Development of Machine Learning Strategies and Integrated Web Platform for the Prediction of Essential Genes

by

Sutanu Nandi
10PP15J26030

A thesis submitted to the
Academy of Scientific & Innovative Research
for the award of the degree of

DOCTOR OF PHILOSOPHY
in
SCIENCE

Under the supervision of
Dr. Ram Rup Sarkar

CSIR-National Chemical Laboratory, Pune

Academy of Scientific and Innovative Research
AcSIR Headquarters, CSIR-HRDC campus
Sector 19, Kamla Nehru Nagar,
Ghaziabad, U.P. – 201 002, India

May, 2021

Certificate

This is to certify that the work incorporated in this Ph.D. thesis entitled, *"Development of Machine Learning Strategies and Integrated Web Platform for the Prediction of Essential Genes"*, submitted by *Sutanu Nandi* to the Academy of Scientific and Innovative Research (AcSIR) in fulfillment of the requirements for the award of the Degree of *Doctor of Philosophy in Science*, embodies original research work carried-out by the student. We, further certify that this work has not been submitted to any other University or Institution in part or full for the award of any degree or diploma. Research material(s) obtained from other source(s) and used in this research work has/have been duly acknowledged in the thesis. Image(s), illustration(s), figure(s), table(s) etc., used in the thesis from other source(s), have also been duly cited and acknowledged.

Sutanu Nandi

Sutanu Nandi

(Research Student)

25th May, 2021

Ram Rup Sarkar

Dr. Ram Rup Sarkar

(Research Supervisor)

25th May, 2021

STATEMENTS OF ACADEMIC INTEGRITY

I, Sutanu Nandi, a Ph.D. student of the Academy of Scientific and Innovative Research (AcSIR) with Registration No. 10PP15J26030 hereby undertake that, the thesis entitled "Development of Machine Learning Strategies and Integrated Web Platform for the Prediction of Essential Genes" has been prepared by me and that the document reports original work carried out by me and is free of any plagiarism in compliance with the UGC Regulations on *"Promotion of Academic Integrity and Prevention of Plagiarism in Higher Educational Institutions (2018)"* and the CSIR Guidelines for "*Ethics in Research and in Governance (2020)"*.

Sutanu Nandi

Signature of the Student

Date: 25th May, 2021

Place: Pune

It is hereby certified that the work done by the student, under my supervision, is plagiarism-free in accordance with the UGC Regulations on *"Promotion of Academic Integrity and Prevention of Plagiarism in Higher Educational Institutions (2018)"* and the CSIR Guidelines for "*Ethics in Research and in Governance (2020)"*.

Ram Rup Sarkar

Signature of the Supervisor

Name: Dr. Ram Rup Sarkar

Date: 25th May, 2021

Place: Pune

Dedicated to my family

Acknowledgments

During the entire period of my doctoral research, I have been supported by many people. Herein I take this opportunity to express my heartfelt gratitude to all of them.

First of all, I would like to express my heartfelt gratitude to my supervisor, Dr. Ram Rup Sarkar, for his constant support and guidance throughout this journey. His enlightened knowledge, wise advice about the research domain, and generous help has made me become not only better researcher but also better person. I started out as a student with computer science background, Sir moulded and guided me to take up the challenge of interdisciplinary research in theoretical and computational biology, sharing his vast experience and knowledge along the doctoral journey and instilling his trade mark "Out of the Box" thinking. I feel immensely privileged to be a part of his research group at CSIR-National Chemical Laboratory. It is the place where I have learned the art of blending hard work and discipline to shape my professional and personal life. What I have gained and learned from him can never be repaid in any possible form. I believe a better way of thanking him would be through my future contributions to the scientific community.

I am grateful to my Doctoral Advisory Committee (DAC) members Dr. Ashok P. Giri, Dr. Leelavati Narlikar, and Dr. Durba Sengupta for evaluating my progress and providing useful suggestions that have not only helped me in my doctoral research but also helped me gain new ideas and perspectives for my future work.

I am thankful to Prof. Dr. Ashish K. Lele (Director, CSIR-NCL), Prof. Dr. Ashwini K. Nangia (Former Director, CSIR-NCL), Prof. Dr. Sourav Pal (Former Director, CSIR-NCL), and the HOD of the CEPD Division, Dr. Sunil Joshi for giving me the opportunity and providing me with advanced research infrastructure and facilities for carrying out my research. I also thank our Finance and Accounts section, Student Academic Office (SAO), CEPD office, DIRC, Library and other departments of CSIR-NCL for providing me the administrative support and necessary infrastructures. I would especially like to express my gratitude to Mr. Kishor Deshpande from DIRC, and his team for helping me with server related technical issues.

My sincere thanks to DST-INSPIRE, Govt. of India for providing me the JRF and SRF fellowships during the entire period of my Ph.D.

My list of acknowledgments would be incomplete without mentioning the name of Professor Manoranjan Maiti (Former Professor at Vidyasagar University, West Bengal), who had effectively incepted in me a deep sense of interest in the field of mathematical modeling. I have got my first exposure and experience in research under his guidance. I would also like to thank Dr. Debashree Ghosh (IACS, Kolkata), Dr. Neelanjana Sengupta (IISER, Kolkata), Dr. Madhura Kulkarni (IISER, Pune), Dr. Chetan Gadgil (NCL), Dr. Anu Raghunathan (NCL), Professor Surjyo Jyoti Biswas (SKBU, West Bengal) for various scientific discussions over the years.

It's my immense pleasure to thank my lab members, especially Dr. Abhishek Subramanian, Dr. Saikat Chowdhury, Ms. Rupa Bhowmick, Ms. Piyali Ganguli, Dr. Noopur Sinha, and Dr. Swarnendu Banerjee, for all their help, scientific discussions, and guidance in the lab. I specifically thank Dr. Abhishek Subramanian for the great scientific discussions, for making me realize my drawbacks and the conceptual understanding of biology. I would also like to thank all the new members of our lab - Anirudh, Kshitij, Priyanka, Gauri, Kiran, Dr. Chandrakala, Bhagyashree for being such wonderful labmates. I also had the opportunity to work with some exceptional Project Assistants of our lab, including Jarjish, Prasun, Jyoti, Rochi, Sanjana, Mudita, Souradeep, Pradeep, Rohit, Anil, Pramod, Arpit, Varsha, Apoorv, Sandipak and many others.

I would also like to especially thank my lab mates Piyali, Rupa, Priyanka, Kshitij and Anirudh for proof reading of my research articles and thesis. I also acknowledge their continuous supports and encouragement on various occasions of my PhD.

A special thanks to my co-authors, Dr. Abhishek Subramanian and Ms. Piyali Ganguli, for their effortless help to understand computational biology.

No words are sufficient to acknowledge my prized friends in and out of NCL who have helped me at various stages of my life and my research work. First of all, I don't know how to quantify the amount of love and support I have got from my hostel mates Dr. Suvendu Karak, Dr. Anup Bhunia, Dr. Mrinmoy Kumar Chini, Dr. Pranab Deb, Dr. Samik Bose, Dr. Himadri Pathak, and Dr. Arijit Mallick to complete this PhD journey. I would love to thank Dr. Krisahnu Show, Dr. Sudip Sashmal, Dr. Atreyee Banerjee, Dr. Soumyajyoti Chatterjee, Dr. Pronay Das, Mr. Himadri Sashmal, Mr. Anirban Sen, Ms. Sanjukta Pahar, Ms. Anushua Biswas, Dr. Milan Bisai, Mr. Anagh Mukherjee, Dr. Sayantan Acharya, Mr. Ujjwal Kumar Nandi, Mr. Tamal Das, Mr.

Subhrashis Banerjee, Mr. Debranjan Mandal, Mr. Tapas Halder, Mr. Narugopal Manna, and Mr. Ramakrishna Gholap for being a valuable part of my NCL family.

I would also like to express my heartfelt gratitude to Dr. Ram Rup Sarkar, Mrs. Mousami Sarkar, Ms. Rupa Bhowmick, Ms. Piyali Ganguli, Dr. Abhishek Subramanian, Dr. Noopur Sinha, Dr. Abhik Banerjee, Dr. Atreyee Banerjee, Dr. Monoj Nandi, and many others for helping me to wade out through one of my difficult phases, i.e., when I was afflicted by Dengue fever. I will never forget all the care and support provided by them.

Words are inadequate to express my feelings and gratitude to my family for their unconditional love, care, and support throughout my life. I would not have achieved anything without my parents' support, who gave me the freedom to explore my world and explore who I am. With immense gratitude and reverence, I acknowledge my mother, Mrs. Gita Nandi, and my father, Mr. Ramgopal Nandi, to shape my life and make me who I am today. I am also immensely thankful to my wife Mrs. Rumpa Mandal who has always supported me emotionally as well as provided me with many practical solutions throughout my journey. I would also like to give special thanks to my brother Krisanu Nandi, sister-in-law Mrs. Mampi Pal Nandi, niece Debadrita Nandi, sister Mrs. Neha Chanda, brother-in-law Mr. Sumit Chugani, niece Katyayani Chugani, and nephew Aditya Chugani for their continuous support during my hard times. I would also like to thank Mr. Manish Chanda for his constant moral support. I would also like to express my sincere thanks to my father-in-law Mr. Sanatan Mandal, mother-in-law Mrs. Uttara Mandal, sister-in-law Ms. Riya Mandal, Mrs. Ambika Pal, brother-in-law Mr. Rajib Mandal and Mr. Arun Kumar Kundu for their constant inspiration throughout my research journey.

I am obligated to the eminent scientific community, whose achievements are a constant source of inspiration for me.

Finally, with immense respect and gratitude, I pay my obeisance to the almighty for all that has been offered to me. Also, I am extremely grateful to everyone who directly or indirectly helped me during this journey.

Sutanu Nandi

Contents

List of Figures

List of Tables

List of Tables (Annexure)

List of Abbreviations

AI	Artificial Intelligence
ANN	Artificial Neural Network
ATP	Adenosine triphosphate
auROC	Area under the receiver operating characteristic curve
BLAST	Basic Local Alignment Search Tool
CAI	Codon Adaptation Index
CEG	Cluster of essential genes
CFS	Correlation-based Feature Selection
CRISPR/Cas9	Clustered regularly interspaced short palindromic repeats/ CRISPR-associated protein 9
DEG	Database of Essential Genes
DR	Dimension Reduction
DNA	Deoxyribonucleic acid
EGGS	Essential Genes on Genome-Scale
ENC	Effective Number of Codons
FBA	Flux Balance Analysis
FCA	Flux Coupling Analysis
FPR	False Positive Rate
FR	Fruchterman Reingold
GO	Gene Ontology
GPR	Gene-Protein-Reaction association
GSRMN	Genome Scale Reconstructed Metabolic Network
GT	Ground Truth
IG	Information gain
KK	Kamada-Kawai
LapSVM	Laplacian Support Vector Machine
LASSO	Least absolute shrinkage and selection operator
MCC	Matthews correlation coefficient
MDS	Metric Dimensional Scaling
ML	Machine Learning
OGEE	Online GEne Essentiality database
PCA	Principal Component Analysis
PCR	Polymerase Chain Reaction
PIN	Protein Interaction Network
PR	Phyletic Retention
RBF	Radial Basis Function
RN	Reaction Network
RNA	Ribonucleic acid
RNAi	RNA interference
SSMSS	Semi-Supervised Model Selection Score
SVM	Support Vector Machine
SVM-RFE	Support Vector Machine - Recursive Feature Elimination
TPR	True Positive Rate

Summary

Gene essentiality information of disease-causing organisms throws light on the minimally essential genes that are absolutely essential for an organism's survival under any environmental condition. Though, the experimental techniques used to conduct genome-wide screens for gene essentiality are costly, labor-intensive, and time-consuming. Various computational techniques are available to annotate gene essentiality. However, the major drawback of existing machine learning-based methods for predicting gene essentiality is that they require a large amount of labeled data from experiments and perform poorly when the labeled dataset is imbalanced or insufficient. Experimentalists frequently encounter these issues when studying new or less studied organisms with a limited number of experimentally annotated genes. The problem is further intensified when the organism shares a small number of conserved orthologous genes with other species, which may not be indispensable, as the organism's various environmental conditions strongly influence gene essentiality.

Machine learning strategies were developed to predict essential genes by considering the issues mentioned above. Two ML-based pipelines were developed to predict essential genes in cases with imbalanced limited labeled training datasets and validated with experimental data that showed high prediction accuracy.

The first strategy, *i.e.*, ML strategy 1 was developed based on the Supervised ML approach for predicting essential genes when sufficient experimental data (labeled data ≥80%) is available, but the dataset is imbalanced. Support Vector Machine-based learning strategy was used for the prediction of essential genes in *Escherichia coli* K-12 MG1655. Dataset combines novel flux-coupled metabolic subnetwork-based features with an appropriate sample balanced training set that characterizes organism-specific

genotype and phenotype. Optimal parameters of the learning algorithm generate the best machine learning model.

Graph-based semi-supervised ML Strategy 2 was developed for the classification of gene essentiality in organisms where the availability of experimental data is minimal (labeled data ≥1%). ML strategy 2 was validated on nine prokaryotes and three eukaryotes, and then the methodology was used to annotate gene essentiality in less-studied organisms like *Leishmania donovani* and *Leishmania major*. It was observed that 80 reaction-gene pairs were predicted to be essential in *Leishmania donovani*. These reactions involved 44 genes that were mostly associated with ATP binding [GO:0005524], oxidoreductase activity [GO:0016491], and AMP deaminase activity [GO:0003876] GO terms. Similarly, in *Leishmania major*, 335 reaction-gene pairs were predicted as essential that involve 194 genes. Predictions for *Leishmania species* are further validated with the experimentally observed pattern of Reaction-Gene combinations occurring in other organisms. These predicted essential reaction-gene combinations were categorized into five different groups (*i.e.*, CEN, ME, MN, SE, and SN) that helps to identify the individual reactions that are regulated by single or multiple essential genes. A similar pattern was also observed for *Leishmania donovani* and *Leishmania major* that further ascertains the validity of predictions. These results indicate the strength of the model in identifying true essential genes using a minimum of 1% labeled data to select biologically relevant features representing gene essentiality. This pipeline (ML Strategy 2) for essential genes prediction shows universality in applying prokaryotes and eukaryotes with limited labeled data.

Existing essential genes prediction platforms such as Geptop, EGP, etc., can only annotate essential genes for model prokaryotic organisms, and in most cases, no source code is publicly available. Also, the preparation of the training datasets and feature tables that includes the calculation of biological features are essential prerequisites for implementing these pipelines, may be quite challenging and time-consuming for users without any prior experience with advanced programming

languages. This necessitates developing a user-friendly ML platform for annotating the essential genes with minimal effort and time. Hence, an online open-source gene prediction server, PRESGENE was developed, by integrating two previously published strategies, machine learning strategy 1 and machine learning strategy 2. Users can easily submit and analyze their data for essential genes prediction through this platform. PRESGENE will provide experimental biologists a well standardized and validated methodology to predict gene essentiality of less-studied organisms. Thus, essential genes, predicted by PRESGENE, will provide important leads to identify novel therapeutic targets in antibiotic and vaccine development.

Chapter 1

Introduction

With the advent of rapid technological development, the molecular biology research has accelerated tremendously in the post-genomic era. Deoxyribonucleic acid (DNA), a polymer of nucleotides with four different bases, *i.e.*, adenine(A), guanine(G), cytosine(C), and thymine(T) which contain genetic information, remains the core of molecular biology. The total amount of DNA in a living organism is known as the genome, and the gene is the functional part of it. The central dogma in molecular biology illustrates that the genetic information flows from DNA to ribonucleic acid (RNA) and RNA to protein by the transcription and translation process (**Figure 1.1 A**). In translation, the triplet of nucleotide bases (a codon) of a gene encodes amino acids that help build macromolecules protein that performs diverse biological functions (**Figure 1.1 B**) such as cell growth, survival, development, and replication. For the survival of an organism, some genes are essential for the cell. These genes are called essential genes (**Figure 1.1 C**).

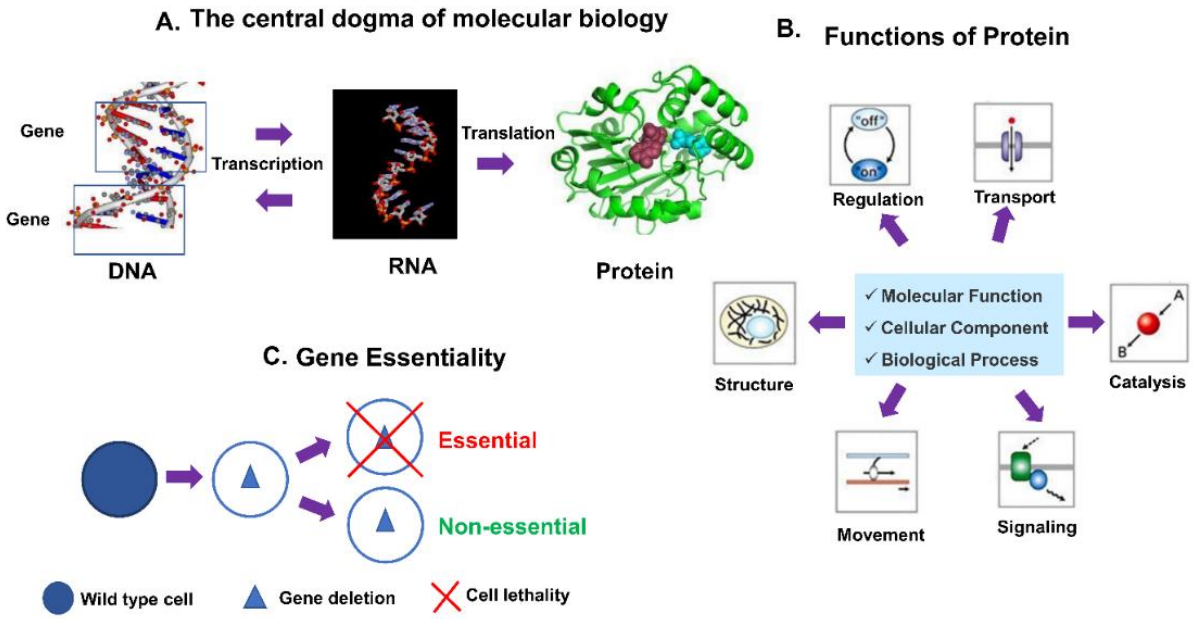


Figure 1.1. The central dogma of biology, linking genotype to phenotype. A. The genetic information flow from DNA to RNA to Protein **B.** Protein plays different biological functions in cell **C.** In a cell, absence of a particular gene, a lethal phenotype is observed, then the gene is termed as an essential gene

1.1. Essential Genes

Essential genes, as the name suggests, are genes necessary for the survival of any cell [1]. The definition of a gene to be essential for survival largely depends upon the environment in which a cell survives and is governed by the underlying function that it performs within the cell [2]. This leads to the classification of essential genes as either – minimally essential, genes absolutely essential irrespective of environmental variations, or conditionally essential, genes essential for cell survival in a particular environment [1,3]. Identification of essential genes is required in many applications like targets of drugs in diseases, systems biology to find out its role (function) within a network, indicators of metabolic microenvironments, and generation of biologically engineered strains of microorganisms [4,5].

1.1.1. Conditionally essential genes

Conditionally essential genes are only essential under specific growth conditions. For example, in a culture medium, the non-availability of the auxotrophic gene, *i.e.*, URA3, catalyzing the sixth enzymatic step in pyrimidine's de novo biosynthesis becomes conditionally essential in S*accharomyces cerevisiae* [6].

1.1.2. Minimally essential genes

Minimally essential genes are absolutely essential to support cellular life and supply the required necessary nutrients in a stress-free environment [6]. For example, *Mycoplasma genitalium* is one of the best organisms to study minimal genes. The 256 genes among 482 protein-coding genes are reported as minimally essential genes that maintain fundamental biological processes such as DNA repair, recombination, replication, and protein translation for cell survival. In our studies, we have considered only minimally essential genes.

1.2. Application of Essential Genes

Identifying essential genes is vital for several reasons, including:

Understanding the pathogen biology: essential genes in a pathogen will help prioritize a set of crucial genes and their functional properties. In *Escherichia coli*, through gene knockout studies, it has been identified that tetracyclines block protein translation by binding to the ribosome. Thereby elucidating the essential function of ribosomes for survival [7].

Drug discovery: the essential genes of disease-causing organisms serve as a list of probable potential drug targets. Undoubtedly, specifying a drug target in the pathogen, gene essentiality is one of the critical criteria. Hu *et al.* demonstrated a lead drug's efficacy on *Aspergillus fumigatus* where 54 genes of *Aspergillus fumigatus* were confirmed orthologs and essential in *Candida albicans* and *Saccharomyces cerevisiae* [8].

Biomarker detection: The reconstruction of the metabolic network of specific human cell lines provides insights into its essential genes. Using essential genes as a parameter, important biomarkers associated with diseases have been identified. In breast and ovarian cancer, homozygous BRCA 1 and BRCA 2 genes loss of function prompt the cancer cell to become dependent on poly (ADP-ribose) polymerase (PARP). This knowledge is exploited to treat ovarian cancer with PARP inhibitor - Olaparib [9].

Evolutionary standpoint: A distinct correlation between gene essentiality and its impact on conservation is suggested in a class or family of organisms. For instance, in *Escherichia coli*, roughly 33% of essential genes are non-essential in *Bacillus subtilis* [10]. Likewise, in *Saccharomyces cerevisiae*, 17% of its essential genes are non-essential for *Schizosaccharomyces pombe* [11].

Synthetic reconstruction of the organism: From the standpoint of synthetic biology, an essential gene set intersects widely with the minimal gene set required

for organism survival; thus, identifying the essential set also leads to a future perspective of the synthetic redevelopment of the organism. Papp *et al.* synthetically reconstructed yeast metabolic network to show that essential genes for growth are necessary for survival [12].

Food microbiology, industrial bioprocessing: Essential genes and their functions in plants, animals, and microorganisms are used to produce food, biofuel, and biocatalyst at a large scale. For example, high-yield strains of *Corynebacterium glutamicum* LYS-12 strain have been generated by globally modifying the pathways and redirecting the flux within the network to synthesize amino acid [13,14].

1.3. Experimental Approaches of Essential Genes Annotation

Innumerable experimental techniques like genetic footprinting, gene knockouts (deletions), double targeted gene replacement, transposon mutagenesis, RNA interference, CRISPR/Cas9, etc., are available for scrutinizing the essentiality of a gene [15–19]. A brief description of these experimental techniques for essential genes prediction is discussed in the following sub-sections.

1.3.1. Genetic footprinting

Genetic footprinting is a technique used for distinguishing between essential and non-essential genes. This method involves following three main steps. First, **transposon mutagenesis,** *i.e.*, generation of the mutation using a transposon that inserts randomly throughout the genome; second, **an outgrowth of the mutagenized cell,** *i.e.*, growth of the initial mutagenized culture over many generations under different conditions that repress transposase expression and plasmid replication; and third, **mutations specific analysis of the cell,** *i.e.*, comparative evaluation of transposon insertions present within specific genes based on analysis of polymerase chain reaction (PCR) [20]. Simultaneously this method determines the genes required for growth under a specific condition [21]. The PCR analysis of the regions

confirms gene essentiality where the transposons are absent in outgrown cells because no cell would have survived if this region was disrupted.

1.3.2. Targeted gene replacement

Targeted gene replacement is a genetic technique that uses homologous recombination to modify a gene. It is a two-step process. Firstly, a region of the gene of interest is replaced by a marker to produce an inactivated gene, which is then retargeted with a second vector to reconstruct the inactivated gene [22]. This method is used to delineate the essentiality of a gene or correct a mutated gene back to wild-type. Construct vector at the targeted site requires a form of DNA double-strand break repair known as homologous recombination [23]. Drawbacks of this method include inaccessible DNA sites to homologous recombination and lack of knowledge on the repair mechanisms [24].

1.3.3. Transposon mutagenesis

Transposon mutagenesis is an experimental technique where genes are transferred to a host's genome by a transposon, thereby creating mutants. It is achieved using plasmids, where transposon is extracted and inserted into the host for creating gene disruptions that eliminate gene function [25]**.** Libraries generated by transposon mutagenesis create well-defined sequence diversity that is two times larger than the length of the target gene. Transposon insertion sites are then verified using direct sequencing. Lethal mutations having loss of essential genes functions are then rescued by transfecting cells containing genomes with the corresponding wild-type genomic fragments [26].

1.3.4. RNA interference (RNAi)

RNAi is a biological process to regulate the expression of protein-coding genes. This mechanism depends on two main steps, the double-strand RNA or messenger RNA is integrated into molecular scissor (RNA-induced silencing complex). This scissor

then targets mRNA by forming a hybrid and then degrading the mRNA with ribonuclease enzyme. The mRNA targeted by the molecular scissor will lose its functionality, thereby acting as a lethal phenotype. Several studies have successfully used RNAi screening for profiling the essential genes landscape [27,28].

1.3.5. CRISPR/Cas9

CRISPR/Cas9 is an experimental technique for annotating gene essentiality in organisms through gene editing. This method determines gene essentiality by knocking out the gene by inducing a site-specific break in the DNA and then rejoining the ends of the broken double-stranded DNA, resulting in a lethal phenotype [29]. This method is less prone to minor errors and has fewer off-target effects than other experimental techniques [30]. This technology is currently being used to identify essential genes in cancer cells [31].

1.4. Limitation of Experimental Approaches for Annotation of Essential Genes

The essentiality of a gene varies from organism to organism depending on the complexities of the cellular structure. To address the differences in the cellular complexities different type of experimental protocols need to be designed [15,17]. However, these techniques work well with model organisms for which a standardized protocol for gene essentiality identification is available and establishing the essentiality for a large set of genes in non-model, less explored organisms is challenging, as the experimental standardization of protocols for determining gene dispensability and sampling for a range of experimental conditions is laborious and time-consuming.

1.5. Database of essential genes

DEG and OGEE are well-established primary databases that collect gene essentiality data from various experimental sources. Apart from these, there are other databases such as EGGS and CEG. Researchers can use the experimentally screened gene

essentiality information to build a robust computational predictive model for annotating gene essentiality. The following sub-sections provide a brief description of these databases.

1.5.1. DEG (Database of Essential Genes)

DEG is an open-source platform for storage information about essential genes from an experiment. This database was released in 2004, and it is still being maintained. The recently updated version of this database, DEG v15.2, contains gene essentiality information collected from different source of experiments and have essential information about archaeal, prokaryotic, and eukaryotic organisms. Due to some experimental limitations, it is impossible to annotate some genes either as essential genes nor as non-essential genes. In this platform, homology searches are also available with the embedded BLAST tool, which can be annotated for single or multiple un-annotated genes [32]. The recently updated information and prediction tool for gene essentiality in DEG make it popular.

1.5.2. OGEE (Online GEne Essentiality database)

OGEE is an open-source database that stores information about both essential and non-essential genes [33]. The text mining process is used to prepare gene essentiality information in this database, and the results are validated with experimental results obtained from large-scale experiments. List of properties, *i.e.*, gene duplication, gene expression profiles, and conservation across species, is organized for an individual gene. This database also includes an online tool for predicting essential genes and an integrated visualization tool to show the proportion of essential genes based on their three properties: singletons, gene duplication, and developmental genes. An improvised version of this database was released in 2006 with more organisms. In addition, this database stores gene essentiality information of nine human cancer cell lines. Researchers can utilize this gene essentiality information for specific cancer types to discover anti-cancer drug development.

1.5.3. Essential Genes on Genome-Scale (EGGS)

EGGS stores only 10 bacterial species genes essentiality information in genome-scale from different sources of experiments [34]. Genes in this database are classified as Essential genes, non-essential genes, and Undefined. EGGS holds far fewer organism's essentiality information compared to DEG and OGEE.

1.5.4. CEG

CEG is a secondary database that accumulates cluster of essential genes derived from the DEG database [35]. CEG_Match tools used the CEG database to annotate genes essentiality based on comparison of the gene name. This database stores genes essentiality information in an orthologous cluster, when genes have similar biological function. However, this database is obsolete.

1.6. Computational Approaches of Essential Genes Prediction

Researchers are developing computational techniques for essential genes prediction based on homology mapping, constraint-based modeling strategies, and machine learning strategies as an effective alternative to complex experimental strategies [36–38].

1.6.1. Homology mapping-based Strategy

Essential Genes prediction using computational methods was first adopted from sequence homology, which depends on comparative analysis of genomes. The basic idea of homology mapping methods is that the genes common in distantly related species are likely to be essential. Researchers have tried to search for a single species sequence data with the available bacterial genomes. Essential genes were identified by comparative genomic analysis in different bacterial species such as *Mycoplasma* [39], *Liberibacter* [39], *Plasmodium falciparum* [40], and *Brucella spp.* [41]. Due to the slower evolutionary rate of essential genes, they are more conserved in bacteria [42]. Essential genes have been predicted using gene duplication and phyletic genes, as

well as other homology-based properties. Gene duplication in same organism is also called paralogs. Paralogous genes have a similar type of function, and these genes are less likely to be essential because another duplicate performs the similar function. So duplicate genes are not likely to be lethal for an organism [42]. Sequence homology mapping can be applicable for annotating gene essentiality based on genomic sequences. However, the limitation of this method is that the conserved orthologous genes between different species form only a small fraction of the entire genome [43]. Also, it has been observed that highly conserved genes across different species are not always essential, as gene essentiality also depends on different environmental conditions where the organism resides.

1.6.2. Constraint-based strategy

Constraint-based modeling strategies, such as Flux Balance Analysis (FBA), employ genome-scale reconstructed metabolic networks to predict the metabolic fluxes at steady-state. This methodology is widely used for predicting essential genes by performing *in-silico* knockout of a gene and estimating its corresponding lethality [44–46]. A limitation of the FBA method is that only a limited number of environmental conditions can be considered for a specific biomass equation (or objective function) for gene essentiality.

1.6.3. Machine Learning-Based Strategies

In recent years, Machine Learning (ML), a subset of artificial intelligence (AI), has been widely used in various fields for data processing and analysis. ML models use data driven approach to automatically learn inherent patterns in the data and make decision for new set of data [47]. Here, we discuss different machine learning algorithms and the applications of recently developed algorithms in the field of essential genes prediction.

Basic terminologies generally used for machine learning are Dataset for Training and Testing, Instance, Features or attribute, Class label, and Cross-validation.

A brief description of each terminology is as follows:

Dataset: It is a matrix in which each row denotes an instance, and each column denotes a feature or attribute. The training dataset is a subset of the entire dataset that is only used to train the model during training time. A testing dataset is a subset of the entire dataset that is not used for training but aids in calculating the error and performance of the classifiers.

Instance: The data points or cases in a study are referred to as instances.

Attribute/feature/variable: It is represented by numeric or categorical values to describe an instance. Feature vector in feature space represents a data point or instance.

Class label is the description of an instance or data points. For essential genes, prediction is a binary classification problem. Here only two class labels, *i.e.*, essential or non-essential, are used.

Cross-validation (CV) is a statistical procedure to generalize classifiers by estimating the errors. Various cross-validation approaches like k-fold cross-validation, bootstrapping, leave one out are used. In k-fold cross-validation, the dataset is split into k subsets by randomly selecting data points from the entire dataset, with (k-1) subsets used for training and one subset used for testing. In "leave one out cross-validation", one sample is chosen at random from the entire dataset to be used for model testing, while the rest of the dataset is used for training. In bootstrap cross-validation, the training dataset is prepared from the whole dataset by sample with replacement technique, and the rest of the dataset is used for testing purposes.

In literature, based on the availability of class labels, ML algorithms are divided into supervised, unsupervised, and semi-supervised. If response variables (Class labels) are known, then it is called a supervised machine learning algorithm, and class labels for unsupervised machine learning algorithms are unknown. In semi-supervised cases, a very small amount of class labels is known. Supervised algorithms are

divided into two categories viz. classification, regression. In classification, the values of the class variable are discrete, whereas, in the regression problem, the values of the class variable are continuous. The workflow of machine learning strategies generally used for essential genes prediction is provided in **Figure 1.2.** The description of some machine learning classifiers used in essential genes prediction is given below.

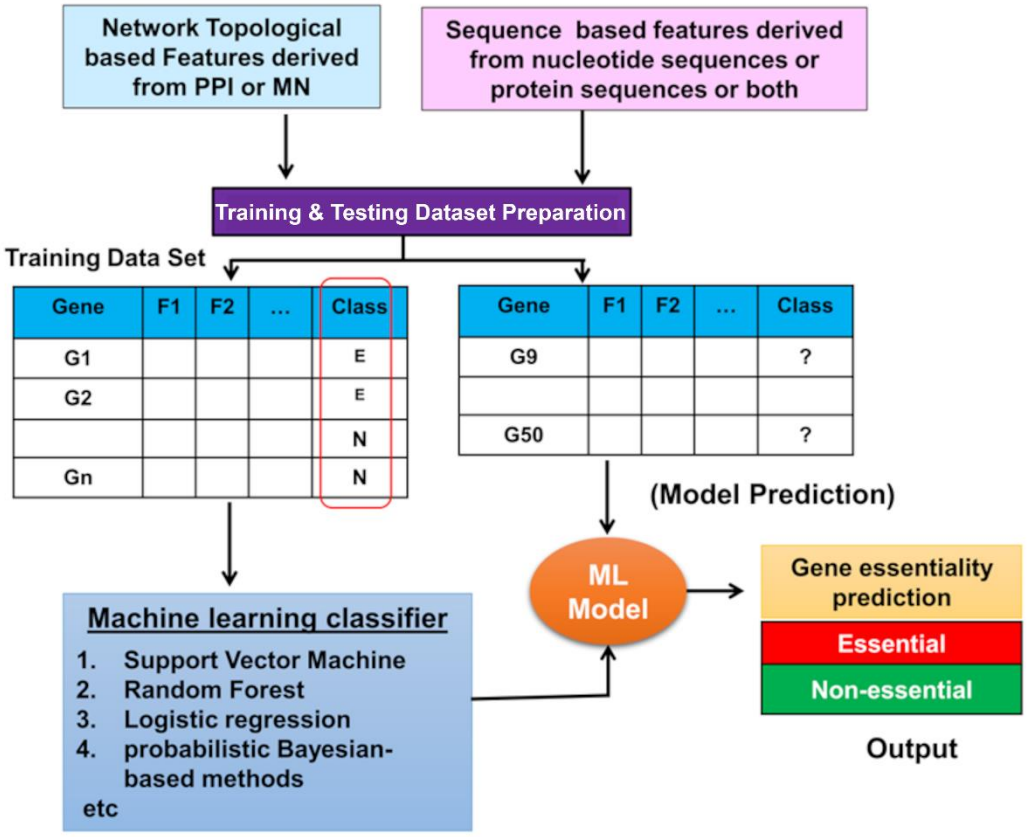


Figure 1.2. Machine learning workflow is generally used for essential genes prediction. The integrated pipeline for prediction of essential genes based on labeled training dataset with sequence, informatics, and topological network features.

1.6.3.1. Calculation of biologically relevant Features used for ML training and prediction

The machine learning-based classifiers predict gene essentiality of unannotated genes based on the pattern of the features of previously annotated genes that have been verified experimentally and labeled as essential and non-essential. In order to achieve this, researchers have curated different combinations of features. Most of the machine learning approaches use calculated features either from coding sequences

[48–50] or network (*e.g.*, protein interaction network, metabolic network) topological features [36,51] or both. Protein interaction networks (PIN) have been used to calculate topological network features (*i.e.*, Degree Centrality, Eigen vector Centrality, Eccentricity, Hub score, Authority Scores, Page Rank, and Betweenness Centrality) to classify gene essentiality [51,52]. On the other hand, only a few studies have used flux-based features derived from metabolic networks to classify genes [53,54] that have been calculated in a single environmental condition that does not represent a universal set of features. The commonly used topological network features, such as centrality measures, highlight the biological significance of an enzyme or protein in a network [55]. Generally, a central and highly connected protein in biological networks is often essential as it represents an important hub within the network [56]. If this hub node is blocked, then the whole pathway might be disrupted. Features, such as amino acid frequency and protein length computed from protein sequence, and codon adaptation index (CAI), Effective Number of Codons (ENC), Phyletic Retention (PR), GC content computed from nucleotide sequence are some of the known features of gene essentiality across bacteria [52,54,57]. A detailed description of these features is discussed in the **Chapter 2 , Section 2.1.**

1.6.3.2. Feature selection methods

The feature selection technique selects a subset with a smaller number of relevant features from a full feature set without changing its original value based on maximizing the feature relevance and minimizing the redundancy. A feature is typically classified as highly relevant, marginally relevant, but not redundant, irrelevant, or redundant. A strongly appropriate feature is often required; it cannot be omitted without impacting the original distribution. An only marginally relevant feature may not always be required for an optimal subset; this may depend on the context. It is unnecessary to include irrelevant features. The redundant features refer to marginally significant features but can be replaced entirely by another set of

features without affecting the target distribution. As a result, feature selection helps to decrease model parameters and enhances model generalization abilities by reducing the execution time complexity at model training time.

The feature selection algorithms are divided into three groups viz., filter, wrapper, and embedded [58].

Filter: Filters are a subset of feature selection algorithms that extract features from data patterns without incorporating machine learning classifiers. For example, *t*-test feature selection, Correlation-based feature selection (CFS), Bayesian networks, and Information gain (IG) are filter-based feature selection algorithms.

Wrapper: Wrapper type of feature selection algorithms considers machine learning classifier to determine the relevant feature set. Sequential search is a wrapper-based feature selection method.

Embedded: Embedded types of algorithms integrate the feature selection and classifier simultaneously. Some examples of embedded feature selection algorithms are recursive feature elimination (RFE), which is integrated with the SVM classifier, Random forests.

A brief description of these feature selection algorithms is provided in Table 1.1.

Table 1.1: Different Feature selection algorithms [58]

Method	Type	Supervised	Linear	Description
t-test feature selection	Filter	—	Yes	It identifies features with the most significant mean difference between groups and the minor variability within each group.
Correlation-based feature selection (CFS)	Filter	—	Yes	It identifies features that are highly correlated with the class but are uncorrelated with one another.
Bayesian networks	Filter	Yes	No	They determine the causal relationships between features and delete those without a causal relationship with the class.

Information gain (IG)	Filter	No	Yes	It assesses how common a feature is in a class compared to all other classes.
Sequential search	Wrapper	—	—	The heuristic-based search algorithm adds one new feature to the set each time to find the highest model training performance (for example, classification accuracy).
SVM method of recursive feature elimination (RFE)	Embedded	Yes	Yes	It constructs the SVM classifier and eliminates features based on their "weight" during the classifier construction process.
Random forests	Embedded	Yes	Yes	They construct several decision trees from the various features and then select the relevant feature subset with the best model training performance.
Least absolute shrinkage and selection operator (LASSO)	Embedded	Yes	Yes	It builds a linear model in which many of the feature coefficients are set to zero, and the nonzero ones are used as the selected features.

1.6.3.3. Dimension Reduction Methods

The feature selection step reduces the dimension of dataset by selecting the relevant features with its original values, whereas the dimension reduction step reduces the higher dimensional features into lower dimension by transforming the original values. There are various dimension reduction techniques, such as Principal Component Analysis (PCA) [59], Metric Dimensional Scaling (MDS) [60], t-distributed stochastic neighbor embedding (t-SNE) [61] etc., have been used in other biological problem such as microarray gene expression data analysis [62], but no previous studies have used dimension reduction technique as a component of machine learning strategy for essential genes prediction purpose. However, dimension reduction techniques help visualize high-dimensional data and solve unsupervised clustering problems. Now we will discuss some widely used dimension reduction techniques.

PCA [59]: It projects the samples with high dimensions based on the eigenvectors (principal components) related to the covariance matrix's largest eigenvalues to conserve most of the variance in the input dataset. Criteria to apply PCA on dataset should follow the Gaussian distribution. For nonlinear data, Kernel PCA is useful with nonlinear kernel mapping.

MDS [60]: It is a distance-conserving dimension reduction technique. It projects the high-dimensional data points into low dimensions by minimizing the difference of the distance between data points of original and projected coordinate.

1.6.3.4. Overview of Machine Learning Classifiers

ML algorithms can be broadly grouped under supervised, semi-supervised, and unsupervised strategies [63,64]. The supervised strategies, such as Decision Tree, Naïve Bayes, Support Vector Machine (SVM), require sufficient amounts of labeled data for model training. In contrast, the unsupervised method relies on clustering algorithms (*e.g.*, K-Means Clustering), where no labeled data is required. The semi-supervised ML algorithms that comprise Generative Models, Self-Training, Transductive SVM, and Laplacian SVM combine the potential of both supervised and unsupervised ML strategies and can train the model with a very limited amount of labeled data.

a. Supervised Machine Learning Classifiers

Widely used, supervised machine learning classifiers for essential genes prediction are SVM, Naive Bayes, Logistic regression, Artificial Neural Network (ANN), Decision tree, Random Forest, and CN2. A brief description of these classifiers is discussed below.

i. Support Vector Machine (SVM)

SVM is a supervised machine learning classifier proposed by Vapnik, which is based on the structural risk minimization technique [65]. SVM builds an optimal

hyperplane by maximizing the distance between two classes. The algorithm can be used for classification and regression problems. SVM transforms the feature space of input data points into high dimensional non-linear feature space by using various kernel functions to distinguish complex real-life datasets. The performance of SVM classifiers is improved significantly by proper use of kernel function and their parameters. Pictorial representation of SVM is given in **Figure 1.3**.

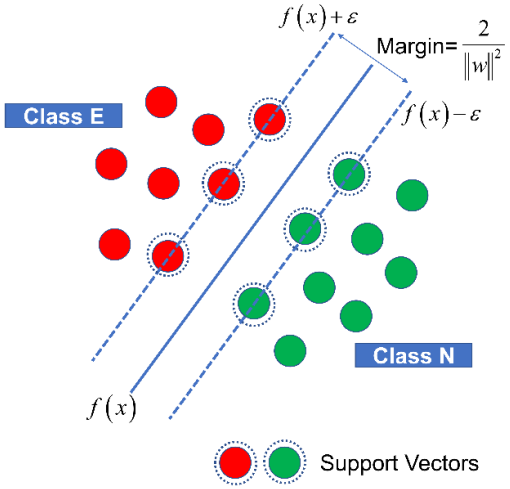


Figure 1.3. Schematic diagram of the support vector machine classifier

The objective function (**Eq. 1.1**) of SVM is defined as follows:

$$\min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\left(\xi_i^+ + \xi_i^-\right) \qquad \text{Eq. 1.1}$$

$$s.t.\begin{cases} y_i - f\left(x_i\right) \le \varepsilon + \xi_i^+ \\ y_i - f\left(x_i\right) \le -\varepsilon - \xi_i^- \\ \xi_i^+, \xi_i^- \ge 0 \end{cases}$$

Where ξ_i^+ and ξ_i^- are the slack variables used to define an error. C is a coefficient of adjusting between the margin and error on the hyper-plane. f is the prediction, y is an actual class label, and ε is a free threshold parameter.

This algorithm has several advantages: high prediction accuracy, less prone to overfitting, good generalization ability with small training data, and robustness to noise and outliers. The disadvantages of this algorithm are as follows: require more

memory for optimization, high time complexity, is not suitable for large training datasets, and requires a proper kernel for a non-linearly separable dataset.

ii. Naive Bayes

The Naive Bayes classifier [66] is a probabilistic method. This classifier assumes that each feature should be conditionally independent of the other. The Naive Bayes method follows Bayes rule to detect the most probable class for classification. Pictorial representation of Naïve Bayes classifier given in **Figure 1.4**.

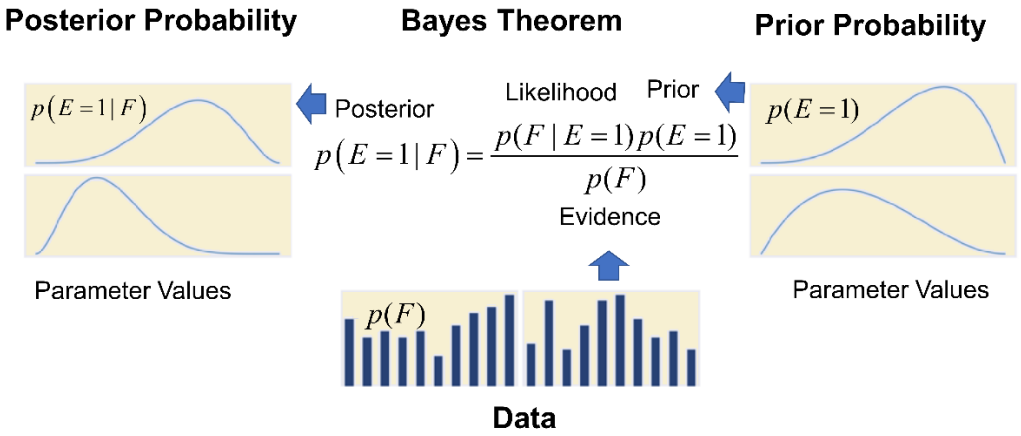


Figure 1.4. Schematic diagram of the Naive Bayes classifier

The posterior probability (**Eq. 1.2**) for the essentiality of a gene by Naïve Bayes classifier can be defined as:

$$p(E=1|F)=\frac{p(E=1)\prod_{i=1}^{n}p(f_i|E=1)}{p(F)} \qquad \textbf{Eq. 1.2}$$

Were, $F=(f_1, f_2, \ldots\ldots f_n)$ is all the features, $p(E=1)$ is the prior probability that can be calculated from a training dataset.

Classifier output can be computed using the following decision rule (**Eq. 1.3**):

$$classify(f_1, f_2, \ldots\ldots f_n)=\arg\max_{E=1,0} p(E)\prod_{i=1}^{n}p(f_i|E) \qquad \textbf{Eq. 1.3}$$

The advantages of this algorithm are simple and easy to implement; converges quickly if the assumption of conditionally independent properties holds.

The disadvantages are, in the complex biological problems, the conditional independence assumption does not always hold; classifier performance reduces drastically with the increased sample size of the training dataset.

iii. Logistic regression classifier

Logistic regression classifier uses the posterior probability using a logistic function. This algorithm is applied for classification problems, not regression. Pictorial representation of Logistic regression classifier is provided in **Figure 1.5**. The logistic function (**Eq. 1.4**) with the feature set $F = \{f_1, f_2, \ldots\ldots\ldots\ldots, f_n\}$ is

$$p(E=1|F) = \frac{1}{1+e^{-(\beta_0+\beta_1 f_1+\ldots\ldots\beta_n f_n)}} \quad \textbf{Eq. 1.4}$$

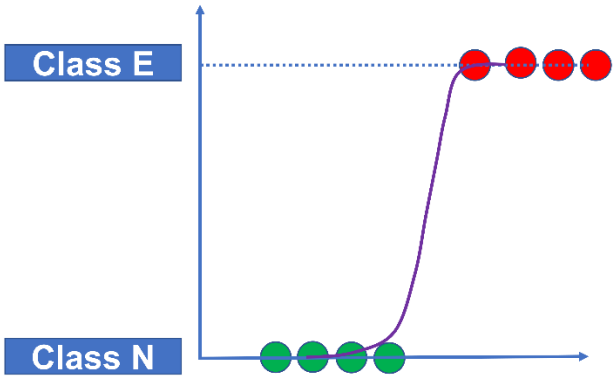


Figure 1.5. Schematic diagram of the Logistic regression classifier

The advantages of this algorithm are simple and easy to implement for a small dataset. The disadvantage is that the classifier performance reduces drastically with the increased sample size and features in the training dataset.

iv. Artificial Neural Network (ANN)

The ANN [67], is built on a multi-layer perceptron and has been generally used to solve challenging classification problems. This algorithm follows the similar logic of biological neurons. It has three layers: input, hidden, and output. It computes the

weights for each layer and then uses backpropagation to reduce the error cost, which is defined by the discrepancy between predictions and actual observations. The schematic representation of ANN is given in **Figure 1.6**.

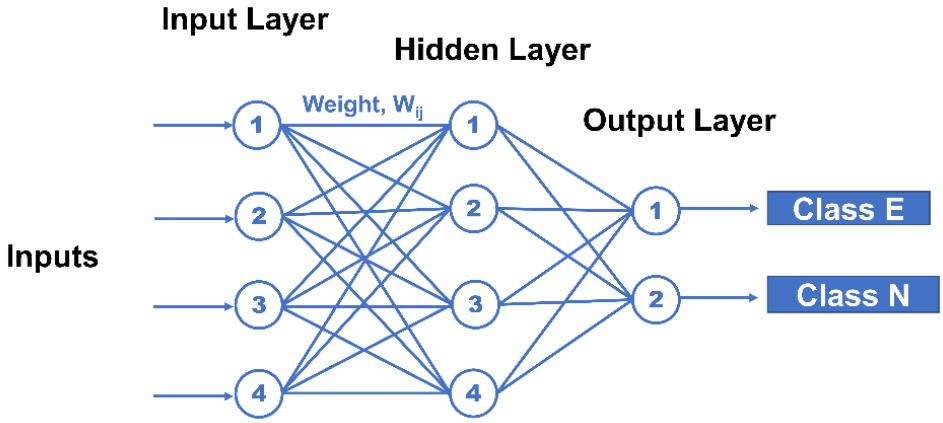


Figure 1.6. Schematic diagram of the artificial neural network classifier

ANN's objective function (**Eq. 1.5**) is as follows:

$$\arg\min_{w} E(w) = \frac{1}{2}\sum_{i=1}^{m}\left(N(w, x_i) - y_i\right)^2 \qquad \textbf{Eq. 1.5}$$

Where w denotes the weight between nodes, x denotes the input vector, and y denotes the target vector. An ANN model's performance is affected by the number of hidden layers and nodes. Although, there are no explicit guidelines for deciding them. Generally, heuristics search (trial and error) determines the number of layers and nodes.

The advantages are flexible and adaptive; performs better on complex non–linear relationships between dependent and independent features.

The disadvantages are, requires a considerable amount of training dataset for good training performance, long training time is required, the algorithm may be stuck into local minima, and prone to overfitting.

v. k –Nearest Neighbors

The k –Nearest Neighbors (k –NN) classifier is a lazy learning method [68]. This method classifies for new data points based on k nearest observations in the training dataset. The pictorial representation of K-NN algorithm is given in **Figure 1.7**.

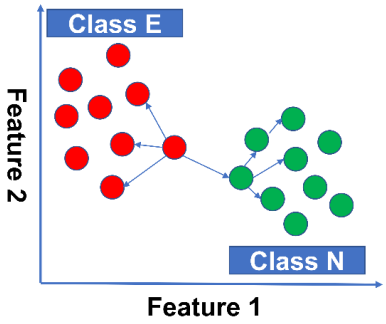


Figure 1.7. Schematic diagram of the k –Nearest Neighbors classifier

The advantages are simple to implement, analytically traceable, highly reserve to local information. The disadvantages are, requires large storage space, requires huge computation time for large dataset, the result varies on different values of k, sensitivity to noise and outliers, larger k values increase the time complexity and high sensitivity to high dimensional data

vi. Decision Trees

It is a supervised machine learning classifier based on the divide and conquers approach of learning from the instance where it follows a tree-based structure (with three types of nodes: root, internal, and leaf) and constructs decision rules [69]. The schematic representation of Decision Tree is given in **Figure 1.8**.

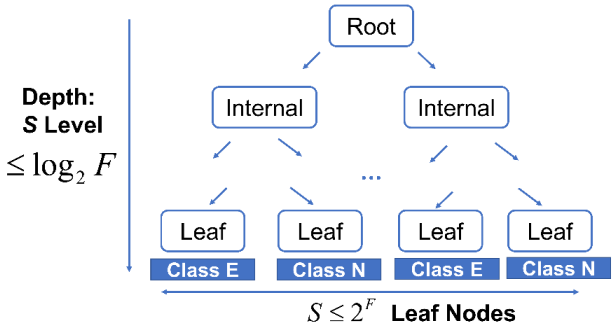


Figure 1.8. Schematic diagram of the Decision Tree classifier

The impurity of a node t in a decision tree can be measure using the following measures (**Eq. 1.6, Eq. 1.7, Eq. 1.8**)

$$Entropy(t) = -\sum_{i\in(0,1)} p(i|t)\log_2 p(i|t) \quad \textbf{Eq. 1.6}$$

$$Gini(t) = 1 - \sum_{i\in(0,1)} \left[p(i|t)\right]^2 \quad \textbf{Eq. 1.7}$$

$$Classification \quad error = 1 - \max_i \left[p(i|t)\right] \quad \textbf{Eq. 1.8}$$

The goodness of split in a tree of a node is measure using (**Eq. 1.9**) the gain ratio

$$\text{Gain ratio} = \frac{I(parent) - \sum_{i=1}^{n} \frac{N(child\quad i)}{N} I(child\quad i)}{-\sum_{i=1}^{n} p(child\quad i)\log_2 p(child\quad i)} \quad \textbf{Eq. 1.9}$$

The feature with the highest gain ratio is chosen to create child nodes in a decision tree. The advantages are simple, easy to understand and interpret, runs fast, can manage irrelevant features, can handle non–linear relationships. The disadvantages are, prone to overfitting without proper tree pruning. Also, finding the optimal decision tree is very difficult. It can be stuck in local minima

vii. Random Forest

This algorithm constructs based on multiple decision trees. This approach is particularly useful when an optimal classifier is infeasible [70]. One of the widely used ensemble classifiers is the Random Forest [71], which operates by constructing multiple decision tree models at the training time. It performs well in multi-class classification. **Figure 1.9** represent the pictorial representation of Random Forest.

The advantages are, the predictive performance is high, easy to interpret class predictions, less chance of overfitting. The random forests algorithm can infrequently suffer from overfitting for noisy datasets.

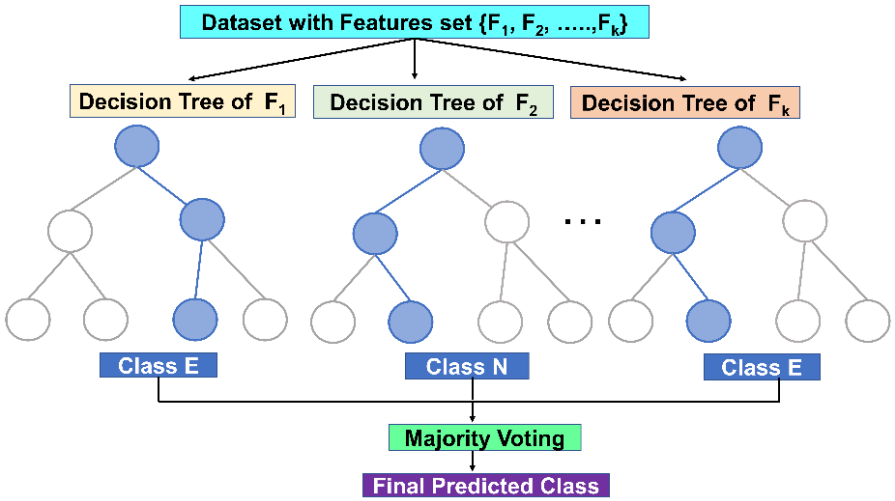


Figure 1.9. Schematic diagram of the Random Forest classifier

viii. CN2 Classifier

The CN2 algorithm proposed by Clark and Niblett [72] is a supervised machine learning algorithm that classifies based on rules. To generate rules from data, iteratively, it uses a decision tree type algorithm in the form of 'if [condition] then predicting [class]' where [condition] is the CN2 rule. The pictorial representation of CN2 classifier is provided in **Figure 1.10**.

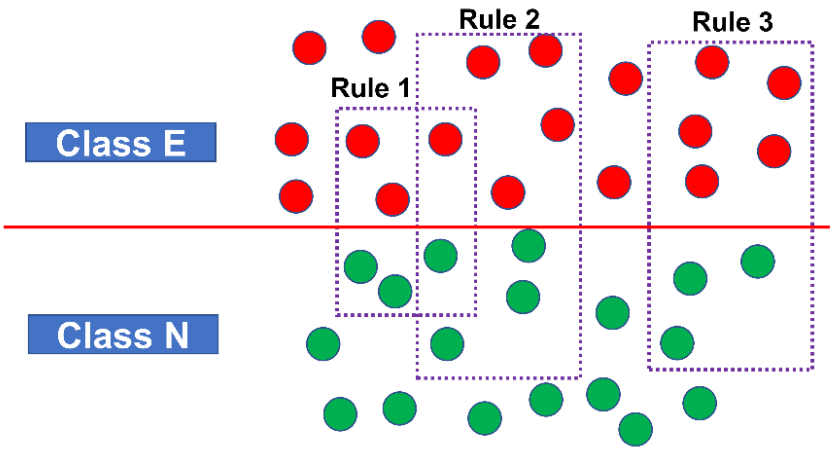


Figure 1.10. Schematic diagram of the CN2 rule-based classifier

The advantages are simple, easy to understand and interpret, runs fast, can manage irrelevant features, can handle non–linear relationships but it is challenging to define rules for training datasets with a large number of features.

b. Semi-supervised classifier

i. Laplacian SVM

The machine learning technique can be a difficult task when a minimal amount of labeled information is available. In this setting, semi-supervised learning is an appropriate approach that builds a trained model from labeled and unlabeled samples [73]. Most of these semi-supervised algorithms follow two common assumptions, *i.e.*, cluster assumption and manifold assumption. Cluster assumption states that data points in the same cluster have a chance of having the same class label. Manifold assumption means that close data points along the manifold area follow similar data structures or similar class labels. However, cluster assumption follows the global feature, and manifold assumption follows the local features in the model.

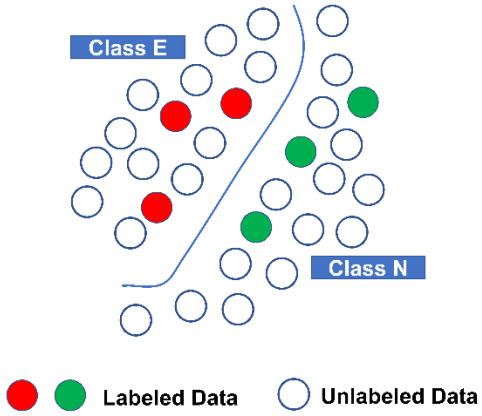


Figure 1.11. Graphical presentation of the Laplacian support vector machine classifier

Laplacian support vector machine (LapSVM) is a graph-based semi-supervised learning method, which is based on a manifold regularization framework [74]. The graph is constructed from labeled and unlabeled data as the node. The similarity between data points in a graph can be assigned by edge weight, which is calculated from the K-NN algorithm. In this way, the information of labeled data points can be passed to another node, and then, the unlabeled nodes can be labeled. Schematic representation of LapSVM is given in **Figure 1.11**.

LapSVM solves the following optimization problem (**Eq. 1.10**).

$$\arg\min_{f\in \mathrm{H}_k} \frac{1}{n_l}\sum_{i=1}^{n_l}\left|1-y_i f\left(x_i\right)\right|_+ + \lambda_a \left\|f\right\|_K^2 + \frac{\lambda_b}{\left(n_l+n_u\right)^2}\times f^T L f \qquad \textbf{Eq. 1.10}$$

Where, $\left\|f\right\|_K^2$ is a regularization function for smoothness,

λ_a, λ_b are hyperparameters,

n_l is the number of labeled data points,

n_u is the number of unlabeled data points,

loss function=$\left|1-y_i f\left(x_i\right)\right|_+ = \max\left(0,1-yf\left(x\right)\right)$,

$\sum_{i,j=1}^{n} W_{ij}\left(f\left(x_i\right)-f\left(x_j\right)\right)^2 = f^T L f$,

W_{ij}, is the edge weights in the graph,

Laplacian operator, $L = D - W$

The advantages of this algorithm are high prediction accuracy, robustness to noise and outliers, less prone to overfitting, and good generalization ability with small labeled data. The disadvantages are required more memory to construct a graph; the time complexity is high, not suitable for large training datasets.

1.6.3.5. Model performance evaluation

Machine learning algorithm requires metrics for selecting the best model. First, the confusion matrix has been prepared with actual and predicted labels. So, the classifier has four outcomes:

True positive (TP): Number of essential genes correctly predicted by the classifier as essential.

False positive (FP): Number of non-essential genes wrongly predicted as essential.

True negative (TN): Number of non-essential genes correctly predicted by the classifier as non-essential.

False negative (FN): Essential genes wrongly predicted as non-essential.

For the above model outcomes, a set of model performance metrics (**Eq. 1.11 - Eq. 1.17**) have been used. The metrics are defined as,

True Positive Rate (TPR) or Sensitivity: The proportion of positive (essential genes) instances predicted correctly by the model.

$$True\ Positive\ Rate\ (TPR)\ or\ Sensitivity = \frac{TP}{TP\ +\ FN}, \quad TPR \in [0,1] \qquad \textbf{Eq. 1.11}$$

False Positive Rate (FPR): Defined as the proportion of negative (non-essential genes) instances predicted as positive by the model.

$$False\ Positive\ Rate\ (FPR)\ = \frac{FP}{FP\ +\ TN}, \quad FPR \in [0,1] \qquad \textbf{Eq. 1.12}$$

Precision: It determines the measure of correctness *i.e.,* how many essential genes are predicted as an essential class belongs to the positive class.

$$\Pr ecision = \frac{TP}{TP\ +\ FP}, \quad \Pr ecision \in [0,1] \qquad \textbf{Eq. 1.13}$$

Recall: It measures the proportion of essential instances correctly predicted by the model.

$$\text{Re}call = \frac{TP}{TP\ +\ FN}, \quad \text{Re}call \in [0,1] \qquad \textbf{Eq. 1.14}$$

F-measure: This performance metric is defined as the harmonic mean between precision and recall. A high value of F-measure suggests that the predictive performance better on essential class.

$$F\text{-}measure = \frac{2\ .\ (\mathrm{Pr}ecision\ .\ \mathrm{Re}call)}{(\mathrm{Pr}ecision\ +\ \mathrm{Re}call)}, \quad F-measure \in [0,1] \qquad \textbf{Eq. 1.15}$$

The area under the receiver operating characteristic curve (auROC): Wilcoxon-Mann-Whitney test statistic is used to calculate auROC. $auROC \in [0,1]$

Accuracy: This performance metric is defined as the total correct prediction of the classifier, *i.e.*, both positive and negative classes.

$$Accuracy = \frac{TP+TN}{TP\ +\ TN+FP+FN}, \quad Accuracy \in [0,1] \qquad \textbf{Eq. 1.16}$$

Matthews correlation coefficient (MCC): The formula of MCC is given below

$$MCC = \frac{(TP\ \cdot\ TN)\ -\ (FP\ \cdot\ FN)}{\sqrt{(TP\ +\ FP)\ \cdot\ (TP\ +\ FN)\ \cdot\ (TN\ +\ FP)\ \cdot\ (TN\ +\ FN)}}, \quad MCC \in [-1,1] \qquad \textbf{Eq. 1.17}$$

When the MCC value is 1, *i.e.*, then the best-trained model is selected, and the MCC value is -1, *i.e.*, then worst trained model is selected.

These performance metrics, *e.g.*, True Positive Rate (TPR), False Positive Rate (FPR), precision, recall, F-measure, Matthews correlation coefficient (MCC), area under the receiver operating characteristic curve (auROC), etc. are used to evaluate the trained model in supervised machine learning technique. These measures are statistically significant if sufficient labeled data are available.

1.6.4. Existing ML Strategies used for essential genes prediction

To illustrate the existing machine learning methods used to identify essential genes, we have included in **Table 1.2**. Most of the studies used supervised machine learning classifiers such as logistic regression [75,76], support vector machine [52–54,77], random forest [78], decision tree [75], ensemble [75] and probabilistic Bayesian-based methods [75,76,79], and instance-based learning methods such as K Nearest neighbor (K-NN) and Weighted KNN (WKNN) [80] have been used for gene essentiality prediction. Deep Learning strategies based on multi-layer perceptron networks have

also been used for essential genes prediction [81,82]. In these studies, researchers have mostly opted for simpler optimization methods for parameter tuning, such as the grid search technique, where the entire parameter space is explored in all possible combinations. The key advantage of these strategies lies in the fact that these models are capable of capturing the inherent patterns of an extensive array of biologically relevant 'features' that are distinctive and reflect the heterogeneous properties of essential genes.

Detailed reviews of the existing machine learning strategies for gene essentiality prediction have been discussed in different works of literature [36–38].

Table 1.2: Comparison of different machine learning approaches for predicting essential genes

Organisms	Type of Biological features	Feature Selection	Dimension Reduction	ML Classifier	Availability of Source Code / Webserver	Availability of Dataset	References
Saccharomyces cerevisiae	PPI, Sequence	No	No	Neural network and SVM	No	No	Chen and Xu, 2005 [83]
Escherichia coli and Saccharomyces cerevisiae	PPI, Sequence	Yes	No	Bayesian algorithm	No	Yes	Gustafson *et al.*, 2006 [79]
Yeast	Sequence	No	No	Bayesian algorithm	No	No	Seringhaus *et al.*, 2006 [84]
Saccharomyces cerevisiae	PPI, Sequence	Yes	No	*k*-Nearest neighbor, SVM	No	No	Saha and Heber, 2006 [80]
Escherichia coli	PPI	No	No	Decision tree	No	No	Silva *et al.*, 2007 [85]
Escherichia coli	MN, Sequence	Yes	No	SVM	No	No	Plaimas *et al.*, 2008 [53]
Saccharomyces cerevisiae	PPI and others	No	No	Decision tree	No	No	Acencio and Lemke,

							2009 [86]
Multiple organisms	MN, Sequence	No	No	Decision trees and SVM	No	No	Plaimas *et al.*, 2010 [54]
Multiple organisms	PPI, Sequence	No	No	Bayesian, logistical regression, decision tree and CN2 rule	No	No	Deng *et al.*, 2011 [75]
Mouse	PPI, Sequence	No	No	Logistic regression, random forest and SVM	No	No	Yuan *et al.*, 2012 [87]
Multiple organisms	PPI, Sequence	No	No	Bayesian algorithm	Yes	No	Cheng *et al.*, 2013 [76]
Yeast	PPI	No	No	Logistic regression	No	No	Li *et al.*, 2013 [88]
Multiple organisms	PPI, Sequence	No	No	Bayesian algorithm	No	No	Cheng *et al.*, 2014 [89]
Aspergillus fumigatus and yeast	Gene Co expression Network, Sequence	No	No	Bayesian algorithm, logistical regression, decision tree and CN2 rule	No	No	Lu *et al.*, 2014 [90]
Multiple organisms	Sequence	No	No	SVM	No	No	Ning *et al.*, 2014 [48]
Homo sapiens	PPI, Sequence	No	No	SVM, logistic regression and decision tree	No	No	Yang *et al.*, 2014 [91]
Arabidopsis thaliana, *Oryza sativa* and *Saccharomyces cerevisiae*	PPI, Sequence	No	No	Random forest	No	No	Lloyd *et al.*, 2015 [92]

Multiple organisms	Sequence	No	No	SVM	No	No	Hua *et al.*, 2016 [93]
Escherichia coli, *Streptococcus pneumoniae* TIGR4	PPI, Sequence	No	No	Principal component regression	No	No	Lin *et al.*, 2017 [94]
Homo sapiens	Sequence	Yes (SVM-RFE)	No	SVM	No	No	Guo *et al.*, 2017 [43]
Multiple organisms	Sequence	No	No	SVM	No	No	Li *et al.*, 2017 [95]
Multiple organisms	Sequence	Yes (LASSO)	No	SVM	No	No	Liu *et al.*, 2017 [96]
Multiple organisms	PPI Network	Yes (LASSO)	No	SVM	Yes	Yes	Azhagesan *et al.*, 2018 [51]
Drosophila melanogaster	Sequence, PPI Network	Yes (LASSO)	No	GLM, SVM, RF, ANN	No	No	Aromolara *et al. 2020* [97]
Multiple organisms (microbes)	Sequence	No	No	deep neural network (DNN)	Yes	Yes	Hasan *et al.* 2020 [82]

1.6.5. Existing Gene Essentiality Prediction Servers and Tools

Information of experimentally screened essential genes is continuously growing in a wide variety of organisms. Now, researchers are trying to annotate gene essentiality rapidly and reliably by various computational techniques using these experimentally screened essential genes information. The following sub-sections will discuss some of the existing essential gene-related web servers and tools such as Geptop, EGP, CEG_Match, and ZCURVE, respectively.

1.6.5.1. Geptop

Geptop is a webserver that utilizes sequenced-based orthology and phylogeny properties to annotate the gene essentiality in sequenced bacterial genomes. If a gene

has been conserved over time, it is more likely to be essential, especially in closely related species. For orthology estimation, the reciprocal best hit method was used. The Composition Vector method was used to calculate the phylogenetic distance between species. On the website, there is also a standalone open-source package available. Geptop can only annotate essential genes with sequenced genomes in bacteria. Furthermore, this platform archives essential genes from Geptop's 968 predicted bacterial genomes. Researchers can utilize this annotated gene essentiality information to conduct additional research [98].

1.6.5.2. EGP (Essential Gene Prediction)

EGP is a webserver for classifying genes' indispensability in bacteria genomes. This server is integrated with an SVM-based approach that only considers sequence-based attributes. Here sequenced-based properties, *i.e.*, codon usage, amino acid usage, di-nucleotide usage, and nucleotide position in three codon positions, are independently and jointly used. The training dataset is made up of 16 different bacterial genomes. Users require only nucleotide sequences as input of EGP. After analysis, the result will be shown in a pop-up web interface [48].

1.6.5.3. CEG_Match

CEG_Match is built on top of the CEG database as a gene's essentiality predicting tool. It annotates essential genes based on their functions and compares conventional gene id with the CEG database. This methodology is more effective than a direct blast against the CEG database. Users can enter either gene id or nucleotide sequences as fasta file format in CEG Match sever. However, CEG Match doesn't annotate gene essentiality when Gene id is unknown [35,99,100].

1.6.5.4. ZCURVE

ZCURVE is a standalone open-source package based on Z-curve theory to annotate gene essentiality in prokaryotic genomes [101]. Its most recent version, ZCURVE 3.0, integrated with Geptop server that can annotate essential genes in genomes, both

bacterial or archaeal. Users will get an output file containing gene essentiality information [102].

1.6.6. Limitations of Existing Machine Learning Strategies and webservers

Existing supervised machine learning algorithms (**Table 1.2**) have some limitations. A previous study by Gustafson *et al.* [79] used a Naïve Bayes model, to train similar features like amino acid composition, aromaticity, codon adaptation index, and frequency of optimal codons, which violate the fundamental assumption of statistical independence [103]. Few other studies by Cheng *et al.* and Deng *et al.* used CN2 rule-based classifier, decision trees, and logistic regression to classify genes [75,76]. Logistic regression suffers from an imbalance of training data [104], and heuristics search of an inappropriate decision tree will be stuck in local optima and might affect the classifier [105]. A similar problem exists with CN2 rule-based classifiers [106]. SVMs are affected by imbalanced training datasets with correlated and redundant features [107]. An SVM-based study by Hwang *et al*. fails to address both these issues appropriately [52], as the number of balanced training sets generated was reasonably low as compared to the number of available instances in each class; and features like clustering coefficient and clique level were redundant, holding similar interpretation.

Apart from this, protein interaction networks (PIN) have been used to calculate topological network features to classify gene essentiality [51,52]. However, these strategies fail for many organisms that do not hold the idea of centrality-lethality hypothesis in PIN [108]. However, only a few studies [53,54] have used flux-based features derived from metabolic networks to classify genes, calculated under a single environmental condition that does not represent a universal set of features.

Another challenge is that the heterogeneity of training and testing data. Training and testing datasets should be prepared from the same source and version of input files. For example, training dataset preparation of the PPI network, topological features for a particular organism were curated from the BioGRID database, and the testing

dataset preparation network was curated from the DIP database. Although two databases store PPI networks for the same organism, node connectivity and structural information differ. Thus, the prediction result is not reliable and accurate as the distribution of features is different between training and prediction.

The ML algorithms for essential gene prediction require a large amount of labeled data to train these models and predict the essentiality of unannotated genes accurately. These ML algorithms show very poor performance when the labeled dataset is imbalanced or limited. On the other hand, no source code is publicly available for these machine learning strategies (**Table 1.2**). The prerequisites for implementing these pipelines are challenging and time-consuming without training in advanced programming languages. Hence, it becomes difficult to apply these strategies for less explored organisms for predicting gene essentiality.

The existing platforms (**Section 1.6.5.**) can annotate essential genes for prokaryotes, not for eukaryotes. The prediction accuracy of the target organism from these webservers and tools is better when it shares a common phylogenetic ancestry with the available reference species having experimentally screened gene essentiality information. However, they underperform when the target organism is newly sequenced, and the proper gene id and phylogeny information are not available.

1.7. Objectives of the Thesis

Based on the previously discussed limitations and the requirements that needs to be addressed in order to identify gene essentiality, the major objective of this thesis is to develop a machine learning strategy for a more precise and accurate annotation of essential genes. The specific objectives are listed as:

- Developing a machine learning strategy (ML Strategy 1) for predicting essential genes in organisms where sufficient essentiality information (labeled data ≥ 80%) is available, but the dataset is imbalanced.

- Creating a machine learning strategy (ML Strategy 2) for predicting essential genes in organisms with limited essentiality (labeled data ≥ 1%) information.
- Developing a web server integrating the two previously established strategies (ML Strategy 1, ML Strategy 2) for essential genes prediction.

1.8. Organization of the Thesis

In **Chapter 1**, we have provided an overview of essential genes and existing strategies for annotating essential genes by covering the recent improvements and limitations of existing methods and their challenges. Various databases that collect gene essentiality information from experiments have been discussed in detail and we have explored diverse feature sets, which help determine a significant pattern between essential and non-essential genes. Finally, we have focused on previously used machine learning algorithms for gene essentiality prediction.

In **Chapter 2**, we have discussed the materials and methods with a detailed description of the two ML-based pipelines (ML Strategy 1, ML Strategy 2) developed to fulfil this thesis's objective. The calculation of biologically relevant features related to gene essentiality and corresponding software packages has been elucidated in detail.

In **Chapter 3,** we have described the results of ML strategy 1, developed to predict gene essentiality and annotate for less studied organisms where an experimentally known and the labeled dataset is sufficient (≥80%) but the dataset is imbalanced. We have combined supervised feature selection technique (SVM-RFE) and ML classifier SVM to predict gene essentiality from genome-scale metabolic networks. We have used a simple support vector machine-based learning strategy to predict essential genes in *Escherichia coli* K-12 MG1655 metabolism.

In **Chapter 4,** we have discussed the results of ML strategy 2, where gene essentiality was predicted with a limited (≥1%) labeled training dataset. This strategy has three components, *i.e.*, unsupervised feature selection technique, dimension reduction

using the Kamada-Kawai algorithm, and semi-supervised ML classifier employing Laplacian SVM to predict gene essentiality from genome-scale metabolic networks. We have validated ML strategy 2 on twelve organisms. We have used the methodology to annotate gene essentiality in less-studied organisms like *Leishmania donovani* and *Leishmania major*. Predictions for Leishmania species are further validated with the experimentally observed pattern of Reaction-Gene combinations occurring in other organisms. Using this semi-supervised ML strategy, we propose a new pipeline for essential gene prediction that shows universality in application to both prokaryotes and eukaryotes with limited labeled data.

In **Chapter 5,** we have described the development of the online gene prediction webserver, PRESGENE, by integrating our two proposed pipelines (ML Strategy 1 and ML Strategy 2). Users can submit and analyze their data for essential genes prediction through a user-friendly platform.

In **Chapter 6,** we have discussed the conclusion and the future direction of the thesis.

Chapter 2

Materials and Methods

In this chapter, we have discussed the methodology to fulfil the main goal of the thesis, *i.e.*, to develop machine learning (ML) strategies for the annotation of essential genes more precisely and accurately. ML models are a data-driven approach that learns the inherent patterns of data and predicts unknown data. This chapter summarizes the description of biologically relevant features to prepare a dataset for machine learning. Further on, an overview of the supervised machine learning strategy (ML Strategy 1) and a semi-supervised machine learning strategy (ML Strategy 2) for predicting essential genes in organisms is described.

2.1. Feature calculation for Training data and Testing data

For the purpose of generating a characteristic training dataset we have considered the metabolic genes from the reconstructed genome scale metabolic networks of both Prokaryotes and Eukaryotes [109]. The reconstruction consists of metabolites, reactions, and genes. In a genome-scale reconstructed metabolic network, the associations between genes, proteins, and reactions and the description of genes products catalyzing the associated reactions are usually described through logical expressions, which are referred to as gene-protein-reaction (GPR) rules. It has been observed that for a subset of reactions, there are either many genes that govern a single reaction (enzyme complexes) or a single gene that governs many reactions (depending upon multiple substrates that it catalyzes). For *e.g.*, gene b0002 within *Escherichia coli* K-12 MG1655 metabolism encodes for both aspartate kinase and homoserine dehydrogenase, whereas acetaldehyde dehydrogenase is encoded by genes b2388 and b0351. Hence from this GPR rule, reaction-gene combinations (R_a_G_b) were created. Creation of reaction-gene combinations directly provides insights into the role of a specific metabolic reaction catalyzed by a gene, deeming it

to be essential. No previous machine learning strategies (**Table 1.2 in Chapter 1**) for essential genes predictions have considered this feature.

For each reaction-gene combination we assembled sequence-based, gene expression-based, metabolic network and flux-coupled subnetwork-based features of target organism. List of features in each type has been given in **Table 2.1**. It is important to note that for situations where there were many genes catalyzing a single reaction, the sequence features related to each reaction-gene pair is distinct, whereas the network topological features remained the same; while, for situations where there were many reactions catalyzed by a single gene, the network features for each reaction-gene pair were different and sequence features remained the same.

2.1.1. Topological analysis of reaction and flux-coupled sub-network

To calculate topological features, we converted the genome scale metabolic network into two sub networks – Reaction Network and Flux Coupled Network.

2.1.1.1. *Reaction Network*

In the metabolic network of each target organism, we transform it into an undirected reaction network (RN), in which each node denotes an enzyme (reaction), and each edge represents the connection between two reactions that have common metabolites. The commonly used topological network features, such as centrality measures, that highlight the biological significance of an enzyme in a network were computed [55]. Generally, a central and highly connected enzyme in biological networks is often essential as it represents an important hub within the network [56]. If this hub node is blocked, then the whole pathway might be disrupted.

2.1.1.2. *Flux coupling network*

Flux-based calculations give a more realistic view of metabolic gene function. Typical computational methods like FBA perform these calculations on genome-scale metabolic networks to compute the flux (flow of metabolites) through an enzyme

(reaction) using a linear optimization procedure to maximize or minimize a defined objective function [110]. However, FBA is limited by environmental (exchange) constraints and the knowledge of an objective function. Hence, flux coupling analysis (FCA) [111,112] was performed on the genome-scale reconstructed metabolic network to avoid these dependencies while considering all input exchanges (representative of all environmental conditions) to be functional. The F2C2 tool v0.95b [112] was used for performing flux coupling analysis. The flux coupling network was derived from the metabolic network after flux coupling analysis (FCA). FCA is a flux-based optimization procedure that calculates reaction subsets that are either coupled with each other via flux or represent a set of block reactions, considering all input exchanges [111,112]. Let v_1 and v_2 be fluxes through reactions R_1 and R_2. Keeping either v_1 or v_2 as objective functions to be optimized, if a non-zero flux in v_1 imposes a non-zero flux in v_2 or vice versa, the two reaction fluxes are termed to be coupled with each other. If zeroing the flux of one reaction does not produce any effect on any other reaction within the metabolic network, then the reaction is termed to be uncoupled. If maximum or minimum of a particular reaction flux objective equals zero, then the reaction is termed to be blocked. Considering v_1 or v_2 to be objective functions, the coupled reactions can be classified into:

Fully coupled: If $v_1 = 0$ implies $v_2 = 0$ and if $v_2 = 0$ implies $v_1 = 0$, and $v_1 = v_2$, then the reaction pair is fully coupled.

Directionally coupled: If $v_1 = 0$ implies $v_2 = 0$ but if $v_2 = 0$ does not imply $v_1 = 0$, then the reaction pair is directionally coupled.

Partially coupled: If $v_1 = 0$ implies $v_2 = 0$ and if $v_2 = 0$ implies $v_1 = 0$, and $v_1 \neq v_2$, then the reaction pair is partially coupled.

Performing FCA on the metabolic network of target organism, we obtained fully, directionally, partially coupled reaction pairs and blocked reactions. As our aim was to find a flux-coupled subnetwork, the nature/property of each reaction pair can be

represented within an adjacency matrix where each reaction pair can be given a value of 1 or 0 corresponding to whether they are either coupled or not. Here, we have assigned a value of 0 to both uncoupled and blocked reaction pairs. The adjacency matrix represents a flux-coupled subgraph, which can be used to extract biologically relevant topological features dependent on predicted physiological flux relationships.

Eight centrality measures have been computed for both the reaction as well as the flux coupled networks, *viz*., Degree Centrality, Eigen vector Centrality, Eccentricity, Hub score, Authority Scores, Page Rank, Betweenness Centrality, and Number of triangles. A detailed description of all these centrality measures has been discussed in different literature [113–115]. These topological features have been calculated using the "igraph" package in R [116].

2.1.2. Features derived from the coding nucleotide sequence

Three types of features (*viz*. nucleotide content, codon usage bias, and information-theoretic features) of the metabolic genes extracted from the nucleotide sequence of the organisms that contribute towards gene essentiality. A brief description of the features has been discussed below.

2.1.2.1. Nucleotide content

Previous studies have elucidated that in bacterial genomes, GC content is correlated with the environmental condition in which the bacterium survives [117]. Hence, the related GC content of the genome of a target organism can be an essential feature for gene essentiality prediction. Another study showed that there is a significant difference in the distribution of the frequency of occurrence of A, T, G, and C nucleotides at the 3rd synonymous position of codons between the essential and non-essential genes [57]. These features were computed using an in-house code.

2.1.2.2. Codon usage bias

Protein abundance in an organism can be predicted by using Codon usage [118–120]. Highly expressing abundant proteins in metabolism might have functional importance and can be essential. Codon usage bias features, like Effective Number of Codons (ENC) [121] and Codon Adaptation Index (CAI) [119], were calculated using EMBOSS package version 6.6.0-1 [122].

a. Relative synonymous codon usage (RSCU)

RSCU is defined as the ratio of the observed frequency of a codon coding for an amino acid to the number of synonymous codons for that amino acid within a gene. The formula (**Eq. 2.1**) of the relative synonymous codon usage (RSCU) usage for the j^{th} codon coding for the i^{th} amino acid is given by,

$$RSCU_{ij} = \frac{c_{ij}}{\frac{1}{a_i}\sum_{j=1}^{a_i} c_{ij}} \qquad \textbf{Eq. 2.1}$$

where c_{ij} is the total number of codons, 'j' codon is coding for amino acid 'i', and a_i is the frequency of alternate codons coding for the amino acid.

b. Codon Adaptation Index (CAI)

Degree of translation selection of a gene can be measured by the codon usage index [119]. It is calculated for each gene with respect to a known set of highly expressing reference genes as a value of geometric mean of RSCU. The formula (**Eq. 2.2**) of CAI is given by,

$$CAI = \exp\frac{1}{L}\sum_{i=1}^{18}\sum_{j=1}^{a_i} c_{ij}\ln(w_{ij}), \qquad \textbf{Eq. 2.2}$$

$$\text{where } w_{ij} = \frac{RSCU_{ij}}{RSCU_{i\max}}$$

Where *L* is the length of the gene. The w_{ij} is calculated from a reference set of highly expressing genes. $RSCU_{imax}$ is the most frequently used codon for the i[th] amino acid of the relative synonymous codon usage. The CAI values are scaled in between 0 and 1. The value of CAI more than 0.5 indication of a high degree of translation selection.

c. Effect number of codons (ENC)

It has a strong relationship with the nucleotide composition at the 3[rd] synonymous position of the codon [121]. The formula (**Eq. 2.3**) of ENC index is,

$$ENC = 2 + \frac{9}{\bar{\hat{F}}_2} + \frac{1}{\bar{\hat{F}}_3} + \frac{5}{\bar{\hat{F}}_4} + \frac{3}{\bar{\hat{F}}_6} \qquad \textbf{Eq. 2.3}$$

where $\bar{\hat{F}}_i$ is the average codon homozygosity of '*i*' codons for amino acids having degeneracy. This index values are scaled between 20 and 61. A gene with ENC value of 20 shows a high codon usage bias whereas the value of 61 indicates a same contribution of each codon which is code for an amino acid.

2.1.2.3. Mutual Information (MI) and Conditional Mutual Information (CMI)

A previous study has used information-theoretic features such as mutual information (MI) and conditional mutual information (CMI), for essential genes prediction [49]. MI and CMI profile of coding nucleotide sequence can be used as genomic signatures which represent the phylogenetic relationship between genomic sequences [123]. A total of 80 features (16 MI and 64 CMI) have been computed by using in house Perl script.

2.1.3. Features derived from protein sequence

In order to investigate the dependence of gene essentiality on protein sequences, various derived and informatic features such as the frequencies of the amino acids, protein length, paralogy score, average Kidera factor, etc. have been considered in this study.

2.1.3.1. *Frequencies of the twenty amino acids and Protein length*

We used protein sequence related to the reaction-gene combination to calculate the occurrences of the 20 amino acids that reflect the physicochemical properties. These twenty features were calculated using EMBOSS package version 6.6.0-1 [122] and named according to their corresponding 20 amino acids.

2.1.3.2. *Paralogy based features (Paralogy score)*

The sequence similarity of a gene in its intragenome is called a paralogous gene of an organism. Paralogous genes have the same or similar types of biological functions. An organism may not be affected by the deletion of one of the paralogous genes because another paralogous gene may compensate for a similar type of function. So there are fewer chances for paralogous genes to be essential [42].

We calculated the paralogy score of a gene by performing a BLAST [version 2.2.26] search against the whole set of protein sequences of a target organism with different E-value threshold ranging from 10^{-3} to 10^{-30} with at least 40 % identity. Features based on paralogy score were labeled as P3 (E-value cut off 10^{-3}), P5 (E-value cut off 10^{-5}), P7 (E-value cut off 10^{-7}), P10 (E-value cut off 10^{-10}), P20 (E-value cut off 10^{-20}), P30 (E-value cut off 10^{-30}). These features have been calculated using in house Perl script.

2.1.3.3. *Homology based features*

A gene might be more important if it has been conserved across evolutionarily related organisms residing in different environments. In bacteria, essential genes were observed to be more evolutionarily conserved as compared to non-essential genes irrespective of the environment [42,124]. Phyletic Retention (PR) is defined as the number of organisms in which ortholog of a given gene is present [17]. For computing PR, protein orthologs amongst 710 bacterial genomes available from the COG database [125] were searched. An ortholog was defined such that, it is the only bi-directional best hit of the query gene in an organism and possessed at least 40%

identity with the query gene along with an E-value cut-off 10^{-7}. Bi-directional best hits for each gene were identified using BLAST version 2.2.26 along with the above parameters. Further, the number of homologs in the 710 genomes with respect to the hits obtained with different E-value cut-offs ranging from 10^{-3} to 10^{-30} (H3, H5, H7, H10, H20, H30) was also calculated [126].

2.1.3.4. Fourier sine and cosine coefficient

We used the Fourier sine and cosine coefficient of protein sequences [127] to observe if there are any inherent patterns that will help to classify between essential and non-essential genes. The Fourier coefficient (FC) is the converted numerical values of protein sequences, which describes the physical properties of corresponding amino acids. These physical properties represent the ten property factors using factor analysis introduced by Kidera *et al.* [128]. Mathematical representations (**Eq. 2.5, Eq. 2.4**) of these coefficients are given below:

$$FC\sin WN_k\_KF_n = a_k^{[n]} = \sum_{l=0}^{N-1} f_l^{[n]} \sin\left(\frac{2\pi kl}{N}\right) \quad \textbf{Eq. 2.4}$$

$$FC\cos WN_k\_KF_n = b_k^{[n]} = \sum_{l=0}^{N-1} f_l^{[n]} \cos\left(\frac{2\pi kl}{N}\right) \quad \textbf{Eq. 2.5}$$

Where the length of the protein sequence is N, $f_l^{[n]}$ is n^{th} property factor of amino acid *l*, and wavenumber is *k* (**Eq. 2.4**, **Eq. 2.5**).

Fourier sine and cosine coefficient in a specific range of Wave Number (WN) and Kidera Factor (KF) was calculated. The range of WN and KF are $0 \le k \le 7$ and $1 \le n \le 10$. It is also reported that global folding information of the protein is encoded in a specific range of wavenumber $0 \le k \le 7$ [127]. A total of 150 features were computed. These features have been calculated using in house Perl script.

2.1.3.5. Average Kidera Factor

The ten Kidera Factors (viz. KF1: Helix/bend preference, KF2: Side-chain size, KF3: Extended structure preference, KF4: Hydrophobicity, KF5: Double-bend preference, KF6: Partial specific volume, KF7: Flat extended preference, KF8: Occurrence in the alpha region, KF9: pK-C, KF10: Surrounding hydrophobicity) were derived by multivariate analysis on 20 amino acids using 188 physical properties and dimension reduction techniques [128]. The protein sequence of the corresponding reaction-gene combination was used to calculate ten features (AKF_i where, i= 1 to 10) by averaging the ten Kidera factors. These features have been calculated using in house Perl script.

2.1.4. Gene expression features

Essential genes tend to express at higher rates as compared to non-essential genes across bacteria [129]. To calculate gene expression-based features, we collected microarray experimental samples that were performed under different environmental stress conditions. The microarray studies for the target organism were curated from gene expression database [130]. Microarray studies carried out on mutant strains was not considered, as our aim was to predict essential genes in a wild type strain, subject to an array of environmental conditions. Also, the microarray experiments related to gene expression-based features were chosen such that it covers the expression profiles of genes under various environmental stress conditions to get a universal definition of gene irrespective of the environment. Average mRNA Expression of a gene (aveEXP) and mRNA Expression Fluctuation (mEF) which is the standard deviation of log2 normalized gene expression values of the Cy3/Cy5 intensity ratio of each gene from the above samples were calculated. From previous studies, it was reported that a gene might be important if it co-regulated with many other genes [131]. Hence, the Number of Genes with Similar Expression (NGSE), (number of gene pairs having a Pearson correlation coefficient: $r < -0.8$ and $r > 0.8$) [126] was also calculated.

Table 2.1: List of curated features

Feature Types	Features name	Abbreviation of features name	# of features	Used in Machine Learning Strategy 1	Used in Machine Learning Strategy 2
Topological analysis of reaction and flux-coupled sub-network					
Reaction Network	Degree Centrality	RN_degree	8	Yes	Yes
	Eigen vector Centrality	RN_eigen_vector_centrality			
	Eccentricity	RN_ecentricity			
	Hub Score	RN_hub_score			
	Authority Score	RN_authority_scores			
	Page Rank	RN_page_rank			
	Betweenness Centrality	RN_betweenness			
	Number of triangle	RN_number_of_triangle			
Flux Coupled Network	Degree Centrality	FCA_degree	8	Yes	Yes
	Eigen vector Centrality	FCA_eigen_vector_centrality			
	Eccentricity	FCA_ecentricity			
	Hub Score	FCA_hub_score			
	Authority Score	FCA_authority_scores			
	Page Rank	FCA_page_rank			
	Betweenness Centrality	FCA_betweenness			
	Number of triangle	FCA_number_of_triangle			
Features derived from the coding nucleotide sequence					
Derived features	Nucleotide content	A3, T3, G3, C3	4	Yes	Yes
	Effective Number of Codons	ENC	1	Yes	Yes
	Codon Adaptation Index	CAI	1	Yes	Yes
Informati on-	Mutual Information (MI)	MI_(X,Y)	16	No	Yes

theoretic features		where $X, Y \in \{A, T, G, C\}$			
	Conditional Mutual Information (CMI)	CMI_(X,Y,Z) where $X, Y, Z \in \{A, T, G, C\}$	64	No	Yes
Features derived from protein sequence					
Derived features	Frequencies of the twenty amino acids	Alanine, Cysteine, AsparticAcid, GlutamicAcid, Phenylalanine, Glycine, Histidine, Isoleucine, Lysine, Leucine, Asparagine, Proline, Glutamate, Arginine, Serine, Threonine, Valine, Tryptophan, Tyrosine, Methionine	20	Yes	Yes
	Protein length	PL	1	Yes	No
	Homology based features (Homology score)	H3, H5, H7, H10, H20, H30	6	Yes	No
	Paralogy based features (Paralogy score)	P3, P5, P7, P10, P20, P30	6	No	Yes
	Phyletic Retention	PR	1	Yes	No
Information-theoretic features	Fourier sine coefficient	$FC \sin WN_k \_ KF_n$ where $1 \le k \le 7$ and $1 \le n \le 10$	70	No	Yes
	Fourier cosine coefficient	$FC \cos WN_k \_ KF_n$ where $0 \le k \le 7$ and $1 \le n \le 10$	80	No	Yes
	Average Kidera Factor	AKF_i where i= 1 to 10	10	No	Yes
Gene Expression based features					
Expression based features	Number of Genes with Similar Expression	NGSE	1	Yes	No
	Average mRNA Expression of a gene	aveEXP	1	Yes	No

	mRNA Expression Fluctuation	mEF	1	Yes	No

2.2. The ML strategy 1 (Supervised): Essential Genes Prediction with sufficient labeled data

With an aim to create a highly precise machine learning model for binary classification of essential and non-essential genes, we have developed an integrated pipeline that addresses the problems of training a model using appropriate balanced dataset of instances with sufficient labeled data (≥80%), automated selection of relevant diverse biological features, identifying an optimized set of model parameters that classifies the chosen instances, and rigorous testing of the obtained data-driven trained model. The schematic view of the designed pipeline is shown in **Figure 2.1.** Algorithm for the pipeline (ML Strategy 1) is given in **Text A- 1, Annexure A**.

2.2.1. Training dataset preparation for ML Strategy 1

In this study to establish the ML Strategy 1, we have only considered the model organism *Escherichia coli*. For generating a training dataset, the metabolic genes from reconstructed genome scale metabolic network (iJO1366) of *Escherichia coli* were considered [109]. The reconstruction consists of 1805 metabolites, 2583 reactions, and 1367 genes. The information for essentiality (class label) of each reaction-gene combination was adopted from a known experimental study [15]. This particular study was selected as the gold standard because the essentiality of nearly all genes in *Escherichia coli* K-12 MG1655 has been tested in a variety of environmental conditions and confirmed using gene knockout techniques. Finally, a total training dataset of 4094 metabolic reaction-gene pairs were enlisted, out of which 384 were essential, 3120 were non-essential, and for around 590 reaction-gene pairs, no phenotype information was available. The known 384 essential and 3120 non-essential reaction-gene pairs were considered as the master (unbalanced) dataset of instances.

Thus, a total of 64 features (*i.e.*, Sequence-based, gene expression-based, metabolic network and flux-coupled subnetwork-based features) were obtained for each reaction-gene pair of *Escherichia coli* K-12 MG1655 metabolism. List of features in each type has been given in **Table 2.1**. We extracted coding nucleotide (CDS) and protein sequences for 1367 metabolic genes from *Escherichia coli* str. K-12 substr. MG1655 genome assembly GCA_000005845.2 available in NCBI GenBank [132] for curation of sequence-based features. It is important to note that, for the first time, flux-coupled subnetwork-based features have been introduced for essential genes classification. The detailed explanation of these features to signify gene essentiality is given in **Section 2.1.**

2.2.2. Components of ML Strategy 1

2.2.2.1. *Balancing the training dataset*

A major obstacle in classification using SVMs is the possibility of a significant class imbalance observed in the dataset used for training [107]. A large disproportion in the two classes may result in poor predictive capability of the model due to over-fitting of decision hyperplane, biased towards class with more number of instances. Previous SVM-based machine learning strategies for essential genes classification have attempted to overcome this problem by introducing a very small set of randomized balanced datasets for training [52]. But, generation of small sets of randomized balanced datasets fail to sample the entire population of genes sufficiently, thereby unable to obtain a perfect sample that can represent a population. This might affect the choice of a perfect training set leading to a sub-optimal model performance. To acquire the perfect training sample, the non-essential class was undersampled for sufficient number of times (1000 samples), so as to have numerous datasets containing equal number of essential and non-essential class labels. Sampling was performed such that each chosen non-essential sample is unique and not repeatedly chosen. The generated 1000 balanced training datasets

ensures that each non-essential reaction-gene pair is probably sampled at least once. The balanced datasets (BD_1 to BD_{1000}) thus generated were used for model training and subsequent testing.

2.2.2.2. *Feature Selection Methodology*

As the contribution of the 64 features towards essentiality of a gene was unknown, there may be a possibility of choosing redundant features for training. Redundant features can affect the predictive capability of the model [133]. Hence, it is incumbent to select a unique, non-redundant subset for training the model. Feature selection helps to enlist the most relevant biological features required for essential/non-essential reaction-gene classification and thereby reduces the feature dimensions for better construction of a hyperplane.

To perform feature selection, each of the balanced datasets was provided to SVM-RFE algorithm [134]. SVM-RFE has been previously established to be a useful strategy for feature selection in the context of essential genes classification [50]. SVM-RFE was performed using WEKA version 3.8 [135]. In SVM-RFE, firstly the features are ranked. To obtain the best set of features, iteratively each top 'n' feature combination, where n = number of ranked features chosen, ranging from 1, 2, …, 64, was selected and given to the Sequential Minimal Optimization (SMO) [136] algorithm for classification while performing 10 fold cross-validation. The feature combination that gave the best performance [with respect to area under Receiver-Operator-Characteristic curve (auROC)] was chosen for each dataset. Thus, corresponding 1000 well-performing feature combinations were shortlisted. Each of these combinations was again trained for the 1000 balanced datasets (BFC_i_BD_j, where i=1 to 1000 and j=1 to 1000). Average auROC of the 1000 trained models for each best feature set was calculated. Out of all the feature sets, the feature set giving the average highest performance was considered to be the best of all best feature combinations (BFC_{best}).

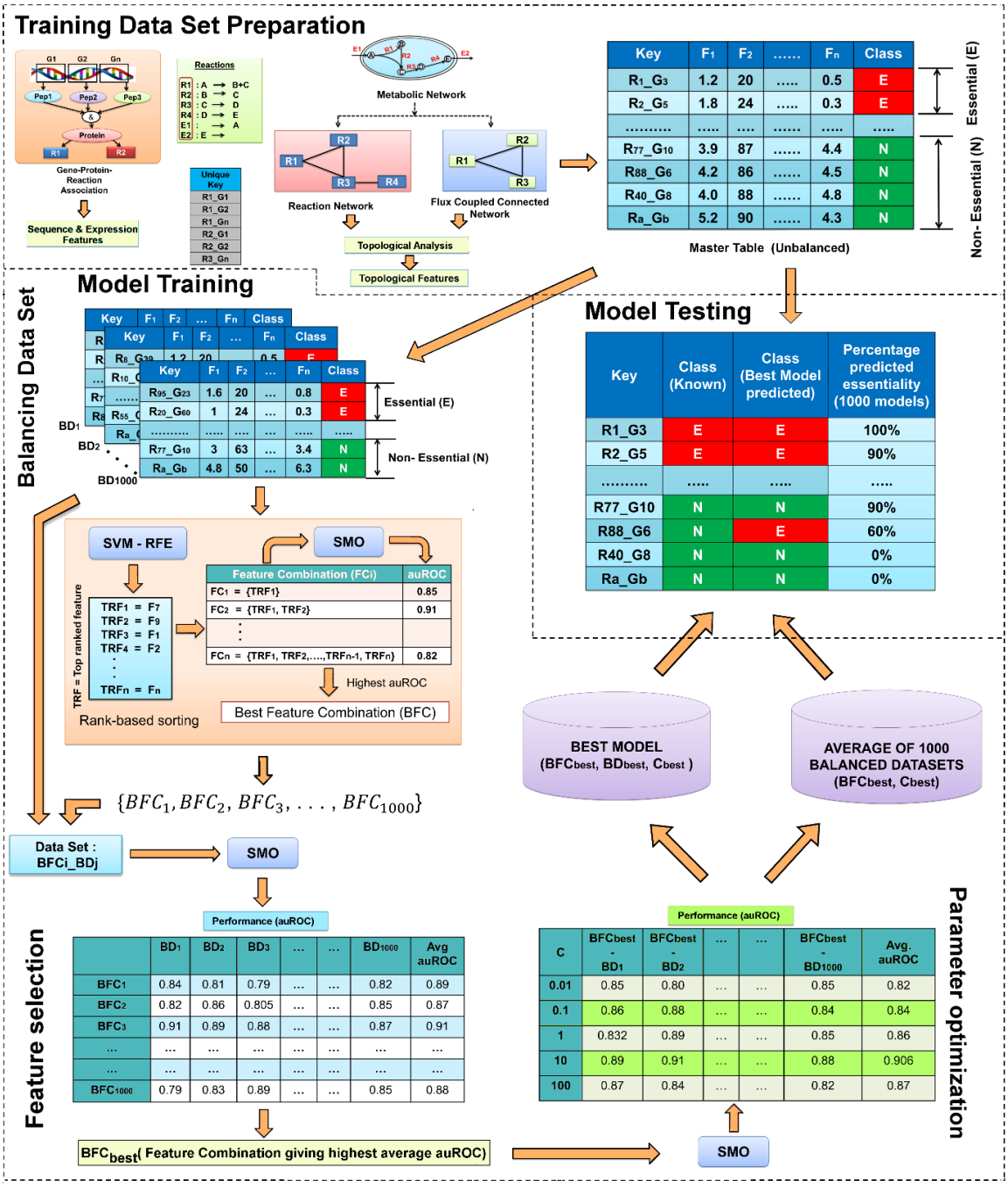


Figure 2.1. The work flow of ML Strategy 1. The integrated pipeline for prediction of essential genes based on a given input unbalanced training dataset consisting of reaction-gene pairs with sequence, expression, and network topological features.

2.2.2.3. *Parameter Optimization of Classifier/SVM model for classification*

In order to obtain globally optimum hyperplane fit, the penalty parameter (C) of the SMO algorithm was fixed at different values (0.01, 0.1, 1, 10 and 100) and trained again for the 1000 datasets while performing a 10-fold cross validation with the above

selected best feature combination (BFC_{best}_BD_i where i =1 to 1000). The penalty parameter that gives highest average auROC was selected (C_{best}). The parameters of the linear kernel function were set to default in each case. Finally, a best feature set, best training dataset, and best parameter combination (BFC_{best}, BD_{best}, C_{best}) can be obtained, which defines the "best" chosen model from our strategy (See **Figure 2.1**). This best-chosen model was further used for comparison with other published models, testing, and predictions.

2.2.2.4. Performance metrics

A number of performance metrics were used to evaluate the model. The definitions of each of these metrics are given in **Section 1.6.3.5.** of **Chapter 1** . Previous SVM-based classification strategies have calculated model performance metrics with respect to only the essential (positive) class [52]. To understand the strength of model to classify instances into both the classes (E and N), a weighted average of each metric was calculated and considered for measuring the true performance of the model strategy.

Let *M* be the total set of performance metrics.

M = {TPR, FPR, Precision, Recall, F-measure, MCC, auROC}.

The **weighted average (Eq. 2.6)** of each metric for measuring model performance was computed by following formula:

$$\text{Weighted\_Metric}_i = \frac{(M_{ip} \times PI) + (M_{in} \times NI)}{PI + NI} \qquad \textbf{Eq. 2.6}$$

where, $i \in M$,

M_{ip}, performance metric for positive class,

M_{in}, performance metric for negative class,

PI is the number of positive instances,

NI is the number of negative instances.

2.2.3. Model testing

Class labels from the model can be predicted with respect to the best training sample chosen from the population of the 1000 balanced datasets of known genes (best chosen model). But, by doing so, there is an inherent bias towards the chosen sample for training. This bias can be avoided by checking whether the class label predicted by the best model compares to the class labels predicted by models trained on all the samples. In our study, we present and compare the class labels of the genes, predicted by both the best model and by the 1000 trained models; which none of the previous studies have provided. Hence, the master unbalanced dataset was provided as a testing dataset to –

1) best model trained for the randomized dataset which gives the best performance (BFC_{best}, BD_{best}, C_{best}). The predicted phenotype of each reaction-gene pair is assigned to be essential (E) or non-essential (N) as predicted by the best model.

2) best model (BFC_{best}, C_{best}) trained for each of the 1000 random datasets generated (1000 trained models): The percentage of models that predict the phenotype of each reaction-gene pair is computed. If 80% (out of 1000) trained models predict same phenotype, we have considered that phenotype (essential or non-essential) for reaction-gene pair. It is worth mentioning that this threshold is user-defined and can be changed.

2.2.4. Dataset curation of other prokaryotes

Published essentiality datasets from two different prokaryotes namely, *Brevundimonas subvibriodes* ATCC 15264 and *Helicobacter pylori* 26695 were obtained from the Database of Essential Genes (DEG) version 13.3. [137] As no curated genome-scale metabolic network was available for *Brevundimonas subvibriodes* ATCC 15264, only nucleotide and amino acid sequence composition-based features were calculated and used for model training. In case of *Helicobacter pylori* 26695, both sequence-based and metabolic network-based features were calculated. A published

genome-scale metabolic network iIT341 [138] was used for computing reaction network and flux-coupled subnetwork based features. As very few gene expression studies were available for both the species, model training in each case was performed without use of gene-expression based features.

2.3. The ML strategy 2 (Semi-Supervised): Essential Genes Prediction with limited labeled data

We have developed the ML Strategy 2 to predict gene essentiality, as elucidated in **Figure 2.2**, which combines feature selection technique based on a space-filling concept, dimension reduction (DR) using the Kamada-Kawai (KK) algorithm, and classification of genes based on a semi-supervised machine learning algorithm employing Laplacian Support Vector Machine (LapSVM). This pipeline combines heterogeneous biological features, such as sequence-based and network-based features. It classifies genes based on a training dataset of very limited information (1% of labeled data) of essential genes from experimental data. Twelve organisms comprising of both Prokaryotes and Eukaryotes (**Table 2.2**) with well-annotated gene essentiality information from the OGEE database [33] have been considered to validate this proposed machine strategy 2, and the subsequent prediction of essential genes in *Leishmania major* and *Leishmania donovani* have been performed. The gene essentiality information has only been considered from the OGEE database as this collates data using text mining as well as manually verified with experimental data, unlike other gene essentiality databases that rely on only text mining.

2.3.1. Training data and Testing dataset preparation and integration of heterogeneous features

The training datasets for the pipeline (ML Strategy 2) of the 12 target organisms were prepared by calculating mainly two types of features: topological features and sequenced-based features. These features were extracted primarily from the genome-scale reconstructed metabolic networks, the fasta files containing the coding

Table 2.2. Organisms considered for model training and validation

Organism Name	Abbreviation	Input files	
		FASTA files of coding nucleotide and protein sequence (RefSeq assembly accession)	**Genome-Scale Reconstructed Metabolic Network**
Organisms used for Model Development and Validation of the Proposed Pipeline (ML Strategy 2)			
Acinetobacter sp. ADP1	ACIAD	GCF_000046845.1_ASM4684v1	iAbaylyiv4 [140]
Bacillus subtilis subsp. subtilis str. 168	BACSU	GCF_000009045.1_ASM904v1	iYO844 [141]
Escherichia coli K-12 MG1655	ECOLI	GCF_000005845.2_ASM584v2	iJO1366 [142]
Helicobacter pylori	HELPY	GCF_000008525.1_ASM852v1	iIT341 [138]
Mycobacterium tuberculosis H37Rv	MYCTU	GCF_000195955.2_ASM19595v2	iNJ661 [143]
Pseudomonas aeruginosa PAO1	PSEAE	GCF_000006765.1_ASM676v1	iPae1146 [144]
Pseudomonas aeruginosa UCBPP-PA14	PSEAB	GCF_000014625.1_ASM1462v1	iPau1129 [144]
Salmonella enterica subsp. enterica serovar Typhimurium LT2	SALTY	GCF_000006945.2_ASM694v2	STM_v1_0 [145]
Staphylococcus aureus subsp. aureus NCTC 8325	STAAB	GCF_000013425.1_ASM1342v1	BMID000000141098 [146]
Saccharomyces cerevisiae	YEAST	GCF_000146045.2_R64	iMM904 [147]
Caenorhabditis elegans	CELEG	GCF_000002985.6_WBcel235	iCEL1273 [148]
Mus musculus	MUSMU	GCF_000001635.26_GRCm38.p6	iMM1415 [149]
Organisms used for Case Study			
Leishmania donovani	LDONO	TriTrypDB-36	iMS604 [150]
Leishmania major	LMAFR	TriTrypDB-36	iAC560 [151]

nucleotide sequences of the genes, and protein sequences of these target organisms (**Table 2.2**) [139]. From the genome-scale reconstructed metabolic network, the information of metabolites, reactions, and genes was collated.

The sequence-based features and the topological features of the metabolic reaction network and flux-coupled sub-network based were calculated and accumulated for each reaction-gene combination. These reaction-gene combinations integrate diverse features of the metabolic adaptation of the organism and give detailed insights into the role of a particular gene in the metabolic reaction network. This helps in the prediction of the essentiality of the gene in the target organism with high accuracy. A total of 289 features were computed for each reaction-gene pair. Brief descriptions of these features are given below, and their abbreviation**s** are enlisted in **Section 2.1.** To establish the model consistency and reproducibility of the proposed pipeline (ML Strategy 2), two different types of datasets for each of the twelve organisms have been used. The first type of dataset consists of 80% data points of the total dataset with limited labeled data that is used for training, while the remaining 20% is used for blind testing to check the model validation. Using this 80% data points of the whole dataset, different types of training dataset are further created with limited labeled data points in the range, *i.e.*, i % Labeled (L) and (100 - i%) Unlabeled (UL) data, where i = 1, 2, 3, 4, 5, 10, 30, 50, 70 and 90. In each category, labeled samples were chosen randomly from the master table. It is to be mentioned here that this selection of labeled data was conditionally randomized to ensure that both the essential and non-essential genes categories appear with equal probability. In this way, 100 datasets in each labeled category have been created.

The second type of dataset consists of the whole dataset with limited labeled data used for model training and prediction purposes for each of the twelve organisms. It is to be mentioned here that, in less-studied organisms where gene essentiality information is very less, a blind test cannot be applied. For those cases, the whole

dataset with limited labeled data will be used for model training and prediction purposes.

2.3.2. Components of ML Strategy 2

2.3.2.1. Feature selection based on the space-filling concept

The contribution of these 289 features towards gene essentiality is unknown; hence, there may be a possibility to select redundant features by the feature selection algorithm. These redundant features may affect the training performance of the machine learning model. Hence, it is important as well as challenging to choose the non-redundant, unique feature subset for training the model. Feature selection helps to capture the most relevant biological features and helps the classifier to learn a better way to predict essential and non-essential genes with high accuracy. Here the unsupervised feature selection method based on the space-filling concept has been used [152]. This unsupervised method selects the features based on a coverage measure that estimates the spatial distribution of the data points in a hypercube and ensures uniform distribution of points in a regular grid in the data space. The method captures the variability of features with new and relevant information about the data. This method has been tested on various datasets and different scenarios with noise injection and data shuffling. The benefits of using this algorithm are two folds. Firstly, being an unsupervised algorithm, prior information of the output variable is not required.

Additionally, here no classifier is required for feature selection. Hence time complexity is less in comparison to other feature selection algorithms, like SVM-RFE. Also, it has been observed that this method gives better information of relevant features than other unsupervised correlation-based feature selection techniques that, although it can remove the redundant features, cannot eliminate the features with low variability that are non-relevant and non-informative for classification [153,154].

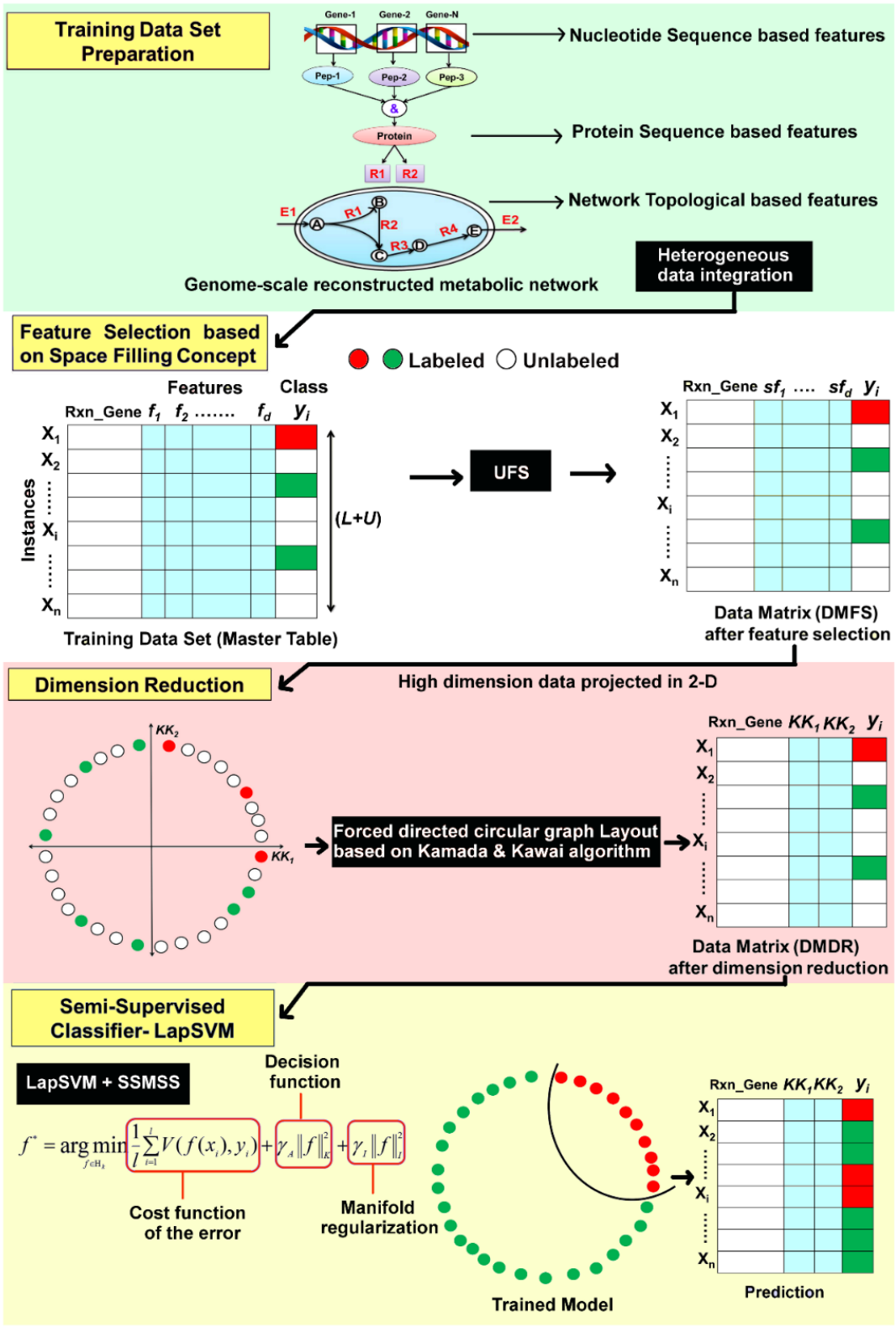


Figure 2.2. The work flow of ML Strategy 2. The integrated pipeline for prediction of essential genes based on limited labeled training dataset consisting of reaction-gene pairs with sequence, informatics, and topological network features.

2.3.2.2. Dimension reduction using force directed graph layout

After feature selection, the dataset was transformed into a lower dimension (2-D) using a dimension reduction technique for visualization. Projected 2-D features set to reserve all the information the same as higher-dimensional data. This is an important step in the pipeline as the classifier works better in 2-D than with the higher dimension data. For dimension reduction, a force-directed graph layout algorithm Kamada-Kawai has been used that considers each data point as a node in a graph having attractive and repulsive forces between them that can be modeled as springs connecting the nodes [155]. This algorithm aims to minimize the system's total energy (**Eq. B- 1, Annexure B**) based on attractive and repulsive forces between them, thereby clustering with similar data points [155]. Here the input of the Kamada-Kawai algorithm is a graph constructed by using the K-Nearest Neighbour (K-NN) algorithm. For known organisms, it has been observed that essential genes are clustered together in one side of an arc in a circle layout, and non-essential genes are clustered in the rest of the circle. A circular layout of each organism has been observed from the Kamada Kawai algorithm with a specific parameter (K Nearest Neighbor) value of the K-NN algorithm. Here it is assumed that if a similar circular layout is observed for less explored organisms related to gene essentiality, the unlabeled genes will be clustered together category wise and reside on the arc of the circle. This analysis had been performed using the "dimRed" package in R [156].

Both the feature selection and the dimensionality reduction methods are used for not only reducing the number of features in a dataset but also to select the important features, which are contributing significantly. Feature selection is used for selecting the relevant features without changing the original values, whereas, the dimensionality reduction step transforms the higher dimensional features into a lower dimension. From the dimension reduction technique, it is very difficult to identify the key features which are contributing for classifications; hence the feature selection step is necessary.

To test the efficiency of this dimension reduction technique combined with unsupervised feature selection and LapSVM classifier, the performance metrics of Kamada-Kawai has been compared against other dimension reduction techniques, such as Principal Component Analysis (PCA) [59], Metric Dimensional Scaling (MDS) [60], Fruchterman Reingold [157] and FastICA [158] using the gold standard dataset of twelve organisms. To test the statistical significance of the results, the one-tailed Mann Whitney U Test has been performed with 1% level of significance (P<0.01).

2.3.2.3. Semi-supervised classifier: Laplacian SVM

Essential genes classification using the machine learning technique can be a difficult task when a minimal amount of gene essentiality information for the target organism is available. In this setting, semi-supervised learning is an appropriate approach that builds a trained model from labeled and unlabeled samples [73]. Most of these semi-supervised algorithms follow two common assumptions, *i.e.*, cluster assumption and manifold assumption. Cluster assumption states that data points in the same cluster have a chance of having the same class label. Manifold assumption means that close data points along the manifold area follow similar data structures or similar class labels. However, cluster assumption follows the global feature, and manifold assumption follows the local features in the model.

Laplacian support vector machine (LapSVM) is a graph-based semi-supervised learning method based on a manifold regularization framework [74]. The graph is constructed from labeled and unlabeled data as the node. The similarity between data points in a graph can be assigned by edge weight, which is calculated from the K-NN algorithm. In this way, the information of labeled data points can be passed to another node, and then, the unlabeled nodes can be labeled. The input dataset being circular (non-linear), Radial Basis Function (RBF) kernel with the classifier LapSVM

have been used. This analysis had been performed using the "RSSL" package in R [159].

2.3.2.4. The score for best model selection

There are various performance metrics, *e.g.*, True Positive Rate (TPR), False Positive Rate (FPR), precision, recall, F-measure, Matthews correlation coefficient (MCC), Area under the receiver operating characteristic curve (auROC), etc. to evaluate the trained model in supervised machine learning technique. These measures are statistically significant if sufficient labeled data are available. However, due to limited labeled data, these metrics will not work for best model selection in a semi-supervised type algorithm. To circumvent the above problem, a new measure has been proposed, called the Semi-Supervised Model Selection Score (SSMSS), for selecting the best model. This SSMSS score is dependent on four different measurements (**Eq. 2.7, Eq. 2.8**). For this, the training dataset, having limited labeled reference, has been labeled as ground truth (GT) reference. Another reference set called the pseudo reference (PR) has been considered by calculating the distance from unlabeled data points to the labeled dataset. The dataset containing the predicted labels by the Laplacian SVM classifier has been labeled as the Laplacian Reference (LR). Thereafter, Silhouette Index (SI) [160] was computed to check the clustering grouping quality. The $\mathrm{CorrectPrediction}_{GT\_LR}$ measure was calculated based on the matches between the predictions of the Laplacian SVM classifier with the Ground Truth data. Here, the calculation of the MCC with the help of Pseudo-reference and Laplacian Reference was represented as MCC_{PR_LR}. Silhouette Index calculation based on Pseudo Reference and Laplacian Reference was denoted by SI_{PR} and SI_{LR} respectively. Based on these parameters, the values of the proposed Semi-Supervised Model Selection Score (SSMSS) may vary from 0 to 1. If any of the above four measurements is low, then the SSMSS value will be drastically decreased. The best model will be selected from 64 models which has the highest SSMSS value for each

dataset in different parameters combinations, *i.e.*, kernel parameter [Radial Basis Function (RBF) kernel parameter sigma (σ)] and LapSVM parameters [lambda (λ): L_2 regularization parameter and gamma (γ): the weight of the unlabeled data]. It may be mentioned here that the score will not consider those models which have negative Silhouette Index and MCC value. The parameters (σ, λ, γ) have been varied with four different values, *i.e.*, 0.01,0.1,1,10. Therefore, by tuning these model parameters using grid search, 64 models for each dataset have been generated. The following equation (**Eq. 2.7, Eq. 2.8**) has been proposed for the calculation of the SSMSS.

$$SSMSS_{k=1\ to\ 64} = \min\left\{Correct\,\Pr ediction^{k}_{GT\_LR}\ ,\ MCC^{k}_{PR\_LR}\ ,\ SI^{k}_{PR}\ ,\ SI^{k}_{LR}\right\}, \quad \textbf{Eq. 2.7}$$

$$\forall MCC^{k}_{PR\_LR} \geq 0, SI^{k}_{PR} \geq 0, SI^{k}_{LR} \geq 0.$$

$$SSMSS_{best} = \max\left\{SSMSS_{k=1}, SSMSS_{k=2}, ..., SSMSS_{k=64}\right\}, \quad \textbf{Eq. 2.8}$$

where k is the k^{th} model with a particular parametric combination and $SSMSS_{best}$ is the best score of the best model among these 64 models.

2.3.2.5. Time complexity of the proposed ML Strategy 2

The proposed pipeline has three components (*i.e.*, Unsupervised Feature Selection, Kamada Kawai Dimension Reduction Technique, and LapSVM semi-supervised classifiers), which work sequentially. To calculate the total time complexity T(n,d) of the proposed strategy (ML Strategy 2), the cumulative effect of all three components have been considered, where n denotes the number of data points (reaction-gene pair) that depends on the size of the metabolic network of the organism, and d is the total number of features.

The time required for each of the three components can be represented as follows

(**Eq. 2.9**) [74,152,155]:

Time required for Unsupervised Feature Selection algorithm $= O\left(\frac{d(d+1)n^2}{2}\right)$

Time required for Kamada Kawai algorithm= $O(n^3)$

Time Required for LapSVM= $O(n^3)$

Therefore, the total time required T(n,d) can be represented as:

$$\therefore T(n,d) = \frac{d(d+1)n^2}{2} + n^3 + n^3 \qquad \textbf{Eq. 2.9}$$

$$or, \quad T(n,d) \le 4n^3 + n^2(d^2+d)$$

$$or, \quad T(n,d) \le (4+d+d^2)n^3$$

$$or, \quad T(n,d) \le Cd^2n^3$$

$$or, \quad T(n,d) = \mathrm{O}(d^2n^3)$$

Where, C is a constant, in particular, $C \ge 6 \quad \forall d, n \in \mathbb{N}$.

Therefore, the total time complexity of the proposed ML Strategy 2 is $\mathrm{O}(d^2n^3)$.

2.3.3. Gene Essentiality Prediction, Experimental Validation, and Pathway Enrichment

The essential genes prediction results for the twelve model organisms have been compared with experimental data obtained from the OGEE database, and the corresponding supervised performance metrics such as TPR, FPR, MCC, auROC, etc., were calculated. Further, the predicted essentiality information of the reaction-gene pairs of all twelve organisms has been categorized into five different groups based on their involvement in different reactions. These five groups are following: **CEN** (Combination of Essential and Non-essential), involving both essential and non-essential genes controlling a reaction; **ME** (Multiple Essential), multiple essential genes involved in a reaction; **MN** (Multiple Non-essential), multiple non-essential genes governed a reaction; **SE** (Single Essential), single essential genes involved in a reaction; **SN** (Single Non-essential), single non-essential genes involved in a reaction.

Thereafter, the distributions of the five categories of reaction-gene pairs from the predicted results have been compared with the distribution observed in experimental data for all the organisms using the Chi-Square Test (1% level of significance).

For *Leishmania donovani* and *Leishmania major,* the best model was selected based on the SSMSS score for the prediction of the essential reaction-gene combinations. These predicted reaction gene combinations were then classified into five categories, like the other twelve species. The list of unique genes that were extracted from these predicted essential reaction-gene pairs was analyzed for their associated Gene Ontology (GO) terms [161,162] from the Uniprot database [163]. The percentages of genes associated with each GO term were calculated for both organisms. Additionally, using the DAVID pathways enrichment tool [164], the essential genes were further analyzed to identify the significantly enriched KEGG pathways [165] that were associated with these essential genes.

Source codes of the entire ML strategy 2 and pipeline are given in **Text B- 2, Annexure B**, which consists, Training dataset preparation and integration of heterogeneous features, Feature selection based on the space-filling concept, Dimension reduction using forced directed graph layout, and Semi-supervised classifier: LapSVM.

Chapter 3

Essential genes prediction using ML strategy 1 (Supervised) for organisms when sufficient gene essentiality information is available

3.1. Motivation

Machine learning algorithms for gene essentiality prediction depend on the diverse, independent features that can classify instances, the distribution pattern of instances with respect to chosen features, and the number of equal instances belonging to each class. Widely used machine learning algorithms for essentiality classification in *Escherichia coli* include support vector machines (SVMs) [52–54], ensemble-based machine learning [75], probabilistic Bayesian-based [75,76,79], logistic regression [75,76] and decision-tree based [75] algorithms. Although these studies use powerful classifiers, a few technical problems are associated with them. A previous study [79] used a Naïve Bayes model, trained for similar features like amino acid composition, aromaticity, codon adaptation index, and frequency of optimal codons, which violate the fundamental assumption of statistical independence [103]. Few other studies use CN2 rule-based classifier, decision trees, and logistic regression to classify genes [75,76]. Logistic regression suffers from an imbalance in training data [104]. Inappropriate decision tree heuristics result in local optima and might affect the decision tree classifier [105]. A similar problem is with CN2 rule-based classifiers [106]. Likewise, SVMs are also affected by imbalanced training datasets and use of correlated or redundant features [107]. A recent SVM-based study fails to address both these issues appropriately [52], as the number of balanced training sets generated was reasonably low as compared to the number of available instances in each class; and features like clustering coefficient and clique level were redundant, holding similar interpretation.

On the other hand, the features required for determining gene dispensability should be biologically relevant, unique, and consider heterogeneous properties of genes. The above machine learning-based studies utilize an array of sequence, structural, and pathway features to classify genes based on their dispensability [37]. With the availability of high number of sequenced and annotated genomes, features from nucleotide and protein sequences were largely used for training machine learning models of classification [166]. Nucleotide sequence properties like codon adaptation index, phyletic retention, GC content, protein sequence properties like amino acid frequency, and protein length are known indicators of gene essentiality across bacteria [167,168]. More recently, features related to gene expression and biological networks gained more importance as compared to static genomic features, as they represented an organismal phenotype. In *Escherichia coli*, topological network features of the protein interaction networks (PIN) were used along with sequence-related features to classify essential from non-essential genes [52]. However, the idea of a centrality-lethality hypothesis in a PIN might not hold true for many organisms [108]. Further, their roles in the context of signaling and metabolic pathways cannot be inferred only using interaction information. For this purpose, different network representations need to be analyzed. Few studies use topological and flux-based features from metabolic networks to classify genes [53,54]. The flux features used in these studies were principally calculated using Flux Balance Analysis (FBA) [110] under a single environmental condition (aerobic glucose input) while optimizing for the biomass objective. Hence, the calculated flux features are condition-specific and do not represent a universal set of features. Previous studies clearly establish that the essentiality of a gene is highly dependent on its adaptability to any environment [169]. To tackle the aforementioned problems inherent within imbalanced training sets, feature bias, and limitations of learning algorithms, we have developed a simple yet powerful integrative machine learning strategy (ML Strategy 1) based on a fundamental SVM-based implementation for binary classification of genes based on gene with sufficient labeled data (≥80%) and protein sequence, gene expression,

network topological and flux-based features for *Escherichia coli* K-12 MG1655 metabolism (detailed description is provide in **Section 2.2. , Chapter 2**). This integrative machine learning strategy attempts selection of the most contributing genotype, phenotype features required for classification by SVM-Recursive Feature Elimination (SVM-RFE), choice of best parameters for the learning model, and removal of the dependency (bias) of a model on a given balanced training dataset to give a highly accurate, unbiased, predictive machine learning model for appropriate classification of genes based on their essentiality. To account for the inherent limitation of environmental dependence to calculate flux distributions through a metabolic network, we perform flux coupling analysis (FCA) [111,112] on the *Escherichia coli* iJO1366 metabolic network while considering all the possible input conditions, and for the first time, incorporate the network topological features of the obtained flux-coupled subnetwork to our model strategy for achieving higher classification performances (**Table 2.1, Section 2.2.1., Chapter 2**). We hypothesize that this curated and selected feature set represents the minimal organism-specific constraints that govern the essentiality of a gene in *Escherichia coli* K-12 MG1655 metabolism. The model (**Section 2.2., Chapter 2**) generated from the selected features, when tested with experimentally known *Escherichia coli* Keio collection dataset [15] predicted 94.28% of total known essential genes to be essential and 82.59% of total known non-essential genes to be non-essential. Further, it is to be noted that, by virtue of the selected features for training, our method is able to capture the minimal set of essential genes that prove to be essential in any given environment. Our method was also able to predict essentiality of 317 genes, previously unidentified by genome-scale knockout experiments. Our methodology (**Section 2.2. in Chapter 2**) was also trained with datasets of other recent supervised classification techniques for essential genes classification and tested using their reported test datasets [75,170]. Test results indicate that our method achieves the highest sensitivity and specificity as compared to the recent supervised classification techniques, irrespective of any input training dataset. Finally, from our analysis, we

establish that a simple machine learning strategy is enough to predict dispensable genes, when provided with an appropriate choice of features combined with the best hyperplane choice in the feature space. Also, as an applicative methodology, our proposed ML Strategy 1 (**Section 2.2., Chapter 2**) can be used in organisms, where essentiality phenotype of genes is majorly (labeled data ≥ 80%) known and dataset is imbalanced.

3.2. Results

3.2.1. Comparison of proposed ML Strategy 1 with a known machine learning strategy for gene essentiality classification

To establish the predictive ability of our proposed pipeline (ML Strategy 1) in identifying essentiality of a gene, classification performance of our strategy (**Section 2.2., Chapter 2**) was compared with a known strategy, both trained with two different input (training) datasets. The former dataset [171] contains sequence and protein-protein interaction network features of *Escherichia coli*. The latter dataset is our curated dataset containing sequence, gene expression and network topological features derived from *Escherichia coli* K-12 MG1655 metabolism. The comparison of the two strategies was performed using different model performance metrics (see **Table 3.1**).

3.2.2. Comparison using a known dataset

To assess the universality of our model performance with any given dataset, training of our model was performed on a known training dataset and its performance was compared with the known strategy [171]. The known strategy involves the Sequential Minimal Optimization (SMO) algorithm with linear kernel based SVM classification for training model on their dataset. Using our method on the known dataset [171] a significantly improved classification performance was achieved as compared to the previous available strategy (**Table 3.1**). This can be observed from the improved MCC (0.675) and F-measure values (0.826). The above comparison indicates that our

model is superior in performance, as compared to Hwang *et al.*'s method [52]. Our method was also compared with other types of supervised machine learning methods, available for essential genes classification by using their training and test datasets (**Table 3.1**). The comparative results indicate that our method outperforms all other known supervised classification methods achieving a very high sensitivity and specificity.

3.2.3. Comparison using our curated dataset

Our curated dataset (**Table 2.1, Section 2.2.1., Chapter 1**) was also used as an alternative dataset for model training, and compared with the known strategy [52]. From **Table 3.1**, it can be observed that our proposed ML Strategy 1 again gave comparatively better performance as compared to the known strategy. An increase in MCC from 0.740 to 0.814 was observed suggesting a significant difference in training accuracy.

Table 3.1: Comparison of our proposed ML strategy 1 with Hwang *et al.* (2009) [171]

Performance metric	Known dataset [52]		Our dataset (Section 2.2.1., Chapter 2)	
	Known strategy [52]*	ML Strategy 1	Known strategy [171]	ML Strategy 1
Precision	0.828	0.877	0.846	0.907
Recall	0.745	0.78	0.903	0.906
F-Measure	0.784	0.826	0.874	0.906
MCC	0.593	0.675	0.740	0.814

*Performance measure as reported in Hwang *et al.* (2009) [52]

3.2.4. Improving model performance by class balancing and feature selection

To test the effect of feature selection and class balancing, four different classification scenarios were simulated and corresponding performance was tested (**Figure 3.1, A**).

The four scenarios were –

1) unbalanced dataset without feature selection,

2) unbalanced dataset with feature selection,

3) balanced dataset without feature selection and

4) balanced dataset with feature selection.

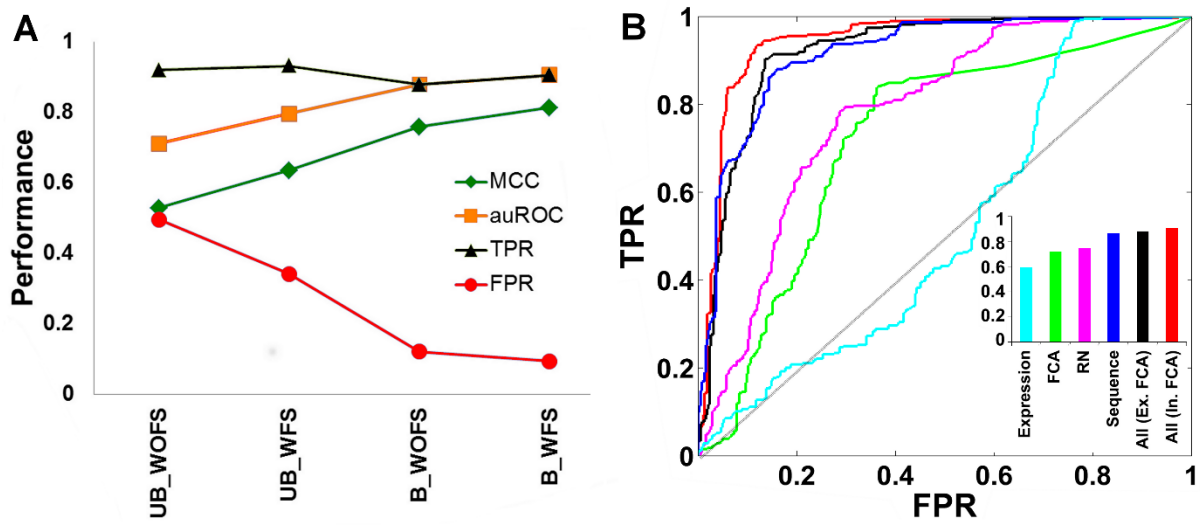


Figure 3.1. Use of balanced training sets and contribution of features. (A) Effect of balancing input training dataset and feature selection. Abbreviation: UB: Unbalanced Dataset, B: Balanced Dataset, WOFS: Without Feature Selection, WFS: With Feature Selection **(B)** Receiver Operating Characteristic (ROC) curves for essential genes prediction using each defined feature set. Inset indicates the auROC values while using each individual feature set.

For comparing these scenarios, MCC, auROC, TPR and FPR were computed. It can be observed that training with the balanced dataset improves the performance of the model, indicated by sharply increasing MCC and auROC values. The TPR seems to remain unchanged whereas FPR drastically reduces after providing balanced dataset to the model. Further, feature selection marginally improves model performance as indicated by increasing MCC, auROC and slightly decreasing FPR. In all the above-

mentioned scenarios, TPR seems to be relatively unchanged, suggesting that the model is able to predict essentiality of genes that are actually known to be essential.

3.2.5. Contribution of ''selected'' features to model performance

In this study, 64 features were considered with respect to their biological relevance (**Table 2.1, Chapter 2**) in classifying essential genes within *Escherichia coli*. To analyze the contribution of each type of feature towards essentiality, four different feature sets based on the type of feature were created namely, i) Genome/Proteome sequence based features ii) Gene expression based features iii) Network topological features in the *Escherichia coli* reaction network (RN) iv) Network topological features in the *Escherichia coli* flux-coupled subnetwork (**Figure 3.1 B, Table 3.2**); each set simulated separately for classification. For these analyses, feature selection on each of the mentioned feature sets while training with the 1000 randomized datasets was performed. The combination of the training dataset (reaction-gene pairs), selected features in each feature set (obtained after running SVM-RFE) and the optimized complexity parameter that gives highest performance was chosen for each feature set and the model classification performance is reported in **Table 3.2**. If only expression features are used for training the model, the model performs poorly with prediction of large number of false positives (FPR = 0.406) but predicts essential genes with a high precision and recall. Training with only sequence features can predict essential genes with high precision and accuracy, and performs best among all individual feature sets (auROC = 0.862). Even while comparing with other methods that use only sequence composition features in *Escherichia coli* (auROC = 0.82 for 5-fold cross-validation) [48], our best model when trained with sequence features shortlisted by feature selection, gives a higher area under the ROC curve (auROC = 0.862 for 10-fold cross-validation). To predict essentiality of genes that are actually known to be essential, the RN feature set performed relatively better than the gene expression subset with a sensitivity of 74%. The novel FCA features, used in this work, also perform comparably to the reaction network features but provide a sub-optimal

performance when given individually. Model performance with respect to both sensitivity and specificity (1 - FPR) improves significantly when all features excluding FCA are given for our training strategy. Including FCA, model performance increases even further. These results indicate the use of unique, non-redundant, heterogeneous features for obtaining a high classification performance.

Table 3.2: Effect of each feature type in model classification performance

Performance metric	Expression (2)	Sequence (14)	RN (5)	FCA (8)	All excluding FCA (22)	All including FCA (26)
TPR	0.594	0.862	0.747	0.717	0.88	0.906
FPR	0.406	0.138	0.253	0.283	0.12	0.094
Precision	0.657	0.862	0.75	0.722	0.881	0.907
Recall	0.594	0.862	0.747	0.717	0.88	0.906
F-Measure	0.548	0.862	0.747	0.716	0.88	0.906
MCC	0.243	0.724	0.497	0.439	0.761	0.814
auROC	0.594	0.862	0.747	0.717	0.88	0.906

The digits in the parentheses indicate the number of features selected from total number of features in each set

After applying SVM-RFE feature selection technique to the whole feature set, a subset of 26 features was obtained (**Figure 3.2**). These features represent the organism-specific determinants of essential genes in *Escherichia coli* K-12 MG1655 metabolism. Additionally, the medians of each feature between the 384 essential and 3120 non-essential reaction-gene pairs were compared using Wilcoxon rank-sum test (*P*-values are indicated in (**Table A. 1, Annexure A**). Among the 26 best features, 21 features were significantly different ($P < 0.05$) for the essential and non-essential reaction-gene pairs. The differences in distributions of feature values between the two classes are represented in **Figure 3.2**. To test this, a correlation analysis was performed between each pair of features among the selected 26 features. The number of feature pairs having weak correlations with each other (Spearman correlation: $\rho < 0.4$ and $\rho > -0.4$, $P < 0.05$) was found. Around 71.69% of the selected feature pairs were weakly

correlated [57] with each other, thereby, indicating the choice of diverse, independent biological features for appropriate classification of gene essentiality.

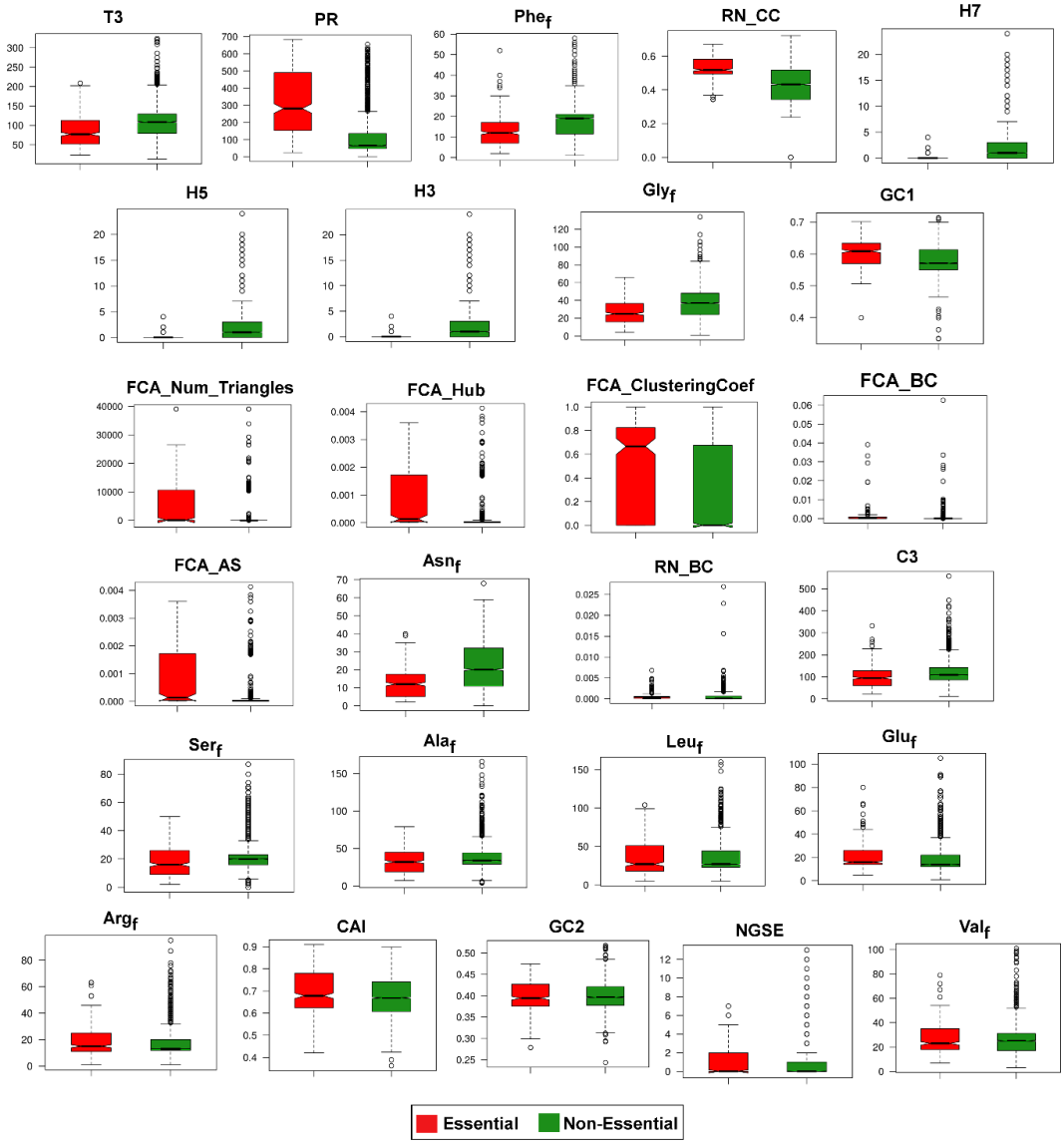


Figure 3.2. Comparisons of distributions of the 26 selected features between the two classes. Each notched box plot represents comparison between the essential (E) and non-essential (N) class based on a single feature. Outliers are indicated by circles in each plot.

3.2.6. Performance of the model and effect of the input balanced training set

In the first scenario where, best model was used for testing with whole unbalanced dataset, 362 out of 384 essential reaction-gene pairs (94.28%) and 2577 out of 3120 non-essential reaction-gene pairs (82.59%) were accurately classified. In the second scenario where best-chosen model (BFC_{best}, C_{best}) was used to train on the 1000 randomized training sets, the percentage of the total 1000 trained models that classify

each reaction-gene pair as essential or non-essential was calculated. If at least 80% of models predict a reaction-gene pair to belong to a particular class, the gene is assumed to be a member of that class. For example, PHETRS_b1713 was predicted to be essential by 80.8% models and 19.2% as non-essential. Accordingly, this gene can be classified as essential. With respect to the threshold of 80% models for assigning essentiality, 356 out of 384 essential reaction-gene pairs (92.7%) and 2485 out of 3120 (79.64%) non-essential pairs were assigned appropriate essentiality phenotypes. Both the testing scenarios were also used for predicting the essentiality of the 590 reaction-gene pairs experimentally unidentified by Baba *et al*. [15].

3.2.7. Model performance for other less-studied organisms

To test the proposed methodology (ML Strategy 1) for organisms in which less or no organism-specific essentiality-based classification models are available, published datasets from two different prokaryotes namely, *Brevundimonas subvibriodes* ATCC 15264 and *Helicobacter pylori* 26695 was further used for training our model. The best trained model after feature selection and 10-fold cross-validation as given in the proposed methodology **(**ML Strategy 1**)**, displayed a high sensitivity and a low FPR (**Table 3.3**). The MCC values are above 0.5 in both the cases demonstrating a comparable accuracy across organisms. The auROC values are lower as compared to the *Escherichia coli* trained best model. The possible reason for this could be that not all features were available for the chosen less-studied organisms in comparison to *Escherichia coli*.

Table 3.3: Model evaluation metrics for two less-studied prokaryotes

Performance metrics	*Brevundimonas subvibriodes* ATCC 15264	*Helicobacter pylori* 26695
TPR	0.783	0.753
FPR	0.217	0.247
Precision	0.784	0.758
Recall	0.783	0.753
F-Measure	0.783	0.752
MCC	0.567	0.512
auROC	0.783	0.752

In case of *Brevundimonas subvibrioides* ATCC 15264, 102 out of 136 essential and 368 out of 486 non-essential genes were correctly classified. In case of *Helicobacter pylori* 26695, 60 out of 73 essential and 174 out of 329 non-essential reaction-gene pairs were correctly classified. It can be observed that the best trained model in both the organisms is highly sensitive in predicting true essential genes. Apart from the known essential and non-essential genes, essentiality of 32 genes in *Brevundimonas subvibriodes* ATCC 15264 and 3 genes in *Helicobacter pylori* 26695, for which essentiality information was not reported in DEG database [137] was predicted newly [57].

3.2.8. Comparison with other available methods – Proof of training set independence

Apart from Hwang *et al.*, 2009, [52] our strategy was also compared with other recent supervised classification studies on essential gene identification [75,170]. To compare the performance of our strategy with these classification methods, training (*Escherichia coli* genes) and test dataset (*Bacillus subtilis* genes) considered in these studies were provided to our methodology for generating a best SVM model and for further testing, respectively. The best model generated from our methodology using the previously available training dataset (consisting of sequence features of *Escherichia coli* genes) was further tested with test dataset (sequence-based features of *B. subtilis* genes) of the available methods.

Table 3.4: Comparison of our proposed strategy (ML Strategy 1) with methods proposed by Song *et al.* 2014 and Deng *et al.* 2011

Performance metric	Our Method (ML Strategy 1)	Song *et al.* 2014 [170]	Deng *et al.* 2011 [75]
auROC	0.966	0.930	0.800
Precision	0.970	0.730	0.540

Testing results in the form of auROC and precision, indicate that the best model generated through our strategy outperforms both the available supervised classification methods (**Table 3.4**). The achieved sensitivity and specificity from our

methodology is the highest suggesting an enhanced model performance, irrespective of the given input training dataset.

3.3. Discussion

As introduced above, the choice of an appropriate training dataset, a flexible and accurate learning algorithm and biologically relevant features that define the essentiality of a gene is absolutely required for accurate essentiality-based classification of genes using supervised machine learning techniques. With respect to these primary requirements, here, we present a simple but comprehensive computational strategy that integrates genotype-phenotype characteristics of *Escherichia coli* K-12 MG1655 metabolism to classify a gene and its corresponding reaction based on their dispensability. Also, attributing to the universal set of features curated for classification, our method can predict minimally essential genes. Here, genotype characteristics correspond to the features calculated from the nucleotide and amino acid sequence that represents the static genome and proteome complement for prediction. Likewise, gene expression, metabolic network, and flux-based features represent the metabolic phenotype complement of the organism that results from interactions of genotype with the environment.

As our method (ML Strategy 1) is a supervised machine-learning strategy, we showcase the predictive capability by comparing it with the previously available supervised classification strategies for essential genes classification [52,75,170]. The comparisons indicate that irrespective of the input training dataset used, our model classification always outperforms all other methods and achieves the highest sensitivity and specificity. Also, our analysis indicates that the SVM model performance is highly dependent on the balance of the input training dataset. It also emphasizes the need to perform model training with many randomly sampled balanced datasets to remove sampling bias and noise, which helps to choose the best training set sample of instances that represent the whole population of genes within that organism. Hence, our strategy ensured that a large number of sample balanced

training sets (1000 sets) were generated such that a particular gene is sampled at least once. Also, the training set of instances in our model represents the reaction-gene combinations. Thus, instead of predicting only essential genes, our model can also predict the metabolic reaction that gene is associated with, for which it was predicted to be essential.

Training the model with specific feature sets indicate that the sequence-based features are the most predictive of gene essentiality in *Escherichia coli* K-12 MG1655 metabolism. Out of the 26 selected features (**Figure 3.2**), homology based features like phyletic retention [17] and number of homologs in other organisms contribute the highest to classify genes in *Escherichia coli* K-12 MG1655 metabolism based on their essentiality, as ranked by SVM-RFE. This indicates that gene essentiality is largely linked to the evolutionary preference among homologous bacterial species. This is followed by CAI, GC content and NGSE that are also significantly high for essential genes. Essential genes have been previously shown to demonstrate a high expression rate as compared to non-essential genes across bacteria [172]. CAI is a predictor of protein abundance and NGSE indicates that a gene sharing a common pattern of expression with large number of other genes tends to be more essential. Median frequencies of glycine, asparagine and phenylalanine are typically low whereas, frequencies of arginine, glutamate and valine are typically high in essential enzymes; an observation supported by a previous study where similar correlations in the *Escherichia coli* K-12 genome was observed for essential enzymes [168].

The definition of a metabolic gene to be essential is highly dependent on the environmental context of the cell. To circumvent this problem, for the first time, flux coupling analysis was performed on the iJO1366 network obtained to obtain a flux-coupled sub-graph that is universal across environments. The topological features of the identified sub-network were used as features in model training. Model training with only FCA-based network features gave a competitive model performance. Further, in conjunction with sequence-based, expression-based and reaction

network-based features, FCA features significantly improved model sensitivity and specificity.

Our results also display that essential reaction-gene pairs exhibit high hub, authority scores, clustering coefficient and betweenness centrality within a physiological flux-coupled sub-network. These features indicate that a large number of reactions demonstrate a metabolic flux dependence on essential reactions. Such reactions also have a tendency to form flux-coupled modules so that metabolites generated or consumed from this reaction once distributed into other reactions within the same module can be regained while maintaining energy or redox balance. Essential reactions also act as connecting links between physiologically important flux-modules. These reactions are probably metabolically important as they catalyze the conversion of highly connected metabolites, like, ATP, H_2O, H^+ and other cofactors, which are used in their coupled reactions. Also, some of these reactions (genes) might also represent a first committed step of conversion of the terminal metabolite of one physiological module as a substrate to enter the other module, for example, acetyl co-A carboxylase, which is both the first committed and rate-limiting step of fatty acid/lipid biosynthesis [173], demonstrates a high betweenness centrality in the obtained physiological flux-coupled subnetwork. Although, it still remains to be established whether there are any relationships between homology-based, amino acid frequency based and flux-coupled subnetwork features; once established, the essentiality predictions from our pipeline can be extrapolated to reveal the evolutionary constraints faced by essential genes as compared to the non-essentials.

Finally, the model was also tested for the experimentally known, complete, gold standard dataset of essential and non-essential genes in the *Escherichia coli* Keio collection [15] for which the essentiality information of each gene was available. With respect to the test set, best model could predict 362 out of 384 known essential reaction-gene pairs (94.28%) and 2577 out of 3120 non-essential reaction-gene pairs (82.59%) which is indicative of the sensitivity of the model to detect true essential and

non-essential reaction-gene pairs. Further, the essentiality of the 590 experimentally unknown reaction-gene pairs as per previous study of Baba *et al.* was also predicted using our strategy. Yamamoto *et al.* [174] provided an update for the *Escherichia coli* K-12 MG1655 Keio collection where essentiality of few of these unknown reaction-gene pairs was experimentally determined. Their new update improves the annotation of five metabolic genes, namely, alanyl-tRNA synthetase (b2697), pantothenate kinase (b3974), dephospho-coA kinase (b0103), isoleucyl tRNA synthetase (b0026), and phosphoglucosamine mutase (b3176) which was previously unknown, as essential. Our best model was precisely able to predict all the corresponding reaction-gene pairs associated with the mentioned genes to be essential. Considering the best model trained with 1000 balanced datasets, 98-100% of the generated models predicted the above genes to be essential. Further, our strategy was also able to predict essentiality of 317 genes (out of which essentiality of 235 genes can be predicted by FCA-based network features alone) which could not be determined by Baba *et al.* [15] or Yamamoto *et al.* [174] Role of few of these genes have also been discussed elsewhere although in different biological contexts. Few mentionable examples that were predicted to be essential from our machine learning framework include ubiE (gene: b3833, reaction: AMMQLT8) enzyme, which catalyzes the carbon methylation reaction in biosynthesis of ubiquinone and menaquinone, that are essential within the respiratory chain, [175] the iron-sulfur cluster YtfE (gene: b4209, reaction: FESR) necessary for repairing damaged iron-sulfur clusters under oxidative or nitrosative stress conditions, [176] β-ketoacyl carrier protein synthase III fabH (gene: b1091, reaction: KAS15) that catalyzes the condensation reaction in the initiation of type II fatty acid synthesis in bacteria, [177] ribosome small subunit-dependent GTPase rsgA (gene: b4161, reaction: NTP3) that is required for ribosomal subunit assembly and 16S-rRNA processing, [178] and triose-phosphate isomerase tpiA (gene: b3919, reaction: TPI), which plays a crucial role in isomerization of triose-phosphate isomers within glycolysis [179]. These examples indicate the application of our designed methodology, given any random

set of training instances, to predict novel essential gene candidates and thereby provide experimentally testable hypotheses for further validation. The functional reasons for these genes to be essential can be further probed using the features selected from our model strategy. Our methodology also provides comparable results in less-studied organisms, like *Brevundimonas subvibrioides* ATCC 15264 and *Helicobacter pylori* 26695, for which less or no organism-specific machine learning studies were previously available.

All the above results indicate the strength of our model (ML Strategy 1) in identifying true essential genes asserting the usage of a balanced training dataset, selection of biologically relevant features to represent gene essentiality and optimal parameters for hyperplane formation to classify essential genes while using a fundamental machine learning based scheme. Further, given the challenges faced by experimental biologists to identify essential genes, the novel putative essential genes and their associated features can provide fresh impetus for further targeted studies in *Escherichia coli* and other related organisms.

Chapter 4

Essential genes prediction using ML strategy 2 (Semi-supervised) for organisms when limited gene essentiality information is available

4.1. Motivation

The major drawback of existing machine learning algorithms for essential genes prediction such as support vector machine [52–54,77], random forest [78], decision tree [75], etc., (**Table 1.2, Chapter 1**) , require a large amount of labeled data that helps to train the models for an accurate prediction of the essentiality of unannotated genes. They show very poor performance when the labeled dataset is imbalanced or limited (1% labeled data). To circumvent these problems, in our previous study (ML Strategy 1), an integrative machine learning strategy (**Section 2.2. in Chapter 2**) was developed using a combination of feature selection algorithm, Support Vector Machine- Recursive Feature Elimination (SVM-RFE) [133] and Sequential Minimal Optimization (SMO) classifier [136] for gene essentiality prediction in the metabolism of *Escherichia coli*, which performed well on imbalanced dataset with diverse features computed from flux coupled connected sub-network along with other sequence-based features [57]. Advantages of using the Flux Coupling Analysis (FCA) based feature for the prediction of gene essentiality with high accuracy and confidence have been reported. FCA analysis help to capture the physiological dependence of one coupled gene-reaction combination to other, under all input exchanges of a reaction, representing all possible environmental conditions, thereby helping the classifier to accurately identify the minimally essential genes that are absolutely crucial for sustaining the metabolic demands of the cell to ensure its survival [57]. However, this technique (ML Strategy 1) was unable to predict gene

essentiality when a very small amount (labeled data ≤ 1%) of experimentally verified labeled data are available.

To mitigate these problems inherent in the existing strategies, we have proposed in this chapter an integrative semi-supervised machine learning strategy (ML Strategy 2) based on Laplacian SVM [74] for the classification of genes using gene sequence, protein sequence, network topological, and flux-based features with very limited labeled data on gene essentiality of metabolic networks for both Prokaryotic and Eukaryotic organisms (**Section 2.3., Chapter 2**). Another objective of this work is, applying ML Strategy 2 for the annotation of essential genes of less explored organisms, like *Leishmania donovani* and *Leishmania major*, the causative organisms for the neglected tropical disease Leishmaniasis, for which very limited experimental data is available. By using the available tools and techniques, the prediction of gene essentiality and targeted therapy for the disease becomes extremely difficult [180]. In the present work, it is hypothesized that using these diverse features, like topological network features of both the genome-scale metabolic reaction network as well as the flux-coupled sub-networks, together with the sequence-based features simultaneously, can capture both the properties of genotype and phenotype and by employing the proposed ML Strategy 2, it is possible to predict the essentiality of uncharacterized genes with high accuracy even in the cases where labeled data is limited. This is in contrast to other machine learning pipelines for essential genes prediction that rely only on sequence-based features and has been applied to only Prokaryotes [75,181]. In this work, we also proposed a new scoring technique (**Section 2.3.2.4. in Chapter 2**), called the Semi-Supervised Model Selection Score (SSMSS) that correlates well with Mathews Correlation Coefficient (MCC) [182] and can be used for the selection of the best model when the calculation of supervised performance metrics like MCC or auROC is difficult due to lack of experimental data. We have validated this proposed pipeline (ML Strategy 2) on twelve organisms (**Table 2.2, Chapter 2**), using as low as 1% labeled data on two types of training

datasets (*i.e.*, with 80% training and 20% blind datasets, as well as using the whole dataset for training) with well-annotated gene essentiality information. Correspondingly the same strategy was applied to predict the essential genes in *Leishmania* as well as categorize the reaction-gene pairs in five different groups-based Gene-Protein-Reaction (GPR) association in metabolism. These groups depict the association of the reactions with different combinations of essential and non-essential genes, which throws light on the probable reaction-gene combination that can be used for targeted therapy. This study promises to lay the foundation for the prediction of gene essentiality information in less explored organisms that will help experimental biologists to identify novel therapeutic targets even when only limited information is available.

4.2. Results

4.2.1. Model Validation with experimental data

The integrative proposed ML Strategy 2 (**Figure 2.2, Chapter 2**) was applied and validated on twelve organisms (**Table 2.2, Chapter 2**) with well-annotated gene essentiality information from experimental data obtained from the OGEE database [33].

4.2.2. Features frequently selected by the feature selection algorithm

The important features chosen by the feature selection algorithm (**Section 2.3.2.1. in Chapter 2**) have been represented in the heat map (**Figure 4.1**), where X-axis represents the name of the 82 features that have been selected at least once by the features selection algorithm and Y-axis corresponds to names of the organism. Red cell color indicates features selected by the feature selection algorithm in the corresponding organism. White-colored cell shows the feature that is not selected or is redundant. Among 289 features (**Table 2.2, Chapter 2**), three features, *viz.*, Reaction Network betweenness centrality (RN_betweenness), Reaction Network Page Rank centrality (RN_page_rank), and Flux Coupled Analysis Network Page

Rank centrality (FCA_page_rank) are selected by the features selection algorithm for every organism. These frequently selected features are topological network features. Apart from these features, Information-theoretic features (Fourier sine or cosine coefficient, Mutual Information, Conditional Mutual Information) from nucleotide and peptide sequences are also selected. If a node is important in the reaction network and flux-coupled network, then there is a chance that the enzyme or protein which controls that particular reaction and its corresponding coding sequence is also essential.

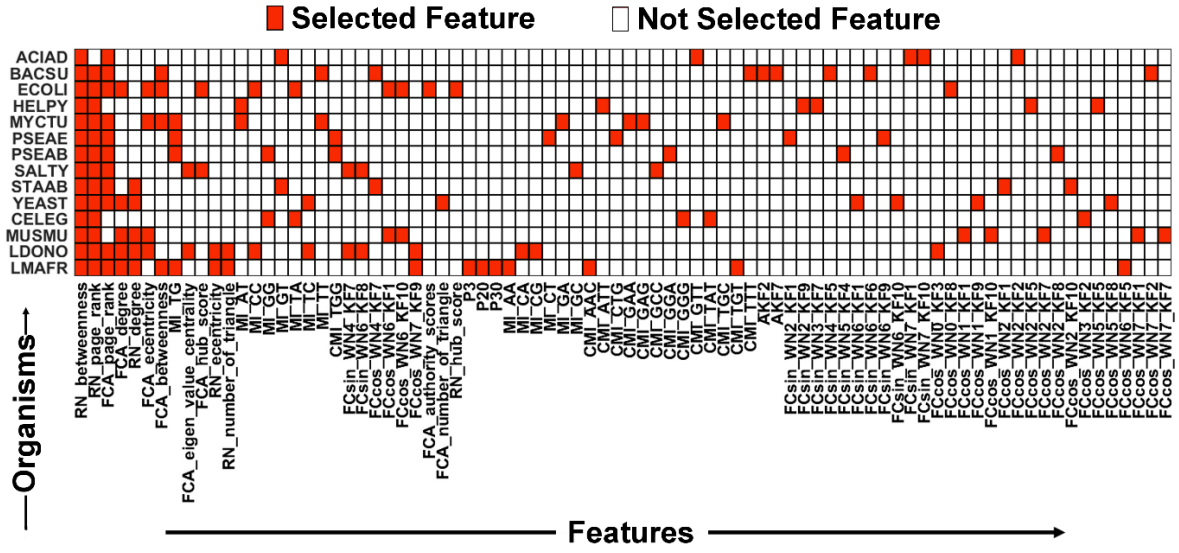


Figure 4.1. Heatmap plot of selected features by the feature selection algorithm. Red cells indicate features selected by the feature selection algorithm in the corresponding organism. White cells show the feature that is not selected or is redundant.

4.2.3. Dimension Reduction

After applying feature selection, we have used the Kamada-Kawai dimension reduction technique [155] for visualization purposes (**Section 2.3.2.2. in Chapter 2**). Here, we observed a circular layout of each organism. While the essential gene-reaction combinations are clustered together in one side of the arc in a 2-D circular layout, the non-essential reaction-gene combinations are clustered in the rest of the circle. Now on applying Laplacian SVM, the classifier was able to easily classify gene essentiality based on their transformed 2-D feature and the limited label information. Now in different parameter combinations of Laplacian SVM, different trained models

are obtained. We have used the proposed SSMSS score to select the best model among trained models.

4.2.4. Robustness of the proposed score (SSMSS)

To check the robustness of the SSMSS score (**Section 2.3.2.4., Chapter 2**), the proposed ML Strategy 2 has been applied on both types of training dataset (*i.e.*, dataset with 80-20% combination of samples and with the whole dataset) for these twelve organisms. Using this 80% data points of the whole dataset, different types of training dataset is further created with limited labeled data points in the range, *i.e.*, i % Labeled (L) and (100 - i%) Unlabeled (UL) data, where i = 1, 2, 3, 4, 5, 10, 30, 50, 70 and 90. In each category, labeled samples were chosen randomly from the master table. It is to be mentioned here that this selection of labeled data was conditionally randomized to ensure that both the essential and non-essential genes categories appear with equal probability. In this way, 100 datasets in each labeled category have been created. For the testing purpose, both the whole training dataset and the 20% blind dataset have been used for prediction. The parameters (σ, λ, γ) were tuned with four different values *i.e.*, 0.01,0.1,1,10. Therefore, by tuning these model parameters using grid search generated 64 models for each dataset have been created. After that, we compared the prediction results with the known gene essentiality information, which is publicly available from the experiment. Six supervised performance metrics have been calculated for the predicted class label with the known class label. After that, the association between the proposed score and auROC was assessed. To verify the linear relationship between auROC and the proposed score (SSMSS), the Pearson correlation coefficient has been calculated, and scatter plots were generated in different limited labeled datasets in each target organism (**Figure 4.2**).

From the scatter plot (**Figure 4.2**), we have observed that in all the cases, Pearson correlation >0.75. Hence, it may be inferred that due to the linear relationship existing between auROC and the proposed score (SSMSS), the applicability of this scoring

technique is asserted and can be used for the calculation of the performance measurement matrix and best model selection for the semi-supervised based classifier.

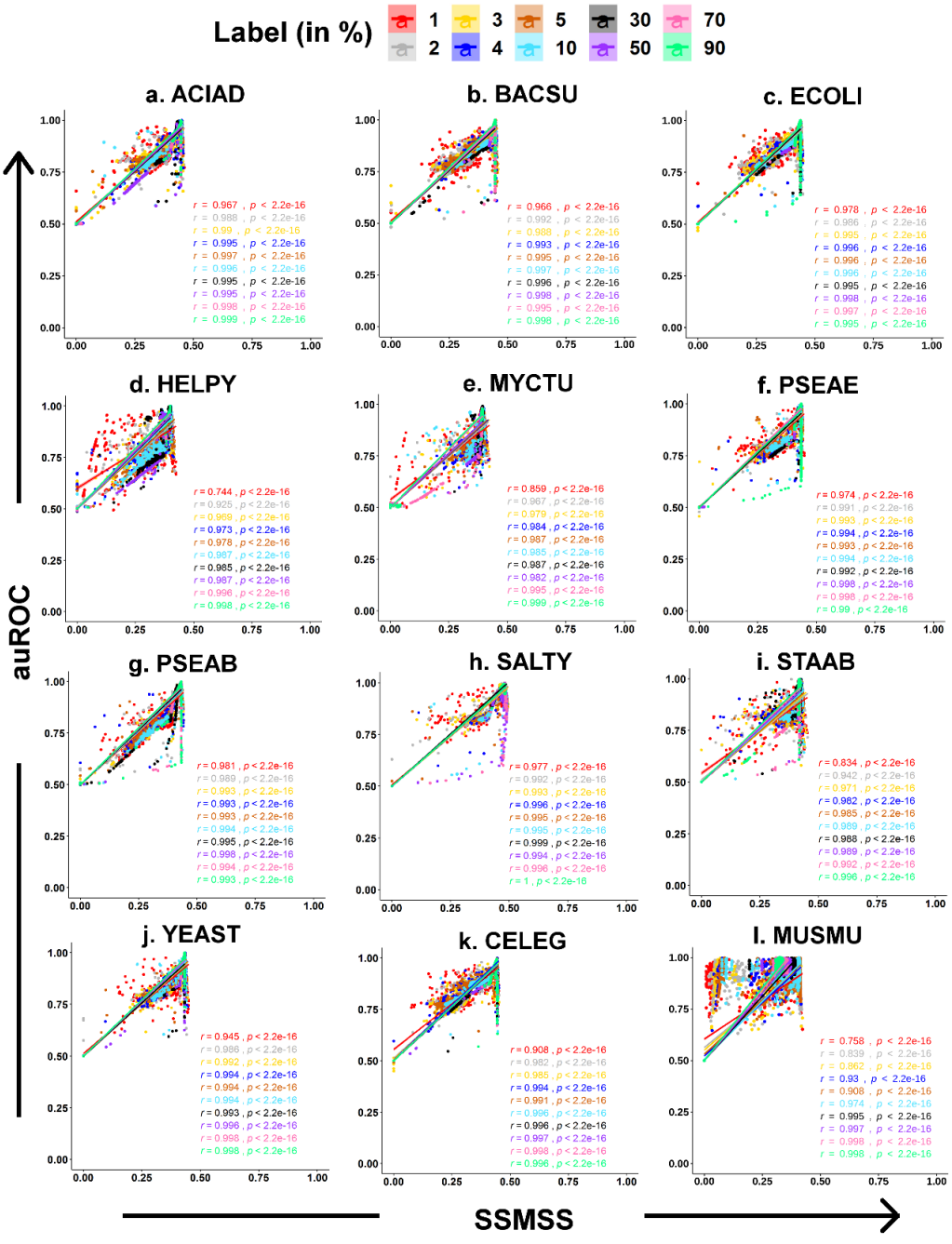


Figure 4.2. Robustness Evaluation of the proposed score (SSMSS). Scatter plots is demonstrating an association between auROC and SSMSS in each labeled category datasets in different model parameters conditions for twelve organisms. The X-axis represents the score (SSMSS), and Y-axis represents the corresponding auROC. To represent each category, ten different colors are used.

4.2.5. Predictive performance of the best models in the different labeled category on training and blind test dataset

In a real-life scenario, only limited gene essentiality information is available for the less explored organisms. However, model building from this limited label data and determining how the highest score will select the best model is difficult. Hence, to test the model performance on known organisms by creating limited labeled datasets (*i.e.*, by varying the limited labeled data from 1% to 90% from the 80% training dataset), six supervised performance metrics have been calculated for each category under different parameter combinations of σ, λ, γ (See **Section 2.3.2.4.** in **Chapter 2** : The score for best model selection for a detailed description of these parameters). Here, within each labeled category, the average behavior of the predictive performance six supervised performance metrics, **Section 1.6.3.5.** in **Chapter 1** and the Score (SSMSS) of the best 100 trained models are plotted in **Figure 4.3**. This has been shown for two different conditions, training dataset (80% of the whole data) and blind testing dataset (20% of the whole data). As observed from the low standard deviations for each metrics (under each category), it is worth to mention that the accuracy for the training and testing are very similar in most of the cases. From these plots, it has been observed that the model selection based on the SSMSS score in each category corresponds to a high auROC value of greater than 0.8 in all cases across all organisms. Also, it is observed that if the label increases, then model performance will also show higher accuracy. However, it is seen that the auROC score remains consistently high, using 1% labeled data or more, which establishes the fact that the proposed method can predict using a minimum of 1% labeled data. It has also been observed that this method is giving a consistent better predictive performance on both Prokaryotic and Eukaryotic organisms for both the datasets (80% training and 20% blind testing) and follow similar patterns for six supervised performance metrics in differently labeled categories. As the predictive performance of 20%, the blind

dataset is similar to training performance, so further, it can be concluded that model overfitting and underfitting is not arising in this case.

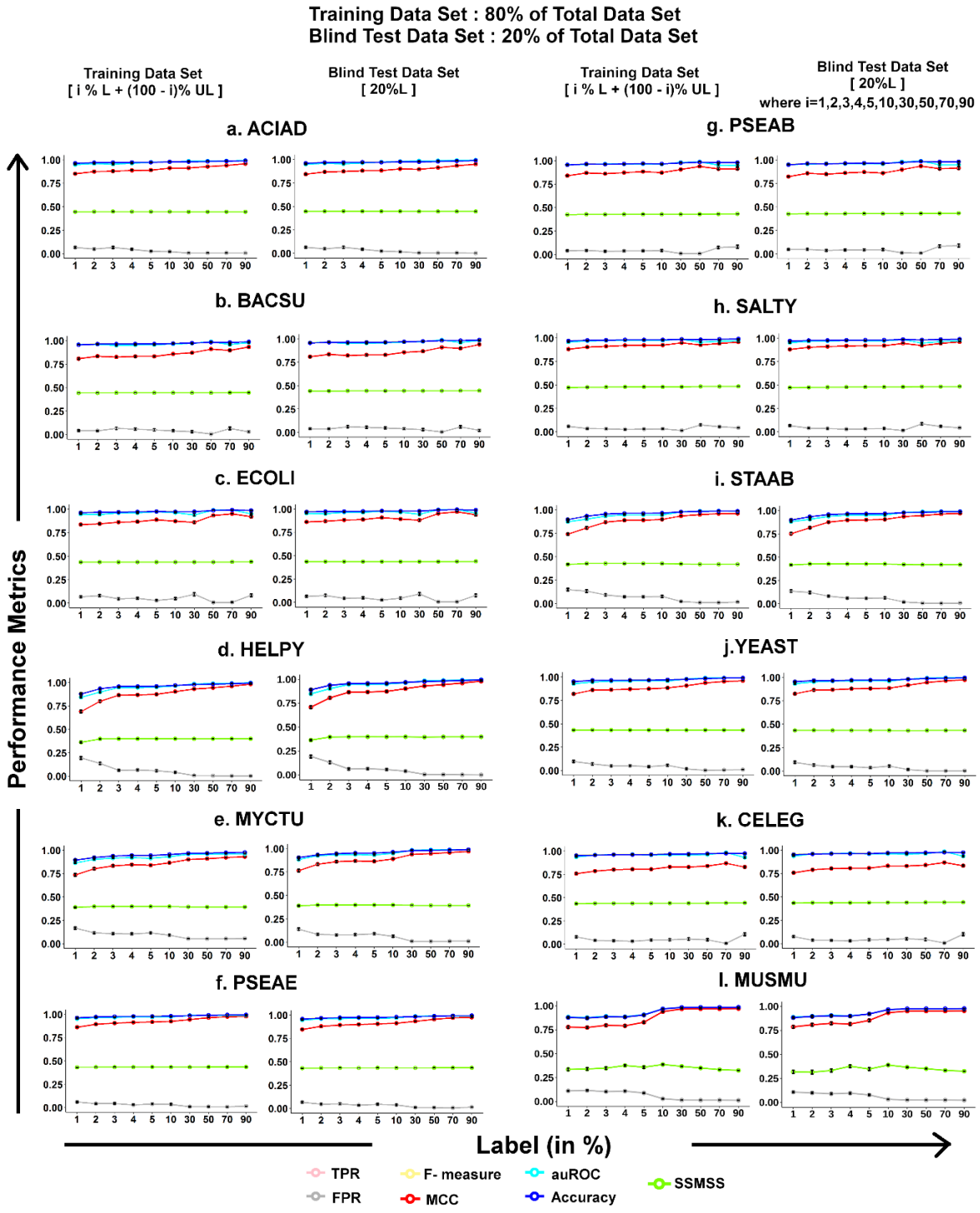


Figure 4.3. Comparison of the Predictive performance of the best models in the different labeled category. The average performance of the best 100 models at training and blind testing for six supervised metrics (*i.e.*, TPR, FPR, F-measure, MCC, auROC, accuracy) and SSMSS for each labeled type. The X-axis represents the category of labeled data, the Y-axis represents the value of performance metrics.

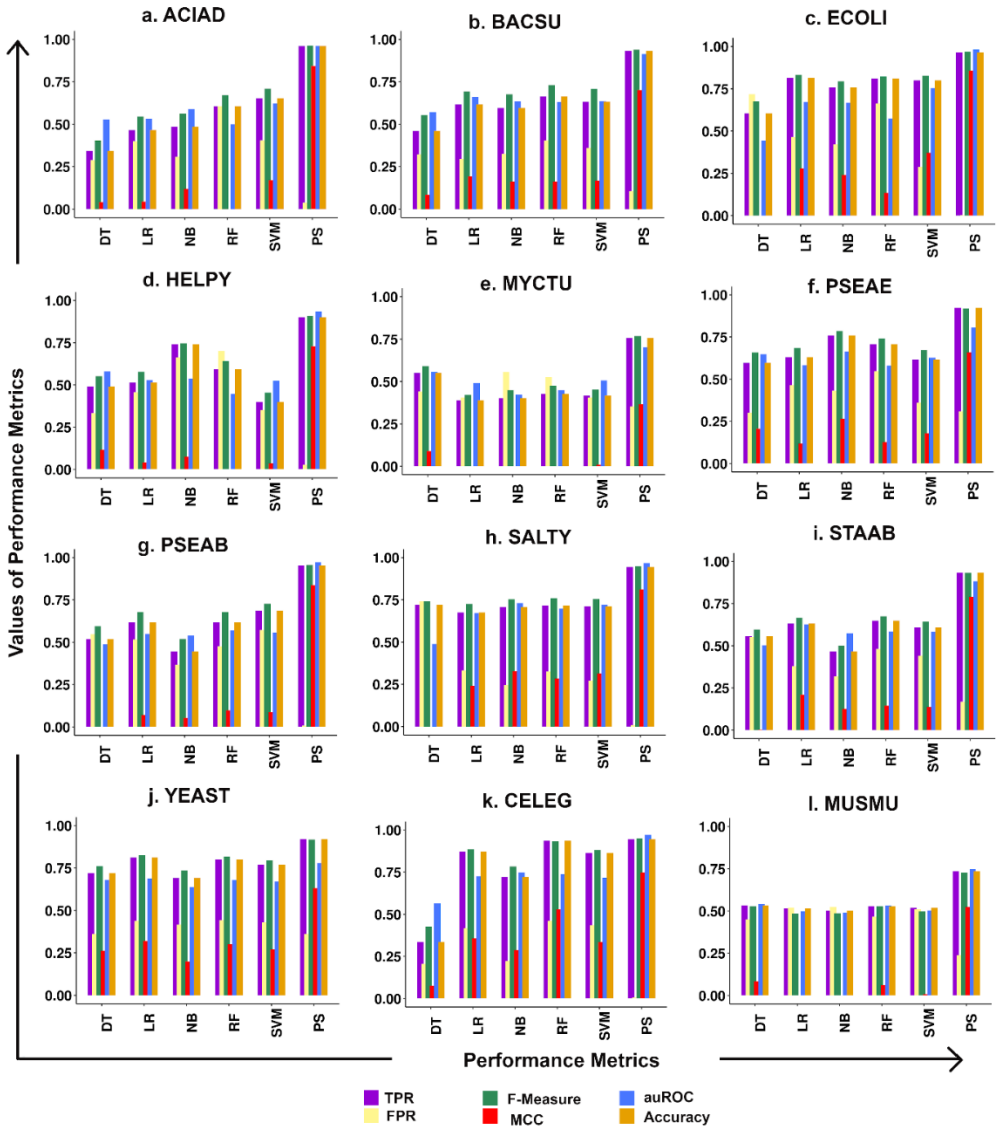


Figure 4.4. Comparison of the predictive performance of the proposed ML Strategy 2 with other supervised methods. Comparison of the performance of proposed strategy (PS) with supervised classifiers [*i.e.*, Decision Tree (DT), Logistic regression (LR), Naive Bayes (NB), Random Forest (RF) and our own previously reported Supervised essential genes prediction pipeline] based on 1% labeled data on twelve organisms. The X-axis represents the different types of performance metrics for machine learning strategies, the Y-axis represents the value of performance metrics. Six different color codes were used to represent six different performance metrics.

To compare the predictive performance of the proposed method (ML Strategy 2), 1% labeled dataset has been considered for each of the twelve organisms. For training, different supervised classifiers have been used, such as Random Forest [71], Naive Bayes [183], Logistic regression [184], J48(C.45) Decision Tree [185] as well as our own (**Section 2.2., Chapter 2**) previously reported Supervised essential genes prediction

pipeline [57] on the whole dataset for testing (**Figure 4.4**). In all of the cases, it is found that the proposed ML Strategy 2 (**Section 2.3., Chapter 2**) performed better than all other methods using only 1% labeled data of the whole training dataset.

4.2.6. Effect of feature selection and dimension reduction in model performance

To compare the effect of feature selection and dimension reduction steps along with the LapSVM classifier, seven different types of classification scenarios, based on different dimension reduction technique such as PCA, MDS, FR, ICA, and KK, were simulated on training dataset (80% data points) and blind testing (20% data points) datasets of twelve organisms. The corresponding performance was calculated on the blind test dataset (**Figure 4.5**). Each training dataset has only 1% labeled data, and the rest of them Unlabeled.

The seven scenarios were created with LapSVM classifier and combinations of features selection and dimension reduction techniques:

Scenario 1 (S1): Without feature selection and Without dimension reduction technique [WOFS +WODR]

Scenario 2 (S2): Without feature selection and With dimension reduction technique (Principal Component Analysis) [WOFS + DR (PCA)]

Scenario 3 (S3): Without feature selection and With dimension reduction technique (Metric Dimensional Scaling) [WOFS + DR (MDS)]

Scenario 4 (S4): Without feature selection and With dimension reduction technique (Fruchterman Reingold) [WOFS + DR (FR)]

Scenario 5 (S5): Without feature selection and With dimension reduction technique (Independent Component Analysis) [WOFS + DR (ICA)]

Scenario 6 (S6): Without feature selection and With dimension reduction technique (Kamada Kawai) [WOFS + DR (KK)]

Scenario 7 (S7): With feature selection (Unsupervised Feature Selection) and With dimension reduction technique (Kamada Kawai) [WFS (UFS) + DR (KK)]

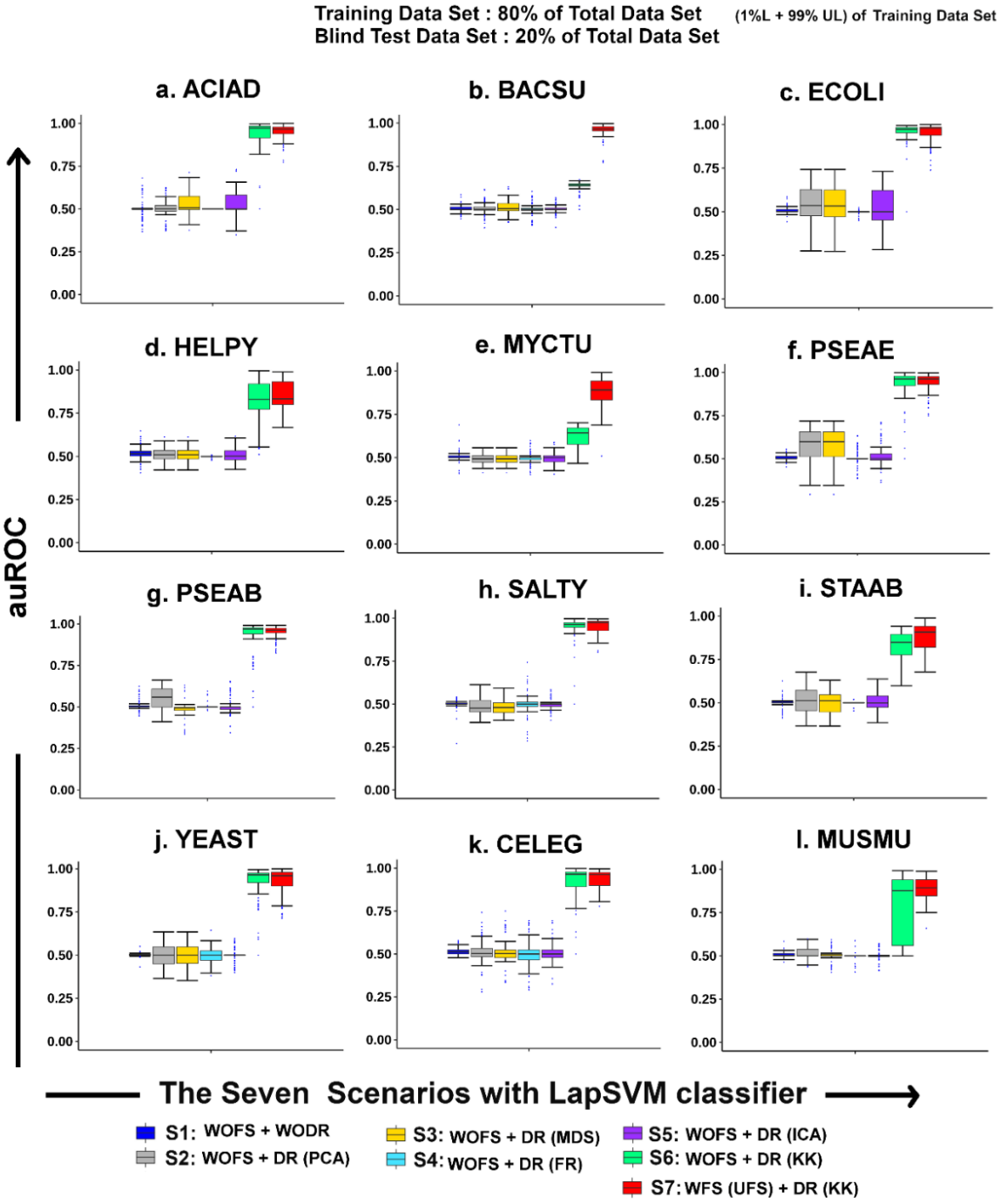


Figure 4.5. Effect of feature selection and dimension reduction on model performance. Comparison of the effect of different dimension reduction techniques PCA, MDS, FR, ICA, and KK (S2 - S6) with S1 (Without Feature Selection and Without Dimension Reduction) and S7 (With Feature Selection and With Dimension Reduction-KK) when combined with LapSVM classifier. Plot represents the auROC value of 100 best models with 1% labeled data across all organisms

From this analysis, it has been observed that for scenarios 1 to 5, the auROC value is very low, which signifies that dimension reduction techniques, *e.g.,* PCA, MDS, FR, ICA, cannot significantly improve the gene essentiality prediction (**Figure 4.5**). On the other hand, for scenarios 6 and 7, it is observed that on applying the Kamada-Kawai method of dimension reduction along with unsupervised feature selection, the model performance (auROC) improves drastically in each target organism. On comparing the efficacy of Kamada-Kawai (KK) with the other dimension reduction methods using the one-tailed Mann-Whitney U Test, a significant improvement in auROC values (P<0.01) for all the twelve organisms was observed (**Table B. 1 in Annexure B**). Scenario 6 highlights the importance of this dimension reduction step, where it is found that even without feature selection, the dimension reduction step [S6: WOFS + DR (KK)] has a huge impact on the results (P<0.01) [**Table B. 2** in **Annexure B**]. However, the feature selection step helped us in identifying the minimal set of features that contribute towards gene essentiality prediction with greater accuracy in all organisms (lower P-values obtained in Scenario 7 with [S7: WFS + DR (KK)]) [**Table B. 2** in **Annexure B**]. Hence, it is observed that the Kamada-Kawai dimension reduction technique, when combined with LapSVM, gives significantly better performance for all twelve organisms even when only 1% labeled data is used (**Figure 4.5**).

4.2.7. Predictive performance using whole training dataset

In model organisms where gene essentiality information is sufficiently available at the genome-scale, blind testing can be applied. However, in less explored organisms where gene essentiality information is very less, a blind test cannot be applied as the reference size is very small. For these cases, the whole dataset with limited labeled data can be used for model training and prediction purposes.

To establish the predictive performance of the proposed ML Strategy 2 (**Section 2.3. in Chapter 2**) on the whole training dataset, 1% labeled data were selected randomly, and the remaining 99% data points were considered unlabeled for the twelve organisms, where the information of gene essentiality in genome-scale was available from the experiments. Now, this whole dataset was trained by the proposed ML Strategy 2. The best model was selected based on the highest score (SSMSS). The same dataset is used for prediction from the best-trained model. The outcome of the proposed ML Strategy 2 can be visualized as three circles (**Figure 4.6**). The first circle represents the circular projection of the whole dataset in 2-D after applying the Kamada Kawai dimension reduction technique with gene essentiality information from the experiment. The second circle shows the training dataset with 1% labeled & 99% Unlabeled data and learning curve of the Laplacian model. The third circle shows the predicted gene essentiality label from the best-trained model. From **Figure 4.6**, it is observed that the proposed model also performed well (as similar circular patterns from experiment and predicted) on the whole training dataset.

The predictive performance on both the datasets (80% and the Whole dataset) has been compared by six supervised performance metrics (*i.e.*, TPR, FPR, F-measure, MCC, auROC, and accuracy) based on actual and predicted labels from the proposed ML Strategy 2. Here it has been observed that the average predictive performance of the 100 trained model with 80% dataset is similar to the performance on the whole dataset (**Figure 4.7**).

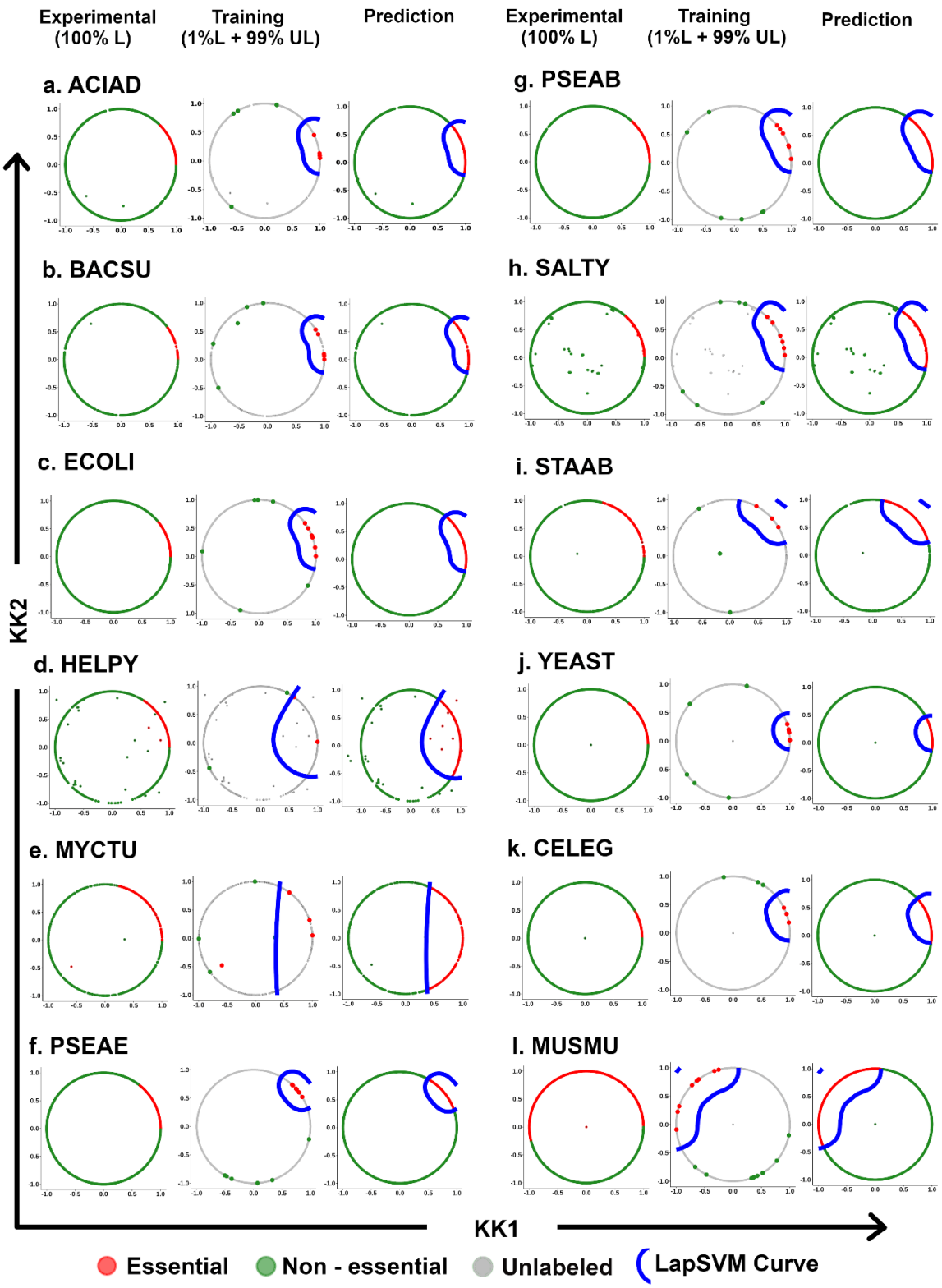


Figure 4.6. Visualization of the outcome of the proposed ML Strategy 2. Essential, non-essential, and Unlabeled reaction gene pairs are colored accordingly Red, Green, and Gray. The learning curve for the best-trained model by LapSVM is colored with blue. The left circle represents the original dataset with labeled data points. The middle circle shows the training dataset with the learning curve, and the Right circle represents the prediction labeled with the learning curve.

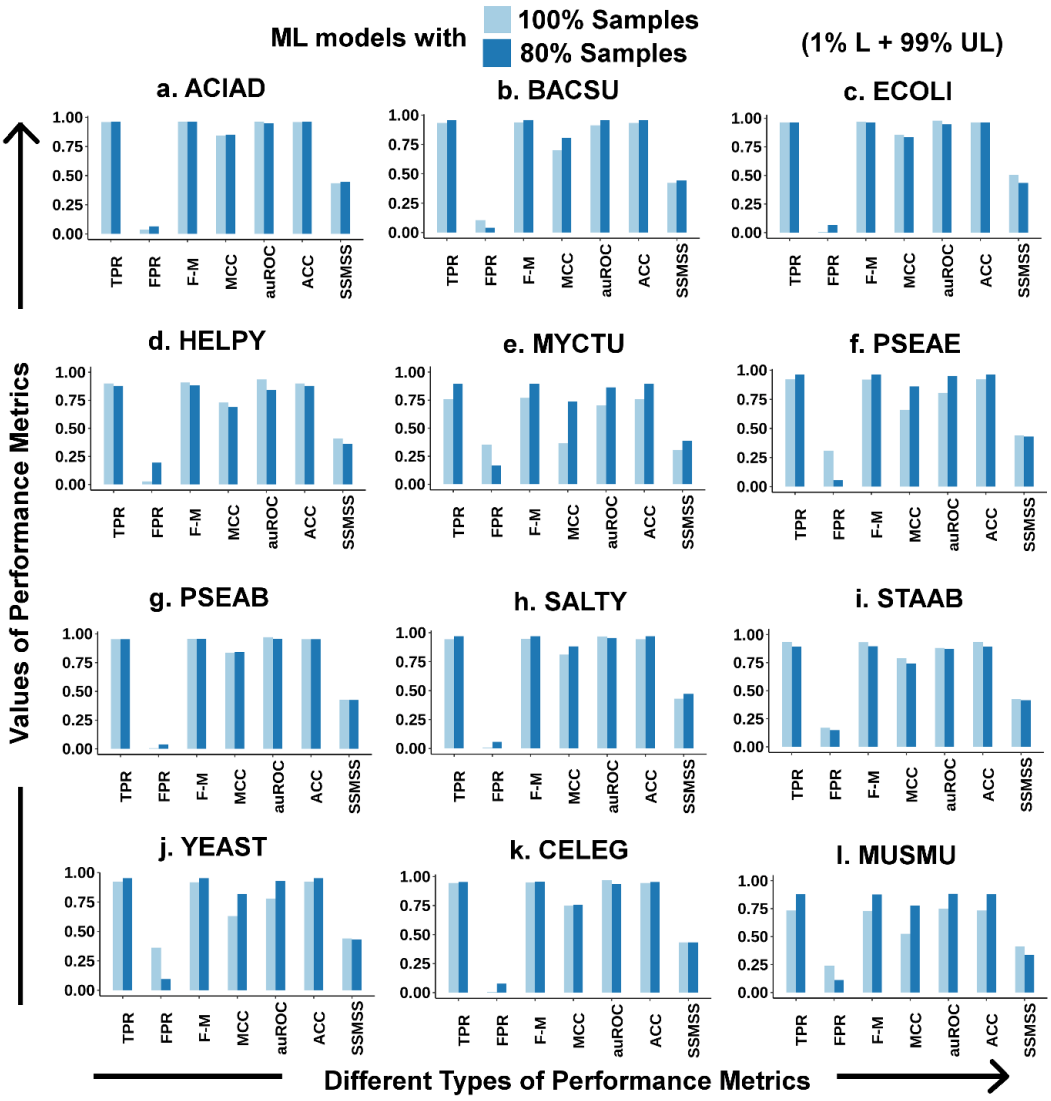


Figure 4.7. Comparison of the predictive performance on both types of datasets (80% and whole dataset). Average predictive performance of the best 100 models on 80% training dataset and performance of whole training dataset containing the Limited Labeled (L=1%) and remaining Unlabeled (UL) data for six supervised metrics (*i.e.*, TPR, FPR, F-measure, MCC, auROC, accuracy) and SSMSS for each labeled type. The X-axis represents the different performance metrics, the Y-axis represents the value of performance metrics.

4.2.8. Categorization of reaction-gene pairs

Categorization of the predicted essentiality information of reaction gene pairs into the five categories, viz. CEN, ME, MN, SE, and SN show that the distribution of reaction of the predicted results matches exactly with the distribution observed with the experimental data for each of the twelve organisms (**Figure 4.8**).

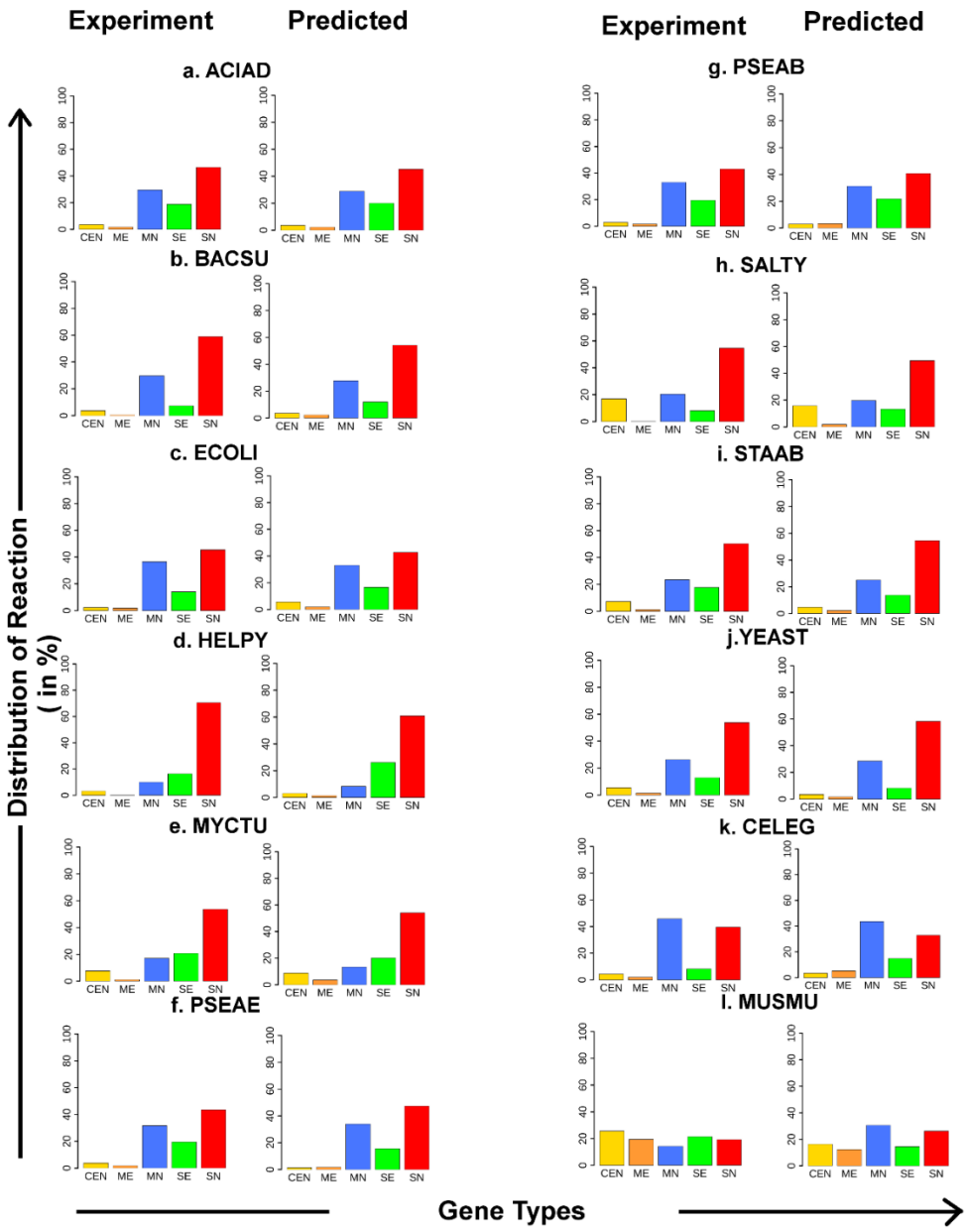


Figure 4.8. Comparison of the distributions of reaction. The reactions have been classified into five categories and the predicted distributions of reaction-gene pairs have been compared with the experimental data across all twelve organisms.

Also, the Chi-square test was performed with a Null Hypothesis (H_0) that the two distributions of reaction (experimental vs. predicted) are similar for all twelve organisms. Here, it has been observed that the *P*-values of the Chi-square test (*P*-values are indicated in **Table B. 3 in Annexure B**) are greater than 0.01 in all the 12 organisms. As *P*-values are large, it can be concluded that the experimental distributions of reaction are not significantly different from the predicted

distributions. This pattern has been fairly consistent over all the organisms, where it is found that the highest fraction of reactions is regulated by single non-essential (red) or multiple non-essential genes (blue). On the other hand, fractions of reaction governed by a single essential gene are low due to a small number of minimally essential genes in all organisms. From this plot (**Figure 4.8**), it is also observed that the fractions of reactions governed by multiple essential genes are extremely low in each of the twelve organisms. These comprise the small set of reactions that are absolutely crucial for the survival of the organisms.

4.2.9. Case Study: *Leishmania donovani* and *Leishmania major*

The proposed ML Strategy 2 has been implemented for less explored organisms like *Leishmania donovani* (11 genes have gene essentiality information [186]) and *Leishmania major* (10 genes have gene essentiality information [186]) using the semi-supervised machine learning strategy. Here it is observed that the network centrality features and information-theoretic features, such as the Fourier cosine coefficient derived from the Kidera factor, have been selected by the feature selection algorithm in both the cases of *Leishmania donovani* and *Leishmania major*. Additionally, certain unique features were also selected for each of the two organisms (**Figure 4.1**). When the Kamada-Kawai dimension reduction technique was applied on *Leishmania* datasets, a similar circular pattern was observed, like the other twelve organisms that helped the classifier in predicting gene essentiality (**Figure 4.9 a**).

For the essential genes prediction, in the case of *Leishmania donovani*, 80 reaction-gene pairs were predicted as essential among 1129 reaction-gene pairs. For Leishmania major, 335 reaction-gene pairs were predicted as essential among 1188 reaction-gene pairs. The categorization of these reaction-gene pairs displayed a pattern similar to the distributions of reaction observed in the twelve model organisms (**Figure 4.9 b**). Predicted gene essentiality information from the proposed pipeline (ML Strategy 2) is

listed in, **Table B. 5** in **Annexure B**). The list of essential genes extracted from these reaction gene pairs consists of 44 essential genes of *Leishmania donovani* and 194 of L. major. These essential genes were associated with 53 and 219 Gene Ontology (Molecular Function) terms for *Leishmania donovani* and L. major, respectively (**Table B. 6** and **Table B. 7** in **Annexure B**). The Gene Ontology term that occurred most frequently with these essential genes were related to ATP binding in both the organisms. The pathway enrichment of these essential genes shows 11 significantly enriched KEGG pathways for *Leishmania donovani* and 20 *Leishmania major*. Although 8 KEGG pathways were found to be common among the two species, certain unique pathways specific to each species were also enriched for each of the two organisms (**Table B. 8 and Table B. 9** in **Annexure B**). Further experimental validation on these predicted results would confirm the role of these genes in these less-studied organisms.

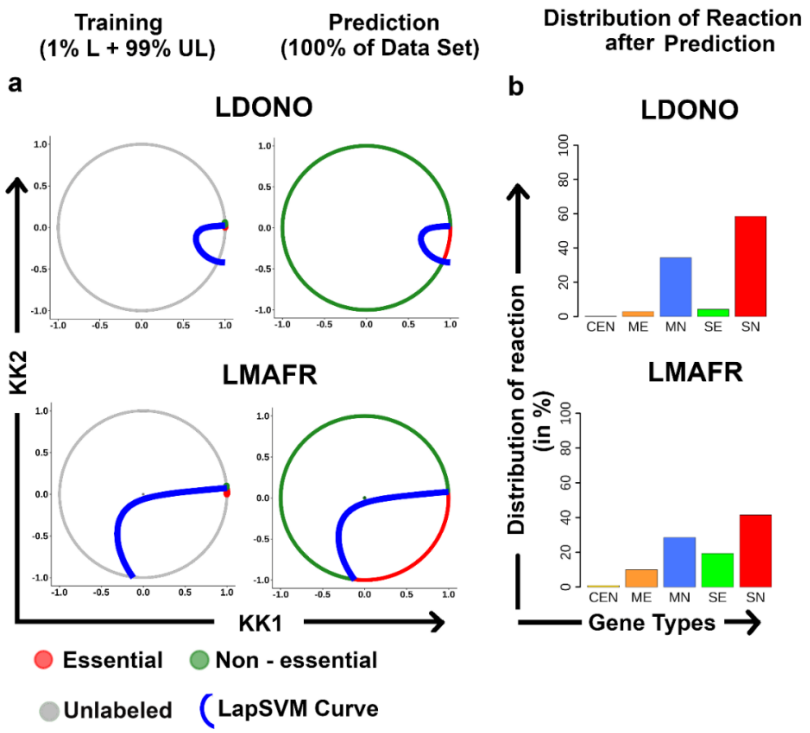


Figure 4.9. Gene essentiality prediction in *Leishmania donovani* and *Leishmania major*. (a) Kamada - Kawai dimension reduction on *Leishmania* datasets showed a circular pattern as observed for other organisms and the learning curve by LapSVM; (b) Distribution of reaction-gene pairs of *Leishmania* species into five categories

4.3. Discussion

Essential genes prediction helps to unveil the complexities and survival strategies of many disease-causing organisms. The prediction of gene essentiality is a challenging task in machine learning due to the unavailability of sufficient experimentally labeled data and lack of proper metric for selection of the best model. Considering this limited gene essentiality information, the proposed ML Strategy 2 (**Section 2.3.**, **Chapter 2**) has been able to predict gene essentiality at genome-scale using as small as 1% labeled genes having gene essentiality information on both training-blind testing dataset (80%-20%) and the whole dataset for training and testing (**Figure 4.3** and **Figure 4.6**). This proposed pipeline (ML Strategy 2) consists of three key steps. First, the unsupervised feature selection algorithm has been used to select the relevant feature set from 289 feature set consisting of different heterogeneous biological features such as sequence-based features, and topological features derived from metabolic reaction network, and flux-coupled sub-network which help to distinguish between essential and non-essential reaction gene combinations. Here, it is observed that for every organism, the features selection algorithm selected three phenotypic features that have shown high correlation with gene essentiality, *viz.*, Reaction Network betweenness centrality (RN_betweenness), Reaction Network Page Rank centrality (RN_page_rank), and Flux Coupled Analysis Network Page Rank centrality (FCA_page_rank). Apart from these, novel features considered in this study, such as Information-theoretic features (Fourier sine coefficient and Fourier cosine coefficient derived from Kidera factor), were also correlated with gene essentiality prediction in most of the organisms. A distinguishing pattern between essential and non-essential genes for the selected features was captured by the feature selection algorithm, which helped the classifier to predict gene essentiality more accurately. Secondly, dataset after feature selection was projected into a 2-D circular layout using the dimension reduction step Kamada-Kawai. This step is essential to project the high dimensional data into a 2-D plane, which helps the classifier LapSVM

to perform significantly better for all the organisms (P<0.01) (**Table B. 1** in **Annexure B**). The results show that this dimension reduction step is capable of improving the prediction accuracy even without feature selection (**Figure 4.5, Table B. 2 in Annexure B**). However, we have also retained the feature selection step in our pipeline to identify the important features that are contributing to gene essentiality classification. After applying Kamada-Kawai, a distinct structured pattern was observed, showing the essential reaction-gene combinations clustered together and the non-essential reaction-gene combination in another cluster, each residing on the arc of a 2-D circular layout for each of the twelve known organisms (**Figure 4.6**). This clustered pattern of reaction-gene pairs helped the semi-supervised classifier (Laplacian SVM) build a non-linear curve that dissects this circle into essential and non-essential classes with significantly higher accuracy. The novelty of the proposed ML Strategy 2 lies in the integration of the Kamada-Kawai algorithm with the semi-supervised LapSVM classifier that contributes to the high accuracy obtained using the pipeline. This is evident from **Table B. 1** in **Annexure B**, where a significantly higher model performance of the Kamada-Kawai step was observed over the other widely used dimension reduction techniques. Further, it has been observed that the LapSVM classifier, when combined with the Kamada-Kawai step, contributes to the higher predictive performance of this pipeline as compared to the other supervised machine learning techniques when only 1% labeled data is available (**Figure 4.4**).

Thereafter, the SSMSS score was used to select the best model. Here it was observed that the selected model based on this scoring technique had a corresponding high auROC value when compared with the experimentally known labels (**Figure 4.2**). This indicated the reliability of the proposed SSMSS score, which, although show high variation for less number of labeled data, is useful as an alternative score when the calculation of supervised metrics is difficult for best model selection.

After the successful validation of this strategy on twelve organisms, the methodology was used to annotate gene essentiality in less-studied organisms like *Leishmania*

donovani and *Leishmania major*, for which less or no organism-specific machine learning studies are available. Here, it was observed that 80 reaction-gene pairs were predicted to be essential in *Leishmania donovani*. These reactions involved 44 genes that were mostly associated with ATP binding [GO:0005524], oxidoreductase activity [GO:0016491], and AMP deaminase activity [GO:0003876] GO terms. Similarly, in the case of *Leishmania major*, 335 reaction-gene pairs were predicted as essential that involve 194 genes. Here it is observed that in addition to the ATP binding and metal-ion binding activities [GO:0005524], some genes that were predicted to be essential were also associated with amino acid transmembrane transporter activity [GO:0015171], magnesium ion binding [GO:0000287], and protein serine/threonine kinase activity [GO:0004674] GO terms that were not observed in the *Leishmania donovani*. On the other hand, in the case of *Leishmania donovani,* the genes involved in flavin adenine dinucleotide binding [GO:0050660] and AMP deaminase activity [GO:0003876] were predicted as essential, which is not observed in *L. major.*

The KEGG pathway enrichment study performed on the essential genes sets of the two organisms – *Leishmania donovani* and *Leishmania major* throw light on the pathways that are crucial for the survival of these micro-organisms and can be considered as probable therapeutic targets. Here, it is observed that apart from the pathways involved in Purine metabolism, Pyrimidine metabolism, Pyruvate metabolism, etc., that were common to both the organisms, a set of unique pathways were also enriched in each of *Leishmania donovani* and *Leishmania major*. While in the case of *Leishmania major*, the pathways involved in Glycolysis / Gluconeogenesis, Glycine, serine and threonine metabolism, Citrate cycle (TCA cycle), Pyruvate metabolism, and Inositol phosphate metabolism were significantly enriched ($P< 0.001$), the essential genes of *Leishmania donovani* show a higher enrichment for Sphingolipid metabolism and Steroid biosynthesis pathways. Further, the predicted essential reaction-gene combinations were categorized into five different groups (*i.e.*, CEN, ME, MN, SE, and SN) that help to identify the individual reactions that are

regulated by single or multiple essential genes. It may be mentioned here that a common pattern in these categories of distributions was observed across all the twelve organisms that corroborate well with the experimental observations (**Figure 4.8**). The Chi-Square Test performed to verify the difference in the experimental and predicted distributions showed no significant difference (**Table B. 3 in Annexure B**). A similar pattern was also predicted for *Leishmania donovani* and *Leishmania major* that further ascertains the validity of the predictions (**Figure 4.9 b**). These results indicate the strength of the model in identifying true essential genes using a small amount labeled data, a selection of biologically relevant features to represent gene essentiality, and optimal parameters for curve formation to classify essential genes. The limitation of the proposed ML Strategy 2 is that, it requires the genome-scale reconstructed metabolic network, and at least 1% genes of this network should be annotated experimentally with gene essentiality information.

Using a graph-based semi-supervised machine learning scheme and combining different well-established methods in ML problems, a novel integrative approach (ML Strategy 2) has been proposed for essential genes prediction that shows universality in application to both prokaryotes and eukaryotes with limited labeled data. The run time of the pipeline is dependent on the size of the metabolic network (n), and the number of features (d) considered and can be represented as T(n,d)= $O(n^3d^2)$. In the case of *Leishmania major* and *Leishmania donovani*, the total runtime was 41 minutes and 48 minutes, respectively, when simulated on a workstation of Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz with 32GB RAM. This strategy will provide experimental biologists a well standardized and validated methodology to predict gene essentiality of less-studied organisms as well as will cater to the theoretical scientists with a novel approach for binary classification problems when limited labeled data is available. The essential genes predicted using the pipeline provide important leads for the identification of novel therapeutic targets for antibiotic and vaccine development against disease-causing parasites, such as *Leishmania sp.*

Chapter 5

PRESGENE: A webserver for PRediction of ESsential GENEs using integrative machine learning strategies

5.1. Motivation

The existing essential genes prediction platforms such as such as Geptop, EGP, etc., (**Section 1.6.5.** in **Chapter 1**) can annotate essential genes for model prokaryotic organisms (with a very limited number of organism), not for eukaryotes. In these web servers, the prediction accuracy of the target organism is better when it shares a common phylogenetic ancestry with the reference species. However, they underperform when the target organism is newly sequenced, and the proper gene id and phylogeny information are not available and existing ML algorithms (**Table 1.2**, **Chapter 1**) for essential gene prediction require a large amount of labeled data to train the models and predict the essentiality of unannotated genes accurately. These ML algorithms show very poor performance when the labeled dataset is imbalanced or limited, and in most cases, no source code is publicly available for these machine learning strategies.

To address these problems, in our previous studies (**Chapter 3, Chapter 4**) we have developed and validated two ML based pipelines (**Section 2.2., 2.3.** in **Chapter 2**) that show high accuracy for prediction of essential genes. These pipelines create highly precise machine learning models with the fundamental requirements for the classification of essential and non-essential genes that mitigates the problems of limited labeled data, automatic selection of relevant, diverse biological features, and identification of an optimized set of model parameters. We have validated both the pipelines on several organisms, including both the prokaryotes and eukaryotes.

However, the preparation of the training datasets and feature tables (*i.e.*, calculation of sequence-based features, network topological properties, Flux coupling features, etc.,) are essential prerequisites for the implementation of these pipelines. It becomes quite challenging and time-consuming for experimental biologists without proper training in advanced programming languages to compute these. This necessitates developing a single user-friendly ML platform for annotating the essential genes with minimal effort and time. Hence, in this chapter, we have developed an online gene prediction server, PRESGENE, by integrating our two previously published strategies (machine learning strategy 1 and machine learning strategy 2, discussed in **Section 2.2., 2.3.** in **Chapter 2**) and considering a diverse set of relevant biological features that influence gene essentiality [57,187]. Users can submit and analyze their data for essential genes prediction through a user-friendly platform. The platform performs well for both Eukaryotes and Prokaryotes, with high accuracy. The open-source policy is maintained to provide access to the server for all researchers in the field.

5.2. Webserver Architecture and Implementation

The proposed webserver has three sections, *i.e.*, Training dataset Preparation, Model training, and Prediction. The workflow of the PRESGENE webserver is elucidated in **Figure 5.1**.

5.2.1. Training Dataset Preparation

Five input files are required for the training dataset preparation: (i) fasta file containing the coding nucleotide sequences of the genes of the organism, (ii) the ribosomal fasta file (iii) fasta file containing the protein sequences, (iv) the genome-scale reconstructed metabolic network in (*.mat) format and (v) available gene essentiality information (*i.e.*, labeled data) from experiments for building the ML model. Broadly two types of features (**Table 2.1**, **Chapter 2**) are calculated for the training and annotation of the essential genes, *viz*, the network topological features and the sequenced based features.

The topological features of the reaction network and flux-coupled sub-network are derived from the genome scale metabolic network of the organism. On the other hand, the sequenced-based features were calculated and integrated for each reaction-gene pair based on the Gene-Protein-Reaction (GPR) rule. Integration of the diverse set of features gives insights into the specific role of the gene in the metabolic network. A total of 289 features (**Table 2.1, Chapter 2**) for each reaction-gene pair can be computed to generate a training and testing dataset using the PRESGENE webserver. **Table 5.1** enlists the list of Features and the background software packages and programming languages used for automation of feature calculation in the PRESGENE webserver. A brief description of each of these features used for the gene essentiality prediction have been discussed **Section 2.1.** in **Chapter 2** in our previous work [57,187].

Table 5.1. List of features and corresponding software packages require for preparing training dataset

Feature Types	Features name	# of features	Software Packages	Programming Languages
Topological analysis of reactions and flux-coupled sub-networks				
Reaction Network	Degree Centrality, Eigen vector Centrality, Eccentricity, Hub Score, Authority Score, Page Rank, Betweenness Centrality, Number of triangle	8	The COBRA Toolbox to generate the reaction network from Genome scale metabolic network (.mat) "igraph" for network analysis	MATLAB, R, Perl
Flux Coupled Network	Degree Centrality, Eigen vector Centrality, Eccentricity, Hub Score, Authority Score, Page Rank, Betweenness Centrality, Number of triangles	8	F2C2 tool v0.95b (Flux Couple Analysis) "igraph" for network analysis	MATLAB, R, Perl

Features derived from the coding nucleotide sequences				
Derived features	Nucleotide content	4	In house Perl script	Perl
	Effective Number of Codons	1	EMBOSS package version 6.6.0-1	Perl
	Codon Adaptation Index	1	EMBOSS package version 6.6.0-1	Perl
Information-theoretic features	Mutual Information (MI)	16	in house Perl script	Perl
	Conditional Mutual Information (CMI)	64	in house Perl script	Perl
Features derived from protein sequence				
Derived features	Frequencies of the twenty amino acids	20	EMBOSS package [version 6.6.0-1]	Perl
	Protein length	1	EMBOSS package [version 6.6.0-1]	Perl
	Paralogy based features (Paralogy score)	6	BLAST [version 2.2.26]	Perl
Information-theoretic features	Fourier sine coefficient	70	in house Perl script.	Perl
	Fourier cosine coefficient	80	in house Perl script.	Perl
	Average Kidera Factor	10	in house Perl script.	Perl

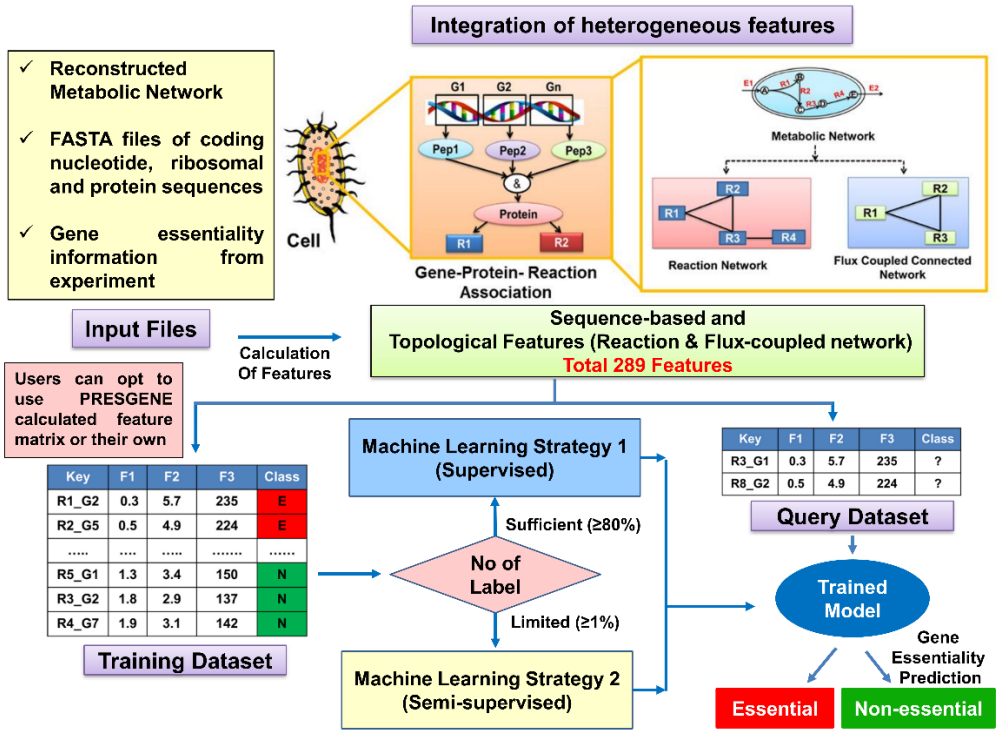
5.2.2. Development of essential genes prediction models

5.2.2.1. The ML strategy 1 (Supervised): Essential Genes Prediction with sufficient labeled data

The ML strategy 1 (**Section 2.2., Chapter 2**) has been developed to annotate and predict gene essentiality information for organisms, where the experimentally known and labeled dataset is sufficient (≥80%) but imbalanced [57]. It combines supervised feature selection technique Support Vector Machine- Recursive Features Elimination

(SVM-RFE) and ML classifier SVM to predict essential and non-essential genes from genome-scale metabolic networks.

However, it is to be noted that the original ML Strategy 1 pipeline has only been applied to prokaryotes by considering only 64 features [57]. The pipeline has now been modified to accommodate all 289 features mentioned in **Table 5.1** and show good performance with both Prokaryotes and Eukaryotes when sufficient labeled data is available.



Key	F1	F2	F3	Class
R1_G2	0.3	5.7	235	E
R2_G5	0.5	4.9	224	E
.....	....	.....	.....	.....
R5_G1	1.3	3.4	150	N
R3_G2	1.8	2.9	137	N
R4_G7	1.9	3.1	142	N

Key	F1	F2	F3	Class
R3_G1	0.3	5.7	235	?
R8_G2	0.5	4.9	224	?

Figure 5.1. Workflow for PRESGENE webserver. Two pipelines are developed and integrated to predict essential genes based on sufficient unbalanced or limited labeled training dataset with sequence, informatics, and topological network features (Reaction network, Flux Coupled Network).

5.2.2.2. The ML strategy 2 (Semi-Supervised): Essential Genes Prediction with limited labeled data

The ML strategy 2 (**Chapter 2.3., Chapter 2**) has been developed for the prediction of gene essentiality where the experimentally known and labeled dataset is limited (≥1%) for model training [187]. It combines unsupervised feature selection technique, dimension reduction using the Kamada-Kawai algorithm, and semi-supervised ML algorithm employing Laplacian SVM to predict essential and non-essential genes from

genome-scale metabolic networks. The strategy uses a novel scoring technique SSMSS (Semi-supervised Model Selection Score) for the selection of the best model [187].

5.3. User Interface Design

For the ease of execution of both the ML pipelines and calculation of the 289 features (**Table 2.1, Chapter 2**), a simple Graphical User Interface has been designed for the PRESGENE webserver. Bootstrap 4 framework has been used for designing the front-end of the server. The programming languages such as MATLAB, Perl, R and PHP have been used to write code for the automation of feature calculation and deployment of the machine learning pipelines (ML Strategy 1 and ML Strategy 2) for essential genes prediction.

5.4. Results

5.4.1. Features and functionalities of PRESGENE

The web interface of PRESGENE is designed in such a way so that users can easily interact and navigate through the interactive web pages. The "Homepage" of the webserver contains all the necessary tabs like "About PRESGENE", "Tutorial", "Sample Dataset", "Machine Learning Strategy", etc. The web server homepage also provides a detailed description of the proposed machine learning strategy 1 (ML Strategy 1) and machine learning strategy 2 (ML Strategy 2) for essential genes prediction. Users can perform analysis with a new dataset by providing the required input files for the calculation of the features based on their choice. Alternatively, the PRESGENE server also has provision for the prediction of essential genes from a user uploaded training dataset containing their own feature table. Model training can be performed using our two strategies (ML Strategy 1 and 2) depending on the availability of the labeled data. In **Figure 5.2**, a snapshot of the home page has been provided. The PRESGENE can be accessed through the following URL: https://presgene.ncl.res.in**.**

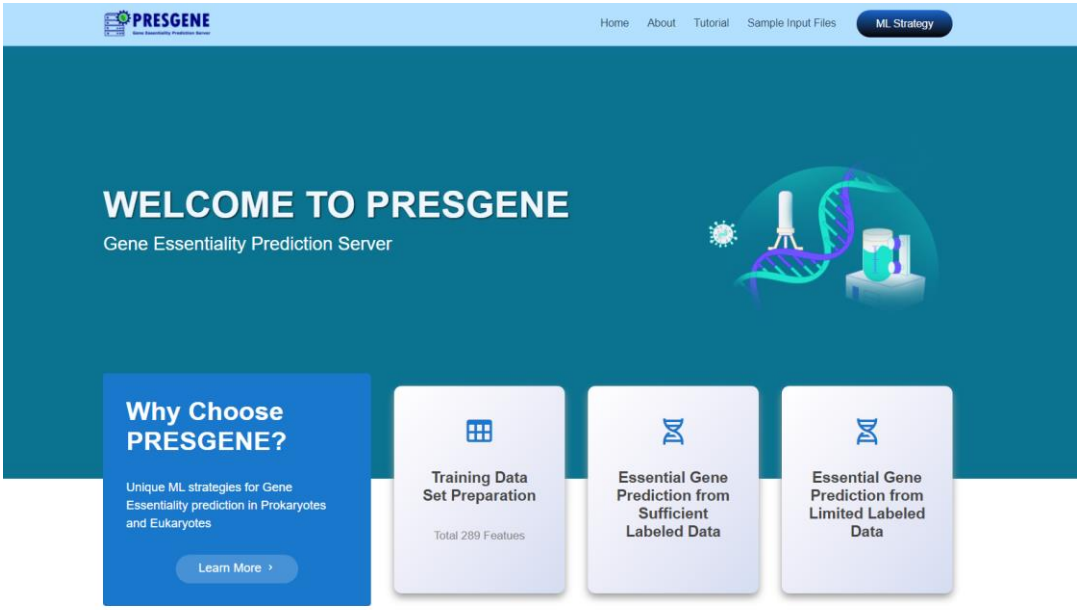


Figure 5.2. Snapshot of the web interface of PRESGENE

5.4.2. Gene Essentiality prediction using PRESGENE

The prediction of essential genes, for a new organism, using the PRESGENE server can be implemented in four simple steps. For the preparation of the training dataset, the user needs to provide the name of the organism and five input files. The input files containing the Genome Scale Reconstructed Metabolic Network (GSRMN) in (*.mat) format, (*.fasta) files of nucleotide sequence, ribosomal sequence, protein sequence and the labeled dataset (*.csv format) can be uploaded through the "Input File" navigation tab (**Figure 5.3**). It is to be noted that all input files should maintain a uniform nomenclature for the genes. A set of sample input files, for both Prokaryote (*Acinetobacter sp*. ADP1) and Eukaryote (*Saccharomyces cerevisiae*), is already made available in the PRESGENE webserver for testing the ML pipelines.

The "Dataset Preparation" tab allows the user to choose the set of biological features that the user wishes to consider for the gene essentiality prediction (**Figure 5.4**). However, it is recommended that all 289 biological features are considered for higher accuracy and better prediction of essential genes.

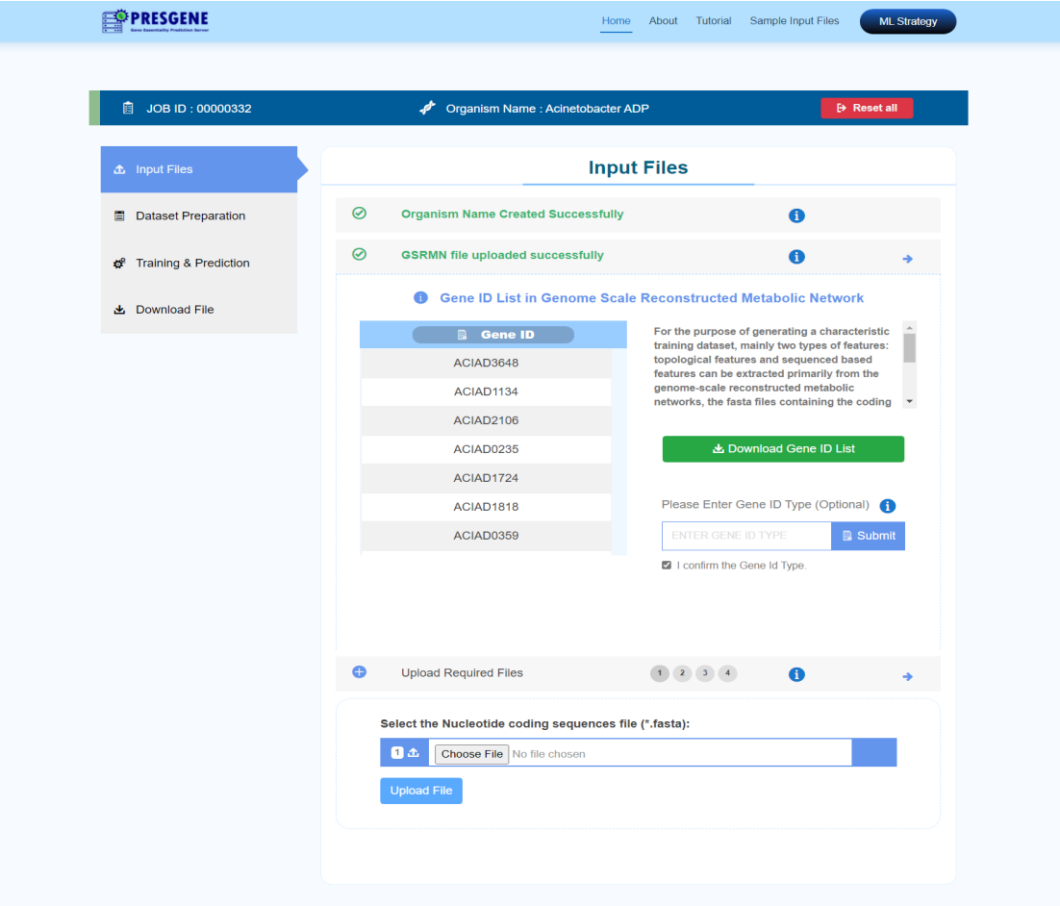


Figure 5.3. Input files navigation tab

Based on the availability of the experimentally labeled data the user can then train the model using either ML Strategy 1 (if labeled data ≥80% of the total dataset) or ML Strategy 2 (if labeled data ≥1% of the total dataset). The performance metrics of the model are displayed on the "Training & Prediction" page (**Figure 5.5**). In addition to the supervised performance metrics (**Section 1.6.3.5., Chapter 1**) such as Precision, Recall, TPR, FPR, auROC and MCC in ML Strategy 1, PRESGENE offers a novel scoring technique SSMSS (Semi-supervised Model Selection Score) for the section of the best model using ML Strategy 2 where the calculation of the supervised metrics is difficult. Additionally, PRESGENE allows the user to vary the feature set and recalculate the feature table to observe the variation in the prediction accuracy.

The results along with the calculated feature table generated for the prediction of the essential genes using the PRESGENE server can be downloaded in .csv format by the user from the "Download File" tab.

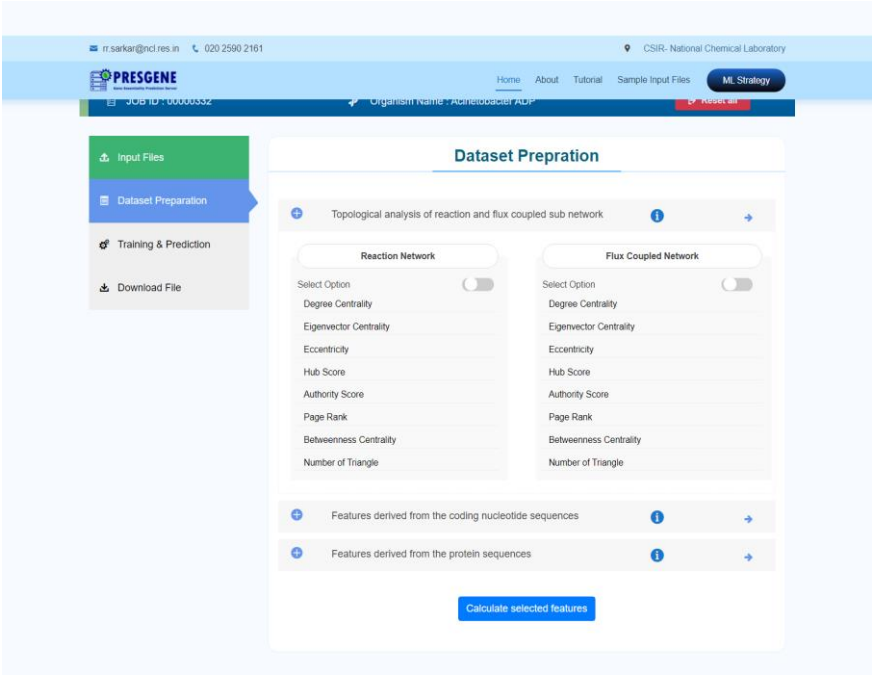


Figure 5.4. Feature Calculation Page for training dataset preparation.

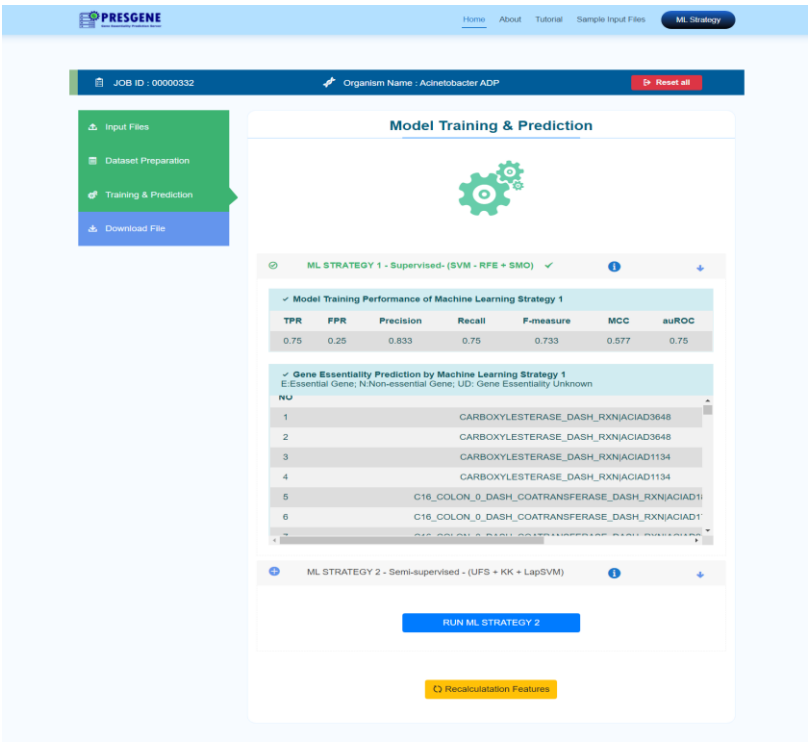


Figure 5.5. Machine learning training and gene essentiality prediction output

5.5. Discussion

Essential genes prediction helps to understand the survival strategies and complexities of many disease-causing organisms. Due to dataset imbalance, limitation

of availability of labeled data from experiments, and lack of a proper metric for selecting the best model, the annotation of gene essentiality is a challenging task in machine learning. The proposed webserver PRESGENE addresses these problems by facilitating annotation of essential genes at genome-scale using only 1% labeled genes. This web server consists of three key steps - Training dataset preparation, Model Training and Model Prediction. First, the training dataset can be generated by the user by calculating 289 heterogeneous biological features consisting of sequence-based features and topological features derived from the metabolic reaction network, and its corresponding flux-coupled sub-network. The feature set considered in the PRESGENE server consists of many novel features such as FCA based features Kidera Factor, Fourier sine and cosine coefficient of protein sequences (**Table 5.1**) that have not been considered in other Essential Genes prediction servers. The server offers the user a choice of two previously validated ML pipelines (ML strategy 1 and ML strategy 2) that can be used to train the model and predict essential genes for both Prokaryotes and Eukaryotes [14,15]. These pipelines integrated with the webserver addresses the issue of dataset imbalance and training using limited labeled data to annotate gene essentiality with high accuracy. The PRESGENE server also offers a unique scoring technique SSMSS to select the best model when the calculation of the supervised performance metrics is difficult due to limited labeled data.

Although a key limitation of the server lies in the fact that both the ML strategies fail to execute if the genome scale reconstructed metabolic network of the organism and a minimum of 1% labeled dataset are not available. Nevertheless, PRESGENE will be invaluable to experimental biologists by providing a well-validated and standardized platform to annotate gene essentiality of less-explored organisms with minimal information of labeled data. The essential genes predicted using the platform has wide applicability and will be useful for the identification of novel therapeutic targets against disease-causing parasites, such as *Leishmania sp., Salmonella sp., Staphylococcus sp., etc.,* for antibiotic and vaccine development.

Chapter 6

Conclusion and Future directions

6.1. Conclusion

Information about the gene essentiality in organism aids in the understanding of the minimally essential genes that are absolutely required for the organism's survival in any environmental condition. However, the experimental techniques used to conduct a genome-wide screen for gene essentiality are costly, labor-intensive, and time-consuming. On the other hand, various computational techniques are available to annotate gene essentiality. However, major drawbacks of existing machine learning-based methods for predicting gene essentiality are that they require a large amount of labeled data from experiments and perform poorly when the labeled dataset is imbalanced or insufficient. Experimentalists frequently encounter these issues when studying new or less studied organisms with a limited number of experimentally annotated genes. The problem is exacerbated further when the organism shares a small number of conserved orthologous genes with other species, which are not always indispensable, as gene essentiality is strongly influenced by the organism's various environmental conditions.

Using computational approaches, we have addressed these issues and we have focused on the development of machine learning strategies for the prediction of essential genes. Here, we have developed and validated two ML-based pipelines (**Section 2.2. - 2.3. in Chapter 2**) that show high accuracy for predicting essential genes in cases with imbalanced labeled training datasets and cases with limited experimental data.

The first strategy, *i.e.*, ML strategy 1 (**Section 2.2., Chapter 2**) based on the Supervised ML approach, was developed for predicting essential genes when sufficient experimental data (labeled data ≥80%) is available, but the dataset is imbalanced [57]. Here we have used a simple support vector machine-based learning strategy for the

prediction of essential genes in *Escherichia coli* K-12 MG1655 metabolism that integrates a non-conventional combination of an appropriate sample balanced training set, a unique organism-specific genotype and phenotype attributes that characterize essential genes. Optimal parameters of the learning algorithm generate the best machine learning model (the model with the highest accuracy among all the models trained for different sample training sets). We also present flux-coupled metabolic subnetwork-based features for enhancing the classification performance for the first time. Our strategy proves to be superior compared to previous SVM-based strategies in obtaining a biologically relevant classification of genes with high sensitivity and specificity [57]. The testing accuracy was high compared to the known techniques, proving that our method outperforms known methods. Observations from our study indicate that essential genes are conserved among homologous bacterial species demonstrating high codon usage bias, GC content and gene expression, and possess a tendency to form physiological flux modules in metabolism.

On the other hand, in the second strategy (**Section 2.3. in Chapter 2**), using a graph-based semi-supervised ML approach, we have proposed another pipeline for the classification of gene essentiality of organisms where the availability of experimental data is very limited (labeled data ≥1%) [187]. After the validation of ML strategy 2 (in **Chapter 4**) on nine prokaryotes and three eukaryotes, the methodology was used to annotate gene essentiality in less-studied organisms like *Leishmania donovani* and *Leishmania major*, for which less or no organism-specific machine learning studies are available. Here, it was observed that 80 reaction-gene pairs were predicted to be essential in *Leishmania donovani*. These reactions involved 44 genes that were mostly associated with ATP binding [GO:0005524], oxidoreductase activity [GO:0016491], and AMP deaminase activity [GO:0003876] GO terms. Similarly, in the case of *Leishmania major*, 335 reaction-gene pairs were predicted as essential that involve 194 genes. Predictions for *Leishmania* species are further validated with the experimentally observed pattern of Reaction-Gene combinations occurring in other organisms. These

predicted essential reaction-gene combinations were categorized into five different groups (*i.e.*, CEN, ME, MN, SE, and SN) that help to identify the individual reactions that are regulated by single or multiple essential genes. A similar pattern was also observed for *Leishmania donovani* and *Leishmania major* that further ascertains the validity of our predictions. These results indicate the strength of our model in identifying true essential genes using a minimum of 1% labeled data, to select biologically relevant features representing gene essentiality, and optimal parameters for curve formation for classifying essential genes. This new pipeline (ML Strategy 2) for essential genes prediction shows universality in application to both prokaryotes and eukaryotes with limited labeled data (**Chapter 4**).

Both the pipelines (**Section 2.2. - 2.3. in Chapter 2**) have been validated with several organisms. These pipelines create highly precise machine learning models with the objective of classifying genes as essential and non-essential. However, the preparation of the training datasets with feature tables and implementation of ML algorithms may be quite challenging and time-consuming for experimental biologists. This necessitates for developing a user-friendly ML platform for annotating the essential genes with minimal effort and time. Hence, we have developed an online open-source gene prediction server, PRESGENE (**Chapter 5**), by integrating our two previously published strategies machine learning strategy 1 and machine learning strategy 2, discussed in **Chapter 2** by considering a diverse set of relevant biological features that influence gene essentiality [57,187]. The user can easily submit and analyze their data for essential genes prediction through a user-friendly platform. The PRESGENE platform also predicts gene essentiality for less studied organisms. The platform performs well for both eukaryotes and prokaryotes, with high accuracy.

6.2. Future directions

In future, this study can be improved in following aspects *i.e.*, to apply this ML strategy on different complex biological problems and experimentally validate the predicted results.

This strategy would be useful to understand the pathogen biology *i.e.*, essential genes in pathogen will help to prioritize a set of crucial genes and their functional properties. The essential genes of disease-causing organisms serve as a list of probable potential drug targets. So, the essential genes prediction using PRESGENE webserver provide important leads for identifying novel therapeutic targets for antibiotic and vaccine development against pathogen.

From an evolutionary standpoint a distinct correlation between gene essentiality and its impact on conservation is suggested in a family of organisms, one can utilize this gene essentiality study in this perspective to construct the phylogenetic relation between class and family of organisms.

From the synthetic biologist's standpoint, essential genes set intersect widely with the minimal gene set required for an organism's survival; thus, identification of the essential set also leads to the synthetic reconstruction of the organism to produce the desired by-product optimally. So, the gene essentiality information can be used for manipulating the organism. For example, in food microbiology, industrial bioprocessing essential genes of plants, animals, and microorganisms are modified to produce food, biofuel, and biocatalyst at a large scale.

Conveniently, this strategy can be applied on biological problems such as protein function prediction, Gene Ontology (GO) analysis, where features can be calculated from different biological data such as genomics and proteomics profile, protein sequence information, physicochemical properties, structural conformation, protein interaction map etc.

Similarly, reconstruction of metabolic network in specific human cell line can provide insights into essential genes. Using these essential genes as a parameter, important biomarker associated with diseases can be identified. This ML strategy can also be used as a diagnostics tool to classify patients as normal or cancerous based on different grades according to features that are available from clinical (demography, habits, age, sex, etc.), biospecimen (tumor tissue morphology), multi-omics data [Gene expression, Simple Nucleotide Variation (SNV), Copy Number Variations (CNV) and Methylation].

On the other hand, one can collaborate with experimental biologist to validate the prediction results. For example, machine learning strategy 2 was applied to annotate gene essentiality in less-studied organisms like *Leishmania donovani* and *Leishmania major*. The experimental testing and validation can be performed for the predicted essential genes in Leishmania species.

In technical aspect one can upgrade the PRESGENE webserver by parallelizing background codes, allowing more CPU utilization for handling simultaneous use of multiple users. Currently, PRESGENE considers 289 biological features for the predicting of essential genes, further on one can include new novel features in the server, based on the recent literature evidences. Hence, further improvement in algorithms and implementation of ML strategies can provide better prediction accuracy with improved run time.

In conclusion, outcomes of the thesis not only contribute to the development of the machine learning strategies but also provide an interactive open-sourced webserver for annotation of the gene essentiality for disease-causing organisms.

Annexure

Annexure A

Text A- 1. Algorithm of our proposed machine learning strategy 1

The instance-feature file (see **Table 2.1** in **Chapter 2**), containing 384 essential and 3120 non-essential reaction-gene pairs (R_a_G_b) was given to the following algorithm. It is noteworthy to mention that, our methodology is not specific to only this problem of classification and hence, can be applied for classifying any other kinds of datasets. Further, 1000 randomized balanced datasets (equal number of positive and negative classes) were generated and given to the integrated pipeline.

Algorithm for choosing best feature combination (Part 1)

The following algorithm was used to choose best features from a total of 64 features:

```
// BD: Set of 1000 Balanced training Datasets
// TRF: Set of Top Ranking Features
// PM: Set of Performance Metrics (auROC)
// perf: Set of best performing metrics
// BD_temp: Set of Balanced Datasets giving high performance
    for i=1: length(BD)
            TRF[ ] = Features ranked (descending) using SVM-RFE
            for j=1: length(TRF[ ])
               PM[i][j] = auROC measured using SMO
            end loop j
            BFC[i] = best features combination set from PM[i][j] which gives
    best auROC
            for k=1: length(BD)
               BFC_BD[i][k] = performance metric with BFC[i] measured
    using  SMO
            end loop k
     end Loop i
     for m=1: length(BD)
       auROC[m] = sum(BFC_BD[m]) / length(BFC_BD[m]);
```

end loop $_m$
Sort (descending) auROC[ ]
Select Best feature combination set (BFC_{best}) which gives highest average performance (auROC[1])

Algorithm for parameter optimization (Part 2)

Training with BFC_{best} (obtained from Part 1) and tuning complexity parameter C

C[ ] = {0.01, 0.1, 1, 10, 100};
for i = 1 : length(C[ ])
for j = 1: length(BD)
PM[i][j] = performance metrics measured using SMO;
end loop $_j$
Sort (descending) PM[i] and choose corresponding BD[j]
perf[i] = PM[1]; // best auROC for C[i]
BD_{temp}[i] = BD[1]; // dataset giving best auROC for C[i]
avePERF_C[i]=**sum**(PM[i])/**length**(PM[i]);
end loop $_i$
Sort (descending) perf[ ] and choose corresponding BD_{temp}[ ]
Sort (descending) avePERF_C[ ] and choose corresponding C[ ]

Output = Best feature combination, best penalty parameter, best dataset (**BFC_{best}, C_{best}, BD_{best}**) which gives highest performance

Algorithm for Model testing

The unbalanced instance-feature file again was given as the total master test set. The testing algorithm returns two results -

1) Predictions from the best model (C_{best}, BFC_{best}, BD_{best})

2) Predictions of the model (C_{best}, BFC_{best}) with respect to the 1000 random datasets -

Given - C_{best}, BFC_{best}
for i=1: length(Ra_Gb)
count = 0;
for j=1: length(BD)
if Ra_Gb[i] == "E" // "E" means essential
count=count+1;
end if
end loop $_j$
PercentagePrediction = [count / **length**(BD)]*100;
End

Table A. 1 : The 26 selected best features

Feature Name	Rank (SVM-RFE)	*P*-value
PR	1	2.2E-16
H3	2	2.2E-16
RN_CC	3	2.2E-16
H5	4	2.2E-16
Ala_f	5	0.000052
Arg_f	6	0.00996
H7	7	2.2E-16
CAI	8	0.2447*
FCA_AS	9	2.2E-16
FCA_Num_triangles	10	2.2E-16
FCA_Hub	11	2.2E-16
Leu_f	12	0.2669*
RN_BC	13	0.000309
GC1	14	2.2E-16
T3	15	1.35E-10
NGSE	16	0.2399*
Gly_f	17	2.2E-16
Val_f	18	0.5293*
Ser_f	19	0.009535
FCA_ClusteringCoef	20	3.08E-12
Phe_f	21	2.21E-15
C3	22	1.65E-05
Glu_f	23	0.001545
GC2	24	0.2302*
Asn_f	25	2.2E-16
FCA_BC	26	2.93E-15

* *P*-values insignificant at a threshold of $P < 0.05$

Annexure B

Text B- 1. Total energy (W_{tot}) of the Kamada-Kawai algorithm

$$W_{tot} = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{1}{2} k_{i,j} \left(\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 - 2 l_{i,j} \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} + \left(l_{i,j}\right)^2 \right)$$

$$\frac{\partial W_{tot}}{\partial x_m} = \frac{\partial W_{tot}}{\partial y_m} = 0, \quad for \quad 1 \le m \le n \qquad \textbf{Eq. B- 1}$$

$$\Delta_m = \sqrt{\left(\frac{\partial W_{tot}}{\partial x_m}\right)^2 + \left(\frac{\partial W_{tot}}{\partial y_m}\right)^2}$$

Where,

- Pairwise distance $\left(d_{i,j}\right)$ = Length of the shortest path between two points $P_i(x_i, y_i)$ and $P_j(x_j, y_j)$
- Pairwise ideal distance $\left(l_{i,j}\right) = L \times d_{i,j}$
- L = Length of single edge, $k_{i,j}$ is spring constant

Text B- 2. Source Code of Proposed Machine learning strategy 2

The proposed pipeline has four following steps: -

a) Training dataset preparation and integration of heterogeneous features [Script No: 1-25]

b) Feature selection based on the space-filling concept [Script No: 26]

c) Dimension reduction using forced directed graph layout [Script No: 27]

d) Semi-supervised classifier: LapSVM [Script No: 28]

We have used the following programming languages (*e.g.*, PERL, MATLAB, R) to develop the proposed machine learning strategy 2.

Script Number of each step is given in square bracket. Inputs for the pipeline are the genome-scale reconstructed metabolic networks, the fasta files containing the coding nucleotide sequences of the genes, protein sequences of these target organisms and limited gene essentiality information from experiment.

Source code of this strategy is available in the following URL: https://doi.org/10.1371/journal.pone.0242943.s015 as supporting information in our published paper [187].

Table B. 1: Comparison of auROC of Kamada-Kawai (KK) dimension Reduction technique with PCA, MDS, FR and ICA. The values reported in the table represent the P-values obtained using the one-tailed Mann-Whitney U Test.

Organisms	Scenario 6 (S6): [WOFS+DR(KK)]	Scenario 2 (S2): [WOFS+DR(PCA)]	Scenario 3 (S3): [WOFS+DR(MDS)]	Scenario 4 (S4): [WOFS+DR(FR)]	Scenario5 (S5): [WOFS+DR(ICA)]
ACIAD		2.45E-30	5.31E-30	8.77E-35	4.34E-30
BACSU		5.99E-29	2.21E-28	1.25E-29	1.87E-29
CELEG		2.09E-30	2.65E-30	1.91E-30	1.47E-30
ECOLI		2.36E-22	2.09E-22	4.42E-20	7.28E-23
HELPY		6.81E-31	6.81E-31	5.02E-36	2.55E-31
MUSMU		2.56E-10	5.55E-11	7.37E-10	2.93E-11
MYCTU		3.56E-26	3.27E-26	3.73E-26	1.25E-25
PSEAB		1.16E-24	2.63E-28	9.94E-24	1.13E-27
PSEAE		4.53E-28	4.53E-28	4.63E-30	1.67E-29
SALTY		9.56E-27	7.08E-27	6.06E-25	1.14E-26
STAAB		7.20E-31	2.08E-31	5.98E-34	2.22E-31
YEAST		1.23E-30	1.22E-30	3.88E-26	2.22E-31

Note: Null Hypothesis (H_0) is that the auROC of Scenario 6 [S6: WOFS + DR (KK)] is not different from the auROC of Scenarios 2 to 5 for all twelve organisms. Alternative Hypothesis (H_1) is that the auROC of Scenario 6 [S6: WOFS + DR (KK)] is greater than the auROC of Scenarios 2 to 5 for all twelve organisms.

Table B. 2: Comparison of the effect of feature selection and Kamada-Kawai (KK) dimension Reduction technique on the model performance (auROC). The values reported in the table represent the P-values obtained using the one-tailed Mann-Whitney U Test.

Organisms	Scenario 1 (S1): [WOFS+WODR]	Scenario 6 (S6): [WOFS+DR(KK)]	Scenario 7 (S7): [WFS (UFS)+DR(KK)]
ACIAD		5.42E-31	4.26E-32
BACSU		4.43E-30	1.95E-30
CELEG		1.08E-30	1.08E-30
ECOLI		5.82E-23	7.41E-30
HELPY		4.44E-32	2.43E-32
MUSMU		1.51E-12	3.59E-19
MYCTU		2.86E-25	4.93E-32
PSEAB		1.35E-26	1.71E-31
PSEAE		7.09E-30	6.58E-33
SALTY		5.85E-27	9.45E-33
STAAB		2.26E-33	8.95E-34
YEAST		2.15E-31	2.15E-31

Note: Null Hypothesis (H_0) is that the auROC of Scenario 1 [S1: WOFS + WODR] is not different from the auROC of Scenarios 6 and 7 for all twelve organisms. Alternative Hypothesis (H_1) is that the auROC of Scenario 1 [S1: WOFS + WODR] is less than the auROC of Scenarios 6 and 7 for all twelve organisms.

Table B. 3: Comparison of percentage distribution of reaction into five categories from experiment vs predicted results. The values reported in the table represent the P-values obtained using the Chi-square test.

Organisms	*P*-value
ACIAD	0.996989
BACSU	0.599506
ECOLI	0.760862
HELPY	0.335664
MYCTU	0.764309
PSEAE	0.760246
PSEAB	0.944716
SALTY	0.504629
STAAB	0.808256
YEAST	0.768774
CELEG	0.391964
MUSMU	0.018437

Note: Null Hypothesis (H_0) is that the two distributions of reaction (experimental vs. predicted) are not different for all twelve organisms. Alternative Hypothesis (H_1) is that the two distributions of reaction (experimental vs. predicted) are different for all twelve organisms.

Table B. 4: Gene essentiality information of Reaction Gene combinations in *Leishmania donovani* predicted using our proposed machine learning strategy 2

[#]Experimental Gene Essentiality Information (Jones,N.G. *et al.* (2018) ACS Infect. Dis., 4, 467–477.)

Abbreviation
E, Essential gene
N, Non-essential gene
UD, Undetermined (Not Known)

GeneReaction	Experiment[#]	Predicted	GeneReaction	Experiment[#]	Predicted	GeneReaction	Experiment[#]	Predicted
4COUCOAL|LdBPK_303150	E	E	TRPDOXtn|LdBPK_323110	UD	E	UDPtn|LdBPK_252480	UD	E
DKMD2K|LdBPK_160590	E	E	TRPDOXtn|LdBPK_353910	UD	E	UDPtn|LdBPK_361410	UD	E
LACDtg|LdBPK_341110	E	E	TRPDRDtn|LdBPK_040270	UD	E	UDPx|LdBPK_241880	UD	E
MLTHFtm|LdBPK_040570	E	E	TRPDRDtn|LdBPK_130870	UD	E	UDPx|LdBPK_242110	UD	E
PMTCOAtm|LdBPK_120105	E	E	TRPDRDtn|LdBPK_322690	UD	E	UDPx|LdBPK_272390	UD	E
THRt6|LdBPK_181500	UD	E	TRPDRDtn|LdBPK_354860	UD	E	UMPK|LdBPK_271940	UD	E
THRt6|LdBPK_181510	UD	E	TRPTRS|LdBPK_311800	UD	E	UMPK|LdBPK_290290	UD	E
THYMDtm|LdBPK_060910	UD	E	TRYPm|LdBPK_241910	UD	E	URAtn|LdBPK_323110	UD	E
THYMDtm|LdBPK_070150	UD	E	TRYR|LdBPK_241910	UD	E	URAtn|LdBPK_353910	UD	E
THYMDtm|LdBPK_282700	UD	E	TYRabc|LdBPK_010470	UD	E	UREAt|LdBPK_360280	UD	E
THYMDtm|LdBPK_352780	UD	E	TYRTA|LdBPK_180440	UD	E	UREAtg|LdBPK_170410	UD	E
TKT1g|LdBPK_341170	UD	E	TYRTA|LdBPK_191380	UD	E	UREAtg|LdBPK_322210	UD	E
TKT2g|LdBPK_341170	UD	E	TYRTA2|LdBPK_120580	UD	E	URItn|LdBPK_210990	UD	E
TMDPP|LdBPK_302920	UD	E	TYRTA2|LdBPK_350840	UD	E	VALt6|LdBPK_050980	UD	E
TMDPP|LdBPK_365410	UD	E	TYRTA3|LdBPK_260680	UD	E	VALt6|LdBPK_170320	UD	E
TMDS|LdBPK_120200	UD	E	UAGDP|LdBPK_050180	UD	E	VALt6|LdBPK_181460	UD	E
TPIg|LdBPK_362740	UD	E	UAGDP|LdBPK_250020	UD	E	VALt6|LdBPK_270300	UD	E
TRNAGLNtm|LdBPK_290920	UD	E	UDPDPS|LdBPK_366170	UD	E	VALt6|LdBPK_270590	UD	E
TRNAGLUtm|LdBPK_230500	UD	E	UDPGALtg|LdBPK_230580	UD	E	VALt6|LdBPK_282170	UD	E
TRNAGLUtm|LdBPK_364510	UD	E	UDPGALtg|LdBPK_230880	UD	E	VALt6|LdBPK_323370	UD	E
TROPtm|LdBPK_040440	UD	E	UDPGtg|LdBPK_181580	UD	E	VALt6|LdBPK_352010	UD	E
VALt6|LdBPK_365620	UD	E	ZYMSTt|LdBPK_330530	UD	E	XYLTRED_D|LdBPK_161340	UD	E
VALTAc|LdBPK_161410	UD	E	ZYMSTter|LdBPK_353280	UD	E	XYLTRED_D|LdBPK_355060	UD	E
VALTRS|LdBPK_060350	UD	E	XANtg|LdBPK_280980	UD	E	XTSN2t|LdBPK_351540	UD	E
VTm|LdBPK_061330	UD	E	XMPtg|LdBPK_365650	UD	E			
XANt2|LdBPK_331010	UD	E	XTSN2t|LdBPK_070210	UD	E			
XANtg|LdBPK_221110	UD	E	XTSN2t|LdBPK_312650	UD	E			
XANtg|LdBPK_271970	UD	E	XTSN2t|LdBPK_350100	UD	E			

Table B. 5: Gene essentiality information of Reaction Gene combinations in *Leishmania major* predicted using our proposed machine learning strategy 2

#Experimental Gene Essentiality Information (Jones,N.G. *et al.* (2018) ACS Infect. Dis., 4, 467–477.)

Abbreviation
E, Essential gene
N, Non-essential gene
UD, Undetermined (Not Known)

GeneReaction	Experiment#	Predicted	GeneReaction	Experiment#	Predicted	GeneReaction	Experiment#	Predicted
MTHFD|LmjF.26.0320	E	E	NICRNTK|LmjF.30.0370	UD	E	ORNDC|LmjF.12.0280	UD	E
DHFOR2|LmjF.06.0860	E	E	NICRNTK|LmjF.32.1810	UD	E	ORPTg|LmjF.16.0550	UD	E
DHFOR2a|LmjF.06.0860	E	E	NICRNTK|LmjF.07.0170	UD	E	P5CDm_i|LmjF.03.0200	UD	E
DHFR|LmjF.06.0860	E	E	NICRNTK|LmjF.11.0250	UD	E	P5CDr|LmjF.03.0200	UD	E
DHFRa|LmjF.06.0860	E	E	NICRNTK|LmjF.27.0100	UD	E	P5CRr|LmjF.13.1680	UD	E
ADNCYC|LmjF.28.0090	E	E	NICRNTK|LmjF.13.0780	UD	E	P5CRrm|LmjF.13.1680	UD	E
GUACYC|LmjF.28.0090	E	E	NICRNTK|LmjF.36.4250	UD	E	PAPA_LM|LmjF.19.1350	UD	E
MTHFC|LmjF.26.0320	E	E	NO3R|LmjF.30.0610	UD	E	PAPA_LM|LmjF.18.0440	UD	E
PIN3K_LM|LmjF.34.4530	E	E	NPHPPH|LmjF.31.2340	UD	E	PAPAm_LM|LmjF.19.1350	UD	E
THFOAi|LmjF.06.0860	E	E	NPHPPH|LmjF.31.2340	UD	E	PAPAm_LM|LmjF.18.0440	UD	E
THFOCi|LmjF.06.0860	E	E	NTRLASE|LmjF.26.2280	UD	E	PDHe1|LmjF.18.1380	UD	E
TMDS|LmjF.06.0860	E	E	NTRLASE|LmjF.26.2280	UD	E	PDHe1|LmjF.35.0050	UD	E
TRYR|LmjF.05.0350	E	E	NTRLASE4|LmjF.26.2280	UD	E	PDHe1|LmjF.25.1710	UD	E
TYRTRS|LmjF.14.1370	E	E	NTRLASE4|LmjF.26.2280	UD	E	PDHe2|LmjF.36.2660	UD	E
TYRTRS|LmjF.14.1370	E	E	OCCOADm|LmjF.28.2510	UD	E	PDHe2|LmjF.21.0550	UD	E
NDPKn5|LmjF.32.2950	UD	E	OCCOADm|LmjF.06.0880	UD	E	PDHe3|LmjF.31.2650	UD	E
NDPKn6|LmjF.35.3870	UD	E	OCDMAT8m|LmjF.05.0520	UD	E	PDHe3|LmjF.32.3310	UD	E
NDPKn6|LmjF.32.2950	UD	E	OCMAT3m|LmjF.05.0520	UD	E	PDHe3|LmjF.29.1830	UD	E
NDPKn7|LmjF.32.2950	UD	E	OCOAT1r_m|LmjF.33.2340	UD	E	PDHe3|LmjF.31.2640	UD	E
NDPKn7|LmjF.35.3870	UD	E	OCOAT1r_m|LmjF.30.1930	UD	E	PDHe3|LmjF.31.2640	UD	E
NDPKn8|LmjF.32.2950	UD	E	OCOAT1r_m|LmjF.30.1940	UD	E	PETOHM_LM|LmjF.31.2290	UD	E
NDPKn8|LmjF.35.3870	UD	E	ODH1mi|LmjF.21.1430	UD	E	PETOHMm_LM|LmjF.31.2290	UD	E
NDPKn9|LmjF.35.3870	UD	E	ODH2mi|LmjF.21.1430	UD	E	PFK26|LmjF.03.0800	UD	E
NDPKn9|LmjF.32.2950	UD	E	ODHmi|LmjF.21.1430	UD	E	PFK26|LmjF.26.0310	UD	E
NICRNTK|LmjF.29.2150	UD	E	OHPHM|LmjF.35.4250	UD	E	PFKg|LmjF.29.2510	UD	E
NICRNTK|LmjF.30.0600	UD	E	OMPDCg|LmjF.16.0550	UD	E	PGCDr|LmjF.03.0030	UD	E
PGDH|LmjF.35.3340	UD	E	PHEt6|LmjF.22.0230	UD	E	PMEVKx|LmjF.15.1460	UD	E

PGI1|LmjF.12.0530	UD	E		PHEt6|LmjF.14.0320	UD	E		PNS1|LmjF.29.2800	UD	E
PGI2|LmjF.12.0530	UD	E		PHEt6|LmjF.27.0670	UD	E		PNS2|LmjF.29.2800	UD	E
PGI3|LmjF.12.0530	UD	E		PHEt6|LmjF.11.0520	UD	E		PNS3|LmjF.29.2800	UD	E
PGK|LmjF.20.0110	UD	E		PI45BPP_LM|LmjF.35.0040	UD	E		PNS4|LmjF.29.2800	UD	E
PGK|LmjF.20.0110	UD	E		PI45BPP_LM|LmjF.30.2950	UD	E		PNTK|LmjF.28.0140	UD	E
PGK|LmjF.20.0100	UD	E		PI4P5K_LM|LmjF.34.3090	UD	E		PNTK2|LmjF.28.0140	UD	E
PGK|LmjF.30.3380	UD	E		PIN3K_LM|LmjF.24.2010	UD	E		PPA|LmjF.03.0910	UD	E
PGKg|LmjF.20.0100	UD	E		PIN3K_LM|LmjF.02.0120	UD	E		PPA_1|LmjF.31.1220	UD	E
PGKg|LmjF.30.3380	UD	E		PIN3K_LM|LmjF.20.1120	UD	E		PPA_1v|LmjF.11.0210	UD	E
PGKg|LmjF.20.0110	UD	E		PIN3K_LM|LmjF.30.1850	UD	E		PPCDC|LmjF.30.1540	UD	E
PGKg|LmjF.20.0110	UD	E		PIN3K_LM|LmjF.34.3940	UD	E		PPCKg|LmjF.27.1805	UD	E
PGL|LmjF.26.2700	UD	E		PIN4K_LM|LmjF.34.3590	UD	E		PPCKg|LmjF.27.1810	UD	E
PGLg|LmjF.26.2700	UD	E		PINOS_LM|LmjF.26.2480	UD	E		PPCOACm|LmjF.28.0490	UD	E
PGM|LmjF.33.2110	UD	E		PIt6|LmjF.03.0500	UD	E		PPCOACm|LmjF.01.0050	UD	E
PGM|LmjF.36.4070	UD	E		PIt6|LmjF.10.0030	UD	E		PPDKg|LmjF.11.1000	UD	E
PGM|LmjF.28.2220	UD	E		PIt6|LmjF.10.1300	UD	E		PPPGO|LmjF.06.1280	UD	E
PGM|LmjF.36.6650	UD	E		PItm|LmjF.05.0290	UD	E		PROTRS|LmjF.18.1210	UD	E
PGM|LmjF.08.0060	UD	E		PItm|LmjF.35.4420	UD	E		PROt6|LmjF.27.0670	UD	E
PGMT|LmjF.21.0640	UD	E		PItm|LmjF.35.4430	UD	E		PROt6|LmjF.22.0230	UD	E
PHCYT_LM|LmjF.26.1620	UD	E		PLAc_LM|LmjF.35.3020	UD	E		PROt6|LmjF.14.0320	UD	E
PHCYTm_LM|LmjF.26.1620	UD	E		PLAe_LM|LmjF.35.3020	UD	E		PROt6|LmjF.11.0520	UD	E
PHETA1|LmjF.35.0820	UD	E		PMANM|LmjF.36.1960	UD	E		PRPPSi|LmjF.08.0510	UD	E
PHETA1|LmjF.36.2360	UD	E		PMANMg|LmjF.34.3780	UD	E		PRPPSi|LmjF.08.1130	UD	E
PHETA1m|LmjF.24.0370	UD	E		PMETM_LM|LmjF.31.3120	UD	E		PRPPSig|LmjF.36.5390	UD	E
PHETRS|LmjF.32.0870	UD	E		PMETMm_LM|LmjF.31.3120	UD	E		PRPPSig|LmjF.33.1930	UD	E
PHETRS|LmjF.19.1040	UD	E		PMEVK|LmjF.15.1460	UD	E		PSD_LM|LmjF.35.4590	UD	E
RNDR4(n)|LmjF.28.0890	UD	E		SPHPLr|LmjF.30.2350	UD	E		PSDm_LM|LmjF.35.4590	UD	E
RNDR4(n)|LmjF.27.2050	UD	E		SPMS|LmjF.04.0580	UD	E		PSUDS|LmjF.26.0420	UD	E
RPE|LmjF.33.1570	UD	E		SQLMer|LmjF.13.1620	UD	E		PSUDS|LmjF.36.1660	UD	E
RPEg|LmjF.35.3680	UD	E		SQLS|LmjF.31.2940	UD	E		PSUDS|LmjF.30.1550	UD	E
RPI|LmjF.28.1970	UD	E		SQLSg|LmjF.31.2940	UD	E		PTHK|LmjF.28.0140	UD	E
S6PFH|LmjF.23.0870	UD	E		SRTMT|LmjF.12.1270	UD	E		PTROPACE|LmjF.25.0020	UD	E
S6PFH|LmjF.27.2340	UD	E		SRTMT|LmjF.23.1200	UD	E		PTROPACE|LmjF.05.0180	UD	E
S6PFH|LmjF.23.0880	UD	E		SSALym|LmjF.36.1760	UD	E		PUNP8I|LmjF.29.2800	UD	E
S6PFH|LmjF.04.0310	UD	E		STFH|LmjF.04.0310	UD	E		PYDAMK|LmjF.30.1250	UD	E
SAM24MTr|LmjF.36.2380	UD	E		STFH|LmjF.23.0870	UD	E		PYDXK|LmjF.30.1250	UD	E
SBPP1r|LmjF.32.2290	UD	E		STFH|LmjF.23.0880	UD	E		PYDXNK|LmjF.30.1250	UD	E
SBPP3|LmjF.19.1350	UD	E		STFH|LmjF.27.2340	UD	E		PYK|LmjF.35.0020	UD	E
SBPP3|LmjF.18.0440	UD	E		SUCD1rm|LmjF.24.1630	UD	E		PYK|LmjF.35.0030	UD	E
SBTD_D|LmjF.33.0520	UD	E		SUCD2_u6m|LmjF.15.0990	UD	E		PYRZAMn|LmjF.26.0210	UD	E
SERAT|LmjF.34.2850	UD	E		SUCOGDPm|LmjF.36.2950	UD	E		PYRZAMn|LmjF.34.2140	UD	E
SERTRS|LmjF.11.0100	UD	E		SUCOGDPm|LmjF.25.2140	UD	E		RAFFH|LmjF.23.0870	UD	E
SERt6|LmjF.11.0520	UD	E		SUCOGDPm|LmjF.25.2130	UD	E		RAFFH|LmjF.27.2340	UD	E
SERt6|LmjF.22.0230	UD	E		SUCR1|LmjF.23.0880	UD	E		RAFFH|LmjF.23.0880	UD	E
SERt6|LmjF.14.0320	UD	E		SUCR1|LmjF.23.0870	UD	E		RAFFH|LmjF.04.0310	UD	E
SERt6|LmjF.27.0670	UD	E		SUCR1|LmjF.27.2340	UD	E		RBK_Dg|LmjF.36.0060	UD	E

SGPL12r|LmjF.30.2350	UD	E	SUCR1|LmjF.04.0310	UD	E	RBLKg|LmjF.36.0060	UD	E
SHSL1|LmjF.35.3230	UD	E	TA6PK|LmjF.02.0030	UD	E	RNDR1(n)|LmjF.28.0890	UD	E
SHSL2r|LmjF.35.3230	UD	E	TA6PK|LmjF.25.2440	UD	E	RNDR1(n)|LmjF.27.2050	UD	E
SHSL4r|LmjF.35.3230	UD	E	TAL|LmjF.16.0760	UD	E	RNDR1(n)|LmjF.22.1290	UD	E
SINCOAL|LmjF.19.1005	UD	E	TCAFCOAL|LmjF.19.0985	UD	E	RNDR2(n)|LmjF.28.0890	UD	E
SINCOAL|LmjF.19.0985	UD	E	TCAFCOAL|LmjF.19.1005	UD	E	RNDR2(n)|LmjF.27.2050	UD	E
SLCBK1|LmjF.18.0440	UD	E	TCINCOAL|LmjF.19.1005	UD	E	RNDR2(n)|LmjF.22.1290	UD	E
SLCBK1|LmjF.19.1350	UD	E	TCINCOAL|LmjF.19.0985	UD	E	RNDR3(n)|LmjF.28.0890	UD	E
SLCYSS|LmjF.36.3590	UD	E	TDCOADm|LmjF.06.0880	UD	E	RNDR3(n)|LmjF.27.2050	UD	E
SO4t6|LmjF.28.1690	UD	E	TDCOADm|LmjF.28.2510	UD	E	RNDR3(n)|LmjF.22.1290	UD	E
SPHK21r|LmjF.32.2290	UD	E	TDPGDH|LmjF.26.2230	UD	E	RNDR4(n)|LmjF.22.1290	UD	E
TRPabc|LmjF.36.0420	UD	E	THDSTIL|LmjF.27.0090	UD	E	UDPDPS|LmjF.13.0020	UD	E
TRPabc|LmjF.31.1820	UD	E	THFAT|LmjF.36.3810	UD	E	UDPG4Ex|LmjF.33.2300	UD	E
TRPabc|LmjF.35.5360	UD	E	THFAT|LmjF.36.3800	UD	E	UDPGALM|LmjF.18.0200	UD	E
TRPabc|LmjF.31.1790	UD	E	THFATm|LmjF.36.3810	UD	E	UPPRTr|LmjF.34.1040	UD	E
TRPabc|LmjF.27.0680	UD	E	THFATm|LmjF.36.3800	UD	E	URIK1|LmjF.31.2470	UD	E
TRYP|LmjF.15.1120	UD	E	THFGLUS|LmjF.36.2610	UD	E	URIK2|LmjF.31.2470	UD	E
TRYP|LmjF.15.1060	UD	E	THRA|LmjF.01.0480	UD	E	URIK3|LmjF.31.2470	UD	E
TRYP|LmjF.15.1160	UD	E	THRLAD|LmjF.01.0480	UD	E	URIK4|LmjF.31.2470	UD	E
TRYP|LmjF.15.1080	UD	E	THRS|LmjF.14.0350	UD	E	URIK5|LmjF.31.2470	UD	E
TRYP|LmjF.15.1100	UD	E	THRTRS|LmjF.35.1410	UD	E	URIK6|LmjF.31.2470	UD	E
TRYP|LmjF.15.1140	UD	E	THRt6|LmjF.11.0520	UD	E	URIK7|LmjF.31.2470	UD	E
TRYPg|LmjF.15.1040	UD	E	THRt6|LmjF.27.0670	UD	E	URIK8|LmjF.31.2470	UD	E
TRYPm|LmjF.23.0040	UD	E	THRt6|LmjF.22.0230	UD	E	URIK9|LmjF.31.2470	UD	E
TRYS|LmjF.27.1870	UD	E	THRt6|LmjF.14.0320	UD	E	URIRHn|LmjF.18.0480	UD	E
TRYS|LmjF.23.0460	UD	E	TKT1|LmjF.24.2060	UD	E	VALTAm|LmjF.27.2030	UD	E
TRYS|LmjF.36.4300	UD	E	TKT1g|LmjF.24.2060	UD	E	VALTRS|LmjF.30.3130	UD	E
TYRTA|LmjF.36.2360	UD	E	TKT2|LmjF.24.2060	UD	E	VALt6|LmjF.35.4410	UD	E
TYRTA|LmjF.35.0820	UD	E	TKT2g|LmjF.24.2060	UD	E	XANMT|LmjF.23.1200	UD	E
TYRTA2|LmjF.35.0820	UD	E	TMDK1m|LmjF.21.1210	UD	E	XANMT|LmjF.12.1270	UD	E
TYRTA2|LmjF.36.2360	UD	E	TNMT|LmjF.12.1270	UD	E	XPRTgr|LmjF.21.0850	UD	E
TYRabc|LmjF.31.1800	UD	E	TNMT|LmjF.23.1200	UD	E	XYLKg|LmjF.36.0260	UD	E
TYRabc|LmjF.27.1580	UD	E	TPIg|LmjF.24.0850	UD	E			
TYRabc|LmjF.33.1420	UD	E	TROPACE|LmjF.05.0180	UD	E			
TYRabc|LmjF.36.0420	UD	E	TROPACE|LmjF.25.0020	UD	E			
TYRabc|LmjF.31.1820	UD	E	TRPTRS|LmjF.29.0060	UD	E			
TYRabc|LmjF.31.1790	UD	E	TRPTRS|LmjF.23.0300	UD	E			
TYRabc|LmjF.35.5360	UD	E	TRPabc|LmjF.35.5350	UD	E			
TYRabc|LmjF.27.0680	UD	E	TRPabc|LmjF.36.6830	UD	E			
TYRabc|LmjF.35.5350	UD	E	TRPabc|LmjF.31.1800	UD	E			
TYRabc|LmjF.36.6830	UD	E	TRPabc|LmjF.27.1580	UD	E			
UAGDPr|LmjF.33.2520	UD	E	TRPabc|LmjF.33.1420	UD	E			

Table B. 6: Gene Ontology (Molecular Function) terms of the predicted essential genes in *Leishmania donovani*

Gene ontology (Molecular Function)	Number of Genes	Gene List (Uniprot IDs)
ATP binding [GO:0005524]	11	[E9BKD6, E9BCR2, E9BG78, E9BUS2, E9BP61, E9BSR1, E9BDY2, E9BTY3, E9BI02, E9BTK0, E9BT25]
oxidoreductase activity [GO:0016491]	8	[E9BFX7, E9BJC7, E9BH73, E9BD53, E9BUR9, E9BPY2, E9BJC4, E9BKH5]
AMP deaminase activity [GO:0003876]	4	[E9B7Z3, E9BBG1, E9BP20, E9BT05]
flavin adenine dinucleotide binding [GO:0050660]	4	[E9B8V1, E9B911, E9BJZ0, E9BSE8]
metal ion binding [GO:0046872]	4	[E9BCR2, E9B8I8, E9BD53, E9BIZ2]
NADH dehydrogenase (ubiquinone) activity [GO:0008137]	3	[E9B8I8, E9BDX7, E9BIZ2]
catalytic activity [GO:0003824]	3	[E9BB03, E9BD62, E9BNX3]
pyridoxal phosphate binding [GO:0030170]	3	[E9BSJ9, E9BB38, E9BRW0]
2 iron, 2 sulfur cluster binding [GO:0051537]	2	[E9BS30, E9BD53]
4 iron, 4 sulfur cluster binding [GO:0051539]	2	[E9B8I8, E9BIZ2]
FAD binding [GO:0071949]	2	[E9BJC4, E9BKH5]
NAD+ kinase activity [GO:0003951]	2	[E9BI81, E9BT25]
acyl-CoA dehydrogenase activity [GO:0003995]	2	[E9B8V1, E9BSE8]
adenylate kinase activity [GO:0004017]	2	[E9BI02, E9BTK0]
diacylglycerol kinase activity [GO:0004143]	2	[E9BCY8, E9BT25]
electron transfer activity [GO:0009055]	2	[E9B917, E9BDX7]
hydrolase activity [GO:0016787]	2	[E9BH53, E9B7B1]
nucleoside diphosphate kinase activity [GO:0004550]	2	[E9BP61, E9BSR1]
oxidoreductase activity, acting on the CH-CH group of donors [GO:0016627]	2	[E9B911, E9BJZ0]
oxidoreductase activity, acting on the aldehyde or oxo group of donors, NAD or NADP as acceptor [GO:0016620]	2	[E9BM35, E9BUP9]

transaminase activity [GO:0008483]	2	[E9BB38, E9BRW0]
transferase activity, transferring glycosyl groups [GO:0016757]	2	[E9BQR2, E9BF85]
ubiquinol-cytochrome-c reductase activity [GO:0008121]	2	[E9BN35, E9BS30]
zinc ion binding [GO:0008270]	2	[E9BKN9, E9BPY2]
(S)-2-(5-amino-1-(5-phospho-D-ribosyl)imidazole-4-carboxamido)succinate AMP-lyase (fumarate-forming) activity [GO:0070626]	1	[E9B810]
3-beta-hydroxy-delta5-steroid dehydrogenase activity [GO:0003854]	1	[E9B8P6]
AMP binding [GO:0016208]	1	[E9BG78]
FMN binding [GO:0010181]	1	[E9B8I8]
N-acetyltransferase activity [GO:0008080]	1	[E9BHA6]
N6-(1,2-dicarboxyethyl)AMP AMP-lyase (fumarate-forming) activity [GO:0004018]	1	[E9B810]
NAD binding [GO:0051287]	1	[E9B8I8]
O-acyltransferase activity [GO:0008374]	1	[E9BQR8]
acetate-CoA ligase activity [GO:0003987]	1	[E9BG78]
acireductone synthase activity [GO:0043874]	1	[E9BUX4]
argininosuccinate synthase activity [GO:0004055]	1	[E9BCR2]
carbamoyl-phosphate synthase (glutamine-hydrolyzing) activity [GO:0004088]	1	[E9BCR2]
coproporphyrinogen oxidase activity [GO:0004109]	1	[E9B8Z2]
guanine deaminase activity [GO:0008892]	1	[E9BKN9]
heme binding [GO:0020037]	1	[E9B917]
iron-sulfur cluster binding [GO:0051536]	1	[E9BDX7]
kinase activity [GO:0016301]	1	[E9BT87]
lyase activity [GO:0016829]	1	[E9BSJ9]
magnesium ion binding [GO:0000287]	1	[E9BUX4]
nicotinate phosphoribosyltransferase activity [GO:0004516]	1	[E9BQ30]

nicotinate-nucleotide diphosphorylase (carboxylating) activity [GO:0004514]	1	[E9BQ30]
phosphotransferase activity, alcohol group as acceptor [GO:0016773]	1	[E9BT87]
proton-exporting ATPase activity, phosphorylative mechanism [GO:0008553]	1	[E9BDY2]
quinone binding [GO:0048038]	1	[E9BIZ2]
ribonucleoside-diphosphate reductase activity, thioredoxin disulfide as acceptor [GO:0004748]	1	[E9BKD6]
tetrahydrofolylpolyglutamate synthase activity [GO:0004326]	1	[E9BTY3]
transferase activity [GO:0016740]	1	[E9B822]
transferase activity, transferring acyl groups [GO:0016746]	1	[E9B8A8]

Table B. 7: Gene Ontology (Molecular Function) terms of the predicted essential genes in *Leishmania major*

Gene ontology (Molecular Function)	Number of Genes	Gene List (Uniprot IDs)
ATP binding [GO:0005524]	43	[Q9U1E1, E9AFL1, Q4Q614, Q4Q8H8, Q4Q273, Q4QD33, Q4QD34, Q4Q6V1, E9AEB3, Q4Q1G0, E9ADK8, Q4QBE4, Q4QDS1, Q4Q288, E9AC86, Q4QCT4, Q4Q2E8, Q4Q2N5, E9AE77, Q4Q7P6, Q4Q7S2, Q4Q598, Q4QIS7, Q4QH55, E9ACY7, Q4QG76, Q4Q0Y9, Q4Q8Q6, Q4QF34, Q4QH70, E9ADF8, E9ADF9, Q4QGX9, Q4QC75, Q4Q1C4, Q4Q5J5, Q4QDB1, Q4Q6X7, E9AEW4, E9ABZ4, Q4QFJ7, E9ACM5, Q4Q9G0]
magnesium ion binding [GO:0000287]	13	[Q4QCF1, E9ACN6, Q4QH59, Q4Q1C4, E9AEH9, E9AEI0, Q4QDB1, Q4QCC2, Q4Q2G5, Q4QIB8, Q4QI56, Q4Q0M2, Q4Q3Z4]
amino acid transmembrane transporter activity [GO:0015171]	12	[E9AD44, Q4QBX3, E9AG08, Q4Q072, Q4Q682, E9ADD7, Q4Q445, Q4Q236, Q4Q680, E9AG09, Q4Q683, E9AD45]

metal ion binding [GO:0046872]	10	[E9AEB3, Q4QAC4, Q4Q7B0, Q4QGX9, Q4QC75, Q4Q7P5, Q4Q431, E9ABZ4, E9AFJ2, Q4Q842]
protein serine/threonine kinase activity [GO:0004674]	10	[Q4Q288, E9AC86, Q4QCT4, Q4Q2E8, Q4Q7P6, Q4Q598, Q4QIS7, Q4QH55, E9ACY7, Q4Q0Y9]
flavin adenine dinucleotide binding [GO:0050660]	9	[Q4QFZ2, Q4QAG8, Q4QJG7, Q4Q5Z6, Q4Q4U1, E9AE44, Q4Q5Z7, Q4QIY9, Q4Q812]
kinase activity [GO:0016301]	7	[Q4Q614, E9ADF8, E9ADF9, Q4QGX9, E9AEH9, E9AEI0, Q4QIB8]
inorganic phosphate transmembrane transporter activity [GO:0005315]	6	[Q4QJH3, E9AFR9, E9AFS0, E9ACJ5, Q4QHL7, Q4QH82]
pyridoxal phosphate binding [GO:0030170]	6	[Q4Q1I5, Q4FX34, E9AFE7, Q4QAU4, Q4Q758, Q4Q159]
ADP binding [GO:0043531]	5	[Q4QD33, Q4QD34, Q4Q6V1, E9AEH9, E9AEI0]
electron transfer activity [GO:0009055]	5	[Q4QAG8, Q4QJG7, Q4Q5Z6, Q4Q4U1, E9AE44]
thioredoxin peroxidase activity [GO:0008379]	5	[Q4QBH2, Q4QF80, Q4QF68, Q4QF76, Q4QF74]
1-phosphatidylinositol-3-kinase activity [GO:0016303]	4	[Q4QAC9, Q4QCT4, Q4Q7B0, Q4Q2E8]
RNA binding [GO:0003723]	4	[Q4QDB1, Q4Q9E9, Q4Q1Q8, Q4Q7E9]
beta-fructofuranosidase activity [GO:0004564]	4	[Q4QB76, E9ACV4, Q4QB75, Q9XTP3]
hydrolase activity [GO:0016787]	4	[E9AG08, E9AG09, E9AFR8, Q4Q546]
ligase activity [GO:0016874]	4	[Q711P7, Q4QBC8, Q4Q0Y4, E9ABZ4]
oxidoreductase activity [GO:0016491]	4	[Q4QBL8, Q5EEK0, Q4QAG8, Q4Q7P5]
peroxiredoxin activity [GO:0051920]	4	[Q4QF80, Q4QF68, Q4QF76, Q4QF74]
ribose phosphate diphosphokinase activity [GO:0004749]	4	[Q4QIB8, Q4QI56, Q4Q0M2, Q4Q3Z4]

Table B. 8: KEGG Pathway enrichment of the predicted essential genes in *Leishmania donovani*

Term	*P*-Value	Genes
ldo01100:Metabolic pathways	1.25198E-11	LDBPK_230580, LDBPK_061330, LDBPK_352010, LDBPK_350100, LDBPK_181460, LDBPK_362740, LDBPK_290920, LDBPK_170410, LDBPK_341110, LDBPK_060910, LDBPK_271970, LDBPK_323110, LDBPK_280980, LDBPK_270590, LDBPK_270300, LDBPK_120580, LDBPK_050180, LDBPK_160590, LDBPK_070210, LDBPK_040440, LDBPK_272390, LDBPK_365650, LDBPK_353910, LDBPK_350840, LDBPK_040570, LDBPK_361410, LDBPK_331010, LDBPK_252480, LDBPK_050980, LDBPK_354860, LDBPK_355060, LDBPK_120105, LDBPK_322690, LDBPK_230880, LDBPK_170320, LDBPK_351540, LDBPK_312650
ldo00190:Oxidative phosphorylation	2.23446E-07	LDBPK_352010, LDBPK_181510, LDBPK_365620, LDBPK_070210, LDBPK_350100, LDBPK_050980, LDBPK_181460, LDBPK_170320, LDBPK_270590, LDBPK_351540, LDBPK_270300, LDBPK_312650
ldo00230:Purine metabolism	0.000647072	LDBPK_252480, LDBPK_361410, LDBPK_271970, LDBPK_040440, LDBPK_354860, LDBPK_323110, LDBPK_280980, LDBPK_322690, LDBPK_353910, LDBPK_290920
ldo01110:Biosynthesis of secondary metabolites	0.002288788	LDBPK_230580, LDBPK_050180, LDBPK_061330, LDBPK_040440, LDBPK_353910, LDBPK_350840, LDBPK_060910, LDBPK_361410, LDBPK_252480, LDBPK_354860, LDBPK_355060, LDBPK_120105, LDBPK_323110, LDBPK_230880, LDBPK_322690
ldo01130:Biosynthesis of antibiotics	0.004028807	LDBPK_230580, LDBPK_050180, LDBPK_252480, LDBPK_361410, LDBPK_060910, LDBPK_040440, LDBPK_354860, LDBPK_323110, LDBPK_120105, LDBPK_322690, LDBPK_230880, LDBPK_353910, LDBPK_350840
ldo00240:Pyrimidine metabolism	0.006067033	LDBPK_170410, LDBPK_160590, LDBPK_341110, LDBPK_271970, LDBPK_323110, LDBPK_280980, LDBPK_353910
ldo00250:Alanine, aspartate and glutamate metabolism	0.024948804	LDBPK_120580, LDBPK_160590, LDBPK_040440, LDBPK_350840
ldo00480:Glutathione metabolism	0.038593104	LDBPK_040570, LDBPK_271970, LDBPK_120105, LDBPK_280980
ldo00620:Pyruvate metabolism	0.082188661	LDBPK_230580, LDBPK_120200, LDBPK_271940, LDBPK_230880
ldo00330:Arginine and proline metabolism	0.098738561	LDBPK_040570, LDBPK_120105, LDBPK_350840
ldo00640:Propanoate metabolism	0.098738561	LDBPK_230580, LDBPK_060910, LDBPK_230880

Table B. 9: KEGG Pathway enrichment of the predicted essential genes in *Leishmania major*

Term	*P*-Value	Genes
lma01100: Metabolic pathways	8.16878E-39	LMJF_33_2300, LMJF_35_0030, LMJF_15_1060, LMJF_36_1760, LMJF_23_0880, LMJF_35_4590, LMJF_06_1280, LMJF_33_1570, LMJF_06_0880, LMJF_21_1430, LMJF_36_1960, LMJF_21_1210, LMJF_27_2340, LMJF_26_2480, LMJF_31_2940, LMJF_23_0870, LMJF_31_2640, LMJF_15_1120, LMJF_24_2060, LMJF_30_1250, LMJF_05_0180, LMJF_16_0550, LMJF_32_2950, LMJF_36_2360, LMJF_29_2800, LMJF_30_2350, LMJF_24_2010, LMJF_25_1710, LMJF_20_0110, LMJF_15_1080, LMJF_34_2850, LMJF_25_2140, LMJF_34_3590, LMJF_26_1620, LMJF_20_0100, LMJF_31_2650, LMJF_34_3780, LMJF_36_3590, LMJF_06_0860, LMJF_13_1620, LMJF_08_1130, LMJF_30_1540, LMJF_04_0580, LMJF_30_2950, LMJF_05_0350, LMJF_35_0020, LMJF_03_0200, LMJF_12_0530, LMJF_21_0550, LMJF_36_2660, LMJF_31_3120, LMJF_20_1120, LMJF_36_2610, LMJF_25_2130, LMJF_15_1100, LMJF_29_2510, LMJF_12_0280, LMJF_36_2380, LMJF_22_1290, LMJF_24_1630, LMJF_14_0350, LMJF_15_1160, LMJF_24_0370, LMJF_27_2050, LMJF_13_1680, LMJF_26_0210, LMJF_34_1040, LMJF_36_0260, LMJF_27_1870, LMJF_36_3810, LMJF_11_1000, LMJF_28_0890, LMJF_36_0060, LMJF_03_0030, LMJF_33_1930, LMJF_31_2470, LMJF_29_1830, LMJF_30_1850, LMJF_33_2520, LMJF_08_0060, LMJF_35_0820, LMJF_30_3380, LMJF_35_3340, LMJF_31_2290, LMJF_15_1040, LMJF_35_0050, LMJF_36_5390, LMJF_36_3800, LMJF_08_0510, LMJF_16_0760, LMJF_15_1140, LMJF_23_0040, LMJF_27_2030, LMJF_15_1460, LMJF_18_0440, LMJF_36_6650, LMJF_28_1970, LMJF_01_0050, LMJF_24_0850, LMJF_35_3020, LMJF_34_3090, LMJF_32_3310, LMJF_36_2950, LMJF_35_3680, LMJF_27_1805, LMJF_35_4250, LMJF_28_0140, LMJF_21_0640, LMJF_27_1810, LMJF_01_0480, LMJF_28_0490, LMJF_18_1380, LMJF_35_3870
lma01110:Biosynthesis of secondary metabolites	4.55232E-27	LMJF_35_0030, LMJF_36_2380, LMJF_24_1630, LMJF_14_0350, LMJF_35_4590, LMJF_06_1280, LMJF_24_0370, LMJF_33_1570, LMJF_06_0880, LMJF_21_1430, LMJF_13_1680, LMJF_36_1960, LMJF_36_3810, LMJF_31_2940, LMJF_31_2640, LMJF_33_1930, LMJF_29_1830, LMJF_24_2060, LMJF_05_0180, LMJF_08_0060, LMJF_35_0820, LMJF_32_2950, LMJF_30_3380, LMJF_35_3340, LMJF_31_2290, LMJF_36_2360, LMJF_35_0050, LMJF_36_3800, LMJF_36_5390, LMJF_08_0510, LMJF_25_1710, LMJF_20_0110, LMJF_16_0760, LMJF_34_2850, LMJF_25_2140, LMJF_27_2030, LMJF_15_1460, LMJF_18_0440, LMJF_20_0100, LMJF_26_1620, LMJF_36_6650, LMJF_28_1970, LMJF_31_2650, LMJF_34_3780, LMJF_36_3590, LMJF_08_1130, LMJF_13_1620, LMJF_24_0850, LMJF_35_3020, LMJF_35_0020, LMJF_32_3310, LMJF_21_0550, LMJF_12_0530, LMJF_36_2660, LMJF_36_2950, LMJF_35_3680, LMJF_27_1805, LMJF_35_4250, LMJF_31_3120, LMJF_21_0640, LMJF_27_1810, LMJF_25_2130, LMJF_01_0480, LMJF_18_1380, LMJF_35_3870, LMJF_29_2510, LMJF_12_0280
lma01130:Biosynthesis of antibiotics	1.40204E-27	LMJF_35_0030, LMJF_36_2380, LMJF_24_1630, LMJF_24_0370, LMJF_33_1570, LMJF_06_0880, LMJF_21_1430, LMJF_13_1680, LMJF_36_3810, LMJF_31_2940, LMJF_03_0030, LMJF_31_2640, LMJF_33_1930, LMJF_29_1830, LMJF_24_2060, LMJF_05_0180, LMJF_33_2520, LMJF_35_0820, LMJF_08_0060, LMJF_32_2950, LMJF_30_3380, LMJF_35_3340, LMJF_36_2360, LMJF_35_0050, LMJF_36_3800, LMJF_36_5390, LMJF_26_2230, LMJF_08_0510, LMJF_25_1710, LMJF_20_0110, LMJF_16_0760, LMJF_25_2140, LMJF_27_2030, LMJF_15_1460, LMJF_20_0100, LMJF_36_6650, LMJF_28_1970, LMJF_31_2650, LMJF_34_3780, LMJF_01_0050,

		LMJF_36_3590, LMJF_13_1620, LMJF_08_1130, LMJF_24_0850, LMJF_35_3020, LMJF_35_0020, LMJF_21_0550, LMJF_12_0530, LMJF_32_3310, LMJF_36_2660, LMJF_36_2950, LMJF_35_3680, LMJF_27_1805, LMJF_21_0640, LMJF_27_1810, LMJF_25_2130, LMJF_01_0480, LMJF_28_0490, LMJF_18_1380, LMJF_35_3870, LMJF_29_2510, LMJF_12_0280
lma01200:Carbon metabolism	2.24838E-24	LMJF_30_3380, LMJF_35_3340, LMJF_35_0030, LMJF_36_5390, LMJF_36_3800, LMJF_08_0510, LMJF_24_1630, LMJF_25_1710, LMJF_20_0110, LMJF_16_0760, LMJF_24_0370, LMJF_33_1570, LMJF_06_0880, LMJF_34_2850, LMJF_25_2140, LMJF_20_0100, LMJF_36_6650, LMJF_28_1970, LMJF_31_2650, LMJF_01_0050, LMJF_36_3590, LMJF_08_1130, LMJF_24_0850, LMJF_36_3810, LMJF_35_0020, LMJF_11_1000, LMJF_12_0530, LMJF_32_3310, LMJF_21_0550, LMJF_36_2660, LMJF_36_2950, LMJF_35_3680, LMJF_27_1805, LMJF_03_0030, LMJF_31_2640, LMJF_33_1930, LMJF_29_1830, LMJF_27_1810, LMJF_25_2130, LMJF_24_2060, LMJF_28_0490, LMJF_18_1380, LMJF_08_0060, LMJF_35_0820, LMJF_29_2510
lma01230:Biosynthesis of amino acids	6.17941E-13	LMJF_30_3380, LMJF_35_0030, LMJF_36_2360, LMJF_36_5390, LMJF_08_0510, LMJF_20_0110, LMJF_16_0760, LMJF_14_0350, LMJF_24_0370, LMJF_33_1570, LMJF_34_2850, LMJF_27_2030, LMJF_13_1680, LMJF_36_6650, LMJF_20_0100, LMJF_28_1970, LMJF_36_3590, LMJF_08_1130, LMJF_24_0850, LMJF_35_0020, LMJF_35_3680, LMJF_03_0030, LMJF_33_1930, LMJF_24_2060, LMJF_01_0480, LMJF_35_0820, LMJF_08_0060, LMJF_29_2510
lma00010:Glycolysis / Gluconeogenesis	5.84107E-12	LMJF_30_3380, LMJF_35_0030, LMJF_35_0020, LMJF_32_3310, LMJF_12_0530, LMJF_21_0550, LMJF_36_2660, LMJF_25_1710, LMJF_20_0110, LMJF_27_1805, LMJF_31_2640, LMJF_21_0640, LMJF_29_1830, LMJF_27_1810, LMJF_36_6650, LMJF_20_0100, LMJF_31_2650, LMJF_34_3780, LMJF_18_1380, LMJF_24_0850, LMJF_08_0060, LMJF_29_2510
lma00480:Glutathione metabolism	9.24113E-09	LMJF_35_3340, LMJF_05_0350, LMJF_15_1040, LMJF_27_1870, LMJF_15_1060, LMJF_28_0890, LMJF_22_1290, LMJF_15_1160, LMJF_15_1080, LMJF_15_1140, LMJF_23_0040, LMJF_27_2050, LMJF_15_1120, LMJF_04_0580, LMJF_15_1100, LMJF_12_0280
lma00230:Purine metabolism	0.035214819	LMJF_35_0030, LMJF_36_5390, LMJF_29_2800, LMJF_35_0020, LMJF_28_0890, LMJF_08_0510, LMJF_22_1290, LMJF_33_1930, LMJF_27_2050, LMJF_21_0640, LMJF_34_3780, LMJF_08_1130, LMJF_35_3870, LMJF_32_2950
lma00020:Citrate cycle (TCA cycle)	2.30064E-06	LMJF_31_2640, LMJF_29_1830, LMJF_27_1810, LMJF_25_2140, LMJF_25_2130, LMJF_31_2650, LMJF_32_3310, LMJF_21_0550, LMJF_36_2660, LMJF_36_2950, LMJF_18_1380, LMJF_25_1710, LMJF_24_1630, LMJF_27_1805
lma00280:Valine, leucine and isoleucine degradation	5.07784E-07	LMJF_35_0050, LMJF_32_3310, LMJF_31_2640, LMJF_30_1940, LMJF_29_1830, LMJF_06_0880, LMJF_21_1430, LMJF_27_2030, LMJF_31_2650, LMJF_05_0180, LMJF_28_0490, LMJF_01_0050, LMJF_30_1930, LMJF_33_2340
lma00030:Pentose phosphate pathway	1.43273E-09	LMJF_35_3340, LMJF_36_5390, LMJF_12_0530, LMJF_08_0510, LMJF_16_0760, LMJF_35_3680, LMJF_33_1930, LMJF_33_1570, LMJF_21_0640, LMJF_28_1970, LMJF_24_2060, LMJF_34_3780, LMJF_08_1130, LMJF_29_2510

lma00620:Pyruvate metabolism	1.59149E-05	LMJF_35_0030, LMJF_35_0020, LMJF_11_1000, LMJF_32_3310, LMJF_21_0550, LMJF_36_2660, LMJF_25_1710, LMJF_27_1805, LMJF_31_2640, LMJF_29_1830, LMJF_27_1810, LMJF_31_2650, LMJF_18_1380
lma00260:Glycine, serine and threonine metabolism	4.21184E-06	LMJF_31_2640, LMJF_29_1830, LMJF_36_3800, LMJF_36_3810, LMJF_36_6650, LMJF_31_2650, LMJF_01_0480, LMJF_32_3310, LMJF_08_0060, LMJF_14_0350, LMJF_03_0030
lma00240:Pyrimidine metabolism	0.046498998	LMJF_31_2470, LMJF_27_2050, LMJF_28_0890, LMJF_06_0860, LMJF_22_1290, LMJF_35_3870, LMJF_16_0550, LMJF_21_1210, LMJF_32_2950, LMJF_34_1040
lma00630:Glyoxylate and dicarboxylate metabolism	0.01064073	LMJF_31_2640, LMJF_29_1830, LMJF_36_3800, LMJF_36_3810, LMJF_31_2650, LMJF_32_3310, LMJF_01_0050, LMJF_28_0490
lma04070:Phosphatidylinositol signaling system	0.0019012	LMJF_34_3590, LMJF_26_1620, LMJF_34_3090, LMJF_30_1850, LMJF_24_2010, LMJF_26_2480, LMJF_30_2950, LMJF_20_1120
lma00562:Inositol phosphate metabolism	0.000202568	LMJF_34_3590, LMJF_34_3090, LMJF_30_1850, LMJF_24_2010, LMJF_26_2480, LMJF_24_0850, LMJF_30_2950, LMJF_20_1120
lma00052:Galactose metabolism	0.003688894	LMJF_33_2300, LMJF_21_0640, LMJF_34_3780, LMJF_23_0880, LMJF_27_2340, LMJF_29_2510, LMJF_23_0870
lma00564:Glycerophospholipid metabolism	0.031259793	LMJF_31_2290, LMJF_26_1620, LMJF_18_0440, LMJF_26_2480, LMJF_35_4590, LMJF_31_3120
lma00330:Arginine and proline metabolism	0.010260779	LMJF_24_0370, LMJF_13_1680, LMJF_03_0200, LMJF_35_0820, LMJF_04_0580, LMJF_12_0280

References

1. Juhas M, Eberl L, Glass JI. Essence of life: Essential genes of minimal genomes. Trends Cell Biol. 2011;21: 562–568. doi:10.1016/j.tcb.2011.07.005

2. Cohen O, Oberhardt M, Yizhak K, Ruppin E. Essential Genes Embody Increased Mutational Robustness to Compensate for the Lack of Backup Genetic Redundancy. PLoS One. 2016;11: e0168444. doi:10.1371/journal.pone.0168444

3. Joyce AR, Reed JL, White A, Edwards R, Osterman A, Baba T, *et al.* Experimental and Computational Assessment of Conditionally Essential Genes in *Escherichia coli.* J Bacteriol. 2006;188: 8259–8271. doi:10.1128/JB.00740-06

4. Ding T, Case KA, Omolo MA, Reiland HA, Metz ZP, Diao X, *et al.* Predicting Essential Metabolic Genome Content of Niche-Specific Enterobacterial Human Pathogens during Simulation of Host Environments. PLoS One. 2016;11: e0149423. doi:10.1371/journal.pone.0149423

5. Juhas M, Eberl L, Church GM. Essential genes as antimicrobial targets and cornerstones of synthetic biology. Trends Biotechnol. 2012;30: 601–607. doi:10.1016/j.tibtech.2012.08.002

6. Zhang Z, Ren Q. Why are essential genes essential? - The essentiality of *Saccharomyces genes*. Microb Cell. 2015;2: 280–287. doi:10.15698/mic2015.08.218

7. Chopra I. Bacterial RNA polymerase: a promising target for the discovery of new antimicrobial agents. Curr Opin Investig Drugs. 2007;8: 600–7. Available: http://www.ncbi.nlm.nih.gov/pubmed/17668362

8. Hu W, Sillaots S, Lemieux S, Davison J, Kauffman S, Breton A, *et al.* Essential Gene Identification and Drug Target Prioritization in *Aspergillus fumigatus*. PLoS Pathog. 2007;3: e24. doi:10.1371/journal.ppat.0030024

9. Bixel K, Hays J. Olaparib in the management of ovarian cancer. Pharmgenomics Pers Med. 2015;8: 127. doi:10.2147/PGPM.S62809

10. Baba T, Ara T, Hasegawa M, Takai Y, Okumura Y, Baba M, *et al.* Construction of *Escherichia coli* K-12 in-frame, single-gene knockout mutants: the Keio collection. Mol Syst Biol. 2006;2: 8–2006. doi:10.1038/msb4100050

11. Kim D-U, Hayles J, Kim D, Wood V, Park H-O, Won M, *et al.* Analysis of a genome-wide set of gene deletions in the fission yeast *Schizosaccharomyces pombe*. Nat Biotechnol. 2010;28: 617–623. doi:10.1038/nbt.1628

12. Papp B, Pál C, Hurst LD. Metabolic network analysis of the causes and evolution of enzyme dispensability in yeast. Nature. 2004;429: 661–664. doi:10.1038/nature02636

13. Dong X, Quinn PJ, Wang X. Metabolic engineering of *Escherichia coli* and *Corynebacterium glutamicum* for the production of L-threonine. Biotechnol Adv. 2011;29: 11–23. doi:10.1016/j.biotechadv.2010.07.009

14. Moritz B, Striegel K, de Graaf AA, Sahm H. Kinetic properties of the glucose-6-phosphate and 6 phosphogluconate dehydrogenases from *Corynebacterium glutamicum* and their application for predicting pentose phosphate pathway flux in vivo. Eur J Biochem. 2000;267: 3442–3452. doi:10.1046/j.1432-1327.2000.01354.x

References

15. Baba T, Ara T, Hasegawa M, Takai Y, Okumura Y, Baba M, *et al.* Construction of *Escherichia coli* K-12 in-frame, single-gene knockout mutants: the Keio collection. Mol Syst Biol. 2006;2: 2006.0008. doi:10.1038/msb4100050

16. Cruz A, Coburn CM, Beverley SM. Double targeted gene replacement for creating null mutants. Proc Natl Acad Sci. 1991;88: 7170–7174. doi:10.1073/pnas.88.16.7170

17. Gerdes S, Scholle MD, Campbell JW, Balázsi G, Ravasz E, Daugherty MD, *et al.* Experimental determination and system level analysis of essential genes in *Escherichia coli* MG1655. J Bacteriol. 2003;185: 5673–5684. doi:10.1128/JB.185.19.5673

18. Reznikoff WS, Winterberg KM. Transposon-Based Strategies for the Identification of Essential Bacterial Genes. Microbial Gene Essentiality: Protocols and Bioinformatics. Springer; 2008. pp. 13–26. doi:10.1007/978-1-59745-321-9_2

19. Agrawal N, Dasaradhi PVN, Mohmmed A, Malhotra P, Bhatnagar RK, Mukherjee SK. RNA Interference: Biology, Mechanism, and Applications. Microbiol Mol Biol Rev. 2003;67: 657–685. doi:10.1128/MMBR.67.4.657-685.2003

20. Hare RS, Walker SS, Dorman TE, Greene JR, Guzman L-M, Kenney TJ, *et al.* Genetic Footprinting in Bacteria. J Bacteriol. 2001;183: 1694–1706. doi:10.1128/JB.183.5.1694-1706.2001

21. Smith V, Botstein D, Brown PO. Genetic footprinting: a genomic strategy for determining a gene's function given its sequence. Proc Natl Acad Sci. 1995;92: 6479–6483. doi:10.1073/pnas.92.14.6479

22. Moore RC, Redhead NJ, Selfridge J, Hope J, Manson JC, Melton DW. Double Replacement Gene Targeting for the Production of a Series of Mouse Strains with Different Prion Protein Gene Alterations. Nat Biotechnol. 1995;13: 999–1004. doi:10.1038/nbt0995-999

23. Kan Y, Ruis B, Lin S, Hendrickson EA. The Mechanism of Gene Targeting in Human Somatic Cells. PLoS Genet. 2014;10: e1004251. doi:10.1371/journal.pgen.1004251

24. Snouwaert JN, Brigman KK, Latour AM, Malouf NN, Boucher RC, Smithies O, *et al.* An Animal Model for Cystic Fibrosis Made by Gene Targeting. Science (80). 1992;257: 1083–1088. doi:10.1126/science.257.5073.1083

25. Rabus R, Venceslau SS, Wöhlbrand L, Voordouw G, Wall JD, Pereira IAC. A Post-Genomic View of the Ecophysiology, Catabolism and Biotechnological Relevance of Sulphate-Reducing Prokaryotes. Advances in microbial physiology. Elsevier; 2015. pp. 55–321. doi:10.1016/bs.ampbs.2015.05.002

26. Brune W, Ménard C, Hobom U, Odenbreit S, Messerle M, Koszinowski UH. Rapid identification of essential and nonessential herpesvirus genes by direct transposon mutagenesis. Nat Biotechnol. 1999;17: 360–364. doi:10.1038/7914

27. Marker S, Carradec Q, Tanty V, Arnaiz O, Meyer E. A forward genetic screen reveals essential and non-essential RNAi factors in *Paramecium tetraurelia*. Nucleic Acids Res. 2014;42: 7268–7280. doi:10.1093/nar/gku223

28. Qin Z, Johnsen R, Yu S, Chu JS-C, Baillie DL, Chen N. Genomic Identification and Functional Characterization of Essential Genes in *Caenorhabditis elegans*. G3 Genes|Genomes|Genetics. 2018;8: 981–997. doi:10.1534/g3.117.300338

29. Doudna JA, Charpentier E. The new frontier of genome engineering with CRISPR-Cas9. Science (80). 2014;346: 1258096. doi:10.1126/science.1258096

References

30. Evers B, Jastrzebski K, Heijmans JPM, Grernrum W, Beijersbergen RL, Bernards R. CRISPR knockout screening outperforms shRNA and CRISPRi in identifying essential genes. Nat Biotechnol. 2016;34: 631–633. doi:10.1038/nbt.3536

31. Bartha I, di Iulio J, Venter JC, Telenti A. Human gene essentiality. Nat Rev Genet. 2018;19: 51–62. doi:10.1038/nrg.2017.75

32. Zhang R, Lin Y. DEG 5.0, a database of essential genes in both prokaryotes and eukaryotes. Nucleic Acids Res. 2009;37: D455–D458. doi:10.1093/nar/gkn858

33. Chen W-H, Minguez P, Lercher MJ, Bork P. OGEE: an online gene essentiality database. Nucleic Acids Res. 2012;40: D901–D906. doi:10.1093/nar/gkr986

34. Overbeek R. The Subsystems Approach to Genome Annotation and its Use in the Project to Annotate 1000 Genomes. Nucleic Acids Res. 2005;33: 5691–5702. doi:10.1093/nar/gki866

35. Ye Y-N, Hua Z-G, Huang J, Rao N, Guo F-B. CEG: a database of essential gene clusters. BMC Genomics. 2013;14: 769. doi:10.1186/1471-2164-14-769

36. Li X, Li W, Zeng M, Zheng R, Li M. Network-based methods for predicting essential genes or proteins: a survey. Brief Bioinform. 2020;21: 566–583. doi:10.1093/bib/bbz017

37. Zhang X, Acencio ML, Lemke N. Predicting essential genes and proteins based on machine learning and network topological features: A comprehensive review. Front Physiol. 2016;7: 1–11. doi:10.3389/fphys.2016.00075

38. Peng C, Lin Y, Luo H, Gao F. A Comprehensive Overview of Online Resources to Identify and Predict Bacterial Essential Genes. Front Microbiol. 2017;8: 2331. doi:10.3389/fmicb.2017.02331

39. Liu W, Fang L, Li M, Li S, Guo S, Luo R, *et al.* Comparative Genomics of *Mycoplasma*: Analysis of Conserved Essential Genes and Diversity of the Pan-Genome. PLoS One. 2012;7: e35698. doi:10.1371/journal.pone.0035698

40. Rout S, Warhurst DC, Suar M, Mahapatra RK. *In silico* comparative genomics analysis of *Plasmodium falciparum* for the identification of putative essential genes and therapeutic candidates. J Microbiol Methods. 2015;109: 1–8. doi:10.1016/j.mimet.2014.11.016

41. Yang X, Li Y, Zang J, Li Y, Bie P, Lu Y, *et al.* Analysis of pan-genome to identify the core genes and essential genes of *Brucella spp*. Mol Genet Genomics. 2016;291: 905–912. doi:10.1007/s00438-015-1154-z

42. Jordan IK, Rogozin IB, Wolf YI, Koonin E V. Essential Genes Are More Evolutionarily Conserved Than Are Nonessential Genes in Bacteria. Genome Res. 2002;12: 962–968. doi:10.1101/gr.87702

43. Guo F-B, Dong C, Hua H-L, Liu S, Luo H, Zhang H-W, *et al.* Accurate prediction of human essential genes using only nucleotide composition and association information. Bioinformatics. 2017;33: 1758–1764. doi:10.1093/bioinformatics/btx055

44. Lu Y, Deng J, Carson M, Lu H, Lu L. Computational Methods for the Prediction of Microbial Essential Genes. Curr Bioinform. 2014;9: 89–101. doi:10.2174/15748936089999140109113434

45. Joyce AR, Palsson BØ. Predicting Gene Essentiality Using Genome-Scale *in silico* Models. Microbial Gene Essentiality: Protocols and Bioinformatics. Springer; 2008. pp. 433–457. doi:10.1007/978-1-59745-321-9_30

46. Basler G. Computational Prediction of Essential Metabolic Genes Using Constraint-Based Approaches. Gene Essentiality. Springer; 2015. pp. 183–204. doi:10.1007/978-1-4939-2398-4_12

References

47. Witten IH, Frank E, Hall MA, Pal CJ. Data Mining: Practical Machine Learning Tools and Techniques. Elsevier; 2011. doi:10.1016/C2009-0-19715-5

48. Ning LW, Lin H, Ding H, Huang J, Rao N, Guo FB. Predicting bacterial essential genes using only sequence composition information. Genet Mol Res. 2014;13: 4564–4572. doi:10.4238/2014.June.17.8

49. Nigatu D, Sobetzko P, Yousef M, Henkel W. Sequence-based information-theoretic features for gene essentiality prediction. BMC Bioinformatics. 2017;18: 473. doi:10.1186/s12859-017-1884-5

50. Yu Y, Yang L, Liu Z, Zhu C. Gene essentiality prediction based on fractal features and machine learning. Mol Biosyst. 2017;13: 577–584. doi:10.1039/C6MB00806B

51. Azhagesan K, Ravindran B, Raman K. Network-based features enable prediction of essential genes across diverse organisms. PLoS One. 2018;13: e0208722. doi:10.1371/journal.pone.0208722

52. Hwang Y-C, Lin C-C, Chang J-Y, Mori H, Juan H-F, Huang H-C. Predicting essential genes based on network and sequence analysis. Mol Biosyst. 2009;5: 1672. doi:10.1039/b900611g

53. Plaimas K, Mallm J-P, Oswald M, Svara F, Sourjik V, Eils R, *et al.* Machine learning based analyses on metabolic networks supports high-throughput knockout screens. BMC Syst Biol. 2008;2: 67. doi:10.1186/1752-0509-2-67

54. Plaimas K, Eils R, König R. Identifying essential genes in bacterial metabolic networks with machine learning methods. BMC Syst Biol. 2010;4: 56. doi:10.1186/1752-0509-4-56

55. Subramanian A, Sarkar RR. Network structure and enzymatic evolution in *Leishmania* metabolism: a computational study. BIOMAT 2015. World Scientific; 2016. pp. 1–20. doi:10.1142/9789813141919_0001

56. del Rio G, Koschützki D, Coello G. How to identify essential genes from molecular networks? BMC Syst Biol. 2009;3: 102. doi:10.1186/1752-0509-3-102

57. Nandi S, Subramanian A, Sarkar RR. An integrative machine learning strategy for improved prediction of essential genes in *Escherichia coli* metabolism using flux-coupled features. Mol Biosyst. 2017;13: 1584–1596. doi:10.1039/C7MB00234C

58. Hira ZM, Gillies DF. A Review of Feature Selection and Feature Extraction Methods Applied on Microarray Data. Adv Bioinformatics. 2015;2015: 1–13. doi:10.1155/2015/198363

59. Pearson K. LIII. On lines and planes of closest fit to systems of points in space. London, Edinburgh, Dublin Philos Mag J Sci. 1901;2: 559–572. doi:10.1080/14786440109462720

60. Torgerson WS. Multidimensional scaling: I. Theory and method. Psychometrika. 1952;17: 401–419. Available: https://link.springer.com/article/10.1007/BF02288916

61. Hinton G, Roweis ST. Stochastic neighbor embedding. NIPS. 2002. pp. 833–840. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.441.8882&rep=rep1&type=pdf

62. Petegrosso R, Li Z, Kuang R. Machine learning and statistical methods for clustering single-cell RNA-sequencing data. Brief Bioinform. 2020;21: 1209–1223. doi:10.1093/bib/bbz063

63. Dey A. Machine learning algorithms: a review. Int J Comput Sci Inf Technol. 2016;7: 1174–1179. Available: http://ijcsit.com/docs/Volume 7/vol7issue3/ijcsit2016070332.pdf

64. Kotsiantis SB, Zaharakis I, Pintelas P. Supervised machine learning: A review of classification techniques. Emerg Artif Intell Appl Comput Eng. 2007;160: 3–24. Available: https://dl.acm.org/doi/10.5555/1566770.1566773

References

65. Cortes C, Vapnik V. Support-vector networks. Mach Learn. 1995;20: 273–297. doi:10.1007/BF00994018

66. Rish I, others. An empirical study of the naive Bayes classifier. IJCAI 2001 workshop on empirical methods in artificial intelligence. 2001. pp. 41–46. Available: https://www.cc.gatech.edu/~isbell/reading/papers/Rish.pdf

67. Hagan MT, Demuth HB, Beale M. Neural network design. PWS Publishing Co.; 1997. Available: https://hagan.okstate.edu/nnd.html

68. Aha DW, Kibler D, Albert MK. Instance-based learning algorithms. Mach Learn. 1991;6: 37–66. doi:10.1007/BF00153759

69. Breiman L, Friedman JH, Olshen RA, Stone CJ. Classification And Regression Trees. Boca Raton, Florida. Routledge; 2017. doi:10.1201/9781315139470

70. Ditterrich TG. Machine learning research: four current direction. Artif Intell Magzine. 1997;4: 97–136. doi:10.1609/aimag.v18i4.1324

71. Breiman L. Random forests. Mach Learn. 2001;45: 5–32. doi:https://doi.org/10.1023/A:1010933404324

72. Clark P, Niblett T. The CN2 induction algorithm. Mach Learn. 1989;3: 261–283. doi:10.1007/BF00116835

73. Chapelle O, Scholkopf B, Zien, Eds. A. Semi-Supervised Learning (Chapelle, O. *et al.*, Eds.; 2006) [Book reviews]. IEEE Trans Neural Networks. 2009;20: 542–542. doi:10.1109/TNN.2009.2015974

74. Belkin M, Niyogi P, Sindhwani V. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. J Mach Learn Res. 2006;7: 2399–2434. doi:10.5555/1248547.1248632

75. Deng J, Deng L, Su S, Zhang M, Lin X, Wei L, *et al.* Investigating the predictability of essential genes across distantly related organisms using an integrative approach. Nucleic Acids Res. 2011;39: 795–807. doi:10.1093/nar/gkq784

76. Cheng J, Wu W, Zhang Y, Li X, Jiang X, Wei G, *et al.* A new computational strategy for predicting essential genes. BMC Genomics. 2013;14: 910. doi:10.1186/1471-2164-14-910

77. Chen L, Zhang Y-H, Wang S, Zhang Y, Huang T, Cai Y-D. Prediction and analysis of essential genes using the enrichments of gene ontology and KEGG pathways. PLoS One. 2017;12: e0184129. doi:10.1371/journal.pone.0184129

78. Qin C, Sun Y, Dong Y. A new computational strategy for identifying essential proteins based on network topological properties and biological information. PLoS One. 2017;12: e0182031. doi:10.1371/journal.pone.0182031

79. Gustafson AM, Snitkin ES, Parker SC, DeLisi C, Kasif S. Towards the identification of essential genes using targeted genome sequencing and comparative analysis. BMC Genomics. 2006;7: 265. doi:10.1186/1471-2164-7-265

80. Saha S, Heber S. *In silico* prediction of yeast deletion phenotypes. Genet Mol Res. 2006;5: 224–32. Available: http://www.ncbi.nlm.nih.gov/pubmed/16755513

81. Jin S, Zeng X, Xia F, Huang W, Liu X. Application of deep learning methods in biological networks. Brief Bioinform. 2021;22: 1902–1917. doi:10.1093/bib/bbaa043

82. Hasan MA, Lonardi S. DeeplyEssential: a deep neural network for predicting essential genes in microbes. BMC Bioinformatics. 2020;21: 367. doi:10.1186/s12859-020-03688-y

References

83. Chen Y, Xu D. Understanding protein dispensability through machine-learning analysis of high-throughput data. Bioinformatics. 2005;21: 575–581. doi:10.1093/bioinformatics/bti058

84. Seringhaus M. Predicting essential genes in fungal genomes. Genome Res. 2006;16: 1126–1135. doi:10.1101/gr.5144106

85. da Silva JPM, Acencio ML, Mombach JCM, Vieira R, da Silva JC, Lemke N, *et al. In silico* network topology-based prediction of gene essentiality. Phys A Stat Mech its Appl. 2008;387: 1049–1055. doi:10.1016/j.physa.2007.10.044

86. Acencio ML, Lemke N. Towards the prediction of essential genes by integration of network topology, cellular localization and biological process information. BMC Bioinformatics. 2009;10: 290. doi:10.1186/1471-2105-10-290

87. Yuan Y, Xu Y, Xu J, Ball RL, Liang H. Predicting the lethal phenotype of the knockout mouse by integrating comprehensive genomic data. Bioinformatics. 2012;28: 1246–1252. doi:10.1093/bioinformatics/bts120

88. Li M, Wang J-X, Wang H PY. Identification of essential proteins from weighted protein–protein interaction networks. J Bioinform Comput Biol. 2013;11: 1341002. doi:10.1142/S0219720013410023

89. Cheng J, Xu Z, Wu W, Zhao L, Li X, Liu Y, *et al.* Training Set Selection for the Prediction of Essential Genes. PLoS One. 2014;9: e86805. doi:10.1371/journal.pone.0086805

90. Lu Y, Deng J, Rhodes JC, Lu H, Lu LJ. Predicting essential genes for identifying potential drug targets in *Aspergillus fumigatus*. Comput Biol Chem. 2014;50: 29–40. doi:10.1016/j.compbiolchem.2014.01.011

91. Yang L, Wang J, Wang H, Lv Y, Zuo Y, Li X, *et al.* Analysis and identification of essential genes in humans using topological properties and biological information. Gene. 2014;551: 138–151. doi:10.1016/j.gene.2014.08.046

92. Lloyd JP, Seddon AE, Moghe GD, Simenc MC, Shiu S-H. Characteristics of Plant Essential Genes Allow for within- and between-Species Prediction of Lethal Mutant Phenotypes. Plant Cell. 2015;27: 2133–2147. doi:10.1105/tpc.15.00051

93. Hua H-L, Zhang F-Z, Labena AA, Dong C, Jin Y-T, Guo F-B. An Approach for Predicting Essential Genes Using Multiple Homology Mapping and Machine Learning Algorithms. Biomed Res Int. 2016;2016: 1–9. doi:10.1155/2016/7639397

94. Lin Y, Zhang F-Z, Xue K, Gao Y-Z, Guo F-B. Identifying Bacterial Essential Genes Based on a Feature-Integrated Method. IEEE/ACM Trans Comput Biol Bioinforma. 2019;16: 1274–1279. doi:10.1109/TCBB.2017.2669968

95. Li Y, Lv Y, Li X, Xiao W, Li C. Sequence comparison and essential gene identification with new inter-nucleotide distance sequences. J Theor Biol. 2017;418: 84–93. doi:10.1016/j.jtbi.2017.01.031

96. Liu X, Wang B-J, Xu L, Tang H-L, Xu G-Q. Selection of key sequence-based features for prediction of essential genes in 31 diverse bacterial species. PLoS One. 2017;12: e0174638. doi:10.1371/journal.pone.0174638

97. Aromolaran O, Beder T, Oswald M, Oyelade J, Adebiyi E, Koenig R. Essential gene prediction in *Drosophila melanogaster* using machine learning approaches based on sequence and functional features. Comput Struct Biotechnol J. 2020;18: 612–621. doi:10.1016/j.csbj.2020.02.022

References

98. Wei W, Ning L-W, Ye Y-N, Guo F-B. Geptop: A Gene Essentiality Prediction Tool for Sequenced Bacterial Genomes Based on Orthology and Phylogeny. PLoS One. 2013;8: e72343. doi:10.1371/journal.pone.0072343

99. Guo F-B, Ning L-W, Huang J, Lin H, Zhang H-X. Chromosome translocation and its consequence in the genome of *Burkholderia cenocepacia* AU-1054. Biochem Biophys Res Commun. 2010;403: 375–379. doi:10.1016/j.bbrc.2010.11.039

100. Guo F-B, Ye Y-N, Ning L-W, Wei W. Three Computational Tools for Predicting Bacterial Essential Genes. Gene Essentiality. Springer; 2015. pp. 205–217. doi:10.1007/978-1-4939-2398-4_13

101. Guo F-B. ZCURVE: a new system for recognizing protein-coding genes in bacterial and archaeal genomes. Nucleic Acids Res. 2003;31: 1780–1789. doi:10.1093/nar/gkg254

102. Hua Z-G, Lin Y, Yuan Y-Z, Yang D-C, Wei W, Guo F-B. ZCURVE 3.0: identify prokaryotic genes with higher accuracy as well as automatically and accurately select essential genes. Nucleic Acids Res. 2015;43: W85–W90. doi:10.1093/nar/gkv491

103. Theodoridis S, Pikrakis A, Koutroumbas K, Cavouras D. Introduction to Pattern Recognition. Elsevier; 2010. doi:10.1016/C2009-0-18558-6

104. Maalouf M, Trafalis TB. Robust weighted kernel logistic regression in imbalanced and rare events data. Comput Stat Data Anal. 2011;55: 168–183. doi:10.1016/j.csda.2010.06.014

105. Sofeikov KI, Tyukin IY, Gorban AN, Mirkes EM, Prokhorov D V, Romanenko I V. Learning optimization for decision tree classification of non-categorical data with information gain impurity criterion. 2014 International Joint Conference on Neural Networks (IJCNN). IEEE; 2014. pp. 3548–3555. doi:10.1109/IJCNN.2014.6889842

106. Tan P-N, Steinbach M, Kumar V. Classification: Alternative Techniques. Introd to data Min. 2013. Available: https://www-users.cs.umn.edu/~kumar001/dmbook/sol.pdf

107. Akbani R, Kwek S, Japkowicz N. Applying Support Vector Machines to Imbalanced Datasets. European conference on machine learning. 2004. pp. 39–50. doi:10.1007/978-3-540-30115-8_7

108. Raman K, Damaraju N, Joshi GK. The organisational structure of protein networks: revisiting the centrality–lethality hypothesis. Syst Synth Biol. 2014;8: 73–81. doi:10.1007/s11693-013-9123-5

109. Orth JD, Conrad TM, Na J, Lerman JA, Nam H, Feist AM, *et al.* A comprehensive genome-scale reconstruction of *Escherichia coli* metabolism—2011. Mol Syst Biol. 2011;7: 535. doi:10.1038/msb.2011.65

110. Orth JD, Thiele I, Palsson BØ. What is flux balance analysis? Nat Biotechnol. 2010;28: 245–248. doi:10.1038/nbt.1614

111. Burgard AP. Flux Coupling Analysis of Genome-Scale Metabolic Network Reconstructions. Genome Res. 2004;14: 301–312. doi:10.1101/gr.1926504

112. Larhlimi A, David L, Selbig J, Bockmayr A. F2C2: a fast tool for the computation of flux coupling in genome-scale metabolic networks. BMC Bioinformatics. 2012;13: 57. doi:10.1186/1471-2105-13-57

113. Barabási A-L. Network science. Philos Trans R Soc A Math Phys Eng Sci. 2013;371: 20120375. doi:10.1098/rsta.2012.0375

References

114. Liu X, Hong Z, Liu J, Lin Y, Rodríguez-Patón A, Zou Q, *et al.* Computational methods for identifying the critical nodes in biological networks. Brief Bioinform. 2020;21: 486–497. doi:10.1093/bib/bbz011

115. Jianxin Wang, Min Li, Huan Wang, Yi Pan. Identification of Essential Proteins Based on Edge Clustering Coefficient. IEEE/ACM Trans Comput Biol Bioinforma. 2012;9: 1070–1080. doi:10.1109/TCBB.2011.147

116. Csardi G, Nepusz T, others. The igraph software package for complex network research. InterJournal, Complex Syst. 2006;1695: 1–9. Available: https://igraph.org

117. Mann S, Chen Y-PP. Bacterial genomic G+C composition-eliciting environmental adaptation. Genomics. 2010;95: 7–15. doi:10.1016/j.ygeno.2009.09.002

118. dos Reis M. Unexpected correlations between gene expression and codon usage bias from microarray data for the whole *Escherichia coli* K-12 genome. Nucleic Acids Res. 2003;31: 6976–6985. doi:10.1093/nar/gkg897

119. Sharp PM, Li W-H. The codon adaptation index-a measure of directional synonymous codon usage bias, and its potential applications. Nucleic Acids Res. 1987;15: 1281–1295. doi:10.1093/nar/15.3.1281

120. Subramanian A, Sarkar RR. Comparison of codon usage bias across *Leishmania* and *Trypanosomatids* to understand mRNA secondary structure, relative protein abundance and pathway functions. Genomics. 2015;106: 232–241. doi:10.1016/j.ygeno.2015.05.009

121. Wright F. The 'effective number of codons' used in a gene. Gene. 1990;87: 23–29. doi:10.1016/0378-1119(90)90491-9

122. Rice P, Longden I, Bleasby A. EMBOSS: The European Molecular Biology Open Software Suite. Trends Genet. 2000;16: 276–277. doi:10.1016/S0168-9525(00)02024-2

123. Bauer M, Schuster SM, Sayood K. The Average Mutual Information Profile as a Genomic Signature. BMC Bioinformatics. 2008;9: 48. doi:10.1186/1471-2105-9-48

124. Ish-Am O, Kristensen DM, Ruppin E. Evolutionary conservation of bacterial essential metabolic genes across all bacterial culture media. PLoS One. 2015;10: 1–15. doi:10.1371/journal.pone.0123785

125. Galperin MY, Makarova KS, Wolf YI, Koonin E V. Expanded microbial genome coverage and improved protein family annotation in the COG database. Nucleic Acids Res. 2015;43: D261–D269. doi:10.1093/nar/gku1223

126. Plaimas K, Eils R, König R. Identifying essential genes in bacterial metabolic networks with machine learning methods. BMC Syst Biol. 2010;4: 56. doi:10.1186/1752-0509-4-56

127. Scheraga HA, Rackovsky S. Global informatics and physical property selection in protein sequences. Proc Natl Acad Sci. 2016;113: 1808–1810. doi:10.1073/pnas.1525745113

128. Kidera A, Konishi Y, Oka M, Ooi T, Scheraga HA. Statistical analysis of the physical properties of the 20 naturally occurring amino acids. J Protein Chem. 1985;4: 23–55. doi:10.1007/BF01025492

129. Grazziotin AL, Vidal NM, Venancio TM. Uncovering major genomic features of essential genes in Bacteria and a methanogenic Archaea. FEBS J. 2015;282: 3395–3411. doi:10.1111/febs.13350

130. Oklahoma University *E. Coli* Gene Expression Database. Available: http://genexpdb.ou.edu/

131. Yu H, Kim PM, Sprecher E, Trifonov V, Gerstein M. The Importance of Bottlenecks in Protein Networks: Correlation with Gene Essentiality and Expression Dynamics. PLoS Comput Biol. 2007;3: e59. doi:10.1371/journal.pcbi.0030059

132. Benson DA, Cavanaugh M, Clark K, Karsch-Mizrachi I, Lipman DJ, Ostell J, *et al.* GenBank. Nucleic Acids Res. 2012;41: D36–D42. doi:10.1093/nar/gks1195

133. Guyon I, Elisseeff A. An introduction to variable and feature selection. J Mach Learn Res. 2003;3: 1157–1182. doi:10.5555/944919.944968

134. Guyon I, Weston J, Barnhill S, Vapnik V. Gene selection for cancer classification using support vector machines. Mach Learn. 2002;46: 389–422. doi:10.1023/A:1012487302797

135. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH. The WEKA data mining software. ACM SIGKDD Explor Newsl. 2009;11: 10–18. doi:10.1145/1656274.1656278

136. Platt JC. Fast training of support vector machines using sequential minimal optimization. Adv kernel methods. 1999; 185–208. Available: https://www.microsoft.com/en-us/research/publication/fast-training-of-support-vector-machines-using-sequential-minimal-optimization/

137. Luo H, Lin Y, Gao F, Zhang C-T, Zhang R. DEG 10, an update of the database of essential genes that includes both protein-coding genes and noncoding genomic elements: Table 1. Nucleic Acids Res. 2014;42: D574–D580. doi:10.1093/nar/gkt1131

138. Thiele I, Vo TD, Price ND, Palsson BØ. Expanded Metabolic Reconstruction of *Helicobacter pylori* (iIT341 GSM/GPR): an *In Silico* Genome-Scale Characterization of Single- and Double-Deletion Mutants. J Bacteriol. 2005;187: 5818–5830. doi:10.1128/JB.187.16.5818-5830.2005

139. O'Leary NA, Wright MW, Brister JR, Ciufo S, Haddad D, McVeigh R, *et al.* Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. Nucleic Acids Res. 2016;44: D733–D745. doi:10.1093/nar/gkv1189

140. Durot M, Le Fèvre F, de Berardinis V, Kreimeyer A, Vallenet D, Combe C, *et al.* Iterative reconstruction of a global metabolic model of *Acinetobacter baylyi* ADP1 using high-throughput growth phenotype and gene essentiality data. BMC Syst Biol. 2008;2: 85. doi:10.1186/1752-0509-2-85

141. Oh Y-K, Palsson BO, Park SM, Schilling CH, Mahadevan R. Genome-scale Reconstruction of Metabolic Network in *Bacillus subtilis* Based on High-throughput Phenotyping and Gene Essentiality Data. J Biol Chem. 2007;282: 28791–28799. doi:10.1074/jbc.M703759200

142. Orth JD, Conrad TM, Na J, Lerman J A, Nam H, Feist AM, *et al.* A comprehensive genome-scale reconstruction of *Escherichia coli* metabolism--2011. Mol Syst Biol. 2011;7: 535. doi:10.1038/msb.2011.65

143. Jamshidi N, Palsson BØ. Investigating the metabolic capabilities of *Mycobacterium tuberculosis* H37Rv using the *in silico* strain iNJ 661 and proposing alternative drug targets. BMC Syst Biol. 2007;1: 26. doi:10.1186/1752-0509-1-26

144. Bartell JA, Blazier AS, Yen P, Thøgersen JC, Jelsbak L, Goldberg JB, *et al.* Reconstruction of the metabolic network of *Pseudomonas aeruginosa* to interrogate virulence factor synthesis. Nat Commun. 2017;8: 14631. doi:10.1038/ncomms14631

145. Thiele I, Hyduke DR, Steeb B, Fankam G, Allen DK, Bazzani S, *et al.* A community effort towards a knowledge-base and mathematical model of the human pathogen *Salmonella Typhimurium* LT2. BMC Syst Biol. 2011;5: 8. doi:10.1186/1752-0509-5-8

146. Li C, Donizelli M, Rodriguez N, Dharuri H, Endler L, Chelliah V, *et al.* BioModels Database: An enhanced, curated and annotated resource for published quantitative kinetic models. BMC Syst Biol. 2010;4: 92. doi:10.1186/1752-0509-4-92

147. Mo ML, Palsson BØ, Herrgård MJ. Connecting extracellular metabolomic measurements to intracellular flux states in yeast. BMC Syst Biol. 2009;3: 37. doi:10.1186/1752-0509-3-37

148. Yilmaz LS, Walhout AJM. A *Caenorhabditis elegans* Genome-Scale Metabolic Network Model. Cell Syst. 2016;2: 297–311. doi:10.1016/j.cels.2016.04.012

149. Sigurdsson MI, Jamshidi N, Steingrimsson E, Thiele I, Palsson BØ. A detailed genome-wide reconstruction of mouse metabolism based on human Recon 1. BMC Syst Biol. 2010;4: 140. doi:10.1186/1752-0509-4-140

150. Sharma M, Shaikh N, Yadav S, Singh S, Garg P. A systematic reconstruction and constraint-based analysis of *Leishmania donovani* metabolic network: identification of potential antileishmanial drug targets. Mol Biosyst. 2017;13: 955–969. doi:10.1039/C6MB00823B

151. Chavali AK, Whittemore JD, Eddy JA, Williams KT, Papin JA. Systems analysis of metabolism in the pathogenic trypanosomatid *Leishmania major*. Mol Syst Biol. 2008;4: 177. doi:10.1038/msb.2008.15

152. Laib M, Kanevski M. A Novel Filter Algorithm for Unsupervised Feature Selection Based on a Space Filling Measure. ESANN 2018 proceedings, Eur Symp Artif Neural Networks, Comput Intell Mach Learn Bruges. 2018. Available: https://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2018-57.pdf

153. Ang JC, Mirzal A, Haron H, Hamed HNA. Supervised, Unsupervised, and Semi-Supervised Feature Selection: A Review on Gene Selection. IEEE/ACM Trans Comput Biol Bioinforma. 2016;13: 971–989. doi:10.1109/TCBB.2015.2478454

154. Mitra P, Murthy CA, Pal SK. Unsupervised feature selection using feature similarity. IEEE Trans Pattern Anal Mach Intell. 2002;24: 301–312. doi:10.1109/34.990133

155. Kamada T, Kawai S. An algorithm for drawing general undirected graphs. Inf Process Lett. 1989;31: 7–15. doi:10.1016/0020-0190(89)90102-6

156. Kraemer G, Reichstein M, Mahecha, Miguel D. dimRed and coRanking - Unifying Dimensionality Reduction in R. R J. 2018;10: 342. doi:10.32614/RJ-2018-039

157. Fruchterman TMJ, Reingold EM. Graph drawing by force-directed placement. Softw Pract Exp. 1991;21: 1129–1164. doi:10.1002/spe.4380211102

158. Hyvarinen A. Fast and robust fixed-point algorithms for independent component analysis. IEEE Trans Neural Networks. 1999;10: 626–634. doi:10.1109/72.761722

159. Krijthe JH. RSSL: Semi-supervised Learning in R. International Workshop on Reproducible Research in Pattern Recognition. 2016. pp. 104–115. Available: https://link.springer.com/chapter/10.1007/978-3-319-56414-2_8

160. Rousseeuw PJ. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math. 1987;20: 53–65. doi:10.1016/0377-0427(87)90125-7

161. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, *et al*. Gene Ontology: tool for the unification of biology. Nat Genet. 2000;25: 25–29. doi:10.1038/75556

162. Consortium GO. The Gene Ontology Resource: 20 years and still GOing strong. Nucleic Acids Res. 2019;47: D330–D338. doi:10.1093/nar/gky1055

References

163. Consortium U. UniProt: a worldwide hub of protein knowledge. Nucleic Acids Res. 2019;47: D506–D515. doi:10.1093/nar/gky1049

164. Huang DW, Sherman BT, Lempicki RA. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. Nat Protoc. 2009;4: 44–57. doi:10.1038/nprot.2008.211

165. Kanehisa M, Sato Y, Furumichi M, Morishima K, Tanabe M. New approach for understanding genome variations in KEGG. Nucleic Acids Res. 2019;47: D590–D595. doi:10.1093/nar/gky962

166. Wang J, Peng W, Wu F-X. Computational approaches to predicting essential proteins: A survey. PROTEOMICS - Clin Appl. 2013;7: 181–192. doi:10.1002/prca.201200068

167. Mann S, Chen YPP. Bacterial genomic G + C composition-eliciting environmental adaptation. Genomics. 2010;95: 7–15. doi:10.1016/j.ygeno.2009.09.002

168. Gong X, Fan S, Bilderbeck A, Li M, Pang H, Tao S. Comparative analysis of essential genes and nonessential genes in *Escherichia coli* K12. Mol Genet Genomics. 2008;279: 87–94. Available: https://link.springer.com/article/10.1007%2Fs00438-007-0298-x

169. Papp B, Notebaart RA, Pál C. Systems-biology approaches for predicting genomic evolution. Nat Rev Genet. 2011;12: 591–602. doi:10.1038/nrg3033

170. Song K, Tong T, Wu F. Predicting essential genes in prokaryotic genomes using a linear method: ZUPLS. Integr Biol. 2014;6: 460–469. doi:10.1039/C3IB40241J

171. Hwang Y-C, Lin C-C, Chang J-Y, Mori H, Juan H-F, Huang H-C. Predicting essential genes based on network and sequence analysis. Mol Biosyst. 2009;5: 1672–1678.

172. Grazziotin AL, Vidal NM, Venancio TM. Uncovering major genomic features of essential genes in Bacteria and a methanogenic Archaea. FEBS J. 2015;282: 3395–3411. doi:10.1111/febs.13350

173. Davis MS, Solbiati J, Cronan JE. Overproduction of acetyl-CoA carboxylase activity increases the rate of fatty acid biosynthesis in *Escherichia coli*. J Biol Chem. 2000;275: 28593–28598. doi:10.1074/jbc.M004756200

174. Yamamoto N, Nakahigashi K, Nakamichi T, Yoshino M, Takai Y, Touda Y, *et al*. Update on the Keio collection of *Escherichia coli* single-gene deletion mutants. Mol Syst Biol. 2009;5: 335. doi:10.1038/msb.2009.92

175. Lee PT, Hsu AY, Ha HT, Clarke CF. A C-methyltransferase involved in both ubiquinone and menaquinone biosynthesis: isolation and identification of the *Escherichia coli* ubiE gene. J Bacteriol. 1997;179: 1748–1754. doi:10.1128/JB.179.5.1748-1754.1997

176. Justino MC, Almeida CC, Teixeira M, Saraiva LM. *Escherichia coli* Di-iron YtfE Protein Is Necessary for the Repair of Stress-damaged Iron-Sulfur Clusters. J Biol Chem. 2007;282: 10352–10359. doi:10.1074/jbc.M610656200

177. Lai C-Y, Cronan JE. β-Ketoacyl-Acyl Carrier Protein Synthase III (FabH) Is Essential for Bacterial Fatty Acid Synthesis. J Biol Chem. 2003;278: 51494–51503. doi:10.1074/jbc.M308638200

178. Hase Y, Yokoyama S, Muto A, Himeno H. Removal of a ribosome small subunit-dependent GTPase confers salt resistance on *Escherichia coli* cells. RNA. 2009;15: 1766–1774. doi:10.1261/rna.1687309

179. Velur Selvamani R, Telaar M, Friehs K, Flaschel E. Antibiotic-free segregational plasmid stabilization in *Escherichia coli* owing to the knockout of triosephosphate isomerase (tpiA). Microb Cell Fact. 2014;13: 58. doi:10.1186/1475-2859-13-58

References

180. Subramanian A, Sarkar RR. Perspectives on *Leishmania* Species and Stage-specific Adaptive Mechanisms. Trends Parasitol. 2018;34: 1068–1081. doi:10.1016/j.pt.2018.09.004

181. Wei W, Ning L-W, Ye Y-N, Guo F-B. Geptop: a gene essentiality prediction tool for sequenced bacterial genomes based on orthology and phylogeny. PLoS One. 2013;8: e72343.

182. Matthews BW. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. Biochim Biophys Acta - Protein Struct. 1975;405: 442–451. doi:10.1016/0005-2795(75)90109-9

183. Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers. Mach Learn. 1997;29: 131–163. Available: https://link.springer.com/article/10.1023/A:1007465528199

184. Hosmer Jr DW, Lemeshow S, Sturdivant RX. Applied logistic regression. John Wiley & Sons; 2013. Available: https://www.oreilly.com/library/view/applied-logistic-regression/9781118548356/

185. Quinlan JR, others. Bagging, boosting, and C4. 5. AAAI/IAAI, Vol 1. 1996. pp. 725–730. Available: https://dl.acm.org/doi/10.5555/645328.650011

186. Jones NG, Catta-Preta CMC, Lima APCA, Mottram JC. Genetically Validated Drug Targets in *Leishmania* : Current Knowledge and Future Prospects. ACS Infect Dis. 2018;4: 467–477. doi:10.1021/acsinfecdis.7b00244

187. Nandi S, Ganguli P, Sarkar RR. Essential gene prediction using limited gene essentiality information–An integrative semi-supervised machine learning strategy. PLoS One. 2020;15: e0242943. doi:10.1371/journal.pone.0242943

ABSTRACT

Name of the Student: Sutanu Nandi
Registration No.: 10PP15J26030
Faculty of Study: Physical Sciences
Year of Submission: 2021
CSIR Lab: CSIR-NCL, Pune
Name of the Supervisor: Dr. Ram Rup Sarkar

Title of the thesis: Development of Machine Learning Strategies and Integrated Web Platform for the Prediction of Essential Genes

---

Essential gene prediction helps to find minimal genes indispensable for the survival of any organism. Machine learning (ML) algorithms have been useful for the prediction of gene essentiality. Existing ML techniques for essential gene prediction have inherent problems, like imbalanced provision of training datasets with sufficient data (labeled ≥80%), limited (labeled ≥1%) experimental labeled data, biased choice of the best model for a given balanced dataset, choice of a complex ML algorithm, and data-based automated selection of biologically relevant features for classification. By addressing these issues, two ML strategies (ML strategy 1 and ML strategy 2) were developed to predict essential genes.

The ML strategy 1 was developed based on the supervised ML classifier - Support Vector Machine (SVM) for predicting essential genes in *Escherichia coli* with sufficient imbalanced experimental data (labeled ≥80%). As a novel feature, we introduced flux-coupled metabolic subnetwork-based features for enhancing the classification performance. Our strategy has proved to be superior when compared with existing SVM-based strategies.

ML Strategy 1 underperforms for limited labeled data (labeled ≥ 1%), and hence ML strategy 2 was developed to circumvent this issue. ML strategy 2 utilizes an unsupervised feature selection technique, dimension reduction (Kamada-Kawai algorithm), and semi-supervised ML algorithm (Laplacian Support Vector Machine). A novel scoring technique, Semi-Supervised Model Selection Score (equivalent to the area under the ROC curve (auROC)), was developed to select the best model when supervised performance metrics calculation difficult due to lack of data. Validation of this ML pipeline gave highly accurate (auROC > 0.85) performance even with 1% labeled data on both Eukaryotes and Prokaryotes. This strategy was used on *Leishmania sp*. to predict essential genes with inadequate experimental known data.

The existing essential genes prediction platforms such as Geptop, EGP, etc., can only annotate essential genes for model prokaryotic organisms, not for eukaryotes and in most cases, no source code is publicly available. Hence, for annotating the essential genes with minimal effort and time, an open-source server, PRESGENE was developed, by integrating these two ML strategies. The user can submit and analyze their data for essential genes prediction through a user-friendly platform. The essential genes predicted using this platform will provide an important lead for predicting gene essentiality and identifying novel therapeutic targets for antibiotic and vaccine development against disease-causing organisms.

List of publications

Publication(s) in SCI Journal(s) Emanating from the Thesis Work

1. **Nandi S**, Subramanian A, Sarkar RR. An integrative machine learning strategy for improved prediction of essential genes in *Escherichia coli* metabolism using flux-coupled features. ***Molecular BioSystems***. 2017;13: 1584–1596. doi:10.1039/C7MB00234C

2. **Nandi S**, Ganguli P, Sarkar RR. Essential gene prediction using limited gene essentiality information–An integrative semi-supervised machine learning strategy. ***PLoS ONE***. 2020;15: e0242943. doi: 10.1371/journal.pone.0242943

3. **Nandi S**, Ganguli P, Panditrao G and Sarkar RR. PRESGENE: A webserver for PRediction of ESsential GENEs using integrative machine learning strategies. 2021 **(Submitted)**

Other Publications

4. Bose S, Dhawan D, **Nandi S**, Sarkar RR, Ghosh D. Machine learning prediction of interaction energies in rigid water clusters. ***Physical Chemistry Chemical Physics***. 2018;20: 22987–22996. doi:10.1039/C8CP03138J

5. Chowdhury S, Sinha N, Ganguli P, Bhowmick R, Singh V, **Nandi S**, Sarkar RR. BIOPYDB: A Dynamic Human Cell Specific Biochemical Pathway Database with Advanced Computational Analyses Platform. ***Journal of Integrative Bioinformatics***. 2018;15. doi:10.1515/jib-2017-0072

6. Saurabh R, **Nandi S**, Sinha N, Shukla M, Sarkar RR. Prediction of survival rate and effect of drugs on cancer patients with somatic mutations of genes: An AI-based approach. ***Chemical Biology and Drug Design***. 2020;96: 1005–1019. doi:10.1111/cbdd.136681.

List of papers with abstract presented (oral or poster) at national or international conferences/seminars.

1. "National Science Day" held at CSIR-NCL, Pune, India, 2018. (Poster Presentation)
2. "Accelerating Biology 2018: Digitizing Life" held at C-DAC, Pune, India, 2018. (Poster and Oral Presentation)
3. "TCGA India 2019 conference", held at IISER, Pune, India, 2019. (Poster Presentation)
4. "Accelerating Biology 2020: SNiPs to SPiNs" held at C-DAC, Pune, India, 2020. (Poster and Oral Presentation)

Molecular BioSystems

PAPER

View Article Online
View Journal | View Issue

Cite this: *Mol. BioSyst.*, 2017, **13**, 1584

An integrative machine learning strategy for improved prediction of essential genes in *Escherichia coli* metabolism using flux-coupled features†

Sutanu Nandi, [ab] Abhishek Subramanian [ab] and Ram Rup Sarkar *[ab]

Received 19th April 2017,
Accepted 14th June 2017

DOI: 10.1039/c7mb00234c

rsc.li/molecular-biosystems

Prediction of essential genes helps to identify a minimal set of genes that are absolutely required for the appropriate functioning and survival of a cell. The available machine learning techniques for essential gene prediction have inherent problems, like imbalanced provision of training datasets, biased choice of the best model for a given balanced dataset, choice of a complex machine learning algorithm, and data-based automated selection of biologically relevant features for classification. Here, we propose a simple support vector machine-based learning strategy for the prediction of essential genes in *Escherichia coli* K-12 MG1655 metabolism that integrates a non-conventional combination of an appropriate sample balanced training set, a unique organism-specific genotype, phenotype attributes that characterize essential genes, and optimal parameters of the learning algorithm to generate the best machine learning model (the model with the highest accuracy among all the models trained for different sample training sets). For the first time, we also introduce flux-coupled metabolic subnetwork-based features for enhancing the classification performance. Our strategy proves to be superior as compared to previous SVM-based strategies in obtaining a biologically relevant classification of genes with high sensitivity and specificity. This methodology was also trained with datasets of other recent supervised classification techniques for essential gene classification and tested using reported test datasets. The testing accuracy was always high as compared to the known techniques, proving that our method outperforms known methods. Observations from our study indicate that essential genes are conserved among homologous bacterial species, demonstrate high codon usage bias, GC content and gene expression, and predominantly possess a tendency to form physiological flux modules in metabolism.

Introduction

Essential genes, as the name suggests, are genes important for the survival of any cell. Identification of essential genes is required in a multitude of applications like determining drug targets in diseases, the role (or function) of a gene within a biological network (systems biology) and indicators of metabolic microenvironments, and the generation of biologically engineered strains of micro-organisms.[1,2] The definition of a gene that is essential for survival largely depends upon the environment in which a cell survives and is governed by the underlying function that it performs within the cell.[3] This leads to the classification of essential genes as either minimally essential (genes that are absolutely essential irrespective of environmental variations) or conditionally essential (genes that are essential for cell survival in a particular environment).[4,5]

Innumerable experimental techniques like gene knockouts (deletions), double targeted gene replacement, genetic footprinting, transposon mutagenesis, RNA interference, *etc.* are available for

[a] *Chemical Engineering and Process Development, CSIR-National Chemical Laboratory, Pune-411008, Maharashtra, India. E-mail: rr.sarkar@ncl.res.in; Fax: +91-20-2590-2621; Tel: +91-20-2590-3040*

[b] *Academy of Scientific & Innovative Research (AcSIR), NCL Campus, India*

† Electronic supplementary information (ESI) available: Text S1: This file contains a description of the model's features and their curation, the algorithm for the proposed pipeline, details of the flux coupling analysis, definitions of the performance metrics used, a comparison of our method with other supervised classification techniques and a list of features selected by SVM-RFE.[53–77] Table S1: The instance-feature file. This file contains a total of 64 features computed for 3504 reaction–gene pairs. Table S2: Correlation matrix of the model's features. This file contains a matrix of Spearman's correlation values for each pair of features and their corresponding *P*-values. Table S3: Model testing and prediction. This file contains the predictions for the experimentally known and unknown essential and non-essential reaction–gene pairs using the best model trained with the best dataset and the best model trained with the 1000 balanced training datasets; Sheet 1 for *E. coli*, Sheet 2 for *H. pylori* and Sheet 3 for *B. subvibriodes*. See DOI: 10.1039/c7mb00234c

scrutinizing the essential role of a gene.[6–10] However, the essential genes identified by each of these experiments differ significantly.[6,8] Further, these techniques work well with model organisms for which a standardized protocol for gene essentiality identification is available. To establish the essentiality of a large set of genes in non-model, less explored organisms is a challenging task, as the experimental standardization of protocols for determining gene dispensability and sampling for a range of experimental conditions is laborious and time-consuming.

Computational techniques help to facilitate this identification by predicting a handful of probable genes that might actually be essential based on features extracted from known experimental data. With an appropriate selection of biological "features", it is possible to train computational (machine learning) models with high precision and sensitivity to predict and classify genes of an organism as essential or non-essential based on a training set of features collected for known essential genes. Once the genes are predicted to be indispensable, they can be rightfully tested for their essentiality and biological role through *in vitro* or *in vivo* experiments designed for that particular organism. An ideal supervised machine learning technique depends upon the choice of a training dataset, appropriate learning algorithm and features (attributes) that can classify the instances. Numerous techniques have been applied for the classification of genes in *Escherichia coli* based on their essentiality.[11–17]

A pre-requisite for an appropriate training dataset is that it spans a large number of instances (genes) with known class labels (essential or non-essential). This would ensure that the sampled instances with experimentally known class labels characterize a large proportion of the population (genome/metabolome) under consideration. In this context, a variety of genome-scale knockout datasets in *E. coli* have been previously used to train machine learning models for essentiality classification. Early machine learning studies have widely used the genome-wide dataset by Gerdes *et al.*,[8] which provides essentiality information for 3746 genes in *E. coli* K-12 MG1655 that are necessary for aerobic growth in a rich, tryptone-based medium using transposon-based genetic footprinting. Subsequent studies used the dataset generated by Baba *et al.*,[6] where the essentiality information for 4288 genes in *E. coli* K-12 MG1655 was determined in two different growth media (rich and glucose-minimal media) by performing a selection of surviving gene deletion mutants through direct inactivation of chromosomal genes. Apart from this, a number of other large-scale experimental studies also exist, albeit with a relatively low coverage of genes.[5,18]

The choice of a machine learning algorithm depends on the available large number of diverse, independent features (high dimensionality) that can classify instances, the distribution pattern of instances with respect to the chosen features, and an equal number of instances belonging to each class. Widely used machine learning algorithms for essentiality classification in *E. coli* include support vector machines (SVMs),[13–15] ensemble-based machine learning,[16] and probabilistic Bayesian-based,[11,16,17] logistic regression[16,17] and decision tree-based[16] algorithms. The objectives of these studies typically include either identification of essential genes within *E. coli*, considering the same dataset for training and testing, or cross-testing, where training is based on *E. coli* and using the trained model, and prediction is done for a test organism. Although these studies use powerful classifiers, there are a few technical problems associated with them. A previous study[11] used a Naïve Bayes model, trained for similar features like amino acid composition, aromaticity, codon adaptation index and frequency of optimal codons which violate the fundamental assumption of statistical independence.[19] A few other studies use CN2 rule-based classifiers, decision trees and logistic regression for classification of genes.[16,17] Logistic regression suffers from imbalances in the training data.[20] Inappropriate decision tree heuristics result in local optima and hence might affect the predictive ability of the decision tree classifier.[21] CN2 rule-based classifiers have a similar problem.[22] Likewise, SVMs are also affected by imbalanced training datasets and the use of correlated or redundant features.[23] A recent SVM-based study failed to address both these issues appropriately,[14] as the number of balanced training sets generated was reasonably low compared to the number of available instances in each class, and features like the clustering coefficient and clique level were redundant, holding similar interpretation.

The features required for determining gene dispensability should be biologically relevant, unique, and should consider the heterogeneous properties of genes. The above machine learning-based studies utilize an array of sequence, structural and pathway features to classify genes based on their dispensability.[24] With the availability of a high number of sequenced and annotated genomes, features from nucleotide and protein sequences are largely used for training machine learning models of classification.[25] Nucleotide sequence properties like the codon adaptation index, phyletic retention, GC content, and protein sequence properties like amino acid frequency and protein length are known indicators of gene essentiality across bacteria.[26,27] More recently, features related to gene expression and biological networks have gained more importance as compared to static genomic features, as they represent an organismal phenotype. In *E. coli*, topological network features of the protein interaction networks (PINs) have been used along with sequence related features to distinguish essential from non-essential genes.[14] However, the idea of a centrality–lethality hypothesis in a PIN might not hold true for many organisms.[28] Further, their roles in the context of signaling and metabolic pathways cannot be inferred using only interaction information. For this purpose, different network representations need to be analyzed. A few studies use topological and flux-based features from metabolic networks to classify genes.[13,15] The flux features used in these studies are principally calculated using flux balance analysis (FBA)[29] under a single environmental condition (aerobic glucose input) while optimizing for the biomass objective. Hence, the calculated flux features are condition-specific and do not represent a universal set of features. Previous studies clearly establish that the essentiality of a gene is highly dependent on its adaptability to any environment.[30]

To tackle the aforementioned problems inherent within training sets, feature bias and the limitations of learning algorithms,

we present here a simple yet powerful integrative machine learning strategy based on a fundamental SVM-based implementation for binary classification of genes based on gene and protein sequences, gene expression, and network topological and flux-based features for *E. coli* K-12 MG1655 metabolism. This integrative machine learning strategy attempts selection of the most contributing genotype and phenotype features required for classification by SVM-Recursive Feature Elimination (SVM-RFE), choice of the best parameters for the learning model, and removal of the dependency (bias) of a model on a given balanced training dataset to give a highly accurate, unbiased, predictive machine learning model for appropriate classification of genes based on their essentiality. To account for the inherent limitations of environmental dependence in calculating flux distributions through a metabolic network, we perform flux coupling analysis (FCA)[31,32] on the *E. coli* iJO1366 metabolic network while considering all the possible input conditions. For the first time, we incorporate the network topological features of the obtained flux-coupled subnetwork into our model strategy for achieving higher classification performances. We hypothesize that this curated and selected feature set represents the minimal organism-specific constraints that govern the essentiality of a gene in *E. coli* K-12 MG1655 metabolism. The model generated from the selected features, when tested with the experimentally known *E. coli* Keio collection dataset,[6] predicted 94.28% of the total known essential genes to be essential and 82.59% of the total known non-essential genes to be non-essential. Further, it is to be noted that, by virtue of the selected features for training, our method is able to capture the minimal set of essential genes that prove to be essential in any given environment. Our method was also able to predict the essentiality of 317 genes previously unidentified by genome-scale knockout experiments. Our methodology was also trained with the datasets of other recent supervised classification techniques for essential gene classification and tested using their reported test datasets.[16,33] Test results indicate that our method achieves the highest sensitivity and specificity as compared to the recent supervised classification techniques, irrespective of the input training dataset. Finally, from our analysis, we establish that a simple machine learning strategy is enough to predict dispensable genes when provided with an appropriate choice of features combined with the best hyperplane choice in the feature space. Also, as an applicable methodology, our proposed strategy can be used in organisms where the essentiality phenotype of genes is mostly unknown.

Materials and methods

With the aim to create a highly precise machine learning model with the above fundamental requirements for the binary classification of essential and non-essential genes, an integrated pipeline that addresses the problems of training a model using an appropriate balanced dataset of instances, automated selection of relevant diverse biological features, identifying an optimized set of model parameters that classifies the chosen instances, and rigorous testing of the obtained data-driven trained model was designed. A schematic view of the designed pipeline is shown in Fig. 1. The algorithm for the pipeline is given in Section 1 of the ESI,† Text S1.

Training data set preparation

For the purpose of generating a characteristic training dataset, the metabolic genes from the *E. coli* genome-scale metabolic network reconstruction iJO1366 were considered.[34] The reconstruction consists of 1805 metabolites, 2583 reactions and 1367 genes. It can be observed that for a subset of reactions, there are either many genes that govern a single reaction (enzyme complexes) or a single gene that governs many reactions (depending upon multiple substrates that it catalyzes). For example, gene b0002 encodes both aspartate kinase and homoserine dehydrogenase, whereas acetaldehyde dehydrogenase is encoded by genes b2388 and b0351. Hence, reaction–gene combinations (R_a_G_b, Fig. 1) were created from the network reconstruction. The creation of reaction–gene combinations directly provides insights into the role of a specific metabolic reaction catalyzed by a gene, deeming it to be essential. No previous machine learning strategies for essential gene predictions have considered this feature. The information for essentiality (class label) of each reaction–gene combination was adopted from a previous experimental study.[6] This particular study was selected as the gold standard because the essentiality of nearly all genes in *E. coli* K-12 MG1655 was tested in a variety of environmental conditions and confirmed using stringent gene knockout techniques. The integrated instance-feature-class file is given in the ESI,† Table S1. Finally, a total training dataset of 4094 metabolic reaction–gene pairs was enlisted, out of which 384 were essential, 3120 were non-essential, and for around 590 reaction–gene pairs there was no phenotype information available. The known 384 essential and 3120 non-essential reaction–gene pairs were considered as the master (unbalanced) dataset of instances.

Sequence-based, gene expression-based, and metabolic network and flux-coupled subnetwork-based features were assembled for each reaction–gene combination within *E. coli* K-12 MG1655 metabolism. Thus, a total of 64 features were obtained for each pair. A list of features for each type is given in Table A of Section 2, ESI,† Text S1. It is important to note that for situations where many genes catalyzed a single reaction, the sequence features related to each reaction–gene pair were distinct, whereas the network topological features remained the same, while for situations where there were many reactions catalyzed by a single gene, the network features for each reaction–gene pair were different and the sequence features remained the same. The coding nucleotide (CDS) and protein sequences for the 1367 metabolic genes extracted from *E. coli* str. K-12 substr. MG1655 genome assembly GCA_000005845.2, available in NCBI GenBank,[35] was used for curation of the sequence-based features.

Curated features for *E. coli* K-12 MG1655

Genome sequence-based features like the nucleotide frequency (A, T, G and C), Coding Sequence length (CDSlen), Codon Adaptation Index (CAI), Effective Number of Codons (ENC), and total number of codons (Num_codon) in a nucleotide

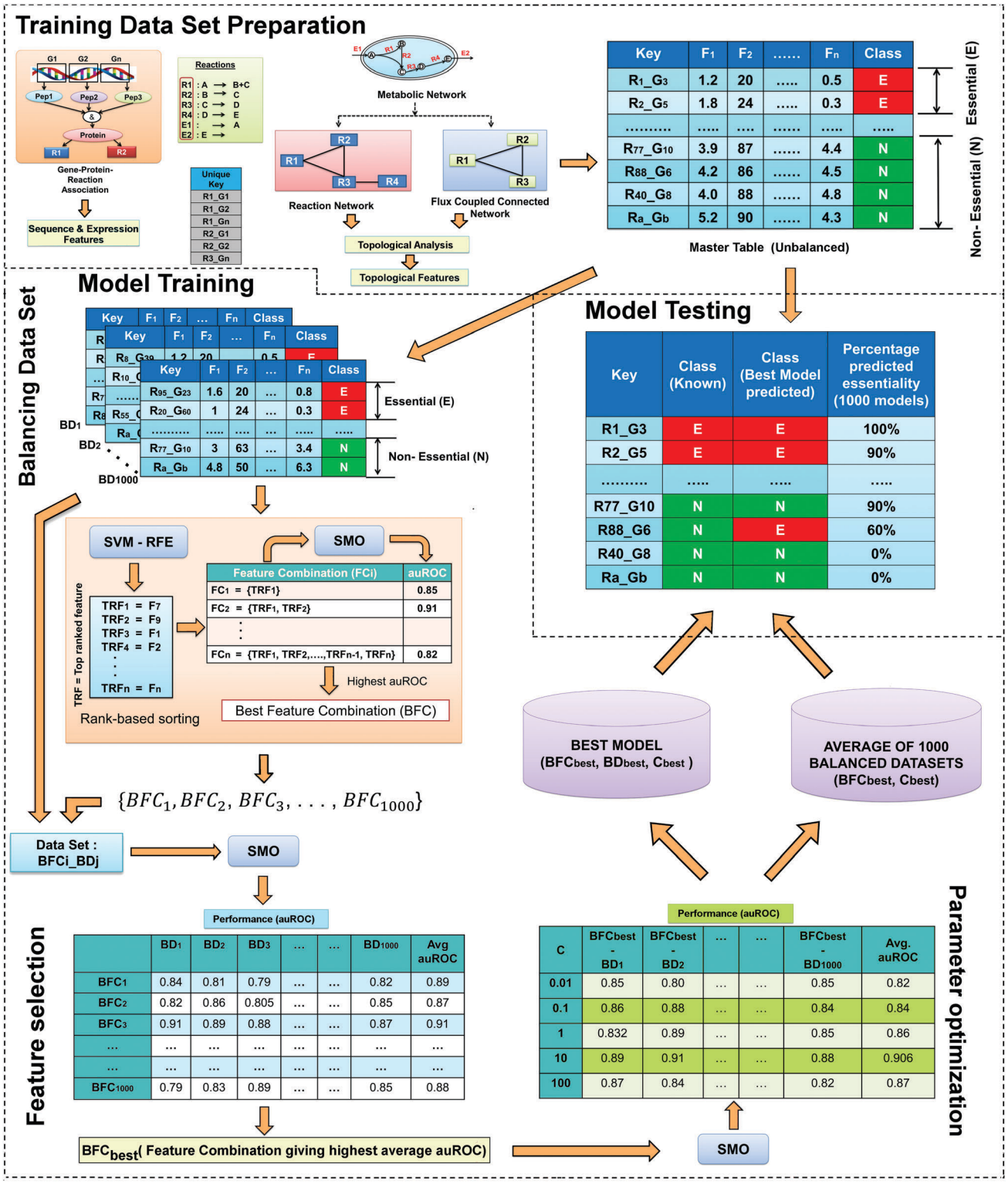


Fig. 1 The proposed machine learning methodology. The integrated pipeline for prediction of essential genes is based on a given input unbalanced training dataset consisting of reaction–gene pairs with sequence, expression, and network topological features.

sequence, protein sequence-based features, frequencies of the 20 amino acids, length of protein sequence (PL), Phyletic Retention (PR), homologs identified from 710 bacterial genomes reported in a cluster of orthologous groups (COG) database[36] with different *E*-value cut-offs, expression-based features like the Number of Genes with Similar Expression (NGSE), average mRNA Expression (aveEXP), mean Expression Fluctuation (mEF), and network-based features like Degree Centrality (RN_DC, FCA_DC),

Eccentricity Centrality (RN_EC, FCA_EC), Closeness Centrality (RN_CC, FCA_CC), Betweenness Centrality (RN_BC, FCA_BC), Eigenvector Centrality (RN_EvC, FCA_EvC), Hub Score (RN_HS, FCA_HS), Authority Score (RN_AS, FCA_AS), Page Rank (RN_PageRank, FCA_ PageRank), Clustering Coefficient (RN_ClustCoef, FCA_ ClustCoef), Number of triangles (RN_Num_triangles, FCA_Num_triangles), and Modularity (RN_M, FCA_M) were calculated for all the reaction–gene pairs. It is important to note that, for the first time, flux-coupled subnetwork-based features have been introduced for essential gene classification. A detailed explanation for the choice of these features to signify gene essentiality is given in Section 2 of the ESI,† Text S1.

Calculation of flux-coupled features

Flux-based calculations give a more realistic view of metabolic gene function. Typical computational methods like FBA perform these calculations on genome-scale metabolic networks to compute the flux (flow of metabolites) through an enzyme (reaction) using a linear optimization procedure to maximize or minimize a defined objective function.[29] However, FBA is limited by environmental (exchange) constraints and the knowledge of an objective function for defining the essentiality of a gene. Hence, to avoid these dependencies, flux coupling analysis (FCA)[31,32] was performed on the iJO1366 network, while considering all input exchanges (representative of all environmental conditions) to be functional. The F2C2 tool v0.95b[32] was used for performing flux coupling analysis (see Section 3 of the ESI,† Text S1 for details). The tool identifies a flux-coupled table (fctable) with 1718 reaction pairs (1718 × 1718 adjacency matrix) that may be coupled or uncoupled. After performing FCA, 1527 fully, 41 049 directionally, and 7438 partially coupled reaction pairs and 865 blocked reactions were obtained from the network. Instead of separately considering these classes, only the information whether the reaction pairs are coupled or not was used to generate a flux-coupled subgraph from the iJO1366 reconstruction. We define the flux-coupled subgraph to be an undirected completely connected reaction graph, where each node is an enzyme (reaction) and each edge represents the flux dependence of one enzyme over the other (coupled – 1, uncoupled/blocked – 0). The flux-coupled subnetwork is phenotypically more relevant than the static reaction network representation of metabolism, as a reaction in a flux-coupled network has a physiological dependence on another enzyme. It is apparent that an enzyme (reaction) might be essential if it is coupled by flux to multiple enzymes within the flux-coupled network under any environmental condition, due to stoichiometric dependence for the provision of metabolites in the form of either cofactors or substrates.

Balancing the training dataset

A major obstacle in classification using SVMs is the possibility of a significant class imbalance being observed in the dataset used for training.[23] A large disproportion in the two classes may result in a poor predictive capability of the model due to overfitting of the decision hyperplane, biased towards the class with a greater number of instances. Previous SVM-based machine learning strategies for essential gene classification have attempted to overcome this problem by introducing a very small set of randomized balanced data sets for training.[14] However, the generation of the small sets of randomized balanced data sets fails to sample the entire population of genes sufficiently, and is therefore unable to obtain a perfect sample that can represent a population. This might affect the choice of a perfect training set, leading to a sub-optimal model performance. To acquire the perfect training sample, the non-essential class was undersampled a sufficient number of times (1000 samples), so as to obtain numerous datasets containing an equal number of essential and non-essential class labels. Sampling was performed such that each chosen non-essential sample is unique and not repeatedly chosen. The generated 1000 balanced training datasets ensure that each non-essential reaction–gene pair is probably sampled at least once. The balanced datasets (BD_1 to BD_{1000}) thus generated were used for model training and subsequent testing.

Feature selection methodology

As the contribution of the 64 features towards the essentiality of a gene is unknown, there may be the possibility of choosing redundant features for training. Redundant features can affect the predictive capability of the model.[37] Hence, it is incumbent to select a unique, non-redundant subset for training the model. Feature selection helps to enlist the most relevant biological features required for essential/non-essential reaction–gene classification and thereby reduces the feature dimensions for better construction of a hyperplane.

To perform feature selection, each of the balanced datasets was provided to the SVM-RFE algorithm.[38] SVM-RFE has been previously established to be a useful strategy for feature selection in the context of essential gene classification.[39] SVM-RFE was performed using WEKA version 3.8.[40] In SVM-RFE, firstly the features are ranked. To obtain the best set of features, iteratively each top 'n' feature combination, where n = number of ranked features chosen, in the range of 1, 2,. . .,64, was selected and given to the sequential minimal optimization (SMO)[41] algorithm for classification while performing 10-fold cross-validation. The feature combination that gave the best performance (with respect to the area under the Receiver–Operator–Characteristic curve (auROC)) was chosen for each dataset. Thus, the corresponding 1000 well-performing feature combinations were shortlisted. Each of these combinations was again trained for the 1000 balanced datasets (BFC_i_BD_j, where i = 1 to 1000 and j = 1 to 1000). The average auROC of the 1000 trained models for each best feature set was calculated. Out of all the feature sets, the feature set giving the average highest performance was considered to be the best of all the best feature combinations (BFC_{best}).

Parameter optimization of classifier/SVM model for classification

In order to obtain a globally optimal hyperplane fit, the penalty parameter (C) of the SMO algorithm was fixed at different values (0.01, 0.1, 1, 10 and 100) and trained again for the 1000 datasets while performing a 10-fold cross validation with

the above selected best feature combination (BFC_{best}_BD_i, where i =1 to 1000). The penalty parameter that gave the highest average auROC was selected (C_{best}). The parameters of the linear kernel function were set to default in each case. Finally, a best feature set, best training dataset, and best parameter combination (BFC_{best}, BD_{best}, and C_{best}) can be obtained, which define the "best" chosen model from our strategy (see Fig. 1). This best chosen model was further used for comparison with other published models, and for testing and predictions.

Performance metrics

A number of performance metrics were used to evaluate the model. The metrics used in our study are given in Table 1. The definitions of each of these metrics are given in Section 4 of the ESI,† Text S1. Previous SVM-based classification strategies have calculated model performance metrics with respect to only the essential (positive) class.[14] To understand the strength of the model to classify instances into both classes (E and N), a weighted average of each metric was calculated and considered for measuring the true performance of the model strategy.

Let M be the total set of performance metrics.

M = {TPR, FPR, precision, recall, F-measure, MCC, auROC}.

The weighted average of each metric for measuring model performance was computed by the following formula:

$$\text{Weighted\_Metric}_i = \frac{(M_{ip} \times \text{PI}) + (M_{in} \times \text{NI})}{\text{PI} + \text{NI}}$$

where $i \in M$, M_{ip} is the performance metric for the positive class, M_{in} is the performance metric for the negative class, PI is the number of positive instances, and NI is the number of negative instances.

Model testing

Class labels from the model can be predicted with respect to the best training sample chosen from the population of the 1000 balanced datasets of known genes (best chosen model). However, by doing so, there is an inherent bias towards the chosen sample for training. This bias can be avoided by checking whether the class label predicted by the best model compares to the class labels predicted by models trained on all the samples. In our study, we present and compare the class labels of the genes predicted by both the best model and by the 1000 trained models, which none of the previous studies have provided. Hence, the master unbalanced dataset was provided as a testing dataset for:

(1) The best model trained for the randomized dataset that gives the best performance (BFC_{best}, BD_{best}, C_{best}). The predicted phenotype of each reaction–gene pair is determined to be essential (E) or non-essential (N), as predicted by the best model.

(2) The best model (BFC_{best}, C_{best}) trained for each of the 1000 random datasets generated (1000 trained models). The percentage of models that predict the phenotype of each reaction–gene pair is computed. In this case, reaction–gene pairs are assigned a phenotype of essential (E) or non-essential (N) only if accurately predicted by 80% of the 1000 trained models. It is worth mentioning that this threshold is user-defined and can be changed.

Supporting analyses

To test whether the features identified through our strategy were significantly different between the two groups, the distributions of each individual selected feature in the master dataset for the essential and non-essential genes were compared. Further, to verify whether the selected features were independent of each other, the correlation between every pair of variables was also calculated.

Dataset curation of other prokaryotes

Published essentiality datasets from two different prokaryotes, namely *Brevundimonas subvibriodes* ATCC 15264 and *Helicobacter pylori* 26695, were obtained from the Database of essential genes (DEG) version 13.3.[42] As no curated genome-scale metabolic network was available for *B. subvibriodes* ATCC 15264, only nucleotide and amino acid sequence composition-based features were calculated and used for model training. In the case of *H. pylori* 26695, both sequence-based and metabolic network-based features were calculated. A published genome-scale metabolic network, iIT341,[43] was used for computing the reaction network and flux-coupled subnetwork-based features. As very few gene expression studies were available for both the species, model

Table 1 Performance metrics used to evaluate the model

Metric	Formula
True positive rate (TPR) or sensitivity	$\frac{\text{TP}}{\text{TP} + \text{FN}}$
False positive rate (FPR) or (1-specificity)	$\frac{\text{FP}}{\text{FP} + \text{TN}}$
Precision	$\frac{\text{TP}}{\text{TP} + \text{FP}}$
Recall	$\frac{\text{TP}}{\text{TP} + \text{FN}}$
F-Measure	$\frac{2(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})}$
Matthews correlation coefficient (MCC)	$\frac{(\text{TP} \times \text{TN}) - (\text{FP} \times \text{FN})}{\sqrt{(\text{TP} + \text{FP}) \times (\text{TP} + \text{FN}) \times (\text{TN} + \text{FP}) \times (\text{TN} + \text{FN})}}$
Area under receiver operating characteristic curve (auROC)	Calculated using Wilcoxon–Mann–Whitney test statistics

training in each case was performed without the use of gene expression-based features.

Results

Comparison of our strategy with a known machine learning strategy for gene essentiality classification

To establish the predictive ability of our proposed pipeline in identifying the essentiality of a gene, the classification performance of our strategy was compared with that of a known strategy. Both were trained with two different input (training) datasets. The former dataset[14] is an available dataset that contains sequence and protein–protein interaction network features of *E. coli*. The latter dataset is our curated dataset containing sequence, gene expression and network topological features derived from *E. coli* K-12 MG1655 metabolism. The comparison of the two strategies was performed using different model performance metrics (see Materials and methods, Section 4 of the ESI,† Text S1).

Comparison using a known dataset. To assess the universality of our model's performance with any given dataset, training of our model was performed on a known training dataset and its performance was compared with the known strategy.[14] The known strategy involves the sequential minimal optimization (SMO) algorithm with linear kernel-based SVM classification for training the model on the dataset. Using our method on the known dataset,[14] a significantly improved classification performance was achieved as compared to the previously available strategy (Table 2). This can be observed from the improved MCC (0.675) and *F*-measure (0.826) values. The above comparison indicates that our model is superior in performance as compared to Hwang *et al.*'s method.[14] Our method was also compared with other types of supervised machine learning methods available for essential gene classification by using their training and test datasets (Table C, Section 5 of the ESI,† Text S1). The comparative results indicate that our method outperforms all the other known supervised classification methods, achieving a very high sensitivity and specificity.

Comparison using our curated dataset. Our curated dataset was also used as an alternative dataset for model training, and compared with the known strategy.[14] From Table 2, it can be observed that our proposed method again gave a comparatively better performance as compared to the known strategy. An increase in the MCC from 0.740 to 0.814 was observed, suggesting a significant difference in training accuracy.

Table 2 Comparison of our proposed strategy with Hwang *et al.* (2009)[14]

	Known dataset[14]		Our dataset	
Performance metric	Known strategy[a 14]	Our strategy	Known strategy[14]	Our strategy
Precision	0.828	0.877	0.846	0.907
Recall	0.745	0.78	0.903	0.906
F-Measure	0.784	0.826	0.874	0.906
MCC	0.593	0.675	0.740	0.814

[a] Performance measure as reported in Hwang *et al.* (2009)[14]

Improving model performance by class balancing and feature selection

To test the effects of feature selection and class balancing, four different classification scenarios were simulated and the corresponding performance was tested (Fig. 2A).

The four scenarios were:

(1) unbalanced dataset without feature selection,
(2) unbalanced dataset with feature selection,
(3) balanced dataset without feature selection, and
(4) balanced dataset with feature selection.

For comparing these scenarios, the MCC, auROC, TPR and FPR were computed (see Materials and methods and Section 4 of the ESI,† Text S1). It can be observed that training with the balanced dataset improves the performance of the model, indicated by the sharp increases of the MCC and auROC values. The TPR seems to remain unchanged whereas the FPR is drastically reduced after providing the balanced dataset to the model. Further, feature selection marginally improves the model's performance, as indicated by the increased MCC and auROC and the slightly decreased FPR. In all the above mentioned scenarios, the TPR seems to be relatively unchanged, suggesting that the model is able to predict the essentiality of genes that are actually known to be essential.

Contribution of "selected" features to model performance

In this study, 64 features were considered with respect to their biological relevance (see Materials and methods) in classifying essential genes within *E. coli*. To analyze the contribution of each type of feature towards essentiality, four different feature sets based on the type of feature were created, namely (i) genome/proteome sequence-based features, (ii) gene expression-based features, (iii) network topological features in the *E. coli* reaction network (RN) and (iv) network topological features in the *E. coli* flux-coupled subnetwork (Table 3 and Fig. 2B); each set was simulated separately for classification. For these analyses, feature selection was performed on each of the mentioned feature sets while training with the 1000 randomized datasets. The combination of the training dataset (reaction–gene pairs), selected features in each feature set (obtained after running SVM-RFE) and the optimized complexity parameter that gives the highest performance was chosen for each feature set and the model classification performance is reported in Table 3. If only expression features are used for training the model, the model performs poorly, with the prediction of a large number of false positives (FPR = 0.406), but predicts essential genes with high precision and recall. Training with only sequence features can predict essential genes with high precision and accuracy, and performs best among all the individual feature sets (auROC = 0.862). Even compared with other methods that use only sequence composition features in *E. coli* (auROC = 0.82 for 5-fold cross-validation),[44] our best model, when trained with sequence features shortlisted by feature selection, gives a higher area under the ROC curve (auROC = 0.862 for 10-fold cross-validation). To predict the essentiality of genes that are actually known to be essential, the RN feature set performed relatively better than the gene expression subset

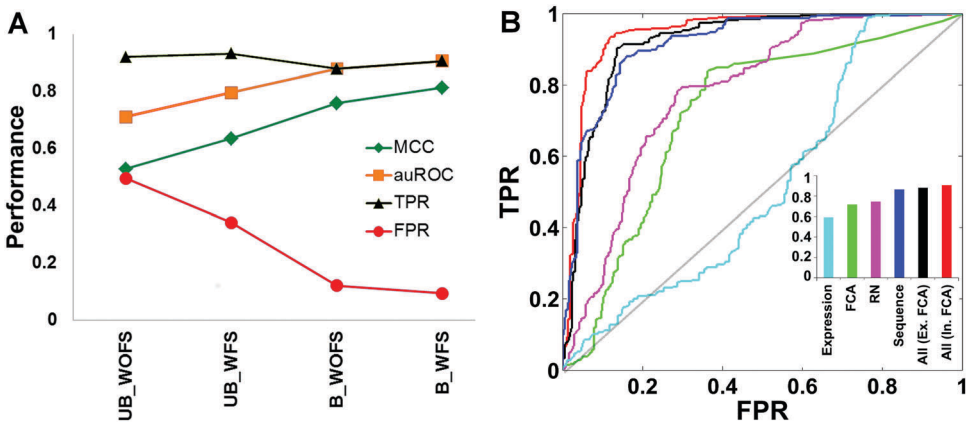


Fig. 2 Use of balanced training sets and contribution of features. (A) Effect of balancing the input training dataset and feature selection. Abbreviations: UB: unbalanced dataset, B: balanced dataset, WOFS: without feature selection, WFS: with feature selection. (B) Receiver operating characteristic (ROC) curves for essential gene prediction using each defined feature set. The inset indicates the auROC values when using each individual feature set.

Table 3 Effect of each feature type on model classification performance

Performance metric	Expression (2)	Sequence (14)	MN (5)	FCA (8)	All excluding FCA (22)	All including FCA (26)
TPR	0.594	0.862	0.747	0.717	0.88	0.906
FPR	0.406	0.138	0.253	0.283	0.12	0.094
Precision	0.657	0.862	0.75	0.722	0.881	0.907
Recall	0.594	0.862	0.747	0.717	0.88	0.906
F-Measure	0.548	0.862	0.747	0.716	0.88	0.906
MCC	0.243	0.724	0.497	0.439	0.761	0.814
auROC	0.594	0.862	0.747	0.717	0.88	0.906

The digits in parentheses indicate the number of features selected from the total number of features in each set.

with a sensitivity of 74%. The novel FCA features used in this work also perform comparably to the reaction network features, but provide a sub-optimal performance when given individually. Model performance with respect to both sensitivity and specificity (1-FPR) improves significantly when all features excluding FCA are given for our training strategy. By including FCA, the model's performance increases even further. These results support the use of unique, non-redundant, heterogeneous features for obtaining a high classification performance.

After applying the SVM-RFE feature selection technique to the whole feature set, a subset of 26 features was obtained (Section 6 of the ESI,† Text S1). These features represent the organism-specific determinants of essential genes in *E. coli* K-12 MG1655 metabolism. The list of selected features and their ranks as predicted by SVM-RFE has been given in Table D, Section 6 of the ESI,† Text S1. Additionally, the medians of each feature between the 384 essential and 3120 non-essential reaction–gene pairs were compared using the Wilcoxon rank-sum test (*P*-values are indicated in Table D, Section 6 of the ESI,† Text S1). Among the 26 best features, 21 features were significantly different ($P < 0.05$) for the essential and non-essential reaction–gene pairs. The differences in the distributions of feature values between the two classes are represented in Fig. 3. To test this, a correlation analysis was performed between each pair of features among the 26 selected features. The number of feature pairs that have weak correlations with each other (Spearman correlation: $\rho < 0.4$ and $\rho > -0.4$, $P < 0.05$) was found. The correlation matrix of the 26×26 features is given in the ESI,† Table S2. Around 71.69% of the selected feature pairs were weakly correlated with each other, thereby indicating the choice of diverse, independent biological features for appropriate classification of gene essentiality.

Performance of the model and effect of the input balanced training set

In the first scenario, where the best model was used for testing the whole unbalanced dataset, 362 out of 384 essential reaction–gene pairs (94.28%) and 2577 out of 3120 non-essential reaction–gene pairs (82.59%) were accurately classified (see ESI,† Table S3 to obtain the total reaction–gene classifications in both scenarios). In the second scenario, where the best chosen model (BFC_{best}, C_{best}) was used for training on the 1000 randomized training sets, the percentage of the total 1000 trained models that classified each reaction–gene pair as essential or non-essential was calculated (ESI,† Table S3). If at least 80% of the models predicted a reaction–gene pair to belong to a particular class, the gene was assumed to be a member of that class. For example, PHETRS_b1713 was predicted by 80.8% of the models to be essential and by 19.2% to be non-essential. Accordingly, this gene can be classified as essential. With respect to the threshold of 80% of the models for assigning essentiality, 356 out of 384 essential reaction–gene pairs (92.7%) and 2485 out of 3120 (79.64%) non-essential pairs were assigned appropriate essentiality phenotypes. Both the testing scenarios were also used for predicting

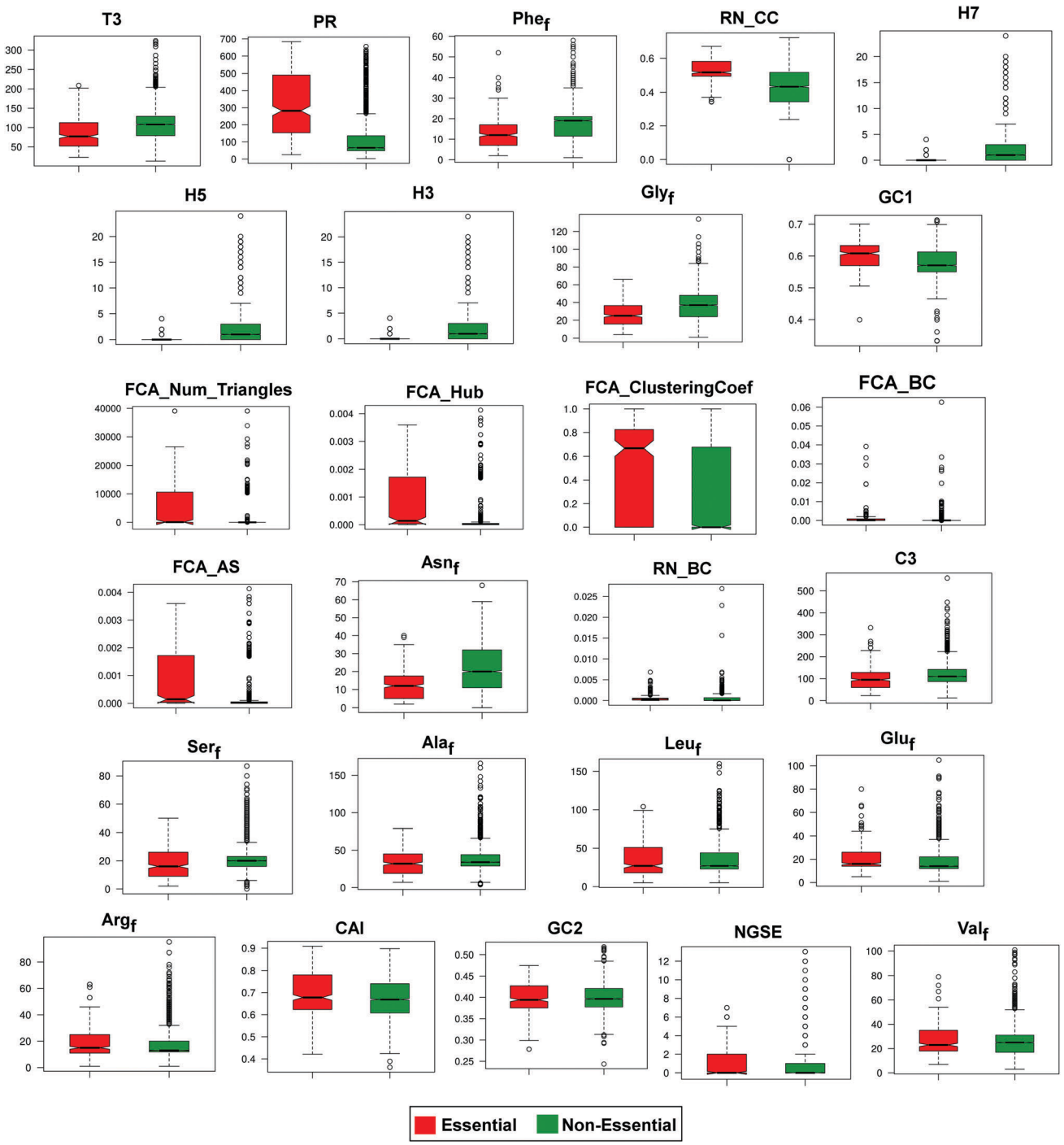


Fig. 3 Comparisons of distributions of the 26 selected features between the two classes. Each notched box plot represents a comparison between the essential (E) and non-essential (N) class based on a single feature. Outliers are indicated by circles in each plot.

the essentiality of the 590 reaction–gene pairs experimentally unidentified by Baba *et al.*[6]

Model performance for other less-studied organisms

To test the proposed methodology for organisms in which fewer or no organism-specific essentiality-based classification models are available, published datasets from two different prokaryotes, namely *Brevundimonas subvibriodes* ATCC 15264 and *Helicobacter pylori* 26695, were further used for training our model. The best trained model after feature selection and 10-fold cross-validation as given in the proposed methodology displayed a high sensitivity and a low FPR (Table 4). The MCC values are above 0.5 in both cases, demonstrating comparable accuracy across organisms. The auROC values are lower as compared to the *E. coli* trained best model. The possible reason for this could be that not all the features were

Table 4 Model evaluation metrics for two less-studied prokaryotes

Performance metrics	*B. subvibrioides* ATCC 15264	*H. pylori* 26695
TPR	0.783	0.753
FPR	0.217	0.247
Precision	0.784	0.758
Recall	0.783	0.753
F-Measure	0.783	0.752
MCC	0.567	0.512
auROC	0.783	0.752

available for the chosen less-studied organisms in comparison to *E. coli*.

The complete dataset was given as an unknown prediction set for the best trained model for both organisms (prediction results are given in the ESI,† Table S3). In the case of *B. subvibrioides* ATCC 15264, 102 out of 136 essential and 368 out of 486 non-essential genes were correctly classified. In the case of *H. pylori* 26695, 60 out of 73 essential and 174 out of 329 non-essential reaction–gene pairs were correctly classified. It can be observed that the best trained model in both the organisms is highly sensitive in predicting true essential genes. Apart from the known essential and non-essential genes, the essentiality of 32 genes in *B. subvibrioides* ATCC 15264 and 3 genes in *H. pylori* 26695, for which essentiality information was not reported in the DEG database,[42] was predicted for the first time (ESI,† Table S3).

Discussion

As introduced above, the choice of an appropriate training dataset, a flexible and accurate learning algorithm and biologically relevant features that define the essentiality of a gene is absolutely necessary for the accurate essentiality-based classification of genes using supervised machine learning techniques. With respect to these primary requirements, here, we present a simple but comprehensive computational strategy that integrates genotype–phenotype characteristics of *E. coli* K-12 MG1655 metabolism to classify a gene and its corresponding reaction based on their dispensability. Also, due to the universal set of features curated for classification, our method can predict minimally essential genes. Here, genotype characteristics correspond to the features calculated from the nucleotide and amino acid sequences that represent the static genome and proteome complements for prediction. Likewise, gene expression, metabolic network and flux-based features represent the metabolic phenotype complement of the organism that results from interactions of the genotype with the environment.

As our method is a supervised machine-learning strategy, we showcase the predictive capability by comparing it with previously available supervised classification strategies for essential gene classification.[14,16,33] The comparisons indicate that irrespective of the input training dataset used, our model's classification always outperforms all other methods and achieves the highest sensitivity and specificity. Also, our analysis indicates that the SVM model performance is highly dependent on the balance of the input training dataset. Hence, model training needs to be performed on a large number of randomly sampled balanced datasets that can remove sampling bias and choose the best training set sample of instances that is representative of the whole population of genes within that organism. Hence, our strategy ensured that a large number of sample balanced training sets (1000 sets) were generated, such that a particular gene is sampled at least once. Also, the training set of instances in our model represents the reaction–gene combinations. Thus, instead of predicting only essential genes, our model can also predict the metabolic reaction that the gene is associated with, for which it was predicted to be essential.

Training the model with specific feature sets indicates that the sequence-based features are the most predictive of gene essentiality in *E. coli* K-12 MG1655 metabolism. Out of the 26 selected features, homology-based features like phyletic retention[8] and the number of homologs in other organisms contribute the most to classifying genes in *E. coli* K-12 MG1655 metabolism based on their essentiality, as ranked by SVM-RFE. This indicates that gene essentiality is largely linked to evolutionary preference among homologous bacterial species. This is followed by the CAI, GC content and NGSE, which are also significantly high for essential genes. Essential genes have been previously shown to demonstrate a higher expression rate as compared to non-essential genes across bacteria.[45] The CAI is a predictor of protein abundance and the NGSE indicates that a gene sharing a common pattern of expression with a large number of other genes tends to be more essential. The median frequencies of glycine, asparagine and phenylalanine are typically low, whereas the frequencies of arginine, glutamate and valine are typically high in essential enzymes. This observation is supported by a previous study, where similar correlations in the *E. coli* K-12 genome were observed for essential enzymes.[27]

The definition of an essential metabolic gene is highly dependent on the environmental context of the cell. To circumvent this problem, for the first time, flux coupling analysis was performed on the obtained iJO1366 network to obtain a flux-coupled sub-graph that is universal across environments. The topological features of the identified sub-network were used as features in model training. Model training with only FCA-based network features gave a competitive model performance. Further, in conjunction with sequence-based, expression-based and reaction network-based features, the FCA features significantly improved the model's sensitivity and specificity.

Our results also show that essential reaction–gene pairs exhibit high hub and authority scores, and high clustering coefficients and betweenness centralities within a physiological flux-coupled sub-network. These features indicate that a large number of reactions demonstrate metabolic flux dependence on essential reactions. Such reactions also have a tendency to form flux-coupled modules, so that metabolites generated or consumed by this reaction can be regained once distributed into other reactions within the same module, while maintaining an energy or redox balance. Essential reactions also act as connecting links between physiologically important flux-modules. These reactions are probably metabolically important as they catalyze the conversion of highly connected metabolites, like ATP,

H_2O, H^+ and other cofactors, which are used in their coupled reactions. Some of these reactions (or genes) may also represent the first committed step of the conversion of a terminal metabolite of one physiological module to a substrate to enter the other module. For example, acetyl co-A carboxylase, which is both the first committed and rate-limiting step of fatty acid/lipid biosynthesis,[46] demonstrates a high betweenness centrality in the obtained physiological flux-coupled subnetwork. However, it still remains to be established whether there are any relationships between homology-based, amino acid frequency-based and flux-coupled subnetwork features; once established, the essentiality predictions from our pipeline could be extrapolated to reveal the evolutionary constraints faced by essential genes as compared to non-essential genes.

Finally, the model was also tested for the experimentally known, complete, gold standard dataset of essential and non-essential genes in the *E. coli* Keio collection,[6] in which the essentiality information for each gene was available. With respect to the test set, the best model could predict 362 out of 384 known essential reaction–gene pairs (94.28%) and 2577 out of 3120 non-essential reaction–gene pairs (82.59%), which is indicative of the sensitivity of the model to detect true essential and non-essential reaction–gene pairs. Further, the essentiality of the 590 experimentally unknown reaction–gene pairs from a previous study by Baba *et al.* was also predicted using our strategy. Yamamoto *et al.*[47] provided an update for the *E. coli* K-12 MG1655 Keio collection, where the essentiality of a few of these unknown reaction–gene pairs was experimentally determined. Their new update improves the annotation of five metabolic genes, namely alanyl-tRNA synthetase (b2697), pantothenate kinase (b3974), dephospho-coA kinase (b0103), isoleucyl tRNA synthetase (b0026), and phosphoglucosamine mutase (b3176), which was previously unknown, as essential. Our best model was precisely able to predict all the corresponding reaction–gene pairs associated with the mentioned genes to be essential (ESI,† Table S3). Considering the best model trained with 1000 balanced datasets, 98–100% of the generated models predicted the above genes to be essential. Further, our strategy was also able to predict the essentiality of 317 genes (out of which the essentiality of 235 genes can be predicted by FCA-based network features alone) which could not be determined by Baba *et al.*[6] or Yamamoto *et al.*[47] The roles of a few of these genes have also been discussed elsewhere, although in different biological contexts. A few mentionable examples that were predicted to be essential by our machine learning framework include the ubiE enzyme (gene: b3833, reaction: AMMQLT8), which catalyzes the carbon methylation reaction in the biosynthesis of ubiquinone and menaquinone, which are essential within the respiratory chain,[48] the iron–sulfur cluster YtfE (gene: b4209, reaction: FESR), which is necessary for repairing damaged iron–sulfur clusters under oxidative or nitrosative stress conditions,[49] the β-ketoacyl carrier protein synthase III fabH (gene: b1091, reaction: KAS15), which catalyzes the condensation reaction in the initiation of type II fatty acid synthesis in bacteria,[50] the ribosome small subunit-dependent GTPase rsgA (gene: b4161, reaction: NTP3), which is required for ribosomal subunit assembly and 16S-rRNA processing,[51] and triose-phosphate isomerase tpiA (gene: b3919, reaction: TPI), which plays a crucial role in the isomerization of triose-phosphate isomers within glycolysis.[52] These examples indicate the applicability of our designed methodology, given any random set of training instances, to predict novel essential gene candidates and thereby provide experimentally testable hypotheses for further validation. The functional reasons for these genes to be essential can be further probed using the features selected by our model strategy. Our methodology also provides comparable results in less-studied organisms, like *Brevundimonas subvibrioides* ATCC 15264 and *Helicobacter pylori* 26695, for which fewer or no organism-specific machine learning studies were previously available.

All the above results indicate the strength of our model in identifying true essential genes, asserting the usage of a balanced training dataset, and the selection of biologically relevant features to represent gene essentiality and optimal parameters for hyperplane formation to classify essential genes while using a fundamental machine learning-based scheme. Further, given the challenges faced by experimental biologists to identify essential genes, the novel putative essential genes and their associated features can provide fresh impetus for further targeted studies in *Escherichia coli* and other related organisms.

Acknowledgements

This work is supported by a grant [BT/PR14958/BID/7/537/2015] from the Department of Biotechnology, Government of India. SN acknowledges the DST-INSPIRE Junior Research Fellowship from DST. AS acknowledges the Senior Research Fellowship from DBT-BINC. The authors also thank Prof. Hsuan-Cheng Huang, National Yang-Ming University, Taipei, for providing the training dataset for comparative analysis.

References

1 T. Ding, K. A. Case, M. A. Omolo, H. A. Reiland, Z. P. Metz, X. Diao and D. J. Baumler, *PLoS One*, 2016, **11**, e0149423.
2 M. Juhas, L. Eberl and G. M. Church, *Trends Biotechnol.*, 2012, **30**, 601–607.
3 O. Cohen, M. Oberhardt, K. Yizhak and E. Ruppin, *PLoS One*, 2016, **11**, e0168444.
4 M. Juhas, L. Eberl and J. I. Glass, *Trends Cell Biol.*, 2011, **21**, 562–568.
5 A. R. Joyce, J. L. Reed, A. White, R. Edwards, A. Osterman, T. Baba, H. Mori, S. A. Lesely, B. Ø. Palsson and S. Agarwalla, *J. Bacteriol.*, 2006, **188**, 8259–8271.
6 T. Baba, T. Ara, M. Hasegawa, Y. Takai, Y. Okumura, M. Baba, K. A. Datsenko, M. Tomita, B. L. Wanner and H. Mori, *Mol. Syst. Biol.*, 2006, **2**, 2006.0008.
7 A. Cruz, C. M. Coburn and S. M. Beverley, *Proc. Natl. Acad. Sci. U. S. A.*, 1991, **88**, 7170–7174.

8 S. Gerdes, M. D. Scholle, J. W. Campbell, G. Balázsi, E. Ravasz, M. D. Daugherty, A. L. Somera, N. C. Kyrpides, I. Anderson, M. S. Gelfand, A. Bhattacharya, V. Kapatral, M. D'Souza, M. V. Baev, Y. Grechkin, F. Mseeh, M. Fonstein, R. Overbeek, A.-L. Barabási, Z. N. Oltvai and A. L. Osterman, *J. Bacteriol.*, 2003, **185**, 5673–5684.
9 W. S. Reznikoff and K. M. Winterberg, *Microbial Gene Essentiality: Protocols and Bioinformatics*, Springer, 2008, vol. 416, pp. 13–26.
10 N. Agrawal, P. V. N. Dasaradhi, A. Mohmmed, P. Malhotra, R. K. Bhatnagar and S. K. Mukherjee, *Microbiol. Mol. Biol. Rev.*, 2003, **67**, 657–685.
11 A. M. Gustafson, E. S. Snitkin, S. C. J. Parker, C. DeLisi and S. Kasif, *BMC Genomics*, 2006, **7**, 1.
12 J. P. M. da Silva, M. L. Acencio, J. C. M. Mombach, R. Vieira, J. C. da Silva, N. Lemke and M. Sinigaglia, *Phys. A*, 2008, **387**, 1049–1055.
13 K. Plaimas, J.-P. Mallm, M. Oswald, F. Svara, V. Sourjik, R. Eils and R. König, *BMC Syst. Biol.*, 2008, **2**, 67.
14 Y.-C. Hwang, C.-C. Lin, J.-Y. Chang, H. Mori, H.-F. Juan and H.-C. Huang, *Mol. Biosyst.*, 2009, **5**, 1672–1678.
15 K. Plaimas, R. Eils and R. König, *BMC Syst. Biol.*, 2010, **4**, 1.
16 J. Deng, L. Deng, S. Su, M. Zhang, X. Lin, L. Wei, A. A. Minai, D. J. Hassett and L. J. Lu, *Nucleic Acids Res.*, 2011, **39**, 795–807.
17 J. Cheng, W. Wu, Y. Zhang, X. Li, X. Jiang, G. Wei and S. Tao, *BMC Genomics*, 2013, **14**, 910.
18 L. K. Smith, M. J. Gomez, K. Y. Shatalin, H. Lee and A. A. Neyfakh, *Genome Biol.*, 2007, **8**, R87.
19 S. Theodoridis, A. Pikrakis, K. Koutroumbas and D. Cavouras, *Introduction to pattern recognition: a MATLAB approach*, Academic Press, 2010.
20 M. Maalouf and T. B. Trafalis, *Comput. Stat. Data Anal.*, 2011, **55**, 168–183.
21 K. I. Sofeikov, I. Y. Tyukin, A. N. Gorban, E. M. Mirkes, D. V. Prokhorov and I. V. Romanenko, *IJCNN*, IEEE, 2014, ISBN: 978-1-4799-1484-5 3548–3555.
22 P. N. Tan, M. Steinbach and V. Kumar, *Classification: Alternative Techniques. Introduction to Data Mining*, 2013.
23 R. Akbani, S. Kwek and N. Japkowicz, *European conference on machine learning*, Springer, 2004, pp. 39–50.
24 X. Zhang, M. L. Acencio and N. Lemke, *Front. Physiol.*, 2016, **7**, 1–11.
25 J. Wang, W. Peng and F.-X. Wu, *Proteomics: Clin. Appl.*, 2013, **7**, 181–192.
26 S. Mann and Y. P. P. Chen, *Genomics*, 2010, **95**, 7–15.
27 X. Gong, S. Fan, A. Bilderbeck, M. Li, H. Pang and S. Tao, *Mol. Genet. Genomics*, 2008, **279**, 87–94.
28 K. Raman, N. Damaraju and G. K. Joshi, *Syst. Biol. Synth. Biol.*, 2014, **8**, 73–81.
29 J. D. Orth, I. Thiele and B. Ø. Palsson, *Nat. Biotechnol.*, 2010, **28**, 245–248.
30 B. Papp, R. A. Notebaart and C. Pál, *Nat. Rev. Genet.*, 2011, **12**, 591–602.
31 A. P. Burgard, E. V. Nikolaev, C. H. Schilling and C. D. Maranas, *Genome Res.*, 2004, **14**, 301–312.
32 A. Larhlimi, L. David, J. Selbig and A. Bockmayr, *BMC Bioinf.*, 2012, **13**, 57.
33 K. Song, T. Tong and F. Wu, *Integr. Biol.*, 2014, **6**, 460–469.
34 J. D. Orth, T. M. Conrad, J. Na, J. A. Lerman, H. Nam, A. M. Feist and B. Ø. Palsson, *Mol. Syst. Biol.*, 2011, **7**, 535.
35 D. A. Benson, M. Cavanaugh, K. Clark, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell and E. W. Sayers, *Nucleic Acids Res.*, 2013, **41**, D36–D42.
36 M. Y. Galperin, K. S. Makarova, Y. I. Wolf and E. V. Koonin, *Nucleic Acids Res.*, 2015, **43**(D1), D261–D269.
37 I. Guyon and A. Elisseeff, *J. Mach. Learn. Res.*, 2003, **3**, 1157–1182.
38 I. Guyon, J. Weston, S. Barnhill and V. Vapnik, *Mach. Learn.*, 2002, **46**, 389–422.
39 Y. Yu, L. Yang, Z. Liu and C. Zhu, *Mol. Biosyst.*, 2017, **13**, 577–584.
40 M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, *ACM SIGKDD Explor. Newsl.*, 2009, **11**, 10–18.
41 J. C. Platt, *Adv. Kernel Methods*, 1999, 185–208.
42 H. Luo, Y. Lin, F. Gao, C.-T. Zhang and R. Zhang, *Nucleic Acids Res.*, 2014, **42**, D574–D580.
43 I. Thiele, T. D. Vo, N. D. Price and B. Ø. Palsson, *J. Bacteriol.*, 2005, **187**, 5818–5830.
44 L. W. Ning, H. Lin, H. Ding, J. Huang, N. Rao and F. B. Guo, *GMR, Genet. Mol. Res.*, 2014, **13**, 4564–4572.
45 A. L. Grazziotin, N. M. Vidal and T. M. Venancio, *FEBS J.*, 2015, **282**, 3395–3411.
46 M. S. Davis, J. Solbiati and J. E. Cronan, *J. Biol. Chem.*, 2000, **275**, 28593–28598.
47 N. Yamamoto, K. Nakahigashi, T. Nakamichi, M. Yoshino, Y. Takai, Y. Touda, A. Furubayashi, S. Kinjyo, H. Dose and M. Hasegawa, *et al.*, *Mol. Syst. Biol.*, 2009, **5**, 335.
48 P. T. Lee, A. Y. Hsu, H. T. Ha and C. F. Clarke, *J. Bacteriol.*, 1997, **179**, 1748–1754.
49 M. C. Justino, C. C. Almeida, M. Teixeira and L. M. Saraiva, *J. Biol. Chem.*, 2007, **282**, 10352–10359.
50 C. Y. Lai and J. E. Cronan, *J. Biol. Chem.*, 2003, **278**, 51494–51503.
51 Y. Hase, S. Yokoyama, A. Muto and H. Himeno, *RNA*, 2009, **15**, 1766–1774.
52 R. S. V. Selvamani, M. Telaar, K. Friehs and E. Flaschel, *Microb. Cell Fact.*, 2014, **13**, 58.
53 S. Mann and Y. P. P. Chen, *Genomics*, 2010, **95**, 7–15.
54 M. dos Reis, L. Wernisch and R. Savva, *Nucleic Acids Res.*, 2003, **31**, 6976–6985.
55 P. M. Sharp and W. H. Li, *Nucleic Acids Res.*, 1987, **15**, 1281–1295.
56 A. Subramanian and R. R. Sarkar, *Genomics*, 2015, **106**, 232–241.
57 F. Wright, *Gene*, 1990, **87**, 23–29.
58 P. M. Sharp, E. Bailes, R. J. Grocock, J. F. Peden and R. E. Sockett, *Nucleic Acids Res.*, 2005, **33**, 1141–1153.
59 P. Rice, I. Longden and A. Bleasby, *Trends Genet.*, 2000, **16**, 276–277.
60 O. Ish-Am, D. M. Kristensen and E. Ruppin, *PLoS One*, 2015, **10**, e0123785.

61 I. K. Jordan, I. B. Rogozin, Y. I. Wolf and E. V. Koonin, *Genome Res.*, 2002, **12**, 962–968.
62 *E. coli* Gene Expression Database (GenExpDB), https://genexpdb.ou.edu/.
63 H. Yu, P. M. Kim, E. Sprecher, V. Trifonov and M. Gerstein, *PLoS Comput. Biol.*, 2007, **3**, e59.
64 A. Subramanian and R. R. Sarkar, *Proc. Int. Symp. Math. Comput. Biol. BIOMAT 2015, World Sci.*, 2015, ISBN: 978-981-3141-90-2, 1-20.
65 G. del Rio, D. Koschützki and G. Coello, *BMC Syst. Biol.*, 2009, **3**, 1.
66 P. I. Wang and E. M. Marcotte, *J. Proteomics*, 2010, **73**, 2277–2289.
67 S. Gerdes, M. D. Scholle, J. W. Campbell, G. Balazsi, E. Ravasz, M. D. Daugherty, A. L. Somera, N. C. Kyrpides, I. Anderson and M. S. Gelfand, *et al.*, *J. Bacteriol.*, 2003, **185**, 5673–5684.
68 E. Almaas, *J. Exp. Biol.*, 2007, **210**, 1548–1558.
69 A. N. Chang, *Protein Networks and Pathway Analysis*, Springer, 2009, vol. 563, pp. 141–156.
70 M. Bastian, S. Heymann and M. Jacomy, *et al.*, *Proc. Third Int. ICWSM Conf.*, 2009, **8**, 361–362.
71 K. S. Jeong, J. Ahn and A. B. Khodursky, *Genome Biol.*, 2004, **5**, 1.
72 P. Boccazzi, A. Zanzotto, N. Szita, S. Bhattacharya, K. F. Jensen and A. J. Sinskey, *Appl. Microbiol. Biotechnol.*, 2005, **68**, 518–532.
73 J. A. Bernstein, A. B. Khodursky, P. H. Lin, S. Lin-Chao and S. N. Cohen, *Proc. Natl. Acad. Sci. U. S. A.*, 2002, **99**, 9697–9702.
74 D. P. Sangurdekar, F. Srienc and A. B. Khodursky, *Genome Biol.*, 2006, **7**, 1.
75 A. G. Franchini and T. Egli, *Microbiology*, 2006, **152**, 2111–2127.
76 J. D. Partridge, C. Scott, Y. Tang, R. K. Poole and J. Green, *J. Biol. Chem.*, 2006, **281**, 27806–27815.
77 P. Baldi, S. Brunak, Y. Chauvin, C. A. F. Andersen and H. Nielsen, *Bioinformatics*, 2000, **16**, 412–424.

PLOS ONE

RESEARCH ARTICLE

Essential gene prediction using limited gene essentiality information–An integrative semi-supervised machine learning strategy

Sutanu Nandi[1,2], **Piyali Ganguli**[1,2], **Ram Rup Sarkar**[1,2]*

1 Chemical Engineering and Process Development, CSIR-National Chemical Laboratory, Pune, Maharashtra, India, **2** Academy of Scientific & Innovative Research (AcSIR), Ghaziabad, India

* rr.sarkar@ncl.res.in

Citation: Nandi S, Ganguli P, Sarkar RR (2020) Essential gene prediction using limited gene essentiality information–An integrative semi-supervised machine learning strategy. PLoS ONE 15(11): e0242943. https://doi.org/10.1371/journal.pone.0242943

Editor: Seyedali Mirjalili, Torrens University Australia, AUSTRALIA

Received: June 10, 2020

Accepted: November 12, 2020

Published: November 30, 2020

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: https://doi.org/10.1371/journal.pone.0242943

Data Availability Statement: All relevant data are within the manuscript and its Supporting Information files.

Abstract

Essential gene prediction helps to find minimal genes indispensable for the survival of any organism. Machine learning (ML) algorithms have been useful for the prediction of gene essentiality. However, currently available ML pipelines perform poorly for organisms with limited experimental data. The objective is the development of a new ML pipeline to help in the annotation of essential genes of less explored disease-causing organisms for which minimal experimental data is available. The proposed strategy combines unsupervised feature selection technique, dimension reduction using the Kamada-Kawai algorithm, and semi-supervised ML algorithm employing Laplacian Support Vector Machine (LapSVM) for prediction of essential and non-essential genes from genome-scale metabolic networks using very limited labeled dataset. A novel scoring technique, Semi-Supervised Model Selection Score, equivalent to area under the ROC curve (auROC), has been proposed for the selection of the best model when supervised performance metrics calculation is difficult due to lack of data. The unsupervised feature selection followed by dimension reduction helped to observe a distinct circular pattern in the clustering of essential and non-essential genes. LapSVM then created a curve that dissected this circle for the classification and prediction of essential genes with high accuracy (auROC > 0.85) even with 1% labeled data for model training. After successful validation of this ML pipeline on both Eukaryotes and Prokaryotes that show high accuracy even when the labeled dataset is very limited, this strategy is used for the prediction of essential genes of organisms with inadequate experimentally known data, such as *Leishmania sp*. Using a graph-based semi-supervised machine learning scheme, a novel integrative approach has been proposed for essential gene prediction that shows universality in application to both Prokaryotes and Eukaryotes with limited labeled data. The essential genes predicted using the pipeline provide an important lead for the prediction of gene essentiality and identification of novel therapeutic targets for antibiotic and vaccine development against disease-causing parasites.

Funding: We thank SERB, Department of Science and Technology, Govt. of India (DST/ICPS/EDA/2018) and DBT, Department of Biotechnology, Govt. of India (File No. BT/PR14958/BID/7/537/2015), for providing financial support to Ram Rup Sarkar. Sutanu Nandi acknowledges DST-INSPIRE for Senior Research Fellowship. Piyali Ganguli acknowledges the Council of Scientific & Industrial Research (CSIR) for the Senior Research Fellowship. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

1. Introduction

Gene essentiality information of disease-causing organisms that throws light on the minimally essential genes that are absolutely required for the survival of the organism under any environmental condition has not only been indispensable for the prediction of novel therapeutic targets for antibiotic and vaccine development but has also contributed towards industrial bioprocessing, food microbiology, and bioremediation. However, experimental techniques [1–5] like genetic foot-printing, gene knockouts, RNA interference (RNAi), transposon mutagenesis have been employed to perform a genome-wide screen to check for gene essentiality are expensive, labor-intensive, as well as time-consuming.

As an efficient alternative to these highly complex experimental strategies, researchers now are employing computational techniques based on homology mapping, constraint-based modeling strategies, and machine learning strategies [6–8]. The homology-based essential gene prediction methods rely on the fact that essential genes are less likely to evolve, tend to remain conserved, and are often shared by distantly related organisms. Essential genes have been identified by comparative genomic analysis in different bacterial species such as Mycoplasma [9], *Liberibacter* [10], *Plasmodium falciparum* [11], and *Brucella spp*. [12]. However, the limitation of this method is that the conserved ortholog genes between different species form only a small fraction of the entire genome [13]. Also, it has been observed that highly conserved genes across different species are not always essential, as gene essentiality also depends on different environmental conditions where the organism resides.

Constraint-based modeling strategies, such as Flux Balance Analysis (FBA), employ genome-scale reconstructed metabolic networks to predict the metabolic fluxes at steady-state. This methodology is widely used for predicting essential genes by performing *in-silico* knockout of a gene and estimating its corresponding lethality [14–16]. A limitation of this FBA method is that only a limited number of environmental conditions can be considered for a certain biomass equation (or objective function) with respect to gene essentiality.

On the other hand, Machine Learning (ML) strategies comprise various data-driven approaches that train a model from the inherent patterns of the training data and make a prediction for the unlabeled data. These ML algorithms can be broadly grouped under supervised, semi-supervised, and unsupervised strategies [17,18]. The supervised strategies such as Decision Tree, Naïve Bayes, Support Vector Machine (SVM), etc. require sufficient amounts of labeled data for model training. In contrast, the unsupervised method relies on clustering algorithms (e.g., K-Means Clustering), where no labeled data is required. The semi-supervised ML algorithms that comprise Generative Models, Self-Training, Transductive SVM, and Laplacian SVM combine the potential of both supervised and unsupervised ML strategies and can train the model with a very limited amount of labeled data. At the same time, optimization of the hyper-parameter is crucial for enhancing the predictive performance of these machine learning classifiers. Various meta-heuristic techniques, such as Particle Swarm Optimization (PSO) [19], Genetic Algorithm (GA) [20], Ant Colony Optimization (ACO) [21], Grey Wolf Optimizer (GWO) [22], Ant Lion Optimizer (ALO) [23], etc. have been used for hyper-parameter tuning.

Based on the availability of labeled data of essential genes, researchers have employed supervised machine learning strategies [6–8] as well as deep learning-based strategies to predict essential genes [24,25]. The key advantage of these strategies lies in the fact that these models are capable of capturing the inherent patterns of a large array of biologically relevant 'features' that are distinctive and reflect the heterogeneous properties of essential genes. Supervised machine learning classifiers such as logistic regression [26,27], support vector machine [28–31], random forest [32], decision tree [26], ensemble [26] and probabilistic Bayesian-based

methods [26,27,33] and instance-based learning methods such as K Nearest neighbor (K-NN) and Weighted KNN (WKNN) [34] have been used for gene essentiality prediction. Deep Learning strategies based on multilayer perceptron networks have also been used for essential gene prediction [24,35]. In these studies, researchers have mostly opted for simpler optimization methods for parameter tuning, such as the grid search technique, where the entire parameter space is explored in all possible combinations.

These machine learning-based classifiers predict gene essentiality of unannotated genes based on the pattern of the features of previously annotated genes that have been verified experimentally and labeled as essential and non-essential. In order to achieve this, researchers have curated different combinations of features. Most of the machine learning approaches use calculated features either from coding sequences [36–38] or network (e.g., protein interaction network, metabolic network) topological features [6,39] or both. Features, such as amino acid frequency and protein length computed from protein sequence, and codon adaptation index (CAI), Effective Number of Codons (ENC), Phyletic Retention (PR), GC content computed from nucleotide sequence are some of the known features of gene essentiality across bacteria [28,29,40]. Protein interaction networks (PIN) have been used to calculate topological network features to classify gene essentiality [28,39]. However, these strategies fail for many organisms that do not hold the idea of the centrality-lethality hypothesis in a PIN [41]. On the other hand, few studies have used flux-based features derived from metabolic networks to classify genes [29,30] that have been calculated under a single environmental condition that does not represent a universal set of features. Detailed reviews of the existing machine learning strategies for gene essentiality prediction have been discussed in different works of literature [6–8].

A major drawback of these existing machine learning algorithms for essential gene prediction is that they require a large amount of these labeled data that helps to train these models for an accurate prediction of the essentiality of unannotated genes, and show very poor performance when the labeled data set is imbalanced or limited. To circumvent these problems, in our previous study, an integrative machine learning strategy has been developed using a combination of feature selection algorithm, Support Vector Machine- Recursive Feature Elimination (SVM-RFE) [42] and classifier, Sequential Minimal Optimization (SMO) [43] for gene essentiality prediction in the metabolism of *Escherichia coli*, which performed well on imbalanced data set with diverse features computed from flux coupled connected sub-network along with other sequence-based features [40]. Here, the advantages of using the Flux Coupling Analysis (FCA) based feature for the prediction of gene essentiality with high accuracy and confidence have been reported. FCA analysis help to capture the physiological dependence of one gene-reaction combination on another, which is coupled to it, under all input exchanges of a reaction, representing all possible environmental conditions, thereby helping the classifier to accurately identify the minimally essential genes that are absolutely crucial for sustaining the metabolic demands of the cell to ensure its survival [40]. However, this technique was unable to predict gene essentiality when a very small amount of experimentally verified labeled data are available.

To mitigate the problems inherent in the existing strategies, we propose an integrative semi-supervised machine learning strategy based on Laplacian SVM [44] for the classification of genes using gene sequence, protein sequence, network topological, and flux-based features with very limited labeled data on gene essentiality of metabolic networks for both Prokaryotic and Eukaryotic organisms. Another objective of this work is the development of a new machine learning pipeline to help in the annotation of essential genes of less explored organisms, like *Leishmania donovani* and *Leishmania major*, the causative organisms for the neglected tropical disease Leishmaniasis, for which very limited experimental data is available. By using the available tools and techniques, the prediction of gene essentiality and targeted

therapy for the disease becomes extremely difficult [45]. In the present work, it is hypothesized that using these diverse features, like topological network features of both the genome-scale metabolic reaction network as well as the flux-coupled sub-networks, together with the sequence-based features simultaneously, that can capture both the properties of genotype and phenotype and by employing the proposed algorithm, it is possible to predict the essentiality of uncharacterized genes with high accuracy even in the cases where labeled data is limited. This is in contrast to other machine learning pipelines for essential gene prediction that relies on only sequence-based features and has been applied to only Prokaryotes [26,46]. In this work, the novel features derived from the genome-scale metabolic reaction network, as well as the flux-coupled sub-networks, contribute towards the better prediction of gene essentiality by capturing the contribution of a gene in sustaining the metabolic demands of the cell under varied environmental challenges that are indispensable for its survival. A new scoring technique has also been proposed, called the Semi-Supervised Model Selection Score (SSMSS) that correlates well with Mathews Correlation Coefficient (MCC) [47] and can be used for the selection of the best model when the calculation of supervised performance metrics like MCC or auROC is difficult due to lack of experimental data. After the successful validation of this proposed pipeline on twelve organisms, with well-annotated genes essentiality information, using as low as 1% labeled data on two types of training datasets (i.e., with 80% training and 20% blind datasets, as well as using the whole dataset for training), the essential genes in *Leishmania* have been predicted as well as categorized the reaction-gene pairs in five different groups based Gene-Protein-Reaction (GPR) association in metabolism. These groups depict the association of the reactions with different combinations of essential and non-essential genes, which throws light on the probable reaction-gene combination that can be used for targeted therapy. This study promises to lay the foundation to the prediction of gene essentiality information for less explored organisms that will help experimental biologists to identify novel therapeutic targets even when only limited information is available.

2. Methods

The Machine learning strategy developed to predict gene essentiality, as elucidated in Fig 1, combines feature selection technique based on a space-filling concept, dimension reduction (DR) using the Kamada-Kawai (KK) algorithm, and classification of genes based on a semi-supervised machine learning algorithm employing Laplacian Support Vector Machine (LapSVM). This pipeline combines heterogeneous biological features, such as sequence-based, as well as network-based features. It classifies genes based on a training dataset of very limited information of essential genes from experimental data. Twelve organisms comprising of both Prokaryotes and Eukaryotes (Table 1) with well-annotated genes essentiality information from the OGEE database [48] have been considered for the validation of this proposed strategy, and the subsequent prediction of essential genes in *Leishmania major* and *Leishmania donovani* have been performed. The gene essentiality information has only been considered from the OGEE database as this collates data using text mining as well as manually verified with experimental data, unlike other gene essentiality databases that rely on only text mining.

2.1 Training data and Testing data set preparation and integration of heterogeneous features

The training datasets for the pipeline of the 12 target organisms were prepared by calculating mainly two types of features: topological features and sequenced based features. These features were extracted primarily from the genome-scale reconstructed metabolic networks, the fasta files containing the coding nucleotide sequences of the genes, and protein sequences of these

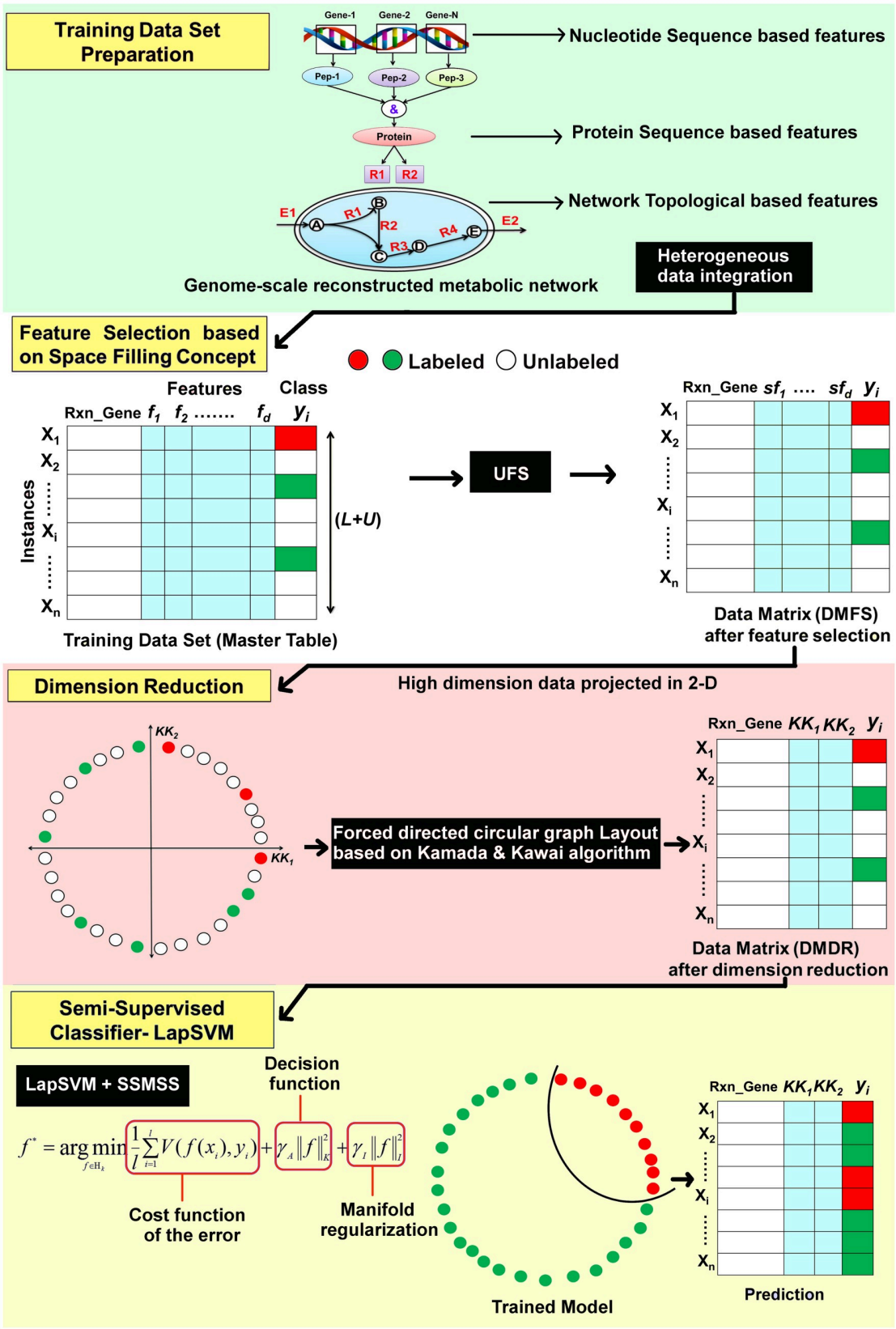

Training Data Set Preparation
Gene-1
Gene-2
Gene-N
Pep-1
Pep-2
Pep-3
&
Protein
R1
R2
E1
E2
R3
R4
A
B
C
D
E
Nucleotide Sequence based features
Protein Sequence based features
Network Topological based features
Genome-scale reconstructed metabolic network
Heterogeneous data integration
Feature Selection based on Space Filling Concept
Labeled
Unlabeled
Features
Class
Rxn_Gene f_1 f_2 ..... f_d y_i
Instances
X_1
X_2
X_i
X_n
(L+U)
Training Data Set (Master Table)
UFS
Rxn_Gene sf_1 .... sf_d y_i
Data Matrix (DMFS) after feature selection
Dimension Reduction
High dimension data projected in 2-D
KK_2
KK_1
Forced directed circular graph Layout based on Kamada & Kawai algorithm
Rxn_Gene KK_1 KK_2 y_i
Data Matrix (DMDR) after dimension reduction
Semi-Supervised Classifier- LapSVM
LapSVM + SSMSS
Decision function
$f^* = \arg\min_{f \in H_k} \frac{1}{l}\sum_{i=1}^{l} V(f(x_i), y_i) + \gamma_A \|f\|_K^2 + \gamma_I \|f\|_I^2$
Cost function of the error
Manifold regularization
Trained Model
Rxn_Gene KK_1 KK_2 y_i
Prediction

Fig 1. The proposed machine learning strategy. The integrated pipeline for prediction of essential genes based on limited labeled training dataset consisting of reaction-gene pairs with sequence, informatics, and topological network features.

https://doi.org/10.1371/journal.pone.0242943.g001

target organisms (Table 1) [62]. From the genome-scale reconstructed metabolic network, the information of metabolites, reactions, and genes was collated.

The sequence-based features and the topological features of the metabolic reaction network, and flux-coupled sub-network based were calculated and accumulated for each reaction-gene combination. These reaction-gene combinations integrate diverse features of the metabolic adaptation of the organism and give detailed insights into the role of a particular gene in the metabolic reaction network. This helps in the prediction of the essentiality of the gene in the target organism with high accuracy. A total of 289 features were computed for each reaction-gene pair. Brief descriptions of these features are given below, and their abbreviations are enlisted in S1 Table.

To establish the model consistency and reproducibility of the proposed pipeline, two different types of data sets for each of the twelve organisms have been used. The first type of data set consists of 80% data points of total data set with limited labeled data that is used for training while the remaining 20% is used for blind testing to check the model validation. Using this 80% data points of the whole dataset, different types of training data set are further created with limited labeled data points in the range, i.e., i % Labeled (L) and (100—i%) Unlabeled (UL) data, where i = 1, 2, 3, 4, 5, 10, 30, 50, 70 and 90. In each category, labeled samples were chosen randomly from the master table. It is to be mentioned here that this selection of labeled data was conditionally randomized to ensure that both the essential and non-essential genes categories appear with equal probability. In this way, 100 data sets in each labeled category have been created.

The second type of data set consists of the whole dataset with limited labeled data, which is used for model training and prediction purposes for each of the twelve organisms. It is to be

Table 1. Organisms considered for model training and validation.

Organism Name	Abbreviation	Input files	
		FASTA files of coding nucleotide and protein sequence (RefSeq assembly accession)	Genome-Scale Reconstructed Metabolic Network
Organisms used for Model Development and Validation of the Proposed Pipeline			
Acinetobacter sp. ADP1	ACIAD	GCF_000046845.1_ASM4684v1	iAbaylyiv4 [49]
Bacillus subtilis subsp. subtilis str. 168	BACSU	GCF_000009045.1_ASM904v1	iYO844 [50]
Escherichia coli K-12 MG1655	ECOLI	GCF_000005845.2_ASM584v2	iJO1366 [51]
Helicobacter pylori	HELPY	GCF_000008525.1_ASM852v1	iIT341 [52]
Mycobacterium tuberculosis H37Rv	MYCTU	GCF_000195955.2_ASM19595v2	iNJ661 [53]
Pseudomonas aeruginosa PAO1	PSEAE	GCF_000006765.1_ASM676v1	iPae1146 [54]
Pseudomonas aeruginosa UCBPP-PA14	PSEAB	GCF_000014625.1_ASM1462v1	iPau1129 [54]
Salmonella enterica subsp. enterica serovar Typhimurium LT2	SALTY	GCF_000006945.2_ASM694v2	STM_v1_0 [55]
Staphylococcus aureus subsp. aureus NCTC 8325	STAAB	GCF_000013425.1_ASM1342v1	BMID000000141098 [56]
Saccharomyces cerevisiae	YEAST	GCF_000146045.2_R64	iMM904 [57]
Caenorhabditis elegans	CELEG	GCF_000002985.6_WBcel235	iCEL1273 [58]
Mus musculus	MUSMU	GCF_000001635.26_GRCm38.p6	iMM1415 [59]
Organisms used for Case Study			
Leishmania donovani	LDONO	TriTrypDB-36	iMS604 [60]
Leishmania major	LMAFR	TriTrypDB-36	iAC560 [61]

https://doi.org/10.1371/journal.pone.0242943.t001

mentioned here that, in less-studied organisms where gene essentiality information is very less, a blind test cannot be applied. For those cases, the whole data set with limited labeled data will be used for model training and prediction purposes.

Topological analysis of reaction and flux-coupled sub-network. The metabolic network of each target organism was transformed into an undirected reaction network (RN), in which each node denotes an enzyme (reaction), and each edge represents the connection between two reactions that have common metabolites. The commonly used topological network features, such as centrality measures, that highlight the biological significance of an enzyme in a network were computed [63]. Generally, a central and highly connected enzyme in biological networks is often essential as it represents an important hub within the network [64]. If this hub node is blocked, then the whole pathway might be disrupted.

Similarly, Flux coupling analysis (FCA) is an optimization procedure based on flux, which represents whether the reaction subsets are coupled or not in certain given specific environmental exchange constraints [65,66]. Flux-coupled subgraph was used to extract biologically relevant topological features dependent on physiological flux relationships.

Eight centrality measures have been computed for both the reaction as well as the flux coupled networks, *viz*., Degree Centrality, Eigenvalue Centrality, Eccentricity, Hub score, Authority Scores, Page Rank, Betweenness Centrality, and Number of triangles. A detailed description of all these centrality measures has been discussed in different literature [67–69]. These topological features have been calculated using the "igraph" package in R [70].

Features derived from the coding nucleotide sequence. Three types of features (*viz*. nucleotide content, codon usage bias, and information-theoretic features) of the metabolic genes have been extracted from the nucleotide sequence of the organisms that contribute towards gene essentiality. A brief description of the features has been discussed below.

Nucleotide content. Previous studies have elucidated that in bacterial genomes, GC content is correlated with the environmental condition in which the bacterium survives [71]. Hence, the related GC content of the genome of a target organism can be an essential feature for gene essentiality prediction. Another study showed that there is a significant difference in the distribution of the frequency of occurrence of A, T, G, and C nucleotides at the 3rd synonymous position of codons between the essential and non-essential genes [40]. These features were computed using an in-house code.

Codon usage bias. Protein abundance in an organism can be predicted by using Codon usage [72–74]. Highly expressing abundant proteins in metabolism might have functional importance and can be essential. Codon usage bias features, like Effective Number of Codons (ENC) [75] and Codon Adaptation Index (CAI) [73], were calculated using EMBOSS package version 6.6.0–1 [76].

Mutual Information (MI) and Conditional Mutual Information (CMI). A previous study has used information-theoretic features such as mutual information (MI) and conditional mutual information (CMI), for essential gene prediction [37]. MI and CMI profile of coding nucleotide sequence can be used as genomic signatures which represent the phylogenetic relationship between genomic sequences [77]. A total of 80 features (16 MI and 64 CMI) have been computed by using in house Perl script.

Features derived from protein sequence. In order to investigate the dependence of gene essentiality on protein sequences, various derived and informatic features such as the frequencies of the amino acids, protein length, paralogy score, average Kidera factor, etc. have been considered in this study.

Frequencies of the twenty amino acids and protein length. Each protein sequence related to the reaction-gene combination was used to calculate the occurrences of the 20 amino acids that reflect the physicochemical properties of these proteins related to each of the

reaction-gene combinations under consideration. These twenty features were calculated using EMBOSS package version 6.6.0–1 [76] and named according to their corresponding 20 amino acids.

Paralogy based features (paralogy score). The sequence similarity of a gene in its intra-genome is called a paralogous gene of an organism. Paralogous genes have the same or similar types of biological functions. An organism may not be affected by the deletion of one of the paralogous genes because another paralogous gene may compensate for a similar type of function. So there are fewer chances for paralogous genes to be essential [78].

The paralogy score of a gene was calculated by performing a BLAST [version 2.2.26] search against the whole set of protein sequences of a target organism with different E-value threshold ranging from 10^{-3} to 10^{-30} with at least 40% identity. Features based on paralogy score were labeled as P3 (E-value cut off 10^{-3}), P5 (E-value cut off 10^{-5}), P7 (E-value cut off 10^{-7}), P10 (E-value cut off 10^{-10}), P20 (E-value cut off 10^{-20}), P30 (E-value cut off 10^{-30}). These features have been calculated using in house Perl script.

Fourier sine and cosine coefficient. The Fourier sine and cosine coefficient of protein sequences [79] have been used to see if there are any inherent patterns which will help to classify between essential and non-essential genes. The Fourier coefficient (FC) is the converted numerical values of protein sequences, which describes the physical properties of corresponding amino acids. These physical properties represent the ten property factors using factor analysis introduced by Kidera et al. [80]. Mathematical representations of these coefficients are given below:

$$FC\sin WN_k\_KF_n = a_k^{[n]} = \sum_{l=0}^{N-1} f_l^{[n]} \sin\left(\frac{2\pi kl}{N}\right) \quad \text{(Eq 1)}$$

$$FC\cos WN_k\_KF_n = b_k^{[n]} = \sum_{l=0}^{N-1} f_l^{[n]} \cos\left(\frac{2\pi kl}{N}\right) \quad \text{(Eq 2)}$$

Where the length of the protein sequence is N, $f_l^{[n]}$ is n^{th} property factor of amino acid l, and wavenumber is k (Eqs 1 and 2).

Fourier sine and cosine coefficient in a specific range of Wave Number (WN) and Kidera Factor (KF) was calculated. The range of WN and KF are $0 \leq k \leq 7$ and $1 \leq n \leq 10$. It is also reported that global folding information of the protein is encoded in a specific range of wavenumber $0 \leq k \leq 7$ [79]. A total of 150 features were computed. These features have been calculated using in house Perl script.

Average Kidera factor. The ten Kidera Factors (viz. KF1: Helix/bend preference, KF2: Side-chain size, KF3: Extended structure preference, KF4: Hydrophobicity, KF5: Double-bend preference, KF6: Partial specific volume, KF7: Flat extended preference, KF8: Occurrence in the alpha region, KF9: pK-C, KF10: Surrounding hydrophobicity) were derived by multivariate analysis on 20 amino acids using 188 physical properties and dimension reduction techniques [80]. The protein sequence of the corresponding reaction-gene combination was used to calculate ten features (AKF_i where, i = 1 to 10) by averaging the ten Kidera factors. These features have been calculated using in house Perl script.

2.2 Feature selection based on the space-filling concept

The contribution of these 289 features towards gene essentiality is unknown; hence, there may be a possibility to select redundant features by the feature selection algorithm. These redundant features may affect the training performance of the machine learning model. Hence, it is

important as well as challenging to choose the non-redundant, unique feature subset for training the model. Feature selection helps to capture the most relevant biological features and helps the classifier to learn a better way to predict essential and non-essential genes with high accuracy. Here the unsupervised feature selection method based on the space-filling concept has been used [81]. This unsupervised method selects the features based on a coverage measure that estimates the spatial distribution of the data points in a hypercube and ensures uniform distribution of points in a regular grid in the data space. The method captures the variability of features with new and relevant information about the data. This method has been tested on various datasets and different scenarios with noise injection and data shuffling. The benefits of using this algorithm are two folds. Firstly, being an unsupervised algorithm, prior information of the output variable is not required.

Additionally, here no classifier is required for feature selection. Hence time complexity is less in comparison to other feature selection algorithms, like SVM-RFE. Also, it has been observed that this method gives better information of relevant features than other unsupervised correlation-based feature selection techniques that, although it can remove the redundant features, cannot eliminate the features with low variability that are non-relevant and non-informative for classification [82,83].

2.3 Dimension reduction using forced directed graph layout

After feature selection, the data set was transformed into a lower dimension (2-D) using a dimension reduction technique for visualization. Projected 2-D features set to reserve all the information the same as higher-dimensional data. This is an important step in the pipeline as the classifier works better in 2-D than with the higher dimension data. For dimension reduction, a force-directed graph layout algorithm Kamada-Kawai has been used that considers each data point as a node in a graph having attractive and repulsive forces between them that can be modeled as springs connecting the nodes [84]. The algorithm then tries to cluster the data points by minimizing the total energy of the system based on attracting and repelling forces between them. Here the input of the Kamada-Kawai algorithm is a graph constructed by using the K-Nearest Neighbour (K-NN) algorithm. For known organisms, it has been observed that essential genes are clustered together in one side of an arc in a circle layout, and non-essential genes are clustered in the rest of the circle. A circular layout of each organism has been observed from the Kamada Kawai algorithm with a specific parameter (K Nearest Neighbor) value of the K-NN algorithm. Here it is assumed that if a similar circular layout is observed for less explored organisms related to gene essentiality, the unlabeled genes will be clustered together category wise and reside on the arc of the circle. This analysis had been performed using the "dimRed" package in R [85].

Both the feature selection and the dimensionality reduction methods are used for not only reducing the number of features in a dataset but also to select the important features, which are contributing significantly. Feature selection is used for selecting the relevant features without changing the original values, whereas, the dimensionality reduction step transforms the higher dimensional features into a lower dimension. From the dimension reduction technique it is very difficult to identify the key features which are contributing for classifications, hence the feature selection step is necessary.

To test the efficiency of this dimension reduction technique combined with unsupervised feature selection and LapSVM classifier, the performance metrics of Kamada-Kawai has been compared against other dimension reduction techniques, such as Principal Component Analysis (PCA) [86], Metric Dimensional Scaling (MDS) [87], Fruchterman Reingold [88] and FastICA [89] using the gold standard dataset of twelve organisms. To test the statistical

significance of the results, the one-tailed Mann Whitney U Test has been performed with 1% level of significance ($P<0.01$).

2.4 Semi-supervised classifier: Laplacian SVM

Essential gene classification using the machine learning technique can be a difficult task when a minimal amount of gene essentiality information for the target organism is available. In this setting, semi-supervised learning is an appropriate approach that builds a trained model from labeled and unlabeled samples [90]. Most of these semi-supervised algorithms follow two common assumptions, i.e., cluster assumption and manifold assumption. Cluster assumption states that data points in the same cluster have a chance of having the same class label. Manifold assumption means that close data points along the manifold area follow similar data structures or similar class labels. However, cluster assumption follows the global feature, and manifold assumption follows the local features in the model.

Laplacian support vector machine (LapSVM) is a graph-based semi-supervised learning method, which is based on a manifold regularization framework [44]. The graph is constructed from labeled and unlabeled data as the node. The similarity between data points in a graph can be assigned by edge weight, which is calculated from the K-NN algorithm. In this way, the information of labeled data points can be passed to another node, and then, the unlabeled nodes can be labeled. The input data set being circular (non-linear), Radial Basis Function (RBF) kernel with the classifier LapSVM have been used. This analysis had been performed using the "RSSL" package in R [91].

2.5 The score for best model selection

There are various performance metrics, e.g., True Positive Rate (TPR), False Positive Rate (FPR), precision, recall, F-measure, Matthews correlation coefficient (MCC), Area under the receiver operating characteristic curve (auROC), etc. to evaluate the trained model in supervised machine learning technique. These measures are statistically significant if sufficient labeled data are available. However, due to limited labeled data, these metrics will not work for best model selection in a semi-supervised type algorithm. To circumvent the above problem, a new measure has been proposed, called the Semi-Supervised Model Selection Score (SSMSS), for selecting the best model. This SSMSS score is dependent on four different measurements (Eq 3). For this, the training data set, having limited labeled reference, has been labeled as ground truth (GT) reference. Another reference set called the pseudo reference (PR) has been considered by calculating the distance from unlabeled data points to the labeled dataset. The dataset containing the predicted labels by the Laplacian SVM classifier has been labeled as the Laplacian Reference (LR). Thereafter, Silhouette Index (SI) [92] was computed to check the clustering grouping quality. The $\text{CorrectPrediction}_{\text{GT\_LR}}$ measure was calculated based on the matches between the predictions of the Laplacian SVM classifier with the Ground Truth data. Here, the calculation of the MCC with the help of Pseudo-reference and Laplacian Reference was represented as $\text{MCC}_{\text{PR\_LR}}$. Silhouette Index calculation based on Pseudo Reference and Laplacian Reference was denoted by SI_{PR} and SI_{LR} respectively. Based on these parameters, the values of the proposed Semi-Supervised Model Selection Score (SSMSS) may vary from 0 to 1. If any of the above four measurements is low, then the SSMSS value will be drastically decreased. The best model will be selected from 64 models which has the highest SSMSS value for each data set in different parameters combinations, i.e., kernel parameter [Radial Basis Function (RBF) kernel parameter sigma (σ)] and LapSVM parameters [lambda (λ): L_2 regularization parameter and gamma (γ): the weight of the unlabeled data]. It may be mentioned here that the score will not consider those models which have negative Silhouette Index and MCC

value. The parameters (σ,λ,γ) have been varied with four different values, i.e., 0.01,0.1,1,10. Therefore, by tuning these model parameters using grid search, 64 models for each data set have been generated. The following equation has been proposed for the calculation of the SSMSS.

$$\mathrm{SSMSS}_{\mathrm{k=1\ to\ 64}} = \min\{\mathrm{CorrectPrediction}^{\mathrm{k}}_{\mathrm{GT\_LR}}, \mathrm{MCC}^{\mathrm{k}}_{\mathrm{PR\_LR}}, \mathrm{SI}^{\mathrm{k}}_{\mathrm{PR}}, \mathrm{SI}^{\mathrm{k}}_{\mathrm{LR}}\} \quad \text{(Eq 3)}$$

$$\forall \mathrm{MCC}^{\mathrm{k}}_{\mathrm{PR\_LR}} \geq 0, \mathrm{SI}^{\mathrm{k}}_{\mathrm{PR}} \geq 0, \mathrm{SI}^{\mathrm{k}}_{\mathrm{LR}} \geq 0.$$

$$\mathrm{SSMSS}_{\mathrm{best}} = \max\{\mathrm{SSMSS}_{\mathrm{k=1}}, \mathrm{SSMSS}_{\mathrm{k=2}}, \ldots\ldots\ldots, \mathrm{SSMSS}_{\mathrm{k=64}}\},$$

where k is the k^{th} model with a particular parametric combination and $\mathrm{SSMSS}_{\mathrm{best}}$ is the best score of the best model among these 64 models.

2.6 Time complexity of the proposed strategy

The proposed pipeline has three components (i.e., Unsupervised Feature Selection, Kamada Kawai Dimension Reduction Technique, and LapSVM semi-supervised classifiers), which work sequentially. To calculate the total time complexity T(n,d) of the proposed strategy, the cumulative effect of all three components have been considered, where n denotes the number of data points (reaction-gene pair) that depends on the size of the metabolic network of the organism, and d is the total number of features.

The time required for each of the three components can be represented as follows [44,81,84]:

Time required for Unsupervised Feature Selection algorithm = $\frac{d(d+1)n^2}{2}$ Time required for Kamada Kawai algorithm = n^3

Time Required for LapSVM = n^3

Therefore, the total time required T(n,d) can be represented as:

$$\begin{aligned} \therefore T(n,d) &= \frac{d(d+1)n^2}{2} + n^3 + n^3 \\ or, \quad T(n,d) &\leq 4n^3 + n^2(d^2+d) \\ or, \quad T(n,d) &\leq (4+d+d^2)n^3 \\ or, \quad T(n,d) &\leq Cd^2n^3 \\ or, \quad T(n,d) &= \mathrm{O}(d^2n^3) \end{aligned}$$

Where, C is a constant, in particular, $C \geq 6\ \forall d, n \in \mathbb{N}$.

Therefore, the total time complexity of the proposed strategy is $O(\mathrm{d}^2\mathrm{n}^3)$.

2.7 Gene essentiality prediction, experimental validation, and pathway enrichment

The essential gene prediction results for the twelve model organisms have been compared with experimental data obtained from the OGEE database, and the corresponding supervised performance metrics such as TPR, FPR, MCC, auROC, etc. were calculated. Further, the predicted essentiality information of the reaction-gene pairs of all twelve organisms has been categorized into five different groups based on their involvement in different reactions. These five groups are following: **CEN** (Combination of Essential and Non-essential), involving both essential and non-essential genes controlling a reaction; **ME** (Multiple Essential), multiple essential genes involved in a reaction; **MN** (Multiple Non-essential), multiple non-essential genes

governed a reaction; **SE** (Single Essential), single essential genes involved in a reaction; **SN** (Single Non-essential), single non-essential genes involved in a reaction. Thereafter, the distributions of the five categories of reaction-gene pairs from the predicted results have been compared with the distribution observed in experimental data for all the organisms using the Chi-Square Test (1% level of significance).

For *Leishmania donovani* and *Leishmania major*, the best model was selected based on the SSMSS score for the prediction of the essential reaction-gene combinations. These predicted reaction gene combinations were then classified into the five categories, like the other twelve species. The list of unique genes that were extracted from this predicted essential reaction-gene pairs was analyzed for their associated Gene Ontology (GO) terms [93,94] from the Uniprot database [95]. The percentages of genes associated with each GO term were calculated for both the organisms. Additionally, using the DAVID pathways enrichment tool [96], the essential genes were further analyzed to identify the significantly enriched KEGG pathways [97] that were associated with these essential genes.

Source codes of the entire machine learning strategy and pipeline are given in S1 Text, which consists, Training data set preparation and integration of heterogeneous features, Feature selection based on the space-filling concept, Dimension reduction using forced directed graph layout, and Semi-supervised classifier: LapSVM.

3. Results

3.1. Model validation with experimental data

The integrative proposed strategy (Fig 1) was applied and validated on twelve organisms (Table 1) with well-annotated genes essentiality information from experimental data obtained from the OGEE database [48].

3.2. Features frequently selected by the feature selection algorithm

The important features chosen by the feature selection algorithm have been represented in the heat map (See methods section for a detailed description of features and S1 Fig), where X-axis represents the name of the 82 features that have been selected at least once by the features selection algorithm and Y-axis corresponds to names of the organism. Red cell color indicates features selected by the feature selection algorithm in the corresponding organism. White-colored cell shows the feature that is not selected or is redundant. Among 289 features, three features, *viz*., Reaction Network betweenness centrality (RN_betweenness), Reaction Network Page Rank centrality (RN_page_rank), and Flux Coupled Analysis Network Page Rank centrality (FCA_page_rank) are selected by the features selection algorithm for every organism. These frequently selected features are topological network features. Apart from these features, Information-theoretic features (Fourier sine or cosine coefficient, Mutual Information, Conditional Mutual Information) from nucleotide and peptide sequences are also selected. If a node is important in the reaction network and flux-coupled network, then there is a chance that the enzyme or protein which controls that particular reaction and its corresponding coding sequence is also essential.

3.3. Dimension reduction

After applying feature selection, the Kamada-Kawai dimension reduction technique [84] is used for visualization purposes. Here, a circular layout of each organism is observed. While the essential gene-reaction combinations are clustered together in one side of the arc in a 2-D circular layout, the non-essential reaction-gene combinations are clustered in the rest of the

circle. Now on applying Laplacian SVM, the classifier was able to easily classify gene essentiality based on their transformed 2-D feature and the limited label information. Now in different parameter combinations of Laplacian SVM, different trained models are obtained. To select the best model among trained models, the proposed SSMSS score has been used.

3.4. Robustness of the proposed score (SSMSS)

To check the robustness of the SSMSS score, the proposed strategy has been applied on both types of training data set (i.e., data set with 80–20% combination of samples and with the whole data set) for these twelve organisms. Using this 80% data points of the whole dataset, different types of training data set is further created with limited labeled data points in the range, i.e., i % Labeled (L) and (100—i%) Unlabeled (UL) data, where i = 1, 2, 3, 4, 5, 10, 30, 50, 70 and 90. In each category, labeled samples were chosen randomly from the master table. It is to be mentioned here that this selection of labeled data was conditionally randomized to ensure that both the essential and non-essential genes categories appear with equal probability. In this way, 100 data sets in each labeled category have been created. For the testing purpose, both the whole training data set and the 20% blind data set have been used for prediction. The parameters (σ,λ,γ) were tuned with four different values i.e. 0.01,0.1,1,10. Therefore, by tuning these model parameters using grid search generated 64 models for each data set have been created. After that, the prediction results were compared with the known gene essentiality information, which is publicly available from the experiment. Six supervised performance metrics have been calculated for the predicted class label with the known class label. After that, the association between the proposed score and auROC was assessed. To verify the linear relationship between auROC and the proposed score (SSMSS), the Pearson correlation coefficient has been calculated, and scatter plots were generated in different limited labeled data sets in each target organism (S2 Fig).

From the scatter plot (S2 Fig), it has been observed that in all the cases, Pearson correlation $>$0.75. Hence, it may be inferred that due to the linear relationship existing between auROC and the proposed score (SSMSS), the applicability of this scoring technique is asserted and can be used for the calculation of the performance measurement matrix and best model selection for the semi-supervised based classifier.

3.5. Predictive performance of the best models in the different labeled category on training and blind test data set

In a real-life scenario, only limited gene essentiality information is available for the less explored organisms. However, model building from this limited label data and determining how the highest score will select the best model is difficult. Hence, to test the model performance on known organisms by creating limited labeled datasets (i.e., by varying the limited labeled data from 1% to 90% from the 80% training dataset), six supervised performance metrics have been calculated for each category under different parameter combinations of σ,λ,γ (See Section 2.5: The score for best model selection) for a detailed description of these parameters). Here, within each labeled category, the average behavior of the predictive performance (six supervised performance metrics) and the Score (SSMSS) of the best 100 trained models are plotted in Fig 2. This has been shown for two different conditions, training data set (80% of the whole data) and blind testing data set (20% of the whole data). As observed from the low standard deviations for each metrics (under each category), it is worth to mention that the accuracy for the training and testing are very similar in most of the cases.

From these plots, it has been observed that the model selection based on the SSMSS score in each category corresponds to a high auROC value of greater than 0.8 in all cases across all

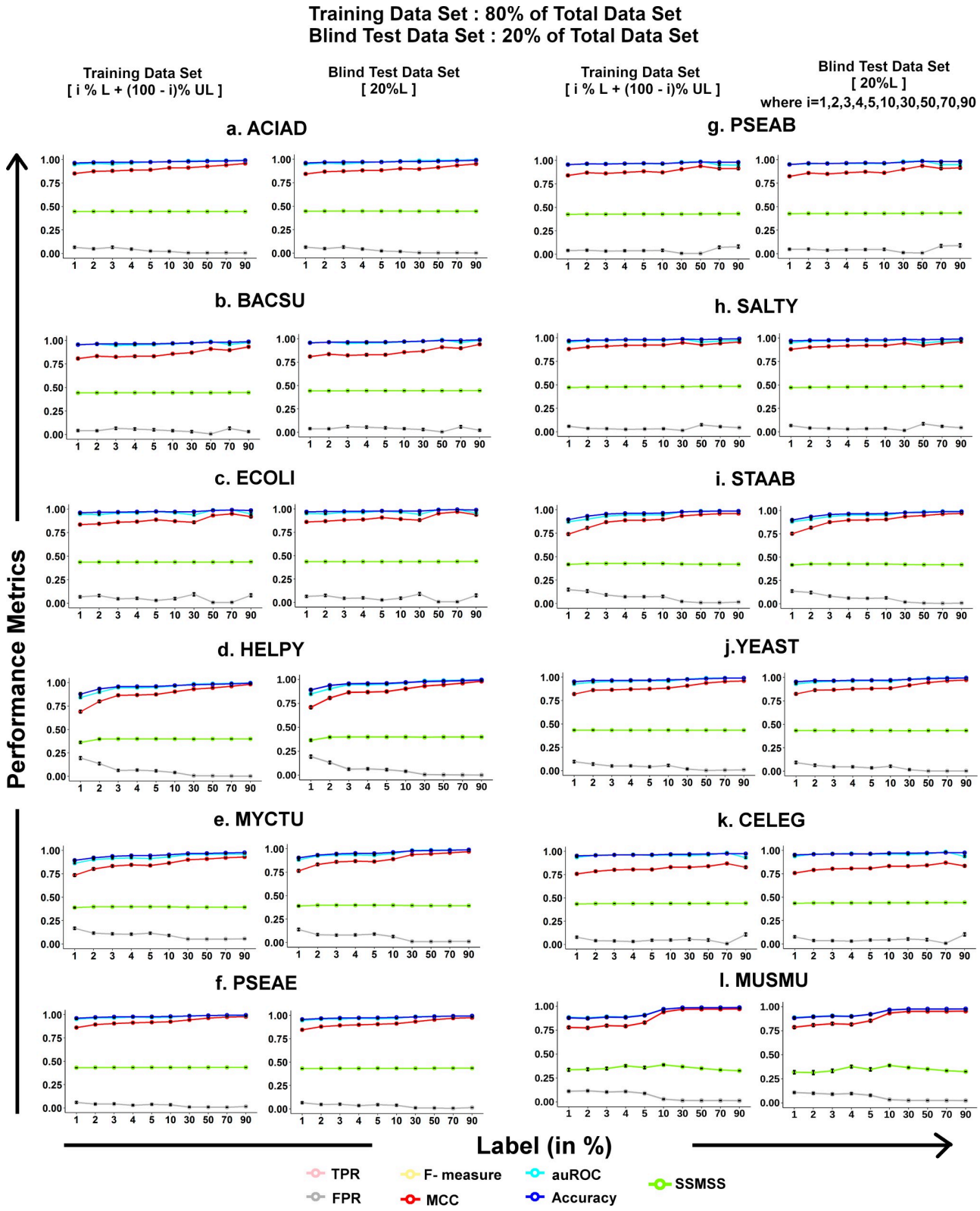


Fig 2. Comparison of the predictive performance of the best models in the different labeled category. The average performance of the best 100 models at training and blind testing for six supervised metrics (i.e., TPR, FPR, F-measure, MCC, auROC, accuracy) and SSMSS for each labeled type. The X-axis represents the category of labeled data, the Y-axis represents the value of performance metrics.

https://doi.org/10.1371/journal.pone.0242943.g002

organisms. Also, it is observed that if the label increases, then model performance will also show higher accuracy. However, it is seen that the auROC score remains consistently high, using 1% labeled data or more, which establishes the fact that the proposed method can predict using a minimum of 1% labeled data. It has also been observed that this method is giving a consistent better predictive performance on both Prokaryotic and Eukaryotic organisms for both the data sets (80% training and 20% blind testing) and follow similar patterns for six supervised performance metrics in differently labeled categories. As the predictive performance of 20%, the blind data set is similar to training performance, so further, it can be concluded that model overfitting and underfitting is not arising in this case.

To compare the predictive performance of the proposed method, 1% labeled data set has been considered for each of the twelve organisms. For training, different supervised classifiers have been used, such as Random Forest [98], Naive Bayes [99], Logistic regression [100], J48 (C.45) Decision Tree [101] as well as our own previously reported Supervised essential gene prediction pipeline [40] on the whole dataset for testing (S3 Fig). In all of the cases, it is found that the proposed method performed better than all other methods using only 1% labeled data of the whole training dataset.

3.6. Effect of feature selection and dimension reduction in model performance

To compare the effect of feature selection and dimension reduction steps along with the LapSVM classifier, seven different types of classification scenarios, based on different dimension reduction technique such as PCA, MDS, FR, ICA, and KK, were simulated on training data set (80% data points) and blind testing (20% data points) data sets of twelve organisms. The corresponding performance was calculated on the blind test data set (Fig 3). Each training data set has only 1% labeled data, and the rest of them Unlabeled.

The seven scenarios were created with LapSVM classifier and combinations of features selection and dimension reduction techniques:

Scenario 1 (S1): Without feature selection and Without dimension reduction technique [WOFS +WODR]

Scenario 2 (S2): Without feature selection and With dimension reduction technique (Principal Component Analysis) [WOFS + DR (PCA)]

Scenario 3 (S3): Without feature selection and With dimension reduction technique (Metric Dimensional Scaling) [WOFS + DR (MDS)]

Scenario 4 (S4): Without feature selection and With dimension reduction technique (Fruchterman Reingold) [WOFS + DR (FR)]

Scenario 5 (S5): Without feature selection and With dimension reduction technique (Independent Component Analysis) [WOFS + DR (ICA)]

Scenario 6 (S6): Without feature selection and With dimension reduction technique (Kamada Kawai) [WOFS + DR (KK)]

Scenario 7 (S7): With feature selection (Unsupervised Feature Selection) and With dimension reduction technique (Kamada Kawai) [WFS (UFS) + DR (KK)]

From this analysis, it has been observed that for scenarios 1 to 5, the auROC value is very low, which signifies that dimension reduction techniques, e.g., PCA, MDS, FR, ICA, cannot significantly improve the gene essentiality prediction (Fig 3). On the other hand, for scenarios

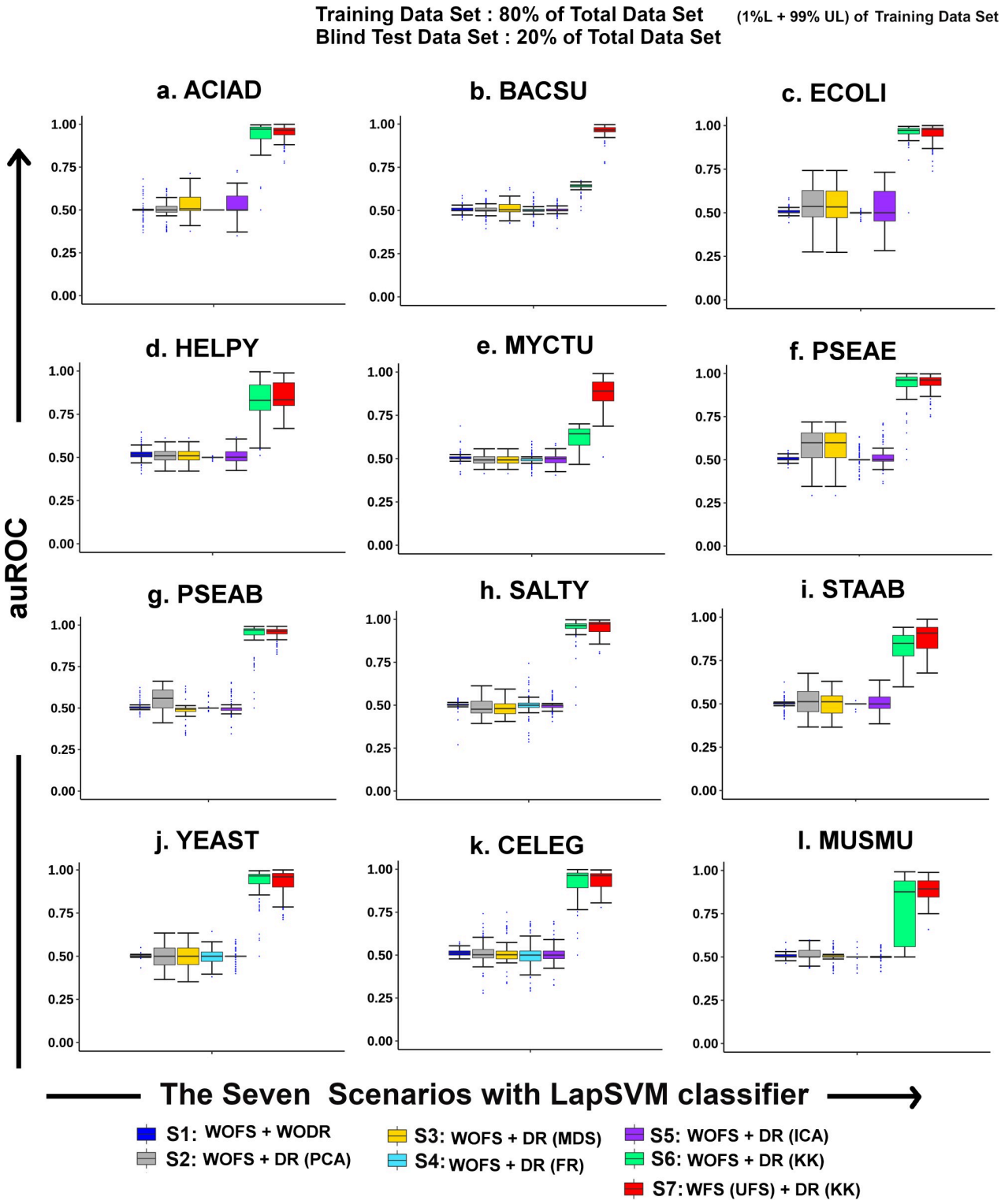


Fig 3. Effect of feature selection and dimension reduction on model performance. Comparison of the effect of different dimension reduction techniques PCA, MDS, FR, ICA, and KK (S2—S6) with S1 (Without Feature Selection and Without Dimension Reduction) and S7 (With Feature Selection and With Dimension Reduction-KK) when combined with LapSVM classifier. Plot represents the auROC value of 100 best models with 1% labeled data across all organisms.

https://doi.org/10.1371/journal.pone.0242943.g003

6 and 7, it is observed that on applying the Kamada-Kawai method of dimension reduction along with unsupervised feature selection, the model performance (auROC) improves drastically in each target organism. On comparing the efficacy of Kamada-Kawai (KK) with the other dimension reduction methods using the one-tailed Mann-Whitney U Test, a significant improvement in auROC values ($P<0.01$) for all the twelve organisms was observed (S2 Table). Scenario 6 highlights the importance of this dimension reduction step, where it is found that even without feature selection, the dimension reduction step [S6: WOFS + DR (KK)] has a huge impact on the results ($P<0.01$) [S3 Table]. However, the feature selection step helped us in identifying the minimal set of features that contribute towards gene essentiality prediction with greater accuracy in all organisms (lower *P*-values obtained in Scenario 7 with [S7: WFS + DR (KK)]) (S3 Table). Hence, it is observed that the Kamada-Kawai dimension reduction technique, when combined with LapSVM, gives significantly better performance for all twelve organisms even when only 1% labeled data is used (Fig 3).

3.7. Predictive performance using whole training data set

In model organisms where gene essentiality information is sufficiently available at the genome-scale, blind testing can be applied. However, in less explored organisms where gene essentiality information is very less, a blind test cannot be applied as the reference size is very small. For these cases, the whole data set with limited labeled data can be used for model training and prediction purposes.

To establish the predictive performance of the proposed strategy on the whole training data set, 1% labeled data were selected randomly, and the remaining 99% data points were considered unlabeled for the twelve organisms, where the information of gene essentiality in genome-scale was available from the experiments. Now, this whole data set was trained by the proposed strategy. The best model was selected based on the highest score (SSMSS). The same data set is used for prediction from the best-trained model. The outcome of the proposed strategy can be visualized as three circles (Fig 4). The first circle represents the circular projection of the whole data set in 2-D after applying the Kamada Kawai dimension reduction technique with gene essentiality information from the experiment. The second circle shows the training data set with 1% labeled & 99% Unlabeled data and learning curve of the Laplacian model. The third circle shows the predicted gene essentiality label from the best-trained model. From Fig 4, it is observed that the proposed model also performed well (as similar circular patterns from experiment and predicted) on the whole training data set.

The predictive performance on both the data sets (80% and the Whole data set) has been compared by six supervised performance metrics (i.e., TPR, FPR, F-measure, MCC, auROC, and accuracy) based on actual and predicted labels from the proposed strategy. Here it has been observed that the average predictive performance of the 100 trained model with 80% data set is similar to the performance on the whole data set (S4 Fig).

3.8. Categorization of reaction-gene pairs

Categorization of the predicted essentiality information of reaction gene pairs into the five categories, viz. CEN, ME, MN, SE, and SN show that the distribution of reaction of the predicted results matches exactly with the distribution observed with the experimental data for each of the twelve organisms (Fig 5). Also, the Chi-square test was performed with a Null Hypothesis (H_0) that the two distributions of reaction (experimental vs. predicted) are similar for all twelve organisms. Here, it has been observed that the *P*-values of the Chi-square test (*P*-values are indicated in S4 Table) are greater than 0.01 in all the 12 organisms. As *P*-values are large, it can be concluded that the experimental distributions of reaction are not significantly different

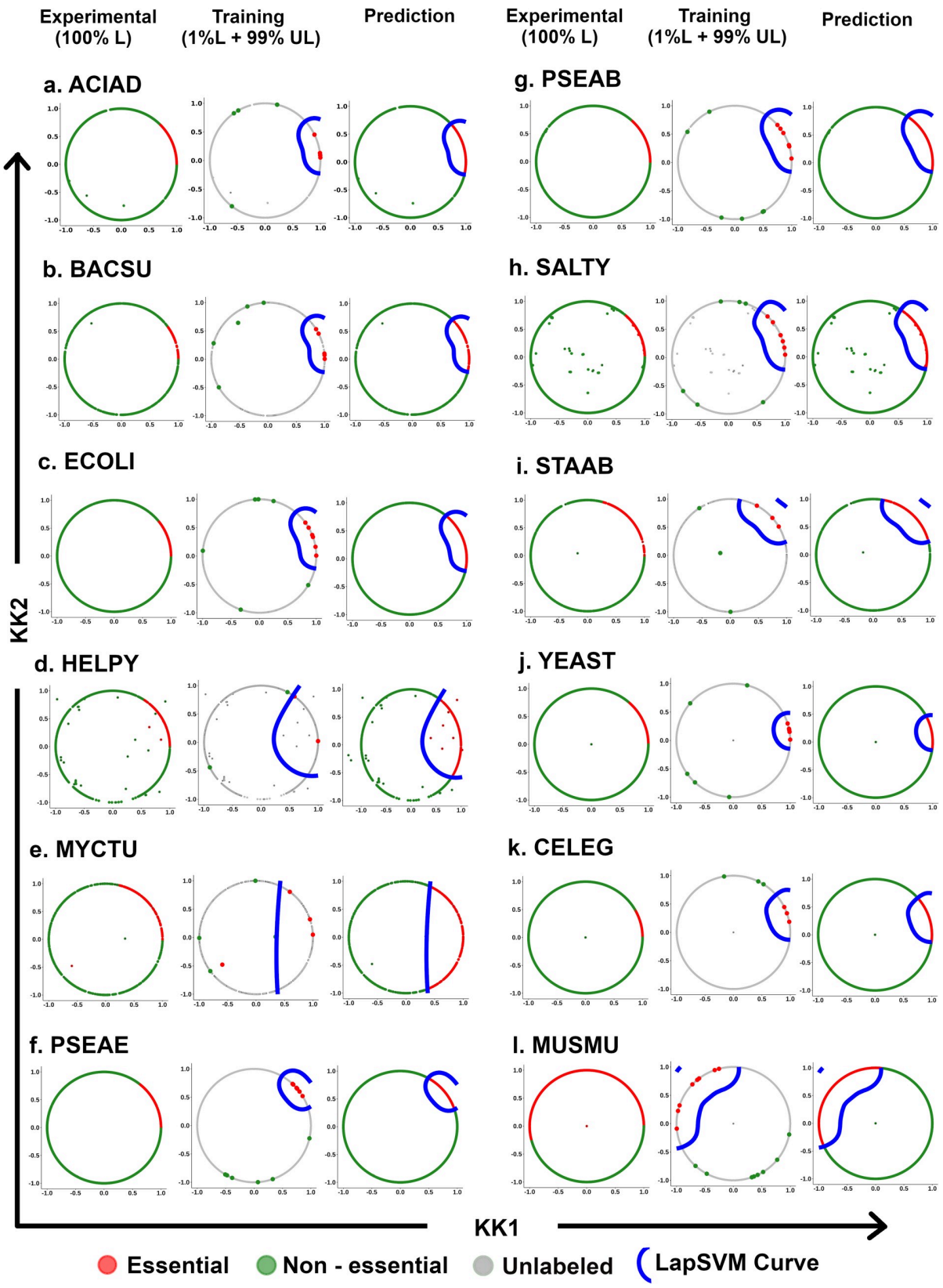

Experimental (100% L)
Training (1%L + 99% UL)
Prediction
Experimental (100% L)
Training (1%L + 99% UL)
Prediction
a. ACIAD
b. BACSU
c. ECOLI
d. HELPY
e. MYCTU
f. PSEAE
g. PSEAB
h. SALTY
i. STAAB
j. YEAST
k. CELEG
l. MUSMU
KK2
KK1
Essential
Non - essential
Unlabeled
LapSVM Curve

Fig 4. Visualization of the outcome of the proposed strategy. Essential, non-essential, and Unlabeled reaction gene pairs are colored accordingly Red, Green, and Gray. The learning curve for the best-trained model by LapSVM is colored with blue. The left circle represents the original data set with labeled data points. The middle circle shows the training data set with the learning curve, and the Right circle represents the prediction labeled with the learning curve.

https://doi.org/10.1371/journal.pone.0242943.g004

from the predicted distributions. This pattern has been fairly consistent over all the organisms, where it is found that the highest fraction of reactions is regulated by single non-essential (red) or multiple non-essential genes (blue). On the other hand, fractions of reaction governed by a single essential gene are low due to a small number of minimally essential genes in all organisms. From this plot (Fig 5), it is also observed that the fractions of reactions governed by multiple essential genes are extremely low in each of the twelve organisms. These comprise the small set of reactions that are absolutely crucial for the survival of the organisms.

3.9. Case Study: *Leishmania donovani* and *Leishmania major*

The proposed strategy has been implemented for less explored organisms like *Leishmania donovani* (11 genes have genes essentiality information [102]) and *Leishmania major* (10 genes have genes essentiality information [102]) using the semi-supervised machine learning strategy. Here it is observed that the network centrality features and information-theoretic features, such as the Fourier cosine coefficient derived from the Kidera factor, have been selected by the feature selection algorithm in both the cases of *L. donovani* and *L. major*. Additionally, certain unique features were also selected for each of the two organisms (S1 Fig). When the Kamada-Kawai dimension reduction technique was applied on *Leishmania* data sets, a similar circular pattern was observed, like the other twelve organisms that helped the classifier in predicting gene essentiality (Fig 6A).

For the essential gene prediction, in the case of *Leishmania donovani*, 80 reaction-gene pairs were predicted as essential among 1129 reaction-gene pairs. For *Leishmania major*, 335 reaction-gene pairs were predicted as essential among 1188 reaction-gene pairs. The categorization of these reaction-gene pairs displayed a pattern similar to the distributions of reaction observed in the twelve model organisms (Fig 6B). Predicted gene essentiality information from the proposed pipeline is listed in (S5 and S6 Tables). The list of essential genes extracted from these reaction gene pairs consists of 44 essential genes of *L.donovani* and 194 of *L. major*.

These essential genes were associated with 53 and 219 Gene Ontology (Molecular Function) terms for *L. donovani* and *L. major*, respectively (S7 and S8 Tables). The Gene Ontology term that occurred most frequently with these essential genes were related to ATP binding in both the organisms. The pathway enrichment of these essential genes shows 11 significantly enriched KEGG pathways for *L. donovani* and 20 *L. major*. Although 8 KEGG pathways were found to be common among the two species, certain unique pathways specific to each species were also enriched for each of the two organisms (S9 and S10 Tables). Further experimental validation on these predicted results would confirm the role of these genes in these less-studied organisms.

4. Discussion

Essential gene prediction helps to unveil the complexities and survival strategies of many disease-causing organisms. The prediction of gene essentiality is a challenging task in machine learning due to the unavailability of sufficient experimentally labeled data and a proper metric for selection of the best model. Considering this limited gene essentiality information, the proposed pipeline has been able to predict gene essentiality at genome-scale using as small as a set of 1% labeled genes having gene essentiality information using both 80%-20% (training-blind

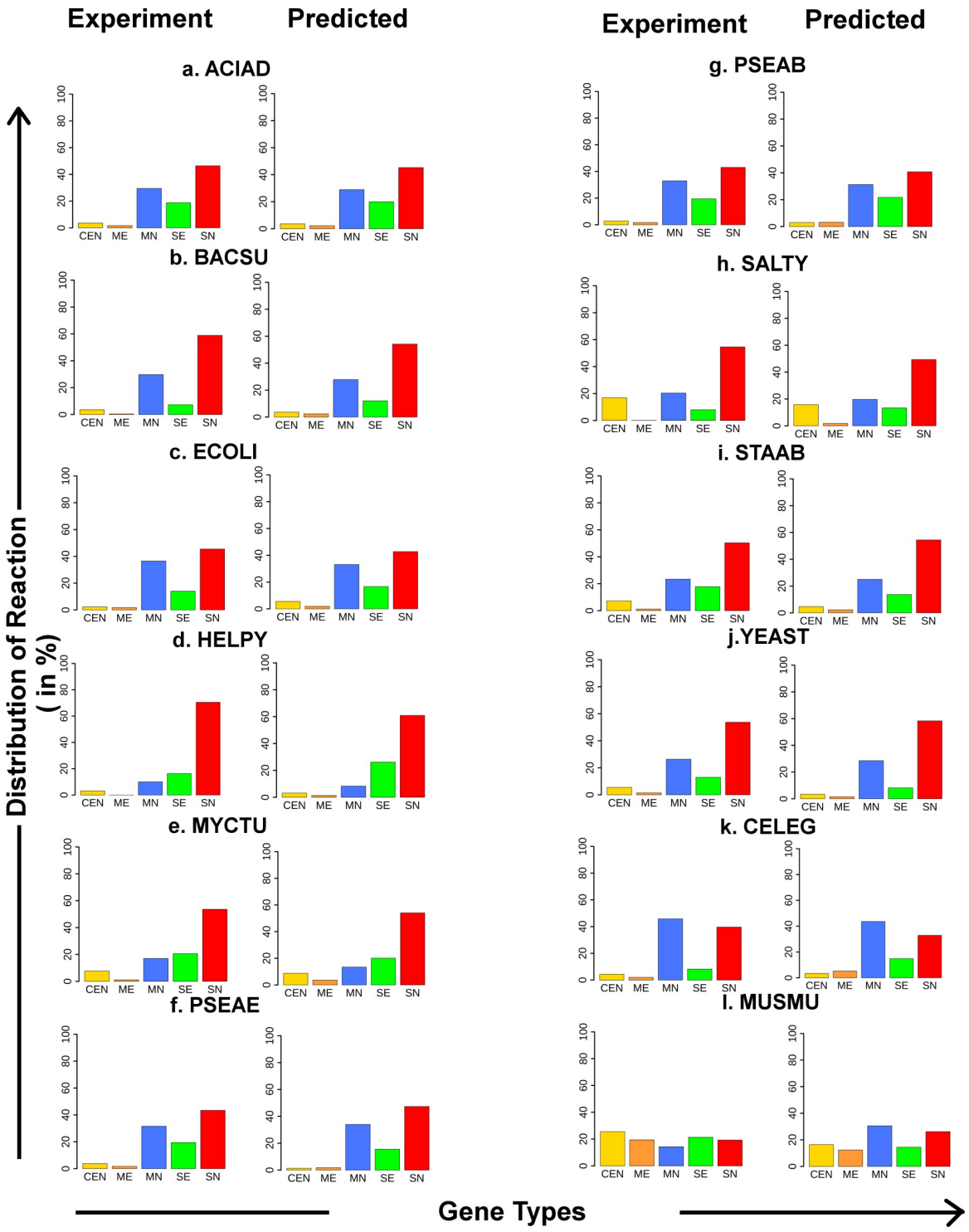


Fig 5. Comparison of the distributions of reaction. The reactions have been classified into five categories and the predicted distributions of reaction-gene pairs have been compared with the experimental data across all twelve organisms.

https://doi.org/10.1371/journal.pone.0242943.g005

testing) dataset as well as the whole dataset for training and testing (Figs 2 and 4). This proposed pipeline consists of three key steps. First, the unsupervised feature selection algorithm

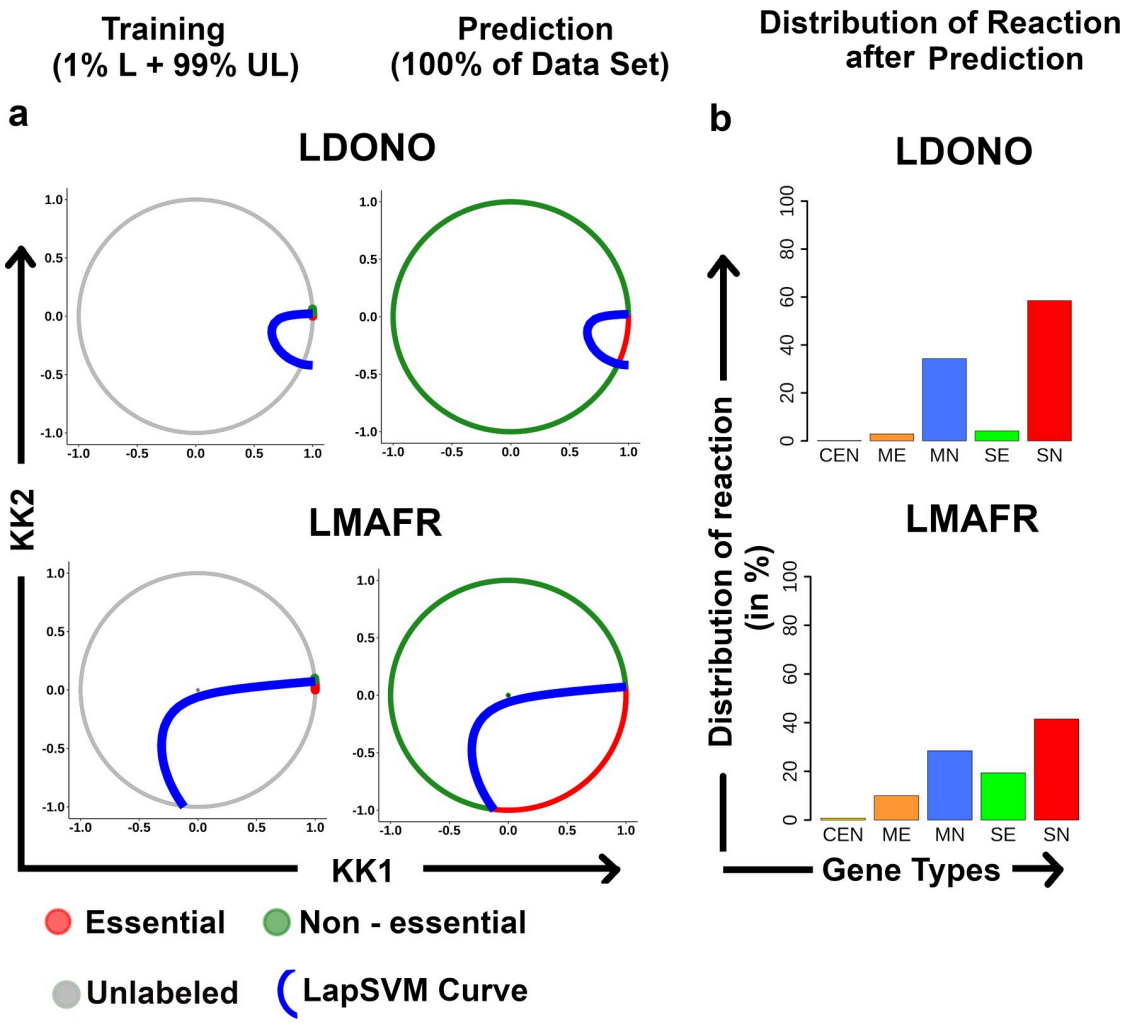


Fig 6. Gene essentiality prediction in *L. donovani* and *L. major*. **(a)** Kamada—Kawai dimension reduction on *Leishmania* datasets showed a circular pattern as observed for other organisms and the learning curve by LapSVM; **(b)** Distribution of reaction-gene pairs of *Leishmania* species into five categories.

https://doi.org/10.1371/journal.pone.0242943.g006

has been used to select the relevant feature set from 289 feature set consisting of different heterogeneous biological features such as sequence-based features, and topological features derived from metabolic reaction network, and flux-coupled sub-network which help to distinguish between essential and non-essential reaction gene combinations. Here, it is observed that for every organism, the features selection algorithm selected three phenotypic features that have shown high correlation with gene essentiality, *viz*., Reaction Network betweenness centrality (RN_betweenness), Reaction Network Page Rank centrality (RN_page_rank), and Flux Coupled Analysis Network Page Rank centrality (FCA_page_rank). Apart from these, novel features considered in this study, such as Information-theoretic features (Fourier sine coefficient and Fourier cosine coefficient derived from Kidera factor), were also correlated with gene essentiality prediction in most of the organisms. A distinguishing pattern between essential and non-essential genes for the selected features was captured by the feature selection algorithm, which helped the classifier to predict gene essentiality more accurately. Secondly, data set after feature selection was projected into a 2-D circular layout using the dimension reduction step Kamada-Kawai. This step is essential to project the high dimensional data into

a 2-D plane, which helps the classifier LapSVM to perform significantly better for all the organisms ($P<0.01$) (S2 Table). The results show that this dimension reduction step is capable of improving the prediction accuracy even without feature selection (Fig 3, S3 Table). However, we have also retained the feature selection step in our pipeline to identify the important features that are contributing to gene essentiality classification. After applying Kamada-Kawai, a distinct structured pattern was observed, showing the essential reaction-gene combinations clustered together and the non-essential reaction-gene combination in another cluster, each residing on the arc of a 2-D circular layout for each of the twelve known organisms (Fig 4). This clustered pattern of reaction-gene pairs helped the semi-supervised classifier (Laplacian SVM) build a non-linear curve that dissects this circle into essential and non-essential classes with significantly higher accuracy. The novelty of the proposed strategy lies in the integration of the Kamada-Kawai algorithm with the semi-supervised LapSVM classifier that contributes to the high accuracy obtained using the pipeline. This is evident from S2 Table, where a significantly higher model performance of the Kamada-Kawai step was observed over the other widely used dimension reduction techniques. Further, it has been observed that the LapSVM classifier, when combined with the Kamada-Kawai step, contributes to the higher predictive performance of this pipeline as compared to the other supervised machine learning techniques when only 1% labeled data is available (S3 Fig).

Thereafter, the SSMSS score was used to select the best model. Here it was observed that the selected model based on this scoring technique had a corresponding high auROC value when compared with the experimentally known labels (S2 Fig). This indicated the reliability of the proposed SSMSS score, which, although show high variation for less number of labeled data, is useful as an alternative score when the calculation of supervised metrics is difficult for best model selection.

After the successful validation of this strategy on twelve organisms, the methodology was used to annotate gene essentiality in less-studied organisms like *Leishmania donovani* and *Leishmania major*, for which less or no organism-specific machine learning studies are available. Here, it was observed that 80 reaction-gene pairs were predicted to be essential in *Leishmania donovani*. These reactions involved 44 genes that were mostly associated with ATP binding [GO:0005524], oxidoreductase activity [GO:0016491], and AMP deaminase activity [GO:0003876] GO terms. Similarly, in the case of *Leishmania major*, 335 reaction-gene pairs were predicted as essential that involve 194 genes. Here it is observed that in addition to the ATP binding and metal-ion binding activities [GO:0005524], some genes that were predicted to be essential were also associated with amino acid transmembrane transporter activity [GO:0015171], magnesium ion binding [GO:0000287], and protein serine/threonine kinase activity [GO:0004674] GO terms that were not observed in the *L. donovani*. On the other hand, in the case of *L. donovani*, the genes involved in flavin adenine dinucleotide binding [GO:0050660] and AMP deaminase activity [GO:0003876] were predicted as essential, which is not observed in *L. major*.

The KEGG pathway enrichment study performed on the essential gene sets of the two organisms–*L. donovani* and *L. major* throw light on the pathways that are crucial for the survival of these micro-organisms and can be considered as probable therapeutic targets. Here, it is observed that apart from the pathways involved in Purine metabolism, Pyrimidine metabolism, Pyruvate metabolism, etc., that were common to both the organisms, a set of unique pathways were also enriched in each of *L.major* and *L.donovani*. While in the case of *L. major*, the pathways involved in Glycolysis/Gluconeogenesis, Glycine, serine and threonine metabolism, Citrate cycle (TCA cycle), Pyruvate metabolism, and Inositol phosphate metabolism were significantly enriched ($P< 0.001$), the essential genes of *L. donovani* show a higher enrichment for Sphingolipid metabolism and Steroid biosynthesis pathways. Further, the

predicted essential reaction-gene combinations were categorized into five different groups (i.e., CEN, ME, MN, SE, and SN) that help to identify the individual reactions that are regulated by single or multiple essential genes. It may be mentioned here that a common pattern in these categories of distributions was observed across all the twelve organisms that corroborate well with the experimental observations (Fig 5). The Chi-Square Test performed to verify the difference in the experimental and predicted distributions showed no significant difference (S4 Table). A similar pattern was also predicted for *L. donovani* and *L. major* that further ascertains the validity of the predictions (Fig 6b). These results indicate the strength of the model in identifying true essential genes using a small amount labeled data, a selection of biologically relevant features to represent gene essentiality, and optimal parameters for curve formation to classify essential genes. The limitation of the proposed strategy is that, it requires the genome-scale reconstructed metabolic network, and at least 1% genes of this network should be annotated experimentally with gene essentiality information.

Using a graph-based semi-supervised machine learning scheme and combining different well-established methods in ML problems, a novel integrative approach has been proposed for essential gene prediction that shows universality in application to both prokaryotes and eukaryotes with limited labeled data. The run time of the pipeline is dependent on the size of the metabolic network (n), and the number of features (d) considered and can be represented as T(n,d) = $O(n^3d^2)$. In the case of *L. major* and *L donovani*, the total runtime was 41 minutes and 48 minutes, respectively, when simulated on a workstation of Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz with 32GB RAM. This strategy will provide experimental biologists a well standardized and validated methodology to predict gene essentiality of less-studied organisms as well as will cater to the theoretical scientists with a novel approach for binary classification problems when limited labeled data is available. The essential genes predicted using the pipeline provide important leads for the identification of novel therapeutic targets for antibiotic and vaccine development against disease-causing parasites, such as *Leishmania sp*.

Supporting information

S1 Fig. Heatmap plot of selected features by the feature selection algorithm. Red cells indicate features selected by the feature selection algorithm in the corresponding organism. White cells show the feature that is not selected or is redundant.
(TIF)

S2 Fig. Robustness evaluation of the proposed score (SSMSS). Scatter plots is demonstrating an association between auROC and SSMSS in each labeled category data sets in different model parameters conditions for twelve organisms. The X-axis represents the score (SSMSS), and Y-axis represents the corresponding auROC. To represent each category, ten different colors are used.
(TIF)

S3 Fig. Comparison of the predictive performance of the proposed strategy with other supervised methods. Comparison of the performance of proposed strategy (PS) with supervised classifiers [i.e., Decision Tree (DT), Logistic regression (LR), Naive Bayes (NB), Random Forest (RF) and our own previously reported Supervised essential gene prediction pipeline] based on 1% labeled data on twelve organisms. The X-axis represents the different types of performance metrics for machine learning strategies, the Y-axis represents the value of performance metrics. Six different color codes were used to represent six different performance metrics.
(TIF)

S4 Fig. Comparison of the predictive performance on both types of data sets (80% and whole data set). Average predictive performance of the best 100 models on 80% training data set and performance of whole training data set containing the Limited Labeled (L = 1%) and remaining Unlabeled (UL) data for six supervised metrics (i.e., TPR, FPR, F-measure, MCC, auROC, accuracy) and SSMSS for each labeled type. The X-axis represents the different performance metrics, the Y-axis represents the value of performance metrics.
(TIF)

S1 Table. List of curated 289 features. List of curated 289 features for essential gene prediction.
(DOCX)

S2 Table. Comparison of auROC of Kamada-Kawai (KK) dimension reduction technique with PCA, MDS, FR and ICA. The values reported in the table represent the *P*-values obtained using the one-tailed Mann-Whitney U Test.
(DOCX)

S3 Table. Comparison of the effect of feature selection and Kamada-Kawai (KK) dimension reduction technique on the model performance (auROC). The values reported in the table represent the *P*-values obtained using the one-tailed Mann-Whitney U Test.
(DOCX)

S4 Table. Comparison of percentage distribution of reaction into five categories from experiment vs predicted results. The values reported in the table represent the *P*-values obtained using the Chi-square test.
(DOCX)

S5 Table. Gene essentiality information of reaction gene combinations in *Leishmania donovani* predicted using the proposed pipeline.
(DOCX)

S6 Table Gene essentiality information of reaction gene combinations in Leishmania major predicted using the proposed pipeline.
(DOCX)

S7 Table. Gene Ontology (Molecular Function) terms of the predicted essential genes in *Leishmania donovani*.
(DOCX)

S8 Table. Gene Ontology (Molecular Function) terms of the predicted essential genes in *Leishmania major*.
(DOCX)

S9 Table. KEGG pathway enrichment of the predicted essential genes in *Leishmania donovani*.
(DOCX)

S10 Table. KEGG pathway enrichment of the predicted essential genes in *Leishmania major*.
(DOCX)

S1 Text. Source code of proposed machine learning strategy. This supplementary text contains source code for the proposed machine learning strategy, including codes for (a) Training data set preparation and integration of heterogeneous features; (b) Feature selection based on

the space-filling concept; (c) Dimension reduction using forced directed graph layout; (d) Semi-supervised classifier LapSVM.
(DOCX)

Acknowledgments

The authors acknowledge Dr. Leelavati Narlikar, Dr. Abhishek Subramanian, Mr. Kshitij Patil and Mr. Jarjish Rahaman for valuable suggestions and insightful comments.

Author Contributions

Conceptualization: Ram Rup Sarkar.

Data curation: Sutanu Nandi.

Formal analysis: Sutanu Nandi, Piyali Ganguli.

Investigation: Sutanu Nandi.

Methodology: Sutanu Nandi.

Supervision: Ram Rup Sarkar.

Validation: Sutanu Nandi, Piyali Ganguli.

Writing – original draft: Sutanu Nandi.

Writing – review & editing: Piyali Ganguli, Ram Rup Sarkar.

References

1. Baba T, Ara T, Hasegawa M, Takai Y, Okumura Y, Baba M, et al. Construction of Escherichia coli K-12 in-frame, single-gene knockout mutants: the Keio collection. Mol Syst Biol. 2006; 2: 2006.0008. https://doi.org/10.1038/msb4100050 PMID: 16738554
2. Cruz A, Coburn CM, Beverley SM. Double targeted gene replacement for creating null mutants. Proc Natl Acad Sci U S A. 1991; 88: 7170–4. https://doi.org/10.1073/pnas.88.16.7170 PMID: 1651496
3. Gerdes SyS, Scholle MD, Campbell JW, Balazsi G, Ravasz E, Daugherty MD, et al. Experimental determination and system level analysis of essential genes in Escherichia coli MG1655. J Bacteriol. 2003; 185: 5673–5684. https://doi.org/10.1128/jb.185.19.5673-5684.2003 PMID: 13129938
4. Reznikoff WS, Winterberg KM. Transposon-based strategies for the identification of essential bacterial genes. Microb Gene Essentiality Protoc Bioinforma. 2008; 13–26. https://doi.org/10.1007/978-1-59745-321-9_2 PMID: 18392958
5. Agrawal N, Dasaradhi PVN, Mohmmed A, Malhotra P, Bhatnagar RK, Mukherjee SK. RNA interference: biology, mechanism, and applications. Microbiol Mol Biol Rev. 2003; 67: 657–685. https://doi.org/10.1128/mmbr.67.4.657-685.2003 PMID: 14665679
6. Li X, Li W, Zeng M, Zheng R, Li M. Network-based methods for predicting essential genes or proteins: a survey. Brief Bioinform. 2019.
7. Zhang X, Acencio ML, Lemke N. Predicting essential genes and proteins based on machine learning and network topological features: A comprehensive review. Front Physiol. 2016; 7: 1–11. https://doi.org/10.3389/fphys.2016.00001 PMID: 26858649
8. Peng C, Lin Y, Luo H, Gao F. A comprehensive overview of online resources to identify and predict bacterial essential genes. Front Microbiol. 2017; 8: 2331. https://doi.org/10.3389/fmicb.2017.02331 PMID: 29230204
9. Liu W, Fang L, Li M, Li S, Guo S, Luo R, et al. Comparative genomics of Mycoplasma: analysis of conserved essential genes and diversity of the pan-genome. PLoS One. 2012; 7: e35698. https://doi.org/10.1371/journal.pone.0035698 PMID: 22536428
10. Fagen JR, Leonard MT, McCullough CM, Edirisinghe JN, Henry CS, Davis MJ, et al. Comparative genomics of cultured and uncultured strains suggests genes essential for free-living growth of Liberibacter. PLoS One. 2014; 9: e84469. https://doi.org/10.1371/journal.pone.0084469 PMID: 24416233

11. Rout S, Warhurst DC, Suar M, Mahapatra RK. In silico comparative genomics analysis of Plasmodium falciparum for the identification of putative essential genes and therapeutic candidates. J Microbiol Methods. 2015; 109: 1–8. https://doi.org/10.1016/j.mimet.2014.11.016 PMID: 25486552

12. Yang X, Li Y, Zang J, Li Y, Bie P, Lu Y, et al. Analysis of pan-genome to identify the core genes and essential genes of Brucella spp. Mol Genet Genomics. 2016; 291: 905–912. https://doi.org/10.1007/s00438-015-1154-z PMID: 26724943

13. Bruccoleri RE, Dougherty TJ, Davison DB. Concordance analysis of microbial genomes. Nucleic Acids Res. 1998; 26: 4482–4486. https://doi.org/10.1093/nar/26.19.4482 PMID: 9742253

14. Lu Y, Deng J, B Carson M, Lu H, J Lu L. Computational methods for the prediction of microbial essential genes. Curr Bioinform. 2014; 9: 89–101.

15. Joyce AR, Palsson BØ. Predicting gene essentiality using genome-scale in silico models. Microbial Gene Essentiality: Protocols and Bioinformatics. Springer; 2008. pp. 433–457.

16. Basler G. Computational prediction of essential metabolic genes using constraint-based approaches. Gene Essentiality. Springer; 2015. pp. 183–204.

17. Dey A. Machine learning algorithms: a review. Int J Comput Sci Inf Technol. 2016; 7: 1174–1179.

18. Kotsiantis SB, Zaharakis I, Pintelas P. Supervised machine learning: A review of classification techniques. Emerg Artif Intell Appl Comput Eng. 2007; 160: 3–24.

19. Kennedy J, Eberhart R. Particle swarm optimization. Proceedings of ICNN'95-International Conference on Neural Networks. 1995. pp. 1942–1948.

20. Bonabeau E, Dorigo M, Marco D de RDF, Theraulaz G, Théraulaz G, et al. Swarm intelligence: from natural to artificial systems. Oxford university press; 1999.

21. Dorigo M, Birattari M, Stutzle T. Ant colony optimization. IEEE Comput Intell Mag. 2006; 1: 28–39.

22. Mirjalili S, Mirjalili SM, Lewis A. Grey wolf optimizer. Adv Eng Softw. 2014; 69: 46–61.

23. Mirjalili S. The ant lion optimizer. Adv Eng Softw. 2015; 83: 80–98.

24. Hasan MA, Lonardi S. DeeplyEssential: a deep neural network for predicting essential genes in microbes. BMC Bioinformatics. 2020; 21: 1–19. https://doi.org/10.1186/s12859-020-03688-y PMID: 32998698

25. Zampieri G, Vijayakumar S, Yaneske E, Angione C. Machine and deep learning meet genome-scale metabolic modeling. PLoS Comput Biol. 2019; 15: e1007084. https://doi.org/10.1371/journal.pcbi.1007084 PMID: 31295267

26. Deng J, Deng L, Su S, Zhang M, Lin X, Wei L, et al. Investigating the predictability of essential genes across distantly related organisms using an integrative approach. Nucleic Acids Res. 2011; 39: 795–807. https://doi.org/10.1093/nar/gkq784 PMID: 20870748

27. Cheng J, Wu W, Zhang Y, Li X, Jiang X, Wei G, et al. A new computational strategy for predicting essential genes. BMC Genomics. 2013; 14: 910. https://doi.org/10.1186/1471-2164-14-910 PMID: 24359534

28. Hwang Y-C, Lin C-C, Chang J-Y, Mori H, Juan H-F, Huang H-C. Predicting essential genes based on network and sequence analysis. Mol Biosyst. 2009; 5: 1672–1678. https://doi.org/10.1039/B900611G PMID: 19452048

29. Plaimas K, Eils R, König R. Identifying essential genes in bacterial metabolic networks with machine learning methods. BMC Syst Biol. 2010; 4: 1. https://doi.org/10.1186/1752-0509-4-1 PMID: 20056001

30. Plaimas K, Mallm J-P, Oswald M, Svara F, Sourjik V, Eils R, et al. Machine learning based analyses on metabolic networks supports high-throughput knockout screens. BMC Syst Biol. 2008; 2: 67. https://doi.org/10.1186/1752-0509-2-67 PMID: 18652654

31. Chen L, Zhang Y-H, Wang S, Zhang Y, Huang T, Cai Y-D. Prediction and analysis of essential genes using the enrichments of gene ontology and KEGG pathways. PLoS One. 2017; 12: e0184129. https://doi.org/10.1371/journal.pone.0184129 PMID: 28873455

32. Qin C, Sun Y, Dong Y. A new computational strategy for identifying essential proteins based on network topological properties and biological information. PLoS One. 2017; 12: e0182031. https://doi.org/10.1371/journal.pone.0182031 PMID: 28753682

33. Gustafson AM, Snitkin ES, Parker SCJ, DeLisi C, Kasif S. Towards the identification of essential genes using targeted genome sequencing and comparative analysis. BMC Genomics. 2006; 7: 1. https://doi.org/10.1186/1471-2164-7-1 PMID: 16403227

34. Saha S, Heber S, et al. In silico prediction of yeast deletion phenotypes. Genet Mol Res. 2006; 5: 224–232. PMID: 16755513

35. Jin S, Zeng X, Xia F, Huang W, Liu X. Application of deep learning methods in biological networks. Brief Bioinform. 2020. https://doi.org/10.1093/bib/bbaa043 PMID: 32363401

36. Ning LW, Lin H, Ding H, Huang J, Rao N, Guo FB. Predicting bacterial essential genes using only sequence composition information. Genet Mol Res. 2014; 13: 4564–4572. https://doi.org/10.4238/2014.June.17.8 PMID: 25036505

37. Nigatu D, Sobetzko P, Yousef M, Henkel W. Sequence-based information-theoretic features for gene essentiality prediction. BMC Bioinformatics. 2017; 18: 473. https://doi.org/10.1186/s12859-017-1884-5 PMID: 29121868

38. Yu Y, Yang L, Liu Z, Zhu C. Gene essentiality prediction based on fractal features and machine learning. Mol Biosyst. 2017; 13: 577–584. https://doi.org/10.1039/c6mb00806b PMID: 28145541

39. Azhagesan K, Ravindran B, Raman K. Network-based features enable prediction of essential genes across diverse organisms. PLoS One. 2018; 13: e0208722. https://doi.org/10.1371/journal.pone.0208722 PMID: 30543651

40. Nandi S, Subramanian A, Sarkar RR. An integrative machine learning strategy for improved prediction of essential genes in Escherichia coli metabolism using flux-coupled features. Mol Biosyst. 2017; 13: 1584–1596. https://doi.org/10.1039/c7mb00234c PMID: 28671706

41. Raman K, Damaraju N, Joshi GK. The organisational structure of protein networks: revisiting the centrality—lethality hypothesis. Syst Synth Biol. 2014; 8: 73–81. https://doi.org/10.1007/s11693-013-9123-5 PMID: 24592293

42. Guyon I, Elisseeff A. An introduction to variable and feature selection. J Mach Learn Res. 2003; 3: 1157–1182.

43. Platt JC. Fast training of support vector machines using sequential minimal optimization. Adv kernel methods. 1999; 185–208.

44. Belkin M, Niyogi P, Sindhwani V. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. J Mach Learn Res. 2006; 7: 2399–2434.

45. Subramanian A, Sarkar RR. Perspectives on Leishmania Species and Stage-specific Adaptive Mechanisms. Trends Parasitol. 2018; 34: 1068–1081. https://doi.org/10.1016/j.pt.2018.09.004 PMID: 30318316

46. Wei W, Ning L-W, Ye Y-N, Guo F-B. Geptop: a gene essentiality prediction tool for sequenced bacterial genomes based on orthology and phylogeny. PLoS One. 2013; 8: e72343. https://doi.org/10.1371/journal.pone.0072343 PMID: 23977285

47. Matthews BW. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. Biochim Biophys Acta (BBA)-Protein Struct. 1975; 405: 442–451. https://doi.org/10.1016/0005-2795(75)90109-9 PMID: 1180967

48. Chen W-H, Minguez P, Lercher MJ, Bork P. OGEE: an online gene essentiality database. Nucleic Acids Res. 2011; 40: D901—D906. https://doi.org/10.1093/nar/gkr986 PMID: 22075992

49. Durot M, Le Fèvre F, de Berardinis V, Kreimeyer A, Vallenet D, Combe C, et al. Iterative reconstruction of a global metabolic model of Acinetobacter baylyi ADP1 using high-throughput growth phenotype and gene essentiality data. BMC Syst Biol. 2008; 2: 85. https://doi.org/10.1186/1752-0509-2-85 PMID: 18840283

50. Oh Y-K, Palsson BO, Park SM, Schilling CH, Mahadevan R. Genome-scale reconstruction of metabolic network in Bacillus subtilis based on high-throughput phenotyping and gene essentiality data. J Biol Chem. 2007; 282: 28791–28799. https://doi.org/10.1074/jbc.M703759200 PMID: 17573341

51. Orth JD, Conrad TM, Na J, Lerman JA, Nam H, Feist AM, et al. A comprehensive genome-scale reconstruction of Escherichia coli metabolism—2011. Mol Syst Biol. 2011; 7: 535. https://doi.org/10.1038/msb.2011.65 PMID: 21988831

52. Thiele I, Vo TD, Price ND, Palsson BØ. Expanded metabolic reconstruction of Helicobacter pylori (iIT341 GSM/GPR): an in silico genome-scale characterization of single-and double-deletion mutants. J Bacteriol. 2005; 187: 5818–5830. https://doi.org/10.1128/JB.187.16.5818-5830.2005 PMID: 16077130

53. Jamshidi N, Palsson BØ. Investigating the metabolic capabilities of Mycobacterium tuberculosis H37Rv using the in silico strain iNJ 661 and proposing alternative drug targets. BMC Syst Biol. 2007; 1: 26. https://doi.org/10.1186/1752-0509-1-26 PMID: 17555602

54. Bartell JA, Blazier AS, Yen P, Thøgersen JC, Jelsbak L, Goldberg JB, et al. Reconstruction of the metabolic network of Pseudomonas aeruginosa to interrogate virulence factor synthesis. Nat Commun. 2017; 8: 14631. https://doi.org/10.1038/ncomms14631 PMID: 28266498

55. Thiele I, Hyduke DR, Steeb B, Fankam G, Allen DK, Bazzani S, et al. A community effort towards a knowledge-base and mathematical model of the human pathogen Salmonella Typhimurium LT2. BMC Syst Biol. 2011; 5: 8. https://doi.org/10.1186/1752-0509-5-8 PMID: 21244678

56. Li C, Donizelli M, Rodriguez N, Dharuri H, Endler L, Chelliah V, et al. BioModels Database: An enhanced, curated and annotated resource for published quantitative kinetic models. BMC Syst Biol. 2010; 4: 92. https://doi.org/10.1186/1752-0509-4-92 PMID: 20587024

57. Monica LM, Palsson B, Herrgård MJ. Connecting extracellular metabolomic measurements to intracellular flux states in yeast. BMC Syst Biol. 2009; 3: 37. https://doi.org/10.1186/1752-0509-3-37 PMID: 19321003

58. Yilmaz LS, Walhout AJM. A Caenorhabditis elegans genome-scale metabolic network model. Cell Syst. 2016; 2: 297–311. https://doi.org/10.1016/j.cels.2016.04.012 PMID: 27211857

59. Sigurdsson MI, Jamshidi N, Steingrimsson E, Thiele I, Palsson BØ. A detailed genome-wide reconstruction of mouse metabolism based on human Recon 1. BMC Syst Biol. 2010; 4: 140. https://doi.org/10.1186/1752-0509-4-140 PMID: 20959003

60. Sharma M, Shaikh N, Yadav S, Singh S, Garg P. A systematic reconstruction and constraint-based analysis of Leishmania donovani metabolic network: identification of potential antileishmanial drug targets. Mol Biosyst. 2017; 13: 955–969. https://doi.org/10.1039/c6mb00823b PMID: 28367572

61. Chavali AK, Whittemore JD, Eddy JA, Williams KT, Papin JA. Systems analysis of metabolism in the pathogenic trypanosomatid Leishmania major. Mol Syst Biol. 2008; 4. https://doi.org/10.1038/msb.2008.15 PMID: 18364711

62. O'Leary NA, Wright MW, Brister JR, Ciufo S, Haddad D, McVeigh R, et al. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. Nucleic Acids Res. 2015; 44: D733—D745. https://doi.org/10.1093/nar/gkv1189 PMID: 26553804

63. Subramanian A, Sarkar RR. Network structure and enzymatic evolution in Leishmania metabolism: a computational study. BIOMAT 2015: International Symposium on Mathematical and Computational Biology. 2016. pp. 1–20.

64. del Rio G, Koschützki D, Coello G. How to identify essential genes from molecular networks? BMC Syst Biol. 2009; 3: 1. https://doi.org/10.1186/1752-0509-3-1 PMID: 19118495

65. Burgard AP, Nikolaev E V, Schilling CH, Maranas CD. Flux coupling analysis of genome-scale metabolic network reconstructions. Genome Res. 2004; 14: 301–312. https://doi.org/10.1101/gr.1926504 PMID: 14718379

66. Larhlimi A, David L, Selbig J, Bockmayr A. F2C2: a fast tool for the computation of flux coupling in genome-scale metabolic networks. BMC Bioinformatics. 2012; 13: 57. https://doi.org/10.1186/1471-2105-13-57 PMID: 22524245

67. Barabási A-L, et al. Network science. Cambridge university press; 2016. https://doi.org/10.1017/nws.2016.2 PMID: 27867518

68. Liu X, Hong Z, Liu J, Lin Y, Rodr'iguez-Patón A, Zou Q, et al. Computational methods for identifying the critical nodes in biological networks. Brief Bioinform. 2019.

69. Wang J, Li M, Wang H, Pan Y. Identification of essential proteins based on edge clustering coefficient. IEEE/ACM Trans Comput Biol Bioinforma. 2011; 9: 1070–1080.

70. Csardi G, Nepusz T, et al. The igraph software package for complex network research. InterJournal, Complex Syst. 2006; 1695: 1–9.

71. Mann S, Chen Y-PP. Bacterial genomic G+ C composition-eliciting environmental adaptation. Genomics. 2010; 95: 7–15. https://doi.org/10.1016/j.ygeno.2009.09.002 PMID: 19747541

72. dos Reis M, Wernisch L, Savva R. Unexpected correlations between gene expression and codon usage bias from microarray data for the whole Escherichia coli K-12 genome. Nucleic Acids Res. 2003; 31: 6976–6985. https://doi.org/10.1093/nar/gkg897 PMID: 14627830

73. Sharp PM, Li W-H. The codon adaptation index-a measure of directional synonymous codon usage bias, and its potential applications. Nucleic Acids Res. 1987; 15: 1281–1295. https://doi.org/10.1093/nar/15.3.1281 PMID: 3547335

74. Subramanian A, Sarkar RR. Comparison of codon usage bias across Leishmania and Trypanosomatids to understand mRNA secondary structure, relative protein abundance and pathway functions. Genomics. 2015; 106: 232–241. https://doi.org/10.1016/j.ygeno.2015.05.009 PMID: 26043961

75. Wright F. The 'effective number of codons' used in a gene. Gene. 1990; 87: 23–29. https://doi.org/10.1016/0378-1119(90)90491-9 PMID: 2110097

76. Rice P, Longden I, Bleasby A. EMBOSS: the European molecular biology open software suite. Trends Genet. 2000; 16: 276–277. https://doi.org/10.1016/s0168-9525(00)02024-2 PMID: 10827456

77. Bauer M, Schuster SM, Sayood K. The average mutual information profile as a genomic signature. BMC Bioinformatics. 2008; 9: 48. https://doi.org/10.1186/1471-2105-9-48 PMID: 18218139

78. Jordan IK, Rogozin IB, Wolf YI, Koonin EV. Essential genes are more evolutionarily conserved than are nonessential genes in bacteria. Genome Res. 2002; 12: 962–968. https://doi.org/10.1101/gr.87702 PMID: 12045149

79. Scheraga HA, Rackovsky S. Global informatics and physical property selection in protein sequences. Proc Natl Acad Sci. 2016; 113: 1808–1810. https://doi.org/10.1073/pnas.1525745113 PMID: 26831093

80. Kidera A, Konishi Y, Oka M, Ooi T, Scheraga HA. Statistical analysis of the physical properties of the 20 naturally occurring amino acids. J Protein Chem. 1985; 4: 23–55.

81. Laib M, Kanevski M. A Novel Filter Algorithm for Unsupervised Feature Selection Based on a Space Filling Measure. ESANN 2018 proceedings, Eur Symp Artif Neural Networks, Comput Intell Mach Learn Bruges. 2018.

82. Ang JC, Mirzal A, Haron H, Hamed HNA. Supervised, unsupervised, and semi-supervised feature selection: a review on gene selection. IEEE/ACM Trans Comput Biol Bioinforma. 2015; 13: 971–989.

83. Mitra P, Murthy CA, Pal SK. Unsupervised feature selection using feature similarity. IEEE Trans Pattern Anal Mach Intell. 2002; 24: 301–312.

84. Kamada T, Kawai S, et al. An algorithm for drawing general undirected graphs. Inf Process Lett. 1989; 31: 7–15.

85. Kraemer G, Reichstein M, Mahecha MD. dimRed and coRanking—unifying dimensionality reduction in R. R J. 2018; 10: 342–358.

86. Pearson K. LIII. On lines and planes of closest fit to systems of points in space. London, Edinburgh, Dublin Philos Mag J Sci. 1901; 2: 559–572.

87. Torgerson WS. Multidimensional scaling: I. Theory and method. Psychometrika. 1952; 17: 401–419.

88. Fruchterman TMJ, Reingold EM. Graph drawing by force-directed placement. Softw Pract Exp. 1991; 21: 1129–1164.

89. Hyvarinen A. Fast and robust fixed-point algorithms for independent component analysis. IEEE Trans Neural Networks. 1999; 10: 626–634. https://doi.org/10.1109/72.761722 PMID: 18252563

90. Chapelle O, Scholkopf B, Zien A. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. IEEE Trans Neural Networks. 2009; 20: 542.

91. Krijthe JH. RSSL: Semi-supervised Learning in R. International Workshop on Reproducible Research in Pattern Recognition. 2016. pp. 104–115.

92. Rousseeuw PJ. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math. 1987; 20: 53–65.

93. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, et al. Gene ontology: tool for the unification of biology. Nat Genet. 2000; 25: 25. https://doi.org/10.1038/75556 PMID: 10802651

94. Consortium GO. The gene ontology resource: 20 years and still GOing strong. Nucleic Acids Res. 2018; 47: D330—D338.

95. Consortium U. UniProt: a worldwide hub of protein knowledge. Nucleic Acids Res. 2018; 47: D506—D515.

96. Huang DW, Sherman BT, Lempicki RA. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. Nat Protoc. 2009; 4: 44. https://doi.org/10.1038/nprot.2008.211 PMID: 19131956

97. Kanehisa M, Sato Y, Furumichi M, Morishima K, Tanabe M. New approach for understanding genome variations in KEGG. Nucleic Acids Res. 2018; 47: D590—D595.

98. Breiman L. Random forests. Mach Learn. 2001; 45: 5–32.

99. Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers. Mach Learn. 1997; 29: 131–163.

100. Hosmer DW Jr, Lemeshow S, Sturdivant RX. Applied logistic regression. John Wiley & Sons; 2013.

101. Quinlan JR, et al. Bagging, boosting, and C4. 5. AAAI/IAAI, Vol 1. 1996. pp. 725–730.

102. Jones NG, Catta-Preta CMC, Lima APCA, Mottram JC. Genetically validated drug targets in Leishmania: current knowledge and future prospects. ACS Infect Dis. 2018; 4: 467–477. https://doi.org/10.1021/acsinfecdis.7b00244 PMID: 29384366