DATA ANALYTICS IN BIOINFORMATICS

A Machine Learning Perspective

Edited by
Rabinarayan Satpathy
Tanupriya Choudhury
Suneeta Satpathy
Sachi Nandan Mohanty
Xiaobo Zhang

Scrivener Publishing

WILEY

Data Analytics in Bioinformatics

Scrivener Publishing
100 Cummings Center, Suite 541J
Beverly, MA 01915-6106

Publishers at Scrivener
Martin Scrivener (martin@scrivenerpublishing.com)
Phillip Carmical (pcarmical@scrivenerpublishing.com)

Data Analytics in Bioinformatics

A Machine Learning Perspective

Edited by
Rabinarayan Satpathy
Tanupriya Choudhury
Suneeta Satpathy
Sachi Nandan Mohanty
and
Xiaobo Zhang

Scrivener Publishing

WILEY

This edition first published 2021 by John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, USA and Scrivener Publishing LLC, 100 Cummings Center, Suite 541J, Beverly, MA 01915, USA
© 2021 Scrivener Publishing LLC
For more information about Scrivener publications please visit www.scrivenerpublishing.com.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, except as permitted by law. Advice on how to obtain permission to reuse material from this title is available at http://www.wiley.com/go/permissions.

Wiley Global Headquarters
111 River Street, Hoboken, NJ 07030, USA

For details of our global editorial offices, customer services, and more information about Wiley products visit us at www.wiley.com.

Limit of Liability/Disclaimer of Warranty
While the publisher and authors have used their best efforts in preparing this work, they make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives, written sales materials, or promotional statements for this work. The fact that an organization, website, or product is referred to in this work as a citation and/or potential source of further information does not mean that the publisher and authors endorse the information or services the organization, website, or product may provide or recommendations it may make. This work is sold with the understanding that the publisher is not engaged in rendering professional services. The advice and strategies contained herein may not be suitable for your situation. You should consult with a specialist where appropriate. Neither the publisher nor authors shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages. Further, readers should be aware that websites listed in this work may have changed or disappeared between when this work was written and when it is read.

Library of Congress Cataloging-in-Publication Data

ISBN 978-1-119-78553-8

Cover image: Pixabay.Com
Cover design by Russell Richardson

Set in size of 11pt and Minion Pro by Manila Typesetting Company, Makati, Philippines

Printed in the USA

10 9 8 7 6 5 4 3 2 1

Contents

Preface xix

Acknowledgement xxi

Part 1 The Commencement of Machine Learning Solicitation to Bioinformatics 1

1 Introduction to Supervised Learning 3
Rajat Verma, Vishal Nagar and Satyasundara Mahapatra
1.1 Introduction 4
1.2 Learning Process & its Methodologies 5
1.2.1 Supervised Learning 7
1.2.2 Unsupervised Learning 8
1.2.3 Reinforcement Learning 9
1.3 Classification and its Types 10
1.4 Regression 12
1.4.1 Logistic Regression 15
1.4.2 Difference between Linear & Logistic Regression 16
1.5 Random Forest 18
1.6 K-Nearest Neighbor 20
1.7 Decision Trees 21
1.8 Support Vector Machines 22
1.9 Neural Networks 24
1.10 Comparison of Numerical Interpretation 26
1.11 Conclusion & Future Scope 27
References 28

2 Introduction to Unsupervised Learning in Bioinformatics 35
Nancy Anurag Parasa, Jaya Vinay Namgiri, Sachi Nandan Mohanty and Jatindra Kumar Dash
2.1 Introduction 36

2.2 Clustering in Unsupervised Learning 37
2.3 Clustering in Bioinformatics—Genetic Data 38
2.3.1 Microarray Analysis 38
2.3.2 Clustering Algorithms 40
2.3.3 Partition Algorithms 41
2.3.3.1 k-Means Clustering 41
2.3.3.2 Cluster Center Initialization Algorithm (CCIA) 41
2.3.3.3 Intelligent Kernel k-Mean (IKKM) 41
2.3.3.4 Clustering Large Applications (CLARA) 42
2.3.4 Hierarchical Clustering Algorithms 42
2.3.4.1 AGNES (Agglomerative Nesting) 43
2.3.4.2 DIANA (Divisive Analysis) 43
2.3.4.3 CURE (Clustering Using Representatives) 43
2.3.4.4 CHAMELEON 43
2.3.4.5 BRICH (Balanced Iterative Reducing and Clustering Using Hierarchies) 44
2.3.5 Density-Based Approach 44
2.3.5.1 DBSCAN 44
2.3.6 Model-Based Approach 45
2.3.6.1 SOM (Self-Organizing Maps) 45
2.3.7 Grid-Based Clustering 45
2.3.7.1 STING (Statistical Information Grid-Based Algorithm) 46
2.3.8 Soft Clustering 46
2.3.8.1 FCM (Fuzzy Class Membership) 46
2.4 Conclusion 46
References 47

3 A Critical Review on the Application of Artificial Neural Network in Bioinformatics 51
Vrs Jhalia and Tripti Swarnkar
3.1 Introduction 52
3.1.1 Different Areas of Application of Bioinformatics 52
3.1.2 Bioinformatics in Real World 53
3.1.3 Issues with Bioinformatics 56
3.1.3.1 Issues Related to Structure 56
3.1.3.2 Sequence Analysis 56
3.2 Biological Datasets 57
3.3 Building Computational Model 58
3.3.1 Data Pre-Processing and its Necessity 58
3.3.2 Biological Data Classification 59

3.3.3 ML in Bioinformatics 60
3.3.4 Introduction to ANN 61
3.3.5 Application of ANN in Bioinformatics 63
3.3.6 Broadly Used Supervised Machine Learning Techniques 64
3.4 Literature Review 64
3.4.1 Comparative Analysis of ANN With Broadly Used Traditional ML Algorithms 67
3.5 Critical Analysis 72
3.6 Conclusion 73
References 73

Part 2 Machine Learning and Genomic Technology, Feature Selection and Dimensionality Reduction 77

4 Dimensionality Reduction Techniques: Principles, Benefits, and Limitations 79
Hemanta Kumar Palo, Santanu Sahoo and Asit Kumar Subudhi
4.1 Introduction 80
4.2 The Benefits and Limitations of Dimension Reduction Methods 81
4.3 Components of Dimension Reduction 83
4.3.1 Feature Selection 84
4.3.2 Feature Reduction 86
4.4 Methods of Dimensionality Reduction 86
4.4.1 Principal Component Analysis (PCA) 88
4.4.2 Missing Values Ratio (MVR) 89
4.4.3 Linear Discriminant Analysis (LDA) 90
4.4.4 Backward Feature Elimination (BFE) 92
4.4.5 Forward Feature Construction (FFC) 93
4.4.6 Independent Component Analysis (ICA) 94
4.4.7 Low Variance Filter (LVF) 95
4.4.8 High Correlation Filter 97
4.4.9 Random Forests (RF)/Ensemble Trees 97
4.4.10 t-Distributed Stochastic Neighbor Embedding (t-SNE) 99
4.4.11 Autoencoder 100
4.4.12 Factor Analysis (FA) 100
4.4.13 Uniform Manifold Approximation and Projection (UMAP) 101
4.4.14 Information Gain (IG) 101
4.4.15 Vector Quantization (VQ) 102

4.5 Conclusion 104
References 105

5 Plant Disease Detection Using Machine Learning Tools With an Overview on Dimensionality Reduction 109
Saurav Roy, Ratula Ray, Satya Ranjan Dash and Mrunmay Kumar Giri
5.1 Introduction 110
5.2 Flowchart 112
5.3 Machine Learning (ML) in Rapid Stress Phenotyping 113
5.4 Dimensionality Reduction 114
5.4.1 Feature Extraction 114
5.4.1.1 PCA (Principal Component Analysis) 115
5.4.1.2 LDA (Linear Discriminant Analysis) 115
5.4.1.3 SIFT (Scale Invariant Feature Transform) 115
5.4.1.4 SURF (Speeded Up Robust Features) 116
5.4.1.5 ORB (Oriented FAST and Rotated BRIEF) 116
5.5 Literature Survey 116
5.6 Types of Plant Stress 128
5.6.1 Biotic Stress 128
5.6.1.1 Fungal Pathogen 129
5.6.1.2 Bacterial Pathogen 129
5.7 Implementation I: Numerical Dataset 130
5.7.1 Dataset Description 130
5.7.2 Results 131
5.7.3 Discussion 133
5.8 Implementation II: Image Dataset 134
5.8.1 Dataset Description 134
5.8.2 Method Used 134
5.8.3 Results 134
5.8.3.1 Results of ORB Feature Extraction and Brute Force Matching 134
5.8.3.2 Color Histogram Comparison: Using Correlation Method 138
5.8.4 Discussions 138
5.9 Conclusion 140
References 141

6 Gene Selection Using Integrative Analysis of Multi-Level Omics Data: A Systematic Review 145
S. Mahapatra and T. Swarnkar
6.1 Introduction 146

6.2 Approaches for Gene Selection 147
6.3 Multi-Level Omics Data Integration 152
6.3.1 Horizontal Integration 153
6.3.2 Vertical Integration 153
6.4 Machine Learning Approaches for Multi-Level Data Integration 153
6.4.1 Unsupervised Integration of Omics Data 159
6.4.2 Supervised Integration of Omics Data 163
6.5 Critical Observation 165
6.6 Conclusion 166
References 166

7 Random Forest Algorithm in Imbalance Genomics Classification 173
Sudhansu Shekhar Patra, Om Praksah Jena, Gaurav Kumar, Sreyashi Pramanik, Chinmaya Misra and Kamakhya Narain Singh
7.1 Introduction 173
7.2 Methodological Issues 175
7.2.1 Decision Tree (DT) Classifier 175
7.2.2 Ensemble Techniques 177
7.2.3 Mathematical Formulation of Ensemble Technique 177
7.2.4 Bagging 178
7.2.5 Bagging Pseudocode 179
7.2.6 Random Forest 180
7.3 Biological Terminologies 181
7.3.1 DNA 181
7.3.2 Genomics 181
7.3.3 Proteins 183
7.4 Proposed Model 183
7.4.1 Balancing the Data 184
7.4.2 Ensembling of Trees 185
7.5 Experimental Analysis 186
7.6 Current and Future Scope of ML in Genomics 188
7.6.1 Gene Sequencing 188
7.6.2 Services to Consumer 188
7.6.3 Gene Editing 188
7.6.4 Pharmacy Genomics 188
7.6.5 Newborn Genetic Screening 188
7.7 Conclusion 189
References 189

8 Feature Selection and Random Forest Classification for Breast Cancer Disease 191
Shubham Raj, Swati Singh, Avinash Kumar, Sobhangi Sarkar and Chittaranjan Pradhan
8.1 Introduction 192
8.2 Literature Survey 192
8.3 Machine Learning 196
8.4 Feature Engineering 202
8.5 Methodology 204
8.5.1 Dataset Collection 204
8.5.2 Proposed Work 204
8.5.2.1 Selection of Feature by the Means of Correlation and Accuracy Calculation Using Random Forest Classification 206
8.5.2.2 Feature Selection Using one Variety and Accuracy Calculation Using Random Forest Classification 207
8.5.2.3 Feature Elimination Using RFE and Classification Using Random Forest 209
8.6 Result Analysis 209
8.7 Conclusion 210
References 210

9 A Comprehensive Study on the Application of Grey Wolf Optimization for Microarray Data 211
Swati Sucharita, Barnali Sahu and Tripti Swarnkar
9.1 Introduction 212
9.2 Microarray Data 213
9.3 Grey Wolf Optimization (GWO) Algorithm 214
9.3.1 Principle of GWO 216
9.3.2 Mathematical Model of GWO 217
9.3.2.1 The Encircling 217
9.3.2.2 Hunting 218
9.3.2.3 Attacking Prey: (Exploitation) 219
9.3.2.4 Search for Prey: (Exploration) 219
9.3.3 Algorithm and Flow Chart of GWO 219
9.4 Studies on GWO Variants 220
9.4.1 Hybridization 221
9.4.2 Extensions 221
9.4.3 Modification 232

9.5 Application of GWO in Medical Domain 232
9.6 Application of GWO in Microarray Data 232
9.7 Conclusion and Future Work 232
References 243

10 The Cluster Analysis and Feature Selection: Perspective of Machine Learning and Image Processing 249
Aradhana Behura
10.1 Introduction 251
10.2 Various Image Segmentation Techniques 254
10.2.1 Clustering 254
10.2.2 Thresholding 254
10.2.3 Edge-Based Segmentation 254
10.2.4 Region-Based Image Segmentation 255
10.2.5 Watershed 255
10.3 How to Deal With Image Dataset 256
10.3.1 Introduction 256
10.3.2 Image Acquisition 256
10.3.3 Image Pre-Processing 256
10.3.4 Image Enhancement 257
10.3.5 Image Segmentation 258
10.3.6 K-Mean Clustering 259
10.3.6.1 Euclidian Distance 259
10.3.6.2 Clustering 260
10.3.7 Density-Based Spatial Clustering of Application With Noise (DBSCAN) 261
10.3.8 SVM Classifier 262
10.4 Class Imbalance Problem 264
10.4.1 Resampling Approaches 264
10.5 Optimization of Hyperparameter 267
10.6 Case Study 270
10.6.1 Pancreatic and Lung Tumor Prediction in the Machine Learning Era: Unique Supervised and Unsupervised Methodologies 270
10.6.2 Pancreatic Cysts (IPMN) 272
10.7 Using AI to Detect Coronavirus 273
10.7.1 BlueDot AI Technology 273
10.8 Using Artificial Intelligence (AI), CT Scan and X-Ray 274
10.9 Conclusion 276
References 276

Part 3 Machine Learning and Healthcare Applications 281

11 Artificial Intelligence and Machine Learning for Healthcare Solutions 283
Ashok Sharma, Parveen Singh and Gowhar Dar
11.1 Introduction 284
11.2 Using Machine Learning Approaches for Different Purposes 284
11.3 Various Resources of Medical Data Set for Research 286
11.4 Deep Learning in Healthcare 287
11.5 Various Projects in Medical Imaging and Diagnostics 288
11.6 Conclusion 289
References 290

12 Forecasting of Novel Corona Virus Disease (Covid-19) Using LSTM and XG Boosting Algorithms 293
V. Aakash, S. Sridevi, G. Ananthi and S. Rajaram
12.1 Introduction 294
12.2 Machine Learning Algorithms for Forecasting 296
12.3 Proposed Method 300
12.3.1 LSTM (Longest Short-Term Memory) 301
12.3.2 XG Boost (eXtreme Gradient Boosting) Algorithm 303
12.3.3 Polynomial Regression 303
12.3.4 Performance Metrics 304
12.4 Implementation 304
12.4.1 The Main Python Code for LSTM 304
12.4.2 The Main Python Code for Polynomial Regression 305
12.4.3 The Main Python Code for XG Boosting Algorithm 306
12.4.4 Libraries or Methods Used in the Proposed Work 306
12.5 Results and Discussion 307
12.6 Conclusion and Future Work 310
References 310

13 An Innovative Machine Learning Approach to Diagnose Cancer at Early Stage 313
Poongodi, P., Udayakumar, E., Srihari, K. and Sachi Nandan Mohanty
13.1 Introduction 314
13.1.1 Multiscale Cancer Detection 315
13.2 Related Work 317
13.3 Materials and Methods 320

13.4 System Design 322
13.4.1 Artificial Neural Network 326
13.4.2 Back Propagation Network (BPN) 326
13.4.3 Support Vector Machine (SVM) 327
13.4.4 Pre-Processing 329
13.4.5 Feature Extraction 329
13.4.6 Database Updation 330
13.4.7 Classification 330
13.4.8 Clustering 330
13.4.9 Segmentation Using FCM Clustering 330
13.5 Results and Discussion 331
13.6 Conclusion 335
References 335

14 A Study of Human Sleep Staging Behavior Based on Polysomnography Using Machine Learning Techniques 339
Santosh Kumar Satapathy and D. Loganathan
14.1 Introduction 340
14.2 Polysomnography Signal Analysis 341
14.3 Case Study on Automated Sleep Stage Scoring 349
14.3.1 Experimental Data 349
14.3.2 The Methodology 351
14.3.3 Experimental Results and Discussion 351
14.4 Summary and Conclusion 356
References 357

15 Detection of Schizophrenia Using EEG Signals 359
Shalini Mahato, Laxmi Kumari Pathak and Kajal Kumari
15.1 Introduction 360
15.1.1 The Human Brain 360
15.1.2 Schizophrenia 361
15.1.2.1 DSM-V Definition and Diagnosis Criteria of Schizophrenia 361
15.1.2.2 Types of Schizophrenia 362
15.1.2.3 Causes of Schizophrenia 363
15.1.2.4 Symptoms of Schizophrenia 363
15.1.3 Electroencephalograph (EEG) 363
15.1.3.1 Characterizations of EEG Signals 364
15.2 Methodology 367
15.2.1 EEG Signal Processing 367
15.2.2 Removing the Artifacts 367

15.2.3 Feature Extraction 368
15.2.4 Normalization 369
15.2.5 Feature Selection/Reduction 369
15.2.6 Feature Classification 369
15.3 Literature Review 372
15.4 Discussion 372
15.5 Conclusion 388
References 388

16 Performance Analysis of Signal Processing Techniques in Bioinformatics for Medical Applications Using Machine Learning Concepts 391
G. Aparna, G. Anitha Mary and G. Sumana
16.1 Introduction 392
16.1.1 Role of Machine Learning in Bioinformatics 393
16.1.2 Machine Learning Applications for Bioinformatics 393
16.1.3 Recent Trends in Bioinformatics 394
16.1.4 Data Analytics in Bioinformatics 394
16.1.5 Machine Learning Algorithms 396
16.2 Basic Definition of Anatomy and Cell at Micro Level 397
16.2.1 Biological Cells 397
16.2.2 DNA, RNA and Proteins 398
16.3 Signal Processing—Genome Signal Processing 403
16.3.1 Identification of Hotspots 404
16.3.2 Advantages of Computational Hotspot Identification Techniques Over Alanine-Scanning Mutagenesis 405
16.3.3 Overview of Protein Sequences 405
16.3.4 EIIP & CPNR-Based Mapping 407
16.3.5 Feature Extraction Technique 411
16.3.5.1 RRM—Resonant Recognition Model 412
16.3.5.2 Discrete Wavelet Transforms 413
16.4 Hotspots Identification Algorithm 414
16.5 Results—Experimental Investigations 416
16.6 Analysis Using Machine Learning Metrics 418
16.6.1 Theoretical Details of Performance Metrics 418
16.6.2 Comparative Analysis of the Protein Sequence Representation 420
16.6.3 Visual Analysis 423
16.7 Conclusion 424

Appendix 424
A.1 Hotspot Identification Code 424
A.2 Performance Metrics Code 425
References 427

17 Survey of Various Statistical Numerical and Machine Learning Ontological Models on Infectious Disease Ontology 431
Yuvaraj Natarajan, Srihari Kannan and Sachi Nandan Mohanty
17.1 Introduction 432
17.2 Disease Ontology 432
17.3 Infectious Disease Ontology 433
17.4 Biomedical Ontologies on IDO 434
17.5 Various Methods on IDO 435
17.6 Machine Learning-Based Ontology for IDO 436
17.7 Recommendation or Suggestions for Future Study 437
17.8 Conclusions 438
References 438

18 An Efficient Model for Predicting Liver Disease Using Machine Learning 443
Ritesh Choudhary, T. Gopalakrishnan, D. Ruby, A. Gayathri, Vishnu Srinivasa Murthy and Rishabh Shekhar
18.1 Introduction 444
18.2 Related Works 445
18.3 Proposed Model 446
18.3.1 Elements of Experimental Methodology 446
18.3.1.1 Experimental Dataset 447
18.3.1.2 Overview of Data & Analysis 447
18.3.1.3 Data Preprocessing 448
18.3.1.4 Standardization 449
18.3.1.5 Label Encoding 450
18.3.2 Model Building 450
18.3.2.1 Support Vector Machines (SVM) 451
18.3.2.2 Logistic Regression 451
18.3.2.3 Naïve Bayes 452
18.3.2.4 Random Forests 452
18.3.2.5 Gradient Boosting 452
18.3.3 Performance Evaluation 453
18.3.4 Performance Optimization 454
18.3.4.1 N-Fold Cross Validation 454

18.4 Results and Analysis 454
18.5 Conclusion 456
References 456

Part 4 Bioinformatics and Market Analysis 459

19 A Novel Approach for Prediction of Stock Market Behavior Using Bioinformatics Techniques 461
Prakash Kumar Sarangi, Birendra Kumar Nayak and Sachidananda Dehuri
19.1 Introduction 462
19.2 Literature Review 463
19.3 Proposed Work 466
19.3.1 Encoding of Stock Market Price Behavior to Binary String 466
19.3.2 Mapping Binary String Into DNA Sequence 467
19.3.3 Sequence Alignment Using BLAST 467
19.3.4 Prediction Method 468
19.3.5 Decoding Predicted DNA Sequence Into Binary String 469
19.3.6 Mismatching Analysis of Stock Market Behavior 470
19.4 Experimental Study 470
19.4.1 Data Analysis 470
19.4.2 Results 471
19.5 Conclusion and Future Work 482
References 484

20 Stock Market Price Behavior Prediction Using Markov Models: A Bioinformatics Approach 485
Prakash Kumar Sarangi, Birendra Kumar Nayak and Sachidananda Dehuri
20.1 Introduction 486
20.2 Literature Survey 487
20.3 Proposed Work 488
20.3.1 Encoding of Stock Market Price Behavior to Binary Sequence 489
20.3.2 Conversion Between Binary Sequences to Nucleotide Sequence 489
20.3.3 Zero-Order Markov Model 490
20.3.4 First-Order Markov Model 490
20.3.5 Second-Order Markov Model 491

20.3.6 Hidden Markov Model 492
20.3.6.1 HMM for Stock Market Behavior Prediction 494
20.3.7 Decoding Predicted DNA Sequence into a Binary String 496
20.3.8 Mismatching Analysis of Stock Market Behavior 497
20.4 Experimental Work 497
20.4.1 Dataset Preparation 497
20.4.2 Results and Analysis 498
20.4.3 Performance Comparison Between Different Orders Markov Models 502
20.5 Conclusions and Future Work 504
References 505

Index 507

Preface

Machine learning has become increasingly popular in recent decades due to its well-defined algorithms and techniques that enable computers to learn and solve real-life problems which are difficult, time-consuming, and tedious to solve traditionally. Regarded as a subdomain of artificial intelligence, it has a gamut of applications in the field of healthcare, medical diagnosis, bioinformatics, natural language processing, stock market analysis and many more. Recently, there has been an explosion of heterogeneous biological data requiring analysis, retrieval of useful patterns, management and proper storage. Moreover, there is the additional challenge of developing automated tools and techniques that can deal with these different kinds of outsized data in order to translate and transform computational modelling of biological systems and its correlated disciplinary data for further classification, clustering, prediction and decision-making.

Machine learning has justified its potential with its application in extracting relevant information in various biological domains like bioinformatics. It has been successful in dealing with and finding efficient solutions for complex biomedical problems. Prior to the application of machine learning, traditional mathematical as well as statistical models were used along with the domain of expert intelligence to carry out investigations and experiments manually, using instruments, hands and eyes, etc. But such conventional methods alone are not enough to deal with large volumes of different types of biological data. Hence, the application of machine learning techniques has become the need of the hour in research in order to find a solution to complex bioinformatics applications for both the disciplines of computer science and biology. With this in mind, this book has been designed with a number of chapters from eminent researchers who relate and explain the machine learning techniques and their application to various bioinformatics problems such as classification and prediction of disease, feature selection, dimensionality reduction, gene selection, etc. Since the chapters are based on progressive collaborative research work on a broad range of topics and implementations, it will be of interest to both

students and researchers from the computer science as well as biological domains.

This edited book is compiled using four sections, with the first section rationalizing the applications of machine learning techniques in bioinformatics with introductory chapters. The subsequent chapters in the second section flows with machine learning technological applications for dimensionality reduction, feature & gene selection, plant disease analysis & prediction as well as cluster analysis. Further, the third section of the book brings together a variety of machine learning research applications to healthcare domain. Then the book dives into the concluding remarks of machine learning applications to stock market behavioural analysis and prediction.

The Editors
December 2020

Acknowledgement

The editors would like to acknowledge and congratulate all the people who extended their assistance for this book. Our sincere thankfulness goes to each one of the chapter's authors for their contributions, without whose support this book would not have become a reality. Our heartfelt gratefulness and acknowledgement also go to the subject matter experts who could find their time to review the chapters and deliver those in time to improve the quality, prominence as well as uniform arrangement of the chapters in the book. Finally, a ton of thanks to all the team members of Scrivener Publishing for their dedicated support and help in publishing this edited book.

Part 1

THE COMMENCEMENT OF MACHINE LEARNING SOLICITATION TO BIOINFORMATICS

1

Introduction to Supervised Learning

Rajat Verma, Vishal Nagar and Satyasundara Mahapatra*

PSIT, Kanpur, Uttar Pradesh, India

Abstract

Artificial Intelligence (AI) has enhanced its importance through machines in the field of present business scenario. AI delineates the intelligence illustrated by machines and performs in a contrasting manner to the natural intelligence signified by all living objects. Today, AI is popular due to its Machine Learning (ML) techniques. In the field of ML, the performance of a machine depend upon the learning performance of that machine. Hence, the improvement of the machine's performance is always proportional to its learning behavior. These Learning behaviors are obtained from the knowledge of living object's intelligence. An introductory aspect of AI through a detailed scenario of ML is presented in this chapter. In the journey of ML's success, data is the only requirement. ML is known because of its execution through its diverse learning approaches. These approaches are known as supervised, unsupervised, and reinforcement. These are performed only on data, as its quintessential element. In Supervised, attempts are done to find the relationship between the independent variables and the dependent variables. The Independent variables are the input attributes whereas the dependent variables are the target attributes. Unsupervised works are contrary to the supervised approach. The former (i.e. unsupervised) deals with the formation of groups or clusters, whereas the latter (i.e. supervised) deals with the relationship between the input and the target attributes. The third aspect (i.e. reinforcement) works through feedback or reward. This Chapter focuses on the importance of ML and its learning techniques in day to day lives with the help of a case study (heart disease) dataset. The numerical interpretation of the learning techniques is explained with the help of graph representation and tabular data representation for easy understanding.

***Keywords*:** Artificial intelligence, machine learning, supervised, unsupervised, reinforcement, knowledge, intelligence

**Corresponding author*: satyasundara123@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (3–34) © 2021 Scrivener Publishing LLC

1.1 Introduction

In today's world, businesses are moving towards the implementation of automatic intelligence for decision making. This is only possible with the help of a well-known intelligence technique otherwise known as Artificial Intelligence (AI). This intelligence technique also plays a vital role in the field of research, which is nothing but taking decisions instantly. The dimension of AI is bifurcated into sub-domains such as Machine Learning (ML) and Artificial Neural Networks (ANN) [1]. The term ML is also termed as augmented analytics [2] and depicts the development of machine's performances. This is achieved through the previous experiences obtained by the machines, but the traditional learning (i.e. the intelligence used in the mid-1800s) works not so efficiently if compared with the ML [3]. In traditional learning, the user deals with data and programs as an input attribute and provides the output or results whereas, in the case of ML the user provides the data and output or desired results as an input attribute and produces the program or rules as an output attribute. This means that data is more important rather than the programs. This is so because the business world depends on the accuracy level of the program which is used for decision making. The block diagram of Traditional learning is shown below in Figure 1.1 for easy understanding.

Traditional Learning is a manual process whereas the functioning of ML is an automated one. Due to ML, the accuracy of analytic worthiness is increased in different diversified domains. These domains are utilized

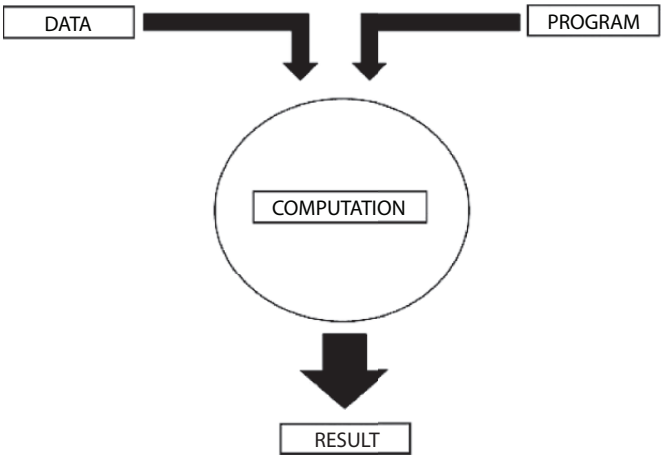


Figure 1.1 Traditional learning.

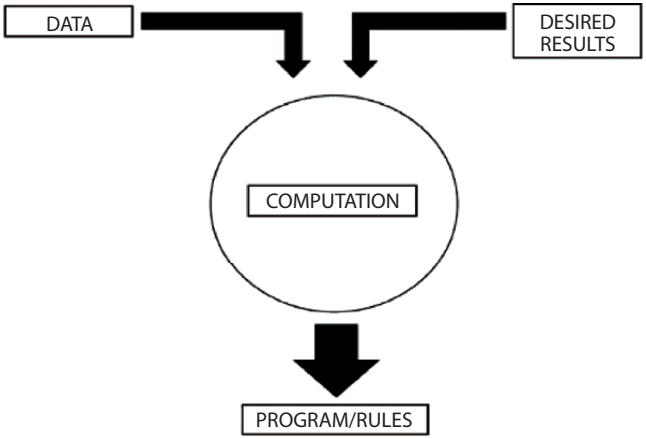


Figure 1.2 Machine learning.

for the preparation of data (raw facts and figures), Outlier Detection (Automatic), Natural Language Interfaces (NLI), and Recommendations, etc. [4]. Due to these domains, the bias factor for taking decisions on a business problem is decreased.

ML is a sub-group of AI and its primary work is allowing systems to learn automatically with the help of data or observations obtained from the environment through different devices [5]. The block-diagram of ML is shown below in Figure 1.2.

ML-based algorithms perform predictions as well as decisions by using mathematical models that are based on some training data [6–8]. Few popular implementations of Machine Learning are Filtering of E-mails [9], Medical Diagnosis [10], Classification [11], Extraction [12], etc. ML works for the growth of the accuracy level of the computer programs. This was done by accessing data from the surrounding, learn the data automatically, and enhancing the capacity of decision making. The main objective of ML is to minimize human intervention and assistance while performing any task. The next section of this chapter highlights the process of learning along with its different methodologies.

1.2 Learning Process & its Methodologies

In AI, Learning means a process to train a machine in such a way so that the machine can take decisions instantly. Hence, the performance of that machine is upgraded because of its accuracy. When a machine performs

in its working environment it may get either success or failure. From these successes or failures machines are gaining experience itself. These newly gained experience, improve the machines through their actions and forms an optimal policy for the working environment. This process is known as learning from experience. This process of learning is possible in an unknown working environment. A general block diagram learning architecture for such a method is presented below in Figure 1.3. This figure tries to present the mechanism of learning a new experience by a machine. The sequence of learning behavior in a stepwise manner is given below.

Step 1. The IoT based sensors received input from the environment.

Step 2. Then, the sensor sends these inputs to the critics for performance evaluation, according to the previously stored performance standards. Simultaneously, the sensor sends the same input to the performance element for checking its effectiveness, if found OK then immediately return the same to the environment through effectors.

Step 3. The Critics provide the feedback to the learning element, if any new feedback occurs then it should be updated in the performance of the element. Then, the updated knowledge comes back to the learning element and send it to the problem generator as a learning goal for evaluating the same through experiments. The updates are sent to the performance of the element for future reference.

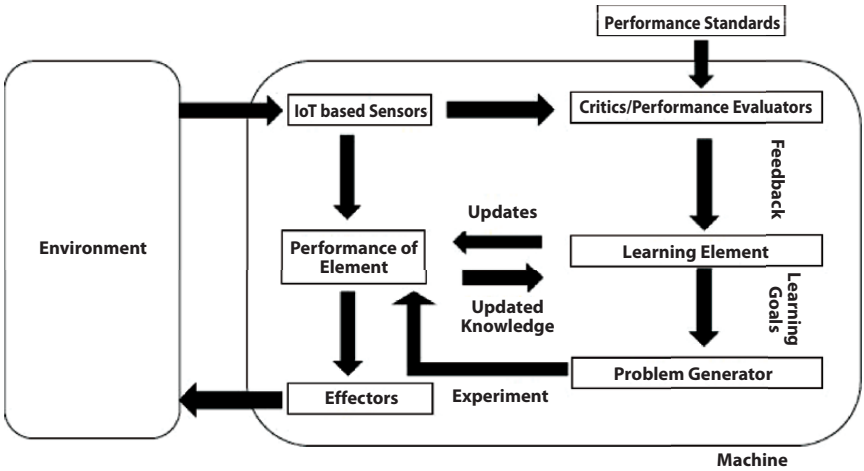


Figure 1.3 Learning behavior of a machine.

The learning process of ML is done in three different ways. These are supervised learning, unsupervised learning, and reinforcement learning. These three learning types have their importance in the different fields of bioinformatics research. Hence, they are explained with suitable examples in the next sections.

1.2.1 Supervised Learning

This is a very common learning mechanism in ML and used by most of the newcomer researchers in their respective fields. This learning mechanism trains the machine by using a labeled dataset in the form of compressed input–output pair as depicted in Refs. [13–15]. These datasets are available in continuous or discrete form. But the important thing is, it needs supervision with an appropriate training model. As supervised learning predicts accurate results [16], hence it is mostly used for Regression analysis and classification purposes. Figure 1.4 shows the execution model of supervised learning.

The figure shows that in supervised learning, a given set of input attributes (i. e. A_1, A_2, A_3, A_4 … … A_k) along with their output attributes (i.e. B_1, B_2, B_3, B_4 … … … B_k) are kept in a knowledge dataset. The Learning Algorithm takes an input A_i and executes with its model and produces the result B_i as the desired output. Supervised Learning has its importance in the field of Bioinformatics as concerning the heart disease scenario where inputs can be a lot of symptoms of heart diseases such as High Cholesterol, Chest Pain, and Blood Pressure, etc. and the output could be a person suffering from heart disease or not. Now all these inputs are passed on to

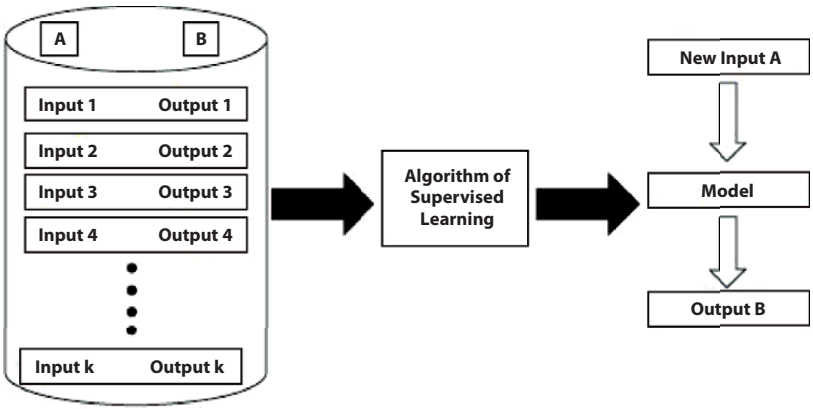


Figure 1.4 Block diagram of supervised learning.

the learning algorithm where it gets trained and if a new input is passed through the model then the machine gives an expected output. If the expected output's accuracy is not up to the mark then there is a need for modification or up-gradation in the model.

An example of supervised learning could be of a person who felt that he has a high cholesterol level and a chest pain and went to the doctor for a check-up. The Doctor fed the inputs given by the patient to the machine. The Machine predicted and told the doctor that the patient is suffering from a cardiac issue in his heart. It acts as an analogy to the supervised learning as the inputs given by the patient are the independent variables and their corresponding output from the machine acts as the dependent attribute. The Machine acted as a model that predicted and gave a relevant output as it is trained by similar inputs. Supervised Learning is itself a huge sub-field of ML and useful for a variety of techniques used in research work. These techniques include Regression Analysis, Artificial Neural Networks (ANN), Support Vector Machines (SVM), etc.

1.2.2 Unsupervised Learning

In Unsupervised Learning, the user doesn't have to supervise the model. Here, the model is allowed to work on its own to find the information. Here clusters are made [17–21]. A block diagram of unsupervised learning is shown in Figure 1.5.

The figure says that in unsupervised learning the inputs are collected as a set of features that are described as $A_1, A_2, A_3, A_4 \ldots \ldots A_k$. But, the output features are not available. The input parameters are passed to a learning

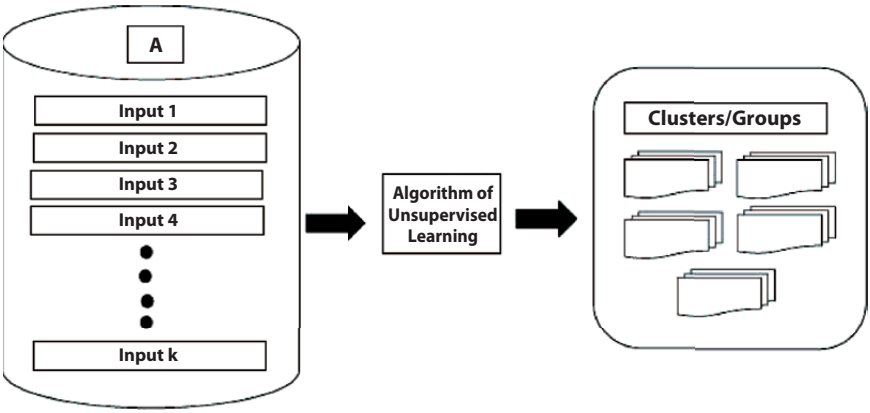


Figure 1.5 Block diagram of unsupervised learning.

algorithm module and diverse groups are formed that are called clusters [22–26].

Unsupervised Learning has its role in Bioinformatics as concerning the heart disease scenario where inputs can be a lot of symptoms of heart diseases such as High Cholesterol, Chest Pain, and Blood Pressure, etc. These symptoms are passed onto the learning algorithm as input where clusters are made by the model and help the patient for identifying a disease (variables/values of similar types in one cluster) that may occur in near future.

1.2.3 Reinforcement Learning

In the field of ML, Reinforcement Learning was developed by John Andreae in the year 1963 when he invented a system called Stella [27]. It is a dynamic approach that works on the concept of feedbacks [28–31]. Reinforcement for a machine is the reward that it receives upon acting in the environment. When the machine acts on its environment, it receives some evaluation on its actions which is called reinforcement but is not told of which action is the correct one for achieving the goal. In this, the machine's utility is defined by the feedback function [32]. The objective is to maximize the expected feedback. The block diagram of reinforcement learning is shown below in Figure 1.6.

The above figure tries to present that, a machine at first performs some actions in the environment. Once the actions are performed, then the machine starts to receive the feedbacks. The collected feedbacks may be positive or negative type. The positive feedbacks are kept inside the machines as knowledge. The machine tries to learn from the negative feedback so that in future such an incident may not happen again. Another important

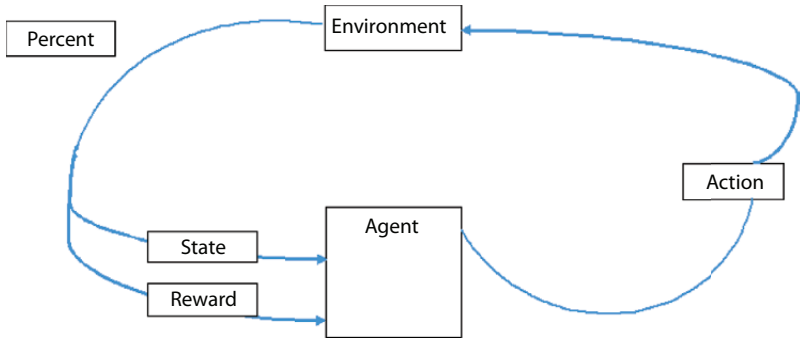


Figure 1.6 Block diagram of reinforcement learning.

aspect of reinforcement learning is the state. The state also provides the input based on the situation to the machine for learning purposes.

A few points of reinforcement learning are as follows:

- The Input of the Reinforcement Learning Process: Initial state
- The Output of the Reinforcement Learning Process: Diversified solutions can be present, depending on the feedbacks obtained
- The training process is purely based on input.
- This Reinforcement Learning model is a continuous process.
- The best solution for this reinforcement learning is the maximum positive feedback.

An example of reinforcement learning could of a person who is suffering from high cholesterol and high blood pressure. He visits his family doctor and requests a medication regarding the same. After analyzing the symptoms, the doctor prescribed a diet chart and a set of medicines to minimize the cholesterol level and blood pressure. He took the medicines and felt better. Here, the patient gets positive feedback in the form of the results of the medication provided by the doctor. Now, the patient will be motivated and will consume only low-fat and low-sodium diet to keep down the levels of blood pressure & cholesterol. If the levels did not go down then the patient will ask the doctor about the same and more tests will be considered for the lowering of the parameters that are required to evaluate the heart of the patient.

1.3 Classification and its Types

Classification is a task in ML, which deals with the organized process of assigning a class label to an observation from the problem domain. It is a sub-group of the supervised form of ML. The traditional classification algorithm was invented by a Swedish botanist Carl Von Linnaeus and depicted in Ref. [33]. In the process of calculating the desired output in supervised learning, this classification is more effective when the input attribute is in the form of a discrete. The Classification approach always helps the user for taking decisions by providing the classified conclusions from the observed data, values as discussed in Refs. [34–36]. Figure 1.7 tries to present a classification graph by executing the data of different persons who are suffering from heart disease or not.

In the above figure, the patients that are suffering from Heart disease are represented by the triangle symbol, and those who are not, are represented

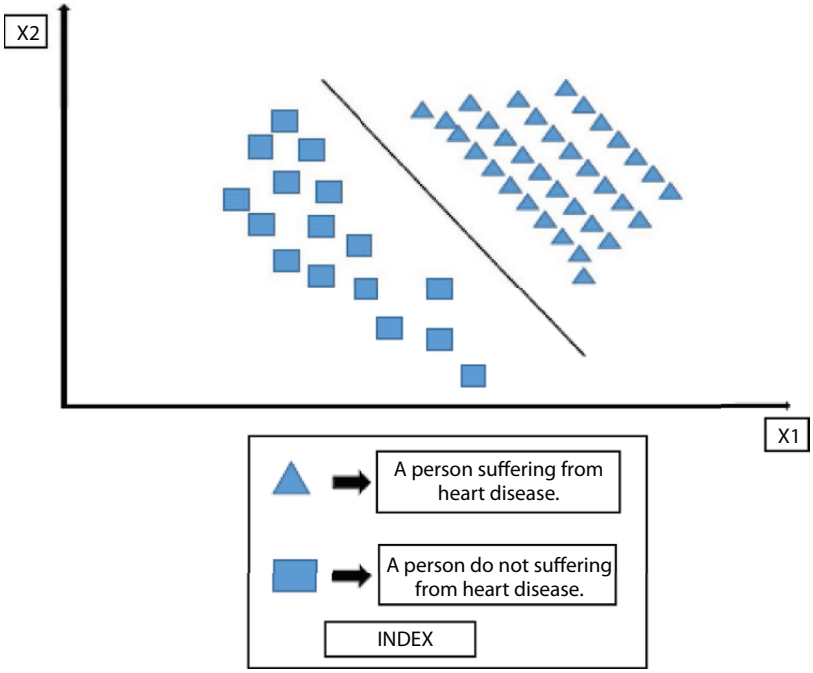


Figure 1.7 Concept of classification.

by rectangle symbols. The hyperplane (partition) line depicts the bifurcation between these two classified entities. In general, there are four types of classification techniques. They are:

- Binary Classification: It considers the tasks of classification where the class labels are two, and the two classes consider one in the normal state and the other in the abnormal state [37].
- Imbalanced Classification: It involves the tasks of classification where the examples are unequally distributed in the class [38].
- Multi-label Classification: It involves the tasks of classification where the number of class labels is two or greater than two where for every example one or more than one class label may be predicted [39].
- Multi-Class Classification: It involves the tasks of classification where the number of class labels is greater than two [40].

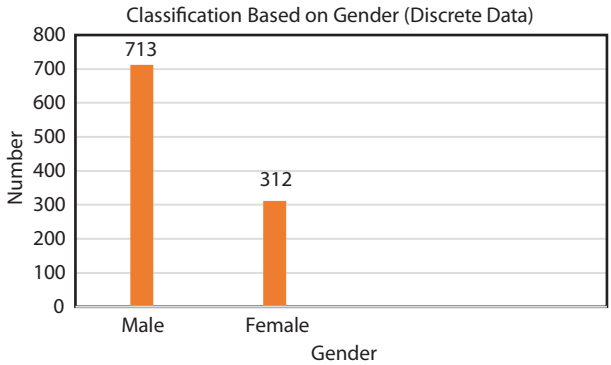


Figure 1.8 Classification based on gender.

For Achieving the Classification approach more precisely, a heart disease dataset [41] has been used that comprises of a total of 1,025 people out of which 312 are females and 713 are males. A particular reason behind taking this dataset is that people are continuously suffering from heart diseases, this is so because people who consume alcohol excessively, consume oily and fast food and also inhale dangerous gases due to pollution. This Classification of gender is given below in Figure 1.8.

1.4 Regression

Regression is a very powerful type of statistical analysis. This is used for finding the strength as well as the character between one dependent variable and a series of independent variables [42–44]. This analysis provides the knowledge on the product that weather any updation in the future is possible or not. The operation of regression provides the ability to a researcher for identifying the best parameter of a topic that can be used for analysis. Also, it provides the parameters that are not to be used for analysis.

In the field of ML, Linear Regression is the most common type of regression analysis for the purpose of prediction [45]. In this process of statistical analysis, equations are made for identifying the useful and not-useful parameters. These are done by linear regression as well as multiple linear regression [46–49]. The representation of Linear Regression is presented in Equation (1.1) and the representation of Multiple Linear Regression is presented in Equation (1.2).

$$B = n + qA + i \tag{1.1}$$

$$B = n + q_1A_1 + q_2A_2 + q_3A_3 \ldots \ldots \ldots q_kA_k + i \tag{1.2}$$

Where,

- *B is known as dependent variable*
- *A or $A_{j \in k}$ are independent variable*
- *n is an intercept*
- *q or $q_{j \in k}$ are slope variables*
- *i is regression residual*
- *k is any natural number.*

For easy understanding, a case study on heart disease is discussed below. In this case study, with the help of the regression approach, a prediction was done whether a person has heart disease or not. Here, the dependent variable is the heart disease and the independent variables are cholesterol levels, blood pressure, etc. After analyzing the data, it was found that the patient has a problem in his heart which is presented below on a 2D plane in Figure 1.9.

The steps required for regression analysis are [50]:

- Select the dependent & independent variables.
- Explore the co-relation matrix along with the scatter plot.
- Perform the Linear or Multiple Regression Operation.
- Accord with the outliers along with the multi-collinearity.
- Perform the t-test.
- Handle the insignificant variables.

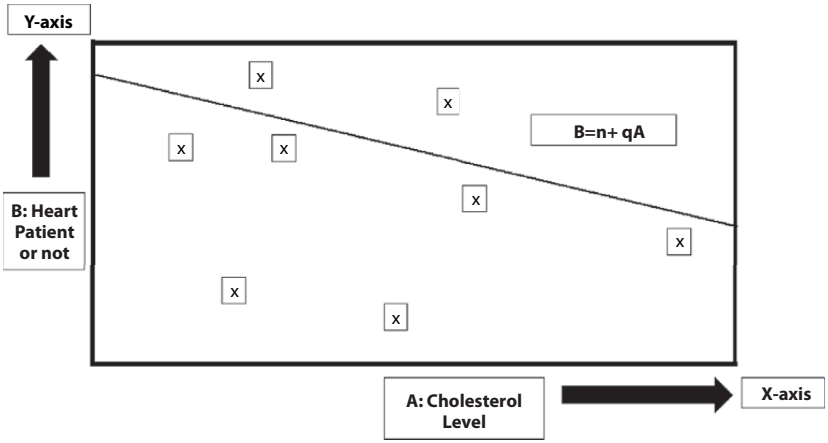


Figure 1.9 Regression.

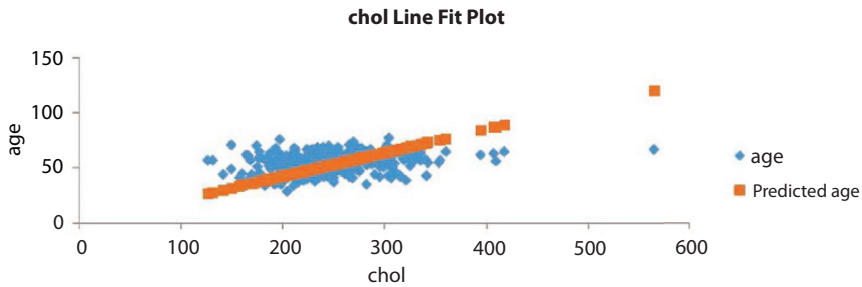


Figure 1.10 Cholesterol line fit plot.

The Regression operation performed on the heart disease dataset concerning the age and cholesterol and got the following results as shown in Figure 1.10.

In the above figure, a line fit plot is mentioned that depicts the line of best fit. This line of best fit is known as the trend line. This trend line is based on a linear equation and try to present the standard cholesterol level of a general human w.r.t. the age. The plot has two axes that include a vertical axis depicting the age and the horizontal axis depicting the cholesterol values. The trend line could be linear, polynomial, or exponential as discussed in Refs. [51–53]. In the process of regression analysis on the heart disease dataset, the following numerical interpretation is obtained and presented in Table 1.1.

Where,

- Multiple R (Co-relation Coefficient): It depicts the strength of a linear relationship between two variables i.e. age and cholesterol of a human. This value always lies between −1 and +1. The obtained value i.e. 0.972834634 indicated that

Table 1.1 Regression statistics.

Regression Statistics	
Multiple R	0.972834634
R Square	0.946407225
Adjusted R Square	0.945430663
Standard Error	12.7814549
Observations	1,025

there is a good relationship between age and cholesterol level.

- R^2**:** It is the coefficient of determination i.e. the goodness of fit. The obtained value is 0.946407225 which indicates that 95% of the values of the heart disease dataset fit the regression model.
- Adjusted R^2: This variable is an upgraded version of R^2. This value tries to adjust the predictor number in the model. This value increases when any new term improves the performance of the model more than the expectation and vice-versa. The obtained value i.e. 0.945430663 indicates that the model is not performing well so there is a need for modification in predictor number.
- Standard Error: It measures the precision of the regression model, the smaller the number, the more accurate the results are. The value obtained is 12.7814549 which indicates that the results are near to accurate value. The Standard Error depicts the measure of how well the data has been approximated.

1.4.1 Logistic Regression

Logistic Regression is a statistical model used for identifying the probability of a class with the help of binary dependent variables i.e. Yes or No. It indicates whether a class belongs to the Yes category or the No category. For example, after executing an event on an object the results maybe Win or Loss, Pass or Fail, Accept or Not-Accept, etc. The mathematical representation of the Logistic Regression model is done by two indicator variables i.e. 0 and 1. It is different from the Linear Regression technique as depicted in Ref. [54]. As logistic regression has its importance in the real-life classification problems as depicted in Refs. [55, 56], different fields like Medical Sciences, Social Sciences, ML are using this model in their various field of operations.

The Logistic Regression is performed on the heart disease dataset [41]. The Receiver Operating Characteristics (ROC) is calculated that is based on the true positive rate that is plotted on the y-axis and the false positive rate that is plotted on the x-axis. After performing the logistic regression in python (Google Colab), the outcome is represented in Figure 1.11 and Table 1.2. Figure 1.11 represents the ROC curve and Table 1.2 represents the Area under the ROC Curve (AUC).

At the time of processing, the AUC value obtained (Table 1.2) on training data is 0.8374022, but when the data is processed for testing then the

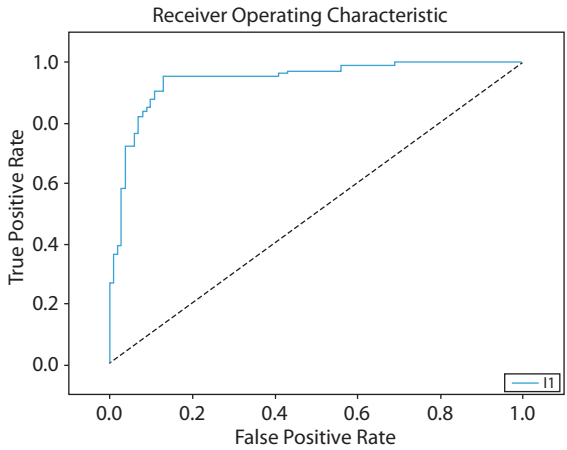


Figure 1.11 ROC curve for logistic regression.

Table 1.2 AUC: Logistic regression.

Parameter	Data	Value	Result
The area under the ROC Curve (AUC)	Training Data	0.8374022	Excellent
	Test Data	0.9409523	Outstanding
	Index: 0.5: No Discriminant, 0.6–0.8: Can be considered accepted, 0.8–0.9: Excellent, >0.9: Outstanding		

obtained result is outstanding (i.e. 0.9409523). This indicates that the model is more than 90% efficient for classification. In the next section, the difference between Linear and Logistic Regression is discussed.

1.4.2 Difference between Linear & Logistic Regression

Linear and Logistics regression are two common types of regression used for prediction. The result of the prediction is represented with the help of numeric variables. The difference between linear and logistic regression is depicted in Table 1.3 for easy understanding.

Linear regression is used to model the data by using a straight line whereas the logistic regression deals with the modeling of probability of events in a bi-variate manner that is occurring as a linear function of

Table 1.3 Difference between linear & logistic regression.

S. No.	Parameter	Linear regression	Logistic regression
1	Purpose	Used for solving regression problems.	Used for solving classification problems.
2	Variables Involved	Continuous Variables	Categorical Variables
3	Objective	Finding of best-fit-line and predicting the output.	Finding of s-curve and classifying the samples.
4	Output	Continuous Variables such as age, price, etc.	Categorical Values such as 0 & 1, Yes & No.
5	Collinearity	There may be collinearity between independent attributes.	There should not be collinearity between independent attributes.
6	Relationship	The relationship between a dependent variable and the independent variable must be linear.	The relationship between a dependent variable and an independent variable may not be linear.
7	Estimation Method Used	Least Square Estimation	Maximum Likelihood Estimation

dependent variables. Few other types of regression analysis are depicted by different scientists and listed below.

- Polynomial Regression: It is used for curvilinear data [57–58].
- Stepwise Regression: It works with predictive models [59–60].
- Ridge Regression: Used for multiple regression data [61–62].
- Lasso Regression: Used for the purpose of variable selection & regularization [63–64].
- Elastic Net Regression: Used when the penalties of lasso and ridge method are combined [65].

1.5 Random Forest

The Random Forest was first invented by Tim Kan Ho [66]. Random Forest is a supervised ensemble learning method, which solves regression and classification problems. It is a method of ensemble learning (i.e. bagging algorithm) and works by averaging the result and by reducing overfitting [67–71]. It is a flexible method and a ready to use in the machine learning algorithm. The Random Forest can be used for the process of regression and known as Regression Forests [72]. It can cope up with the missing values but deals with complexity as well as a longer training period. There are two specific causes for naming it as Random that are:

- When building trees, then a random sampling of training data sets is followed.
- When Splitting nodes, then a random subset of features is considered.

The functioning of random forests is illustrated in Figure 1.12.

In the above figure, five forests are there and each one representing a disease, such as blue represents liver disease, orange represents heart disease, the green tree represents stomach disease, yellow represents lung disease. It was observed that as per the majority of color, Orange is the winner.

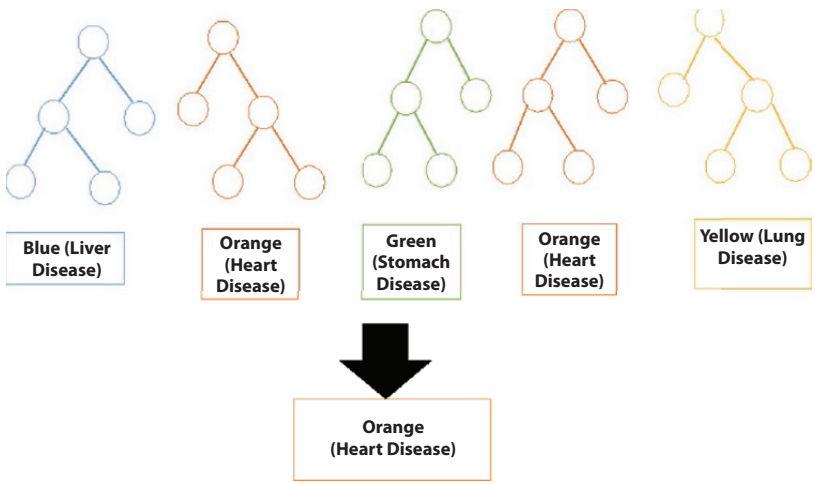


Figure 1.12 Random forest.

This concept is known as the Wisdom of crowd as discussed in Ref. [73]. The execution of this method is achieved with the help of two concepts, which is listed below

- Bagging: The Data on which the decision trees are trained are very sensitive. This means a small change in the data can bring diverse effects in the model. Because of this, the structure of the tree can completely change. They take benefit of it by allowing each tree to randomly sample the dataset with a replacement that results in different trees. This is called bagging or bootstrap aggregation [74–75].
- Random Feature Selection: Normally, when we split a node, every possible feature is considered. The one that produces the most separation is considered. Whereas, in the random forest scenario we can consider a random subset of features. This allows more variation in the model and results in a greater diversification. [76]

The Concept of Random Forest took place in the heart disease dataset also. The low correlation is the key, between the models. The Area under the ROC Curve (AUC) characteristic of Random Forest performed in python (Google Colab) is shown in Table 1.4 and Figure 1.13.

In the above table, the area under the receiver operating characteristic curve (AUC) is mentioned.

AUC measures the degree of separability. The obtained value of Training Data is 1.0000000 that attains an outstanding remark and the value of the testing data is 1.0000000 that attains an outstanding remark in the AUC score. The result indicates that the used models perform outstandingly on the heart disease dataset.

Table 1.4 AUC: Random forest.

Parameter	Data	Value	Result
The area under the ROC Curve (AUC)	Training Data	1.0000000	Outstanding
	Test Data	1.0000000	Outstanding
	Index: 0.5: No Discriminant, 0.6–0.8: Can be considered accepted, 0.8–0.9: Excellent, >0.9: Outstanding		

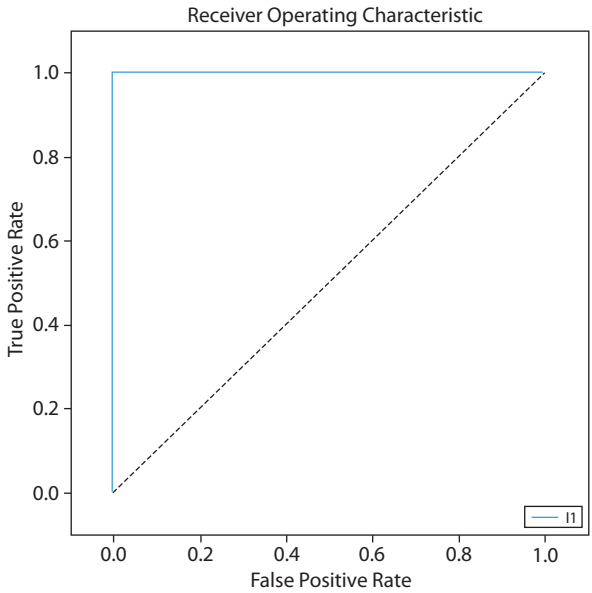


Figure 1.13 ROC curve for random forest.

1.6 K-Nearest Neighbor

K-Nearest Neighbor belongs to the category of supervised classification algorithm and hence, needs labeled data for training [77, 78]. In this approach, the value of K is suggested by the user. It can be used for both the classification and regression approaches but the attributes must be known. By performing the KNN algorithm, it will give new data points according to the k-number or the closest data points.

In the heart disease dataset also, The Area under the ROC Curve (AUC) has been used. It is the most basic tool for judging the classifier's performance in a medical decision making concerns [79–81]. It is a graphical plot for judging the diagnostic ability with the help of a binary classifier. The generated ROC curve for KNN on the heart disease dataset [41] is presented below in Figure 1.14.

In the above figure, the true positive rate (probability of detection) is mentioned on the Y-axis, and on the x-axis, the false positive rate (probability of false alarm) is mentioned. The False Positive rate depicts the unit proportion with a known negative condition for which the predicted condition is positive.

The Area under the ROC Curve (AUC) of K-nearest neighbor is performed on the heart disease dataset [41] in python (Google Colab) and shown below in Table 1.5.

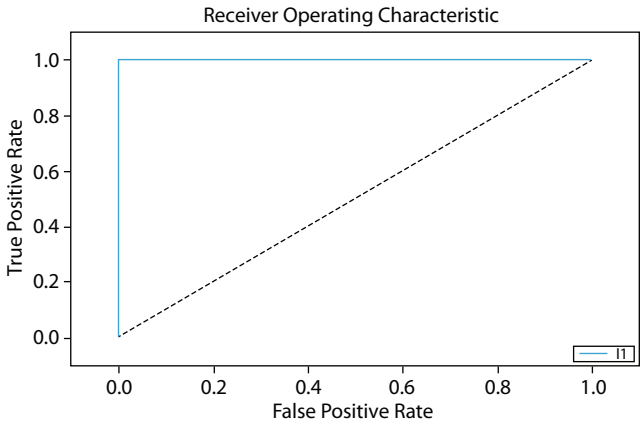


Figure 1.14 ROC curve for k-nearest neighbor.

Table 1.5 AUC: K-nearest neighbor.

Parameter	Data	Value	Result
The area under the ROC Curve (AUC)	Training Data	1.0000000	Outstanding
	Test Data	1.0000000	Outstanding
	Index: 0.5: No Discriminant, 0.6–0.8: Can be considered accepted, 0.8–0.9: Excellent, >0.9: Outstanding		

The obtained value of Training Data is 1.0000000 that attains an outstanding remark and the value of the testing data is 1.0000000 that attains an outstanding remark in the AUC score. The result shows that KNN performs outstandingly on the dataset.

1.7 Decision Trees

Decision Tree is a form of supervised machine learning and was invented by William Belson in the year 1959 [82]. It predicts the response values by learning the decision rules that were derived from features [83–84]. They are good for evaluating the options. It is used in operation research and decision analysis. An example of Decision Trees considering a person is having heart disease or not is presented below in Figure 1.15 for easy understanding.

The above figure depicts the answer to the Question "A person having Heart Disease or not?" by concerning various conditions and reaching a

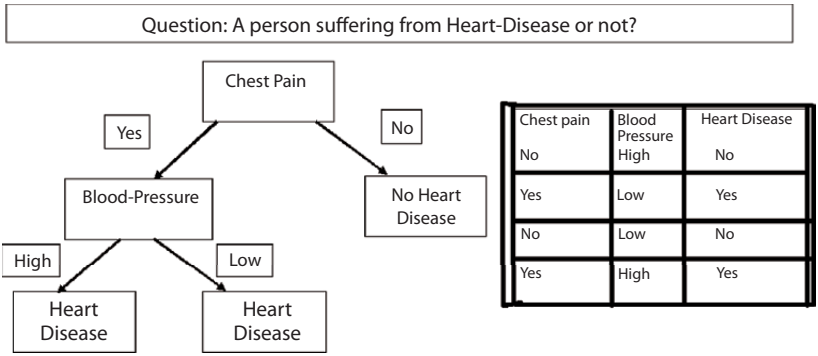


Figure 1.15 Decision tree.

Table 1.6 AUC: Decision trees.

Parameter	Data	Value	Result
The area under the ROC Curve (AUC)	Training Data	0.9588996	Outstanding
	Test Data	0.9773333	Outstanding
	Index: 0.5: No Discriminant, 0.6–0.8: Can be considered accepted, 0.8–0.9: Excellent, >0.9: Outstanding		

conclusion. Initially, it is checked that a person having chest pain or not. If yes, then it is checked that the person has high blood pressure or not. If the blood pressure if high or even low, then the person is suffering from heart disease. If the person doesn't have chest pain then he is not suffering from heart disease. After implementing the Decision tree on the heart disease dataset [41] the AUC values are generated and presented in Table 1.6. The implementation was done in Python (Google Colab).

The obtained value of Training Data is 0.9588996 that attains an outstanding remark and the value of the testing data is 0.9773333 that attains an outstanding remark in the AUC score. The result indicates that the decision tree model performs outstandingly on the heart disease dataset.

1.8 Support Vector Machines

The original Support Vector Machine (SVM) algorithm was invented by Vladimir N. Vapnik & Alexey Ya. Chervonenkis in 1963 [85]. In machine learning, the Support Vector Classifier fits the data that the user provides,

and returns the best-fit hyper-plane that categorizes the data. After getting the hyperplane, the user can feed some features to the classifier to check the predicted class [86–87]. SVM is used for analyzing data that can be used for the process of regression or classification. Taking a similar example of the bifurcation of a person suffering from heart disease or not but giving it a more detailed view, it is depicted in Figure 1.16.

In the above figure, there is an illustration of Support Vector Machines that amalgamates the Hyperplane, Support Vectors, Maximum Margins, and Data Points in a single concept and belongs to either a person is suffering from heart disease or not. Support Vectors are the points that are present very close to the hyperplane and it affects the position and orientation. If they are removed then the position and orientation of the hyperplane will be altered and the maximum margin will also get affected [88–90]. The maximum margin is the distance/length between the nearest points to both classes. Here, Class 1 belongs to the person suffering from heart diseases and Class 2 belongs to the persons who are not suffering from heart diseases. After implementing SVM on the heart

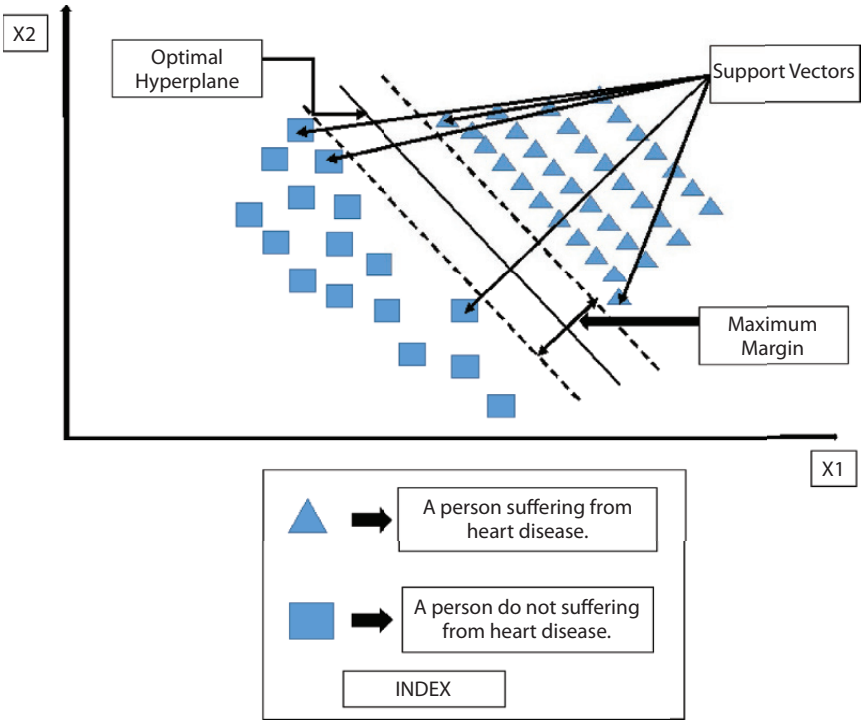


Figure 1.16 Support vector machine.

Table 1.7 AUC: Support vector machines.

Parameter	Data	Value	Result
The area under the ROC Curve (AUC)	Training Data	1.0000000	Outstanding
	Test Data	0.9773333	Outstanding
	Index: 0.5: No Discriminant, 0.6–0.8: Can be considered accepted, 0.8–0.9: Excellent, >0.9: Outstanding		

disease dataset [41] through python (Google Colab), it was observed that the generated AUC values presented in Table 1.7 show that the model performs outstandingly.

1.9 Neural Networks

The Artificial Neural Network (ANN) was invented by Frank Rosenblatt in 1958 [91]. They are inspired by biological neural networks. It is a collection of connected nodes that are called neurons but artificial. The Original goal of Artificial Neural network (ANN), is to solve the problems as the human brain does [92–93]. It does so by taking the inputs, processing them, and calculating the output. Neural Networks can learn by themselves. The outputs that are generated by the neural networks are not limited to the input attributes provided by the user. It doesn't require a database, rather it stores the input in its network. The general form of an artificial neural network is shown below in Figure 1.17 and its detailed version is shown in Figure 1.18. Its other name is the connectionist system. This system learns

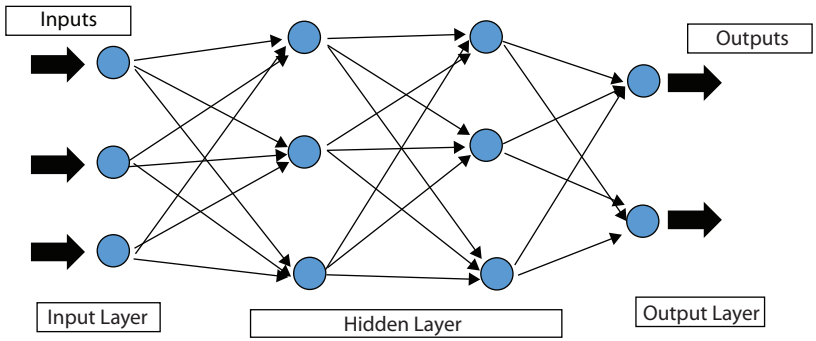


Figure 1.17 Neural network (general).

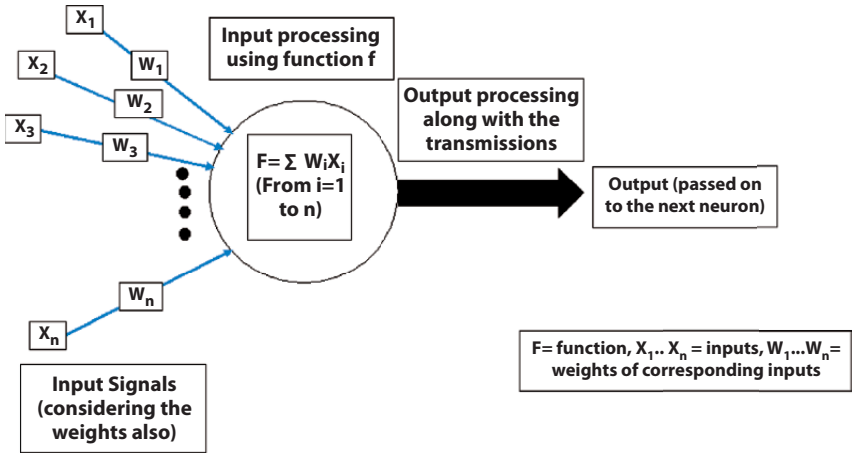


Figure 1.18 Neural network (detailed).

by considering examples and performing tasks. Neural Networks has its applications in various fields such as:

- Speech Recognition [94]
- Signature Verification Application [95]
- Human face Recognition [96]
- Character Recognition [97]
- Natural Language Processing [98].

A basic execution procedure of a neural network [99] is presented in Figure 1.15. In general, it has consisted of three layers. These are the Input layer, Hidden Layer, and the Output layer. These layers are consist of neurons and these neurons and are connected among themselves. In this figure, the input layer contains the health parameters. Depending on the number of inputs received, the hidden processing layer will work to provide an output. Here, only two hidden processing layers are considered but could be any depending on the nature of the purpose of the machine. The outputs are attained and these outputs will act as an input for the next neuron and this process goes on forever. A detailed form of the neural network is explained briefly below with the help of Figure 1.18 for easy understanding.

In Figure 1.16, x_1, x_2, x_3….x_n are the inputs, and the weights they carry are represented by w_1… w_n. Their processing is done by the function F, where it performs summation with values up to n. After processing, the output is transmitted to the next neuron as an input. The AUC obtained

Table 1.8 AUC: Neural network.

Parameter	Data	Value	Result
The area under the ROC Curve (AUC)	Training Data	0.8366730	Excellent
	Test Data	0.9415238	Outstanding
	Index: 0.5: No Discriminant, 0.6–0.8: Can be considered accepted, 0.8–0.9: Excellent, >0.9: Outstanding		

after implementation of the neural network on the heart disease dataset is presented in Table 1.8. It shows that the model is performing excellently on the training dataset and outstanding on the testing dataset. The implementation is done on python (Google Colab).

Some additional Points obtained from the implementation are also presented below:

- Neural Score: 83.78
- Neural Test Score: 90.24
- Accuracy: 0.9024390.

Some other types of Neural Networks are available and listed below for reference:

- Multilayer Perceptron [100]
- Convolutional Neural Network [101]
- Recursive Neural Network [102]
- Recurrent Neural Network [103]
- Long short term memory [104]
- Sequence to Sequence Model [105]
- Shallow neural Network [106].

1.10 Comparison of Numerical Interpretation

A summarized version of the AUC results of the above discussed supervised learning methods is given below in Table 1.9 as a comparison of methods. The result indicates that the performance of Random Forest, K-Nearest Neighbor, Decision Tree, and Support Vector Classifier performs outstandingly in both Train and test data sets. Whereas the Logistic Regression and Neural Network perform Outstanding on the testing data

Table 1.9 AUC: Comparison of numerical interpretations.

S. No.	Supervised Learning Parameter	AUC Training Data Value (T1)	AUC Test Data Value (T2)	Result
1	Logistic Regression	0.8374022	0.9409523	T1: Excellent T2: Outstanding
2	Random Forest	1.0000000	1.0000000	T1: Outstanding T2: Outstanding
3	K-Nearest Neighbor	1.0000000	1.0000000	T1: Outstanding T2: Outstanding
4	Decision Tree	0.9588996	0.9773333	T1: Outstanding T2: Outstanding
5	Support Vector Classifier	1.0000000	0.9773333	T1: Outstanding T2: Outstanding
6	Neural Networks	0.8366730	0.9415238	T1: Excellent T2: Outstanding
Index: 0.5: No Discriminant, 0.6–0.8: Can be considered accepted, 0.8–0.9: Excellent, >0.9: Outstanding				

set only. It indicates that the models used in Logistic Regression and Neural Network need improvement in the training data set. Hence, the accuracy level will be achieved.

1.11 Conclusion & Future Scope

The contribution of AI has been significant from the past six decades. It consisted of the sub-domain, Machine Learning that has also made its mark in the field of research. Its main constituent Supervised Learning is highlighted in this chapter along with its different sub techniques such as a k-nearest neighbor algorithm, classification, regression, decision trees, etc. This chapter also depicts the analysis of a popular dataset of Heart Disease [41] along with its numerical interpretations. The implementation was done on python (Google Colab). A small introductory part of unsupervised learning along with reinforcement learning is also depicted in this chapter.

In the future, the research will be continued in the field of supervised learning (i.e. Logistic Regression and Neural Networks) and its subfields and will try to find out more similarities that may enhance the research perspective.

References

1. Guo, J., He, H., He, T., Lausen, L., Li, M., Lin, H., Zhang, A., Gluoncv and gluonnlp: Deep learning in computer vision and natural language processing. *J. Mach. Learn. Res.*, 21, 23, 1–7, 2020.
2. Abas, Z.A., Rahman, A.F.N.A., Pramudya, G., Wee, S.Y., Kasmin, F., Yusof, N., Abidin, Z.Z., Analytics: A Review Of Current Trends, Future Application And Challenges. *Compusoft*, 9, 1, 3560–3565, 2020.
3. Géron, A., *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, O'Reilly Media, United State of America, 2019.
4. Alshemali, B. and Kalita, J., Improving the reliability of deep neural networks in NLP: A review. *Knowl.-Based Syst.*, 191, 105210, 2020.
5. Klaine, P.V., Imran, M.A., Onireti, O., Souza, R.D., A survey of machine learning techniques applied to self-organizing cellular networks. *IEEE Commun. Surv. Tut.*, 19, 4, 2392–2431, 2017.
6. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Kudlur, M., Tensorflow: A system for large-scale machine learning, in: *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283, 2016.
7. Alpaydin, E., *Introduction to machine learning*, MIT Press, United Kingdom, 2020.
8. Larranaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., Robles, V., Machine learning in bioinformatics. *Briefings Bioinf.*, 7, 1, 86–112, 2006.
9. Almomani, A., Gupta, B.B., Atawneh, S., Meulenberg, A., Almomani, E., A survey of phishing email filtering techniques. *IEEE Commun. Surv. Tut.*, 15, 4, 2070–2090, 2013.
10. Kononenko, I., Machine learning for medical diagnosis: History, state of the art and perspective. *Artif. Intell. Med.*, 23, 1, 89–109, 2001.
11. Kotsiantis, S.B., Zaharakis, I., Pintelas, P., Supervised machine learning: A review of classification techniques, in: *Emerging Artificial Intelligence Applications in Computer Engineering*, vol. 160, pp. 3–24, 2007.
12. Freitag, D., Machine learning for information extraction in informal domains. *Mach. Learn.*, 39, 2–3, 169–202, 2000.
13. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., Improving language understanding by generative pre-training, URL https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/languageunderstanding paper.pdf, 2018.

14. Garcia, V. and Bruna, J., Few-shot learning with graph neural networks, In *Proceedings of the International Conference on Learning Representations (ICLR)*, 3, 1–13, 2018.
15. Miyato, T., Maeda, S.I., Koyama, M., Ishii, S., Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41, 8, 1979–1993, 2018.
16. Tarvainen, A. and Valpola, H., Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results, in: *Advances in Neural Information Processing Systems*, pp. 1195–1204, 2017.
17. Baldi, P., Autoencoders, unsupervised learning, and deep architectures, in: *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, 2012, June, pp. 37–49.
18. Srivastava, N., Mansimov, E., Salakhudinov, R., Unsupervised learning of video representations using lstms, in: *International Conference on Machine Learning*, 2015, June, pp. 843–852.
19. Niebles, J.C., Wang, H., Fei-Fei, L., Unsupervised learning of human action categories using spatial-temporal words. *Int. J. Comput. Vision*, 79, 3, 299–318, 2008.
20. Lee, H., Grosse, R., Ranganath, R., Ng, A.Y., Unsupervised learning of hierarchical representations with convolutional deep belief networks. *Commun. ACM*, 54, 10, 95–103, 2011.
21. Memisevic, R. and Hinton, G., Unsupervised learning of image transformations, in: *2007 IEEE Conference on Computer Vision and Pattern Recognition*, 2007, June, IEEE, pp. 1–8.
22. Dy, J.G. and Brodley, C.E., Feature selection for unsupervised learning. *J. Mach. Learn. Res.*, 5, Aug, 845–889, 2004.
23. Kim, Y., Street, W.N., Menczer, F., Feature selection in unsupervised learning *via* evolutionary search, in: *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2000, August, pp. 365–369.
24. Shi, Y. and Sha, F., Information-theoretical learning of discriminative clusters for unsupervised domain adaptation, *Proceedings of the International Conference on Machine Learning*, 1, pp. 1079–1086, 2012.
25. Balakrishnan, P.S., Cooper, M.C., Jacob, V.S., Lewis, P.A., A study of the classification capabilities of neural networks using unsupervised learning: A comparison with K-means clustering. *Psychometrika*, 59, 4, 509–525, 1994.
26. Pedrycz, W. and Waletzky, J., Fuzzy clustering with partial supervision. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)*, 27, 5, 787–795, 1997.
27. Andreae, J.H., The future of associative learning, in: *Proceedings 1995 Second New Zealand International Two-Stream Conference on Artificial Neural Networks and Expert Systems*, 1995, November, IEEE, pp. 194–197.

28. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M., Playing atari with deep reinforcement learning, in: *Neural information Processing System (NIPS) '13 Workshop on Deep Learning*, 1, pp. 1–9, 2013.
29. Abbeel, P. and Ng, A.Y., Apprenticeship learning *via* inverse reinforcement learning, in: *Proceedings of the Twenty-First International Conference on Machine learning*, 2004, July, p. 1.
30. Wiering, M. and Van Otterlo, M., Reinforcement learning. *Adapt. Learn. Optim.*, 12, 3, 2012.
31. Ziebart, B.D., Maas, A.L., Bagnell, J.A., Dey, A.K., Maximum entropy inverse reinforcement learning, in: *Aaai*, vol. 8, pp. 1433–1438, 2008.
32. Rothkopf, C.A. and Dimitrakakis, C., Preference elicitation and inverse reinforcement learning, in: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2011, September, Springer, Berlin, Heidelberg, pp. 34–48.
33. Anderson, M.J., *Carl Linnaeus: Father of Classification*, Enslow Publishing, LLC, New York, 2009.
34. Becker, H.S., Problems of inference and proof in participant observation. *Am. Sociol. Rev.*, 23, 6, 652–660, 1958.
35. Zaffalon, M. and Miranda, E., Conservative inference rule for uncertain reasoning under incompleteness. *J. Artif. Intell. Res.*, 34, 757–821, 2009.
36. Sathya, R. and Abraham, A., Comparison of supervised and unsupervised learning algorithms for pattern classification. *Int. J. Adv. Res. Artif. Intell.*, 2, 2, 34–38, 2013.
37. Tao, D., Li, X., Hu, W., Maybank, S., Wu, X., Supervised tensor learning, in: *Fifth IEEE International Conference on Data Mining (ICDM'05)*, 2005, November, IEEE, p. 8.
38. Krawczyk, B., Woźniak, M., Schaefer, G., Cost-sensitive decision tree ensembles for effective imbalanced classification. *Appl. Soft Comput.*, 14, 554–562, 2014.
39. Wang, B., Tu, Z., Tsotsos, J.K., Dynamic label propagation for semi-supervised multi-class multi-label classification, in: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 425–432, 2013.
40. Valizadegan, H., Jin, R., Jain, A.K., Semi-supervised boosting for multi-class classification, in: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2008, September, Springer, Berlin, Heidelberg, pp. 522–537.
41. Lapp, D., Heart Disease Dataset, retrieved from https://www.kaggle.com/johnsmith88/heart-disease-dataset.
42. Yildiz, B., Bilbao, J.I., Sproul, A.B., A review and analysis of regression and machine learning models on commercial building electricity load forecasting. *Renewable Sustainable Energy Rev.*, 73, 1104–1122, 2017.
43. Singh, Y., Kaur, A., Malhotra, R., Comparative analysis of regression and machine learning methods for predicting fault proneness models. *Int. J. Comput. Appl. Technol.*, 35, 2–4, 183–193, 2009.

44. Verrelst, J., Muñoz, J., Alonso, L., Delegido, J., Rivera, J.P., Camps-Valls, G., & Moreno, J., Machine learning regression algorithms for biophysical parameter retrieval: Opportunities for Sentinel-2 and-3. *Remote Sens. Environ.*, 118, 127–139, 2012.
45. Razi, M.A. and Athappilly, K., A comparative predictive analysis of neural networks (NNs), nonlinear regression and classification and regression tree (CART) models. *Expert Syst. Appl.*, 29, 1, 65–74, 2005.
46. Lu, Q. and Lund, R.B., Simple linear regression with multiple level shifts. *Can. J. Stat.*, 35, 3, 447–458, 2007.
47. Tranmer, M. and Elliot, M., Multiple linear regression, *The Cathie Marsh Centre for Census and Survey Research (CCSR)*, vol. 5, pp. 30–35, 2008.
48. Kutner, M.H., Nachtsheim, C.J., Neter, J., Li, W., *Applied Linear Statistical Models*, vol. 5, McGraw-Hill Irwin, New York, 2005.
49. Noorossana, R., Eyvazian, M., Amiri, A., Mahmoud, M.A., Statistical monitoring of multivariate multiple linear regression profiles in phase I with calibration application. *Qual. Reliab. Eng. Int.*, 26, 3, 291–303, 2010.
50. Ngo, T.H.D. and La Puente, C.A., The steps to follow in a multiple regression analysis, in: *SAS Global Forum*, 2012, April, vol. 2012, pp. 1–12.
51. Hargreaves, B.R. and McWilliams, T.P., Polynomial trendline function flaws in Microsoft Excel. *Comput. Stat. Data Anal.*, 54, 4, 1190–1196, 2010.
52. Dethlefsen, C. and Lundbye-Christensen, S., *Formulating State Space Models in R With Focus on Longitudinal Regression Models*, Department of Mathematical Sciences, Aalborg University, Denmark, 2005.
53. Brown, A.M., A step-by-step guide to non-linear regression analysis of experimental data using a Microsoft Excel spreadsheet. *Comput. Methods Programs Biomed.*, 65, 3, 191–200, 2001.
54. Tripepi, G., Jager, K.J., Dekker, F.W., Zoccali, C., Linear and logistic regression analysis. *Kidney Int.*, 73, 7, 806–810, 2008.
55. Press, S.J. and Wilson, S., Choosing between logistic regression and discriminant analysis. *J. Am. Stat. Assoc.*, 73, 364, 699–705, 1978.
56. Menard, S., *Applied Logistic Regression Analysis*, vol. 106, Sage, United State of America, 2002.
57. Demartines, P. and Hérault, J., Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. *IEEE Trans. Neural Networks*, 8, 1, 148–154, 1997.
58. Max, T.A. and Burkhart, H.E., Segmented polynomial regression applied to taper equations. *For. Sci.*, 22, 3, 283–289, 1976.
59. Bendel, R.B. and Afifi, A.A., Comparison of stopping rules in forward "stepwise" regression. *J. Am. Stat. Assoc.*, 72, 357, 46–53, 1977.
60. Mahmood, Z. and Khan, S., On the use of k-fold cross-validation to choose cutoff values and assess the performance of predictive models in stepwise regression. *Int. J. Biostat.*, 5, 1, 1–21, 2009.
61. Hoerl, A.E., Kannard, R.W., Baldwin, K.F., Ridge regression: Some simulations. *Commun. Stat.-Theory Methods*, 4, 2, 105–123, 1975.

62. Fearn, T., A misuse of ridge regression in the calibration of a near infrared reflectance instrument. *J. R. Stat. Soc.: Ser. C (Appl. Stat.)*, 32, 1, 73–79, 1983.
63. Hans, C., Bayesian Lasso Regression. *Biometrika*, 96, 4, 835–845, 2009.
64. Zou, H. and Hastie, T., Regularization and variable selection *via the* elastic net. *J. R. Stat. Soc.: Series B (Stat. Methodol.)*, 67, 2, 301–320, 2005.
65. Ogutu, J.O., Schulz-Streeck, T., Piepho, H.P., Genomic selection using regularized linear regression models: ridge regression, lasso, elastic net and their extensions, in: *BMC Proceedings*, 2012, December, BioMed Central, Vol. 6, No. S2, p. S10.
66. Brieuc, M.S., Waters, C.D., Drinan, D.P., Naish, K.A., A practical introduction to Random Forest for genetic association studies in ecology and evolution. *Mol. Ecol. Resour.*, 18, 4, 755–766, 2018.
67. Jurka, T.P., Collingwood, L., Boydstun, A.E., Grossman, E., van Atteveldt, W., RTextTools: A Supervised Learning Package for Text Classification. *R J.*, 5, 1, 6–12, 2013.
68. Criminisi, A., Shotton, J., Konukoglu, E., Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Found. Trends® Comput. Graph. Vis.*, 7, 2–3, 81–227, 2012.
69. Shi, T. and Horvath, S., Unsupervised learning with random forest predictors. *J. Comput. Graph. Stat.*, 15, 1, 118–138, 2006.
70. Settouti, N., Daho, M.E.H., Lazouni, M.E.A., Chikh, M.A., Random forest in semi-supervised learning (Co-Forest), in: *2013 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA)*, 2013, May, IEEE, pp. 326–329.
71. Gu, L., Zheng, Y., Bise, R., Sato, I., Imanishi, N., Aiso, S., Semi-supervised learning for biomedical image segmentation *via* forest oriented super pixels (voxels), in: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2017, September, Springer, Cham, pp. 702–710.
72. Fiaschi, L., Köthe, U., Nair, R., Hamprecht, F.A., Learning to count with regression forest and structured labels, in: *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, 2012, November, IEEE, pp. 2685–2688.
73. Welinder, P., Branson, S., Perona, P., Belongie, S.J., The multidimensional wisdom of crowds, in: *Advances in Neural Information Processing Systems*, pp. 2424–2432, 2010.
74. Oza, N.C., Online bagging and boosting, in: *2005 IEEE international conference on systems, man and cybernetics*, 2005, October, vol. 3, IEEE, pp. 2340–2345.
75. Wang, G., Hao, J., Ma, J., Jiang, H., A comparative assessment of ensemble learning for credit scoring. *Expert Syst. Appl.*, 38, 1, 223–230, 2011.
76. Yerima, S.Y., Sezer, S., Muttik, I., High accuracy android malware detection using ensemble learning. *IET Inf. Secur.*, 9, 6, 313–320, 2015.
77. Weinberger, K.Q. and Saul, L.K., Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.*, 10, Feb, 207–244, 2009.

78. Keller, J.M., Gray, M.R., Givens, J.A., A fuzzy k-nearest neighbor algorithm. *IEEE Trans. Syst. Man Cybern.*, 15, 4, 580–585, 1985.
79. Jain, R., Camarillo, M.K., Stringfellow, W.T., *Drinking Water Security for Engineers, Planners, and Managers*, Oxford: Elsevier, Inc, United Kingdom 2014.
80. Meyer-Baese, A. and Schmid, V.J., *Pattern Recognition and Signal Analysis in Medical Imaging*, Elsevier, Netherlands, 2014.
81. Staelens, S. and Buvat, I., Monte Carlo simulations in nuclear medicine imaging, in: *Advances in Biomedical Engineering*, pp. 177–209, Elsevier, Netherlands, 2009.
82. Murthy, S.K., Automatic construction of decision trees from data: A multi-disciplinary survey. *Data Min. Knowl. Discovery*, 2, 4, 345–389, 1998.
83. Criminisi, A., Shotton, J., Konukoglu, E., Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Found. Trends® Comput. Graph. Vis.*, 7, 2–3, 81–227, 2012.
84. Tanha, J., van Someren, M., Afsarmanesh, H., Semi-supervised self-training for decision tree classifiers. *Int. J. Mach. Learn. Cybern.*, 8, 1, 355–370, 2017.
85. Zahir, N. and Mahdi, H., Snow Depth Estimation Using Time Series Passive Microwave Imagery *via* Genetically Support Vector Regression (case Study Urmia Lake Basin). *The Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.*, 40, 1, 555, 2015.
86. Gualtieri, J.A. and Cromp, R.F., Support vector machines for hyperspectral remote sensing classification, in: *27th AIPR Workshop: Advances in Computer-Assisted Recognition*, 1999, January, vol. 3584, International Society for Optics and Photonics, pp. 221–232.
87. Brefeld, U. and Scheffer, T., Co-EM support vector learning, in: *Proceedings of the Twenty-First International Conference on Machine Learning*, 04, July, p. 16, 20.
88. Chang, C.C. and Lin, C.J., LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol. (TIST)*, 2, 3, 1–27, 2011.
89. Hsu, C.W. and Lin, C.J., A comparison of methods for multiclass support vector machines. *IEEE Trans. Neural Networks*, 13, 2, 415–425, 2002.
90. Shawe-Taylor, J. and Cristianini, N., *Support Vector Machines*, vol. 2, Cambridge University Press, Cambridge, 2000.
91. Marto, A., Hajihassani, M., Jahed Armaghani, D., Tonnizam Mohamad, E., Makhtar, A.M., A novel approach for blast-induced flyrock prediction based on imperialist competitive algorithm and artificial neural network. *Sci. World J.*, 2014, 1–11, 2014.
92. Zhang, Z. and Friedrich, K., Artificial neural networks applied to polymer composites: A review. *Compos. Sci. Technol.*, 63, 14, 2029–2044, 2003.
93. Maind, S.B. and Wankar, P., Research paper on basic of artificial neural network. *Int. J. Recent Innovation Trends Comput. Commun.*, 2, 1, 96–100, 2014.

94. Dahl, G.E., Yu, D., Deng, L., Acero, A., Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Trans. Audio Speech Lang. Processing*, 20, 1, 30–42, 2011.
95. Baltzakis, H. and Papamarkos, N., A new signature verification technique based on a two-stage neural network classifier. *Eng. Appl. Artif. Intell.*, 14, 1, 95–103, 2001.
96. Zhao, Z.Q., Huang, D.S., Sun, B.Y., Human face recognition based on multi-features using neural networks committee. *Pattern Recognit. Lett.*, 25, 12, 1351–1358, 2004.
97. Patil, V. and Shimpi, S., Handwritten English character recognition using neural network. *Elixir Comput. Sci. Eng.*, 41, 5587–5591, 2011.
98. Davydova, 10 Applications of Artificial Neural Networks in Natural Language Processing, Retrieved from https://medium.com/@datamonsters/artificial-neural-networks-in-natural-language-processing-bcf62aa9151a.
99. Murakawa, M., Yoshizawa, S., Kajitani, I., Yao, X., Kajihara, N., Iwata, M., Higuchi, T., The grd chip: Genetic reconfiguration of dsps for neural network processing. *IEEE Trans. Comput.*, 48, 6, 628–639, 1999.
100. Mozolin, M., Thill, J.C., Usery, E.L., Trip distribution forecasting with multilayer perceptron neural networks: A critical evaluation. *Transport. Res. Part B: Meth.*, 34, 1, 53–73, 2000.
101. Kalchbrenner, N., Grefenstette, E., Blunsom., P., A Convolutional Neural Network for Modelling Sentences, in: *Proceedings of ACL*, vol. 1, pp. 655–665, 2014.
102. Setiono, R., Baesens, B., Mues, C., Recursive neural network rule extraction for data with mixed attributes. *IEEE Trans. Neural Networks*, 19, 2, 299–307, 2008.
103. Gregor, K., Danihelka, I., Graves, A., Rezende, D.J., Wierstra, D., Draw, in: *Proceedings of the 32nd International Conference on Machine Learning, PMLR*, vol. 37, pp. 1462–1471, 2015.
104. Zen, H. and Sak, H., Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis, in: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, April, IEEE, pp. 4470–4474.
105. Sutskever, I., Vinyals, O., Le, Q.V., Sequence to sequence learning with neural networks, in: *Advances in Neural Information Processing Systems*, pp. 3104–3112, 2014.
106. Oymak, S. and Soltanolkotabi, M., Towards moderate overparameterization: global convergence guarantees for training shallow neural networks. *IEEE J. Sel. Areas Inf. Theory*, 1, 84–105, 2020.

2

Introduction to Unsupervised Learning in Bioinformatics

Nancy Anurag Parasa[1], Jaya Vinay Namgiri[1], Sachi Nandan Mohanty[2] and Jatindra Kumar Dash[1]*

[1]Department of Computer Science and Engineering, SRM University-AP, Andhra Pradesh, Amaravathi, India
[2]Department of Computer Science and Engineering, IcfaiTech, ICFAI Foundation for Higher Education, Hyderabad, India

Abstract

Unsupervised learning algorithmic techniques are applied in grouping the data depending upon similar attributes, most similar patterns, or relationships amongst the dataset points or values. These Machine learning models are also referred to as self-organizing models which operate on clustering technique. Distinct approaches are employed on every other algorithm in splitting up data into clusters. Unsupervised machine learning uncovers previously unknown patterns in data. Unsupervised machine learning algorithms are applied in case of data insufficiency. Few applications of unsupervised machine learning techniques include: Clustering, anomaly detection. Clustering algorithms in bioinformatics are mostly used to decrypt the salient data in gene expression which is used to acknowledge biological processes in an organism. These models aid in drug design through gene expression profiling. Self organising maps are used in data reduction which provides a better understanding of genomics. Various clustering algorithms are deployed in microarray analysis which is useful in clinical research in keeping track of gene expression data. To define in simpler terms unsupervised learning is a technique which works on the input data to produce the output which is hidden or undetermined. This chapter presents various unsupervised algorithms used for knowledge exploration in the field of bioinformatics and highlights several novel works reported in the recent literature.

***Keywords*:** Clustering, self-organizing-maps, microarray

**Corresponding author*: jatinkdash@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (35–50) © 2021 Scrivener Publishing LLC

2.1 Introduction

Machine Learning can be coined as equipping the machine (computers) to learn from the environment through experience by facilitating the machines with some tasks whose performance can be measured using some metrics and algorithms. This broad spectrum of machine learning is subdivided into few areas as mentioned below.

Supervised learning—In the above categories supervised learning is stipulated as learning system where the data (input) is provided and the output is also known which states that output is dependent on the input provided. From the experience of learning from the data provided this approach predicts labels for the newly given data.

Reinforcement Learning—This learning approach drives on a goal oriented approach in an interactive environment, and functions on the basis of feedback system using the cases rewards and punishments based on the interaction with the data and its outcomes.

Unsupervised Learning—This learning approach explores all the hidden patterns from the input provided as the output is unknown. Prediction is performed on the dataset where the algorithms are applied and the resultant outcome is produced [1].

As the biological data is vast because of compound protein structures and genome sequences, understanding and decrypting the function of cells is resilient. So as to study the rudimentary biological processes, machine learning approaches paves a way to make the system hassle free in developing tools, software and algorithms. This chapter dives in introducing the unsupervised learning approaches, algorithms and their practices in bioinformatics domain which is an interdisciplinary field of science grouping together biology, statistics and computer science in order to analyse and assess the huge amounts of biological data [2].

In unsupervised learning approach the machine learns from the dataset given as input and labels or groups data accordingly [1]. This can also be referred as self-organization, where the algorithm applied structures the data based on the input provided with minimum human intervention. This approach draws all the hidden patterns that exist in the data and also reveals the relationship of the patterns present.

Unsupervised learning basically operates on few common algorithms [3]

- Clustering
- Association
- Anomaly detection

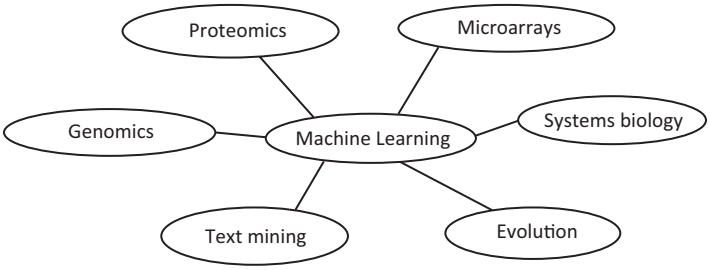


Figure 2.1 Machine learning in bioinformatics.

- Latent variable
- Dimensionality reduction.

Among the above approaches this chapter explores about the algorithmic techniques that are widely applied in bioinformatics paradigm.

Unsupervised learning in bioinformatics—Machine learning in bioinformatics is spread across 6 realms [6] as shown in Figure 2.1.

Genomics and proteomics—the complete set of genes in a cell of an organism is called genome. Genes are structures in which DNA is stored produced from RNA (mRNA-messenger RNA) that is made up from proteins [7]. Every cell of an organism is developed from proteins which are dynamic in nature because every other tissue produces non identical set of proteins. This dynamic nature of proteins is based on the gene expression data. Unlike proteomes, genomes are constant. The set of proteins present in a cell provides insights about the structure and function of a cell [8]. It is difficult to handle gene expression data manually due to its size. Hence machine learning approach such as clustering algorithms are deployed upon varied gene expression data so as to group up similar functions and structures of tissues and explore hidden information.

2.2 Clustering in Unsupervised Learning

Clustering is a technique through which the unlabeled dataset is being grouped based upon the similarity and the characteristics of the data from which a structured output is obtained [4]. The popular algorithms [5] of clustering include

- k-means (partitions the data)
- hierarchial (AGNES—agglomerative nesting)

- Density-based (DBSCAN—Density based spatial clustering with noise)
- Model-based (SOM—self-organizing maps)
- Grid-based (STING—statistical information grid)
- Soft clustering (FCM—Fuzzy Class Membership).

2.3 Clustering in Bioinformatics—Genetic Data

Clustering algorithms in bioinformatics are mostly used to decrypt the salient data in gene expression which is used to acknowledge biological processes in an organism. Gene expression exhibits divergent nature under varying clinical conditions, different tissues and different organisms. The observations drawn from the above conditions enhance the study and analysis of gene functionality. This in turn supports in drug discovery based on the diseased area and the varying nature of genes with diseased condition [28]. With the presence of voluminous amount of genes, it is complex to interpret the data. Therefore, the hidden patterns are being revealed by applying clustering techniques providing a better understanding of functioning of genes and the cellular and biological process of a cell as well [9]. Clustering can be either hard or overlapping. Contemplating every gene as a single cluster is referred as hard clustering whereas overlapping cluster is defined as degree of integration of relatable genes in diverse clusters to every gene expression [10].

2.3.1 Microarray Analysis

The exploration of genomics is centered upon the study of single genes; contrarily microarray analysis is a technology where thousands of gene expressions and its levels are identified under a microscopic slide upon which chips are placed to collect the data which in turn referred as gene chips or DNA chips [12].

In microarray analysis mRNA molecules are gathered from a reference sample (e.g. sample of diseased patient) and test sample (molecules of any individual). The data is combined using probes and if similarities are identified then they are moved into a cluster. If they are found to be dissimilar, they are moved to another cluster. Hence, the clusters are now labeled based on their similarities [12].

During the microarray analysis the data gathered from microarray images are used to construct matrices (refer to Table 2.1) in which rows hold genes and columns hold different conditions viz. different tissues,

Table 2.1 Gene expression data matrix representation.

	Sample 1	Sample 2	Sample 3	Sample 4	Sample m
Gene1	C11	C12	C13	C13	C1m
Gene2	C21	C22	C23	C24	C2m
Gene3	C31	C32	C33	C34	C3m
Gene4	C41	C42	C43	C44	C4m
Gene n	Cn1	Cn2	Cn3	Cn4	Cnm

clinical condition, different biological processes. Expression level data is maintained in each cell of the matrix (refer to Table 2.1) which is uniquely numbered using every gene in every other sample.

These gene expression matrices are summed up into a database as shown in Figure 2.2 which acts as a repository for all the genetic information like gene regulation, genetics functionality of diseases, drug discovery based upon gene structures, response to drug projection and so on. These databases with all the genomic information when searched for particular instance of genes, it retrieves all the relevant information with high similarities using these clustering algorithms [32].

The sequence of actions for Clustering of Gene Expression (GE), include

- A matrix which hold the data of GE structure which consists of size, dimension, number, origin, etc.

	TCGA-A2-A0CV-01A-31R-A115-07	TCGA-E9-A1NG-11A-52R-A14M-07
CHRFAM7A|89832	40.42	53.74
TBC1D24|57465	1126.77	483.00
TCL6|27004	7.00	4.00
TMEM179B|374395	5087.00	1505.00
FLII|2314	8087.00	8557.00
GPC1|2817	4611.00	3561.00
C10orf4|118924	1408.00	976.00
IL20|50604	86.00	33.00
ATP5B|506	32088.00	20368.00
HHEX|3087	711.00	563.00

Figure 2.2 Example matrix of gene expression (10 genes in a row and 2 samples in columns) [29–31].

- Features are extracted to identify the similarities (feature extraction). The input data is reduced without loss of information making it more suitable to application is called feature selection or feature reduction. The feature reduction can be achieved using few common algorithms for example Principal Component Analysis [11].
- Based upon similarities or features, all the genes which express similar expression are made into one cluster and which are distinct are grouped into another cluster. The degree of similarity or the proximity levels are computed using certain distance metrics applied on gene expression data provided. Euclidean distance, Manhattan distance, Mahalanobis distance, etc. are few metrics applied to quantify the level of similarity in order to form a cluster.

2.3.2 Clustering Algorithms

The genetic data (genes) are closely connected. Based upon the dataset appropriate clustering algorithms need to be chosen. These clustering approaches also reveals about the relationship between the clusters based upon the similarity. This section explains various partition algorithms. Figure 2.3 summarizes various clustering based partitions those can be used in Bioinformatics.

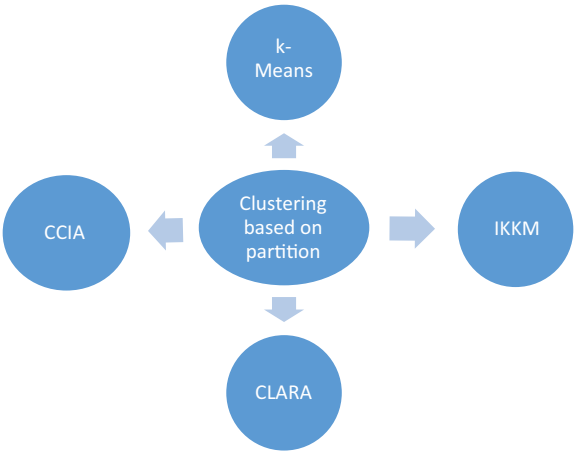


Figure 2.3 Partition clustering algorithms.

2.3.3 Partition Algorithms

2.3.3.1 k-Means Clustering

This algorithm partitions the dataset of n objects into k clusters in the feature space. The value of k (number of clusters) is fixed beforehand. Initially the algorithm randomly assumes k number of cluster centers (cluster mean) in the feature space. The distance (computed using Euclidean distance) of each gene sample is computed from all these cluster means. The sample is then grouped under the cluster that has minimum distance from the sample. The same process is followed for all the samples. Once all the samples are grouped under one of these clusters, each cluster mean is computed again considering the entire samples in that cluster and the cluster mean is updated. The above steps are iterated until the algorithm forms k stable clusters.

The drawback of such algorithm is that determining the clusters (k) prior to implementation reduces the effectiveness of cluster formation and is prone to noisy data in gene expression values [13].

2.3.3.2 Cluster Center Initialization Algorithm (CCIA)

In traditional k-means algorithm the centroids of clusters are randomly selected initially, yielding low quality clusters. As an attempt in producing high quality clustering CCIA is introduced. This algorithm can be referred as an enhancement to k-means algorithm in which centroids (k) are accurately selected which are distant from the outliers. This selection process is done by computing mean and standard deviation on the data elements. CCIA now examines most nearest set of elements from the dataset (D) and fabricates into a new dataset (X) after discarding the outliers. Later those set of elements are eliminated from the dataset (D).This process is iterated until the dataset (X) equals with (k).

Drawback—This algorithm is incapable in dealing with closely connected clusters and cannot deal with high dimensional data [14, 15].

2.3.3.3 Intelligent Kernel k-Mean (IKKM)

Generally gene expression data is nonlinearly separable and the gene data is represented in a matrix format using linear kernel function as shown in Figure 2.2. IKKM is applied on the kernel matrix in which clusters need not be predefined.

The algorithmic approach of IKKM is carried by calculating the centre of mass of the entire dataset and then identifies the element (×1) which is distant from the centre of mass (C) another element (×2) is calculated distant from c1 is calculated. Now the elements near to c1 are moved into cluster (S1) and the elements near to c2 are moved into another cluster (S2).This process is repeated until no change is observed in the clusters. In contrast to traditional k-means approach this algorithm implementation doesn't require predicting the number of clusters in advance.

Drawback—Cannot deal with high dimensional dataset [16].

2.3.3.4 Clustering Large Applications (CLARA)

Rather than considering and calculating medoids for the entire dataset CLARA considers a subset from the dataset with a constant size upon which PAM(partitioning around medoids) algorithm is applied. PAM algorithm aims at reducing the degree of similarity by calculating the distance between the cluster medoids and the elements in the subset space. As a result an optimal set of medoids are obtained which requires less store space when compared to other algorithms. In contrast to other partitioning algorithms this measures the degree of dissimilarity amongst the elements and the medoid of the cluster. This algorithm is applied on huge datasets producing accurate results with less computation time [17].

Drawback—Cannot deal with higher dimensional data and closely connected clusters which leaves this algorithm a second choice in gene expression data.

2.3.4 Hierarchical Clustering Algorithms

Hierarchical clustering technique is again divided into two categories

Agglomerative clustering—This follows a bottom up approach as depicted in Figure 2.4(a). Initially, every other object in the dataset is

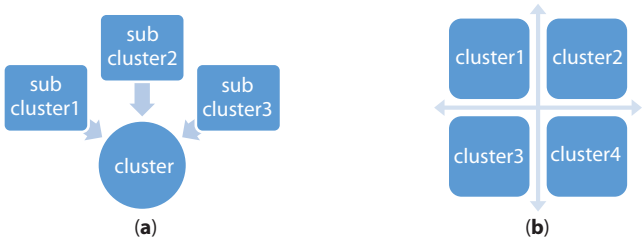


Figure 2.4 (a) Agglomerative clustering, (b) divisive clustering.

assumed as a single cluster. Distance measures are computed upon each object and based upon the similarities they are merged to form clusters.

Divisive Clustering—This follows top down approach as depicted in Figure 2.4(b). Initially, it considers all the data points as single cluster or model. The model separation is performed recursively until required criteria is met based upon the model.

Various algorithms of hierarchical clustering are described below.

2.3.4.1 AGNES (Agglomerative Nesting)

In this algorithm subsequent merging operations are performed on the dataset to arrange them into clusters. Dissimilarity coefficients are acquired using either Euclidean or Manhattan distance metrics [17]. These measured dissimilarities forms a matrix using unweighted pair group method with arithmetic mean [18].

Drawbacks of this method are that the objects once merged cannot be separated and different computation metrics leads to varying results.

2.3.4.2 DIANA (Divisive Analysis)

This follows divisive clustering approach in which the whole dataset is divided into 2 sub parts (clusters) and these clusters are further separated into sub clusters until no splitting can be performed further [17].

Drawback—once separated clusters cannot be merged, this cannot handle huge dataset [19]. This may not be applicable for gene expression data as it cannot handle missing data [17].

2.3.4.3 CURE (Clustering Using Representatives)

This algorithm operates on defining stable scatter points which are able to detect the shape and extent of a cluster. These scatter points based upon the similarities moves towards centroid in order to form a cluster. The scatter points capture the shape and extent of clusters which enables to detect non-spherical clusters. This algorithm is applied to gene expression data which produces the clusters efficiently [20].

2.3.4.4 CHAMELEON

This algorithm is based on dynamic modeling which is executed in two stages. In the first stage graph partitioning algorithm is applied resulting into smaller clusters. As graph partitioning technique is applied the

elements in the clusters are strongly connected. This establishes a quality separation among the clusters by discarding the noisy data. The later stage is executed with the application of agglomerative hierarchical clustering combining all the smaller clusters iteratively. This algorithm doesn't fall short of incorrect merging because of its 2 stage execution. And also CHAMELEON overcomes all the shortcomings of AGNES and DIANA in which the data once split cannot be merged [21].

2.3.4.5 BRICH (Balanced Iterative Reducing and Clustering Using Hierarchies)

In this algorithmic approach entire data in the dataset is maintained in the form of a tree structure called as CF (clustering Feature) which is based on clustering feature. The CF tree also holds the information of clustering. As the outliers are eliminated the sub clusters data is made as leaf nodes. With iterative updating of CF BRICH provides the potential to deal with huge datasets. This works effectively for gene expression data [22].

Drawback—This algorithm gets skewed with the presence of nonspherical clusters [22].

2.3.5 Density-Based Approach

Density based clustering algorithmic approach is composed by splitting the low point density regions and high point density regions in a given data space. The low point density region is observed as outliers [26].

2.3.5.1 DBSCAN

The DBSCAN algorithm proposed by Ref. [27] is based on the following criterion:

Eps(€)—this depicts the space around the data point, if the difference of distance between 2 data points is greater than or equal to € then they are declared as neighbors, if the (€) value is very small then they are termed as outliers, if the (€) value is high then that data point is merged to form cluster.

Minpts—this represents the minimum number of data points within the radius (€) for the larger datasets like microarrays Minpts are obtained from the dimensions of the dataset.

From the following definition of data points clusters and outliers are identified

Core point—presence of higher Minpts within radius of (€)

Border point—presence of less Minpts within radius of (€) but close to core point

Outlier—point which is neither core nor border point.

2.3.6 Model-Based Approach

This clustering algorithm handles the shortcomings of K-Means and Fuzzy K-Means. In this approach a probabilistic model is developed to which the data from the dataset is fit into [23]. Self-Organizing Maps implement the model based approach technique.

2.3.6.1 SOM (Self-Organizing Maps)

This algorithm is popularly known in gene expression data clustering in which the expression values are closely related. SOM adapts a neural network template with a single layer in which data is split over neurons as shown in Figure 2.5. As every neuron is connected to every other neuron and is associated with a reference vector, the data objects are plotted to the nearest reference vectors to form quality clusters [24].

SOM's are highly effective in mapping high dimensional data. The representation of data in the form of map provides quick visualization and interpretation [24].

2.3.7 Grid-Based Clustering

In grid based algorithm rather than working with the data points this operates on space around the data points. The grid structure is made up

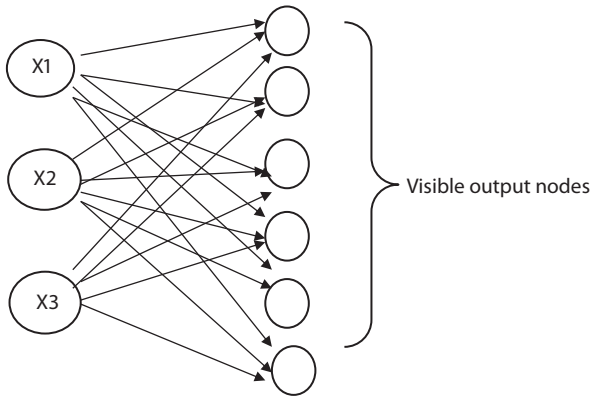


Figure 2.5 Self Organizing Map (SOM).

of cells which segregate the data into number of closely connected cells. Density of each cell is calculated and sorted accordingly [25]. The computational complexity of grid based algorithm is less when compared with other algorithms this makes this approach more adaptable in dealing with large datasets.

2.3.7.1 STING (Statistical Information Grid-Based Algorithm)

The working principle of this algorithm is based on the users input density value with respect to the cluster areas, space with the low density is referred as non-relevant which is discarded as noisy data. Computationally STING requires fewer resources. The grid structure along with the statistical information associated with the cells gives a graphical representation of cluster formed [26].

2.3.8 Soft Clustering

In this approach of soft clustering the data points in the dataset can belong to any cluster this is also defined as a probabilistic model. In simpler terms a single data point can appear in other clusters sharing the similarities. Among the soft clustering approaches FCM is most popular.

2.3.8.1 FCM (Fuzzy Class Membership)

This algorithm is mostly applied in microarray data analysis as microarrays are collection of tens of thousands of genes and analysing them concurrently. This uses a membership function upon which a membership matrix is built from the dataset. This is updated at every instance of similarity check with the data points. The degree of membership is given by the weights of the matrix [25] which specifies the data point how similar it is to the mean of a cluster. The membership values ranges from 0 to 1.

2.4 Conclusion

This chapter provides an overview of unsupervised learning algorithms and approaches used in the field of bioinformatics for the exploration of gene expression data. The chapter provide insights about various clustering algorithms used in the field of bioinformatics. These algorithms when applied on gene expression data helps in building gene expression profiling in which co expressed genes are clustered together that exhibits similar cell

function, identification of gene homology which aids researchers in drug discovery based on the diseased targets using the microarray analyses. These clustering algorithms also comprehend the genetic data in studying about gene functions, identifying sub types of cells which assist in diseased target identification.

References

1. Simeone, O., A Very Brief Introduction to Machine Learning With Applications to Communication Systems. *IEEE Trans. Cognit. Commun. Networking*, 4, 4, 648–664, 2018.
2. Dixit, P. and Prajapati, G.I., Machine Learning in Bioinformatics: A Novel Approach for DNA Sequencing. *2015 Fifth International Conference on Advanced Computing & Communication Technologies*, Haryana, pp. 41–47, 2015.
3. https://en.wikipedia.org/wiki/Unsupervised_learning.
4. Jain, A.K., Data clustering: 50 years beyond k-means. *Pattern Recognit. Lett.*, 31, 8, 1, 651–666, 2010, https://doi.org/10.1016/j.patrec.2009.09.011.
5. Oyelade, J. *et al.*, Data Clustering: Algorithms and Its Applications. *2019 19th International Conference on Computational Science and Its Applications (ICCSA)*, *Saint Petersburg, Russia*, pp. 71–81, 2019.
6. Larrañaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., Lozano, J.A., Armañanzas, R., Santafé, G., Pérez, A., Robles, V., Machine learning in bioinformatics. *Briefings Bioinf.*, 7, 1, 86–112, March 2006, https://doi.org/10.1093/bib/bbk007.
7. National Research Council (US) Committee on Intellectual Property Rights in Genomic and Protein Research and Innovation, Merrill, S.A. and Mazza, A.M. (Eds.), *Reaping the Benefits of Genomic and Proteomic Research: Intellectual Property Rights, Innovation, and Public Health*, National Academies Press (US), Washington (DC), 2006, 2, Genomics, Proteomics, and the Changing Research Environment, Available from: https://www.ncbi.nlm.nih.gov/books/NBK19861/.
8. Boundless.com. License: *CC BY-SA: Attribution-ShareAlike*.
9. Oyelade, J., Isewon, I., Oladipupo, F. *et al.*, Clustering Algorithms: Their Application to Gene Expression Data. *Bioinform. Biol. Insights*, 10, 237–253, 2016.
10. Kerr, G., Ruskin, H.J., Crane, M., Doolan, P., Techniques for clustering gene expression data. *Comput. Biol. Med.*, 38, 3, 283–293, 2008.
11. Jain, A.K., Murty, M.N., Flynn, P.J., Data clustering: A review. *ACM Comput. Surv.*, 31, 3, 264–323, 1999.
12. ©Nature Education, CC-BY-NC-SA.

13. Jiang, D., Tang, C., Zhang, A., Cluster analysis for gene expression data: A survey. *IEEE Trans. Knowl. Data Eng.*, 16, 11, 1370–1386, 2004.
14. Chandrasekhar, T., Thangavel, K., Elayaraja, E., Effective clustering algorithms for gene expression data. *Int. J. Comput. Appl.*, 32, 4, 25–9, 2011.
15. Khan, S.S. and Ahmad, A., Cluster Center Initialization Algorithm for K-Means Clustering. 25, 11, 1293–1302, 2004.
16. Handhayani, T. and Hiryanto, L., Intelligent Kernel K-Means for Clustering Gene Expression. *Procedia Comput. Sci.*, 59, 171–7, 2015.
17. Kaufman, L. and Rousseeuw, P.J., *Finding Groups in Data: An Introduction to Cluster Analysis*, vol. 344, John Wiley & Sons, New York, 1990.
18. Sokal, R.R. and Michener, C.D., A statistical method for evaluating systematic relationships. *Univ. Kansas Sci. Bull.*, 28, 1409–38, 1958.
19. Domany, E., Superparamagnetic clustering of data—The definitive solution of an ill-posed problem. *Physica A Stat. Mech. Appl.*, 263, 1, 158–69, 1999.
20. Guha, S., Rastogi, R., Shim, K., CURE: an efficient clustering algorithm for large databases, in: *ACM SIGMOD Record*, vol. 27, New York, NY, pp. 73–84, ACM, USA, 1998.
21. Karypis, G., Han, E.H., Kumar, V., Chameleon: Hierarchical clustering using dynamic modeling. *Computer (Long Beach Calif.)*, 32, 8, 68–75, 1999.
22. Zhang, T., Ramakrishnan, R., Livny, M., *BIRCH: an efficient data clustering method for very large databases*, vol. 25, New York, NY, pp. 103–14, ACM, ACM Sigmod Record, USA, 1996.
23. Grun, B., Model Based Clustering, arXiv:1807.01987v1 [stat.ME], 5 Jul 2018.
24. Kohonen, T., The self-organizing map. *Proc. IEEE*, 78, 9, 1464–80, 1990.
25. Grid-Based Clustering Algorithms, Data Clustering: Theory, Algorithms, and Applications, 209–217, https://doi.org/10.1137/1.9780898718348.ch12.
26. Sander, J., Density-Based Clustering, in: *Encyclopedia of Machine Learning*, C. Sammut and G.I. Webb (Eds.), Springer, Boston, MA, 2011, https://doi.org/10.1007/978-0-387-30164-8.
27. Ester, M., Kriegel, H.-P., Sander, J., Xu, X., A density-based algorithm for discovering clusters in large spatial databases with noise, in: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96)*, AAAI Press, pp. 226–231, 1996.
28. Sharan, R., Elkon, R., Shamir, R., Cluster Analysis and Its Applications to Gene Expression Data, in: *Bioinformatics and Genome Analysis, Ernst Schering Research Foundation Workshop*, vol. 38, Springer, Berlin, Heidelberg, 2002, https://doi.org/10.1007/978-3-662-04747-7_5.
29. Colaprico, A. *et al.*, TCGAbiolinks: an R/Bioconductor package for integrative analysis of TCGA data. *Nucleic Acids Res.*, 44.8, e71–e71, 2015.
30. Silva, T.C. *et al.*, TCGA Workflow: Analyze cancer genomics and epigenomics data using Bioconductor packages. *F1000Research*, 5, 2016, (https://f1000research.com/articles/5-1542/v2).

31. Mounir, M. *et al.*, New functionalities in the TCGAbiolinks package for the study and integration of cancer data from GDC and GTEx. *PLoS Comput. Biol.*, 15, 3, e1006701, 2019, https://doi.org/10.1371/journal.pcbi.1006701.
32. Brazma, A. and Vilo, J., Gene expression data analysis. *FEBS Lett.*, 480, 1, 17–24, 2000.

3

A Critical Review on the Application of Artificial Neural Network in Bioinformatics

Vrs Jhalia[1]* and Tripti Swarnkar[2]

[1]*Department of Computer Science and Engineering, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, India*
[2]*Department of Computer Application, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, India*

Abstract

Proper diagnosis of disease requires deep analysis and classification of disease. In the last two decades the exponential increase in the biological data emerged as an opportunity for many researchers. "Bioinformatics" is the integration of biology and computer science to develop methods and software tools for understanding this large biological dataset. Analyzing the hunks of data by manual means is difficult. Here computer science plays a pivotal role for extracting the hidden patterns. Among all Artificial Intelligence (AI) techniques, Artificial Neural Network (ANN) is considered as the most efficient model in pattern recognition by an automated process. ANN is a type of data structure inspired by networks of biological neurons organized in layers, in which data set is fed to the model, model learns the features from the input and predicts the output. The performance is compared based on classification accuracy. In this chapter we will make a review by comparing the classification accuracy of the ANN model with other existing classification model to get more insights to draw hypothesis.

***Keywords*:** Machine learning (ML), artificial neural network (ANN), prediction model, disease classification

**Corresponding author*: vjhalia@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (51–76) © 2021 Scrivener Publishing LLC

3.1 Introduction

We wear smart watch to measure heart beat rate, hold smart phones that scan fingerprint and retina. All these are possible because of study and research on bioinformatics. Bioinformatics brings biological themes into computers. It is basically the integration of biology with computer science, mathematics and statistics. It analyzes the biological data such as genes, proteins, genomes, cells and deals with the computational management of ecological system, genetic engineering, medical information, drug development. This can be used for prediction, modeling, visualizing and designing.

Bioinformatics is a technique that uses computers to build mathematical model using different bioinformatics tools to acquire, manage and visualize biological data [1]. To infer relationship between components of complex biological system it develops software tools and methods to understand and analyze clinical data and biological data. This requires collecting and interpreting large chunks of biological data so as to gain information about some biological processes. In such situation computational biology is the actual program used to manipulate the data and bioinformatics puts those data to use.

Bioinformatics has a great significance in many areas of biology. In the sphere of genetics, it allows gene prediction, understanding the gene flow, gene sequencing, interpreting genomes and their observed mutations. For image and signal processing, bioinformatics techniques are applied in experimental molecular biology for mining useful information from large amount of raw data. In the text mining bioinformatics tools help to study and organize biological data of many biological literatures, which lead to the expansion of many biological and gene ontologies. It has a significant impact in the study of protein and gene expression and regulation. In system biology bioinformatics tools are used to analyse and catalogue the biological process and networks. It also assists to gain knowledge about the evolutionary characteristics of molecular biology for analysing, comparing and interpreting genetic and genomic data [2]. Bioinformatics helps to predict the 3D structure of different macromolecules such as RNA, DNA and proteins. It also assists in the analysis and simulation of different biomolecular interactions that are an important part of structural biology [3].

3.1.1 Different Areas of Application of Bioinformatics

Figure 3.1 represents different fields in which bioinformatics has a great range of application. It also describes various functions of bioinformatics in each of the field.

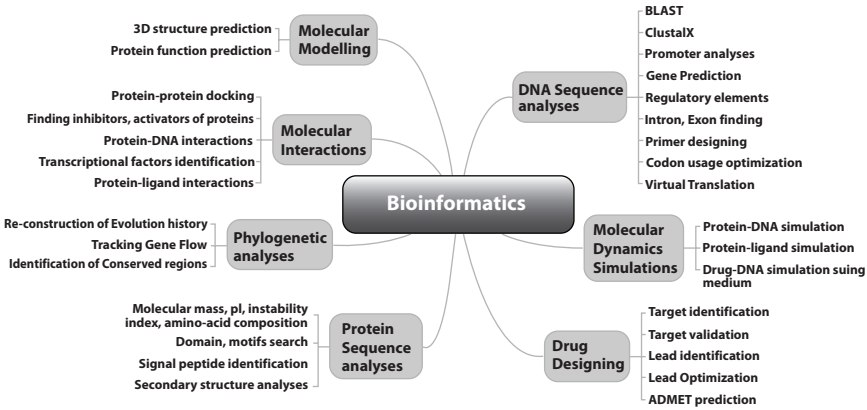


Figure 3.1 Areas of research of bioinformatics [4].

3.1.2 Bioinformatics in Real World

Health Sector

- Rapid progression of various diseases creating more scope for gene analysis to detect and classify the disease and to improve medical diagnosis.
- *Bioinformatics* tools are used to study genes or proteins that are associated in the pathogen and pathogenesis of the *disease.*
- Today by combining the clinical data with genetic data survival risk of patients are predicted.

Drug Development

- Bioinformatics has a visible impact in drug discovery and drug development. Acquiring deep knowledge on association of gene with disease using computational tools, new disease specific drugs are discovered and validated rapidly [4].
- Bioinformatics tools are also very powerful and effective in analysis, prediction and interpretation of clinical and preclinical findings.
- Procuring more understanding by conducting research in this field will assist to develop drugs that attack the cause not merely the symptoms.

Molecular Medicine

- With the study of genetic mechanisms we can hunt for genes that are directly related with different diseases and resulting

knowledge will enable better treatment, speed up development of drugs and diagnostic tools to measure a person's vulnerability to harmful diseases.

- With the application of bioinformatics tools we can identify the molecular and genetic error associated with different disease. We can also understand the molecular structures and develop molecular interventions to correct them.

Preventative Medicine

- Study of human genes with use of bioinformatics tools allow us to find the cause of a disease.
- Development of preventive medicine will help avert and avoid the disease before it causes a disease.

Gene Therapy

- Gene therapy is the approach to deal with gene expression of a person. This method is used to alter the gene expression of an individual to treat, cure and prevent disease.

Waste Clean-Up

- With the use of bioinformatics tools Bacterium like *Deinococcus radiodurans* is identified which help cleaning the waste [4, 5].
- These organisms are efficient in cleaning up toxic chemicals and harmful radiations of waste sites.

Study of Climate Change

- Excessive use of fossil fuels for obtaining energy contributes to high emission of harmful carbon dioxide gas, which is a major cause of global climate change.
- The study of different microorganisms that uses carbon dioxide as their leading source will help to reduce atmospheric carbon dioxide levels.

Biotechnology

- In bioinformatics, biotechnology is used to identify organisms and micro-organisms which can be useful in dairy industries and food manufacturing companies. For example micro-organisms like *Lactococcus lactis* involved in dairy industry for the manufacture of buttermilk, cheese, yogurt, etc.

Crop Improvement

- Study of bioinformatics involves study of DNA, RNA sequence, prediction of function and structure of protein of plant genomes.
- Genetic Knowledge of plants has shown the organisation of genes of plants and this knowledge is used for producing improved insect resistant crops and makes plants more productive and the protein model helps to improve genes of plants.

Insect Resistance

- Soil-borne bacteria like *Bacillus thuringiensis* makes proteins that are toxic to some insects.
- Genes of these soil-borne bacteria have been studied and successfully transferred to cotton, potatoes and maize to control many serious pests [4, 5].
- These bacteria facilitate to repel insect attack so the practice of using insecticides in plants can be reduced with the study of the protein produced by them and hence the nutritional content of the plants can be improved.

Development of Drought Resistance Varieties

- Genetic knowledge of plants helps to develop varieties of crops with a great tolerance of soil alkalinity, iron toxicities and have the capability to grow in reduced water condition. This also allows crop development in substandard soil regions to create more agricultural land and to increase crop production [7].

Comparative Studies

- To understand the functions of genes, inherited diseases mechanisms and evolution of species we need to analyze and compare the genetic substance of different species.
- Bioinformatics tools are also applied to make comparisons between the numbers, locations and biochemical functions of genes in different organisms [5, 6].

There are a wide range of applications of bioinformatics in the domain of diagnosis, medicine, agriculture, biotechnology. Studying and using different tools of bioinformatics will allow researchers to extend knowledge

far more efficiently and effectively through data analysis and experiments. This will fasten the major discoveries more accurately.

3.1.3 Issues with Bioinformatics

Section 3.1.1 discusses different applications of bioinformatics. These applications come with many challenges when it is associated with some issues related with the data or the devices used for collection or analysis of it. So addressing and analysing these issues are required for proper execution and effective result. This subsections below discusses different issues that are faced when the biological study is conducted.

3.1.3.1 Issues Related to Structure

Study of DNA and protein includes problems like protein structure prediction as they are represented in 3D data, so structure prediction, alignment and analysis become a difficult task. The prediction of protein three-dimensional structure from sequence can be solved with the application of ANN.

Most of the biological networks such as protein–protein interaction networks, gene regulatory network, etc. are difficult to interpret and build due to the complexity of biological system. So using graph-theoretic methods these massive range of networks are displayed in graphs which makes classification very difficult using traditional methods.

3.1.3.2 Sequence Analysis

Classification of RNA, Protein Sequence and DNA become a challenge because of difference and similarity of many organisms.

Issue with Genome Sequence
A Genome denotes to the complete set of chromosomes of an organism consisting of DNA**.** Genome sequencing, is a way of mapping out DNA or ordering DNA for organizing, processing and interpreting the sequences, which again requires improvements in sequencing strategies. Each sequencing of DNA faces challenges in searching the sequence pattern, designing, analyzing and interpreting the data.

- In gene findings and genome annotation: Gene finding suggests for prediction of nucleotide sequence such as introns

and exons in DNA-sequence segments, whereas genome annotation is a process of gene sequencing to find out the gene coding regions to analyze protein sequence [8]. It involves study of the repetitive DNA within the genome, emulated from either same or nearly same sequence.

- In sequence comparison: Sequence comparison is the process of comparing two or more than two sequences. Availability of large amount of sequences in genomic database requires proper categorization of DNA and protein sequence. So sequence comparison helps assigning a hypothetical structure and function to a sequence for identification, design and interpretation of sequence [8].

Analysis of sequencing or DNA sequencing is an important task because it helps to detect individual genes that are associated with a disease. When a disease affects an individual, its protein or genes get altered, that causes gene sequence alteration. So it becomes very important to detect these genes to find the cure of the disease. Traditional methods of gene detection were based on trial and error method. Now the advancement in Data mining and machine learning like Neural Network (NN) allow more precise study of genes and its sequence to simplify the task [9]. Many machine learning algorithms are used to classify the normal and abnormal genes with a great accuracy.

Solution to above problems involves following steps

- Collection of Biological Data
- Building Computational model
- Analyze and solve problems of computational model
- Test the computation algorithm
- Evaluate the performance of the model.

3.2 Biological Datasets

Bioinformatics deals with various biological datasets being collected at different levels of omics data such as

- Genomic Sequence data
- Protein Sequence data

- Microarray data
- Structure data (Structure of RNA and protein)
- Chemical data
- Disease data.

Based on the type of data Biological database can be divided in to two categories:

a. Primary Database
   These kinds of databases are archival in nature because these databases are created by the experimental results submitted directly by researchers. These databases are populated with protein sequence, nucleotide sequence or macromolecular structure etc. [10].
   Example: Protein Data Bank (PDB), GenBank, DNA Data Bank of Japan (DDBJ), Gene Expression Omnibus (GEO).
b. Secondary Database
   These databases are either manually created or extracted from result analysis of primary database to create more structured records for easy retrieval of data [10].
   Example: Swiss-port (it is protein sequence database maintained by Swiss Institute of Bioinformatics, Switzerland and the European Bioinformatics Institute, UniProt Knowledgebase.

3.3 Building Computational Model

Building Computational model includes study of different behavior of complex system to get some new insights for deepening our understanding. In this section we will discuss some prerequisites which are required for building the computational model.

3.3.1 Data Pre-Processing and its Necessity

After collecting data from database it goes through several processes because data present in the databases are often raw, noisy, incomplete or inconsistent due to these reasons data cannot be used directly for mining process because it may produce unsatisfactory mining result. In order to enhance the classification result, a pre-processing step is initiated as an essential step before mining the data. It usually includes following methods such as data cleaning, data integration, data transformation,

dimensionality reduction and so on [11]. Data pre-processing technique significantly improves the quality of data, performance of the classification model and minimizes the time required for actual mining.

We will address some of the problems which need to be solved to achieve better classification result. It involves cleaning of noisy data, missing data, duplicate data, etc. from the database to smoothly conduct the classification process. Noisy data refers to the unnecessary information available in the dataset which is meaningless and cannot be interpreted by machines. It can also be called corrupt data. Presence of these data can affect the data preparation process. To smooth the noisy data binning, clustering, regression methods can be used or can simply be deleted from large datasets based on the amount of noise present [11, 12]. Another biggest problem in biological data is the absence of values. In complex biological datasets this issue greatly impacts the performance of accuracy of the model. So to handle missing values various imputation techniques have to be used [12]. Data duplication are ongoing data quality problem testified in diverse domains, including health care, business and molecular biology, etc. [13] Presences of duplicate data leads to data inconsistency and redundancy which produces several consequences in classification problem. This issue can be handled by detecting and eliminating duplicate values.

Biological data may contain thousands of features because they are highly dependent on comparing the behavior of various biological units. These data often contains large amount of irrelevant data which affect the classification accuracy and machine learning efficiency [14]. Dimensionality reduction technique focuses on reducing the number of input features which aids to reduce computation time and redundant data. Dimensionality can be reduced using different method such as Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), feature selection, etc.

3.3.2 Biological Data Classification

Databases are the rich source of hidden information which can be extracted for intelligent decision making. Classification is a process by which data is organized into different categories by determining a class for an element in database. Data is grouped into different classes based on the training dataset. The classification process helps analyze data and to build models that define data classes and predict future trends.

The availability of large amount of microarray data has created new scopes in classification methods. Like classification of DNA microarrays contributes significantly for diagnosis and prognosis in many clinical practices [15].

Also the classification of gene expression data addresses the fundamental problem of many diseases. Classification of tumor types has been achieving a great success in cancer diagnosis and precision drug development. However, earlier cancer classification studies were clinical based and they have inadequate analytical expertise. Today thousands of gene expressions data is simultaneously monitored because of the advancement in classification algorithms.

3.3.3 ML in Bioinformatics

Machine learning (ML) is a technique to develop computer program to access data and to learn knowledge automatically from experience without human interventions and assistance. Machine learning techniques and deep learning algorithms enable the classifier to make use of automatic feature learning technique, to make reasonably complex predictions when the model is being trained on large datasets. It means the algorithm is able to learn based on the dataset alone and can discover ways to integrate numerous features of the input information into one intellectual set of features from which further learning can be done [9]. Machine learning technique uses two different methods to train the model: supervised learning and unsupervised learning method [16]. In subsequent sections we will discuss machine learning models used for supervised learning problems.

In recent years, availability of biological datasets have risen abruptly, this has enabled bioinformatics researchers to make use of these machine learning algorithms. ML techniques has been applied to many biological domains such as Microarrays, Systems biology, Genomics, Proteomics, Stroke diagnosis and Text mining, etc. [8].

- Machine learning in bioinformatics helps to explore, analyze, manage and store data to extract relevant information from biological data.
- Gene identification and nucleotide identification help to understand gene and gene association with disease.
- Machine learning tools are also used to determine genomic sequence and examine gene pattern.
- Gene sequence classification allows us to grasp the concept of nucleic acid and protein sequence.

Advancement of this field can ultimately leads to the development of automated diagnostic tools, personalized and precision medicine, gene therapy, food analysis, biodiversity management and many more which will

target Individual's lifestyle, environment and custom medical treatments considering person's vulnerability to disease.

There are many machine learning techniques among which Artificial Neural Network is an effective technique for the identification, selection, classification and prediction of the gene in the DNA Sequences.

3.3.4 Introduction to ANN

ANN is a computing system that resembles human brain, consisting of highly interconnected network of processing units called neurons. It has the ability to handle complex features within data in order to process the information.

Figure 3.2 represents the simple work flow of artificial neural network with a perceptron architecture, which consists of one input layer with few input units representing multiple features present in the database, depending on the objective being defined. Each inputs collected from the dataset are multiplied with weights and fed to a function called activation function to produce the actual output.

Depending on the difference between desired output and actual output weights are modified in each layer of connection and finally output is predicted. Weights are machine learnt values from neural networks. Neural network learns by a feedback process called backpropagation. In backpropagation, weights are modified from the output layer to the input layer, going backward through hidden layer to lessen the difference between actual output and desired output to a certain point [18]. Therefore, backpropagation algorithm helps to reduce the overall computational loss of the network while learning the network. The number of hidden layers used for computation and the type of computation being done in each layer together determine the processing speed of the network.

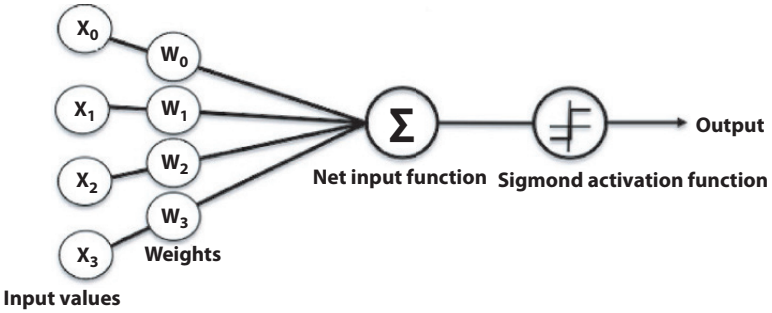


Figure 3.2 Simple network architecture of ANN with four input unit [17].

Neural networks are classified into different types based on the structure, neuron density, number of layers, etc. Most commonly used ANNs generally follow the three-layer architecture having an input layer, one or more hidden layers and an output layer as we have seen in Figure 3.2. The number of neurons to be used in each layer of the network depends on the number of features available in a dataset and the complexity of the problem [19].

We will discuss some of ANN model that are widely used as supervised learning models.

- Perceptron Network: Perceptron model is a binary classifier that means it separates input data in to two categories. It is the simplest and oldest model of neural network which can implement linearly separable problems such as AND, OR, NOT gate but does not work for non-linear problems like XOR gate. So to deal with non-linear problems or more complex problems we utilize multilayer perceptron. Left side of Figure 3.3 shows a simple architecture of perceptron model consisting of an input layer and an output layer.
- Multilayer Perceptron (MLP) Network: MLP supports multi-class classification. It consists of at least one hidden layer along with an input layer and an output layer. In this network every single node is connected to all other nodes in the next layer by connecting weights to develop a fully connected neural network. It may involve one or more than

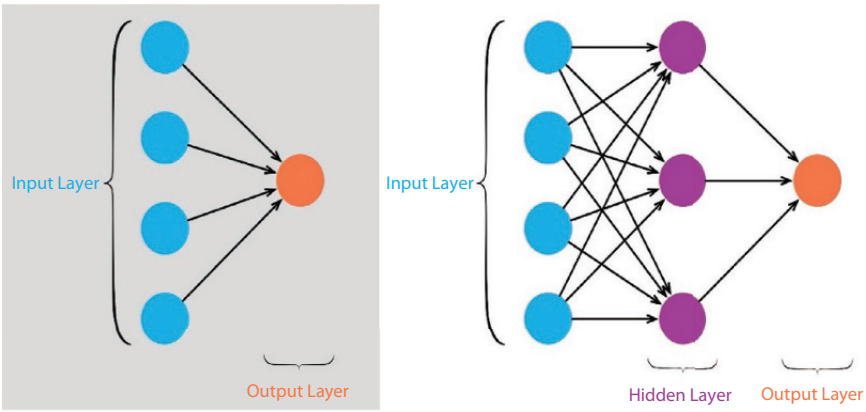


Figure 3.3 Single layer perceptron (left) and multilayer perceptron with one hidden layer (right) [20].

one hidden layer in the network for complex problem classification. MLP implements non-linear activation function for predicting the output units. It has a unique ability to propagate in both the direction i.e. forward propagation and backward propagation [18]. Right of Figure 3.3 shows the architecture of MLP with an input layer, one hidden layer and an output layer.

Backpropagation: It is a supervised learning algorithm implemented in MLP network that helps to change the weight to minimize the calculated error. It traverses backward from output layer to input layer through hidden layers.

Unsupervised learning occurs when the target output is unknown. In this kind of learning network supervises how to group the output result based on the given input data. So this is also defined as self-organization. Some of the applications of unsupervised learning include image processing, speech recognition, text mining, etc. There are some well-established form of neural network based unsupervised learning algorithms available, such as principal components analysis, Kohonen's self-organizing maps, independent components analysis, Hebbian learning, etc. [9]. Though unsupervised learning algorithms have a proven track in many areas, unwrapping their application for a comprehensive review is beyond the scope of our discussion. This chapter will only focus on the application of supervised neural networks. To get more insights on unsupervised learning in proteomics and genomics you can refer [21].

3.3.5 Application of ANN in Bioinformatics

Artificial neural network has been used in many areas of bioinformatics and has proved to be one of the most powerful tools in solving many bioinformatics problems. Some of the areas of bioinformatics where ANN is applied are listed below and discussed in detail in further sections.

1. In DNA, RNA alignment
2. Image and signal processing
3. In the problems of genes identification
4. In the coding region recognition of genes
5. features detection, classification and Sequencing
6. In signal Identification and analysis generated from regulatory sites
7. Protein structure prediction from different sequences

8. Expression of genetic and genomic data
9. In monitoring the treatment of patients based on DNA sequences.

3.3.6 Broadly Used Supervised Machine Learning Techniques

Apart from ANN there are many supervised machine learning algorithms such as Support Vector Machine (SVM) [22], Logistic Regression [23], Decision tree [24], K-nearest neighbors (KNN) [25] and Random Forest [26] which are widely used in the field of bioinformatics and obtaining a high classification accuracy. These models with most popular Artificial Neural Network architecture being used in the literature are further discussed.

3.4 Literature Review

Over the years, artificial neural network has been widely used in gene expression data processing due to its ability to identify the complicated relationships between different attributes in the large data sets. ANN has achieved great success due to their potential to manage the complexity and nonlinearity of biology datasets. Gene expression analysis have aimed for defining more specific biological aspects to enhance patient risk stratification and to guarantee the highest benefits and least toxicity from a specific treatment.

Wisconsin Prognostic Breast Cancer (WPBC) dataset was collected by Samundeeswari *et al.* [27] to perform an experiment using ANN model. ANN was used to handle the predicting status of patients at a particular endpoint and to predict the time of disease occurrence. Dataset consisted of 35 features and 194 instances. Feedforward neural network model was used, with two hidden layers and 20 neurons in each layer and the entire experiment was carried out in Matlab environment. Model was trained with backpropagation technique and the sigmoid activation function was used for hidden and output nodes. In this research Neural Network proved to perform remarkably with 96% specificity and 97.68% accuracy.

An ANN model was used by Narayanan *et al.* [28] to identify positive and negative genes related to cancer from a large dataset. A dataset of 74 patients with 7,129 gene expressions was collected. Out of 74, 31samples were normal bone marrow cases and rest of the patients were diagnosed with multiple myeloma. Different experiments were carried out using single layer ANN model. At the end the authors concluded that the requirement of hidden layers in a network is dependent on the complexity of the

gene expression dataset. For gene expression analysis single layer neural network was very useful for the sake of simplicity and generalizability and could solve many complex problems with a suitable architectural modification.

Hu *et al.* [29] designed a classification model to classify bladder cancer cell for six different tumor classes using 467 images. Using both supervised and unsupervised learning methods accuracy of the model was estimated. In supervised learning, MLP of one hidden layer along with backpropagation algorithm was applied to classify and mitigate the error rate. In unsupervised learning, fuzzy and non-fuzzy c-means clustering methods were implemented. Different activation functions such as Gaussian, sigmoid and sinusoid were studied for different network configurations. Using all the available data, and 5 different features neural network classifier was able to capture the information about cancer cells and obtain 96.9% classification rate whereas fuzzy c-means obtained only 76.50%.

In the year 2003, Won *et al.* [30] classified leukemia dataset consisting of 72 samples with two different classes such as acute lymphoblastic leukemia and acute myeloid leukemia. Each sample had 7,129 gene expression levels representing the input for the model. Model was trained using 38 samples and rest of the samples were used for testing. Researchers used a 3-layered MLP for data classification with 8 hidden nodes and 2 output nodes. Result showed that ANN outperformed with an accuracy of 97%.

Thein *et al.* [31] used the breast cancer medical dataset with 699 instances and 10 attributes with one class attribute. The dataset was made available by university of Wisconsin hospital, Madison. Attributes 1 to 9 were used to represent features to be used in the model. Each instance belonged to one of two possible classes: Benign or Malignant. According to the class distribution 458 were Benign and 241 instances were Malignant. The dataset was classified using multilayer neural network (MLP) with backpropagation technique and achieved an accuracy of 99.97%. Authors finally depicted that ANN has the greatest tolerance of noisy data and a great ability to classify the untrained data pattern.

Peterson *et al.* [32] analyzed DNA microarray cancer data set by comparing different machine learning algorithms such as ANN, logistic regression, linear discriminant analysis, SVM and k-nearest neighbor for survival analysis of patients. One of the main findings here was that ANN is dependent on the statistical significance of the features so despite large sample size, ANN outperformed all other classifiers, in achieving greatest area under the curve.

Soto *et al.* in 2020 [33] dealt with 11_Tumors database, a well-recognized database of gene microarrays related to cancer disease to

generate the likelihood of types of cancer. This database had 12,533 gene expression microarrays for 174 samples and 11 different categories of cancer. Since the dataset had large number of features so the dimensionality reduction technique, PCA was used to reduce the number of features from 12,533 to 113. Classification was done using softsign activation function in the multilayer feedforward network (MLP) consisting of three hidden layers with 100 neurons in each layer. Using sigmoid activation function an output layer of 11 neurons was generated. Upon using 10-fold cross validation classification model was evaluated and achieved an accuracy of 97.14%.

Wei *et al.* [34] collected 56 cDNA microarray tumor samples form 49 neuroblastoma patients to predict the survival rate of high risk patients. To remove poor quality data principle component analysis algorithm was used and total 42,578 features were reduced to 37,920 data points. Each sample was analysed using a powerful ANN based predictor model. Despite this high complexity, 88% accuracy was seen to be achieved. An ANN-based gene minimization strategy had also been implemented by the author in a separate analysis of 19 genes. In this analytical process high risk patients were further divided in to subgroups based on their survival status. This derived subset of 19 genes correctly classified 98% of the patients. Finally they concluded that ANN-based approach has a significant ability in prediction of survival rate. This would allow personalized therapies for patients according to their gene expression profiles.

Cangelosi *et al.* [35] developed a robust classification model using ANN to predict neuroblastoma patient's outcome with minimized error rate by defining a gene expression signature (NB-hypo), which measures the hypoxic status of 100 neuroblastoma tumor gene expression profiles. The ANN-MLP was applied to build a hypoxia predictor of 62 probe sets of the signature with potential clinical application to evaluate the hostile effect of tumor hypoxia on the progression of the disease. The result showed that MLP achieved a similar or better performance when compared to SVM, Naïve Bayes and logistic regression model. ANN proved itself to be a better competitive tool for predicting 'poor' or 'good' outcome of a patient while making an analysis of complex gene expression data.

Nayeem *et al.* [36] designed a classifier using 3 different datasets collected from UCI Machine Learning Repository for the diagnosis of heart disease, liver disorder and lung cancer. The proposed network showed an accuracy above 80% for each kind of dataset. MLP along with gradient descent optimization algorithm was used to minimize the error and Levenberg–Marquardt algorithm was used to avoid curve fitting problem.

The feedforward–backpropagation algorithm achieved a good performance even when the number of features was more. Authors depicted that the proposed model was able to show high accuracy than exiting models because of the use of more number of samples for training data, more than one hidden layer and large number of neurons in each hidden layer.

Bordoloi *et al.* [37] compared amino acid sequence of a particular protein structure to predict secondary structure of protein and for this work authors used a fully connected MLP with backpropagation algorithm with only one hidden layer. Here amino acid structures were the input to the network and predicted structures representing four different classes such as hemoglobin, myoglobin, sickle cell anemia and insulin were the output. The network was trained using back propagation algorithm to update weights and for evaluating the error. To minimize the error rate produced by the model, Levenberg–Marquardt optimization technique was implemented. The ANN model showed 100% accuracy in successfully identifying the required parameters. It was seen that when the number of epochs to train the model was increased, model produced high accuracy rate however, training time of the network also increased slightly. It was also observed that, since the dataset was large ANN model produced some computational constraints while training the model.

Shanthi *et al.* [38] used a feature selection method for stroke disease classification using multilayer perceptron. Research was carried out using the dataset having 20 attributes of 150 patients with symptoms of stroke disease. To reduce the number of features from 20 to 14 genetic algorithm was used in a neuro-genetic approach. This new feature set of symptoms were given to the model for prediction of type of stroke disease. The model was trained using backpropagation algorithm with 7 hidden neurons and sigmoid activation function. The result showed that this neuro-genetic approach obtained a better accuracy than traditional ANN with reduced number of features.

3.4.1 Comparative Analysis of ANN With Broadly Used Traditional ML Algorithms

Table 3.1 shows recent studies conducted in the field of bioinformatics for classification, prediction, and feature selection using different machine learning techniques. Most of the studies have used datasets like genomic, clinical, molecular, etc. From the large set of publications we have filtered out some of the recent and relevant papers for our research to make a comparative analysis between ANN and other machine learning models.

Table 3.1 Shows published articles of ANN used for biological data.

Referred articles	Dataset used	Model used	Purpose	Findings
Rahman *et al.* [39]	Ovarian Cancer Data	• ANN • SVM • KNN	Feature Selection with Classification	ANN achieved a 98.7% accuracy, higher than other two models being compared.
Ghani *et al.* [40]	Breast Cancer Data	• ANN • Decision Tree • Naïve Bayes • KNN	Feature Selection with Classification	ANN-MLP best classify the attribute in comparison to other three models with an accuracy of 80%.
Song *et al.* [41]	Cardiac patient record	• Logistic Regression • ANN-MLP • ANN-SLP (ANN-Single layer perceptron) • LS-SVM (Least Square SVM • LSS (Least Square Separation)	Risk Prediction	ANN-MLP performed better than single layer ANN, as well as, other compared models.

(*Continued*)

Table 3.1 Shows published articles of ANN used for biological data. (*Continued*)

Referred articles	Dataset used	Model used	Purpose	Findings
Tiwari et al. [42]	Microarray Cancer Dataset	• KNN • SVM • ANN-MLP • CNN • Logistic Regression	Outcome Prediction	With the application of deep learning algorithm accuracy of ANN maximized to 99.3%.
Kaur *et al.* [43]	• Cervical cancer • Breast cancer	• Decision tree • SVM • KNN • Logistic regression • ANN • Ensemble	Feature Selection with Classification	ANN and Ensemble classifier outperformed other classifiers for Breast cancer and Cervical dataset respectively.
Román *et al.* [44]	Dengue Patient Dataset	• ANN-MLP • SVM polynomial • (3 kernel function used in SVM)	Classification with missing value imputation	ANN-MLP attained 96% accuracy when compared to SVM Classifier.

(*Continued*)

Table 3.1 Shows published articles of ANN used for biological data. (*Continued*)

Referred articles	Dataset used	Model used	Purpose	Findings
Subhadra *et al.* [45]	Heart disease Data (UCI repository)	• ANN-MLP • Logistic Regression • Random Forest • SVM • Decision Tree • Naïve Bayes	Disease Prediction	MLP with backpropagation algorithm procured an accuracy 94%, which is higher in comparison to all other classifiers being used in the paper.
Dwivedi [46]	Microarray gene expression	• ANN-MLP • KNN • Logistic regression • Classification tree • Naïve Bayes • SVM	Cancer Classification	ANN-MLP procured 98% accuracy with negligible error rate when compared to other five classifiers.
Sethi *et al.* [47]	• Yeast • Pima Indian Diabetes	• MLP + PCA • PSO-MLP (Particle swarm optimization-MLP) + PCA	Feature Selection	PSO-MLP + PCA obtained high accuracy when compared with MLP + PCA with a small increase in computational cost.

(*Continued*)

Table 3.1 Shows published articles of ANN used for biological data. (*Continued*)

Referred articles	Dataset used	Model used	Purpose	Findings
Padmavati *et al.* [48]	Breast Cancer	• RBFNN (Radial Basis Function Neural Network) • MLP • Logistic regression	Prediction	RBFNN yielded an accuracy more than 90% in less time when compared with MLP and logistic regression.
Pellakuri *et al.* [49]	Medicinal Dataset	• MLP back propagation • Adaline Network	Classification	Adaline neural network is a simple, easy and fastest method that gives effective result for decision making.
Sonawane *et al.* [50]	Cleveland Heart Disease (UCI repository)	• Improved-MLP • BPN • DSS (Decision Support System)	Prediction	BPNN obtained 98% accuracy when number of neurons in the hidden layer increased to 20 and epochs increased to 1,000.

3.5 Critical Analysis

In this study we observed that ANN classifier outperformed all other classifiers with a reasonable accuracy result. Here we discuss our critical observation on various factors that improve the performance of ANN model in achieving high accuracy. ANN learns to solve complex problems because of their tremendous parallel processing, adaptive learning, fault tolerance and self-organization capability which ensure high classification performance. ANN has been the most powerful tool in classification and prediction. The performance of ANN algorithm depends on various factors such as pre-processing of data, (dimension of dataset, availability of incomplete data or noisy data, selection of features) activation function to be used, selection of number of epochs and neurons. Selecting huge amount of features or wrong features could affect the performance of model. The performance of ANN also depends on selection of right combination of input variables and other parameters.

In case of SVM model it is seen that when number of features exceeds number of samples the model tends to perform slow so more work on feature selection is required. But, when substantial amount of DNA sequencing data is present for two-class disease classification we can say SVM is a great classification model. We also observed that when inputs are noisy or incomplete, neural networks are still able to produce reasonable result. So the correct use of Data pre-processing technique could improve the performance of the classification model.

Another factor that influences the performance of the model is the right choice of activation function for classifying linear and non-linear data. One of the fastest learning activation functions is ReLU function that gives more accurate result because it is easy to optimize with gradient descent and result in global optimal solution.

It is observed that when number of hidden layers increases, the model gives relatively high accuracy. So we can say that performance of the model also depends on number of hidden layers and number of neurons used in each hidden layer. As we found in many articles that performance of the model hardly improves when less hidden layers are used in the network. It also increases the risk of convergence to local minima. The network fits the training data poorly if number of neurons used in the network is very less as a result the network becomes too simple and may not be able to model complex data. Similarly, if the model is trained using too many neurons, it may take excessively long time for training the input data and there is a high chance of overfitting of data. In such case the network may begin to

model some random noise in the output, as a result the model generalizes the training data extremely well, but fails to generalize new or future data. Integration of ANN algorithm with different optimization algorithms minimizes the error rate produced by the classification model which in result improves the model performance.

3.6 Conclusion

In this paper we have discussed the different applications of ANN related to different fields of bioinformatics. We have also made a comparative study between various machine learning algorithms and ANN algorithm to get some useful variants about how ANN works and what affects the performance of an ANN classification model and how the performance of the model can be improved to get more accurate result. The problems associated with the traditional approach to solve classification problem can be overcome with the concept of deep learning, which will allow faster learning by reducing the computational cost of the classification model even for the large dataset with inbuilt feature engineering that reduces the requirement of domain expertise. The observation from the study shows that the ANN and its variations can be used to solve complex problem of disease diagnostics or prognosis related with bioinformatics, resulting in the improved lifestyle and environment.

References

1. https://en.wikipedia.org/wiki/Bioinformatics.
2. https://microbenotes.com/bioinformatics-introduction-and-applications/.
3. https://en.wikipedia.org/wiki/Structural_biology.
4. Mehmood, M.A., Sehar, U., Ahmad, N., Use of Bioinformatics Tools in Different Spheres of Life Sciences. *J. Data Mining Genomics Proteomics*, 5, 1–13, 2014.
5. Singh, H., Bioinformatics: Benefits to Mankind. *Int. J. Pharm. Tech. Res.,* 9, 4, 242–248, 2016.
6. https://www.biotecharticles.com/Bioinformatics-Article/Applications-of-Bioinformatics-3270.html.
7. Rhee, S.Y., Dickerson, J., Xu, D., Bioinformatics and its applications in plant biology. *Annu. Rev. Plant Biol.*, 57, 335–360, 2006.
8. https://www.analyticssteps.com/blogs/understanding-bioinformatics-application-machine-learning.

9. Lancashire, L.J., Lemetre, C., Ball, G.R., An introduction to artificial neural networks in bioinformatics—Application to complex microarray and mass spectrometry datasets in cancer studies. *Briefings Bioinf.*, 10, 3, 315–329, 2009.
10. https://microbenotes.com/biological-databases-types-and-importance/.
11. https://ceoworld.biz/2019/12/13/data-preprocessing-what-is-it-and-why-is-important/.
12. Herbert, K.G. and Wang, J.T.L., Biological data cleaning: A case study. *Int. J. Inf. Qual.*, 1, 1, 60–82, 2007.
13. Chen, Q., *Duplication in biological databases: definitions, impacts and methods* (Doctoral dissertation), University of Melbourne, Australia, 2017.
14. Lee, G., Rodriguez, C., Madabhushi, A., An empirical comparison of dimensionality reduction methods for classifying gene and protein expression datasets, in: *International Symposium on Bioinformatics Research and Applications*, Springer, Berlin, Heidelberg, pp. 170–181, 2007.
15. Lausen, B., Bioinformatics and classification: The analysis of genome expression data, in: *Classification, Clustering, and Data Analysis*, pp. 455–461, Springer, Berlin, Heidelberg, 2002.
16. https://towardsdatascience.com/explore-the-world-of-bioinformatics-with-machine-learning-47c62c482aaf.
17. https://www.kdnuggets.com/2018/10/simple-neural-network-python.html.
18. https://www.tutorialspoint.com/artificial_neural_network/artificial_neural_network_supervised_learning.html.
19. https://www.mygreatlearning.com/blog/types-of-neural-networks/.
20. https://towardsdatascience.com/multi-layer-neural-networks-with-sigmoid-function-deep-learning-for-rookies-2-bf464f09eb7f.
21. Boutros, P.C. and Okey, A.B., Unsupervised pattern recognition: An introduction to the whys and wherefores of clustering microarray data. *Briefings Bioinf.*, 6, 4, 331–343, 2005.
22. Wang, J.-Y., *Application of support vector machines in bioinformatics*, National Taiwan University, Taiwan, 2002.
23. https://towardsdatascience.com/logistic-regression-classifier-8583e0c3cf9.
24. Polaka, I., Tom, I., Borisov, A., Decision tree classifiers in bioinformatics. *Appl. Comput. Syst.*, 42, 1, 118–123, 2010.
25. https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761.
26. Qi, Y., Random forest for bioinformatics, in: *Ensemble machine learning*, pp. 307–323, Springer, Boston, MA, 2012.
27. Samundeeswari, E. and Saranya, P., An artificial neural network model for prediction of survival time of breast cancer dataset. *Int. J. Res. Eng. Appl. Sci.*, 6, 1, 161–168, 2016.
28. Narayanan, A., Keedwell, E.C., Gamalielsson, J., Tatineni, S., Single-layer artificial neural networks for gene expression analysis. *Neurocomputing*, 61, 217–240, 2004.

29. Hu, Y., Ashenayi, K., Veltri, R., O'Dowd, G., Miller, G., Hurst, R., Bonner, R., A comparison of neural network and fuzzy c-means methods in bladder cancer cell classification, in: *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, vol. 6, IEEE, pp. 3461–3466, 1994.
30. Won, H.-H. and Cho, S.-B., Paired neural network with negatively correlated features for cancer classification in DNA gene expression profiles, in: *Proceedings of the International Joint Conference on Neural Networks, 2003*, vol. 3, pp. 1708–1713, IEEE, Portland, Oregon, 2003.
31. Thein, H.T.T. and Tun, K.M.M., An approach for breast cancer diagnosis classification using neural network. *Adv. Comput.*, 6, 1, 1, 2015.
32. Peterson, L.E. and Coleman, M.A., Machine learning-based receiver operating characteristic (ROC) curves for crisp and fuzzy classification of DNA microarrays in cancer research. *Int. J. Approximate Reasoning*, 47, 1, 17–36, 2008.
33. Tabares-Soto, R., Orozco-Arias, S., Romero-Cano, V., Segovia Bucheli, V., Rodríguez-Sotelo, J.L., Jiménez-Varón, C.F., A comparative study of machine learning and deep learning algorithms to classify cancer types based on microarray gene expression data. *PeerJ Comput. Sci.*, 6, e270, 2020.
34. Wei, J.S., Greer, B.T., Westermann, F., Steinberg, S.M., Son, C.-G., Chen, Q.-R., Whiteford, C.C. *et al.*, Prediction of clinical outcome using gene expression profiling and artificial neural networks for patients with neuroblastoma. *Cancer Res.*, 64, 19, 6883–6891, 2004.
35. Cangelosi, D., Pelassa, S., Morini, M., Conte, M., Bosco, M.C., Eva, A., Sementa, A.R., Varesio, L., Artificial neural network classifier predicts neuroblastoma patients' outcome. *BMC Bioinf.*, 17, 12, 347, 2016.
36. Nayeem, Md O. G., Ning Wan, M., Hasan, Md K., Prediction of disease level using multilayer perceptron of artificial neural network for patient monitoring. *Int. J. Soft Comput. Eng. (IJSCE)*, 5, 7–23, 2015.
37. Bordoloi, H. and Sarma, K.K., Protein structure prediction using multiple artificial neural network classifier, in: *Soft Computing Techniques in Vision Science*, pp. 137–146, Springer, Berlin, Heidelberg, 2012.
38. Shanthi, D., Sahoo, G., Saravanan, N., Input Feature Selection using Hybrid Neuro-Genetic Approach in the diagnosis of Stroke. *Int. J. Comput. Sci. Netw. Secur.,* 8, 12, 2008.
39. Rahman, Md A., Chandren Muniyandi, R., Islam, Kh T., Rahman, Md M., Ovarian Cancer Classification Accuracy Analysis Using 15-Neuron Artificial Neural Networks Model, in: *2019 IEEE Student Conference on Research and Development (SCOReD)*, IEEE, pp. 33–38, 2019.
40. Ghani, M.U., Alam, T.M., Jaskani, F.H., Comparison of Classification Models for Early Prediction of Breast Cancer, in: *2019 International Conference on Innovative Computing (ICIC)*, IEEE, pp. 1–6, 2019.
41. Song, X., Mitnitski, A., Cox, J., Rockwood, K., Comparison of machine learning techniques with classical statistical models in predicting health outcomes, in: *Medinfo*, pp. 736–740, 2004.

42. Tiwari, M., Bharuka, R., Shah, P., Lokare, R., Breast Cancer Prediction Using Deep Learning and Machine Learning Techniques, Available at SSRN 3558786, 2020.
43. Kaur, R. and Ginige, J.A., Comparative Evaluation of Accuracy of Selected Machine Learning Classification Techniques for Diagnosis of Cancer: A Data Mining Approach. *Int. J. Biomed. Biol. Eng.*, 12, 19–25, 2018.
44. Mello-Román, J.D., Mello-Román, J.C., Gomez-Guerrero, S., García-Torres, M., Predictive Models for the Medical Diagnosis of Dengue: A Case Study in Paraguay, in: *Computational and Mathematical Methods in Medicine*, 2019.
45. Subhadra, K. and Vikas, B., Neural network based intelligent system for predicting heart disease. *Int. J. Innov. Technol. Exploring Eng. (IJITEE)*, 8, 5, 484–487, 2019.
46. Dwivedi, A.K., Artificial neural network model for effective cancer classification using microarray gene expression data. *Neural Comput. Appl.*, 29, 12, 1545–1554, 2018.
47. Sethi, L., Jha, A.K., Chandrakar, H.K., A novel PSO-MLP framework for feature selection and classification of gene expression data. *Yeast*, 1484, 17, 2013.
48. Padmavati, J., A comparative study on breast cancer prediction using RBF and MLP. *Int. J. Sci. Eng. Res.*, 2, 1, 1–5, 2011.
49. Pellakuri, V., Rajeswara Rao, D., Murthy, J.V.R., Modeling of supervised ADALINE neural network learning technique, in: *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, IEEE, pp. 17–22, 2016.
50. Sonawane, J.S. and Patil, D.R., Prediction of heart disease using multilayer perceptron neural network, in: *International Conference on Information Communication and Embedded Systems (ICICES2014)*, IEEE, pp. 1–6, 2014.
51. Öz, E. and Kaya, H., Support vector machines for quality control of DNA sequencing. *J. Inequal. Appl.*, 1, 2013, 85, 2013.

Part 2

MACHINE LEARNING AND GENOMIC TECHNOLOGY, FEATURE SELECTION AND DIMENSIONALITY REDUCTION

4

Dimensionality Reduction Techniques: Principles, Benefits, and Limitations

Hemanta Kumar Palo*, Santanu Sahoo and Asit Kumar Subudhi

Institute of Technical Education and Research, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India

Abstract

In machine learning, the effectiveness of the classification and regression analysis is often based on the reliability and dimension of the features representing the patterns. Nevertheless, there are several factors in the feature space which are correlated, overlapped, and can be considered irrelevant. These redundant data may occupy unnecessary storage space, and increase the response time, thus affecting the classification accuracy. The higher the redundant data, the complex it becomes to visualize the training set and simulating on it. Thus, there is a need to reduce the dimension of the data by keeping only relevant information. A 3-D pattern classification task is difficult to contemplate, whereas it is easy to map a 2-D pattern into a simple 2-D space or a 1-D task into a simple line. This is the reason, the dimension reduction techniques have been a growing trend of research in the field of pattern classification and regression. It is a process of reducing the number of features to the desired set by considering the principle variable which can adequately describe the given pattern. This motivates the authors to investigate several dimensional reduction techniques in the light of their principles, advantages, and limitations in this work. It also provides a comparative analysis of the discussed techniques and their suitability on a particular field of application.

***Keywords*:** Dimensionality reduction, feature selection, feature reduction, variables, machine learners

**Corresponding author*: hemantapalo@soa.ac.in

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (79–108) © 2021 Scrivener Publishing LLC

4.1 Introduction

The recent surge in device technology shows a growing deployment of many advanced sensors, actuators, or wearable devices, recording numerous data continuously in real-time for visualization. These data can be used for drawing inferences in many vivid application domains such as agriculture, healthcare, banking and finances, environmental science, remote sensing, artificial intelligence, cognitive sciences, astronomy, engineering, consumer transactions, pattern recognition, etc. The processing and management of such a huge amount of data daily basis posed a new challenge to the community as compared to the conventional and smaller datasets mostly studied in the past [1–5].

It has been observed that the generalization and learning ability of mostly machine learning algorithms suffer from small data inputs. On the contrary, the presence of large data adequately allows separation of the class boundaries. The machine learners struggle or find it difficult to model the desired pattern effectively. When the extracted feature set happens to be very large in comparison to the number of data samples or observations, it suffers from Curse of Dimensionality [6]. In data science, all of the data recorded or stored are not useful in every field. Many times, the recorded data are overlapped with correlated patterns, thus, confuse the machine learners. The use of low quality or irrelevant data may introduce noise into the learning system causing a delayed system response. Although a few computationally effective machine learners can predict the model accurately, there is still a need to eliminate redundant data for better system response and modeling.

Dimensionality reduction is a process of mapping a high-dimensional feature space into a low-dimensional space for effective learning [3]. The lower feature space contains only relevant data of the represented classes leading to a faster system response, low storage requirement, and better handling. This motivates the author to investigate a few state-of-the-art and conventional Dimension Reduction Techniques (DRT) explored by earlier researchers. The steps to mathematically formulate any DRT is explained below.

- Let's consider n-number of observations with each represented by a feature space as given by $s = (s_1, s_2,\ldots,s_p)^T$. Each p–dimensional feature space has a mean of $E(s) = (m_1, m_2,\ldots, m_p)^T$ and the covariance matrix of $E\{(s - m)(s - m)^T\} = \Sigma_{p\times p}$. Each observation matrix can be denoted as $S = \{s_{k,l}: 1 \le k \le p, 1 \le l \le n\}$.

- We need to form a low dimensional feature space $r = (r_1, r_2,\ldots,r_q)^T$, $q \leq p$ representing the attributes of the p–dimensional features using some selection criterion.
- There are two major DRTs: (a) Linear and (b) Non-linear. In linear DRT, each $q \leq p$ new feature is obtained by linearly combining the original features as

$$r_k = w_{k,1}s_1 + w_{k,2}s_2 + \cdots + w_{k,p}s_p \quad \text{for } k = 1, 2, \ldots, q \quad (4.1)$$

or

$$r = Ws \quad (4.2)$$

where $w_{q \times p}$ represents the linear transformation weight matrix
- A similar relation exists in the new feature space called the latent or hidden features as

$$s = Br \quad (4.3)$$

where $B_{p \times q}$ represents the linear transformation weight matrix. Following the $n \times p$ observation matrix S

$$R_{k,l} = w_{k,1}R_{1,l} + w_{k,2}R_{2,j} + \cdots + w_{k,p}R_{p,j}, \text{ for } k = 1, 2, \ldots, q$$

$$\text{and } l = 1, 2,\ldots,n \quad (4.3)$$

where the variable l denotes the jth realization, or, equivalently

$$R_{q \times n} = W_{q \times p}S_{p \times n} \quad (4.4)$$

$$S_{p \times n} = B_{p \times q}R_{q \times n} \quad (4.5)$$

4.2 The Benefits and Limitations of Dimension Reduction Methods

The development of advanced and large-throughput sources, generates huge amounts of data. These data comprise of thousands of variables much greater than the number of observations. The management and handling of such high-dimensional data becomes cumbersome due to

shortage of storage space and complex learning mechanisms. It warrants for the application of the dimensionality reduction method that retain only the relevant information. These methods compress the data to a desired dimension with an efficient utilization of the memory or storage space. Due to the removal of redundant data, the system training time and computational complexity decreases [7]. It allows a fewer degree of freedom with good network generalization capabilities. Ultimately, it prevents overfitting of new data during the training of a classifier. Additionally, the feature selection and optimization techniques allow the promotion of important variables as inputs to the machine learners with the improved model interpretability and visualization. The processing of low-dimensional, but relevant data arguably enhances the system response and machine efficiency. Further, some of the machine learners not suitable to handle large amounts of data can be effectively tested for their utility to their utmost potential using small data inputs. The obvious removal of noise and unwanted data using multicollinearity are likely to boost the system integrity.

Some of the major benefits of utilizing dimension reduction methods are:

- Reduced memory or storage space due to less amount of data.
- Faster training time and system response.
- Many efficient machine learners fail to perform adequately in the presence of large feature sets, that requires dimensionality reduction.
- Due to the handling of less data, their management and processing become easy and less costly.
- The removal of redundant data and retaining only relevant information provides better visualization of the patterns.
- The learning system becomes less complex while dealing with small data size.
- It results in less overfitting and better network generalization due to the presence of only useful information.

The limitations of dimension reduction methods are:

- It results in loss of data.
- The use of dimension reduction techniques makes the overall system expensive, hence remains application-specific.

- Most of the dimensionality reduction techniques are based on assumptions, hence may not be generalized in a true sense.
- It is not applicable for fine-grained classes since there is a chance of losing useful information.

4.3 Components of Dimension Reduction

A classic instance where the impact of feature size plays a significant role is the classification of e-mail for detecting spam versus genuine mail. The possible features can be

- Has the e-mail a generic title?
- The content or structure of the e-mail
- Is there a template?

In such a detection system, on can encounter many features that are correlated and overlapped. These features may not provide very useful information, hence can be treated redundant. The elimination of such irrelevant data is an essential step of data-science as it can improve the system response. Further, due to the less-amount of data, the operating platform is relaxed and easy to handle without any ambiguities. Similarly, consider a weather-forecasting system. The input to this system may contain many variables for accurate prediction. However, among these variables, the information obtained from parameters such as the rainfall and humidity is correlated as these come under a similar category. Under such condition it is advisable to discard one while keeping the other for weather prediction.

There are two approaches to reduce the data dimension and is briefed below:

- Selecting only the most relevant features and discarding the redundant features from the initial extracted feature set. This approach is known as feature selection.
- By forming a new reduced feature set of the original feature set using some statistical parameter or by applying some combination mechanism to the input feature set. In this case, each new feature variable extracted by combining a set of input features will retain the attributes of the original feature set. This approach is known as feature reduction.

4.3.1 Feature Selection

It aims to derive a smaller but more relevant feature subset from the extracted input feature set that can reduce the storage space and the system computational complexity. Feature selection facilitates achieving the followings:

- It helps the machine learners to train faster.
- The complexity of constructing a model is reduced.
- It helps an easier interpretation of results.
- The modeling accuracy is enhanced with the choice of right feature subsets.
- Less overfitting and better network generalization.

It usually involves three ways [8]:

(a) Filter method: It is a pre-processing step and is not associated with any machine learning technique. The selection of features is done based on the scoring in different statistical tests and their correlation with the extracted variables. Some of the statistical tests used for this purpose are the chi-square test, Pearson's Correlation, ANOVA, LDA, etc. The filtering method can be performed using the following steps.
   - Extract the input feature set
   - Choice of best feature subset
   - Use of a suitable machine learner
   - Performance appraisal.

   Some of the popular filter methods of feature selection are Missing Values Ratio, Low Variance Filter, Information Gain, Correlation Coefficient Score, Chi-Square Test, etc.

(b) Wrapper method: In this method, a chosen feature subset is used to train a machine learning model. With the drawn inferences of the previous outcome, either the addition or removal of features is made with the chosen subset. It is an iterative or search algorithm, hence is computationally expensive. The steps of the wrapper method are as follows:
   - Extract the input feature set
   - Choice of best feature subset (i) Generation of subset (ii) choice of a Machine learner
   - Performance appraisal.

In this method the Boruta package is used which creates shadow features to determine the significance of a feature. Initially, shadow features or shuffled copies of all features are created by adding randomness to the extracted feature set. Using some statistical tests such as the Z-score, t-score, or ANOVA, the importance of each feature is estimated by comparing with the shadow features. The least important features are constantly discarded iteratively. The algorithm stops either when all the features are checked and unimportant features are discarded or when it reaches the specified limits set by the user. Some of the wrapper methods of feature selection techniques are Forward Feature Construction (FFC), Backward Feature Elimination (BFE), Recursive Feature Elimination (RFE), etc.

(c) Embedded Method:
This method combines the virtues of both wrapper and filter methods. It is usually implemented using built-in feature selection machine learning algorithms. The popular embedded methods are LASSO and RIDGE regression. These algorithms have built-in penalization functions to tackle the overfitting issue. The LASSO algorithm solemnizes L1 regularization to add penalties. The penalty is equivalent to the absolute value of the feature magnitude. The RIDGE algorithm solemnizes L2 regularization to add penalty which is equivalent to the square of the feature magnitude

A few methods of choosing the variables are:

- Greedy algorithms: It allows the removal or addition of features subjected to fulfilment of certain conditions.
- Shrinking and penalization algorithm: It assigns a cost function corresponding to each variable so as to decide whether to discard it or not. For example, in an L1 regularization algorithm, some variables are assigned with zero values based on their importance. Similarly, the regularization algorithm restricts the space where the features can reside.
- Filtering of variables using correlation, VIF, or some "distance measure" between the features.

The major differences between the wrapper and filter methods are as follows [9].

I. While the filter method uses the correlation between the dependent variables to determine the importance of each feature, the wrapper method practically trains the system model to determine the relevance of each feature subset.
II. The filter method doesn't require to train the model, hence it is faster than the wrapper method.
III. While the former uses statistical techniques to select the relevant features, the latter uses the cross-validation approach for the same.
IV. The filter method sometimes fails to find the desired subset whereas the wrapper method always provides the best feature set.
V. The wrapper method is more prone to overfitting as compared to the filter models.

4.3.2 Feature Reduction

It maps a higher dimensional feature space into a reduced feature space. The usual procedure is to remove irrelevant columns in a dataset that do not contribute any new, or useful information while modelling a system. Feature reduction can be achieved mainly in two ways: (i) factor/component-based (ii) Project-based [12]. Examples of factor-based feature reduction techniques are Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), Factor Analysis (FA), Independent Component Analysis (ICA), etc. Similarly, a few examples of project-based feature reduction techniques are ISOMAP, t-distributed Stochastic Neighbor Embedding (t-SNE), Uniform Manifold Approximation, and Projection (UMAP), etc. A brief investigation of these techniques is given in the following sections.

4.4 Methods of Dimensionality Reduction

The explosion of advanced storage devices, accessibility of millions of attributes and records, sophisticated sensors, actuators, wearables, etc. has triggered several big-data issues besides the developments of many parallel data analytic algorithms. It has ultimately pushed the efficient utilization of valid data and redundant data elimination by way of different DRTs. The DRTs not only help to increase the speed of system

response, but also provides better recognition or clustering accuracy, using compatible machine learning algorithms [10]. It intends to remove unwanted noise or irrelevant data to achieve the desired amount of algorithm efficiency. This motivates the authors to investigate the foundations and the properties of several dimensionality reduction techniques. There are many methods based on Empirical Mode Decomposition (EMD) or wavelets specifically applied in the time series that exploits the frequent information of the signal under consideration. These reduction methods are sparse in the designated frequency space and are not applied to reduce the data size in general, hence they are not discussed here. For example, the wavelets tend to project the original time series into a fixed dictionary which makes the sparse projection (since it has very few features having a large value). In this case, the desired reduction of features has been carried out by fixing most of the features (usually of small values) to zero. Similarly, in EMD, it is essential to develop a dictionary adaptable to every individual input sample, hence not suits for feature reduction in general application domains. On the contrary, many DRTs consider the supervised method of classifications such as neural network, Naïve Bayes, decision tree, random forest, logistic regression, etc. Most of these algorithms find it unsuitable to handle large amounts of data due to their internal structure [11]. It is not the size of the data that poses this difficulty, rather the number of input columns that increase the learning time, hence are not fit for big-data analysis.

The different dimensionality reduction methods have been explained in the following section.

- PCA
- Missing Values Ratio (MVR)
- LDA
- ICA
- High Correlation Filter
- Low Variance Filter (LVF)
- Ensemble Trees/Random Forests
- t-distributed Stochastic Neighbor Embedding (t-SNE)
- Autoencoder
- Factor Analysis (FA)
- Backword Feature Elimination (BFE)
- Forward Feature Construction (FFC)
- Vector Quantization (VQ)
- Information Gain (IG)
- UMAP

4.4.1 Principal Component Analysis (PCA)

This is an unsupervised and statistical method of feature reduction technique introduced by Karl Pearson. It uses an orthogonal transformation to map the original high-dimensional feature set of n–dimension into a low-dimensional feature set of m–dimension comprising of Principle Components (PC). The first PC provides the highest possible variance signifying as large as variation in the input features followed by the succeeding PCs. However, all the PCs are uncorrelated or orthogonal to each other. The PCs are the eigenvectors of the symmetric covariance matrix, hence tend to be orthogonal. Since the PCA is based on the variance which reduces the dimension of the original feature set into a lower feature space, the m–PCs ($m < n$) retain most of the data information [13]. The technique tends towards maximum variance in the estimation of PCs during the mapping of features from higher-to-lower space. The PCA technique is either linear or non-linear, based on the technique employed. In calculating the PCs, the original data need to be normalized since the transformation process is sensitive to the relative scaling of the extracted feature set. Further, the PCs are the projected variables and are not real system-produced variables anymore, hence the feature set lacks interpretability. Thus, the technique is not suitable for tasks that require the interpretability of the outcomes.

In this transform the new projected variables are the linear combination of the original variables computed using eigenvalue decomposition, thus, the original feature set loses its meaning. If the original feature matrix is of size $p \times q$, then each row of p provides a single eigenvector or PC along with the original coordinates while the corresponding eigenvalue signifying the first column. A higher value of PC denotes a large variance of the feature on the corresponding eigenvector. The rows of the feature tables or the eigenvectors are arranged in descending order of eigenvalues or the variances. A subset of the estimated PCs can provide the most of the information of the original input feature set, hence the choice of the number of PCs remains an important parameter. It is chosen based on the task in hand and the type or the feature matrix as a higher number may aggressively reduce the data which may result in loss of information. The PCA works on numerical values of the features, hence doesn't act on the nominal columns. Further, it doesn't consider the missing values, thus not effective in case the columns comprising of many missing values.

The prime or linear PCA is discussed below.

It involves the following steps:

- Normalize the extracted feature set.
- Develop the covariance matrix of the feature matrix.

- Estimate the eigenvectors of the covariance matrix.
- Arrange the eigenvectors in descending order of magnitude signifying the variance of the input feature set.
- Select a suitable number of PCs. This way, a lesser number of PCs can be obtained.

The following are the major features of the PCA technique.

- Each PC is a linear combination of the original feature set.
- The PCs signify the maximum variance in the feature set.
- The second PC explains the remaining variance in the feature set and is uncorrelated to the first PC.
- The third PC explains the variance which is not explained by the first two PCs and so on

Disadvantages of PCA [14]:

- Loss of data.
- It determines the linear correlations among features, which is not always desired.
- The extraction of PCs is based on the estimation of the mean and covariance, hence the technique cannot be applied to all types of datasets.
- The choice of the number of PCs is ambiguous and based on thumb rules.
- Not suitable for columns having many missing values.
- PCs are the projected variable of the original feature set, hence the latter loses its meaning and interpretability.

Advantages of PCA [15]:

- It is a versatile technique that works well in practice.
- Its implementation is simple and fast.
- It is easy to test and compared the feature extraction algorithms with and without PCA.
- It offers many extensions and variations such as sparse PCA, kernel PCA, etc., to suit different task domains.

4.4.2 Missing Values Ratio (MVR)

In a feature set having columns with a large number of missing values usually carries a little relevant information. Hence, those columns having

missing values more than a certain threshold can be discarded. The higher the value of the threshold, the more the feature reduction process. Such a situation exists due to a recent increase in the number of sensors or actuators that continuously record and store data for future use. However, not all the data are useful for a particular application causing redundant feature overload. In such a case, it is advisable to explore the recorded data before generating the desired model. When there are missing values in the recorded data, either we have to compute it or drop them using an appropriate technique [16]. If the missing values are larger in number, i.e., greater than 50%, naturally it cannot provide much of the information on the variable. Thus, it is rather preferred to drop the variable containing the missing values rather than compute them. The choice of threshold is an essential parameter as it determines the percentage of missing values above the chosen threshold. A larger value of threshold makes the data reduction process more aggressive, which can hamper the performance. On the contrary, a lower threshold may not provide the desired reduction ratio. In general, a proper threshold can be set based on some optimization algorithms and then testing the recognition accuracy. It is usually preferable to test several classification algorithms, as different thresholds may provide a different amount of accuracy. A major advantage of this technique is that it can be applied to feature having numerical or nominal columns.

A straightforward approach to use the missing value data reduction technique is to count the number of missing values using some sort of interpolation strategy. After that, it is essential to remove the columns having missing values more than 50% (i.e. Percentage of missing values than the chosen threshold depending on the task in hand). Using statistical measures, the ratio of missing values corresponding to each column can be found out as

$$The\ Ratio\ of\ missing\ values = \frac{Number\ of\ missing\ values}{Total\ number\ of\ rows}$$

4.4.3 Linear Discriminant Analysis (LDA)

The LDA reduction is a supervised technique that dramatically facilitates class separability to obtain a decision region among the several patterns. However, its performance degrades in comparison to PCA for a smaller number of variables [17–18]. In the LDA technique, the low-dimensional feature set of m–dimension can be estimated as a linear combination of

the n–dimensional original features. It provides the m-discriminant functions ($m < n$) that are uncorrelated, maximizing the class separation. All the numeric columns in the original feature set are projected onto these linear discriminant functions resulting in dimensionality reduction. In this technique, initially, the input feature column is chosen. The maximum number of reduced dimensions corresponds to the classes in the input feature column less one. In general, the LDA assumes that the designated classes follow a multivariate normal distribution having the same variance but with a different meaning for individual classification [19]. The LDA is based on Fisher's linear discriminant and has been efficiently used in the field of pattern recognition, statistics, machine learning, etc. While the PCA maximizes the variance, the LDA maximizes the class separability. Thus, the LDA suits in applications having labeled data. Similar to PCA, the LDA is scaled sensitive, hence require the normalization of data before its application. Like PCA, the LDA features are also projected variables, hence are not interpretable. Thus, the original data loses its meaning in the transformation process. The choice of the number of LDA component is another issue and has to be decided with the task in hand. The use of labeled data in this technique makes it task-specific. The steps of LDA are shown in Figure 4.1.

The followings are the few major advantages of LDA.

- It is preferable for applications having a small number of samples with several variables.
- It performs better when the class conditional densities of clusters are approximately normal.
- The PCA algorithm ignores class labels in estimating the principal components and is unsupervised. On the contrary, LDA aims to provide the linear discriminants representing axes which can maximize class boundaries.
- The performance of LDA is better than PCA for multi-class classification problems.

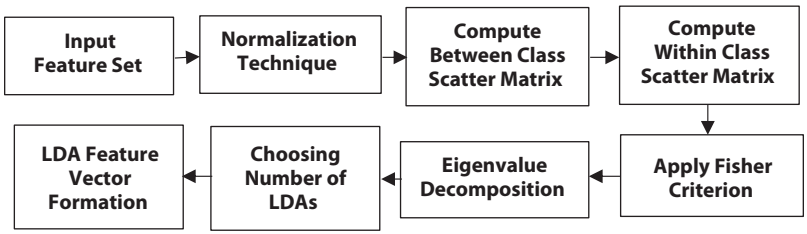


Figure 4.1 The Steps of LDA reduction technique.

- The shape and location of the original input feature set convert to another projected feature space in the PCA. On the other hand, the LDA only provides class separability, without any variation in the shape and location of the original feature space during transformation.
- Its training algorithm is simple which can be scaled well with several training examples.

The following are the few disadvantages of LDA [20]:

- The performance of LDA is poorer than PCA for a small sample size.
- In non-linearly separable classes, it is not possible to find a low-dimensional feature space using LDA.

4.4.4 Backward Feature Elimination (BFE)

In this technique, we begin with, the entire feature set and start discarding the least important feature iteratively for the enhancement of the model performance [21]. The process is repeated until there is no further improvement in the performance of the model by discarding the features. The steps of BFE are carried out as below.

- The chosen classifier is fed with n–input features and is trained at each iteration.
- Determine the model performance.
- The classifier is retrained with $(n - 1)$ input features after eliminating a one-input feature at a time.
- This way the classification model is trained n–times.
- The input feature that produces the lowest increase in the error rate is eliminated, providing $(n - 1)$ features.
- The model performance is determined by eliminating one feature each time and the model is retrained with the $n - 1$ remaining features.
- Now the classification process is repeated with $(n - 2)$ input features and so on.
- Thus, the classification model is developed by training on $(n - i)$ input features on each iteration i with an error rate $e(i)$.
- By choosing a maximum allowable error rate, a reduced set of features is obtained that provides the desired recognition accuracy.

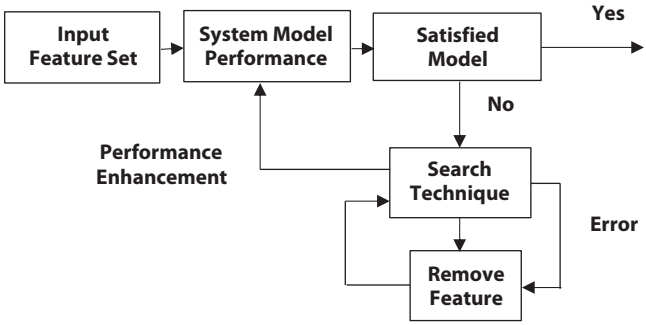


Figure 4.2 The steps of backward feature elimination technique.

Thus, in this technique, beginning with all n–dimensions, the sum of a square of error is estimated after eliminating every variable by n–number of times (n times). Consequently, it is imperative to remove the variables that produce the least increase in the error to obtain $n - 1$ input features. The process is repeated until it is not possible to find any variables which can be removed. Such a reduction technique is suitable to construct Logistic Regression or Linear Regression models.

The steps of the BFE algorithm have been shown in Figure 4.2.

4.4.5 Forward Feature Construction (FFC)

This is similar to FFC but is accomplished in the opposite direction. In this case, the process begins iteratively having no feature in the system model. It starts adding features one by one for the performance improvement of the model. This process continues until there is no further improvement in model performance. Thus, in this technique, we can start with a single feature and add one feature at a time progressively [22]. The feature to be added must provide the highest improvement in system efficiency. The biggest limitation of both FFC and BFE is that both are time-consuming and computationally expensive.

The steps followed in FFC are shown in Figure 4.3 and is explained below.

- Begin with a single best variable that provides the highest model performance.
- Train the system n–times separately with each feature.
- Repeat the process to improve the performance by adding one feature at a time. The feature, providing the greatest performance improvement is retained.
- Continue with the process until there is no further significant enhancement in the system performance.

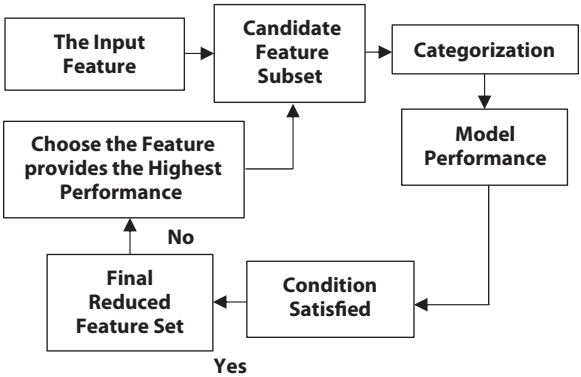


Figure 4.3 The steps of backward feature elimination technique.

4.4.6 Independent Component Analysis (ICA)

ICA is a popular feature reduction technique based on information-theory. In reducing the feature dimension, the ICA seeks independent components contrary to the PCA, which considers the correlation among features. Alternatively, the ICA intends to optimize the higher-order statistics such as kurtosis whereas the PCA provides due importance to the covariance matrix that describes the second-order statistics [6]. The ICA assumes that the chosen variable is a linear mixture of some random latent variables which are mutually independent, Thus, the extracted variables remains to be independent factors of the observed data. ICA is a computational and statistical feature reduction technique that reveals hidden contents in a random feature set. It assumes the features as a linear mixture of other unknown non-Gaussian independent latent features with an unknown mixing system. It performs better in finding the sources or factors when both the PCA or the factor analysis fails [18]. It decomposes the complex feature set into independent segments. It is a black-box approach and is popular in the blind source separation algorithm.

ICA has many variations like projection pursuit, FastICA, and Infomax [1, 3]. Its objective is to fetch independent components, mostly using the following techniques.

- maximizing the non-Gaussianity
- minimizing mutual information
- Utilizing the maximum likelihood (ML) estimation method.

ICA is an extension of PCA. The PCA decorrelates and maximizes the variance of features providing the PCs. The PCs are dependent and extracted assuming that the latent features obey the Gaussian distribution which is not always true for many processes. On the contrary, the ICA generates uncorrelated independent factors that are non-Gaussian and realistic. Thus, ICA remains more versatile than PCA in many applications.

The advantages of ICA are:

- ICA minimizes both second and higher-order dependencies present in the extracted input feature set.
- It aims to obtain the basis along which the features projected onto them are statistically independent.

The ICA has the following difficulties.

- In this, the initialization of the weight vectors is done randomly. The weight vectors are updated iteratively and rotated to obtain the ICs. Hence, it is not possible to find the ICs in a specific order, although, the weight vectors provide the source signals.
- The performance of the system model provides a very meagre different when the sign of the ICs is changed. Alternatively, it is possible to multiply the weight vectors either by +1 or −1. Thus, the weight matrix in an n–dimensional space results in two-local maxima for each ICs.

4.4.7 Low Variance Filter (LVF)

It is a feature reduction method that helps to determine the amount of information available in a data column. The data columns having similar or very little changes in feature values carries very little information. Hence, in this case, a threshold value of the feature is set, and the variance below the chosen threshold in the data column is discarded, thus the feature columns having variance below a chosen threshold are excluded. However, as the variance is range-dependent, the technique is formalized after normalization of the feature values [23]. Further, the normalization technique can make the estimated variances independent of the column domain range. Similarly, in case a data column has feature values of a constant value say '2', the features produce a zero variance. In such a situation, the modeling of the patterns cannot be efficient. The choice of a threshold

value is an important parameter for data column reduction as a higher threshold can degrade the accuracy.

The low variance filter technique after normalization computes the variance of each column. The low variance node discards the columns having a variance lower than the threshold. The de-Normalizer block finally normalizes the features of the selected columns to their original values as shown in Figure 4.4. Since the variance can be estimated from numerical columns, hence, the method is suitable for modeling the patterns having feature values expressed in numeric terms. The steps of the LVF technique are shown in Figure 4.4.

LVF is more suitable to discard features using threshold when the feature values do not vary much between consecutive observations. In this case, their variance remains below the chosen threshold. For example, if a health care dataset comprising 95% observations of 30-year-old male patients, then the information on the gender or age may carry no information at all. A major benefit of this technique is that it is relatively safe and easy to reduce the dimension at the beginning of the system modeling. Further, the choice of the desired variance threshold is simply based on solid intuition as similar or values of features varying very little are more likely to carry very little information. The limitation of this method is that it doesn't suit much for dimensionality reduction based only on the variance threshold. Again, it is desired to set the threshold manually, which remains tricky and task-dependent. Thus, it is advisable to start the process by choosing a lower threshold and move on iteratively.

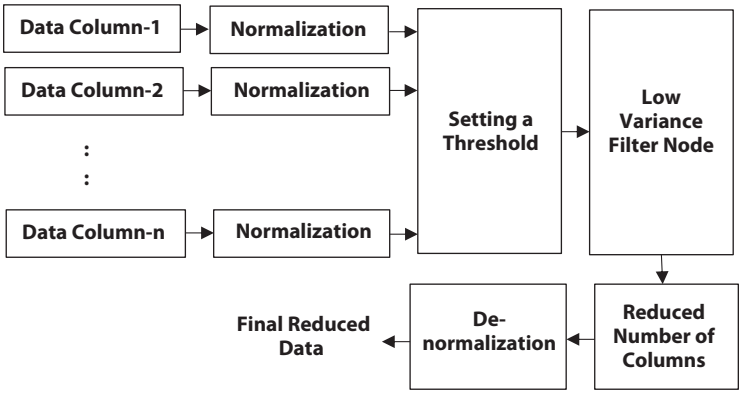


Figure 4.4 Low variance ratio feature reduction techniques.

4.4.8 High Correlation Filter

Data columns containing features with similar trends in general carry very similar information. Thus, it is desirable to discard all keeping only one of them. The procedure is to estimate the correlation coefficient among features or columns that include both numerical and nominal attributes. This way, one can remove redundant data. For example, if a floor area is expressed both in square foot and in the square meter as two separate variables, it is advisable to safely ignore one. For this purpose, initially one should estimate all the pairwise correlations. Then remove the features having a higher correlation above a chosen threshold by finding the feature having a greater mean absolute correlation corresponding to the other feature. Some examples of the correlation algorithms are Pearson's chi-square value or the Pearson's Product Moment Coefficient. In this case, also, a threshold is set to discard all the columns having a higher correlation coefficient than the chosen threshold whereas keeping only one [24]. The major benefit of this method is that the correlation is based on solid intuition. Further, a few machine learners are not compatible with correlated features, hence discarding them can enhance the system performance. The major disadvantage of this technique is that the correlation remains scale sensitive, hence the normalization of the column has to be carried out to obtain a fruitful correlation comparison. Thus, the choice of correlation factor is an important parameter as the dimensions having a larger correlation may lead to inefficient modelling. Similarly, setting a low threshold may lead to the elimination of relevant information. Thus, it is advisable to tune the correlation threshold manually, hence remains a tricky situation which makes the PCA a better alternative. Another problem is that the method is not suitable for tasks having multiple variables providing similar information. To determine the higher correlation, one can explore the Pearson correlation matrix to select the best coefficient using the Variance Inflation Factor.

4.4.9 Random Forests (RF)/Ensemble Trees

The ensemble Trees technique can solve multiple issues such as outliers, missing values, and identifying relevant features. It is the random forests or a Decision Tree (DT) Ensembles architecture which can be applied to select relevant features out of a large feature set. It creates a carefully constructed large tree with target attributes with several sets of branch trees using statistical parameters describing feature attributes [25]. However, this technique tends to bias towards variables with several distinct values.

For example, it favors the numeric variables over the categorical or binary values.

Thus, RF uses many DT can be formalized using the following steps.

- Let there be n–classes, each having m–input variables that assist for the decision at a tree node. It is imperative that $m \ll n$.
- Now select a training set by selecting n–times with replacement from all n–existing training sets.
- Use the remaining cases to compute the error of the tree and forecasting the corresponding classes.
- For a tree node, select m–variables randomly for the decision to be made at that node.
- Estimate the best split based on these m–variables corresponding to the input training set.
- Each tree is now considered to be fully grown, but not pruned.

The steps that need to be performed for each DT in an RF algorithm are shown in Figure 4.5.

The benefits of the RF method of feature reduction are as follows.

- It can overcome the overfitting issue by combining or averaging the outcomes of various DTs.
- It performs better for large-size data with a large range as compared to a single DT.

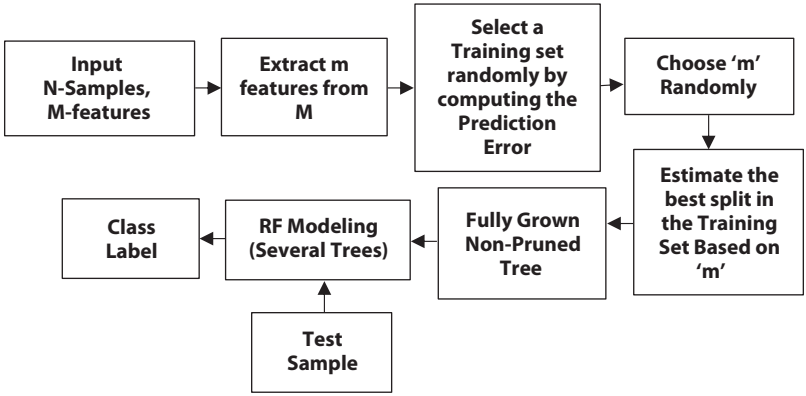


Figure 4.5 The steps of the random forest algorithm.

- The variance of RF is very less as compared to a single DT.
- The algorithms are more flexible and highly accurate in dealing with large-feature sets.
- It requires no scaling and maintains the desired accuracy without scaling of data.
- It suits in cases, where a large chunk of data is missing and provides good accuracy even when a considerable part of the data is missing.
- The learning is faster with a better visualization.
- It is not essential to detect variables.
- The RF estimates proximities between pairs for locating outliers, clustering and scaling providing interesting visualization of data. It is an experimental technique to identify variable interactions.
- It determines the test error without any cost involved about the repetition of model training associated with cross-validation.

The limitations of the RF algorithms are briefed below.

- It is a complex algorithm and building the RFs is much harder.
- It is computationally complex with large training time than the DT.
- For RF implementation, it requires a large computational resource.
- When the number of DTs is large, the RF remains less intuitive and interpretable.
- It has a slower prediction process as compared to most of the other conventional algorithms.
- It requires a large memory space.

4.4.10 t-Distributed Stochastic Neighbor Embedding (t-SNE)

This method eliminates the n–numeric feature columns in a feature set to a lower set of m–dimensions using a nonlinear relationship, where $m < n$. This way, the technique expresses every high-dimensional object with either two or three-dimensional points. Thus, similar objects can be expressed using nearby points, whereas the dissimilar objects can be defined by distant points in the reduced feature space [26].

Initially, a multivariate normal distribution is used to model the numeric columns of the feature set. Further, a low dimensional t-distribution is used to replace the distribution obtained. The replaced distribution resembles

the original multivariate normal distribution. In a low-feature space, the t-distribution can select another feature in the feature set as a neighbor to the current point. For a chosen point, the perplexity parameter is able to control the data density signifying the effective number of neighbors. A larger perplexity provides a better global structure. The t-SNE reduction method performs well on the current feature set and is not suitable to export the application domain of the system model to new data. The major advantages of this technique are:

- This technique can retain both the local and global structure of the data concurrently.
- It estimates the probability similarity of features both in high and low-dimensional space.
- In this, the high-dimensional Euclidean distances between features can be changed into conditional probabilities signifying similarities.

4.4.11 Autoencoder

It is a neural network having n–number of inputs and a similar number of outputs. It has a minimum one hidden layer with m- units, where $m < n$. The training is based on the backpropagation algorithm to map the input layer onto the output layer [27]. It minimizes the numeric feature columns using the hidden layer output to describe the input feature vector. In this, the encoder lies between the input and the hidden layer. The encoder reduces the n–dimensional input feature set into m–dimensional. The decoder part lies between the hidden and the output layer. The decoder expands the m–dimensional feature space at the output of the hidden layer into n–dimensional feature space in the output layer. This way, the original feature set is returned. Thus, autoencoder is an unsupervised technique helps to reconstruct the original feature set present in the input rather than used for classification purpose. The major advantage of this technique is due to the neural network structure that makes them suitable in many applications such as speech, or image processing. However, the inherent disadvantages of neural structures remain its main limitation. Hence, the structure is not a popular dimensionality reduction technique.

4.4.12 Factor Analysis (FA)

In Factor analysis, the features are segmented based on their correlations, wherein an individual group is described by a single underlying construct

or factor. In other words, each group or segment is represented by a factor [28]. In general, these factors are very small in number, hence factor analysis can provide the desired dimensionality reduction. A major issue of this technique is that the denoted factors are difficult to visualize or observe. Factor analysis is carried out using two techniques. These are:

- Exploratory Factor Analysis
- Confirmatory Factor Analysis.

For example, let's consider two variables: Education and Income. These variables arguably provide a high correlation since educated people are likely to have a higher income and vice versa. Thus, in FA all variables corresponding to a specified segment can have a higher correlation among themselves, However, the correlation remains low with other variables corresponding to other segments.

4.4.13 Uniform Manifold Approximation and Projection (UMAP)

t-SNE technique performs better in the presence of a large feature set. However, it has limitations like slow computation time, loss of large-scale information, and incapable of representing a very large feature set meaningfully [29]. This issue can be resolved using the UMAP dimension reduction technique. The technique can preserve the local, and the global data structure more efficiently as compared to t-SNE with faster runtime. A few major benefits of this algorithm are:

- Ability to manage and process large feature sets or high-dimensional data easily.
- It has better visualization of the data besides dimensionality reduction attributes.
- It can preserve both local and global structure of the feature set.
- It can map both nearby and faraway data points in a high-dimensional feature space into the low dimensional feature space using a similar technique.

4.4.14 Information Gain (IG)

It is an entropy-based feature selection technique popularly used in machine learning. It signifies the amount of information the feature set provides for

the categorization of the pattern. It estimated the reduction in surprise or entropy when a transformation is carried out from one feature space to another [31]. Mostly the technique is applied to form DTs using the training feature set utilizing the IG provided by each variable. In the process, it selects the suitable variables that provide the maximum IG ultimately minimizing the entropy using the best data split. It determines the gain of every individual variable corresponding to the target variable. The larger the IG, the lower the entropy provided by the feature set and the lesser the surprise. In general, a skewed feature distribution result in a lower entropy, whereas the features having equal probability causes a large entropy. Thus, the features having a lower probability provide more surprise and hence more information. The IG is computed through entropy comparison of the feature set before and after the process of transformation. In other words, the mutual information estimates the statistical dependence between two features and is known as IG for feature selection application. IG can be used to rank the features corresponding to their order of importance [32]. A feature ranks higher when its IG is higher and is more reliable in classifying the pattern. Unrelated features have a low ranking and lower IG as compared to the perfectly partitioned features.

The major advantages of the IG feature selection technique are:

- It applies to attributes with a large number of distinct values with improved training of the input feature set.
- It determines the most relevant attributes for testing near the root of a DT.
- It helps to reduce the bias towards multi-valued attributes by considering several branches of different during the selection of an attribute.

The major limitations of the IG feature selection algorithm are:

- Although IG can focus on relevant features based on ranking, it is not fully perfect.
- When the feature set comprises of a large number of distinct values, the IG technique causes the overfitting of data.

4.4.15 Vector Quantization (VQ)

It is a data compression technique which greatly helps to reduce the computation time and memory. It is a non-parametric method mostly reliable for spectral analysis. It has been efficiently explored in the field of speech/

speaker and emotion detection/recognition [33–36]. The algorithms can effectively eliminate redundant data by utilizing four interrelated properties. These are:

- linear dependency (correlation)
- nonlinear dependency
- the shape of the probability density function (*pdf*)
- vector dimensionality.

As compared to the VQ, the scalar quantization technique explores the properties such as the linear dependency and *pdf* shape. The use of the nonlinear dependency property allows the VQ in quantizing the signal spectral components. On the other hand, the vector dimensionality facilitates waveform quantization. Nevertheless, the VQ technique is expensive, hence is task-specific. The cost involved increases exponentially corresponding to an increase in feature dimension as well as with an increase in the number of bits. With the advances in storage facilities and computational platforms, VQ can be utilized at a transmission rate as low as 1 bit per parameter or even less. Nevertheless, it is difficult to operate scalar quantizers at these bit rates due to degraded performance.

The major features of the VQ technique are:

- Signal representation produces a feature vector sequence.
- Applicable for signal compression, image, and speech coding.
- The transmission of information is better than that of scalar quantization (at a rate less than 1 bit/parameter is possible).
- It has been used for discrete acoustic modelling using several conventional clustering approaches.

The major advantages of VQ are:

- It is suitable to solve specific tasks related to speech/speaker recognition, such as speaker adaptation using codebook mapping.
- Several techniques, such as fuzzy VQ and semi-continuous modelling of HMM, have also been investigated to cope with the quantization distortion problem.
- The VQ technique has the potential to produce new ideas in speech/speaker recognition using advanced algorithms, such as neural networks, stochastic variables, statistical approaches, etc.

The major limitations of VQ technique are:

- It has large computational and storage costs.
- The cost of computation increases exponentially with an increase in the number of bits corresponding to a vector in full-search coding.
- The cost of storage and computation doubles with an increase of one bit in the transmission rate. With a further increase in transmission rate, the cost of computation and storage becomes too expensive that prohibits its use.
- The judicious selection of the codebook is very much essential for its effectiveness.

4.5 Conclusion

Missing Value Ratio is used for dimensionality reduction when the feature set has too many missing values. It is possible to discard the columns with several missing values in them. The Low Variance Filter (LVF) approach helps to detect and discard the constant parameters in a feature set. The target feature column is not affected by features with low variance. Thus, it is possible to discard these features variables safely. The High Correlation Filter (HCF) is suitable when a pair of features with a high correlation in the extracted set increases multicollinearity. It is more advantageous to detect highly correlated features and discard them as required. Random Forest (RF) remains popular which identifies the significance of each feature present in the feature set. This way we can retain the most relevant features resulting in redundant data removal. The Backward Feature Elimination (BFE) and Forward Feature Construction (FFC) algorithms require a large computational time, hence mostly preferred for a small database. Factor Analysis (FA) is preferred when the features are highly correlated as it segments the features into sub-groups based on their correlation. In this case, each sub-group is denoted by a factor. Principal Component Analysis (PCA) is a widely tested technique to deal with linear data. It expressed the large feature space into a smaller feature space using a set of projected variables known as principal components. Independent Component Analysis (ICA) helps to transform the feature set into a smaller set known as independent components. Both ISOMAP and t-SNE is more suitable when the extracted feature set is strongly non-linear. However, t-SNE can visualize the feature set better. UMAP is computationally efficient with short runtime as compared to t-SNE, hence suitable in cases where a faster response

is desired. Further, UMAP can preserve the global structure of the feature set better than t-SNE. Thus, it provides the "big picture" view of the feature set better by preserving the local neighbor relations as well.

References

1. Hyvarinen, A., Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. Neural Netw.*, 10, 3, 626–634, 1999.
2. Palo, H.K., Mohanty, M.N., Chandra, M., Efficient feature combination techniques for emotional speech classification. *Int. J. Speech Technol.*, 19, 1, 135–150, 2016.
3. Hyvärinen, A. and Oja, E., Independent component analysis: Algorithms and applications. *Neural Netw.*, 13, 4, 411–430, 2000.
4. Mohanty, M.N. and Palo, H.K., Child emotion recognition using probabilistic neural network with effective features. *Measurement*, 152, 107369, 2020.
5. Palo, H.K. and Mohanty, M.N., Wavelet based feature combination for recognition of emotions. *Ain Shams Eng. J.*, 9, 4, 1799–1806, 2018.
6. Tharwat, A., Independent component analysis: An introduction. *Appl. Comput. Inform.*, 2018. https://www.emerald.com/insight/2210-8327.htm)
7. Mohanty, M.N. and Palo, H.K., Segment based emotion recognition using combined reduced features. *Int. J. Speech Technol.*, 22, 4, 865–884, 2019.
8. Guyon, I. and Elisseeff, A., An introduction to variable and feature selection. *J. Mach Learn. Res.*, 3, 1157–1182, 2003.
9. Saurav, K., *Introduction to Feature Selection methods with an example (or how to select the right variables)?*, Analytics Vidhya, Gurgaon, India, 2016.
10. Palo, H.K., Chandra, M., Mohanty, M.N., Emotion recognition using MLP and GMM for Oriya language. *Int. J. Comput. Vis. Robot.*, 7, 4, 426–442, 2017.
11. Palo, H.K. and Mohanty, M.N., Modified-VQ features for speech emotion recognition. *J. Appl. Sci.*, 16, 9, 406–418, 2016.
12. Van der Maaten, L.J.P., An introduction to dimensionality reduction using matlab. *Report*, 1201, 07, 62, 2007.
13. Chai, Y., Tao, S., Mao, W., Zhang, K., Zhu, Z., Online incipient fault diagnosis based on Kullback–Leibler divergence and recursive principle component analysis. *Can. J. Chem. Eng.*, 96, 2, 426–433, 2018.
14. Lee, S., Drawbacks of Principal component analysis, *Academia Sinica*, arXiv preprint arXiv:1005.1770, 95, 1–4, 2010.
15. Karamizadeh, S., Abdullah, S.M., Manaf, A.A., Zamani, M., Hooman, A., An overview of principal component analysis. *J. Signal Inf. Pr.*, 4, 3B, 173, 2013.
16. Toutenburg, H. and Srivastava, V.K., Amputation versus imputation of missing values through ratio method in sample surveys. *Stat. Pap.*, 49, 2, 237–247, 2008.

17. Navarrete, P. and Ruiz-del-Solar, J., Analysis and comparison of eigenspace-based face recognition approaches. *Int. J. Pattern. Recognit.*, 16, 07, 817–830, 2002.
18. Delac, K., Grgic, M., Grgic, S., A comparative study of PCA, ICA, and LDA, in: *Proc. of the 5th EURASIP Conf. focused on Speech and Image Processing, Multimedia Commun. and Serv.*, Jul. 2005, pp. 99–106.
19. Wang, J.S., Chiang, W.C., Hsu, Y.L., Yang, Y.T.C., ECG arrhythmia classification using a probabilistic neural network with a feature reduction method. *Neurocomputing*, 116, 38–45, 2013.
20. Tharwat, A., Gaber, T.I., Abdelhameed, H., Aboul, E., Linear discriminant analysis: A detailed tutorial. *Al Commun.*, 30, 169–190, 2017.
21. Kostrzewa, D. and Brzeski, R., The data dimensionality reduction in the classification process through greedy backward feature elimination, in: *Int. Conf. on Man–Machine Interactions*, Oct. 2017, Springer, Cham, pp. 397–407.
22. Mahapatra, B. and Patnaik, S., Data reduction in MANETS using forward feature construction technique, in: *2015 Int. Conf. on Man and Machine Interfacing (MAMI)*, Dec. 2015, IEEE, pp. 1–3.
23. Wood, L.B. and Asada, H.H., Low variance adaptive filter for cancelling motion artifact in wearable photoplethysmogram sensor signals, in: *2007 29th Annual Int. Conf. of the IEEE Eng. in Med. and Bio. Soc.*, Aug. 2007, IEEE, pp. 652–655.
24. Choi, J., Jin Chang, H., Yun, S., Fischer, T., Demiris, Y., Young Choi, J., Attentional correlation filter network for adaptive visual tracking, in: *Proc. of the IEEE Conf. on Comp. Vis. and Patt. Recog.*, pp. 4807–4816, 2017.
25. Breiman, L., Random forests. *Mach. Learn.*, 45, 1, 5–32, 2001.
26. Belkina, A.C., Ciccolella, C.O., Anno, R., Halpert, R., Spidlen, J., Snyder-Cappione, J.E., Automated optimized parameters for T-distributed stochastic neighbor embedding improve visualization and analysis of large datasets. *Nat. Commun.*, 10, 1, 1–12, 2019.
27. Lee, K. and Carlberg, K.T., Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *J. Comput. Phys.*, 404, 108973, 2020.
28. Poncela, P., Rodríguez, J., Sánchez-Mangas, R., Senra, E., Forecast combination through dimension reduction techniques. *Int. J. Forecast.*, 27, 2, 224–237, 2011.
29. Húsek, D., Moravec, P., Snášel, V., Frolov, A., Řezanková, H., Polyakov, P., Comparison of neural network boolean factor analysis method with some other dimension reduction methods on bars problem, in: *Int. Conf. Patt. Recog. Mach. Intell.*, Dec. 2007, Springer, Berlin, Heidelberg, pp. 235–243.
30. Becht, E., McInnes, L., Healy, J., Dutertre, C.A., Kwok, I.W., Ng, L.G., Newell, E.W., Dimensionality reduction for visualizing single-cell data using UMAP. *Nat. Biotechnol.*, 37, 1, 38–44, 2019.

31. Lei, S., A feature selection method based on information gain and genetic algorithm, in: *2012 Int. Conf. Compu. Sc. and Electr. Eng.*, Mar. 2012, vol. 2, IEEE, pp. 355–358.
32. Alhaj, T.A., Siraj, M.M., Zainal, A., Elshoush, H.T., Elhaj, F., Feature selection using information gain for improved structural-based alert correlation. *PLoS One*, 11, 11, e0166017, 2016.
33. Palo, H.K., Mohanty, M.N., Chandra, M., New features for emotional speech recognition, in: *2015 IEEE Power, Communication and Information Technology Conference (PCITC)*, Oct. 2015, IEEE, pp. 424–429.
34. Palo, H.K. and Mohanty, M.N., Comparative analysis of neural networks for speech emotion recognition. *Int. J. Eng. Technol.*, 7, 4, 111–126, 2018.
35. Palo, H.K. and Sagar, S., Comparison of neural network models for speech emotion recognition, in: *2018 2nd International Conference on Data Science and Business Analytics (ICDSBA)*, Sept. 2018, IEEE, pp. 127–131.
36. Palo, H.K., *Spectral prosodic and hybrid features for emotion recognition*, PhD Theses, Shodhganga: A reservoir of Indian theses @ INFLIBNET, Siksha O Anusandhan, Bhubaneswar, Odisha, India, 2018.

5

Plant Disease Detection Using Machine Learning Tools With an Overview on Dimensionality Reduction

Saurav Roy[1], Ratula Ray[1†], Satya Ranjan Dash[2] and Mrunmay Kumar Giri[1*]

[1]School of Biotechnology, KIIT Deemed to be University, Patia, Bhubaneswar, India
[2]School of Computer Applications, KIIT Deemed to be University, Patia, Bhubaneswar, India

Abstract

Food security is a matter of concern for different countries, irrespective of their developmental status. Plant stress response is one of the major concerns for crop productivity. "Climate change and global warming turned out to be the main contributors to crop loss by imparting both abiotic and biotic stress to the plant. Identifying the reasons and solutions to the problems related to crop loss is one of the major areas of concentration for researchers worldwide. The traditional method of identification, classification and quantification of plant stress by visual phenotyping needs specialized human resources and is often associated with erroneous results. An array of different parameters contributes significantly to the decrease in crop yield. Machine Learning (ML) has revolutionized the way in which we approach and deal with such problems nowadays, i.e., predicting the outcome with a reliable accuracy score, thus preventing the crop loss significantly. Researches concerned with Plant Stress Biology (PSB) have gained significant impetus due to the advent of ML techniques, where prediction studies can be carried out based on the feature space for both image and numeric datasets. This feature space is generally large, owing to the correlation between different features with the final outcome for classification. In this book chapter, we have particularly aimed towards introducing the concept of Dimensionality Reduction and have dealt with the part of "Feature Extraction" in particular. Dimensionality

**Corresponding author*: mrunmay.giri@kiitbiotech.ac.in; mrunmayjnu@gmail.com
[†]Equal contribution as first author

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (109–144) © 2021 Scrivener Publishing LLC

Reduction is a part of data pre- processing and is focused on reducing the original feature space by getting rid of the redundant and less important attributes. Our chapter will focus on the understanding the different feature extraction techniques for both numeric and image data for effective prediction of plant disease as a result of biotic stress. We will also focus on the implementation of some of these techniques on publicly available datasets to understand how the dimensionality reduction methods work to provide us with reliable outcomes for prediction of crop diseases and assist in their management.

***Keywords*:** Feature extraction, smart agriculture, principal component analysis (PCA), dimensionality reduction, plant disease detection, image analysis, numerical data analysis, machine learning (ML)

5.1 Introduction

The temperature of the earth is increasing day by day. It is expected that there will be an increase in earth's temperature 1–6 °C at the end of this century. Climate change has a profound effect in limiting the growth of agricultural production. It's becoming a challenge now to feed the entire human population in the coming future. Due to reducing availability of land for cultivation and proportionate increase in human population, we need implementation of modern scientific techniques and use of climate resilient crop varieties in agriculture for better stress management. Plant stress can be explained as a state in a period of a plant's life cycle, where the plant growth is taking place under non-ideal conditions, which in turn, affects its growth, and hinders its ability to grow normally. Plants, in their life cycle, are exposed mainly to two types of stress: abiotic and biotic. Abiotic stresses are those which include the environmental factors like, temperature, availability of water, soil salinity, etc. On the contrary, biotic stress factors include all the organisms that coexist in vicinity, and interact with the plant (e.g. nematodes, microbes, viruses, insects, etc.) [1, 2]. There can be various effects of stress. Some of them include deficits in growth, reproductive impairment, affected crop yields, permanent damage, death, etc.

Under stress conditions, plants undergo certain morphological [3], mechanical [3] and biochemical [4] changes, which lead to a decrease in crop productivity. So, in order to ensure agricultural sustainability in the near future, early detection of stress phenotype is needed for proper management. The symptoms developed by plants due to various types of abiotic stresses share common phenotypic signatures like wilting of leaves, decrease in chlorophyll content, closure of stomatal aperture, etc. However, the symptoms developed by plants in response to various types of biotic

stresses are very specific and are needed to be identified properly for efficient control of the disease. The common practice which is being followed till now by farmers, breeder and agriculture scientists is the manual inspection and identification of the disease symptoms based on prior experience and knowledge. In traditional way, manual assessment is done by trained raters; by using a particular scale [5]. With time, like all other systems, this system of manual inspection was also found out to be outdated and inefficient owing to the quick analysis of a large amount of data, (i.e. many hectares of crops; drying up of forests, etc.) that is needed to be assessed in a faster and smarter way to conclude the onset of a stress (disease) in a population of healthy plants. This method of manual inspection and categorization of stress phenotype is quite time consuming and often associated with inter- and intra-rater variabilities. In this modern era of farming, extensive use of automated vehicles (like unmanned aerial vehicles—UAVs) and sensors in the agricultural field have resulted in an extensive inflow of data. These UAVs and ground robots take image data at various time points of a day throughout the cultivated field, over the entire season for monitoring the health of the plants. Even the cellular mobile phone images of the farmers and breeders add to the massive repertoire of datasets. Hence, it is practically impossible for the trained personals to analyze such a huge collection of datasets and make sense out of this without making erroneous conclusions. This necessitates the use of machine learning (ML) tools and artificial intelligence (AI) to play a lead role in storing and analyzing the large complex datasets for the identification, classification, quantification and accurate prediction of biotic stress phenotype in plants.

The arrival of a new age of computational science and technology, has *revolutionized* the approach of stress phenotyping. Different methods of imaging technologies used for remote sensing like Visible light imaging (VLI) [6], Chlorophyll fluorescence imaging (CFI) [7], Hyperspectral and thermal imaging came into play [8]. So, instead of manual rating, computer algorithms are now being used for collection, analysis, and interpretation of the data. Marr's three levels of information processing was a widely used framework that contributed largely to this system at the beginning [9, 10]. Further development in the algorithms made the system more efficient and sensitive towards detecting even smaller recognizable plant phenomics changes. Softwares like ASSESS 2.0 [11] created by Lakhdar Lamari, from University of Manitoba, Canada do the aforementioned functions but lack in flexibility. Using databases like Plantvillage [12], UCI [13], etc. scientists can make artificial intelligence to carry out similar plant phenomics function with much flexibility.

The world is headed towards a data-driven revolution, with so much of secondary data being generated every second. A huge chunk of this data gets wasted due to lack of awareness regarding how they can be processed, to derive more information. Harnessing the strengths of data-driven technologies will help in addressing the problems related to food security. Smart farming practices can help save a lot of cost and time as opposed to manual inspection, and also help produce greater yield. Big data analytics can assist in crop management on and off the field, by providing farmers with necessary information so that they can make an educated decision regarding various aspects of farming. Data collected from the sensors on-field can help a lot in case of improving resource utilization. Drones (like Intel "Falcon" 8+) are also used to provide coverage of the survey area, in order to monitor the vegetation, condition of the crops, etc.

To understand the correlation between plant stress and the diminished yield from an image and numerical datasets, developing different image processing algorithms is essential. Computational pipelines can be optimized to do so with greater ease, based on the most important aspect of the data that needs to be analyzed in relation to the result, and it is irrespective of the data type. In order to optimize huge chunks of data and filter out the most important parameters, different statistical tools (such as dimensionality reduction; along with Principal Component Analysis, Heatmap, t-SNE graphs, MDS graphs) have been employed over the years [14, 15]. For example, data dimensionality reduction had been used for the preprocessing and interpretation of the data on metabolomic and proteomic level [16]. Data dimensionality reduction is one such lite, exiguous, useful technique, which filters out the feature set by dropping the number of random variables present in the data. Here, in the following chapter, we aim to dive deep into the process of feature extraction in dimensionality reduction and its application to predict plant stress responses and their relation with the disease which occurs, using both numerical and image datasets.

5.2 Flowchart

We have tried to map out the structure of our book-chapter in the flowchart below (showed in Figure 5.1). Our aim here is to deal with two primary case studies for implementation, which include both the numerical and image datasets related to plant stress. We have tried to understand how feature extraction plays a role in understanding the features, that are associated to the final output in case of a supervised classification task.

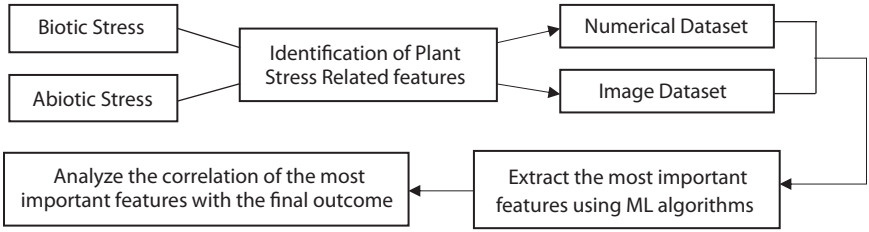


Figure 5.1 Flowchart to depict the structure of the book chapter.

5.3 Machine Learning (ML) in Rapid Stress Phenotyping

Machine learning is a multidisciplinary method used in various types of data analysis. Often, this uses algorithms that are based on statistics, data visualization, probability theory, decision theory and optimization for the needful output [17]. Over the years, automated and high-throughput imaging approaches have evolved. Techniques have changed from manual raters, to data entry employees and data officers to auto-sorting algorithms that comprise lines of codes [18]. Now, recognition of plant stress data is done depending on images and numerical datasets based on morphological and genetic characteristics of plants.

For real world applications, computer aided learning methods are used, such as computer vision [19, 20], deep learning [21], etc. This allows the primary and precise detection and enumeration of all stress-causing agent. In an attempt to maximize efficient control of weeds, pests and diseases, it's equally significant to know the stress causing agent as well as the severity of the stress signs or symptoms. Rapid diagnosis and proper quantification of biotic stress are the main purposes in the discipline of phytopathology [22]. The recommendation of control and prediction of yield is done by looking into what percentage of plant is symptomatic and what is the severity of the infection [5]. These are identified by botanists, plant biotechnologists agricultural scientists, etc. as sometimes identification of plant diseases is challenging even for humans with experience [23].

With machine learning, extracting patterns and features for the required data is the utmost need of the hour. It's not just only about the quantity of the data anymore, but also the quality and specificity. In recent years deep learning and machine learning have provided significantly impressive performances in various domains [24]. Generally, in Machine Learning (ML), the focus is on solving one problem, such as identifying diseases.

To perform this efficiently, a model, or sometimes, a set of few similar models are trained to solve specific tasks. Then, they are provided with a dataset to perform the pre-trained tasks. The four stages of ML deployment in agriculture namely, Identification, Classification, Quantification and Prediction; together known as ICQP it is one most systematic and scientific approaches towards analysis which is one smart tool to aid agriculture of tomorrow [17].

5.4 Dimensionality Reduction

Dimensionality reduction can be explained as the transformation of high-dimensional data into a meaningful representation of reduced dimensionality [25]. It has already been used in solving various big-data problems concerned with technologies in the biological domain, such as precision agriculture [26], stress phenotyping [6], predictive analysis [27], etc. The problem that lies in developing this system is the identification of distinctive features for various types of stress morphology. The symptoms (features) which are observed for a particular type of stress are often the same across all the plants (for example, yellowing of leaf may be caused by a nutrient deficiency, salt stress or pest attack, droopiness of leaf can be caused by high heat stress and also lack of water). So, Feature extraction is essential part of dimensionality reduction, and is very useful in extracting important features and understands the type of stress that it related to.

5.4.1 Feature Extraction

Machine learning employs feature extraction in pattern recognition, identification and assessment of different datasets like numerical, image or datasets of various other types. Feature extraction can be explained as the process to find out the most important data in a bunch of raw data or datasets [28]. A dataset comprises a set of features and the corresponding class labels for each of the instances present in the dataset. Using these dimensionality reduction methods, this set of parameters can be further reduced. The selected features contain the appropriate information from the input data that are easy to analyze and work as more effective representation of the initial data [29]. Few commonly used features extraction techniques (for both numerical and image datasets) are explained in the following subsections.

5.4.1.1 PCA (Principal Component Analysis)

PCA or principal component can simply be explained as a perpendicular, linear transformation system, that transforms each of the new data to a new coordinate system in a way that the greatest variance by the data clusters on each of the coordinate system [30]. In this way, the data corresponding to the first coordinate (called the first principal component) has the most variation, second one has the second most and so on. The components with the highest variation consist of most of the information regarding the classification task. Each of these components is influenced by all of the features and the ones which influences the components the most are the most important ones which contribute to the classification. The orientation of the first PC is such that geometrically it causes greatest variance in the dataset, hence retaining the maximum classification information. The Eigen values, associated with the Eigen vectors that gives the direction of the axes that determine the variance for each of these principal components.

5.4.1.2 LDA (Linear Discriminant Analysis)

LDA quite closely related to PCA or principal component analysis. Factor analysis is something in common in both of these techniques and they both look for linear combinations of variables that explain the data in the best possible way. However, instead of just looking into the value of the variances like in case of PCA, LDA organizes a much simpler method to 'fit' the required data points (variables). It does so in such a way that there is maximum clear distinction amid the classes of data. LDA is best suited for use when independent variables are used and where each of the observations is a continuous quantity [31].

5.4.1.3 SIFT (Scale Invariant Feature Transform)

It is a commonly used feature extraction method for images. For SIFT algorithm [32], DoG (Difference of Gaussian) operator is used in order to find the local maxima for detecting the feature-points. It looks for extracting the 16×16 pixels in the neighborhood of the specific feature points and further divide this region into sub-blocks. This algorithm is invariant to the scale (which can result in change of size and quality of an image) of the images or the rotations of the images, when the two images are being compared against one another for feature matching. They also have limited

affine variation. The only problem with this technique is the high computational cost that this process entails.

5.4.1.4 SURF (Speeded Up Robust Features)

This SURF is also a feature extraction technique for image analysis [20], which uses the determinant of the Hessian matrix in order to locate the feature position and the scale of the image. The main advantage of SURF over SIFT is that SURF has a much lower computational cost as compared to SIFT and like SIFT, it is also invariant to rotation and scale changes. It also has lesser amount of affine invariance.

5.4.1.5 ORB (Oriented FAST and Rotated BRIEF)

The ORB method uses a combination of modified version of FAST (Features from Accelerated Segment Test) and rBRIEF (rotation aware Binary Elementary Features) algorithms, in order to extract the most important feature points of the image [19]. It is a good alternative to the SIFT and SURF algorithms in the sense that the other two are patented algorithms and ORB is a publicly available one, which can be easily used to extract the important features. It performs almost as good as the SIFT algorithm in practice and gives a better performance than SURF. The ORB algorithm uses a multi-scale pyramid for the FAST features to extract key-points based on the multi-scale representation of the image. The BRIEF part of the algorithm converts the key-points extracted by FAST into binary vectors. As BRIEF in general is not invariant to rotation, ORB uses the rotation aware BRIEF (rBRIEF) in this case. It is a very fast process and computationally inexpensive when compared to SIFT or SURF, while giving almost the same level of performance.

5.5 Literature Survey

Table 5.1 contains a brief literature survey over the past decade, regarding the use of Machine Learning techniques in case of detection of plant stress phenotyping.

Table 5.1 Highlights the important work of literature published in the last two decades regarding the use of Machine Learning in plant stress phenotyping.

No.	Type of Stress	Plant Name	Method Used (Algorithm)	Experimental setups	Year
1.	Abiotic (Drought)	*Impatiens hawkeri*	Machine vision (Automated inspection from images through image acquisition and feature extraction)	Used for determining the Top canopy area of plant in order to detect the water stress through a non-contact sensing technique. Gray level histogram was used for finding out the features of the images. The diurnal movement of the plants due to water stress was used as a marker for detection, where the coefficient of relative variation was calculated to predict the amount of stress being induced [33].	2002
2.	Abiotic (Drought)	Potted Forsythia	Image processing using single layer perceptron	Used for calculating the shape parameters (fractal dimension, invariant moments and development of terminal branches) by keeping the plant "black" in color and the background "white". Perceptron with a single hidden layer was used in order to perform the classification task. The shape parameters were used to interpret the reaction of water stress in plants [34].	2004

(*Continued*)

Table 5.1 Highlights the important work of literature published in the last two decades regarding the use of Machine Learning in plant stress phenotyping. (*Continued*)

No.	Type of Stress	Plant Name	Method Used (Algorithm)	Experimental setups	Year
3.	Biotic (Rice blast) Pathogen: *Magnaporthe grisea*	Rice	Support vector machine (SVM), Generalized regression neural network (GRNN)	A cross location model was developed by use of five different places. Moreover, a model was made and validated using GRNN. Further comparison with SVM. The method employed Vector Machine data for development of weather-based prediction models. This experimentation showed that SVM is better than conventional machine learning techniques in forecasting plant diseases [35].	2006
4.	Biotic Pathogen: (*Uromyces betae*, *Cercospora beticola)*	Sugar beet	K-nearest neighbor (k-NN) and Bayesian classifier	RGB and multispectral image was taken. Some of the leaves were heavily infected whereas some were healthy. The two images, RGB and multispectral one was fused together, and a 3D map was made. pixelwise methods of classification K-NN and Bayesian Classifier was used to achieve median pairwise classification for both the pathogen and fungus [36].	2011

(*Continued*)

Table 5.1 Highlights the important work of literature published in the last two decades regarding the use of Machine Learning in plant stress phenotyping. (*Continued*)

No.	Type of Stress	Plant Name	Method Used (Algorithm)	Experimental setups	Year
5.	Abiotic (Drought)	Barley	Dirichlet aggregation regression (DAR)	The Dirichlet aggregation regression (DAR) allows high dimensional data to embed into the lower dimensional data spaces. This can, therefore lessen effects due to noise, expose latent relations and do much more. So, this method was employed to find out patterns in the system of the datasets [27].	2012
6.	Biotic Pathogen: *Candidatus* Liberibacter spp.	Citrus [*Citrus sinensis*]	Support Vector Machine (SVM)	Fluorescence imaging spectroscopy was used in case where Huanglongbing disease affected citrus leaves. The images taken were then processed for segmentation by the use of MATLAB, separating pixels into rectangular areas, which were used for feature extraction. The four-matrix which were used were uniformity, contrast, correlation, and homogeneity. The extraction was done by the classifiers [37].	2013

(*Continued*)

Table 5.1 Highlights the important work of literature published in the last two decades regarding the use of Machine Learning in plant stress phenotyping. (*Continued*)

No.	Type of Stress	Plant Name	Method Used (Algorithm)	Experimental setups	Year
7.	Biotic Disease: Powdery mildew	Tomato	Bayesian classifier (BC) and self-organizing map (SOM)	Non-supervised learning and supervised learning comparison was done based on BC and SOM separating diseased plant grown in uncontrolled environment. SOM were created in two groups using K-mean and conditional probabilities. This system showed better results than previously used methods [38].	2014
8.	Abiotic (Drought)	Spinach	Gaussian processes classifier (GPC) and Support vector machine (SVM)	In this work literature, visible light and thermal imagery were combined with the existing machine learning techniques to find out the regions of plants showing a response to soil water deficit. Statistical dimensions were extracted from the images and used to classify between canopies growing in watered and water deficit soil using SVM and GPC and then again by using both classifiers. This work of ML also claims that the technique is highly sensitive up to 97% [6].	2014

(*Continued*)

Table 5.1 Highlights the important work of literature published in the last two decades regarding the use of Machine Learning in plant stress phenotyping. (*Continued*)

No.	Type of Stress	Plant Name	Method Used (Algorithm)	Experimental setups	Year
9.	Various	Various Plants	Similarity incorporating; Affinity propagation clustering algorithm (APC)	Microarray data analysis was done by Clustering algorithm, the gene ontology was taken into consideration. APC algorithm was chosen to analyze the effect of the biological likeness over the dataset. The neighborhood set is used to select repetitive gene from the cluster in this study. The prediction accuracy achieved in this experiment was claimed to be better than the other state-of-the-art methods [39].	2015

(*Continued*)

Table 5.1 Highlights the important work of literature published in the last two decades regarding the use of Machine Learning in plant stress phenotyping. (*Continued*)

No.	Type of Stress	Plant Name	Method Used (Algorithm)	Experimental setups	Year
10.	Abiotic (Drought)	*Verticillium dahliae*	Support Vector Machine (SVM) Linear and discriminant analysis (LDA)	High-resolution thermal and hyperspectral imagery was used in the experiment as a mode of showing the remote sensor data as a useful decision system. The hyperspectral sensor attached to the device had the visible and near-infrared (VNIR) micro-hyperspectral images. Radiometric and atmospheric adjustments were applied to find out spectral influence in the experiment from which an LDA and SVM map was made depending on the spatial distribution. This method was claimed to be a robust and accurate system [8].	2015

(*Continued*)

Table 5.1 Highlights the important work of literature published in the last two decades regarding the use of Machine Learning in plant stress phenotyping. (*Continued*)

No.	Type of Stress	Plant Name	Method Used (Algorithm)	Experimental setups	Year
11.	Biotic (26 diseases)	Various Plants (14 crop species)	Deep Convolutional Neural Network (Alex Net and Google Net)	Both the concepts of Transfer learning and training the data set from scratch were used for three versions of the image data set: colored, grayscale and segmented. The main aspect of Deep Learning to be used for image analysis is the feature extraction part, which extracts useful features in a much efficient manner than traditional methods of image processing and gives much better performance [21].	2016
12.	Abiotic (Drought)	Various Plants	k-nearest neighbor (k-NN), Artificial Neural Networks	A series of different techniques were used for the analysis of drought stress in various plants. The literature reviewed basically discusses the establishment of the ML paradigm on the subject of drought stress. Many of the processes are extensively discussed i.e., Time of light, stereo vision, other contact and non-contact measurement systems. One important system, hyperspectral imaging of the plants, among many others, were done to form a histogram in the analysis of vitality [40].	2016

(*Continued*)

Table 5.1 Highlights the important work of literature published in the last two decades regarding the use of Machine Learning in plant stress phenotyping. (*Continued*)

No.	Type of Stress	Plant Name	Method Used (Algorithm)	Experimental setups	Year
13.	Biotic and abiotic	soybean	Framework based on convolutional neural networks (CNNs)	The authors in this study present one of the most avid stress detection systems as they claim the framework developed by them can identify and rank a set of 8 different stresses (abiotic and biotic) affecting the leaves of soybean. Along with that, they presented a brief mechanism of selection and extraction of the feature maps that can point to visual symptoms in a way, that is quite similar to humans [41].	2018

(*Continued*)

Table 5.1 Highlights the important work of literature published in the last two decades regarding the use of Machine Learning in plant stress phenotyping. (*Continued*)

No.	Type of Stress	Plant Name	Method Used (Algorithm)	Experimental setups	Year
14.	Biotic Diseases: (rust, powdery mildew, blight, bacterial, viral, etc.)	Various Plants	Convolutional neural networks (CNNs)	In this study, the authors present a rapid image-based disease assessment network. The network also emphasizes on the severity of the diseases apart from just relying on the symptoms. It is known as "Plant Disease Diagnosis and Severity Estimation Network" The authors claim that this is the only computer-aided way that simultaneously estimates severity, marks diseases, while also recognizing species, and classifying diseases for plants using deep learning. The dataset used by the authors were taken from a synthetic dataset [42].	2019
15.	Biotic	Coffee	Convolutional neural networks (CNNs)	Using computer vision and deep learning methods, the authors of this study worked exclusively in the domain of biotic stresses in case of coffee plantation. They employed a multi-tasking system based CNNs. Moreover, they took both the symptoms and severity parameter into account. They claimed their results showed more than 90% accuracy [43].	2020

(*Continued*)

Table 5.1 Highlights the important work of literature published in the last two decades regarding the use of Machine Learning in plant stress phenotyping. (*Continued*)

No.	Type of Stress	Plant Name	Method Used (Algorithm)	Experimental setups	Year
16.	Biotic Disease: Rhizoctonia root rot Pathogen: *Rhizoctonia solani*.	Sugar beet	Recursive feature elimination method (RFE)	Hyperspectral imaging with visible light and near-infrared range of the sugar beet plant has been done in this case. The experiment was done in assessment of the fungus *Rhizoctonia solani*. More than hundred spectral features of different time series were selected in two levels by RFE. The best classifier was obtained by a square modelling. The experiment proves that, under the experimental conditions rapid detection of indirect symptoms of *Rhizoctonia* root rot in sugar-beet plants is possible [44].	2020
17.	Biotic	Espeletia	Analysis of occurrence, GLCM, Fourier transform, maximum likelihood classification and SVM	Unmanned aerial vehicles (UAV) acquired the images of Espeletia plant, from Colombia, was used in the assessment of biotic stress affecting the area. This has been done so as to perform environmental monitoring in the area [45].	2020

(*Continued*)

Table 5.1 Highlights the important work of literature published in the last two decades regarding the use of Machine Learning in plant stress phenotyping. (*Continued*)

No.	Type of Stress	Plant Name	Method Used (Algorithm)	Experimental setups	Year
18.	Biotic Pathogen: *Bursaphelenchus xylophilus*	Maritime pine (*Pinus pinaster*)	Random Forest classification	In this experimental assessment UAV based flight hyperspectral data was collected over a span of two years. The algorithm was seen to provide accuracies higher than 0.91 in test data. The study claims that the early detection of possible even before visible symptoms appears [46].	2020

5.6 Types of Plant Stress

Plants respond to stress in various ways. For example, drought-stressed *Arabidopsis thaliana* shows the wilting of leaves [7]. *Solanum lycopersicum* i.e. Tomato, on the other hand shows stunted growth, droopy leaves and it has also been seen to limit root growth when put in drought-like condition [47]. However, in the cases of biotic stress, the visible traits are much easily observable than those of the abiotic stress. for example, diseases like blight, rust, etc. causes visible necrosis which changes color of the leaves from green to yellow or red. Whereas, the droopy, drying up leaf caused by drought or salinity occurs much later when the plant is in severe stress. One example of biotic stress is common maize rust, which is mainly caused by the fungus, *Puccinia sorghi*. In this case, reddish brown spots (pustules) are formed in the leaf lamina of *Zea mays*. Rust is caused mainly by a fungus named *P. sorghi* [48].

When multiple types of stress acts on the same plant at the same time, often it causes adverse condition, owing to either death or impairments of the plant parts. However, on rare occasions, it has also been seen that one type of stress acclimatizes the plant for another type of stress. This is another system of classification of stress where based on the number of correlated factors, the stresses are grouped (like single stress, multiple individual stress, and combined stresses) [49]. However, this is beyond the scope of the discussion of this chapter, so we'll keep the discussions majorly limited to the biotic stress classification and identification.

5.6.1 Biotic Stress

Biotic stress is primarily caused by a living organism(s), where almost all types of organisms, irrespective of their size or numbers are capable of contributing to the stress. Organisms like microbes, fungi, nematodes, insects, viruses and weed are commonly known to cause significant biotic stress. These can cause either partial damage to the plants that they can recover from, or extensive damages that might cause the death of the plant [50].

Biotic stress is caused by microorganisms, they are easily visible on plant body (fungal, bacterial and viral plant diseases). Some diseases can also occur due to the action of parasites. Insects and animals, however, lead to physical damages through predation, whereas weeds or unwanted plants cause stresses as a part of competition with plants for light, nutrients, space, etc. [1]. Fungus and bacteria are the two major causes of microorganism

induced stress. We will discuss about both of them with the context of using ML algorithm on their detection and diagnosis.

5.6.1.1 Fungal Pathogen

One major example is the infamous Magnaporthe grisea, a.k.a. the rice blast fungus. It's a fungus from the class Sordariomycetes, which affects all the parts of the rice plant. It affects the node, leaf, parts of panicle, neck and leaf sheath. It grows from the spores and has a whitish texture. The plant part which it affects initially shows white, grey-green abrasions or small spots with dark green border. A necrotic reddish-brown border also develops around it, giving it a distinctive appearance [51]. This distinction is easily observed under visible light, which can be recognized using ML algorithms. The methodology of the system depends on use of the K-Nearest Neighbor (KNN) or Artificial Neural Networks (ANN) Classifiers. Pre-Processing is done before the model training then the feature extraction is done based on the requirement. It is often seen that KNN is frequently outperformed by ANN in specificity, accuracy and precision. ML is an excellent and established system for analysis of the rice blast disease [52]. Systems like this can be trimmed, optimized, trained and extrapolated as per the requirement to fit the needs of the analysis for stress in different plants.

5.6.1.2 Bacterial Pathogen

Bacterial diseases of commercially important crops is a problem for most countries in the world. Bacterial blight of rice is caused due to *Xanthomonas oryzae* pv. oryzae. It's a disease, which accounts for major crop loss almost every year in various parts of the world where rice cultivation is prevalent. It is known to affect in places that have weeds or increased growth of already diseased plants. It occurs in both tropical and sub-tropical climatic conditions, particularly in watered and rainfed low-lying crop fields. In general, the disease prefers temperatures at 25–34 °C, with relatively high humidity [53, 54].

Neural networks are one of the technologies that has been used recently for crop protection. The strategies employed are explained as follows. firstly, the leaves are collected from the fields and are checked for blast, brown spots, bacterial blight, and sheath rot diseases. The images are then pre-processed and further segmented. Then classification of the exact disease-causing pathogen is identified by using a deep neural network

(DNN) and an optimizing computer algorithm, which is further confirmed by wet lab tests. The feature extraction for the same is done by a suitable matrix, using color and texture features to arrive into the output data. The system was also evaluated with an artificial neural network (ANN) and others. But it was seen that, the new system showed better results with DNN rather than the ANN. In this chapter, we will see how the steps are selected and followed in order to use machine learning for identification of bacterial disease [55].

5.7 Implementation I: Numerical Dataset

5.7.1 Dataset Description

The dataset [56] comprises of 307 instances in total and the rows for the missing values have been imputed. The dataset deals with 35 numerical features/parameters responsible for causing diseases in soybeans. The diseases in the soybean are further divided into 19 classes, based on these features.

Method used: We have used PCA (Principal Component Analysis) in order to extract out the most important features and highlight the relevancy of these features with respect to the final outcome of the classification. We have tried to explain how the features influences the first principal component (PC1) using an example in Figure 5.2. Though we have shown only how PC1 can be created in this case, similar steps are also followed for creating the other principal components. PCA is one of the most commonly used dimensionality reduction techniques which helps in minimizing information loss by choosing the components with maximum variance.

This method is typically used in cases where the feature space is very large. PCA creates new independent variables (by combining the original inputs) and the feature and we have to look out for the components in which the maximum information for the classification is retained, i.e. the ones with the highest variance Using ML technologies, this visible distinction is segregated by Pixel reading based on Image.

For the classification task, we have used Random Forest classifier to predict the correct plant disease depending upon the features which contribute most for the most important principal components. We have a total of 307 instances in our dataset, out of which we have imputed the ones with missing values by dropping the rows. Out of the remaining 265 instances, we have used 80% is used for training our ML model and 20% for testing.

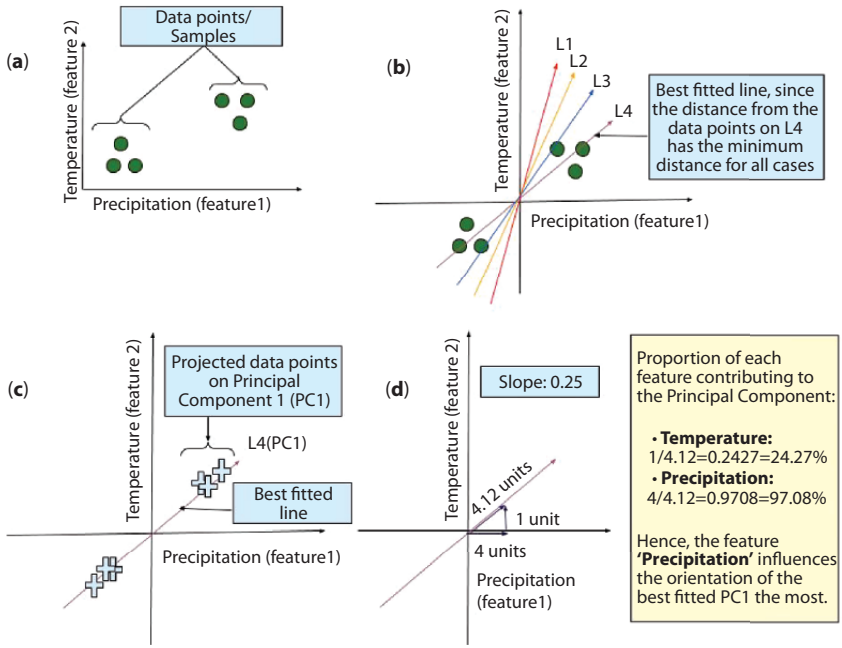


Figure 5.2 Depicts the different steps in which PC1 is created, by considering an example. (a) Shows the distribution of the data points (which we have taken as the health plant samples in this case) on the plot with respect to the two features: Temperature and Precipitation. (b) Shows how the PC1 is selected. The data points are shifted with respect to the origin. Lines passing through the origin are drawn in various orientations, until the best fit line (based on the minimum distance from all the data points are found). (c) Shows the projection of the data points on the best fitted line. (d) Shows up to what degree, these features (Precipitation and Temperature) influence the orientation of PC1, so that it can be considered as the best fitted line. We have assumed that the slope is 0.25 for this case. Going by this assumption, we have found that Precipitation influences the health of the plant samples in this case for almost about 97.08% and Temperature influences it for only about 24.27%

5.7.2 Results

The bar graph below shows the 16 principal components that we have used for our classification task. The graph (in Figure 5.3) depicts the values of the variance corresponding to each of these components, which helps us to determine the most important features.

The 16 components that we have selected to consider for the classification task contain the maximum information regarding the classification task. Each of the 35 features have played a role in construction of these components, but we have tried to shortlist the ones which have contributed

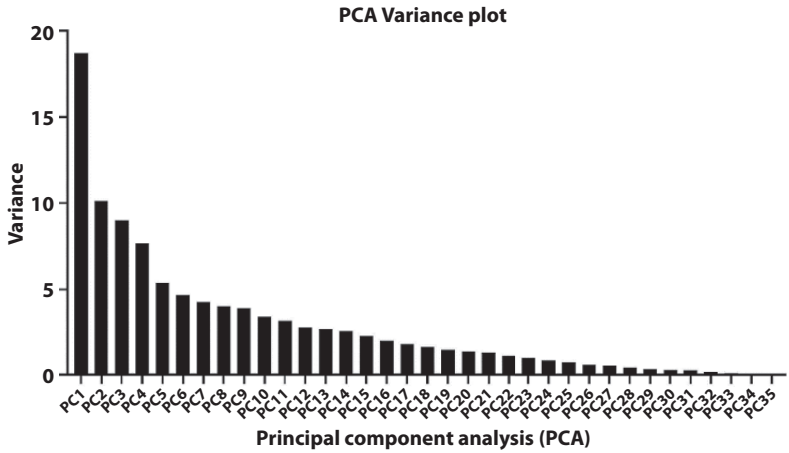


Figure 5.3 Shows the screen plot which depicts the variance of each of the principal components.

the most for the variance, based on the Eigen values. Here we have shown the features (in Table 5.2) that the first nine components are primarily influenced by, resulting in the maximum variance. We can hence consider them to be the most important features that have been extracted with relation to the classification task of plant stress.

Table 5.2 Depicts the features influencing each of the 9 principal components.

Principal components	**Feature primarily influencing each of the components**
Principal Component: 1	Fruit spots
Principal Component: 2	Season
Principal Component: 3	Stem Cankers
Principal Component: 4	Area damaged
Principal Component: 5	Canker lesion
Principal Component: 6	Crop history
Principal Component: 7	Germination
Principal Component: 8	Severity
Principal Component: 9	Precipitation

The 16 components are decided by observing the variance values from the graph. We have also cross checked the performance of the Random Forest classifier by selecting these 16 components and running the classifier model on it. We have seen that the accuracy keeps on increasing and reaches a maximum of 79.24% till 16 components are used. Increasing the components even further causes the accuracy to decline. Hence, we have used 16 components for our classification task.

5.7.3 Discussion

The important features are the ones which influence the principal components the most. Each of these principal components can be further explained by the values of the Eigen vectors, where higher the Eigen value, more important is the corresponding feature. These Eigen values influence the components to obtain the variance in PCA. We have obtained the most important features in this way. Nine of these extracted features that we have discussed in the table above are also biologically relevant in contributing to the diseases that occur in case of soybean plants. We have discussed the importance of these nine features related to the plant stress from a biological point of view as follows:

Fruit spots: Fruit spots are one of the major indicators of the various diseases which cause biotic stress in plants. They are mostly caused by fungal pathogens, out of which, genus *Botryosphaeria* [57], *Anthracnose* [58] and *Magnaporthe* are well known. The colors and size of the spots may vary plant to plant which is a true and accurate indicator for stress.

Season: The season is one of the most vital parameters for ensuring the growth of crops. From skilled farmers to agricultural scientists to plant biologists all have monitored the effect of season, and changes caused by it in plant growth, development and survival. From early to modern literature suggests that effect of season causes direct change in plant stress [59].

Stem Cankers: Stem canker is one of the diseases of various species of plant including but not limited to soybean. Tomato, guava, potato, tea, cocoa, etc. also have been known to get affected by *Phytophthora* sp., the causative organism for stem canker. The *Phytophthora* is a species that is quite versatile and also known to cause blackening of pods of legumes in some cases. It is a group of the opportunistic pathogens, which is quite diverse [60].

Area damaged: The area damaged of the plant is a quite visible and well-known way of assessment that has been used for centuries for stress assessment.

Apart from these, the other factors, as seen from the result, play quite a vital role in determining biotic stress onset and its assessment. Factors

such as, Canker lesion, Crop history, Germination, Severity, Precipitation also have important aspects in the detection and its assessment. However, the factors discussed, significantly contribute more with respect to the rest mentioned here.

5.8 Implementation II: Image Dataset

5.8.1 Dataset Description

The dataset [61] is a multivariate dataset that contains 120 instances in total and the rows for the missing values have not been imputed. The dataset deals with three types of diseases caused by pathogens (harmful microorganisms) responsible for causing different diseases in rice. The diseases in the rice are namely rice blast, rice bacterial blight and brown spot of rice.

5.8.2 Method Used

We have used the ORB (Oriented FAST and Rotated BRIEF) algorithm in this case to extract the most important key points using the FAST and BRIEF algorithm. This algorithm has been explained elaborately in the section above. In an image, features are basically the vector representations, that help to detect edges, corners and other parts of an image. This helps in motion detection, image alignment, segmentation, etc. In this case, we have used this feature extractor to extract the important key-points or features from each of the images and have compared the trained image with a query image that our model has never previously encountered. The dataset (described in the previous sub-section) has three types of images of the diseases. Namely, Leaf smut, Bacterial leaf blight and Brown spot. We have taken images from each of these classes of diseases, one for training and one for testing. We have extracted the key-points from each of these images and have tried matching them using these features. We have also used a Brute Force Matcher to match these features amongst the two images based on distance.

5.8.3 Results

5.8.3.1 Results of ORB Feature Extraction and Brute Force Matching

5.8.3.1.1 Leaf Smut on Rice Leaves

Rice leaf smut is a widely distributed, airborne disease of rice. It is caused by fungus *Entyloma oryzae*, and generally occurs late in the growing

season, causing formation of the black or black-ish smut on the leaf surface. They often appear as distinguishable, angular or elliptical dots and causes necrotic damage on the leaf [53]. Figure 5.4 shows the extracted features in case of Leaf Smut disease using ORB algorithm.

- Number of key-points identified on training image: 414
- Number of key-points identified on testing image: 471
- Number of key-points matched between training and testing images: 125

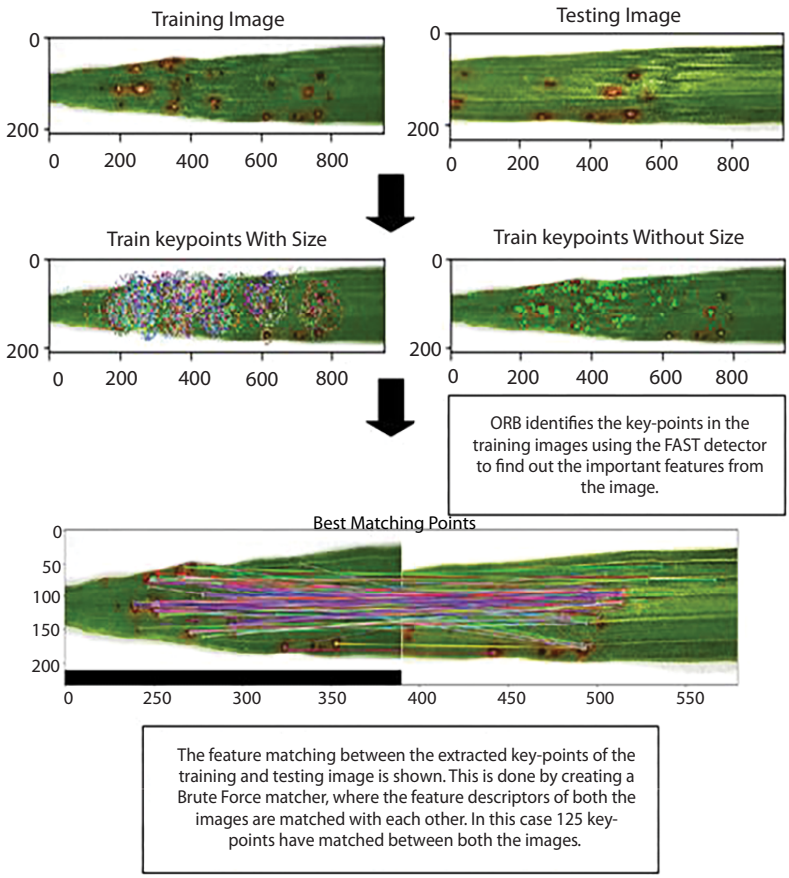


Figure 5.4 Shows the steps followed in extraction of the features by ORB in case of Leaf Smut disease in rice.

5.8.3.1.2 Brown Spots on Rice Leaves

As the name suggests, Brown Spots a condition that appears when a number of brown, long ellipsoidal spots keep appearing in rice leaf. It is caused by a fungus named *Cochliobolus miyabeanus* and is known to affect the crop from seedling to harvest stage. It is a very serious disease of rice and is a major reason for crop loss in many countries [62]. Figure 5.5 shows the extracted features in case of Brown Spots disease using ORB algorithm.

- Number of key-points identified on training image: 481
- Number of key-points identified on testing image: 480
- Number of key-points matched between training and testing images: 127.

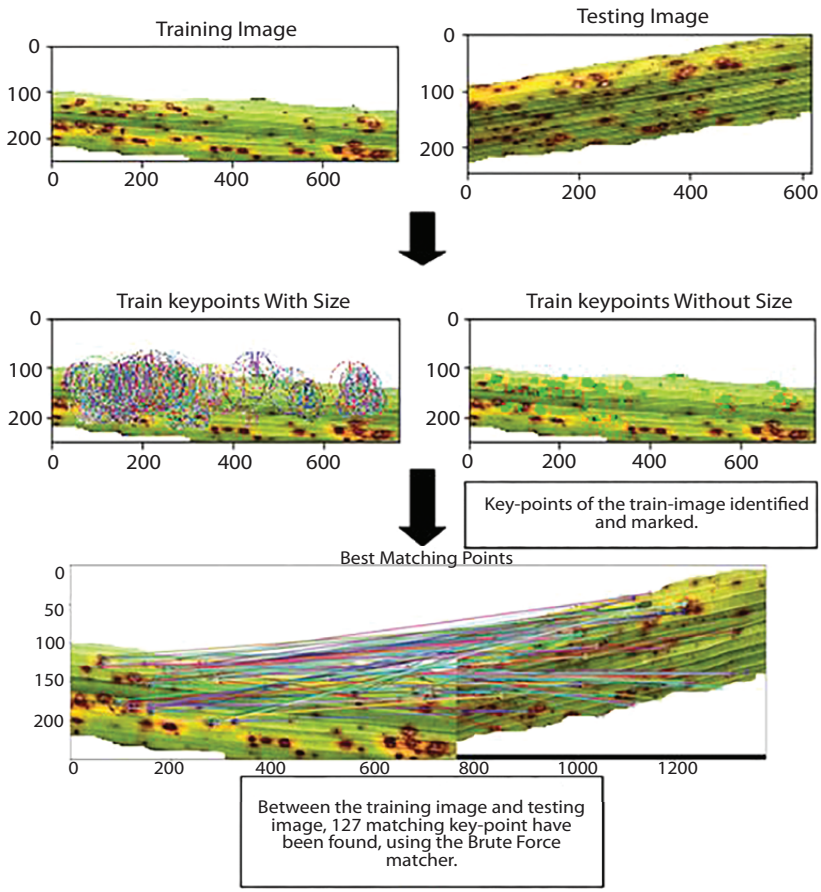


Figure 5.5 Shows the steps followed in extraction of the features by ORB in case of Brown Spots disease in rice.

5.8.3.1.3 Bacterial Leaf Blight on Rice Leaves

Bacterial blight of rice is caused by a proteobacteria known as Xanthomonas oryzae. It occurs at the growth stages of rice and the phenotypic changes include acute weltering of a part or whole of the leaf. X. oryzae invades plants either by pore or by wounds. Damage is generally seen from the tip of the leaf [63]. Much like brown spot, bacterial blight is also a major damaging disease of rice. An early occurrence of the disease accounts for higher yield loss. Figure 5.6 shows the extracted features in case of Bacterial Leaf Blight disease using ORB algorithm.

- Number of key-points identified on training image: 500
- Number of key-points identified on testing image: 500
- Number of key-points matched between training and testing images: 66.

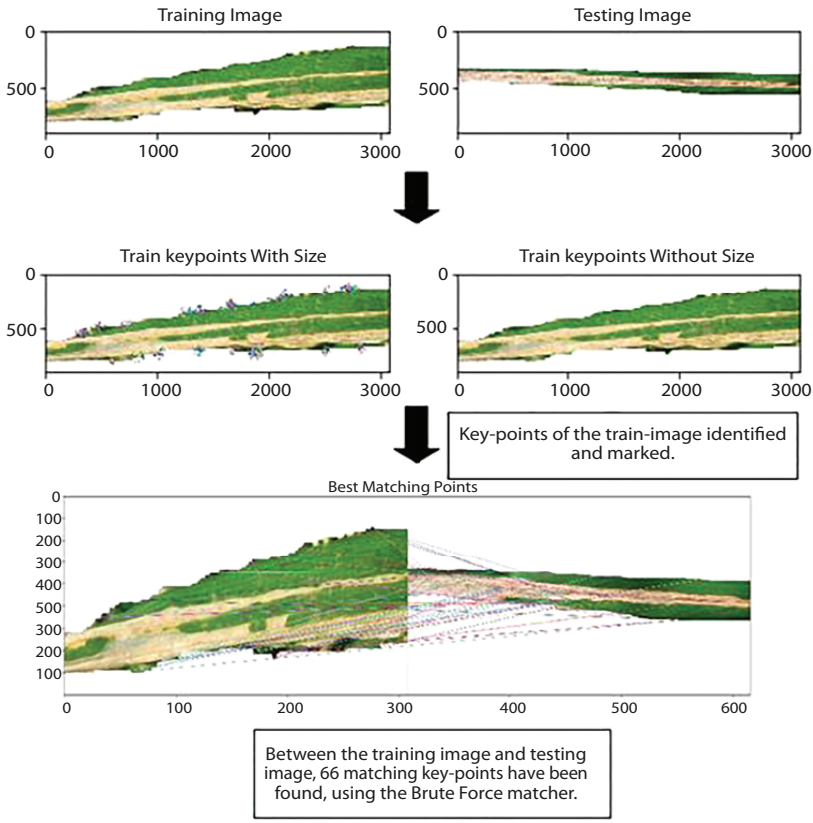


Figure 5.6 Shows the steps followed in extraction of the features by ORB in case of Bacterial Leaf Blight disease in rice.

5.8.3.2 Color Histogram Comparison: Using Correlation Method

We have plotted the intensity histogram (using BGR color channels) of the rice leaves with the diseases (as shown in Figure 5.7) and tried to compare the images of the leaves of similar categories and of separate categories. Based upon these comparisons, we have tried to find out the correlation coefficients to understand if our results turned out as we would have expected it to be.

- Value of correlation coefficient for the first pair of images (Both the images are of Bacterial Leaf blight): 0.9707
- Value of correlation coefficient for the second pair of images(Bacterial Leaf bight Vs. Brown Spots): −0.016
- Value of correlation for the third pair of images (Bacterial Leaf blight and Leaf Smut): −0.0346.

5.8.4 Discussions

We have used the ORB feature extraction technique to extract out the most important features and have used a Brute Force matcher to match between the different key-points of images with similar morphological features. One of the main reasons for choosing ORB algorithm for our work was that it is rotation invariant and scale invariant. Images with different orientations and scales can be used for comparison. Though the computational cost of using ORB is also very low, there is a need to check for a better distribution of the key-points over the surface of the images by comparing with other algorithms with SIFT and SURF.

The information extracted from color histogram, by using the Blue (B), Green (G) and Red (R) color channels, help us to understand the intensity of each of these colors in the image. As our distinction in case of this particular image analysis task was based on the phenotypic features, 'color' becomes an essential feature based on which the distinctions are made. Hence trying to find the correlation between the two images based on the color histograms has been an important measure in comparing the images between different types of diseases on the leaves of rice plants. Our results, in this case, have been consistent with our expectations. The value for the correlation coefficient is the highest (0.9707) in cases where two images of the same leaf disease (bacterial leaf blight) have been compared. But when we tried to show the comparison between the images of two different diseases (bacterial leaf blight and brown spots; bacterial leaf blight and leaf smut), we obtained negative correlation in both the cases. This clearly shows us the importance of using color as a feature descriptor while

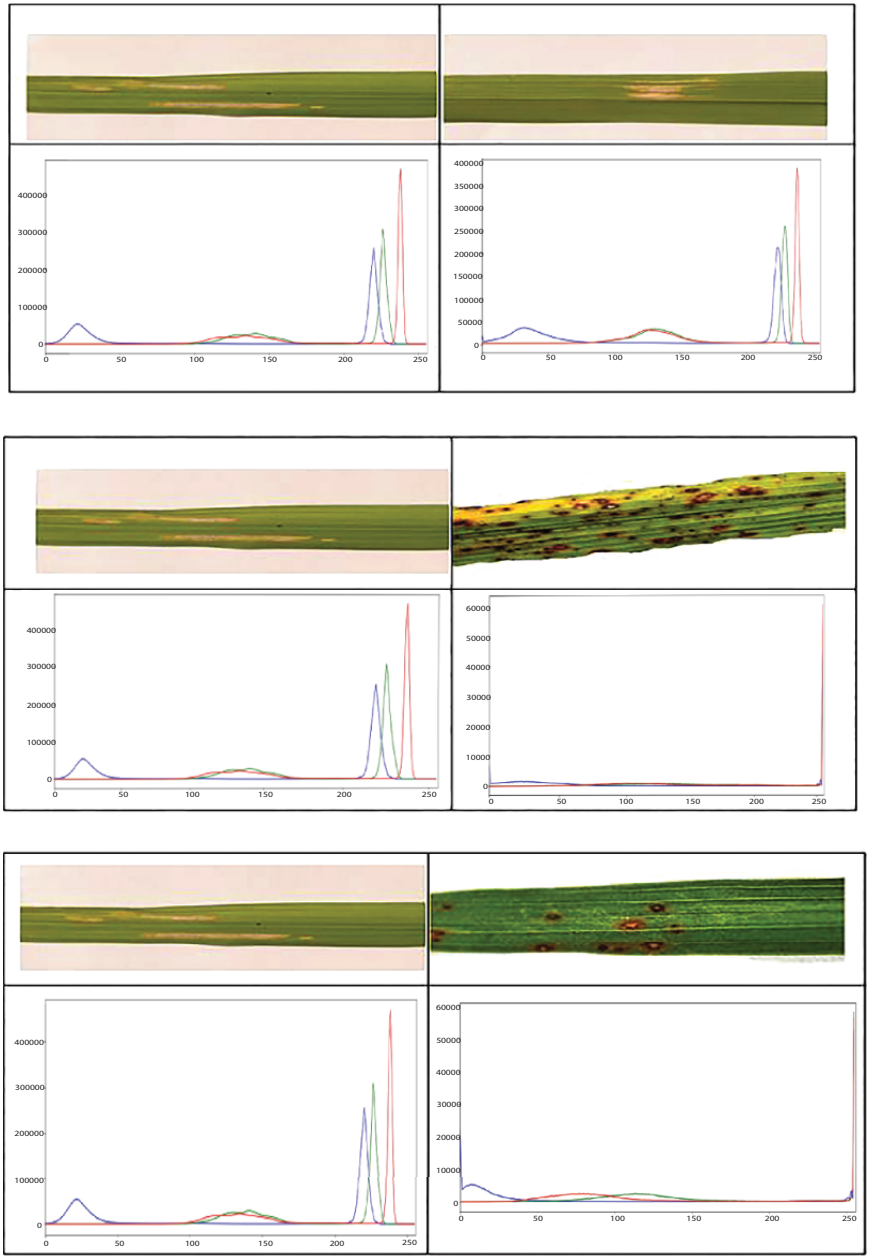


Figure 5.7 Color histogram comparison between three pairs of images: images of both the leaves infected with Bacterial Leaf Blight (first image from the top), images of rice leaves with Bacterial Leaf Blight and brown spots (second image from top), images of leaves with Bacterial Leaf Blight and Leaf smut (third image from top).

comparing between two images, when the distinction is largely based on the morphological features.

5.9 Conclusion

In this chapter, we have discussed some important feature extraction techniques that are most commonly used in case of numerical and image datasets. We have mainly dealt with plant stress in this case and have tried to understand the important parameters that contribute to the stress using ML techniques. We have discussed the important works done in this field in an extensive literature survey. We have also carried out two-fold implementation using the feature extraction techniques—PCA in case of numerical dataset and ORB and color histogram analysis in case of image dataset. In case of PCA, we have seen that the features primarily influencing the first few components are in fact biologically relevant to the associated diseases too. In case of the implementation with image dataset, we have seen that color histogram analysis produces a much better result in case where the training image is compared with the query image. This can be because of the characteristic difference in color (like brown spots on leaves) associated with certain diseases. Hence, color becomes an important feature that can help in classification of plant diseases. We have also experimented with ORB feature extraction technique, which tries to extract the key features of each of the images irrespective of the scale of the pictures or the orientation in which these pictures are taken. We have also used a brute force matcher to compare between the training and the query images which uses these key points for comparison.

Deep learning techniques are becoming very popular these days and work as excellent tools for feature extraction. Convolutional Neural Network (CNN) [64], a very popular deep learning model, uses convolutional layers to extract the most significant features present in images prior to classification. It also results in achieving higher accuracy in case of classification using the lower dimensional dataset. Autoencoders can be used for feature extraction for obtaining even better results with the final classification task [65]. These techniques can be used in future with these datasets to get even better results.

Dimensionality reduction is a very important topic to focus on, as in case of real world problems, there are a lot of features associated with a particular disease. Computing the accuracy of a classifier model based on all of these features is computationally very expensive. Hence, extraction of the most important features contributing to the classification has become very

important. Tech giants like Intel are working on developing ML-enabled platforms merged with the Internet of Things (IoT), to gather real time data from the fields to monitor the crops. This concept is known as Precision Agriculture, and this field is gaining rapid impetus owing to the huge amount of data that can be collected and processed quickly. Applications such as Google Lens are also becoming widely popular in allowing the farmers and gardeners to take informed decisions based on the visual similarity search. The ML models that work behind such applications are being constantly updated to increase the accuracy of the results. In case of quality assessment, work has also been done in predicting quality of rice grains using computer vision techniques to analyze the texture of the grains [66]. So in order to create such smart farming platforms which can provide us with even better prediction results, it is important to constantly develop better ML models that can handle the problems related with big data analysis. With smart farming options becoming easily available through user-friendly platforms like smart-phones, the field of agriculture is on its way to get digitally revolutionized.

References

1. Mosa, K.A., Ismail, A., Helmy, M., Omics and system biology approaches in plant stress research, in: *Plant stress tolerance*, pp. 21–34, Springer, Cham, 2017.
2. Zhao, W. *et al.*, Different pathogenicities of Rice stripe virus from the insect vector and from viruliferous plants. *New Phyto.,* 210, 1, 196–207, 2016.
3. Rahman, M.-u. *et al.*, Genetics and genomics of cotton leaf curl disease, its viral causal agents and whitefly vector: A way forward to sustain cotton fiber security. *Front. Plant Sci.,* 8, 1157, 2017.
4. Su, T. *et al.*, Dynamics of peroxisome homeostasis and its role in stress response and signaling in plants. *Front. Plant Sci.*, 10, 705, 2019.
5. Bock, C. *et al.*, Plant disease severity estimated visually, by digital photography and image analysis, and by hyperspectral imaging. *Crit. Rev. Plant Sci.,* 29, 2, 59–107, 2010.
6. SeA, R. *et al.*, *Automatic Detection of Regions in Spinach Canopies Responding to Soil Moisture Deficit Using Combined Visible and Thermal Imagery*, PLoS ONE 9(6):e97612, San Francisco, California, 2014.
7. Yao, J. *et al.*, Phenotyping of Arabidopsis drought stress response using kinetic chlorophyll fluorescence and multicolor fluorescence imaging. *Front. Plant Sci*., 9, 603, 2018.
8. Calderón, R., Navas-Cortés, J.A., Zarco-Tejada, P.J., Early detection and quantification of Verticillium wilt in olive using hyperspectral and thermal imagery over large areas. *Remote Sens.*, 7, 5, 5584–5610, 2015.
9. Mutka, A.M. and Bart, R.S., Image-based phenotyping of plant disease symptoms. *Frontiers in Plant Pcience*, 5, 734, 2015.

10. Pridmore, T.P., French, A.P., Pound, What lies beneath: Underlying assumptions in bioimage analysis. *Trends Plant Sci.*, 17, 12, 688–692, 2012.
11. Lamari, L.J.U.o.M., Winnipeg, R3T2N2, Canada: The American Phytopathological Society, in: *ASSESS 2.0: Image Analysis Software for Plant Disease Quantification*, Department of Plant Science, Winnipeg, 2008.
12. Hughes, D. and Salathé, M.J. An open access repository of images on plant health to enable the development of mobile disease diagnostics, Computers and society, arXiv preprint arXiv:1511.08060, 2015.
13. Dua, D. and Graff, C.. UCI Machine Learning Repository http:// archive.ics.uci.edu/ml. Irvine, CA: University of California, School of Information and Computer Science, 2019.
14. Shaik, R. and Ramakrishna, W.J.P., Machine learning approaches distin- guish multiple stress conditions using stress-responsive genes and identify candidate genes for broad resistance in rice. *Plant Physiol.*, 164, 1, 481–495, 2014.
15. Souza, G.M. and Cardoso, V., Toward a hierarchical concept of plant stress. *Israel J. Plant Sci.*, 51, 1, 29–37, 2003.
16. Weckwerth, W., Integration of metabolomics and proteomics in molecular plant physiology—Coping with the complexity by data-dimensionality reduction. *Physiol. Plant.,* 132, 2, 176–189, 2008.
17. Singh, A. *et al.*, Machine learning for high-throughput stress phenotyping in plants. *Trends Plant Sci.*, 21, 2, 110–124, 2016.
18. Chlingaryan, A. *et al.*, Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review. *Comput. Electron. Agric.*, 151, 61–69, 2018.
19. Rublee, E. *et al.*, ORB: An efficient alternative to SIFT or SURF, in: *2011 International Conference on Computer Vision*, IEEE, 2011.
20. Bay, H. *et al.*, Speeded-up robust features (SURF). *Comput. Vis. Image Underst.*, 110, 3, 346–359, 2008.
21. Mohanty, S.P., Hughes, D.P., Salathé, M., Using deep learning for image-based plant disease detection. *Front. Plant Sci.*, 7, 1419, 2016.
22. Kranz, J., Measuring plant disease, in: *Experimental techniques in plant disease epidemiology*, pp. 35–50, Springer, Berlin, Heidelberg, 1988.
23. Barbedo, J.G.A., Koenigkan, L.V., Santos, T.T., Identifying multiple plant diseases using digital image processing. *Biosyst. Eng.*, 147, 104–116, 2016.
24. Alom, M.Z. *et al.*, A state-of-the-art survey on deep learning theory and architectures. *Electron.*, 8292, 3, 2019.
25. Van Der Maaten, L., Postma, E., Van den Herik, J.J.J.M.L.R, Dimensionality reduction: A comparative. *J. Mach. Learn. Res.,* 10, 66–71, 13, 2009.
26. Sabarina, K. and Priya, N., Lowering data dimensionality in big data for the benefit of precision agriculture. *Procedia Comput. Sci.,* 48, 548–554, 2015.
27. Kersting, K. *et al.*, Pre-symptomatic prediction of plant drought stress using dirichlet-aggregation regression on hyperspectral images, in: *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.
28. Kumar, G. and Bhatia, P.K., A detailed review of feature extraction in image processing systems, in: *2014 Fourth International Conference on Advanced Computing & Communication Technologies*, IEEE, 2014.

29. Alpaydin, E., *Introduction to machine learning*, MIT Press, Cambridge, 2020.
30. Jolliffe, I.T., Principal components in: *regression analysis, in*: *Principal component analysis*, pp. 129–155, Springer, New York, NY, 1986.
31. Perriere, G., Thioulouse, J., Biomedicine, Use of correspondence discriminant analysis to predict the subcellular location of bacterial proteins. *Comput. Methods Programs Biomed.*, 70, 2, 99–105, 2003.
32. Lowe, D.G., Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.*, 60, 2, 91–110, 2004.
33. Kacira, M., Ling, P.P., Short, Machine vision extracted plant movement for early detection of plant water stress. *Trans. ASAE*, 45, 4, 1147, 2002.
34. Foucher, P. *et al.*, Morphological image analysis for the detection of water stress in potted forsythia. *Biosyst. Eng.*, 89, 2, 131–138, 2004.
35. Kaundal, R., Kapoor, A.S., Raghava, G.P., Machine learning techniques in disease forecasting: a case study on rice blast prediction. *BMC Bioinform.*, 7, 1, 485, 2006.
36. Bauer, S.D., Korč, F., Förstner, W., The potential of automatic methods of classification to identify leaf diseases from multispectral images. *Precis. Agric.*, 12, 3, 361–377, 2011.
37. Wetterich, C.B. *et al.*, A comparative study on application of computer vision and fluorescence imaging spectroscopy for detection of citrus huanglong-bing disease in USA and Brazil, in: *Laser Science*, Optical Society of America, FL, 2013.
38. Hernández-Rabadán, D.L., Ramos-Quintana, F., Juk, G.., Integrating SOMs and a Bayesian classifier for segmenting diseased plants in uncontrolled environments. *The Scientific World Journal*, 2014, 13, 214674, 2014.
39. Meng, J., Li, R., Luan, Y., Classification by integrating plant stress response gene expression data with biological knowledge. *Math Biosci.*, 266, 65–72, 2015.
40. Kirongo, A.C., A Review of Image Processing Software Techniques for Early Detection of Plant Drought Stress, *IJCATR*, 5, 6, 376–379, 2016.
41. Ghosal, S. *et al.*, An explainable deep machine vision framework for plant stress phenotyping. *Proc. Natl. Acad. Sci.*, 115, 18, 4613–4618, 2018.
42. Liang, Q. *et al.*, PD2SE-Net: Computer-assisted plant disease diagnosis and severity estimation network. *Comput. Electron. Agric.*, 157, 518– 529, 2019.
43. Esgario, J.G. *et al.*, Deep learning for classification and severity estimation of coffee leaf biotic stress. *Comput. Electron. Agric.*, 169, 105162, 2020.
44. Barreto, A. *et al.*, Hyperspectral imaging of symptoms induced by Rhizoctonia solani in sugar beet: Comparison of input data and different machine learning algorithms. *J. Plant. Dis. Prot.*, 1–11, 2020.
45. Martin, L.D., Medina, J., Upegui, E.J., Assessment of Image-Texture Improvement Applied to Unmanned Aerial Vehicle Imagery for the Identification of Biotic Stress in Espeletia. Case Study: Moorlands of Chingaza (Colombia), 30, 1, 27–44, 2020.
46. Iordache, M.-D. *et al.*, A Machine Learning Approach to Detecting Pine Wilt Disease Using Airborne Spectral Imagery. *Remote Sens.*, 12, 14, 2280, 2020.
47. Cortina, C. and Culiáñez-Macià, F.A.J.P.S., Tomato abiotic stress enhanced tolerance by trehalose biosynthesis. *Plant Sci.*, 169, 1, 75–82, 2005.

48. Zúñiga-Silvestre, C.A. *et al.*, Induced resistance to common rust (*Puccinia sorghi*), in maize (*Zea mays*). *EJFA*,11–18, 2020.
49. Pandey, P. *et al.*, Impact of combined abiotic and biotic stresses on plant growth and avenues for crop improvement by exploiting physio-morphological traits. *Front. Plant Sci.*, 8, 537, 2017.
50. Kranner, I. *et al.*, What is stress? Concepts, definitions and applications in seed science. *New Phytol.*, 188, 3, 655–673, 2010.
51. Zeigler, R.S., Leong, S.A., Teng, P.S., Rice blast disease. *Int. Rice Res. Inst.*, 111–134, 1994.
52. Ramesh, S. and Vydeki, D.J., Application of machine learning in detec-tion of blast disease in South Indian rice crops. *J. Physiol.*, 31–37, 2019.
53. Ou, S.H., *Rice diseases*, IRRI, Great Britain, 1985.
54. Ezuka, A. and Kaku, A historical review of bacterial blight of rice. *Bulletin of the National Institute of Agrobiological Resources*, 15, 1–207, 2000.
55. Ramesh, S. and Vydeki, D.J., Recognition and classification of paddy leaf diseases using Optimized Deep Neural network with Jaya algorithm. *Inf. Process Agric.*, 7, 2, 249–260, 2020.
56. Michalski, R.S., Learning by being told and learning from examples: an experimental comparison of the two methods of knowledge acquisition in the context of development an expert system for soybean disease diagnosis. *International Journal of Policy Analysis and Information Systems*, 4, 2, 125–161, 1980.
57. Van Niekerk, J.M. *et al.*, Botryosphaeria *spp.* as grapevine trunk disease pathogens. *Phytopathol. Mediterr.*, 45, S43–S54, 2006.
58. Crouch, J. and Beirn, L.J.F.D., Anthracnose of cereals and grasses. *Fungal Divers.*, 39, 19, 2009.
59. Mcmillin, J.D. and Wagner, M.R., Season and intensity of water stress: Host-plant effects on larval survival and fecundity of Neodiprion gillettei (Hymenoptera: Diprionidae). *Environ. Entomol.*, 24, 5, 1251–1257, 1995.
60. Hansen, E.M., Reeser, P.W., Sutton, W.J., Phytophthora beyond agriculture. *Annu. Rev. Phytopathol.*, 50, 359–378, 2012.
61. Prajapati, H.B., Shah, J.P., Dabhi, V.K., Detection and classification of rice plant diseases. *Intelligent Decision Technologies*, 11, 3, 357–373, 2017.
62. Sunder, S., Singh, R., Agarwal, R.J., Brown spot of rice: An overview. *Indian Phytopath.*, 67, 3, 201–215, 2014.
63. Niño-Liu, D.O., Ronald, P.C., Bogdanove, A.J., Xanthomonas oryzae pathovars: Model pathogens of a model crop. *Mol. Plant Pathol.*, 7, 5, 303–324, 2006.
64. Cascianelli, S. *et al.*, Dimensionality reduction strategies for cnn-based classification of histopathological images, in: *International Conference on Intelligent Interactive Multimedia Systems and Services*, Springer, 2018.
65. Wang, Y., Yao, H., Zhao, S.J.N., Auto-encoder based dimensionality reduction. *Neurocomputing*, 184, 232–242, 2016.
66. Panigrahi, J. *et al.*, Rice Quality Prediction using Computer Vision, in: *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, IEEE, NSW, Australia, 2020.

6

Gene Selection Using Integrative Analysis of Multi-Level Omics Data: A Systematic Review

S. Mahapatra* and T. Swarnkar

Department of Computer Application, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, India

Abstract

Technological advances in the DNA microarray technology allow the researchers to analyze the expression levels of thousands of genes simultaneously in order to understand the genetic etiology behind a disease. But all the genes in the set are not relevant. Gene selection is the process of identifying a few informative genes that achieve the best performance for the respective disease of interest. Emerging high-throughput technologies produce abundance of omics data at multiple levels of biological systems - from genomics to epigenomics, transcriptomics, proteomics and metabolomics. Traditional approaches of gene selection focuses on independent analysis of biological data at different levels. Independent analysis of omics data may leads to many false positives while identifying reliable and robust biomarkers. Interdependency existing across diverse biological levels results in molecular complexity in disease pathogenesis. Integrative analysis has emerged as a gateway to integrate multiple data artifacts from the same objects which can overcome any misleading of results due to single omics data type analysis. In this article we reviewed different existing integrative approaches of gene selection using various supervised and unsupervised machine learning techniques. We also explored the shortcomings of traditional approaches and investigated the computational aspects and open challenges of various integrative approaches for gene selection.

Keywords: Gene selection, integrative analysis, multi-level omics data, unsupervised omics data integration, supervised omics data integration

**Corresponding author*: saswatimahapatra2015@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (145–172) © 2021 Scrivener Publishing LLC

6.1 Introduction

It has been an open quest for many researchers to understand the genetic basis of complex traits. Technological advances in high throughput technology generate rapid dissemination of data from multiple levels of biological systems—including DNA sequence data [1], RNA expression levels [2], methylation patterns [3], proteomics [4] and metabolomics [5]. Consequently, this results in a plethora of data associated with these "-omes" also known as "omics" data [13], summarized in Figure 6.1. This broadens up the horizon of research to explain complex cellular mechanisms, identify key player in molecular mechanisms and predict diagnostic conclusions from large biomedical datasets.

DNA microarray technology allows the researchers to detect and analyze the expression levels of thousands of genes simultaneously [6, 7]. Although gene expression profiling has been used to study infectious and immunological diseases, gene expression analysis predominantly focuses on the study of cancer. It has given the scientists, a remarkable opportunity to analyze the expression pattern or genetic signature of thousands of genes concurrently in a solitary platform. But all the genes in the microarray experiment are not relevant for any particular disease of interest [8]. Thus a group of genes need to be identified according to their expression patterns. Gene selection or biomarker discovery is the process of identifying

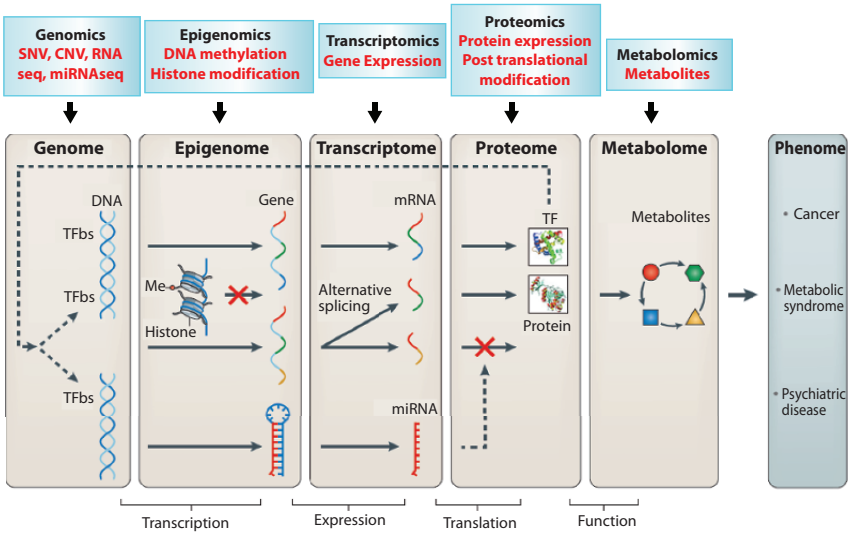


Figure 6.1 Multiple levels of omics data in biological system, from genome, epigenome, transcriptome, proteome and metabolome to phenome [13].

a sub set of relevant and informative genes from the original set of genes with diagnostic and prognostic capability [9]. Their discriminative ability allows classifying samples into disease categories (diagnostic), while their predictive power enables assessing the cause of a disease and discovery of new therapy (prognostic).

Considering the data matrix that represents the microarray data, columns in the matrix represent genes or features where as rows represents samples. Each entry of the matrix explains the expressive power of a particular gene in a sample or time point. The major limitation in the microarray is its large dimension [10, 11], i.e. the number of genes/features are significantly high than the number of samples. Thus gene selection can be redefined as a task of feature selection, i.e. identification of a group of revealing features out of large number candidates there by eliminating the noisy and redundant ones. The set of discriminative features can be used to finalize classifier model with enhanced prediction accuracy, comprehensibility, and adaptability and generalization ability. It also reduces computational complexity and storage, provides a faster and cost-effective model, and plays an important role in knowledge discovery [12].

6.2 Approaches for Gene Selection

Feature subset selection works by selecting a set of relevant features which can be used for model construction. In machine learning, many supervised and unsupervised approaches for gene/feature subset selection have been proposed considering the single level of biological data as an isolated component. Supervised technique utilizes the data with known class labels in the feature evaluation procedure. Nevertheless the supervised technique is the most commonly used; it relies on the labeled data. However, as the labeled data is obtained from peripheral processes and techniques, it may be wrongly, attributed unpredictable and costly as well [14]. As a result the learning process may face the problem of over-fitting by unintentionally removing the relevant features and selecting the irrelevant ones. Unsupervised techniques are unassisted by the labeled data. It is unbiased as it doesn't utilize the class labels of the data. The key disadvantage of unguided approach are that it is based on a definite mathematical model irrespective of the data sets used and also it does not take into account any correlation among the relevant features. Hence the resultant subsets may not be optimal for the defined discrimination task [15]. There are multiple reviews focusing on most popular approaches for feature selection in gene expression data [21, 22].

The process of feature selection is a group of search techniques which proposes new subsets and a measuring approach to evaluate and rank the newly created feature subsets. Depending on the evaluation metric used, the feature selection algorithms are of three types: Filter, Wrapper and embedded approaches [16].

Usually to learn and train, Filter methods are used as a precursor step. The feature selection process proceeds by statistically scoring the features, without involvement of any learning algorithm. Statistical measures such as mutual information, Relief based algorithm, Pearson coefficient of correlation, inter/intra class similarity or the scores of significance for individual class/feature group are used for scoring the features. This approach is computationally faster and can provide a generalized solution for many classifiers. The feature subset generated in this approach does not calibrate a predictive model correctly [17]. So it may result in varying prediction performance, when the same feature subset is applied to various classifiers [18].

Wrapper method uses predictive models to select the features, utilizing the classification error rate and prediction accuracy as the selection criteria [19]. This method selects features with best discriminative ability by minimizing the prediction error of that particular classifier. However, the wrapper methods are less general than filter because the features are selected based on a particular learning algorithm. The performance of the same set of features may not be same if another learning algorithm is utilized. Additionally, it is more inclined towards over-fitting in comparison to filter approach since the classifier is invoked repetitively for the evaluation of the feature subsets.

Embedded method takes the advantages of both filter and wrapper approach. It performs feature selection as a part of model construction and uses its properties to guide feature evaluation. LASSO is a most common approach among others for feature selection in multi dimensional data [20]. It constructs a linear model by reducing constantly the regression co-efficient of the feature to zero. Finally, features with non zero regression coefficients are selected by LASSO algorithm. This approach do not invoke the classifier repetitively for the evaluation of each of the feature subsets. Hence the efficiency of this approach is more and it is observed that it is a well-managed computational methodology vis-à-vis wrapper method, with a sustained performance.

Apart from these feature selection can also be outlined as a Hybrid or Ensemble approach. A Hybrid approach is the combination of both filter and wrapper methods which inherits the advantages of both the methods by combining their complementary strengths [42, 43]. Ensemble method is based on different sub sampling strategies to form the groups of feature subsets deploying different feature selectors. Finally a stable feature subset is formed by merging the groups. Hybrid or ensemble approach is a better

technique which is able to calculate the optimal subsets of features based on approximation technique and also offers an improved estimated rank to the features by combining the outcome of various feature selectors [44].

Gene selection acts as a key player in analyzing the different types of omics data at single level to identify the key genomic factors and their associations, explaining the disease risk or other biological insights. Traditionally there is an extensive use of afore mentioned feature selection methods, in application to gene selection. It predominantly focuses on analysis of single level omics data. Majority of the work for gene selection had been carried out in gene expression data deploying the aforementioned methods of feature selection. Mandal *et al.* [23] proposed an improvised minimum redundancy, maximum relevance approach of feature selection for gene expression data, taking the mutual information between a feature and class labels as the measuring criteria for relevance of features. A supervised gene selection method combining locality sensitive Laplacian score (LSLS) and wrapper approach had been proposed in [24]. It worked in two steps. Firstly it incorporated discriminative information into local geometrical structure, by minimizing local within-class information and then maximized local between-class information. A wrapper approach had been proposed by Sharma *et al.* that overcomes the limitations of the traditional feature selection methodologies [25]. Low ranked genes which are generally skipped in the traditional methodologies were used in the feature selection process that enhanced the prediction ability of the classifier. Xiang *et al.* proposed a least square regression framework for feature selection and multi class classification. $L_{2,1}$-norm had been used to generate a robust model for feature selection [26]. A regularized sparse multinomial logistic regression with $L_{1/2}$-penalty had been established [27]. Features were measured sparsely between high and low weight. This method was proved to be significant in reducing the memory requirement for storing the large data matrix. A combination of fuzzy classifiers and kernel machines had been proposed [28]. A fuzzy support vector machine was applied combining filter or wrapper feature selection methods to develop a robust model with higher accuracy. An ensemble framework of fast clustering-based algorithm had been proposed [29] where the whole feature set was clustered using graph based clustering and further a representative feature from each cluster is selected producing a subset of useful and independent features. The prediction accuracy of the identified markers was further validated using machine learning methods. Huang *et al.* [32] and Singh *et al.* [33] proposed hybrid gene selection techniques for prostate and lung cancer microarray data. Huang *et al.* [32] attempted to overcome the high dimensionality of the micro array data by introducing a sparse logistic regression including a linear combination of $L_{1/2}$ and L_2 penalties. Singh *et al.* [33] contributed to

identify informative genes using a hybrid approach combining Spearman's Correlation Coefficient (SCC) and Genetic Algorithm (GA). Adrian analyzed the feature value space instead of feature space and proposed a rule based algorithm to identify detect relevant, non-redundant and interacting genes in gene expression data [34]. Songyot proposed a hybrid filter wrapper gene subset selection algorithm employing interaction information to rank candidate genes [35]. Wang *et al.* [37] proposed a weighted group lasso cancer gene selection approach for micro array data, where the importance of genes and groups is presented in terms of joint mutual information. Many hybrid gene selection pipelines had been proposed in [38–41] employing nature inspired and filter search techniques to produce optimal feature subsets of genes. Study reveals epigenetic modifications play a key role in regulating the gene expression [30]. DNA methylation and histone modifications are two major mechanisms of epigenetic regulation. An ensemble of feature selectors of Multi-category Receiver Operating Characteristic (Multi-ROC), Random Forests (RFs) and Maximum Relevance and Minimum Redundancy (mRMR) methods had been incorporated [31] to select DNA methylation markers that simultaneously distinguish the lung cancer subtypes. A supervised case-control analysis of RNA sequence data had been carried out using Random Forest, Variational Autoencoders and regressors and identified subsets of genes associated with survival [35].

Nevertheless, single level analysis of biological data identifies biomarkers that provide a new insight in detecting disease in individuals, classify the disease type, and predict the response of therapy and so on. The major challenges in the analysis of micro array data using various machine learning methodologies are [45, 46]: (i) High noise: Sporadic noise and skewness present in the gene expression data lead to distorted scientific outputs and cost escalation. It may lead to erroneous prediction of phenotypes which ultimately impact disease therapy. (ii) High dimensionality with small sample size: The number of genes in the gene expression data is massively high in comparison to the number of samples. Among all, a large portion of genes are irrelevant, only a small fraction of genes are significant for disease of interest. These mass of improper genes become noise in the microarray data which ultimately significantly affects the performance of machine learning algorithms. (iii) Batch effect: A systematic processing noise is induced to multiple batches of microarray samples which affects the integrity of the generated microarray data. The microarray experiment is mostly limited with erroneous measurement which is affected by the factors including experimental principle, data collection standards and personal differences. (iv) Low reproducibility of significant biomarkers: Key biomarkers marked by internal validation rarely overlap with other

research group. Thus the restrictive single omics data type analysis may lead to generation of many false positives, which contributes to partial exploration of genetic etiology of many complex traits and biological networks. However, interdependency existing across diverse biological levels (genomics, epigenomics, transcriptomics, proteomics and metabolomics) permits to reveal hidden genotype–phenotype relationships, with the aim of better understanding the onset and progression of many diseases, and identifying new diagnostic and prognostic biomarkers. Integrative analysis has emerged as a gateway to integrate multiple data artifacts from the same objects which can compensate for missing or unreliable information in any single omics data type. The chances of false positives decrease with the findings from multiple sources which point to the same gene or pathway.

Ritchie *et al.* defined two different hypotheses explaining the linear and nonlinear molecular interplay among diverse omics data [13]. Hypothesis 1: Any alteration at genome level will bring changes in gene expression at transcriptome, which will lead to deviation in protein expression at proteome resulting in an altered phenotype. This ascertains a linear assumption of disease etiology. Hypothesis 2: Molecular variations at different biological level are nonlinear, interactive and thus complex. Unfavorable molecular alterations at multiple biological levels facilitate the disease progression. The schematic representations of the hypothesis are shown in Figure 6.2. Sequential or simultaneous integration of data at multiple omics level facilitate in assessing the information flow across omics layers and thus help to bridge the gap in understanding the phenotypic variations

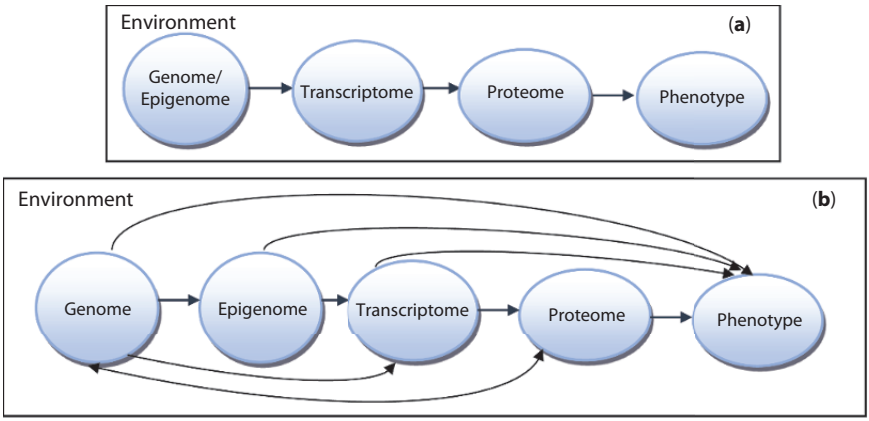


Figure 6.2 (a) Hypothesis 1: Molecular variations propagates linearly in a hierarchy of omic levels and finally results in disease phenotype. (b) Hypothesis 2: Genetic basis of complex traits are by virtue of aggregate molecular variations across all possible omics levels.

in disease [47]. Gene selection using integrative approaches, by virtue of its ability to systematically study the complex biological system, is capable of enhancing the prognostic and predictive accuracy of phenotype.

6.3 Multi-Level Omics Data Integration

Statistically data integration approaches are broadly categorized as (1) horizontal integration and (2) vertical integration. Horizontal integration aggregates same types of omics data across multiple studies where as vertical integration combines different genomic data from the same set of samples [48]. The horizontal and vertical integration pipeline is illustrated in Figure 6.3.

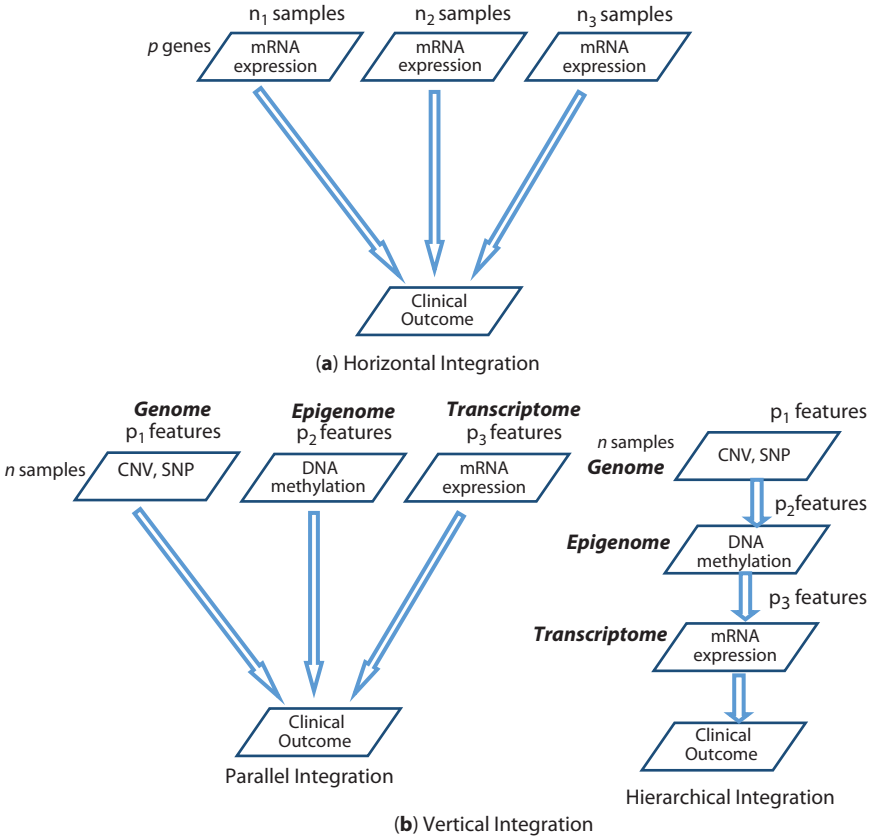


Figure 6.3 Different integration pipelines of multi-level omics data. (a) Horizontal integration (b) Vertical integration.

6.3.1 Horizontal Integration

Data generated from different studies using high throughput technologies are often restricted to the sample size. In other words the number of gene features (difference in nucleotides, gene symbols in expression profiles or methylation sites) is significantly higher than the number of samples. It puts a restriction on the predictive ability of the data. Conventionally, the data set is represented as a matrix where samples are generally arranged in columns and gene features are in rows. In horizontal integration, data from multiple genomic studies for the same gene features are combined horizontally (Figure 6.3(a)). Such multi-study approach of integration combines consensus information from heterogeneous study to enhance statistical power in biomarker detection [49, 50]. Nevertheless the horizontal integration combines data from multiple studies for different cohort of samples and utilizes the heterogeneities across studies. Still, it considers data from single omics level neglecting the information flow across levels.

6.3.2 Vertical Integration

Vertical integration is based on the fact that cellular activities are trusted up on several molecular phenomena acting at distinct omics level. It combines data from multiple omics layers for the identical group of samples and jointly analyzes genomic information across different levels. The vertical integration can be lined up in a step wise hierarchical manner or simultaneous parallel manner (Figure 6.3(b)). The hierarchical integration reflects hypothesis 1 and parallel integration reflects hypothesis 2 as illustrated in Figure 6.2. In this review we explore strategies used for the vertical integration of data at multiple omics level.

6.4 Machine Learning Approaches for Multi-Level Data Integration

Machine learning techniques used for vertical integration of multiple layers of omics data are mostly build upon three different viewpoints: concatenation-based integration, transformation-based integration and model-based integration. The inputs are multiple data matrices representing data at distinct omics layer for the same set of samples with known class labels. The schematic diagram of three statistical integrative frameworks is illustrated in Figure 6.4. Various unsupervised and supervised machine learning techniques has been proposed in literature to implement various

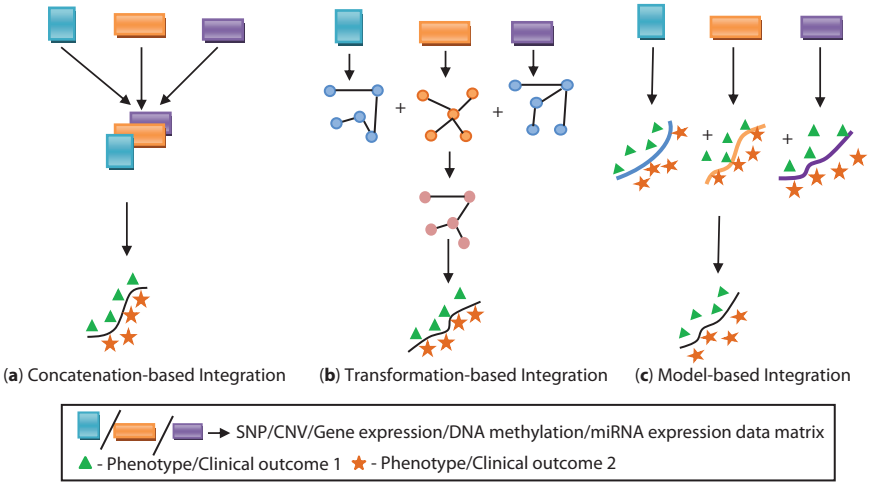


Figure 6.4 Methods for parallel integration. (a) Concatenation-based, (b) transformation based, and (c) model-based approaches.

integration schemes. In this review we have outlined various techniques used for unsupervised and supervised integration of omics data across multiple biological levels. Various studies performing integrative analysis of multi-level omics data using supervised and unsupervised machine learning approaches is summarized in Tables 6.1 and 6.2 respectively.

Concatenation-based integration combines data from manifold of data matrices for the identical set of sample and configure one composite input matrix before establishing the statistical or computational model. It is further used to model and analyze the data matrix to study and explore the interactions between different genomics data. The raw or processed input data matrices may be from different scales. E.g. Single Nucleotide Polymorphism (SNP) data value are either 0, 1 or 2 representing the difference of a single nucleotide in a human genome; copy number variation (CNV) data comprised of values in the range {−2, −1, 0, 1, 2} corresponding to a status of structural variation in a given chromosomal region; DNA methylation values are in the range of 0 and 1; mRNA expression are continuous valued values explaining the expression level of a gene in a specific condition. This approach may escalate the dimension of the data to very high, with number of gene features being much larger than the number of samples [51]. Thus, selection of appropriate method to assemble different omics level data to generate a combined data matrix in a meaningful way is the main challenge in this approach.

Table 6.1 Published studies of multi-level omics data integration using unsupervised approaches.

Method	Techniques used	Omics Data	Studied Disease
ICAN [58]	CCA + RWR Algorithm	CNV + mRNA expression + DNA methylation	Ovarian cancer
CCA sparse group [61]	SCCA + CCA-l1 + CCA-elastic net + CCA-sparse group	SNP + mRNA expression	Glioblastoma
SELP [62]	sCCA+Linear programming	mRNA expression + DNA methylation	Breast cancer
CuRSaR [63]	SRCCA+ maximum relevance maximum significance + Rough set	mRNA expression + protein expression + DNA methylation	Breast cancer Ovarian cancer
FaRoC [64]	SRCCA + Roughset	mRNA expression + protein expression + DNA methylation	Breast cancer Ovarian cancer
SCCA [65]	SCCA + PMCDCCA + relaxed PMCDCCA	mRNA expression + Lipid measurement	Nutrimouse Data
Matrix Factorization [67]	Non negative matrix factorization	mRNA expression + miRNA expression + DNA methylation	Ovarian cancer
iCluster [68]	Gaussian latent variable + EM Algorithm	CNV + mRNA expression	Lung cancer

(*Continued*)

Table 6.1 Published studies of multi-level omics data integration using unsupervised approaches. (*Continued*)

Method	Techniques used	Omics Data	Studied Disease
iCluster+ [69]	Penalized likelihood approach + LASSO penalty	CNV + mRNA expression + somatic mutation	A colorectal cancer
iClusterBayes [70]	iCkuster+ + basian latent variable	CNV + mRNA expression + DNA methylation + somatic mutation	Glioblastoma Ranal cancer
rMKL-DR [72]	MKL + graph embedding + regularization parameter	mRNA expression + miRNA expression + DNA methylation	Glioblastoma Breast cancer Lung cancer Ranal cancer Colon cancer
MKL [73]	MKL + SOM	mRNA expression + miRNA expression + DNA methylation	Breast cancer
RMKL [74]	MKL + SVM	miRNA expression + Clinical Data	Ovarian cancer
Co-relation + information gain	Feture selection + gain ratio + Chisquare + c	PPI + Protein sequence	Alzhemier's disease
Co-relation based	Co-regulation network + RWR	SNP + PPI	Alzhemier's disease
Network module-based	Gibb's sampling algoriyhm + Enrichment analysis	CNV + DNA methylation + mRNA expression + miRNA expression	Penile carcinoma

(*Continued*)

Table 6.1 Published studies of multi-level omics data integration using unsupervised approaches. (*Continued*)

Method	Techniques used	Omics Data	Studied Disease
Clustering-based network + Dense Subgraph [78]	K-means + Goldberg's algorithm	mRNA expression + PPI	Prostate cancer
Co-expression based-network + cDense subgraph [79]	Pesrson co-expression + Gpldberg's algorithm	mRNA expression + PPI	Prostate cancer
Multiplex network [82]	Similarity network fusion + generality modularity	mRNA expression + miRNA expression + DNA methylation	Glioblastoma Breast cancer

Transformation-based integration transforms each input data matrix into an intermediate form of network or kernel matrix. Multiple networks or kernels formed are then assembled which is further used for elaborating any model. The intermediate network formed for each data matrix consists of nodes as gene features and edges as association among the gene features. In transformation-based approaches are widely used in identifying the modules of interacting genes and subsequently selecting the genes within the module which are significant in the disease pathways [52, 53]. This approach is suitable for integrating variants of genomic data including continuous valued, categorical or sequence data. The transformation step in this approach preserves exclusive property specific to each genomic level data. However, separate transformation of original feature space may make it difficult to identify the interaction between different types of omics level. Thus the main objective in this approach is to find an appropriate intermediate representation of data such that the transformation preserves data type specific properties without missing the interaction effects across omics level.

Table 6.2 Published studies of multi-level omics data integration using supervised approaches.

Method	Techniques used	Omics Data	Studied Disease
FSMKL [83]	Composite kernel + confidence score	CNV + Clinical factors	Breast cancer
ATHENA [85]	Grammatical evolution neural networks	mRNA expression + miRNA expression + DNA methylation + CNV + Clinical outcome	Ovarian cancer
Network propagation [86]	Laplacian normalization + Network propagation	mutation + mRNA expression + microRNA expression	Glioblastoma Breast cancer
Ensemble model [87]	PCA + PLS + least absolute shrinkage + LASSO + COX survival model	mRNA expression + miRNA expression + DNA methylation + CNV + Clinical outcome	Breast cancer Glioblastoma Leukemia Lung cancer
Model-based [88]	Elastic net + SPCA + SPLS	mRNA expression + miRNA expression + DNA methylation + CNV + Clinical outcome	Cuteneous melanoma

Model-based integration is a meta dimensional approach. Multiple predictive models are constructed using different types of omics data as training set. Later a final ensemble model is generated from the multiple models created during the training phase. This approach preserves the data specific properties across biological levels, which can be exploited to study and analyze the association among copy number alterations, DNA methylation, miRNA, mRNA expression and metabolomic data. It is worthy to note here that model-based integration is built upon a specific hypothesis and relevant mechanisms to analyze data at individual level and further combine the resulting models in a meaningful way. As an example consider the measured data set of DNA sequence, methylation and metabolites for normal and tumor tissue of cancer. Each of the three data set can analyzed individually to generate three predictive models. The resultant DNA sequence model, methylation model and metabolomic model can further be integrated to produce a meta dimensional model. In the process of generating the combined model there is a chance of missing some of feature interactions between different omics data types. Because the individual participating models are built upon the features selected in the data type specific process and discarding the irrelevant features. Furthermore, model-based integration approaches are widely known for overfitting [54]. Hence it is advisable to deploy model-based integration only when each genomic data type is highly heterogeneous and concatenation or transformation based integration can't be possible.

6.4.1 Unsupervised Integration of Omics Data

Unsupervised integration approaches aim at finding hidden patterns in the observed data where the inputs are data matrices with unknown labels or phenotypes.

Correlation based Integration

Canonical correlation analysis (CCA) is an established multivariate feature selection process to explore the linear relationship between variables of very high dimension [55]. The objective of CCA is to find an appropriate transformation for two multidimensional data sets such that the correlation between two linear combinations of variable is maximized. Consider the data sets M and N at different omics levels. Data sets M and N are described with r and s features respectively and e is the number of samples in both M and N. CCA aims at finding bidirectional weight vectors named as basis vectors, $W_m \in R^r$ and $W_y \in R^s$ such that the correlation between the

respective projections of feature vectors into these weight vectors is maximized. The correlation coefficient ρ is given as:

$$\rho = \max_{W_m W_n} \frac{W_m^T C_{mn} W_n}{\sqrt{W_m^T C_{mm} W_m W_n^T C_{nn} W_n}} \tag{6.1}$$

Here, $C_{mm} \in R^{r \times r}$ represents the cross covariance matrix between M and N and $C_{mm} \in R^{r \times r}$ and $C_{nn} \in R^{s \times s}$ are the co variance matrices of M and N respectively.

In bioinformatics domain CCA has been widely used in concatenation of multiple feature vectors for extracting correlated features across omics level [56, 57]. Zhou *et al.* deployed CCA followed by random walk with restart algorithm (RWR) and investigated the co-alteration of copy number variation, DNA methylation and gene expression to uncover the genes associated with ovarian cancer [58]. However, the main computational challenge in traditional CCA is computing inverse of a covariance matrix, as the number of features (p) is quite large in comparison to number of samples (n). Extension of CCA, the sparse CCA (sCCA), has been developed in penalized matrix decomposition framework [59, 60] to produce a stable and robust solution for basis vector. sCCA reformulates the basis vector by mutual agreement of penalization and regularization terms. Lin *et al.* exploited sCCA to study the association among different genomics data for feature selection [61]. A comparative analysis of CCA–l_1 with l_{-1} norm penalizations, CCA-elastic net with the fusion of l_{-1} and l_{-2} norm and CCA-sparse group had been done to identify markers in human glioma. An integrated approach of Sparse Estimation with Linear Programming (SELP) on canonical correlation analysis was proposed to identify few marker genes in breast carcinogenesis [62]. Supervised regularized CCA (SRCCA) along with rough set [63] and its extension [64] was formulated using multi modal omics data for extracting features which are clinically relevant and statistically significant. Suggested algorithms outperform in extracting a feature, maximally pertaining to phenotype of interest and meaningful towards already extracted features across layers of multi dimensional data sets. Recently, sCCA was formulated as supervised learning to investigate how the computed latent variables can be used for predicting complex traits [65].

Clustering-Based Integration

The matrix factorization approach of feature selection primarily focuses on addressing the observed variations in projecting the high dimensional

multimodal data into a space of lower dimension [66]. Zhang *et al.* adopted a joint matrix factorization technique and proposed a non negative matrix factorization framework (NMF) for integrative analysis of expression profile and methylation data for ovarian cancer [67]. These high multi dimensional data were extrapolated into a common platform to investigate the association among the variables. Further, the coherent pattern in within the data was identified by scrutinizing the elements with significant z-score. The NMF approach uncovered interdependency among different layers of biological actions and identified clinically distinct patient subgroups. Computational time and space requirement was the major challenge of this approach. It is significant to mention here that NMF requires a non negative input matrices and appropriate normalization step to preprocess the input data because of highly heterogeneous nature of biological data. iCluster: a regression model with penalized parameter had been proposed for integrative clustering of multi-level omics data through a single cluster assignment [68]. The Gaussian Latent Variable framework had been adopted for integrative clustering of multi-platform biological data to identify distinct subtypes of cancer. An advancement to iCluster, iCluster+ further accommodates the desperate nature of omics measurements. iCluster+ can be applicable for different types of omics data (like continuous data, sequential data) in conjunction with various modeling assumptions (like generalized logistic regression, penalty likelihood with LASSO penalty [69]). Mo *et al.* suggested an enhancement to iCluster+ (iclusterBayes) by introducing a Bayesian latent variable and utilized the binary indexes as prior variable during probabilistic feature selection [70]. It is further generalized in various models using Metropolis–Hasting algorithm. Computation cost is a constraint for both iCluster and iCluster+.

Network-Based Integration

Epistatic interactions among various genomic objects account for disease status and progression. These interactions are materialized over multiple biological networks across various genomic layers. Network based integrative approaches are based on analyzing dependency structure among the biological entities, identifying modules co-altered genes/genomic objects or pathway enrichment analysis. Study reveals that genes participating in a common pathway lie in a close proximity in the entire network. In this context, genes represent the nodes in the network and edges correspond to association among genes. The association between genomic features is represented by a matrix A, where each entry a_{ij} in the matrix represents strength of association between the features. Study of association among the biological entities can be carried out using kernel based methods,

graph based methods or a mixture of both. Unsupervised network based integration approaches are generally applied for identifying marker genes participating in pathways and locating sub networks of co-expressed to uncover the functionality of genes. Biologically meaningful biomarkers can be identified by mapping the key genes/gene products onto known pathways or familiar networks, such as protein-protein interaction (PPI) network and analyzing their properties.

Kernel methods are based on kernel functions using positive semi definite kernel. The kernel corresponds to inner product among projections of all pair wise combinations of data in a huge dimensional feature space [71]. This method emphasizes in formulating nonlinear methods by projecting the data in input space to a high dimensional space called Kernel Hilbert space which allows successful prediction of properties of interest. An extension of multiple kernel learning for dimensionality reduction had been proposed for feature reduction and cancer subtypes identification (rMKL-DR) [72]. Multiple kernel learning (MKL) was incorporated along with a graph embedding framework for dimensionality reduction. A regularization term is appended which avoids over fitting during learning and allows use of several kernels types in data integration. Mariette *et al.* proposed a multiple kernel learning framework that combines several kernels into a single meta kernel and allows integration of multiple data sets of various types into a single exploratory analysis [73]. Wilson *et al.* proposed a multiple kernel learning framework for selecting genes that play a key role in prognostic prediction of various cancer types [74]. Wrapper functions were formulated to compute the kernel matrix and perform cross-validation in SVM.

Graph-based integration approaches rely on analyzing the associations within and across the networks of biological entities. The elementary method to quantify the pair wise significance is by measuring correlation coefficient or mutual information. Networks of co-expressed genes can be generated using Pearson correlation coefficient as the similarity measure. The resultant networks are undirected and weighted/un-weighted type. An integrative frame work of ensemble methods had been proposed for identification of marker genes associated with Alzhemier's Disease (AD) by combining structural properties of the genes in the PPI networks, sequence features and functional annotations [75]. Gene–gene co-regulation network had been constructed by incorporating expression quantitative trait loci (eQTLs) information followed by RWR approach for mining genes associated with AD [76]. A network module based integrative mechanism had been deployed [77] to identify and contextualize driver genes in pathways involved in penile carcinogenesis. Combining transcriptomic data

along with the know-how of existing gene network provides a new way for interaction based gene selection or functional module identification. In this contexture, cluster of co-expressed genes are mapped to existing network of proteins that confirms their association [78]. Clusters of co-expressed genes were formed considering Euclidean distance as similarity measure among genes. Further, the resultant induced network generated in the previous step was exploited to identify densely connected modules in the network. Genes identified in this the two step integration are statistically competent and biologically significant as well. Mahapatra *et al.* proposed an enhancement to the said approach [78] and incorporated Pearson correlation co-efficient as the similarity measure to generate the modules of co-expressed genes. The suggested approach not only selects putative and functionally coherent genes, provides a greater insight of disease associated biological process as well.

Multiplex network; assemblages of distinct networks provides a way to model and analyze such type complex real world networks by integrating multiple types of interactions among different biological units. Each individual network encodes for a different layer of multiplex system. Each layer in the multiplex system contains the same set of entities (nodes) but interconnected by different types of links (edges). The layers are connected (and hence coupled) to each other via inter layer edges [80, 81]. It is represented as network of networks, each encoding different types of association among participating units/nodes. Multiplex network was used to construct a multiplex structure to understand the effect of social relationship on employee performance and uncover eco-evolutionary dynamics of animal social behavior. The exploitation of multilayer network analysis in biological networks is sparse; however, clearly relevant with the context. Wang et al proposed a multiplex network based data integrative approach and identified 3 subtypes for glioblastoma multiforme (GBM) and breast invasive carcinoma (BIC) [82].

6.4.2 Supervised Integration of Omics Data

Supervised techniques of data integration is build up on considering the class labels (tumor or non-tumorous) and use different machine learning methods to evaluate the models. Models are constructed using known class labels from the training data.

Kernel based framework offers a pioneering approach for multi-level omics data integration. Data set at each omics level was denoted by a explicit kernel function which represents a transformed data with elicitation of relevant information from the data set. Distinct kernels are

then combined using semi definite programming. Saeone *et al.* proposed a multiple kernel based feature selection technique (FSMKL) and constructed composite kernels, which are linear combination of a large set of base kernels that encodes for a particular type of data [83]. Classifiers were constructed utilizing the pathway kernel technique and decision function obtained from various types of input data like gene expression or CNV. The accuracy of the prediction methodology is more effective when the high throughput genomic measurements and the clinical attributes are combined together.

Analysis Tool for Heritable and Environmental Network Associations (ATHENA) is a collection of analysis techniques to analyze system wide analysis of omics data and inspect their associations with clinical outcomes [84]. Kim *et al.* suggested a model based integration using ATHENA and examined the association among copy number, DNA methylation, miRNA and mRNA expression data for ovarian cancer survival analysis [85]. ATHENA utilizes neural networks to select the relevant features and further uses grammatical evolution algorithm in the training of each specific model using data from different omic levels. Then each distinct model is clubbed and a resultant generative model is produced; latter on can be applied for many determinations like diagnostically and forecasting. Ruffalo *et al.* integrated mutation and gene expression data sets for identification of sample specific important proteins using propagated PPI networks [86]. Based on existing evidence and information, propagated profile for individual genes was engineered in ppi network. In a controlled environment, most relevant features are selected and marked on these propagated profiles such that they produce the phenotype of interest. Finally a group of proteins is carefully chosen and the deciding factor is the propinquity of the samples in the network. The utility of this technique is that it shows a concealed range of genes and proteins at pathway level which have a bearing on the clinical diagnosis and prognosis of the tumor. In most cases, such outcomes are unnoticed during the mutational expression analysis.

Integrating the four genomic features (CNV, mRNA expression, DNA methylation) along with the clinical outcomes had been done in Zhao *et al.* [87]. LASSO, principal component analysis (PCA), partial least square were adopted to select marker genes. The prediction ability of the extracted features was evaluated using C-statistics (higher value of C-statistics indicate better predictive power). mRNA expression were found to be showing a higher value of C-statistics in comparison to other genomic measurements. This is a strong evidence that mRNA expression carry ample of information for disease prognosis and has a direct effect on cancer clinical outcomes.

Jiang *et al.* had developed a model based approach by combining multi-dimensional genomic features for Cutaneous melanoma [88]. Bench mark dimensionality reduction techniques of Elastic net, sparse PCA (SPCA), sparse Partial Least Square (SPLS) were utilized for extracting relevant features were selected from all platforms. Result confirmed the improved prediction ability of the selected markers due to integration of multi-level omics profiles, specially DNA methylation variables showed higher prediction accuracy for all the approaches (Elastic net, SPCA, SPLS).

6.5 Critical Observation

Technological advances in high throughput technology generate rapid dissemination of data from multiple levels of biological systems. Integrative analysis has emerged as a gateway to integrate multiple data artifacts from the same objects in a meaningful way. The main objective of integrating omics data across multiple levels is to mine remarkable biological insights from multiple biological platforms that are hard to obtain from singleton analysis of any omics data as isolated component. The above stated objective will face challenges concerning size, format and dimensionalities of the data being integrated along with the noise present in the data and level of compatibility between data sets. Mentioned data integration approaches are capable of eliminating the challenges to some extent. This is because most of the integrative approaches are built up on the techniques designed for analyzing data at exclusive omics level, that are further exploited for analyzing multi-level omics data. The state-of-the-art methods widely adopted for integrating omics data at different biological layers is summarized in Tables 6.1 and 6.2. CCA based integration are mostly applicable for small-size data sets due to the short comings in handling data of high dimension. At the same time the sparse version of CCA (sCCA) with regularization term and CCA elastic net give a comparatively stable solution for integration and feature selection. The markers identified in this approach are statistically competent and biologically interpretable as well. Heterogeneous data sets cannot be handled through kernel based methods even though it can handle large number of datasets. NMF methods have shown promising throughput in handling heterogeneous datasets also. Integration methods which are based on intermediate data integration strategies have yielded best performing results in the parameter of accuracy. Most of the network based integrative approaches rely on transforming individual data set by projecting it to a different space or merging individual networks to a single

composite network. This course of integration is limited in considering the network topology of generated networks and biological inference content in them simultaneously [89, 90]. However, the multiplex network is designed as network of networks, where individual isolated network encodes for different layers in the multiplex system with coupling edges in between. It preserves the interplay among the interacting units across layers. This type of multiplex structure can be adopted for analysis of connectivity structure of multiple networks and deduce biological inference from studied networks simultaneously. Despite of various challenges and short-comings, study of different data integration approaches and strategies have shown promising results. This has set the path open for further research on data integration mythologies and for its advanced study.

6.6 Conclusion

Emerging high-throughput technologies produce plethora of omics data at multiple levels of biological systems—from genomics to epigenomics, transcriptomics, proteomics and metabolomics. Traditional approaches of gene selection focuses on independent analysis of biological data at different levels which reduces the efficiency of the constructed prognostic models. As a result, the identified markers are less informative. In contrast, the integrative approaches promote the combined association study of biological variables across multiple layers of biological system with a higher complexity level. The markers selected are highly significant which play a key role in progression of disease. The integrative approaches are essentially suitable for life threatening diseases like cancer and Alzheimer's disease, which demands considerable efforts in multi-omics research. To conclude, we expect that this review will act as a guide for researchers to identify appropriate integrative strategy pertaining to their problems.

References

1. Metzker, M.L., Sequencing technologies—the next generation. *Nat. Rev. Genet.*, 11, 1, 31–46, 2010.
2. Ozsolak, F. and Milos, P.M., RNA sequencing: Advances, challenges and opportunities. *Nat. Rev. Genet.*, 12, 2, 87–98, 2011.
3. Laird, P.W., Principles and challenges of genome-wide DNA methylation analysis. *Nat. Rev. Genet.*, 11, 3, 191–203, 2010.

4. Maarten Altelaar, A.F., Munoz, J., Heck, A.J.R., Next-generation proteomics: Towards an integrative view of proteome dynamics. *Nat. Rev. Genet.*, 14, 1, 35–48, 2013.
5. Shulaev, V., Metabolomics technology and bioinformatics. *Brief. Bioinformatics*, 7, 2, 128–139, 2006.
6. Murphy, D., Gene expression studies using microarrays: principles, problems, and prospects. *Adv. Physiol. Educ.*, 26, 4, 256–270, 2002.
7. Simon, R., Analysis of DNA microarray expression data. *Best Pract. Res. Clin. Haematol.*, 22, 2, 271–82, 2009.
8. Boulesteix, A.-L. *et al.*, Evaluating microarray-based classifiers: An overview. *Cancer Inform.*, 6, CIN-S408, 77–97, 2008.
9. Alba, E. *et al.*, Gene selection in cancer classification using PSO/SVM and GA/SVM hybrid algorithms. *2007 IEEE Congress on Evolutionary Computation*, IEEE, 2007.
10. Singh, R.K. and Sivabalakrishnan, M., Feature selection of gene expression data for cancer classification: A review. *Procedia Comput. Sci.*, 50, 52–57, 2015.
11. Ang, J.C. *et al.*, Supervised, unsupervised, and semi-supervised feature selection: A review on gene selection. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 13, 5, 971–989, 2015.
12. Singh, R.K. and Sivabalakrishnan, M., Feature selection of gene expression data for cancer classification: A review. *Procedia Comput. Sci.*, 50, 52–57, 2015.
13. Ritchie, M.D. *et al.*, Methods of integrating data to uncover genotype–phenotype interactions. *Nat. Rev. Genet.*, 16, 2, 85–97, 2015.
14. Kalakech, M. *et al.*, Constraint scores for semi-supervised feature selection: A comparative study. *Pattern Recognit. Lett.*, 32, 5, 656–665, 2011.
15. Cai, D., Zhang, C., He, X., Unsupervised feature selection for multi-cluster data. *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2010.
16. Guyon, I. and Elisseeff, A., An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3, Mar, 1157–1182, 2003.
17. Zhang, Y. *et al.*, Divergence-based feature selection for separate classes. *Neurocomputing*, 101, 32–42, 2013.
18. Peng, Y., Wu, Z., Jiang, J., A novel feature selection approach for biomedical data classification. *J. Biomed. Inf.*, 43, 1, 15–23, 2010.
19. El Aboudi, N. and Benhlima, L., Review on wrapper feature selection approaches. *2016 International Conference on Engineering & MIS (ICEMIS)*, IEEE, 2016.
20. Tibshirani, R., Regression shrinkage and selection *via the* lasso. *J. R. Stat. Soc.: Ser. B (Methodol.)*, 58, 1, 267–288, 1996.
21. Hira, Z.M. and Gillies, D.F., A review of feature selection and feature extraction methods applied on microarray data. *Adv. Bioinf.*, 2015, 1–13, 2015.

22. Singh, R.K. and Sivabalakrishnan, M., Feature selection of gene expression data for cancer classification: A review. *Procedia Comput. Sci.*, 50, 52–57, 2015.
23. Mandal, M. and Mukhopadhyay, A., An improved minimum redundancy maximum relevance approach for feature selection in gene expression data. *Procedia Technol.*, 10, 20–27, 2013.
24. Liao, B. *et al.*, Gene selection using locality sensitive Laplacian score. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 11, 6, 1146–1156, 2014.
25. Sharma, A., Imoto, S., Miyano, S., A top-r feature selection algorithm for microarray gene expression data. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 9, 3, 754–764, 2011.
26. Xiang, S. *et al.*, Discriminative least squares regression for multiclass classification and feature selection. *IEEE Trans. Neural Netw. Learn. Syst.*, 23, 11, 1738–1754, 2012.
27. Liang, Y. *et al.*, Sparse logistic regression with a L 1/2 penalty for gene selection in cancer classification. *BMC Bioinf.*, 14, 1, 198, 2013.
28. Hajiloo, M., Rabiee, H.R., Anooshahpour, M., Fuzzy support vector machine: an efficient rule-based classification technique for microarrays. *BMC Bioinf.*, 14, S13, 1–11, 2013.
29. Song, Q., Ni, J., Wang, G., A fast clustering-based feature subset selection algorithm for high-dimensional data. *IEEE Trans. Knowl. Data Eng.*, 25, 1, 1–14, 2011.
30. Li, J. *et al.*, Using epigenomics data to predict gene expression in lung cancer. *BMC Bioinf.*, 16, 5, S10, 2015.
31. Cai, Z. *et al.*, Classification of lung cancer using ensemble-based feature selection and machine learning methods. *Mol. BioSyst.*, 11, 3, 791–800, 2015.
32. Huang, H.-H., Liu, X.-Y., Liang, Y., Feature selection and cancer classification *via* sparse logistic regression with the hybrid L1/2+ 2 regularization. *PLoS One*, 11, 5, e0149675, 2016.
33. Singh, P., Shukla, A., Vardhan, M., Hybrid approach for gene selection and classification using filter and genetic algorithm. *2017 International Conference on Inventive Computing and Informatics (ICICI)*, IEEE, 2017.
34. Pino Angulo, A., Gene selection for microarray cancer data classification by a novel rule-based algorithm. *Information*, 9, 1, 6, 2018.
35. Nakariyakul, S., A hybrid gene selection algorithm based on interaction information for microarray-based cancer classification. *PLoS One*, 14, 2, e0212333, 2019.
36. Wenric, S. and Shemirani, R., Using supervised learning methods for gene selection in RNA-Seq case-control studies. *Front. Genet.*, 9, 297, 2018.
37. Wang, Y., Li, X., Ruiz, R., Weighted general group lasso for gene selection in cancer classification. *IEEE Trans. Cybern.*, 49, 8, 2860–2873, 2018.
38. Elyasigomari, V. *et al.*, Development of a two-stage gene selection method that incorporates a novel hybrid approach using the cuckoo optimization algorithm and harmony search for cancer classification. *J. Biomed. Inf.*, 67, 11–20, 2017.

39. Shukla, A.K., Singh, P., Vardhan, M., A two-stage gene selection method for biomarker discovery from microarray data for cancer classification. *Chemom. Intell. Lab. Syst.*, 183, 47–58, 2018.
40. Jain, I., Jain, V.K., Jain, R., Correlation feature selection based improved-binary particle swarm optimization for gene selection and cancer classification. *Appl. Soft Comput.*, 62, 203–215, 2018.
41. Alshamlan, H.M. *et al.*, Genetic Bee Colony (GBC) algorithm: A new gene selection method for microarray cancer classification. *Comput. Biol. Chem.*, 56, 49–60, 2015.
42. Kabir, Md M., Islam, Md M., Murase, K., A new wrapper feature selection approach using neural network. *Neurocomputing*, 73, 16–18, 3273–3283, 2010.
43. Peng, Y., Wu, Z., Jiang, J., A novel feature selection approach for biomedical data classification. *J. Biomed. Inf.*, 43, 1, 15–23, 2010.
44. Shen, Q., Diao, R., Su, P., Feature Selection Ensemble, in: *Turing-100*, vol. 10, pp. 289–306, 2012.
45. Russo, G., Zegar, C., Giordano, A., Advantages and limitations of microarray technology in human cancer. *Oncogene*, 22, 42, 6497–6507, 2003.
46. Yang, Z.-Y. *et al.*, Multi-view based integrative analysis of gene expression data for identifying biomarkers. *Sci. Rep.*, 9, 1, 1–15, 2019.
47. Subramanian, I. *et al.*, Multi-omics data integration, interpretation, and its application. *Bioinf. Biol. Insights*, 14, 1–24, 2020.
48. Richardson, S. *et al.*, Statistical Methods in Integrative Genomics. *Annu. Rev. Stat. Appl.*, 3, 181–209, 2016.
49. Li, Q. *et al.*, Meta-analysis based variable selection for gene expression data. *Biometrics*, 70, 4, 872–880, 2014.
50. Bhattacharjee, S. *et al.*, A subset-based approach improves power and interpretation for the combined analysis of genetic association studies of heterogeneous traits. *Am. J. Hum. Genet.*, 90, 5, 821–835, 2012.
51. Clarke, R. *et al.*, The properties of high-dimensional data spaces: Implications for exploring gene and protein expression data. *Nat. Rev. Cancer*, 8, 1, 37–49, 2008.
52. Wang, B. *et al.*, Similarity network fusion for aggregating data types on a genomic scale. *Nat. Methods*, 11, 3, 333, 2014.
53. Bonnet, E., Calzone, L., Michoel, T., Integrative multi-omics module network inference with Lemon-Tree. *PLoS Comput. Biol.*, 11, 2, e1003983, 2015.
54. Opitz, D. and Maclin, R., Popular ensemble methods: An empirical study. *J. Artif. Intell. Res.*, 11, 169–198, 1999.
55. Hotelling, H., Relations between two sets of variates, in: *Breakthroughs in Statistics*, pp. 162–190, Springer, New York, NY, 1992.
56. Yuan, Y.-H. and Sun, Q.-S., Multiset canonical correlations using globality preserving projections with applications to feature extraction and recognition. *IEEE Trans. Neural Netw. Learn. Syst.*, 25, 6, 1131–1146, 2013.

57. Lee, G. *et al.*, Supervised multi-view canonical correlation analysis (sMVCCA): Integrating histologic and proteomic features for predicting recurrent prostate cancer. *IEEE Trans. Med. Imaging*, 34, 1, 284–297, 2014, [] D. Lin *et al.*, "Group sparse canonical correlation analysis for genomic data integration," BMC Bioinformat., vol. 14, no. 1, p. 245, 2013.
58. Zhou, Y. *et al.*, ICan: An integrated co-alteration network to identify ovarian cancer-related genes. *PLoS One*, 10, 3, e0116095, 2015.
59. Witten, D.M., Tibshirani, R., Hastie, T., A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 10, 3, 515–534, 2009.
60. Tibshirani, R. *et al.*, Sparsity and smoothness *via the* fused lasso. *J. R. Stat. Soc.: Ser. B (Stat. Methodol.)*, 67, 1, 91–108, 2005.
61. Lin, D. *et al.*, Group sparse canonical correlation analysis for genomic data integration. *BMC Bioinf.*, 14, 1, 1–16, 2013.
62. Safo, S.E. *et al.*, Sparse generalized eigenvalue problem with application to canonical correlation analysis for integrative analysis of methylation and gene expression data. *Biometrics*, 74, 4, 1362–1371, 2018.
63. Maji, P. and Mandal, A., Multimodal Omics Data Integration Using Max Relevance–Max Significance Criterion. *IEEE Trans. Biomed. Eng.*, 64, 8, 1841–1851, 2016.
64. Mandal, A. and Maji, P., FaRoC: Fast and robust supervised canonical correlation analysis for multimodal omics data. *IEEE Trans. Cybern.*, 48, 4, 1229–1241, 2017.
65. Rodosthenous, T., Shahrezaei, V., Evangelou, M., Integrating multi-OMICS data through sparse Canonical Correlation Analysis for the prediction of complex traits: A comparison study. *Bioinformatics*, 1–10, 2020.
66. Lee, D.D. and Seung, H.S., Algorithms for non-negative matrix factorization. *Adv. Neural Inform. Process. Syst.*, 556–562, 2001.
67. Zhang, S. *et al.*, Discovery of multi-dimensional modules by integrative analysis of cancer genomic data. *Nucleic Acids Res.*, 40, 19, 9379–9391, 2012.
68. Shen, R., Olshen, A.B., Ladanyi, M., Integrative clustering of multiple genomic data types using a joint latent variable model with application to breast and lung cancer subtype analysis. *Bioinformatics*, 25, 22, 2906–2912, 2009.
69. Mo, Q. *et al.*, Pattern discovery and cancer gene identification in integrated cancer genomic data. *Proc. Natl. Acad. Sci.*, 110, 11, 4245–4250, 2013.
70. Mo, Q. *et al.*, A fully Bayesian latent variable model for integrative clustering analysis of multi-type omics data. *Biostatistics*, 19, 1, 71–86, 2018.
71. Hofmann, T., Schölkopf, B., Smola, A.J., Kernel methods in machine learning. *Ann. Stat.*, 36, 3, 1171–1220, 2008.
72. Speicher, N.K. and Pfeifer, N., Integrating different data types by regularized unsupervised multiple kernel learning with application to cancer subtype discovery. *Bioinformatics*, 31, 12, i268–i275, 2015.
73. Mariette, J. and Villa-Vialaneix, N., Unsupervised multiple kernel learning for heterogeneous data integration. *Bioinformatics*, 34, 6, 1009–1015, 2018.

74. Wilson, C.M. *et al.*, Multiple-kernel learning for genomic data mining and prediction. *BMC Bioinf.*, 20, 1, 426, 2019.
75. Jamal, S. *et al.*, Integrating network, sequence and functional features using machine learning approaches towards identification of novel Alzheimer genes. *BMC Genomics*, 17, 1, 807, 2016.
76. Li, J. *et al.*, Mining disease genes using integrated protein–protein interaction and gene–gene co-regulation information. *FEBS Open Bio*, 5, 251–256, 2015.
77. Marchi, F.A. *et al.*, Multidimensional integrative analysis uncovers driver candidates and biomarkers in penile carcinoma. *Sci. Rep.*, 7, 1, 1–11, 2017.
78. Swarnkar, T. *et al.*, Identifying dense subgraphs in protein–protein interaction network for gene selection from microarray data. *Netw. Model. Anal. Health Inform. Bioinform.*, 4, 1, 33, 2015.
79. Mahapatra, S., Mandal, B., Swarnkar, T., Biological networks integration based on dense module identification for gene prioritization from microarray data. *Gene Rep.*, 12, 276–288, 2018.
80. Tomasini, M., *An introduction to multilayer networks*, pp. 1–14, BioComplex Laboratory, Florida Institute of Technology, Melbourne, USA, 2015.
81. Kanawati, R., Multiplex Network Mining: A Brief Survey. *IEEE Intell. Inf. Bull.*, 16, 1, 24–27, 2015.
82. Wang, H. *et al.*, Integrating omics data with a multiplex network-based approach for the identification of cancer subtypes. *IEEE Trans. Nanobiosci.*, 15, 4, 335–342, 2016.
83. Seoane, J.A. *et al.*, A pathway-based data integration framework for prediction of disease progression. *Bioinformatics*, 30, 6, 838–845, 2014.
84. Holzinger, E.R. *et al.*, ATHENA: The analysis tool for heritable and environmental network associations. *Bioinformatics*, 30, 5, 698–705, 2014.
85. Kim, D. *et al.*, ATHENA: Identifying interactions between different levels of genomic data associated with cancer clinical outcomes using grammatical evolution neural network. *BioData Min.*, 6, 1, 23, 2013.
86. Ruffalo, M., Koyutürk, M., Sharan, R., Network-based integration of disparate omic data to identify "silent players" in cancer. *PLoS Comput. Biol.*, 11, 12, e1004595, 2015.
87. Zhao, Q. *et al.*, Combining multidimensional genomic measurements for predicting cancer prognosis: observations from TCGA. *Briefings Bioinf.*, 16, 2, 291–303, 2015.
88. Jiang, Y. *et al.*, Integrated analysis of multidimensional omics data on cutaneous melanoma prognosis. *Genomics*, 107, 6, 223–230, 2016.
89. Dutkowski, J. *et al.*, A gene ontology inferred from molecular networks. *Nat. Biotechnol.*, 31, 1, 38–45, 2013.
90. Sun, K. *et al.*, The integrated disease network. *Integr. Biol.*, 6, 11, 1069–1079, 2014.

7

Random Forest Algorithm in Imbalance Genomics Classification

Sudhansu Shekhar Patra[1]*, Om Praksah Jena[2], Gaurav Kumar[3], Sreyashi Pramanik[1], Chinmaya Misra[1] and Kamakhya Narain Singh[1]

[1]*School of Computer Applications, KIIT Deemed to be University, Bhubaneswar, India*
[2]*School of Information and Computer Sciences, Ravenshaw University, Cuttack, India*
[3]*Scientist, Virology Lab, Government Medical College, Shahdol, India*

Abstract
Random Forest (RF) is a well-known ensemble technique based on decision tree designed to improve the accuracy of CART and can be applied for classification as well as regression problems. It can be applied to problems which are classified as "large p and small n" problems with highly data adaptive. It accounts the interaction among the features and correlation among features. It works by constructing many CART trees on the training dataset making the model less interpretable. To predict for a new data, each tree votes on the outcome, and we aggregate the outcome by majority voting. Genomics data are usually high dimensional multi-class imbalance data. This chapter describes and shows the usefulness of RF algorithm for high dimensional genomic data analysis and shows the accuracy of the result.

Keywords: Random forest, decision tree, bioinformatics, imbalance, genomics

7.1 Introduction

Bioinformatics involves a group of methods which is used to capture diverse sources of biological information and to prepare models to analyze that information and other advanced research. It offers opportunities for developing innovative data analysis methods. Machine learning [1] has

**Corresponding author*: sudhanshupatra@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (173–190) © 2021 Scrivener Publishing LLC

wide applications in bioinformatics [2, 3], and of the tool decision tree has many successful applications in this domain. This chapter briefly reviews the use of decision trees and the ensemble algorithms related to decision trees and shows the key areas of such advances on resolving biological complexities. By using the classification technique of decision trees (DT) and related ensemble learning classifiers, biologists gets benefited how ML algorithms applied in biological science. Conversely, the computer science researchers can able to identify for which categories of bioinformatics topics the open area is there and they can do future research [4].

Decision tree [6] follows two important properties i.e., low bias along with high variance. The low bias property builds the DT up to the full depth and trains the data set and the training error is very less. On the other hand high variance in a new test data, it gives small errors. If a DT is created for a full depth it handles the issues of overfitting. Random forests (RF) [5] is a boosting technique follows DT-based ensemble ML tool which is extremely data adaptive, having low bias along with low variance, applies a number of DTs as the base predictor, aggregates them and predicts the final output w.r.t majority vote able to convert the DT feature to low variance. This feature of RF makes it invoking for multi-dimensional imbalance genomic data analysis. The robustness of the algorithm w.r.t large feature sets, gives better accuracy in the result as compared to other ensemble-based classifiers.

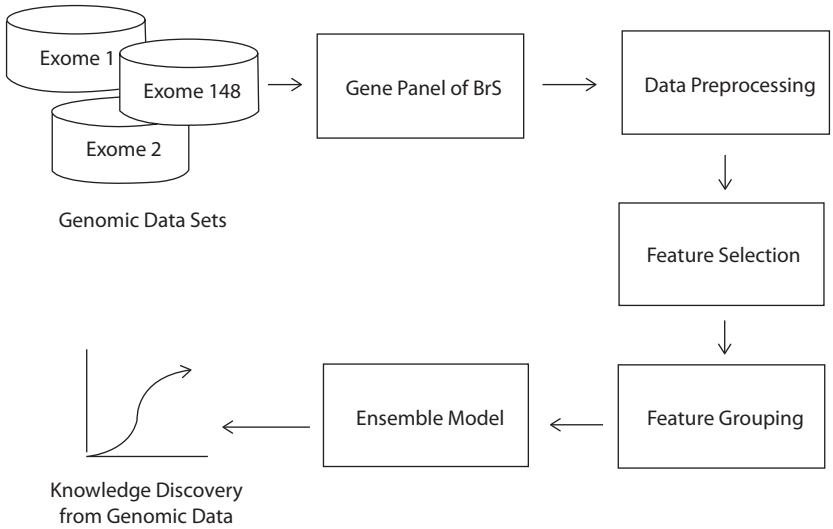


Figure 7.1 Knowledge discovery process in genomic data.

In this chapter, we proposed a modified RF algorithm which uses similarity measures for genomic datasets which performs better in prediction, classification, variable selection, as well as unsupervised learning. Figure 7.1 shows the knowledge discovery process in genomic.

The chapter is organized as follows. The methodological issues and the solutions for the issues are presented in Section 7.2, Section 7.3 defines many biological terminologies, the proposed model is depicted in Section 7.4, the experimental results depicted in Section 7.5, the current and future scopes of ML in the field of genomics is presented in Section 7.6 and finally the conclusion is presented in Section 7.7.

7.2 Methodological Issues

7.2.1 Decision Tree (DT) Classifier

For the multi-class classification decision tree (DT) is an appropriate nonparametric classification tool. DT classifier builds a tree-like decision structure to process the training dataset, generates multiple leaves starting from a root node as shown in Figure 7.2. The internal nodes are the nodes in between the root nodes and leaves. Each node either split into internal node or as a leaf node keeping in view there is increase in purity in the dataset after splitting. The split must be such that it should cause the dataset into separate groups such that they belong to predominantly to one class. DT classifier is sensitive due the nature of high variance in the classifier. Hyperparameter optimization is needed to minimize variance by increasing the bias. Bias can be minimized by increasing the variance by splitting the tree to greater depth with more splits.

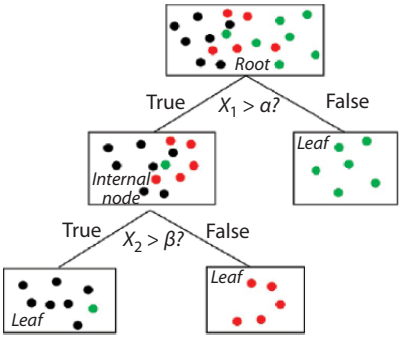


Figure 7.2 Decision tree classifier on a dataset having two features (X_1 and X_2) and three classes.

ID3, C4.5 and CART are the three different DT algorithms which adopt greedy technique where DTs are constructed following divide and conquer technique which are recursive in nature. The decision tree follows the approach in which the algorithm starts with a training set of tuples and the associated class labels. The training dataset is splitting into smaller subsets. The attribute selection measure is a heuristic for the selection of the splitting criteria that best labelled the training dataset into individual classes. For the best splitting each partition must be pure i.e., the tuples falling into a node after partition must belong to the identical class. There are 2 different feature selection criteria are popularly used—Information gain and Gini Index

Entropy—It helps to measure the purity of the split. It is denoted by E(S) and formulated as in Equation (7.1):

$$E(S) = -Pc_{(+)} \log_2 Pc_{(+)} - Pc_{(-)} \log_2 Pc_{(-)} \tag{7.1}$$

Here $Pc_{(+)}$ is the percentage of + ve class and $Pc_{(-)}$ is the percentage of the – ve class.

Because of the information is encoded in bits, a logarithmic function with base 2 is used.

Information gain—ID3 algorithm uses information gain as its information gain measures. Let node N is the feature or attribute which has maximum information gain, then it is considered as the best choice for splitting node N.

The expected information required for classifying a tuple in T is denoted by,

$$Info_Gain(T, A) = E(T) - \sum_{V \in VAL} \frac{|T_V|}{|T|} E(T_V) \tag{7.2}$$

Here E(T) is the Entropy for the specific split

T is the subset, T_V subset after splitting, $E(T_V)$ is the Entropy of the subset after splitting.

Gini Impurity—The Gini Impurity is used in CART. It measures the impurity.

$$GI = 1 - \sum_{i=1}^{n} (pc)^2 \tag{7.3}$$

$$= 1 - [(Pc_{(+)})^2 + (Pc_{(-)})^2] \tag{7.4}$$

Where $Pc_{(+)}$ is the percentage of + ve class
and $Pc_{(-)}$ is the percentage of the − ve class.

7.2.2 Ensemble Techniques

The core idea of Ensemble technique is the multiple predictors for the same dataset and makes the final prediction by aggregating the predictions resulting from the multiple predictors. Here the problem is a supervised learning prediction problem by multiple inputs, job is to predict the class. Here the base predictor is the same and what we are generating are the different versions. So if multiple predictors are creating different regression equations, then technicality they are ensemble method. Figure 7.3 shows the ensemble learning model to increase the accuracy of the classification algorithm Ensemble techniques are basically of 2 types:

1. Bagging (Bootstrap Aggregation)—One of the technique in bagging is called Random Forest. Random Forest uses multiple decision trees.
2. Boosting—Boosting is classified into 3 categories: AdaBoosting, Gradient Boosting(GB) and Extra Gradient boosting (Xgboost).

7.2.3 Mathematical Formulation of Ensemble Technique

The goal of Ensemble technique is the improvement of prediction performance of any particular model fitting problem. The idea of the model is to

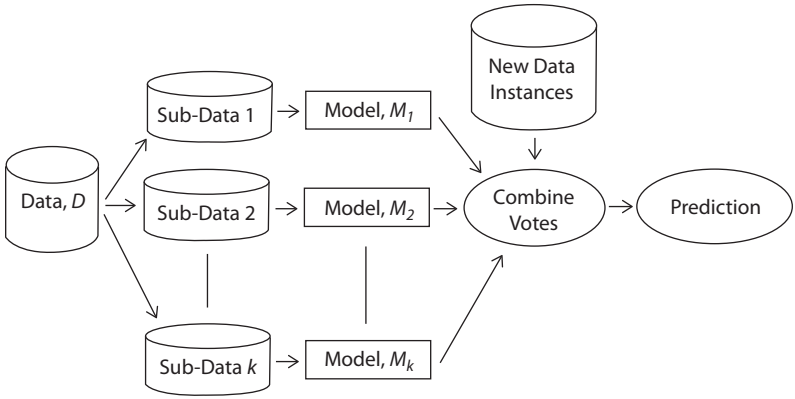


Figure 7.3 Ensemble learning model to increase the accuracy of classification model.

build a linear combination of multiple predicted models, instead of the use of a single fit.

The ensemble learning helps in predicting a real-valued function f: $R^d \rightarrow R$.

Based on data $\{(P_1,Q_1)),\ldots.(P_n, Q_n)\}$ where P is the d-dimensional predictor variable and Q a univariate response. By taking a base procedure for some testing data let the prediction be estimated as a function $\hat{f}(.)$. The base procedure can be a non-parametric kernel estimator (for small d) or a non-parametric statistical method for d greater than or equals 2 either regression tree or class prediction using classification tree.

The base procedure can be run several times by replacing the training data. The basic idea of ensemble technique is to take the aggregated value of the different estimates $\hat{f}_1(.),\hat{f}_2(.),\hat{f}_3(.),\ldots.$ on separate reweighted training data. After than we can design $\hat{f}_{ens}(.)$ a function estimate using the linear combination of individual $\hat{f}_k(.)$.

7.2.4 Bagging

Bagging a shortened for bootstrap aggregating, is a well-known ensemble technique for improving the classification scheme. Breiman [4] suggested Bagging as a technique to reduce high variance for a given base procedure, for e.g., decision tree or fittings in linear model. Besides the tool to reduce the variance it also reduces the mean squared error (MSE).

Bootstrap aggregation is a technique which is used in reducing the variance of the algorithms having high variance. The algorithms with high variance are DTs used as classifiers and also the regression trees (CART). DTs are specific to a training data. When the training data gets changed the generated DT will be different and the predictions will also be different. Bagging is an application of bootstrap procedure which gives solution for the high variance classification algorithms such as DTs.

Let us consider a sample dataset having 5,000 instances and let we are using the CART DT algorithm. Bagging works as follows:

1. Build random sub samples (for e.g., 250) of the training dataset with replacement. This step uses row sampling with replacement and is known as bootstrap.
2. Apply CART model on each sub sample to train the data and get a prediction result.
3. Apply the majority voting scheme (classification problem) otherwise mean or median (in case of regression problem) and aggregate the output. This step is known as the aggregation.

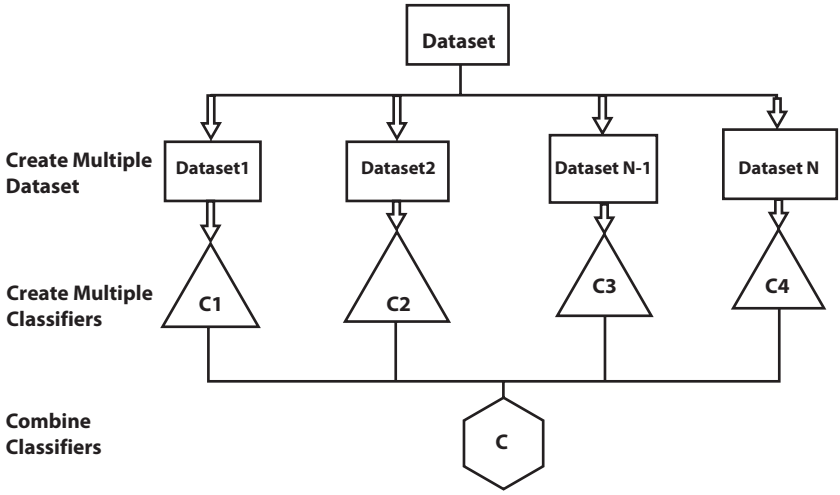


Figure 7.4 Bootstrap aggregation (bagging) technique.

These are the important characterize of the sub-models when combing predictions using bagging.

The parameters needed to implement bagging are the number of decision trees to be included and the samples in each training set. This can be done by incremental way by increasing the number of decision trees during run after run till the accuracy is being constant and the algorithm doesn't show further improvement. Many models take a long duration for the prevarication, but it doesn't overfit the training data. Similar to decision trees, bagging is used for classification as well as regression problems. Figure 7.4 shows the working principal of Bagging technique.

7.2.5 Bagging Pseudocode

Step 1: Build a bootstrap sample $(P_1^\star, Q_1^\star), \ldots (P_n^\star, Q_n^\star)$ by random selection n times with replacement from the data $\{(P_1, Q_1), \ldots (P_n, Q_n)\}$.

Step 2: Calculate bootstrap estimator $\hat{f}^\star(.)$ by $\breve{f}^\star(.) = h_n((P_1^\star, Q_1^\star), \ldots (P_n^\star, Q_n^\star)(.)$.

Step 3: Repeatedly do step 1 and 2 N times; where $N <= 100$, giving $\hat{f}^\star(.)$ (k = 1,2,…N). The bagging estimator is $\hat{f}_{Bag}(.) = N^{-1} \sum_{k=1}^{N} \hat{f}^{*k}(.)$.

7.2.6 Random Forest

RF classifier is an ensemble technique which trains multiple decision trees in parallel first bootstrapping and then followed by aggregation, referred as bagging. Bootstrapping have the feature in which multiple decision trees parallelly trained in subsets of the training dataset called row sample with replacement. Bootstapping takes care of each individual decision tree in the RF is unequalled, which takes care of the high variance property of the decision tree in the RF classifier. For the final decision, the RF classifier aggregates the predictions of the individual base classifiers i.e., the decision tree here. RF classifier follows good generalization. Compared to other classifiers Random Forests outperforms in terms of accuracy taking care of the overfitting issues. Feature scaling is not required in RF classifier which is in DT classifier. The method is more robust in case of noise data, easier to tune the hyper parameters in comparison of DT classifier. Figure 7.5 shows the use of RF classifier.

Feature of RF provides fast computation of the internal measure of variable importance (VIMP) which is used for ranking variables. For multidimensional imbalance genomic data this feature is very useful. Gini index or the node impurity index and permutation importance are the two measurements for this. In classification model, the Gini index is depends on the impurity measure of a node for splitting a node. Modified VIMP evaluates have been suggested for genomics. The use of sub sampling without

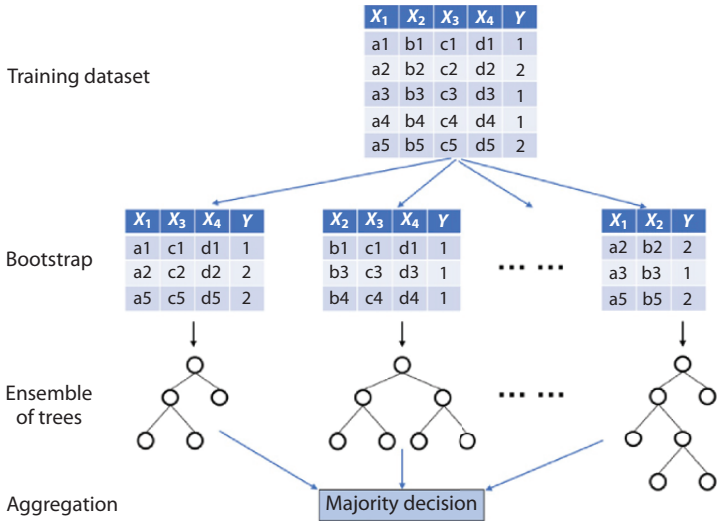


Figure 7.5 Implementing RF classifier in a dataset having four features ($X_1 \ldots X_4$) and two classes (Y_1, Y_2).

replacement instead of bootstrapping was proposed by Strobl *et al.* [11]. For correlated variables a conditional permutation VIMP was suggested for correcting the bias [12]. A importance of chi-square measure was developed in Ref. [8].

7.3 Biological Terminologies

7.3.1 DNA

DNA carries all the genetic information; these are the instructions that make a living being. DNA stands for deoxyribonucleic acid, and it's found in nearly all the living cells. The DNA molecule is made up of two single strands of DNA that bind together to form a structure known as the double helix, which looks like a ladder that has been twisted. Structurally, DNA is what's known as a polymer: this means that it's made up of many small units. In DNA, these units are known as nucleotides, and they are the key to forming the double helix. The nucleotides are themselves, made up of three different components. Two of them sugar molecule called deoxyribose, and a phosphate, bond together to form a backbone which makes up the sides of the DNA ladder. The rungs of the ladder, that connect these two backbones, are made up of chemical bases that are weakly bonded together. Four different kinds of chemical bases present in DNA: adenine, thymine, cytosine and guanine. They always bond in a specific way, and this is known as the base pairing pattern: A pairs with T and C pairs with G. Understanding the properties of DNA, such as base pairing rule, is important, as these can be exploited in many molecular tests that are carried out in clinical genetics laboratories.

7.3.2 Genomics

Gene is a piece of DNA which encodes the concoction of gene product like protein and RNA, that contains the information for the production of biological molecules usually proteins. Human body has 20,000 to 25,000 genes, each containing the information needed to build more than one protein. Some genes regulate physical characteristics, such as the our eyes color, skin color; others can affect our risk of progressing certain conditions which are inherited, such as diabetes, but genes only account for around 2% of all genetic information which is made up of more than 3 billion of nitrogen bases. While we don't understand the exact function of all the other 98%, we know do know that changes to this DNA can also impact

on our health. So where is all this DNA stored? Almost all of our DNA is arranged into tightly coiled structures called chromosomes. Humans have 23 pairs of chromosomes, are found in the nucleus of the cell, with one of each pair inherited from our father and the other from our mother. A very small amount of DNA is also found in the Mitochondria, a structure which produces energy in the cell. All the DNA contained in one cell is known as genome. We have only a single copy of our genome in nearly every cell in a human body. Every other human being has approximately 99.8% similarity with each other through genomes, it's the 0.2% variation in our genome that interests in health-care, as understanding it can help in the projection, prohibition, diagnose and cure diseases. Figure 7.6 shows the structure of genes, proteins and molecular machines.

Genomics is a versatile field of biology where we study about structural data, functionality, depiction and modifying of gene. A genome is a whole set of DNA of an organism; it includes all of the genes. We can divide genomics into different classifications:

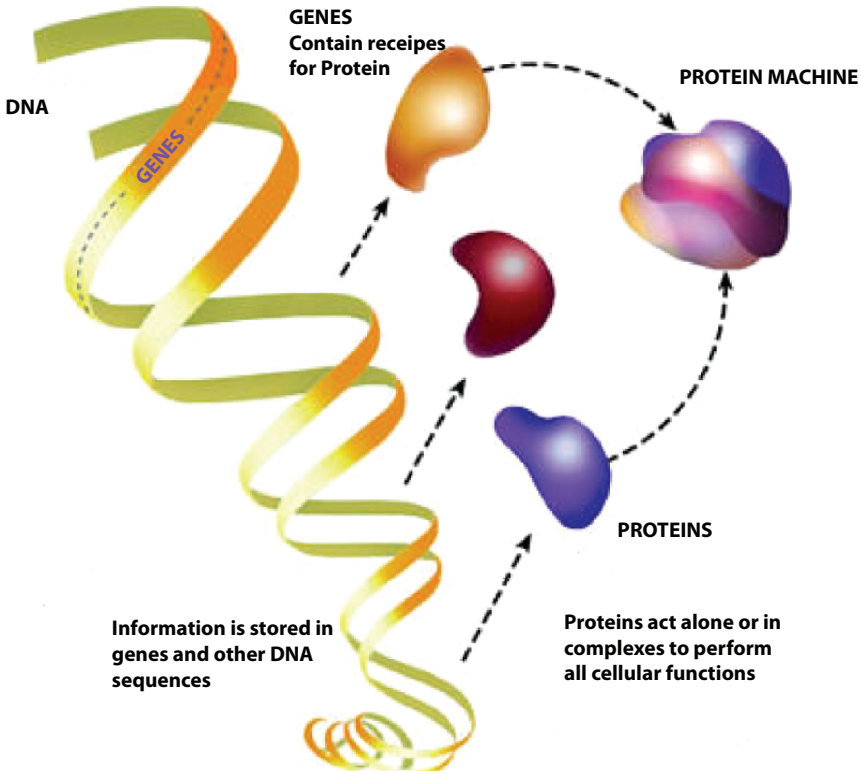


Figure 7.6 Genes, proteins and molecular machines.

1. Regulatory genomics—In this area, the researchers study the genomics features and the regulating expression. The ML applications in this area predict the classification of gene expression and produces transcription factors as well as RNA-binding proteins. Also ML predicts the promoters and the enhancers for the gene expression.
2. Functional genomics—This is where the researchers study the gene functionalities and the interactions. ML helps in studying the mutation classification, producers, promoters and enhancers and classifies the sub cellular localization.
3. Structural genomics—The researchers explore the genome structures. ML helps in classify the structure of protein, classify protein tertiary and secondary structure.

7.3.3 Proteins

Proteins are essentials for life and involved in all the processes in our body. Proteins are the complex molecules made of smallest sub units called amino acids, molecules containing an amine and a carboxyl functional group and are the most important building blocks. The amino acid sequence gives the primary structure of a protein. There are 20 essential amino acids. The secondary structure refers to the folding the chain of amino acids of proteins. It provides information about local inter-residue reaction by the formation of the hydrogen bonds. The most prevalent secondary structures are α-helices β-sheets. The tertiary structure of the protein is the 3-D structure formed by the folding of the secondary structure (α-helice β-sheets) upon themselves. Sometimes many folded protein molecules combine to make a multi-subunit complex (like hemoblobin) usually bound by hydrogen bonds. This complex structure is called the quaternary structure of the protein.

7.4 Proposed Model

This section describes the model used for the classification of genomics data, a high dimensional multi-class data. The model implements the data pre-processing, attribute selection and uses ensemble technique and constructed an ensemble classifier for classification. The method commutes the imbalance genomic dataset into several balanced data and then made multiple DTs on these data with specified feature. The classification results of these DTs are aggregated by the help of majority voting technique for the classification of the new genomics.

7.4.1 Balancing the Data

One of the common issues found in datasets that are used for classifications is imbalanced class issues. Data imbalance usually occurs because of data sampling and the properties of the domain. The solution for the imbalances dataset is up-sampling and down-sampling. In the down-sampling the new dataset has minority instances will be the same and the majority instances are eliminated from the original data. In the other hand, the up-sampling increases the minority class instances and makes the data balanced. However, the drawback of down-sampling is the data may lose certain critical information while up-sampling causes overfitting [7]. The bagging including random forest is used for imbalance data set.

For dealing with imbalance data, the majority class instances are divided into many clusters and every cluster have same number of instances as the minority instances. Clustering with similarity-measure (Csim) is being used for cluster majority class instances. Csim is beneficial to put the similar instances in one set and dissimilar data in different clusters. The algorithm 1 shows the algorithm for similarity measure between a cluster center and an instance. The distance measure can be anyone out of Euclidean distance, Manhattan distance or Minkowski distance between the instances.

Let's consider Csim (r_i , r_l) is the similarity measure between l^{th} cluster center r_l and r_i. The aim is to search d_l to maximize the whole similarity measure.

$$\mathrm{Js}(C) = \sum_{l=1}^{m} \sum_{i=1}^{N} f(\mathrm{CSim}(r_i, r_l)) \tag{7.5}$$

Algorithm 1: Clustering based on similarity measure

1. Input: R= {r_1, r_2,…..,r_N}
2. Output : collection of clusters, Cluster= {$\mathrm{Cluster}_1$,$\mathrm{Cluster}_2$,… $\mathrm{Cluster}_m$}.
3. Cluster <- ϕ
4. m=1
5. $\mathrm{Cluster}_m$={r_1}
6. Cluster=Cluster ∪ $\mathrm{Cluster}_m$
7. for i <-2 to N do
8. for l <- 1 to m do
9. Compute the lth cluster center $r_l \in Cluster_l$ for maximizing the similarity measure
   CSim(r_i,r_l)

```
if CSim(r_i,r_l)>= threshold then
  Cluster_m ← Cluster_m ∪ r_i
else
  m ← m+1
  Cluster_m ← {r_l}
  Cluster ← Cluster ∪ Cluster_m
```

7.4.2 Ensembling of Trees

We used ensembling of DT on genomic data for the generation of better predictive performance of DNA categories. Many DTs are combined together to get out of the risk of overfitting. Let the initial balanced data is D. It is classified into several balanced data denoted as $D_1, D_2, \ldots, D_m$. Then the output of each D_p is grouped together and build a tree D_q. The error rate of D_q is computed on balanced data sets D_p. If the rate of error of D_q is <= the threshold then the tree is considered for ensembling. Generally the threshold is considered as 0.5. The new instance is predicted by aggregating the predictions done by voting of the all the trees of that group. The learning technique is depicted in algorithm 2.

Algorithm 2: Ensembling of DTs

Input: C4.5 decision tree algorithm, Genomic data
Output: Group of Trees, DT^*

```
DT* ← ∅
From Imbalance genomic data create balanced data sets,
{D_1,D_2,…,D_k}
for p <- 1 to m
do
    Divide the features in D_p into k groups
    for q <- 1 to k
     do
          Construct a DT_q having q feature group
          Calculate error(DT_q) on D_p
          if error (DT_q) ≤ threshold
                    DT* = DT* ∪ D_q
Apply aggregating function as the majority voting to classify a
new instance x_new
```

7.5 Experimental Analysis

For the analysis of the above methodology Python is used. Scikit-learn library with 8 GB RAM Intel i5 processors is being used to study the performance of the proposed method. Classification accuracy, F-score, recall, precision and 10-fold cross validation are being used for testing the classifier. Equation (7.6) is used to measure the classification accuracy.

$$\text{accuracy} = \frac{\sum_{j=1}^{|X|} \text{assess}(x_j)}{|X|}, x_j \in X \tag{7.6}$$

If x_j is correctly classified then assess $(x_j) = 1$, otherwise $(x_j) = 0$. Equations (7) to (9) show the calculations of recall (sensitivity, SE), Precision (Positive predictive value, PPV) and F-Score where the symbols used there are

QTP is the quantity of true positive
QTN is the quantity of true negative
QFP is the quantity of false positive and
QFN is the quantity of false negative

$$\text{SE} = \frac{\text{QTP}}{\text{QTP} + \text{QFN}} \tag{7.7}$$

$$\text{PPV} = \frac{\text{QTP}}{\text{QTP} + \text{QFP}} \tag{7.8}$$

$$\text{F_score} = \frac{2*\text{PPV}*\text{SE}}{\text{PPV} + \text{SE}} \tag{7.9}$$

The 10-fold cross validation is used here which divides the data into 10 groups of same size each. With nine datasets it trains the classifier and the remaining one is used for testing. 10 times it's repeated and the mean accuracy rate is considered.

The performance of the ensemble technique with the proposed method on Brugada syndrome (BrS) of 2,143 variants which is from 148 Extome

datasets [9]. The performance of RF, Bagging, Boosting and proposed method are shown in Table 7.1. Table 7.2 depicts the performance of the proposed method against ensemble methods using 10-folds cross validation. 103 Exome data sets are being used for training and 45 Exome data sets are used for testing the model, Table 7.3 shows the results on 45 Exome data sets.

Table 7.1 The classification accuracy, recall, precision and F-score of ensemble techniques using training data.

Learning Method	Accuracy (%)	recall	precision	F-Score
Bagging	87.63	0.86	0.84	0.84
Boosting	91.67	0.90	0.93	0.91
RF	92.33	0.93	0.96	0.93
Proposed Method	94.12	0.94	0.93	0.93

Table 7.2 The classification accuracy, recall, precision and F-score of ensemble techniques using 10 folds cross-validation.

Learning Method	Accuracy (%)	recall	precision	F-Score
Bagging	58.35	0.57	0.33	0.44
Boosting	66.67	0.65	0.60	0.64
RF	62.34	0.55	0.38	0.45
Proposed Method	68.78	0.67	0.57	0.62

Table 7.3 The classification accuracy, recall, precision and F-score of ensemble techniques using testing data.

Learning Method	Accuracy (%)	recall	precision	F-Score
Bagging	51	0.6	0.26	0.34
Boosting	51	0.55	0.34	0.41
RF	61	0.55	0.81	0.65
Proposed Method	76	0.76	0.57	0.64

7.6 Current and Future Scope of ML in Genomics

The current applications of ML in Genomics are:

7.6.1 Gene Sequencing

Experts search the pattern in high volume dataset of gene using ML techniques and then translated to computer model to predict the individual chance of developing any disease or inform the patient the therapy.

7.6.2 Services to Consumer

Some companies gives genomics sequencing services to individuals using ML approaches and advices to their patients and interpret the genetic information.

Few of the future applications of ML are:

7.6.3 Gene Editing

Through gene editing the DNA is altered at the cellular or at the organism level. CRISPR [13] is a gene editing technology which is faster as well as less expensive. Target sequence has to be chosen for the gene editing. This is a daunting process involving many possibilities and unpredictable outcomes for which ML is used to predict the outcome with less cost, reducing time and helps in identifying a target sequence.

7.6.4 Pharmacy Genomics

To study the precision medicine which examines the role of genomics how a medicine responds to an individual. ML models are applied in which the researchers determine the response of a particular treatment or drug on individual.

7.6.5 Newborn Genetic Screening

Some experts believing that in the next decade the new born genetic screen become a standard practice. Through this the EHR (Electronic Health Record) profile of the new born is integrated with the birth data. The ML techniques increase the accuracy of the newborn screening system. ML tools will reduce the number of false positive in many diseases.

7.7 Conclusion

Ensemble technique is a model which is implemented for classifying the high dimensional genomics data which are imbalance in nature. The conventional ensemble learning methods like Bagging and Boosting are working fine and do well in case of known gene classification problems, but the accuracy declines in case of distinct genes or unknown genes. Usually, the genomics data are imbalance and high dimensional data. The proposed technique converts the high dimensional data to several balanced data sets using the clustering method and then generates decision tree on each balanced data. Each genomic data has 143 features, out of which 96 features have been taken and converted into 16 clusters based on the correlation between the features. So, our algorithm constructed 16 decision trees, out of this 16 some decision trees are not taken because they violate the threshold value. The output generated from the decision trees are aggregated using majority voting mechanism for the classification of the new gene type. This chapter investigated the RF bagging technique for classifying the gene variants of Brugada symptom. The experimental results clarified that RF machine learning technique is accurately classify the unknown variants of gene as well.

References

1. Baldi, P. and Brunak, S., *Bioinformatics: The Machine Learning Approach (Adaptive Computation and Machine Learning)*, Second Edition, MIT, Cambridge, MA, 2001.
2. Bhaskar, H., Hoyle, D.C., Singh, S., Machine learning in bioinformatics: A brief survey and recommendations for practitioners. *Comput. Biol. Med.*, 36, 1104–1125, 2006.
3. Hossain, M.A., Islam, S.M.S., Quinn, J.M., Huq, F., Moni, M.A., Machine learning and bioinformatics models to identify gene expression patterns of ovarian cancer associated with disease progression and mortality. *J. Biomed. Inform.*, 100, 103313, 2019.
4. Zou, Q. and Liu, Q., Advanced machine learning techniques for bioinformatics. *IEEE Comput. Archit. Lett.*, 16, 04, 1182–1183, 2019.
5. Probst, P., Wright, M.N., Boulesteix, A.L., Hyperparameters and tuning strategies for random forest. *Wiley Interdiscip. Rev.: Data Min. Knowl. Discov.*, 9, 3, e1301, 2019.
6. Hammann, F., Gutmann, H., Jecklin, U., Maunz, A., Helma, C., Drewe, J., Development of decision tree models for substrates, inhibitors, and inducers of p-glycoprotein. *Curr. Drug Metab.*, 10, 339–46, 2009.

7. Sun, Z., Song, Q., Zhu, X., Sun, H., Xu, B., Zhou, Y., A novel ensemble method for classifying imbalanced data. *Pattern Recognit.*, 48, 5, 1623–1637, 2015.
8. Sun, Z., Song, Q., Zhu, X., Sun, H., Xu, B., Zhou, Y., A novel ensemble method for classifying imbalanced data. *Pattern Recognit.*, 48, 5, 1623–1637, 2015.
9. Hofman, N., Tan, H.L., Alders, M., Kolder, I., de Haij, S., Mannens, M.M., Lombardi, M.P., Dit Deprez, R.H., van Langen, I., Wilde, A.A., Yield of molecular and clinical testing for arrhythmia syndromes: report of a 15 years' experience. *Circulation*, 128, 1513–1521, 2013.
10. Strobl, C., Boulesteix, A.L., Zeileis, A., Hothorn, T., Bias in random forest variable importance measures: illustrations, sources and a solution. *BMC Bioinf.*, 8, 25, 2007.
11. Strobl, C., Boulesteix, A.L., Kneib, T., Augustin, T., Zeileis, A., Conditional variable importance for random forests. *BMC Bioinf.*, 9, 307, 2008.
12. Wang, M.H., Chen, X., Zhang, H.P., Maximal conditional chi-square importance in random forests. *Bioinformatics*, 26, 6, 831–837, 2010.
13. Ledford, H., CRISPR, the disruptor. *Nature News*, 522, 7554, 20, 2015.

8

Feature Selection and Random Forest Classification for Breast Cancer Disease

Shubham Raj, Swati Singh, Avinash Kumar, Sobhangi Sarkar and Chittaranjan Pradhan*

School of Computer Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar, India

Abstract

Medical applications such as detection of the type of cancerous cells has a great use of machine learning in it. As the modern day is progressing the researches in the world of Machine Learning and Artificial Intelligence are expanding speedily. Machine Learning gets better if we keep discovering manifold applications on a wider basis. Predictive models are the foundation of machine learning. If the accuracy is perfect the model will be more efficient and better will be the solution to a specific obstacle. Breast cancer disease is reason for a heavy count of deaths every year. It is the most common type of cancer and the main cause of women deaths worldwide. Breast Cancer is one of the most typical type of disease which occurs in women. It's a fact that cancer can be cured if it's found at an earlier stage but a large number of people are examined with cancer very late. We have discussed the techniques and mathematics behind some algorithms used in machine learning and how that can help in detection of Breast Cancer. Our research aims at determining the features which are mostly responsible for breast cancer. We have used different methods of feature selection for detection of Breast Cancer and further, we have applied Random Forest Classification on our model. We have got a good result and further it can be preferred for the detection of Breast Cancer.

Keywords: Breast cancer, cancerous cell, feature selection, machine learning, random forest classification

**Corresponding author*: chitaprakash@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (191–210) © 2021 Scrivener Publishing LLC

8.1 Introduction

Cancer that occurs in cells of breast or which develops in breast tissue is basically termed as breast cancer. The cells generally form a tumor that can be felt as a lump or can also be detected on an X-ray. There is a need to know that maximum lumps are mild and not breast cancer. In terms of incidence or mortality, breast cancer is presently the utmost common type of cancer among women who are in younger age group.

Types of breast cancer majorly include the following: i) Invasive Ductal Carcinoma (IDC) arise when milk ducts cancer cells get spread outside into other parts of breast tissue. This type of breast cancer is common in men too. ii) Ductal Carcinoma *In Situ* (DCIS) is nonaggressive type of cancer where antidromic cells have been detected in the lining of the milk duct in the breast. Situ means indigenous place. iii) Metastatic breast cancer is contemplated as phase 4 breast cancer which escalates to different major body parts that includes the liver and brain. Symptoms are bones that are more easily fractured or broken (bone), vision disturbances (brain), chronic cough (lungs) and many more. iv) In Inflammatory breast cancer, the cells invade the vessels of skin and lymph of the breast. Primary symptoms include persistent itching and the appearance of a rash or small irritation. In this the breast often turns red, swollen, and warm [1].

Breast cancer is detected by biopsy. In this process, medical practitioner uses a needle device specific for the above purpose and followed by some X-Ray or some other imaging test to pull out a core of tissue from dubious area. Machine learning algorithms are mainly utilized in finding the type of cancerous cells which is done more effectively and accurately. Categorization of tumor cells may take complete 2 days. Machine learning is all about analyzing the given data and extracting the important and useful functions of bond and information from the given dataset and finally creates a computational model for best presentation of the data provided. According to scientist and researchers, machine learning can detect cancer disease at early stage [2]. Ensemble learning technique tries to enhance the predictive models by improving their accuracy.

8.2 Literature Survey

Chandan Banerjee *et al.* discussed about detecting breast cancer by their paper [3]. Here, they chose the dataset named as Wisconsin Breast Cancer dataset. This dataset has a benefit that the quantity of data in dataset is less

making it genuine for establishing ensemble learning models. This benefit gave a reason to choose this model. The phases of their experiment include firstly, preprocessing the dataset then normalizing the dataset then splitting the dataset then constructing three different ensemble models which itself includes Bagging based Ensemble Classifier, Boosting based Ensemble Classifier and Voting based Ensemble Classifier and lastly they compared the results got from each technique by the Python library known as "scikit-learn" the techniques discussed in this paper are experimented accordingly. A detailed analysis of non-identical Ensemble Learning techniques was done by the authors which were applied only on the Classification models. To find the apt Ensemble model for this comparative research should help the researchers for their purposes. Further it can be helpful in the advancement of the outcomes. One of the limitations of this work is that they took the predefined settings of the Classification models. Adjustment of parameters of the base models was decided to overcome the limitation and eliminate by them through so that better outcomes can be achieved.

Shubham Sharma *et al.* carried out a research regarding development of model for observation of breast cancer using latest technique. They particularly explained various sections of ML which was favorable for development of model for finding whether the person is suffering from breast cancer or not [4]. They further presented a comparison of the famous ML algorithms and latest techniques that are being used for observation of breast cancer. The algorithms are performed using a computer structure known as Intel core i7 having RAM size of 16GB. Different open source libraries in python, a web application which is an open source also, to execute the program are elaborately explained in this paper. The algorithms explained here had an efficiency of more than 94%, resolving different type of tumor. Keeping the parameters as accuracy, Recall, F1 Score and precision the performance of various latest techniques of machine learning was compared using the Wisconsin Diagnosis Breast Cancer data set. Obtaining competitive results, it was concluded from the paper that the algorithms used are highly efficient for detection of the breast cancer and its treatment. KNN was recognized to be most efficient as it had the best accuracy, precision and F1 Score. Proving its efficiency to be the best, supervised learning techniques are regarded to be best in cancer research.

Anusha Bharat *et al.* used Breast Cancer dataset which was obtained from the UCI machine learning Repository [5]. Initially the dataset was divided into two sets named as training sets and testing sets. Further 20% data was used to test while 80% data was focused to train the system. They used SVM, KNN, CART and Naive Bayes and compared the accuracy of prediction. KNN technique was the best for overall methodology. For

predictive analysis, SVM was strong though. At last, he used support vector machine using Gaussian kernel which was the most accurate method for finding of breast cancer. Multiclass SVM was used to solve the problem of requirement of binary number of class variable. Further creation of LIBSVM (SVM classes) took place. For better accuracy, they concluded Fine tuning of parameters can help.

Ebru Aydindag Bayrak *et al.* carried out a research work which revolves around different techniques of machine learning used for diagnosis of breast cancer [6]. The dataset used was classification of Breast Cancer (Original) dataset acquired from open source Repository which consists of 699 records classified into benign and malignant. With the help of WEKA tool, they classified WBC (Original) dataset. The modern ML technique and ANN was used. To find the best from the applied ML technique, they are compared on the basis of performance metrics such as precision, accuracy, recall and ROC area. Hence, from the comparison they found out that SVM (Sequential Minimal Optimization Algorithm) came up with best performance with an outstanding accuracy of 96.9957%. The applied technique was SVM and ANN for finding the best performance of ML methods. Vladimir Vapnik firstly explained Support Vector Machines (SVMs). Being the most popular ML classification technique, SVM indicates better performance from its competitive classification technique.

In the work done by Gouda I. Salama *et al.*, they tried to prove by their experimental results that fusion between J48 classifiers and MLP with PCA have the upper hand on other classifiers [7]. Their proposed a model that consists of two phases, namely, training and testing. The training phase was starting with sensor carried with pre-processing then feature extraction and lastly followed by feature selection, testing phase carried the same four steps with the addition of the classification step. The Dataset which they used was from the UCI Machine Learning Repository. Further, they took the help of Confusion matrix for their performance evaluation criteria, Confusion matrix shows the relation between predicted classes and outcomes.

They performed three experiments to evaluate their proposed model, the first experiment was carried out with the help of WBC dataset in which they performed five different steps, firstly, they compared the accuracy of all the five different classifiers (MLP, NB, IBK, J48 and SMO) individually, then testing accuracy with a fusion of SMO and other classifiers at a time, carried on with a fusion of SMO and two other classifiers at a time and then with a fusion of SMO and three classifiers at a time and finally they got that the best accuracy was with using feature selection with MLP

and J48 classifiers which was 97.568%. The second experiment was carried out with WDBC dataset, they again repeated all the steps, But when they recorded the accuracy of fusion of SMO and other two classifiers, they founded decrease in accuracy percentage, and when they did a fusion of SMO and other three classifiers, the accuracy was still lower than a first step and it was found to be around 97.7153%. The third experiment was carried out with the help of WPBC dataset. In the first step they found out that SMO and J48 have better accuracy than others. In the second and third step, the accuracy rate was the same as the accuracy of J48 and SMO individually. In the fourth step, they found that while the fusion of classifiers, using IBK instead of NB as the fourth classifier has the upper hand and after using feature selection with classifiers did not create a difference. Finally, they came to conclusion that fusion of J48 and MLP classifiers with PCA have upper hand on other classifiers.

S.N. Singh *et al.* compared different classifiers such as Simple CART, J48, Logistic Regression, Naive Bayes, Bayesian on the basis of time complexity and accuracy [8]. Experiments were carried out in WEKA. WBCO dataset was used from UCI repository. Their dataset contained 697 instances which included 241 Benign and 456 Malignant. The description of attributes is following: SECS, Bare Nuclei, Bland Chromatin, SCN, CT, Uniformity of cell size, Uniformity of Cell shape, Marginal Adhesion, NN, Mitosis and Class label. Firstly, they pre-processed the input dataset filling the missing values with median. Then it is passed from four data algorithm classified under two category Bayesian and decision tree named as Bayesian Logistic Regression, J48, Naive Bayes, and simple CART. Performance was based in the accuracy formula.

$$Accuracy = true\ negative + true\ positive + true\ positive/true\ negative + false\ positive + false\ negative \tag{8.1}$$

Using Equation (8.1), they got the accuracy percentage of 98.1349, 65.4232, 97.274 and 95.2654% for J48, Bayesian Logistic Regression, Simple CART and Naïve Bayes algorithms respectively. The execution time for Naïve Bayes from four classifiers was least with 0.02 s followed by 0.06 (Bayesian Logistic Regression), 0.29 (Simple CART), 0.20 (J48) respectively.

At last they concluded their research with coming up with conclusion in which they mentioned Simple CART decision tree algorithm is the best classifier considered from the point of accuracy and with least Root relative squared error percentage. Though they accepted that time complexity of simple CART is more and decided to work on it in coming future.

8.3 Machine Learning

Machine Learning is the study or implementation of algorithms; this is achieved on the basis of two phases: train and test. Machine learning algorithms are used in many areas. In today's world everyone wants future prediction, disease diagnosis, etc. Machine learning has established itself as one of the most dependent things which humans need in their life from predicting the weather to finding a disease in the body [9, 10].

Machine learning is based on the concept of training itself by experience and implementing it.

Machine learning is classified in three parts:

1. Supervised learning
2. Unsupervised learning
3. Reinforcement learning.

Supervised Learning
This is the method which is used to enable machines to classify or predict problems or situation based on labeled data given to the machine. For example, you have the data of geometric figures that how circle; square and triangle are on the basis of number of corners. Now you have given some data and you have to find out the shape. You will come with an outcome of what shape it was with help of labeled data provided to you.

Unsupervised Learning
This is method is used when we don't have any labeled data. For example: We have the data of geometrical figure and we don't have any other information about it. So, we find out the hidden structures in the data and relate them with each other and we will have outcome as there are three shapes with 0, 3, 4 edges respectively.

Reinforcement Learning
It is considered as an important type of Machine learning. It will be the biggest demanding type of machine learning in future. In this, an agent learns how to behave in situation or condition by getting performing action and observing the results. For example: Suppose baby is in initial state and is happy then it sees a fire and goes towards it (action performed by the baby) and when it touches fire, it cried (Result) and now baby have understood that going near fire will cause the pain. Like this Reinforcement learning is used.

The most common algorithms in learning Machine learning you will come across would be:

Classification
Suppose you have to predict whether the price of stocks will increase or decrease tomorrow. You will have your outcome as yes or no. This is example of classification. Let's discuss more about classification below

Logistic regression
Logistic regression is a classification algorithm. Here, y can take on only two values 0 and 1. For example, if we are making a model which can predict whether the order is fake or not. Our output can be 0 which means it's not fake or 1 which means it is fake. We take the help of sigmoid function such that our hypothesis lies between 0 and 1.

$$0 \leq h_{\alpha}(A) \leq 1 \tag{8.2}$$

We can now write our hypothesis as,

$$h_{\propto}(A) = g(\propto_0 + \propto_1 A) \tag{8.3}$$

Where $g(\omega)$ represents sigmoid function.

$$g(\omega) = \frac{1}{1+e^{-\omega}} \tag{8.4}$$

Therefore, our hypothesis can be written as:

$$h_{\propto}(A) = \frac{1}{1+e-^{(\propto_0+\propto_1 A)}} \tag{8.5}$$

In order to achieve our discrete classification, we can modify our output from hypothesis as:

$$h_{\propto}(a) \quad \begin{matrix} \geq 0.5, R \rightarrow 1 \\ \leq 0.5, R \rightarrow 0 \end{matrix} \tag{8.6}$$

Our logistic function will behave according to input, like when input is greater than or equal to zero, our output will be greater than 0.5.

$$g(\omega) \geq 0.5, \textit{ when } \omega \geq 0 \tag{8.7}$$

Where $\omega = \propto_0 + \propto_1 A$
Some important equations:

$$\omega = 0, e^0 = 1 \Rightarrow g(\omega) = \frac{1}{2} \tag{8.8}$$

$$\omega \to \infty, e^{-\infty} \to 0 \Rightarrow g(\omega) = 1 \tag{8.9}$$

$$\omega \to -\infty, e^{\infty} \to \infty \Rightarrow g(\omega) = 0 \tag{8.10}$$

Now, here comes the term decision boundary. Decision boundary is the line which separates the value of R. It is not necessary that input of sigmoid function should be always linear; it can even be an equation of circle.

Cost Function:

$$J(\propto) = \frac{1}{n}\sum_{i=1}^{n} Cost(h_{\propto}(A^{(i)}), R^{(i)}) \tag{8.11}$$

$$Cost(h_{\propto}(A), R) = \begin{cases} -\log(h_{\propto}(A)) \textit{ if } R = 1 \\ -\log(1 - h_{\propto}(A)) \textit{ if } R = 0 \end{cases} \tag{8.12}$$

Generalized Cost Function:

$$Cost(h_{\propto}(A), R) = -R \times \log(h_{\propto}(A)) - (1 - R) \times \log(1 - h_{\propto}(A)) \tag{8.13}$$

Finally,

$$J(\propto) = -\frac{1}{n}\sum_{i=1}^{n}[R^{(i)} \times \log(h_{\propto}(A^{(i)})) - (1 - R^{(i)}) \times \log(1 - h_{\propto}(A^{(i)}))] \tag{8.14}$$

Gradient Descent:

$$\propto_j = \propto_j - \beta[derivative\ of\ J(\propto_0, \propto_1)] \tag{8.15}$$

Finally, we get

$$\propto_j = \propto_j - \beta \frac{1}{n} \sum_{i=1}^{n} \left((h_\alpha (A^{(i)}) - R^{(i)}) A_j^{(i)} \right) \tag{8.16}$$

Repeat the above equation until convergence. This equation is similar to what we used in linear regression, so update all the α values simultaneously.

Random Forest Classifier

Random forest classifier is a type of ensemble learning algorithm. It is a supervised learning algorithm. To have a good understanding of it, let's first discuss about ensemble learning. Ensemble Learning is a method in which an object is classified with a combination one or different sets on algorithm. For example: We made a combination of different classifiers such as SVM, Naive Bayes and Decision Tree and then took a vote for the final consideration of result.

For real life example, Suppose you want to buy a mobile phone and you have the certain budget, and suppose you bought phone only by taking opinion of one friend, it may be possible that his opinion about the phone was wrong but if you had asked many different friends opinion and then selected a phone, there would be higher chance that your phone is best among your budget.

In Figure 8.1, we tried to explain the ensemble learning working strategy. Suppose L_1, L_2, L_3, L_4, L_5, L_6 are different types of classifiers. We obtain the output from all these classifiers and make a very strong model L∗. Random forest classifier creates a set of random decision tree from subset of training data. Then it takes the average output.

Regression

Suppose you have to predict the age of man/woman on the basis of height, weight and other factors. You will have outcome as series of numbers and not just yes or no. This turns out to be example of regression.

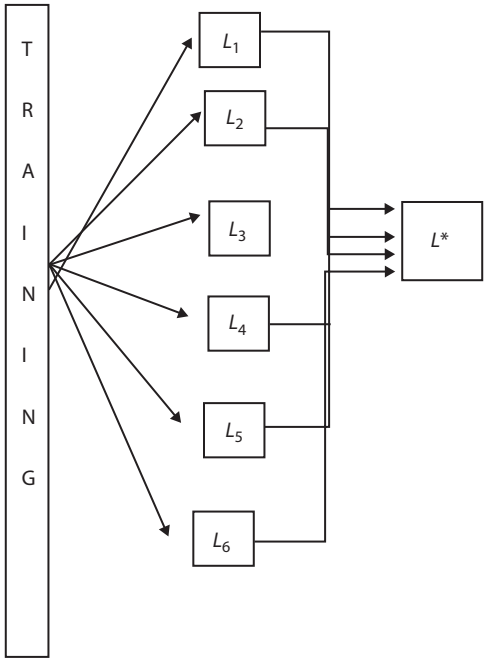


Figure 8.1 Ensemble learning.

Linear Regression

Linear regression model the relationship between two different variables by fitting a linear equation to purified data, where one of the variables is considered as explanatory variable, and the other referred as dependent variable. The linear regression is represented as Equation (8.17).

$$R = a + bA \tag{8.17}$$

Where A: Explanatory variable, R: Dependent variable, a: the intercept, b: slope of the line.

For example, suppose we have to predict the price of a house. We are provided with the training set to train our model. The data which contains SIZE IN FEET^2 which will be our A and the PRICE of the house which will be our R will be divided in two set test and train. We will use Train part to train our model and test part to test the accuracy of our model or find the percentage error.

Now, suppose the number of training set is considered to be 'n'. We will have training set as (A_1, R_1), (A_2, R_2) … (A_n, R_n).

Where, A: Input variable R: Output variable.

Now we will input our input data (SIZE IN FEET^2) in our model and predict the estimated price of the house.

Since we have assumed that our training set has only one input variable or better to say only one feature. Therefore, this is single variable Linear Regression.

Therefore, our hypothesis will be

$$h_{\propto}(A) = \propto_0 + \propto_1 A \tag{8.18}$$

We can clearly see this is in linear equation form.

Now, α_0 and α_1 should be chosen as that the predicted price is very close to actual price or to actually have least error. Here comes the role of cost function. This takes an average difference of all the predictions with the help of the hypothesis with inputs from A's and the actual output R's.

$$J(\propto_0, \propto_1) = \frac{1}{2n}\sum_{i=1}^{n}(h_{\propto}(A_i) - R_i)^2 \tag{8.19}$$

We can also write it as $0.5\,\bar{A}$. This function is also known as "squared error function" or "Mean squared function". We multiplied it with the term 0.5 for simplification of gradient descent (to find the bottom). The derivative of square term will be cancelled out.

Now, our aim is to find the deepest place in the graph where cost function is least. This can be achieved by taking the derivative of our cost function. The direction to move will be found by the slope of the tangent which represents the derivative at that point. We move step by step to the deepest point of the graph. Each step should be the size of β which is considered as learning rate. Smaller the β smaller will be the step and vice versa.

Gradient descent is given by:

$$\propto_j = \propto_j - \beta[derivative\ of\ J(\alpha_0, \alpha_1)] \tag{8.20}$$

j: feature index number.

For each j one should update both the α values simultaneously.

Finally, Gradient descent for Linear Regression will be written as:

$$\propto_0 = \propto_0 - \beta \frac{1}{n} \sum_{i=1}^{n} (h_\propto(A_i) - R_i) \tag{8.21}$$

$$\propto_1 = \propto_1 - \beta \frac{1}{n} \sum_{i=1}^{n} ((h_\propto(A_i) - R_i) A_i) \tag{8.22}$$

We will repeat this until convergence. This will be how we will be training our model. In short to minimize the cost function or to bring it to deepest point of the graph we will keep iterating gradient descent by taking small steps.

If you observe first decrement and then increment, you have passed the deepest point. Try taking lower β and finally you will achieve the point. This was for single variable or feature. Now suppose if there are multiple variables, for example take m features, then our hypothesis will be

$$h_\propto(A) = \propto_0 + \propto_1 A_1 + \propto_2 A_2 + \ldots + \propto_m A_m \tag{8.23}$$

And gradient descent will be

$$\propto_j = \propto_j - \beta \frac{1}{n} \sum_{i=1}^{n} \left((h_\alpha(A^{(i)}) - R^{(i)}) A_j^{(i)} \right) \tag{8.24}$$

where j = 0, 1, 2, …, m and $A_0^{(i)} = 1$. Repeat this until convergence.

Clustering

Suppose you have the data of customers of a brand. Now you have to find group of people who spend more on the basis of their past records. Finding that group of people will be an example of clustering.

8.4 Feature Engineering

As we said above, Machine learning can also be used to predict. However, for best outcome, we should have the best data. Feature engineering plays a major role in it.

Initially, all data are in raw form and we need to extract useful features from it before giving it to algorithm for better training. Extraction of useful

data from raw data is the most important step. Nearly every data is in form of tables. Each column represent feature, but not every feature will be useful. Modifying these features in a new set of data will be more useful. Feature engineering is not just about selecting useful features.

Feature engineering plays an important role in increasing performance of machine learning model. Feature engineering creates a huge difference between a good and bad model of machine learning, so it becomes more important for the researcher to implement the feature engineering more accurately.

To solve any problems, there are some basic steps which are implemented:

1. Gathering the raw data.
2. Making it useful data, to be more precise cleaning data.
3. Feature engineering.
4. Describing the model.
5. Training and Testing your model.
6. Prediction.

This technique aids in converting data into a design that the selected machine learning algorithm can easily understand. Also, it is a technique that helps in extracting more detailed information and components from the raw data.

Human readers can easily comprehend the raw input types, like longitudinal data timestamps, or categorical variables but there is a requirement of some unboxing to be easily understandable by the chosen machine learning algorithm. Hence, feature engineering is not only dependent on deep understanding of the form that different algorithms exhibit and depict data, but it also depends on an ardent skill to perform domain specific knowledge to the selected modeling objective.

While predicting, the most common information available is the date of birth of a person and this date of birth is converted into a single number—age of the person at the time of prediction. Well, there are also more options available like in place of using the age in years; we can define Boolean variables for various groups of age ranges. We can take an example like under 10 years, under 18 years, under 28 years, under 50 years and so on. The efficiency of the machine learning algorithm increases if we use Booleans instead of age number.

Prediction becomes more difficult when some complex data is available. Like in some health sectors, diagnosis codes ICD-9 or ICD-10 are available which indicates the sole reason for which the medical services were provided.

Now using diagnosis codes for the machine learning methods becomes inappropriate. Different variables can be assigned with particular ICD codes but as there are large numbers of ICD-10 codes (about 68,000) it becomes difficult as well as inconvenient. Alternatively, we can choose different sets of codes representing a single variable and signifying specific medical issue and name these sets as groupers. For Example, a category ICD-10 code 209.5 representing hypertension, ICD-9 code 219.5 and likewise many categories can be created. Hence it becomes meaningful to use these diagnosis codes in sets or groups instead of using them individually.

8.5 Methodology

8.5.1 Dataset Collection

Our dataset has been collected from "UCI ML" repo. Wisconsin Diagnosis Breast Cancer dataset is the topic of the work. The dataset has 32 attributes which are important for detection of kidney disease. The dataset have different type of data types that are being used. Data types which are used are object and float64. Diagnosis, radius_mean, texture_mean, perimeter_mean, area_mean, etc. are the most important variables. Table 8.1 shows the total attributes which are used here.

The output label consists whether the cancer is Benign or Malignant. We have total 357 Benign and 212 Malignant. Figure 8.2 shows the total number of output label.

8.5.2 Proposed Work

The dataset had 569 attributes which was used for the development of model. The basic work was started with the preprocessing which is the root of development of any model for better accuracy. First of all we checked for null values if there any and was removed if found. The first column that is id was not taken into consideration as it will not help in development of model. We splitted our dataset into x and y. The x contains our features and y is the target which contains the values of breast cancer. The number of benign cancer was 357 and malignant was 212. Later on we decided to do something new and we tried feature selection which undoubtedly was a great method to explore the data and withdraw valuable conclusions from it. So we started with feature selection and evaluated our model with classification algorithm which is explained in more detail in next section.

Table 8.1 Dataset Attributes.

Attribute	Data Type	Attribute	Data Type
Diagnosis	object	area_se	float64
radius_mean	float64	smoothness_se	float64
texture_mean	float64	compactness_se	float64
perimeter_mean	float64	concavity_se	float64
area_mean	float64	radius_worst	float64
smoothness_mean	float64	texture_worst	float64
compactness_mean	float64	perimeter_worst	float64
concavity_mean	float64	area_worst	float64
concave_points_mean	float64	smoothness_worst	float64
symmetry_mean	float64	compactness_worst	float64
fractal_dimension_mean	float64	concavity_worst	float64
radius_se	float64	concave_points_worst	float64
texture_se	float64	symmetry_worst	float64
perimeter_se	float64	fractal_dimension_worst	float64

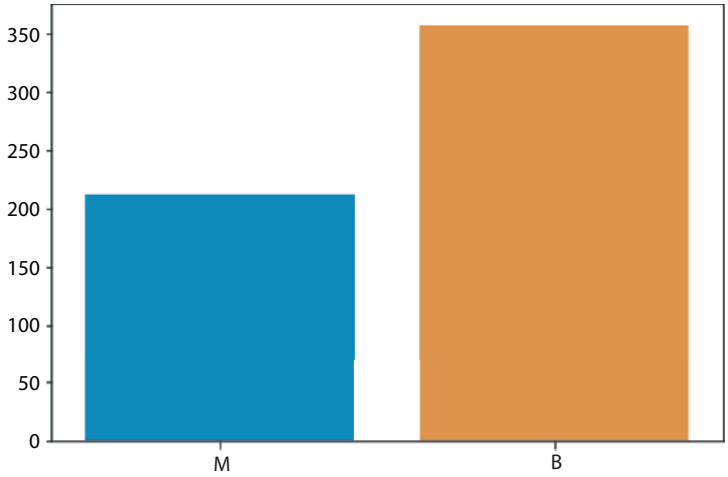


Figure 8.2 Output label.

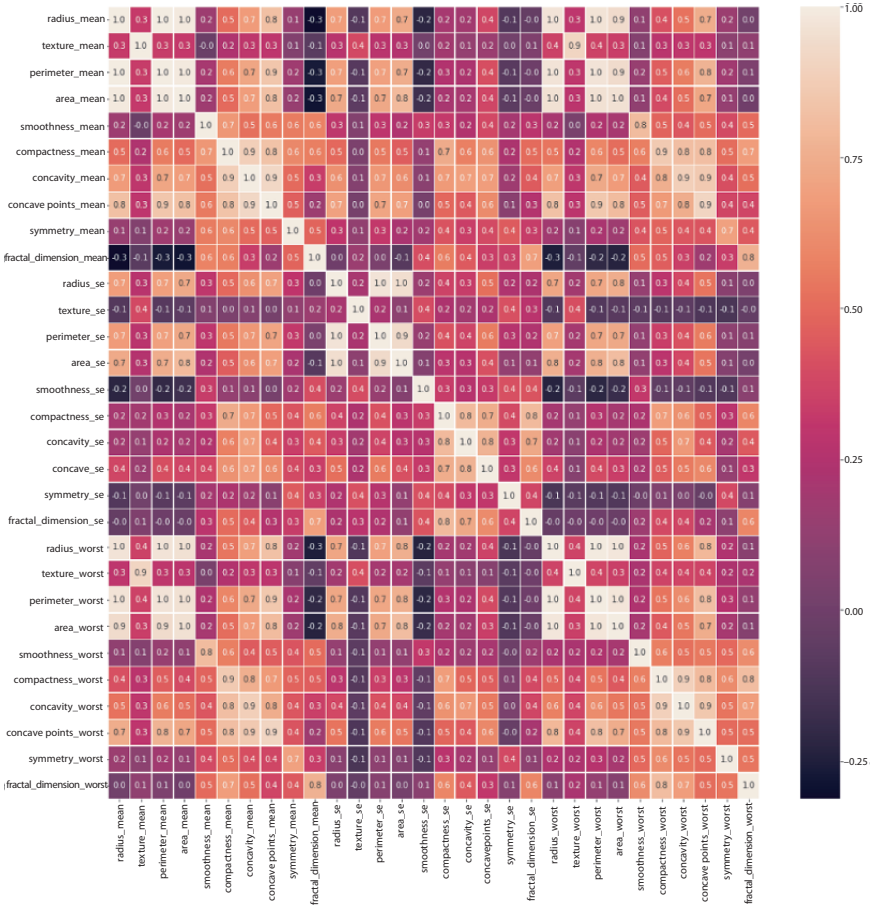


Figure 8.3 Heat map.

Here, we have used feature selection with different methods that are discussed in the next section .We have used classification algorithm to train our model and predict. Figure 8.3 shows the heat map which tells us how to choose features i.e. feature selection while using heatmap (correlation matrix).

8.5.2.1 Selection of Feature by Means of Correlation and Accuracy Calculation Using Random Forest Classification

We can see in the heat map figure, that attribute radius_mean, perimeter_mean and area_mean are correlated with each other so we will use only area_mean. Similarly different attributes were correlated that's why we did

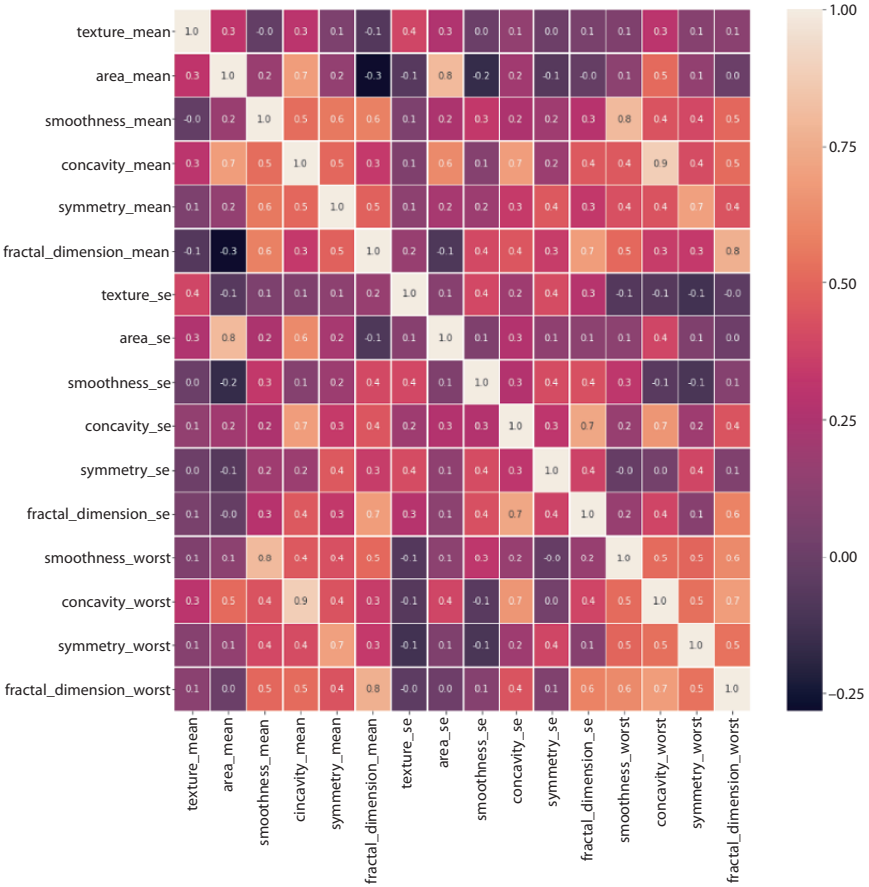


Figure 8.4 Correlation matrix.

not considered it and we only choose concavity_mean, area_se, area_worst, concavity_worst, concavity_se, texture_mean, area_mean for our model.

We can see that we have dropped the correlated features in Figure 8.4, there are no more correlated features.

On applying Random forest, we get the accuracy of 95.32%. The confusion matrix is shown in Figure 8.5.

8.5.2.2 Feature Selection Using one Variety and Accuracy Calculation Using Random Forest Classification

Here, we need to select the number of features that we will use. Let's say, k where k can be number of features that we will use can be 5 or 10 or 15?

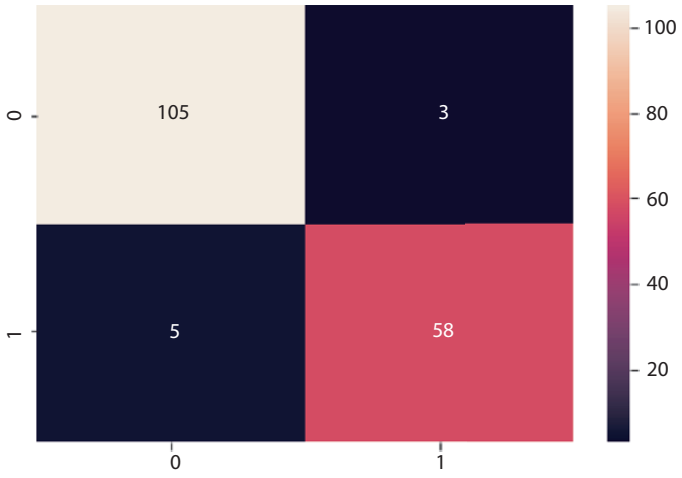


Figure 8.5 Confusion matrix.

The answer is only trying or intuitively. We choose k = 5 and find best 5 features. Top 5 features to classify are area_mean, area_se, texture_mean, concavity_worst and concavity_mean. We only used these best five features and got the accuracy of 94.15%. Figure 8.6 shows the confusion matrix.

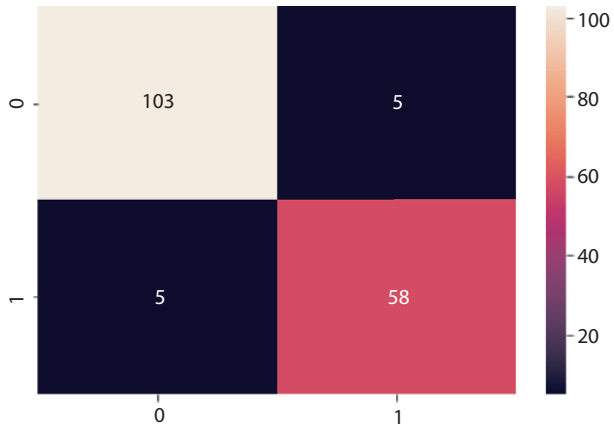


Figure 8.6 Confusion matrix.

8.5.2.3 *Feature Elimination Using RFE and Classification Using Random Forest*

It uses one of the classification methods (random forest in our example), allocate weights to each of features. The feature whose absolute weights are the smallest are trimmed from the available features set. This process is duplicated recursively on the trimmed set until the desired number of features.

Chosen 5 best features by rfe is area_mean, texture_mean, concavity_mean, concavity_worst, area_se. They are exactly similar with previous (selectkBest) method. Therefore, we do not need to calculate accuracy again. In short, we can say that we made good feature selection with rfe and selectkBest methods.

8.6 Result Analysis

In the first method, we used Feature Selection with correlation and observed the correlation between different features and got the accuracy of 95.32%. In the second method we used Univariate feature selection and selected top 5 feature using Method selectKBest and got the accuracy of 94.15%. In the last method we have used Recursive feature elimination (RFE) with random forest and observed that the top 5 feature are same as we got in the previous method so we did not need to find the accuracy for it.

The comparison analysis of all the methods that had been used in the development of the model is shown below in Figure 8.7.

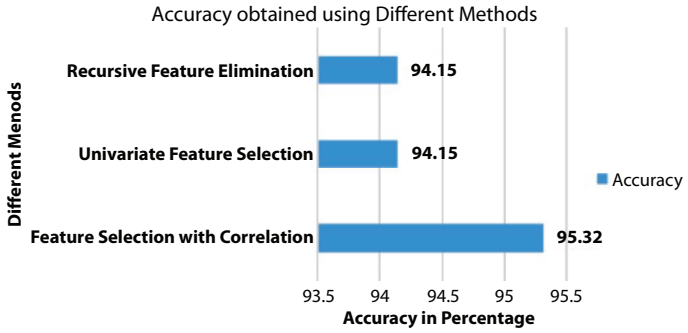


Figure 8.7 Comparison of different methods.

8.7 Conclusion

In this chapter, we tried to present the different aspects of feature selection and visualization of data which can lead to a better decision-making model. Initially, dataset contains 33 features but after selection of feature, we drop this number from 33 to 5 with accuracy 95%. We got the satisfactory accuracy using selection of feature and classification method. In the future, we will try to develop more features and obtain good accuracy on our model.

References

1. Gupta, C. and Gill, N.S., Machine Learning techniques and Extreme Learning Machine for Early Breast Cancer Prediction. *Int. J. Innov. Technol. Exploring Eng.*, 9, 4, 163–167, 2020.
2. Maity, N.G. and Das, S., Machine Learning for Improved Diagnosis and Prognosis in Healthcare. *IEEE Aerospace Conference*, USA, 2017.
3. Banerjee, C., Paul, S., Ghoshal, M., A Comparative Study of Different Ensemble Learning Techniques using Wisconsin Breast Cancer Dataset. *IEEE International Conference on Computer, Electrical & Communication Engineering*, India, 2018.
4. Sharma, S., Aggarwal, A., Choudhury, T., Breast Cancer Detection using Machine Learning Algorithms. *IEEE International Conference on Computational Techniques, Electronics and Mechanical Systems*, India, 2018.
5. Bharat, A., Pooja, N., Reddy, R.A., Using Machine Learning Algorithms for Breast Cancer Risk Prediction and Diagnosis. *IEEE International Conference on Circuits, Control, Communication and Computing*, India, 2018.
6. Bayrak, E.A., Kirci, P., Ensari, T., Comparison of Machine Learning Methods for Breast Cancer Diagnosis. *IEEE Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science*, Turkey, 2019.
7. Salama, G.I., Abdelhalim, M.B., Zeid, M.A., Experimental Comparison of Classifiers for Breast Cancer Diagnosis. *IEEE International Conference on Computer Engineering & Systems*, Egypt, 2012.
8. Singh, S.N. and Thakral, S., Using Data Mining Tools for Breast Cancer Prediction and Analysis. *IEEE International Conference on Computing Communication and Automation*, India, 2018.
9. Jamil, S., Abbas, M.S., Umair, M., Habibb, F., Fawad, Hussain, Y., A Novel Deep Neural Network CanNet for Malignant Detection. *IEEE International Conference on Information Science and Communication Technology*, Pakistan, 2020.
10. El-Kafrawy, P., Manhrawy, I.I.M., Fathi, H., Qaraad, M., Kelany, A.K., Using Multi-Feature Selection with Machine Learning for De novo Acute Myeloid Leukemia in Egypt. *IEEE International Conference on Intelligent Systems and Advanced Computing Sciences*, Morocco, 2019.

9

A Comprehensive Study on the Application of Grey Wolf Optimization for Microarray Data

Swati Sucharita[1], Barnali Sahu[1]* and Tripti Swarnkar[2]

[1]*Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneshwar, India*
[2]*Department of Computer Application, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneshwar, India*

Abstract

The new research trend is to achieve improved elucidation efficiency in the area of global optimization for various functional and real-life applications**.** Various nature inspired computational algorithms are now a promising approach to discovering new and robust methods, grounded on the theory of biological evolution of nature. The current survey focuses on a recent emerging optimization algorithm i.e. Grey Wolf Optimization (GWO) in various health care domains, and specifically in the Microarray data application. There is various nature inspired algorithms present in the literature and GWO is a recent developed algorithm. It has been applied in different engineering fields and successfully achieved the objectives. In the current research we have given a systematic review on the application of GWO in health care domain and gene selection in microarray data. Gene selection in health care is an impending arena that attains huge importance for extracting useful genes from the thousands of genes in microarray dataset. The gene selection plays a vital role in large datasets to increase the efficiency of classification to derive the important genes for high dimensional classification. The study also focuses on the variations of GWO and its application in medical field as well as in medical data classification and gene selection.

**Corresponding author*: sahu.barnali08@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (211–248) © 2021 Scrivener Publishing LLC

Keywords: Nature inspired algorithm, optimization algorithm, Grey wolf optimization, Microarray data, gene selection, variations of Grey wolf optimization, medical data, classification

9.1 Introduction

DNA Microarray technology has transmuted biology in the last years. This technology has facilitated researchers to study thousands of genes simultaneously rather than researching a single gene. This collective approach has shown a pathway to researchers to study about a disease caused in humans. Microarray data can be used for various drug design, disease prediction, classification of cancer, treatment, prevention and many more. The prime contribution of DNA microarray is towards solving gene expression problems [1]. The major issue with microarray technology is due to the presence of many irrelevant genes, and these are more in number as compared to the sample numbers in a gene expression problem. While solving the general classification problem, the technique of Machine Learning does not perform well for more genes compared to samples. The processing of the original microarray data with redundant genes has a significant effect on classification algorithms efficiency [2]. The identification of appropriate genes is a challenging task for microarray dataset as small number of genes is a major role player in many biological processes. In Machine learning approach the collection of features is also known as collection of variable subset or attributes [2]. The selection of data mining features is a pre-processing approach for the active analysis of the data in order to select the most appropriate data by reducing the redundancy resulting in the reduction of the feature space to predetermined targets [3]. In gene expression data analysis, gene selection or biomarker selection is choosing the appropriate genes for the construction of classifiers to classify a sample as diseased or normal [3]. The gene selection generally minimizes the number of genes by removing unwanted, irrelevant genes out of number of genes. The feature selection process is the common problem encountered today in the field of global combinatory optimization. Finding out the appropriate gene subset is usually tough and deriving the optimal subset of genes has been shown to be NP-hard [3]. Due to this several meta heuristic algorithms were used to make gene selection of training and testing data, i.e. Genetic algorithm (GA) [5], Artificial Bee colony (ABC) [6], Ant Colony Optimization (ACO) [4], Particle Swarm Optimization (PSO) [7], Harmony Search Algorithm (HSA) [8], Firefly Algorithm (FA) [9], Bat Algorithm (BA) [10] and many more. In this study a systematic review on Grey Wolf Optimization (GWO) [11] for gene selection in microarray data

and a comparative analysis of GWO in various healthcare applications is presented. GWO is a meta-heuristic Swarm intelligence optimization algorithm now used by many researchers to solve the problem of optimization [5]. Meta heuristic algorithms are the higher-level approach intended to hunt for the top solution with less overhead computation over a large search [10]. The meta heuristic algorithms are having very simple idea, easy implementation and no need of inclination of data to find out the near optimal solution [6]. These algorithms are mostly designed to be applied over any optimization problems. The major difference between heuristic and meta-heuristic is that heuristics find best solution for a particular problem but they are designed for a specific problem. Metaheuristic on the other hand are the general-purpose approach which can be applied to solve almost all kind of optimization problem. In this study Grey wolf Optimization (GWO) is used which is a population-based optimization algorithm which mimic the animals especially inspired by the hierarchy of leadership and the process for chasing grey wolves [11]. Since the literature lack with a systematic study on GWO application for healthcare data specifically microarray data. Therefore, we motivated to present a systematic analysis of the applications of GWO on healthcare data, the variants of GWO and its application on the health care data along with microarray data.

The chapter remains' is structured as follows. Section 9.2 deals with details on microarray data. The overview of GWO Algorithm and its variants are outlined in Section 9.3. The literature review for GWO in Medical applications and the application in Medical domain for Microarray Dataset is included in Sections 9.4 and 9.5 of the chapter respectively. Finally, conclusion and future scope is in Section 9.6.

9.2 Microarray Data

This section presents an outline on DNA microarray technology and the key rudiments of microarray data. Microarray or Chip technology allows simultaneous analysis on many samples containing thousands of genes on a solid substrate. The schematic representation of a DNA microarray technology is presented in Figure 9.1 depicting the hybridization process. This influential technology has helped many researchers to monitor infectious genes present in an organism. Many up and downs or swipe of genes can be analyzed using this microarray technology. Many genes are in normal state but there is abnormality in many genes which can be a cause of a disease. The DNA microarray technology helps the biologists to analyse thousands of genes in a gene expression profile.

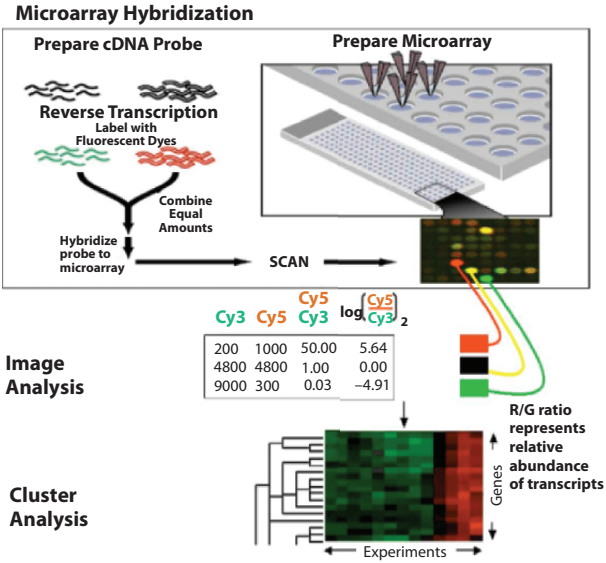


Figure 9.1 Schematic representation of DNA microarray technology [12].

Microarray data Analysis deals with extraction of the relevant genes out of thousands of genes in a gene expression. The global analysis of genes allows researchers for finding out the cause of a specific disease [12]. Cancer classification, prediction and diagnosis are the major applications of microarray data analysis. The marker genes refer to the genes due to which changes occur in gene expression profile [13]. The identification of marker genes from the thousands of genes is a major issue with microarray data. There are three different techniques for which the microarray data can be analysed: (1) class comparison (Mechanistic Analysis), (2) class prediction, and (3) class discovery (clustering) [14]. Class comparison focus on deriving the genes those are expressed differently in comparison to other genes. Class prediction (classification) focus on deriving the class of samples and Class discovery is the discovery of the groups which are related (similarity in gene expression). Figure 9.2 represents the workflow of analysis of microarray data.

9.3 Grey Wolf Optimization (GWO) Algorithm

GWO is a recent nature inspired metaheuristic optimization algorithm developed by Australian Scholar Mir Jalili in 2014 [15, 16]. It mimics

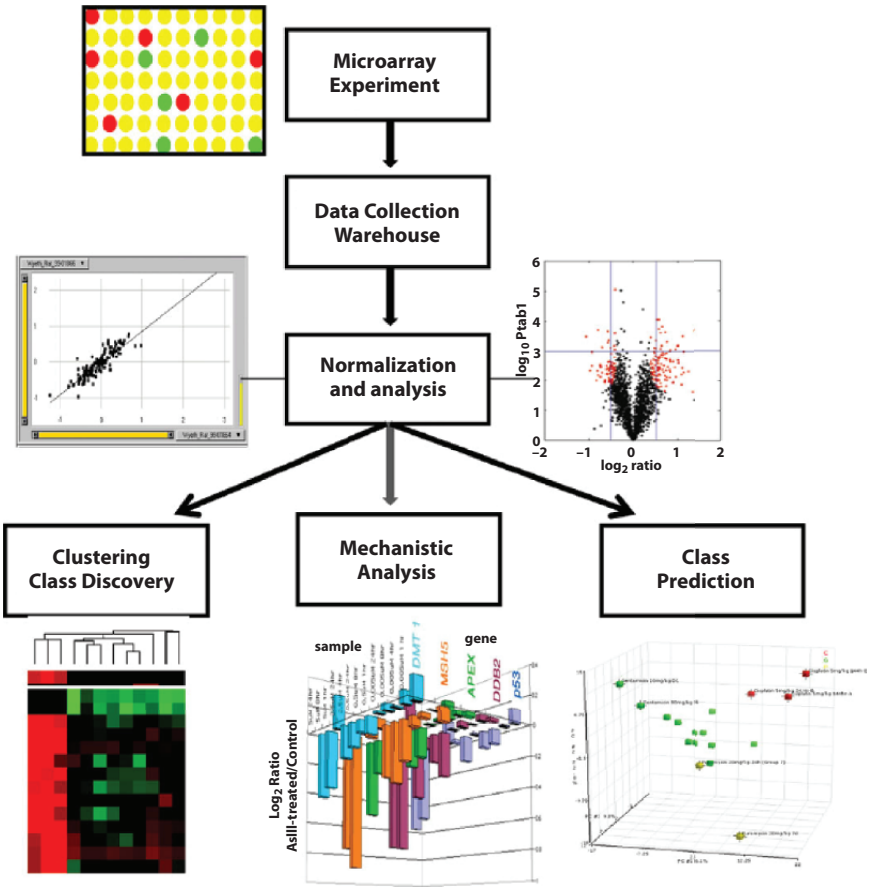


Figure 9.2 Model for microarray data analysis [13].

the hierarchical mechanism depicted in Figure 9.3 and behavior of grey wolves. The wolves capture the prey through a series of processes, such as 1. Surrounding, 2. Hunting and 3. Attacking [16]. The wolves are categorized based on their strength for attacking the prey. Four categories of wolfs are formed called alpha (α), beta (β), delta (δ) and omega (ω) [15, 16], each having its own behavior while hunting. The dominant and most powerful among all is alpha (α) which acts as leader in the entire pack while hunting. The alpha can be male or female wolf and the decision is always taken by the alpha which is followed by rest of the wolves. When the alpha pack is sick or deceased, beta (β) takes the role of alpha (α). Beta (β) acts as an advisor of alpha of which wolf may be male or female [16]. The delta (δ) and omega (ω) wolfs lack in power than alpha and beta.

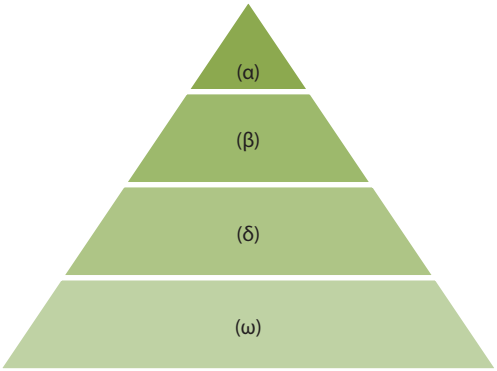


Figure 9.3 Social hierarchy of grey wolves.

The major steps followed while searching and hunting the prey in GWO is as follows:

Step 1. When a prey is detected, monitor first, chase and then approach the prey.

Step 2. When the prey runs, the wolves carry on, surround and attack before the prey stops moving.

Step 3. The assault eventually begins.

9.3.1 Principle of GWO

The exploration and exploitation are the two basic principles of almost every meta heuristic algorithm. In exploration the search space for the better optimal solutions out of many candidate solutions is found out. On the other hand, in exploitation the exploration phase is verified to meet the global optimization. Every meta heuristic algorithm should balance these two promising factors to find the accuracy in getting the global optimum [11]. In GWO algorithm, prey searching is the ability to explore and prey attacking is the ability to exploit [11]. In exploration the algorithm tries to search a wide range of area to find out more search space of the problem [16]. The exploitation is limited to a small portion and the main aim is to advance the predictable solutions reached by deriving the nearest solution for each solution [16]. That is why the normal differences in the results should be brought together to the global optimum.

9.3.2 Mathematical Model of GWO

The best solution is called as alpha (α), the second best solution is called as beta (β), the third best solution is called as delta (δ) and the rest are called omega (ω) [16]. All the omegas are guided by alpha, beta and delta during searching for optimization and the entire hunting process. Omega (ω) acts like a scapegoat [17].

9.3.2.1 The Encircling

The behavior of the grey wolves can be clearly understood during encircling around the prey using the mathematical equations below. Using these equations, the grey wolves change their position in the solution area just about the prey and the wolves can go on updating their location around the search space with n dimensions.

"t" signifies the current iteration.

$\vec{A}$, $\vec{C}$, $\vec{D}$ represent the coefficient vectors.

$\vec{X}_P$ signifies the present location of prey.

$\vec{X}$ signifies present the location of the grey wolves

$$\vec{D} = \left|\vec{C}.\vec{X}_P - \vec{X}(t)\right| \tag{9.1}$$

$$\vec{X}(\mathrm{t}+1) = \vec{X}_P - \vec{A}.\vec{D} \tag{9.2}$$

The parameters $\vec{A}$ and $\vec{D}$ are the combination of the controlling parameters $\vec{a}$, $rand_1$ and $rand_2$. The interval of random numbers $rand_1$ and $rand_2$ are the random numbers that lie between .The regulatory parameter $\vec{a}$ changes $\vec{A}$ and roots the omega (ω) to approach or run away from the dominant wolves alpha (α), beta (β), delta (δ) [17]. $\vec{A}$ and $\vec{C}$ can be computed as follows.

$$\vec{A} = 2.\vec{a}\ .rand_1 - \vec{a} \tag{9.3}$$

$$\vec{C} = 2.rand_2 \tag{9.4}$$

$\vec{C}$ helps in exploration which returns a random value between interval [0,2]. $\vec{a}$ is defined to be decreasing linearly from 2 to 0 as the iteration t raises i.e.

$$\vec{a} = 2(1 - t/t_{max}) \tag{9.5}$$

If $|\vec{A}| > 1$, the omega (ω) run away from the dominant or the prey & explore more space. If $|\vec{A}| < 1$, then omega (ω) approaches the dominant alpha (α), beta (β), delta (δ).

9.3.2.2 Hunting

The next major task of grey wolf is to start hunting for prey once encircling of prey. The wolf alpha (α) leads the hunting process. The other wolves obey the decision of alpha (α) during the entire process of hunting. However, we have no clue on the exact location of the optimal solution (prey) in the search space. We assume alpha (α) as the best initial solution. Beta (β) and Delta (δ) have the knowledge about the location of the prey [11]. The behavior of alpha (α), beta (β) and delta (δ) can be computed as below.

$$\vec{D}_\alpha = |\vec{C}_1 \quad \vec{X}_\alpha - \vec{X}| \tag{9.6}$$

$$\vec{D}_\beta = |\vec{C}_2 \quad \vec{X}_\beta - \vec{X}| \tag{9.7}$$

$$\vec{D}_\delta = |\vec{C}_3 \quad \vec{X}_\delta - \vec{X}| \tag{9.8}$$

The location of the wolves can be calculated using the equations

$$\vec{X}_1 = |\vec{X}_\alpha - \vec{A}_1 \quad \vec{D}_\alpha| \tag{9.9}$$

$$\vec{X}_2 = |\vec{X}_\beta - \vec{A}_1 \quad \vec{D}_\beta| \tag{9.10}$$

$$\vec{X}_3 = |\vec{X}_\delta - \vec{A}_1 \quad \vec{D}_\delta| \tag{9.11}$$

The best location can be found out using the equation

$$\vec{X}(t+1) = \vec{X}_1 + \vec{X}_2 + \vec{X}_3 / 3 \tag{9.12}$$

9.3.2.3 *Attacking Prey: (Exploitation)*

The grey wolves complete hunting process by attacking the prey once the prey stops moving [16]. To construct a model mathematically for approaching the prey, we need to reduce the value of $\vec{a}$ and the variation range of $\vec{A}$ also reduced by $\vec{a}$ [11]. In other words, a random value is in the interval $[-a, a]$ where *a is* reduced from 2 to 0 during the iterations [11]. When random values are in [−1,1], the next position of a search agent may be in any location between its current position and prey position [11]. If $|A|<1$, which forces the wolves to attack towards the prey.

9.3.2.4 *Search for Prey: (Exploration)*

The grey wolves searching is done according to the location of alpha (α), beta (β), delta (δ). $\vec{C}$ is a used for exploration purpose that returns a random ranges of value between intervals [0,2]. It modifies the prey's role in determining the next position. This is a robust input when C>1, the result sinks further in the direction of the prey. This parameter delivers random values instead of the iteration number in the case of any local optima stagnation, exploration during optimization will be emphasized [11]. $\vec{A}$ is also used for exploration based on $\vec{a}$, where it decreases from 2 to 0 linearly. Since random components are comprised in this parameter, the range in the interval of [−2,2] for $\vec{A}$ changes. Exploration is done when A>1 or A<−1, whereas there is highlight on exploitation when $-1<\vec{A}<1$ [11].

9.3.3 Algorithm and Flow Chart of GWO

Grey Wolf Optimization Algorithm

Step 1: Set t = 1, where t is the current iteration.

Step 2: Locate all the required parameters a, A, C and choose the population size (grey wolves) randomly, calculate the fitness and derive alpha(α)which is the best fittest search element (X_α), beta(β) the next best element (X_β), delta(δ) the next element (X_δ) and rest are omega(ω) i. e(X_ω).

Step 3: Update the location of alpha(α), beta(β), delta(δ) and rest omega(ω) using the equation (3.1) and (3.2) of the mathematical model

Step 4: Check the stopping criteria. if it satisfies i.e. $t < t_{max}$ go to step no 2 otherwise the search elements are X_α,X_β,X_δ,X_ω. Using Equations (3.3) to (3.12) of the mathematical model.

Step 5: Stop

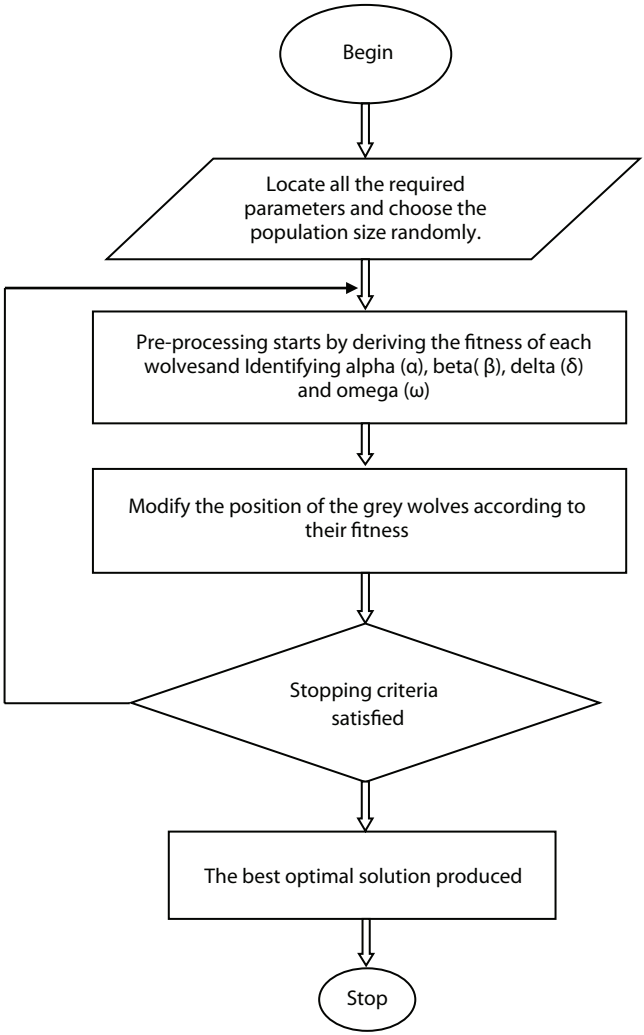


Figure 9.4 Flow chart of GWO.

The flow chart of GWO is shown in Figure 9.4.

9.4 Studies on GWO Variants

This section focuses on the advances of GWO algorithm. In order to resolve the real world optimization problem the various alteration of GWO algorithm has been done. The mechanisms of GWO can be altered by adding new operators, updating mechanisms, combining GWO with other

metaheuristics to enhance the performance and so on. We divide advances of GWO into three different aspects.

i) Hybridization of GWO with other metaheuristic methods include Genetic Algorithm (GA) [5], Particle Swarm Optimization algorithm (PSO) [7], Firefly algorithm (FA), Crow search algorithm (CS), Harmony Search algorithm (HS) [8], Ant Lion Optimization algorithm (ALO), Sine Cosine algorithm (SCA), immune system (AIS), Tabu Search (TS), Ant colony optimization algorithm (ACO), simulated annealing (SA), Artificial Bee Colony optimization algorithm, Differential Evolution algorithm (DE), biogeography-based optimization (BBO), and other meta heuristic algorithms.
ii) Extensions of GWO to other optimization fields, including binary, multi-objective, discrete, constrained optimization and other extensions.
iii) Modifications of GWO, including fuzzy, quantum-behaved, chaotic, topology, bare-bones, opposition-based GWO and various other modifications.

9.4.1 Hybridization

Hybridization plays a lively role in refining the search performance of an algorithm. Two or more algorithms can be combined together for getting a hybridized algorithm. The main motto behind hybridization to have improvements in computation and to achieve high accuracy. It combines the advantage of both the algorithms to an improved version of GWO [18]. The hybridized GWO helps in empowering the exploration and exploitation property of GWO. In this section Table 9.1 presents the literature survey on the different hybridized GWO algorithm not restricted to medical domain. The review has focused on the method objective and findings.

9.4.2 Extensions

The better performance of GWO algorithm in solving single optimization problems has motivated many researchers to extend the algorithm to solve different multi objective real-world problems in the field of engineering, science and medical domains consisting of multi objective optimization, binary optimization, constrained optimization and so on. A brief analysis of extended GWO is shown in a tabular form in the present section (Table 9.2).

Table 9.1 Literature Review for Hybridization of GWO Algorithm.

Paper	Methodology Used	Objective of the method	Dataset used	Findings of the Paper
[19]	Hybrid GWO-PSO	The goal is to provide solution to the question of Economic Load Dispatch (ELD).	Three different test system for solving ELD load demand.	The proposed algorithm shows more effective as compared to other metaheuristic algorithms.
[20]	Hybrid GWO-SCA	To develop a hybridized algorithm for classical and real-life optimization problem. The GWO is used for exploitation and SCA is used for exploration.	1. Twenty-two benchmark tests data 2. Five bio-medical datasets 3. A set of sine data.	The computational and statistical results achieved using the hybrid GWO-SCA method is compared to other metaheuristics and the results show that the hybrid model proposed can be very effective in solving benchmark and real-life problems [20].
[21]	Hybrid GWO-FPA (Flower Pollination Algorithm)	The objective is to use the hybridized algorithm to get better solution in solving optimization problems with or without un known search areas.	Six benchmark test data.	The experimental result shows that the proposed algorithm gives better performance in solving the practical real-life problems as compared to other algorithms.

(*Continued*)

Table 9.1 Literature Review for Hybridization of GWO Algorithm. (*Continued*)

Paper	Methodology Used	Objective of the method	Dataset used	Findings of the Paper
[22]	Hybrid GWO-ABC (Artificial Bee Colony Algorithm) [22]	The novel initialization approach is to build an initial population of already successful individuals to establish a steady ground for the rest of the GWO-ABC algorithm to be implemented [22].	Twenty-seven synthesis benchmark functions [22]	The algorithm proposed is matched with other standard intelligent meta heuristics and the result shows the better performance in case of the proposed algorithm. GWO-ABC as a viable alternative for the design of a controller with optimum parameters and system efficiency improvement.
[23]	Hybrid GWO-Bat Algorithm	The objective is to solve the global function optimization problem by utilizing the GWO exploration skill and BA's local search capability.	Thirty standard benchmark function from CEC2017 benchmark suite.	The test results show that the hybridized algorithm give better accuracy as compared to other bench mark algorithms such as original GWO, BA and whale optimization algorithm [23].

(*Continued*)

Table 9.1 Literature Review for Hybridization of GWO Algorithm. (*Continued*)

Paper	Methodology Used	Objective of the method	Dataset used	Findings of the Paper
[24]	Hybrid GWO-Pattern Search Algorithm	The major objective of the method is to solve the various optimization problems such as 1. Non-linear optimization 2. Highly constrained engineering design problems 3.Engineering optimization problems.	1. Twenty-three benchmark problem 2. Two real life biomedical problems.	The operating process of the proposed GWO-PS algorithm was found to be better than the regular GWO optimizer, Ant Lion Optimizer, Month Flame Optimizer, Sine–Cosine Optimizer and other newly recorded metaheuristic algorithms.
[25]	Hybrid GWO-CSA (Crow Search Algorithm).	A hybrid GWO-CSA is proposed for function optimization and selection of the features [25].	Twenty-three benchmark functions for function optimization and twenty-one UCI repository datasets are used.	The result indicates improved function optimization capability of the proposed algorithm, faster convergence and takes less time in giving optimization result and gives greater classification accuracy and number of optimal features selected in accordance with other metaheuristics [25].

(*Continued*)

Table 9.1 Literature Review for Hybridization of GWO Algorithm. (*Continued*)

Paper	Methodology Used	Objective of the method	Dataset used	Findings of the Paper
[26]	Grey Wolf Optimization and Neural Network algorithms are used.	The objective of the hybrid algorithm is to solve global numerical optimization problem.	Twenty-three unconstrained benchmark functions are used [26].	The proposed algorithm is superior or equal when compared with thirteen different metaheuristics algorithm. It has better solution quality and is more efficient in computation as compared to other metaheuristics.
[27]	Hybrid GWO-GA (Genetic Algorithm)	The proposed approach has the objective of solving the combinatorial problem i.e. Service composition and optimal selection (SCOS) using crossover and mutation operators to improve exploration capability [27].	Two Testing Scenarios and QOS preferences of different users.	Not only does the proposed method allow a continuous structure algorithm such like GWO to solve a combinatorial problem like SCOS, but also ensures greater exploration by crossover and mutation operators [27].

(*Continued*)

Table 9.1 Literature Review for Hybridization of GWO Algorithm. (*Continued*)

Paper	Methodology Used	Objective of the method	Dataset used	Findings of the Paper
[28]	Hybrid GWO-Differential Evolution	The objective is to solve complex nonlinear systems as a problem of optimization [28].	Thirteen unconstrained optimization problem and seven benchmark system of nonlinear equation are used.	Empirical results show that GWO-DE has the potential to achieve optimal consequences for full structures of nonlinear equations and optimization problems [28].
[29]	Hybrid GWO-PSO and C4.5 approach	To develop an effective Rheumatoid Arthritis prediction system [29].	Real time patient data.	The proposed model is the first prediction model that contains feature selection and classification of RA prediction [29].
[30]	Hybrid GWO-CS (Cuckoo Search algorithm)	The objective of the method to extract parameters of PV cell models in different operating conditions alongside experimental data.	Ten complex benchmark functions.	The proposed algorithm maintains a balance between exploration and exploitation. The experimental result depicts that the suggested algorithm is the finest candidate approach to extract the parameters of solar photovoltaic models.

(*Continued*)

Table 9.1 Literature Review for Hybridization of GWO Algorithm. (*Continued*)

Paper	Methodology Used	Objective of the method	Dataset used	Findings of the Paper
[31]	Hybrid GWO-WO (Whale Optimization algorithm)	Objective is to overcome WOA's drawbacks in the exploitation and solution of real-world optimization problem, such as design of pressure vessel [31].	Three different benchmark functions such as 1. 23 common functions 2. 25 CEC2005 3. 10 CEC2019 functions.	The method proposed was compared with original GWO, and based on Wilcoxon rank-sum test WOA and three widely used algorithms showed better result.
[32]	Hybrid GWO-PSO	The Objective is to design a new hybridized algorithm for feature selection.	Twenty benchmark datasets from UCI Machine Learning repository and https://www.openml.org/ search site.	The combined features chosen from the 20 datasets were 196 out of 773.The average accuracy is 90% compared to other algorithms such as GWO (81.6) and PSO (86.8). The overall processing time is less compared to GWO (272) and PSO (245.6) i.e. 184.3.
[33]	Hybrid GWO-GA	The goal of the IOTD algorithm is to reduce the size of massive wireless network traffic by choosing the most informative traffic features.	AWID real wireless intrusion datasets [33]	The experimental result show that the proposed algorithm enhances the performance of IoTIDS and enable it to detect high performance with low false alarm levels. The proposed algorithm is more effective as compared to other metaheuristics [33].

Table 9.2 Literature Review for GWO Extension Algorithms.

Paper	Methodology Used	Objective of the method	Dataset used	Findings of the Paper
[34]	Multi-objective discrete Grey Wolf Optimizer (MODGWO) [34]	To provide solution to real world welding scheduling problem [34].	A real-world scheduling problem [34].	The efficiency of the proposed algorithm is compared with well-known MOEAs i.e. NSGA-II and SPEA2, in terms of speed, coverage and convergence metrics and shows better result.
[35]	Binary Grey Wolf Optimizer	To develop a heuristic algorithm to solve binary unit commitment problem.	Test systems with dimensions starting from 10 to 100 units for 24 h.	The Binary Grey wolf optimizer is more efficient when compared with other existing algorithms in solving unit commitment problem.
[36]	Non dominated sorting Grey Wolf Optimizer (NS-GWO) [34]	The objective is to apply the proposed algorithm to a set of standard problems mentioned below 1. unconstrained 2. constrained 3. engineering design.	Nineteen standard test functions	The algorithm proposed is capable of finding pareto front of any sort. The proposed algorithm is validated by complex real world ECEDWP problem for solving nonlinear and complex problem.

(*Continued*)

Table 9.2 Literature Review for GWO Extension Algorithms. (*Continued*)

Paper	Methodology Used	Objective of the method	Dataset used	Findings of the Paper
[37]	Binary Grey Wolf Optimizer (BGWO)	The goal is to enhance binary Grey Wolf Optimizer [37] within wrapper feature selection to handle the problem of Arabic text classification.	Three public datasets in Arabic i.e. Alwatan, Akhbar Alkhaleej, Al-jazeera-News	The proposed algorithm comprises of various learning models i.e. KNN, Decision trees [37], Naive Bayes and SVM classifier, shows better performance in case of SVM feature selection technique on Arabic document classification problem.
[38]	Binary Grey Wolf Optimizer [38]	The objective of the method is to resolve multi-dimensional Knapsack problem	Two sets benchmark problem from the OR library [38].	The proposed method with transfer function (V-shaped) gives better performance in solving MKP [38].
[39]	Multi objective Grey Wolf Optimizer based on decomposition (MOGWOD) [39]	To provide solution to multi objective optimization problem.	1. Well-known benchmark problems Deb–Thiele–Laummans–Zitzler test suits (DTLZ) and UF) 2. Two real life engineering problems [39]	The proposed algorithm provides better results for the used data set [39].

(*Continued*)

Table 9.2 Literature Review for GWO Extension Algorithms. (*Continued*)

Paper	Methodology Used	Objective of the method	Dataset used	Findings of the Paper
[40]	Hierarchy Strengthened Grey Wolf Optimizer (HSGWO)	To avoid trapping at local optima while solving large scale optimization problem.	30 unconstrained complex functions taken from CEC2014 benchmark functions.	The proposed algorithm gives better performance in terms of solution quality and convergence speed [40].
[41]	Binary GWO	The major objective of the method is to solve binary optimization problem.	1. Binary suites consisting of Set Union Knapsack Problem 2. Incapacitated Facility Location Problem [41].	The GWO algorithm adapts two mechanisms such as 1. Evolutionary mechanism 2. Adaptive inheritance mechanisms to work right in the binary domain.
[42]	Improved Grey Wolf Optimization	To predict subset of features for protein structure prediction.	The dataset used for this work is available at http://bit.ly/ RMSD Classification-DS	The test analysis shows that ANN classifier predicts more precisely the protein structure [42]. The proposed method is stable enough to maximize the precision and minimize the number of optimum features [42].

(*Continued*)

Table 9.2 Literature Review for GWO Extension Algorithms. (*Continued*)

Paper	Methodology Used	Objective of the method	Dataset used	Findings of the Paper
[43]	(CEEMDAN), MOGWO-Kernel based nonlinear extension of the ARPs decline (MGWO-KNEA) [43]	To develop an algorithm to predict the Wind power (WP) of an OWFs.	Historical date on wind power taken as case study for offshore wind farms in Belgium. The raw data are referred from Elia [43]. i.e. http://www.elia.be/en	The predictive outcomes are compared with six comparison models, and the model does better [43]. The proposed model is superior in terms of stability and prediction accuracy than the six models.

9.4.3 Modification

In the complexity of real-world optimization problems, many complications may exist, GWO has been updated to comply complex domain search space. Amendments are made in the update mechanism in order to overcome the limitation of the GWO in resolving many practical problems and similarly it can be done to expand the actions of GWO. The literature review of various GWO modifications is summarized below in Table 9.3.

9.5 Application of GWO in Medical Domain

Grey Wolf Optimization algorithm has been categorized as Hybrid, Extended and Modified and in the present section a brief review on all these variants of GWO in medical domain is concentrated. Table 9.4 focuses on the method used in the paper, categorization of variant and the objective of the method. Subsequently, the researcher can have a wider view on the variant of GWO in medical domain.

9.6 Application of GWO in Microarray Data

There are several ways based on which microarray data can be analysed such as prediction, classification and clustering. GWO can be used for optimizing these techniques. Table 9.5 gives a detailed review on GWO on microarray data for several analysis.

9.7 Conclusion and Future Work

In this study the GWO algorithm for solving optimization problem in different domain as well as how effectively it can be used to analyse the microarray data is presented. The analysis includes the key motivation and necessity of GWO algorithm in medical application and microarray dataset, structure of GWO, mathematical model, the exploration and exploitation features of GWO and the variants of GWO are discussed. From the review of GWO variants in the basis of hybridization, extension, modification it is clear that this algorithm can be used in many different applications by combining, extending and modifying GWO to solve different kinds of optimization problems. A study on GWO algorithm in medical application and Microarray data set was also presented in this paper. From

Table 9.3 Literature Review for GWO Modification Algorithm.

Paper	Methodology used	Objective of the Paper	Dataset used	Findings of the Paper
[44]	Grey Wolf Optimizer with a fuzzy hierarchical operator (GWOFHO) [44]	The proposed operator applies to hunting process simulation in GWO [44]	Twenty-three benchmark functions.	During the simulation of hunting process, the proposed operator is employed and has five different variants. The performance of Hierarchical GWO is better when fuzzy logic is used [44].
[45]	Experienced Grey Wolf Optimization	The aim is to improve GWO performance.	Twenty-one data sets from UCI machine learning repository	The performance is accessed using feature selection problem and neural network weight adaption. Results is accessed in both applications using various performance indicators.
[46]	Levy Embedded GWO(LGWO)	A modified version of Grey Wolf Optimization algorithm is developed for solving real world and global optimization problems.	Thirty artificial and fourteen real world problems from CEC 2014 and CEC 2011 to evaluate proposed algorithm.	To measure the performance of LGWO, the result is equated with other state-of-art optimization algorithms and gives better result. To measure the effectiveness of LGWO, 30 benchmark functions are used, and the result shows the proposed algorithm is highly effective.

(*Continued*)

Table 9.3 Literature Review for GWO Modification Algorithm. (*Continued*)

Paper	Methodology used	Objective of the Paper	Dataset used	Findings of the Paper
[47]	Modified Grey Wolf Optimizer	The objective is to apply MGWO algorithm for Automatic Generation Control of power system in presence of plug in electric vehicle.	Standard benchmark function.	The AGC of single power systems are considered and MGWO proves superior than the other approaches are demonstrated.
[48]	COBGWO-DEDO) [48]	In this paper a variant of GWO is proposed to enhance the exploration and exploitation capability to solve global optimization problem.	Two benchmark function i.e. twenty-three unconstrained function from classical benchmark function for CEC 2005 and thirty unconstrained function for CEC 2014	The performance of the proposed algorithm is evaluated by applying it to the classification of galaxy images where the algorithm acts as a feature selection method to find out the subset of features of galaxy images. The result shows that the algorithm is more efficient in solving global optimization problem with high accuracy.

(*Continued*)

Table 9.3 Literature Review for GWO Modification Algorithm. (*Continued*)

Paper	Methodology used	Objective of the Paper	Dataset used	Findings of the Paper
[49]	(COGWO2D)	The goal is to use the oppositional GWO algorithm to derive the optimum economic load dispatch problem operating strategy [49]	Small scale Medium scale Large scale test systems (solving ELD problems of 13, 40, 140, 160 units).	The proposed algorithm is used to solve nonlinear ELD problems that have a higher integration rate and are efficient in producing possible solutions.
[50]	Oppositional Grey Wolf Optimizer (OGWO)	To cluster Large scale datasets using the proposed method.	Seven UCI benchmark datasets	The proposed algorithm is compared with specific metaheuristics and other clustering methods to calculate the output. The tentative result shows that the proposed variant is an effective method to cluster large scale datasets [50].
[51]	Map Reduce Enhanced Grey Wolf Optimizer (MR-EGWO)	The objective is to improve the search ability of GWO [51]	Thirty standard IEEE CEC 2014 benchmark problems	The proposed algorithm gives better leadership in searching for a prey and is highly reliable to solve both continuous and real-world problems.

(*Continued*)

Table 9.3 Literature Review for GWO Modification Algorithm. (*Continued*)

Paper	Methodology used	Objective of the Paper	Dataset used	Findings of the Paper
[52]	Random Walk Grey Wolf Optimization	The objective is to improve the efficiency of GWO by effective exploration and exploitation mechanisms.	Complete set of novel and state of art optimizer on Twenty-three test and 30 CEC problems [52].	The algorithm proposed for KELM is applied to solve real world optimization problems [52]. The results illustrate that OBLGWO is better performer than GWO and other existing variants in terms of result quality and convergence speed.
[53]	Opposition Based Learning Grey Wolf Optimizer	To use in visual object tracking under complex environments [53].	Visual tracking database [53].	The MGWOPF is equated with other algorithms but the result shows this proposed algorithm gives reliable and efficient tracking result.
[54]	Modified Grey Wolf Optimizer based Particle Filter (MGWOPF)[54]	The objective is to use a mechanism based on β-chaotic sequence for the improvement of GWO algorithm.	Two benchmark suites i.e. twelve shifted and biased function, twenty-nine CEC 2017 functions	The efficiency of the proposed algorithm is evaluated by comparing it with other models and addressing two real-life engineering issues. The experimental results indicate that the approach proposed is highly efficient and has greater potential for exploration and exploitation.

(*Continued*)

Table 9.3 Literature Review for GWO Modification Algorithm. (*Continued*)

Paper	Methodology used	Objective of the Paper	Dataset used	Findings of the Paper
[55]	β-GWO	The objective is to overcome premature convergence in GWO using the proposed algorithm.	CEC 2014 functions.	The algorithm efficiency is measured by comparing the proposed algorithm with other GWO variants. A multi UAV tracking problem is also solved to measure the performance.
[56]	Gaussian Estimation of Distribution Grey Wolf Optimization. (GEDGWO)	The objective is to analyse the earthquake catalogues in an ergodic framework using Quantum GWO.	Twenty-Four benchmark functions	The proposed algorithm possesses better convergence rate than other algorithms. The performance is also measured in comparison with other variants with seven benchmark functions and the super performance is measured on comparing the algorithm with a PSO based model.
[57]	Quantum Grey Wolf Optimizer (QGWO)	To solve Harmonic estimation problem	CEC-2017 benchmark functions.	The proposed algorithm improves GWO's efficiency based on solution quality. The proposed algorithm helps GWO to achieve better accuracy.
[58]	Evolutionary operators equipped GWO (EOGWO)	Use opposition based learning to boost GWO's exploratory behavior	Twenty-Three benchmark Functions	The proposed approach allows Grey Wolf Optimization to give better performance as compared to other algorithms.

Table 9.4 Literature Review for GWO in Medical Applications.

Paper	Hybrid	Extend	Modified	Method objective	Medical dataset used	Findings
[59]	×	×	✓	GWO is used to optimize the created Fuzzy rules	PIMA data set from UCI machine learning repository	A diabetes prediction model is created based on the GWO optimization with fuzzy logic. The model learns fuzzy rules and optimizes them according to GWO algorithm.
[60]	×	×	✓	The Objective is to Diagnosis Heart Disease.	Cleveland Heart Database from UCI repository	The experimental analysis shows that the proposed algorithm possesses better classification accuracy than other algorithms.
[61]	×	×	✓	The Objective is to classify the disease using different symptoms	1. Five Bio Medical data set 2. sine dataset	The result shows that the proposed GWO and SCA algorithm is highly effective in classifying disease by symptoms using optimization and real-life application.
[62]	×	✓	×	FMI-Binary GWO is used for feature selection	Leukemia and Breast cancer data from UCI repository	The proposed method is compared with BGWO and has a better classification accuracy with the selected features in comparison to BGWO
[63]	×	×	✓	The Objective is to predict blood Glucose	The data sets are available in Diabetes Research in Children Network	The proposed GWO-KELM possess high prediction accuracy.

(*Continued*)

Table 9.4 Literature Review for GWO in Medical Applications. (*Continued*)

Paper	Hybrid	Extend	Modified	Method objective	Medical dataset used	Findings
[64]	×	×	✓	To develop a process which will help in identifying Alzheimer Disease	Real time data sets from Chettinad health city.	The proposed GWO algorithm with Deep Learning classifier provides better accuracy compared to other techniques to assess output.
[65]	×	✓	×	The goal of the method is to select the optimal feature subset	Eighteen standard benchmark data set from UCI repository	The proposed Binary Grey wolf Optimization and Particle Swarm Optimization (BGWOPSO) algorithm gives high accuracy, selecting best subset of features as compared to other techniques.
[66]	×	×	✓	The objective is to identify paraquat poisoned patients.	PQ-poisoned Patients data [66].	To measure the performance four measures are used and the proposed fusion GC-MS, EGWO, ELM algorithm is compared to other methods and results better than others [66].
[67]	×	×	✓	Objective of CAIMGWO-NB is used for heart disease diagnosis	Cleveland Heart Database from UCI machine learning repository	CAIMGWO-NB shows better classification accuracy in comparison to the traditional NB classifier.

Table 9.5 Literature Review for GWO in Medical Application for Microarray Dataset.

Paper	Methodology Used	Objective	Dataset Used	Data Analysis Type	Findings of the Paper
[68]	GWO and Decision tree classifier	The objective is to identify cancer by selecting small number of relevant genes out of many.	Ten gene expression Cancer datasets	classification	The method proposed is contrasted with other methods, and the result shows that the method proposed provides better accuracy then other methods.
[69]	Multi objective GWO and Non dominated sorting based GWO [69] (MOGWO-NSGWOO)	The basic objective is to find out optimal subset features in order to classify the malignant and benign lesions present in CECT images on the cervix region [69]	UCI Machine learning repository [69] and online available microarray dataset.	classification	The proposed algorithm shows improved efficiency in comparison to others. The future work of this paper is to find the most suitable meta heuristic for optimal feature selection.

(*Continued*)

Table 9.5 Literature Review for GWO in Medical Application for Microarray Dataset. (*Continued*)

Paper	Methodology Used	Objective	Dataset Used	Data Analysis Type	Findings of the Paper
[70]	Antlion Optimization-Grey Wolf Optimization	The main objective is to select significant features from data sets	1. Seven Microarray gene expression data 2. Five image(face) detection data [70]	classification	The proposed algorithm shows a good stability between exploration and exploitation [70]. The algorithm is capable of generating diverse solutions and preventing premature convergence [70].
[71]	(ALO-GWO)	The objective is to classify Cancer.	Five publicly available benchmark microarray data set are used.	Classification	The test result shoes that the scoring method used in this paper for the proposed algorithm gives high classification accuracy of 100% for both scoring methods when compared with other techniques.

(*Continued*)

Table 9.5 Literature Review for GWO in Medical Application for Microarray Dataset. (*Continued*)

Paper	Methodology Used	Objective	Dataset Used	Data Analysis Type	Findings of the Paper
[72]	Intelligent Dynamic Grey Wolf Optimizer	The objective of the method is to categorize the data into groups for analysing the gene expression effectively.	Microarray data	Clustering	The proposed algorithm gives better precision rate, recall rate, f-measure rate and minimal time consumption as compared to other standard techniques.
[73]	Generalized hierarchical fuzzy C means and Grey Wolf Optimizer [73]	The major objective is clustering of Microarray data to reduce the variability within the cluster and optimize cluster variability between the cluster [73].	Four Artificial data sets and three real life gene expression data sets	Clustering	The proposed method shows better efficiency as compared to other techniques [73].

the survey it is clear that out of so many variants, many works have already been done in hybridization and modifications of GWO algorithm but there is still work to be done on extensions of GWO to solve multi objective optimization problems. From the study it has been shown that mostly the benchmark functions like unimodal, multimodal, fixed dimension multimodal and composite functions are used to measure the performance of the GWO algorithm to solve real world optimization problem. The UCI Machine learning bench mark functions are mostly used. GWO has been applied in very few works for Microarray data set. It should be noted that from the GWO is able to demonstrate high performance not only in solving unconstrained problems but also in solving constrained optimization problems.

More different variants of the algorithm can be developed by combining with other algorithms, the algorithms exploration and exploitation capability can be further strengthened. There is currently no work has been done using GWO for dynamic optimization problems. Very less work has been done in GWO using microarray dataset in order to solve the optimization problem in the field of medical application.

References

1. Almugren, N. and Alshamlan, H., A Survey on Hybrid Feature Selection Methods in Microarray Gene Expression Data for Cancer Classification. *IEEE Access*, 7, 78533–7854, 2019.
2. Zheng, X., Zhub, W., Tang, C., Wang, M., Gene selection for microarray data classification *via* adaptive hypergraph embedded dictionary learning. *Gene*, 706, 188–200, 2019.
3. Hammer, B. and Gersmann, K., A note on the universal approximation capability of support vector machines. *Neural Process. Lett.*, 17, 1, 43–53, 2003.
4. Bir Jmel, A., Douiri, S.M., Elbernoussi, S., Gene Selection *via a* New Hybrid Ant Colony Optimization Algorithm for Cancer Classification in High-Dimensional Data. *Comput. Math. Methods Med.*, 2019, 1–20, 2019.
5. Shetye, A. and Tike, R., A Survey on Applications and Optimizations of Genetic Algorithm. *IJERT,* 6, 3, 2017.
6. Xu, Y., Fan, P., Yuan, L., A Simple and Efficient Artificial Bee Colony Algorithm. *Math. Prob. Eng.*, 2013, 1–9, 2013.
7. Zhang, Y., Wang, S., Ji, G., A Comprehensive Survey on Particle Swarm Optimization Algorithm and its Application. *Math. Prob. Eng.*, 2015, 1–38, 2015.
8. Gao, X.Z., Govindasamy, V., Xu, H., Wang, X., Zenger, K., Harmony Search Method: Theory and Applications. *Comput. Intell. Neurosci.*, 2015, 1–10, 2015.

9. Yang, X.S., Firefly Algorithms for Multimodal Optimization. *International Symposium on Stochastic Algorithm*, Springer, Berlin, Hedinberg, pp. 169–178, 2009.
10. Gandomi, A.H. and Yang, X.-S., Chaotic bat algorithm. *J. Comput. Sci.*, 5, 2, 224–232, 2014.
11. Faris, H., Aljarah, I., Al-Betar, M.A., Mirjalili, S., Grey wolf optimizer: a review of recent variants and applications. *Neural Comput. Appl.*, 30, 2, 413–435, 2018.
12. Darwish, A., Bio-inspired computing: Algorithms review, deep analysis, and the scope of applications. *Future Comput. Inf. J.*, 3, 2, 231–246, 2018.
13. Mukhopadhyay, A., Identifying Non-redundant Gene Markers from Microarray Data: A Multiobjective Variable Length PSO-based Approach. *IEEE/ACM Trans. Comput. Biol. Bioinf.*, 11, 6, 1170–1183, 2014.
14. Tarca, A.L., Romero, R., Draghici, S., Analysis of microarray experiments of gene expression profiling. *Am. J. Obstet. Gynocol.*, 195, 2, 373–388, 2006.
15. Darwish, A., Bio-inspired computing: Algorithms review, deep analysis, and the scope of applications. *Future Comput. Inf. J.*, 3, 2, 231–246, 2018.
16. Thanoon, R.B. and Mitras, B.A., Modified Grey Wolf Optimization Algorithm by using Classical Optimization Methods. *IJCNCS,* 7, 3, 49–60, 2019.
17. Noori, N.M. and Qasim, O.S., Improving cancer diseases classification using a hybrid filter and wrapper feature subset selection. *Ann. Proteom. Bioinform.,* 4, 006–0011, 2020.
18. Ting, Yang, X.-S., Cheng, S., Huang, K., Hybrid Metaheuristic Algorithms: Past, Present, and Future, in: *Recent Advances in Swarm Intelligence and Evolutionary Computation*, pp. 71–83, Springer, Cham, 2015.
19. Chopra, N., Kumar, G., Mehta, S., Hybrid GWO-PSO Algorithm for Solving Convex Economic Load Dispatch Problem. *Int. J. Res. Advent Technol.*, 4, 6, 37–41, 2016.
20. Singh, N. and Singh, S.B., A novel hybrid GWO-SCA approach for optimization problems. *Eng. Sci. Technol., An International Journal*, 20, 6, 1586–1601, 2017.
21. Pan, J.-S., Dao, T.-K., Chu, S.-C., Nguyen, T.-T., A Novel Hybrid GWO-FPA Algorithm for Optimization Applications. *International Conference on Smart Vehicular Technology, Transportation, Communication, and Application*, Springer, pp. 274–281, 2017.
22. Gaidhane, P.J. and Nigam, M.J., A Hybrid Grey Wolf Optimizer and Artificial Bee Colony Algorithm for Enhancing the Performance of Complex Systems. *J. Comput. Sci.*, 27, 84–302, 2018.
23. El Gayyar, M., Emary, E., Sweilam, N.H., Abdelazeem, M., A Hybrid Grey Wolf–Bat Algorithm for Global Optimization. *International Conference on Advance Machine Learning Technology and Applications*, Springer, Cham, pp. 3–12, 2018.
24. Kamboj, V.K., Bhadoria, A., Gupta, N., A Novel Hybrid GWOPS Algorithm for Standard Benchmark Optimization Problems. *INAE Lett.*, 3, 4, 217–241, 2018.

25. Arora, S., Singh, H., Sharma, M., Sharma, S., Anand, P., A New Hybrid Algorithm based on Grey Wolf Optimization and Crow Search Algorithm for unconstrained function optimization and feature selection. *IEEE Access*, 7, 26343–26361, 2019.
26. Zhang, Y., Jin, Z., & Chen, Y. Hybridizing grey wolf optimization with neural network algorithm for global numerical optimization problems. *Neural. Comput. Appl.,* 32, 14, 10451–10470, 2020.
27. Bouzary, H., Chen, F.F., Shahin, M., Optimal Composition of Task in cloud manufacturing platform: A novel Hybrid GWO-GA approach. *Procedia Manuf.*, 34, 961–968, 2019.
28. Tawhid, M.A. and Ibrahim, A.M., A hybridization of grey wolf optimizer and differential evolution for solving nonlinear systems. *Evol. Syst.*, 11, 1, 65–87, 2020.
29. Sundaramurthy, S. and Jayavel, P., A hybrid Grey Wolf Optimization and Particle Swarm Optimization with C4.5 approach for prediction of Rheumatoid Arthritis. *Appl. Soft Comput. J.*, 94, 106500, 2020.
30. Long, W., Caia, S., Jiao, J., Xu, M., Wu, T., A new hybrid algorithm based on grey wolf optimizer and cuckoo search for parameter extraction of solar photovoltaic models. *Energy Convers. Manage.*, 203, 112243, 2020.
31. Mohammed, H. and Rashid, T., A novel hybrid GWO with WOA for global numerical optimization and solving pressure vessel design. *Neural Comput. Appl.*, 1–18, 2020.
32. El-Hasnony, I.M., Barakat, S.I., Elhoseny, M., Mostafa, R.R., Improved Feature Selection Model for Big Data Analytics. *IEEE Access*, 8, 66989–67004, 2020.
33. Davahli, A., Shamsi, M., Abaei, G., Hybridizing genetic algorithm and grey wolf optimizer to advance an intelligent and lightweight intrusion detection system for IoT wireless networks. *J. Ambient Intell. Hum. Comput.*, 2020.
34. Lu, C., Xiao, S., Li, X., Gao, L., An effective multi-objective discrete grey wolf optimizer for a real-world scheduling problem in welding production. *Adv. Eng. Software*, 99, 161–176, 2016.
35. Panwar, L.K., Srikanth, R.K., Verma, A., Panigrahi, B.K., Kumar, R., Binary Grey Wolf Optimizer for large scale unit commitment problem. *Swarm Evol. Comput.*, 38, 251–66, 2018.
36. Jangir, P. and Jangir, N., A new Non-Dominated Sorting Grey Wolf Optimizer (NS-GWO) algorithm: Development and application to solve engineering designs and economic constrained emission dispatch problem with integration of wind power. *Eng. Appl. Artif. Intell.*, 72, 449–467, 2018.
37. Chantar, H., Mafarja, M., Alsawalqah, H., Heidari, A.A., Aljarah, I., Faris, H., Feature selection using binary grey wolf optimizer with elite-based crossover for Arabic text classification. *Neural Comput. Appl.*, 32, 16, 12201–12220, 2020.
38. Luo, K. and Zhao, Q., A binary grey wolf optimizer for the multidimensional knapsack problem. *Appl. Soft Comput. J.*, 83, 105645, 2019.

39. Zapotecas-Martínez, S., Garcia-Najera, A., Lopez-Jaimes, A., Multi-objective grey wolf optimizer based on decomposition. *Expert Syst. Appl.*, 120, 357–371, 2019.
40. Tu, Q., Chen, X., Liu, X., Hierarchy Strengthened Grey Wolf Optimizer for Numerical Optimization and Feature Selection. *IEEE Access*, 7, 78012–78028, 2019.
41. Gölcük, İ. and Ozsoydan, F.B., Evolutionary and adaptive inheritance enhanced Grey Wolf Optimization algorithm for binary domains. *Knowl.-Based Syst.*, 194, 105586, 2020.
42. Sharma, P., Gupta, A., Aggarwal, A., Gupta, D., Khanna, A., Hassanien, A.E., de Albuquerque, V.H.C., The health of things for classification of protein structure using improved grey wolf optimization. *J. Super Comput.*, 76, 2, 1226–1241, 2020.
43. Lu, H., Ma, X., Huang, K., Azimi, M., Prediction of offshore wind farm power using a novel two-stage model combining kernel-based nonlinear extension of the Arps decline model with a multi-objective grey wolf optimizer. *Renewable Sustainable Energy Rev.*, 127, 109856, 2020.
44. Rodriguez, L., Castillo, O., Soria, J., Melin, P., Valdez, F., Gonzalez, C.I., Martinez, G.E., Soto, J., Fuzzy Hierarchical Operator in the Grey Wolf Optimizer Algorithm. *Appl. Soft Comput.*, vol 57, 315–328, 2017.
45. Emary, E., Zawbaa, H.M., Grosan, C., Experienced Gray Wolf Optimization Through Reinforcement Learning and Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.*, 29, 3, 681–694, 2017.
46. Heidari, A.A. and Pahlavani, P., An Efficient Modified GreyWolf Optimizer with Levy Flight for Optimization Tasks. *Appl. Soft Comput. J.*, 60, 115–134, 2017.
47. Padhy, S., Panda, S., Mahapatra, S., A modified GWO technique based cascade PI-PD controller for AGC of power systems in presence of Plug in Electric Vehicles. *Eng. Sci. Technol.—Int. J.*, 20, 2, 427–442, 2017.
48. Ibrahim, R.A., Elaziz, M.A., Lu, S., Chaotic Opposition-Based Grey-Wolf Optimization Algorithm based on Differential Evolution and Disruption Operator for Global Optimization. *Expert Syst. Appl.*, 108, 1–27, 2018.
49. Pradhan, M., Roy, P.K., Pal, T., Oppositional based grey wolf optimization algorithm for economic dispatch problem of power system. *Ain Shams Eng. J.*, 9, 4, 2015–2025, 2018.
50. Tripathi, A.K., Sharma, K., Bala, M., A Novel Clustering Method Using Enhanced Grey Wolf Optimizer and MapReduce. *Big Data Res.*, 14, 93–100, 2018.
51. Gupta, S. and Deep, K., A novel Random Walk Grey Wolf Optimizer. *Swarm Evol. Comput.*, 44, 101–112, 2019.
52. Heidari, A.A., Abbaspour, R.A., Chen, H., Efficient boosted grey wolf optimizers for global search and kernelextreme learning machine training. *Appl. Soft Comput. J.*, 81, 105521, 2019.

53. Narayana, M., Nenavath, H., Chavan, S., Koteswara Rao, L., Intelligent visual object tracking with particle filter based on Modified Grey Wolf Optimizer. *Optik-International Journal for Light and Electron Optics*, 193, 162913, 2019.
54. Saxena, A., Kumar, R., Das, S., β-Chaotic map Enabled Grey Wolf Optimizer. *Appl. Soft Comput.*, 75, 84–105, 2019.
55. Wang, X., Zhao, H., Han, T., Zhou, H., Li, C., A grey wolf optimizer using Gaussian estimation of distribution and its application in the multi-UAV multi-target urban tracking problem. *Appl. Soft Comput. J.*, 78, 240–260, 2019.
56. Vijay, R.K. and Nanda, S.J., A Quantum Grey Wolf Optimizer based declustering model for analysis of earthquake catalogs in an ergodic framework. *J. Comput. Sci.*, 36, 101019, 2019.
57. Saxena, A., Kumar, R., Mirjalili, S., A Harmonic Estimator Design with Evolutionary Operators Equipped Grey Wolf Optimizer. *Expert Syst. Appl.*, 145, 113125, 2020.
58. Dhargupta, S., Ghosh, M., Mirjalili, S., Sarkar, R., Selective Opposition based Grey Wolf Optimization. *Expert Syst. Appl.*, 151, 113389, 2020.
59. Siva Shankar, G. and Manikandan, K., Diagnosis of diabetes diseases using optimized fuzzy rule set by grey wolf optimization. *Pattern Recognit. Lett.*, 125, 432–438, 2019.
60. Babu, S.B., Suneetha, A., Charles Babu, G., Jeevan Nagendra Kumar, Y., Karuna, G., Medical Disease Prediction using Grey Wolf optimization and Auto Encoder based Recurrent Neural Network. *Period. Eng. Nat. Sci.*, 6, 1, 229–240, 2018.
61. Swathi, K., Nithya, P.R., Priyadharshini, S., Danapaquiame, N., RamaKrishna, V., Janardhan, N., Detection of Disease in Grey Wolf Optimization using Sine Cosine Algorithm and Medical dataset. *Int. J. Pure Appl. Math.*, 118, 14, 403–409, 2018.
62. Noori, N.M. and Qasim, O.S., Improving cancer diseases classifi cation using a hybrid fi lter and wrapper feature subset selection. *Ann. Proteom. Bioinf.*, 4, 006–011, 2020.
63. Chen, X., Tuo, J., Wang, Y., A Prediction Method for Blood Glucose Based on Grey Wolf Optimization Evolving Kernel Extreme Learning Machine. *Chinese Control Conference*, IEEE, pp. 3000–3005, 2019.
64. Chitradevi, D. and Prabha, S., Analysis of brain sub regions using optimization techniques and deep learning method in Alzheimer disease. *Appl. Soft Comput. J.*, 86, 105857, 2020.
65. al Tashi, Q., Abdul Kadir, S.J., Rais, H. M., Mirjalili, S., Alhussian, H., Binary Optimization Using Hybrid Grey Wolf Optimization for Feature Selection. *IEEE Access*, 7, 39496–39508, 2019.
66. Zhao, X., Zhang, X., Cai, Z., Tian, X., Wang, X., Huang, Y., Chen, H., Hu, L., Chaos enhanced Grey Wolf Optimization wrapped ELM for diagnosis of paraquat-poisoned patients. *Comput. Biol. Chem.*, 78, 481–490, 2019.

67. El Bakrawy, L.M., Grey Wolf Optimization and Naive Bayes classifier Incorporation for Heart Disease Diagnosis. *Aust. J. Basic Appl. Sci.*, 11, 7, 64–70, 2017.
68. Vosooghifard, M. and Ebrahimpour, H., Applying Grey Wolf Optimizer-based decision tree classifer for cancer classification on gene expression data, in: *2015 5th International Conference on Computer and Knowledge Engineering (ICCKE)*, IEEE, pp. 147–151, 2015.
69. Sahoo, A. and Chandra, S., Multi-Objective Grey Wolf Optimizer for Improved Cervix Lesion Classification. *Appl. Soft Comput.*, 52, 64–80, 2017.
70. Zawbaa, H.M., Emary, E., Grosan, C., Snasel, V., Large-dimensionality small instance set feature selection: A hybrid bio-inspired heuristic approach. *Swarm Evol. Comput.*, 42, 29–42, 2018.
71. Uthayan, K.R., A Novel Microarray Gene Selection and classification using Intelligent dynamic Grey Wolf Optimization. *Genetika*, 52, 3, 805–828, 2019.
72. Edwin Dhas, P. and Sankara Gomathi, B., A novel clustering algorithm by clubbing GHFCM and GWO for microarray gene data. *J. Supercomput.*, 76, 1–15, 2019.
73. Achom, A., Das, R., Pakray, P., Saha, S., Classification of Microarray Gene Expression Data using Weighted Grey Wolf Optimizer based Fuzzy Clustering, in: *TENCON 2019-2019, IEEE Region 10 Conference (TENCON)*, pp. 2705–2710, 2019.

10

The Cluster Analysis and Feature Selection: Perspective of Machine Learning and Image Processing

Aradhana Behura

Veer Surendra Sai University of Technology, Burla, Sambalpur, India

Abstract

Medical Imaging is the process of producing visible images of inner structures of the body for scientific and medicinal study and treatment as well as visible view of the function of interior tissues. These medical images play an important role for treatment of various pathological disorders. Medical imaging is developing rapidly due to developments in image processing techniques including image recognition, analysis, enhancement, segmentation, etc. It increases the percentage and amount of detected tissues detected through imaging. A number of methods are studied in the literature using different methodologies. In literature reviews, the authors classified the retinal segmentation techniques into two broad categories. These are rule based methods and machine learning based methods. Rule based methods generally deals with specific rules regarding the framework of algorithm such as kernel based, vessel tracking, adaptive local thresholding, etc. Machine Learning method uses a manually segmented image (ground truth) to form a labelled dataset which is further used as training dataset. In training images pixels are identified by either they belong to the vessels or not. The machine learning based methods use supervised learning approach (requires training images) and unsupervised approaches (does not require training images). However, noise occurs in the process of exaggeration of the retinal images due to the lower intensity difference between the background and the vessel surface. Further, it is challenging to preserve the minute vessels, while segmenting. Usually, the contrast enhancement techniques enrich the information content as well as the noise present in its background. Therefore, selection of a suitable enhancement technique for retinal vessel segmentation plays a vital role.

Email: aradhanabehura@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (249–280) © 2021 Scrivener Publishing LLC

Machine learning is an important subset of artificial intelligence which helps user to predict the problem and it's solution in an informatics way which consumes less time as well as cost. There are many clustering algorithms are used to predict the disease from the data set by taking the similarity instances among the data set. The well known clustering algorithms are K-means, C-means, DBSCAN which is a density based clustering algorithm used to remove noise from the data set. The well known neural network SOM (self organizing feature map) is used for clustering as well as pattern recognition purposes. This neural network not only an information visualization procedure but also helps to reduce dimension to observe pattern of the feature from the data set. The C-means is an iterative cluster analysis procedure, in which objectives to minimize the dissimilarity among the data points. In this algorithm, first decide the number of clusters then the level of fuzziness and initialize randomly the membership matrix. After that, calculate the dimension of the cluster center and evaluate the Euclidean distance among the cluster center and the data points then update the membership matrix which is described in Figure 10.8. The Health division can be modernized utilizing the most recent innovations which will build the future of the general populace. Driving ailments like Heart Disease, cyst, Cancer comprises passings around the world. Passing rate including Cardiovascular Disease is expanding at a disturbing rate each year. World Health Organization report for the year 2016 recommends that, 31% of the passings overall included cardiovascular sickness in which 85% were because of coronary failure and stroke. The expanding prominence of liquor, tobacco in the creating and created nations legitimately adds to the danger of heart-related illness. Increasing weight rates in created nations like United States, England, Canada, and New Zealand prompt the rising danger of heart-related issues. Thinking about the effect of cardiovascular sicknesses on total populace, AI model for early location turns out to be very helpful. Steady endeavors utilizing diverse mechanical progressions are made to manage this rising hazardous issue. Diverse bioengineering procedures are created lately to adapt up to the ever-developing medical issues. The proceeded with research in the region is demonstrating an aid to the improve decrease rate. Because of mechanical progressions, information assortment and capacity from different therapeutic establishments and medical clinics around the world has gotten conceivable. The information gathered is then broke down utilizing diverse AI calculations that find certain examples, relationships, and likeness among each characteristic in the dataset. The initial step is to utilize various information representation techniques to speak to the content based information into a visual organization for distinguishing undetected patterns. One second from now step is to utilize include choice to lessen excess and trifling information which improves the forecast rate essentially. The third step is to utilize order strategies to prepare the model and anticipate on the testing dataset. The fourth step is to propose a strategy for boosting the forecast rate for procedure.

Keywords: Lung cancer, SVM, IPMN, clustering, pancreatic cancer, particle swarm optimization (PSO)

10.1 Introduction

Multiple investigation techniques and algorithms have worked on breast cancer classification according to their different perspectives. In medical field, AI, DL (Deep Learning) and Machine Learning (ML) show a vital part in distinguishing the malignant and benign breast tissue with high accuracy and efficiency [29, 30, 40]. There are dissimilar type of ways belongs to Artificial Intelligence i.e. (Supervised, Unsupervised and Reinforcement)

- Supervised Learning
  In supervised learning beginning with the medical datasets, it can identify themselves through the connected level. This well-known algorithm contains Gradient Boosting, Classification, Regression and Prediction. The objective of this Machine Learning is properly detecting the new testing dataset, concluded the supervised process and considering the preceding trained dataset, lastly this procedure can study the method to classify the dataset. The supervised learning algorithm takes the responses that the result is previously known. The machine is then modeled subsequently. With the help of such type procedures, a machine creates a procedure of supervised learning method to attempt and ensure the forecast model of label standards on unlabeled data by the exploitation applicable outlines. Supervised learning application catches in such zones where the long span proceedings are estimated from end to end the historic data. Supervised learning problems can be auxiliary clustered into 'Regression' and 'Classification' problems. The Regression problem, output variable is a continuous value or real value. A Classification problem, output variable is a category like filtering emails "spam" or "not spam".
- Unsupervised Learning
  In unsupervised learning studies however, systems will learn to represent specific input patterns in a manner that reflects the applied math structure of the assortment of input patterns. By contrast with supervised learning or reinforcement learning, there aren't any express target outputs or environmental evaluations related to every input; rather the unattended learner brings in contact previous biases on what aspects of the structure of the input ought to be captured

within the output. A specific output is not having by unsupervised learning. Finding the structures and patterns in the data is aimed by the learning agent.

- Semi-Supervised Learning
  In this type of learning procedure, the machines are skilled to learn each label and untagged data for the training tenacity. This is mainly involves in training the model through a low magnitude of labeled information at the outlay of an oversize amount of untagged information. This kind of machine learning is working often times with the procedures such as regression, prediction, and classification.
- Reinforcement Learning (RL)
  This type algorithms run through the trial and error approach that offers the simplest results with a positive action and its application found in the field of play, (AI) artificial intelligence and navigation. It regularly used for AI, Gaming, task scheduling and robot navigation. There are three fundamentals that principally employ this ML sort—agent, learner and atmosphere with the help of the agent to do the interaction. It is based on fault learning pattern where a computational agent mug up to achieve suitable accomplishment by getting evaluative response. The learning agent reinforces itself from failures which ultimately lead to success. Q learning, temporal difference is an example of reinforcement learning. In this technique, each node is evaluated and a credit is assigned to the path or node. In structural terms, it is necessary to determine which part will be assigned how much of credit. This leads to overall performance enhancement. RL can be basically categorized into two types.
- Delayed reinforcement learning
- Immediate reinforcement learning
  Delayed reinforcement learning deals with a temporal sequence of state vectors used to optimize the function. In this case, the learning does not wait for delays between action and resultant reinforcement. Immediate reinforcement is purely dependent on input & output sequence which is purely dependent on the result of the action. The complete objective of the RL is to arrange the agent choice, which will

easy to take full advantage of reward over the preferred total time with simplest policy and best results.

- Collaborative Learning
  Collaborative learning is to the first step to Clustering Technique. Surrounded by the huge amount of varieties and founded on assessment of predilections, it assists the manipulators to catch item of significance. Due to no specific structural assortments of data previously, collaborative filtering techniques are an unsupervised learning. From end to end testing dissimilar assets of the datasets obviously happen in information is learned by the clustering algorithm. The clustering is splitting technique of considering huge quantity of dataset into reduced groups and analysis of tuning for each one, which fit in for the exploratory analysis. Each group member of a group obeys the property of individual one.
- Clustering
  It is an unsupervised learning method that doesn't have any proper information about the target value. The main objective of the clustering technique is to establish relations between different data and distribute the data into their corresponding cluster and help to find out the hidden structure.
- Classifications
  Classification technique comes under supervised learning which involves training of dataset that has well-known labels. Applications involved in data classification are trained by a set of non-spam or spam massages model will ultimately acquire to notice undesirable emails. From end to end the training of foregoing accounts, the system will aware to classify the upcoming risk possibility.

An important issue of AI techniques are the high dimensionality of the medical dataset [10, 31, 39]. The inspection requires a lot of memory and prompts an overfitting [11–15]. Dimensionality reduction in the dataset, improve the information [16]. Principal component analysis (Pca) makes a segment that stocks the significant data of the highlights by catching a huge fluctuation [21]. As of late, a few examinations have utilized PCA as a feature extraction method which provides better results.

10.2 Various Image Segmentation Techniques

Image segmentation is the process of partitioning an image into different groups or portions and it helps to classify the images by analyzing its pixel correctly [35]. There are various methods used in image processing for segmentation. Some of them are:

1. Clustering
2. Thresholding
3. Edge based segmentation
4. Region based segmentation
5. Watershed.

10.2.1 Clustering

Image clustering helps to implement the efficient retrieval of algorithms and to create a user-friendly interface in the database. Image clustering mainly involves these data:

- Image features—define how we are representing the image.
- Organization of the featured data—how we are organizing the data.
- Classifier—how the image is classify according to the cluster.

10.2.2 Thresholding

Threshold is another simple method of segmentation. In this method the threshold value is calculated by converting image to binary image. Thresholding methods are of two types, global and local thresholding, which have methods like Otsu global thresholding and Adaptive local thresholding [20].

Abudullakasim *et al.* [14] introduced a procedure for measurement of severity degree of brown spot in cassava leaves. He introduced the Otsus' method for this purpose. He used image enhancement techniques for image analysis and for segmenting the image used in the Otsus' method.

10.2.3 Edge-Based Segmentation

Edge detection is a method in which we detect the edge of an image. Understanding image features detection of detection of edge of an image is a very important stage. Edge of an image contains some meaningful

features and information regarding the image [20]. Edge detection technique helps to reduce the size of the image and it also delete the unnecessary information regarding the image, only required image size is saved. For image segmentation purpose also edge detection is used.

Majorly there are two methods to detect the edge of an image:

Gradient: gradient method used to detect the edge of the image by looking the maximum and minimum value. Methods used are Prewitt, Roberts and Sobel operators to detect vertical and horizontal edges. Sharp edges are defined using thresholding method.

Laplacian: The Laplacian method is based on the zero crossings in the second derivative of an image to detect edges.

Researchers introduced a fuzzy algebraic morphology calculation to distinguish the edge. An effectiveness of edge credit in light of the fuzzy morphology is demonstrated by contrasting the results of alternate calculations, for example, Sobel and binary morphology, the edge identification calculation proposed by this paper have pointed of interest like no threshold required, high invulnerability to noise, extraction of texture and the vein. Additionally, research is to figure the area of tobacco leaf and the length of a vein in light of the consequence of edge recognition.

10.2.4 Region-Based Image Segmentation

In region based segmentation we divide the images into different regions. And then we decide which pixels belong to which region or classes according to their similarity index value. A variety of techniques are used for region based segmentation. Although the methods are different, the results are almost same. The common types of region-based segmentation are region splitting, region growing and region merging [20].

Darshana *et al.* [18] in this paper used region-based technique for finding the disease area. Among the three region-based segmentation techniques, they found the best one. On the basis of time and quality analysis they decided that, Region growing gave the best result to represent distinct regions having least discrete entropy value and the highest gray level energy as compared to the mean-shift segmentation methods.

10.2.5 Watershed

Researcher presented the idea of texture gradient and had utilized it to create a successful watershed segmentation strategy for common images in light of intensity and texture boundaries. Moreover, an efficient algorithm has been implemented to check the problem of over-division while holding

key slope restrictions providing no little lingering districts. Utilizing this choice of idea with a typical image slope will regularly prompt powerful division for non-surface pictures. By using this, markers removed from the created least district, moving edge and locale developing strategy, homogeneously finished areas can be distinguished. This marker extraction technique utilizes a similar inclination picture as the along these lines executed watershed change. All little scale inclination elements are consequently safeguarded regularly making a more powerful division.

10.3 How to Deal With Image Dataset

10.3.1 Introduction

This chapter briefly describes about the principles, procedures and methodologies used to achieve the expected results and outcome of the research including the flow chart and algorithms of the entire research works. These procedures include the procedure for acquiring the images, pre-processing, methods of segmenting and algorithm to detect the area of the infected part of the leaf through the image.

10.3.2 Image Acquisition

In digital image processing image acquisition is the first step where we retrieve an image from hardware. The query image is taken for various operations that we carry out in image processing techniques, the image is pass through various process for proper studying of image [7]. The main aim of image acquisition is to transform the real world data into the numerical data so, that we can able to read the details of image using their pixel values. The numerical value of the image later manipulated in the computer [41]. Image Acquisition process consists of three steps [42]:

1. Optical system that focus on the energy
2. Energy is reflected from the object
3. A sensor that measures the amount of energy on the object.

10.3.3 Image Pre-Processing

An image is the optical data that represents an object [44]. Image processing is just like signal processing. In image processing we give image as an input and as output we get either an image or some characteristics of the image to

be read out [43]. Image processing includes altering, enhancement, compressing, etc. to the existing image. Image processing is used to convert an image into digital form so that we can perform some operations on it, so that we can retrieve the enhanced image and some important information from it. The first step of pre-processing is to bring the query image into the processing field for further process. We perform the pre-processing so that it can reduce the time taken for further processing of the image. Image processing generally has three steps:

- Retrieve the real world data by some image acquisition tool;
- Analysing the numerical data of the image;
- Output is generated using image analysis.

Image processing is done so that we can improve the image quality for further processing of the image. Unwanted noise and distortions are removed in image processing [7]. Various techniques are used in pre-processing of image like morphological operations are done on image, changing shape and size of the image, filtering of noise on image, image enhancement is done and conversion of image. Here we are using various MATLAB code for resize of an image, and to enhance contrast and for RGB to gray scale conversion.

10.3.4 Image Enhancement

Image enhancement is done to sharpen the image features like edges, boundaries and contrast for better graphical display that is very useful for analysis of the image [43]. Enhancing is done for better display of the image. For subjective evaluation of the image quality contrast value is much needed one. By the difference in luminance reflected from two adjacent surfaces contrast is created. Contrast is the visual properties of the image which distinguish the object from any other object and from background also. According to visual perception, contrast is defined by the color and brightness of the object as compare to the other object. Various algorithms have been developed for accomplishing contrast enhancement, and it is used in the problem part of the image processing. According to visual perceptions of human beings image enhancement is used to improve the quality of an image. It is used for low level vision applications. Image enhancement is the technique which is used to enhance the image for better display of the image, better quality of image is used for further processing of the image [45]. Image enhancement helps to retrieve the important information regarding image and which helps in better visibility for the observer.

10.3.5 Image Segmentation

The most basic and important part in image processing is image segmentation. Image segmentation is used to segments the image into various meaningful parts. The segmented parts are divided according to their gray level, spectrum, texture, colour, and so on. The goal of image segmentation is to partition an image into a set of disjoint regions with uniform and homogeneous attributes such as intensity, color, tone or texture, etc. Image segmentation is one of the most fundamental, useful, and studied topics in image processing and analysis [31, 32, 37–40, 47, 48]. While performing every task our main goal is to represent every regions of the image into some meaningful information, just like the urban areas, crops, and forests of a satellite image. While analyzing the task the regions may set as the border pixels just like the structure of line segments or circular arc segments in 3D image. The studied region may consist of both the border and different types of shapes having circular, triangular or polygonal shape. If the segmented region does not cover the whole image we can still divide it into foreground region and background region of the image. Segmentation is done with respect to two objectives. Our first objective is two divide the image into different parts. The segmented parts are taken into consideration, for our further study we take the reasonable parts for further study. As we know in the chapter colour, an algorithm was proposed to segments the human face using color image video. Segmentation is very difficult in case of having different color components objects unlike human face. Because the human face is of same color components that make it easier to segments the parts. But in case of different color components objects we need high domain applications for better segments of the image and the second objective is to change the representation of the image in segmentation. The pixels of the image should be arranged in high level unit to make it meaningful and proficient for our further analysis. Segmentation algorithm must be required the following steps:

1. Segmented image should have uniform and homogeneous regions according to the segmented properties like gray lever and tone.
2. Interior regions should be simple and without having small holes.
3. The properties of adjacent region of each segment should vary according to their uniform characteristics.
4. Boundaries of the segment region should be not raged, simple, and must be spatially accurate.

10.3.6 K-Mean Clustering

K-mean clustering is popularly used to segmenting the image into various clusters. Clustering is the unsupervised learning method where we create uniform groups [46]. Different types of clustering are there: Fuzzy C-means clustering, hierarchical clustering and K-means clustering. K-means clustering is mostly used in different types of application [47]. K-means is a partition-based clustering method. K-means clustering is commonly used in various applications, it is used for low level segmentation. Choosing of initial cluster center is very important because it helps the clustering algorithm to produce right results [46]. Generally the initialization procedure chooses the initial cluster centre randomly from input data [48]. While performing the real image segmentation, we create an array of all clusters generated during initial phase. Then implement a loop to iterate the generated array, in each iteration of the loop we step through the array of super-pixel that is for the current cluster a centroid is created. For the super-pixel, we perform a linear search for finding those image pixels that are having the particular color distance which corresponds to the applied boundary condition. Later we create a new image of the selected pixels for the new cluster. According to the color of centroid super pixel the color of each pixels are replaced. The pixels which contain the same outline colour of the original image also contain by the new image. Then the new clusters are added with the array. Then, we decide the coordinates of the central point after finding the centre of mass of the array, of the super-pixels of the parent cluster and then, we check whether the centroid super-pixel of new cluster is moved according to central point of parent cluster. Then, only we add the new cluster to the array of clusters, otherwise not. Generally, we repeat these processes for each particular cluster of the array of clusters until no more clusters left to be processed. To obtain the final image, the segmented one we have to merge images of all the clusters and built into an overall image. The images which are associated with the clusters are also have some part of original image having similar types of colour.

10.3.6.1 Euclidian Distance

For image segmentation we are using k-means clustering algorithm. And this method is totally based on Euclidian distance formula. Here, we deal with the two main variants of the given formula for calculating the actual distance between two pixels, and for determining the similarity of two 3D-vectors colors.

The representation of the Euclidian distance formula:

$$(\mathbf{L})^2 = \sum_{\mathbf{i=1}}^{\mathbf{n}} (\mathbf{x_i} - \mathbf{y_i})^2 \tag{10.1}$$

Here L—determines the value of Euclidian distance

X and Y—are the two vectors for which we calculate the actual distance among them.

n—it represents the number of dimensions of the vectors, (n = 2). Here we deal with either 2D or 3D vectors. To calculate the actual distance between the two vectors (pixels) having their coordinates value we will use this Euclidian distance formula:

$$\mathbf{L} = \sqrt{(\mathbf{X_1} - \mathbf{Y_1})^2 + (\mathbf{X_2} - \mathbf{Y_2})^2} \tag{10.2}$$

Here L defines Euclidian distance value

(x1;x2) and (y1;y2)—the coordinate values of the Cartesian plane of the two pixels containing x and y coordinate values.

And, to find the magnitude of similarity value between the two 3D-vectors (R; G; B) colour value, we will use the same 3D-variant of Euclidian distance formula:

$$\mathbf{s} = \sqrt{(\mathbf{r_1} - \mathbf{r_1})^2 + (\mathbf{g_1} - \mathbf{g_2})^2 + (\mathbf{b_1} - \mathbf{b_2})^2} \tag{10.3}$$

S defines the value of similarity magnitude between the two vectors of colors (r1;g1;b1) and (r2;g2;b2) respectively.

10.3.6.2 Clustering

Image clustering and the categorization of image are done to find the high level unit that contain the image. Our main objective is finding a planning among the documentation of the images into various clusters (classes). And the classes deliver the same essential data of the image collection. The set of classes deliver us a summarization and the visualization of the image content that can be castoff for dissimilar purpose regarding the management of image database. Proposed flow chart for K-mean clustering is given below. In k-means, data will be included in one particular cluster, whereas in FCM, a data can be included in all existing clusters, but with varying degrees of membership, in a range of values [0 1]. Fuzzy clustering is also referred to as soft clustering or soft k-means. From Figure 10.1, we know briefly about K - means clustering algorithm.

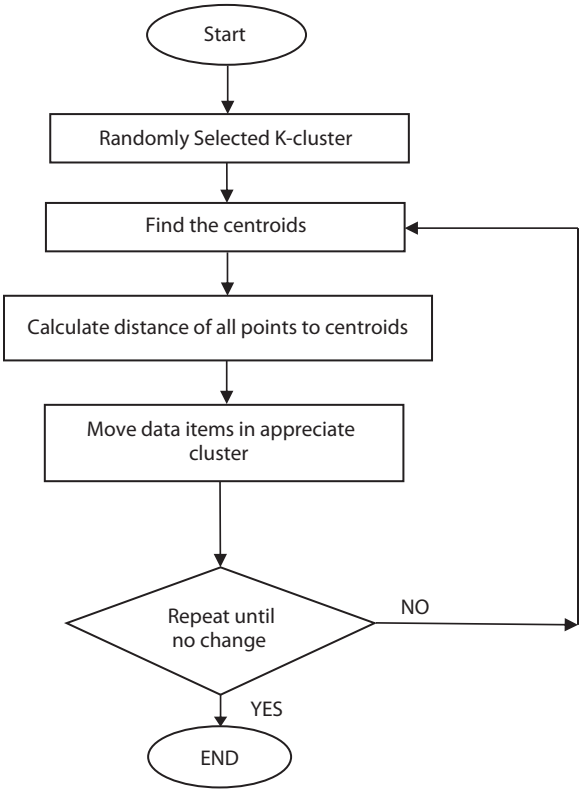


Figure 10.1 Flowchart for K-mean clustering.

10.3.7 Density-Based Spatial Clustering of Application With Noise (DBSCAN)

Clustering is usually an unsupervised learning method in which it divides the data points into a number of clusters [15]. Clusters are nothing but the data points having similar properties. Data in different groups or cluster have different properties. DBSCAN is based on clusters and noise. The basic idea is that for each value or data point of a cluster, the neighborhood of a given radius has to contain at least a minimum number of points. The partitioning method and hierarchical clustering method works for finding spherical or convex cluster. They are highly affected by outliers or noise. In DBSCAN two parameters are required.

eps: It is nothing but the radius of the cluster. The data points which come under the circle of radius *eps* are considered as one cluster. We assume the radius is also the part of the cluster.

Minpoints: The minimum number of data points required to form a dense cluster. If we set the *minpoint* at 5, then we require at least 5 points to form a cluster.

10.3.8 SVM Classifier

Support Vector Machines are a supervised classification technique. SVMs are used to manage a classical two class pattern recognition problem. Because it is supervised learning algorithm it can be used to solve both classification and regression problem. SVM is generally used to solve classification problems. Traditional SVM gives binary value, the class of an object. We basically focus on these two classes: the dissimilarities of the same species images, and dissimilarities of the different species images. If the data has exactly two classes then we can use SVM. SVM classifies the data according to the hyper-plane concept that helps to separate the data points of one class to another class. If the hyper plane gives the large margin between the classes then it gives the best result. The three cases to classify the data points and to separate them into their respective classes:

- Linear SVM (separable Case)
- Linear SVM (non-separable Case)
- Non-Linear SVM.

An example is given below for the linear SVM in separable case. Here the line is considered as the boundary line or the hyper plane which divide the two classes. One is red and the other one is green. The principle of SVM is shown in Figure 10.5. The data points which belong to red class are shifted to left side of the plane and the data points those belong to the green class are shifted to right side of the plane. According to their uniformity they are divided into two classes.

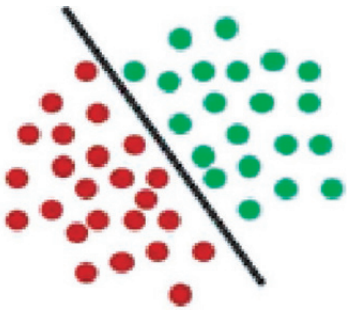


Figure 10.2 Linear classifier.

Figure 10.2 shows about the hyper-plane classifier. The hyper plane classification is done using a curve line, which is more difficult to classify the data points as compared to the linear classifier. The minimum distance is based between the data point and the center of class of the data classified, so it is very difficult to decide in hyper plane. But SVM makes it easier somehow.

Here we are using an optimal line to separate the data points into their suitable classes. From Figure 10.3, we can know about the hyper plane classifier. Left side classification is based on the curve line which makes it difficult for separation. A mathematical calculation is used to decide the classes. Kernel trick is used here to separate the data points. But in linear case it can be easily handled. Right side we are using linear classification. Two classes are used here: red and green. A diagram or Figure 10.4 is drawn below to demonstrate the working of SVM.

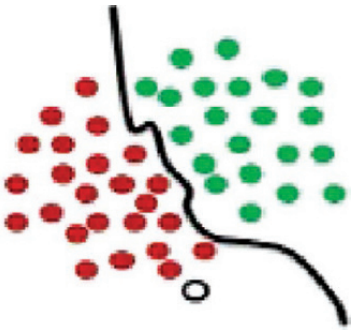


Figure 10.3 Hyper plane classifier.

Figure 10.4 Optimal line.

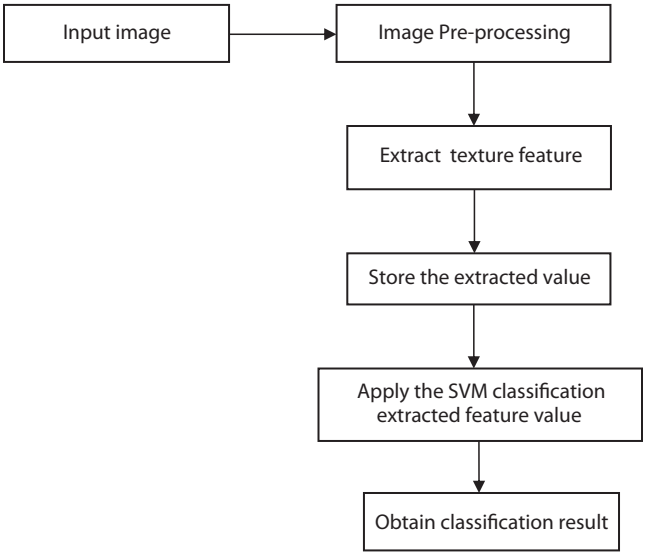


Figure 10.5 Working process of SVM.

10.4 Class Imbalance Problem

It is a problem in machine learning when the class distribution is highly imbalanced which means there is an unequal distribution of positive and negative class. In such case, many classification learning algorithms give false sense of accuracy because unbalanced dataset bias the model towards the more frequent class. Almost all machine learning algorithms give best results when the sum total of instances of one class is almost equal to the other class. To overcome the class imbalance or skewness problem following techniques are followed.

10.4.1 Resampling Approaches

This approach reduces the skewness in the dataset. It is accomplished either by increasing the count of minority class sample to accomplish equivalent proportion with the majority class sample or decreasing the count of majority class sample to accomplish equivalent proportion with minority class sample. Three basic resampling methods include Random Undersampling (RUS), Random Oversampling (ROS) and a hybrid of both the methods.

(a) Under-Sampling Approach

Under-sampling method is a non-heuristic method which removes the count of majority class sample so that it accomplishes equal distribution with minority class sample. By applying this method the size of the training dataset decreases so that it improves the running time and storage problem.

(b) Over-Sampling Approach

Over-sampling is a data balancing method in which we increase the count of minority class sample so that it accomplishes equal distribution with majority class sample. In this method there is no informal loss of data samples but it is prone to over fitting as we have simply copying of same information.

(c) Hybrid Approach

By combining under sampling and over sampling approach, we can prepare a hybrid approach which is shown in Figure 10.6. AI (ML) methods have been utilized for quite a long time in clinical picture examination and processing for unraveling shrouded tissue properties ready to productively portray pathophysiology and advance the improvement of Computer Assisted Diagnosis (CAD) clinical frameworks, showing amazing outcomes and in any event, coordinating or outperforming the exhibition of clinical specialists [1, 2].

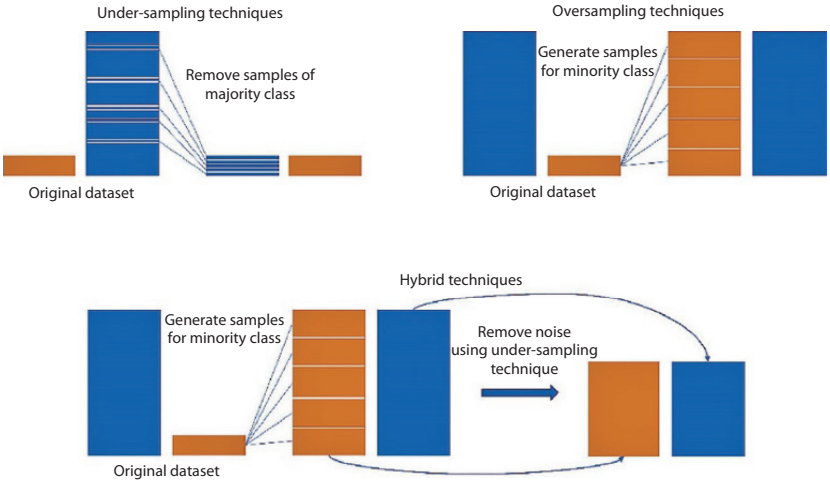


Figure 10.6 Hybrid algorithm.

During the most recent years, a steady move of exploration patterns in picture handling and PC vision has occurred from shallow carefully assembled calculations to DL structures [3, 6–8]. The greatest preferred position of profound structures is the concurrent preparing procedures of highlight related undertakings like extraction, determination, decrease and arrangement inside a blended, start to finish learning process. All things considered, not many endeavors have been made to join the best of the two universes as in profound Fisher piece systems [7, 8, 11–13] where rather than independent advances, the Fisher pipeline is converged into a profound layer with the goal to take care of one raised enhancement issue. Additionally in Refs. [4, 5, 9, 10], half breed models of DL systems followed by Fisher Vector encoding are proposed for picture recovery and surface division, individually, without being capable anyway to coordinate best in class level execution. Simultaneously, the accessibility of "enormous information" appears to step by step become a sine qua non-condition for preparing systems ready to show noteworthy outcomes past the best in class.

This can be an issue in clinical picture investigation DL applications dependent on clinical datasets which are limited in size because of: a) the lawful and moral limitations securing the secrecy of patients, b) distinctive picture obtaining conventions and strategies utilized in the clinical daily schedule, c) significant expense to make and minister such databases, and d) the heterogeneity of pathologies (for example sub-atomic heterogeneity of malignant growth), directing the formation of a wide range of, case-explicit datasets. Thus, the ordinary number of tests for preparing and testing objects, is altogether littler contrasting with different areas. To conquer these impediments, analysts have proposed compensatory procedures, the greater part of them tended to in Refs. [14–18, 51, 21–28] including half breed organize structures, move learning, antagonistic systems, eager layer-wise preparing and solo pre-preparing. In Refs. [19], a clinical imaging application identified with sectioning organs from clinical sweeps accomplished promising outcomes in little example size setting with changed models, for example, fell systems. Notwithstanding the dataset size impediments, DL structures and explicitly CNNs have been effectively actualized into the clinical imaging area with a few exceptional applications in grouping, division, recognition and picture enlistment, beating traditional ML and picture investigation strategies dependent on natural component extraction [20]. Ongoing instances of clinical picture grouping incorporate bosom malignant growth order [11], glioma portrayal [12], head

and neck disease arrangement [13], and greasy liver evaluating [14]. Different CNNs incorporate cerebrum MRI tumor division [15], entire heart division [16], liver tissue division [17] and MRI pancreas division [18]. Noteworthy division results have likewise been as of late accomplished in 3D CNNs, for example, in Ref. [19] for sore division from multi-divert MRI and in Ref. [20] for liver division from 3D Computed Tomography (CT) outputs and entire heart and incredible vessels division from 3D MRI.

10.5 Optimization of Hyperparameter

Hyperparameters are including the number of layers, learning rate, number of scholarly pieces, size of respons ive field, neurons of completely associated to layer and the quantity of expanded examples are dictated thorough system learning. The objective was to boost precision, limit overfitting and keep away from remembrance of the dataset by the system. Measurements like preparing misfortune and preparing exactness were compared to their approval partners. Different hyperparameters, for example, the analyzer, the initializer, the actuation work or the step were at first decided by the writing. Softmax was picked as initiation work for the order layer abusing the extra likelihood scores of the forecasts that we intend to use in CAD framework for this medical image application. Then sigmoid function enactment was additionally tried however prompted comparable (marginally second rate) results. All previously mentioned boundaries were balanced during this streamlining procedure [34, 35]. Support Vector Machines (SVMs) are classifiers broadly used to an assortment of administered learning issues showing great execution, regularization and arched advancement and it helps in tissue classification. From Figure 10.7, we know about c means clustering. Fuzzy clustering is a form of clustering in which each data point can belong to more than one cluster. Clustering or cluster analysis involves assigning data points to clusters such that items in the same cluster are as similar as possible, while items belonging to different clusters are as dissimilar as possible. This algorithm works by assigning membership to each data point corresponding to each cluster center and the data point. More the data is near to the cluster center more is its membership towards the particular cluster center. From Figure 10.8, we conclude about workflow of the various medical problem statements (data sampling, data augmentation, model selection, optimization).

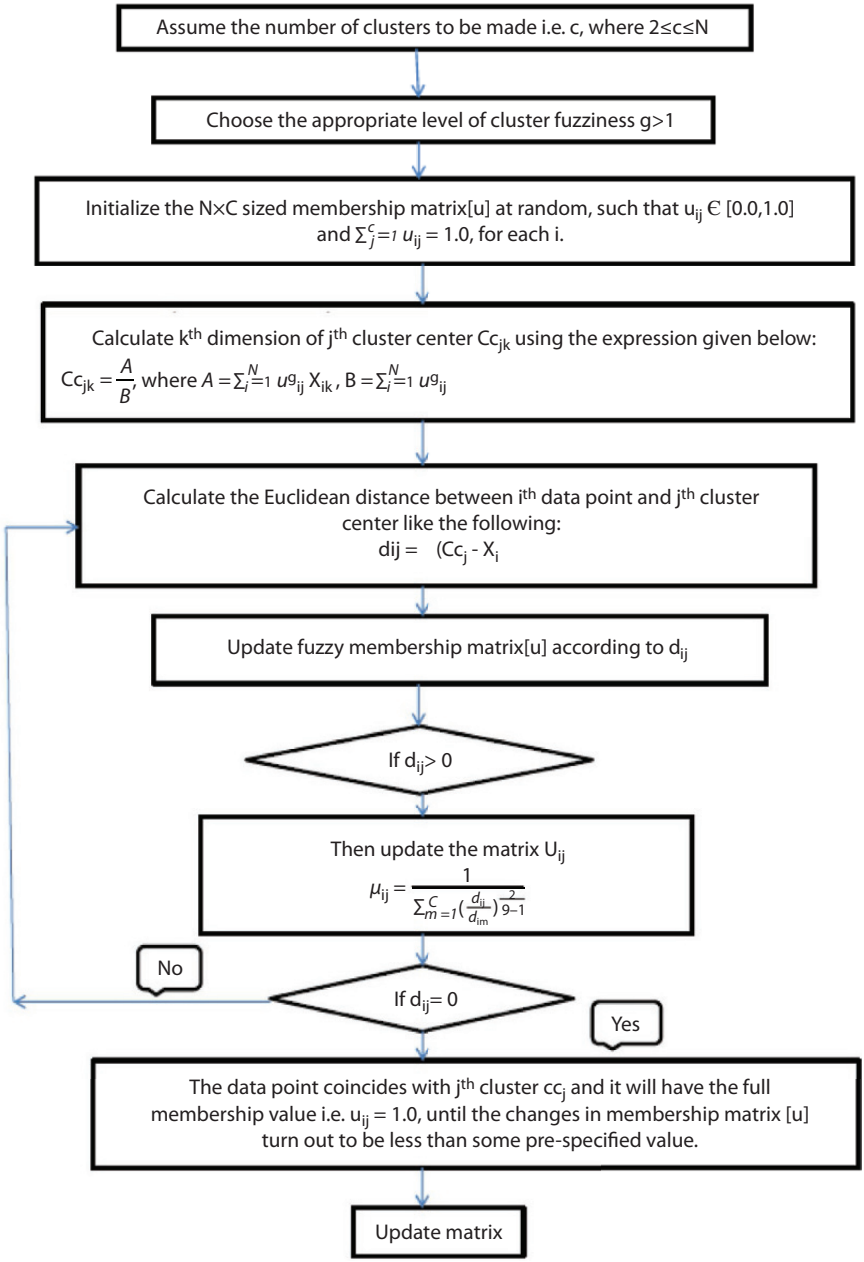


Figure 10.7 C—means clustering algorithm.

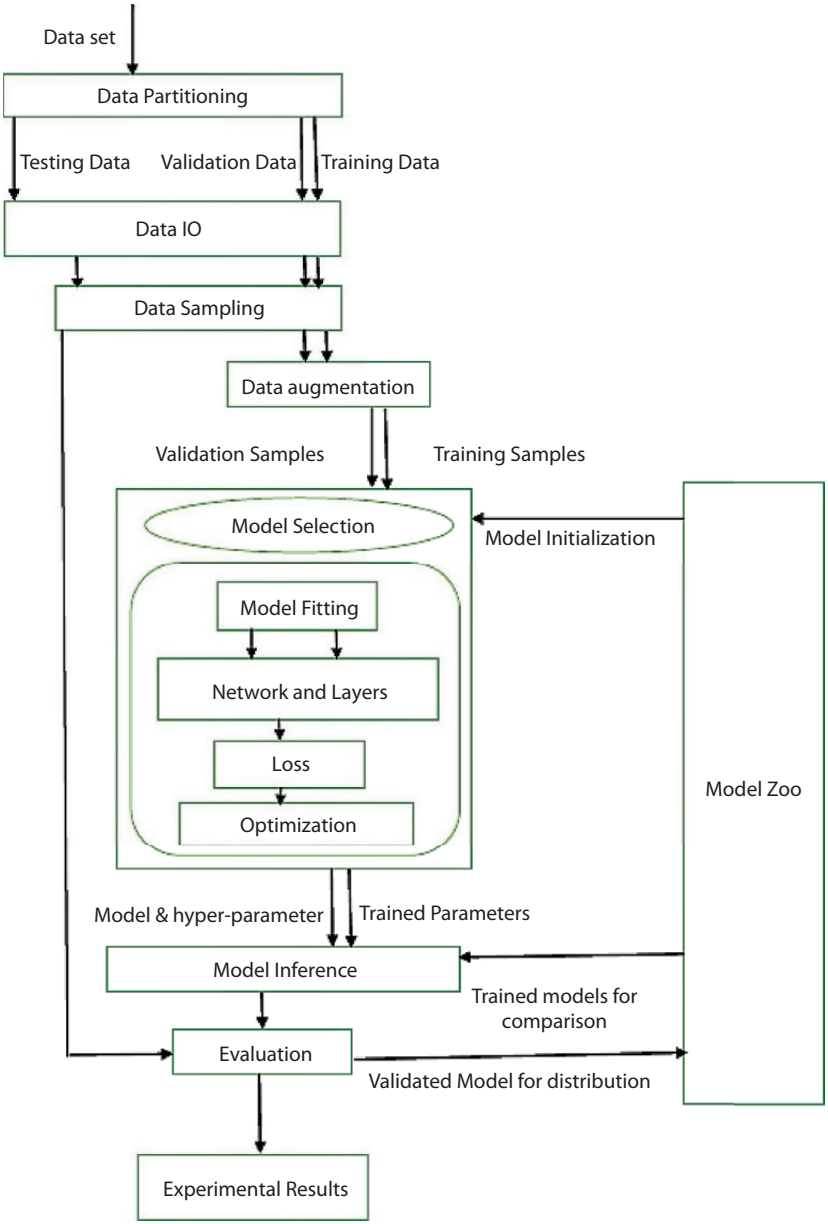


Figure 10.8 Model of data prediction.

10.6 Case Study

10.6.1 Pancreatic and Lung Tumor Prediction in the Machine Learning Era: Unique Supervised and Unsupervised Methodologies

Disease separation (tumors) from radiology pictures can be progressively exact and quicker with computer aided conclusion (CAD) apparatuses [36, 49–51]. Tumor portrayal through such apparatuses can likewise empower non-intrusive disease organizing, guess, and encourage customized treatment arranging as a piece of exactness medication. In this investigation, we propose both directed and solo AI procedures to improve tumor portrayal.

Our first methodology depends on regulated learning for which we show huge increases with deep learning calculations, especially by using a 3D CNN and Transfer Learning. Around 40% of individuals will be determined to have disease sooner or later during their lifetime with a general mortality of 170.2 per 100,010 individuals for every year (in view of passings between 2008 and 2012) [1]. How semi supervised learning is useful, we can know it from Figure 10.9. Risk stratification of tumors from radiology images can

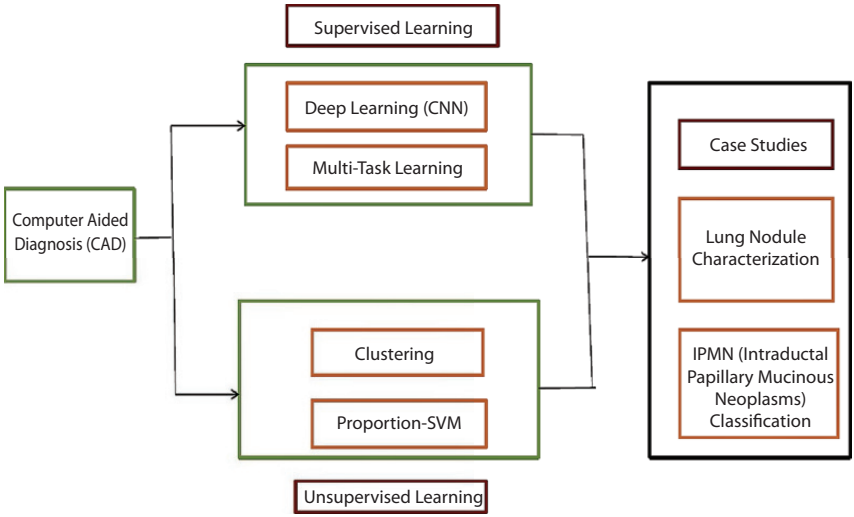


Figure 10.9 The flow-chart signifies different systems, approaches and investigational case trainings offered in this research. We improve both supervised as well as unsupervised algorithms to describe about tumors. For the supervised system, we suggest a 3D CNN design based on a graph regularized sparse multi-task learning and achieve assessments for the lung nodule description from CT scans images. For unsupervised system, we suggest a clustering algorithm and test it for the classification of lung nodules and the pancreatic cysts (IPMN) from CT and MRI scans report.

be more accurate and faster with computer aided diagnosis (CAD) tools. Tumor characterization through such tools can also enable non-invasive cancer staging, prognosis and foster personalized treatment planning as a part of precision medicine. In this study, we propose both supervised and unsupervised machine learning strategies to improve tumor characterization. Our first approach is based on supervised learning for which we demonstrate significant gains with deep learning algorithms, particularly by utilizing a 3D Convolution Neural Network and Transfer Learning. Motivated by the radiologists' interpretations of the scans, we then show how to incorporate task dependent feature representations into a CAD system via a graph-regularized sparse Multi-Task Learning (MTL) framework. In the second approach, we explore an unsupervised algorithm to address the limited availability of labeled training data, a common problem in medical imaging application. Inspired by learning from label proportion (LLP) approaches in computer vision, we propose to use proportion SVM for characterizing tumors. Lung and pancreatic tumors are two of the most widely recognized malignant growths. While lung malignant growth is the biggest reason for disease related passings on the planet, pancreatic malignancy has the least fortunate forecast with a 10-year endurance pace of just 8% in the USA [1]. With respect to pancreatic malignancy, explicitly in this work, we center around the difficult issue of programmed determination of Intraductal Papillary Mucinous Neoplasms (IPMN). This is a pre-threatening situation and whenever left untreated, it can advance to obtrusive malignancy. IPMN is mucin-delivering neoplasm that can be found in the primary pancreatic pipe and its branches. Recognition and portrayal of these lung and pancreatic tumors can help in early conclusion; thus, expanded endurance chance through suitable treatment/medical procedure plans. Customarily, the CAD frameworks are intended to help radiologists in settling on precise and quick choices by decreasing the quantity of bogus positives and bogus negatives. For indicative dynamic, a higher accentuation is laid on expanded affectability: a bogus banner is more mediocre than a tumor being missed or mistakenly named favorable. In such manner, an electronic investigation of imaging highlights turns into a key elements for radiologists to develop their demonstrative choices.

We use the data around six elevated level knob qualities, for example, calcification, spiculation, sphericity, lobulation, edge, and surface to improve programmed kind hearted dangerous arrangement. At that point, we coordinate these elevated level highlights into a novel chart regularized perform multiple tasks learning (MTL) system to yield the last harm yield. We dissect the effect of the previously mentioned lung knob properties inside and out for threat assurance and see these qualities as reciprocal

while getting the danger scores. From a specialized point of view, we additionally manipulate distinct regularizers and perform multiple tasks learning approaches, for example, follow standard and diagram regularized MTL and SVM classifier for to select features from the images and Classifiers.

Expectedly, the hazard separation (order) of lung knobs may require knob division, calculation and determination of low-level highlights from the picture, and the utilization of a classifier/regressor. In the methodology by Ref. [42], diverse physical insights including power measures were removed and class names were gotten utilizing Artificial Neural Networks. In Ref. [3], lung knobs were divided utilizing appearance-based models followed by shape examination utilizing circular sounds. The last advance included k-closest neighbor based order. Another methodology broadened 2D surface highlights including LBP (Local Binary Patterns), RF (random forest). In an alternate report, knob division by means of 3D dynamic forms, and afterward applied elastic band fixing change. A LDA (Linear Discriminant Analysis) was used to find the class level. Researchers [6] presented a component determination based methodology using medical information. Information substance and highlight pertinence were estimated utilizing a group of hereditary calculation and irregular subspace technique. In conclusion, LDA was applied to get last grouping on the dense list of capabilities. In an ongoing work, round music highlights were intertwined with profound learning highlights [8] and afterward RF order was utilized for lung knob portrayal. Up to this point, the use of convolution neural network for knob portrayal has been restricted to two dimensional space [43], subsequently missing the mark concerning fusing essential relevant and volumetric data.

10.6.2 Pancreatic Cysts (IPMN)

In spite of the fact that there has been impressive advancement in creating programmed ways to deal with fragment pancreas and its sores [19, 20], the utilization of cutting edge AI calculations to perform completely programmed hazard definition of IPMNs is constrained. Typically, the visual acknowledgment and grouping errands are tended to utilizing marked information (oversight). In any case, for errands where physically producing marks relating to huge datasets is relentless and costly, the utilization of solo learning techniques is of huge worth. Solo procedures had been utilized to tackle issues in different areas running from object order [23, 33], discourse preparing [44] and sound arrangement [25]. These strategies ordinarily depended on some correlative data gave the information to improve realizing, which may not be accessible for a few characterization errands in clinical imaging. Autoencoder and GAN (generative adversarial network) are used for better classification.

10.7 Using AI to Detect Coronavirus

10.7.1 BlueDot AI Technology

BlueDot is a start-up company in Toronto which uses a platform built based on Artificial Intelligence, machine learning and Bid data. They use them to locate and predict any diseases which can be infectious in nature and how they can be spread around the globe. They were the first to alert the government as well as the private sector officials about the cluster of coronavirus occurring around Wuhan, China which was quite unusual as a phenomenon. After this detection, within nine days the World Health Organisation sent the news of the outbreak of coronavirus along with 4,000 deaths.

Dr. Kamar Khan from the University of Toronto had studied the science behind spreading of infectious diseases across the globe and therefore had worked to create a technology in order to predict, locate and mitigate the outbreaks. The Blue dot gathers information of around 150 diseases along with their syndromes from around the world within 15 min and keeps updating them regularly within 24 h a day. They also retrieve information from WHO or Centre for Disease Control. The BlueDot's prediction is based on the data it receives from sources outside such as climate data from satellites, movement of travellers on flights, information from local healthcare centers and even the journalists.

The data is then manually classified by the experts, so that they can create a taxonomy. On doing so, the keywords could be scanned and then be used to train the machine with the help of machine learning and natural Language processing. Regular alerts can be sent to the government, health care centers, business and public clients by BlueDot. These alerts are in the form of brief synopses that their AI engine has found out about the outbreak of this weird disease. They even predict whether these diseases can pose any threats to the community or not.

The system had found out that the cases of pneumonia in China were reported as 27 which were related to a market which used to sell seafood and live animals. BlueDot sent the alert alarms and helped in identifying the cities which were connected to Wuhan and that too correctly. They used all information sources ranging from data from global airline ticketing to minor traveling data of a person to predict whether is affected by coronavirus through traveling or not. BlueDot predicted the international destinations which would have the highest number of travelers from Wuhan were Hong Kong, Taipei, Phuket, Bangkok, Tokyo, Seoul and Singapore. And there prediction turned into a true event because these cities were the first places to have COVID-19 patients.

10.8 Using Artificial Intelligence (AI), CT Scan and X-Ray

The use of AI over the preceding few decades has become more of a norm than a gimmick that it was considered in its early years. Due to increased data volumes, advanced algorithms, and improvements in computing capabilities and storage, AI has advanced into scientific innovation that can possibly pave way towards future advancements in scientific studies. But still the reliability of an intelligence system made by man itself, to solve human level issues might be received with criticism. Therefore its employment in critical sectors like Healthcare, Business, and other complicated core sectors had been fairy minimal. However, with the onset of the global pandemic of Covid-19, scientists and doctors were forced to their knees with the tremendous amount of pressure that came over this highly contagious virus proliferating over almost the entire world. The escalation of the situation was such a sudden hit that leaders across the globe had to take on unconventional decisions that would have usually taken at least a couple of years of testing and implementation. One such step was taken by the Chinese technology giant Alibaba. They have started implementing an AI system that successfully diagnoses the COVID-19. Alibaba is a conglomeration of various business departments such as software, gaming, online marketing, healthcare, etc. All of these vastly diverse sectors are technologically guided by the company's world-class AI department.

China had the first reported case of COVID-19 infection, and Wuhan was the first city to face the outbreak. It's obvious to say that China as a country had more time dealing with the crisis quantitatively. Being one of the worst affected cases (not 'the' worst affected) China too was to take steps that were previously unaccounted for.

A report states that, the AI system detects the virus using CT scans and the process and outcome is as far as 96% accurate against viral pneumonia cases, a common symptom of COVID-19. Moreover, it takes average personnel about 15 min to completely diagnose the images and confirm the infection, but this AI takes about 20 s only to make a determination over the same. It's claimed to have been trained over 5,000 positive cases of the infection and has yielded good results over testing in various COVID-19 hospitals. In fact, this system is under use by a good number of healthcare facilities. According to other sources, Alibaba isn't the only tech company to have thought and taken these steps but even other competitors have started to take initiatives and joined in the cause. Ping An, another Chinese company focused over healthcare sector recently revealed that they have

too developed an AI based system that also determines the infection status based upon image reading. According to the co-president of the company, more than 1,500 healthcare firms have benefitted from their technology and have been carrying out diagnosis using the same. About 5,000 people have already been diagnosed by the AI system free of cost. This system too can generate unerring analysis within 15 s with pin-pointed accuracy.

Coming out of China's secretive research and enforcements, other countries have also openly embraced the idea of implementing AI into healthcare sectors to maximize efficiency and minimizing the load over the healthcare personnel. UK has also taken a pioneering position in this segment of research. Cranfield University, UK has seen its computer and machine vision students proposing a project which uses AI to analyse chest X-rays images, and detecting minute details usually missed by the naked eye. This AI tech not only can pick up anomalies that lines up to pneumonia, a COVID-19 symptom, but also determine if the particular case is due to the SARS-CoV-2. Conventional machine learning algorithms and deep learning frameworks were put under use for development of this AI program.

The system responded with pronounced accuracy, which could be further improved with finer algorithms encoded via future updates. The team is now hoping to work with computer engineering professionals to take the next step by putting in advanced algorithms and training the AI to work better with Computer Tomography (CT) scans [35, 36, 45, 51] to increase its productivity. With proper fine tuning and an interface to work with this system could significantly contribute to the cause of containing the global crisis of COVID-19, as said by Dr. Zeeshan Rana, lecture at Cranfield, and the person leading this pilot project. King's College London, Massachusetts General Hospital, and health tech firm Zoe have put an AI-based diagnostic under trials, trying to correlate symptoms and results of traditional COVID-19 tests to enhance the AI predictions. Benevolent-AI, UK, put up its platform to distinguish a marketed drug—Eli Lilly's rheumatoid arthritis therapy Olumiant (baricitinib)—which can be a potential treatment for the SARS-CoV-2 virus. The drug was committed to trials by the US National Institute for Allergies and Infectious Diseases (NIAID) in February. Exscientia, an Oxford-based company took up the initiative to first ever trial an AI-discovered drug over a Human Subject. The company is now sifting through 15,000 drugs along with Scripps research institute, US, trying to isolate the ones that possibly show any trace of reactivity towards the SARS-CoV-2 virus. Cambridge-based Healx used its AI platform to analyse 4,000 pre-existing drugs that can be repurposed to work against the Global Pandemic. The point is, even though humanity has come

out of worse situations before, the COVID-19 pandemic hits the very core lifestyle of people that has been established since decades. Therefore, we are looking at such a pressure misbalance in our society, where one sector is heavily burdened and another can only sit back and watch. But what we have today against the old times is tremendous advancements in technology. So, it's only sensible to implement it to our advantage and relieving the pressure off of our Healthcare Sectors. Implementations of such ground breaking tech would definitely provide a breath of freshness if not complete salvation.

10.9 Conclusion

The World Health Organization (WHO) [50] records cyst formation in the organs as the main source of death all around with 17.9 million individuals kicking the bucket each year. Clustering technique used for segmenting the image into various cluster. Image is first converted into histogram and then clustering is applied on it. There are different types of clustering methods like, optimized clustering by using nature inspired algorithm, K-means clustering, DBSCAN, Fuzzy C-means clustering method. One of the basic clustering algorithms is K-means, which is intended for segmentation in textured images [20]. It clusters same pixels to segment the image. Conclusion is an issue for experts due the side effects' tendency of being normal to different conditions or mistook for indications of maturing. The development in clinical information assortment presents another open door for doctors to improve tolerant conclusion. As of late, professionals have expanded their use of PC advances to improve dynamic help. In the human services industry, AI is turning into a significant answer for help the conclusion of patients. AI is an investigative instrument utilized when an undertaking is huge and hard to program, for example, changing clinical record into information, pandemic forecasts, and genomic information examination [4].

References

1. Srinivas, M., Roy, D., Krishna Mohan, C., Discriminative Feature Extraction of X-ray Images using Deep Neural Networks. *Proc. 41st IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2016)*, Shanghai, China, Mar 2016.

2. Srinivas, M. and Krishna Mohan, C., Classification of Medical Images Using Edge Based Features and Sparse Representation. *Proc. 41st IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2016)*, Shanghai, China, Mar 2016.
3. LeCun, Y., Bengio, Y., Hinton, G., Deep learning. *Nature*, 521, 7553, 436–444, 2015.
4. Srinivas, M. and Krishna Mohan, C., Medical Image Indexing and Retrieval using Multi-Feature Extraction Method, in: *Proc. IEEE Int. Conf. on Computational Intelligence and Information Technology (CIIT)*, Mumbai, Oct. 2013, Elsevier.
5. Srinivas, M. and Krishna Mohan, C., Efficient clustering approach using incremental and hierarchical clustering methods, in: *Proc. IEEE Int. Conf. on International Joint Conference on Neural Networks (IJCNN)*, Barcelona, July 2010.
6. Donoho, D., Compressed sensing. *IEEE Trans. Inf. Theory*, 52, 1289–1306, April 2006.
7. Guha, T. and Ward, R., A sparse reconstruction based algorithm for image and video classification, in: *Proc. IEEE Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3601–3604, 2012.
8. Brezeale, D. and Cook, D., Automatic video classification: A survey of the literature. *IEEE Trans. Syst. Man Cybern. Part C: Appl. Rev.*, 38, 3, 416–430, May 2008.
9. Xiang, M., Schonfeld, D., Khokhar, A.A., Video event classification and image segmentation based on non causal multidimensional hidden Markov models. *IEEE Trans. Image Process.*, 18, 6, 1304–1313, June 2009.
10. Smeulder, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R., Content based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22, 12, 1349–1380, 2000.
11. Cai, W., Feng, D., Fulton, R., Content-based retrieval of dynamic PET functional images. *IEEE Trans. Inf. Technol. Biomed.*, 4, 2, 152–158, June 2000.
12. Pourghassem, H. and Ghassemian, H., Content based medical image classification using a new hierarchical merging scheme. *Comput. Med. Imaging Graph.*, 32, 8, 651–661, 2008.
13. Krawczyk, B. and Schaefer, G., Ensemble fusion methods for medical data classification, in: *Proc. IEEE Int. Symposium. Neural Network Applications in Electrical Engineering (NEUREL)*, 20–22 Sept. 2012, pp. 143–146.
14. Peng, F., Li, L., Xu, W., Liu, W., Zhang, J., Shao, G., The Identification of Breast Mass based on Multi agent Interactive Information Fusion Method, in: *Proc. of IEEE Int. Conf. Bioinformatics and Biomedical Engineering*, 11–13, June 2009.
15. Mitchell, T., *Machine Learning*. McGraw Hill, 1997.
16. Abu-Mostafa, Y.S., Magdon-Ismail, M., Lin, H.-T., *Learning From Data*, AMLBook, 1999.

17. Hastie, T., Tibshirani, R., Friedman, J., *The elements of statistical learning. Data mining, inference, and prediction*. Springer, 2009.
18. Bishop, C., *Pattern Recognition and Machine Learning*. Springer, 2006.
19. Duda, R.O., Hart, P.E., Stork, D.G., *Pattern Classification*, Wiley, 2011.
20. Bengio, *Practical Recommendations for Gradient-Based Training of Deep Architectures, in Sutskever, 2013, Training Recurrent Neural Networks* (PhD Thesis), p. 24, 2012.
21. Le, J. and Hinton, A Simple Way to Initialize Recurrent Networks of Rectified Linear Units, p. 18, 2015.
22. Sutskever, *et al.*, On the importance of initialization and momentum in deep learning, p. 18, 2013.
23. Li, and Karpathy, CS231n: Convolutional Neural Networks for Visual Recognition (Course Notes).
24. Srivastava, *et al.*, Dropout: A Simple Way to Prevent Neural Networks from Overfitting, p. 18, 2014. *Ibid*.
25. Zeiler, and Fergus, Stochastic Pooling for Regularization of Deep Convolutional Neural Networks, p. 18, 2013.
26. Wertheim, J.A., Petrowsky, H., Saab, S., Kupiec-Weglinski, J.W., Busuttil, R.W., Major challenges limiting liver transplantation in the United States. *Am. J. Transplant.*, 11, 1773e84, 2011.
27. Weismuller, T.J., Fikatas, P., Schmidt, J. *et al.*, Multicentric evaluation of model for end-stage liver disease-based allocation and survival after liver transplantation in Germany—Limitations of the "sickest first"-concept. *Transpl. Int.*, 24, 91e9, 2011.
28. Dutkowski, P., Linecker, M., DeOliveira, M.L., Mullhaupt, B., Clavien, P.A., Challenges to liver transplantation and strategies to improve outcomes. *J. Gastroenterol.*, 148, 307e23, 2015.
29. Han, G., Liu, F., Tian, Y., Wang, H., Wang, J., Wang, Y., Detection of glucose concentration in a turbid medium using a stacked auto-encoder deep neural network. *Infrared Phys. Technol.*, 105, 103198, 2020.
30. Koshimizu, H., Kojima, R., Kario, K., Okuno, Y., Prediction of Blood Pressure Variability Using Deep Neural Networks. *Int. J. Med. Inform.*, 1, 104067, 2020.
31. Kandi, H., Mishra, D., Gorthi, S.R.K.S., Exploring the learning capabilities of convolutional neural networks for robust image watermarking. *Comput. Secur.*, 65, 247–268, 2017.
32. Dvorák, P. and Menze, B., Structured prediction with convolutional neural networks for multimodal brain tumor segmentation. *MICCAI Multimodal Brain Tumor Segmentation Challenge (BraTS)*, pp. 13–24, 2015.
33. Simonyan, K. and Zisserman, A., Very deep convolutional networks for large-scale image recognition, p. 24, 2014. [Online], Available: http://arxiv.org/abs/1409.1556, arXiv:1409.1556v6.

34. Hafiz, A.M. and Bhat, G.M., A Survey of Deep Learning Techniques for Medical Diagnosis. *Advances in Intelligent Systems and Computing*, pp. 161–170, 2019.
35. Siva Raja, P.M. and Rani, A.V., Brain tumor classification using a hybrid deep autoencoder with Bayesian fuzzy clustering-based segmentation approach. *Biocybern. Biomed. Eng.*, 27, 2020.
36. Khan, H., Shah, P.M., Shah, M.A., ul Islam, S., Rodrigues, J.J.P.C., Cascading handcrafted features and Convolutional Neural Network for IoT-enabled brain tumor segmentation. *Comput. Commun.*, 27, 2020.
37. Das, H., Naik, B., Behera, H.S., Medical disease analysis using Neuro-Fuzzy with Feature Extraction Model for classification. *Inf. Med. Unlocked*, 18, 100288, 2020.
38. Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Larochelle, H., Brain tumor segmentation with Deep Neural Networks. *Med. Image Anal.*, 35, 18–31, 2017.
39. https://medium.com/analytics-vidhya/.
40. Jiao, Z., Gao, X., Wang, Y., Li, J., A deep feature based framework for breast masses classification. *Neurocomputing*, 197, 221–231, 2016.
41. Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., Sánchez, C.I., A survey on deep learning in medical image analysis. *Med. Image Anal.*, 42, 60–88, 2017.
42. Abdelaziz Ismael, S.A., Mohammed, A., Hefny, H., An Enhanced Deep Learning Approach for Brain Cancer MRI Images Classification using Residual Networks. *Artif. Intell. Med.*, 6, 101779, 2019.
43. Ghassemi, N., Shoeibi, A., Rouhani, M., Deep neural network with generative adversarial networks pre-training for brain tumor classification based on MR images. *Biomed. Signal Process. Control*, 57, 101678, 2020.
44. Amin, J., Sharif, M., Gul, N., Yasmin, M., Shad, S.A., Brain Tumor Classification based on DWT Fusion of MRI Sequences using Convolutional Neural Network. *Pattern Recognit. Lett.*, 1, 2019.
45. Saba, T., Sameh Mohamed, A., El-Affendi, M., Amin, J., Sharif, M., Brain Tumor Detection using Fusion of Hand Crafted and Deep Learning Features. *Cognit. Syst. Res.*, 27, 2019.
46. Toğaçar, M., Ergen, B., Cömert, Z., BrainMRNet: Brain Tumor Detection using Magnetic Resonance Images with a Novel Convolutional Neural Network Model. *Med. Hypotheses*, 1, 109531, 2019.
47. Vijh, S., Sharma, S., Gaurav, P., Brain Tumor Segmentation Using OTSU Embedded Adaptive Particle Swarm Optimization Method and Convolutional Neural Network, *Lecture Notes on Data Engineering and Communications Technologies*, 10, 171–194, 2019.
48. Amin, J., Sharif, M., Gul, N., Raza, M., Anjum, M.A., Nisar, M.W., Bukhari, S.A.C., Brain Tumor Detection by Using Stacked Autoencoders in Deep Learning. *J. Med. Syst.*, 44, 2, 10, 2019.

49. Nema, S., Dudhane, A., Murala, S., Naidu, S., RescueNet: An unpaired GAN for brain tumor segmentation. *Biomed. Signal Process. Control*, 55, 101641, 2020.
50. Hafiz, A.M. and Bhat, G.M., A Survey of Deep Learning Techniques for Medical Diagnosis. *Adv. Intell. Syst. Comput.*, 161–170, 2019.
51. Siva Raja, P.M. and Rani, A.V., Brain tumor classification using a hybrid deep autoencoder with Bayesian fuzzy clustering-based segmentation approach. *Biocybernetics*, 27, 2020.

Part 3

MACHINE LEARNING AND HEALTHCARE APPLICATIONS

11

Artificial Intelligence and Machine Learning for Healthcare Solutions

Ashok Sharma[1]*, Parveen Singh[2] and Gowhar Dar[1]

[1]*School of Computer Science and Engineering, Lovely Professional University, Phagwara, India*
[2]*Government SPMR College of Commerce, Jammu, India*

Abstract

Trends of teaching and learning has changed its shape from offline teaching to online teaching as full mode and physical mode of teaching may become substitution of academic keeping in view the pandemic covid-19.

Data science has become part of parcel of our daily life and most of the technical apps we are using contains machine Learning algorithms and helps us in many ways.

With rising conditions, artificial intelligence will be the most prominent transformative technology and enabler for society in the present era. here is no uncertainty that AI and analogous frameworks are built to change global efficiency, working habits, and lifestyles and support healthcare, Pharma Industry and Transformation in diagnosis process, disease treatment and early identification of symptoms has been fuelled machine learning techniques and tools such as Generative Adversarial Networks (GAN), Deep Convolutional Networks, Deep Reinforcement Learning (DRL), Gradient-boosted-tree models (GBM), etc. MRI and other sophisticated imaging systems immensely used for neural disorders, cancer diagnostics. In this chapter we are discussing various resources of medical datasets which can be used for diagnosis of dementia with the usage of machine learning approaches. We are presenting how various machine learning approaches can be useful in early diagnosis of many diseases and explained where machine learning and deep learning can be used on electronically stored medical data.

**Corresponding author*: drashoksharma@hotmail.co.in

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (283–292) © 2021 Scrivener Publishing LLC

Recent developments are achieved in what way machine learning can be applicable in multi-disciplinary research areas. The main emphasis of this chapter is to elaborate on the applicability of machine learning in the domain of healthcare. In the past, there had been substantial signs of progress in the way where machine learning can apply in innumerable research and industries. This chapter deliberates the prospect of using machine learning technologies in the healthcare sector and sketches several industry ingenuities implementing machine learning initiatives.

Keywords: Deep learning, machine learning, ADNI, dementia

11.1 Introduction

Healthcare is among fast-growing sectors and is now at the focus of a comprehensive global revamp and Transition. Public healthcare costs projected at approximately 6 to $7 trillion predicted to hit additional $12 trillion, merely in seven years [1]. The cumulative expenditure in USA on healthcare rose to 5.3% and lead to a high cost of $3 trillion nationally [2]. Besides, the cost of healthcare throughout the USA accounts for 17% of overall gross domestic product (GDP). Moreover, healthcare expenditure in the United States accounts for 17% of the global gross domestic product (GDP), our healthcare expenses increase at rates nearly twice our economic growth. Besides an increase in the amount the customers spend on healthcare, the federal government pays a considerable amount for healthcare as charges become unaffordable to patients. In 2014, the federal government's amount of money for healthcare expenses bounce by 11.7% to an unprecedented $844 billion in 2015 [3]. This surge in federal support reflects a substantial gap between healthcare costs and customer financial burden [4, 5]. Despite this dramatic rise in prices, a range of acts is taken to ensure that healthcare expenses are not taken any further Winding out of sight. Timetable follow-up, patient–physician condition communication, and availability specialists were apparent, too. Our healthcare system has a lot of drawbacks, and to tackle this problem is innovation. These novelties like swallowable microchips aware doctors when medicine had been taken to great measure data scrutiny to settle which medicines are top operative.

11.2 Using Machine Learning Approaches for Different Purposes

Machine Learning gives healthcare providers a unique capability to unify patient care proficiently, systematize contracts and payments, and speed up

diagnosis and treatment development procedures. Data are growing across sectors in today's connected world in exponential terms. As data volumes rise, new innovative ways of communicating with and extracting value from data emerge. Human interference was used in the past to analyse the information; however, this is cumbersome, and a significant amount of unidentified structures contained inside the data are not found [6], however it is very vital for healthcare area from external point of view. Machine learning is an emerging data analytics tool which Automates model construction, as it extends to the design of models. the ability of machine learning to employ algorithms, we can extract unknown insights from the data. The recursive essence of machine learning enables machines to acclimate their approaches and results as they expose to new circumstances and data [7]. The two basic methods to be followed while going deeper into machine learning are supervised and unsupervised learning. Training algorithms cover the field of supervised learning using different instances. The computer is having a range of inputs with the right number of outputs; Learning compares empirical findings with proper efficiency exception identifier [8]. Mostly supervised learning is applied to forecast future events by using previous data [9].

Unsupervised learning is applied to derive presumptions from datasets where input data is unlabelled. Cluster analysis is one of the most common form of unsupervised learning, which is used to discover hidden insights or to aggregate data for data exploration [6].

Healthcare is specific area where machine learning has widespread social ramifications. For a Series Growing smart watch industry, fit-up bits and gadgets Constantly gathering a whole host of health data, the frequency of implementing machine learning to examine this data is gaining incitation [10]. Machine learning should determine the solution to both reduce the rising health-care costs and help Build better relationship between patient and doctor. The Big Data and Machine learning applications can be used for a multitude of uses related to health. These include helping physicians identify more tailored medications and therapies for patients, and thus helps patients decide whether and when follow-up appointments will be scheduled. There's plenty of exciting possibilities for machine learning to boost health care quality. Models segregating the patients in various risk groups have management strategies to educate huge potential health-care effect [11]. This will become increasingly important to recognize the critical patients and high cost to undertake tailored intervention as healthcare providers take over the financial burden to handle their patients [12]. The tactics to machine learning strategies are applied to characterize and anticipate a wide range of health threats. Latest study used Logistic regression

to classify patients with misdiagnosed peripheral artery disease and to forecast their death rate, found that such a significantly achieves a more straightforward step-by-step logistical regression in terms of precision, configuration and net reclassification [13]. Machine learning was also applied to management of hospitals and services, to simplify processes and enhance patient satisfaction. For instance, Models are constructed for forecasting emergency rooms demand beds [14], and number of cases for elective surgery to notify decisions on hospital staffing.

Ultimately, using machine learning isn't a perfect solution, and not anything predicted will be actionable. For instance, we can forecast precisely headway to stage 4 from stage 3 of chronic kidney failure. In the dearth of any efficient care options—in addition to dialysis and kidney transplantation—prediction does not do anything to improve patient care.

11.3 Various Resources of Medical Dataset for Research

Machine learning is bursting in the field of healthcare sector. The different areas where machine learning has done revolution, Healthcare is at top priority. This all is possible because of higher processing power of machine learning algorithms. Different datasets specifically for healthcare research purposes are available. These datasets are divided into three different kinds:

I. Image Datasets
II. Genome Datasets
III. Hospital Datasets

Image datasets for medicine, healthcare and life sciences: Different image datasets are available for research purpose from different resources. Some important resources are as under:

I: ADNI: Alzheimer's Disease Neuroimaging initiative is a collaboration of medical centers and universities located in Canada and US. the sole purpose of ADNI is to develop biomarker procedures and imaging techniques in patients suffering from cognitive disorders [15]. The primary objective of ADNI is to develop effective methods that will adhere to unified norms for acquisition of functional and longitudinal PET data and multi-site MRI on AD patients, normal controls and mild cognitive impairments to build data repository which will define longitudinal changes in metabolism and brain structure while accumulating biochemical, parallel clinical and cognitive data to expound methods that will enhance the power to ascertain

the drug effects in clinical trials, and evaluate a set of biomarker and clinical based hypothesis.

II: OASIS: the open Access series of Imaging Studies (OASIS). The main goal of this project is to make neuroimaging dataset freely available to research community who are doing research on brain disorders. Central.xnat.org hosts OASIS datasets and facilitates open access platform for use in cognitive, neuroimaging and clinical studies on cognitive decline and normal aging with open access. The whole image data is available at www.oasis-brain.org.

III: CHDS: Datasets of Child Health and Development studies datasets are helpful to examine how disease and health transmit through generations. The dataset contained in CHDS are helpful to examine how cultural, social, and environmental factors and other genomic expressions are helpful in health and disease.

IV. Genome datasets: this kind of data repository contains genetic data for research purposes. Some common genome datasets are GEO Datasets, 1000 Genomes project and Genome in a Bottle. The 1000 Genomes dataset is a project with international collaboration and has setup the most detailed catalog of variation in human genes. It is comprised of more than 2,500 individuals from twenty-six distant populations across the globe.

V. Hospital datasets: datasets are also available from hospitals. This is primary data which further needs processing. MIMIC Critical Care Database is openly available dataset, which is designed by MIT lab for Computational physiology, consisting of data of more than 40 thousand critical care patients. Medications, biochemistry, demographics and much more are included in this dataset.

11.4 Deep Learning in Healthcare

Deep learning, Artificial Intelligence and neural networks are reframing patient care and medical expertise in the healthcare sector. Data analysis at unprecedented speed without conceding on precision in healthcare sector by applying deep learning algorithms is the breakthrough. The positive impacts of deep learning in healthcare sector are quick, accurate effective and abundant. Deep learning is a subclass of machine learning with multi-layered neural networks. The most commonly used deep learning networks are CNNs, which are used for image processing. However other deep neural networks used for processing text or audio are also present. Deep learning algorithms are applied for various purposes in medical sector like brain tumour segmentation on MRI images [16], lung disease

classification using CT [17], breast cancer detection using mammograms [18], and liver metastases with segmentation on CT [19], etc.

11.5 Various Projects in Medical Imaging and Diagnostics

With the rapid innovations and drastic advancements in healthcare and medicines since last few decades, the world is totally driven by data of different forms and decisions and inventions are the result of this data analytics.

Depending upon the data type whether it is text, images, audio, video, we need analytical and technological upgradation of existing systems and processes. As data collected can be in the form of readings, texts, speeches, images or videos, etc. so we have different version of AI approaches like Machine Learning, Deep Learning, etc.

Deep Learning has become one of the promising approach among the existing and it is nothing but an Artificial Neural Network with different layers to process the datasets.

With the variant of Neural Network like CNN, RNN, Autoencoders, etc., AI has potential in understanding datasets to find patterns and extracting useful attributes from a complex dataset. There are numerous Projects based on AI and Machine Learning which have been proven a new scope for medical research which includes:

PathAI's Technology utilizes machine learning to aid pathologists make faster and more precise diagnoses and recognizes patients who may benefit from novel types of therapies or treatments. As per records, this project has got funding of $15 million till 2018.

Microsoft's Project Inner Eye tools use machine learning based learning techniques to distinguish between specific tumors and healthy anatomy using 3D radiological image datasets to assist medical practitioners, radiotherapy healthcare experts, and improve decision-making and preparation, including medical diagnosis. However, this tool is very beneficial in identifying the exact and accurate location of the tumors, enabling more precisely targeted radiotherapy.

In examining the body immunity against cancer, R&D in Pfizer uses machine learning for immuno-oncology research with the assistance of IBM's Watson AI technology.

Insitro is another project based on machine learning and data science very helpful in development of drugs aiming towards providing cheap patient care.

Beta has developed an iLet platform that normalizes the level of blood sugar in type1 diabetes patients around the clock.

Enlitic developed a deep learning medical tool to simplify radiology diagnoses using a dataset of distinct kinds to evaluate unstructured clinical records (patient history, ECGs, radiology images, genomics, blood tests), which provides the best insight into the actual-time needs of a patient allowing health professionals to respond appropriately.

Freenome implements artificial intelligence in diagnostic tests, blood work, and screening to test for cancer. the sole purpose of freenome to use AI is to detect cancer as early as possible to discover new treatments.

Beth Israel Deaconess, a Medical Center at Harvard University, has designed an artificial intelligence-based model for early diagnosis of fatal blood diseases. AI-based microscopes are used by doctors to identify hurtful bacteria like staphylococcus in blood samples fast as compared to manual scanning. The drug discovery sector is embroiled by soaring production costs and research that happens to take thousands of human hours with the use of AI-based strategies and technology advancements; bio-pharmaceutical companies are efficiently noticing the knowledge, accuracy, and efficiency that AI can provide.

Atomwise implements AI to treat almost severe diseases, including multiple sclerosis and Ebola, by implementing the AtomNet neural network that helps to predict bioactivity and classify patient characteristics for clinical studies.

11.6 Conclusion

Machine learning methods implemented to EHR (electronic health records) data can yield attainable perceptions, from ameliorative upon patient menace slash systems, to predict the Illness initiation, to optimize hospital procedures. Modeling figures take advantage of the diversity and fullness of EHR data (as divergent to the use of EHR data). Modeling figures take advantage of the diversity and richness of EHR data (as opposed to the use of EHR data). Constrained range of features chosen by experts and/or frequently used) are still relatively rare and provide a promising route for further studies. Dynamic data types, such as wearable devices, put forth their own chances and challenges The complexities of using the machine learning approach effectively encompass the accessibility of workers with the expertise to create, test, and apply models, the evaluation of the real-world economic analysis of implanting a pattern into a system in health care.

References

1. Bouwens, J., *Embracing Change: The healthcare industry focuses on new growth drivers and leadership requirements*. Atlanta, GA, 2020. https://www.russellreynolds.com/sites/default/files/rr-gl-embracing-change_v4.1.pdf
2. Pianin, E., US Health Care Costs Surge to 17 Percent of GDP. *Financ. Times*. 2013. https://healthcare.mckinsey.com/big-data-revolution-us-healthcare/
3. Kayyali, B., Knott, D., Kuiken, S., *The Big Data Revolution in US Health Care: Accelerating value and innovation*, McKinsey & Company, New York, 2015.
4. Nambiar, R., Sethi, A., Bhardwaj, R., Vargheese, R., A look at challenges and opportunities of Big Data analytics in healthcare. *IEEE Big Data Conference*, 2013.
5. Bhardwaj, R., Adhiraaj Sethi, A., Nambiar, R., Big data in genomics: An overview. *IEE Big Data Conference*, 2014.
6. SAS Institute, *Machine Learning: What it is and what it matters*. 2020. www.sas.com/en_ca/insights/analytics/machine-learning.html
7. Wikipedia, *Machine Learning*. Australia, 2020. https://en.wikipedia.org/wiki/Machine_learning
8. Maddux, D., *The Human Condition in Structured and Unstructured Data*, Acumen Physician Solutions. 2014. https://acumenmd.com/blog/human-condition-structured-unstructured-data/#
9. Brownlee, J., *What is Machine Learning: A Tour of Authoritative Definitions and a Handy One-Liner You Can Use*, www. machinelearningmastery.com.
10. Dolley, S., *Big Data Solution to Harnessing Unstructured Data in Healthcare*. 2018. https://www.ncbi.nlm.nih.gov/pubmed/29594091
11. Parikh, R.B., Kakad, M., Bates, D.W., Integrating Predictive Analytics into High-Value Care: The Dawn of Precision Delivery. *JAMA*, 315, 7, 651–652, 2016.
12. Bates, D.W. *et al.*, Big data in health care: using analytics to identify and manage high-risk and high-cost patients. *Health Aff.*, 33, 7, 1123–1131, 2014.
13. Ross, E.G. *et al.*, The use of machine learning for the identification of peripheral artery disease and future mortality risk. *J. Vasc. Surg.*, 64, 5, 1515–1522, 2016.
14. Peck, J.S. *et al.*, Generalizability of a simple approach for predicting hospital admission from an emergency department. *Acad. Emerg. Med.*, 20, 11, 1156–1163, 2013.
15. Mueller, S.G., Weiner, M.W., Thal, L.J., Petersen, R.C., Jack, C., Jagust, W., Beckett, L., The Alzheimer's disease neuroimaging initiative. *Neuroimaging Clin.*, 15, 4, 869–877, 2005.
16. Havaei, M. *et al.*, Deep learning trends for focal brain pathology segmentation in MRI, in: *Machine Learning for Health Informatics*, pp. 125–148, Springer, Cham, 2016.
17. Anthimopoulos, M. *et al.*, Lung pattern classification for interstitial lung diseases using a deep convolutional neural network. *IEEE Trans. Med. Imaging*, 35, 5, 1207–1216, 2016.

18. Becker, A.S. *et al.*, Deep learning in mammography: Diagnostic accuracy of a multipurpose image analysis software in the detection of breast cancer. *Invest. Radiol.*, 52, 7, 434–440, 2017.
19. Drozdzal, M. *et al.*, Learning normalized inputs for iterative estimation in medical image segmentation. *Med. Image Anal.*, 44, 1–13, 2018.

12

Forecasting of Novel Corona Virus Disease (Covid-19) Using LSTM and XG Boosting Algorithms

V. Aakash[1], S. Sridevi[1], G. Ananthi[2]* and S. Rajaram[2]

[1]*Department of Information Technology, Thiagarajar College of Engineering, Madurai, India*
[2]*Department of Electronics and Communication Engineering, Thiagarajar College of Engineering, Madurai, India*

Abstract

The viruses are called as enteric viruses developed using ingestion termed as fecal oral transmission and is relicated using intestinal tract. Enteric viruses are genus Enterovirus phrased as Caliciviridae, Picoornaviridae, Coronaviride, Astroviridae, Orthoreovirus, genera Rotavirus, and Reoviridae phrase as Adenovirida and Reoviridae. Coronaviruses belong to the Coronaviridae family. It belongs to one of the Ribo-Nucleic Acid (RNA) families of the order Nidovirales, the others are pathegens of birds and insects of Arteriviridae and the Roniviridae families. The coronaviruses consists of single stranded RNA genome of 30kb in length in size. An epidemic of novel corona virus called as SARS-CoV-2 irritates the COVID-19 disease is reported recently. It is enveloped, plus stranded RNA viruses with extra ordinarily large genomes and helical nucleocapsids. During the pandemic situations, it is necessary to predict Covid cases in advance to take the preventive measures and thus saving the human life and other living beings. To predict the count of Covid-19 in advance and to improve the accuracy, this chapter proposes Machine and Deep learning algorithms such as Long Short Term Memory (LSTM), eXtreme Gradient Boost (XG Boost) algorithms and polynomial regression for forecasting. The real time dataset is taken from Kaggle which contains around 36,000 samples. The sample is taken from around 187 countries from the world and the dataset contains the details which included from the month of

**Corresponding author*: gananthi@tce.edu

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (293–312) © 2021 Scrivener Publishing LLC

January to May, 2020. The algorithm is tested using test dataset and the performance is evaluated through the performance metrics.

***Keywords*:** Forecasting, covid-19, LSTM, XG boosting, polynomial regression

12.1 Introduction

A corona virus belongs to the group of coronavirinae, nidovirales and coronavitdiae and having single-stranded RNA viruses. The generations of corona viruses are termed as alpha, beta, delta and gamma corona viruses [1]. It is observed that gene sources of microbat and rodents arise from alpha and beta corona viruses. Delta and gamma viruses' gene sources are from avian species. Corona viruses are crossed species barriers are very important in human pathogens. The corona viruses are Middle East respiratory syndrome corona virus and severe acute respiratory syndrome corona virus.

The above mentioned viruses are originated from bats, mammalian host and the dromedary camel and in turn infect the human beings. In literature, it is observed that two alpha corona viruses and four beta corona viruses infected human beings. The viruses are having upper and lower respiratory tract infections in immunocompetent and immunocompromised hosts [3]. The health issues are deranged liver, renal function tests, diarrhea and multiorgan dysfunction syndromes. Figure 12.1 shows the impact of Covid-19 across the world [11].

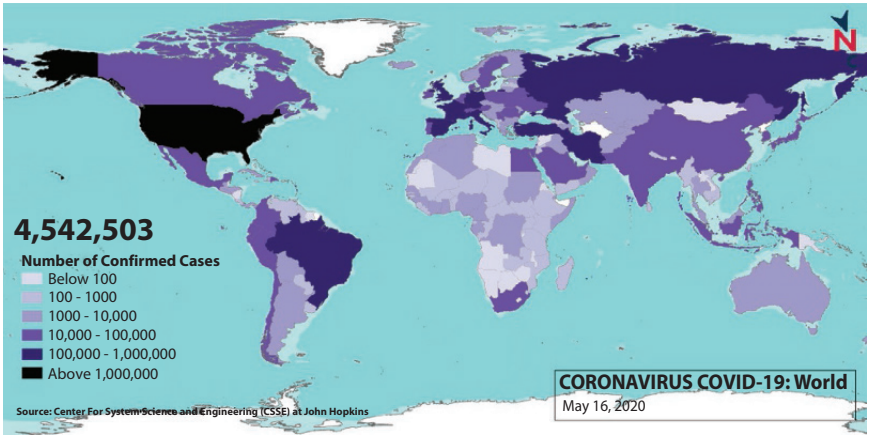


Figure 12.1 Total Confirmed Cases across the world on May, 2020. (Source : https://www.newsclick.in/covid-19-graphs-cases-recoveries-deaths).

The novel virus nature is not a predictable one due to its uncertainty based on the decisions on the vanishing the disease. Hence, short term forecasting is significant for the management of economical, public and societal and cultural health issues [6]. Intuitively, statistical methods are useful to forecast the corona virus. Initially, small time interruption active system works better forecasting analysis for the public data [6]. The use of generalized logistic model represents the growth in epidemic exponential in nature.

To forecast the Covid-19 cases, this chapter addresses the need and applications of forecasting algorithms. Figure 12.2 shows that confirmed Covid-19 cases in Tamilnadu. In the diagram, only highly affected cities (where 5,000 cases) like Thiruvallur, Chengalpattu, Vellore and Virudhunagar, etc. are considered for doing analysis.

In earlier days, various Machine learning algorithms were used for forecasting. Machine Learning (ML) is learning based on the data without being explicitly programmed. Learning from the labeled data is supervised and it has two categories: regression and classification. Predicting the discrete number of values comes under classification and predicting the continuous values comes under regression. Existing Machine Learning algorithms used for forecasting are as follows: Autoregressive Integrated Moving Average (ARIMA), Exponential smoothing, Artificial Intelligence (AI), Evolutionary Algorithms (EA), Fuzzy Logic Systems (FLS) and Support Vector Machines (SVM), etc. The above algorithms provide a high fault grade in prediction. To reduce the fault grade, this work uses LSTM (Long Short Term Memory), XG Boost (eXtreme Gradient Boost) algorithms and polynomial regression for forecasting.

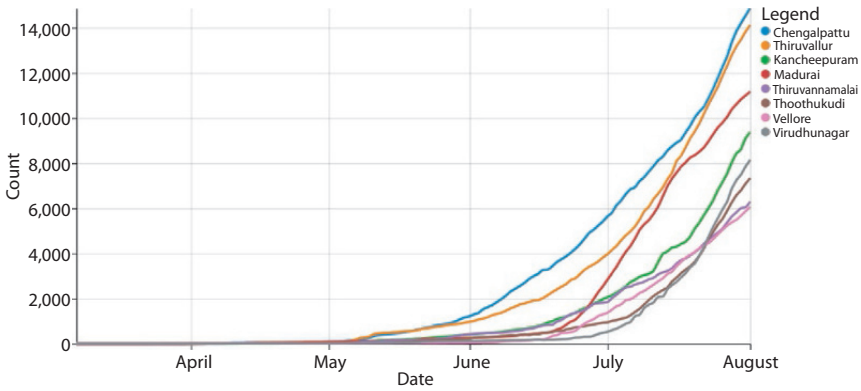


Figure 12.2 Confirmed Covid-19 cases in Tamilnadu—Highly affected cities. (Source: https://en.wikipedia.org/wiki/Statistics_of_COVID-19_pandemic_in_Tamil_Nadu).

Recently, forecasting has been implemented in several applications deals with numerical data or spatio temporal data. The predictions have been implemented using short term, medium term and long term basis. The forecasting has been done in short term predicts the result within a period of fewer than three months. Medium term forecasting predicts the results up to one year. Long term forecasting predicts the results more than a year. This chapter addresses only on short term forecasting only [2].

The objective of this chapter is to predict the corona virus using the proposed algorithm. The chapter is structured below. Section 12.2 describes the various machine learning algorithms for forecasting. Section 12.3 explains the proposed algorithms. Section 12.4 discusses implementation. Results and discussion of the proposed algorithm is explained in Section 12.5. Conclusion and future work is explained in Section 12.6.

12.2 Machine Learning Algorithms for Forecasting

Machine Learning algorithms can be used for association rule mining, classification, prediction and clustering. Various supervised and unsupervised algorithms are used for the above task.

The machine learning algorithm phases are business kind, data kinds, preparing data, modeling and evaluation. Data kind is the analyzing of individual attributes and their relationship with other attributes. By applying appropriate data pre-processing techniques such as data transformation, normalization, dimensionality reduction, etc. data is to be prepared for further process. Modeling includes supervised or unsupervised models and also includes either simple or hybrid models [4]. Hybrid models include combination of one or more models. Using testing dataset, the proposed model is evaluated through performance metrics. Classification is tested for the following metrics such as an accuracy, sensitivity, specificity, precision and recall etc. Prediction model is tested using the metrics like Root Mean-Squared Error, Mean-Squared Error and Mean Error Relative, etc. The applications are Stock Market prediction, Weather Prediction, Health care, Pattern Recognition, Electricity Demand Prediction, Forest fire prevention, Traffic pattern analysis, Market Basket Analysis and Yield prediction, etc. Various Machine learning algorithms for forecasting are discussed as follows:

Exponential Smoothing

It is the traditional method used for many predictions for demand forecasting and weather forecasting. The required data is forecasted based on the training data [7].

Naive Method
This method forecasts the imminent value based on the previous period. It assigns the future value as the previous period value short of doing calculations. Hence, this method provides large errors.

Auto Regression (AR)
The autocorrelation method is to forecast the next time step based on the previous input time steps observations. It is the type of time series technique uses regression equation. It is useful on a range of time series problems to predict accurate forecasting. The significance of this model is to find the series of data with respect to time for autocorrelation. The autocorrelation models help to make predictions.

Moving Average (MA)
The Moving Average method consists of average number of previous data points in a time series.

Autoregressive Moving Average (ARMA)
This statistical method uses moment-based estimation method is used to estimate the parameters. It is obviously known that the corona virus time series data non stationary sequences. It is to be converted into stationary time series date.

Autoregressive Integrated Moving Average (ARIMA)
The stationary process consists of constant mean and short term random time pattern in an algebraic manner. The autocorrelation and its power spectrum are constant with respect to time. The use of filter in ARIMA model is to separate the signal component from noise component. The extracted signal component is extrapolated to obtain the forecasts.

Seasonal Autoregressive Integrated Moving-Average (SARIMA)
This method consists of univariate time series data of seasonal component.

The seasonal components are

- Autoregression
- Differencing
- Moving average
- Parameter for the seasonal period.

Genetic Algorithm
This algorithm is a random search algorithm based on historical data to search the space. These algorithms pretend the natural selection process

means to adapt to environment changes and in turn reproduce the next generation. These algorithm uses chromosome behaviours of the population and its genetic structure. The details are as follows:

- Population in Individual completes for resources
- Fittest individuals produce larger issues than existing methods
- The fittest parent propagate gene to the entire generation and hence the parents propagate issues.
- The consecutive generation is suitable for the situation.

The parameter fitness score shows the individual ability for completion. The types of operators are selection operator, cross over operator and mutation operator.

Regression
Regression analysis is an algebraic tool to relate two variables. The first is the variable called predictor variable, the rate is collected from experiments made. The second is the variable called as the response variable, the rate is resulting from the predictor variable.

Multiple Regressions
Using previous data [10], this model calculates the regression coefficients using weighted least squares estimations method to predict the future data.

Support Vector Machine
The two cases of SVM include linearly separable and non-linear separable case. It depends upon the dataset. Overfitting, Local minimum problems and high dimensional sample data are the major challenges in neural networks. These problems are eliminated in support vector machines [8].

Random Forest (RF)
It is applicable for classification and regression. The advantage of using random forest algorithm is it that it reduces overfitting problems by using ensemble averaging and single decision trees. The algorithm is explained as follows:

- Select the random samples from the dataset.
- Construct decision tree in each and every sample.
- Get the voting from predicted result.
- Select the maximum voted prediction result.

Ensemble Methods
The significance of ensemble method is to group weedy beginners to method a husky beginner used to increase the precision. In literature, this algorithm provides higher rate than regression tree, classification and Naïve Bayes algorithms [9].

Fuzzy Approach
Fuzzy Clustering Means (FCM) algorithm is useful to the fuzzy time series for forecasting. It has three steps:

- Fuzzification
- Determining fuzzy relationship
- Defuzzification.

Fuzzification is useful for calculating the cluster centres. The rules are well defined between the relationships. Using these relationships the forecasting was done using defuzzified rules [12, 13].

Gradient Boosting Algorithm
Gradient boosting algorithm helps to solve regression and classification issues. This is sequential ensemble learning algorithm which uses several iterations. The absolute differentiable loss function optimization framework has been modeled. The parameters used in gradient boosting algorithm are loss function, weak learner and additive model.

Deep Learning Algorithms
The issues in Machine Learning are: i) Difficult to work high dimensional data or extremely large sets of inputs and outputs. ii) Difficult to do feature extraction. Deep learning is the class of machine learning algorithms in the form of a neural network that uses a cascade of layers (tiers) of processing units to extract features from data and make predictive guesses about new data.

Types of Basic Deep Learning Algorithms

- Convolutional Neural Network
  - used in computer vision and image recognition.
- Recurrent Neural Network
  - used in time series issues and forecasting.

Convolutional Neural Network
This algorithm consists of convolution layer, Rectified Linear unit layer, pooling layer, flattening layer and full connection layer.

Recurrent Neural Network
The advantage of this network has a memory and hence the all the information is being calculated. Hidden states are useful to maintain the memory.

Hyper Parameter Tuning in Deep Learning Algorithms
The parameters called as hyper parameters are variables determine the amount of hidden units in a network structure. The variables also determine the network training called as rate of learning. To improve the accuracy and considering time complexity into mind, this chapter uses deep learning algorithms and polynomial regression for forecasting

12.3 Proposed Method

The objective of the work is to forecast the number of cases, and number of fatalities using LSTM (Long Short Term Memory), XG Boost (eXtreme Gradient Boost) algorithms and polynomial regression for forecasting. The proposed model has two modules. The first module involves implementation of various machine learning algorithms. The second module compares the performance of all the models using performance metrics RMSLE (Root Mean Squared Logarithmic Error). Figure 12.3 shows that the proposed model for Covid-19 forecasting.

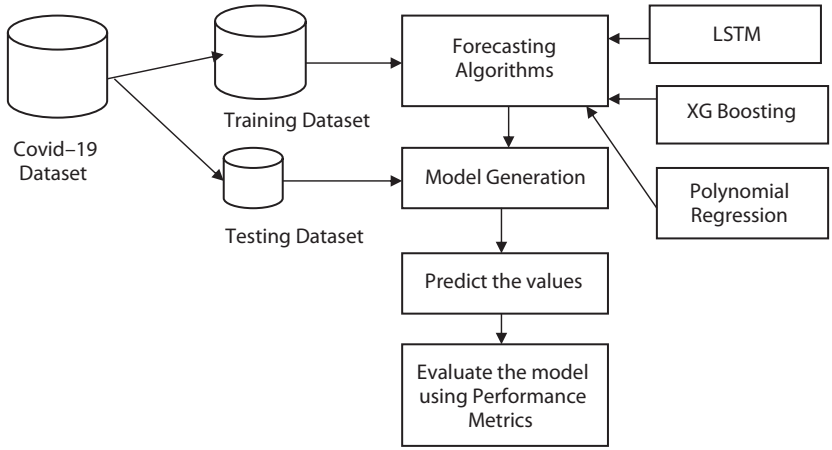


Figure 12.3 Proposed model for forecasting covid-19.

To improve the accuracy, ML algorithms used data pre-processing steps such as missing value estimation and data transformation before applying the models. If deep learning algorithm is employed, then the algorithm itself is doing data pre-processing, modeling and generating the output. To handle the sequence data, earlier days RNN (Recurrent Neural Network) is used. It takes the input and applies multifaceted calculations and using random variables called weights and biases. To overcome this gradient vanishing problem in RNN, Long Short-Term Memory (LSTM) is introduced in this work. LSTM is an advanced RNN variant; it can capture long-term dependencies and nonlinear characteristics of temporal or time series data.

In this work, the author used popular time series deep learning algorithm called Longest Short-Term Memory (LSTM) and gradient boosting algorithm called XG boosting algorithm for sequence prediction. To compare their performance and to improve the accuracy further polynomial regression is also employed in this work.

12.3.1 LSTM (Longest Short-Term Memory)

Long Short Term Memory uses temporal sequences and extended series of dependencies. This network has the type of a chain of repeating modules. This recurring module is a simple tanh structure.

RNN's uses previous layer's activation in addition to the current input and hence can handle sequential data. But due to vanishing gradients and exploding gradients it cannot memorize the long sequences and thus to overcome these issues LSTM and GRU were introduced. Figure 12.4

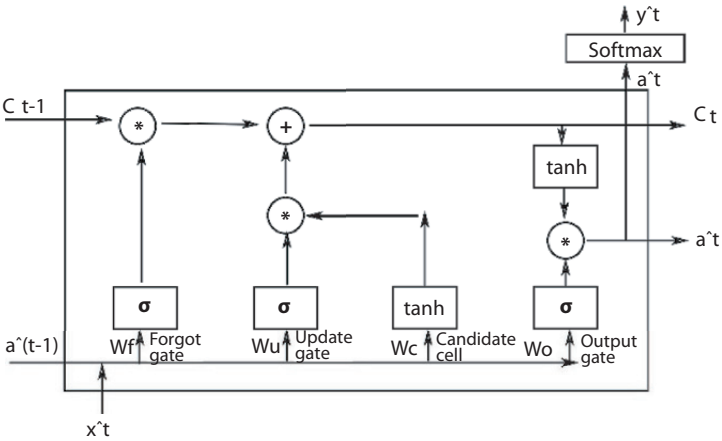


Figure 12.4 Working of LSTM.

explains the working of LSTM architecture. LSTM network consists of 4 parts namely Forget gate, Update gate, memory cell and output gate. These parts functionality is based on the following equations.

Let us assume that

d^{t-1} = previous time stamp activation
X^t = current time stamp input
W = weight matrix
b = bias term
d^t = current time stamp activation
C_t = Memory Cell
C_{t-1} = previous time stamp cell state
C_a = Candidate cell

$$C_a = \tanh\,(W_c[d^{(t-1)}, X^t] + b_c) \tag{12.1}$$

$$F_u = \sigma\,(W_u[d^{(t-1)}, X^t] + b_c) \tag{12.2}$$

$$F_f = \sigma\,(W_f[d^{(t-1)}, X^t] + b_f) \tag{12.3}$$

$$F_o = \sigma\,(W_o[d^{(t-1)}, X^t] + b_o) \tag{12.4}$$

$$C_t = F_u{*}C_{a+}\,F_f{*}C_{t-1} \tag{12.5}$$

$$d^t = F_o * C_t \tag{12.6}$$

Forget gate eliminates the information from the earlier activation. Update gate changes the information on the entrant cell. Output of each gate is then added and the memory state is obtained. The information to be sent out from the memory state is determined by yield gate.

The following parameters are considered in the proposed architecture to get the better accuracy:

- Input data Size, rate of learning
- Count of Hidden Layers and batch Normalization
- Size of the Batch and Activation Function
- Loss Function, Optimizer
- Filters Size (Input to state), Number of Filters (State to State)
- Kernel Size (Input to state), Kernel Size (State to State)
- Drop out, Regularizer, Feature Scaling.

12.3.2 XG Boost (eXtreme Gradient Boosting) Algorithm

XG Boost optimizes the loss function of the previous learner. Boosting is the process of converting the feeble learners in to strong learners. Initially a base model is used and the residuals are calculated from the actual values and the predicted values, then again next decision tree is built as x as features and residual r1 of base model as y and the predictions of current model are made as next residual. Now next decision tree is built as x as features and residual r2 of previous model as y. Now F(x) is calculated as the sum of the outputs of the entire model. Figure 12.5 shows that the working of extreme Gradient Boosting algorithm. The residuals after F0(x) to make h1(x) .h1(x) acquires from the residuals of F0(x) and defeats it in F1(x). h1(x) is a regression tree reduces the residuals from the previous step. F0(x) is to be defined initially.

$$F(X) = h_0(x) + a_1h_1(x) + a_2h_2(x)+.... \tag{12.7}$$

The XG Boost is quicker and accurate than the gradient boosting algorithm and it is parallelizable. Here, learning can be distributed across several CPU's also it can handle the missing values. It also uses L1 and L2 regularization to control over-fitting as well as built-in cross validation to give a better performance.

Unique features of XG Boost are: Handling sparse data, Regularization, Weighted quantile sketch, Cache awareness, Block structure for parallel learning and Out-of-core computing.

12.3.3 Polynomial Regression

This method is the association among the autonomous and dependent variables. Among various regression types, regression in linear manner is

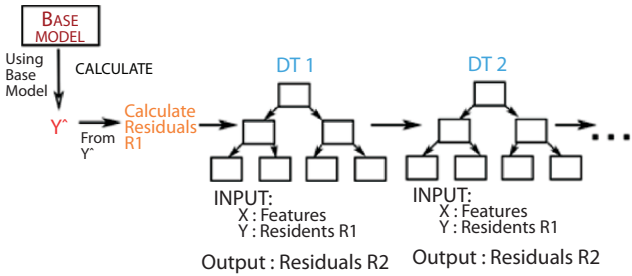


Figure 12.5 Working of the gradient boosting algorithm.

the most basic and commonly used model for predicting the continuously varying data. It is finding the line that fits best with the input and predicting the new value based on the hypothesis line. It works based on the equation

$$y = \theta_1 + \theta_2 * X + \theta_3 * X^2 + ... \tag{12.8}$$

(θ_1 = intercept, θ_2 = coefficient of X; X is the input variables). It can fit with the linearly varying data and can make predictions from the hypothesis.

12.3.4 Performance Metrics

The output of the test dataset is evaluated through the following metrics.

$$\text{RMSLE (Root mean squared logarithmic error)} = \sqrt{\frac{1}{n}\sum (Log(Y\hat{} + 1) - Log(Y + 1))^2} \tag{12.9}$$

Y is the real value and Y^ is the forecast value.

12.4 Implementation

The algorithms have implemented using Python and the system uses Intelcore 2 duo processor, Windows 2010 with 4GB of main memory. The python code for the proposed algorithm is given in this section.

12.4.1 The Main Python Code for LSTM

```
" generator = TimeseriesGenerator (new_train, length=length, batch_size=batch)\n",
" model = Sequential()\n",
" model.add(LSTM(100,input_shape=(length,new_train.shape[1])))\n",
" model.add(Dense(new_train.shape[1]))\n",
" model.compile(optimizer='adam',loss='mse')\n",
" model.fit_generator(generator,epochs=30)\n",
" first_eval_batch1 = new_train[-length:]\n",
" first_eval_batch1 = first_eval_batch1.reshape((1, length, new_train.shape[1]))\n",
" n_features = new_train.shape[1]\n",
```

```
"  test_predictions = \n",
"  first_eval_batch1 = new_train[-length:]\n",
"  current_batch = first_eval_batch1.reshape((1, length, n_features)) \n",
"  for i in range(30):\n",
"     current_pred = model.predict(current_batch)[0]\n",
"     test_predictions.append(current_pred) \n",
"     current_batch = np.append(current_batch[:,1:,:],[[current_pred]],ax-
is=1)\n",
"  pred = scaler.inverse_transform(test_predictions)\n",
"  cases = [i[0] for i in pred]\n",
"  death = [i[1] for i in pred]\n",
"   new = pd.DataFrame(cases,death,columns=['Fatalities']).reset_index()
\n",
"  new.columns = ['Fatalities','ConfirmedCases']\n",
    "   Res = pd.concat([Res,new],axis=0)\n",
"   Res = Res.reset_index(drop=True)\n"
```

12.4.2 The Main Python Code for Polynomial Regression

```
"for i in tqdm(locations):\n",
"  df = data[data['Location']==i].reset_index().drop(['index'],axis=1).
reset_index() \n",
"   X_c = poly.fit_transform(np.array(df['index']).reshape(-1,1))\n",
"   df[[0,1,2,3,4]] = pd.DataFrame(X_c)\n",
    "                train = df[df['Date']<='2020-04-14']
[['ForecastId','ConfirmedCases','Fatalities',0,1,2,3,4]]\n",
    "                test = df[df['Date']>='2020-04-15']
[['ForecastId','ConfirmedCases','Fatalities',0,1,2,3,4]]\n",
"  model = LinearRegression()\n",
"model.fit(train.drop(['ConfirmedCases','ForecastId','Fatalities'],axis=1),
train['ConfirmedCases'])\n",
"test['ConfirmedCases']=model.predict(test.drop(['ConfirmedCases',
'ForecastId','Fatalities'],axis=1))\n",
"  model = LinearRegression()\n",
"model.fit(train.drop(['ConfirmedCases','ForecastId','Fatalities'],ax-
is=1),train['Fatalities'])\n",
"test['Fatalities']=model.predict(test.drop(['ConfirmedCases','ForecastId',
'Fatalities'],axis=1))\n",
"          pred = pd.concat([train[-13:][['ForecastId','ConfirmedCases',
'Fatalities']],test[['ForecastId','ConfirmedCases','Fatalities']]])\n",
"  Res = pd.concat([Res,pred],axis=0)"
```

12.4.3 The Main Python Code for XG Boosting Algorithm

```
"source": [
  "x_train = data[data['Date']<='2020-04-14']\n",
  "#x_train.sort_values(['Day','Location'],inplace=True)\n",
  "y_train = x_train['Target']\n",
  "x_val = data[(data['Date']>='2020-04-02')&(data['Date']<='2020-04-14')]\n",
  "#x_val.sort_values(['Day','Location'],inplace=True)\n",
  "y_val = x_val['Target']\n",
  "test_ = data[data['part']=='test']\n",
  "#test_.sort_values(['Day','Location'],inplace=True)"
  ]
"source": [
  "train_set = DMatrix(x_train[features],y_train)\n",
  "val_set = DMatrix(x_val[features],y_val)\n",
  "model = train(params,train_set,num_boost_round=100,evals=[(val_set,'validation')],verbose_eval=50)\n",
  "ctr = 1\n",
  "for j in days:\n",
  "    test_set = DMatrix(test_[test_['Day']==j][features])\n",
  "    test_.loc[test_['Day']==j,'Target'] = model.predict(test_set)\n",
  "    test_ = feature(test_,ctr)\n",
  "    ctr+=1"    ]
```

12.4.4 Libraries or Methods Used in the Proposed Work

```
numpy
matplotlib.pyplot
pandas
xgboost
math
seaborn
load_boston
sklearn.metrics
mean_squared_error
from sklearn.model_selection
train_test_split
TimeseriesGenerator
Sequential
LSTM
```

12.5 Results and Discussion

The real time dataset—Covid-19 dataset is used in this work is taken from "Kaggle". The dataset contains the global data which involves around 184 countries. The dataset contains five fields: country, province state, date, established cases and mortalities. It has 36,000 samples by considering the period from January, 2020 to May, 2020. The algorithm is used to forecast the established cases and mortalities of corona virus. Training dataset is considered from the period January to April in order to train the model. Then the mode is tested using test data by considering the data of May, 2020. The task of the challenge is to predict the corona spread and the fatalities for the next 30 days. The sample dataset is shown in Table 12.1.

The Covid-19 is growing in an exponential rate and is strictly non-linear. Polynomial features include x^2 x^3 x^4 for the original value x. By using these polynomial features linear regression can handle the non-linear data. Figures 12.6 and 12.7 show that how linear regression fits the linear data and how polynomial regression fits the non-linear data.

Table 12.1 Sample dataset.

Province_State	Country_Region	Date	Confirmed Cases	Fatalities
2	Afghanistan	10-04-2020	521	15
2	Afghanistan	11-04-2020	555	18
2	Afghanistan	12-04-2020	607	18
2	Afghanistan	13-04-2020	665	21
2	Afghanistan	14-04-2020	714	23
2	Afghanistan	15-04-2020	784	25
2	Afghanistan	16-04-2020	840	30
2	Afghanistan	17-04-2020	906	30
2	Afghanistan	18-04-2020	933	30
2	Afghanistan	19-04-2020	996	33
2	Afghanistan	20-04-2020	1,026	36
2	Afghanistan	21-04-2020	1,092	36

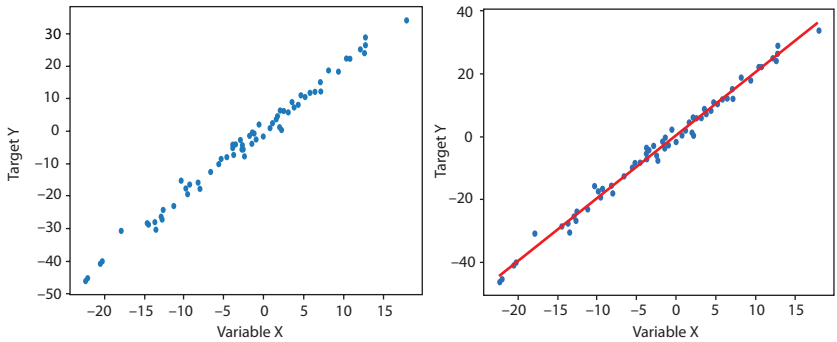


Figure 12.6 Output of linear regression for linear data.

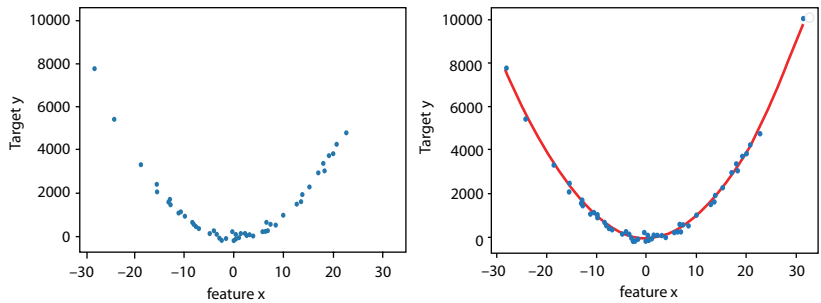


Figure 12.7 Working of polynomial regression with the non-linear data.

Basic pre-processing was applied in the time series data, as well as several time lagged features were created and the results were analysed. For analyzing the efficiency of the algorithms, Root Mean Squared Logarithmic Error (RMSLE) was used as a metric. Forecasting with XG Boosting is done with the entire data as well as country-wise. In LSTM and Polynomial Regression forecasting is done country-wise. Table 12.2 compares the performance of all the algorithms, out of which LSTM yield better results.

Table 12.2 Performance analysis of the proposed algorithms.

S. No.	Model	RMSLE
1.	Xg boost overall data	0.78360
2.	Xg Boost country-wise	0.93560
3.	LSTM country-wise	0.69394
4.	Polynomial Regression	0.85

The LSTM gives the best result on the test data with the RMSLE of 0.69394 for the forecasting of the next 30 days. Due to build in pre-processing step and parameter tuning, LSTM yields better accuracy than XG Boosting and polynomial regression. Figures 12.8 and 12.9 show that actual vs predicted values of Confirmed cases and fatalities using polynomial Regression.

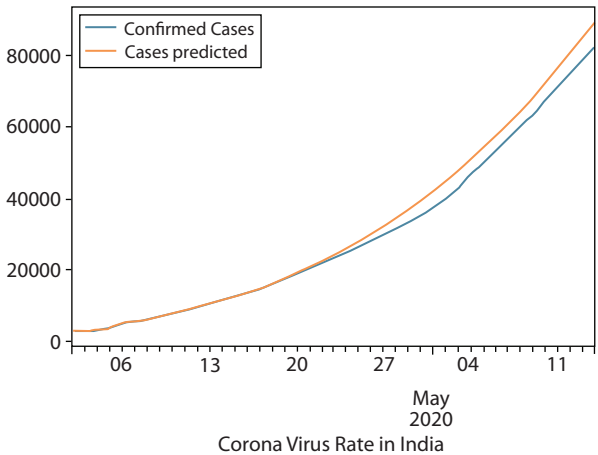


Figure 12.8 Actual and predicted values of Confirmed cases using polynomial Regression.

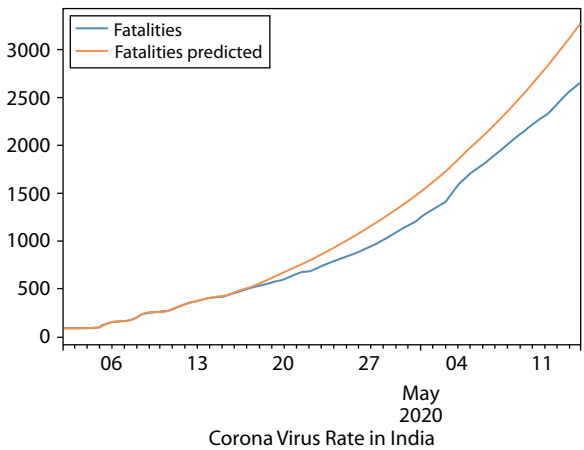


Figure 12.9 Actual and predicted values of fatalities using polynomial regression.

12.6 Conclusion and Future Work

The prediction of established cases and mortalities related to Covid-19 dataset was done using LSTM, XG Boosting and Polynomial Regression algorithms. To analyze the performance of the algorithms, the real time Covid-19 dataset is taken from Kaggle and its outcomes are measured. Among three algorithms, it is evident that LSTM works better than XG Boosting and Polynomial Regression algorithms due to its time series features and parameter tuning. Future work will focus on combining temporal features with spatial features i.e. use of ConvLSTM to provide the better accuracy.

References

1. Chan, J.F., To, K.K., Tse, H. *et al.*, Interspecies transmission and emergence of novel viruses: lessons from bats and birds. *Trends Microbiol.*, 21, 10, 544–555, 2013 Oct.
2. Chen, Y., Cheng, J., Jiang, X., Xu, X., *The reconstruction and prediction algorithm of the fractional TDD for the local outbreak of COVID-19*, arXiv, Cornell University, February 2020.
3. Cheng, V.C., Lau, S.K., Woo, P.C. *et al.*, Severe acute respiratory syndrome coronavirus as an agent of emerging and reemerging infection. *Clin. Microbiol. Rev.*, 20, 4, 660–694, 2007 Oct.
4. Mahalakshmi, G., Sridevi, S., Rajaram, S., A Survey on Forecasting of time series data, in: *IEEE International Conference on ICCTIDE'16*, 2016.
5. Peiris, J.S., Lai, S.T., Poon, L.L., Yuen, K.Y. *et al.*, SARS study group, Coronavirus as a possible cause of severe acute respiratory syndrome. *Lancet*, 361, 9366, 1319–1325, 2003 Apr. 19.
6. Petropoulos, F. and Makridakis, S., Forecasting the novel corona virus COVID-19. *PLos One*, 15, 3, 1–8, 2020, Braunstein LA, ed.
7. Sridevi, S., Parthasarathy, S., Rajaram, S., An Effective Prediction System for Time Series Data Using Pattern Matching Algorithms. *Int. J. Ind. Eng.: Theory Appl. Pract.*, 25, 2, 123–136, 2018.
8. Sridevi, S., Rajaram, S., Parthiban, C., SibiArasan, S., Swadhikar, C., Imputation for the analysis of missing values and prediction of time series data. *ICRTIT'11*, pp. 1158–1163, 2011.
9. Abirami, U. and Sridevi, S., Traffic flow prophecy with Map Reduce job for Big Data driven Approach. *Eighth International Conference on Advanced Computing (ICoAC'17)*, 2017.

10. Woo, P.C., Lau, S.K., Chu, C.M. *et al.*, Characterization and complete genome sequence of a novel coronavirus, coronavirus HKU1, from patients with pneumonia. *J. Virol.*, 79, 2, 884–895, 2005 Jan.
11. World Health Organization, Novel coronavirus, [Cited 2020 Jan 16], Available from: https://www.who.int/westernpacific/emergencies/novel-coronavirus.
12. Cheng, Y.-C. and Li, S.-T., Fuzzy Time Series Forecasting With a Probabilistic Smoothing Hidden Markov Model. *IEEE Trans. Fuzzy Syst.*, 20, 2, 291–304, 2012.
13. Yu, T.H.K. and Huarng, K.H., A bivariate fuzzy time series model to forecast the TAIEX. *Expert Syst. Appl.*, 34, 2945–2952, 2008.

13

An Innovative Machine Learning Approach to Diagnose Cancer at an Early Stage

Poongodi, P.[1]*, Udayakumar, E.[1], Srihari, K.[2] and Sachi Nandan Mohanty[3]

[1]*Department of ECE, KIT-Kalaignarkarunanidhi Institute of Technology, Coimbatore, India*
[2]*Department of CSE, SNS College of Engineering, Coimbatore, India*
[3]*Department of CSE, ICFAI foundation of Higher Education, Hyderabad, India*

Abstract

For high level application perceiving the low-level objects in an image has a great deal. Multiscale representation of the image can be used to detect the various cancers whereas the cancers appear in various ways based on their scale. Several cancer detection method and its applications are described in this chapter and they are compared fairly. The comparison defines about various cancers and its own strength and short coming. Automatic detection of cancers from data sets of images is a significant step in analyzing the huge scale of datas. Tumor positions in Magnetic resonance imaging or computerized tomography scan information, nuclear organization in refined colony and homogeneous sections in geophysical information can be represented by cancers. Before probing into cancer detection, initially few definitions are provided. Cancer is well-defined as a region associated by minimum of one limited extreme, maximum to minimum no. of local extreme, to represent bright or dark cancer. In the image intensity function, the saddle point limits the spatial level of a cancer. The saddle point is point at which the intensity breaks diminishing and jumps aggregating for optimistic cancers and for dim cancers.

***Keywords*:** Cancer, supervised learning, machine learning, SVM, deep learning, medical diagnosis, artificial neural networks

**Corresponding author*: poongodiravikumar@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (313–338) © 2021 Scrivener Publishing LLC

13.1 Introduction

The chapter provides complementary data around regions which is not possible to detect from corner or edge detectors. Cancer detection is used for identifying the regions which are to be processed further. These regions will indicate the occurrence of substances or portions which are to be recognized or to be tracked. Cancer descriptions are the fundamental for texture analysis and recognition. In present scenario, cancer descriptors have been found popular for wide reference point matching and to indicate the existence of information regarding the image structures. It presently the cancer detectors were deliberated in the gray-scale images rather than using these detectors in color images which leads to complications [1].

In general, the color image derivatives are calculated by calculating the channel to yield the last color gradient. The each and every color channel can have results of a color edge in the differing directions. Hence the summation of the derivatives per channel will eliminate the correlation between color channels. But the drawback of utilizing the cancer detector is that, it is based on the gray-scale where as there are two merits of utilizing the color vision. Color vision is an important discriminative property which allows differentiating between the textures. Second it provides additional information about physical causes for the variation in colors [2]. Thus, black object on the road can be identified as a shadow by the second merit.

In PC vision, mass recognition techniques are planned for distinguishing in a propelled picture that change in properties, for instance, magnificence or concealing, stood out from including regions. A mass is a territory of an image wherein a couple of properties are consistent or around steady, crucial classes of mass markers: (i) differential systems, which rely upon auxiliaries of the limit concerning position, and (ii) techniques reliant on neighborhood extrema, which rely upon finding the close by maxima and minima of the limit. With the later wording on the other hand intrigue locale administrators additionally intrigue point recognition and corner discovery [33].

There are several inspirations for considering and making mass identifiers. One key explanation is to give equivalent data about zones, which isn't picked up from edge pointers or corner identifiers. In early work in the zone, mass distinctive verification was utilized to get territories of vitality for extra preparing. These territories could hail the closeness of sections or parts of things in the picture space with application to battle insistence and also object following. In different spaces, for example, histogram assessment, mass descriptors can in like way be utilized for top territory with

application to division. Another fundamental use of mass descriptors is as basic local people for surface evaluation and surface insistence. In later work, mass descriptors have discovered ceaselessly standard use as intrigue habitats for wide check sound framework organizing and to hail the closeness of enlightening picture features for appearance-set up thing affirmation based as for close by picture estimations. There is also the related thought of edge acknowledgment to signal the proximity of delayed things.

There are a few inspirations why mass finders are examined and created. One primary explanation is to give corresponding data about locales, which can't be become from control locators or turn indicators. In initial works in this field, mass location was utilized to acquire districts of enthusiasm for additional handling. These locales could banner the proximity of chapters or parts with application to battle insistence and besides object following. In different locales, for example, histogram evaluation, mass descriptors can in like way be utilized for top affirmation with application to division. Another ordinary utilization of mass descriptors is the standard local people use for surface assessment and attestation. In later works, mass descriptors have been discovered intelligently remarkable use as intrigue communities for wide standard sound structure sorting out and to hail the vicinity of enlightening picture highlights for appearance-set up part attestation based with respect to near to picture encounters. There is likewise the related thought of edge unmistakable confirmation to flag the closeness of postponed parts. Be that as it may, the majority of mass indicators these days are concentrated in dark scale pictures while these finders don't function admirably in shading pictures with complex shading foundation.

The fundamental way to deal with processing shading picture subordinates is to figure independently the subsidiaries of the channels and add them to create the last shading inclination. Be that as it may, the subsidiaries of a shading edge can be in the contradicting bearings for the different shading channels. Subsequently, the summation of the subordinates per channel will dispose of the relationship between shading channels [35].

13.1.1 Multiscale Cancer Detection

Scale space hypothesis is a structure for multi-scale signal portrayal created by the PC vision, picture handling and sign preparing networks. It is a conventional hypothesis for taking care of picture structures at various scales so that fine-scale highlights can be progressively smothered and a scale parameter t can be related with each level in the scale-space portrayal. All in all, when a neighborhood picture structure is available in a specific

scope of scales, the focuses are then recognized at each scale inside this range. As an outcome, there are numerous focuses, which speak to a similar structure, yet the restriction and the size of the focuses is extraordinary [34]. Nonetheless, the pointlessly high number of focuses expands the likelihood of bungles and the multifaceted nature of coordinating and acknowledgment calculation. In this manner, techniques for dismissing the bogus matches and for confirming the outcomes are vital at further strides of the calculations.

Laplacian of the Gaussian. One of the first and furthermore most regular mass finders depends on the Laplacian of the Gaussian. The difference of Gaussians (DoG) approach. This administrator is in essence like the Laplacian and can be seen as an estimate of the Laplacian administrator. In improved structure, the determinant of Hessian registered from the Haar wavelets was utilized as a fundamental intrigue point administrator in the SURF descriptor for picture coordinating and object acknowledgment. Additionally, proposed a way to deal with mass location dependent on wavelet change modulus maxima.

The mixture Laplacian and determinant of the Hessian administrator (Hessian–Laplace). A half and half administrator between the Laplacian and the determinant of the Hessian mass indicators was likewise proposed, where spatial determination is finished by the determinant of the Hessian and scale choice is performed with the scale-standardized Laplacian. Relative adjusted differential mass locators [23]. The mass descriptors got from these mass finders with programmed scale determination are invariant to interpretations, turns and uniform rescaling's in the spatial space.

Dim level masses, dark level mass trees and scale-space masses. An area with spatial degree characterized from a watershed similarity was related with every [37] nearby most extreme, too a neighborhood differentiate characterized from a so-called delimiting saddle point. A neighborhood extremum with degree characterized thusly was alluded to as a gray-level mass. In addition, by continuing with the watershed relationship past the delimiting saddle point, a dim level mass tree was characterized to catch the settled topological structure of level sets in the force scene, in the way that is invariant to relative misshapenness in the picture space and monotone power changes. By concentrating how these structures develop with [21] expanding scales, the idea of scale-space masses was presented. Past nearby difference and degree, these scale-space masses additionally estimated how stable picture structures are in scale-space, by estimating their scale-space lifetime.

Maximally stable extremal regions (MSER) were keen on characterizing picture descriptors that are vigorous under point of view changes. They

contemplated level sets in the force scene and estimated how stable these were along the power measurement. In light of this thought, they de-fined an idea of maximally stable extremum districts and indicated how these picture descriptors can be utilized as picture highlights for sound system coordinating.

The fundamental downside of these mass identifiers depends on dim scale space. There are two significant favorable circumstances of utilizing shading vision. To start with, shading is a significant discriminative property of chapters, permitting us to recognize mustard and mayonnaise. Second, shading gives additional data which permits the differentiation among different physical foundations for shading varieties on the planet, for example, changes [38] because of shadows, light source reflections, and chapter reflectance varieties. This serves to rapidly recognize a dark chapter out and about as a shadow.

13.2 Related Work

Optical character acknowledgment frameworks have been successfully created for the acknowledgment of printed characters. Optical character acknowledgment is a great PC vision method with different applications running from sparing continuous contents carefully and inferring setting based knowledge utilizing regular language handling from the compositions. One such claim is the affirmation of machine reproduced fonts. This chapter demonstrates the framework to perceive machine printed characters using Cancer acknowledgment method and Image dealing with. When in doubt of such machine printed characters there is likeness between character concealing and establishment concealing. There is work up of reflected light and scattered light. Shading is steady across character region or foundation zone. Chapter clarifies how Cancer location method is utilized for acknowledgment of these machines printed characters.

A Cancer examination calculation is utilized to discover and tally objects, and to make essential estimation of their qualities. The motivation behind examination is to decide if the outcomes got from an activity are exact, sensible and valid. Picture preparing programming includes complex calculations that have pixel esteems as sources of info [22]. The present picture investigation programming bundles incorporate both old and new innovations. Most huge is the connection between the old mass examination strategy and the new age-identification method. For picture handling, a mass is characterized as a district of associated pixels. Mass investigation is the recognizable proof and investigation of these districts

in a picture. The calculation observes pixels by their worth and spots them in one of two classifications. Viz. closer view (normally a pixel has a non-zero worth) or the foundation (pixels with a zero value). If straightforward division is preposterous because of poor lighting or masses with a similar dim level as parts of the foundation, we should build up a division calculation proper to a specific picture [39].

Malignant growth ID strategies with incorporate strength close by the usage of KNN calculation are proposed inside this part. A scale-space tree is made dependent upon the lion's share that were delivered utilizing a development of pictures in the wake of obscuring. Highlights and spatial data give the movement in scale space tree improvement. After the system of mass extraction, clients pick the kind of masses that were seen inside the picture by seeing classes to make ground truth pictures. Inside a practically identical technique, k-Nearest Neighbor (KNN) calculation is applied to see classes of the picture's mass to show its showcase.

After the strategy of a scale space tree improvement, all masses are tended to as a middle and will be assessed for its significance. The mass lifetime is depicted as a discrete worth assessed from number of scales which the structures exist. The system of highlight ampleness in a scale space tree headway. As lifetime broadens, the mass acquires centrality. The lifetime can be depicted when a specific mass at a particular scale shows up or vanishes. After a scale-space tree is surrounded, it is investigated to [20] compute the lifetime of each mass. The dominant part with a most exceptional extent of lifetime in each branch will be picked as section applicants.

The limit for lifetime can in like way be applied and balanced in this applicant determination method. On the off chance that the mass remains longer at a specific limit, the mass will be picked as the object of intrigue. This give an outline of invariant intrigue point indicators, how they advanced after some time, how they work, and what their individual qualities and shortcomings are. We start with characterizing the properties of the perfect nearby element finder [9]. This is trailed by a review of the writing in the course of recent decades sorted out in various classifications of highlight extraction techniques. We at that point give a progressively definite investigation of a determination of strategies which had an especially noteworthy effect on the examination field. We finish up with an outline and promising future research bearings.

The confinements of the worldwide highlights are to portion the picture in a set number of areas or sections, with each such locale comparing to a solitary item or part thereof. The most popular case of this methodology is the mass world framework, proposed in, which sections the picture dependent on shading and surface, at that point scans a database for pictures with comparative

picture masses. A model dependent on surface division is the wide standard coordinating work portrayed in Ref. [7]. An approach to manage the issues experienced with worldwide highlights or picture divisions is to thoroughly test various subparts of the picture at every area and scale. For each such picture subpart, worldwide highlights would then be able to be registered.

The fast improvement of clinical imaging innovation has drastically expanded the spatial and worldly goals, and accordingly size, of clinical imaging information. A few instances of chapters in pictures for clinical investigations remember cell cores for histopathology pictures or cerebral veins in attractive reverberation (MR) pictures to analyze vascular illness, cerebrum tumors [40] in MR pictures to survey treatment, bosom sores in ultrasound pictures to arrange bosom malignant growth injuries and glomeruli in kidney MR pictures to portray renal ailment, specifically noteworthy in this exploration is a typical sort of item that is little in structure and raised elliptic fit as a fiddle. In PC vision, the issue of identifying such chapters is recognized as mass recognition. Broad research has proposed [18] different mass indicators, among which one well known methodology is to utilize neighborhood extrema in a changed space related to a vector of inferred highlights (e.g., nearby power histogram or direction histogram) to recognize the masses.

All in all, this sort of mass finder can be arranged as an intrigue point locator or intrigue locale indicator. As intrigue point locators, the Radial-Symmetry and the Radial Gradient Transform detector outspread symmetric space as the change base. Be that as it may, for outspread topsy-turvy masses, these two indicators may lose their sensitivities. Different sorts of intrigue point locators like SIFT, BRISK and SURF, are established change. Related [22] with relative invariant features, for instance, the bearing histogram to depict the close. In this manner, these finders will in general be less lenient of nearby commotion.

Inspired by the relative invariant intrigue point indicator, intrigue area identifiers are acquainted with determine collected proportions of various pixels/voxels inside areas of nearby extrema. A few instances of intrigue district finders are the Harris-relative identifier, Hessian-relative indicator, and Hessian–Laplace locator. While these indicators are demonstrated to be increasingly powerful to tumor, they are computationally costly to adjust the shape estimation. Furthermore, the enormous number of neighborhood extrema requires cautious. It is likewise testing to distinguish the proper will in general be emotional intrigue district finders. Albeit both intrigue point and area finders well portray [15] relative nearby picture seen in the investigation the exhibition indicators mass ID issues obsessive pictures are unsuitable. Rather, another kind of locator, Laplacian of Gaussian (LoG) finder, under the scale space hypothesis, has pulled in incredible consideration for

the mass discovery issue. Scale space hypothesis is a proper hypothesis that thinks about one picture as a heap of pictures constrained.

This implies neighboring mass items will diffuse and in the long run consolidate to be identifiable masses at a particular scale. On the off chance that a multiscale portrayal concerning is built, exists one "ideal" scale for masses with comparative sizes. Singular masses would then be able to be distinguished with relating. By applying the LoG parts can effectively identify cancers doesn't increment, and the quantity of nearby maxima in a splendid mass doesn't diminish.

In any case, the symmetric idea of the LoG indicator restrains its exhibition in rotational awry mass identification expense. In this way, various LoG augmentations. The instance, the DoG is created to estimate the LoG and improve computational execution. Thus gLoG is in this manner ready to recognize general circular structures, for example, rotationally symmetric and uneven masses [16]. All indicators looked into beforehand utilize a similar work process, pinpointing the mass & cautiously evacuate the covering nearby execution. An ordinary circle with an appropriate sweep that is related. Therefore, the determined highlights are just gauges as opposed to precise estimations characterized by evident limits. Besides, while scale space hypothesis gives the establishment to complex finders, it depends on a multiscale portrayal, which may squander registering exertion in ideally choosing masses around. Consistently measured masses regular various clinical applications, for example, in recognizing cells and cores in pathologic or fluorescent minute pictures and fragmenting kidney glomerulus in 3-D MR pictures [25]. These pictures for the most part contain huge number of little masses, each with a curved circular shape.

13.3 Materials and Methods

The proposed method, named Gray-Level Co-Occurrence Matrix (GLCM) will be a novel method for small cancer detection. In the proposed method, GLCM helps to identify the Multiscale representation for separate image whereas the lobs are analogous in scope and the optimal rule can be recognized from the gray-level study which provides the only scale representation from images [3].

GLCM analysis first performs pre-segmentation process in the cancer candidates because the pre-segmentation will identify and differentiate all true cancers and non-cancer objects which lead to precise results stated the theoretical foundation of Hessian. The proposed block diagram is shown in Figure 13.1. Threshold based fine pruning routine was inked in

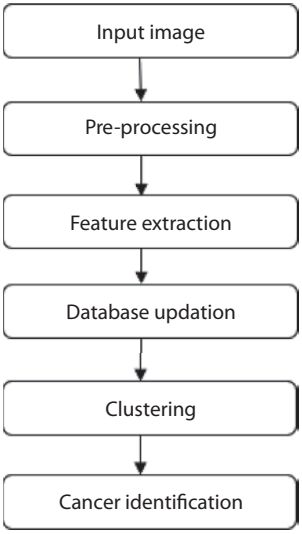


Figure 13.1 Block diagram.

detectors such as LoG, gLoG and Radial Symmetry but they are sensitive to noise. In the nick of time, K-means pre-segmentation reduces the data size by removing the non-cancer sections and visualizing the border of the lob candidates. As an outcome it is possible to excerpt several features from minor cancer candidates [11].

A quantifiable technique for examining surface that considers the spatial relationship of pixels is the dull level co-event framework (GLCM), regardless called the lessen level spatial reliance cross section. As far as possible delineate the outside of a picture by figuring out how routinely matches of pixel with unequivocal qualities and in a predefined spatial relationship happen in a picture, making a GLCM, and sometime later removing credible measures from this cross area. The surface channel limits, portrayed in Texture Analysis can't give data about shape, i.e., the spatial relationship of pixels in an image. After you make the GLCMs, you can get two or three estimations from them utilizing the work.

To prompt a GLCM, to utilize the decrease comatrix work. The dull comatrix work causes a to decrease level co-event arrange (GLCM) by figuring how sometimes a pixel with the force (reduce level) respect I happens in a particular spatial relationship to a pixel with the worth j. Typically, the spatial relationship is portrayed as the pixel of intrigue and the pixel to its expedient right (on a level plane adjoining), yet you can choose other spatial relationship between the two pixels. Each part (i,j) in the resultant glcm is on a very basic level the total of the occasions that the pixel with respect

I happened in the predestined spatial relationship to a pixel with respect j in the information picture [19].

The measure of lessen levels in the picture picks the size of the GLCM. According to typical system, diminish co-arrange utilizes scaling to decrease the measure of power respects in a picture to eight, in any case you can utilize the Num Levels and the Gray Limits parameters to control this scaling of lessen levels. See the diminish comatrix reference page for extra data. The dull level co-event structure can uncover certain properties about the spatial dispersing of the lessen levels in the surface picture. For instance, if most by a wide margin of the portions in the GLCM are locked in along the slanting, the surface is coarse concerning the fated balanced.

13.4 System Design

The skin malignant growth analysis where a few factors assume job in finding, it is generally hard to confine the contributions of the neural organize and to overlook the effect of certain elements for a superior system learning. Notwithstanding, knowledge quality may advance by lessening the quantity of information factors, a decision insignificant of significant info highlights is important. In any case, a bigger number of layers are worse than few, it is commonly realized that a system containing hardly any shrouded neurons sums up superior to one with numerous neurons. Likewise, the best system among various independently prepared systems is then chosen by experimentation.

Calculation of preparing fake neural system the learning calculation systems alluded as backpropagation calculation. This was better after some time and different sorts of backpropagation calculation were introduced for preparing the system. When utilizing neural systems, the best possible learning calculation is generally decided dependent on inner mental drugs and by means of experimentation techniques. The strategy that finds at least a capacity that is communicated aggregate of straight capacities. An exceptionally basic, however powerful, strategy for approximating a capacity. For the most part in its utilization of nonlinear hunt system. This makes the calculations diminished in a solitary age.

This examination is predominantly planned for giving execution investigation of these learning calculations for the instance of example acknowledgment for skin malignant growth finding. Melanoma is at present one of the most hazardous kinds of malignancy. In spite of the fact thatskin cancer outside area, the conclusion is frequently has late phases sickness improvement because generally clinical attendances. Simultaneously, in spite of the

high passing conclusion of cancer and auspicious treatment will deliver practically 100% endurance [30]. Threatening outwardly apparent can for the most part be recognized by specialists of all claims to fame during routine assessment. The most generally utilized analytic the essential shaped authorities utilizing perceptible & dermatoscopic photos by ground-breaking. A progression of different qualities permitting threatening neoplasms to be segregated from amiable utilizing just a picture [28].

The idea of utilizing PC vision to illuminate the undertaking of distinguishing skin malignant growths emerged moderately as of late. PC innovation comprises of a usual of techniques accommodating distinguishing proof and sorting of various articles. Logical examinations meaning to utilizing PC methods. Clinic as of now utilize exceptional dermatology frameworks giving advanced arrangements permitting diagnostics utilizing a dermatoscopic, camera and unique expert programming [10]. As a rule, such programmable frameworks give PC examination of pictures. The calculation discriminates among generous and dangerous skin injuries with precision of in any event 91% utilizing programmed investigation of pictures of pigmented skin developments.

The profoundly subject to understanding of the specialist. Examination and conclusion of cutaneous sores can be upgraded by applying PCs to play out the assignment. As of late a wide range of programming bundles have been created to help doctors. Some of them consider geometry, shading and surface [12], some utilization picture handling techniques. A few bundles consolidate both the dermoscopic pictures with the clinical history setting. The PC helped clinical conclusion is particularly important on the off chance that it both groups perceptions and gives a clarification to the choice too. This is why we have chosen to consider the static information investigation as well as likewise a corresponding methodology called information mining [14]. Three extra-normal comes nearer from the true learning frameworks have been picked, the Naive Bayes classifier, the Random Forest and the K case based classifier. The WEKA instrument use was applied. For rules explaining the clarified plan portrayal the Attributional Calculus executed in the AQ21 perform different assignments learning and data mining program was used.

Man-made reasoning (AI) is the execution of machines customized such that they think and act like human. Computerized reasoning gives a bit nearer to robotization in each area. It furnishes productive outcome with exactness [21] and without human mistake, making it one of the most significant headways in the coming innovation. Man-made reasoning depends on working of system of neurons in a human cerebrum, it is a fake reproduction of a neuron which is made to be figured out how to

perform errand of an organic neuron through different calculations and systems. The organic neural system collaborates through electrical driving forces which go about as a sign between the neuron to play out a particular assignment. The fake neural organize (ANN) works likewise, it utilizes a few fake neurons going from 10 to 10,000 which is a lot lesser than organic neurons in a cerebrum which is in billions. ANN are computational models which are able to do design acknowledgment and AI [20]. Counterfeit insight fills our heart with joy to-day life proficient while limiting human endeavors. Computer based intelligence is executed in different fields to facilitate our everyday schedules. Man-made intelligence innovation handles a few exercises in banks and money related frameworks [38]. It incorporates more intelligent visit bots for better and fast client administrations.

It oversees money related tasks, shrewd putting resources into stocks, overseeing properties and substantially more. Assembling organizations and enterprises additionally use AI, for the most part in the creation unit. They are utilized as robots to move an item starting with one spot then onto the next, give various shapes to an object. Organizations use AI for the board of representative records, recovery of explicit data. It gives time the board and gives precise outcome [36]. Artificial intelligence plays an imperative job in Air transport frameworks, most exercises acted in air transport incorporate AI advances. Various programming projects in light of AI give a superior flight understanding to the clients. Simulated intelligence calculations have given bots against whose player could play in gaming console. Simulated intelligence innovation has helped in headway of playing against a prevalent contender for better gaming experience.

Computer based intelligence assumes a fundamental job in the field of medication. Machines are fit for examining immense measures of information and distinguishing designs that people can't. Working of AI programs that complete determination and make suggestions is key worry of clinical man-made consciousness. A few clinical applications depend on completely numerical and probabilistic strategies, in contrast to this, programs created utilizing AI are based on representative models ofailment units and their relationship with persistent variables and clinical showings. Early, the imminent of such an innovation in human services and medication interested researchers and specialists [26]. Progressively, supercomputers and homegrown frameworks are being utilized by the specialists and medical clinics to recognize patients who may experience coronary illness, kidney disappointment or on the other hand postoperative diseases. Investigation of affiliation between avoidance or treatment techniques and results of persistent is the key target of man-made reasoning in medication (AIM). It was normal that shrewd PCs, with the capacity to store and procedure

expansive amount of information, would furnish help or exceed expectations specialists with various errands.

A wide range of PC systems are utilized for location and counteraction of this hazardous ailment. The best and effective route is to utilize diverse AI methods to recognize the event of malignancy at beginning periods. The idea of AI can be considered for creating models for finding of malignant growth. Capacity of a PC to learn without being modified is known as machine learning [27]. Different models can be utilized to learn and anticipate the event of malignancy from CT sweeps, and picture handling strategies would be useful for this situation. Models can likewise be prepared based on different components for example, hereditary legacy, tobacco and liquor utilization, stoutness, presentation to radiation or in any event. There are AI strategies accessible, however not all can be utilized for malignancy discovery.

Every single strategy has its own component which decides if it tends to be utilized for a particular explanation or not. There are a few driving fake strategies, for example, neural systems, hereditary calculations, fluffy grouping, bolster vector machines (SVMs), molecule swarm enhancement (PSO), choice tree classifier, Bayesian systems, straight relapse (LR) and PC supported discovery (CAD). These strategies decide the grouping or relapse approach and may likewise decide a complete procedure for disease discovery. Different strategies utilized for the location of lung malignant growth, liver disease what's more, bosom disease are assessed in this chapter. It is watched that k-NN, SVM, fluffy bunching and profound learning are hardly any methods which give the best outcome to the ID of malignant growth.

In any case, different kind injuries can imitate these malignant growths, bringing about pointless dreariness through intrusive biopsies and medicines. The utilization of man-made reasoning as an analytic help is a developing pattern in dermatology. These frameworks by and large use some type of AI including techniques empower forecasts dependent earlier information what's more, encounters. As opposed to regular models that are expressly customized to deal with an arrangement [24]. Be that as it may, the easiest methodology is advanced photography, frequently upgraded by a dermatoscopy.

The near all inclusive use of electronic cameras and dermatoscopes in dermatologic practice, propelled picture based ML models have the best potential for clinical execution and are thusly the point of convergence of this study. Past overviews of man-made cognizance and skin dangerous development have focused on melanoma [17]. Undoubtedly, the current examination addresses the principle effective review of robotized ID of NMSC using propelled picture assessment. The PRISMA plan is associated with Search philosophy of the databases while securing the method of reasoning of the

request string. Artificial intelligence terms were taken from course books on AI and address the most by and large used models. The interest to dismiss examinations tumors. To assurance 2 specialists expelled data, cross-affirmed at separately movement of the decision show. We have itemized recently the NMSC-express results in ponders that gathered both melanoma [18].

13.4.1 Artificial Neural Network

Neural networks/Neural systems are farsighted models inaccurately subject to the action of natural neurons. The assurance of the name—neural network was one of the exceptional PR achievements of the Twentieth Century. It irrefutably sounds more stimulating than a specific portrayal, for instance [13], an arrangement of weighted, included substance regards with nonlinear trade functions. In any case, regardless of the name, neural frameworks are far from speculation machines or fake brains. A common fake neural framework may have a hundred neurons. In assessment, the human tactile framework is acknowledged to have about 3 × 1,010 neurons. We are still light an extremely significant time-frame from Data.

Rosenblatt's model contained three layers, a retina|| that coursed commitments to the resulting layer, affiliation units|| that solidify the commitments with burdens and trigger an edge step work which feeds to the yield layer, the yield layer which joins the characteristics. Amazingly, the use of a phase work in the neurons mentioned the objective facts irksome or hard to plan [29]. A different essential deficiency of perceptron and for some time, eagerness for perceptron vanished. They proposed a multilayer neural framework with nonlinear yet differentiable trade works that kept up a key good ways from the snares of the principal perceptron's movement limits [31].

13.4.2 Back Propagation Network (BPN)

Back Propagation network (BPN) and General Regression Neural Networks (GRNN) have comparative designs, however there is a principal distinction: Probabilistic systems perform order where the objective variable is downright, though broad relapse neural systems perform relapse where the objective variable is persistent. In the event that we select a BPN/GRNN arrange, DTREG will naturally choose the right kind of system dependent on the kind of target variable [32].

The input layer is one neuron in the data layer for each pointer variable. Because of straight out variables, N-1 neurons are used where N is the amount of arrangements. The data neuron (or getting ready before the data layer) standardizes the extent of the characteristics by deducting the center

and detaching by the interquartile run. The data neurons by then feed the characteristics to all of the neurons in the covered layer.

The Hidden layer has one neuron for each case in the planning educational assortment. The neuron stores the estimations of the [41] marker factors for the case nearby the goal worth. At the point when given the x vector of data regards from the data layer, a covered neuron enlists the Euclidean partition of the test from the neuron's center point and a while later applies the RBF divide work using the sigma value(s). The resulting worth is passed to the neurons in the model layer.

The Pattern layer or Summation layer is layer in the framework is different for BPN frameworks and for GRNN frameworks. For BPN frameworks there is one model neuron for each arrangement of the goal variable. The genuine target class of every planning case is taken care of with each disguised neuron; the weighted regard coming out of a covered neuron is taken consideration. The subordinate F'(x) is enlisted in two phases are Feed-forward: the data x is dealt with into the framework. The unrefined limits at the center points and their subordinates are evaluated at each center point [30]. The subordinates are taken care of. Back spread: The predictable 1 is dealt with into the yield unit and the framework is run backward. Moving toward information to a center point is incorporated and the result is expanded by the value set aside in the left bit of the unit. The result is transmitted to the other side of the unit. The result accumulated at the data unit is the auxiliary of the framework work with respect to x.

13.4.3 Support Vector Machine (SVM)

In AI, support vector machines are coordinated learning models with related getting the hang of calculating's that different information and see plans, utilized for ask for and fall away from the faith assessment. Given a huge amount of preparing models, each set apart for having a spot with one of two courses of action, a SVM arranging calculation produces a model that disperses new models into one request or the other, making it a non-probabilistic joined straight classifier. A SVM model is a delineation of the models as focuses in space, mapped with the target that the instances of the different classes are separated by a reasonable opening that is as wide as could be ordinary thinking about the current circumstance. New models are then mapped into that indistinguishable space and predicted to have a spot with a class subject to which side of the initial they fall on. In spite of performing direct solicitation, SVMs [15] can competently play out a non-straight assembling utilizing what is known as the bit stunt, certainly

mapping their responsibilities to high-dimensional part spaces. A help vector with machining creates a hyper plane or set of hyper planes in a high- or boundless dimensional space, which can be utilized for depiction, fall away from the faith, or different undertakings. Normally, an unprecedented detachment is polished by the hyper plane that has the best parcel to the closest arranging information explanation behind any class (suggested utilitarian edge), since everything considered the more noteworthy the edge the lower the hypothesis mistake of the classifier.

The hyper planes in the higher-dimensional space are depicted as the strategy of focuses whose spot thing with a vector in that space is steady. The vectors depicting the hyper planes can be picked to be straight blends in with parameters of pictures of highlight vectors that happen in the information base. With this decision of a hyper plane, the focuses in the part space that are mapped into the hyper plane are depicted by the affiliation, if winds up being little as ends up being further away from , each term in the whole measures the level of closeness of the test point to the seeing information base point [20]. In this way, the total of bits above can be utilized to quantify the general closeness of each test point to the server farms beginning in both of the sets to be disengaged. Note the way that the arrangement of focuses mapped into any hy an outcome, permitting inside and out all the more amazing disengagement between sets which are not twisted at all in the chief space.

The parameters of the best edge hyper plane are controlled by clarifying the improvement. There exist a couple of explicit computations for quickly dealing with the QP issue that rises up out of SVMs, generally relying upon heuristics for isolating the issue into tinier, progressively reasonable irregularities. Another philosophy is to use an inside point procedure that uses Newton-like cycles to find an answer of the Karush–Kuhn–Tucker conditions of the fundamental and twofold issues. As opposed to understanding a gathering of isolated issues, this technique clearly deals with the issue far and away. To keep away from handling a straight system including the tremendous part organize, a low position estimation to the cross section is consistently used in the bit stunt [18].

Another typical technique is Platt's Sequential Minimal Optimization (SMO) count, which isolates the issue into 2-dimensional sub-gives that are seen experimentally, clearing out the necessity for a numerical smoothing out computation and lattice amassing. This computation is mindfully clear, easy to execute, generally faster, and has better scaling properties for problematic Support vector machine (SVM) issues. The one of a kind example of straight assistance vector machines can be seen even more capably by a comparable kind of estimations used to smooth out its close by cousin, vital backslide; this

class of counts joins sub-incline drop (e.g., PEGASOS]) and encourage plunge (e.g., LIBLINEAR). LIBLINEAR makes them appeal getting ready time properties. Each get together accentuation requires some genuine vitality direct in the time taken to examine the train data and the emphases [8] moreover have a Q-Linear Convergence property, making the count incredibly fast.

13.4.4 Pre-Processing

It is defined by converting RGB image to Gray scale image. The RGB components of the innovative given image are changed to gray components and blurring of the images along with noise is removed. The pre-processing of an image is done by using 3×3 median filters. This step helps to improve the appearance and quality of the image to be ease in detecting cancers [4]. The pictures are dependent upon different sorts of commotions, for example, anomalies and so forth. These clamors may corrupt the nature of the picture and therefore it can't give right data to ensuing picture division and edge recognition. So as to improve the nature of the picture, activities should be performed to evacuate or diminish debasements endured in its securing. Pre-processing is additionally required so as to homogenize and isolate the power disseminations of the dangerous and considerate tissues [18]. This can be accomplished by utilizing a few denoising methods, viz., Gaussian channel, middle channel.

13.4.5 Feature Extraction

This is used to extracting the features of the image and it is done by using GLCM algorithm. GLCM algorithm extract features such as energy, correlation, homogeneity, contrast, skewness and entropy. The potency of texture based tumor detection; segmentation and classification are done to outline the variation of the tumor surface from the non-tumor region which leads to fine extraction. Feature extraction is tactful for identifying the brain tumor and skin cancer and its authentic locations [5].

This component extraction stage examines the chapters from the info picture, to remove the most noticeable highlights. The motivation behind element extraction is to lessen the first information by estimating certain properties, or highlights, that recognize one information design from another example. The separated element ought to give the qualities of the info type to the classifier by thinking about the portrayal of the applicable properties of the picture into highlight vectors.

The viability of surface based tumor discovery, division and order has been talked about to describe the tumor surface variety which is relied upon

to be unique in relation to the non-tumor district, which further expands the assurance of fine extraction. In plan affirmation and in picture getting ready, incorporate extraction is a one of a kind sort of dimensionality decline. Right when the data to an estimation is too colossal to even think about being in any capacity arranged and it is suspected to be broadly overabundance then the data will be changed into a diminished depiction set of. Highlight Extraction is useful in recognizing mind tumor and skin malignant growth where is actually found and aides in anticipating next stage.

13.4.6 Database Updation

Here, Normal brain and skin images are stored. Then features of each image is identified using GLCM.

13.4.7 Classification

Finally, classification is applied to identify whether the brain and skin images are affected by cancer or not. To classify an image SVM algorithm is used [6].

13.4.8 Clustering

If the cancer is identified in a given image, Clustering operation is performed to highlight the affected portion [14].

13.4.9 Segmentation Using FCM Clustering

The -means clustering is generally based on the pixel scheme. The pixel based method is the simplest, less complex than the edge and region based methods. Proposed method along with existing will produce a better result which causes segmentation method to play a vital role. Segmentation is used to classify the images. K means algorithm was used as an efficient segmentation [8]. Information bunching is the path toward dividing data segments from remarkable pictures into classes or gatherings so things in a comparative class are as practically identical as could be normal the situation being what it is, and things in different classes are as disparate as could be normal the situation being what it is.

Dependent upon the possibility of the data and the explanation behind which gathering is being used, different extents of likeness may be used to place things into classes, where the closeness measure controls how the bundles are modeled. A couple of occurrences of measures that can be used as in packing join partition, accessibility, and power. In bundling,

data is separated into unquestionable gatherings, datum part has a spot with correctly one gathering. In feathery gathering (moreover suggested as sensitive batching), data segments can have a spot with more than one gathering, and associated with each segment is a ton of interest levels.

13.5 Results and Discussion

In our chapter we are using FCM clustering for to segment the tumor part of the image and cancer part of the skin purpose.

Figure 13.2 is the medical image to be taken as an input for the analysis of CANCERs in the particular area. The input image can be of different medical image like Tumor cells, Cancer cells, Lesions, dead cells.

Figure 13.3 is the gray scale image for the input image taken. The conversion of the input image into the grey scale image will get a better view and identification of the cancers in the particular area.

Figure 13.4 is the filtered image for the input image taken. After the conversion of input into gray scale image, the gray scale image is sent

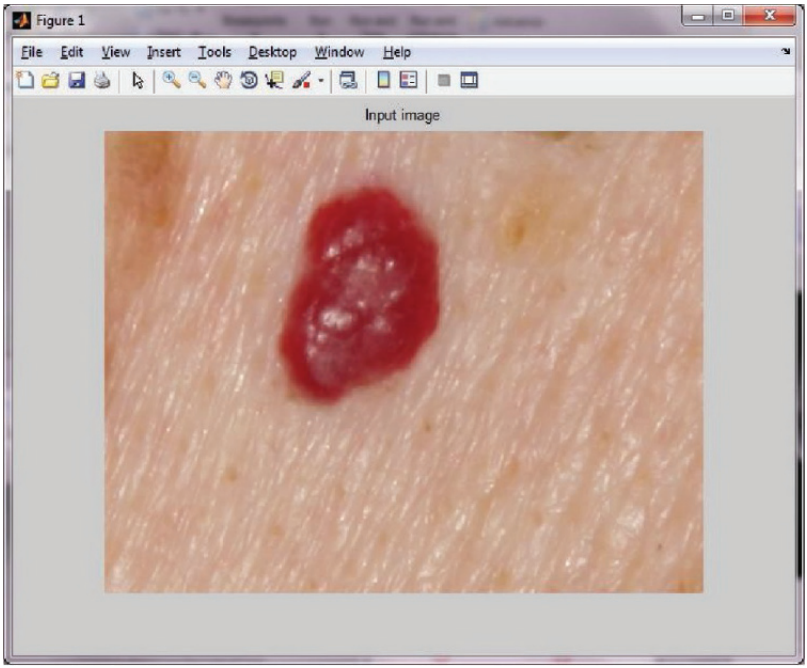


Figure 13.2 Input image.

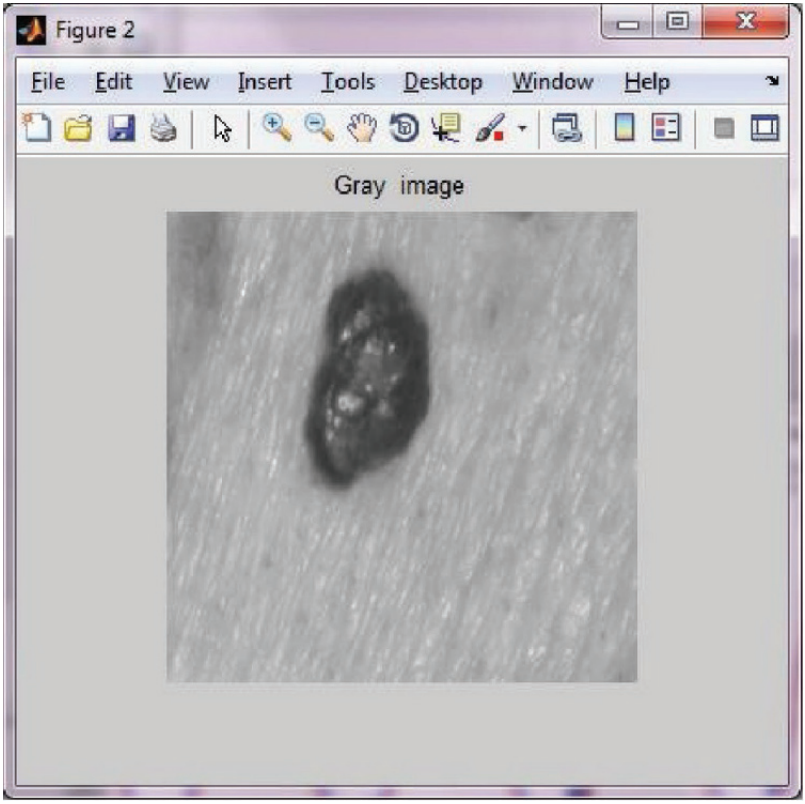


Figure 13.3 Gray image.

to the filtering process. In the filtering method, the gray scale image is filtered and filtered image is obtained. Table 13.1 displays the SNR comparison of dissimilar medical images by using the existing and proposed algorithm.

Figure 13.5 shows the graph for comparison of the image taken. It is the comparison of the existing & proposed algorithm. It shows the SNR value of the two different algorithms used for the cancer identification.

Figure 13.6 is the final output of the input image taken. It shows the exact location of the cancers located in the medical image. The white region in this figure is the area where the cancers are located. It will be more helpful to operate the cancers during operation.

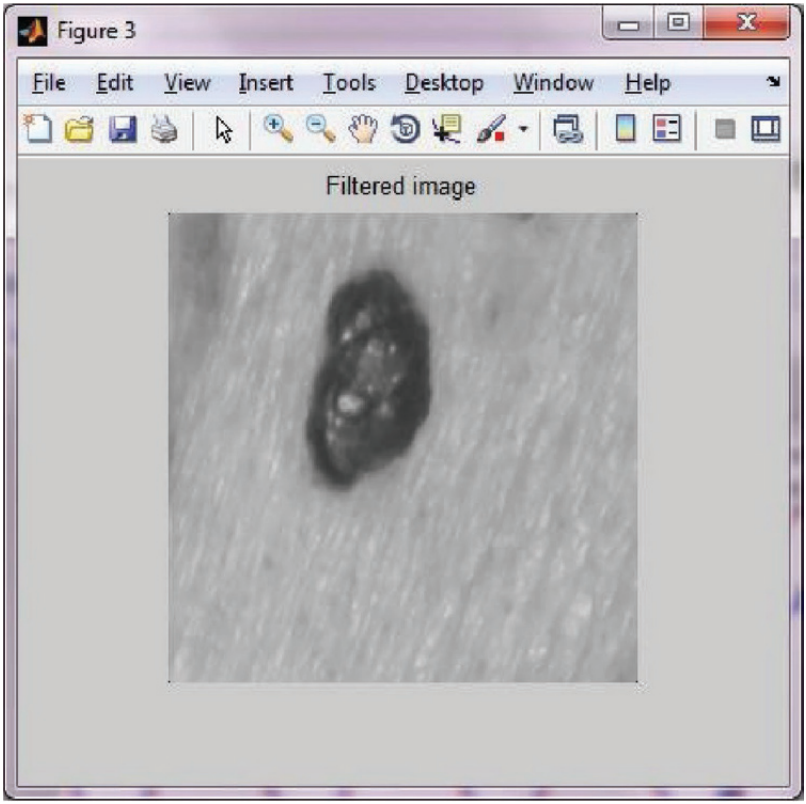


Figure 13.4 Filtered image.

Table 13.1 SNR Comparison of different images.

Image	Existing SNR value	Proposed SNR value
Image 1	40	68.3707
Image 2	40	68.2665
Image 3	40.439	68.2316
Image 4	38.26	68.3098
Image 5	41.93	68.2435
Image 6	39.114	68.3603
Image 7	68.2783	79.86
Image 8	68.2925	85

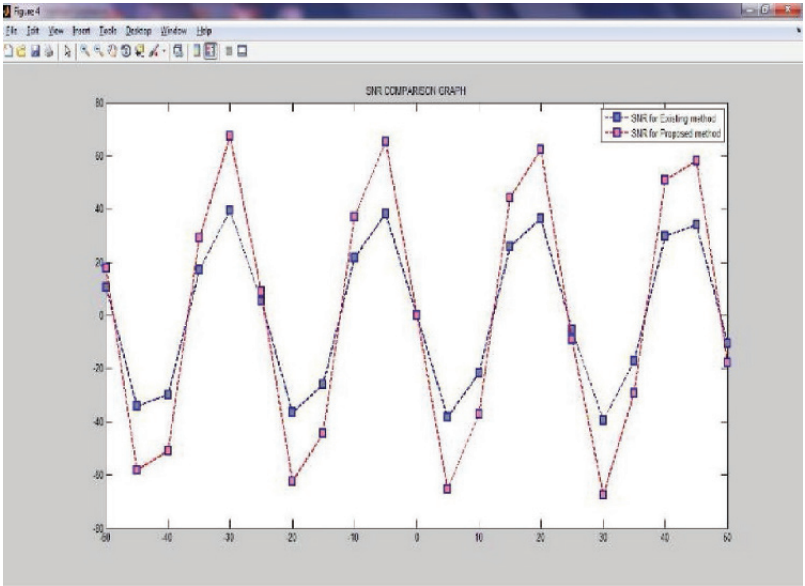


Figure 13.5 SNR comparison.

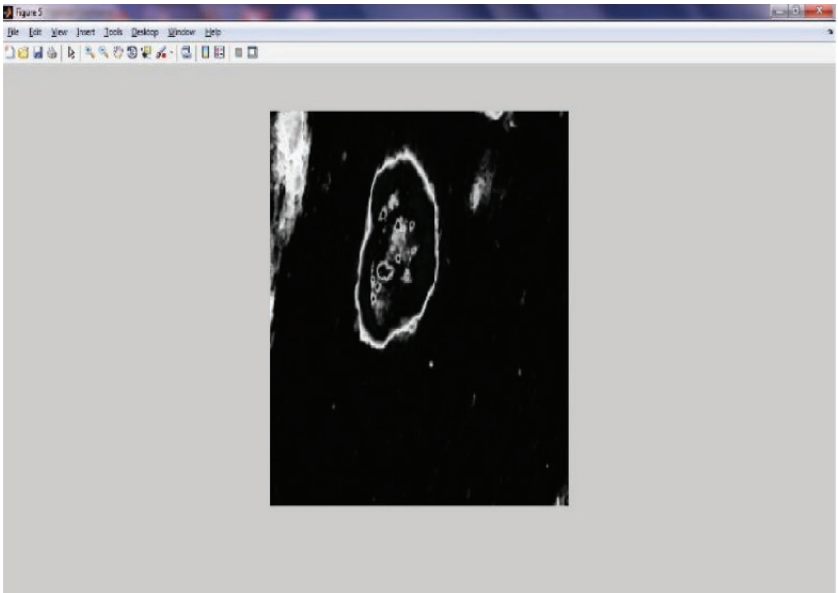


Figure 13.6 Identification of cancer.

13.6 Conclusion

A use of mass identifier with GLCM, incorporate sufficiency and the upside of FCM portrayal proposed in this part. The intensity of the estimation helps in recognizing variable size and condition of chapters satisfactorily makes to concentrate on this work. GLCM give the noteworthiness in removing feature vectors of each mass by tangling the primary picture. SVM portrayal is arranged and examined for the figuring's ampleness. Every methodology helps in performing mass acknowledgment and request effectively. Feature vectors can be loosened up or changed to suit different applications and pictures. The ideal aftereffect of proposed counts helps in various applications, for instance, in clinical for diagnosing the reactions and pinpointing interest even more unequivocally. Human examinations are up till now required during the time spent pinpointing the component which prompts update of delayed consequences of the yield close by computation upgrades. Better execution and precision are gotten from the PC assessment.

References

1. Al-Kofahi, Y., Lassoued, Lee, W., Roysam, B., Improved automatic detection and segmentation of cell nuclei in histopathology images. *IEEE Trans. Biomed. Eng.*, 57, 841–852, 2010.
2. Gowrishankar, R. *et al.*, An Unified Reeb Analysis for Cortical Surface Reconstruction of MRI Images. *Biomed. Pharmacol. J.*, 10, 939–945, 2017.
3. Patil, A. and Dhanvijay, M., Cancer detection technique using image processing for identification of machine printed characters. *Int. J. Innov. Eng. Res. Technol.*, 2, 10, 2015.
4. Sindhumathy, S. *et al.*, Analysis of Magnetic Resonance Image Segmentation using spatial fuzzy clustering algorithm. *J. Glob. Pharma Technol.*, 10, 12, 88–94, December 2018.
5. Beeman, S.C., Zhang, M. *et al.*, Measuring glomerular number and size in perfused kidneys using MRI. *Am. J. Physiol. Renal Physiol.*, 300, 1454–1457, 2011.
6. Vetrivelan, P. *et al.*, TB screening using SVM and CBC techniques. *Curr. Pediatr. Res.*, 21, 2, 338–342, May 2017.
7. Martinez-Martin, E. and del Pobil, A.P., Visual object recognition for robot tasks in real life scenarios. *10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, 2013.
8. Rama, J. *et al.*, Automatic Detection of Diabetic Retinopathy through Optic Disc Using Morphological Methods. *Asian J. Pharm. Clin. Res.*, 10, 28–31, 2017.

9. Harder, N. *et al.*, Automated recognition of mitotic patterns in fluorescence microscopy images of human cells. *Proc. IEEE Int. Symp. Biomed. Image.*, pp. 1016–1019, 2006.
10. Vetrivelan, P. *et al.*, An Identification of efficient vessel feature for Endoscopic Analysis. *Res. J. Pharm. Technol.*, 10, 8, 2633–2636, August 2017.
11. Inthajak, K. *et al.*, Medical image cancer detection with feature stability and KNN classification. *2011 Eighth International Joint Conference on Computer Science and Software Engineering (JCSSE)*, 2011.
12. Sivaganesan, S. *et al.*, Design and Development of Smart Glucose Monitoring System. *Int. J. Pharma Biosci.*, 8, 3, 631–638, July 2017.
13. Kouroua, K. *et al.*, Machine learning applications in cancer prognosis and prediction. *Comput. Struct. Biotechnol. J.*, 13, 8–17, 2015.
14. Vetrivelan, P. *et al.*, An Investigation of Bayes Algorithm and Neural Networks for identifying the Breast Cancer. *Indian J. Med. Paediatr. Oncol.*, 38, 3, 340–344, July 2017.
15. Zhang, M., Wu, T., Bennett, K.M., Small Cancer Identification in Medical Images Using Regional Features from Optimum Scale. *IEEE Trans. Biomed. Eng.,* 1051–1062, April 2015.
16. Ramesh, C. *et al.*, An Efficient Tissue Segmentation of Neonatal Brain Magnetic Resonance Imaging. *Res. J. Pharm. Technol.*, 12, 6, 1–8, June 2019.
17. Yang, Q. and Parvin, B., CHEF: Convex hull of elliptic features for 3D cancer detection. *Object Recognition Supported by User Interaction for Service Robots*, 2002.
18. Yogadinesh, S. *et al.*, Certain Investigation of Identify the New Rules and Accuracy using SVM algorithm. *Middle-East J. Sci. Res.*, 23, 2074–2080, 2015.
19. Xiaowei, C., Xiaobo, Z., Wong, S.T.C., Automated segmentation, classification, and tracking of cancer cell nuclei in time-lapse microscopy. *IEEE Trans. Biomed. Eng.*, 53, 762–766, 2006.
20. Jony, M.H., Tuj Johora, F., Khatun, P., Rana, H.K., Detection of Lung Cancer from CT Scan Images using GLCM and SVM. *International Conference on Advances in Science, Engineering and Robotics Technology*, pp. 1–6, 2019.
21. Ramesh, C. *et al.*, Region Growing Image Segmentation for Newborn Brain MRI. *BioTechnol.: Ind. J.*, 12, 12, 1–8, December 2016.
22. Mohanty, A., Beberta, S., Lenka, Classifying Benign and Malignant Mass using GLCM and GLRLM based Texture Features from Mammogram. *Eng. Res. Appl.*, 1, 687–693, 2013.
23. Mohan, N. *et al.,* Classification of Malignant Melanoma and Benign Lung Cancer by using Deep Learning Based Neural Network. *Int. J. Innov. Technol. Exploring Eng.*, 9, 3, 2958–2963, Jan 2020.
24. Acharya, V. and Kumar, P., Detection of acute lymphoblastic leukemia using image segmentation and data mining algorithms. *Med. Biol. Eng. Comput.*, 57, 1783–1811, 2019.

25. Ramesh, C. *et al.*, Detection and Segmentation of Optic Disc in Fundus Images. *Int. J. Curr. Pharm. Res.*, 10, 5, 20–24, 2018.
26. Humeau-Heurtier, A., Texture Feature Extraction Methods: A Survey. *IEEE Access*, 7, 8975–9000, 2019.
27. Tseng, H.-H. *et al.*, Machine Learning and Imaging Informatics in Oncology. *Oncol. Informatics – Rev.*, 98, 344–362, Apr 2018.
28. Sali, L. *et al.*, Computer-based self-training for CT colonography with and without CAD. *Eur. Radiol.*, 28, 4783–4791, 2018.
29. Han, K.T.M. and Uyyanonvar, B., A Survey of Cancer Detection Algorithms for Biomedical Images. *International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES)*, pp. 57–60, 2016.
30. Al-Milli, N., Backpropagation Neural Network for Prediction of Heart Disease. *J. Theor. Appl. Inf. Technol.,* 56, 1, 131–135, Oct 2013.
31. Ansari, A.Q., Automated Diagnosis of Coronary Heart Disease Using Neuro-Fuzzy Integrated System. *2011 World Congress on Information and Communication Technologies*, pp. 1383–1388, 2011.
32. Atel, D., Shah, Y., Thakkar, N. *et al.*, Implementation of Artificial Intelligence Techniques for Cancer Detection. *Augment. Hum. Res.*, 5, 6, 2020.
33. Marka, A., Carter, J.B., Toto, E. *et al.*, Automated detection of nonmelanoma skin cancer using digital images: A systematic review. *BMC Med. Imaging*, 19, 21, 2019.
34. Masood, A., Al-Jumaily, A.A., Adnan, T., Development of Automated Diagnostic System for Skin Cancer: Performance Analysis of Neural Network Learning Algorithms for Classification, in: *Artificial Neural Networks and Machine Learning—ICANN 2014. ICANN 2014. Lecture Notes in Computer Science*, vol. 8681, S. Wermter *et al.* (Ed.), Springer, Cham, 2014.
35. Gavrilov, D.A., Melerzanov, A.V., Shchelkunov, N.N. *et al.*, Use of Neural Network-Based Deep Learning Techniques for the Diagnostics of Skin Diseases. *Biomed. Eng.*, 52, 348–352, 2019.
36. Grzesiak-Kopeć, K., Nowak, L., Ogorzałek, M., Automatic Diagnosis of Melanoid Skin Lesions Using Machine Learning Methods. *Artificial Intelligence and Soft Computing*, ICAISC, 2015.
37. Konstantina, K. *et al.*, Machine learning applications in cancer prognosis and prediction. *Comput. Struct. Biotechnol. J.*, 13, 8–17, 2015.
38. Adaline Suij, R. *et al.*, An Efficient Classification of medical Images using Deep Learning Technique. *Int. J. Pharm. Res. Adv. Sci. Res.*, 12, 4, 119–129, 2020.
39. Valavanidis, A., Improving Early Cancer Diagnosis by Artificial Intelligence Algorithms and Machine Learning Models. *Sci. Rev.,* 1, 1–26, April 2020. WEBSITE: chem-tox-ecotox.org/ScientificReviews
40. Dhahri, H. *et al.,* Automated Breast Cancer Diagnosis Based on Machine Learning Algorithms. *J. Healthc. Eng.,* 2019, 1–9, 2019.
41. Cruz, J.A. *et al.,* Applications of Machine Learning in Cancer Prediction and Prognosis. *Cancer Inf.,* 6, 59–77, 2007.

14

A Study of Human Sleep Staging Behavior Based on Polysomnography Using Machine Learning Techniques

Santosh Kumar Satapathy* and D. Loganathan

Pondicherry Engineering College, Puducherry, India

Abstract

The rapid changes in life styles may cause to not get proper sleep in daily basis. This may create a lot health issues in later period of life of the subject. Among those health issues, the major challenges have seen across global as sleep related disorder. The major cause behind this was stress in social and professional life and it becomes difficult to get proper sleep in night time. The consequences of sleep deprivation may create risky of human lifes. To consider with this all points,our main objective of this chapter is to describe the sleep behavior and its changes patterns during sleep cycles. Additionally we also consider one case studies where we proposed an automated sleep stage scoring approach for analysis and classification of sleep stages. We are considering the electroencephalogram signals for studying the brain behaviors and followed by this we are considering the three major steps such as signal preprocessing, analysis of signal characteristics in terms to time and frequency features and finally considers the machine learning classification models for classifying the sleep stages. We obtained sleep records from ISRUC-Sleep public repository. It has reported from experiment that the proposed research work, the SVM classification techniques turned out is most useful for classifying the sleep stages of subject-16 through C4-A1 channel with an accuracy of 97.20%.

Keywords: Computer aided diagnosis, sleep stages, EEG analysis, 30-s EEG signal segments, feature extraction, feature screening, classification, machine learning

**Corresponding author*: santosh.satapathy@pec.edu

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (339–358) © 2021 Scrivener Publishing LLC

14.1 Introduction

To maintain both physical and mental fitness proper sleep required for each category of subjects with respect to different ages. The sleep ratio is a different requirement from born baby to elders. A quality amount of sleep is required for each individual for proper functioning of the different parts of the body. The National Sleep Foundation has published a sleep requirement chart with different age groups. When the number of hours during the night is reduced a state called "sleep debt" is accumulated and it becomes towards the disease of sleep disorder. Sleep disorder is the term given to disturbance in sleeping patterns of person in terms of quality and quantity. Due to disturbed sleep, there is an imbalance between mental and physical functioning which further worsens health conditions. Nowadays sleep-related disorders are one of the major reasons of disease among human deaths in many developed countries. Not only India but the whole world is facing widespread infectious sleep disorders. The major cause of death is unawareness among people related to sleep disorder. Also, people are very careless regarding their lifestyle and eating habits which cause a sleep disorder to grow at a fast rate in their body and finally lead to the end of life. Computer science is growing rapidly, giving new fields of research arena and progress interfaces to tackle the disastrous situations that can be a great source of assistance for an ordinary human being to diagnose and further get know about the deadly sleep disorders with which they are suffering.

As per the survey, 91% of the population in India is suffering from sleep-related issues; however, just 3% of Indians talk about their sleep-related issues with physicians. Sleep disorders constitute a global epidemic. Along with this sleep disorder has also a significant economic impact. These economic, socio and personal consequences which the people with sleep disorders face are much higher than the cost involved in the treatment. According to the report from the University of Maryland Medical Centre, the annual cost of direct sleep-related problems in the USA is $16 billion while indirect costs reaching $50–100 billion which includes accidents, litigation, property destruction, hospitalization, and death. Various impacts occurred due to various sleep disorders and its effect either may be on a short or long term basis. In shorter manner, the subjects may not be able to put full concentration and efforts in their day to day basis activities, in some cases it may happen that some accidents are placed either in working place or on the road. But on long term basis, it becomes highly risky of life for increasing blood pressue, high

possibilities of occurring heart attack, cardio vascular diseases, mental disorder and obesity issues.

The sleep diseases turn out to be vital because of its commonality in whole world basis. Hence many researchers proposed various techniques and methods to diagnose sleep disorders and its various issues. In the early stages the researchers have used the quantitative approach for the detection of sleep disorders. But the accuracy for the enhancement of the existing system was the main challenge as the entire investigation and diagnosis for recognizing the disorder level. Different types sleep related have seen irrespective of age and genders. Some of the disorders which are generally really challenge nowadays for clinicians, these disorders are sleep apnea, insomnia, hypersomnia, periodic limb movement sleep disorder and parasomnia.

The whole sleep cycle are witnessed with major two phases that are non-rapid eye movement (NREM) and rapid eye movement (REM). Both sleep stages occur at regular interval of times within 6–7 h of sleep. The first sleep standards was documented by the Rechtschaffen and Kales (R&K) in 1968 [1]. As per R&K rules, the NREM stages are divided into four sub-sleep stages that are N1, N2, N3 and N4. After some years of gap the American Academy of Sleep Medicine (AASM) [2] has published a new sleep analysis guideline with minor updating of existing R&K rules, the changes was reflected with NREM sleep phase, instead of 4 sub stages, here only mentioned 3 stages, by merging of N3 and N4 into one that is called as N3 stage. The major requirement of sleep stage scoring is polysomnography (PSG) signals [3]. In general PSG signals are combination of electroencephalogram, electrooculogram, electromyogram and electrocardiogram. Out of these signals, the most effective signal for monitoring the brain activities and its changes behavior are through EEG signals because it provides direct information about brain behavior, so that it is more effective for the purpose of diagnosis and types of sleep related diseases.

14.2 Polysomnography Signal Analysis

In general all sleep labs followed uniformity during sleep stage scoring. During test the sleep disorder, first of all, consider the PSG signal of those particular subjects, sometimes combination of multiple channels was extracted for identifying the sleep irregularities. But due to some difficulties raised during multiple signal cases, the major difficulty was

inconvenience for patient to wear out number of sensors in the different part of the body, which may cause sometimes disturbances during sleep recordings and it may become a problem to reach the accurate diagnosis solutions. For maximum results, clinicians referred to only one channel of EEG signals during testing. The major advantage from EEG signals is analysis of the brain wave form by segmenting the whole signals into different sub-signals.The different EEG sub-signals help to identify the major events that happened during sleep and apart from that it also support further analysis of the sleep characteristics. Basically five EEG sub-bands are generally used for analyzing the characteristics of the sleep patterns that are delta (δ), theta (θ), alpha (α), beta (β) and sigma (σ).

During sleep, the brain behavior reflects the transition between sleep stages and these changes are reflected on EEG signals. These signals are collected using different placed electrodes placed in the subject's head. The EEG signals are composed of α wave, δ wave, β wave and θ wave, K complex, sleep spindles and sawtooth. The W (wakefulness) corresponds to alpha 8-12 Hz rhythm. The N1 (Non-REM 1) corresponds to alpha 8-12 Hz rhythm. The N2 (Non-REM 2) corresponds to K-complexes such as negative peak followed by a positive complex and a final negative voltage. Furthermore, the N2 also corresponds to <1.5 Hz range and sleep spindles (burst of oscillatory waves) at sigma of 12-15 Hz band. Finally, the N3 (Non-REM3) present a slow-wave activity (delta rhythm) of 0.5-3Hz, the eye movements are unusual, and EMG tone is low. Moreover, the existence of rapid eye movements is visible in EOG, relatively low-amplitude and mixed frequency activity, and also presents the lowest muscle tones on EMG.

1. Delta Wave (δ): In general the range of delta wave patterns is approximately 3 Hz or lower. The wave patterns are in slow and highest in amplitude. These types of patterns are seen in specifically infants and it continues up to one year.
2. Theta Wave (θ): The frequency range of theta patterns are between 3.5 and 7.5 Hz and its nature is slow activities. These wave patterns generally occur from children up to age of 13 years.
3. Alpha Wave (α): The frequency range of alpha waves occur from the range 7.5 to 13 Hz. These wave patterns happen when the subjects are under relaxed states and closed eyes; once the subjects are in alert mood then these wave patterns disappear. It is one of the major parts of the EEG signal and generally it appears mostly for normal adults subjects.

4. Beta Wave (β): The frequency range of beta wave patterns are 14 Hz and greater. These types of wave forms are visualized for subjects who have taken drugs like sedative–hypnotic. In general its patterns are normal in brain waves. These wave behaviors occur in those subjects who were alert and most anxious in behavior manners.
5. Sigma Wave (γ): This type of wave patterns generally occurs during NREM sleep. Its frequency range from 12 to 16 Hz. Basically the sigma sub-band provides the information about sleep spindle activities. Both sigma and beta sub band behave reciprocally within NREM sleep periods but both these signal sub-bands reported their lowest value during REM sleep periods.

The sleep behavior of the subject is shown in Figure 14.1. The figure represents the 30 s epochs of all the sleep stages where we highlighted

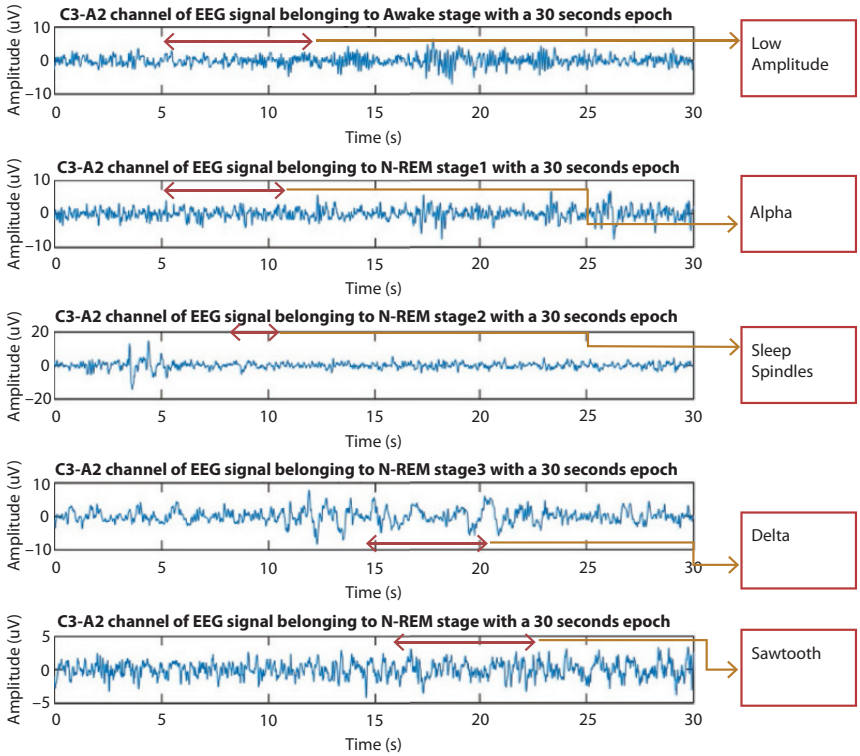


Figure 14.1 C3-A2 channel of EEG signal for different stages behavior of sleep: Awake, N-REM (stage 1, stage 2, stage 3) and REM stage.

some important sleep characteristics, which becomes more helpful for clinicians to know about the irregularities that occurred in which stages of sleep cycle.

A sample of a healthy subject with a time frame of 60 s EEG and its sub-band signals are shown in Figures 14.2 and 14.3. Similarly the sleep

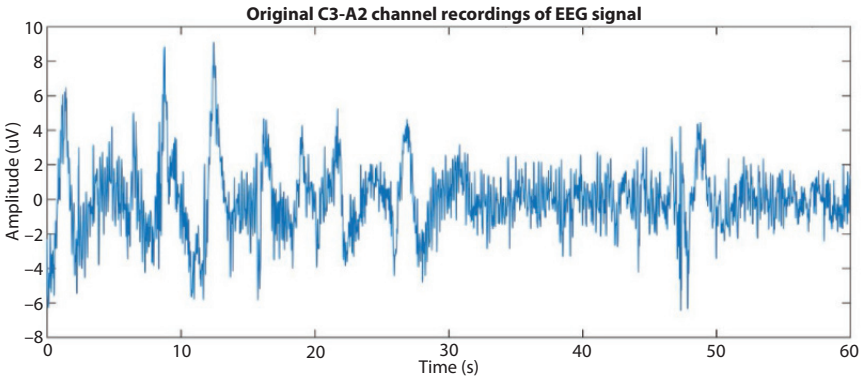


Figure 14.2 60 s epoch original EEG signal behavior from Healthy Subject.

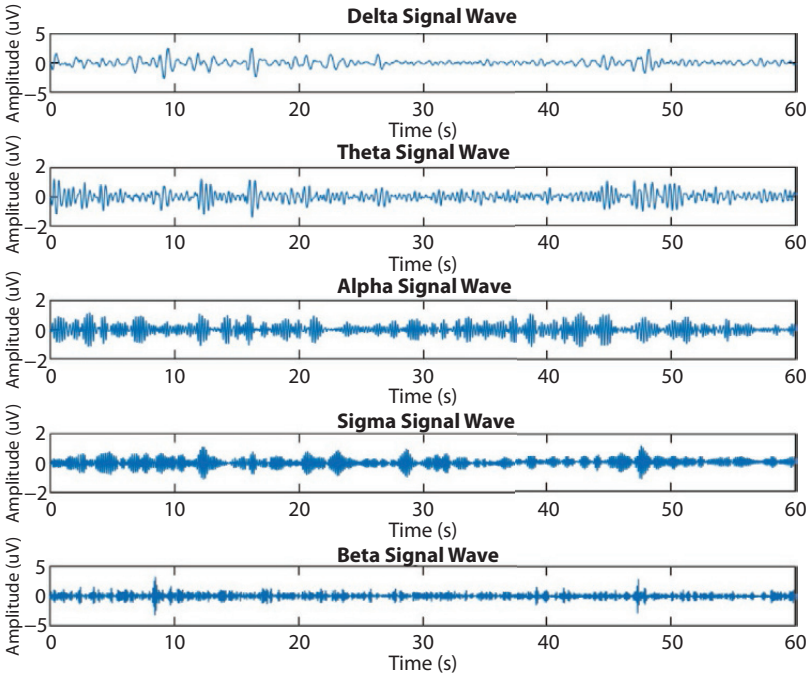


Figure 14.3 Different brain behaviors of healthy subject during different sleep stages.

characteristics of subject who is affected with sleep-related disorders with epoch length 60 s and its sub-band representation are shown in Figures 14.4 and 14.5.

The whole recording procedure followed according to 10–20 electrode placement standards. These 10–20 electrode placement standards were

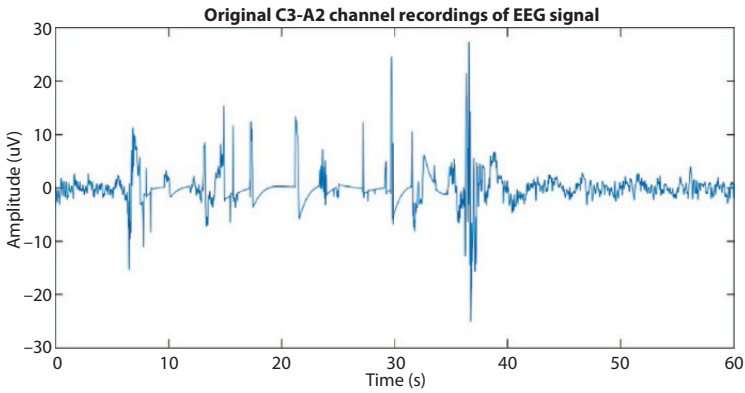


Figure 14.4 EEG signal behavior from sleep disorder subject with duration of 60 s epoch length.

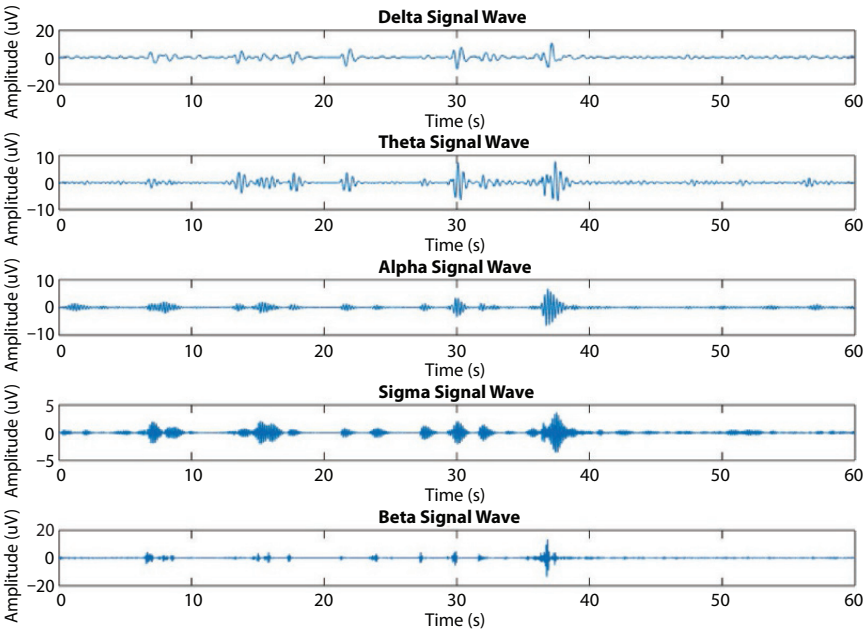


Figure 14.5 Different sleep stage characteristics from subject affected with sleep disorder.

used during recordings of EEG signal which assign a number to each fixed EEG electrode to specify the location in the left and right hemispheres. In this chapter we will include a discussion of the specific parameters used for sleep staging and a summary of the various behaviors of EEG activities needed during sleep stage scoring [4–6].

According to this placement standard the head is segment in to from specific skull landmarks such as nasion, preauricular, inion like to this adequately cover all the regions of the brain as shown in Figure 14.6. The labeling of the electrode placement 10–20 indicates that the electrode placed in the central midline at 10, 20, 20, 20, 20 and 10% of the total nasion–inion distance, like that the other electrodes are placed in the same way of fractional distances. From fixed electrodes the respective parts of the body behavior are monitored and used for the purpose of analysis different types of neurological disorder. The sample EEG recordings from different positional electrodes are shown in Figures 14.7 to 14.12, similarly EOG recordings are presented in Figures 14.13 to 14.14 and EMG recordings are shown in Figures 14.15 to 14.17.

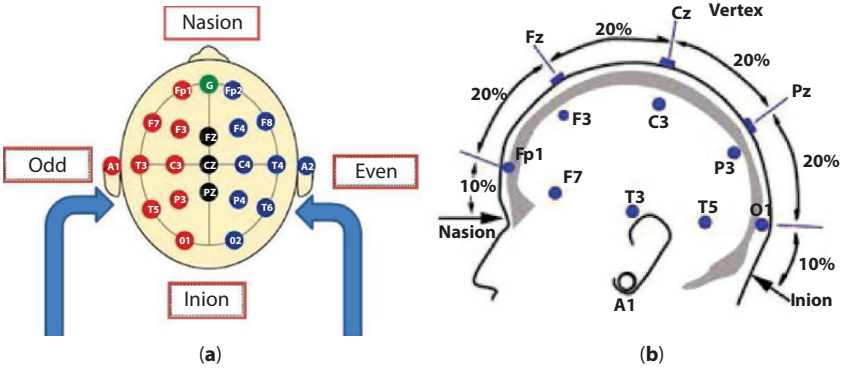


Figure 14.6 (a) 10–20 Electrode placement system, (b) Notations for placement of electrodes on the scalp [7].

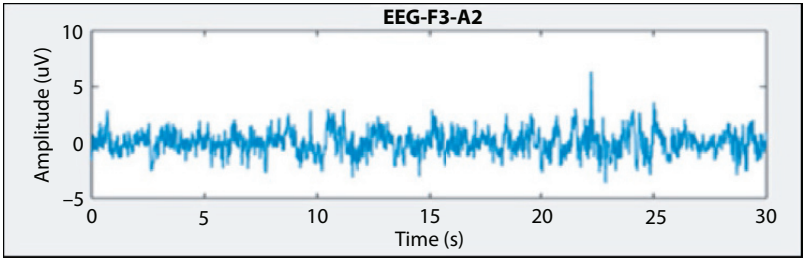


Figure 14.7 Left frontal region recording.

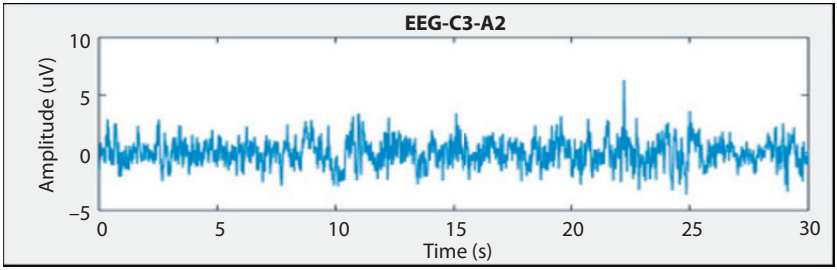


Figure 14.8 Left central region recording.

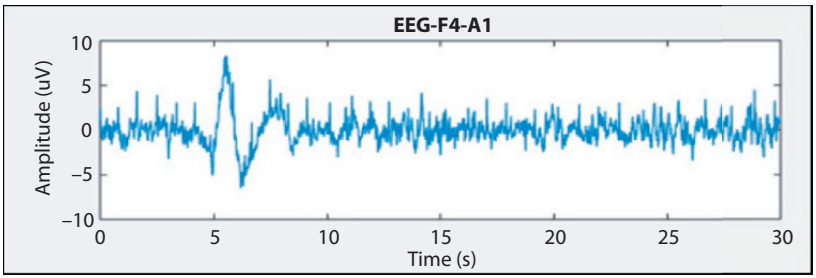


Figure 14.9 Right frontal recording.

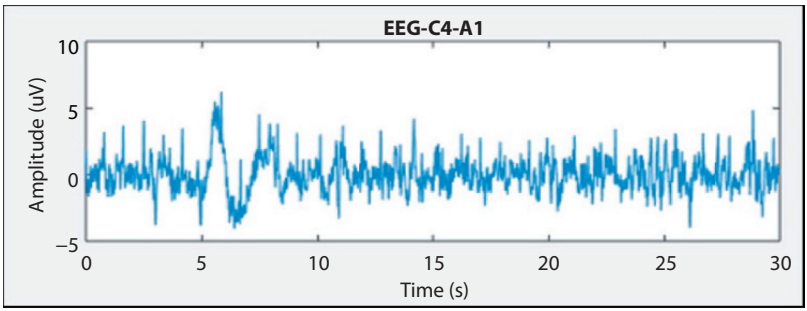


Figure 14.10 Right central recording.

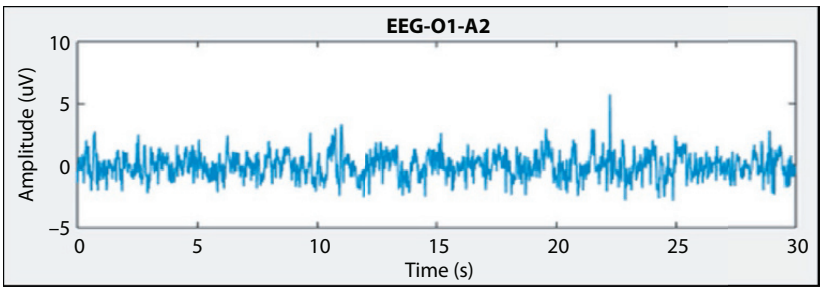


Figure 14.11 Left occipital recording.

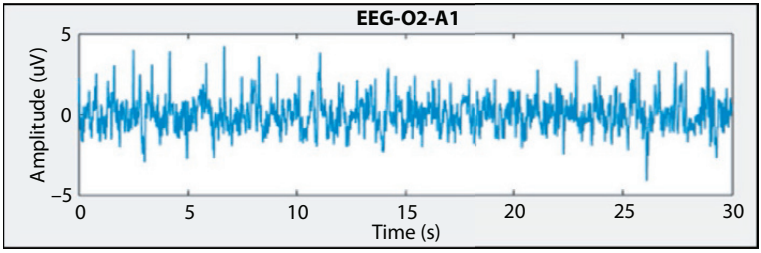


Figure 14.12 Right occipital region recording.

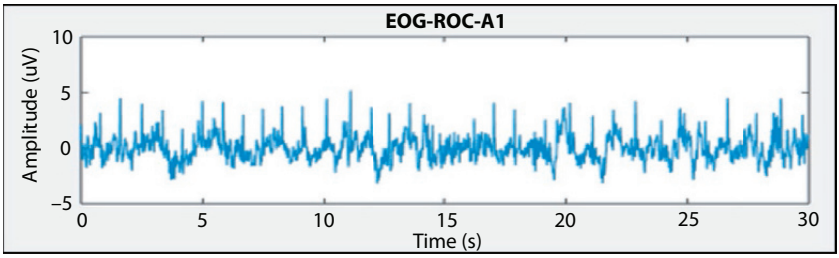


Figure 14.13 Right outer canthus region recording.

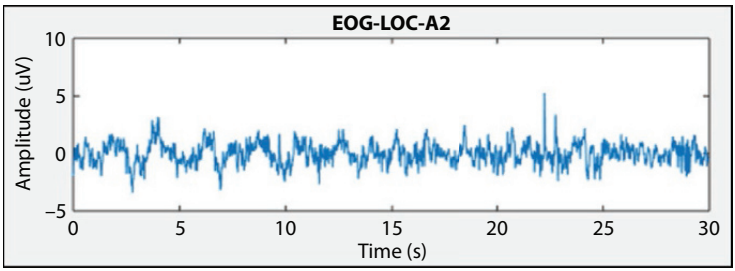


Figure 14.14 Left outer canthus region recording (typical activity of the EOG (LOC-A2)).

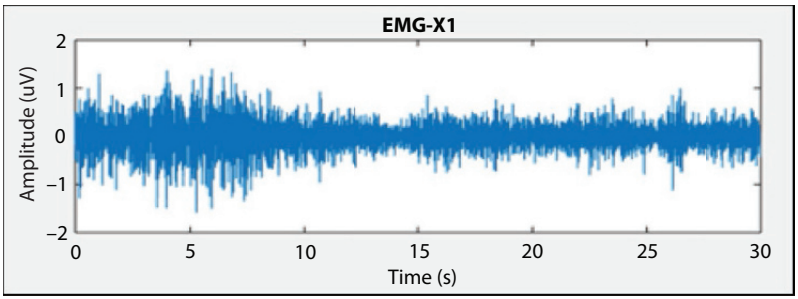


Figure 14.15 Chin EMG region recording (typical activity of the EMG (EMG-X1)).

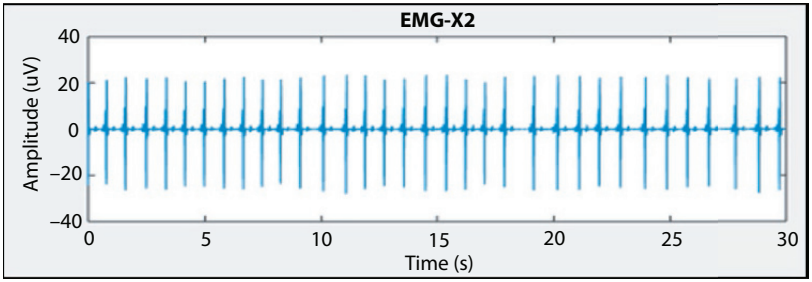


Figure 14.16 Limb EMG (left leg) region recording (typical activity of the EMG (EMG-X2)).

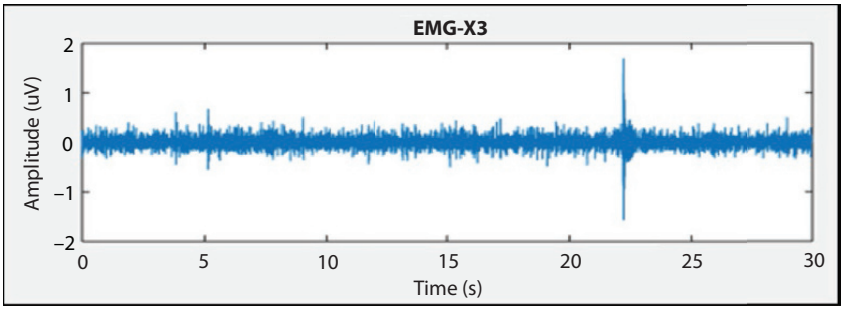


Figure 14.17 Limb EMG (Right Leg) Region Recording (Typical activity of the EMG (EMG-X3)).

14.3 Case Study on Automated Sleep Stage Scoring

14.3.1 Experimental Data

To implement the SleepEEG test model, we have used the ISRUC-Sleep publicly available dataset for evaluating the proposed objectives. It is a polysomnographic dataset and it is prepared for sleep researchers and it is open-access comprehensive dataset for sleep stage classification and this dataset has contained the different characteristics oriented subject information which is more helpful for researchers for their experimental work. ISRUC-Sleep dataset obtained data from adults, out of that some data has collected from healthy subjects and few from subject's effect with sleep disorder and subjects under medical condition with sleep disorder. All those recordings were acquired by the Sleep Medicine Centre of the Hospital of Coimbra University (CHUC) in Portugal during 2009–2013 [8]. This dataset categorized the data into 3 subsections; first sub-section contains data of 100 adult subjects with information about sleep disorders.

Here one session recordings for individual subjects and two sleep experts prepared two hypnograms per individual subject. In second sub-section carried information of only 8 subjects with symptoms of sleep disorders and in this part, two-session recordings have to be recorded with two different dates for individual subjects. For these recordings, two hypnogram files were used to be maintained as per session recording dates. In the third sub-section, only 10 healthy subjects' data were to be contained for one session recording only and two sleep experts have to prepare two hypnograms for this sub-section category data. This dataset comprised of 8–9 h whole night polysomnographic (PSGs) recordings from subjects at the sampling rate of 200 MHz. The whole experiment carried through ISRUC-Sleep subsection-I subjects. All recorded signals from this dataset were generally segmented into 30 s epoch length. All types of operations conducted at the time of recording strongly followed AASM standard. This dataset recorded different bio-signals from subjects through 11 electrodes which were placed in various parts of the body. The recordings were extracted from subjects using standard 10–20 electrode placement rules. The collected data from different bio-signals in subsection1 is briefly shown in Table 14.1. Each epoch was annotated by sleep specialist with one sleep class stage. Generally the sleep stages include wake (w), non-rapid eye movement (N1, N2, N3, N4) and rapid eye movement (REM). As per AASM rule in this study we have combined N3 and N4 into one stage called as N3. Fundamentally wake and N1 stage is light sleep state but N2 and N3 stages are being considered as deep sleep state. Generally this stage takes 40–60% of total sleep in sleep cyclic process. In REM stage, the subject's eye movements happened and normally in this phase dreaming occurred. This sleep states takes around 90–120 min in our sleep cycle process. By observing these imbalanced sleep stages numbers, we propose two-state classification for sleep scoring and here also we considered two channels so that we can verify which channel data to be more effective for sleep scoring. In this study we have considered single channel such as

Table 14.1 Composed channels of polysomnography.

Polysomnographic signals	**Data acquisition from different electrodes position**
EEG	$EEG_{C3\text{-}A2}$, $EEG_{C4\text{-}A1}$, $EEG_{F3\text{-}A2}$, $EEG_{F4\text{-}A1}$, $EEG_{O1\text{-}A2}$, $EEG_{O2\text{-}A1}$
EOG	$EOG_{LOC\text{-}A2}$, $EOG_{ROC\text{-}A1}$
EMG	EMG_{LEG1}, EMG_{LEG2}, EMG_{CHIN}

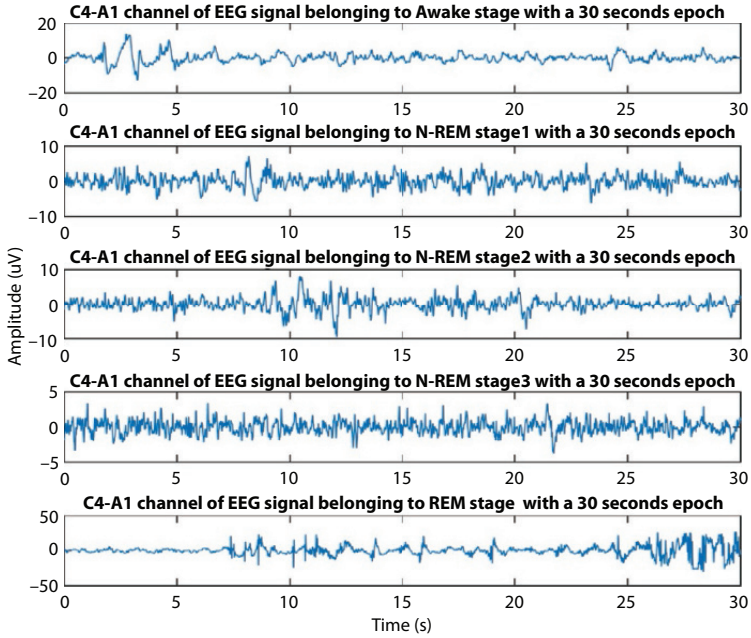
C4-A1 of EEG signals for experimental work of our SLEEPEEG test model. In this proposed study we have obtained four subjects from Subgroup-I, ISRUC-Sleep database. The details of ISRUC-Sleep subgroup-I dataset structure are shown in Table 14.1. The subjects enrolled for this study were affected with sleep disorder problems. From each subject, the brain behavior was monitored and collected through one obtained input channel such as C4-A1 and all these recording processes were to be done as per AASM guidelines and have been annotated by a sleep expert at sleep laboratory on 30 s basis. Therefore each epoch's length is 30 s or 6,000 samples. Each input channel is processed in 30 s time-frames in our proposed experiment work. Figure 14.18(a–d) presented samples of five different stages of sleep with input channel of C4-A1 for subject-16, subject-18, subject-23 and subject-77 respectively. The details of the subject undertaken for this proposed study are described in Table 14.2. Table 14.3 described the distribution of sleep epochs for each participated subjects.

14.3.2 The Methodology

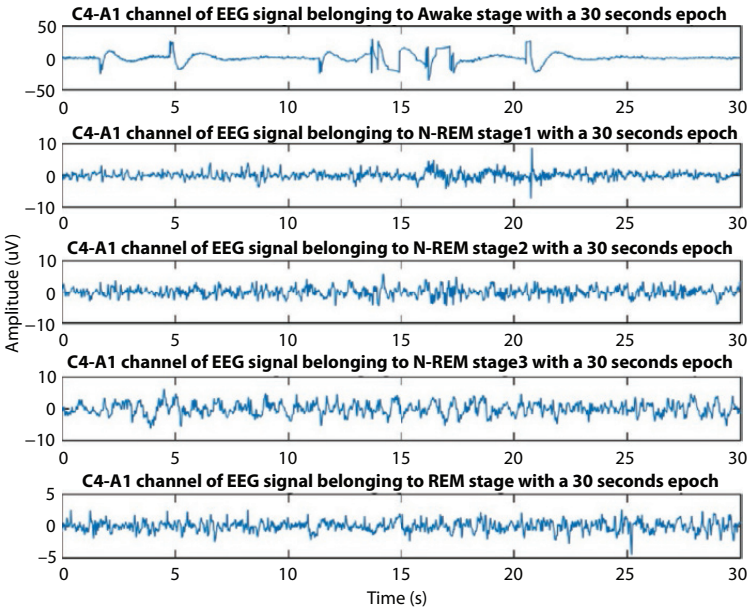
This proposed research work proposes an efficient two-state sleep stage classification based on single channels. This SleepEEG test model consists of several steps and its total workflow is shown in Figure 14.19. As shown in Figure 14.19, our proposed research work starts from signal acquisition from enrolled subjects through single channel of EEG signal. Once the signal acquisition is over, what follows is the pre-processing step to be obtained for excluding the irrelevant noises and artefacts from recorded channels, followed by feature extraction step to be obtained. Next to that feature selection step to be considered for identifying the suitable features for classification model. Finally the selected features are forwarded to the obtained different machine learning classification models for evaluation of sleep abnormality during sleep.

14.3.3 Experimental Results and Discussion

The main contributions of the proposed sleep study are how proposed methodology identifies sleep irregularities during sleep through proper sleep stage scoring in two-state sleep stage classification. For this implementation work, we have considered one channel. We have taken four subjects having different genders: two of them belong to male category and the other two belonging to female category, all who have sleep problem symptoms. In this work we have used C4-A1 channel of EEG signals for the purpose of recordings the information from obtained subjects



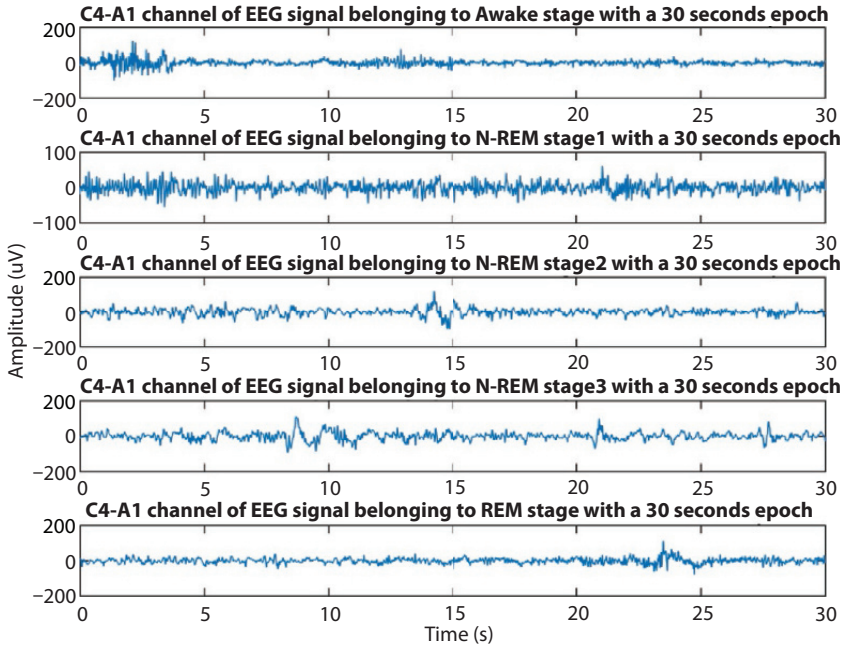
(**a**)



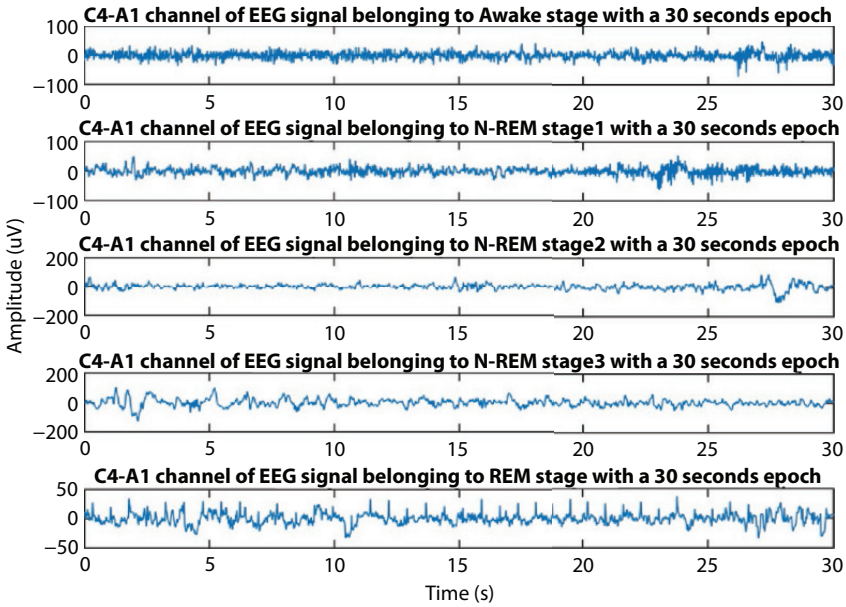
(**b**)

Figure 14.18 Sleep stages behavior of the subjects (a) Subject-16, (b) Subject-18.

(*Continued*)



(**c**)



(**d**)

Figure 14.18 (Continued) Sleep stages behavior of the subjects (c) Subject-23 and (d) Subject-77.

Table 14.2 Details of enrolled subjects in this proposed work.

Sl. No.	Subject Number	Age	Sex	Diagnosis/Other problems
1	Subject-16	50	Male	Sleep Apnea/No
2	Subject-18	38	Male	Sleep Apnea/PLMS
3	Subject-23	50	Female	Sleep Apnea/PLMS
4	Subject-77	46	Female	Sleep Apnea/No

Table 14.3 Description of epochs of various sleep stages from four subjects used in this study.

Subject Number	Wake	N1	N2	N3	REM
16	128 (17.07%)	125 (16.67%)	280 (37.33%)	120 (16.00%)	97 (12.93%)
18	133 (17.73%)	92 (12.27%)	301 (40.13%)	141 (18.80%)	83 (11.07%)
23	212 (28.27%)	99 (13.20%)	270 (36.0%)	65 (8.67%)	104 (13.87%)
77	81 (10.80%)	146 (19.47%)	309 (41.20%)	160 (21.33%)	54 (7.20%)

in this research work. This complete SleepEEG study worked upon the ISRUC-Sleep dataset, which is publicly available online collected from the Department of Sleep Medicine, Hospital of Coimbra University (CHUC) in Portugal. The entire work of this proposed study followed AASM standard rules to recognize the different patterns of sleep. In this study we have extracted total thirty-four features including both time and frequency domains. In this experimental work, we have used SVM classifiers for discriminating between the sleep states. Table 14.4 shows the confusion matrix results of the subject-16, subject-18, subject-23 and subject-77 for SVM classifiers.

The performances of each stages of sleep evaluation metrics achieved from each enrolled subject for C4-A1 channel with SVM classifier of this SleepEEG study were described briefly in Table 14.5 in terms of accuracy, recall, precision, specificity and F1-Score. The performance graphs

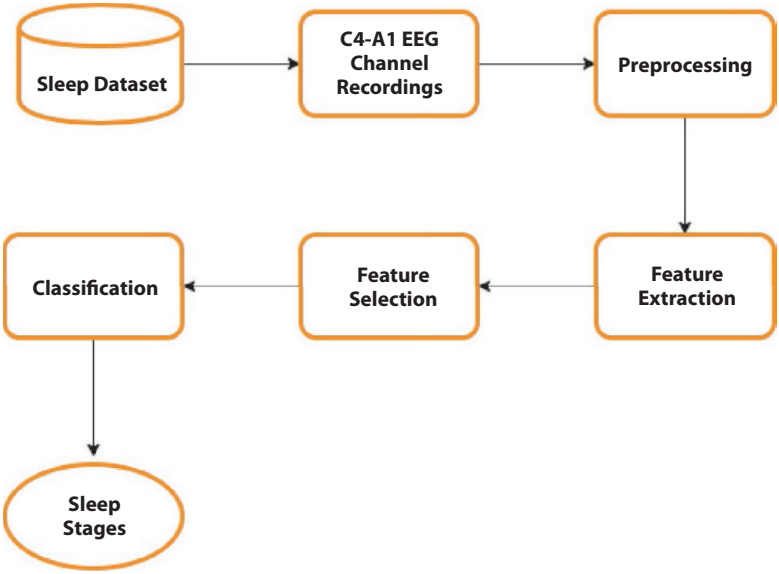


Figure 14.19 Framework of the proposed research work.

Table 14.4 Confusion matrix performance achieved by the SleepEEG study using C4-A1 EEG channel from Subject-16, Subject-18, Subject-23 and Subject-77.

C4-A1	**Subject-16**	**Wake**	**Sleep**	**C4-A1**	**Subject-23**	**Wake**	**Sleep**
SVM	Wake	302	7	**SVM**	Wake	359	21
	Sleep	14	427		Sleep	13	357
C4-A1	**Subject-18**	**Wake**	**Sleep**	**C4-A1**	**Subject-77**	**Wake**	**Sleep**
SVM	Wake	152	18	**SVM**	Wake	290	13
	Sleep	8	572		Sleep	13	434

of the proposed SleepEEG test model with the C4-A1 channel of EEG signal for the SVM classifier are shown in Figure 14.20. The proposed SleepEEG study based on single channel of EEG signals by using SVM classifier shows a higher accuracy of 97.20% for subject-16 with C4-A1 input channel.

Table 14.5 Performances Statistics of Subjects with input of C4-A1 channel sleep recordings.

C4-A1	Subject-16	Subject-18	Subject-23	Subject-77
Accuracy	97.20%	96.53%	95.47%	96.53%
Precision	98.39%	96.95%	94.44%	97.09%
Sensitivity	96.83%	98.62%	96.49%	97.09%
Specificity	97.73%	89.41%	94.47%	95.71%
F1-Score	97.60%	97.78%	95.45%	97.09%

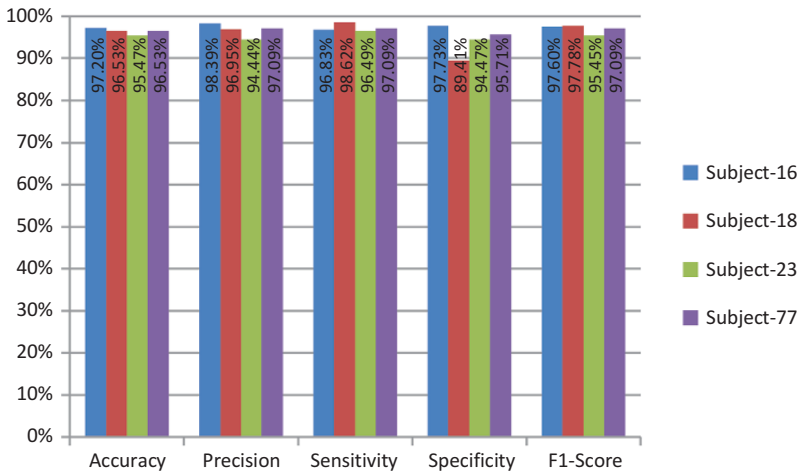


Figure 14.20 Performance graphs of the proposed SleepEEG test model with the C4-A1 channel of EEG signal for the SVM classifier.

14.4 Summary and Conclusion

An improper sleep causes serious effects on both physical and mental conditions. Sleep is important for our life for maintaining the day to day activities smoothly and it also makes one important impact in our body such as the secretion of hormone and management of our immunity system. In the later time period, it may lead to sleep-related disorders. In the current scenario, it is found across the globe that sleep-related diseases have increased

and its consequences also have very seriously impacts on our health. In view of this, some of the researchers came forward and made the analysis of sleep-related irregularities and their causes. From primary investigations through different surveys by health organization and sleep laboratory, it has been found that the major reason for sleep-related diseases is improper sleep and excessive daytime sleepiness. With this problem, the sleep experts initially go through a sleep study in manual techniques by following the edited sleep standards by R&K, and in the later time of research so many advance techniques came as introduced by different researchers or laboratory with new sleep standards edited by AASM through new automated sleep stage detection. It has found from the contributed work with regards to investigating sleep-related disorders. One major solution proposed for identifying whether the subject is effective through any sleep-related disorders or not is proper analysis of "Sleep Stage Classification or Sleep Staging Scoring". Therefore one of the important challenges across the global health sector is to find the proper diagnosis solutions for different types of sleep-related disorders.

In summary therefore we have said that sleep irregularities and different types of sleep disorders caused a great loss in terms to his/her personality and it can also affect the physical, mental and spiritual processes. Subsequently it also lowers the physical and mental performances. It also affects the attention and concentration of the subjects. It can also affect the quality of life and additionally affects the social relationship and competencies skills. With all information and concepts, we finally conclude that the proper sleep diagnostics and possibly preventive mechanisms are essential to prevent the complications and comorbidities from different sleep diseases.

References

1. Rechtschaffen, A. and Kales, A. (Eds.), *A Manual of Standardized Terminology Techniques and Scoring System for Sleep Stages of Human Sleep*, Brain Information Service/Brain Research Institute, UCLA, Los Angeles, 1968.
2. Iber, C.; Ancoli-Israel, S.; Chesson, A.L.; Quan, S.F. The AASM Manual for the Scoring of Sleep and Associated Events: Rules, Terminology and Technical Specification. *American Academy of Sleep Medicine*, 2007.
3. Caraskadon, M.A. and Rechschaffen, A., Monitoring and staging human sleep, in: *Principles and Practice of Sleep Medicine*, 2nd ed., M.H. Kryger, T. Roth, W.C. Dement (Eds.), pp. 1359–1377, WB Saunders, Philadelphia, 2005.

4. Millán, J.d.R. and Mouriño, J., Asynchronous BCI and local neural classifiers: An overview of the adaptive brain interface project. *IEEE Trans. Neural Syst. Rehabil. Eng.*, 11, 159–161, 2003.
5. Klem, G.H., Lüders, H.O., Jasper, H., Elger, C., The ten-twenty electrode system of the International Federation. *Electroencephalogr. Clin. Neurophysiol.*, 52, 3, 1999.
6. Satapathy S.K., and Loganathan D. A Comprehensive Survey and New Investigation on Sleep Disorder Detection Using EEG Signal. In: Purohit S., Singh Jat D., Poonia R., Kumar S., Hiranwal S. (eds.) Proceedings of International Conference on Communication and Computational Technologies. Algorithms for Intelligent Systems. Springer, Singapore, 2021, http://doi-org-443.webvpn.fjmu.edu.cn/10.1007/978-981-15-5077-5_45
7. https://www.medicine.mcgill.ca/physio/vlab/biomed_signals/eeg_n.html.
8. Khalighi S., Sousa T., Santos J.M., Nunes U. ISRUC-Sleep: A comprehensive public dataset for sleep researchers. *Comput. Methods Programs Biomed.*, 124, 180–92, Feb 2016.

15

Detection of Schizophrenia Using EEG Signals

Shalini Mahato*, Laxmi Kumari Pathak† and Kajal Kumari‡

Amity Institute of Information Technology, Amity University Jharkhand, Ranchi, India

Abstract

Schizophrenia is a psychiatric disorder in which people live outside the real world and live in hallucination and delusion. Schizophrenia negatively affects the mental ability and overall behavior of an individual. Presently there exists no biomarker for detection of schizophrenia. The electroencephalogram (EEG) is a non-intrusive and an affordable way for measuring a person's brainwaves. EEG is an important application by which one can understand the state of the mind as well as the behavior. Electrodes are placed on the scalp to record the EEG signal (micro-Volt) which are generated from neurons of the brain. EEG-based detection of schizophrenia would help in early as well as accurate diagnosis of it. In this chapter, a detailed study has been done for detecting the disorder schizophrenia using EEG signal based on present state of art which would contribute in future study for detecting schizophrenia using EEG signal. It leads to the conclusion that EEG has a significant role in detection of schizophrenia in the early stage, classification and can act as an adjunct tool for detection of schizophrenia.

Keywords: Electroencephalogram (EEG), event-related potential (ERP), schizophrenia, classification, biomarker

**Corresponding author*: swarup.shalini@gmail.com
†*Corresponding author*: laxmipathak1@gmail.com
‡*Corresponding author*: Kajal.jaiswal797@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (359–390) © 2021 Scrivener Publishing LLC

15.1 Introduction

Disorders related to mental health is now expanding across the globe and affects around one out of two persons in their lifespan. Schizophrenia is the most recurrent neuropsychiatric problem that involves a lifelong chance of 1% in the overall population. According to WHO, schizophrenia is an acute mental problem and more than 20 million people are suffering from it. A high mortality rate is recorded in the persons diagnosed with schizophrenia. It can create an overload on an individual as well as on the society. Person diagnosed positive with schizophrenia may encounter delusions where antipsychotic medicines and psychological support can give relief. If not taken care, then schizophrenia can be the situation where affected people start to harm themselves and in worst cases they can commit suicide also [1].

15.1.1 The Human Brain

The human body has its utmost important part—the brain, which accepts messages from our senses. It can be divided into three parts: cerebellum, brain stem, and cerebrum. Cerebrum fills in as a control to willful development, sensation and insight. Since cerebrum is huge, it is isolated in four different projections, frontal, temporal, parietal and occipital. Table 15.1 displays functions performed by the cerebrum according to its different parts.

Table 15.1 Different part of cerebrum and its operations.

Cerebrum Parts	Operations
Parietal Lobes	It receives signals from our five senses (vision, touching, smelling, listening and tasting)
Frontal Lobes	It controls movements of the body and the emotion. Human are able to speak because of this part.
Occipital Lobes	It receives signals from our eyes.
Temporal Lobes	It receives the signals through our ears. Human can recognize messages and sounds because of this part. Any sort of damage to this area can cause epilepsy with severe behavioral problems and an aggressive outpouring.

Understanding functional and cognitive behavior of human brain is an interesting area for researchers to find out better solution for various brain related issues or disorders. The activity of the brain can be recorded using EEG signal.

15.1.2 Schizophrenia

Schizophrenia is an interminable, serious, weakening psychological instability portrayed by disarray of thoughts, strange practices, and rebellious practices. This is a psychotic disorder, in which the affected person could not relate to actuality every so often. People who can be affected by schizophrenia are:

(i) Schizophrenia is more often analyzed between the ages of 16 and 25.
(ii) This can be innate (runs in families)
(iii) Men can get influenced by schizophrenia 1.5 occasions more normally than women.
(iv) Amount spent in the treatment of Schizophrenia normally costs in between $32.5 and $65 billion per year that badly affects the economy.
(v) Schizophrenia is uncommon in children. The National Institute of Mental Health (NIMH) assesses that just 1 of every 40,000 children encounter the beginning of schizophrenia manifestations before 13 years of age.

15.1.2.1 DSM-V Definition and Diagnosis Criteria of Schizophrenia

Schizophrenia is an acute and persistent mental disorder that can be classified by disruption in thought, insight and conduct. A variety of apprehension, behavioral, and psychological manifestations are included in Schizophrenia. Its diagnosis is quite difficult also. Clinical or physical tests to diagnose schizophrenia are yet to be discovered. The diagnosis includes identification of a constellation of manifestations that are fatalistically influencing social or professional operating. As per the Mental Disorders' Diagnostic and Statistical Manual, Fifth Edition (DSM V), schizophrenia has the lifelong pervasiveness of around 0.3–0.7%. The deranged characteristics of schizophrenia generally show up during the age of mid-teenagers to mid-thirties. The pinnacle age of onset of experiencing schizophrenia for the first time is in the age of early to middle of twenties for males while in females in the late twenties.

The DSM-V criteria for diagnosis of schizophrenia are:

If any two or more symptoms of the following persists for a month or longer time period, and as a minimum of one of them must be A, B or C

a. Delusions
b. Hallucinations
c. Unorganized talk
d. Abnormal or catatonic behavior
e. Detrimental indications, like lessen emotional expression.

A. Deterioration in any of the main areas of operation for particular time duration from the inception of the disorder: Profession, personal relationships, or personal care.
B. Few symptoms of this ailment must persist for an unceasing time interval of 6 months or more. This complete period must have at least symptoms for one month (or less if the person is under treatment) that comply with the criterion A (is called active phase symptoms) and can have intervals of remaining symptoms. In the remaining time periods, there may be only negative symptoms.
C. Schizoaffective problem and depressive or bipolar problem with deranged characteristics have been solved:
   - Any depressive or manic event must not be experienced concurrently with symptoms of active phase
   - If depressive or manic event has been occurred during symptom of active phase, these have been persisted for a minimum of the overall time span of the active and residual phases of this disorder [2].

15.1.2.2 Types of Schizophrenia

Schizophrenia can be classified into five types. They are classified by the kinds of indications the individual shows when they are evaluated:

(i) Paranoid schizophrenia: Paranoid behavior, including delusions and auditory hallucinations
(ii) Disorganized schizophrenia: Inappropriate emotions or reactions
(iii) Catatonic schizophrenia: State nothing for a considerable length of time, or they may rehash all that you state or do

(iv) Undifferentiated schizophrenia: Delusions, hallucinations, muddled discourse or conduct, mental conduct
(v) Residual schizophrenia: Currently has no symptoms.

15.1.2.3 Causes of Schizophrenia

It is still not known exactly what causes schizophrenia. There are many researches that conclude a mixture of genetic, physical, psychological and environmental aspects can lead a person towards the condition of schizophrenia. The causes for schizophrenia may be different from individual to individual.

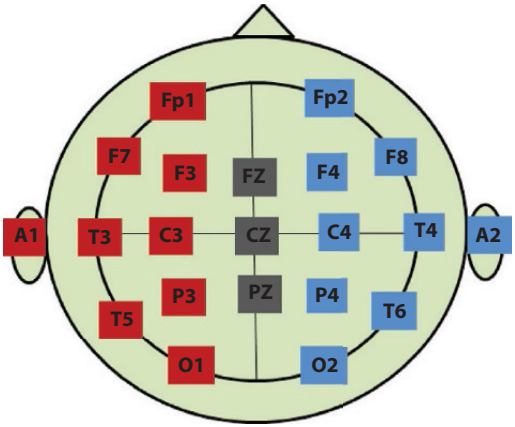
15.1.2.4 Symptoms of Schizophrenia

Numerous individuals with schizophrenia don't show up sick. Be that as it may, numerous conduct changes will make the individual appear 'off' as the malady advances. The symptoms are: social disconnects uneasiness, misapprehension, delirium, irrationally anxious or sentiments of mistreatment, loss of appetite or skipping meals, lack of cleanliness, etc.

Schizophrenia is a mainstream psychiatric problem that influences 1 of every 100 individuals worldwide and gives many burdens on the person. People influenced for the most part will have a low quality of life, expanded bleakness and mortality, upsets relational connections and family structures, and has noteworthy financial expenses to society. There are an increasing number of patients in Malaysia which affects social functioning. EEG is used in schizophrenia patients to measure the brain's response and the wave's tone that is different in length. Mismatch field (MMF) in the event-related potential (ERP) of the patient is reduced compared to a healthy person. ERP is used in measuring brain response because of the output of some particular sensory cognitive.

15.1.3 Electroencephalograph (EEG)

Electroencephalograph (EEG) is an amazing clinical methodology that can recognize, observe and register electrical exercises of the mind. In spite of the fact that EEG delivers high caliber in catching worldly data and less expensive contrasted with utilitarian Magnetic Resonance Imaging (fMRI) and Positron Emission Topography (PET), the spatial goals in EEG is low. Thereby EEG becomes the utmost reasonable clinical methodology for examining remembering, observation, feeling, learning, and data handling undertakings.



KEY WORDS	
F	Frontal Lobe
T	Temporal Lobe
C	Central Lobe
P	Parietal Lobe
O	Occipital Lobe
Z	Electrode Place on Midline
A	Ear Clip reference node

Figure 15.1 Electrode placement method.

Electroencephalography is a method in which electrical signals of the brain are measured using a special device Electroencephalogram (EEG). Electrodes, amplifiers, conductive gel, and Analog to Digital Converter are the main components of an EEG device. Electrical reactions from the brain are conducted using the electrodes or leads. For EEG analysis, many types of electrodes are generally used.

EEG is a non-intrusive strategy for diagnosis that can be sheltered to be utilized in the body of a human and may be recorded in a given time span. The Electroencephalogram is the recorded EEG. One of the strategies by the International Federation of EEG Societies to record EEG signal is setting up a placement of 10–20 EEG electrodes. There are four standard focuses in which a human head can be separated: the left preauricular points (ears), the right preauricular points, the nasion (nose), and the inion (external occipital protuberance or projection). Figure 15.1 shows picture of standard EEG electrode attachment in the brain involved while taking the EEG recording.

15.1.3.1 Characterizations of EEG Signals

Frequency, amplitude, shape, and position of the electrodes on the scalp are utilized to commonly arrange the EEG waveforms. For determining normal or abnormal rhythms, frequency is taken as the basic unit. Based on signal frequency, the waveforms are classified such as- Alpha, Beta, Gamma, Theta, and Delta. Identifying some waves is done based on their shape, head distribution and symmetry property. Some specific pattern of waveform is considered common at certain age, situation of vigilance, and sleep. The continuous rhythms of the brain waves are divided by the bands

of frequency. Brain wave frequency relates to various behaviors and mental conditions of the human mind.

a. Delta (0.1–4 Hz)
   This is considered that the highest in amplitude are the Delta waves but are the slowest. Gray matter of the mind is represented by it. Almost in all stages of sleep especially in stages 3 and 4, this slowest brain wave activity is recorded. In infants, it is common and paramount rhythms and seldom identified by expert intermediary. In adults who are not sleeping, Delta waves are more common, and it also instigates hormone growth.
b. Theta (4–8 Hz)
   The Theta waves have their range between 4 and 8 Hz which is also associated with subconscious activity. When someone is in deep relaxation or meditation, these types of brain waves are observed. These brain waves or signals influence on spiritual related researches in getting extrasensory skills that can be achieved through meditations by reaching the activity of theta level. This level can also be utilized by the psychotics to collect information regarding awareness. It is uncommon in aged persons but it is quite normal for children under the age of 13 years. It influences the production of hormone for human growth, the cortical hormone that helps in improving learning and remembrance, and serotonin hormone that maximizes the level of relaxation to have comfort from pain.
c. Alpha (8–13 Hz)
   Alpha waves are from the range 8–13Hz. The Alpha waves are something that can be found across all age groups usually seen in adults, those are in eyes closed mode but are awake and in relaxation. Both the side of the head can be affected by it but slight higher amplitude is found in non-dominant portion. The occipital and parietal areas of a human brain are the regions from where it can be recorded. The white matter of the human brain is represented by it. It plays the role of connector between conscious and subconscious parts of mind.
d. Beta (13–30 Hz)
   Beta waves are from the range 13–30 Hz. Behavior and actions are the two most important aspects that are related

to Beta waves. These waves are related to all the five senses of what we hear, see, smell, touch, and taste. Both portions of frontal and parietal lobes can encounter this. Beta wave instigates the creation of Cortisol. For the age accelerating hormone in the mind, it is considered to be the main factor and also links with learning and remembrance. These waves can arise in the state of mindfulness such as talking, analyzing, solving some problems, decision making, and judgment. Beta waves can be identified around cortical actions and it has the ability to think and react.

e. Gamma (30–100 Hz)
The Gamma waves have their range between 30 and 70 Hz, and it is concerned with observation and consciousness. These waves are found to be of innate capacity and a balanced mix of tactile sources of information. The senses and memory are together perfectly consolidated by Gamma waves.

Figure 15.2 shows the characteristics of delta, theta, alpha, beta and gamma wave.

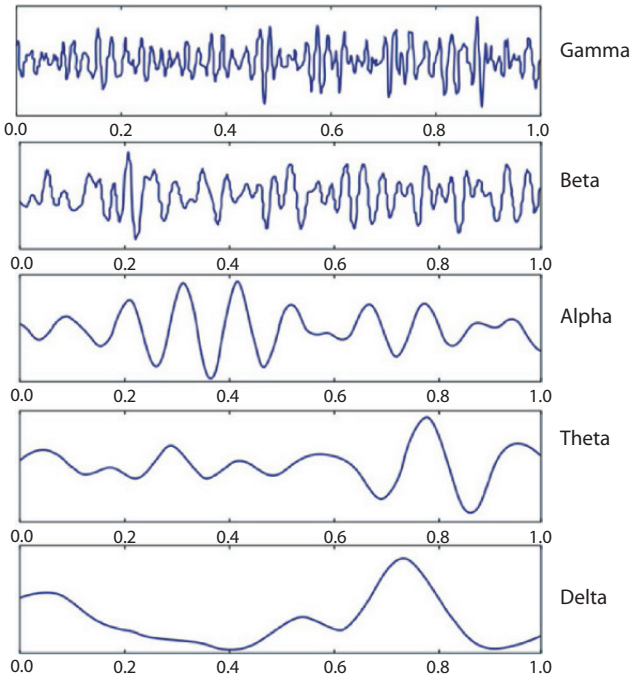


Figure 15.2 EEG delta, theta, alpha, beta and gamma [7].

15.2 Methodology

The general procedure followed when classification of Schizophrenia and healthy controls are done is represented in Figure 15.3.

15.2.1 EEG Signal Processing

The first step is always recording of EEG signal which is done using channel placed on scalp using standard 10–20 system or any other international electrode placement system is followed.

15.2.2 Removing the Artifacts

Whenever the recording of EEG is done, there are chances of EEG signal getting contaminated by external noise like electrical signal, body movement, biological activity (muscle movement, eye blinks). For accurate analysis of EEG signal, it is very important to remove the noise from the EEG signal.

Thus, the second step in analyzing EEG signal is to eliminate the noise while enhancing the signals of brain. There are a few procedures which

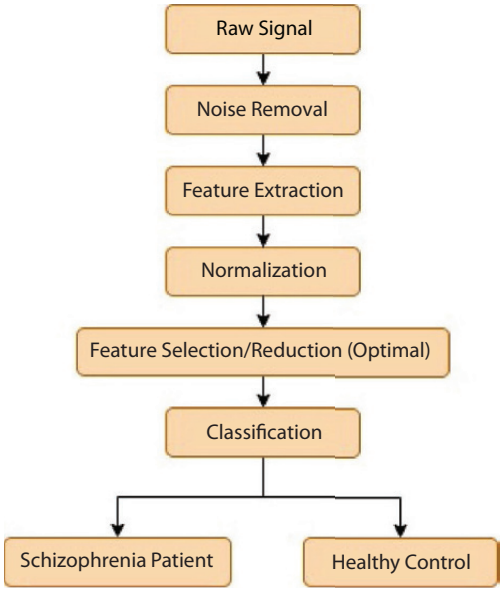


Figure 15.3 Steps involved in processing of EEG signal.

have been utilized in isolating and evacuating the artifacts, including notch filter and fixed linear filter. For filtering the line interference noise, notch filter is utilized, in situations where the EEG's data acquisition system is unable to cancel out the line frequency of 50 Hz.

Independent Component Analysis (ICA) is one of the component-based methods that are rigorously utilized in analyzing the patterns and analysis of bio-signals. By decomposition of the signals into several independent components depending on the signal's statistical independence, the noises are separated from the EEG signals using ICA. The most advantageous aspect of ICA is any additional channel reference is not required because a priori information is not needed by the algorithm itself. But, ICA needs a visual inspection that is completely manual and identify the artifact independent components and make it time consuming and subjective.

The adaptive filtering method is used to filter coefficients that are updated utilizing a recursive algorithm based on a given property of signal to eliminate the noises. A reference signal is required, which carries significant data and statistical properties of the artifacts. Until the output has been minimized, the filter coefficients are fine-tuned in an ongoing basis. Due to the fast convergence of adaptive filtering method, it is well suited for online applications.

15.2.3 Feature Extraction

Different types of features can be extracted from EEG signal. Various approaches like frequency-domain analysis, time-domain analysis, and time-frequency domain analysis are used for feature extraction.

The various features such as standard deviation, min, max, mean and energy are commonly used in time-domain analysis. The time-domain approach has some inherent disadvantages such as: very high susceptibility of identified characteristics and the need for huge storage are.

By utilizing the Fast Fourier Transform (FFT) in frequency-domain analysis, the EEG signals are converted into frequency domain. In certain cases while using merely frequency information, providing signal characteristic for classification is not enough. Thus a substitute to improve the classification performance is the time-frequency analysis.

There are two divisions possible for the features: linear feature and non-linear feature. Linear analysis comprises of frequency and time domain features like band power, inter-hemispheric asymmetry, EEG recordings (frequency, amplitude, and power). Non-linear features have the ability analyze the chaotic behavior of the brain. Non-linear features

include Lyapunov Exponent, Higuchi's Fractal Dimension, Approximate Entropy, Correlation Dimension, etc.

15.2.4 Normalization

Various types of time domain, frequency domain features as well as non-linear and linear features are retrieved from the EEG signal. All these feature values needs to be normalized in the same range to be further analyzed. So normalization is an important step during the analysis. Normalization can be done in a number of ways of which the most commonly used ones are min-max normalization and z-score normalization.

15.2.5 Feature Selection/Reduction

During the feature extraction phase, quite a big number of features can be extracted from the EEG. From all the features, only the significant features needs to be selected or feature needs to be reduced into smaller number so that less amount of computation is required for analysis as well higher accuracy is achieved. So when the large number of features is there, then only feature selection or reduction step is required.

15.2.6 Feature Classification

In feature classification state, the selected features from EEG signals will be used as input to classifier models. The number of features, linearity between input and output, and data distribution, are among the factors, which will affect the accuracy of the classification. Some of important classifiers are discussed below:

a. Support Vector Machine (SVM)
   This SVM technique is broadly utilized in classification of data, both non-linear and linear. By utilizing a mapping of non-linear type, the real training data gets transformed into a quite larger dimension by SVM. In this newly specified dimension, a linear differentiating hyperplane is searched for that must be optimal. Data of two different classes can be distinguished by a hyperplane, employing an appropriate non-linear mapping into a very large dimension. The time taken by SVMs for training is very slow; but the method is quite accurate because they are able in modeling nonlinear decision boundaries that are too complex. SVMs normally

don't lead to over-fitting problem as compared to other techniques. SVMs can also perform numeric prediction and also the learned model can be depicted by them very briefly. The areas where SVMs can be applied are: recognition of handwritten digit, identification of speaker, object identification, tests for predicting benchmark time-series.

b. Decision Tree (DT)

It is similar to a flowchart-like tree structure. Each non-leaf node shows an examination on an attribute, each branch shows an output of the examination, and a class label is stored by each leaf node. The topmost node in a tree is called a root node.

For tuple, *T*, the related class label is unknown; the attribute values of the tuple are examined against the decision tree. The path from root to a leaf node is tracked, where the tuple's class prediction is stored. It is quite easy to convert decision to classification rules according to the requirement. It is best suited for exploratory knowledge mining as no details regarding domain or setting of parameter is needed to create decision tree classifier. Multidimensional data can be very well managed by decision trees. The knowledge demonstration done by the decision tree is very user friendly. It is very easy to learn the steps of classification done by decision tree and also they are very accurate. There are various application areas where decision trees can be implemented such as: production of medicine, analyzing finance, astronomy, system for commercial rule induction, molecular biology, etc.

c. *k*-Nearest-Neighbor Classifier (kNN)

This is the widely used method for pattern recognition. This classifier compares a test tuple with training tuples dataset that employs similarity, and this is called learn by analogy. Let *n* attributes are used to denote training tuples. In an *n*-dimensional region, each tuple denotes a point. So, following this process, every training tuples are stored in a pattern region that is *n*-dimensional. If an unidentified tuple is encountered, a pattern region is looked for the k training tuples by *k*-nearest-neighbor classifier keeping in mind that it must be the closest one to the tuple that is unidentified. So all of these *k* training tuples are considered as *k* "nearest neighbors" of the unidentified tuple.

d. Artificial Neural Network (ANN)
It is a computational model which depicts the biological neural networks structure and its operations. ANN can be considered as the parts of a computing system that is made for simulating the course of action a human brain undergoes for analysis and processing of information. An ANN can be thought of as an oriented graph. It comprises of nodes that is used to denote neurons, joined by arcs which is associated with a weight. It also has dendrites and synapses. It implements the input values of nodes and establishes an activation function with the incoming arcs, which is balanced by the weights of the arcs. ANN has a very important advantage that it can self-learn from observing data sets. So it can be used as a tool for random function approximation. ANN is quite less expensive and time saving method as it takes input as samples of data and not the complete data set for reaching at a point of solution.

e. Probabilistic Neural Network (PNN)
It is a feed-forward neural network, and extensively utilized in problems like pattern recognition and classification. In PNN algorithm, to approximate the parent probability distribution function (PDF) of each class, a Parzen window and a non-parametric function are utilized. The probability of a class for a newly input can be approximated by utilizing each class's PDF, and Bayes' rule is used for input data that is new to allocate it the highest posterior probability to the. Mis-classification probability is reduced by using this technique. Various operations in a PNN are arranged in a multilayered feed-forward network that includes four layers: i) Input layer, ii) Output layer, iii) Pattern layer, and iv) Summation layer.

f. Convolutional Neural Network (CNN)
It is considered to be a class of deep neural networks. It is used to analyze visual imagery. They often called shift invariant or space invariant artificial neural networks (SIANN). CNNs can be seen as versions of multilayer perceptron which means completely connected networks. Neuron of one layer is connected to all the neurons of other layer. This leads to over-fitting of data. CNNs are based on biological processes. CNNs require much lower pre-processing as than other algorithms of image classification.

g. Random Forest
It is a supervised learning method for classification, regression and all the other jobs that can be carried out by creating a numerous decision trees that operates like an ensemble. The main use of this algorithm is to solve classification problems. In the random forest, each decision tree is used to perform a class prediction and the highest voted class is termed as the model's prediction. It averages the results of different decision trees and decreases the over-fitting problem. It is very flexible and has maximum accuracy even if a huge amount of data is missing. Random forest is suitable for large data item ranges and has comparatively less variance than a single decision tree.

15.3 Literature Review

In the present state of the art, an extensive amount of study has been done related detection of Schizophrenia using EEG signal. Some of the literature has been discussed in the table given below.

15.4 Discussion

From Table 15.2, it can be found that higher classification accuracy is obtained when Random forest, SVM and CNN is used as classifier. Highest classification of 96.77% is achieved when random forest is used. Particularly it can also be observed the now a days more and more research is being done the field deep neural network and more particularly CNN because its ability to extract and select features automatically. It can also be observed that CNN provides high classification accuracy. So for future works, on could go for implementing CNN as well as SVM and Random Forest. One could for analyzing more and more linear and non-features to get the more significant features which could be able to provide higher accuracy.

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia.

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
1	[3]	In this study, EEG spectra along with Random Forest classifier have been utilized to differentiate between paranoid schizophrenic and non-schizophrenic people. An open dataset was used that had 499, one min epoch of 28 subjects (14 healthy controls + 14 paranoid schizophrenic).	Total 28 participants. 14 patients (7 females + 7 males) and 14 healthy controls (7 females + 7 males) from Institute of Psychiatry and Neurology, Poland [4]. Recording done for 15min at eyes closed resting state sitting position. Sampling rate = 250 Hz the ICD-10 diagnosis F20.0	19 EEG Channels	EEG spectra	Random Forest Classifier	96.77%

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
2	[5]	In this study, features used were from Energy, Entropy, Power spectral density, Contrast from Gray Level Co-occurrence Matrix (GLCM) which is based on image processing method. In this model technique used for classification is Fuzzy Mathematical model of Emotional Learning (FMEL)	Total 27 participants (All males). 16 patients & 11 age-matched controls were recruited from major psychiatric hospital in Western Australia based on DSM-IV and ICD-10 criteria. Resting in sitting posture eyes closed 4 min EEG data. Sampling rate = 200 Hz	20 Electrodes	Power spectral density, Energy, Entropy, and Contrast were extracted from Gray Level Co-occurrence Matrix (GLCM) from spectrogram image	Fuzzy Mathematical model of Emotional Learning (FMEL)	81.5%

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
3	[6]	In this study, 157 non-linear features were extracted and t-test was used for feature selection (14 features). EEG signal was segmented into non-overlapping 25 s of epoch which lead to 6,250 samples. 5 different classifiers were used i.e. DT, LDA, KNN, PNN and SVM.	Total 28 participants. 4 patients (7 females + 7 males) and 14 healthy controls (7 females + 7 males) from Institute of Psychiatry and Neurology, Poland [7]. Recording done for 15 min at eyes closed resting state sitting position. Sampling rate = 250 Hz	19 Channels	Entropy, Kolmogorov Sinai entropy, Hjorth complexity and mobility, Shannon, cumulant, permutation entropy, Largest Lyapunov Exponent, Kolmogorov complexity, bispectrum, Rényi and Tsallis	Feature selection using t-test and classification using DT, LDA, KNN, PNN & SVM	92.91% SVM (RBF)

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
4	[9]	In this study, Pearson Correlation Coefficient is applied among the channels to find the relations between the channels. Only selected channel's information was sent to Convolutional Neural Network (CNN) for classification. 16 * 7,680 samples was used.	Total 84 participants. 45 patients and 39 healthy controls Recording done for 1min at eyes closed resting state sitting position. Sampling rate = 128 Hz [8].	16 Channels	Coefficient matrix as image	Convolutional Neural Network (CNN)	90%

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
5	[10]	EEG signals were recorded by showing the universal affective picture framework. The features utilized were entropy and classifier used was SVM. Signals were grouped based on 3 three kinds of emotions and five frequency groups. Absolute scores of positive and negative condition scale's correlation and regression were utilized.	Total 44 participants.10 healthy controls (4 males, 6 females), 17 moderately ill subjects (8 males, 9 females) and 17 markedly ill subjects (9 males, 8 females) from, Taichung Veterans General Hospital, Chiayi, Taiwan. Recording was done with visual stimulation. Sampling rate = 256 Hz	3 Channels (Fz, Pz, Cz)	frequency, and entropy	SVM	81.5%, (normal and markedly ill) and 79.5% (moderately and markedly ill)

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
6	[11]	Working memory job was provided to patients with schizophrenia and healthy control subjects. This helps the health provider to look at essential somatosensory encoding of vibrotactile upgrades. The results show that the neural entrainment of vibrotactile stimuli in necessary somatosensory cortex is hampered in patients of schizophrenia. Neural oscillatory also connects of dynamic stimuli data were decreased in patients during working memory upkeep.	Total 21 participants. 9 patients and 12 healthy control subjects from outpatient clinic of the Charité University Hospital (Psychiatry Department), Berlin	68 Channels	EEG Somatosensory Steady-state evoked potential (SSEP), induced beta-band EEG oscillations	Statistical analysis	SSEPs were found to be less in schizophrenic patients during vibrotactile stimulation than in healthy patients. Lesser prefrontal beta power modulation was found schizophrenic patients compared to normal subjects

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
7	[12]	In this study, deep learning framework has been modified by using Random forest in place of softmax layer and an additional layer is for voting is added on top of the neural network to help in the classification of 3 groups i.e. first episode subjects (FES), characteristics in high risk (CHR) subjects and healthy controls (HC)	Total 140 participants. 40 CHR patient, 40 1st episode schizophrenia (FEH) and 40 healthy controls (HC). The data was recorded from the outpatients at Shanghai Mental Health Center. Recording was done eye-open resting conditions, each for 300 s at sampling frequency of 1,000 Hz	64 Channels	Features are extracted automatically using the Modified deep learning framework	Convolution Neural Network Voting-Random Forest (CNNV-RF)	CHR:81:6% FES: 96.7% HC: 99:2%

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
8	[13]	In this study, EEG was recorded based on response based stimuli using Time frequency feature representation and using Real Time ANOVA and Regression analysis for classification and severity scaling respectively. Single channel F2 was selected for analysis as it worked the best among all channels	Total 50 participants. 25 patients and 25 healthy controls Recording based on the response from the stimulus. Total 78 stimuli was presented for 150 ms with an in between interval of 1 s. Sampling rate = 512 Hz [14]	64 Channels	Time Frequency representation using Stockwell transform at a specific time frame i.e. after presentation of stimuli	Real Time ANOVA, Regression based correlation	91.5% (F2) (Classification) 71.7% (scaling of severity)

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
9	[15]	In study, EEG as well fMRI recording was done. Analysis was done using correlation between Joint Independent Component score and schizophrenia symptoms standards scales.	Total 53 Participants. 30 patients, 23 Healthy controls. EEG as well fMRI recording was done at the same time and subjects were asked to be in resting state with eyes open condition to look at the cross symbol and at the same time listen to the tones and press a button. Sampling frequency = 250 Hz	32 Channels	Joint Independent Component score (JICs) between "N100' and "P200"	Correlation	Correlation was found between "N100" and "P200" JICs and correlated "N100" and "P200" JICs to Scale for the Assessment of Negative Symptoms & Scale for the Assessment of Positive Symptoms

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
10	[16]	In this study, Ultimatum Game (UG).was used for the analysis of mental status of patients based on social decision-making was checked. The EEG data was recorded while subjects were asked to play Ultimatum Game on the system and the Event-Related Potential (ERP) was recorded. Statistical analysis was done based on ERP	Total 35 Participants. 16 patients, 19 Healthy controls from Mental Health Network Switzerland. The EEG data was recorded while the subjects were asked to Ultimatum Game and the ERP was recorded.	128 Channels	Local Autoregressive Average Regularization Approach (LAURA), ERP (P2 and feedback-related negativity (FRN),	Statistical Analysis (ANOVA)	Significant difference was found in feedback-related negativity between Schizophrenia subjects and healthy individual

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
11	[17]	In this study, SVM & Gaussian processes classifiers (GPC) were used as classifier and Average ERP, mismatch negativity (MMN) response, stimulus paradigm were used as attribute.	Total: 21 patients (6 females, 15 males) and 22 healthy controls (8 females, 14 males) from Royal Brisbane Hospital, Australia. Recording was done using auditory-odd ball test which consisted of sequence of short auditory stimuli at an interval 500 ms. Sampling frequency = 500 Hz	64 Channels	Average ERP, mismatch negativity (MMN) response, stimulus paradigm	Statistical parametric mapping (SPM) SVM & Gaussian processes classifiers (GPC)	80.49% (GPC), 78.05% (SVM)

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
12	[18]	In this study, statistical analysis (TANOVA and ANOVA) was done to find the similarity between 22q11.2 deletion syndrome and schizophrenia using EEG microstates.	Total 112 Participants. 30 patients with 22q11DS, 28 Healthy controls from Switzerland Total 54 Participants. 27 patients, 27 Healthy controls from Psychosocial Rehabilitation Centre in Tbilisi, Georgia. Recording was done for 5 min resting state eyes closed state. Sampling frequency = 500 Hz	256 Channels	EEG microstates	Statistical analysis (TANOVA and ANOVA)	Microstate class showed the similarity between subjects with 22q11DS and schizophrenia
13	[19]	In this study, cross-correlation between the Multiscale entropy, Lempel–Ziv complexity and Lyapunov exponent was computed to find the characteristics of the schizophrenia patients.	Total 84 participants. 45 patients and 39 healthy controls Recording done for 1 min at eyes closed resting state sitting position. Sampling rate = 128 Hz [8]	16 Channels	Entropy, Lempel–Ziv Complexity, Lyapunov Exponent	cross-correlation	Cross-correlation value for Lempel–Ziv was found to be decreed in schizophrenia patient in left half of the brain as compared to right half

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
14	[20]	In this study, dimension complexity of EEG signal in spatial domain is analyzed for schizophrenia patient as compared to normal subjects	Total 36 participants. 18 male patients and 18 male healthy controls Recording done in eyes closed resting. Sampling rate = 100 Hz	14 Electrodes	correlation dimension,	Statistical Analysis	Schizophrenia patients have lower CD compared to healthy subjects.
15	[21]	Lyapunov exponent by Wolf's method, Kantz method and Rosenstein method was computed and compared for identification of schizophrenia patients. The study was to reflect the chaotic nature of EEG signal which was present there in schizophrenia patients.	Total 84 participants. 45 patients and 39 healthy controls Recording done for 1 min at eyes closed resting state sitting position. Sampling rate = 128 Hz [8]	16 Channels	Lyapunov exponent by Wolf's method, Kantz method and Rosenstein method Spectrum of computed exponents	Statistical Analysis (t-test)	Lyapunov exponents value was lower in all electrodes for schizophrenic patients as compared to healthy controls. Largest Lyapunov exponent reflected by schizophrenia patients was computed using Rosenstein method

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
16	[22]	In this study, features used N100, P200 and Peak-to-Peak P300-N100 from ERP EEG signal which were further utilized in the classifier Random forest for identification of schizophrenia patients.	Total 81 participants. 49 patients and 32 healthy controls Recording was done while subjects were asked to do sensory task like pressing the button to generate a tone [23]	64 Channels	N100, P200 and Peak-to-Peak P300-N100.	Random Forest	81.10%
17	[24]	In this study, 2 modified odd ball pictures test has been used during EEG recording to check the mental status of the subject.	Total 78 participants. 55 patients and 23 healthy controls	23 Channels	Spectral entropy, Information entropy, Kolmogorov complexity, Approximate Entropies, Shannon entropy and Higuchi's Fractal Dimension	Neural Network, SVM	88.5 % (SVM)

(*Continued*)

Table 15.2 Literature Review on the Analysis of EEG Signal for Detection of Schizophrenia. (*Continued*)

S. N.	Ref.	Methodology Used	Details of Data Set	No. of Electrodes Used	Features	Analysis Methodology	Accuracy
		The features extracted are Spectral entropy, Information entropy, Kolmogorov complexity, approximate Entropies, Shannon entropy and Higuchi's Fractal Dimension which are further utilized in the classifier to identify the schizophrenia patients					
18	[25]	In this study, 11-layered CNN are used for detection of schizophrenia patients	Total 28 participants. 14 patients (7 males, 7 females) and 14 (7 males, 7 females) healthy controls from Institute of Psychiatry and Neurology in Poland [5]	19 Electrodes	Automatic feature extraction at convolution stage	Convolutional Neural Network (CNN)	Non-subject based testing - 98.07% Subject based testing - 81.26%

15.5 Conclusion

From the study it can be analyzed that EEG signal has the potential for identifying biomarker for identification of Schizophrenia. Hence more research needs to be done in order to develop a framework for identification of schizophrenia with very high accuracy. Thus development of such a framework would help in in detection of Schizophrenia in the early stage detection of schizophrenia. The framework could act as an adjunct tool for detection of Schizophrenia which would help in the betterment of the condition of life of the patient as help the society as whole.

References

1. https://www.who.int/news-room/fact-sheets/detail/schizophrenia.
2. American Psychiatric Association, *Diagnostic and statistical manual of mental disorders*, 5th edition, American Psychiatric Association, Washington, DC, 2013.
3. Buettner, R., Beil, D., Scholtz, S., Djemai, A., Development of a Machine Learning Based Algorithm to Accurately Detect Schizophrenia based on One-minute EEG Recordings. *Proceedings of the 53rd Hawaii International Conference on System Sciences*, pp. 3216–3225, 2020, https://doi.org/10.24251/hicss.2020.393.
4. Olejarczyk, E. and Jernajczyk, W., Graph-based analysis of brain connectivity in schizophrenia. *PLoS One*, 17, 12, 11, e0188629, 2017, https://doi.org/10.1371/journal.pone.0188629.
5. Khasraghi, B.J., Setayeshi, S., Price, G., Applying brain emotional learning based fuzzy inference system for EEG signal classification between schizophrenics and control participant. *2017 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology, CIBCB*, pp. 1–8, 2017, https://doi.org/10.1109/CIBCB.2017.8058555.
6. Jahmunah, V., Oh, S.L., Rajinikanth, V., Ciaccio, E.J., Cheongf, K.H., Kumar, N.A., Acharya, U.R., Automated detection of schizophrenia using nonlinear signal processing methods. *Artif. Intell. Med.*, 100, 101698, 2019.
7. Abhang, P.A., Gawali, B.W., Mehrotra, S.C., Technological Basics of EEG Recording and Operation of Apparatus, in: *Introduction to EEG- and Speech-Based Emotion Recognition*, pp. 19–50, Elsevier, London, UK, 2016, http://brain.bio.msu.ru/eeg_schizophrenia.htm.
8. Naira, C.A.T. and Del Alamo, C.J.L., Classification of people who suffer schizophrenia and healthy people by EEG signals using deep learning.

Int. J. Adv. Comput. Sci. Appl., 10, 10, 511–516, 2019, https://doi.org/10.14569/ijacsa.2019.0101067.

9. Chu, W.L., Huang, M.W., Jian, B.L., Cheng, K.S., Analysis of EEG entropy during visual evocation of emotion in schizophrenia. *Ann. Gen. Psychiatry*, 16, 1, 1–9, 2017, https://doi.org/10.1186/s12991-017-0157-z.
10. Ludwig, S., Spitzer, B., Jacobs, A.M., Sterzerd, M.S.P., Blankenburg, F., Spectral EEG abnormalities during vibrotactile encoding and quantitative working memory processing in schizophrenia. *NeuroImage: Clin.*, 11, 578–587, Elsevier, 2016.
11. Chu, L., Qiu, R., Liu, H., Ling, Z., Zhang, T., Wang, J., Individual Recognition in Schizophrenia using Deep Learning Methods with Random Forest and Voting Classifiers, *Insights from Resting State EEG Streams*., arXiv, 1707.03467 1–7, 2017.
12. Dvey-aharon, Z., Fogelson, N., Peled, A., Intrator, N., Schizophrenia Detection and Classification by Advanced Analysis of EEG Recordings Using a Single Electrode Approach. *PLoS ONE*, 10, 4, 1–12, 2015, https://doi.org/10.1371/journal.pone.0123033.
13. Fogelson, N., Shah, M., Scabini, D., Knight, R.T., Prefrontal cortex is critical for contextual processing, Evidence from brain lesions. *Brain*, 132, 3002–3010, 2009.
14. Forda, J.M., Roacha, B.J., Palzesa, V.A., Mathalon, D.H., Using concurrent EEG and fMRI to probe the state of the brain in schizophrenia. *NeuroImage: Clin.*, 12, 429–441, Elsevier, 2016.
15. Horat, S.K., Favre, G., Prévot, A., François, J.V., Herrmann, R., Gothuey, I., Merlo, M.C.G., Missonnier, P., *Impaired social cognition in schizophrenia during the Ultimatum Game: An EEG study*, National Library of Medicine, National Center for Biotechnology Information, Amsterdam, Netherlands 2018.
16. Taylor, J.A., Matthews, N., Michie, P.T., Rosa, M.J., Garrido, M.I., Auditory prediction errors as individual biomarkers of schizophrenia. *NeuroImage: Clin.*, 15, 264–273, Elsevier, 2017.
17. Tomescu, M.I., Rihs, T.A., Roinishvili, M., Karahanoglu, F.I., Schneider, M., Menghetti, S., Ville, D.V.D., Brand, A., Chkonia, E., Eliez, S., Herzog, M.H., Michel, C.M., Cappe, C., Schizophrenia patients and 22q11.2 deletion syndrome adolescents at risk express the same deviant patterns of resting state EEG microstates: A candidate endophenotype of schizophrenia. *Schizophr. Res.: Cogn.*, 2, 159–165, Elsevier, 2015.
18. Kutepov, E., Krysko, V.A., SPavlov, S.P., Zigalov, M.V., Papkova, I.V., Saltykova, O.A., Yaroshenko, T.Y., Krylova, E.Y., Yakovleva, T.V., Dobriyan, V.V., Erofeev, N.P., Complexity of EEG Signals in Schizophrenia Syndromes. *29th International Conference on Computer Graphics and Vision, At Bryansk, Russia*, 2019.

19. Lee, Y.J., Zhu, Y.S., Xu, Y.H., Shen, M., Zhang, H., Thakor, N.V., Detection of non-linearity in the EEG of schizophrenic patients. *Clin. Neurophysiol.*, 112, 1288–1294, Elsevier, 2001.
20. Kutepov, I.E., Dobriyan, V.V., Zhigalov, M.V., Stepanov, M.F., Krysko, A.V., Yakovleva, T.V., Krysko, V.A., EEG analysis in patients with schizophrenia based on Lyapunov exponents. *Inf. Med. Unlocked*, 18, 100289, Elsevier, 2020.
21. Zhang, L., EEG Signals Classification Using Machine Learning for the Identification and Diagnosis of Schizophrenia. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, pp. 4521–4524, 2019, https://doi.org/10.1109/EMBC.2019.8857946.
22. https://www.kaggle.com/broach/button-tone-sz/home.
23. Thilakvathi, B., Shenbaga Devi, S., Bhanu, K., Malaippan, M., EEG signal complexity analysis for schizophrenia during rest and mental activity. *Biomed. Res. (India)*, 28, 1, 1–9, 2017.
24. Oh, S.L., Vicnesh, J., Ciaccio, E.J., Yuvaraj, R., Acharya, U.R., Deep Convolutional Neural Network Model for Automated Diagnosis of Schizophrenia Using EEG Signals. *Appl. Sci.*, 9, 14, 2870, 2019, https://doi.org/10.3390/app9142870.

16

Performance Analysis of Signal Processing Techniques in Bioinformatics for Medical Applications Using Machine Learning Concepts

G. Aparna[1], G. Anitha Mary[2] and G. Sumana[3]*

[1]ECE Department, University College of Engineering, Osmania University, Hyderabad, Telangana, India
[2]MCA Dept., Loyola Academy, Old Alwal, Hyderabad, India
[3]Command Control Center, Sri Padmavati Mahila University, Tirupati, India

Abstract

Bioinformatics is an emerging multidisciplinary area that is evolving into a discipline of high demand for research scope with the assistance of computer scientists. With the aid of computers demands them to accelerate the work for accessing the details easily and in a faster manner. The term Bioinformatics is a union of two phases Biology and Information Sciences. Bioinformatics is an interdisciplinary area for research that requires the knowledge of computer & information sciences to extract the outcome to biological problems.

The researchers in this field identify the biological issues in living organisms. They then try to unravel the biological unsolved issues that may be the cause of the problems. By using computational approaches also known as the in-silico approaches. Thus, Bioinformatics is currently one among the several thrust domains of research in Computer Science & details Technology.

Genomic Signal Processing (GSP) is an evolving area in bioinformatics that uses genomic data as input to perform logical analytics applying the Digital Signal Processing (DSP) procedures. In this process the protein data sequences denoted in the form of characters are mapped into digital representation termed as genomic signal. These genomic signals are further processed to achieve the biological knowledge required for the analysis. The concepts depicted in the literature

**Corresponding author*: sumanaspmvv@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (391–430) © 2021 Scrivener Publishing LLC

of the GSP has generated prolific interest in this area. Hence bioinformatics is considered as an integrative field which includes the computational procedures for approximate and speedy exegesis of bio data.

Reliable and economical techniques for distinctive hotspots identification and their respective position in proteins area unit extracting are being pursued.

The evolution of hotspot identification for genomic sequences is an evolutionary progress through digital signal processing techniques, efficient identification of hotspots requires more attention and further improvements. Sequence discriminating DSP technique, initial protein sequence is to be portrayed into number sequences to remodel it in the signal form of notation. Transforming techniques are performed in spatial domain for feature extraction of the hotspot and statistical analysis is done so as to understand the performance metrics using Machine Learning concepts. The numerical representation, mapping process and transforming techniques play a vital role in the complete process.

Keywords: Bioinformatics, proteins, hotspots, EIIP, CPNR, DWT, accuracy, F1 score, macro average (avg), weight average (avg), precision, recall

16.1 Introduction

In the past few years, fast growth in biological facts is demanding in computational analytics [2]. Bioinformatics is one among such important implementations of computer technology for accessing details in biological databases. Biological molecules and sequences may be accelerated for the details required to write different works like pulling out the specifics, ordering the details of data, perfectly analyzing the details and exposition of the details and also visualization of data which is used for gaining various forms of processes in bioinformatics. Important sources are DNA sequencing and protein mapping of bioinformatics [1]. Nowadays vast development of different molecules, genomic & genetic analyst in the domain pertaining to biology at micro level is generating huge amounts of details. The bioinformatics aim is to upgrade the view of gaining knowledge in the biological details. Machine learning is a promising approach peculiarly for approximate categorization and forecasting the dynamic environments by using huge quantities of data [2]. Analysis includes the process of a very large collection of biological details like gene searching, gene categorization, gene expression, microarray details, unhealthy gene forecasting, analytical shell of protein to protein communication, grouping of same gene details and so on. Factors uprooting from various types of inlet and outlet interrelation of bio details can also be approximately analyzed by machine learning [2].

16.1.1 Role of Machine Learning in Bioinformatics

Conventional computer systems were designed to decode bio details but unworthy to accept for complex details and impotent to trade in with speedy growth of details existing worldwide Machine training achieves an important part to decode the drawbacks present in the approaches existing in bioinformatics. Bio details requires diagnoses with different approach searches to find out various types of processes in genome detail residues, which is impulsively trained by the systems and give up hand for the bioinformatics domain. Machine training gives a reliable platform in gaining the knowledge of different varieties like learning through experience, learned with analogy, learning by sample. These types of learning proficiency of computers intensively enhance the results production based on time [3].

16.1.2 Machine Learning Applications for Bioinformatics

Machine learning for bioinformatics embrace Markov model, graphical models and support-vector-machines and neural networks, which were applied in managing probabilistic details of raw details and applied for analyzing genome like code forecasting or non-coding area of genome and RNA forecast [8].

Applications of genomics in machine learning are listed in Figure 16.1, based on the requirements for Gene data analytics by applying grouping flowcharts, Gene searching flowcharts applying easy Markov design,

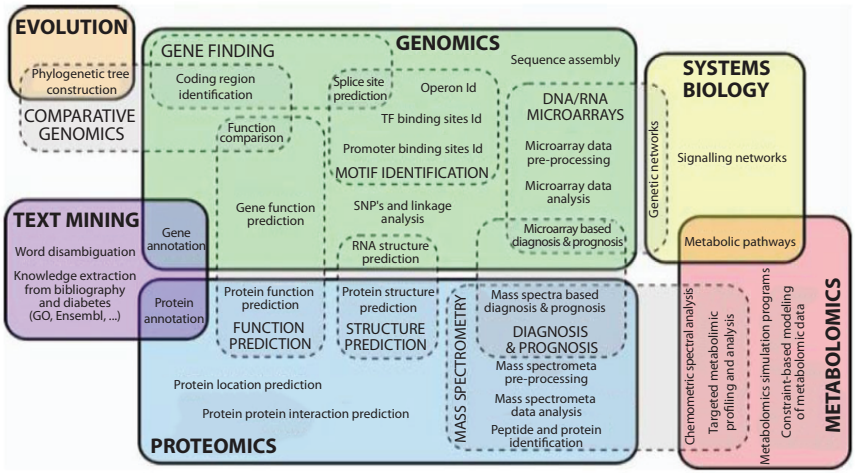


Figure 16.1 Bioinformatics suit.

factor estimation, Hidden Markov Model (HMM), and Viterbi design. Motif searching designs applying for various discrete data distribution, Alignment of genome is application of HMM. Combining various biological raw details for feature retrieving by applying Classification Algorithms, Decision Trees, Naive Bayes, Regression and Random Forest. Categorizing different types of genes ground on various types of group tables such as ANOVA, F-test and t-test [2].

16.1.3 Recent Trends in Bioinformatics

The field of bioinformatics is growing exponentially into a fountain with many opportunities for each individual from different disciplines such as biology, biotechnology, medicine, chemistry, statistics, mathematics, computer science, Information Technology & data analytics. There is an expanding urge for everlasting demand fields and its collaboration. Translating raw data details into knowledge is the important aim of this domain and there is an urge to take up innovative startups through different scientific & official organizations at a broader level globally. Bioinformatics is an everlasting field in expansion in many other crucial research areas like systems biology, pharmacogenomics and synthetic biology. With the enhancements in technologies and storage of data at an extraordinary rate, it is predicted for a stream of demand for bioinformatics scientists to solve the problems of technology evolving and manage the challenges in it [4].

16.1.4 Data Analytics in Bioinformatics

In the bioinformatics domain the research can be carried out on the data which is growing at an exponential rate. Big data resources are not bound to physics trials and web crawlers to find the indexes and logs. Advancement of digitization for every activity, along with present high end devices with less price, quantity of data popped in every domain and also same with bioinformatics domain. For example a human genome sequence is accurately 200 GB [5]. As the demand increased for exponential data which in turn lowered the computing cost and increased in the growth of analytics performance with enhancing big data procedures. Biologists need not use conventional wet labs to find better biomarkers for a particular health, but they can depend upon the large amounts of genome details made peruse by researchers of distinguished groups. Procedures applied for bio data are lowered by cost and become reliable for genome sequencers, this evolved to a new branch of big data in the bioinformatics domain.

The voluminous details of bioinformatics were grown at a faster rate in the past few years. The institutes like European Bioinformatics Institute (EBI), is one among the high bio details repositories and had accurately 40 petabytes storage of genes, molecules and proteins in 2014 when related with 18 peta data bytes in 2013 [6]. The amount of data updated to databases is multiplying year wise. The increased curve in their genome details and stored details are shown in Figure 16.2.

To process their data EBI erected a group, Hinton data center group, 17,000 cores of amount worth and RAM with 74 TB. EBI computing capability is upgraded almost each month. On a note EBI is one of the organizations existing in huge bio details databases and also many different organizations are existing as servers for massive collection of bio data to database stores, which are shared across the globe, like United States of America and National Institute of Genetics, NCBI and Japan.

The present availability of voluminous data is favoring better approximate analytics, in specific to analysts in bioinformatics [35]. Big data tasks

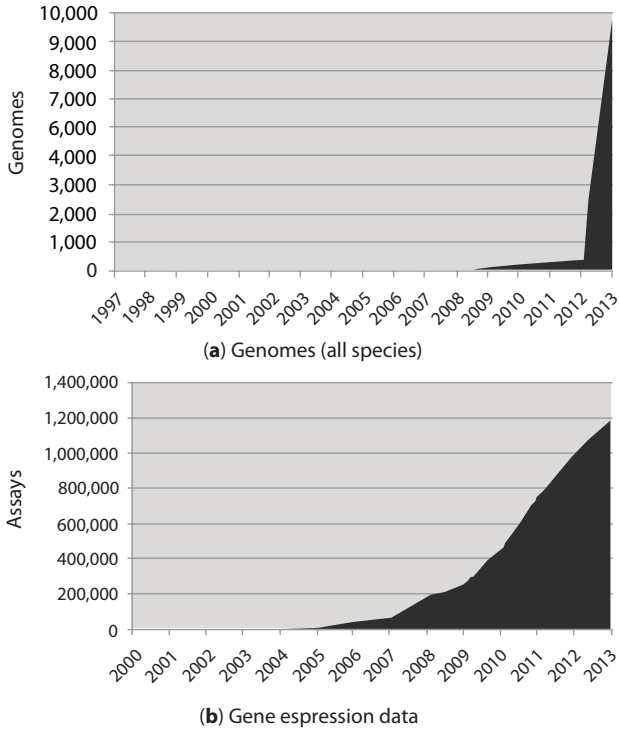


Figure 16.2 Amount of data stored by EBI over the years [3].

in bioinformatics can be distinguished from the other domain issues as particle physics details acquired at CERN or high configuration satellite details obtained at NRSC/ISRO open data archive2. Deviation can be enlisted into two categories. First, the data in bioinformatics is highly diversified in nature. To solve problems in bioinformatics, analytics needs many diversified and independent data stores for checking and validating them. In fact bioinformatics data stores are updated rapidly for a huge number of organisms and their repositories in a variety of forms of data from their sources. Second data of bioinformatics is voluminous growing wide in dimensions & instance occurrences across the globe. Only a partial amount of data is uploaded on the net whereas leftover data cannot be uploaded because of reasons like amount of data (not sufficient storage), cost, ethical and privacy matters. This issue forces researchers to carry out with the existing data and in turn to do only partial analytics and render results. Hence bioinformatics based big data problems are besides volume, velocity, and variety across global distributed data.

In order to manage the expectations of bioinformatics in big data, cloud computing procedures are applied which achieve good results. The excellent work using cloud for dual purpose is one for storage and other for computation [7]. Through this work it favored big data expectations on bioinformatics is achieved on huge, increasing and isolated distributed details. BGI abbreviated as Beijing Genomics Institute, which is one among the leading gene centers for sequencing, has erected a cloud depending on an analysis algorithm named Gaea, applied on Hadoop structure. Gaea is applied for the activities of genome analysis on a large scale, which is aligned wide among hundreds of computers dependent on clouds. One more important cloud baked genome analytics outcome is made available by Bina Technologies3, a Stanford University and UC Berkeley derivative from a hardware device called Bina box, apple for preprocessing on gene details and applied to cloud devices for performing analytics. Bina efficiently can also minimize the size of genome data and is effectively related to cloud devices. This success achieves to advance the output of genome analytics in terms of magnitude, more than the conventional procedures [8].

16.1.5 Machine Learning Algorithms

Machine learning procedures are well known in handling various challenges in bioinformatics. These algorithms [9] are grouped as namely Supervised Learning, Unsupervised Learning, Semi supervised Learning, Reinforcement Learning and Optimization. Each type of machine learning procedure gains up hand in the knowledge about the input protein

data sets and its transformation under various signal processing steps. The basic definitions of each type of algorithm is described based on the process of machine learning. In supervised learning a database which constitutes details of training data that has inputs and outputs to be attained. By implementing these training datasets on the procedures developed, generates a function which in turn is used to forecast the approximate results. By employing these training samples of data the algorithm obtains a function which is used to predict accurate models that makes further predictions of output values. The supervised learning is classified into classification and regression. In the classification category the label is determined as a finite set of discrete values, whereas in the regression category the label is determined as a finite set of continuous values. In unsupervised type learning a database contains training data that has both labeled and unlabeled examples, the objective of this type is to partition the training samples into clusters so that the data in each cluster displays a higher level of proximity. Note that the labels are not given to train the database in the type of learning. The third type of machine learning is semi-supervised learning, in this type a database contains both labeled and unlabeled examples. The basic requirement of this is to design a model capable of predicting the target of future output cases exactly for which its output values are not known. This can be based on the process of uncertainty accordingly. The database holds a little amount of labeled data together with a large amount of unlabeled data. In the reinforcement process of machine learning software agents are allowed to automatically manipulate the optimal behavior within a specified situation to improve the performance. A sample beneficial feedback is required for the agent so as to learn its behavior. Optimization is the last type of machine learning algorithm in which an optimal solution is searched in a space of multiple solutions of possibility as searching for the model that best suits and fits the data as well.

16.2 Basic Definition of Anatomy and Cell at Micro Level

The cell anatomy is viewed in the form of biological cells.

16.2.1 Biological Cells

Every organism has cells. They have nuclei that have the entire genetic material. Organisms without nucleus are tagged as prokaryotes & the organisms with presence of nucleus are tagged as eukaryotes. Normally,

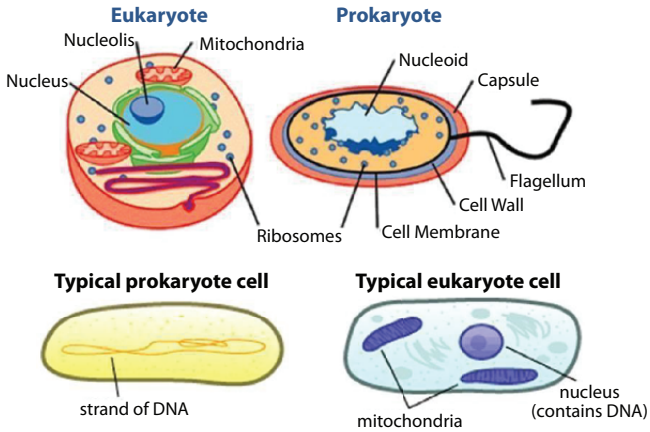


Figure 16.3 Depicts prokaryotic & eukaryotic cells.

prokaryotes are having single cells & eukaryotes are having many cells. The genetic material is available in both organisms but the variation availability of the nuclear membrane, which wraps the genetic material. Figure 16.3 depicts the cell content of an organism. Genetic carnal available substance tagged as chromatin. It is available either in shrink or unshrinks form based on cell life cycle and the shrink state is said as chromosome. In chromatin twenty percentages compose of nucleic acids and the rest 80% of the body substance is composed of proteins. Nucleic acids are the degree of linearity.

Macromolecules synthesized, which are tagged as polynucleotides. DNA & RNA [39] are a kind of elements of nucleotides. Proteins that are primarily building blocks play an elementary role in factor regulation of the organisms with presence of nucleus (eukaryotes) [10, 11].

16.2.2 DNA, RNA and Proteins

Chromosomes are the ones which have full details of the cells in the body, consisting of tiny molecules known as DNA. The location of DNA in the chromosomes is depicted in Figure 16.4. It is having the genetic paths used in the action event and the functionality of all living organisms. Each and every organism is completely unique just because of the chemical hereditary details. DNA strand is formed by nucleotides in DNA grouped in linear fashion. The complete stretch of DNA molecules sticks on to a single cell of the *Homo sapiens* that exist in pairs of meters. The helix shape formed by the dual single strands of DNA by bonding one another in volute pattern [12]. Each DNA has a phosphate term, a pentose sugar and the element nitrogen

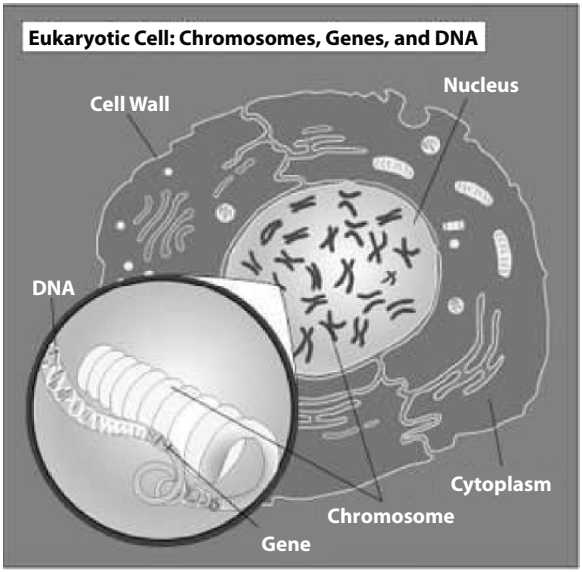


Figure 16.4 Eukaryotic nuclear DNA within the chromosomes.

having a base. The foundation of each strand of DNA are phosphate & sugar chemical compounds. The elements Guanine (G), Thymine (T), Cytosine (C) & Adenine (A), are base containers of four amino acids specifically. One strand of A joins with another antonym stand of T and similarly G joins with the C. The nucleotides A & T are called purine bases and similarly C & G are called pyrimidine bases. The deoxyribonucleic acid strands are held along principally by hydrogen bonds between the bases. These are linked with the sugars and protrude sideways. Nucleotides are a unit joined by linking the phosphate with the five units of carbon (deoxy) another sugar & one to the three-unit position of the next. The 2 strands of a deoxyribonucleic acid area unit opposed parallel, as they purpose in opposite directions [13].

RNA is commonly viewed as a single strand molecule. It is a counterpart of DNA which exists by the procedure called transcription. So, the dual stranded DNA gets isolated into one stranded RNA that is similar to the DNA by swapping of Thymine with Uracil (U). In the nucleus genetic details present in DNA, which is passed through ribosomes for progress of proteins [14], which is the primary responsibility of RNA. Figure 16.5 depicts RNA transcription & its sequence generated through a DNA strand. The DNA strands get isolated for the synthesis of message passing with RNA (mRNA). Just one strand is traced. The opponent of RNA bases are combined with the RNA enzyme.

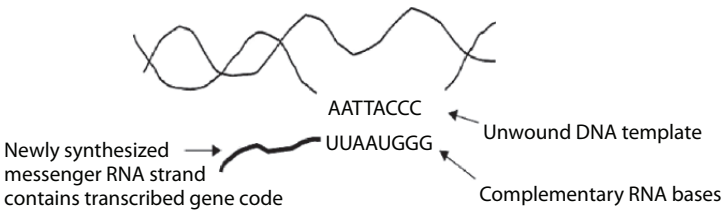


Figure 16.5 Depicts RNA transcription process shows a strand RNA given from a double strand and DNA and T swapped by U.

Transcription process is the starting point of the DNA is made by RNA polymerase, which synthesizes the messenger RNA at the developer site. The genetic details found in the cells are carried by mRNA. This process in organisms with nucleus isolated from the nucleus and enters into the cytoplasm. The Ribosomes are found in the cytoplasm group the triplet nucleotides in a group and translate into a link of amino acids called protein [10, 12]. This is the translation process. The cycle of transcription & translation to form of protein from DNA shown in Figure 16.6 below. The proteins are also known as polypeptide links because they are linked with peptide bonds.

A codon is a block which is formed by a triplet of nucleotides present in RNA. These codons are answerable for the making up of the specific chemical compound. As there are four fully totally different nucleotides in RNA, there are sixty-four possible codons inside the order as given in Table 16.1. There are twenty unique amino acids that conjure for varied sorts of proteins. The sequence AUG encoded as essential methionine, which acts as the initialization sequence of translation. The three blocks namely UAA, UAG and UGA aid to end the process of translation and thus referred to as stop codons. The set of laws by which the messages available in DNA sequences are translated into proteins by the living cells is known as code [14].

No matter what variety of knowledge, one must clearly perceive that the living organism is created of cells and these cells are nothing however evolved from differing kinds of chemical compounds. The physiological activities within the living organism are numerous chemical reactions happening in these chemical compounds. A number of these chemical reactions is also auxiliary and useful to the organism, leading to physical or physiological progress associate degreed growth of an organism. The chemical reactions which will be non-supportive may result in sickness. The fundamental chemical compounds that are accustomed kind a cell of associate degree organism are four bases,

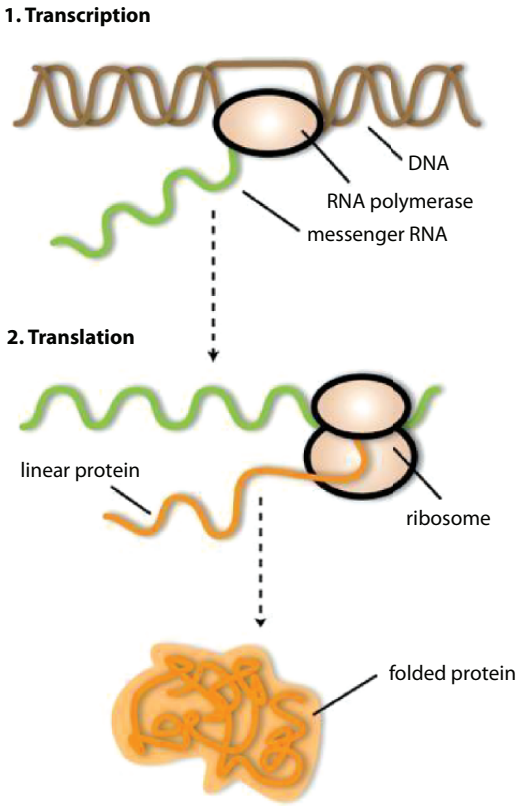


Figure 16.6 Depicts transcription and translation describing the process of conversion of DNA to protein.

conjointly referred as these four nucleotides are A, C, G, T. They are chains with one another, in numerous ways in a DNA is then coded into Proteins and later on the cell. Protein- Proteins area unit the essential building blocks of cells. Proteins area unit molecules created from many links of amino acids in a particular definite order. Triplet sets of nucleotides combined into an organic compound. The nucleotide sequences present in the sequence will code for the macromolecule that determines this order. Proteins are the essential units for right functioning of the body. Every macromolecule has distinctive performance that's necessary for the structure functioning & controlling the cells available in the body, tissues, and organs. They are essential for anatomical structures like skin or hair and for substances like enzymes, hormones, cytokines or antibodies. There are unique twenty amino acids as shown

Table 16.1 Genetic code describing 64 possible codons and the corresponding amino acids present in protein.

first letter	second letter: U	C	A	G	third letter
U	UUU, UUC } Phe UUA, UUG } Leu	UCU, UCC, UCA, UCG } Ser	UAU, UAC } Tyr UAA stop UAG stop	UGU, UGC } Cys UGA stop UGG Trp	U C A G
C	CUU, CUC, CUA, CUG } Leu	CCU, CCC, CCA, CCG } Pro	CAU, CAC } His CAA, CAG } Gin	CGU, CGC, CGA, CGG } Arg	U C A G
A	AUU, AUC, AUA } Ile AUG Met	ACU, ACC, ACA, ACG } Thr	AAU, AAC } Asn AAA, AAG } Lys	AGU, AGC } Ser AGA, AGG } Arg	U C A G
G	GUU, GUC, GUA, GUG } Val	GCU, GCC, GCA, GCG } Ala	GAU, GAC } Asp GAA, GAG } Glu	GGU, GGC, GGA, GGG } Gly	U C A G

in Table 16.2 are combined according to functionality in protein order arrangements.

A sequence could be a triplet nucleotide. Every sequence represents a specific organic compound. As their square measures four nucleotides in deoxyribonucleic acid, their square measures 43 attainable triplets. Every of those twenty amino acids has over one sequence to represent it. For instance, amino acids (gln) are often coded by CAA and CAG, and essential amino acids (arg) by CGU, CGC, CGA, CGG, title of respect and AGG as shown in Table 16.1. It's not necessary for all organisms to use all the codons equally. Men use twenty second of title of respect and twenty third of AGG and rest terribly very proportionally. Drosophila melanogaster uses forty ninth of CGC and also the rest terribly very low proportion. This is often termed

Table 16.2 Amino acids listed in table with 3-letter and 1-letter codes.

Amino acid	3-Letter code	1-Letter code	Amino acid	3-Letter code	1-Letter code
Alanine	Ala	A	Leucine	Leu	L
Arginine	Arg	R	Lysine	Lys	K
Asparagine	Asn	N	Methionine	Met	M
Aspartic acid	Asp	D	Phenylalanine	Phe	F
Cysteine	Cys	C	Proline	Pro	P
Glutamic acid	Glu	E	Serine	Ser	S
Glutamine	Gln	Q	Threonine	Thr	T
Glycine	Gly	G	Tryptophan	Trp	W
Histidine	His	H	Tyrosine	Tyr	Y
Isoleucine	Ile	I	Valine	Val	V

sequence bias, wherever a specific organism is a lot biased towards a particular sequence for organic compounds and generally to check dual organisms [15, 16].

Example for Protein Sequence

AQVINTFDGVADYLQTYHKLPDNYITKSEAQALGW
VASKGNLADVAPGKSIGGDIFSNREGKLPGKSGRTWR
EADINYTSGFRNSDRILYSSDWLIYKTTDHYQTFTKIR

16.3 Signal Processing—Genome Signal Processing

Genomic Signal Process (GSP) cope with the action of genomic signals, that area unit measured events sourcing from DNA. Proteomic signal process reveals with the process of signals sourcing from proteins. Each uses approaches from engineering & statistics to point out and survey the data combined with genetic materials. Here is the need of approaches to explore & give the end outcome in a biologically favorable manner. The specific nature of biological sequences gives apt data for the application of Digital Signal Process (DSP) approaches to analyze. DSP cope with the mathematical calculations on close details signal to enhance it into digital signal

form. It is featured by the presentation of signals in individual time & frequency or different domains by sequence of figures or symbols and also the action performed by those signals. As biological sequences are represented in alphabetical form, they are reborn into data sequences on which the signal processing approaches are implemented for investigation [30]. In present days, pc algorithms encourage DSP for knowing the features of DNA, RNA & protein sequences. Capable signal process approaches as wavelet transforms & digital filters are applied successfully to conclude biologically wrathful data on genomic & proteomic sequences. These scenarios are played on the spectral energy features in Fourier domain or spectrum calculation played on signal modeling. Though the existing strategies offer less correct in terms of accuracy, computationally advanced solutions with increased ground noise. The human genome sequence termed as genome is described in the introduction is the constructive tool designed and developed to explore human growth and health disorder [38]. The genetically code and the inter connection among the input data structures of bases termed as nucleotides ATCG in DNA that encodes the proteins and the input data sequence of the 20 amino acids in those proteins. The genetic flow step details are from DNA to function is the process of DNA to RNA to Protein finally to formation of Function.

16.3.1 Identification of Hotspots

Protein-target is an interaction between proteins with other proteins considered as a target. Protein-target interconnection is an important step to address the protein structure functionality which is unique in its own way. Active sites are the particular regions in the protein-target molecules that depicts the protein-target chemical interactions. Identification of these active sites which are involved in describing the functionality is challenging and critical as well in understanding the protein-target interaction process. Apart from identification of protein-target interaction, locating the active sites and identifying the dominating groups among the protein functions is to be performed in the next steps of the process. These dominating amino acid groups are entitled as hot spots. These hot spots are the major key role players for various applications in bioinformatics. Site-directed mutagenesis is a well-known experimental technical process in the identification of the hot spots [40]. The process of amino acids at a particular location is replaced by another type of amino acid in a natural means termed as mutations. In the process of mutation there are chances of changes in the biological properties which in turn change the functionality also. Alanine is the amino acid that is usually selected as the replaceable acid to perform the

mutation process. In order to conclude that a very particular specific location belongs to a hotspot, therefore the biological function of the protein and its mutation at that location will restrict the movement of the bonds of a particular amino acid. This is a continuous process repeated so as to locate the remaining hot spots [30]. This repetition process of amino acid is called as alanine-scanning mutagenesis (ASM).

16.3.2 Advantages of Computational Hotspot Identification Techniques Over Alanine-Scanning Mutagenesis

ASM is simple in terms of concept as the procedure involves many fragile steps that require perfect execution at microscopic levels. Tough biological experiment procedures like Alanine Scanning Mutagenesis (ASM) have been implemented to find hotspot locations to conclude but computational procedures like using digital filters can be applied at the inception stage which would be more helpful for the biologists in advance before they conduct wet lab experiments. With computational results as starting points, biologists can analyze only identified hotpots in their experiments and finalize hotspots. By doing so the amount of time, resources and effort can be reduced. Based on computational results 80% of them are hotspots identified. So, if computational procedures achieve reliability then biologists can reduce the work stretch of amino acids in their experiment.

Most of the present computational procedures need 3D structure of the protein in detailed description for investigation [21–23]. But for only newly found proteins, protein sequences of amino acids are available. Whereas 3D structure details are available only after researchers investigated and analyzed. Therefore computational procedures play a crucial role in identifying hotspots.

16.3.3 Overview of Protein Sequences

Proteins are clusters of proteins unify from ribonucleic acid by the activity called translation. They are to blame for completing the cellular scheme and the metabolic scheme of the cells mostly. The protein sequences are lengthy inter linked amino acids bonded by amide joints observed as peptide chains. These amino acids are 20 in count and they are drawn from protein strings of character notations with varying stretch from 100 to 10k [14]. Proteins are inclined to fold into 3D structures. These 3D structures are associated with biological operations [15] so, structure plays a vital role. Folding operation is very much complicated, as the polypeptide chain reaches a fixed 3D state with small and large chemical communication variations among

amino acids and in numerous states of the molecules, severally. The peptide chain twists during this folding method and bends till a state of lowered energy is attained. This increases the steady state of the ensuing shape. Here this allows the protein to act among various molecules considered to be the goals at particular points that are identified as energetic sites of protein [16]. These active sites possess different shapes that are specific in nature, so as to match the target molecules throughout the interaction process during a method similar to a leg fitting into a shoe is given in the Figure 16.7. The energetic sites in and around these sub regions are identified as hotspots that are chargeable for each and every chemical stability of active sites as an additive provision of the energy for the protein-target interactions. A hotspot could contain one or a lot of amino acids organized in a very distinctive shape within the protein sequence. Because the hotspots activate a crucial role in granting proteins to activate their roles, a specific data concerning the hotspot locations is very important in knowing the protein performance. Figure 16.7 demonstrates the protein handshake. The extraction for protein activities exploring the finding and characterization of all proteins further data concerning in their interaction with the neighboring proteins and deoxyribonucleic acid molecules is required in-depth.

Binding of the protein with the target protein is very much similar to the key fitting to its lock given in Figure 16.7 [24] depicts the interaction process of one protein with the other. The amino acids present at communication junctions are identified as hotspots. These hotspots are very much responsible for exhibiting a characteristic frequency and its operations. Obtaining the characteristic frequency of a selected hotspot, new similar hotspots are often foretold in alternative unannotated protein sequences. Often, cancer-causing mutations cluster in hotspots wherever tumors from completely different patients harbor a similar continual mutation. Some hotspot mutations could

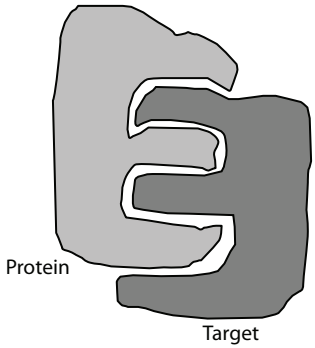


Figure 16.7 Depicts a protein handshake.

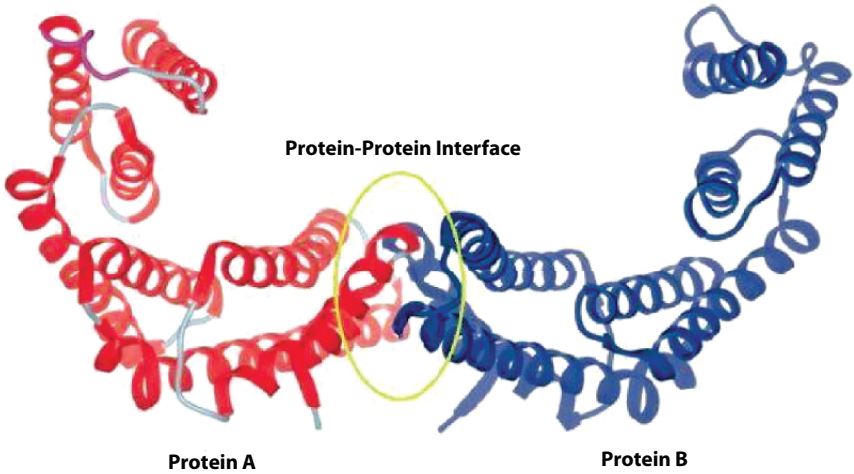


Figure 16.8 Depicts Protein–Protein interaction render to hotspot.

occur often, whereas others are rare. In cancer, corporeal driver mutations usually target paralogous remain hotspots across the root connected members of one protein family. These hotspots as shown in Figure 16.8, usually represent the foremost drug able genetic alterations.

16.3.4 EIIP & CPNR-Based Mapping

Protein sequence consists of twenty amino acids. Correct illustration of amino acids is vital to extract the correct relation of activities in proteins [26]. Related to alphabetical illustration of amino acids, numerically illustration of amino acids will massively elaborate the power to analyze them. Among the progressive procedures, electron–ion interaction potential (EIIP) could be a default illustration pattern littered with degeneracy problems that to functionally completely different proteins could get terribly similar or maybe constant illustration [25]. The degeneracy issue in amino acids mapped may be the same representation of numbers. To manage such a task in literature as a replacement procedure, Complex Prime Numerical Representation (CPNR) of amino acids representation is done [26] in Table 16.3 and Table 16.3 shown below the notation of Amino acid and Codons its EIIP values respectively.

By considering the 5 amino acids P, T, V, A, and G as an instance then every amino acid has 4 codons. In order there are 5 prime numbers (7, 13, 19, 37 and 43) in the gap [2, 67], everything is less than the next non-composable number by four. That order of correlation

Table 16.3 Depicts EIIP mapping for amino acids.

Amino acid	EIIP Value	Amino	EIIP Value
M	0.0823	Q	0.0761
W	0.0548	S	0.0829
F	0.0946	A	0.0373
Y	0.0516	N	0.0036
P	0.0198	G	0.005
C	0.0829	R	0.0959
T	0.0941	I	0
H	0.0242	D	0.1263
V	0.0057	E	0.0058
L	0	K	0.0371

was inspired for representation of amino acids based on prime numbers Table 16.4 shown below represents the differential pattern between each non-composable number and so on.

CPNR is supported by the amount of codons present in amino acids as given in Table 16.5. A good commonality among the amounts of codons present in amino acids is identified and a different scheme of prime numbers. To form the illustration a lot of biologically important, prime numerical are represented into complicated fields. Contrast with EIIP, CPNR reveal no existence of degeneracy drawback [26], and also the representation in CPNR are comparatively dependent with one another, which implies that one range

Table 16.4 Number of codons for 20 amino acids.

No. of codons	Amino Acids
1	M, W
2	F, Y, C, H, Q, N, D, K, E
3	I
4	P, T, V, A, G
6	L, S, R

Table 16.5 Different pattern among each of the non-composable number and its beside.

Difference with the next non-composable number	Non-composable (1 or prime) numbers
1	{1, 2}
2	{3, 5, 11, 17, 29, 41, 59}
3	Φ
4	{7, 13, 19, 37, 43}
6	{23, 31, 47, 53, 61}

can't be generated from another by addition, multiplication, or involution of correct numeric value. Therefore protein sequence is often evolving again by numerical representation sequence by substituting every alphabet by their respective CPNR values [27]. Table 16.6 shown below gives the tabular values used to represent the Amino acids with 20 prime numbers which depicts the importance of the amino acids notation for further processing and plays a very vital role in machine learning aspects and analysis point of view for evaluating the performance metrics like accuracy, precision, F1 Score.

Table 16.6 Complex Prime Numerical Representation (CPNR).

S. no.	Amino acid	CPNR	S. no.	Amino acid	CPNR
1.	M	1	11.	Q	29
2.	W	2	12.	S	31
3.	F	3	13.	A	37
4.	Y	5	14.	N	41
5.	P	7	15.	G	43
6.	C	11	16.	R	47
7.	T	13	17.	I	53
8.	H	17	18.	D	59
9.	V	19	19.	E	61
10.	L	23	20.	K	67

The observations tabulated in Table 16.6 are constructed from a set Q from prime numericals ranging from 2 to 67 and the numerical 1. Here defined a one-to-one representation of: G to Q so as to 20 amino acids are represented to 20 non composable numbers in Q (Table 16.6) [32]. In representation f, the gap among two represented numericals is big enough. For instance: 67–2 = 65. To remove a few amino acids with more amplitudes when compared with others, we still represent the 20 numericals in Q onto a unit circle given in Figure 16.9 in a complex numerical domain. Hence, a protein a = [a1, _ _ _ , an] is given in an array of complex numericals x = [x1, . . . , xn]. Here complex number array x is the complex numerical array x and the complex prime numerical representation (CPNR) of the protein [26].

$$x_j = e^{-\frac{2\pi f(a_j)}{67}}, \forall j \in [1..n].$$

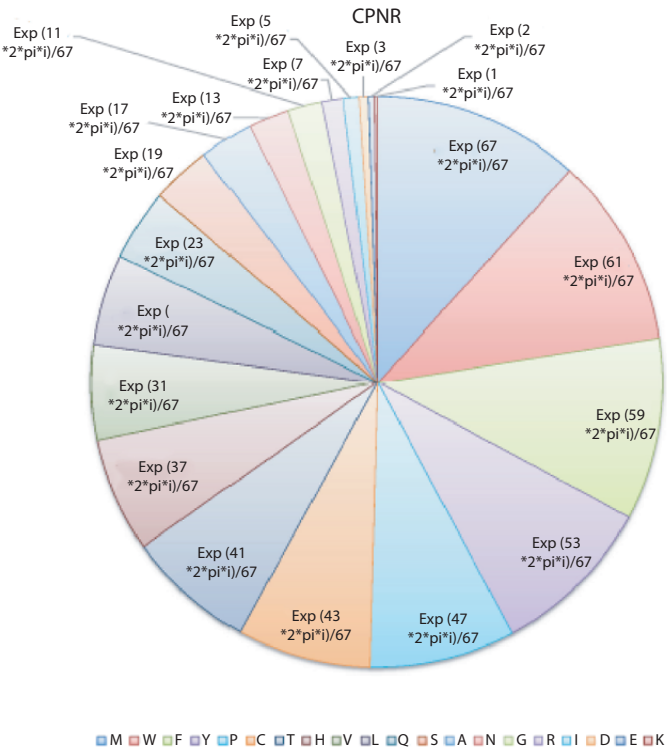


Figure 16.9 Complex plane of unit circle taken for 20 complex numbers in CPNR.

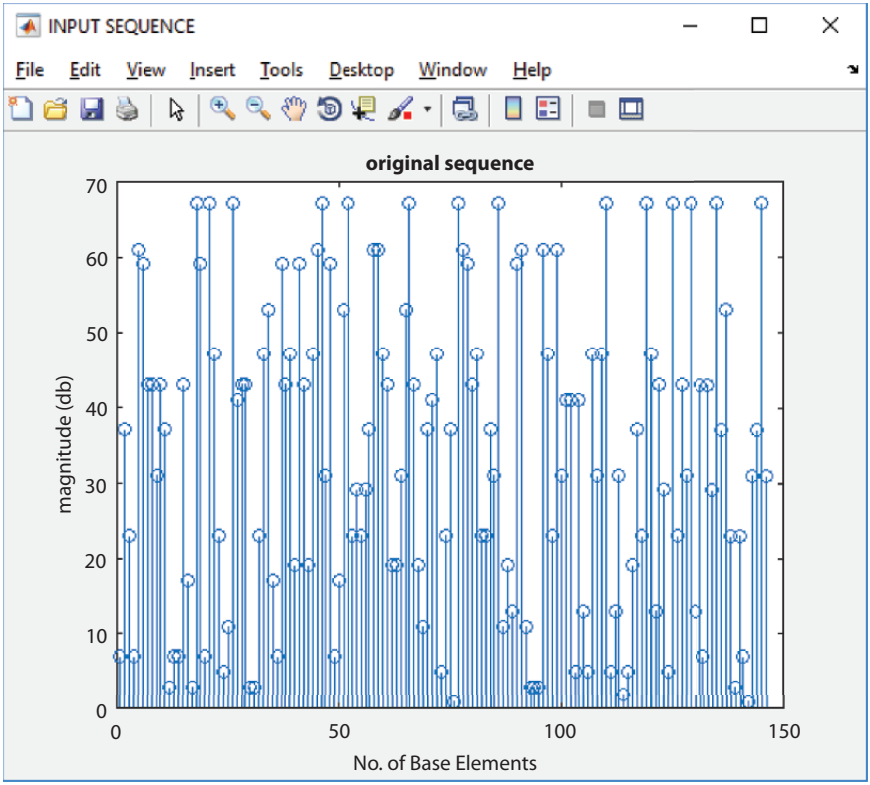


Figure 16.10 Mapping of amino acids using CPNR.

For example, the Human Fibroblast Growth Factor (HFGF) protein sequence is considered as input to the procedure with length l = 146 and its CPNR shown in Figure 16.10 and EIIP in Figure 16.11 respectively.

16.3.5 Feature Extraction Technique

Genomic Signal Processing (GSP) is viewed as an advanced evolving discipline in bioinformatics nowadays [28], that method with genomic logic analytics and their details by involving Digital Signal Process (DSP) techniques [29–31]. Genomic details like protein sequences that exist in character type mapped into a digital type known as genomic signal. Bioinformatics is an associate degree knowledge base space that contains process procedures for quick and correct illustration of biological details.

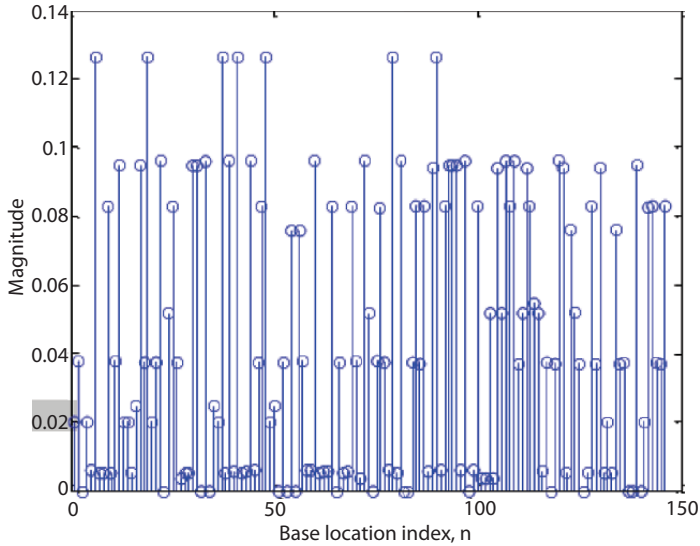


Figure 16.11 Mapping of amino acids using EIIP.

16.3.5.1 RRM—Resonant Recognition Model

Resonant Recognition Model (RRM) is known as the model to determine the biological functions of proteins, which is a physio-mathematical technique that depicts the protein sequence details by means of techniques like digital signal processing. The literature on the role of signal-processing concepts in genomics and proteomics evidently describes the sequence details and significant correlation between the frequency spectrum of the numerical notation of amino acid sequences and its functional activities. In the process of analyzing the function of the protein RRM is performed in two stages [40]. Firstly the symbolic sequences as mentioned in the table of EIIP values of the 20 amino acids are converted into the numerical representation using Electron Ion Interaction Pseudopotential (EIIP)/CPNR accordingly. The DFT of the protein is computed to evaluate the consensus spectrum. The spectra of the common frequency exhibit a peak in the cross-spectrum function. The determination of the character frequency helps in identification of the individual amino acids that are involved in contributing the hotspots. The character frequency is calculated by the cross-spectral function.

16.3.5.2 Discrete Wavelet Transforms

The significant disadvantage of the Fourier transform is that it cannot reflect the time resolution of the frequency and it has been widely used in signal processing to analyze stationary signals. Therefore in order to handle this challenge wavelet transforms approach can be used. Wavelets have the ability to analyze signals with discontinuities and sharp spikes. DWT is one of the most famous transforms having excellent signal compaction properties for many classes of real-world signals and is computationally efficient. These properties and advantages of the wavelets support in applying these transforms in several fields of signal processing, image processing and communications and also pattern recognition. The specific application of wavelets is in medical applications which have a huge scope in machine learning approach for data analysis. As stated in the above sections, machine learning is a multidisciplinary field and has a wide range of research domains in which medical applications is one among them [9]. The analysis of medical data from various sources for clinical diagnosis is throwing a challenge for medical practitioners today. This aspect has become the main motivation for looking at machine learning approaches in the medical domain.

Wavelet transform is a reliable technique well known for the analytics and processing of non-stationary like bio signals in which both the time and frequency details are required. The wavelet transform gives a solution to the problem of time & frequency resolution because of the multi resolution strategy of wavelet transforms [31]. Wavelets are well defined in both time & frequency domains. Wavelet analysis can be compressed or denoised as a signal without effective reduction. Wavelet analysis is the decomposing of a signal into shifted and scaled versions of the original (or mother) wavelet equation below [34]. Wavelet transform is given in continuous & discrete domain.

Mathematical formula Continuous Wavelet Transform (CWT) describe as

$$C(p,q) = < \Psi_{p,q}(t), x(t) > = \int_{-\infty}^{+\infty} x(t) 1/\sqrt{p} \Psi((t-q)/p) dt$$

The terms p and q denote dilation and translations respectively. Ψ is the basic wavelet termed as mother wavelet. The DWT is evaluated by successively processing discrete time domain signals through band pass and stop band filters. DWT coefficients will have energies of hotspot signals required [32]. DWT technique is one of the feature extraction technique used to extract the coefficients to represent the sequences in its compact and unique form of single values or matrix vectors. This feature extraction process is the major steps which gives the scope of the research work towards machine learning. The performance analysis of the algorithm employed in representation of the sequence the transformation process performed on the mapped sequences to evaluate the hot spots depends on the transformation techniques like STFT—Short Time Fourier Transform, STDFT—Short Time Discrete Fourier Transform, and DWT—Discrete Wavelet Transform. The comparison of various transforming techniques applied to evaluate the coefficients and hotspot identification and filters employed to locate the hotspots depicts the computational efficiency of an algorithm chosen, filter applied and the performance analysis for comparative analysis of the representation of the protein sequences.

16.4 Hotspots Identification Algorithm

The algorithm to identify the hotspots [33, 34] is given below and flowchart in Figure 16.12:

Step 1: Read the input protein sequences from NCBI.
Step 2: Convert the protein alphabets in the obtained sequence from the step 1 into numeric sequence using CPNR/EIIP mapping scheme.
Step 3: In order to obtain the energy components in the characteristic frequencies apply narrow band pass filtering technique on the obtained numeric sequences.
Step 4: Apply DWT filtering on the obtained coefficients from step 3, so as to get the signal energies available in hotspots.
Step 5: The peak values in the obtained plot, denote the hot spots.
Step 6: Performance metrics evaluation.

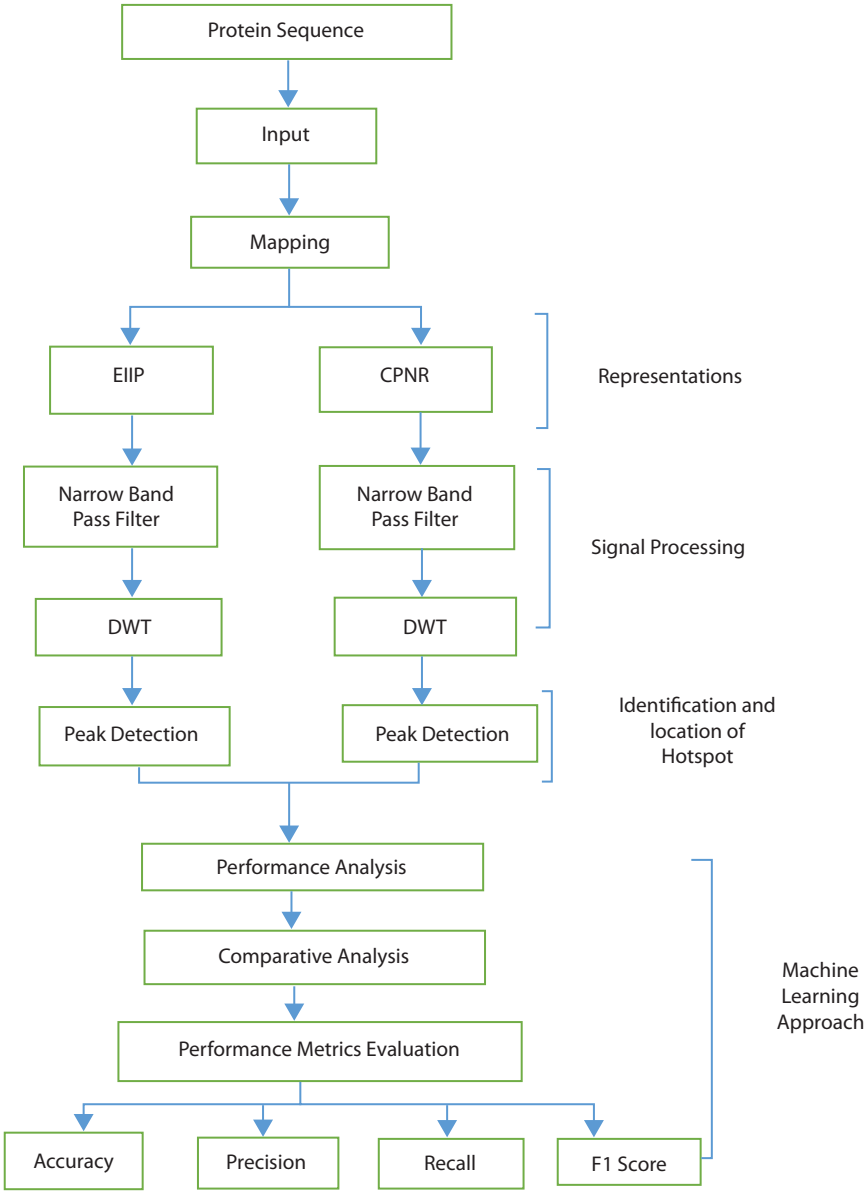


Figure 16.12 Depicts Flow of EIIP, CPNR mapping with DWT for hotspots identification.

16.5 Results—Experimental Investigations

The Dataset used for evaluating Process is given in Table 16.7.

Table 16.7 Dataset of protein sequences used for experiment.

Organism	Protein Name	PDB ID	Sequence Length	Characteristic Frequency
Human	Fibroblast growth factor (FGF)	4fgf	146	0.904
Human	Human Growth Hormone (HGH)	3hhr	190	0.270
Bacteria	Barstar	1brs	89	0.321
E. Colo	Colicin-E9 Immunity IM9	1bxi	86	0.190
Bacteria	Barnase	1brs	110	0.321
Bacteria	TRAP	1wap	75	0.247
Human	Human alpha hemoglobin	1vwt	142	0.023

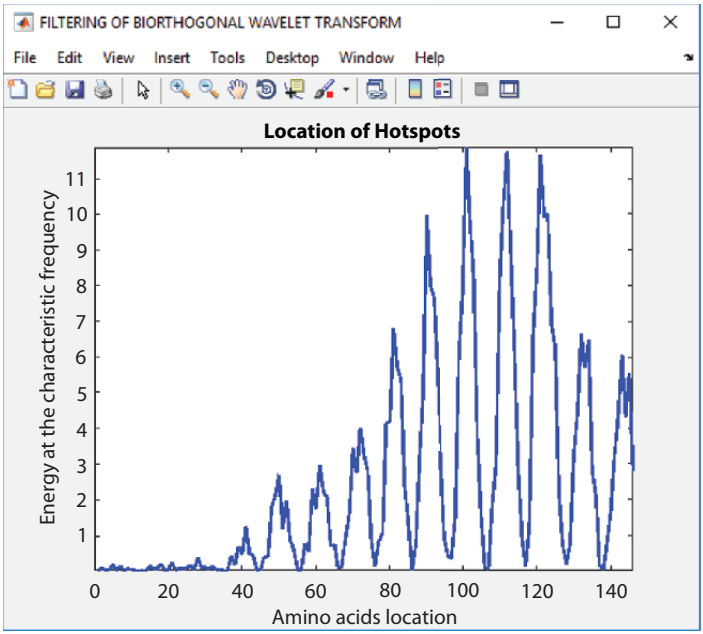


Figure 16.13 CPNR results.

The results obtained from CPNR AND EIIP are shown in Figures 16.13 and Figure 16.14 respectively.

The experiment results obtained using CPNR are tabulated in Table 16.8 and the Experiment results obtained using EIIP are tabulated in Table 16.9 respectively.

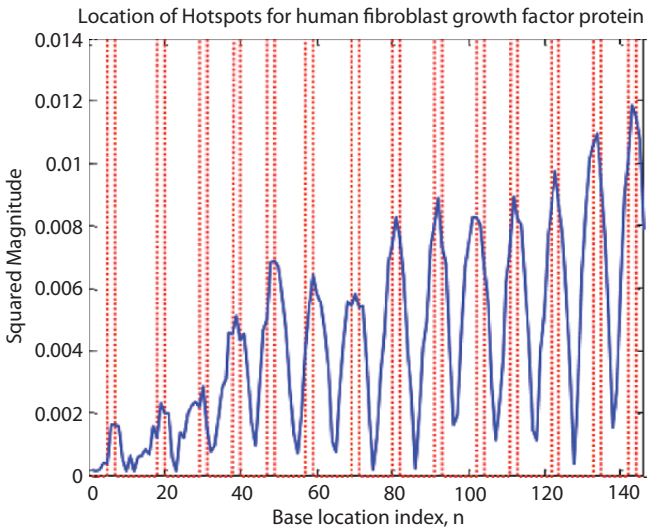


Figure 16.14 EIIP results.

Table 16.8 Details of identified hotspots using CPNR.

Input Sequence	Hotspots identified using CPNR approach
Fibroblast growth factor	5 17 24 28 41 50 61 72 81 96 103 111 121 134 140
Human Growth Hormone (HGH)	4 11 125 129 133 141 145 148 151 155 159 163 168 171 172 178 179 188
Barstar	3 6 9 12 15 18 22 25 29 31 35 39 42 44 47 50 53 56 59 62 66 69 72 76 78 81 84 87
Colicin-E9 Immunity IM9	5 9 15 21 25 33 34 41 45 50 51 55 64 68 71 79
Barnase	3 7 10 16 23 27 32 35 41 43 46 52 58 60 63 67 70 73 77 80 83 87 90 93 96 99 102 106 108
TRAP	4 7 13 16 20 25 29 33 37 40 45 49 56 58 61 65 69 73
Human alpha hemoglobin	14 18 20 22 28 36 56 59 63 95 98 100 103 117 126 139 141

Table 16.9 Details of identified hotspots using EIIP.

Input Sequence	Hotspots identified using EIIP approach
Fibroblast growth factor	4 7 21 31 41 51 60 71 81 92 103 114 124 134 145
Human Growth Hormone (HGH)	20 43 71 75 80 83 109 113 117 121 125 129 133 136 140 143 147 151 155 171 183 187
Barstar	19 37 41 49 55 61 65 75
Colicin-E9 Immunity IM9	3 11 19 26 29 33 39 43 51 55 62 67 72 78 83
Barnase	13 57 77 83 97 101 104 107
TRAP	27 51 68 71
Human alpha hemoglobin	10 15 35 43 64 66 74 79 107 111 119 127

16.6 Analysis Using Machine Learning Metrics

16.6.1 Theoretical Details of Performance Metrics

Metrics for confusion matrix to measure the performance of functional comparison of proteins can be computed as shown in Figure 16.15

Protein amino acids having the same hotspot in comparison with databases then it is considered 'positive.' Otherwise, it's 'negative.' Algorithm output has four outcomes of computational functional relation. (1) True Positive (TP) (2) False Positive (FP) (3) True Negative (TN) and (4) False Negative (FN).

		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

Figure 16.15 Confusion matrix.

Analysis of algorithm performance:

1. True positive rate (sensitivity), TPR = TP/(TP + FN)
2. True negative rate (specificity), TNR = TN/(FP + TN)
3. Postivie predictive value (precision), PPV = TP/(TP + FP)
4. Negative predictive value (NPV) = TN/(FN + TN)
5. Accuracy = (TP + TN) / (TP + TN + FP + FN)
6. Recall=TP/(TP+FN) and
7. F1 Score=2*((Precision*Recall)/(Precision + Recall))

i) Precision Calculation details as shown in Figure 16.16

		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

True Positive + Positive = Total Predicted Positive

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive + False Positive}}$$

$$= \frac{\text{True Positive}}{\text{Total Predicted Positive}}$$

Figure 16.16 Precision calcuation details.

ii) Recall Calculation details as shown in Figure 16.17

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive + False Negative}}$$

$$= \frac{\text{True Positive}}{\text{Total Actual Positive}}$$

		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False positive
	Positive	False Negative	True Positive

True Positive + False Negative = Actual Postive

Figure 16.17 Recall calculation details.

iii) F1 Score Calculation details

$$F1 = 2 \times \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

F1 Score is required to measure the balance between Precision and Recall. And accuracy can be largely contributed by a large number of True Negatives which in most of the circumtances, not given much attention. Whereas False Negative and False Positive usually has the costs (tangible & intangible) thus F1 Score might be a better measure to use if required to seek a balance between Precision and Recall and when there is an uneven class distribution (large number of Actual Negatives).

16.6.2 Comparative Analysis of the Protein Sequence Representation

Table 16.10 shows the tabulated values evaluated for EIIP and CPNR mapping representation respectively. The comparison is done for the performance metric values like accuracy, precision recall and F1 score calculations using machine learning concepts. Each metric has its own importance in understanding how best a parameter is in measuring the performance of the implementation process, given in Figure 16.18.

Table 16.10 Analysis of experimental results.

Protein	EIIP Mapping					CPNR Mapping			
		Precision	Recall	F1 score	Support	Precision	Recall	F1 score	Support
1	Accuracy	–	–	0.88	147	–	–	0.92	147
	Macro Average	0.52	0.58	0.52	147	0.60	0.84	0.64	147
	Weighted Average	0.95	0.88	0.92	147	0.97	0.92	0.94	147
2	Accuracy	–	–	0.83	191	–	–	0.87	191
	Macro Average	0.49	0.48	0.48	191	0.32	0.31	0.32	191
	Weighted Average	0.88	0.83	0.85	191	0.90	0.87	0.89	191
3	Accuracy	–	–	0.86	90	–	–	0.74	90
	Macro Average	0.47	0.45	0.46	90	0.59	0.86	0.7	90
	Weighted Average	0.89	0.86	0.87	90	0.95	0.74	0.81	90
4	Accuracy	–	–	0.80	87	–	–	0.80	87
	Macro Average	0.57	0.61	0.57	87	0.57	0.61	0.57	87
	Weighted Average	0.86	0.80	0.83	87	0.86	0.80	0.83	87

(*Continued*)

Table 16.10 Analysis of experimental results. (*Continued*)

Protein	EIIP Mapping					CPNR Mapping			
		Precision	**Recall**	**F1 score**	**Support**	**Precision**	**Recall**	**F1 score**	**Support**
5	Accuracy	–	–	0.86	111	–	–	0.73	111
	Macro Average	0.47	0.46	0.46	111	0.33	0.26	0.29	111
	Weighted Average	0.87	0.86	0.87	111	0.93	0.73	0.82	111
6	Accuracy	–	–	0.89	76	-	-	0.82	76
	Macro Average	0.47	0.47	0.47	76	0.61	0.90	0.63	76
	Weighted Average	0.89	0.89	0.89	76	0.96	0.82	0.86	76
7	Accuracy	–	–	0.90	143	–	–	0.90	143
	Macro Average	0.53	0.56	0.54	143	0.61	0.85	0.66	143
	Weighted Average	0.94	0.90	0.92	143	0.97	0.90	0.93	143

16.6.3 Visual Analysis

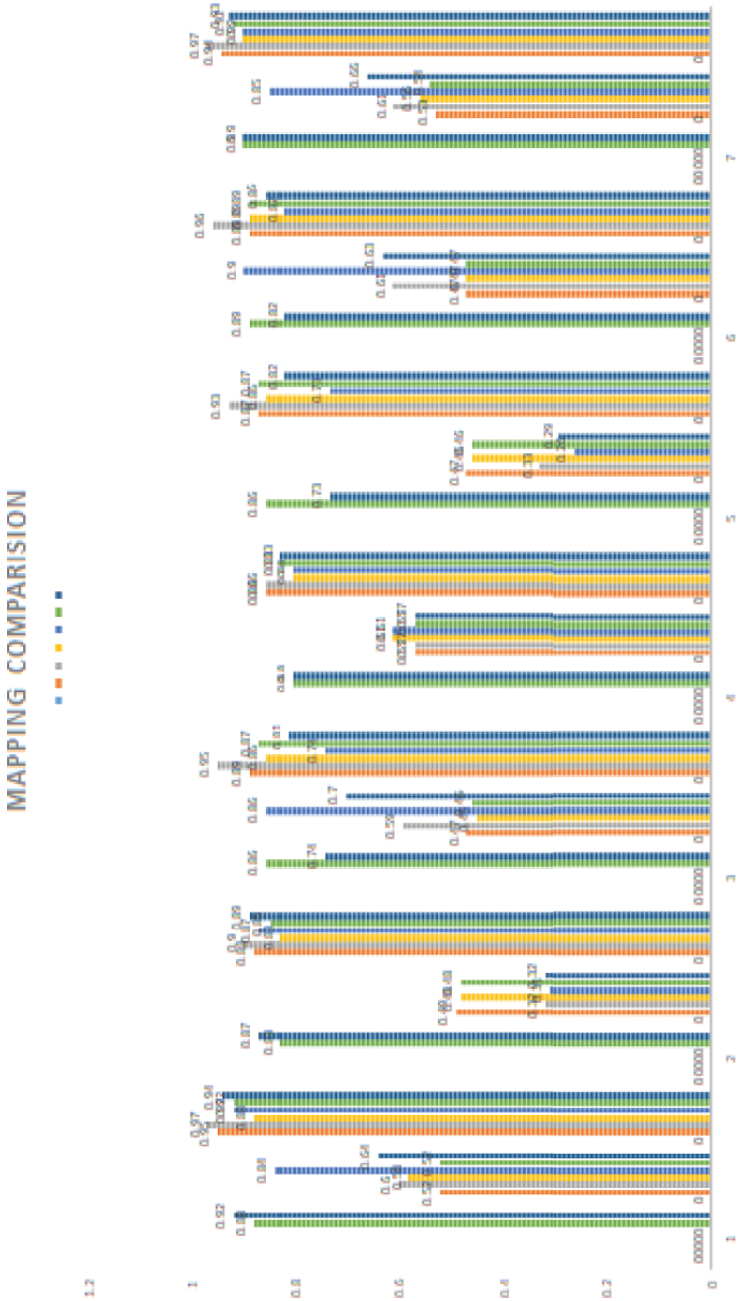


Figure 16.18 Overall performance analysis visualization.

16.7 Conclusion

A machine learning approach for performance analysis of protein sequences identification of hot-spots using signal processing techniques of bioinformatics for medical data is explained in this chapter. The technique was applied using both EIIP and CPNR numerical representation of the proteins. Comparative analysis is done with the corresponding results obtained. The performance metrics are also evaluated in determining the accuracy, precision, recall and F1 score which play a vital role in machine learning algorithmic approach. The future scope of this approach is to predict hotspots involved in cancer detection and computer aided diagnosis as well. The improved machine learning algorithms for enhancing the metrics required for better analysis of protein sequences for various applications.

Appendix

A.1 Hotspot Identification Code

```
% read protein sequence
fid1 = fopen ('numericalsequencefilename.txt', 'r'); if fid1 == -1
disp ('error, check') else
disp ('ok') end
x = fastaread('numericalsequencefilename.txt');%fasta file has
%header and Sequence
p = x.Sequence; % getting info of Sequence in string format
    PROTEIN = str2num(p);% converting string to number
s = ones(size(PROTEIN)); L = length(s)
figH = figure;
set(figH,'Name','INPUT SEQUENCE','NumberTitle','off') stem
    (PROTEIN);
subplot(1,1,1);
stem(PROTEIN);
title('original sequence');
xlabel('No.of Base Elements');
ylabel ('magnitude(db)');
N = 170;
F1 = 0.220;
F2 = 0.226;
 flg = 'scale'; % Sampling Flag
```

```
beta = 0.5; % Window Parameter
% creating window
wind = kaiser(N+1, Beta);
% Coefficients FIR1 functn.
b= fir1(N, [F1 F2],'bandpass', wind, flg);
Hd = dfilt.dffir(b);
w3 = filter(Hd,PROTEIN);
[ca0,cd0]=dwt(w3,'bior3.9');
b0=idwt(ca0,cd0,'bior3.9'); figH = figure;
set(figH,'Name','BIORTHOGONAL WAVELET TRANSFORM',
    'NumberTitle','off') subplot(3,1,1);
plot(ca0);
title('biorthogonal approx wavelet transform');
xlabel('No.of Base Elements'); ylabel('magnitude(db)');
subplot(3,1,2); plot(cd0);
title('biorthogonal detail wavelet transform');
xlabel('No.of Base
Elements');
figH = figure;
set(figH,'Name','FILTERINGOFBIORTHOGONALWAVEL-
    ETTRANSFORM',
'NumberTitle','off')
u3=(abs(w3)); e=u3.^2;
%[pks,locs] = findpeaks(u3);
plot(e,'b','linewidth',2);
axis([0 L    1.1027e-04 54.1340])%when plot e
%plot(u3,'b','linewidth',2);
%axis([0 L 8.5113e-07 4])%when plot abel('Amino acids location');
ylabel('Energy at the characteristic frequency');
```

A.2 Performance Metrics Code

```
#Python Code for evaluating
#performance metrics
# cpnr 1
# importing libraries
from sklearn.metrics import confusion_matrix from sklearn.
    metrics import accuracy_score from sklearn.metrics import
    classification_report
```

```
# Actual hotspots from ASEDb 18 25 42 45 46 64 168 171 172
    175 178 179
actual      =
[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0
    ,0,0,0,0,0,0,0,1,0,0,1,1,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,1,0,0,1,0,0,1,1,0,0,0,0,0,
    0,0,0,0,0,0]
len(actual) 191
actual.index(1) 18
# Predicted hotspot from cpnr mapping for Protein Sequence1
    20 43 71 75 80 83
109 113 117 121 125 129 133 136 140 143 147 151 155 171 183
    187
Predicted  =
[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
    ,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,0,1,0,0,1,0,
    0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,
    0,1,0,0,0,1,0,0,1,0,0,0,1,0,0,
0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,0,
    0,0,1,0,0,0]
len(predicted) 191
predicted.index(1)
20
res=confusion_matrix(actual,predicted) res
array([[158, 21],
[11,1]],dtype=int64)cls=classification_report(actual,predicted)
```

	Precision	recall	f1 score	support
0	0.93	0.88	0.91	179
1	0.05	0.08	0.06	12
accuracy			0.83	191
macro avg	0.49	0.48	0.48	191
weighted avg	0.88	0.83	0.85	191

References

1. Li, J., Wong, L., Yang, Q., *Data Mining in Bioinformatics*, IEEE Intelligent System, IEEE Computer Society, 2005. (https://ieeexplore.ieee.org/abstract/document/1556510/)
2. Jayanthi, K. and Mahesh, C., Need of Machine Learning In Bioinformatics. *Int. J. Innov. Technol. Exploring Eng. (IJITEE)*, 8, 11, pg 2608, September 2019.
3. Larranaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., Lozano, J.A., Armananzas, R., Santafe, G., Perez, A., Robles, V., Machine learning in bioinformatics. *Briefings Bioinf.*, 7, 86–112, 2006.
4. Recent Trends and Development in Bioinformatics: Challenges and Opportunities article in CSI Communications 40(7):28-31, Oct 2016.
5. Robison, R.J., How big is the human genome?, Precision Medicine, January 2014 (https://medium.com/precision-medicine/how-big-is-the-human-genome-e90caa3409b0)
6. EMBL-European Bioinformatics Institute, *EMBL-EBI Annual Scientific Report 2013*, Cambridge, 2014.
7. Marx, V., Biology: The big challenges of big data. *Nature*, 498, 7453, 255–260, 2013.
8. Rojahn, S.Y., Breaking the Genome Bottleneck. *MIT Technol. Rev.*, May 2012. (https://www.technologyreview.com/2012/05/07/186187/breaking-the-genome-bottleneck/)
9. https://www.researchgate.net/publication/329329261_Machine_Learning_from_Theory_to_Algorithms_An_Overview.
10. Darned, J., Lodish, H., Baltimore, D., Molecular cell biology. *Sci.*, 1, 1, 20, 1986.
11. Lodish, H., Berk, A., Zipursky, S.L., Matsudaira, P., Baltimore, D., Darnell, J.E., *Molecular Cell Biology* - NCBI Bookshelf, 4th edition, 2000.
12. Crick, F. and Watson, J., Molecular structure of nucleic acids. *Nature*, 171, 4356, 737–738, 1953.
13. Stansfield, W.D., *Schaum's outline of theory and problems of Genetics*, Third edition, McGraw Hill, Europe, 1991.
14. Anastassiou, D., Genomic signal processing. *Signal Process. Mag. IEEE*, 18, 4, 8–20, 2000.
15. Clark, D.P. and Pazdernik, N.J., *Molecular biology*, second edition, Academic Cell, 2012. (https://iubmb.onlinelibrary.wiley.com/doi/abs/10.1002/bmb.20760)
16. Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walter, P., *Molecular Biology of the Cell*, 4th edition, Garland Science, New York, 2002.
17. Old, R.W. and Primrose, S.B., *Principles of Gene Manipulation: An Introduction to Genetic Engineering*, 4th ed., Blackwell Scientific, Oxford, U.K., 1989.
18. Cunningham., B.C., Jhurani., P., Ng, P., Wells, J.A., Receptor and antibody epitopes in human growth hormone identified by homolog-scanning mutagenesis. *Science*, 243, 4896, 1330–1336, Mar 1989.

19. Wells, J.A., Systematic mutational analyses of protein–protein interfaces. *Meth. Enzymol.*, 202, 390–411, 1991.
20. Bogan, A.A. and Thorn, K.S., Anatomy of hot spots in protein interfaces. *J. Mol. Biol.*, 280, 1–9, 1998.
21. Gao, Y., Wang, R., Lai, L., Structure-based method for analyzing protein–protein interfaces. *J. Mol. Model.*, 10, 1, 44–54, 2004.
22. Li, X., Keskin, O., Ma, B., Nussinov, R., Liang, J., Protein–protein interactions: Hot spots and structurally conserved residues often locate in complemented pockets that pre-organized in the unbound states: Implications for docking. *J. Mol. Biol.*, 344, 3, 781–795, 2004.
23. Huo, S., Massova, I., Kollman, P.A., Computational alanine scanning of the 1:1 human growth hormone-receptor complex. *J. Comput. Chem.*, 23, 1, 15–27, 2002.
24. Kakumani, R., Omair Ahmad, M., Devabhaktuni, V., Identification of CpG islands in DNA sequences using statistically optimal null filters. *EURASIP J. Bioinform. Syst. Biol.*, 12, 1–14, 2012.
25. Cosic, I., Macromolecular bioactivity: Is it resonant interaction between macromolecules? Theory and applications. *IEEE Trans. Biomed. Eng.*, 41, 12, 1101–1114, 1994.
26. Chen, D., Wang, J., Yan, M., ShengBao, F., A Complex Prime Numerical Representation of Amino Acids for Protein Function Comparison. *J. Comput. Biol.*, 23, 8, 2016. https://doi.org/10.1089/cmb.2015.0178
27. Chrysostomou, C. and Seker, H., Prediction of protein allergenicity based on signal-processing bioinformatics approach, in: *Proceedings of the IEEE 36th Annual International Conference of Engineering in Medicine and Biology Society (EMBC '14)*, Aug 2014, IEEE Computer Society, Chicago, Ill, USA, pp. 808–811.
28. Sahu, S.S. and Panda, G., Efficient localization of hot spots in Proteins using a novel S-transform based filtering approach. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 8, 05, 1235–1246, Aug 2014.
29. Ramachandran, P. and Antoniou, A., Localization of Hot Spots in Proteins Using Digital Filters. *Proc. IEEE Int'l Symp, Signal Processing and Details Technology*, pp. 926–931, Aug 2006.
30. Ramachandran, P. and Antoniou, A., Identification of Hot-Spots Locations in Proteins Using Digital Filters. *IEEE J. Sel. Top. Signal Process.*, 2, 3, 378–389, June 2008.
31. George, T.P. and Thomas, T., Discrete wavelet transform de-noising in eukaryotic gene splicing. *BMC Bioinf.*, 11, supplement 1, 2010. https://www.researchgate.net/publication/51441695_Discrete_wavelet_transform_de-noising_in_eukaryotic_gene_splicing
32. Chakraborty, S. and Gupta, V., DWT based cancer identification using EIIP. *Second International Conference on Computational Intelligence & Communication Technology*, 2016.

33. G., Anitha Mary, G., Aparna Raghava Rao, G., Anjan Babu, Identification of hotspots in protein sequences using CPNR and DWT. *Int. J. Adv. Res. Comput. Sci.*, 9, 3, 219–233, May–June 2018.
34. Inbamlar, T.M. and Sivakumar, R., Improved algorithm for analysis of DNA Sequences using multiresolution transformation. *Sci. World J.*, 2015, 786497, 2015.
35. Kashyap, H., Ahmed, H.A., Hoque, N., Roy, S., Bhattacharyya, D.K., Big data analytics in bioinformatics: Architecture, techniques, tools and issues. *Netw. Model Anal. Health Inform. Bioinform.*, 13, 9, 1–20, 2016.
36. Jabbar, M.A., Deekhatulu, B.L., Chandra, P., Computational intelligence techniques for early diagnosis of heart disease. *IEEE*, 2015.
37. Chakraborty, S. and Gupta, V., DWT Based Cancer Identification Using EIIP. *Second International Conference on Computational Intelligence & Communication Technology (CICT)*, 2016.
38. Deergha Rao, K. and Swamy, M.N., *Digital Signal Processing*, Springer Science and Business Media LLC, Singapore, 2018.
39. Ostrovskii, V. and Kadyshevich, E., IntechOpen, Licensee MDPI, Basel, Switzerland, 2011.
40. Sahu, S.S. and Panda, G., Efficient Localization of Hotpots in Proteins Using a Novel S-Transform Based Filtering Approach. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, ACM 8(5):1235-46, 2011.

17

Survey of Various Statistical Numerical and Machine Learning Ontological Models on Infectious Disease Ontology

Yuvaraj Natarajan[1]*, Srihari Kannan[2] and Sachi Nandan Mohanty[3]

[1]*Research and Development, ICT Academy, Chennai, India*
[2]*Department of Computer Science and Engineering, SNS College of Technology, Coimbatore, India*
[3]*Department of Computer Science and Engineering, ICFAI Foundation of Higher Education, Hyderabad, India*

Abstract

In this chapter, we present a survey on various decision support methods that uses ontology as its tool to provide the reasoning, medical knowledge list, attributes and relationship between the clinical reasoning ontology on an infectious disease. The study in specific discusses the support of the ontology technique over infectious disease domain. Various technologies involved to improve the decision making by ontology in finding or controlling the infectious disease is discussed deeply that involves the strategies associated with testing, development and refinement of ontology in this domain. The chapter further presents the recommendations on improving the infectious disease ontology computation on translational medicine and computational application.

Keywords: Infectious disease, ontology, unsupervised learning, machine learning, decision support systems

**Corresponding author*: yraj1989@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (431–442) © 2021 Scrivener Publishing LLC

17.1 Introduction

Informatics of Infectious Diseases is a field that learns the knowledge in infectious diseases, modeling, sharing and management. The development is driven by increasing clinical and biomedical data and the demands on data analytics. The results integrating the computer data and experimental data are restructured in a way such that the research on infectious disease is carried out by increasing the expectation that infectious diseases will better be controlled [2].

The aims of the information technology on infectious diseases are elevated: to optimize the antimicrobials development, to improve the vaccine and to identify clinical results and biomarkers of infections to understand interactions of host–pathogen [2].

The computer industry has been decisive in changing the microbial world from statistic to dynamic. In the microbial world, chromosomes are not considered to change gradually in time as strictly defined genotypes but rather as temporary dynamical stability, which form tightly linked network areas. It is important to consider the whole infection cycle and to evaluate relationships between growth and transmission [2].

Biological interactions differ heterogeneously in terms of spatial and temporal relationships. The networks can be built by means of functionally assigned computational linkages and are often referred to as hubs as highly connected and central proteins. There are several commonalities that lies free from scale or smaller networks in the world. Increasingly, pathogenesis of diseases and drug action mechanisms are seen from the perspective of the biological system. A deeper understanding can be derived from a characterization of interactions between the infected cells and virus-coded proteins [2].

17.2 Disease Ontology

The semantically consistent annotations of diagnostic evaluation and comparison of data between studies, such as those available in the Disease Ontology (DO) system may be greatly encouraged [40–66].

In order to systematically record the classifications of mortality and morbidité, standardise clinical and event health reporting, and index Medline papers, or interconnect biomedicine-defined concepts through centuries of vocabulary, coding systems, thesauri and classifications that have been developed in various fields. While these vocabularies and

ontologies include concepts and expressions associated with diseases and diseases, none are 'ordered' by the disease concept [3].

The DO has been designed to build a single classification structure that integrates disease representativeness into a relational ontology among the various terminologies and vocabulary that allows the deduction and reasoning of disease relations and concepts and is optimized in the direction of disease annotation [3].

For each infectious disease, the domain ontology provides a definition within the etiologically based disease classification that enables the consistent use and use of biomedical data. The DO deals with the complexity of the nomenclature by including MeSH, OMIM, ICD, and SNOMED CT concept names and identification IDs. A framework for data mining, analysis and inference will be provided by DO web browsers that allow the examination of biomedical diseases and gene data for ongoing research and novel research based on a shared disease representation [3].

17.3 Infectious Disease Ontology

The collected DNA samples with patient health data are developed for annotation by the Disease Ontology. The broader motivation for establishing the DO to provide a vocabulary in the public domain against medical data and an annotating model for the phenotype of human illness terms.

The Disease Ontology is as a disease taxonomy that refers mainly to disease types from the international classification. The hierarchy reflects the relationship between the types of diseases. Few terms are included to date are natural speech, typically taken from the NCI thesaurus, SNOMED CT or MeSH. A strategy to align DO to the SNOMED CT-type disease improves the hierarchy of Disease Ontology and is planned to further improve DO.

Notwithstanding the DO claim that disease terms are organized on the basis of types that use an is–a relation, the DO hierarchy is structured poorly such that the it mixes infection disease types and the anatomic mixing types, infectious properties, the infection type, the stage of development, the disease's geography and the properties of the infectious agent The combination of classification modes and heritage models leads to inheritance of properties which are not subject to a type. Because of its limited extent and unorganized hierarchy of categorisation, the disease ontology is a limited utility for the infectious disease domain. However, the developers of disease ontology have made efforts to implement the reforms required.

In recent five years, biomedical ontology has shown an increase in interest, but there is a lack of computable resources for infectious diseases. The need for ontology and an opportunity to coordinate the efforts to achieve wide interoperability in the fields of disease specifics and clinical, public and biomedical research.

Various methodologies are developed on ontology modules over the infectious diseases together to form the basis for ontology effort. The methodology relies on use for domain extensions development over general infectious disease ontology (IDO). There are many advantages to this methodology.

The core infectious disease ontology assures that extensions that are interoperable, where the modular schema tends to offer the research expert to develop and maintain each module. The progress is rapid using the division of labor towards the necessary ontological sets, guarantees the biological precision and increases the widespread probability likelihood by use of research community ontologies on infectious diseases.

It is created through related terms from Open Biomedical Foundry ontologies based on the relationships between the ontology and creating new terms and relationships when necessary. The building of IDO and Open Biomedical Foundry ontologies have many benefits. Formality behind ontologies ensures sophisticated calculations are subsequently supported both within and between ontologies is a key step in ontological development.

Finally, since the use and extension of the open biomedical Foundries ontologies, in particular the gene ontology, improves the chances to accept the biological ontology and database communities by the infectious disease ontology use of foundry ontologies. Interoperability and use in a range of computing applications are tested for infectious disease ontology extensions. The ontologies are refined to improve continuously in response to these tests [1].

17.4 Biomedical Ontologies on IDO

Biomedical ontologies are a number of hierarchical terms and relationships that represent and illustrate the relationship between entities in biomedical science. The ontological terms are often expressed in formal logic, along with documentation and definitions to support automated reasoning [4].

In fields like data integration, knowledge management, semantic and exchange interoperability, reasoning and decision support, where biomedical ontology play an important role. An infectious disease ontology consortium has been established to support translational health research

in infectious diseases. Infectious disease ontology intends to develop an interoperable ontology that the infectious diseases, covering the specialties of infectious diseases and clinical, public and biomedical research [4].

The ontology suite of infectious disease ontology is developed using a core expansion approach that developed disease or pathogenic ontologies as a common infectious disease ontology core expansion ontology. The infectious disease ontology core ontology covers those entities that generally deal with infectious diseases [4].

In the meantime, the infectious disease ontology extensions include entities which are relevant to particular types of disease or pathogen. The infectious disease ontology core has natural language processing for logical expressions among infectious disease relationship ontology, such as Gene Ontology (GO) [4].

Experts from specific areas of infectious diseases maintain specific IDO extension ontologies. Various IDO expansion ontologies are coordinated from the core to ensure that all IDO expansion ontologies are interoperable [4].

The CDSMS is designed and built on the basis of CDSMKB. Mobile chronic disease auto-management system. The web framework of the Jena semantic web processes data into semantic data, which is stored in a data semantic models for transforming the clinical data of patients. This method further provides treatment and adaptation plans for the chronic disease based on reasoning rules and it further improves the effectiveness and operatively of the self-management of chronic diseases [15].

17.5 Various Methods on IDO

A Brucellosis Ontology development is reported by Ref. [4] as an infectious disease ontology that targets the brucellosis domain. It uses OWL format, which is an automated OWL reasoning can be carried on OWL editors using reasoning programs, which develops a SPARQL query and a rationalization programs. This approach combines an advanced and an advanced methodology of realism. With regard to the top down method we started with the development of the IDO core and its terms belongs to superclass types. The bottom-up procedure is to detect and then generalize the most specific terms into higher ontology classes by means of scenarios.

The Schistosomiasis domain knowledge and dynamics propagation have been analyzed and designed by Ref. [5]. IDOSCHISTO—the ontology of schistosomiasis—has been developed to extend the infectious disease ontology. This ontology aims at monitoring the schistosomiasis

during the time of spreading and the risk assessment is carried out via data integration, cooperation, semantic interoperability and risk analysis and decision-making.

The ontology development complemented by reusing the Open Biomedical Ontologies (OBO) foundry of terms from orthogonal ontology [6]. The study of plant disease illustrates the ontology framework utilisation over models, biological events, and the difference and similarity between infectious diseases in relation with other organisms.

A system which automatically categorizes and classifies reports of infectious diseases by type and by geographical location for the analysis of domain experts was described by Ref. [7]. Infectious disease references are identified by using ontology of a disease. The system uses the STEWARD textual and spatial search capabilities to allow queries, like reports, within a certain time period. Documents are tagged to allow for the graphical presentation of spatial relations with certain conditions via a map interface. Furthermore, the tag and index of newspaper articles can support the monitoring of continuing epidemics.

The Protegé OWL files evaluate on BCIDO dataset and it uses BCIDO to get operated on a secured environment [8].

The estimation of decision support system that optimizes an ontology for diagnosing the antibiotic treatment and infectious diseases [9]. The ontology is constructed through the collection into the hierarchical conceptual schema of ontology linked to infectious diseases, syndromes and drugs. This method identifies the infectious disease using self-described condition of a patient. In the following, the system looks and offers an appropriate antibiotic therapy specifically adapted to its patients on basis of factors such as the body temperature of the patient, infection sites, symptoms/signs, complications, antibacterial spectrum, contraindications, drug–drug interactions and the therapeutic route.

The ontology decision support method is further used for diagnosing the infectious disease and antibiotics therapy, which is introduced by [10]. This stand-alone method identifies the illnesses associated with infectious and offers antibiotic therapy based on patient-entry information. A backbone for the representation was developed for rare diseases in humans through cross-cutting methods [11].

17.6 Machine Learning-Based Ontology for IDO

A knowledge base risk prediction model is the mechanism for generating a Bayesian Network (BN) and has examined the accuracy of the prediction

result. The research in this document was initiated with the encoding in an infectious disease risk ontology of declarative knowledge from the Atlas of Human Infectious Diseases. The BN generation automatically uses two tools from this knowledge. A rule converter generates an ontological BN structure. A tool to supply joint and marginal probability adds probabilities to the BN. These tools enable BN to be automatically recreated when knowledge and data change [12].

An introduced human risk prediction system based on the environments and the features to mitigate the infectious disease [13]. The prediction system assists the experts on risk prediction of infectious diseases. The prediction system is designed to representation the medical knowledge that assists in encoding the knowledge on infectious diseases risk and developing an algorithm to convert automatically the coded knowledge into a probability model for risk assessment. The knowledge about the risk on infectious disease representation has the ontology and the set of rules that represents and quantifies the structure of knowledge that allows predictive model generated via Bayesian Network. The Bayesian Network-Builder algorithm converts the risk knowledge base to Bayesian Network risk for infectious disease that includes the prediction based on infectious disease risk. The epidemiological knowledge can be coded into an infectious disease risk know how base using the user interface.

A biomedical ontology is developed for chronic liver disease with the UMLs as a database source. Since the data is of unstructured format and upon failure to access the information, ontologies help structuring existing data [14]. This method adopts resource description frame (RDF) for the development of secured schemes in order to describe concepts and relationships within chronic liver diseases. This is how the ontology is formed for the biomedical data.

17.7 Recommendation or Suggestions for Future Study

In future, the study can be improved by integrating the machine learning and deep learning models [16–39] with ontologies in finding the ontologies related to infectious disease spread. The existing techniques on ontologies needs to be studied more to improve the studies related to ontologies on infectious disease spread. Further the studies have to be carried out with more recommendations to improve the ontology computational on different translational medical with infectious disease spread.

17.8 Conclusions

In this chapter, a deep survey on various techniques of clinical decision support methods including statistical, numerical and machine learning techniques is discussed. These techniques enhance the ontology domain to offer improved reasoning, medical knowledge list, attributes and relationship between the clinical reasoning ontology over infectious disease control. The study shows that the ontology utilized under different technologies improves well the decision-making ability of ontology in controlling the infectious disease rate and this enhances the testing, development and ontology refinement in infectious disease domain. Finally, it is seen that the study provides a bridge between the medical field and the computer science field in the encoding and prediction of infectious disease.

References

1. Cowell, L.G. and Smith, B., Infectious disease ontology, in: *Infectious Disease Informatics*, pp. 373–395, Springer, New York, NY, 2010.
2. Sintchenko, V., Informatics for Infectious Disease Research and Control, in: *Infectious Disease Informatics*, pp. 1–26, Springer, New York, NY, 2010.
3. Schriml, L.M., Arze, C., Nadendla, S., Chang, Y.W.W., Mazaitis, M., Felix, V., Kibbe, W.A., Disease Ontology: A backbone for disease semantic integration. *Nucleic Acids Res.*, 40, D1, D940–D946, 2012.
4. Lin, Y., Xiang, Z., He, Y., Brucellosis Ontology (IDOBRU) as an extension of the Infectious Disease Ontology. *J. Biomed. Semantics*, 2, 1, 9, 2011.
5. Camara, G., Despres, S., Djedidi, R., Lo, M., Design of schistosomiasis ontology (IDOSCHISTO) extending the infectious disease ontology. *Stud. Health Technol. Inform.*, 192, 466–470, 2013.
6. Walls, R., Smith, B., Justin, E., Albert, G., Dennis, W.S., A plant disease extension of the Infectious Disease Ontology, in: *Proceeedings of the Third International Conference on Biomedical Ontology (CEUR 897)*, pp. 1–5, 2012.
7. Lieberman, M.D., Sankaranarayanan, J., Samet, H., Sperling, J., Augmenting spatio-textual search with an infectious disease ontology, in: *2008 IEEE 24th International Conference on Data Engineering Workshop*, IEEE, pp. 266–269, 2008.
8. Gordon, C.L. and Weng, C., Bacterial clinical infectious diseases ontology (BCIDO) dataset. *Data Brief*, 8, 881–884, 2016.
9. Shen, Y., Yuan, K., Chen, D., Colloc, J., Yang, M., Li, Y., Lei, K., An ontology-driven clinical decision support system (IDDAP) for infectious disease diagnosis and antibiotic prescription. *Artif. Intell. Med.*, 86, 20–32, 2018.

10. Shen, Y., Deng, Y., Zhang, J., Li, Y., Du, N., Fan, W., Lei, K., IDDAT: An Ontology-Driven Decision Support System for Infectious Disease Diagnosis and Therapy, in: *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, IEEE, pp. 1417–1422, 2018.
11. Bernasconi, A. and Masseroli, M., Biological and Medical Ontologies: Disease Ontology (DO), in: *Reference Module in Life Sciences: Encyclopedia of Bioinformatics and Computational Biology*, vol. 1, pp. 838–847, 2019.
12. Vinarti, R.A. and Hederman, L.M., A Personalized infectious disease risk prediction system. *Expert Syst. Appl.*, 131, 266–274, 2019.
13. Vinarti, R.A. and Hederman, L., Personalization of Infectious Disease Risk Prediction: Towards Automatic Generation of a Bayesian Network, in: *2017 IEEE 30th International Symposium on Computer-Based Medical Systems (CBMS)*, IEEE, pp. 594–599, 2017.
14. Wahab, K., Qamar, U., Arif, K.S., Ali, U., Building a Biomedical Ontology for Chronic Liver Disease, in: *2019 International Conference on Computer, Information and Telecommunication Systems (CITS)*, IEEE, pp. 1–5, 2019.
15. Zheng, C., Gou, L., Zhao, Y.X., Lu, Y., Wang, F., Zhou, T.S., Li, J.S., A chronic disease self-management system based on OWL-based ontologies and semantic rules, in: *2016 8th International Conference on Information Technology in Medicine and Education (ITME)*, IEEE, pp. 1–6, 2016.
16. Ayadi, A., Samet, A., de Beuvron, F.D.B., Zanni-Merk, C., Ontology population with deep learning-based NLP: A case study on the Biomolecular Network Ontology. *Procedia Comput. Sci.*, 159, 572–581, 2019.
17. Greenbaum, N.R., Jernite, Y., Halpern, Y., Calder, S., Nathanson, L.A., Sontag, D.A., Horng, S., Improving documentation of presenting problems in the emergency department using a domain-specific ontology and machine learning-driven user interfaces. *Int. J. Med. Inf.*, 132, 103981, 2019.
18. Lakshmi, K.S. and Vadivu, G., Network based approach for discovering novel gene-phenotypic association and disease co morbidities using ontological data. *Procedia Comput. Sci.*, 167, 819–829, 2020.
19. Lamy, J.B., Sedki, K., Tsopra, R., Explainable decision support through the learning and visualization of preferences from a formal ontology of antibiotic treatments. *J. Biomed. Inf.*, 104, 103407, 2020.
20. Albukhitan, S., Alnazer, A., Helmy, T., Framework of Semantic Annotation of Arabic Document using Deep Learning. *Procedia Comput. Sci.*, 170, 989–994, 2020.
21. Crawford, J. and Greene, C.S., Incorporating biological structure into machine learning models in biomedicine. *Curr. Opin. Biotechnol.*, 63, 126–134, 2020.
22. Harrow, I., Balakrishnan, R., Jimenez-Ruiz, E., Jupp, S., Lomax, J., Reed, J., Woollard, P., Ontology mapping for semantically enabled applications. *Drug Discov. Today*, 24, 10, 2068–2075, 2019.
23. Gao, W., Guirao, J.L., Basavanagoud, B., Wu, J., Partial multi-dividing ontology learning algorithm. *Inf. Sci.*, 467, 35–58, 2018.

24. Gao, W., Baig, A.Q., Ali, H., Sajjad, W., Farahani, M.R., Margin based ontology sparse vector learning algorithm and applied in biology science. *Saudi J. Biol. Sci.*, 24, 1, 132–138, 2017.
25. Liu, K., Hogan, W.R., Crowley, R.S., Natural language processing methods and systems for biomedical ontology learning. *J. Biomed. Inf.*, 44, 1, 163–179, 2011.
26. Cardoso, S.D., Pruski, C., Da Silveira, M., Supporting biomedical ontology evolution by identifying outdated concepts and the required type of change. *J. Biomed. Inf.*, 87, 1–11, 2018.
27. Ali, M., Fathalla, S., Ibrahim, S., Kholief, M., Hassan, Y., Cross-lingual ontology enrichment based on multi-agent architecture. *Procedia Comput. Sci.*, 137, 127–138, 2018.
28. Alobaidi, M., Malik, K.M., Hussain, M., Automated ontology generation framework powered by linked biomedical ontologies for disease-drug domain. *Comput. Methods Programs Biomed.*, 165, 117–128, 2018.
29. Kastrati, Z., Imran, A.S., Yayilgan, S.Y., The impact of deep learning on document classification using semantically rich representations. *Inf. Process. Manag.*, 56, 5, 1618–1632, 2019.
30. Roohi, A., Faust, K., Djuric, U., Diamandis, P., Unsupervised Machine Learning in Pathology: The Next Frontier. *Surg. Pathol. Clin.*, 13, 2, pp. 349–358, 2020.
31. Cardoso, S.D., Da Silveira, M., Pruski, C., Construction and exploitation of an historical knowledge graph to deal with the evolution of ontologies. *Knowl.-Based Syst.*, 105508, 2020.
32. Le, D.H. and Dao, L.T., Annotating diseases using human phenotype ontology improves prediction of disease-associated long non-coding RNAs. *J. Mol. Biol.*, 430, 15, 2219–2230, 2018.
33. Annane, A., Bellahsene, Z., Azouaou, F., Jonquet, C., Building an effective and efficient background knowledge resource to enhance ontology matching. *J. Web Semant.*, 51, 51–68, 2018.
34. Phan, N., Dou, D., Wang, H., Kil, D., Piniewski, B., Ontology-based deep learning for human behavior prediction with explanations in health social networks. *Inf. Sci.*, 384, 298–313, 2017.
35. Phillips, M.H., Serra, L.M., Dekker, A., Ghosh, P., Luk, S.M., Kalet, A., Mayo, C., Ontologies in radiation oncology. *Physica Med.*, 72, 103–113, 2020.
36. Kassahun, Y., Perrone, R., De Momi, E., Berghöfer, E., Tassi, L., Canevini, M.P., Kirchner, F., Automatic classification of epilepsy types using ontology-based and genetics-based machine learning. *Artif. Intell. Med.*, 61, 2, 79–88, 2014.
37. Shen, Y., Yuan, K., Chen, D., Colloc, J., Yang, M., Li, Y., Lei, K., An ontology-driven clinical decision support system (IDDAP) for infectious disease diagnosis and antibiotic prescription. *Artif. Intell. Med.*, 86, 20–32, 2018.
38. Huitzil, I., Dranca, L., Bernad, J., Bobillo, F., Gait recognition using fuzzy ontologies and Kinect sensor data. *Int. J. Approximate Reasoning*, 113, 354–371, 2019.
39. Huitzil, I., Dranca, L., Bernad, J., Bobillo, F., Gait recognition using fuzzy ontologies and Kinect sensor data. *Int. J. Approximate Reasoning*, 113, 354–371, 2019.

40. Dramé, K., Diallo, G., Delva, F., Dartigues, J.F., Mouillet, E., Salamon, R., Mougin, F., Reuse of termino-ontological resources and text corpora for building a multilingual domain ontology: an application to Alzheimer's disease. *J. Biomed. Inf.*, 48, 171–182, 2014.
41. Zhang, Y.F., Gou, L., Zhou, T.S., Lin, D.N., Zheng, J., Li, Y., Li, J.S., An ontology-based approach to patient follow-up assessment for continuous and personalized chronic disease management. *J. Biomed. Inf.*, 72, 45–59, 2017.
42. LePendu, P., Musen, M.A., Shah, N.H., Enabling enrichment analysis with the Human Disease Ontology. *J. Biomed. Inf.*, 44, S31–S38, 2011.
43. Topalis, P., Dialynas, E., Mitraka, E., Deligianni, E., Siden-Kiamos, I., Louis, C., A set of ontologies to drive tools for the control of vector-borne diseases. *J. Biomed. Inf.*, 44, 1, 42–47, 2011.
44. Cannataro, M., Guzzi, P.H., Milano, M., GoD: An R-package based on ontologies for prioritization of genes with respect to diseases. *J. Comput. Sci.*, 9, 7–13, 2015.
45. Liaw, S.T., Taggart, J., Yu, H., de Lusignan, S., Kuziemsky, C., Hayen, A., Integrating electronic health record information to support integrated care: Practical application of ontologies to improve the accuracy of diabetes disease registers. *J. Biomed. Inf.*, 52, 364–372, 2014.
46. Chi, Y.L., Chen, T.Y., Tsai, W.T., A chronic disease dietary consultation system using OWL-based ontologies and semantic rules. *J. Biomed. Inf.*, 53, 208–219, 2015.
47. Colombo, G., Merico, D., Boncoraglio, G., De Paoli, F., Ellul, J., Frisoni, G., Antoniotti, M., An ontological modeling approach to cerebrovascular disease studies: The NEUROWEB case. *J. Biomed. Inf.*, 43, 4, 469–484, 2010.
48. Yu, M.K., Ma, J., Ono, K., Zheng, F., Fong, S.H., Gary, A., Ideker, T., DDOT: A Swiss Army Knife for Investigating Data-Driven Biological Ontologies. *Cell Syst.*, 8, 3, 267–273, 2019.
49. Liaw, S.T., Rahimi, A., Ray, P., Taggart, J., Dennis, S., de Lusignan, S., Talaei-Khoei, A., Towards an ontology for data quality in integrated chronic disease management: A realist review of the literature. *Int. J. Med. Inf.*, 82, 1, 10–24, 2013.
50. Marquet, G., Mosser, J., Burgun, A., A method exploiting syntactic patterns and the UMLS semantics for aligning biomedical ontologies: The case of OBO disease ontologies. *Int. J. Med. Inf.*, 76, S353–S361, 2007.
51. Refolo, L.M., Snyder, H., Liggins, C., Ryan, L., Silverberg, N., Petanceska, S., Carrillo, M.C., Common Alzheimer's disease research ontology: National Institute on Aging and Alzheimer's Association collaborative project. *Alzheimers Dement.*, 8, 4, 372–375, 2012.
52. Baneyx, A., Charlet, J., Jaulent, M.C., Building an ontology of pulmonary diseases with natural language processing tools using textual corpora. *Int. J. Med. Inf.*, 76, 2–3, 208–215, 2007.
53. Tang, X., Chen, W., Zeng, Z., Ding, K., Zhou, Z., An ontology-based classification of Ebstein's anomaly and its implications in clinical adverse outcomes. *Int. J. Cardiol.*, 316, 79–86, 2020.

54. Gupta, A., Ludäscher, B., Grethe, J.S., Martone, M.E., Towards a formalization of disease-specific ontologies for neuroinformatics. *Neural Netw.*, 16, 9, 1277–1292, 2003.
55. Singleton, M.V., Guthery, S.L., Voelkerding, K.V., Chen, K., Kennedy, B., Margraf, R.L., Huff, C.D., Phevor combines multiple biomedical ontologies for accurate identification of disease-causing alleles in single individuals and small nuclear families. *Am. J. Hum. Genet.*, 94, 4, 599–610, 2014.
56. Kiourtis, A., Nifakos, S., Mavrogiorgou, A., Kyriazis, D., Aggregating the syntactic and semantic similarity of healthcare data towards their transformation to HL7 FHIR through ontology matching. *Int. J. Med. Inf.*, 132, 104002, 2019.
57. Hausknost, D., & Haas, W., The Politics of Selection: Towards a Transformative Model of Environmental Innovation. *Sustainability*, 11, 2, pp. 506, 2019.
58. Nasiri, S., Zahedi, G., Kuntz, S., Fathi, M., Knowledge representation and management based on an ontological CBR system for dementia caregiving. *Neurocomputing*, 350, 181–194, 2019.
59. Meyer, D.E., Bailin, S.C., Vallero, D., Egeghy, P.P., Liu, S.V., Hubal, E.A.C., Enhancing life cycle chemical exposure assessment through ontology modeling. *Sci. Total Environ.*, 712, 136263, 2020.
60. Groza, T., Köhler, S., Moldenhauer, D., Vasilevsky, N., Baynam, G., Zemojtel, T., Vasant, D., The human phenotype ontology: Semantic unification of common and rare disease. *Am. J. Hum. Genet.*, 97, 1, 111–124, 2015.
61. Zheng, L., Chen, Y., Elhanan, G., Perl, Y., Geller, J., Ochs, C., Complex overlapping concepts: An effective auditing methodology for families of similarly structured BioPortal ontologies. *J. Biomed. Inf.*, 83, 135–149, 2018.
62. Fraser, S., Doing ontopolitically-oriented research: Synthesising concepts from the ontological turn for alcohol and other drug research and other social sciences. *Int. J. Drug Policy*, 82, 102610, 2020.
63. Shen, Y., Yuan, K., Chen, D., Colloc, J., Yang, M., Li, Y., & Lei, K., An ontology-driven clinical decision support system (IDDAP) for infectious disease diagnosis and antibiotic prescription. *Artif. Intell. Med.,* 86, 20–32, 2018.
64. Cardoso, S.D., Pruski, C., Da Silveira, M., Supporting biomedical ontology evolution by identifying outdated concepts and the required type of change. *J. Biomed. Inf.*, 87, 1–11, 2018.
65. Stevens, R., Lord, P., Malone, J., Matentzoglu, N., Measuring expert performance at manually classifying domain entities under upper ontology classes. *J. Web Semant.*, 57, 100469, 2019.
66. Robinson, P.N., Köhler, S., Bauer, S., Seelow, D., Horn, D., Mundlos, S., The Human Phenotype Ontology: A tool for annotating and analyzing human hereditary disease. *Am. J. Hum. Genet.*, 83, 5, 610–615, 2008.

18

An Efficient Model for Predicting Liver Disease Using Machine Learning

Ritesh Choudhary[1], T. Gopalakrishnan[2]*, D. Ruby[2], A. Gayathri[3], Vishnu Srinivasa Murthy[2] and Rishabh Shekhar[4]

[1]*School of Computing and IT, Manipal University Jaipur, India*
[2]*School of Computer Science and Engineering (SCOPE), Vellore Institute of Technology, Vellore, India*
[3]*School of Information Technology (SITE), Vellore Institute of Technology, Vellore, India*
[4]*VIT Business School (VITBS), Vellore Institute of Technology, Vellore, India*

Abstract

Liver Diseases account for more than 2.8% of Indian adult deaths each year. The major issue with finding out the liver disease is difficulty in diagnosing its early stages which will be showing subtle symptoms. Mostly the liver disease will show its symptoms only in the advanced stage, which causes difficulty in identification and diagnosing. This article looks to improve diagnosis methods for liver diseases identification by verifying different patient parameters and their respective genome expression. This paper also analyzes the various computational algorithms that can be used in the above-mentioned methodologies to find out the best models. It also proposes various methods to improve the efficiency of these algorithms.

Keywords: Machine learning, bioinformatics, liver disease, SVM, gradien boosting

**Corresponding author*: gopalakrishnan.ct@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (443–458) © 2021 Scrivener Publishing LLC

18.1 Introduction

The Liver is an intense and vital organ in the human body. It consists of two large portions, known as the left projections and privilege. The Gallbladder, Pancreas, and Stomach are some organs that reside nearby the Liver. Liver and their organs collaborate to ingest and process substances. A job of the Liver is to remove the harmful substances from the blood beginning from the framework, which is related to the stomach prior crossing it to whatsoever is left in the body. Liver disease can be adopted either from outside factors like infections or alcohol. Corpulence has additionally been related to this illness. Overtime harm to the Liver can cause liver failure and sometimes lead to a life-threatening condition.

Therefore, accurate and efficient diagnosis of a person at initial traceable and risk of Liver Diseases might provide huge aids for prevention, diagnosis, or the right treatment. From the past decade, biopsy was the method used to stratify the patients, called a diagnostic reference standard for finding of liver with fatty intrusions. This procedure is large expensive & cumbersome; it might also provide the sampling errors and side effects when the application of this method conducted. Ultrasonography acts as a functional tool for the diagnosis tic of Liver Diseases with colossal accuracy, and identifying efficiency which is mostly operator dependent [1]. Liver Diseases are tricky to trace provided the subtlety of the symptoms in their initial stages. It is claimed by doctors that the problems with a liver are not traceable till it is not too late (as nearly 50%), until which, the Liver remains to function even after some portion damaged [2]. Initial tracing can be a life-saving act.

Hence, traditional detection techniques now do not prevail of much importance until a late stage diagnosis. So, apart from clinical methods, modern-age technology believes in highly sophisticated computational techniques which help to simplify work extensively with a higher rate of accuracy and minimal error. Artificial Intelligence and machine learning are among the major domains and applications of the computer science. It uses the implementation of the computer algorithms to determine the massive dataset patterns and also helps to forecast the different data-based outcomes [3]. In the presence of the clinical data, ML plays a vital role for the decision making in the medical field [4], [5]. Various machine learning models of development will act as essential aids to determine disease and would result in better decisions. It allows the hospital resources optimization by dividing the right patients with critical various earlier risk factors.

The prediction model is based on the various variables of clinical test which will help the lab clinicians and technicians to calculate the disease accurately and making a specific decision for early diagnosis, prevention, and specific treatment. There are not much implementations that have been done in past to identify the liver disease for the available electronic data received by the various medical sensors. Hence, we aim to design the forecasting model for the liver disease by applying the recent machine learning technologies which will be very useful for classification purpose. This article will use various methods of mathematical ML models for the liver disease diagnosing.

18.2 Related Works

SVM allows solving linear and non-linear binary problems that help to construct a hyper-plane, specifically in the space of hyper-dimensional area. The main vital role of this research work is to attain class separation in the liver disease diagonising. The distance is calculated with respect to the adjoining training data point of any class available or present in the specific hyper plane separation process will be carried out. This can be briefed as, greater the margin value, the rationalization or the normalized fault in the classifier is dropped and reduced. For SVM activity, the values are in between the range of data [11]. The advantage of SVM is the presence of large dimensional space for storage and efficient in memory space occupation.

In statistical method, logistic regression is used to determine the binary outcome based on the dataset. In the dataset, either at least one or much independent variable is considered for calculation. The results are derived by the variables which are dichotomous in nature. This results into two possible outcomes [9]. This is a special regression category which is used to analyze the inputs of binary and definite values.

Logistical regression is performed to adjust the influence of autonomous variables in a large manner at the same instance of time. In addition, logistical regression methods analyze any one of the two independent types of variables. The design involved in logistical regression incorporates the best fitting function that helps to the maximum likelihood method to provide maximum probability of categorizing the recognized data into the correct division [10]. Further, there exist various appliances associated in logistic regression to predict and explore the success and failure results of the latest market trends [12, 14].

Naïve Bayes comes under a generative model which involves in dealing with missing values much easier [13]. In this classification, prediction can be done by y, the class label which has a feature vector $x = [y1, y2, \ldots yd]^T$ and aids to produce interference for a new sample $Y_{new} = [y1, y2 \ldots yd]^T$ with a missing feature y_m.

18.3 Proposed Model

18.3.1 Elements of Experimental Methodology

The outline of the experimental approach is shown in Figure 18.1. The experimental analysis in this work contains five steps. The first is an overall overview and analysis of the dataset generated for better understanding of liver diseases. This includes analyzing features which have a greater dependency of having a liver disease or not, and so on. The second step is to apply Data Preprocessing techniques like Standardization of the feature variables and label encoding the target variable, for better predictions. Third step is applying various machine learning algorithms over the dataset to build our models. Fourth step is computing various performance metrics of the models built in step 3, and comparing each model over these metrics to choose the best-fitted model. The final step is optimizing the results of the best model by applying a technique called Cross Validation, by which the best fit for the model can be found and the best probable accuracy can be computed for our prediction whether or not the patient is suffering from a liver disease. A comprehensive discussion of the phases is provided below.

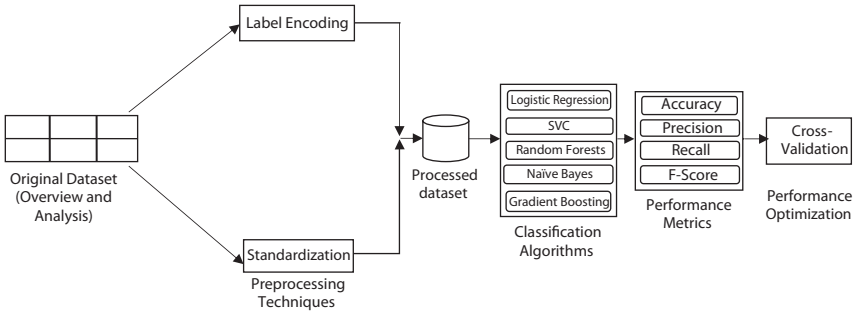


Figure 18.1 Overview of experimental methodology.

18.3.1.1 Experimental Dataset

The data are gathered from various medical laboratories, hospitals and medical centers. From this a mock Indian Liver Patient Dataset (ILPD) dataset is generated for analysis of liver disease. This collection of data includes 416 tuples of liver decease patient and 167 tuples of non-liver patients from Andhra Pradesh, India. The "Dataset" class label is accessed to classify groups into liver disease patient or not. The collection of data includes 142 records of female patients and 441 records of male patients. The attributes in this Dataset are: Age and Gender of the patient, Direct Bilirubin, Albumin and Globulin Ratio, Total Bilirubin, Alamine Aminotransferase, Albumin, Total Proteins, Alkaline Phosphotase. The data was further cleaned to remove noises and missing values and outliers were also dealt in the process. We also applied Data Normalization techniques to normalize certain variables for better feature preprocessing and getting better outputs.

18.3.1.2 Overview of Data & Analysis

We do this step for gaining valuable insights from the data. For this, we will be using Numpy and Pandas libraries available by the Python community. First, we check the shape of the dataset, which is (58,311). Then we check the data types of all variables to check for any non-string values present in the features (as shown in Table 18.1). The output to the code for checking it is shown in Figure 18.2. Hence, what we see are all integers and floating-point numbers. Next, we check for missing values, if present in the dataset. We replace these with the mean of the entire feature space.

Finally, we check for the correlation amongst each and every feature with the other features. The correlation coefficient is used in the algorithm is based on the direction and strength from the linear relationship between two numerical variables x and y. 'r' denotes correlation coefficient for a data sample. The correlation coefficient (r) is evaluated and calculated using the expression given as mentioned,

$$r = \frac{1}{n-1}\left[\frac{\sum_x \sum_y (x-x')(y-y')}{s_x s_y}\right] \tag{18.1}$$

where number of pairs of data is denoted by n; x' repersents the sample means of all the x-values and y' represents the y-values; and s_x represents

Table 18.1 Data types of features.

Features	Data Type
Age	Integer
Gender	Object
Alkaline Phosphotase	Integer
Direct Bilirubin	Float
Total Bilirubin	Float
Alamine Aminotransferase	Integer
Albumin and Globulin Ratio	Float
Albumin	Float
Dataset	Object
Total Proteins	Integer
Aspartate Aminotransferase	Float

the sample standard deviation of x and s_y represents the sample standard deviations of all y-values, respectively. Hence, we obtain a correlation matrix depicted by a Correlation Heatmap as shown in Figure 18.2.

As we can see infer from Figure 18.2, we need to check how well are all variables correlated or are dependent on the 'Dataset', which is our target variable. As we see, the only variables that show positive correlation coefficient with the target variable are 'Albumin and Globulin Ratio' 'Albumin' and 'Total Proteins'. Hence, these features are interdependent with the target variable, while other variables showing negative correlation coefficient, which signifies that, these variables shows a direct opposite relationship with the target variable.

18.3.1.3 Data Preprocessing

In this step, Dataset is classified into Training and Test sets using Scikit-Learn's method called "train_test_split". Training set is that set by which we train or fit the model with appropriate machine learning models. This is basically the data from which the models learn by which it can make future predictions. On the other hand, Test set is the set by which we evaluate the model which was trained using the training data. Please note that while training the model, the test data is left hidden from the model and is just provided post

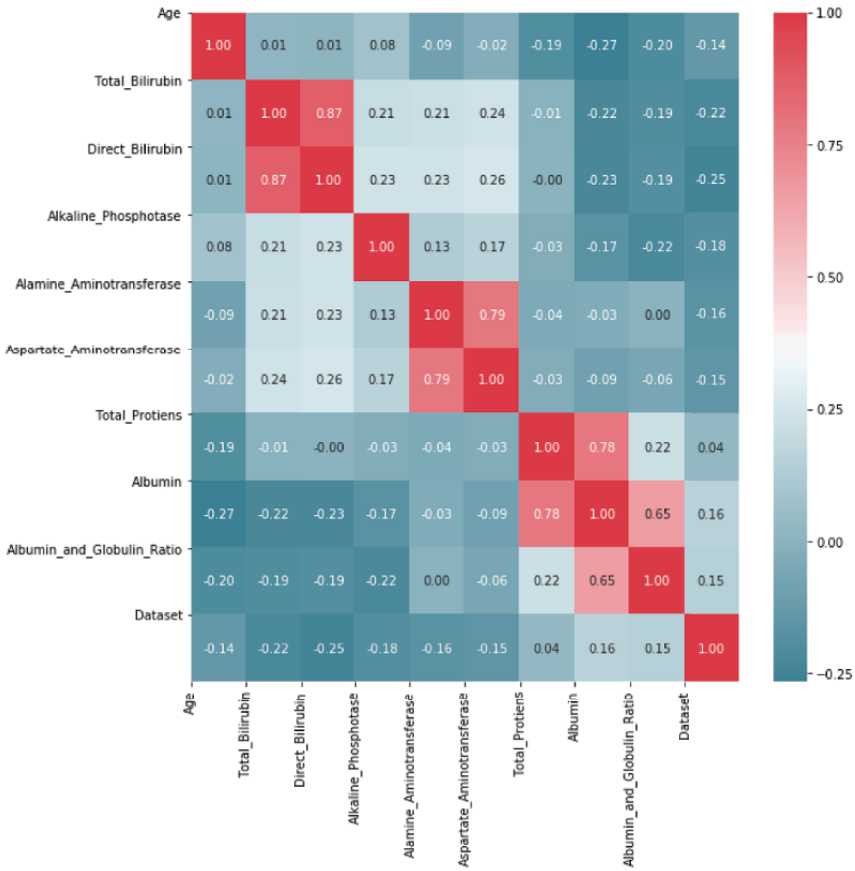


Figure 18.2 Correlation heatmap.

training the model. In our case, the data is divided into training and test sets in 70:30 ratios. If we keep a larger proportion to training the model, it will lead to overfitting during prediction, while on the contrary, if we take a larger proportion of test set, then the model will underfitting during prediction. Next, we apply the process called as feature scaling or Data Standardization (commonly referred as Data Normalization) and Label Encoding.

18.3.1.4 Standardization

The features get rescaled by using Standardization [7] method, standard normal distribution properties $\mu = 0$ (Average) and $\sigma = 1$ (Mean) is calculated; The z scores or standard scores are computed as follows:

$$z = \frac{x - \mu}{\sigma} \tag{18.2}$$

Standardizing the features in such a way that they are focused around 0 with a standard deviation of 1represesntation as it helps in eliminating the bias and variance to some degree which actually helps in enhancing the training model.

18.3.1.5 Label Encoding

The key factor to obtain the numerical columns from the categorical columns is to make a machine learning algorithm recognizes it. The categorical encoding is the process which converts categories to numbers. There are two types of encoding method namely label and One-hot encoding.

Label encoding in any machine learning application refers to the processes of converting labels into numeric values in order to make it understandable for machines. For illustration (as shown in Figure 18.3), let's have any column of level in a dataset which includes beginners, intermediate and advanced. After applying the label encoder, it will be converted into 0, 1 and 2 respectively.

18.3.2 Model Building

Different types of machine learning algorithms are incorporated for model building, to construct various models. The model consists of Random Forests, Support Vector Machines (SVM), Gradient Boosting, Logistic Regression (LR) and Naïve Bayes.

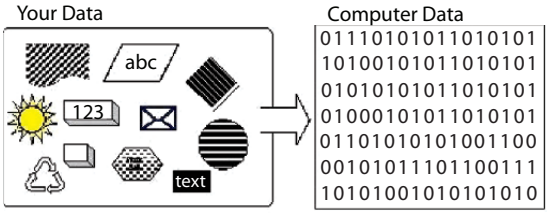
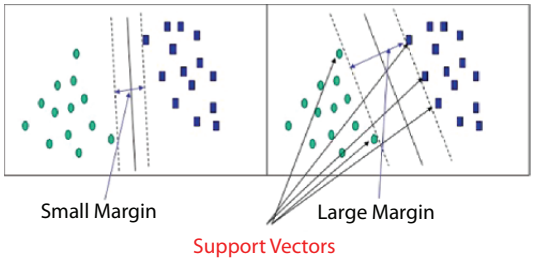


Figure 18.3 Encoding process.

18.3.2.1 Support Vector Machines (SVM)

In SVM, generally hyperplanes are considered to be the decision boundaries based on the specific data points in the classification process. SVM are the critical elements of the training set. Data points that fall on different boundaries are classified as two different classes [10].



In mentioned SVM algorithm the objective is to always have a maximum distances between the two data points or the hyperplane. The loss function of the margin can be calculated as the following:

$$c(x, y, f(x)) = \begin{cases} 0, & \text{if } y * f(x) \geq 1 \\ 1 - y * f(x), & \text{else} \end{cases} \tag{18.3}$$

$$c(x,y,f(x)) = (1 - y * f(x))_{+} \tag{18.4}$$

18.3.2.2 Logistic Regression

Logistic regression can be adopted for binary classification problem which is also used as go-to method. This method is labeled for the function which has the logistic function. It also known termed as logistic functions.

$$1/(1 + e\text{^-value}) \tag{18.5}$$

Given below is a sample equation of logistic regression:

$$y = e\text{^} (a_0 + a_1 * x)/(1 + e\text{^} (a_0 + a_1 * x)) \tag{18.6}$$

Where the predicted output is denoted by Y, a_0 is the intercept or bias value and a_1 is the coefficient for the single input value (x).

18.3.2.3 Naïve Bayes

Naive Bayes are a cluster of classification algorithms that comes under Bayes' Theorem. It is a mass collection of algorithms that show a principle that is common in nature, i.e. each pair of features is classified and independent to one other.

Consider a dataset that has fictional type. The environment explains the conditions of weather for playing a golf game. The condition of weather indicates, each tuple classifies as fit ("Yes") or unfit ("No") for playing golf.

Bayes' Theorem is given as:

$$P(h|d) = (P(d|h) * P(h))/P(d) \tag{18.7}$$

18.3.2.4 Random Forests

Random forest otherwise called as random decision forests for a specific type of algorithm namely ensemble that uses a mixture combination of decision based trees and data subsets. Mostly random forest algorithm won't produce a decision tree for lesser decision trees for prediction in parallel manner. Furthermore, random forest algorithms is very accurate when compare to single decision trees.

Random forest considers the CART-classification and the regression trees which will train on various collections of datasets bootstraps. To test set, OOB samples are used. To organize new information, every CART tree (in Figure 18.1) votes with respect to one class in addition timberland calculates class that require majority of votes.

18.3.2.5 Gradient Boosting

GBM—Gradient boosting machines are the algorithm that proved success history across various machine learning domains. In addition it is also one of the superior methods for gaining success in Kaggle competitions. In, random forests build a group of deep independent trees where else in the GBMs constructed with the right sequence with each tree learning and improving comparing the existing with the clustering of shallow trees. Overall the predicted week shallow trees can be used to boost powerful committee by properly tuning. This chapter will cover the fundamental understanding and popular implementations of GBMs.

18.3.3 Performance Evaluation

Accuracy
The best intuitive performance measure is the accuracy and it measures the number of correctly predicted observation of data points out of total number of observations. The model is considered to be the best if it has high accuracy. Certainly, accuracy is a best indicator and an issue is it works well only if we have similar datasets where values of false negative and false positive are almost the same. So, it is necessary to have a look at various other parameters also to assess the performance of the model. Accuracy is the one of the most intuitive performance measure.

$$\text{Accuracy} = \text{True Negative} + \text{True Positive}/\text{All Samples}$$

Precision
The ratio of the number of correctly predicted positive observations that belong to the class of predicted positive observations is called Precision (P). The query is that, how many actually survived from all passengers that labeled as survived? High precision relays to the low false positive rate.

$$\text{Precision} = \text{True Positive}/\text{Total positive predicted}$$

Recall
The ratio of correctly anticipated positive readings to all the readings in actual class is Recall (R) also called as Sensitivity.

$$\text{Recall} = \text{True Positive}/(\text{False Negative} + \text{True Positive})$$

F1 Score
F1 Score is the mean or combination part of both the Precision and Recall, and the F1 will take possibility of the false cases (Both False positives and false negatives). F1 score will be more useful than accuracy, particularly if you have an unequal class distribution of data. Accuracy is always the better when both the false positives and false negatives costs are balances and more similar.

F1 Score can be calculated as $2 * (\text{Precision} * \text{Recall})/(\text{Precision} + \text{Recall})$

So, if you are building a model, this work will help you to find out what the above discussed parameters mean and how well your designed model has achieved.

18.3.4 Performance Optimization

Based on above mentioned performance metrics, the best model is identified and its accuracy is further optimized. N/K-Fold Cross Validation technique is considered to be the one of the best methods which helps us to find the samples from all the possible training dataset.

18.3.4.1 N-Fold Cross Validation

Cross-validation is a best method in which any model is been used on the subset of the data-set and after then, secondly it will evaluate using the complementary subset of the data-set. Understanding any dataset and assessing any of the parameters from the dataset in the prediction function on the same data is specified as an operational error: this model will replicate the labels of the samples which are already seen, but sometimes fail to predict something meaningful on data that which is not been available for accessing or not yet seen. This situation may lead to the overfitting. To prevent this overfitting, it is a general practice to keep a set of the available data as a **test set** while performing a supervised learning model. Remember that the word "experiment" is not intended merely to denote academic use, as machine learning starts out its experiments even in commercial settings.

18.4 Results and Analysis

This section discusses the findings of the experimental study which was addressed in detail in Section 3. The Scikit-Learn library in Python is used for building machine learning models. Numpy, Pandas in Python were used for Data Preprocessing and Analysis and Seaborn was used to draw the visualization of Correlation Heatmap shown in Figure 18.2. All techniques and models used in the paper contain default values of hyper parameters in models.

The observations gained through thorough experiments are shown in Table 18.2.

The results obtained in Table 18.2 can be summarized as:

- Based on performance metric as Accuracy, Logistic Regression shows an accuracy of 70.5%, followed by Support Vector

Table 18.2 Comparison of Performance Metrics on various Machine Learning Models.

Models	**Performance Metrics**			
	F-Score	**Precision**	**Recall**	**Accuracy**
Logistic Regression	0.81	0.72	0.92	0.7054
Support Vector Classifier	0.81	0.68	0.99	0.6849
Naïve Bayes	0.57	0.99	0.40	0.5890
Random Forests	0.81	0.72	0.91	0.6986
Gradient Boosting	0.79	0.72	0.87	0.6788

Classifier with 68.49%, Naïve Bayes with 58.9%, Random Forests with 69.86% and Gradient Boosting with 67.88%. Among all models, Logistic Regression shows the highest accuracy.

- Based on performance metric as Precision, Naïve Bayes performs the best with 0.99. Based on Recall, Support Vector Classifier performs the best with 0.99. Based on F-score, Logistic Regression, Support Vector Classifier and Random Forests, all perform equally well with 0.81.
- Logistic Regression performs best overall amongst all performance metrics with a highest Accuracy and significantly higher Precision, Recall and F-Score, while other models tend to fail on certain performance metrics.

Hence, we deduce that amongst all models taken into consideration, Logistic Regression, performs best on all performance metrics. There are various reasons for such an observation, which are explained below.

- The dependent variable or the target variable is dichotomous, i.e., binary (presence or absence of liver disease) due to the technique called Label Encoding.
- There are no outliers in the data because of the application of Standardization. It reduces bias and variance by standardizing the features in range of [0,1] making it a normalized dataset.
- There is no high correlation (multicollinearity) among the predictors or feature variables. As observed from the

Table 18.3 Accuracy on application of 10-fold cross validation of logistic regression model.

N(N = 10)Fold Cross Validation									
1-Fold	**2-Fold**	**3-Fold**	**4-Fold**	**5-Fold**	**6-Fold**	**7-Fold**	**8-Fold**	**9-Fold**	**10-Fold**
0.6666	0.7045	0.75	0.7272	0.7954	0.75	0.7441	0.6976	0.7674	0.6976

correlation Heatmap, all features have correlation values less than 0.90 making it statistically significant tending to best results in Logistic Regression.

Next, as discussed earlier, we next apply N-Fold Cross Validation technique for optimization purposes, with N = 10, to get best possible accuracy value. The results of the 10-folds are shown in Table 18.3.

Hence, from Table 18.3, we infer that post the application of 10-fold Cross Validation, the maximum accuracy is observed in Fold-5, with accuracy as high as 79.54% which is ~80%, inferring a good result.

18.5 Conclusion

A classification process is used in this research work to find accurate and efficient diagnosis of persons at risk and also the early diagnosis of Liver Diseases. Comparisons of SVM, Random forest, Gradient Booster and Naive Bayesian algorithms were implemented and tested based on the various evaluation metrics such as Fscore and classification accuracy. Based on the results, it is inferred that the Logistic regression achieves better classification efficiency, yields accurate results and is thus considered to be the best classifier compared to the other classifier algorithms listed above. Practically, Logistic classifier classifies the data with precise and effective diagnosis of persons at risk and early identification of Liver Diseases with 71% overall accuracy.

References

1. Anstee, Q.M., Targher, G., Day, C.P., Progression of NAFLD to diabetes mellitus, cardiovascular disease or cirrhosis. *Nat. Rev. Gastroenterol. Hepatol.*, 10, 330–344, 2013.

2. Lin, R.-H., An Intelligent Model for Liver Disease Diagnosis. *Artif. Intell. Med.*, 47, 1, 53–62, 2009.
3. Motwani, M., Dey, D., Berman, D.S., Germano, G., Achenbach, S., Al-Mallah, M.H., Andreini, D., Budoff, M.J., Cademartiri, F., Callister, T.Q., Machine learning for prediction of all-cause mortality in patients with suspected coronary artery disease: A 5-year multicenter prospective registry analysis. *Eur. Heart J.*, 38, 500–507, 2016.
4. Raghupathi, W. and Raghupathi, V., Big data analytics in healthcare: promise and potential. *Health Inf. Sci. Syst.*, 2, 3, 2014.
5. Groves, P., Kayyali, B., Knott, D., Van Kuiken, S., The 'big data' revolution in healthcare. Accelerating value and innovation. McKinsey & Company, New York, 2013.
6. Wu, C.-C., Yeh, W.-C., Hsu, W.-D., Islam, M.M., Nguyen, P.A., Nasrin Poly, T., Wang, Y.-C., Yang, H.-C., Li, Y.-C., Prediction of fatty liver disease using machine learning algorithms. *Comput. Methods Programs Biomed.*, 170, 23–29, 2019.
7. Patro, S.G. and Sahu, K.K., Normalization: A Preprocessing Stage. *IARJSET,*. 2, 3, 20–22, 2005.
8. Huang, S., Cai, N., Pacheco, P.P., Narrandes, S., Wang, Y., Xu, W., Applications of Support Vector Machine (SVM) Learning in Cancer Genomics. *Cancer Genomics Proteomics*, 15, 1, 41–51, 2017.
9. Noble, W.S., What is a support vector machine. *Nat. Biotechnol.*, 24, 12, 1565–1557, 2006. [PubMed].
10. https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47-SVM-figure.
11. Cortes, C. and Vapnik, V., Support-Vector Network. *Mach. Learn.*, 20, 3, 273–297, 1995.
12. Anjali, M., Maria, D., and Shinoj R., An Overview of Deep Learning, *J. Computing Intelligent Systems*, 1, 1, 15–21, 2017.
13. Rish, I., An empirical study of the naive Bayes classifier: IJCAI 2001 workshop on empirical methods in artificial intelligence. *IBM*, 3, 41–46, 2001.
14. Dreiseitl, S. and Ohno-Machado, L., Logistic regression and artificial neural network classification models: a methodology review. *J. Biomed. Inform.*, 35, 352–359, 2002.

Part 4

BIOINFORMATICS AND MARKET ANALYSIS

19

A Novel Approach for Prediction of Stock Market Behavior Using Bioinformatics Techniques

Prakash Kumar Sarangi[1]*, Birendra Kumar Nayak[2] and Sachidananda Dehuri[3]

[1]*Department of Computer Science, NM Institute of Engineering and Technlogy, Bhubaneswar, India*
[2]*Department of Mathematics, Utkal University, Bhubaneswar, India*
[3]*Department of Information and communication Technology, Fakir Mohan University, Balasore, India*

Abstract

A six-phase "A Novel Approach for Prediction of Stock Market Behavior Using Bioinformatics Techniques" is pursued in this work. The six phases are working in a pipeline i. e. the output of one phase is passed as the input to the second phase and so on. In phase one, we have encoded of stock market price behavior to binary string; phase two mapped the binary string to DNA sequence; phase three Sequence alignment occurs using BLAST; phase four Produce the behavior of next Sensex days; phase five decode predicted sequence into binary string and in last phase find error or Mismatching analysis of stock market behavior. We have carried out an extensive experimental way on Bombay stock market (BSE) Sensex data for validity out purposed frame work. Finally we conclude the result extracted from experimental study between predicted stock market price behavior with actual behavior. Some better inference has been done and using other bioinformatics models same theme extract is to be explored more.

Keywords: Local alignment, nucleotides, binary strings, DNA sequences

**Corresponding author*: prakashsarangi89@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (461–484) © 2021 Scrivener Publishing LLC

19.1 Introduction

The Stock market Behavior study relies on how the examples of file esteem or stock costs are demonstrating verifiably. The crucial plan to fruitful securities exchange conduct expectation in accomplishing best outcomes limit the off base gauge the stock cost through previous history study. Forecasting stock records conduct is exceptionally troublesome in view of the irregularity in financial exchange that needs a precise estimate model. The securities exchange files are exceptionally fluctuating that is fall the stock cost or raising the stock cost. Changes are influencing the financial specialist's conviction. Deciding progressively successful methods of financial exchange record conduct expectation is significant for securities exchange speculators so as to settle on increasingly educated and precise venture choices. There are a few models and methods, for example, Neural Network, Data Mining, Markov Model, and Neuro-Fuzzy framework that offer valuable apparatuses for determining uproarious conditions like financial exchange conduct study. This examination intends to give smart methods like bioinformatics strategies to foresee day by day financial exchange lists conduct and securities exchange costs conduct. A securities exchange file speaks to personal conduct standards of a few individual stocks.

Bioinformatics can be used to solve very complex biological problems. It also used for sequence mining from big biological datasets. It has lots of advantages like genomic sequence analysis, DNA micro array analysis, gene prediction, protein structure prediction, and phylogenetic analysis. In the last two decades, application of bioinformatics plays a vital role in different field of medical science, computer science, and other areas.

In this work, our goal in securities exchange conduct is how we can anticipate the conduct of up and coming stock files through bioinformatics procedures. Here conduct forecast happens in the type of DNA arrangement mining. Succession mining is the crudest activity in computational science. This activity comprises of discovering which some portion of the organic arrangements are similar and which part varies during clinical investigation and genome mapping forms. The sequencing emulating infers oppressing a DNA or peptide arrangement to succession arrangement, arrangement databases, rehashed arrangement look, or different bioinformatics strategies on a PC. For an extraction of comparative examples, arrangement is expected from other comparable qualities. In such cases, BLAST is a significant device that encourages us to discover exceptionally comparative examples that exist in quality succession.

Finally daily stock market behavior prediction can done using some algorithms like Naïve pattern matching algorithm and hamming distance algorithm. Prediction result is derived for BSE daily stock closing price. We have expressed our efforts of daily data year wise and also prediction results are satisfactory in yearly basis. The details of the above are described in next sections. In Section 19.2 we discuss about several models for the prediction of stock market behavior. Our real description of work is clearly explained in Section 19.3 in the form of algorithm steps or examples. The results of prediction of stock market behavior are discussed in Section 19.4. Finally, the conclusion is drawn in Section 19.5.

19.2 Literature Review

Predicting stock market behavior with customary time arrangement investigation has demonstrated to be troublesome where a counterfeit neural system might be appropriate for the undertaking. The neural system can extricate valuable data from a huge arrangement of information. They predicted the stock rate since it is a difficult and overwhelming errand to discover which is a progressively successful and precise strategy with the goal that a buy or sell sign can be produced for given stocks [1]. Notwithstanding neural system models learning methods can foresee nonlinear regularities in resource value developments [2]. Employments of the Neural system in view of their abilities in taking care of nonlinear relationships and furthermore execute another fluffy time arrangement model to improve estimates. The fuzzy relationship is utilized to conjecture the Taiwan stock data [3].

In the neural system, fuzzy time arrangement model whereas in test perceptions is utilized for preparing and out example perceptions are utilized for estimating. The disadvantage of taking all the levels of enrollment for preparing and anticipating is that it may influence the presentation of the neural system. To keep away from this take the contrast between perceptions. These decrease the scope of the universe of talk [4] for ARIMA model and vector ARMA model with fuzzy time arrangement strategy for forecasting. Fuzzy time arrangement technique particularly heuristic model performs better gauging capacity in the momentary period forecast. The ARIMA model makes little determining blunders in a longer analysis timeframe. In this paper, the creator examines whether the length of the span will impact the gauging capacity of the models or not [5].

Artificial neuron-fuzzy are surmising framework for taking choices dependent on the estimations of specialized markers. The framework

utilized weighted moving midpoints, disparity, and RSI. RSI assists with flagging overbought and oversold conditions in security. The errand of the preparation for ANFIS was changing all the modifiable boundaries to make the ANFIS yield coordinate the preparation information. A half and half learning calculation contains both least-square strategy and back engendering and furthermore used to recognize the ideal qualities for the boundaries. The creator dole out loads or rate for connoting the portion of absolute speculation that ought to be a limit hazard and augment benefit in securities exchange [6]. The Efficient Market Hypothesis was utilized to improve the expectation in transient securities exchange patterns. This implies gains from the right forecast and misfortunes from wrong expectations. The neuro-fuzzy framework obviously shows the potential for money related market expectations [7]. Investigation Stock expectations dependent on the neuro-fuzzy framework, the procedures were utilized to forecast the pattern of the stock costs and results determined. Elliott's wave hypothesis can't continually clarify the market impeccably yet the fuzzy evaluations of the market conduct precisely to improve the securities exchange estimating [8].

AI is used for information mining to assess the prescient relationship of various money related and financial factors. In this work they have utilized the Neural system model used for estimation and arrangement which are then inspected for their capacity to give a powerful figure of future qualities. A cross-approval procedure was utilized to improve the speculation capacity of a few models. The exchanging systems guided by characterization models create higher hazard balanced benefits than the purchase-and-hold procedure just as guided by the level-estimation based gauge of the neural system and relapse models. The creator chooses to send the conjecture the stock profits, exchange expenses, and an individual-charge section to duplicate the sensible speculation rehearses [9]. The information mining strategies can reveal the concealed example; foresee future patterns and practices in money related market. For example, coordinating procedures is seen as an illustrative in time arrangement examination. In this paper, the creator applied insect calculation to oblige an adaptable and dynamic example coordinating undertaking in time series calculation. Aside from portion size, the subterranean insect to sub-time-arrangement size influences the framework execution. In this paper, the proportion was set to 1 and furthermore the proportion decreased to get a superior outcome [10].

Predicted Markov Models (HMM) way is dealing and with determining the stock cost for interrelated markets. Gee was utilized for design acknowledgment and characterization issues in light of its

demonstrated appropriateness for displaying dynamic framework. The creator summed up the benefit of the HMM which was a solid factual establishment. It's ready to deal with new information vigorously and is computationally productive to create and assess comparable examples. The creator chooses to create a crossbreed framework utilizing AI ideal models with HMM to improve the exactness and proficiency of gauge the financial exchange [11]. A combination model of HMM, ANN and GA was used to estimate money-related market conduct. This instrument was utilized inside and out an investigation of the securities exchange. The everyday stock cost was changed to free arrangements of qualities that become a contribution to HMM and GA to advance the underlying boundaries of HMM. At that point HMM was used to recognize and find comparable examples. A weighted normal of the value contrast of comparable examples are obtained to set up a gauge for the requirement for the following day. To streamline the execution, picking the number of states as the number qualities in the perception vectors may not be appropriate for some example. The creator chooses to utilize another successful GA to locate the best HMM engineering for the given informational index [12].

Hybridization of Markov chain ideas into a fuzzy stochastic forecast of stock files is used to accomplish better precision and certainty. In this paper, the examination of the ANN and Markov model has resulted in significant focal points. It creates a high precise outcome requiring just one contribution of information. The main hour's stock file information was utilized as the info and it leads the expectation of the likely record at some random hour. This methodology did not require the standard deviation of the forecast. This methodology not just gave improved benefit execution but likewise was used to decide and stop 'misfortunes' with more prominent certainty [13]. A region of Neural Network, Data Mining, Hidden Markov Model, and Neuro-Fuzzy framework changes expectation for the securities exchange. Neural Networks and Neuro-Fuzzy frameworks are recognized to be the main AI strategies in financial exchange list expectation territory. Traditional procedures do not spread all the conceivable connection of the stock value changes. There are new ways to deal with known inside and out of an investigation of stock value varieties. NN and Markov Model can be utilized only in the fund markets and determining of stock cost. Markov Model can be utilized solely in the money markets and anticipating stock cost. They propose an estimating strategy to give better precision rather than customary strategies. In future work, the neural system and Markov model can likewise investigate different applications [14].

19.3 Proposed Work

Our approach is passing through six phases like (I) encoding of stock market price behavior to binary string, (II) Mapping binary string to DNA sequence, (III) Sequence alignment using BLAST, (IV) Production, (V) Decoding of predicted sequence, and (VI) Mismatching analysis of stock market behavior. Figure 19.1 illustrates the overall framework of our approach.

19.3.1 Encoding of Stock Market Price Behavior to Binary String

The stock market closing prices of each day are categorized as '0' and '1'; '0' means that the next day index is lower or same to the index of current day (i.e., the non-increasing behavior) and '1' means the next day index is higher than the index of current day (i.e. the increasing behavior). An algorithm 1 converts the stock market price behavior to binary string.

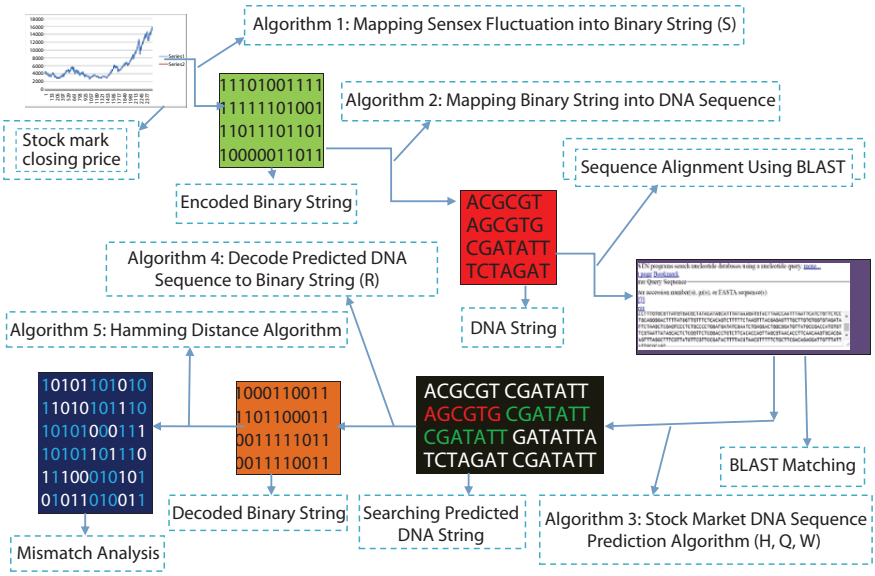


Figure 19.1 Framework of the book chapter.

Algorithm 1: Mapping Sensex Fluctuation into Binary String (S)

```
Input: Stock market closing price index-S
Output: Bits of string-B
N = Length(S);
FOR  i = 1  to  N-1
        IF  (S[i]<S(i+1))
                B[i] = 1;
        ELSE
                B[i] = 0;
END FOR
Return B;
```

19.3.2 Mapping Binary String Into DNA Sequence

Here we are converting stock market price behavior or stock market indices behavior in form of binary string to DNA sequence by taking two bits of binary numbers into a single character of nucleotides (A, T, C, G), where A = (0,0), T = (1,1), C = (0,1), and G = (1,0) according to their pairing like 'A' pair with 'T' and 'C' pair with 'G'. For this representation we are developing a small algorithm which can convert entire binary sequence to DNA sequence.

Algorithm 2: Mapping Binary String into DNA Sequence (B)

```
Input: Bits of String-B
Output: Nucleotides of Strings-P
M= Length(B);
FOR j = 1 to M-1
        IF  (B[j]B[j+1] = 00)
                P[j] = A;
        ELSE IF (B[j]B[j+1] = 11)
                P[j] = T;
        ELSE IF (B[j]B[j+1] = 10)
                P[j] = G;
        ELSE (B[j]B[j+1] = 01)
                P[j] = C;
END  FOR
Return P;
```

19.3.3 Sequence Alignment Using BLAST

BLAST (Basic Local Alignment Search Tool) is a family of sequence alignment algorithms developed by Altschul *et al.* in 1997 [15]. These programs

are utilized for succession comparability ID. They recognize districts of neighborhood arrangement to help with distinguishing connections among successions, which permits the client to distinguish similitude's between the question nucleotide or protein grouping with groupings in open databases and furthermore distinguishes bunches of close by or locally thick "comparative" k-tuples (number of series of letters), Additionally, it used to recognize whether a given grouping is novel, homologous to a known grouping, or if the arrangement contains themes which may give pieces of information to a potential job of the arrangement being questioned.

The favored inquiry succession organization of the BLAST program is the FASTA group which accepts contribution as DNA arrangements. Presently each piece of example is lined up with the biggest DNA database of BLAST. It is discovered that 5 years of stock example behavior plays out a match with the DNA information base of BLAST by giving decisions exceptionally comparable examples.

19.3.4 Prediction Method

Discovering all events of an example in a pattern is difficult that emerges oftentimes in content editing programs. Normally, the content is a record that is altered, and the example looked for is a specific word provided by the client. Proficient calculations for this issue can extraordinarily help the responsiveness of the content editing program. String-coordinating calculations are additionally utilized, for instance, to look for specific examples in DNA groupings. For the definition of the string coordinating issue as we accept that the content in a variety of length K and that the example is a variety of length L. The string-coordinating issue is the issue of discovering every single substantial move with which a given example happens in a given book via looking through character by character. We are developing a modified searching algorithm like Naïve string matching algorithm to find stock market as a pattern sequence for a particular month with the text pattern human genomic sequence. Our aim is to predict daily stock market behavior for a particular year. An algorithm 3 predicts the output of phase 2 into predicted DNA sequence of future trading days.

Algorithm 3: Stock Market DNA Sequence Prediction Algorithm (H, Q, W)

Input: Human genome-H
Nucleotides of one month stock market index behavior-Q
Number of trading days in next month-W
Output: Predicted nucleotides of next month stock market index behavior-R

```
K=Length(H);
L=Length(Q);
O=W/2;
FOR u=0  to K-L
        IF(Q[1,2,…L]=H[u+1,u+2,…u+L)
                Then matching  occurs  with  shift  u;
                FOR  v=L+1 to  L+O
                     R[v]=H[u+v];
                END FOR
        END IF
    END FOR
   Return R;
```

19.3.5 Decoding Predicted DNA Sequence Into Binary String

The fluctuation in the stock market is encoded into binary string described by phase-I. Similar manner after getting predicted DNA we have to decode into find the behavior of stock market in reverse way. In this phase we have developed an algorithm 4, which converts the output predicted DNA sequence of phase-3 into predicted binary string.

Algorithm 4: Decode Predicted DNA Sequence to Binary String (R)

Input: Predicted nucleotides of next month stock market index behavior-R

Output: Predicted binary string of next month stock market index behavior-Z

```
V= Length(R)
FOR k=1  to  V
       l = 2k-1;
         IF(R[k]==A)
                    Z[l]Z[l+1]=00;
         ELSE  IF (R[k]==T)
                    Z[l]Z[l+1]=11;
           ELSE  IF (R[k]==G)
                    Z[l]Z[l+1]=10;
              ELSE
                         Z[l]Z[l+1]=01;
      END FOR
       Return Z;
```

19.3.6 Mismatching Analysis of Stock Market Behavior

The mismatch analysis between predicted binary string and actual Sensex data can be measured using hamming distance algorithm approach. The Hamming distance is named after Richard Hamming, who presented it in his basic paper on Hamming codes mistake distinguishing and blunder rectifying codes in 1950. The Hamming separation between two strings of equivalent length is the number of positions at which the comparing images are extraordinary. In this way, it quantifies the base number of replacements required to transform one string into the other or the number of mistakes that changed one string into the other. We have improved the examination calculation hamming separation since it finds the request for the letters, yet in addition, thinks about the substance of the words in any case the request and size of the two strings. A calculation 5 discovers bungle happens between financial exchange real information with anticipated information.

Algorithm 5: Hamming Distance Algorithm (X, Z)

```
Input: Bits of actual next month stock market index behavior-X
       Bits of predicted next month stock market index behavior-Z
Output: Hamming distance-D
C= length(X) or Length(Z);
D=0;
FOR w=1 to C
        IF (X(w)!=Z(w))
        D=D+1;
END FOR
RETURN  D;
```

19.4 Experimental Study

The entire data set contains fifteen years daily close price of BSE, which covers the period from the first trading day of January, 2001 to last trading day of December, 2010. The data sets are divided into yearly and prediction results derived for daily trading days of a particular month.

19.4.1 Data Analysis

Bombay Stock Exchange (BSE) data is collected from Yahoo Finance. Entire data is divided into five years each. Encoding takes place by using

above algorithm to get a nucleotide sequence of stock market. Each character of the sequence contains two-day behavior. Align the sequence and put into the BLAST to get highly similar pattern from gene. In our work we have taken ten years daily closing price into three parts. Each part contains sequentially five years trading day's behavior. After local alignment we found that stock behavior highly similar with human gene. Find all human genes from gene data base and match month wise patterns of stock market behavior. Using above algorithms we have able to predict the behavior of daily stock market in year wise. The prediction results are derived by using MATLAB tools R 2013b are as follows.

19.4.2 Results

The experimental results between actual stock market behaviors with predicted behavior of each month year wise explain figures below. Figure 19.2 describes the BSE Sensex behavior for the month of February in the year 2001. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.3 describes the BSE Sensex behavior for the month of February in the year 2004. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.4 describes the

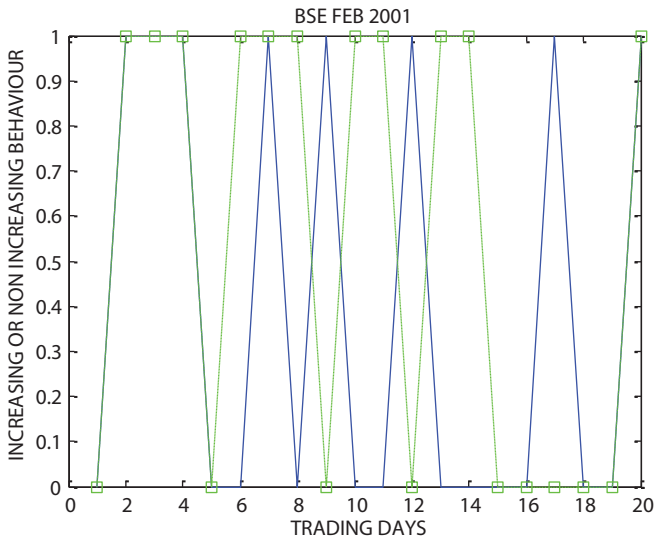


Figure 19.2 Actual vs predicted behavior BSE February, 2001.

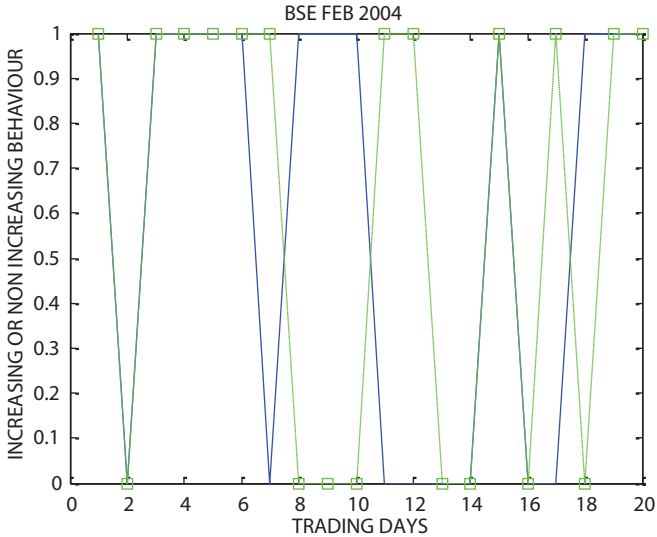


Figure 19.3 Actual vs predicted behavior BSE February, 2004.

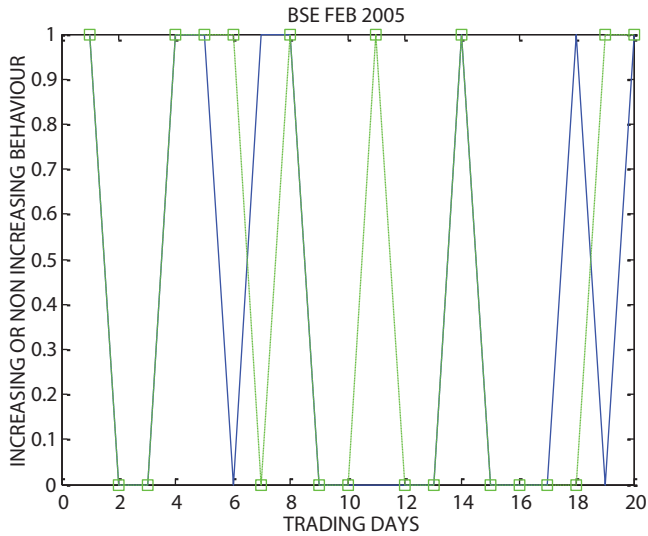


Figure 19.4 Actual vs predicted behavior BSE February, 2005.

BSE Sensex behavior for the month of February in the year 2005. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.5 describes the BSE Sensex behavior for the month of February in the year 2008. Here X-axis represents number of trading days and Y-axis

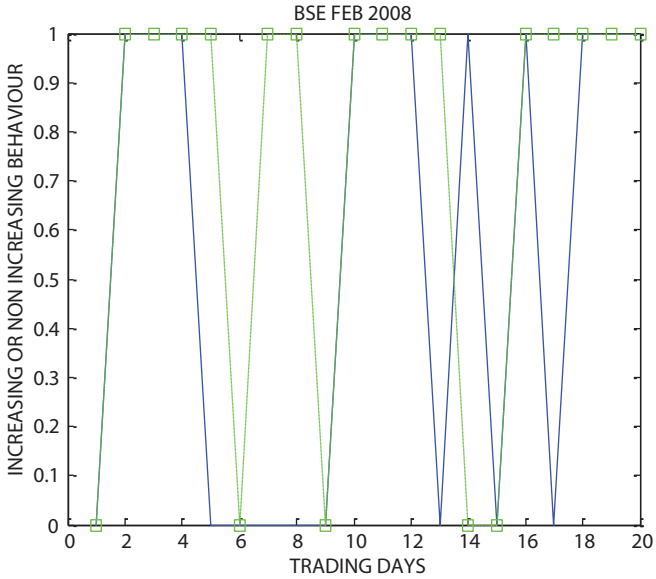


Figure 19.5 Actual vs predicted behavior BSE February, 2008.

represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.6 describes the BSE Sensex behavior for the month of February in the year 2010. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string.

Figure 19.7 describes the BSE Sensex behavior for the month of March in the year 2001. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.8 describes the BSE Sensex behavior for the month of March in the year 2002. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.9 describes the BSE Sensex behavior for the month of March in the year 2003. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.10 describes the BSE Sensex behavior for the month of March in the year 2006. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.11 describes the BSE Sensex behavior for the month of March in the year 2008. Here X-axis represents number of trading days and Y-axis represents increasing

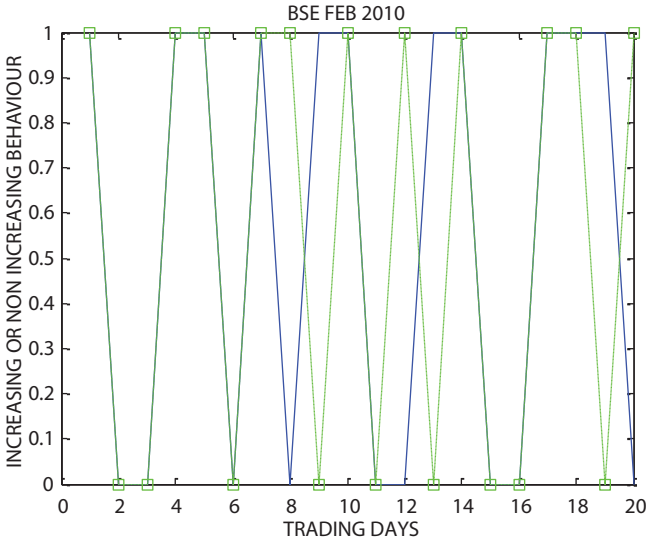


Figure 19.6 Actual vs predicted behavior BSE February, 2010.

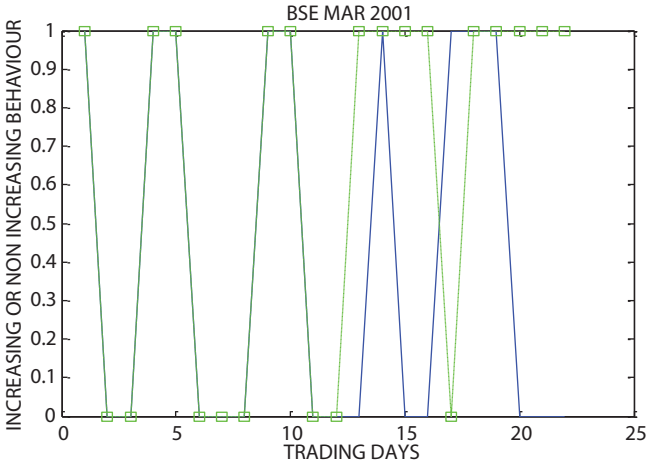


Figure 19.7 Actual vs predicted behavior BSE March, 2001.

and non-increasing behavior between actual vs predicted binary string. Figure 19.12 describes the BSE Sensex behavior for the month of March in the year 2010. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string.

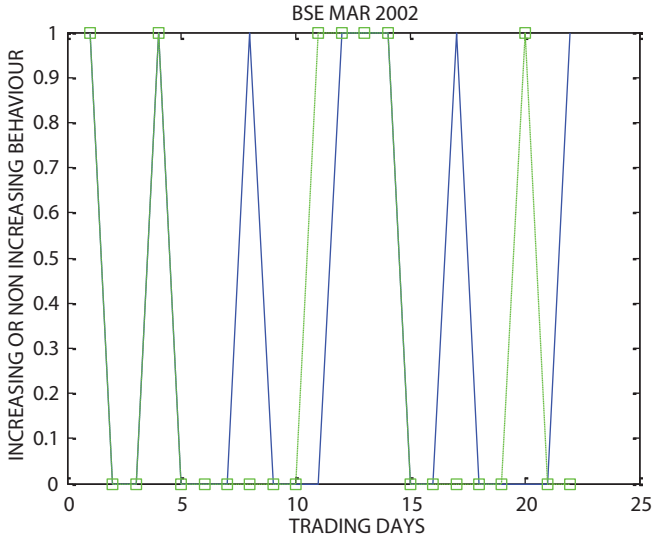


Figure 19.8 Actual vs predicted behavior BSE March, 2002.

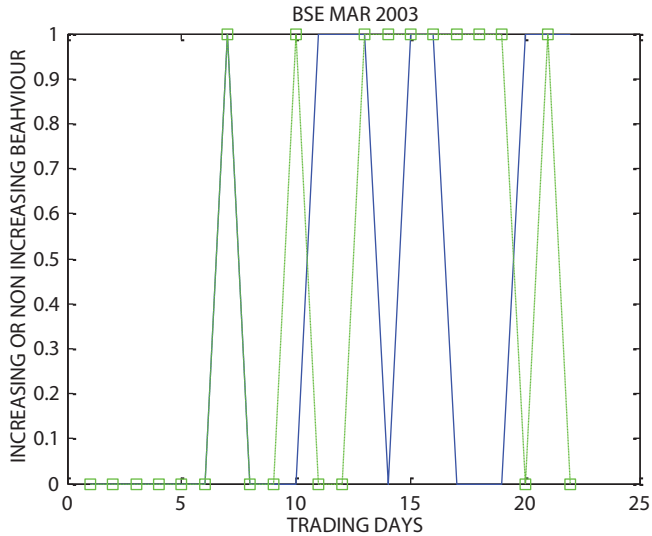


Figure 19.9 Actual vs predicted behavior BSE March, 2003.

Figure 19.13 describes the BSE Sensex behavior for the month of April in the year 2001. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.14 describes the BSE Sensex behavior for

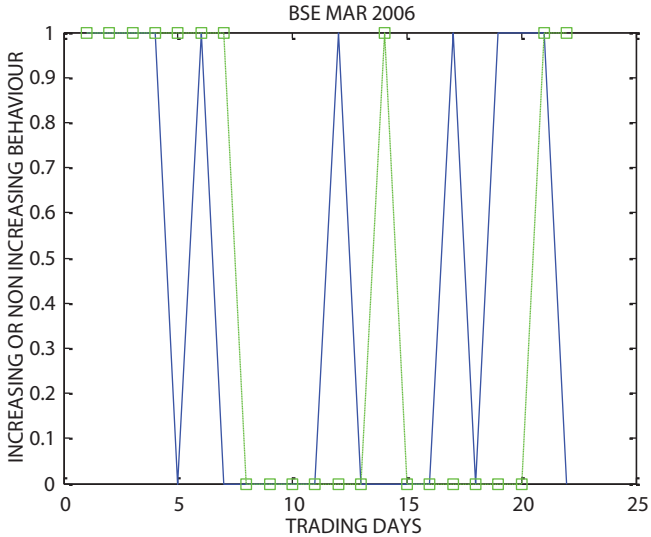


Figure 19.10 Actual vs predicted behavior BSE March, 2006.

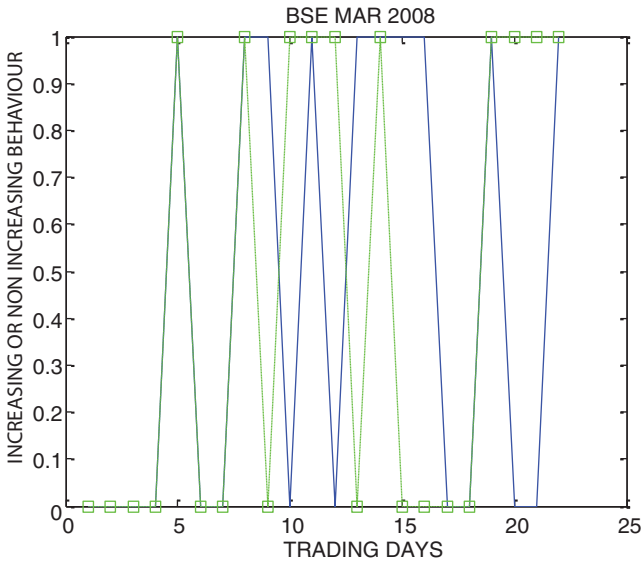


Figure 19.11 Actual vs predicted behavior BSE March, 2008.

the month of April in the year 2003. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.15 describes the BSE Sensex behavior for the month of April in the year 2004. Here X-axis

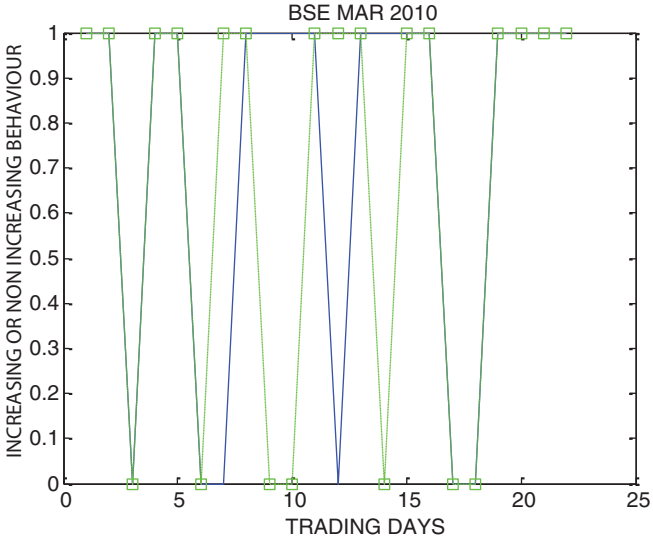


Figure 19.12 Actual vs predicted behavior BSE March, 2010.

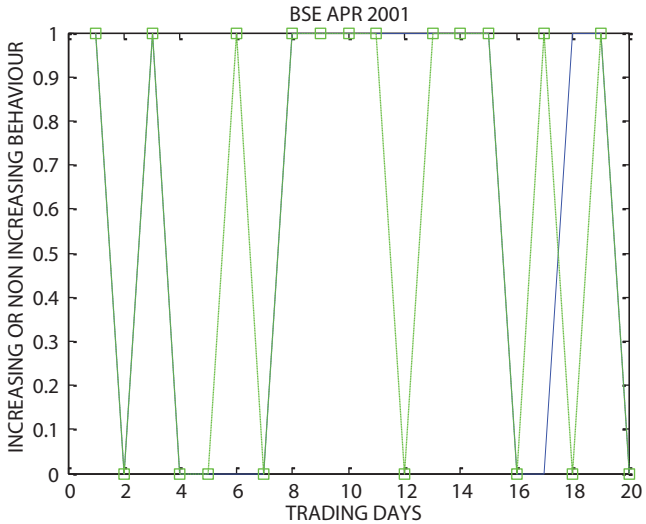


Figure 19.13 Actual vs predicted behavior BSE April, 2001.

represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.16 describes the BSE Sensex behavior for the month of April in the year 2007. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary

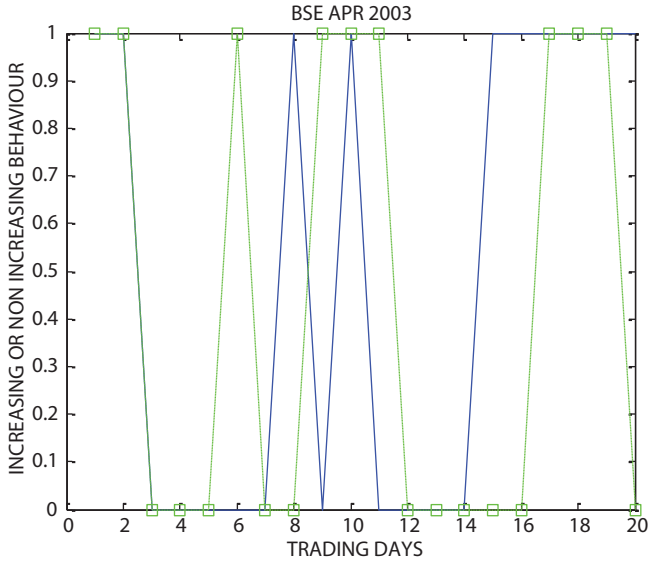


Figure 19.14 Actual vs predicted behavior BSE April, 2003.

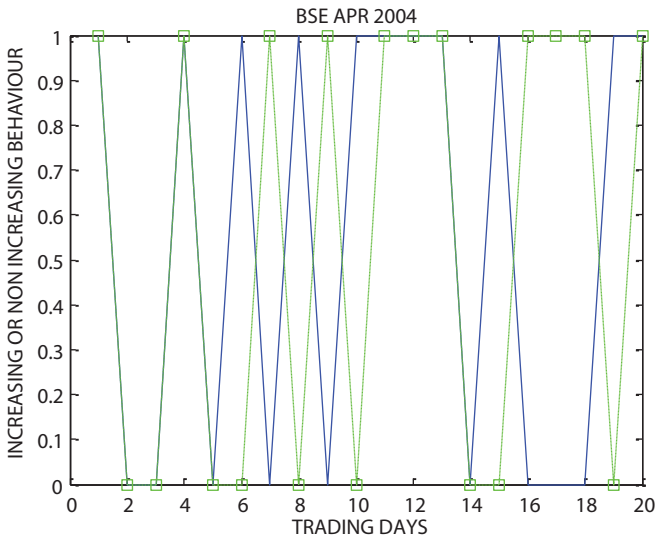


Figure 19.15 Actual vs predicted behavior BSE April, 2004.

string. Figure 19.17 describes the BSE Sensex behavior for the month of April in the year 2008. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string.

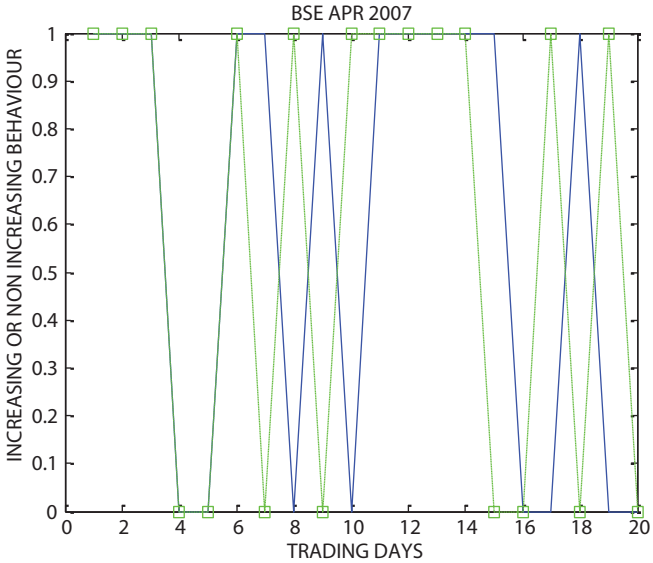


Figure 19.16 Actual vs predicted behavior BSE April, 2007.

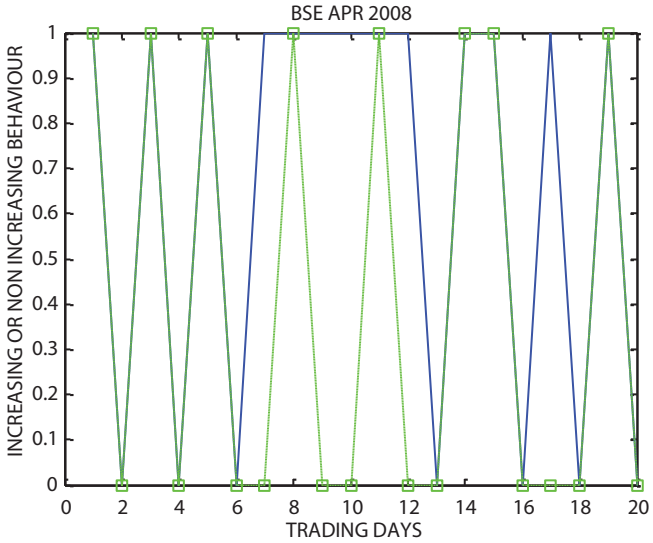


Figure 19.17 Actual vs predicted behavior BSE April, 2008.

Figure 19.18 describes the BSE Sensex behavior for the month of May in the year 2001. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.19 describes the BSE Sensex behavior for

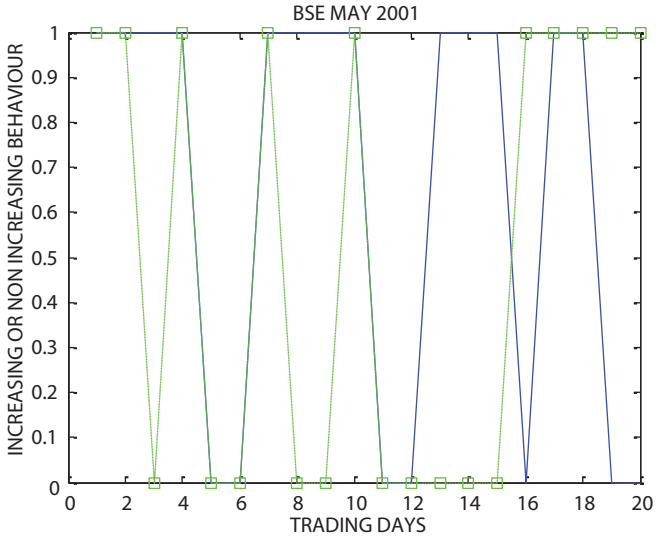


Figure 19.18 Actual vs predicted behavior BSE May, 2001.

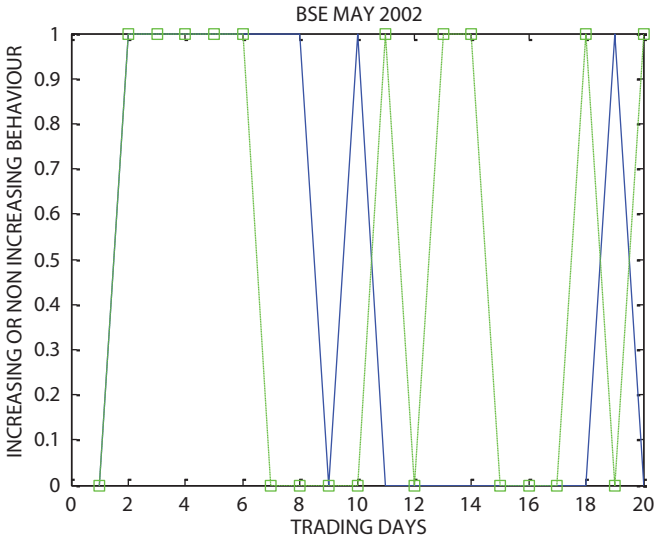


Figure 19.19 Actual vs predicted behavior BSE May, 2002.

the month of May in the year 2002. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string.

Figure 19.20 describes the BSE Sensex behavior for the month of October in the year 2001. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string. Figure 19.21 describes the BSE Sensex behavior for the month of October in the year 2003. Here X-axis represents number of trading days and Y-axis represents increasing and non-increasing behavior between actual vs predicted binary string and so on.

From above experimental results, if we are considering prediction for first six trading days then the comparison results have been displayed in below table. Table 19.1 represents each month's first six day prediction result comparison between actual stock market behaviors with predicted behavior. First and forth columns denote months, second and fifth column denotes different year from 2001 to 2010 and columns three and six calculated results number of mismatch finding using Hamming distance algorithm.

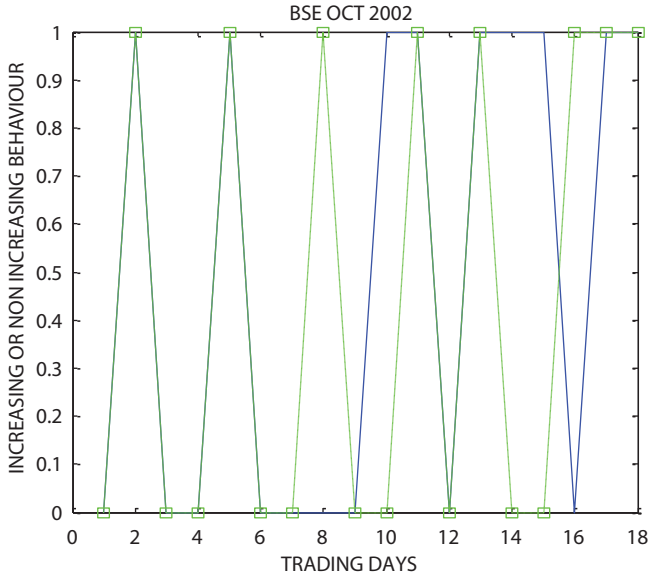


Figure 19.20 Actual vs predicted behavior BSE October, 2002.

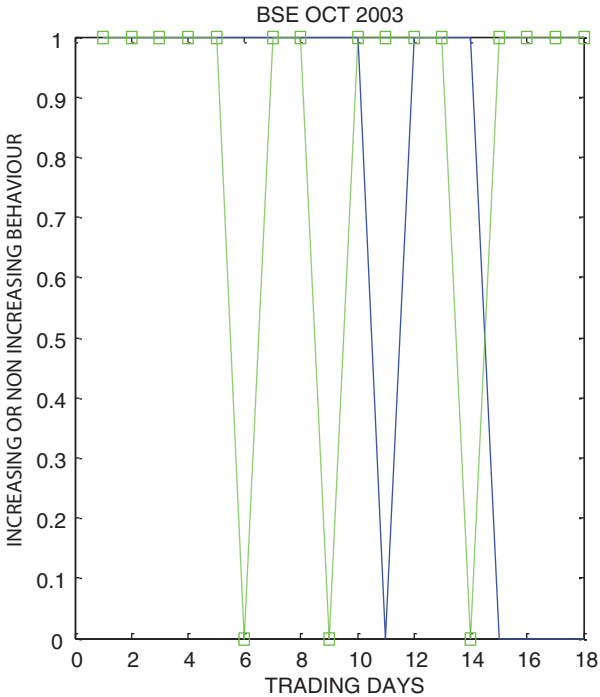


Figure 19.21 Actual vs predicted behavior BSE October, 2003.

19.5 Conclusion and Future Work

In practice, behavior of stock markets varies through different markets and different countries. In Indian stock market the rising and falling depends upon various factors. In this work we focus on BSE Sensex data and predict their behavior in different approaches. Our model consists of six phases and all the phases pass sequentially one by one. Our objective is to apply bioinformatics techniques which are applied to stock market prediction process. We are converting Sensex behavior into DNA string and prediction done again compare to the actual behavior. Finally our findings are clearly explained in experimental results. From this work we conclude that on an average 90% prediction can made for the year using this technique. Another aspect to note is considering a six-day prediction where the number of mismatch calculated is very minimal. For future research we are trying to interlude hybridization and gene production techniques into stock market behavior study.

Table 19.1 Hamming distance result year wise.

Month	Year	Hamming Distance	Month	Year	Hamming Distance
February	2001	1	August	2001	0
	2004	0		2002	0
	2005	1		2003	0
March	2001	0		2010	0
	2002	0	September	2005	0
	2003	0		2008	1
	2006	1	October	2002	0
	2008	0		2006	1
	2009	1		2007	0
	2010	0		2008	0
April5	2007	0		2009	0
	2008	0	November	2001	0
May	2001	1		2002	1
	2002	0		2003	0
June	2002	1		2004	0
	2003	1		2005	0
	2004	0		2006	0
July	2001	0		2007	0
	2002	1		2008	0
	2004	0		2009	0
	2005	1		2010	0
	2006	1	December	2004	0
	2009	1		2005	0
	2010	0		2010	0

References

1. Dase, R.K. and Pawar, D.D., Application of Artificial Neural Network for stock market predictions: A review of literature. *Int. J. Mach. Intell.*, 2, 2, 14–17, 2010.
2. Halbert, W., *Economic prediction using neural networks: The case of IBM daily stock returns*, Department of Economics University of California, San Diego. 1988.
3. Yao, J. and Tan, C.L., Guidelines for Financial Prediction with Artificial neural networks. *Proc. International Conference on Neural Information Processing,* Shanghai, China, pp. 757–761, 2001.
4. Hui-Kuang Yu, T. and Huarng, K.-H., A Neural network-based fuzzy time series model to improve forecasting. *Expert Syst. Appl.,* 37, 4, 3366–3372, 2010.
5. Wong, H.-L., Tu, Y.-H., Wang, C.-C., Application of fuzzy time series models for forecasting the amount of Taiwan export. *Expert Syst. Appl.*, 37, 2, 1456–1470, 2010.
6. Agrawal, S., Jindal, M., Pillai, G.N., Preduction using Adaptive NeuroFuzzy Inference System (ANFIS), in: *Proceeding of the International Multiconference of Engineers and Computer Scientists*, vol. I, 2010.
7. Atsalakis, G.S. and Valavanis, K.P., Forecasting stock market short-term trends using a neuro-fuzzy based methodology. *Expert Syst. Appl.*, 36, 7, 10696–10707, 2009.
8. Atsalakis, G.S., Dimitrakakis, E.M., Zopounidis, C.D., Elliot Wave Theory and neurofuzzy systems, in stock market predictions: The WASP system. *Expert Syst. Appl.*, 38, 8, 9196–9206, 2011.
9. Enke, D. and Thawornwong, S., The use of data mining and neural networks for forecasting stock market returns. *Expert Syst. Appl.*, 29, 4, 927–940, 2005.
10. Suresh Babu, M., Geethanjali, N., Sathyanarayana, B., Forecasting of Indian Stock Market Index Using Data Mining & Artificial Neural Nework. *Int. J. Adv. Eng. Appl.*, 2011.
11. Hassan, Md. R. and Nath, B., Stock Market forecasting using Hidden Markov Model: A New Approach. *Proceeding of the 2005 5th International Conference on Intelligent Systems Design and Application*, 0-7695-2286-06/05, IEEE, 2005.
12. Hassan, Md. R., Nath, B., Kirley, M., A fusion model of HMM, ANN and GA for stock market forecasting. *Expert Syst. Appl.*, 33, 1, 171–180, 2007.
13. Wang, Y.-F., Cheng, S., Hsu, M.-H., Incorporating the Markov chain concepts into fuzzy stochastic prediction of stock indexes. *Appl. Soft Comput.*, 10, 2, 613–617, 2010.
14. Preethi, G. and Santhi, B., Stock market forecasting techniques: A survey. *J. Theor. Appl. Inf. Technol.*, 46, 1, 24–30, 2012.

20

Stock Market Price Behavior Prediction Using Markov Models: A Bioinformatics Approach

Prakash Kumar Sarangi[1]*, Birendra Kumar Nayak[2] and Sachidananda Dehuri[3]

[1]Department of Computer Science, NM Institute of Engineering and Technology, Bhubaneswar, India,
[2]Department of Mathematics, Utkal University, Bhubaneswar, India
[3]Department of Information and Communication Technology, Fakir Mohan University, Balasore, India

Abstract

Financial stock market price behavior prediction is very important problem which can be expressed briefly using computational biology or bioinformatics techniques. Markov Models have newly used in stock market price behavior prediction. In this work, we have applied the zero order, first order, second order, and Hidden Markov Model (HMM) techniques for forecasting the daily stock market price behavior (i.e. increasing and non-increasing) in form of 1's and 0's for given closing price of day to day financial data. The fluctuations are characterized by a number 1and 0, where '0' denoting non-increasing behavior (i.e. Sensex either steady or falling down) and '1' denoting increasing behavior (i.e. Sensex rising day to day) in the financial Stock market. The behavior of financial market 0 and 1 are changes to nucleotides A, T, C, and G. Now behavior of financial market converted in to biological DNA sequence and we have used the zero order, first order, second order, and Hidden Markov Model (HMM) procedure to predict future behavior of daily stock market Sensex. In this work, our motivations is prediction of stock market behavior through bioinformatics tools like HMM and get successfully closest behavior results from the historical training Sensex data to the test sequences. Finally we have drawn a comparison analysis between all four models. In future other higher order Markov Models have explored.

Keywords: Binary string, nucleotides, markov model, finite state machine, HMM

**Corresponding author*: prakashsarangi89@gmail.com

Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty and Xiaobo Zhang (eds.) Data Analytics in Bioinformatics: A Machine Learning Perspective, (485–506) © 2021 Scrivener Publishing LLC

20.1 Introduction

The finance related securities exchange is a significant parameter for the monetary development of a nation. All major monetary dealings in a country depend on a system and which changes progressively. The dynamic paces of progress esteems are called securities exchange cost or financial exchange record. Anticipating this securities exchange value conduct offers immense exchange benefit openings which are a gigantic inspiration for inquiring about around there. Information on a financial exchange value conduct in advance by even a small amount of a second can influence an exchange opening for benefit. Essentially, a probabilistically precise expectation can be the massive business analysis. This allure of an answer has empowered scientists, in both industry and the world business to discover a route past issues like unpredictability, irregularity, and reliance on schedule, economy, and different market. Already procedures of AI and soft computing techniques have been utilized to take care of this issue.

There are so many types of Markov models used to depict datasets of various complexities. In each kind of Markov model, diverse some mathematical probabilistic formulas are determined. A zero-order Markov model portrays the probability of the current state independent of the previous state. Here we have to calculate individual probability of each symbol. A first-order Markov model portrays the probability of the current state being determined by the previous state. This compares to the exceptional frequencies of two connected deposits happening at the same time.

Also, a second-order Markov model depicts the circumstance wherein the likelihood of the probability of the current state is determined by the previous two states. This relates to the special trimmer frequencies in DNA sequences. For instance, in a protein-coding succession, the recurrence of special trimmers ought to be not quite the same as that in a non-coding or irregular arrangement. This disparity can be depicted second-order Markov model. Also, significantly third-order Markov models or higher-order Markov Models are accessible for DNA sequence analysis.

The Hidden Markov Model is a probabilistic limited state machine that depends on a lot of in secret basic states and utilized in advances as for an arranged measurement. Each state is related with perceptions. The securities exchange value conduct can likewise be found in a comparative methodology. The hidden states, which are controlled by the conduct of the securities exchange cost, are generally undetectable to the investor. The gathering changes between these basic states dependent on party approach, choices and monetary conditions, and so on. The obvious impact which

mirrors these is the estimation of the stock. Obviously, the HMM adjusts well to this genuine situation.

The rest of this chapter is containing as like. In Section 20.2 we have reviewed for stock market price behavior predictions using AI, and some Markov model techniques, especially in Section 20.3, we have elaborated details of our approach using the zero order, the first order, the second order, and the Hidden Markov Model (HMM) with algorithm steps. In Section 20.4, we have described the experimental study, data set collection, and result analysis with comparison using different Markov Models. At last in Section 20.5, conclude our observation and interpret future work to be explored.

20.2 Literature Survey

Securities exchange value conduct expectation has been one of the unprecedented research zones previously, given the recognizable enthusiasm of a great deal of significant organizations. In this examination, some AI strategies have likewise been applied to questionable degrees of progress. Be that as it may, securities exchange estimating is still carefully constrained because of its non-fixed, occasional, and when all is said in done erratic nature. Foreseeing conjectures from simply the past financial exchange information is a still an increasingly troublesome undertaking since it overlooks a few remote elements, (for example, the condition of the organization, proprietorship, and so forth).

The AI strategies which can be comprehensively used to the behavior of stock market price techniques like Machine learning approaches such as ANNs, FL, and SVM [2,3,7]. As they conclude, from above machine learning techniques ANNs performance is better than other models [4,8].

From time series data analysis Markov models are used for DNA sequencing, Speech reorganizations, etc. [6,9], used HMMs to predict airline data detection, stock price, etc. Starting late Hassan joined HMMs and feathery basis rules to improve the desire precision on non-fixed budgetary trade costs, [4]. Our methodology is like Ref. [5]. They have likewise demonstrated the day by day opening, closing, high and low records as consistent perceptions from basic shrouded states. The principle distinction between our methodology and theirs lies in the highlights utilized (we have utilized fragmentary changes in the above amounts) and in the way of anticipating. While they have searched for comparable information designs in the history information, we have augmented the probability

of a succession of perceptions over all conceivable determined securities exchange value conduct.

20.3 Proposed Work

In this paper our motivation is to predict stock market behavior through computational biology. Now computational biology or bioinformatics algorithms can be used in several fields of research. The Sensex fluctuation moves to binary sequences characterizing the stock market behavior. The behaviors of stock market are defining a binary sequence in form of increasing in Sensex represented by 1 and non-increasing represented by 0. Such sequences are converting in to nucleotides sequences A, T, C, G, where A = (0, 0), T = (1, 1), C = (0, 1) and G = (1, 0). Now our objective is to predict stock market behavior form this nucleotide by using the zero order, first order, second order, and Hidden Markov Model (HMM). Finally decode the predicted nucleotides sequence behavior into binary and compare actual vs predicted behavior to find their mismatch analysis using Hamming Distance Algorithm. The details of framework are described in Figure 20.1.

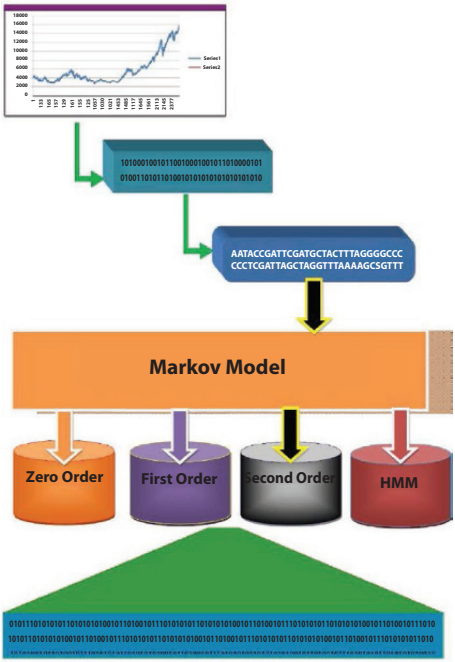


Figure 20.1 Framework for prediction using Markov model.

20.3.1 Encoding of Stock Market Price Behavior to Binary Sequence

Considering stock market day to day closing price, if current day closing price is higher than the previous day closing price then the encoded symbol put into '1' i.e. increasing behavior and similarly if current day index is lower or same as previous day then consider the symbol '0' i.e. non-increasing behavior.

Algorithm 3.1: Stock Behavior Algorithm(S)

```
Input: Stock market closing price index-F
Output: Bits of string-H
M=Length(F);
FOR I= 1 toM-1
    IF (F[I]<F(I+1))
        H[I]=1;
    ELSE
        H[I] =0;
END FOR
Return H;
```

20.3.2 Conversion Between Binary Sequences to Nucleotide Sequence

In this section we are converting stock market prices behavior or stock market indices behavior in form of binary sequence to DNA sequence by taking two bits of binary numbers into a single character of nucleotide (A, T, C, G) where A = (0,0), T = (1,1), C = (0,1)and G = (1,0) according to their pairing like 'A' pair with 'T' and 'C' pair with 'G'. For this representation we are developing a small algorithm which can convert entire binary sequence to DNA sequence.

Algorithm 3.2: Binary to DNA sequence Encoding Algorithm

```
Input: Bits of String-H
Output: Nucleotides of Strings-P
M= Length(H);
FOR J=1 to M-1
    IF (H[J]H[J+1] =00)
            P[J]=A;
    ELSE IF (H[J]H[J+1] =11)
            P[J] =T;
```

```
        ELSE IF (H[J]H[J+1] =10)
                P[J] =G;
        ELSE (H[J]H[J+1] =01)
                P[J] =C;
END FOR
Return P;
```

20.3.3 Zero-Order Markov Model

In this sub-session, we have applied the encoded DNA sequence to Zero order Markov Model. As we know, the zero order Markov model is memory less model. Here we count the number of probability occurrence between each symbols. All symbols are independent to each other. In this model, our input is a sequence of nucleotides consist of A, T, C, and G. Calculate probability occurrence of A, T, C, and G between given sequence and highest probability occurrence symbol will be the next two days prediction behavior of stock market.

For example in Zero order Markov model consider a sequence of A, T, C, and G.

S = TGGCTTTTGGCTCTCGTAACGCGATACATGAGGACT
TTGCGTTATTATGTAAACGGAGACGGAGCTGAACA
CACGTGCATATCAACAAAACGCCTATCTTTCCAATG
GGTGTGTCCATTGAACAAT

After prediction the new sequence is

R=TGGCTTTTGGCTCTCGTAACGCGATACATGAGGACTTT
GCGTTATTATGTAAACGGAGACGGAGCTGAACACG
TGCATATCAACAAAACGCCTATCTTTCCAATGGTGT
GTCCATTGAACAATT.

Figure 20.2 has represented the probability of occurrence of A, T, C, and G for the year 2001. Each occurrence is represented by different colors.

20.3.4 First-Order Markov Model

Like zero order Markov model, first order Markov model has also no memory. The only difference is all symbols are depends upon the previous state symbols. In this model we have calculated probability values of a pair of nucleotide sequence consist of pair between A, T, C, and G. Among all

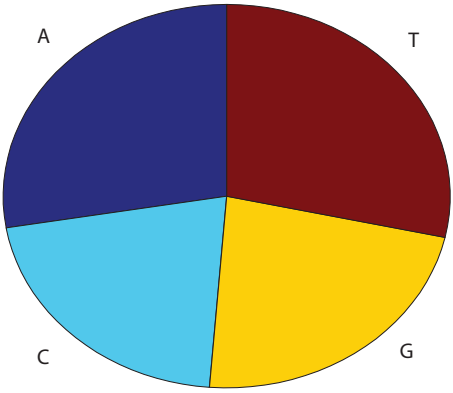


Figure 20.2 Example for zero order Markov model of BSE (2001).

pair probability values, highest occurrence score will be the next four days behavior prediction of stock market. Here input is same as previous model but get output as probability value with pair of nucleotides.

For example in First-order Markov model consider a sequence of A, T, C, and G.

S = TGGCTTTTGGCTCTCGTAACGCGATACATGAGGACT
TTGCGTTATTATGTAAACGGAGACGGAGCTGA
ACACGTGCATATCAACAAAACGCCTATCTTTCCA
ATGGTGTGTCCATTGAACAAT

After prediction the new sequence is

R = TGGCTTTTGGCTCTCGTAACGCGATACATGAGGACT
TTGCGTTATTATGTAAACGGAGACGGAGCTGA
ACACGTGCATATCAACAAAACGCCTATCTTTCCA
ATGGTGTGTCCATTGAACAATTG.

Figure 20.3 has represented the probability of occurrence of pairs for the year BSE 2001. Each occurrence has represented in different colors according to their starting pairs.

20.3.5 Second-Order Markov Model

Second order Markov model is applied for stock market behavior prediction because of two reasons: (1) It is an advanced Markov model and has memory. (2) It can predict the time series data for both long- and

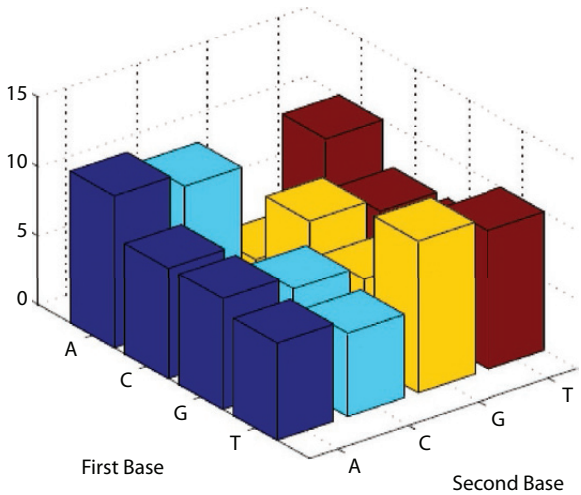


Figure 20.3 Example for first order Markov model of BSE (2001).

short-term basis. In this model, input data consists of sequences of nucleotides. From given biological sequence find the probability occurrence of each codon. Whichever codon occurrence is more considered then that codon serves as output and behaves as next six days behavior of stock market. Here current state depends upon previous states.

For example in Second order Markov model consider a sequence of A, T, C, and G.

S = TGGCTTTTGGCTCTCGTAACGCGATACATGAGGACT
TTGCGTTATTATGTAAACGGAGACGGAGCTG
AACACGTGCATATCAACAAAACGCCTATCTTT
CCAATGGTGTGTCCATTGAACAAT

After prediction the new sequence is

R=TGGCTTTTGGCTCTCGTAACGCGATACATGAGGACT
TTGCGTTATTATGTAAACGGAGACGGAGCTGA
ACACGTGCATATCAACAAAACGCCTATCTTT
CCAATGGTGTGTCCATTGAACAATTTG.

20.3.6 Hidden Markov Model

The Hidden Markov Model (HMM) is a limited state machine with a limited number of states, each related with a likelihood circulation.

Table 20.1 illustrates the probability of occarances of each codons that define a second order Markov model. The advances between states can't be straightforwardly estimated (covered up), yet in a specific express a perception can be created. It is simply the perceptions and not the states which are noticeable to an outside eyewitness. Be that as it may, by applying a progression of factual methods it is conceivable to pick up knowledge into the shrouded states by means of the perceptions they create. On account of DNA examination, we watch a strand of DNA which we accept can be portioned into homogeneous areas to recognize the particular elements of the DNA strand. Using HMMs we can figure out which parts of the strand have a place with which fragments by coordinating sections to shrouded states. It has been applied to a wide scope of uses including discourse acknowledgment, signal preparing and econometrics to give some examples. Here we will talk about the hypothesis behind HMMs and the applications that they have in DNA grouping investigation.

Table 20.1 Example for second order Markov model of BSE (2001).

AAA - 1	AAC - 2	AAG - 0	AAT - 1
ACA - 0	ACC - 0	ACG - 2	ACT - 1
AGA - 0	AGC - 0	AGG - 1	AGT - 0
ATA - 0	ATC - 2	ATG - 2	ATT - 0
CAA - 0	CAC - 1	CAG - 0	CAT - 1
CCA - 2	CCC - 0	CCG - 0	CCT - 1
CGA - 1	CGC - 0	CGG - 0	CGT - 1
CTA - 0	CTC - 1	CTG - 0	CTT - 1
GAA - 1	GAC - 1	GAG - 0	GAT - 0
GCA - 0	GCC - 0	GCG - 0	GCT - 2
GGA - 2	GGC - 0	GGG - 0	GGT - 0
GTA - 2	GTC - 0	GTG - 2	GTT - 0
TAA - 0	TAC - 1	TAG - 0	TAT - 2
TCA - 0	TCC - 0	TCG - 0	TCT - 0
TGA - 0	TGC - 0	TGG - 1	TGT - 1
TTA - 0	TTC - 0	TTG - 3	TTT - 1

We will be explicitly keen on examining how might we decide from which express our perceptions are being produced?, How do we decide the parameters of our model? and How do we decide the arrangement of concealed states given our perceptions? While there are numerous uses of HMMs we will just be worried about their utilization as far as DNA grouping examination and will cover models in this writing where HMMs have been effectively utilized.

HMM is a probabilistic finite state machine consists of 5 tuple machine (Q, O, X, Y, π) characterized by the following

1. Q = Set of finite number of states
2. O = Set of finite number of observation symbols
3. $X = \{x_{ij}\}$ = State transition probabilities matrix
4. $Y = \{y_j(O_t)\}$ = Observation emission probability matrix
5. $\pi = \{\pi_i\}$ Initial state distribution matrix

Where

$N = |Q|$

$M = |O|$

T = Observation sequence total length

$\lambda = (X, Y, \pi)$ the overall HMM model

To find λ three algorithms are required like In Forward algorithm, Viterbi algorithm, and Baum–Welch algorithm. Since it is a tool based model and we have generated a prediction algorithm to predict next state sequence from given observation sequence. The detail of the algorithm is described below.

20.3.6.1 HMM for Stock Market Behavior Prediction

In this area we have built up a HMM based apparatus for time arrangement determining, for example for the financial exchange conduct forecast. While actualizing the HMM, the decision of the model, decision of the quantity of states and perception image (ceaseless or discrete or multiblend) become a dreary assignment. For example, we have utilized left-right HMM with 4 states. In the investigation, our goal was to anticipate the following day's end cost for a particular financial exchange share utilizing previously mentioned HMM model. For preparing the model, past one and a half years (roughly) day by day information were utilized and ongoing most recent multi month's information were utilized to test the productivity of the model.

1. Given the parameters of the model, figure the likelihood of a specific yield succession.

This requires summation over all conceivable state successions however should be possible productively utilizing the Forward calculation, which is a type of dynamic programming.

2. Given the parameters of the model and a specific yield grouping, discover the state succession that is well on the way to have created that yield arrangement. This requires finding a most extreme over all conceivable state successions, however can comparably be unravelled effectively by the Viterbi's calculation.
3. Given a yield arrangement or a lot of such successions, locate the most probable arrangement of state progress and yield probabilities. At the end of the day, infer the most extreme probability gauge of the parameters of the HMM given a dataset of yield arrangements. No tractable calculation is known for taking care of this issue precisely, yet a neighborhood most extreme probability can be inferred productively utilizing the Baum-Welch calculation. The Baum–Welch calculation is a case of a forward-in reverse calculation and is an extraordinary instance of the Expectation–augmentation calculation.

In this session, we have mentioned HMM with 4 states A, T, C, and G as follows.

Algorithm 3.3: Observation sequence prediction using HMM (N,M,T,Q,X,Y,I)

Input: N=4, M=4, State sequence-Q (A, T, C, G), Initial Probability-π, Transition Probability-X, Observation Emission Probability-Y, Length of observation symbol-T

Output: Predicted observation symbol-O

```
FOR     i = 1 to T
  V[i,1]= π= {π_i};
  P[i,1]= π= {π_i} I;
   O[i]=φ;
END FOR
    FOR i = 1 to  T
            FOR j = 1 to  M
                    FOR k = 1 to  N
    V [i,j]+ = V [i,k] * X[k,j];
                    END FOR
  V [i,j] = max[k] *Y [j,Q[i]];
  P[i,j] = argmax[k];
    END FOR
```

```
14. O[i] = P[i,argmax[v[i,j]]];
15. END FOR
16. Return O;
```

From above algorithm, numbers of observation symbols are required for specified number of future trading days. The output symbols are in form of nucleotide having four variables (A, T, C, G), and this predicted behavior decoded into binary string in next session.

20.3.7 Decoding Predicted DNA Sequence into a Binary String

The behavior of stock market is predicted using pair wise sequence alignment. The predicted sequences are derived by searching the pattern that exists in the human gene bank. The predicted behavior is also a set of nucleotide sequences which are decoded into binary by mapping A, T, G, and C to 00, 11, 10, and 01 respectively. Where 00, 11, 10, and 01 are denoted non-increasing–non-increasing, increasing–increasing, increasing–non-increasing and non-increasing–increasing behavior. In this phase, we have developed an algorithm 3.4, which is converted the output predicted DNA sequence of different Markov model results into a predicted binary string.

Algorithm 3.4: Decode predicted DNA sequence to binary string (R)

```
Input: Predicted nucleotides of next month stock market index
    behavior-R
Output: Stock market predicted Binary string -Z
1.  V= Length(R);
2.  FOR k=1 to V
3.      l = 2 *k – 1;
4.      IF(R[k]==A)
5.          Z[l]Z[l+1]==00;
6.      ELSE IF (R[k]==T)
7.          Z[l]Z[l+1]=11;
8.      ELSE IF (R[k]==G)
9.          Z[l]Z[l+1]=10;
10.     ELSE
11.         Z[l]Z[l+1]=01;
12.     END IF
13.     END FOR
14.     Return Z;
```

20.3.8 Mismatching Analysis of Stock Market Behavior

The mismatch analysis between predicted binary string and actual Sensex data can be measured using hamming distance algorithm approach. Hamming distance algorithm is based on principle XOR operation. In this session, we have given two binary string of equal length; we have calculated the XOR between two bit strings. An algorithm 3.5 finds mismatch occurs between stock market actual data with predicted data.

Algorithm 3.5: Hamming distance algorithm (X, Z)

```
Input: Bits of actual stock market index behavior-X,
       Bits of predicted stock market index behavior-Z
Output: Hamming distance-D
E= length(X)= Length(Z);
D=0;
FOR w=1 to E
    IF (X(w)! = Z(w))
    D=D+1;
    END IF
END FOR
Return D;
```

20.4 Experimental Work

For implementation the system requirements are I3 processor, 4.00 GB RAM, Windows 8.1 operating system, Matlab R2013b.

20.4.1 Dataset Preparation

We have used daily historical data for stocks of the Bombay Stock Exchange data from yahoo finance. The total dataset contains 15 year BSE closing price starting from 01/01/2001 to 1/1/2015. The analyses of the stocks are done using the closing values of each day. Such Sensex dataset computes the probability of occurrence using different Markov models to predict the stock market price behavior. Zero-order Markov model predicts next two days stock market behavior. Similarly, in first-order Markov model, we will get a pair of nucleotides as a predicted sequence which can show four days behavior. Second order Markov model will predict the codon of DNA sequence and it contains next six days behavior whereas another model called HMM with

some hidden state to predict a sequence of nucleotides for next upcoming Sensex days. The prediction results and their analysis are explained below.

20.4.2 Results and Analysis

The predicted results of different Markov model are explained two days, four days, six days or next month or year for stock market price behavior. In this part, we have evaluated a comparative report that is mismatch analysis between predicted stock market behavior vs actual stock market behavior. The mismatch analysis computes the hamming distance results model wise. On the starting point, we are making the results produce zero-order Markov model. Here we will evaluate the probability occurrence of each nucleotide A, T, C, and G. Highest probably measure nucleotide will be the next two Sensex days behavior. The outcomes of the zero order Markov model are given in Table 20.2.

Table 20.2 illustrates the hamming distance calculated between actual stock market behavior with respect to predicted stock behavior. Here the

Table 20.2 Hamming distance of BSE (2002 to 2014) using zero order Markov model.

Year	Hamming Distance (D)
2002	0
2003	1
2004	0
2005	0
2006	0
2007	0
2008	0
2009	2
2010	1
2011	1
2012	0
2013	0
2014	0

first column indicates years represent one by one row wise and column two contains the hamming distance results of BSE Sensex from 2002 to 2014 using zero-order Markov models. To analyze the prediction result out of thirteen years prediction result nine years 100 percentage match with our predicted behavior and except one year rest of the years match 50 percentage i.e. partial matching occurs with our predicted behavior. Table 20.3 illustrates the hamming distance calculated between actual stock market behavior with respect to predicted stock behavior. Here the first column indicates years represent one by one row wise and column two contains the hamming distance results of BSE Sensex from 2002 to 2014 using First-order Markov models. To analyze the prediction result out of thirteen years prediction result five years 100 percentage match with our predicted behavior and except two years rest of the years match 50 percentage i.e. partial matching occurs with our predicted behavior.

Table 20.4 illustrates the hamming distance calculated between actual stock market behaviors with respect to predicted stock behavior. Here the first column indicates years represent one by one row wise and column two

Table 20.3 Hamming distance of BSE (2002 to 2014) using first order Markov model.

Year	Hamming Distance (D)
2002	1
2003	3
2004	1
2005	0
2006	1
2007	0
2008	1
2009	1
2010	0
2011	2
2012	0
2013	0
2014	1

Table 20.4 Hamming distance of BSE (2002 to 2014) using Second order Markov model.

Year	Hamming Distance (D)
2002	1
2003	3
2004	1
2005	0
2006	2
2007	0
2008	1
2009	1
2010	0
2011	2
2012	0
2013	0
2014	1

contains the hamming distance results of BSE Sensex from 2002 to 2014 using second-order Markov models. To analyze the prediction result out of thirteen years prediction result one year 100 percentage matches with our predicted behavior and one of the year 75 percentage matching occurs with our prediction result. In few cases, 50 percentage matching occurs between the years.

In order to predict the behavior of the stock market, we have divided the dataset into training and testing data. Suppose we have taken four your data set as a training data (i.e. from BSE 2001 to 2004) and test data will be BSE 2005.

Using HMM computes the observation nucleotides equal length with respect to test data and study their mismatch analysis using hamming distance algorithm. Figure 20.4 illustrates the BSE January 2005 increasing and non increasing behavior of stock market in form of 0s and 1s between actual fluctuation in the stock market and predicted behavior using HMM. Here X-axis represents the trading days for the month of January 2005 and Y-axis represents the increasing and non-increasing behavior.

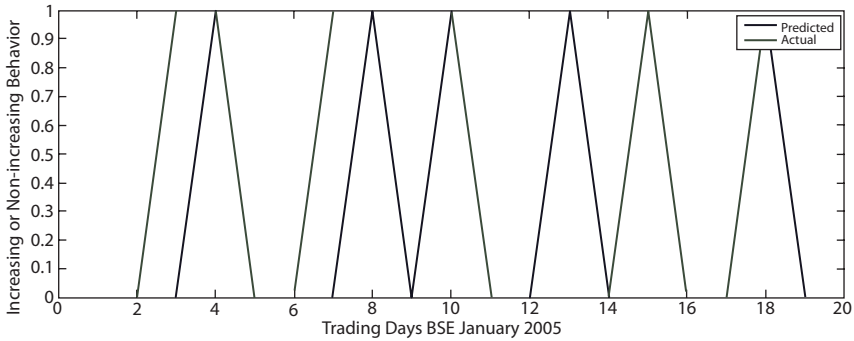


Figure 20.4 **B**ehavior of BSE, January 2005 (actual vs predicted).

Figure 20.5 illustrates the BSE February 2005 increasing and non-increasing behavior of stock market in form of 0s and 1s between actual fluctuation in the stock market and predicted behavior using HMM. Here X-axis represents the trading days for the month of February 2005 and Y-axis represents the increasing and non-increasing behavior.

Figure 20.6 illustrates the BSE March 2005 increasing and non increasing behavior of stock market in form of 0s and 1s between actual fluctuation in the stock market and predicted behavior using HMM. Here X-axis represents the trading days for the month of March 2005 and Y-axis represents the increasing and non-increasing behavior.

Figure 20.7 illustrates the BSE March 2010 increasing and non increasing behavior of stock market in form of 0s and 1s between actual fluctuation in the stock market and predicted behavior using HMM. Here X-axis represents the trading days for the month of March 2010 and Y-axis represents the increasing and non-increasing behavior.

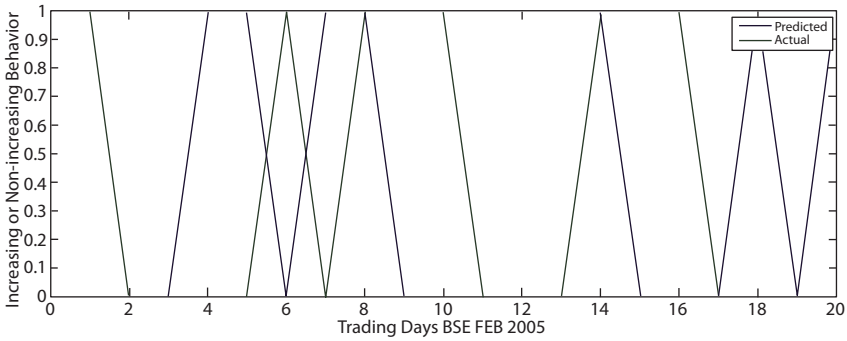


Figure 20.5 Behavior of BSE, February 2005 (actual vs predicted).

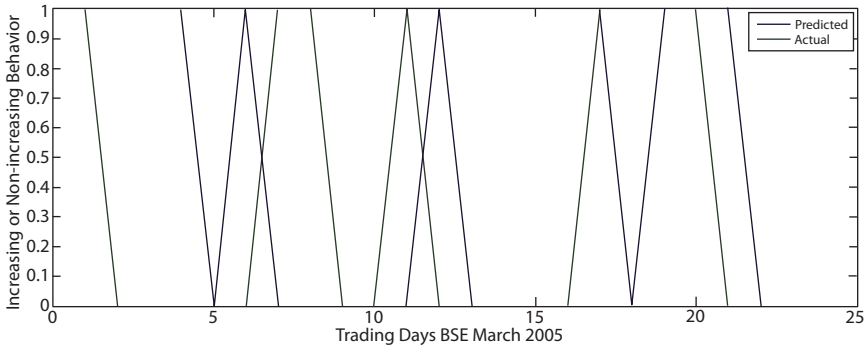


Figure 20.6 Behavior of BSE, March 2005 (actual vs predicted).

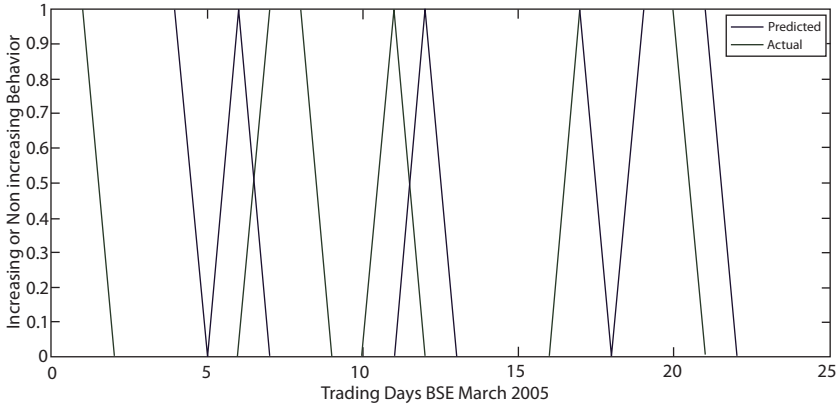


Figure 20.7 Behavior of BSE, March 2010 (actual vs predicted).

20.4.3 Performance Comparison Between Different Orders Markov Models

The performance evaluation of stock market price behavior prediction using different Markov models (MM) techniques are described in this subsection. The performance comparison results illustrate between actual stock market behavior and proposed model using Markov models (MM) techniques.

In terms of performance evaluation among proposed models, we have found the matching percentage of stock market price behavior using the zero order, first order, second order Markov model, and HMM techniques to predict the behavior for two days. The percentage of matching with actual behavior is considered as the order of accuracy of that model. In the month of November the order of accuracy in terms of percentage for first two days

stock market price behavior are presented in Figure 20.8. Here X-axis represents the different proposed prediction models and Y-axis represents the percentage of matching with stock market actual behavior. From Figure 20.8, we have found that among all four models HMM model is more accurate i.e. 96 percentages than other models.

In terms of performance evaluation among proposed models, we have found the matching percentage of stock market price behavior using the first order, second order Markov model, and HMM techniques to predict the behavior for four days. The percentage of matching with actual behavior is considered as the order of accuracy of that model. In the month of November the order of accuracy in terms of percentage for first two days stock market price behavior are presented in Figure 20.9. Here X-axis represents the different proposed prediction models and Y-axis represents the percentage of matching with stock market actual behavior. From Figure 20.9, we have found that among all three models HMM model is more accurate i.e. 96 percentages than other models.

In terms of performance evaluation among proposed models, we have found the matching percentage of stock market price behavior using the second order Markov model, and HMM techniques to predict the behavior for four days. The percentage of matching with actual behavior is considered as the order of accuracy of that model. In the month of November, the order of accuracy in terms of percentage for first two days stock market price behavior are presented in Figure 20.10. Here X-axis represents the different proposed prediction models and Y-axis represents the percentage of matching with stock market actual behavior. From Figure 20.10, we

Figure 20.8 First two days prediction accuracy percentage of BSE (November).

Figure 20.9 First four days prediction accuracy percentage of BSE (November).

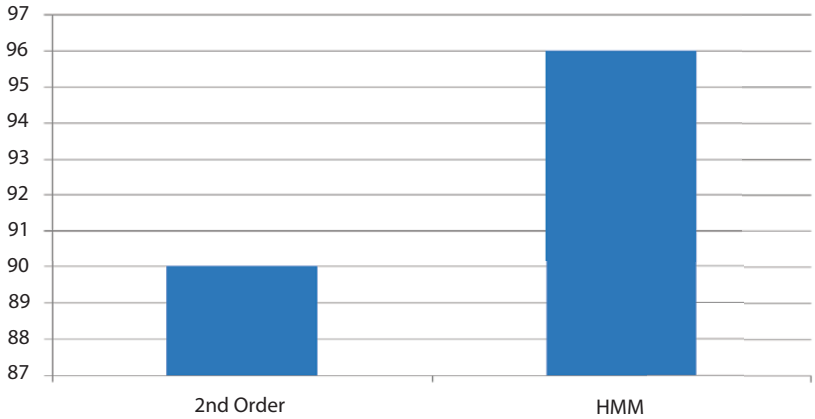


Figure 20.10 First six days prediction accuracy percentage of BSE (November).

have found that among two models HMM model is more accurate i.e. 96 percentages than other models.

20.5 Conclusions and Future Work

In practice, the behavior of stock markets varies through different markets and different countries. In Indian stock market, the rising and falling depend upon various factors. In this work, we have focused on BSE Sensex data and predict their behavior using different order Markov Models. Our

model consists of eight phases and from these eight phases, four phases are different Markov models. Phase three is zero order which predicts next two days stock market behavior. Similarly, phase four explains four days prediction using first order Markov model. Second order Markov model discusses in phase five which can predict next six days stock market behavior and phase six indicates prediction using HMM.

From the experimental results, we are examined that zero order, first order, and second order Markov model prediction on an average 88 percentage or more. But they are limited to a number of days' prediction. Using HMM to the study of stock market price behavior that numbers of days prediction can possible by defining a number of observation symbols you needed and its accuracy is somehow high in many cases i.e. more than 90%. In a future study, we will predict the stock market behavior using higher order Markov model and hope it can give more accuracy than other models.

References

1. Welch, L.R., Hidden Markov models and the Baum–Welch algorithm. *IEEE Inf. Theory Soc. Newsl.*, 53, 4, 10–13, 2003.
2. Cao, L. and Tay, F.E., Financial forecasting using support vector machines. *Neural Comput. Appl.*, 10, 2, 184–192, 2001.
3. Choi, J.H., Lee, M.K., Rhee, M.W., Trading S&P 500 stock index futures using a neural network, in: *Proceedings of the third annual international conference on artificial intelligence applications on Wall Street*, pp. 63–72, June 1995.
4. Hassan, M.R., A combination of hidden Markov model and fuzzy model for stock market forecasting. *Neurocomputing*, 72, 16–18, 3439–3446, 2009.
5. Gupta, A. and Dhingra, B., Stock market prediction using hidden Markov models, in: *2012 Students Conference on Engineering and Systems, Proceedings of the IEEE*, pp. 1–4, 2012.
6. Rabiner, L.R., A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, 77, 2, 257–286, 1989.
7. Vercher, E., Bermúdez, J.D., Segura, J.V., Fuzzy portfolio optimization under downside risk measures. *Fuzzy Sets Syst.*, 158, 7, 769–782, 2007.
8. Wang, J.H. and Leu, J.Y., Stock market trend prediction using ARIMA-based neural networks, in: *Proceedings of International Conference on Neural Networks (ICNN'96), Proceedings of the IEEE*, vol. 4, pp. 2160–2165, 1996.
9. Weigend, A.S. and Shi, S., Predicting daily probability distributions of S&P500 returns. *J. Forecast.*, 19, 4, 375–392, 2000.

Index

Accuracy, 4, 5, 8, 26, 27, 32, 404, 409, 420
  true negative, 453
  true positive, 453
ADNI (Alzheimer's Disease Neuroimaging initiative), 286–287
AI-based microscopes, 289
Alanine-scanning mutagenesis (ASM), 405
ANFIS, 464
ANN, 465
Anomaly detection, 35
ANOVA, 84–85, 394
Application of bioinformatics,
  agriculture, 55
  environment, 54
  health sector, 53–54
Application of GWO in microarray data, 232, 240–242
Applications of ML in genomics,
  gene editing, 188
  gene sequencing, 188
  newborn genetic screening, 188
  pharmacy genomics, 188
  service to consumer, 188
ARIMA model, 463
Artificial intelligence, 3, 4, 28, 324
Artificial intelligence-based model, 289
Artificial neural network, 4, 8, 24, 29, 33, 34, 288, 326
  architecture, 61–62
  backpropagation, 63
  classification, 62–63
  features, 61, 72
  performance analysis, 72–73
Association, 36
AtomNet neural network, 289
Attributes, 35, 431, 438
AUC, 15, 16, 19–22, 24–27
Auto regression (AR), 297
Autoencoder, 87, 100, 250, 288
Automated sleep stage scoring, 349
Autoregressive integrated moving average (ARIMA), 297
Autoregressive moving average (ARMA), 297

Back propagation network, 326
Backpropagation, 322
Backward feature elimination (BFE), 85, 92–94, 104, 106
Bagging, 177, 178
Beijing Genomics Institute, 396
Beta, 289
Beth Israel Deaconess, 289
Big data, 285
Bina Technologies, 396
Binary string, 496–497
Biodetails repositories, 395
Bioinformatics, 35–42, 44, 46, 392
  data analytics in, 394–396
  machine learning and, 393–394
  recent trends in, 394
Bioinformatics algorithms, 488
Biologcal database,
  primary, 58
  secondary, 58
Biological cells, 397–398

Biological processes, 35, 36, 38, 39
Biomarker, 359, 388
Biomarker selection, 212
Biomedical, 432–435, 439
BLAST (basic local alignment search tool), 467–468, 471
Bombay Stock Exchange (BSE), 470–481, 497–504
Boosting, 177
Bootstrap, 452
Breast cancer types, 192

C4.5, 176
Cancer detection, 314
CART, 176
Categorical column, 450
CERN, 396
CHDS (datasets of child health and development), 287
Classification, 5, 7, 10–12, 15–18, 20, 23, 27–33, 79, 83, 90–92, 100, 105, 197, 363, 367, 368–372, 374–376, 379, 380
Classification algorithms, 394
Clinical, 431, 432, 434, 435, 438
Clinical conditions, 38
Clinical research, 35
Cloud computing, 396
Cluster analysis, 285
Clustering, 35–46, 202, 253, 330
CNNs, 287, 288
Color vision, 314
Complex prime numerical representation (CPNR), 407–410, 416–417, 421–423
Computational complexity, 82, 84
Confirmed cases, 295
Continuous wavelet transform (CWT), 413–414
Corona virus, 294
Correlation coefficient, 84, 447, 448
Curse of dimensionality, 80

Data, 431–438
Data pre-processing, 296, 448
  necessity, 58–59
  techniques, 58–59
Dataset, 3, 7, 12, 14, 15, 19, 20–22, 24, 26, 27, 30, 436, 438
Decision, 431, 434, 436, 438–440
Decision tree, 21, 22, 26, 27, 30, 174, 394
Deep learning, 299
Deep learning in healthcare, 287–288
Dementia, 283
Density based, 38, 44
  border point, 45
  core point, 44, 45
  DBSCAN, 38, 44
  EPS(€), 44
  minpts, 44, 45
Dependent variable, 455
Digital signal processing (DSP), 391–392, 403–404
Dimension, 79–83, 88, 90, 94, 96, 101, 103, 106
Dimension reduction, 79–83, 101, 106
Dimensionality reduction, 37, 79–80, 82–83, 85–87, 91, 96, 100–101, 104, 106, 114
  feature extraction, 114
    LDA (linear discriminant analysis), 115
    ORB (oriented fast and rotated brief), 116
    PCA (principal component analysis), 115
    SIFT (scale invariant feature transform), 115–116
    SURF (speeded up robust features), 116
Discrete wavelet transforms, 413–414
DNA, 181
DNA microarray technology, 212
DNA, RNA and proteins, 398–403
DRT, 80, 81, 86–87

Drug design, 35
Ductal carcinoma *in situ* (DCIS), 192

Efficient market hypothesis, 464
Eigenvectors, 88–89
Electroencephalogram (EEG), 359, 364, 365, 367, 369, 372–388
Electron–ion interaction potential (EIIP), 407–409
Embedded method, 85, 148
Empirical mode decomposition (EMD), 86
Enlitic, 289
Ensemble approach, 148–150, 158–159, 162–169
Ensemble methods, 299
Ensembling technique, 177, 185
Entropy, 176
Error, 14, 15
European Bioinformatics Institute (EBI), 395
Event-related potential (ERP), 363, 382
Experimental data, 349

F1 score, 409, 420, 453
Factor analysis (FA), 86–87, 100, 104
False positive, 453
FCM clustering, 330
Feature extraction, 329
Feature extraction technique, 411–414
  discrete wavelet transforms, 413–414
  resonant recognition model (RRM), 412
Feature selection, 79, 82–85, 101, 105, 107, 147–149, 159–161, 164–165, 167–170, 212
Feature space, 79–81, 86, 88, 92, 99–102, 104
Features, 40, 79, 81, 83–89, 91–98, 100, 102–104, 105, 107
  feature reduction, 40
  principal component analysis, 40
Feedback, 3, 6, 9, 10
Filter methods, 84, 86, 148
Financial exchange record, 486
Finite state machine, 494
Flowchart, 112–113
Forward feature construction (FFC), 85, 87, 93, 106
Freenome, 289
F-test, 394
Fuzzy approach, 299
Fuzzy time arrangement model, 463

GA, 465
Gaea, 396
Gene expression, 146–147, 149–151, 154, 160, 164, 167–170
Gene expression data, 40–46
  gene homology, 47
  gene regulation, 39
  genetic information, 39
  genetics functionality, 39
Gene expression problem, 212
Gene finding, 56
Gene selection, 145–147, 149–152, 164, 166
General regression neural networks, 326
Genetic algorithm, 297
Genome annotation, 57
Genome datasets, 287
Genomic signal process (GSP), 391–392, 411
  EIIP & CPNR-based mapping, 407–411
  feature extraction technique, 411–414
  hotspots identification, 404–405
  protein sequences, overview, 405–407
Genomics,
  functional genomics, 183
  regulatory genomics, 183
  structural genomics, 183
Gini impurity, 176
Global combinatory optimization, 212
Gradient boosting, 452

Graphical models, 393
Gray-level co-occurrence matrix, 320
Greedy algorithms, 85
Grid-based, 38, 45, 46
  statistical information grid (STING), 38, 46
GWO,
  algorithm, 219
  flowchart, 220
  grey wolf optimization, 212–216
  mathematical model, 217–219
  principle, 216
GWO variants,
  extension, 221, 228–231
  hybridization, 221–227
  modification, 221, 232–237

Hadoop structure, 396
Hamming distance algorithm, 488, 497
Hamming distance algorithm approach, 470
Heart disease, 3, 7, 9–15, 18–23, 26, 27, 30
Hessian–Laplace, 316
Hidden layer, 327
Hidden Markov Model (HMM), 394, 486–488, 492–496, 500–504
Hierarchical clustering, 42, 43, 44
  agglomerative clustering, 42
  agglomerative nesting (AGNES), 37, 43, 44
  balanced iterative reducing and clustering using hierachires (BRICH), 44
  CHAMELEON, 43, 44
  clustering using representation (CURE), 43
  divisive analysis (DIANA), 43, 44
  divisive clustering, 42, 43
High-dimensional, 80, 81, 88, 99–101
Hinton data center group, 395
Histogram evaluation, 315
Hospital datasets, 287
Hotspots identification, 392, 404–405
  algorithm, 414–415
Human fibroblast growth factor (HFGF) protein sequence, 411
Human interference, 285

IBM's Watson AI technology, 288
ID3, 176
iLet platform, 289
Image datasets, 286–287
Implementation I: numerical dataset, 130
  dataset description, 130–131
  discussion, 133–134
  results, 131–133
Implementation II: image dataset, 134
  dataset description, 134
  method used, 134
  results, 134
    bacterial leaf blight on rice leaves, 137
    brown spots on rice leaves, 136
    color histogram comparison:using correlation method, 138
    leaf smut on rice leaves, 134–135
    results of ORB feature extraction and brute forematching, 134
Independent component analysis (ICA), 86, 94, 104–105
Infectious disease, 431–438
Information gain, 176
Insitro, 288
Invasive ductal carcinoma (IDC), 192
ISOMAP, 86
Issues in bioinformatics,
  solutions, 57
  types, 56

K-means pre-segmentation, 321
K-nearest neighbor, 20, 21, 26, 27, 32, 318

KNN, 20, 21
Knowledge, 3, 6, 7, 9, 12, 29, 30

Label, 36, 38
Label encoding, 450
Laplacian of Gaussian, 319
LASSO, 85
Learning process, 5, 7, 10
Liblinear, 329
Linear discriminant analysis (LDA), 86, 90, 106
Linear regression, 200
Literature survey, 116–127
Location, 436
Logistic, 15–17, 26–28, 31
Logistic regression, 197, 285–286, 451
Long short term memory (LSTM), 300–302
Low variance filter (LVF), 84, 87, 95–96, 104
Low-dimensional, 80, 82, 88, 90, 92, 100

Machine learners, 79–80, 82, 97
Machine learning, 3, 4, 5, 18, 21, 22, 27–30, 33, 36, 37, 79–80, 84–85, 87, 91, 101, 212, 392, 431, 436–438, 444–446, 448, 450, 452, 454
  algorithms, 396–397
  applications, 285
  applications for bioinformatics, 393–394
  approaches for different purposes, 284–286
  metrics, analysis using, 418–420
  role in bioinformatics, 393
  software agents, 397
Machine learning (ML) in rapid stress phenotyping, 113–114
Machine learning for classification,
  advantages, 60
  application in bioinformatics, 60
  definition, 60
  different models, 61, 64
  model comparison, 68–71
Machine learning techniques, 145, 153
  supervised techniques, 163
  unsupervised techniques, 147
Markov model, 393
Markov models (MM) techniques, 502–504
Maximally stable extremal regions, 316
Medical dataset for research, 286–287
Medical imaging and diagnostics, 288–289
Melanoma, 322
Metaheuristic algorithms, 212
Methodology, 353
Microarray, 149–150
Microarray analysis,
  DNA chips, 38
  matrices, 38–39
  microscopic slide, 38
  probes, 38
Microarray data analysis,
  microarray data, 212–215, 243
Microsoft's Project Inner Eye tools, 288
MIMIC critical care database, 287
Missing values ratio (MVR), 84, 87, 89
Model-based,
  neuron, 45–46
  self organizing maps (SOM), 35
Modules, 434
Moving average (MA), 297
Multi scale cancer detection, 315

Naïve Bayes, 394, 452
Naïve Bayes classifier, 323
Naïve method, 297
Network generalization, 82, 84
Networks, 432, 436, 437, 439
Neural networks, 393
Neural system models, 463
Neuron-fuzzy framework, 463–464
Neurons, 322
N-fold cross validation,
  overfitting, 454

NLP, 28
Noisy data, 41, 44, 46
NP hard, 212
NRSC/ISRO open data archive2, 396
Numerical, 431, 438

OASIS (Open Access series of Imaging Studies), 287
Omics, 145–171
  epigenomics, 145, 146, 151, 166, 168
  genomics, 145, 146, 151, 154, 160, 166, 168–169, 171
  multi-omics, 166, 169–170
  proteomics, 145, 146, 151, 166–167
  single omics, 145, 151, 153
  transcriopomics, 145, 146, 151, 166
Omics data integration, 145, 152, 155–158, 163, 169, 170
  clustering-based integration, 160
  concatenation-based integration, 153–154
  correlation-based integration, 159
  horizontal integration, 152–153
  model-based integration, 153–154, 159
  network-based integration, 161–162, 171
  transformation-based integration, 153–154, 157
  vertical integration, 152–153
Ontology, 431–438
Optimization, 82, 90, 397
Optimization problem, 213
Outliers, 41, 44
Overfitting, 82, 84–86, 98, 102

Partition algorithms, 40, 41
  centroids, 41
  cluster center initialization algorithm (CCIA), 40, 41
  clustering large applications (CLARA), 40, 42
  intelligent kernel k-mean (IKKM), 40–42
  K-means, 37, 40, 41, 42, 45
PathAI's technology, 288
Pattern classification, 79
Pattern layer, 327
Patterns, 79, 80, 82, 90, 95–96
PC vision, 323
PCA, 86–92, 95, 97, 104, 106
Pearson's correlation, 84
PEGASOS, 329
Performance, 3, 5, 6, 15, 20, 26, 31
Pfizer, 288
Platt's sequential minimal optimization, 328
Polynomial regression, 303
Polysomnography signal analysis, 341
Precision, 409, 453
Predicted Markov models (HMM), 464–465
Principal component, 86, 88, 104–105
Processing, 15, 24, 25, 28, 29, 32–34
Project inner eye tools (Microsoft), 288
Proteins, 183
Protein-target interconnection, 404–405
Proteomes, 37
Proximity, 40
  Euclidean distance, 40, 41
  Mahalanobis distance, 40
  Manhattan distance, 40, 43

Q-linear convergence property, 329

Random forest, 18–20, 26, 27, 32, 87, 97, 106, 180, 298, 394
  CART-classification, 452
  decision tree, 452
Random forest classifier, 199
Recall,
  false negative, 453
Redundant, 79–80, 82–83, 86, 90, 97, 103–104
Regression, 7, 8, 12–18, 20, 23, 26–28, 30–33, 79, 85, 87, 93, 199, 394
Reinforcement, 3, 7, 9, 10, 27, 29, 30

Reinforcement learning, 36, 196
Relation, 432–436
Reliability, 79
Resonant recognition model (RRM), 412
RIDGE regression, 85
RNN, 288
ROC, 15, 16, 19–22, 24, 26
RSI, 464

Schizophrenia, 359–363, 367, 372, 379–383
Securities exchange cost, 486
Sequencing,
  genome sequencing, 56
  issues, 56–57
  sequence comparison, 57
Site-directed mutagenesis, 404–405
Soft clustering, 38, 46
  fuzzy class membership, 38, 46
  probabilistic model, 45, 46
Speech recognition, 25, 33
Standardization,
  average, 449
  mean, 449
  standard deviation, 450
  variance, 450
  z scores, 449
Statistical tests, 84–85
Stock market behavior and bioinformatics,
  approach, 466–470
  experimental study, 470–479, 481
  literature review, 463–465
Stock market price behavior prediction using Markov models,
  experimental work, 497–504
  literature survey, 487–488
  proposed work, 488–497
Supervised, 3, 7, 8, 10, 18, 20, 21, 26–30, 32, 33
Supervised learning, 35–37, 46, 196, 251, 285, 397
Supervised learning model, 454
Support, 431, 434, 436, 438
Support vector classifier, 22, 26, 27
Support vector machine, 22, 23, 24, 298, 327, 393
  hyperplane, 451
SVM, 8, 22–24

Task, 5, 10
t-distributed stochastic neighbor embedding (t-SNE), 86–87, 99
Tissue, 37, 38
Traditional learning, 4
Training algorithms, 285
t-test, 394
Types of plant stress, 128
  biotic stress, 128–129
    bacterial pathogen, 129–130
    fungal pathogen, 129

Ultrasonography, 444
Uniform manifold approximation and projection (UMAP), 86–87, 101, 104–106
Unlabled dataset, 37
Unsupervised, 3, 7–9, 27–30, 32
Unsupervised learning, 35–37, 46, 196, 285, 397, 431, 440

Vector quantization (VQ), 87, 102–104
Viterbi design, 394

Watershed, 255
WEKA instrument, 323
Wrapper method, 84, 86, 148–150, 162, 167, 169

XG boost (extreme gradient boosting) algorithm, 303

Z-score, 85