

# PROCESS MINING BASED CRITICAL PATH RECOMMENDATION IN HEALTHCARE MANAGEMENT

Thesis

Submitted in partial fulfillment of the requirements for the degree of  
DOCTOR OF PHILOSOPHY

by

**LIKEWIN THOMAS**



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING  
NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA,  
SURATHKAL, MANGALORE - 575025

FEBRUARY 2018



*Dedicated to my  
Parents, Wife, Son, Guide and  
Dear Friend (late) Madhu*





**DECLARATION**

*by the Ph.D. Research Scholar*

I hereby **declare** that the Research Thesis entitled **Process Mining Based Critical Path Recommendation in Healthcare Management** which is being submitted to the **National Institute of Technology Karnataka, Surathkal** in partial fulfilment of the requirements for the award of the Degree of **Doctor of Philosophy in Computer Science and Engineering** is a **bonafide report of the research work carried out by me**. The material contained in this Research Thesis has not been submitted to any University or Institution for the award of any degree.

**(CS12F05, Likewin Thomas)**

(Register Number, Name & Signature of Research Scholar)

Department of Computer Science and Engineering

Place: NITK, Surathkal.

Date: February 9, 2018

---



**CERTIFICATE**

This is to *certify* that the Research Thesis entitled **Process Mining Based Critical Path Recommendation in Healthcare Management** submitted by **Likewin Thomas**, (Register Number: **CS12F05**) as the record of the research work carried out by him, is *accepted as the Research Thesis submission* in partial fulfillment of the requirements for the award of degree of **Doctor of Philosophy**.

Dr. Annappa B

Research Supervisor

(Name and Signature with Date and Seal)

Chairman - DRPC

(Signature with Date and Seal)

---



# Acknowledgements

Five years of my Ph.D. journey was filled with lots of challenges and learning moments. The experience what I am carrying out is incredible and life-changing. It has not only given me a new dimension in life but also has changed the way I approach a problem. This journey could not have been completed without support and love of few people. I would like to acknowledge individually for all what they have done to me.

As with every research scholar, my journey was filled with lots of ups and downs. Few of my close associates were always there to hold me and never let me fall. I thank each one of them, for walking through this passage of life.

My sincere *thanks* to my guide **Dr. Annappa**, Assoc. Prof., Dept. of CSE. Without him, I could not even dream of Ph.D. He encouraged me to pursue my education. His support and guidance are immense. He will always remain my **God Father**. He never let me down and always helped me to be on track. The discussion with him has helped to widen my thinking towards research problem. His enthusiastic nature always accelerated my work. The wide knowledge and generosity have helped me to move to a new level and become a better person. The moments spent with him is incredible and cherishing. I could have achieved this milestone only because of him. Thank you sir, for all the support and always holding my hand.

I want to thank **Dr. Ramesh Kini M** (Assoc. Prof., Dept. of E&C), and **Dr. K Ram Chandar** (Asst. Prof., Dept. of Mining) for being my Research Progress Assessment Committee (RPAC) members. Very few research scholars are fortunate to get good RPAC members and I was one of those. The complete credit goes to my guide. They never gave any excuses even in their busy schedule to share knowledge and attend my seminars. Their comment, observation, understanding, and discussion, encouraged me to think beyond boundaries. Thank you **Dr. Ramesh Kini M** and **Dr. K Ram Chandar**, for all the support and guidance.

I humbly extend my acknowledgement to **Prof. K. Chandrasekaran** (Prof., Dept. of CSE). He is an inspiration for everyone. He carries a personality that provokes us and make us to thinks beyond our reach. His door was always open with a smiling face for me and ready for any discussion. Thank you sir, for being a part of my life and giving me a space in your heart.

I also want to thank **Dr. Mohit P Tahiliani** (Asst. Prof., Dept. of CSE). Though, being from a network background, he was always ready for assisting me. He never sent me back with a reason that he was busy. He was always ready to listen to my research problems and give the solution. There hasn't been an instance, where I came out of his chamber without a solution to my problem. Thank you sir, for sharing space with me and helping me in countless ways to move forward.

I am grateful to **Dr. Chandavarkar** and **Mrs. Soumya Hegde** (Asst. Prof., Dept. of CSE) for their cheering words and thoughts. Though being the faculties, they always treated me as their close associate. Thank you sir and mam, for every single moment shared with me. I also humbly thank my Head of the Department, **Dr. Santhi Thilagam**, (Assoc. Prof., Dept. of CSE) for providing all the necessary facilities and assistance needed for the completion of my Ph.D. I also extend my sincere thank to **Dr. Jeny Rajan**, Secretary (DRPC) for helping me in research related aspects.

I could not have thought of mathematical models without the assistance of two great professors, **Dr. Gnanasekaran** (Asst. Prof., Dept. of Mechanical) and **Dr. Vishwanath K. P** (Asst. Prof., Dept. of MACS). They were ever ready for the discussion in their busy schedule. They helped in building the optimization models. Thank you sir, for all the support and guidance.

For sailing a boat, we need two *oar*, that helps in its rowing. My journey was paddled by two of my close friends, **Manoj** and **Priyanka**. They were there almost all the time. They never let me alone. They helped me to be strong and focused. They shared their food when I was busy researching. With them, I lived a life at NITK. I must thank my institution for introducing them to me. Expressing love for them would not end by words. I just want to thank them for being my lifeguards always.

My research lab was more exciting and fun with many friends around me. **Praveen** my late night lab-mate, **Vishnu** who never fails to bring a smile on my face and always ready to help at any time, **Sachin** a man with lots of positive energy, **Sumith** a lady of discipline, **Bane Rahman** my tutor and senior, along with **Vishal**, **Girish**, **Bhimappa**,

**Khyamling, Manjunath Mulimani, Marimuttu, Raghavan, and Fathima.** Though we all come from different corners of life, we shared fun and love in this research lab. They managed to make me smile whenever I was down. They held my hand through bad times. It was a my pleasure to share the lab with you kind people, you all contributed, big or small, to this work.

I would humbly thank the non-teaching staff of my department. **Mr. Dinesh Kamath, Mrs. Yashawanthi** and **Mr. Vairavanathan** were supportive in ensuring that research-related seminars go well and uninterrupted. I should be thankful to **Ms. Vanitha, Mrs. Seema Shivaram,** and **Mrs. Mohini** for acknowledging me through the academic work and helping me all the time. I can't forget **Mr. Dayanand** for his help regarding academic and office related matters. Thank to you all.

I again thank my guide **Dr. Annappa** for providing me an opportunity to get to know all these people. *Thank you sir.*

My Ph.D. is a dream of my parents **Mr. T C Thomas** and **Mrs. Mary Kutty Thomas.** They sacrificed everything for this. In their old age, they suffered and faced lot of hurdles without informing me, so that I didn't get disturbed. They supported me in every sense. Their prayer on knees is paid off. They lived away from me alone for five years and stood like a pillar. My mother is staying with me for last two months to help me in this thesis writing. My dad went through many health issues but even faced it alone. *Thank you God for giving me such a good parents.*

My Wife **Evy** and Son **Alfy** (my family) always stood my frustrations. Danced with me when I was happy and cried when I was down. More than me, they wanted to see me happy. They supported in every sense to keep me calm. Thank you dear for coming in my life and love you both always. I also want to thank you, my sister, **Lovina** and brother **Dareal,** along with my niece, **Cia** and **Joffy** for all the love and support. My in-laws (**Mr. Mathew P A** and **Mrs. Elsy K M**) were supporting me in every possible way to manage and go through this period of life. *Thank you Papa and Mommy.*

Above all, I want to thank my priests **Fr. Jacob, Fr. Yesu** and **Fr. Jose,** especially for their prayers and blessings. I am sure, without their prayers and blessings I could not have completed this journey. Finally, I must thank God Almighty for being with me always. **Thank you God.**

Place: Surathkal

**Likewin Thomas**

Date: February 9, 2018





# Abstract

From the literature it was studied that, most of the medical error was due to the faulty system/ process, because of which there is a delay in treatment management, leading to complications in later stages. Proper management of healthcare system is necessary to provide good medical care. Medical error due to failure in the healthcare system can be reduced by employing an appropriate clinical decision support system (*CDSS*). *CDSS* helps in identifying the severity of disease, predicting its progression, and recommending required resources for proper management of the disease. In the recent years, the information system is employed in the healthcare system to improve the management of healthcare.

*CDSS* are being used to predict the disease progression and length of stay in the hospital. In our work, a *CDSS* was developed with the help of process mining techniques for providing improved treatment management. Process mining with its ability to build efficient process models was used for discovering this critical treatment path. The critical treatment path is a sequence of clinical and non-clinical activities that are critical. Process mining helps in stream-lining these activities along with the efficient resources for performing those activities. The gallstone disease treatment management is considered as a case study in this work.

Modified Cascade Neural Network (ModCNN) was built upon the architecture of Cascade-Correlation Neural Network (CCNN) and, was trained and tested using the ADaptive LInear NEuron (ADALINE) circuit. In *CDSS* the performance of ModCNN was evaluated and compared with Artificial Neural Network (ANN) and CCNN. *CDSS*, using ModCNN stratified the cases that may need Endoscopic Retrograde Cholangio-Pancreatography (ERCP) as the treatment progresses. Our result shows improvement in accuracy of prediction and reduction in waiting time. ModCNN showed better accuracy of 96.42% for predicting the disease progression when compared with CCNN (93.24%) and ANN (89.65%). *CDSS* developed in this work is aimed at providing better treatment planning to reduce medical error.



# Contents

|  |             |
|--|-------------|
| <b>Abstract</b>  | <b>i</b>    |
| <b>Table of Contents</b>   | <b>ii</b>   |
| <b>List of Figures</b>   | <b>ix</b>   |
| <b>List of Tables</b>  | <b>xiii</b> |
| <b>Abbreviations and Nomenclature</b>                                  | <b>xix</b>  |
| <b>1 Introduction</b>  | <b>1</b>    |
| 1.1 Healthcare System . . . . .  | 1           |
| 1.1.1 Medical error: Its cause and types . . . . .                     | 3           |
| 1.1.2 Process error: <i>A main cause of medical error</i> . . . . .    | 4           |
| 1.2 Process Engineering Approach for Reducing Medical Errors . . . . . | 6           |
| 1.2.1 Deployment of electronic health record . . . . .                 | 6           |
| 1.3 Process Mining . . . . .   | 8           |
| 1.3.1 Basic types and techniques . . . . .                             | 10          |
| 1.3.2 Petri net . . . . .  | 12          |
| 1.3.3 Process mining in healthcare system . . . . .                    | 12          |
| 1.4 Case Study: A Gallstone Disease (GSD) . . . . .                    | 13          |
| 1.4.1 Gallstone disease (GSD) . . . . .                                | 13          |
| 1.5 Proposed System . . . . .  | 14          |
| 1.5.1 System architecture . . . . .                                    | 15          |
| 1.6 Thesis Outline . . . . .   | 15          |
| <b>2 Related Work</b>  | <b>19</b>   |
| 2.1 Complications and Stages of Gallstone Disease . . . . .            | 19          |
| 2.2 Disease Severity Scoring System . . . . .                          | 20          |
| 2.2.1 Intensive Care Unit (ICU) scoring system . . . . .               | 20          |
| 2.3 Meta-Data Analysis . . . . .                                       | 22          |

|          |   |           |
|----------|---|-----------|
| 2.4      | Regression . . . . .  | 23        |
| 2.4.1    | Introduction to regression . . . . .  | 23        |
| 2.4.2    | Regression in epidemiology . . . . .  | 25        |
| 2.4.3    | Significance of regression in gallstone disease . . . . .                       | 26        |
| 2.5      | Artificial Neural Network (ANN) Outperforming Regression . . . . .              | 28        |
| 2.5.1    | Artificial Neural Network . . . . .   | 29        |
| 2.5.2    | Application of ANN . . . . .  | 29        |
| 2.5.3    | ANN in epidemiology . . . . .   | 30        |
| 2.5.4    | ANN for disease management of GSD . . . . .                                     | 31        |
| 2.5.5    | ANN as an expert system for predicting the disease progression . . . . .        | 32        |
| 2.5.6    | Observations and limitations of ANN . . . . .                                   | 32        |
| 2.5.7    | If not ANN then what? . . . . .   | 33        |
| 2.6      | Cascade-Correlation Neural Network (CCNN) . . . . .                             | 33        |
| 2.6.1    | Study on application of CCNN . . . . .  | 34        |
| 2.7      | Bridging Statistical Analysis with Process Mining . . . . .                     | 36        |
| 2.8      | Electronic Health Record Process Mining . . . . .                               | 37        |
| 2.8.1    | Assistance of EHR in Clinical Decision Support System ( <i>CDSS</i> ) . . . . . | 37        |
| 2.8.2    | EHR in healthcare system . . . . .  | 38        |
| 2.8.3    | Adoption of EHR in India . . . . .  | 39        |
| 2.9      | Process Mining in Healthcare . . . . .  | 40        |
| 2.9.1    | Spaghetti-like process model . . . . .  | 41        |
| 2.9.2    | Careflow: A patients journey within the hospital . . . . .                      | 42        |
| 2.10     | Motivation and Contributions . . . . .  | 44        |
| 2.11     | Problem Statement . . . . .   | 45        |
| 2.11.1   | Research objectives . . . . .   | 45        |
| 2.12     | Synergies of process mining in Healthcare . . . . .                             | 46        |
| <b>3</b> | <b>Framework and Study Material</b>   | <b>49</b> |
| 3.1      | Experimental Set-up . . . . .   | 51        |
| 3.1.1    | Description of ModCNN . . . . .   | 52        |
| 3.2      | Electronic Health Record for Healthcare Process Mining . . . . .                | 54        |
| 3.3      | Trace Clustering and Trace Matching . . . . .                                   | 56        |
| 3.4      | Control-Flow: A Causal Relationships . . . . .                                  | 58        |
| 3.5      | Representing and Storing Event Log: Mining eXtensible Markup Language . . . . . | 59        |

|          |  |           |
|----------|--|-----------|
| <b>4</b> | <b>Modified Cascade Neural Network (ModCNN)</b>                              | <b>65</b> |
| 4.1      | Introduction . . . . .   | 66        |
| 4.2      | Artificial Neural Network . . . . .  | 66        |
| 4.2.1    | Flow of information in ANN . . . . .   | 67        |
| 4.3      | Cost Function $J(\theta_0, \theta_1)$ . . . . .                              | 69        |
| 4.3.1    | Cost function for ANN . . . . .  | 70        |
| 4.4      | Cascade Correlation Neural Network (CCNN) . . . . .                          | 70        |
| 4.4.1    | Evolution of CCNN . . . . .  | 71        |
| 4.4.2    | Architecture of CCNN . . . . .   | 72        |
| 4.4.3    | Training phase . . . . .   | 73        |
| 4.5      | Modified Cascade-correlation Neural Network (ModCNN) . . . . .               | 74        |
| 4.5.1    | Architecture of ModCNN . . . . .   | 75        |
| 4.5.2    | ADALINE circuit . . . . .  | 75        |
| 4.5.3    | Least Mean Square Algorithm: (LMS algorithm) . . . . .                       | 77        |
| 4.5.4    | Gradient descent . . . . .   | 79        |
| 4.5.5    | Finding correlation among the hidden units . . . . .                         | 82        |
| 4.5.6    | Training phase . . . . .   | 84        |
| 4.5.7    | Testing phase . . . . .  | 85        |
| 4.5.8    | Master-slave model . . . . .   | 86        |
| 4.6      | Summary . . . . .  | 87        |
| <b>5</b> | <b>Experimental Result</b>   | <b>89</b> |
| 5.1      | Background . . . . .   | 89        |
| 5.2      | Attribute Distribution . . . . .   | 90        |
| 5.2.1    | Chi-squared test $((\chi)^2)$ . . . . .                                      | 90        |
| 5.3      | Testing Relative Risk of each Factor for Different Spectrum of GSD . . . . . | 94        |
| 5.4      | Performance Comparison of ModCNN with ANN and CCNN . . . . .                 | 97        |
| 5.5      | Validation of ModCNN . . . . .   | 99        |
| 5.6      | Testing for Accuracy in Prediction and Detection of Critical Cases . . . . . | 100       |
| 5.7      | Performance Measurement of ANN, CCNN and ModCNN . . . . .                    | 102       |
| 5.8      | Accuracy Measurement Using the Concept of $A_Z$ . . . . .                    | 102       |
| 5.9      | Summary . . . . .  | 102       |

|          |  |            |
|----------|--|------------|
| <b>6</b> | <b>Process Mining in Healthcare System</b>   | <b>109</b> |
| 6.1      | Introduction . . . . .   | 111        |
| 6.1.1    | Electronic Health Record (EHR) . . . . .   | 111        |
| 6.1.2    | EHR process mining . . . . .   | 112        |
| 6.2      | Preliminaries . . . . .  | 113        |
| 6.3      | Construction of Transition System . . . . .  | 117        |
| 6.3.1    | Initial design . . . . .   | 118        |
| 6.3.2    | Current state . . . . .  | 118        |
| 6.3.3    | Abstraction of event log for generating transition system . . . . .  | 120        |
| 6.3.4    | Performance information . . . . .  | 121        |
| 6.3.5    | Activity metric . . . . .  | 121        |
| 6.3.6    | Transition metric . . . . .  | 124        |
| 6.3.7    | Causal metric . . . . .  | 125        |
| 6.3.8    | Construction of annotated transition system for analysing the re-<br>source performance based on TDABC . . . . . | 127        |
| 6.3.9    | Construction of annotated transition system based on remaining<br>turn-around time . . . . .                     | 131        |
| 6.4      | Prediction Function . . . . .  | 135        |
| 6.4.1    | Trace clustering and trace matching using similarity check . . . . .   | 135        |
| 6.4.2    | Identifying the resource load and their performance . . . . .  | 138        |
| 6.4.3    | Identifying critical path activities . . . . .   | 143        |
| 6.4.4    | Steps involved in identifying critical path . . . . .  | 144        |
| 6.5      | Summary . . . . .  | 146        |
| <b>7</b> | <b>Performance Evaluation of Annotated Transition System</b>   | <b>149</b> |
| 7.1      | Length of Stay in the Hospital . . . . .   | 150        |
| 7.1.1    | Statistics . . . . .   | 151        |
| 7.2      | An approach to develop a decision support system for GSD management .  | 153        |
| 7.2.1    | Resource performance . . . . .   | 154        |
| 7.2.2    | Trace execution . . . . .  | 158        |
| 7.3      | Accuracy of prediction . . . . .   | 160        |
| 7.4      | Summary . . . . .  | 163        |
| <b>8</b> | <b>Conclusions</b>   | <b>165</b> |
| 8.1      | Summary of Contribution . . . . .  | 166        |

|   |            |
|---|------------|
| 8.2 Direction for Future Work . . . . . | 167        |
| <b>References</b>                       | <b>169</b> |
| <b>List of Publications</b>             | <b>192</b> |





# List of Figures

|      |  |    |
|------|--|----|
| 1.1  | A Simple illustration of healthcare system. . . . .  | 2  |
| 1.2  | Swiss cheese model illustrating causing of medical error (source (Reason et al., 2001a)). . . . .                            | 3  |
| 1.3  | Block diagram showing the causes and types of medical error (source (Leape et al., 1993)). . . . .                           | 5  |
| 1.4  | Example structure of EHR. . . . .  | 7  |
| 1.5  | Model of Clinical Decision Support System. Source Nair (2007) . . . . .  | 9  |
| 1.6  | Relationship between process mining and BPM. . . . .   | 10 |
| 1.7  | Basic types of process mining techniques. . . . .  | 10 |
| 1.8  | Petri net Model illustrating hospital treatment Process. . . . .   | 11 |
| 1.9  | Framework of proposed clinical decision support system. . . . .  | 14 |
| 1.10 | Design structure of statistical comparator for identifying the efficient model. . . . .                                      | 16 |
|      |  |    |
| 2.1  | Illustration of Logistic Regression . . . . .  | 24 |
| 2.2  | Illustration of ANN . . . . .  | 29 |
| 2.3  | Overview of the main applications of ANN in medicine . . . . .   | 30 |
| 2.4  | Flow of Study . . . . .  | 36 |
| 2.5  | Example spaghetti model discovered from standard process mining event log repository (4TU.Centre for Research Data). . . . . | 43 |
| 2.6  | Lasagne: Simplified model of spaghetti process model shown in Figure 2.5. . . . .  | 44 |
| 2.7  | Roadmap of the Problem approach . . . . .  | 47 |
| 2.8  | Roadmap of the Problem approach . . . . .  | 48 |
|      |  |    |
| 3.1  | Clinical readings showing SYMPTOMS observed through lab investigations in the study cases. . . . .                           | 49 |
| 3.2  | Clinical readings showing SIGNS observed through lab investigations in the study cases. . . . .                              | 50 |

|      |   |    |
|------|---|----|
| 3.3  | Clinical readings showing COMORBID Conditions observed through lab investigations in the study cases. . . . . | 50 |
| 3.4  | Clinical readings showing TESTS conducted on the study cases. . . . .   | 50 |
| 3.5  | USG Findings . . . . .  | 51 |
| 3.6  | Classification result of GSD patients . . . . .   | 51 |
| 3.7  | Statistical analysis of EHR. . . . .  | 52 |
| 3.8  | Experimental setup for analysis of EHR data using ModCNN. . . . .   | 53 |
| 3.9  | Accuracy testing of ANN, CCNN, and ModCNN. . . . .  | 54 |
| 3.10 | Empirical testing of ANN, CCNN, and ModCNN. . . . .   | 54 |
| 3.11 | Illustration of master-slave model. . . . .   | 55 |
| 3.12 | Bar chart showing overall distribution of traces based on the duration. . .                                   | 56 |
| 3.13 | Bar chart showing the cluster of BEST traces. . . . .   | 57 |
| 3.14 | Bar chart showing the cluster of BETTER traces. . . . .   | 57 |
| 3.15 | Bar chart showing the cluster of GOOD traces. . . . .   | 58 |
| 3.16 | Activity position distribution showing the execution of activities. . . . .                                   | 58 |
| 3.17 | Traces at each variants. . . . .  | 59 |
| 3.18 | cluster of variants showing the traces belonging to different variants. . . .                                 | 60 |
| 3.19 | Petri-net model of hospital treatment process. . . . .  | 60 |
| 3.20 | Event log structure in XES format. . . . .  | 64 |
| 4.1  | Neural Network model . . . . .  | 68 |
| 4.2  | Schematic diagram of CCNN architecture . . . . .  | 71 |
| 4.3  | Moving target problem illustration of ANN, compared with CCNN . . . . .                                       | 71 |
| 4.4  | Illustration of frozen and training state in CCNN . . . . .   | 73 |
| 4.5  | Schematic diagram of ModCNN architecture. . . . .   | 76 |
| 4.6  | ModCNN training and frozen states . . . . .   | 76 |
| 4.7  | Architecture of ADALINE . . . . .   | 77 |
| 4.8  | Paraboloid of the cost function . . . . .   | 79 |
| 4.9  | Learning Rate in gradient descent . . . . .   | 80 |
| 4.10 | Learning Rate of gradient descent using contour graph . . . . .   | 80 |
| 4.11 | Illustration of gradient descent . . . . .  | 83 |
| 4.12 | Empirical testing of ANN, CCNN, and ModCNN. . . . .   | 86 |
| 4.13 | Gradient descent identifying the optimal neuron for different hidden units. .                                 | 88 |
| 4.14 | Gradient descent identifying the optimal hidden units using MSE. . . . .                                      | 88 |

|      |   |     |
|------|---|-----|
| 5.1  | Classification result of GSD patients. . . . .  | 91  |
| 5.2  | Clinical readings, showing the feature distribution of 260 patients. . . . .  | 91  |
| 5.3  | Relative strength of treatment effects in different spectrum of GSD. . . . .  | 96  |
| 5.4  | Analysing the disease progression and detecting the critical cases . . . . .  | 97  |
| 5.5  | Performance comparison of ModCNN, CCNN and ANN for classifying spec-<br>trum of GSD . . . . .   | 98  |
| 5.6  | Classification performance of ModCNN, CCNN and ANN. . . . .   | 99  |
| 5.7  | Comparison of ANN, CCNN and ModCNN for different spectrum of GSD .  | 100 |
| 5.8  | Initial Stage: Decrease in MSE showing classification performance . . . . .   | 103 |
| 5.9  | Intermediate Stage: Decrease in MSE showing classification performance .  | 103 |
| 5.10 | Final Stage: Decrease in MSE showing classification performance . . . . .   | 104 |
| 5.11 | Comparison of accuracy of prediction for cholangitis using $A_Z$ . . . . .  | 104 |
| 5.12 | Comparison of accuracy of prediction for pancreatitis using $A_Z$ . . . . .   | 105 |
| 5.13 | Comparison of accuracy of prediction for cholecystitis using $A_Z$ . . . . .  | 105 |
| 5.14 | Comparison of accuracy of prediction for choledocholithiasis using $A_Z$ . .  | 106 |
| 5.15 | Accuracy comparison of ModCNN with ANN and CCNN $A_Z$ . . . . .   | 107 |
| 6.1  | Typical process model of a healthcare system. . . . .   | 110 |
| 6.2  | Example structure of EHR. . . . .   | 112 |
| 6.3  | Illustration of Yerkes-Dodson Law of Arousal. . . . .   | 113 |
| 6.4  | Illustration of design approach for predicting the future behaviour. . . . .  | 117 |
| 6.5  | Illustration of current state to future state transition. . . . .   | 117 |
| 6.6  | Tree structure of process log. . . . .  | 119 |
| 6.7  | Illustration of different process life time. . . . .  | 122 |
| 6.8  | Gantt chart for the trace shown in Table 6.3. . . . .   | 122 |
| 6.9  | Analysis of process behaviour. . . . .  | 124 |
| 6.10 | Position of different activity in the example log $\mathcal{L}$ . . . . .   | 128 |
| 6.11 | Process model with the information of processing time, waiting time, suc-<br>cessor and predecessor for the example log $\mathcal{L}$ . . . . . | 129 |
| 6.12 | Annotated transition system based on log shown in Table 6.10. . . . .   | 134 |
| 6.13 | Activity position distribution showing the execution of activities. . . . .   | 137 |
| 6.14 | Traces at each variants. . . . .  | 138 |
| 6.15 | Cluster of variants showing the traces belonging to different variants. . . .   | 139 |
| 6.16 | Cost function . . . . .   | 142 |

|      |   |     |
|------|---|-----|
| 6.17 | Learning rate of gradient descent using contour graph . . . . .   | 142 |
| 6.18 | Learning rate in gradient descent . . . . .   | 144 |
| 6.19 | Illustration of process time execution. . . . .   | 145 |
| 6.20 | Flow chart showing the chapter summary. . . . .   | 147 |
|      |   |     |
| 7.1  | Frequency distribution of length of stay. . . . .   | 150 |
| 7.2  | Frequency distribution of cholecystitis and choledocholithiasis. . . . .  | 151 |
| 7.3  | Frequency distribution of pancreatitis and cholangitis. . . . .   | 152 |
| 7.4  | Gallstone management for ERCP of cholangitis. . . . .   | 153 |
| 7.5  | Gallstone management for ERCP of pancreatitis. . . . .  | 153 |
| 7.6  | Gallstone management for ERCP of gallstone related jaundice and cholan-<br>gitis. . . . .                               | 153 |
| 7.7  | Process model of current system. . . . .  | 155 |
| 7.8  | Analysing the activities for their processing and waiting time 1. . . . .   | 156 |
| 7.9  | Analysing the activities for their processing and waiting time 2. . . . .   | 156 |
| 7.10 | Time taken by resources for the completion of assigned task. . . . .  | 158 |
| 7.11 | Time taken by resources for the completion of assigned task after applica-<br>tion of TDABC. . . . .                    | 159 |
| 7.12 | Conventional v/s recommended traces. . . . .  | 159 |
| 7.13 | Frequency distribution of cholecystitis and choledocholithiasis after opti-<br>mizing the resource performance. . . . . | 160 |
| 7.14 | Frequency distribution of pancreatitis and cholangitis after optimizing the<br>resource performance. . . . .            | 161 |
| 7.15 | Analysing the activities for their processing and waiting time after opti-<br>mizing the resource performance. . . . .  | 161 |
| 7.16 | ROC showing the accuracy of recommendation. . . . .   | 162 |

# List of Tables

|     |  |     |
|-----|--|-----|
| 2.1 | Different scoring system . . . . .   | 21  |
| 2.2 | Different statistical techniques categorised based on different objectives . .   | 41  |
| 4.1 | Illustration of feature selection and classification . . . . .   | 65  |
| 4.2 | Asymptotic equivalents for ANN, CCNN, and ModCNN . . . . .   | 86  |
| 5.1 | Presentation of different spectrum of GSD in the current study, compared<br>with California study (Glasgow et al., 2000) . . . . .   | 90  |
| 5.2 | Description of the attribute . . . . .   | 92  |
| 5.3 | Observed risk factors for a different spectrum of GSD . . . . .  | 93  |
| 5.4 | Expected frequency calculated for the observed values in Table 5.3 . . . . .   | 93  |
| 5.5 | Result of $(\chi)^2$ . . . . .   | 94  |
| 5.6 | The statistical analysis of current study . . . . .  | 95  |
| 5.7 | Illustration of feature selection and classification . . . . .   | 96  |
| 5.8 | Factors associated with each spectrum of GSD and $A_Z$ of ModCNN. Each<br>factor is parenthesized with its $P$ value. . . . .        | 101 |
| 5.9 | Representation of TP (A), FN (B), FP (C) and TN (D) . . . . .  | 101 |
| 6.1 | Event log of hospital treatment process. . . . .   | 116 |
| 6.2 | Activities in the process model shown in the Figure 6.1 . . . . .  | 119 |
| 6.3 | <i>Initials</i> : Arrival and processing time of $\mathcal{L}_1$ . . . . .   | 122 |
| 6.4 | Arrival Metric of $\mathcal{L}_1$ . . . . .  | 123 |
| 6.5 | <i>Initials</i> : Succeeding and preceding activities $\mathcal{L}_1$ . . . . .  | 123 |
| 6.6 | Causal relationship for all the activities based on turn around time for the<br>event log $\mathcal{L}$ shown in Table 6.1 . . . . . | 127 |
| 6.7 | Causal relationship for all the activities based on waiting time for the event<br>log $\mathcal{L}$ shown in Table 6.1 . . . . .     | 127 |
| 6.8 | Traditional activity based costing . . . . .   | 130 |

|      |  |     |
|------|--|-----|
| 6.9  | Impact of practical capacity . . . . .                                     | 131 |
| 6.10 | Event log of Table 6.1 with the information of turn around time . . . . .  | 132 |
| 6.11 | Load and Service Time . . . . .  | 141 |
| 6.12 | Cost function with $\theta_0$ fixed as 0 . . . . .                         | 142 |
| 6.13 | Cost function with $\theta_1$ & $\theta_2$ . . . . .                       | 143 |
| 7.1  | Length of stay and its cost in the hospital . . . . .                      | 150 |
| 7.2  | Delay in length of stay . . . . .  | 152 |
| 7.3  | Comparison in length of stay before and after applying the proposed system | 152 |
| 7.4  | Representation of TP (A), FN (B), FP (C) and TN (D) . . . . .              | 162 |
| 7.5  | Frequency distribution comparison . . . . .                                | 163 |

# Abbreviations and Nomenclature

## Abbreviations

|                 |  |
|-----------------|--|
| <b>ABC</b>      | Activity Based Costing                         |
| <b>ADALINE</b>  | ADaptive LInear NEuron                         |
| <b>AFM</b>      | Abrasive Flow Machining                        |
| <b>AIIMS</b>    | All India Institute of Medical Science         |
| <b>ANN</b>      | Artificial Neural Network                      |
| <b>AP</b>       | Acute Pancreatitis                             |
| <b>APACHE</b>   | Acute Physiology And Chronic Health Evaluation |
| <b>APACHE-O</b> | Modified APACHE scoring system                 |
| <b>BPM</b>      | Business Process Management                    |
| <b>C-DAC</b>    | Centre for Development of Advanced Computing   |
| <b>CBDS</b>     | Common Bile Duct Stones                        |
| <b>CCNN</b>     | Cascade Correlation Neural Network             |
| <b>CDSS</b>     | Clinical Decision Support System               |
| <b>CECT</b>     | Contrast-Enhanced Computed Tomography          |
| <b>CLC</b>      | Conventional LC                                |
| <b>CT</b>       | Computed Tomography                            |

|                |  |
|----------------|--|
| <b>CV</b>      | Cross-Validated  |
| <b>CoM</b>     | Committee of Machines  |
| <b>DPM</b>     | Dossier Medical Personnel  |
| <b>EBM</b>     | Evidence-Based Medicine  |
| <b>EF</b>      | Earliest Finish time   |
| <b>EHR</b>     | Electronic Health Record   |
| <b>ERCP</b>    | Endoscopic Retrograde Cholangio-Pancreatography                    |
| <b>ES</b>      | Earliest Start time  |
| <b>ET</b>      | Estimated Time   |
| <b>FN</b>      | False Negative   |
| <b>FP</b>      | False Positive   |
| <b>GD</b>      | Gradient Descent   |
| <b>GREPCO</b>  | GRoup for Epidemiology and Prevention of ChOlelithiasis            |
| <b>GSD</b>     | GallStone Disease  |
| <b>HIE</b>     | Health Information Exchange  |
| <b>HITECH</b>  | Health Information Technology for Economic and Clinical Health Act |
| <b>HOMA-IR</b> | HOmeostatic Model Assessment-Insulin Resistance                    |
| <b>ICU</b>     | Intensive Care Unit  |
| <b>ID</b>      | Ideal Time   |
| <b>LC</b>      | Laparoscopic Cholecystectomy                                       |
| <b>LCS</b>     | Longest Common Subsequence   |
| <b>LF</b>      | Latest Finish time   |



|                |  |
|----------------|--|
| <b>LMS</b>     | Least Mean Square  |
| <b>LS</b>      | Latest Start time  |
| <b>MIMO</b>    | Multi-dimensional Input and Output                       |
| <b>MSE</b>     | Mean Squared Error                                       |
| <b>MXML</b>    | Mining eXtensible Markup Language                        |
| <b>ModCNN</b>  | Modified Cascade-correlation Neural Network              |
| <b>NHS</b>     | National Health Service                                  |
| <b>NPA</b>     | Next Probable Activity                                   |
| <b>NPfIT</b>   | National Program for IT                                  |
| <b>OC</b>      | Open Cholecystectomy                                     |
| <b>OEE</b>     | Overall Equipment Effectiveness                          |
| <b>OR</b>      | Overall Runtime  |
| <b>PAIS</b>    | Process Aware Information System                         |
| <b>RFT</b>     | Renal Functional Test                                    |
| <b>RMRS</b>    | Regenstrief Medical Record System                        |
| <b>ROC</b>     | Receiver Operating Characteristic Curve                  |
| <b>SA-MXML</b> | Semantically Annotated Mining eXtensible Markup Language |
| <b>SILC</b>    | Single-Incision LC                                       |
| <b>SOFA</b>    | Sepsis-related Organ Failure Assessment score            |
| <b>TC</b>      | Task Completed   |
| <b>TDABC</b>   | Time Driven Activity Based Costing                       |
| <b>TN</b>      | True Negative  |

|            |                          |
|------------|--------------------------|
| <b>TP</b>  | True Positive            |
| <b>USG</b> | UltraSonoGraphy          |
| <b>WFM</b> | Workflow Management      |
| <b>XES</b> | eXtensible Event Streams |

## Nomenclature

|                      |                                     |
|----------------------|-------------------------------------|
| $\mathcal{P}$        | Probability function                |
| $\frac{1}{1+e^{-z}}$ | Sigmoid function                    |
| $\in$                | Belongs                             |
| $\Delta$             | Change in factor                    |
| $\partial$           | Partial differentiation             |
| $\theta$             | Input weights                       |
| $\sigma$             | Partial trace                       |
| $\cup$               | Union                               |
| $\times$             | Times                               |
| $\subseteq$          | Subset                              |
| $\lambda$            | Learning rate                       |
| $\varepsilon$        | Error                               |
| $\nabla$             | Gradient operator                   |
| $\mu$                | Coefficient of convergence          |
| $\xi$                | MSE                                 |
| $\alpha$             | Learning rate: Small baby steps     |
| $\gamma$             | Gain function                       |
| $\theta^T$           | Transpose of weight vector $\theta$ |
| $(\chi)^2$           | Chi-squared test                    |
| $AN$                 | Set of all attributes               |
| $A_Z$                | Area under ROC                      |

|                         |   |
|-------------------------|---|
| $E$                     | Empirical risk  |
| $H$                     | Hidden units  |
| $I$                     | Input units   |
| $J(\theta_0, \theta_1)$ | Cost Function   |
| $O$                     | Output units  |
| $R$                     | Relation  |
| $S$                     | Sum over all output units $O$   |
| $W$                     | Weight vector   |
| $[V]_{n \times p}^1$    | Weight vector of dimension $n \times p$ between hidden and output units |
| $[W]_{k \times n}^1$    | Weight vector of dimension $k \times n$ between input and hidden units  |
| $h_\theta(x)$           | Predicted output  |
| $y$                     | Desired output  |
| $  $                    | Absolute value  |

# Chapter 1

## Introduction

“ Nearly every patient hospitalized have a life threat at some point of time during their stay.

”

---

Peter Pronovost, *"Safe Patients, Smart Hospitals: How One Doctor's Checklist Can Help Us Change Health Care from the Inside Out"*, (Pronovost and Vohr, 2010)

It is been studied that the one of the common cause of medical error is a faulty healthcare system. This could be overcome or prevented by providing proper attention towards improving the healthcare system and the resources involved in it (Donaldson et al., 2000).

In this chapter, we will introduce the healthcare system, types and causes of medical error, along with their statistics. We will also introduce our approach to reduce this medical error and provide "care-flow" to the patients.

### 1.1 Healthcare System

The focus of hospitals in a comprehensive healthcare system is to streamline their process (Anyanwu et al., 2003) and reduce medical error (Fanjiang et al., 2005). Healthcare process is compiled of series of clinical and non-clinical activities, performed by different resources. Resources are the people assisting in conducting those activities and taking critical clinical decisions. Streamlining processes in a healthcare is needed to provide proper "care-flow" by providing quality healthcare at reduced cost and waiting time. The biggest challenge in the healthcare system is the extraction of knowledge, as they are

associated with highly complex processes. Complexity of the process is due to flexibility in patients movement and dynamic nature of care. Along with that, multidisciplinary make them more complicated to understand and build a model out of it.

*Dynamic* in nature is due to sudden changes in the path followed by patients as-well-as, changes in administration, drugs and treatment procedures. *Multidisciplinary* is because there involves many internal and external departments for the successful execution of the process. Hence, there are many people/ resources involved in the completion of an activity in a process. And each resource has his way of process execution. Above this, the free application developed by the hospital make it more ad-hoc (Lenz et al., 2002). Due to this complexity and ad-hoc nature of healthcare system, it is tough to make them complete automated. Moreover, the entire healthcare process can't be replaced by supercomputers, as they need clinicians expertise. This is because, here we are dealing with the human life, a huge number of diseases and their causes, and multiple ways of managing those conditions, unlike any other research areas. A simple healthcare system is shown in the Figure 1.1 and it can see that, a lot of time goes in waiting. Application of process mining in healthcare process enables better healthcare as it helps in reducing delay and confirms the proper execution of a process.

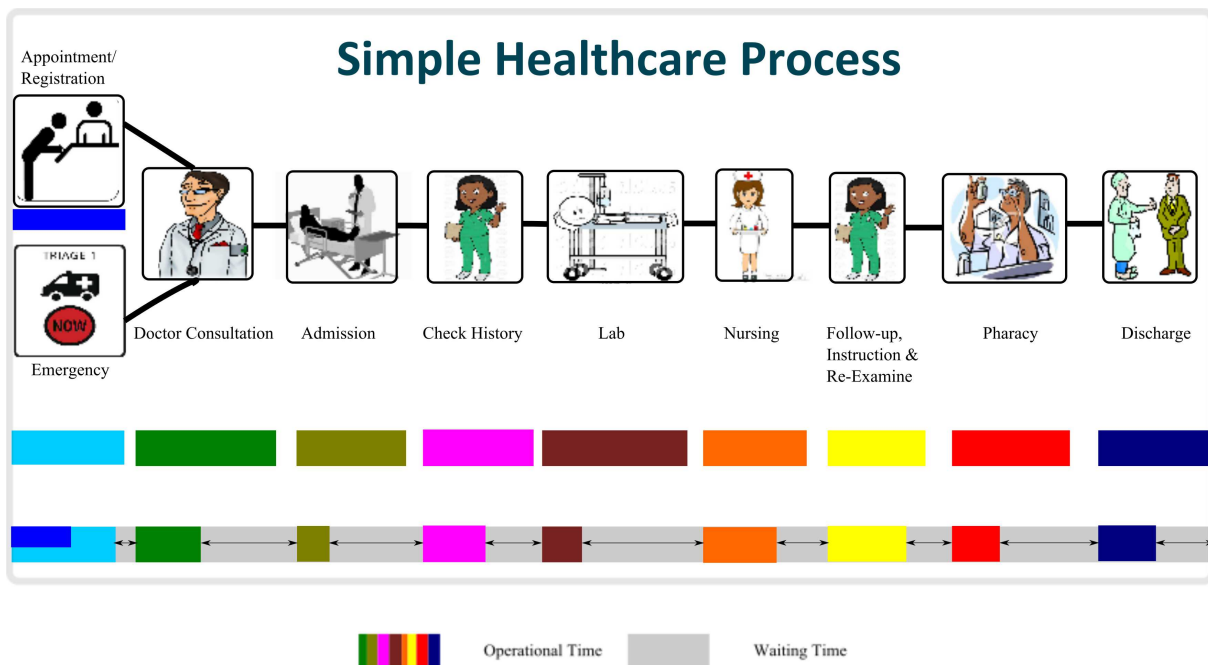


Figure 1.1: A Simple illustration of healthcare system.

### 1.1.1 Medical error: Its cause and types

In the studies conducted during the 1950s regarding patients safety, a medical error was defined as *disease of medical progress* (Moser, 1956). Later in the 1990s three most important studies on medical error: the "Harvard Medical Practice Study" (Leape et al., 1991), the "Quality in Australian Health Study" (Wilson et al., 1999) and the "Utah and Colorado Medical Practice Study" (Thomas et al., 2000), defined it as an *adverse event*. An adverse event is a failure in medical management, causing unintended injury. As a result of this, the patients may suffer from a disability or longer hospital stay, sometimes even both. But later studies showed that the outcome of an adverse event was a subset of medical error. According to Reason et al. (2001b), any hospital is protected by multiple layers of protections. This protection should defend the patients from any adverse events. But, even then there is an error due to flaws in an individual layer of protection as shown in Figure 1.2. Hence, there was a need to understand the processes that cause such errors (Thomas and Brennan, 2001).

From the studies of Donaldson et al. (2000); Hatch (2001); Leape (1994); Fish (2001); Liang and Storti (1999) it was understood that, a safer healthcare system could be built only by properly design of the processes involved in it. Hence, according to Reason (2000); Donaldson et al. (2000) medical error is a *failure in completing the planned action in a pre-defined way or application of an alternative plan (can also be called as the wrong plan)*.

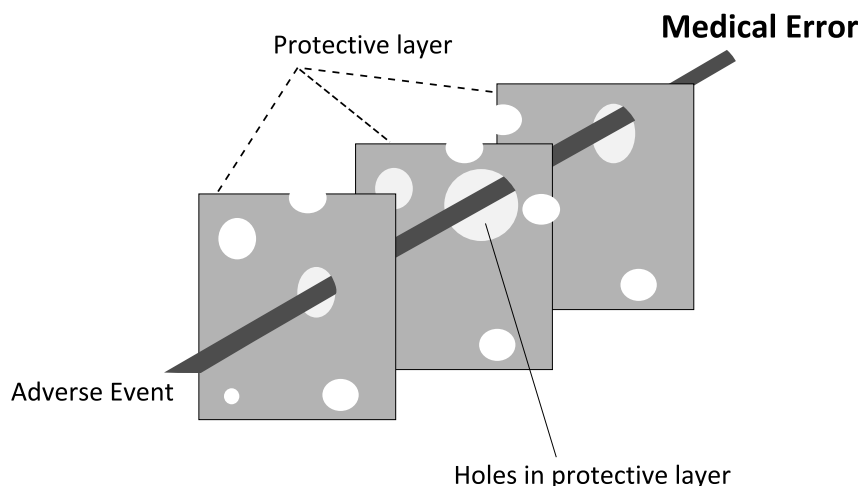


Figure 1.2: Swiss cheese model illustrating causing of medical error (source (Reason et al., 2001a)).

## A Medical error a global issue

Donaldson et al. (2000) conducted a comprehensive study to reduce medical error and

improve patients safety. The study reported that in the US about 98,000 people die every year due to this avoidable medical error, at the rate of at least two errors in ICU every day. The latest study by Leapfrog (2013) estimated the death rate of 4,40,000 annually, making the medical error as the third leading cause of deaths in the US (Tait Shanafelt, 2017). A UK study conducted by Avery et al. (2012) observed 12% of primary care patients were affected by medical error annually. The chance of medical error for 75 and older aged patients was 38% and 30% for the patients receiving five or more drugs. A Swedish study conducted by Soop et al. (2009) observed an adverse event of 12.3% out of which 70% was avoidable. A Saudi Arabian study conducted by Khoja et al. (2011) showed one-fifth of primary care prescription had error. A Mexican study Zavaleta-Bustos et al. (2008) observed 58% of prescription error. In India, a Harvard study conducted by Jha et al. (2013) showed that the medical error was estimated around 5.2 million. Agrawal et al. (2012) conducted a study to evaluate the medication error in general hospitals at Delhi. The study revealed that 8.2% of patients have the risk of being affected by an adverse event.

Unfortunately, these studies did not give much insight to the methodological issues such as failure in treatment planning and management. And from this, we observe that *medical error is a global issue.*

## **B Cause of medical error**

Among these adverse events, half of the errors were due to "surgery" and rest were due to "medication," "diagnostic," and "therapeutic" error. The *medication error* is due to prescribing, dispensing and administering illegal drugs. *Diagnostic error* is frequent non-operative error. The error due to omission and commission were the *therapeutic errors*. The error in *omission* was due to failure in appropriate action such as missed diagnosis (missing any predominant symptoms, neglecting or not identifying some significant parameters and missing to prescribe the correct drugs). The error in *commission* was due to inappropriate action such as administering incorrect drugs (Eldar, 2002). Figure 1.3 shows the types and causes of medical errors.

### **1.1.2 Process error: A main cause of medical error**

Leape et al. (1995); Bates et al. (1995), together re-defined medical error. According to them, healthcare process is a complex chain of events, and medical error is a result of



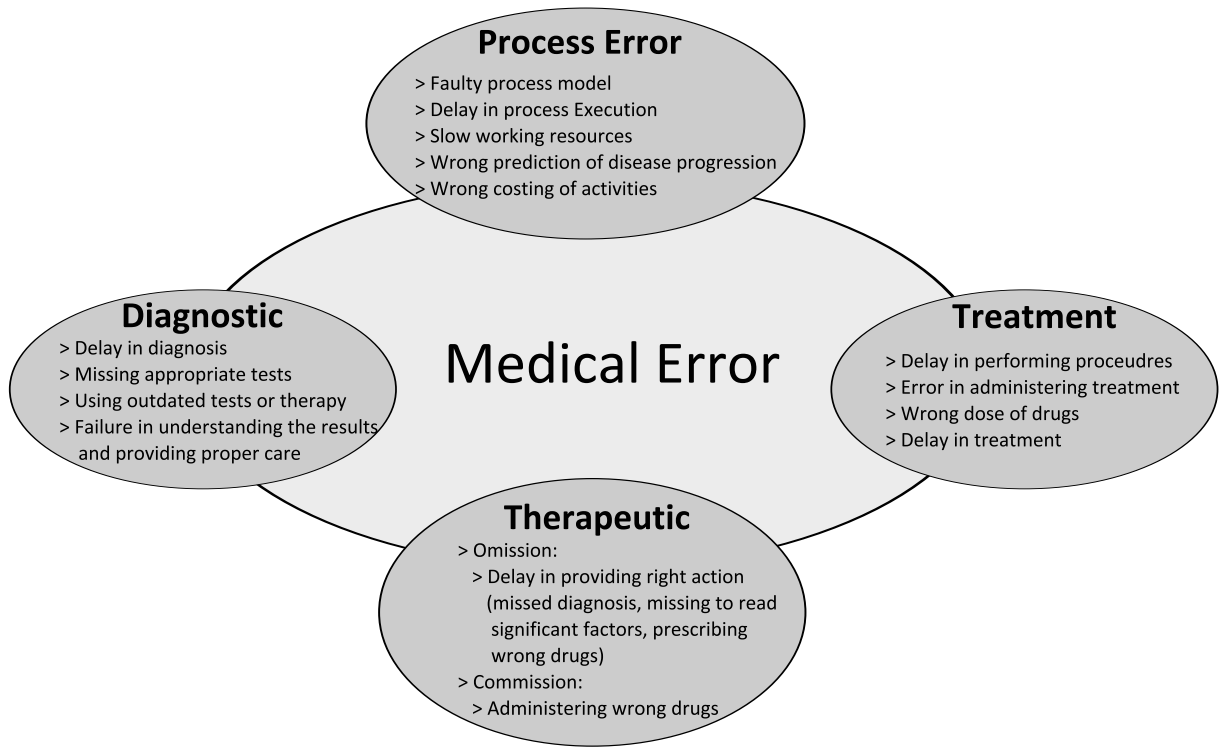


Figure 1.3: Block diagram showing the causes and types of medical error (source (Leape et al., 1993)).

this chain. Stump (2000), with an interdisciplinary team, re-designed healthcare process. Stump (2000) observed that "*A medical error is due to series of events in action, in a faulty process model.*" Further research on cognitive psychology and human factors by Leape (1997) and studies conducted by Fanjiang et al. (2005) suggested that many serious medical errors were due to failure in the process rather by any individual. A survey carried out by Institute of Medicine Donaldson et al. (2000), clearly stated that "*medical error are more commonly caused due to faulty systems, processes and the conditions due to which resources involved commit mistakes or even fail to prevent them*". The process error is the failure in process execution, events and working condition. Medication, therapeutic and diagnostic error were more likely to happen due to process error.

The most disturbing is the absence of relevant knowledge to improve healthcare process and avoid this medical error (Donaldson et al., 2000). The key for improving healthcare process is by extracting knowledge to understand its weakness and vulnerability present in the medication process (Stump, 2000). In the initial research, Leape et al. (1995) showed that the existing healthcare process models are seldom defined only at a higher level. Such model failed to deal with a drifted situation. Moreover, those models were built using traditional process mining techniques and were very complex/ spaghetti and vague, making it harder to understand (Kaymak et al., 2012).

## 1.2 Process Engineering Approach for Reducing Medical Errors

Recently, medical error and its consequences were recorded for statistical analysis. The result of this study has astonished both, the doctors as-well-as a common man (Quaglioni, 2009). As a solution for this, an approach was needed that could identify the possible error in healthcare process and recommend an alternative path of execution. This would provide a smoother and safer execution of treatment process (care-flow) (Mans et al., 2009). Motivated by this, and by the advancement in process mining, we aimed at investigating process engineering research and analysis to assist in reducing the medical error. Thereby, providing proper care-flow to the patients.

The process analysis tools such as Business Process Management (BPM) and Workflow Management (WFM) has assisted in understanding the concept of process engineering. Process engineering monitors and standardizes the behaviour of a business process. This concept was mainly applied to improve the quality of a business process. But, in recent time there is a paradigm shift towards healthcare process. The medical community wanted to build a better healthcare system by reducing the medical error. Hence, they started recording each activity. Such recorded data is known as Electronic Health Record (EHR). Using this EHR in process mining a well-established healthcare system could be built, in-order to assist in taking the critical clinical decision with less medical error (Davidoff et al., 1995).

This awareness of medical error and EHR in medical community was due to clinical pathways (Weiland, 1997) proposed by the principles of Evidence-Based Medicine (EBM) (Davidoff et al., 1995). The clinical pathways are the protocols written by the multidisciplinary team, who timely intervened and examined the healthcare process (Hunter and Segrott, 2008). These protocols aim to avoid any medical negligence in healthcare, assist in taking critical clinical decisions (Blaser et al., 2007) and avoid any unseen errors in clinical treatment (Donaldson et al., 2000).

### 1.2.1 Deployment of electronic health record

With an intention of digitization, deployment of clinical pathways and reducing the medical error in healthcare process, Lary Weed in 1960 introduced the concept of problem oriented medical record. This was named as electronic medical record (EMR)/ (EHR) (Weed, 2017). EHR assists in retrieving the patient-centric records having an information

of treatment plan, medical history and medications of the patients. It also provides an access to clinical data and assist in taking appropriate clinical decisions by streamlining the workflow. The illustration of EHR is shown in Figure 1.4.


| Help                  | Patients Details  |                           |   | Healthcare Service Providers                          |                    |                      |                   |                 |                    |  |
|-----------------------|---|---------------------------|---|---|--------------------|----------------------|-------------------|-----------------|--------------------|--|
| Logout                |  | IME0011                   | Aadhar No:<br>125678943652                                | <b>Name</b>   | <b>Dept.</b>       | <b>Last Visit</b>    | <b>Next Visit</b> |                 |                    |  |
|                       |   | Shwetha                   |   | Laxman, Srinivas                                      | Cardiology         | 01/2006              | 07/2006           |                 |                    |  |
|                       |   | <b>Sex:</b><br>Female     | <b>Phone:</b><br>91-0825-25369                            | Meenaxi, Madhu  | RN                 | 08/2005              | 11/2005           |                 |                    |  |
|                       |   | <b>DOC:</b><br>1940/01/01 | <b>Address:</b><br>19-Kotiabele<br>Mangalore<br>Karnataka | Mohan, Jacob  | Dermatology        | 07/2005              | 12/2005           |                 |                    |  |
|                       |   |                           |   | <b>Medications of Chelecystectomy Done on 05/1981</b> |                    |                      |                   |                 |                    |  |
|                       |   |                           |   | <b>Name</b>   | <b>Dept.</b>       | <b>Activity Name</b> | <b>Start Time</b> | <b>End Time</b> |                    |  |
|                       |   |                           |   | Ramu (green)  | Front office       | Registration         | 01/05/1981        | 01/05/1981      |                    |  |
|                       |   |                           |   |   |                    |                      | 09:00 AM          | 10:25 AM        |                    |  |
|                       |   |                           |   | Mohan (green)   | Front office       | Taken to doctor      | 01/05/1981        | 01/05/1981      |                    |  |
|                       |   |                           |   |   |                    |                      | 10:30 AM          | 10:45 AM        |                    |  |
|                       |   |                           |   | Sarita (blue)   | Nursing            | Preliminary check    | 01/05/1981        | 01/05/1981      |                    |  |
|                       |   |                           |   |   |                    |                      | 11:35 AM          | 12:35 PM        |                    |  |
|                       |   |                           |   | Saxena (red)  | Surgery            | Doc. Consultation    | 01/05/1981        | 01/05/1981      |                    |  |
|                       |   |                           |   |   |                    |                      | 01:00 PM          | 01:55 PM        |                    |  |
|                       |   |                           |   | <b>Encounter History</b>                              |                    |                      |                   |                 |                    |  |
|                       |   |                           |   | <b>Date</b>   | <b>Facility</b>    | <b>Speciality</b>    | <b>Clinicians</b> | <b>Reason</b>   | <b>Type</b>        |  |
|                       |   |                           |   | 02/2006   | GP                 |                      |                   | Hypertension    | -                  |  |
|                       |   |                           |   | 01/2006   | Cardio Assoc.      | Cardiology           | Laxman            | CAD             | OP                 |  |
|                       |   |                           |   | 12/2005   | GP                 |                      |                   | Diabetes        | -                  |  |
|                       |   |                           |   | 10/2005   | General Hosp.      | Dietician            | John              | Diab. Teaching  | OP                 |  |
|                       |   |                           |   | 08/2005   | GP                 |                      |                   | Diabetes        | -                  |  |
|                       |   |                           |   | 08/2005   | GP                 |                      | Rajesh            | Cellulitis      | -                  |  |
|                       |   |                           |   | 08/2005   | Home Visit         | RN                   |                   | Cellulitis      | -                  |  |
|                       |   |                           |   | <b>Immunization</b>                                   |                    |                      |                   |                 |                    |  |
|                       |   |                           |   | <b>Type</b>   | <b>Most Recent</b> | <b>No.</b>           | <b>Type</b>       | <b>Value</b>    | <b>Most Recent</b> |  |
|                       |   |                           |   | Influenza   | 11/2005            | 7                    | A1C               | 0.071           | 12/2005            |  |
|                       |   |                           |   | Preumovax   | 03/2005            | 1                    | LDL               | 2.41            | 12/2005            |  |
|                       |   |                           |   | Twinrix   | 08/2005            | 1                    | BP                | 135/75          | 02/2006            |  |
|                       |   |                           |   | Td  | 04/1996            | 1                    | Microalb          | 0.02            | 04/2006            |  |
|                       |   |                           |   |   |                    |                      | Eye Exam          |                 | 05/2004            |  |
| <b>Patient Record</b> | <b>Alerts</b>   |                           |   |   |                    |                      |                   |                 |                    |  |
| >Summary              | Allergies -Sulfa Drugs  |                           |   |   |                    |                      |                   |                 |                    |  |
| >Lab Result           | > Pap Smear Due   |                           |   |   |                    |                      |                   |                 |                    |  |
| >Diagnostic           | > Tp Due  |                           |   |   |                    |                      |                   |                 |                    |  |
| >Images               | > A1C above target  |                           |   |   |                    |                      |                   |                 |                    |  |
| >Details              |   |                           |   |   |                    |                      |                   |                 |                    |  |
| >Notes or Comment     |   |                           |   |   |                    |                      |                   |                 |                    |  |
|                       | <b>Diagnosis</b>  | <b>State</b>              | <b>Status</b>   |   |                    |                      |                   |                 |                    |  |
|                       | Hypertension  | 11/1989                   | Ongoing   |   |                    |                      |                   |                 |                    |  |
|                       | Diabetes  | 05/1996                   | Ongoing   |   |                    |                      |                   |                 |                    |  |
|                       | Coronary  |                           |   |   |                    |                      |                   |                 |                    |  |
|                       | Artery Diabetes   | 02/2002                   | Ongoing   |   |                    |                      |                   |                 |                    |  |
|                       | Fasting lipids  | 12/2005                   | Resolved  |   |                    |                      |                   |                 |                    |  |
|                       | Chelecystectomy   | 05/1981                   | Resolved  |   |                    |                      |                   |                 |                    |  |

Figure 1.4: Example structure of EHR.

In 1972, Dr. McDonald came up with an idea of an advanced medical recording system known as Regenstrief Medical Record System (RMRS), which was not encouraged by many physicians (McDonald, 1972). In 1991 Institute of Medicine in the US recommended the use of EHR in all the hospitals by 2000, due to the increasing prevalence of medical errors (Weed, 2017). This forced all the hospitals to record the clinical data, and since then there is a substantial growth in clinical data digitization.

As an impact of Health Information Technology for Economic and Clinical Health (HITECH) Act 2009, in the US it became mandatory to maintain EHR (Almasalha et al., 2013). The HITECH was highly concerned about improving the patients care by reducing medical errors. In early 2002, National Program for IT (NPfIT) proposed a nation wide EHR project in the UK. The aim was to have an EHR system within four years. This was very difficult for National Health Service (NHS) and EHR vendors. The government spent 12.7 billion pounds on this project, but it was incomplete even after nine years

(Bowers, 2013). As a result, they failed in building a healthcare IT domain that could connect everyone. France is one of the nations having best healthcare system and are currently in the same stage of the US in implementing EHR (Blackstone and Taylor, 2012). In 1998, Carte Vitale launched and proposed computer based medical information system, and this was extended in 2004 as Dossier Medical Personnel (DPM). The goal was to improve the overall quality of care. In 2011 Ministry of Health, made it compulsory to have "DPM-compatible" EHR system (Stone, 2014). At present, EHR adoption rate is 67% when compared to 69% of the USA. In 2011, Government of India had initiated implementation of EHR. The Center for Development of Advanced Computing (C-DAC) is responsible for building a comprehensive EHR system (Stone, 2014). But, the challenge is in providing better security and privacy. This is because, in October 2013, 90,000 patients information was breached in the USA (Stone, 2014). India having even bigger population than the USA this would be a major concern while building an EHR system.

In 2009, it was seen that 73% of EHR deployed healthcare system are not correctly using it (Renner, 2009), and in 2013 the statistics remain same. With such a little advancement in deploying EHR in a healthcare system, implementing a Clinical Decision Support System *CDSS* was a challenge. *CDSS* is needed for assisting clinicians in taking appropriate clinical decisions and recommending a treatment pathway for the healthcare process (Kong et al., 2008). In this work, we installed an information system that collected the treatment related data along with the patient's journey in the hospital. Information collected was converted into EHR format for further analysis in process mining. Thus, we developed a *CDSS* for assisting clinicians in taking clinical decisions as-well-as in recommending an alternative path of treatment for the critical cases using process mining techniques. The *CDSS* model is shown in Figure 1.5. It is an information system, which extracts the knowledge from the clinical inputs by running the computer-based algorithms and provides an inference from the knowledge extracted to assist the clinicians in taking critical decisions.

### 1.3 Process Mining

The recent advancement in the field of process mining has been able to stratify and standardize the treatment process. This improvement is due to an evolution of Process Aware Information System (PAIS), which extract knowledge from EHR data and build an information model (Ma, 2007). Developing such an information model for a structured

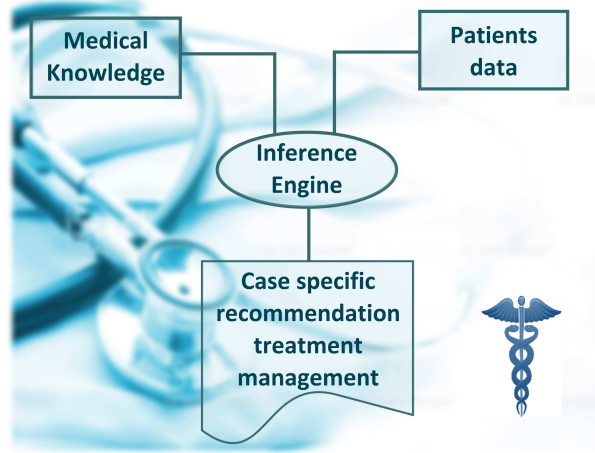


Figure 1.5: Model of Clinical Decision Support System. Source Nair (2007)

non-linear representation of clinical data is a challenge. This problem is due to the uncertainty of clinical data and complexity of diagnostic information. Healthcare process employed with such an information system is essential to have a low-cost service, to meet the clinician's demands, reduce the waiting time of patients, and provide better process transparency.

Process mining is a technique for managing the executable processes by analyzing the data recorded. With the help of various machine learning techniques, it *discovers* what is happening inside the process, does the *conformance* for the executed path by comparing with the actual path of execution and *enhances* the existing model by recommending the new way of execution. Discovery plays a vital role in extracting knowledge from the event log and building a process model (Van Der Aalst et al., 2007).

Process mining provides a detailed insight into the process execution using EHR data. It bridges the gap between the process-oriented nature of BPM and the data-oriented nature of machine learning/ data mining. It is a research discipline that discovers, monitors, and improves the real processes (not the assumed processes) by extracting knowledge from the EHR (Van der Aalst, 2011). The advantage of using process mining is two fold.

- It offers information on how processes are to be carried out in the real-world environment.
- And, it offers the possibility to compare the actual behavior of the model with discovered one. The expected behavior is usually defined either in a formal way (by defining a formal process model) or in an informal way. By this comparison, deviation from the intended behavior could be analyzed and can be used to improve the process execution.

Process mining, in short, is a reverse engineering of BPM and is shown in Figure 1.6. BPM usually starts with high-level process design followed by configuration and implementation phases. But, in case of process mining, the behavior observed by the information system is utilized to discover the real process model.

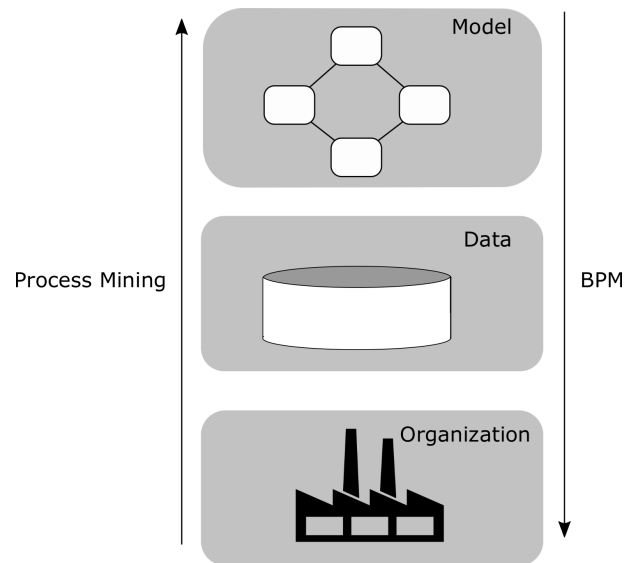


Figure 1.6: Relationship between process mining and BPM.

### 1.3.1 Basic types and techniques

The underlying architecture of process mining is shown in Figure 1.7. The figure shows three most important dimensions of process mining techniques: discovery, conformance, and enhancement.

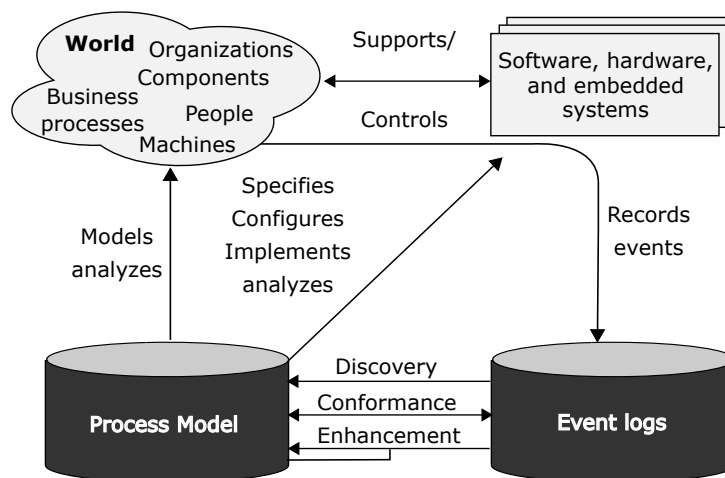


Figure 1.7: Basic types of process mining techniques.

## A *Discovery*

In process mining, event log is read and analyzed to *discovers* the process models. These models give the best description about the behavior observed in the event log. Process discovery methods are used to provide insights into what occurs in reality. Discovery techniques produce control-flow, data, organizational, time, and case models. Process mining has a huge number of defined and tested methods to discover models of various notations. As an illustration of healthcare process, we developed a simple hospital treatment process model, shown in Figure 1.8. This model is generated using  $\alpha$  process discovery algorithm (Van der Aalst et al., 2004).

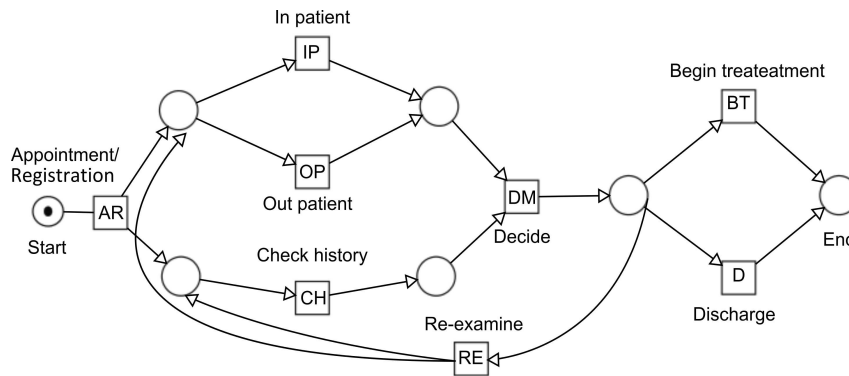


Figure 1.8: Petri net Model illustrating hospital treatment Process.

## B *Conformance*

In *conformance* checking the discovered process model is analyzed with the event log of same process. It measures and quantifies, how closely a given process model conforms to the reality and evaluates the quality of a discovered model. Simplicity, precision, fitness, and generalization are the dimensions used for measuring the conformance.

## C *Enhancement*

*Enhancement* improves the discovered process model to make it more informative. With the information extracted from event log, process models can be enhanced further to accommodate the changing requirement. For example, a control-flow model can be enhanced and made more readable by overlaying the additional information such as time-stamps, bottlenecks, service levels, throughput times, frequencies, resources, decision rules quality metrics, etc.

### 1.3.2 Petri net

$\alpha$  algorithm defined by Van der Aalst et al. (2004) and available in process mining can generate control flow of a process in the petri net notation. Petri net model of hospital admission process is given in Figure 1.8. A petri net is a triplet  $N = (P, T, F)$  where,

- $P$  is a set of places.
- $T$  is a set of transitions such that  $P \cap T = \emptyset$
- $F \subseteq (P \times T) \cup (T \times P)$  is a set of directed arcs called flow relations.

Petri net consists of places and transitions. The structure of petri net model is static, but it is controlled by firing rule. Distribution of tokens over network is referred as marking and it determines the state of petri net.

### 1.3.3 Process mining in healthcare system

A process in healthcare system is a sequence of activities recorded during the treatment procedure and medication of patients. EHR records both the clinical and administration (non-clinical) activities. Process mining discovers healthcare process models using EHR and is known as EHR process mining. The objectives of healthcare system could be met by building an efficient process model and by improving the resource efficiency.

Process mining is relatively young research discipline known for its existence more than a decade (Van der Aalst, 2011). Since then a lot of investigations have been conducted on its application, and it has now become mature enough to be applied on any type and complex process. This was initially developed with an intention of assisting the business management for taking critical business decisions and was successful. This success was because, the processes involved in business execution were recorded as event logs. Event log is a set of traces containing sequence of process performed for a successful completion of a particular process instance.

EHR process mining monitors and diagnose the workflow deadlocks, errant process, and ideal and idle resources. It was observed from the study of Schuld et al. (2011) that, by applying process mining techniques on EHR, performance of healthcare staff/ resources could be improved and it also enhances the process efficiency. Such a well-designed healthcare process not only employs a clinical pathway efficiently but also, improves the diagnosis and treatment options to the patients, providing better care.



## 1.4 Case Study: A Gallstone Disease (GSD)

GSD once known as the western disease, is showing high prevalence in India since last one decade. So, GSD was considered as the case study. This is a retrospective analysis of 260 complicated cases of GSD from the tertiary care center in North Malabar, Kerala, India, from 2014 to 2015. In our experiment, by the help of machine learning techniques 49% of complicated cases were classified and rest were uncomplicated.

Progression of chronic disease is usually slower when compared to that of acute disease. For example chronic obstructive pulmonary disease may take more than ten years to progress from stage I (*mild*) to stage IV (*very severe*) (Pauwels et al., 2001). It is same for congestive heart failure (Wang et al., 2014). The patients suffering from chronic pancreatitis have all the possibility to have the episodes of Acute Pancreatitis (AP). Similarly, chronic GSD causes a scar on the gall bladder, making it rigid and giving a lot of abdominal pain. Hence determining the progression of the disease having the episodes of chronic as-well-as acute is a challenge. In this study, the proposed system predict the disease progression, and for the cases that may become critical, it recommends a safer treatment path known as *critical path*.

### 1.4.1 Gallstone disease (GSD)

Gallstone disease is a heterogeneous disease (Cetta et al., 1995) and the most common biliary pathology. After the appearance of Laparoscopic Cholecystectomy (LC) in the late 1980s, the incidence of gall bladder surgery has raised all over the world (Steiner et al., 1994). Due to its unpredictability in progressive organ failure, the mortality rate has been observed from one-third to one-half during the first week of diagnosis. The previous study conducted by Hong et al. (2013) showed that most of the death occurred after admission was due to local complication such as pancreatic necrosis, with the symptoms of sepsis and multi organ failure. GSD is also studied to be a significant risk factor for gall bladder cancer (Kapoor, 2006). An early detection and timely management of GSD would prevent the progression towards an adverse complication.

Thus, there is a need for an optimal classification technique for identifying the spectrum of GSD and significant factors/ predictors associated with each class. These significant factors help in predicting the disease progression from which any unseen complications could be avoided. This would assist the physicians to have close surveillance and provide alternative treatments. Significant factors also known as risk factors are identified

from the features detected by lab investigations and observations.

## 1.5 Proposed System

Here we aimed in devising a mandated process model to be followed in a healthcare system. Especially, for the cases which are identified as at high risk. The healthcare process model would reduce the waiting time between the activities and employ an efficient resource for the completion of a task. In this work, we in assistance of medical experts re-defined the healthcare process using technique of EHR process mining. The model not only captured the standard path of execution but also predicted the exception cases that may happen. Those exception cases were known as critical cases, initially treated at triage unit and needed special attention.

Emergency department in-order to handle such cases would want to reduce the waiting time and provide a hassle free care-flow known as critical treatment path. This could be achieved by identifying the bottlenecks and improving the resource utilization. Paths assisted by adequate resources for successful completion of events were chained in healthcare process making it a critical treatment path. The proposed system is shown in Figure 1.9. This works in two phase, in the *first phase*, disease progression is predicted using statistical tools, and in the *second phase*, safest care-flow is recommended. The error in prediction and recommendation were analyzed to make the model more optimal, thus significantly decreasing the space for medical error.

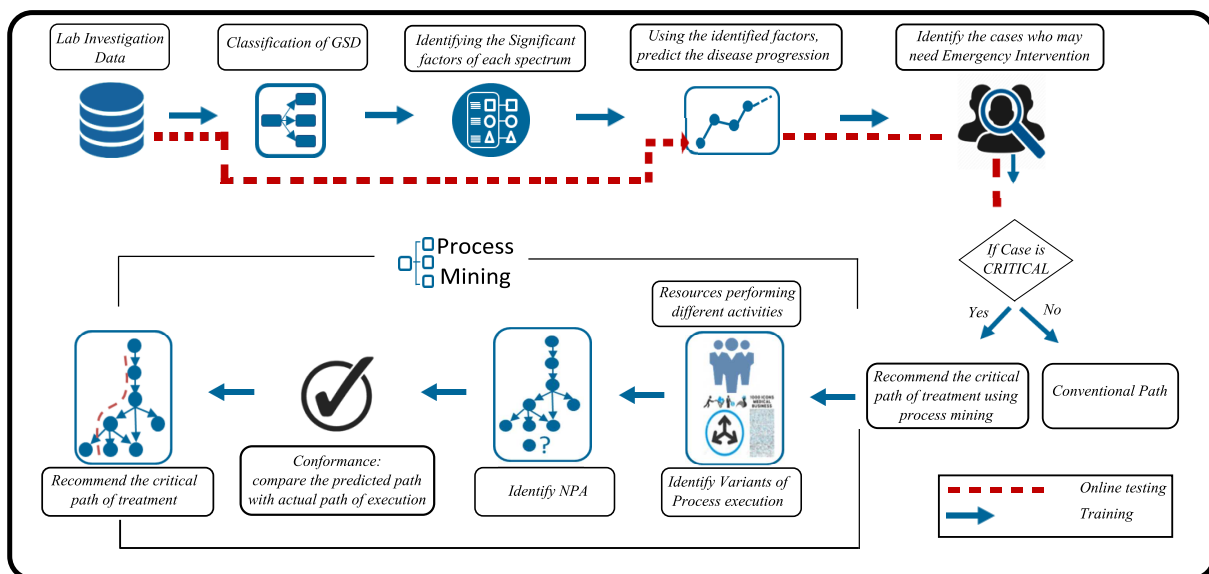


Figure 1.9: Framework of proposed clinical decision support system.

### 1.5.1 System architecture

For accurately predicting the disease progression, we Modified the architecture of Cascade-correlation Neural Network (ModCNN) and compared its performance with Artificial Neural Network (ANN) and Cascade Correlation Neural Network (CCNN). From the literature, we understood that ANN and CCNN were well-established models for conducting the statistical analysis in clinical field. These statistical tools identified the significant factors associated with the disease. It is vital to find right significant factors as they would be fed into the system to predict the disease progression. Hence, the accuracy of prediction purely depends on this identified significant factors. This prediction would help in finding the cases that are critical. A case is known as critical if they need emergency intervention or may require in near future hours. A medical error can happen when clinicians fail to notice this and plan a wrong treatment management. On identifying the critical case, we try to provide the quickest and safest care-flow. Hence, an early detection and management of GSD will prevent the progression towards an adverse complication.

The complete system function for predicting the disease progression is shown in Figure 1.10. Here in the figure, data collected from the retrospective study is fed into Committee of Machines (CoM). ANN, CCNN, and ModCNN are optimized and included in CoM. Each statistical tools find their significant factors, which are fed back to the system to predict the disease progression eliminating all those features, which are not significant. The accuracy of prediction is measured using the concept of  $A_Z$ . Thus the system first finds the best suitable tool and then using that predicts progression of the disease.

## 1.6 Thesis Outline

This section provides the brief overview about the structure of this thesis.

- **Chapter 2:** This chapter provides the insight study for finding different statistical techniques along with the scoring system for predicting the disease progression. The study was also conducted to find the process mining application in the field of healthcare system.
- **Chapter 3:** Here, the framework of study material is detailed. The lab investigations conducted were used to identify the significant factors and then predict the progression. The patient's journey in the hospital, recorded by EHR system is used to provide better treatment management using process mining.

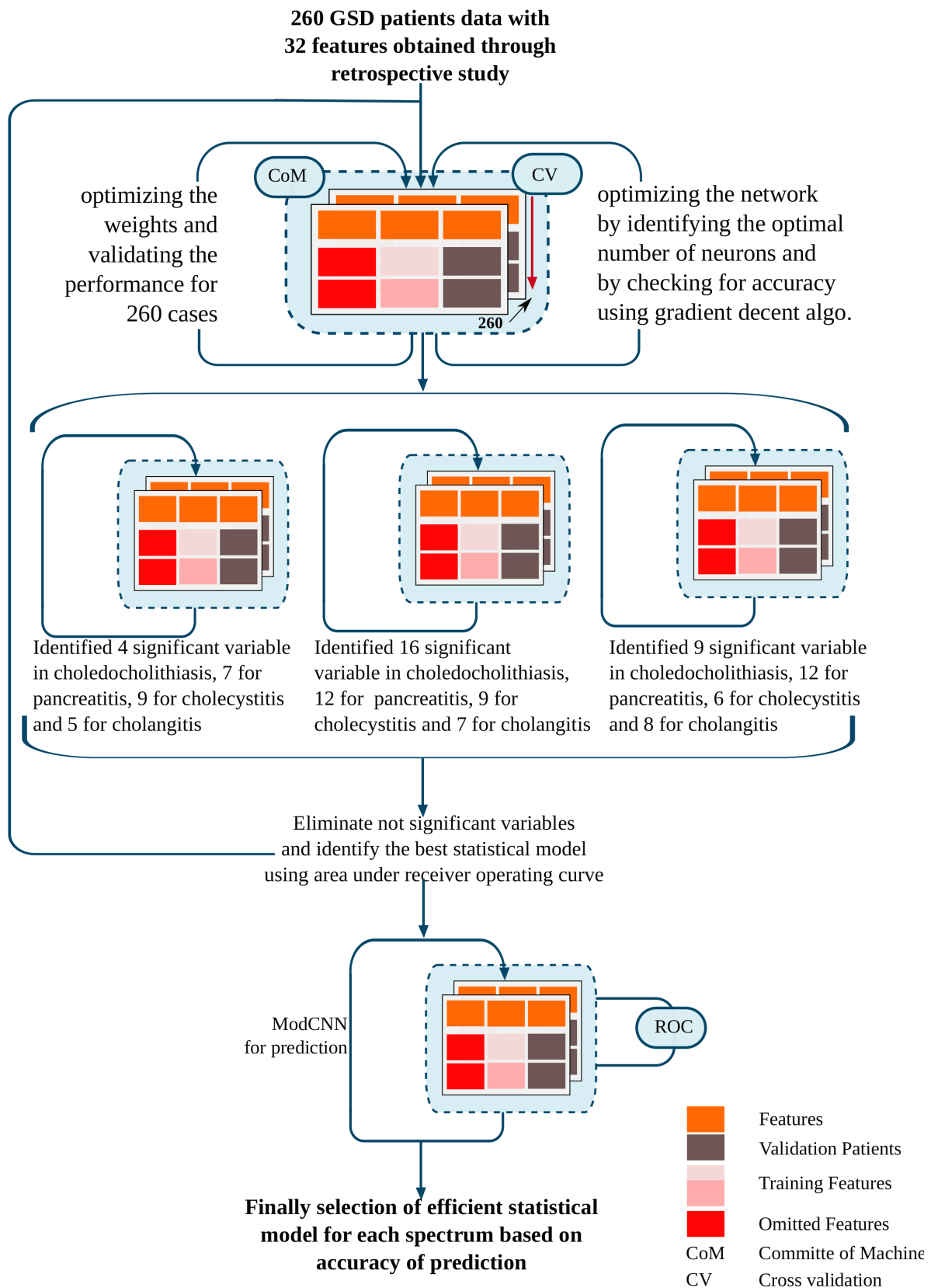


Figure 1.10: Design structure of statistical comparator for identifying the efficient model.

- **Chapter 4:** Here the architecture of proposed ModCNN is explained using the concept of ADALINE circuit and gradient descent algorithm.

- **Chapter 5:** The ModCNN is tested, and the result is presented in this chapter. Here the model is compared for its accuracy in prediction of critical cases.
- **Chapter 6:** In this chapter, the annotation system is developed for predicting the future state of the partially executed trace. Annotated system used the application of process mining for achieving this. Here, the critical treatment path along with an efficient resource for conducting the activity streamlined along the treatment path is recommended. This would decrease the patient's journey in the hospital.
- **Chapter 7:** The result of recommendations made by the process mining application is presented in this chapter.
- **Chapter 8:** This chapter summarizes the total work conducted along with the future direction of research.



# Chapter 2

## Related Work

This chapter details the literature about an application of process mining in healthcare management. Along with this, an analysis of different statistical techniques for predicting the disease progression is made. As an outcome of this analysis, we could find and understand the performance of well-established statistical tools and techniques, along with process mining for managing the critical cases.

### 2.1 Complications and Stages of Gallstone Disease

GSD is studied to be a significant risk factor for gall bladder cancer (Hundal and Shaffer, 2014). It is a heterogeneous disease and the most common biliary pathology (Cetta et al., 1995). The process of gallstone formation is referred to as *cholelithiasis*. It is a slow process and usually doesn't show any pain or other symptoms. 10% of patients with *cholelithiasis* passes the stone into common bile duct resulting in a condition called as *choledocholithiasis* (Almadi et al., 2012). In few conditions, these stones are likely to cause infection leading to *cholangitis*. Between 1-3% of people with symptomatic gallstones develop an inflammation in the gallbladder accounting into *cholecystitis* (Times, 2008). This occurs when stones or sludge block the duct. *Pancreatitis* is a process of inflammation of the pancreas. 80% of cases are mild with interstitial edema which normally recovers within weeks (Beger and Rau, 2007). But, 15%-20% of cases become severe by systemic or local complications leading to severe morbidity and even death (Beger and Rau, 2007). Early death within the first week is seen due to multiple organ dysfunctions. The mortality rate has been observed from one-third to one-half during the first week of diagnosis due to unpredictability in progressive organ failure of the pancreas (Johnson and Abu-Hilal, 2004). Hong et al. (2013) showed that most of the death occurred after admission was due

to a local complication with the symptoms of multi organ failure. Late mortality is usually a consequence of organ dysfunction and local or systemic infections, including infected pancreatic necrosis (Blum et al., 2001). Thus, an early detection and timely management of GSD would prevent the disease progression towards an adverse complication in later stages.

From the study we could observe that GSD leads to a serious complication if neglected and may cause death. The first week of admission is considered as the high-risk period. But, as the disease progresses there are later stage complications. If not properly treated, it may grow to become cancer. Due to its heterogeneity, predicting its progression was highly difficult and challenging. Hence there was a need for a technique that could categorize the study material, identify the risk factors and predict the disease progression using the identified risk factors.

## 2.2 Disease Severity Scoring System

Initial clinical assessment is inadequate for classifying the disease severity (McMahon et al., 1980). For that, an algorithm of action is required from an interdisciplinary team. Accurate prediction of the disease severity at the time of admission is vital as GSD has a high risk during its first week of admission. During recent years several scoring systems were applied for classifying the disease severity.

### 2.2.1 Intensive Care Unit (ICU) scoring system

ICU Scoring system can be categorized based on the *clinical criteria* and *radiologic features at computerized tomography (CT)*.

#### A *Clinical criteria*

The two clinical criteria scoring systems are Ranson and APACHE (Acute Physiology And Chronic Health Evaluation). *Ranson scoring system* is the first meta-analysis model developed by Ranson et al. (1974). In the current study, we identified eight significant factors for predicting the morbidity and mortality. Out of that, five were identified at the time of admission and three based on the treatment response in first 48 hours. The major shortcoming was, we had to wait for 48 hours after admission of the patient for finding the significant factors. Imrie et al. (1978) proposed *Glasgow scoring system* but, even this had the same shortcoming as of Ranson.



Knaus et al. (1985) introduced the *APACHE II scoring system*. It classified the severity of adult patients admitted to ICU. Here, the severity was calculated using age, chronic health condition and the score calculated for acute physiological measurements. In the Atlanta symposium conducted in 1992, it was awarded as the best scoring system, that could assess within 24 hours of admission. It identified obesity as the major risk factor for mortality and development of severe AP. Hence forth, it was named as modified APACHE scoring system (APACHE-O). But to our knowledge, we could not find any major prospective study using this scoring system.

## **B Radiologic features at CT**

The radiologic features at CT is a Balthazar scoring system proposed by Balthazar et al. (1990). This is the currently used scoring system and was developed in 1985. It classifies the patients into five classes: (*class A, class B, class C, class D and class E*), based on the severity. But even this scoring system had the same disadvantage of being able to complete its evaluation only after 48 hours of admission.

Table 2.1: Different scoring system

| <b>Scoring System</b> | <b>Author and Year of Introduction</b> | <b>Significant Factors Identified</b>              | <b>Comment</b>                                       |
|-----------------------|--|--|--|
| Ranson                | Ranson et al. (1974)                   | 08   | Complete assessment only after 48 hours of admission |
| Glasgow               | Imrie et al. (1978)                    | 05   | Complete assessment only after 48 hours of admission |
| Balthazar             | Balthazar et al. (1990)                | Classify the severity into five groups of severity | Not well implemented                                 |
| APACHE II             | Knaus et al. (1985)                    | 11   | Widely used  |
| Marshall              | Marshall et al. (1995)                 | 5  | Widely used based on organ failure                   |
| SOFA                  | Vincent et al. (1996)                  | 5  | Widely used based on organ failure                   |

*Marshall score* and *Sepsis-related Organ Failure Assessment (SOFA) score* were proposed in 1995 and 1996 respectively (Marshall et al., 1995; Vincent et al., 1996). The idea of SOFA and Marshall was not to predict the disease progression but to sequence the complications. As observed by Andersson (2010), there are several scoring systems, but they are not widely used for assisting in identifying the significant factors, at the time of admission. The generalized performance of the different scoring system is summarized in the Table 2.1.

The following are the challenges observed in the existing scoring systems.

- There exist many hidden covariates involved in the progression of the disease. The scoring systems failed to identify those hidden covariates.
- Identifying the chronic and acute cases were a real challenge.
- Due to the incomplete patient record regarding the disease progression, it was hard to build and train a model in a directed way.
- Due to abnormal check-ups, it was hard to identify the continuity in the disease progression. This caused irregularity in record maintenance because of the inconsistent data.

## 2.3 Meta-Data Analysis

Meta analysis is a statistical model for combining the data from multiple studies. It is used for analysing the existence of a common pattern in the treatment outcome. These identified patterns are found to be consistent with one study to other. It also determines the parameters responsible for variation in the treatment response pattern. According to Plackett (1958) the invention of meta-analysis was made in 17<sup>th</sup> century for the studies related to astronomy. But, the first medical application was made by Simpson and Pearson (1904) where they collected data from typhoid affected patients and observed its outcome using meta-analysis.

In several papers presented by GREPCO (Group for Epidemiology and Prevention of Cholelithiasis), it is observed that meta-analysis is used as a statistical technique. GREPCO is a cross-sectional study conducted in Rome, Italy. For GSD, we found its application in identifying the risk factors associated with gallbladder cancer by Larsson

and Wolk (2007). Clayton et al. (2006) used it for finding the better therapeutic approach among endoscopic and surgical interventions. With the help of meta-analysis Gurusamy et al. (2010) was able to determine the effect of early and delayed LC for acute cholecystitis and find a better approach among them. Geng et al. (2013) compared the procedure of Single-Incision LC (SILC) with Conventional LC (CLC) and identified the better approach among them using meta-analysis. Zhang et al. (2008) discovered the prognostic factors of race and tumour size in carcinosarcoma of gallbladder using meta-analysis.

*On studying the different application of meta-analysis, we could observe that:*

- With the limited study of meta-analysis application on GSD, we observed that they were applied for determining the risk factors associated with the disease. Using these risk factors, the prognosis could be conducted.
- But, it failed in accurately predicting the expected results for a single large study (LeLorier et al., 1997).
- This failure is due to its source of bias, on which the model failed to have control (Slavin, 1986).
- Hence, we could find the limited successful application of meta-analysis in identifying the factors associated with GSD.

## 2.4 Regression

Due to the limitation of a meta-analysis of being non-operable on a single large experiment, researchers started statistical analysis using regression. Regression application in the epidemiology of GSD is relatively recent.

### 2.4.1 Introduction to regression

Regression relates the probability of an event to the regressor variables. *Let,  $x$  be the factor of disease progression, and  $y$  be any complications due to the disease. The *linear relationship* is a slope connecting  $x$  and  $y$  and is defined as "a unit change in  $x$  on  $y$ ".* The challenge here is to estimate and calculate the correct slope, i.e., the effect of  $y$  for a unit change in  $x$ . The Figure 2.1 shows the use of regression model for predicting the progression of lung cancer mortality ( $y$ ) due to the consumption of cigarettes ( $x$ ).

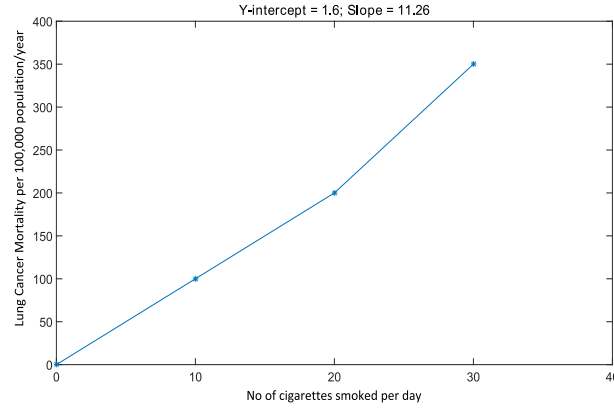


Figure 2.1: Illustration of Logistic Regression

One of the objective of multivariate analysis in epidemiology is to identify the risk factors. On finding the risk factors, its presence and absence in test data would assist in predicting the disease progression. So that, the complexity of the patient's health condition is understood with more clarity, based on which an appropriate treatment could be initiated. Risk factors associated with the disease is computed using the equation 2.1 where  $\Delta$  (Delta) is "change in" slope.

$$\beta_{RiskFactor} = \frac{\text{Change in Disease Progression}}{\text{Change in Risk Factors}} = \frac{\Delta \text{DiseaseProgression}}{\Delta \text{RiskFactors}} \quad (2.1)$$

The equation 2.1 could be re-written as a straight line equation 2.2 where  $\beta_0$  is an *intercept* for the straight line and  $\beta_{RiskFactors}$  is the *slope* found in equation 2.1.

$$\text{Disease Progression} = \beta_0 + \beta_{RiskFactors} \times \text{RiskFactors} \quad (2.2)$$

To make the equation more generalized, we need to consider *other factors* other than the determined risk factors. The updated equation is shown in 2.3.

$$\text{Disease Progression} = \beta_0 + \beta_{RiskFactors} \times \text{RiskFactors} + \text{other factors} \quad (2.3)$$

Then the generalized regression equation using equation 2.3 could be written as shown in equation 2.4, for each distinct cases with  $i = 1, 2, \dots, n$ .

$$y_i = \beta_0 + \beta_i x_i + \Delta u_i \quad (2.4)$$

In the equation 2.4, for  $n$  distinct cases:

- $y_i$  is the result of the treatment (progression of the disease) of  $i^{th}$  case
- $x_i$  is the factor seen by testing various clinical tests, including UltraSonoGraphy (USG), Computed Tomography (CT), and
- $u_i$  determines the other factor that may influence the change in disease progression.

Therefore, equation 2.4 is a simple linear regression model with  $y$  being the only regressor and is linearly dependent on  $x$ , where  $x$  is the independent factor or known as *regressor*.

## 2.4.2 Regression in epidemiology

Epidemiology is a branch of medicine and deals with the factors related to identifying the incidence of the disease, along with the possible way to control its progression. In epidemiology, the regression model assists in not only identifying the risk factors associated with disease but also in predicting its progression (Montgomery et al., 2015). Regression model achieves this by building a relationship between the disease and identified risk factors. Historically, the main advantage of a regression model is its computational and theoretical simplicity. This allows the statisticians to have closer view towards the data behaviour.

But, it was observed that an application of these methods became a challenge as number of variable that were to be investigated increased Attili et al. (1995). Suppose, there are ten variables and from these variables, we are intended to identify the associated risk factors. In regression, an association of each of these variables is analysed at two levels, developing 1024 cells of multiple cross classification and each cell needed rigorous observation and analysis. This may lead to information loss during the analysis. This challenge becomes bigger while dealing with a highly complex pattern of risk factors generated by cross multiplication between the factors. By the introduction and application of maximum likelihood (Walker and Duncan, 1967) and also by the advancement in machine learning algorithms, regression was able to overcome these challenges.

This invited a lot of researchers to diffuse the model in identifying the risk factors associated with and predicting the disease progression. Truett et al. (1967) first applied regression in cardiovascular epidemiology. Since then a lot of researchers have analysed its application in various other streams of medicines. We could find their application in estimating the volume of liver for liver transplantation (Olthoff et al., 2015), for identifying

the prognostic factors for predicting the disease progression (Kumarasinghe et al., 2015) and finding the risk factors associated with cancer along with predicting their prevalences (Group, 2016).

### 2.4.3 Significance of regression in gallstone disease

Alvi et al. (2011) conducted a retrospective analysis of odd ratio among *case-control* for a period of 19 years. *Cases* were the patients with gallbladder and *control* were the patients with no gallbladder. They used regression technique to find the odd ratio of control and cases.  $Odd\ ratio = \frac{odds(cases)}{odds(control)}$ , where  $odd = \frac{probability\ of\ success}{probability\ of\ failure}$  (Bruin, 2011). Using the regression model and methodology of finding odd ratio, Alvi et al. (2011) were able to find the risk factors (age > 55 years, solitary stone and stone > 1 cm). Kim et al. (2011) applied the regression model for analysing the significant association of insulin resistance with the formation of gallstone. This study included 4125 Korean women between the age group of 30-79 years. The regression model categorized the study and showed the gallbladder stones, age, obesity, abdominal obesity, hyperinsulinemia, and high Homoeostatic Model Assessment-Insulin Resistance (HOMA-IR) index, were the significant independent factors in post-menopausal women and low high density lipoprotein-cholesterol in pre-menopausal women.

Wang et al. (2012) applied the technique for identifying prognostic factors for gallbladder. They compared the result of univariate analysis with regression model and found that earlier statistical method identified (five) significant factors while later identified (three). Using these significant factors, they were able to diagnose the complication at an early stage so that the prognosis of the gallbladder could be done more effectively. Wang et al. (2017) analysed the per-operational condition causing the risk for post-operation of gallbladder ejections. They utilized the technique of uni-variable and multi-variable regression model. They found the two factors: wall thickness of the gallbladder and lithotritry that are to be analysed before conducting an operation. The finding suggested that they may cause gallbladder ejection even after the operation.

Srivastava et al. (2010) used regression model for finding the significance and association of toll-like receptors polymorphism with gallbladder cancer. They used the technique of odds ratio for calculating the probability of association. Mønsted Shabanzadeh et al. (2016) found the symptoms associated with the abdominal pain due to a newly formed gallstone. They applied the technique of logistic regression for analysing the significance

of newly formed gallstone and the reason for abdominal pain along with the projected pain for a longer duration. Using regression model, they were able to find that newly formed gallstone does show the indication of abdominal pain at upper abdominal and for a longer duration. Muszynska et al. (2017) investigated using multiple logistic regression for identifying the predictors that could predict the gallbladder cancer. On statistical analysis, they could find that older aged women suffering from jaundice and had undergone cholecystitis were under higher risk for gallbladder cancer. Gautham et al. (2011) conducted a study using regression to identify the prevalence of GSD in India based on geographic and gender distribution along with the progression of the disease over time. Ryu et al. (2016) observed the risk level of gallstone progression towards cancer. Using the regression model they were able to identify that gallstones are significantly associated with mortality due to hepatobiliary cancer. Creasy et al. (2017) analysed the probability of residual disease at the time of re-operation. They used logistic regression and classification, and regression tree for conducting the analysis. The model showed  $A_Z$  of 0.78 for predicting the risk-level of the patients at the time of operation. Using this result, patients at high risk were stratified.

*On studying performance of regression we could observe that:*

- Regression models assume that all the identified significant risk factors are available in all the test cases. But their absence in the test cases is one of the limitations in applying heterogeneity in regression analysis. Such a limitation of regression is known as *data dredging* (Marshall, 2001).
- This situation could be avoided by pre-identifying the significant factors. But, with the property of data dredging, it fails in accounting adequately the large errors and cant be widely applied for computational purpose (Draper and Smith, 2014).
- The associative relationship observed by the regression models are less interpretable when compared to the casual relationship. However, identifying the casual variances which are inverse and used for regression is very difficult. With these restrictions, regression models could be recommended only with pre-analysed significant factors. But, this purely depends on the methodologies and planning of systematic reviews executed by the clinicians to get an insight to the clinical data.
- Further, the performance of regression model is determined by the sample size. But,

with the restricted samples, discovering the right interpretation is always a challenge in regression.

- Regression models were built with an unrealistic assumptions of data and error distributions, due to which they had a lot of limitations. Hence, this might be few of the reasons for the statistical shift from regression analysis to Artificial Neural Network (ANN).
- As in many cases, ANNs have proved to perform well we continued our interest to find their utilization and application in the field of medicine.

## 2.5 Artificial Neural Network (ANN) Outperforming Regression

In literature, a lot of regression related studies in many diversified areas were found. But at the same time, many comparative studies between ANN and regression model were performed. And it was observed that ANN had outperformed the regression models. Jovanovic et al. (2014) compared their work on regression with ANN for selecting the patients with higher risk towards Endoscopic Retrograde Cholangio-Pancreatography (ERCP). They could find that, ANN showed better accuracy with  $A_Z = 0.884$  when compared to their earlier regression model  $A_Z = 0.787$  (Jovanović et al., 2011). Vukicevic et al. (2016) proposed an expert system which can automatically build ANN and validate its performance. They also mentioned in their study that among various statistical tools, ANN was more suitable for the prediction and diagnosing concurrent Common Bile Duct Stones (CBDS). Suarez et al. (2016) aimed in accurately predicting the choledocholithiasis using the impact of laboratory trends. They compared the performance of regression model with ANN and found that ANN was more accurate than the earlier one. Wall (2013) used ANN for predicting the possibility of survival of the patients suffering from pancreatic ductal adenocarcinoma. In his study, he found that ANN outperformed the regression model. Hong et al. (2013) compared the performance for predicting the organ failure in the people suffering from AP and observed that ANN shows higher accuracy for the prediction.



## 2.5.1 Artificial Neural Network

ANNs mimics the human brain and are composed of a non-linear combination of computational elements known as neurons. Neurons formed by the biologically non-decomposable units is a mathematical function in ANN. Its task is to receive the inputs, perform some mathematical processing and produce the calculated output. The output is processed through the processing units simulating the neurons. For this, neurons are interconnected using the synaptic connections as in the human brain. This synaptic connection allows the signal to pass through the network. The signals are the processing elements in ANN which are processed through the interconnecting weights.

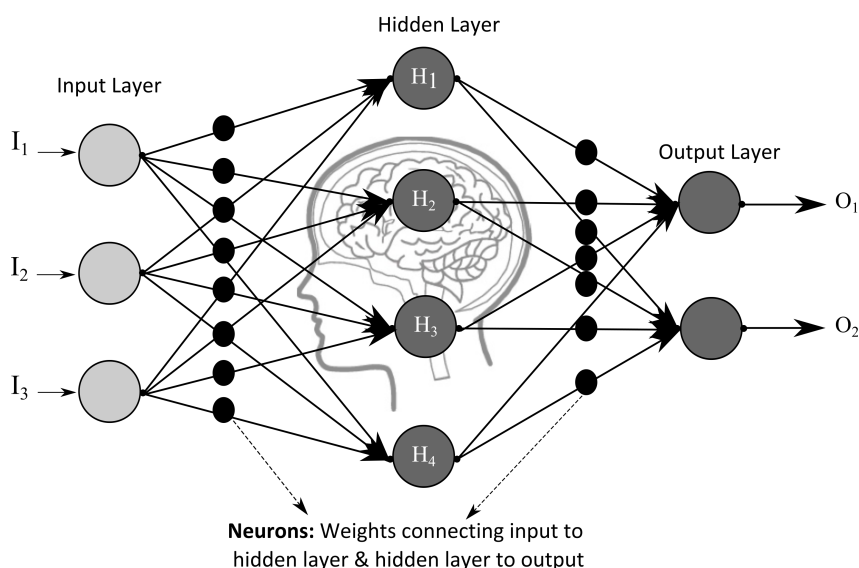


Figure 2.2: Illustration of ANN

## 2.5.2 Application of ANN

The accuracy of ANN could be evaluated in many ways based on their properties. The first feature is *variability*. It is a property of ANN to evaluate the performance of risk-adjusted models (Anderson et al., 2003). *Calibration* is the second feature which defines the property of ANN to assign the rightly identified risk factors to the individual cases (Hosmer Jr et al., 2013). The third feature is about discriminating the cases based on the outcome of interest and is known as *discrimination* (Swets, 1988). The fourth feature is related to the *precision* of the model. The model can repeatedly give an accurate and similar result for the same input variables using the same statistical techniques leading the accurate output. This makes the model more *reliable* and *stable* (Altman, 1990).

### 2.5.3 ANN in epidemiology

Due to its non-linearity classification nature, we could see a lot of its application in medical science. Analysing the samples of blood and urine (Catalogna et al., 2012), classification of leukemia (Dey et al., 2011), for identifying and diagnosing tuberculosis (Elveren and Yumuşak, 2011), analysing the complicated effusion samples (Barwad et al., 2012). We could find a lot of their application in speech and image processing. They were seen in radiography analysis and even analysis of living tissues (Saghiri et al., 2012). They have been successfully applied for classification of; benign from malignant breast lesions (Chan et al., 1997), lung disease and coronary artery disease (Ashizawa et al., 1999) and for the outcome analysis of pancreatitis (Keogan et al., 2002).

In general we could see their application in oncology (Saxena and Burse, 2012), urology (Mantzaris et al., 2011), paediatric (Mantzaris et al., 2010), cardiology (Karabulut and İbrikçi, 2012), ophthalmology (Güven and Kara, 2006), neurology (Blahuta et al., 2012) and others (Dobchev and Karelson, 2016). Keogan et al. (2002) investigated ANN predictive model, devised with both *clinical and radiologic features*. Since then, they have been successfully applied for diagnostic radiology including pulmonary embolism on ventilation-perfusion scans and differentiation of benign from malignant breast lesions (Chan et al., 1997).

Though ANNs have been applied in diversified branches of medicines (Figure 2.3), we continued our interest to know how successfully they are involved in the analysis of gallstone. We aimed to find the efficiency of ANN for the prediction and classification and if there any scope for further research?

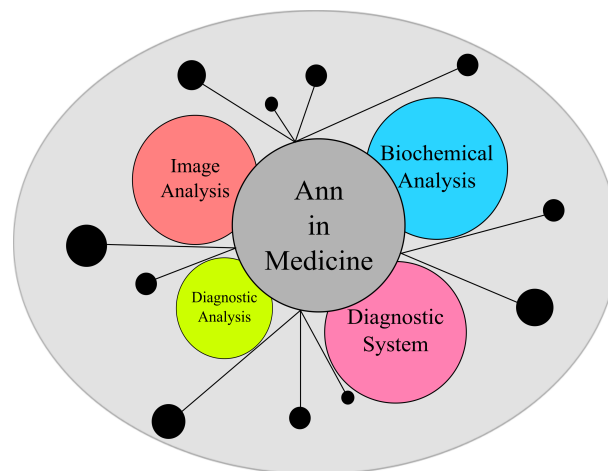


Figure 2.3: Overview of the main applications of ANN in medicine

## 2.5.4 ANN for disease management of GSD

Due to the limitations of traditional statistical techniques for predicting the disease severity and mortality, several statisticians (Mofidi et al., 2007; Yoldas et al., 2008) have recommended the use of ANN as the predictive model for predicting and assessing the patients with GSD. The application of ANN can be broadly classified into *prediction*, identification of *significant factors* and *classification*. In *prediction* ANNs were applied for predicting the severity of choledocholithiasis (Adams et al., 2015), Acute Pancreatitis (AP) (Andersson et al., 2011), disease progression and organ failure in AP (Hong et al., 2013), conversion of LC to Open Cholecystectomy (OC) (Eldar et al., 2002), CBDS on ERCP (Vukicevic et al., 2016) and whether the stones in distal ureter would pass or may need intervention (Cummings et al., 2000). It was used for *identifying the significant factors* associated with pancreatitis (van den Heever et al., 2014) and CBDS (Vukicevic et al., 2016). Opačić et al. (2015) applied ANN for *classifying* the benign and malignant in pancreatic cancer. Where in Yang et al. (2013) classified pancreatic adenocarcinoma from non-neoplastic tissue. They were also efficiently applied in regions segmentation in ultrasound image for analysing high intense region and identifying the percentage of gallstone (Lian et al., 2017).

Ikeda et al. (1997) compared the performance of ANN with Bayesian analysis, Hayashi's quantification method II, and the observation of radiologists. They found that ANN was better for classifying pancreatic ductal adenocarcinoma and mass-forming pancreatitis. Due to the complications in removing CBDS by laparoscopic procedure, Golub et al. (1998) with the help of ANN conducted experiments on it. They accurately screened the patients, who are at high risk for CBDS so that a pre-operative ERCP could be avoided. Jovanovic et al. (2014) studied 291 patients undergoing ERCP for suspected choledocholithiasis. They applied ANN and found that it accurately predicted the patients with positive and negative findings on ERCP. This study was further investigated and analysed by Adams et al. (2015). Cummings et al. (2000) conducted analysis for assisting the clinicians in taking appropriate decisions about "*will the small stones present in the distal ureter would pass out or may need interventions*".

Mofidi et al. (2007) developed a model to identify AP and predict the reason for causing mortality. On comparison with APACHE II and Glasgow severity scoring system, ANN showed better accuracy in prediction of disease progression, organ failure and mortality of the patients suffering from AP. Organ failure during the first week after onset of AP is

one of the death threat (Hong et al., 2013). Hong et al. (2013) used ANN for accurately predicting the patients who may have organ failure. Another challenge was in identifying the prognostic factors for prognosis of pancreatic disease (Bartosch-Härlid et al., 2008) so that proper diagnosis could be provided. For that, Andersson et al. (2011) using ANN identified the significant factors and predicted the progression of AP for finding its severity.

With an advancement in image processing techniques, Das et al. (2008) applied ANN for classifying the pancreatic adenocarcinoma from a non-neoplastic tissue of chronic pancreatitis, benign from malignant patterns and diagnosing pancreatic cancer (Yang et al., 2013). After filtering the noise in the image to get a narrow region of the gallbladder, Lian et al. (2017) used the pulse-coupled neural network to identify the high-intensity region in the image obtained through an ultrasound of GSD.

### **2.5.5 ANN as an expert system for predicting the disease progression**

Vukicevic et al. (2016) proved that ANNs were slow learners, so they developed ANN expert system by applying evolutionary assembling approach. They used a genetic algorithm for automatically configuring ANN and overcoming the limitations of being a slow learner. The model identified the significant predictors and accurately predicted the patients who may have CBDS when compared to all other statistical techniques. ANN were known as slow learners due to its high computational steps needed for identifying the right combination of neurons and hidden unit. Fahlman and Lebiere (1990) stated this slowness as a *moving target problem*. The back-propagation algorithm used for learning this combination was slower for highly complex data. The model proposed by Vukicevic et al. (2016) not only increased the accuracy of prediction but also built a simplified expert system. Keogan et al. (2002) used ANN for predicting the length of stay in the hospital for the patients suffering from AP. A similar kind of study was later conducted by Muhammet and Guneri (2015).

### **2.5.6 Observations and limitations of ANN**

We understand that ANNs are well-established models for classifying the disease severity and pattern recognition. But, with its high processing units and limited room for computational combinations, it was not only slow for identifying the infrequent patterns, but also failed (Ohno-Machado, 1996). Halonen et al. (2003) compared the performance

of regression with ANN for predicting the fatal outcome of severe AP. But, surprisingly the earlier one showed better accuracy ( $A_z=0.862$ ) than the later ( $A_z=0.847$ ). Chang and Hsu (2009) developed a screening test for prediction of pancreatic cancer. They compared stepwise logistic regression, ANN, Genetic algorithm-logistic regression. Genetic algorithm-logistic regression showed better performance  $A_z = 0.921$  than ANN  $A_z = 0.895$ .

The performance of ANN is unstable due to the presence of local minimum in back propagation (Akande et al., 2014). The convergence towards the local minimum is done by backpropagation and is very slow. The convergence at local minimum is the objective of ANN and is never ending learning process (Balázs, 2009). Further, backpropagation requires functions in a networked structure and has a high impact on learning capabilities. This limitation of ANN was widely reported by Cunningham et al. (2000).

### **2.5.7 If not ANN then what?**

Though we found a lot of successful application of ANN, other statistical models have performed well. The major limitation in ANN was in training and identifying the optimal combination of neurons and hidden units, where the error was minimum. Because of these reasons, there was a need for effective training algorithms which could build model adaptively during the training phase. This prompted us to think and research further on constructive training algorithms. These algorithms have two main classes. One set of classes uses the traditional model for structuring the network by training and accumulating several networks. This needs high computation and must be infeasible. Other categories are represented by the CCNN (Fahlman and Lebiere, 1990). CCNN automatically adapts the model based on the training process. The training process here takes lesser computation and address the problems as mentioned earlier of backpropagation. So we continued our research to find the application of CCNN and their suitability in the field of medicine.

## **2.6 Cascade-Correlation Neural Network (CCNN)**

Fahlman and Lebiere (1990) introduced a new CCNN architecture with cascade correlation of network which learns by experience. Here the weights are frozen as the hidden units are added to the network. CCNN works on two key architecture. *First* during the training phase, if the network demands that addition of new neurons would assist in solv-

ing the complex problem more accurately, then CCNN would add new neurons. Secondly, addition and training are sequential.

### 2.6.1 Study on application of CCNN

Fahlman and Lebiere (1990) built a model which adaptively identified the optimal network connectivity and weights. The model could solve the classification task more efficiently than the existing models through supervised learning. Shavlik et al. (1991) conducted experiments on linearly separable data (audiology and soybean) as well as non-linearly separable data (chess). It was noted that CCNN used 1-2 magnitude lesser epochs than backpropagation and perceptron algorithm. As-well-as they needed fewer hidden nodes (Shavlik et al., 1991). On testing for accuracy, CCNN found to be better than backpropagation on soy-bean data, but backpropagation was better on chess and radiology data. On comparison with perceptron algorithm, CCNN was better in all the tested experiments. Itchhaporia et al. (1996) compared the performance with ANN by applying in cardiology for diagnosis of coronary artery disease and myocardial infarction. They found that ANN was too complicated and complex in training process, which made it too slow to get modelled when compared by CCNN. Hirayama et al. (1993) experimented feedforward controlling of arm movement. They were successful in planning time-accuracy trade off and quasi-power-law type of speed-accuracy trade-off.

Doering et al. (1997) modified CCNN and built an optimal CCNN which converged faster than the existing techniques. They proved that the linear output of neurons could be solved within a finite number of steps. The model was optimized by choosing optimal weights. Thus the proposed model showed better performance when compared to CCNN proposed by Fahlman and Lebiere (1990). The model was further generalized by Chudova et al. (1998). The ability and time elapsed in discovering an optimal model were comparably faster. But on further investigation, we observed that the existing model needed different training and retraining techniques to improve its performance. Song et al. (2011) regularized the correlation method and reduced complexity of CCNN. This improved the efficiency of CCNN and helped in faster convergence. Using this efficient model Song et al. (2011) developed first break the auto-picking model. The model identified five significant risk factors which were adequate for separating the first break and non-first break. The model achieved good efficiency in testing seismic data.

Chandra and Varghese (2007) used CCNN for identifying cipher system from cipher text generated by block cipher and stream cipher and found CCNN was 14% more accurate than ANN. Zhao et al. (2011) observed that CCNN showed better performance for fault detection in sensor and recovering data. (Lam and Smith, 1998) modeled and improved the performance of Abrasive Flow Machining (AFM) using CCNN. AFM is a part of an automotive engine. CCNN was able to predict the termination point where the AFM would meet the airflow specification. The result showed that CCNN outperformed regression model. Diamantopoulou et al. (2005) modified ANN by applying cascade correlation algorithms for its training. The weights interconnecting neurons were changed using Kalman's learning rule (Kalman, 1960). The modified ANN here successfully proved to be a useful model for identifying the monthly values of water quality parameters and predicting the quality parameters in water (Diamantopoulou et al., 2007). Bathen et al. (2007) used CCNN for predicting the progression of breast cancer. The objective of this study was to discover the hormone status, histological grade and axillary lymphatic spread the diseased patients.

*On studying performance of CCNN we could observe that:*

- Though we found very limited applications of CCNN, it outperformed ANN and other statistical techniques.
- They addressed the limitations of ANN and gave better accuracy though with its limited application. We understood that on some data their performance was not as expected. The comparison study conducted by Burke et al. (1994) showed that ANN had better accuracy than CCNN. Chudova et al. (1998) on evaluating the ability of CCNN understood that the model needs training and retraining for optimizing their performance.
- However, CCNN had a challenge of identifying where to add new neuron, but traditionally it was studied that the neurons were added sequentially. The other challenge was to find when to add a new node and develop the connection of these new node (Yang and Honavar, 1991).
- It is learned that though CCNN performs better than ANN, it needed further modification/ optimization for showing better performance. This made to propose a ModCNN.

- Here in ModCNN the neurons and hidden units are adapted automatically/ dynamically for giving better accuracy. ModCNN assisted in identifying the independent factors, which were fed to predict and identify the cases which may need emergency interventions in later stages of treatment. The experimental comparison of the proposed study with ANN and CCNN showed that ModCNN was better than later techniques.

## 2.7 Bridging Statistical Analysis with Process Mining

The Figure 2.4 explains about the analysis and how we are trying to bridge a gap between the statistical analysis and process mining for meeting our objective. The proposed work aims to recommend the critical treatment path for the identified critical cases. Critical treatment path is a sequence of activities in a healthcare process to be performed with minimum waiting and processing time with the help of adequate resources. We applied the technique of EHR process mining for finding the critical path. It is observed that EHR improves the quality of care and process mining assist in modelling the treatment management by quantifying the resource utilization using EHR data (Baker et al., 2017). In this section, we would discuss the application of EHR process mining in the healthcare process.

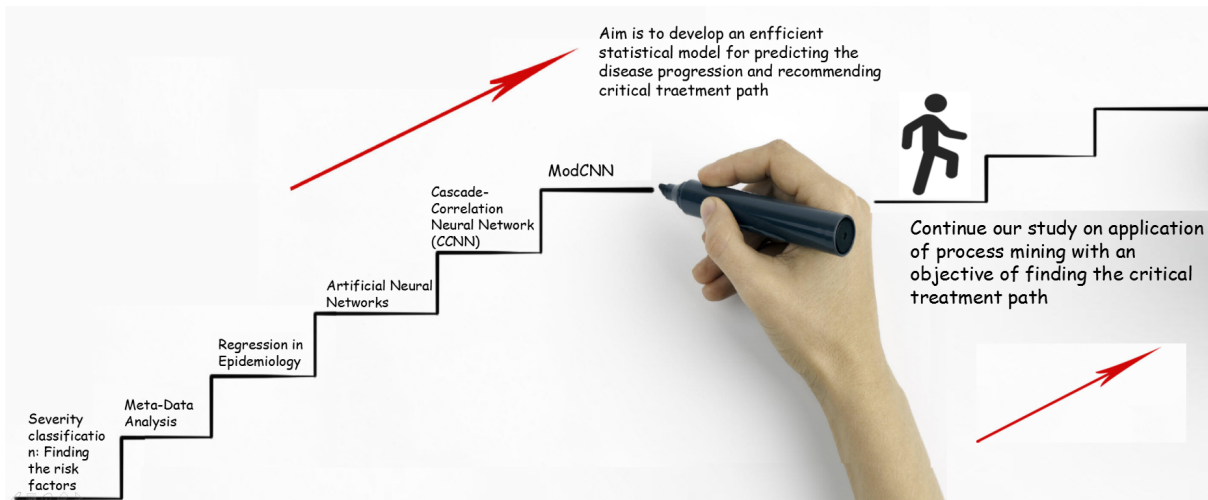


Figure 2.4: Flow of Study



## 2.8 Electronic Health Record Process Mining

EHR is a prospective and actual information of a patient medication, history of operation, treatment including past diagnostics and follow-ups, and result of laboratories and signal/ image processing recording which are carried retrospectively (Rosenthal, 2013). They also provide assistance for non-clinical information. This would help in developing a generalized protocols and treatment pipelines. The primary care centre, hospitals, pharmaceutical and laboratories are interconnected by Health Information Exchange (HIE) to record EHR and make them to play a vital role in communication between healthcare and statisticians (Mertz, 2014). Sittig and Wright (2015) addressed the use-cases of EHR in-order to assist:

- *Clinicians* : For providing safe and efficient healthcare.
- *Researchers* : To mine and extract information about disease health-care process.
- *Administrators* : The reliability on single EHR developer is decreased.
- *Software Developers* : To overcome the limitations of current EHR and build an innovative application.
- *Patients* : To access their personal health information from anywhere, where they are getting health check-up.

### 2.8.1 Assistance of EHR in Clinical Decision Support System (*CDSS*)

Kim et al. (2008) developed an independent as well as inter-operable and extensible *CDSS* using EHR. The interoperability was between the *CDSS* and knowledge engine. Knowledge engine is the key for understanding EHR and assisting in taking appropriate clinical decisions. Focsa (2010) re-engineered EHR for assistance in taking clinical decision in clinical workflow system. This was achieved by management of process in workflow along with knowledge extracted using semantic models. Further re-engineering on EHR was performed to make it more optimal. (Peleg, 2013) integrated EHR with organizational workflow to find the non-compliance pattern (Riha et al.). By discovering frequent non-compliance pattern, *CDSS* could be made more efficient and accurate. Such an *CDSS*

that could identify positive correlation between prior study and EHR about a case was highly needed in emergency department (Grana and Jackwoski, 2015). With the assistance of model that could identify the non-compliance pattern, Chen and Sarkar (2014) developed a knowledge discovery framework for mining EHR data. Using this framework, they were able to find an association of disease-disease, disease-drug, and disease-gene. The identification of this association would assist in data selection, preprocessing, transformation, data mining, and validation. Jonnagaddala et al. (2017) used EHR data having information of demographic, billing, medication and lab reports and investigated the advantage of text mining on it. This analysis helped in providing better healthcare by identifying cohorts, correlations in disease, phenotypes in genome-wide association and the associated risk factors.

### 2.8.2 EHR in healthcare system

It is observed that for better EHR based event logs, it is important to have better process-aware EHRs and healthcare information technology. Most of the issues raised and analysed about the EHR application is in its workflow system. For this, an application of process-aware EHR and healthcare information technology system would enhance the adaptability of EHR in healthcare. Using a well established and adopted EHR process model,

- The best treatment process could be identified.
- The communication about the treatment among the clinical staffs (task assignments) could be made.
- Provide best possible care to the patients.
- Healthcare resources could be used in a best possible way.

Delias et al. (2013) clustered the activity patterns in emergency department. Basole et al. (2015) used EHR data to develop an interactive model that could mine the data and provide the visual interaction by exploring process. Helm and Paster (2015) applied process mining to get an insight of complex healthcare process. Kumar et al. (2014) developed a prototype (AsthmaFlow). This framework helped clinicians in visually analysing and exploring the process involved in emergency department of paediatric asthma. The framework proposed here assisted clinicians to predicting the drifts in clinical cases by combining process mining with machine learning and understanding the

process flow (Bostock et al., 2011). Wu et al. (2010) developed a model that could predict the heart failure, six months prior to the event along with actual date of diagnosis using EHR data. Almasalha et al. (2013) mined the hidden patterns in EHR and extracted knowledge in nursing care. Using this model the authors were able to predict whether the patients admitted in the hospital would be able to meet pain relief goals. Here the patients with less than three days of hospital stay were compared with the patients suffering with end-of-life disease for longer hospital stays. Jensen et al. (2012) mined the EHR data to classify the patients more accurately based on the disease. As-well they could find the correlation-ship among the diseases. Li et al. (2013) proposed a semi-automatic model for phenotype (Phenotype studies are set of observable characteristics in an individual due to interaction of its genotype).

From the study of Jalloh and Waitman (2006) it was understood that EHR were the best data source for reducing the medical error. Using this EHR Chazard and Beuscart (2009) built a framework to identify the risk factors associated and prevent its prevalence. Helm and Paster (2015) investigated whether EHR data generated is suitable for applying in process mining. For this they developed a simplified simulation of radiological workflow process using Petri nets. Hence it is observed that EHR event logs could be used for process mining of healthcare process.

### **2.8.3 Adoption of EHR in India**

Most of the healthcare industry is trying to implement EHR based system. This is achieved by encouraging the conversion of paper-based application to EHR-based (Black et al., 2011). Very soon we can see them becoming a default healthcare application in India, marking beginning of digital India. Sharma and Aggarwal (2016) in their study could find around twenty hospitals where EHR system has been successfully implemented.

Karthikeyan and Sukanesh (2012) observed that the hospitals which have successfully implemented EHR are more efficient and consistent. The medical error in those hospitals are exponentially very low and patients satisfaction is very high. In a report on "Electronic Health Record standards for India" by Ministry of Health & Family Welfare, Government of India (2013), EHR vendors were classified based on their new scientific creation and workflow processes. Hence it is not just important to adopt any EHR system, but one which is scientifically tested and has a better process model for the healthcare system under consideration.

The recent paradigm shift in process mining has seen lot of its research applications in healthcare domain. EHR process mining extracts the information by building the relationship between activities and the resources conducting those activities. Mans et al. (2013) showed several related applications of process mining in a healthcare process. Rebuge and Ferreira (2012) designed and proposed a process model for an emergency care in a public hospital in Portugal. Perimal-Lewis et al. (2012) from Australia studied the patients journey within the hospital. Poelmans et al. (2010) developed a process model for breast cancer treatment. On literature survey, we found several related works on healthcare process.

## 2.9 Process Mining in Healthcare

EHR records the healthcare activities and assist in maintaining proper and complete information. Nasiri et al. (2013) suggested the standard format for data exchange in EHR so that it can be further analysed using process mining techniques (Grana and Jackowski, 2015). By adopting EHR in healthcare, application of process mining in healthcare system has increased largely in recent years. This would assist in understanding the clinical process and its complexity (Kaymak et al., 2012).

The data that are recorded in a healthcare process are events and it varies based on the department. These recordings are limited to a particular department. For example in billing department, it records the information about the payment related to healthcare services. This could be overcome by proper coordination among intra-department and inter-department known as cross-organizational process mining. Tomar and Agarwal (2013) worked on cross-organizational process mining and its application in healthcare to identify the regular and exceptional cases. Yang and Hwang (2006) built a process model to detect the fraudulent and abusive cases in claiming expensive health insurance. The model was built using data mining technique with an adaptable and extensible detection model. Gupta (2007) built a process mining model to assist in cross function and multidisciplinary process. The model was built with the combination of clustering and association rule techniques. They helped in grouping similar characteristic patients.

Its been two decades since the clinical data were recorded as EHR (Shortliffe and Cimino, 2013) and data mining been first applied for statistical analysis on them (Klößen and Zytkow, 2002). On research we saw a lot of statistical technique for analysing the clinical data (Benneyan, 2001). Obenshain (2004) surveyed various statistical techniques

which can be applied for analysing the healthcare data and is shown in Table (2.2).

Table 2.2: Different statistical techniques categorised based on different objectives

| <b>Objective</b> | <b>Supervised</b>  | <b>Unsupervised</b>   | <b>Comment</b>   |
|------------------|--|---|--|
| Prediction       | Ordinary least, Squares regression, Logistic regression, Neural networks, Decision trees, Memory-based reasoning, Support vector machines, Multi-adaptive regression splines | Not applicable  | From analysis we understood all these models performed well. |
| Classification   | Decision trees, Neural networks, Discriminant analysis, Bagging and boosting ensembles, Naïve Bayes classifiers.   | Clustering (eg, K means), Kohonen networks, Self-organizing maps. | Performed well, but needed further research                  |
| Exploration      | Decision tree  | Principal components Clustering (eg, K means), Link analysis      | Not well implemented   |
| Affinity         | Not applicable   | Associations, Sequences Factor analysis                           | Well used  |

### 2.9.1 Spaghetti-like process model

Healthcare system integrated with clinical guidelines could be used to dynamically guide clinicians in taking critical decisions (Dumas et al., 2005). Even though EBM has strictly asked all the hospitals to follow the clinical guidance, often most of the hospitals fail to obey them due to their complex policies and patients characteristics. So, Rovani et al. (2015) applied process mining and built a declarative model to act as a mediator between the clinical guidance and a healthcare process. They improved the existing system using process mining technique to analyse the deviations/ drift and assist in following the

guidelines more effectively. But, Kaymak et al. (2012) argued that current algorithms discovered spaghetti-like process (Kaymak et al., 2012) for healthcare system, due to which adapting them was a challenge. The complexity of healthcare process is due to *dynamic* (Gupta, 2007), *complex* (Mans et al., 2009), *ad-hoc* (Mans et al., 2009) and *multidisciplinary* (Gupta, 2007) in nature. An Example spaghetti-like process is shown in Figure 2.5. But, this could be overcome by:

- Incorporating medical knowledge in process mining algorithm.
- Preprocessing the data based on clinical knowledge and decreasing the search space. Smaller the search space, simpler model could be discovered.
- Healthcare processes are not always simple sequence of events and we cant expect the process to follow the predefined sequence. Hence they are dynamic in nature. This is due to physical system that are dynamically described. Thus, build an expert system that could predict the changes in the sequence.
- Clinicians aim at multiple goals, following the single healthcare process. The goals are needed to be known and subjected, if the algorithm has to be properly depicted.

Due to highly flexibility and heterogeneous nature of healthcare process it is important to have a model which is ready to adapt the sudden changes in the process. This is needed, because most of the healthcare process believes that the process remains in the steady-state from the beginning to the end of its process flow, but this is not true. Since, the patients frequently deviate from the actual treatment path (Song et al., 2009), so Bose et al. (2011) proposed the concept drift. They understood that the process may be changing on the course of execution, while the cases are getting handled. This is known as second-order dynamics and was well analysed. The result of simplified process is shown in Figure 2.6.

### **2.9.2 Careflow: A patients journey within the hospital**

Kim et al. (2013) evaluated a patient care process in a healthcare clinic in-order to reduce the waiting time. For this, they built a machine-driven model which could identify frequent path of execution. The frequent path of execution is the treatment path through which the patients make their journey in the hospital. McGregor et al. (2011) built a

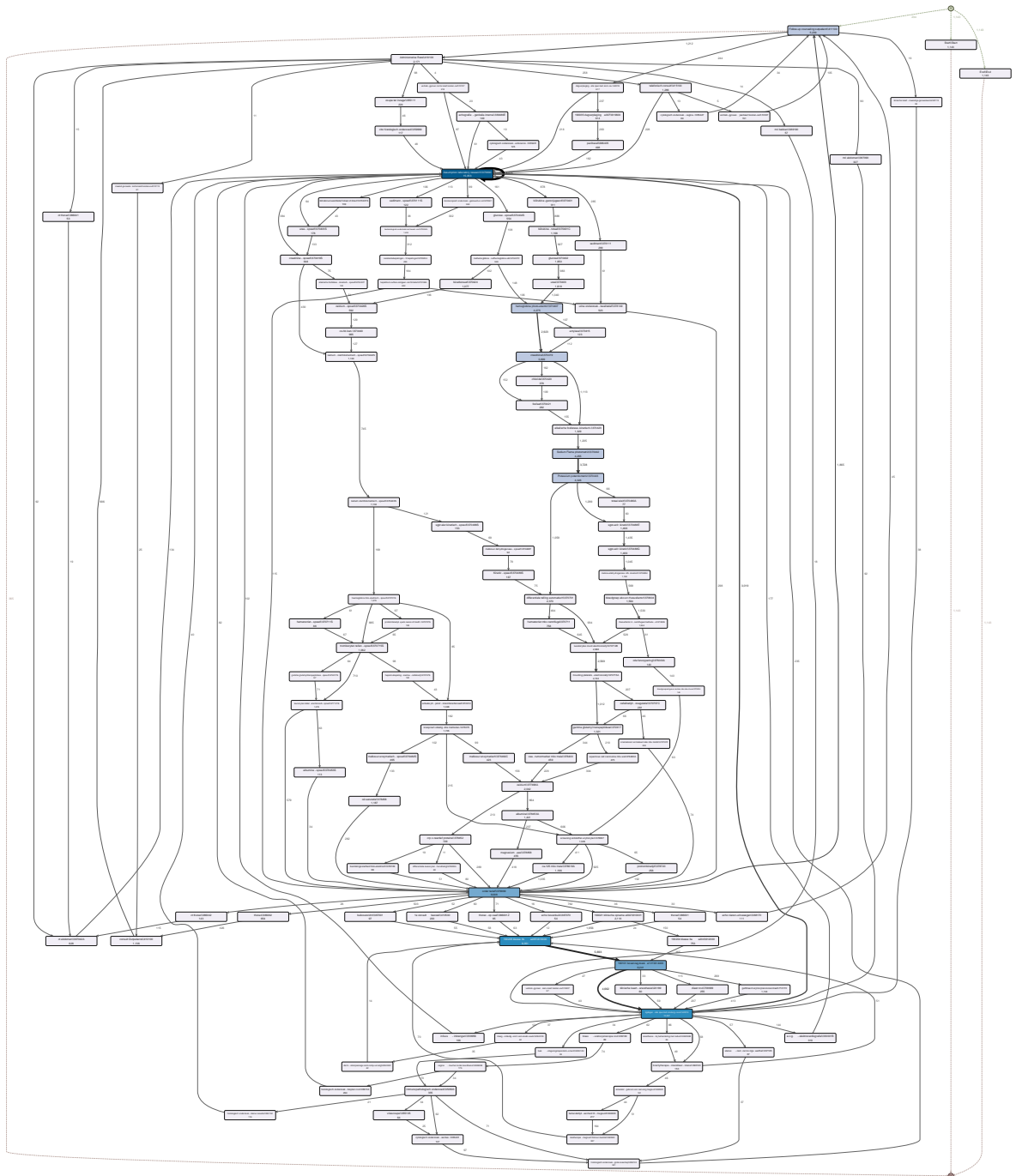


Figure 2.5: Example spaghetti model discovered from standard process mining event log repository (4TU.Centre for Research Data).

framework that could be used to update the patients journey and provide critical care. An approach was made to discover knowledge and develop a process flow mappings. The framework can be used to update the patients journey and provide critical care. A model was built using "*patients journey modelling architecture*" of MacDougall et al. (2011). This journey or process flow needed to be monitored so that proper treatment management could be provided. For that, the processes in the healthcare should be streamlined.





which was capable of analysing EHR data and converting it to a healthcare model. The proposed model was tested for its accuracy, thus reducing the medical error.

- From study we found very less research done on GSD. Hence, there was a need for analysing the disease progression so that any emergency situation could be well handled without making any medical error. Our aim was to identify those case which may become complicated and suggest critical treatment path. We built a model for identifying the significant factors associated with GSD and to predict its progression. This model could identify critical cases which needed emergency interventions and recommend the critical treatment path.
- We could find few work that predicted the patient's journey in the hospital. This motivated us to work further on this. In this work, we recommend the journey path, along with right resources for handling the activities in the path and provide proper care-flow.
- ANN has shown good accuracy in clinical research. But, few studies revealed that it was slow due to its very complicated backpropagation technique. This made us to think of a model, that could perform better than ANN. Later studies has shown that CCNN has performed better than ANN. But, it too had limitations so we modified it and proposed our model, ModCNN.
- Process mining was studied to be applied for the analysis of business application. But its application in clinical research was very less. This was due to improper data recording. On having EHR data, we could run EHR process mining techniques on them and recommend the critical treatment path along with the efficient resources.

## 2.11 Problem Statement

*Design and development of a clinical decision support system for identifying the cases which may need emergency intervention and recommend a critical treatment path using machine learning and process mining techniques.*

### 2.11.1 Research objectives

- To develop a clinical decision support system by dynamically selecting the best suitable machine learning technique discovered by comparing their performance for

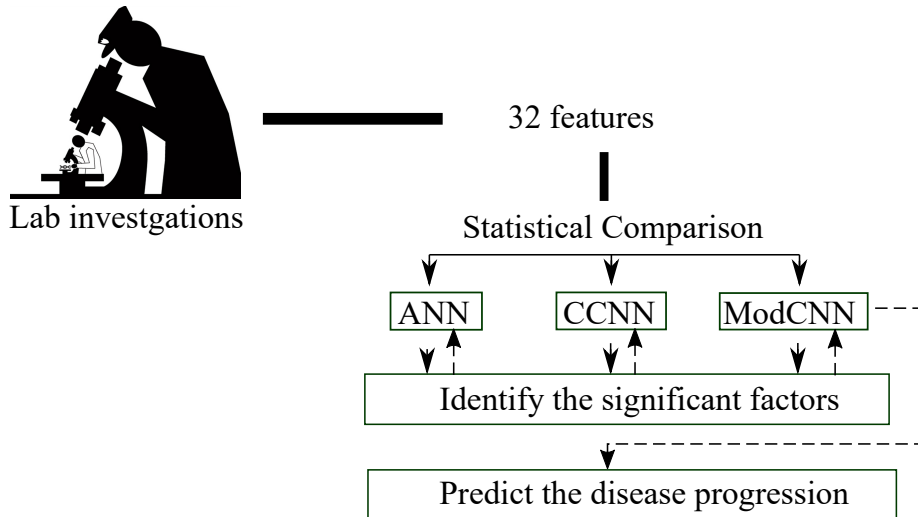
the input data.

- Identifying the critical cases by predicting the disease progression using the significant factor associated with each spectrum. (Gallstone disease – a case study).
- Recommending the treatment path and an efficient resource for each evidence in the treatment path using electronic health record process mining technique, for the identified critical cases.

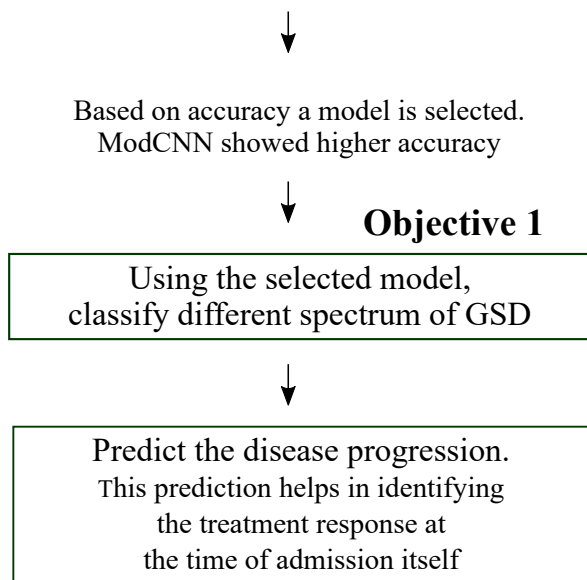
## 2.12 Synergies of process mining in Healthcare

Aim of this work is to assist the clinicians in taking appropriate clinical decisions regarding the treatment management and reducing the medical errors using process mining and machine learning techniques. From the study it was understood that most of medical errors were due to process error. For that matter, we designed and developed a *CDSS* to recommend the critical treatment path for the cases which needed immediate interventions. But, it was retrospectively observed in the case study of GSD, that few cases initially showed positive response towards the treatment, later needed interventions. The *CDSS* developed in this work identifies the critical cases at the time of admission using ModCNN and then recommends the critical treatment path for them using process mining techniques.

The *challenge and issues* in this work was in developing an information system to record EHR of the patients who came with the complaint of abdominal pain. It was hard to maintain the system in our study area. The recorded information was later converted into the compatible format. Yet, another challenge was in selecting an optimal model that could stratify the cases and predict the disease progression more accurately. Precision was more important here as the experiments were conducted on real life study. The proposed ModCNN though showed better accuracy in prediction failed to perform well, when tested with varying feature size. This challenge was addressed by adding neurons in parallel for different hidden units using master-slave model. The identified critical cases were further needed to be recommended with the critical treatment path to avoid later complications. The real challenge was in finding the critical activities and right set of available resource for performing the critical activities along the treatment path. The complete approach towards the defined research problem is showed as a roadmap in the Figure 2.7 and 2.8.



Accuracy of prediction is evaluated using AZ



**ModCNN**

1. Built upon the architecture of CCNN
2. CCNN was modified to dynamically identify the optimal number of neurons and hidden units.
3. ADALINE circuit is adopted to find the optimal number of neurons and hidden units.
4. ADALINE circuit uses LMS algorithm for finding the right patterns of neurons and hidden units.
5. LMS circuit with the help of gradient descent find the patterns in an optimal way.
6. Though, ModCNN showed higher efficiency than ANN and CCNN was less prominent when the features size was increased over 100.
7. This was overcome by adopting master-slave model, thus making ModCNN more optimized.

Prediction is tested by:

1. Chi-squared test
2. Relative risk of each different spectrum of GSD
3. Accuracy of prediction is tested by AZ

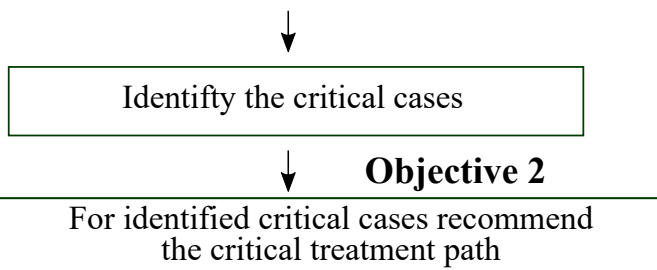


Figure 2.7: Roadmap of the Problem approach

## Objective 3: Application of process mining

Complicated cases identified by ModCNN

Recommend the critical treatment path

Construct the annotated transition system

*Aim of annotated transition system is to*

1. find and recommend the future state of execution

|  |
|--|
| $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow F \rightarrow ?$<br>Partrial trace $\rightarrow$ Future state |
|--|

2. Identify the adequate resources

1. Identify the next succeeding activity in the future state.

This was achieved by:

1. **Activity metric:** To identify the waiting time at each activity
2. **Transition metric:** To identify the performance of activity and resource at different position of execution
3. **Causal metric:** Build a causal relationship between the succeeding and preceeding activity. This is needed to find the reason of occurance of an activity

2. On identifying the activity to succeed the current state, we measure the cost of occurance of that activity using TDABC

3. On identifying the activity to succeed the current state, we need to find the efficient resource who is available and capabale of performing the recommended activity.

1. **Theory of Arousal:** Using this theory proposed by yerkes-dodson, we identified the optimal load of each resource where his performance is better. Based on this finding the recommendation was made.

2. **Analytic Hierarchy Process:** The concept of AHP was used to rank the resources. This means that higher the rank of the reource, his performance is higher.

3. Hence, Based on ranking and availability, the adequate resource was recommended to perform the activity in the future state.

Figure 2.8: Roadmap of the Problem approach

# Chapter 3

## Framework and Study Material

The proposed study aimed in identifying the risk level of each patient at the time of admission. Our study population included the patients with abdominal pain, dyspeptic symptoms, and evidence of gallstone in radiography, USG, CT. The patients with pancreatitis who were not clinically improving, underwent Contrast-Enhanced Computed Tomography (CECT) abdomen to rule out severe pancreatitis. The percentage of each observation of these investigations are represented as stack bar in the Figures 3.1,3.2,3.3,3.4 and 3.5. In these figures the length of stack bar shows the percentage of observation and is shown on top of the bar. Patients with chronic alcohol abuse, elevated Renal Functional Test (RFT), salivary gland pathology, consuming drugs causing pancreatitis, concomitant abdominal conditions like a perforated peptic ulcer, mesenteric vascular occlusion, intestinal obstruction and who had undergone ERCP in the past, were excluded from the study. The pie chart in the Figure 3.6 shows the classification of different spectrum of GSD. The treatment procedure of these patients were recorded using EHR system for management of GSD.

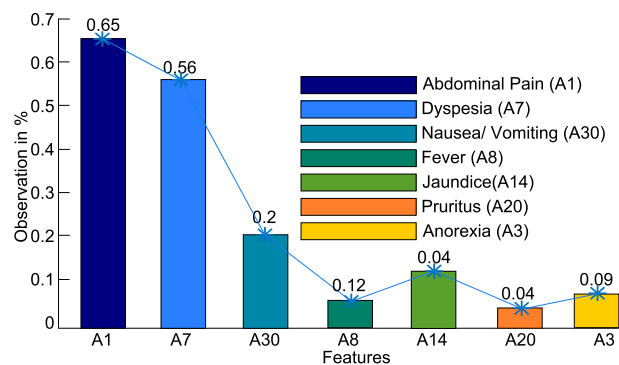


Figure 3.1: Clinical readings showing SYMPTOMS observed through lab investigations in the study cases.

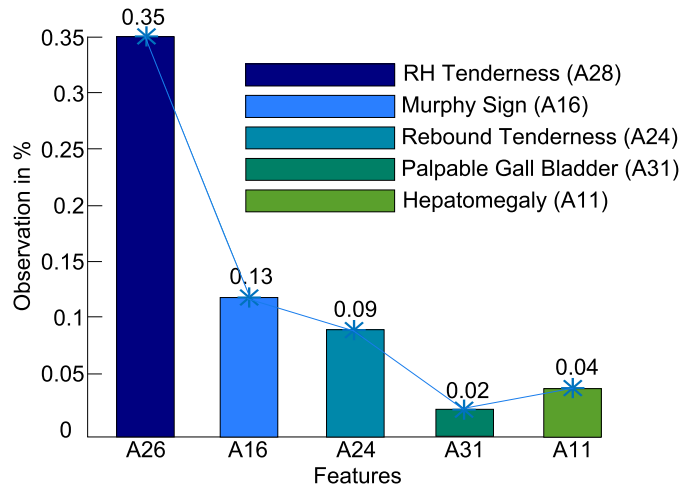


Figure 3.2: Clinical readings showing SIGNS observed through lab investigations in the study cases.

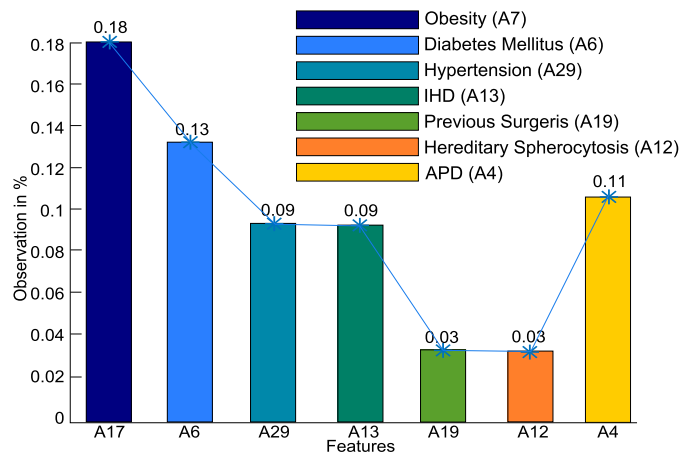


Figure 3.3: Clinical readings showing COMORBID Conditions observed through lab investigations in the study cases.

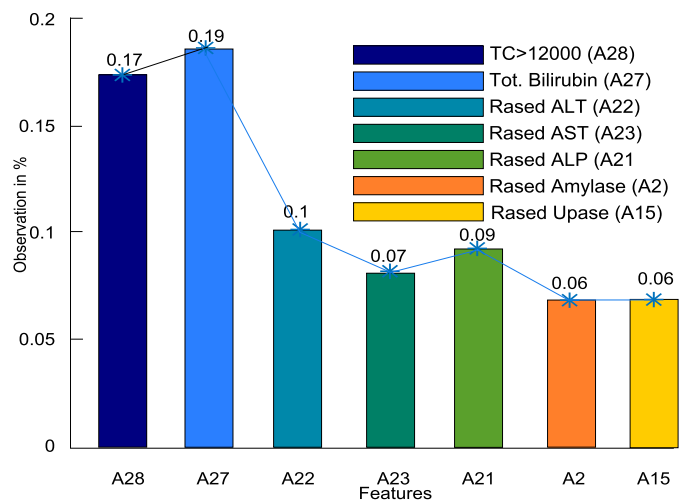


Figure 3.4: Clinical readings showing TESTS conducted on the study cases.

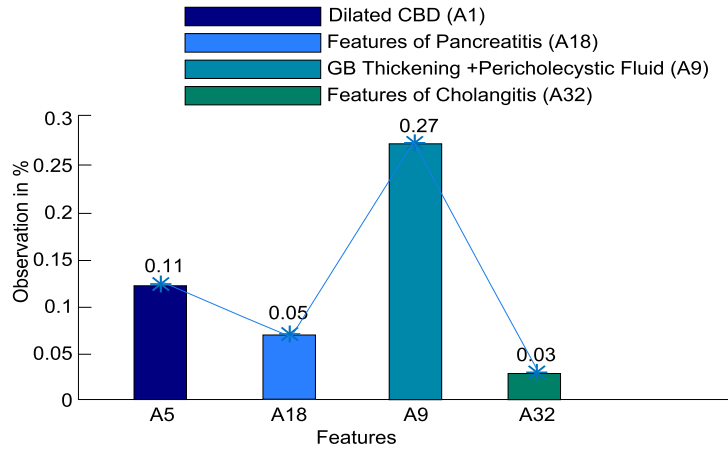


Figure 3.5: USG Findings

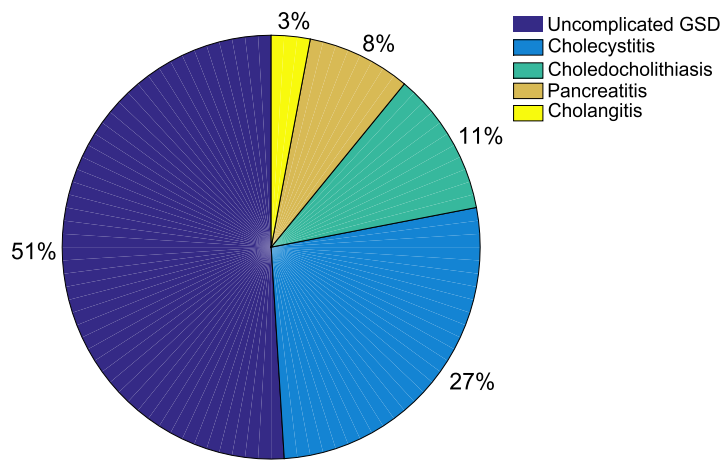


Figure 3.6: Classification result of GSD patients

### 3.1 Experimental Set-up

The Figure 3.7 illustrates how the data from inter-department (pharmacy, laboratory, radiology, and narrative) are processed. Narratives are free text but coded with clinical terms. The information system stores this information in EHR format in a repository. The statistical analysis could be conducted to classify the patients and identify their risk factors using the stored EHR data. This would be helpful if any clinician wants to read a particular patients data. The system would help him in getting the statistical summary of the data. This statistical analysis is illustrated in Figure 3.8. Here the patient’s data associated with the clinical features are considered. The clinical features are diagnosis, medication and laboratory test data. The diagnosis is shown in *pink*, medication in *blue* and laboratory in *green*. This data is fed into ModCNN, which identify the prevalence of significant factors: C2 and C4. Based on the presence of this factors, an expected number

of patients having the significant factors are stratified. Thus finding those cases which may have a higher risk. This would help the clinicians in understanding the patients health status and also provide them with the information about how the disease may progress. In this work, we installed an information system that collected the treatment related data along with the patient’s journey in the hospital. The information collected was converted into EHR format for further analysis in process mining.

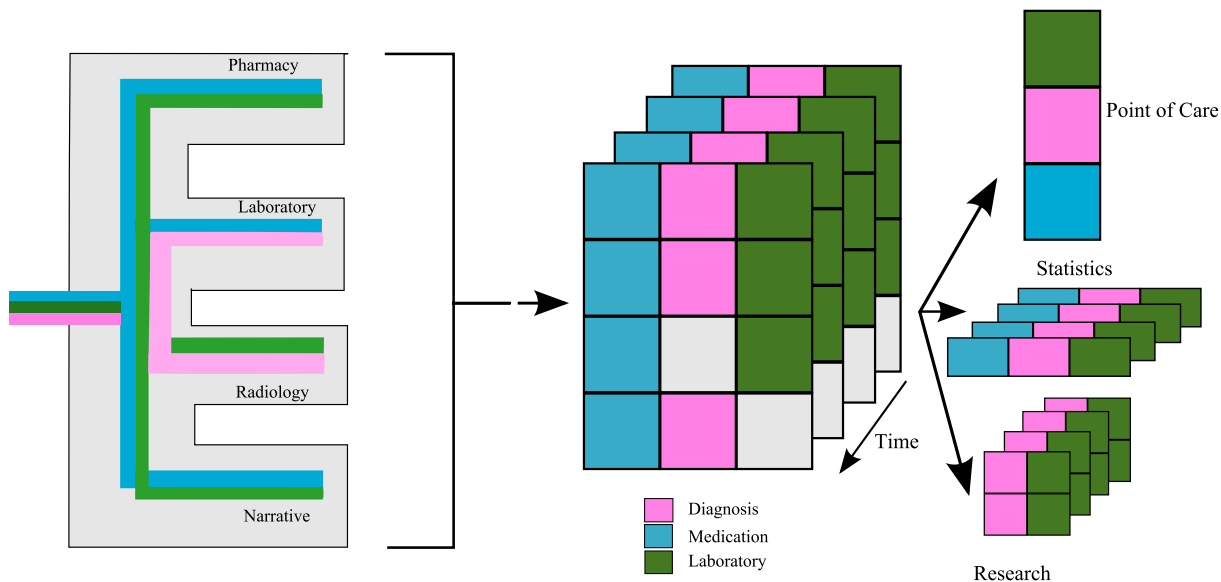


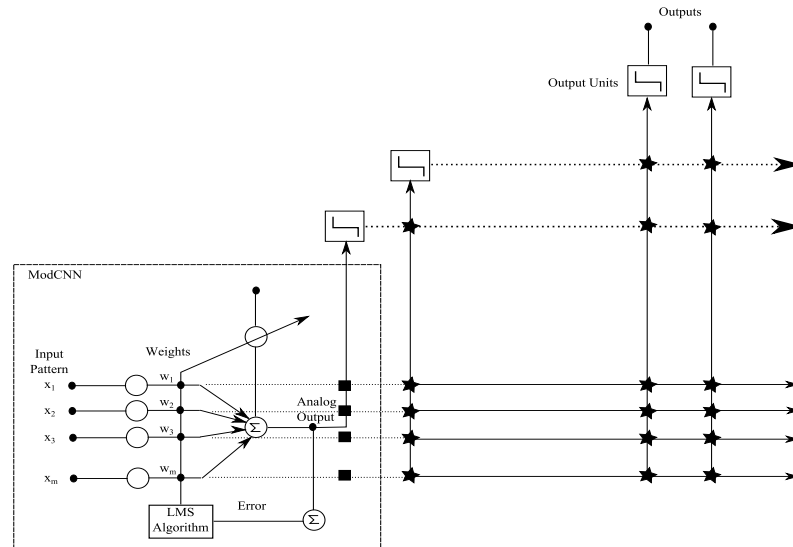
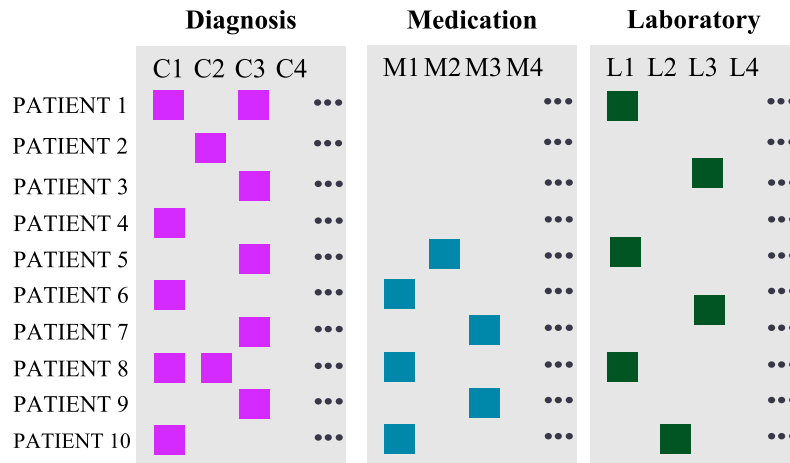
Figure 3.7: Statistical analysis of EHR.

### 3.1.1 Description of ModCNN

ModCNN was built using the architecture of CCNN using Matlab platform. Due to its high computational capability, it was run on workstation built with Intel i7 core processor with the processing speed up to 3.6 GHz. ModCNN was developed in a attempt to address and overcome the slow learning algorithm of ANN and challenge of adding new neurons, when to add them and develop a connection between the neurons as in case of CCNN. From the literature it was studied that CCNN performed better but needed some modification and optimization. In this work we developed a ModCNN, where neurons and hidden units are adapted automatically/ dynamically for giving better accuracy. Here, we examined the performance of ModCNN and compared it with ANN and CCNN. It was observed that ModCNN performed better than other two. This is shown in Figure 3.9

When it was tested for accuracy with varying feature size, it was seen that the Mod-CNN’s accuracy started drifting as the feature size was increased above 100. This information is shown in the Figure 3.10.





**Comorbidity**

|     | C4  | -C4 |      |
|-----|-----|-----|------|
| C2  | 10  | 40  | 50   |
| -C2 | 90  | 860 | 950  |
|     | 100 | 900 | 1000 |

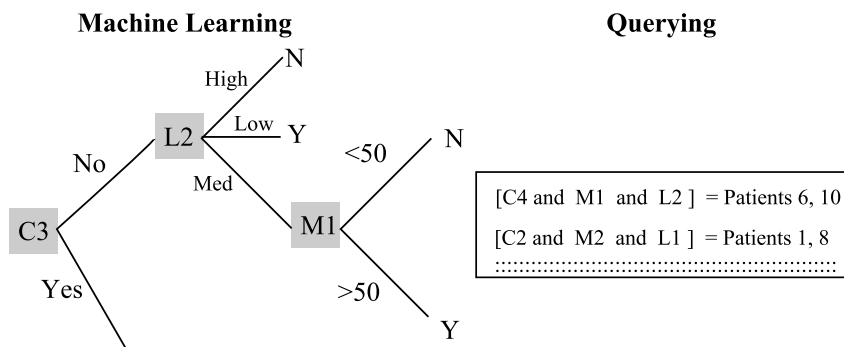
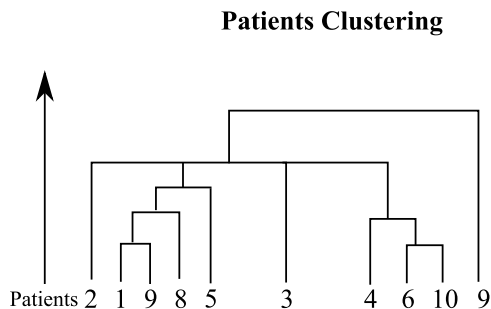


Figure 3.8: Experimental setup for analysis of EHR data using ModCNN.

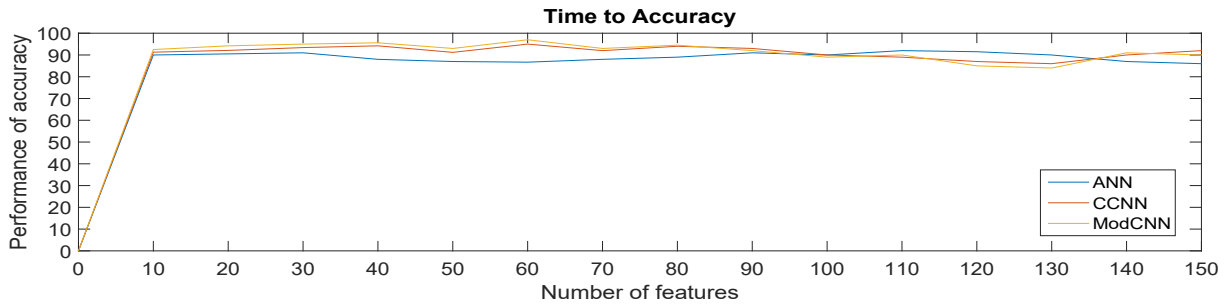


Figure 3.9: Accuracy testing of ANN, CCNN, and ModCNN.

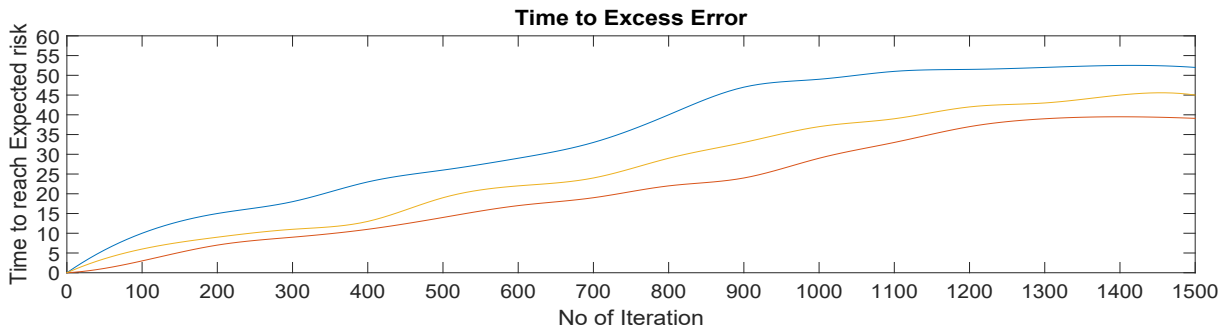


Figure 3.10: Empirical testing of ANN, CCNN, and ModCNN.

To overcome this situation, we parallelized the learning function using *master-slave* model. The Master assigns tasks to slaves with the information about number of neurons, input data and expected result. Slaves built and learns the model. Slaves on reaching the threshold would return number of iteration it took to reach the threshold along with the MSE. Master on receiving this information from slaves would run the gradient descent algorithm to find the optimal set of neurons that would yield least MSE with minimum number of iterations. This master-slave model is shown in Figure 3.11.

## 3.2 Electronic Health Record for Healthcare Process Mining

In Process mining, the sequence of activities are known as trace (Van Oirschot et al., 2014). A patient in his journey within the hospital goes through different activities which are conducted by the hospital staff. Clinical staff are responsible for performing the clinical activities. Administrative based activities such as registration, taking patient to the ward, managing their diet, preparing discharge procedure and so on were performed by non-clinical staff. The information system recorded and maintained the events as EHR. Every activity used the time stamp to add information such as waiting and service time, along with resource information. The occurrence of an event in a healthcare system, not only

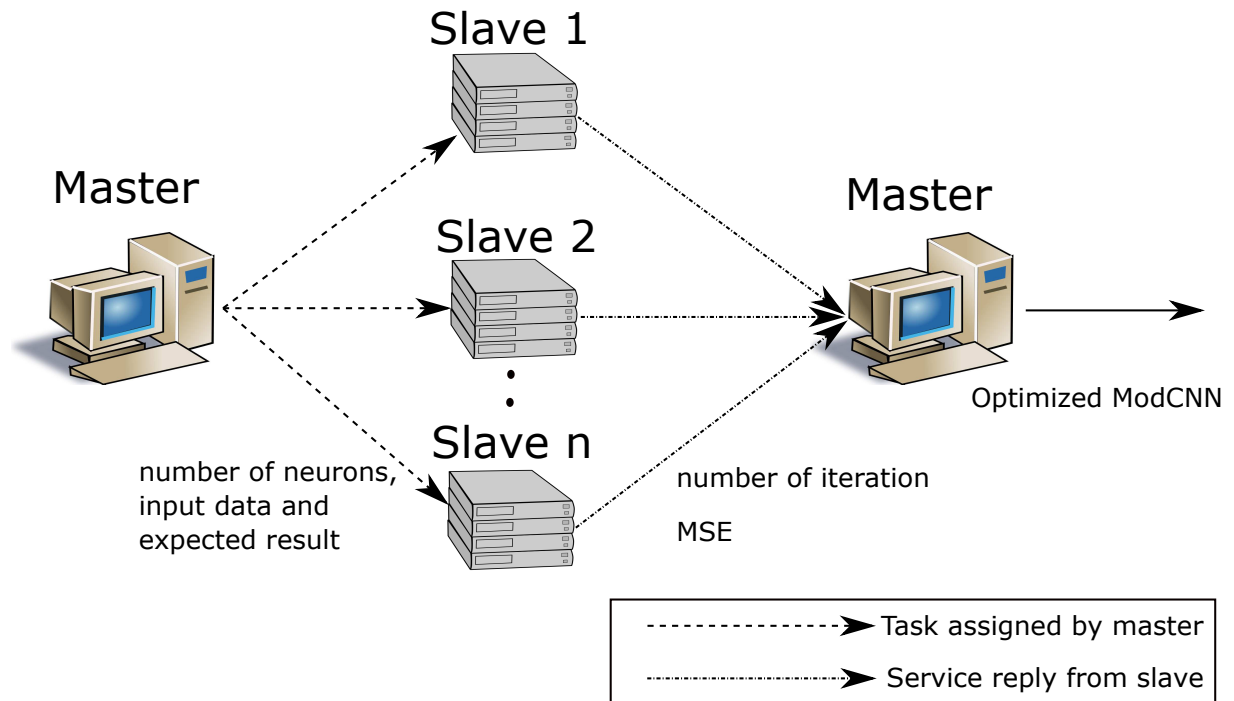


Figure 3.11: Illustration of master-slave model.

depends on the completion of the previously assigned task, but also need to consider the patient's health condition and his/ her response towards the treatment (Partington et al., 2015). Along with this, various other factors such as: sudden change in treatment options based on how patients respond towards the ongoing treatment, shared decision making by multidisciplinary professionals and availability of resources are also to be considered before starting an activity. Hence a healthcare system is a non-trivial as they may not get executed in the way it has been sequenced/ built.

In the current study, the process starts its recording when a patient comes with the complaint of *abdominal pain* at the emergency care unit. The patient is then advised to undergo some lab investigations, to get the parameters mentioned in the Figure 3.1,3.2,3.3,3.4 and 3.5. ModCNN is run on the vitals to retrieve the information about the patients health condition. Using this information, the process mining technique recommends different treatment path. The process ends with a definitive treatment management based on the disease progression. The complete model consisted 23 events with 575 traces and 58 resources. Here the trace is the sequence of events and resources are the people conducting those events. The goal of the proposed technique is to identify the trace match for a partial trace  $\sigma$  and recommend the path of execution along with an efficient resource who can handle the assigned task.

### 3.3 Trace Clustering and Trace Matching

Traces were needed to be clustered in-order to complete the partial trace. The clustering helps in identifying the remaining time and recommending the optimal path for the partially executed trace. The overall distribution of the traces based on the time taken for the completion of an assigned task is shown in the Figure 3.12. And the clustered output showing the traces that took the best, better and good time for the completion of an entire process is shown in Figure 3.13, 3.14, and 3.15 respectively. The K-means clustering algorithm clustered the traces that took an time duration from 0 – 36 *hours* as the *best traces*. Similarly the traces that took the duration of *upto 85 hours* were clustered as *better* and the traces that took more than 82 *hours* were considered as *good*.

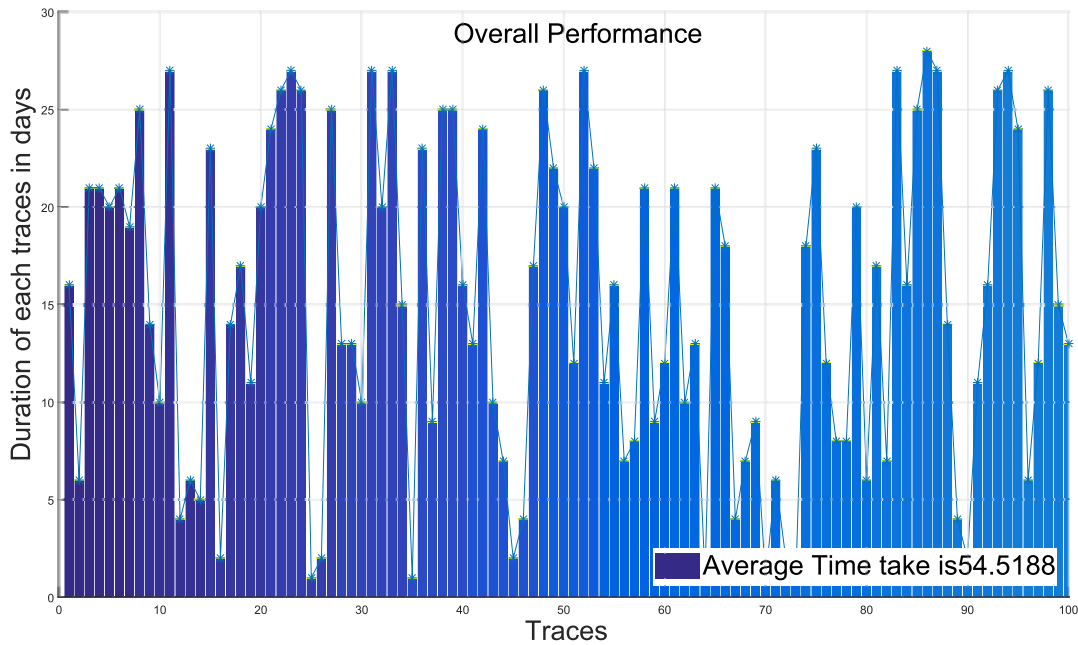


Figure 3.12: Bar chart showing overall distribution of traces based on the duration.

The distribution of activities based on the position of their occurrence is shown in the Figure 6.13. This information is extracted from the event logs recorded. On running Longest Common Subsequence (LCS) algorithm on the distributed set of activities, we identified the sequence of occurrence that repeatedly occurred and is shown in the Figure 6.14. The common sequence of events that repeatedly occur is known as variant. The traces occurring in the variants are clustered as good, better and best cluster and is shown in the Figure 6.15. The knowledge extracted from this information is useful for matching the sequence for a partial trace  $\sigma$ . Using this discovered sequence of activities,

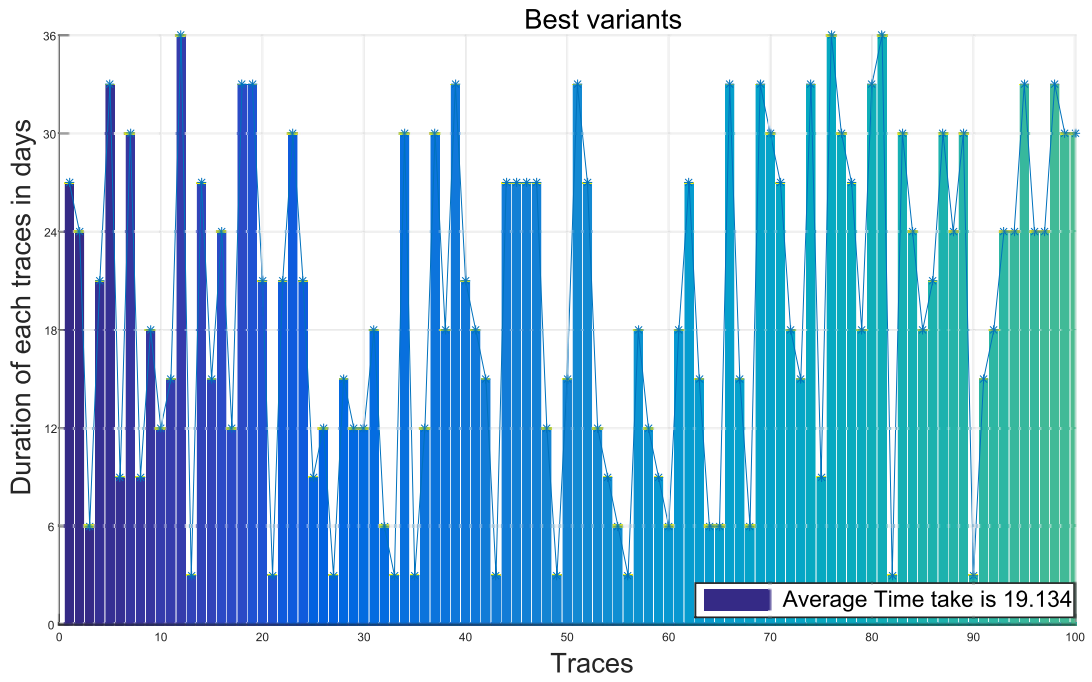


Figure 3.13: Bar chart showing the cluster of BEST traces.

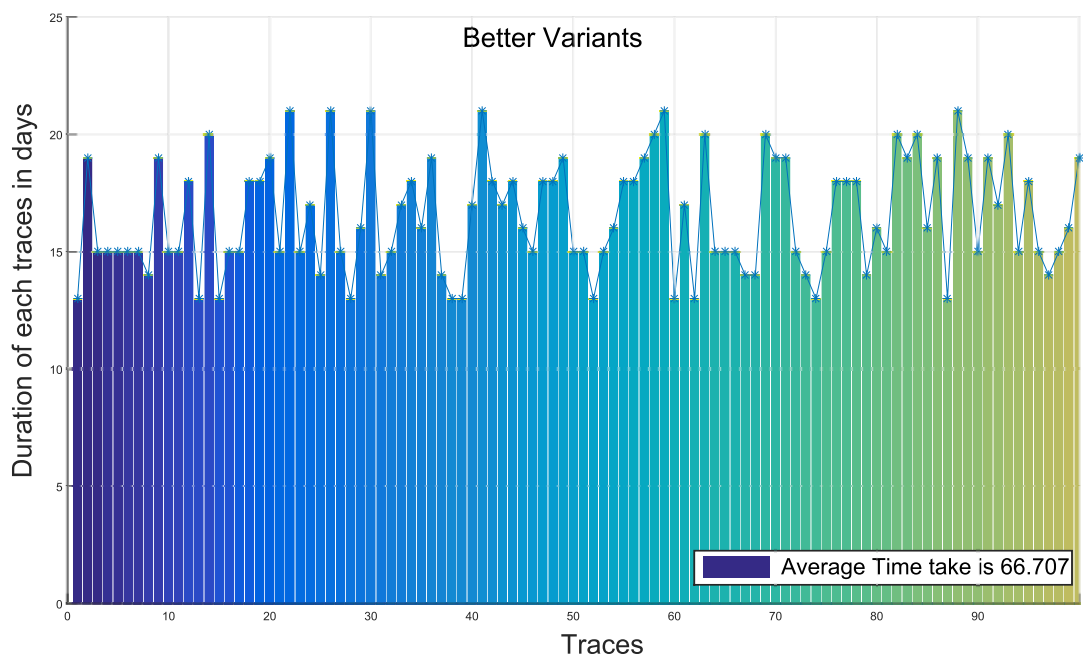


Figure 3.14: Bar chart showing the cluster of BETTER traces.

an alternative path of execution was recommended if any delay in process execution was observed.

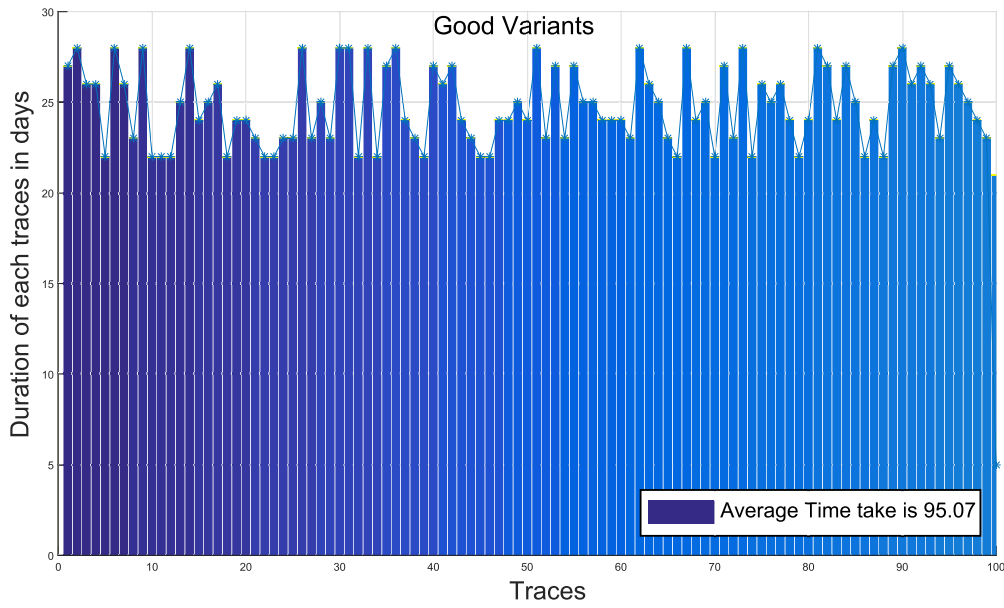


Figure 3.15: Bar chart showing the cluster of GOOD traces.

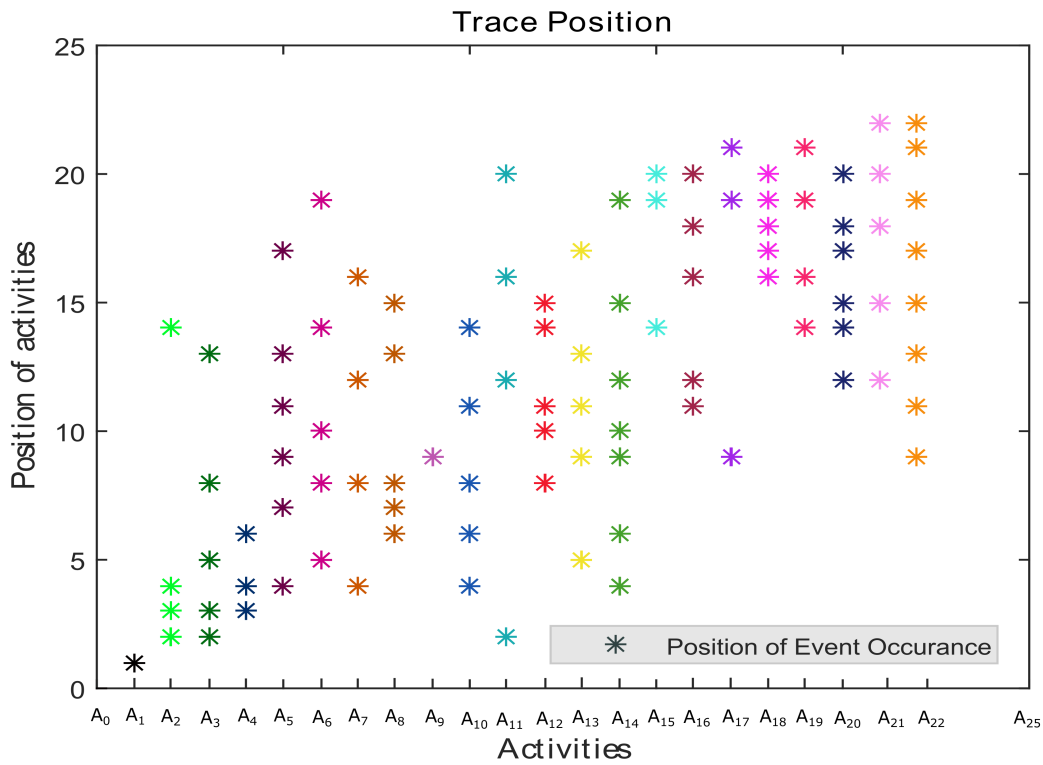


Figure 3.16: Activity position distribution showing the execution of activities.

### 3.4 Control-Flow: A Causal Relationships

An activity is the minimum requirement for discovering the control-flow perspective of a process. The control-flow shows the *causal relationships between activities in a process*. Control-flow model depicting the causal relationship between activities of hospital treat-

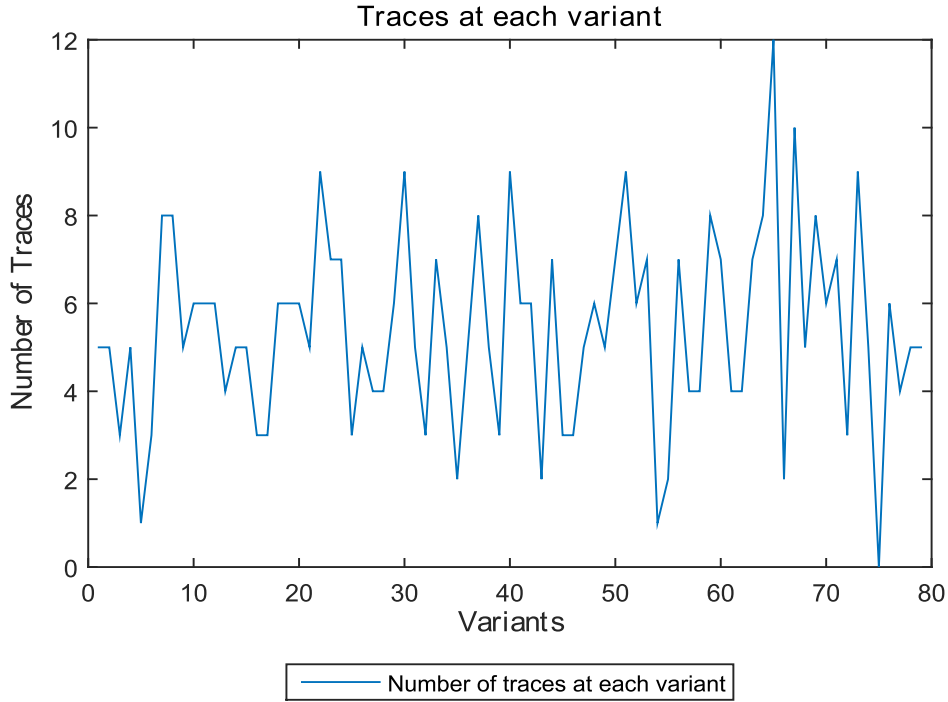


Figure 3.17: Traces at each variants.

ment process is shown in Figure 3.19. It is built using Petri net notations <sup>1</sup>. According to control flow process model shown in Figure 3.19, a patient has to take an *appointment/registration* (AR) and visit the hospital, based on the condition of patient, the case is considered as *out patient* (OP) or *in patient* (IP). Concurrently, the *check history*(CH) of the patient is verified. On the basis of verification outcome, the *decision making* (DM) is taken either to *begin treatment* (BT) or *discharge* (D) the patient. If the case still needs further evaluation, case is *re-examined* (RE) and the process is repeated.

### 3.5 Representing and Storing Event Log: Mining eX-tensible Markup Language

Until recently, Mining eXtensible Markup Language (MXML) (shown in listing 3.1) and its variant such as Semantically Annotated Mining eXtensible Markup Language (SA-MXML) are the de-facto standards for storing and exchanging event logs in digital format. Based on many practical limitations with MXML (and SA-MXML), the eXtensible Event Streams (XES) format has been accepted as a standard event log format. XES has been made less restrictive and truly extensible.

<sup>1</sup>A Petri net is a triplet  $N = (P, T, F)$  where  $P$  is a finite set of places,  $T$  is a finite set of transitions such that  $P \cap T = \phi$ , and  $F \subseteq (P \times T) \cup (T \times P)$  is a set of directed arcs, called the flow relation.

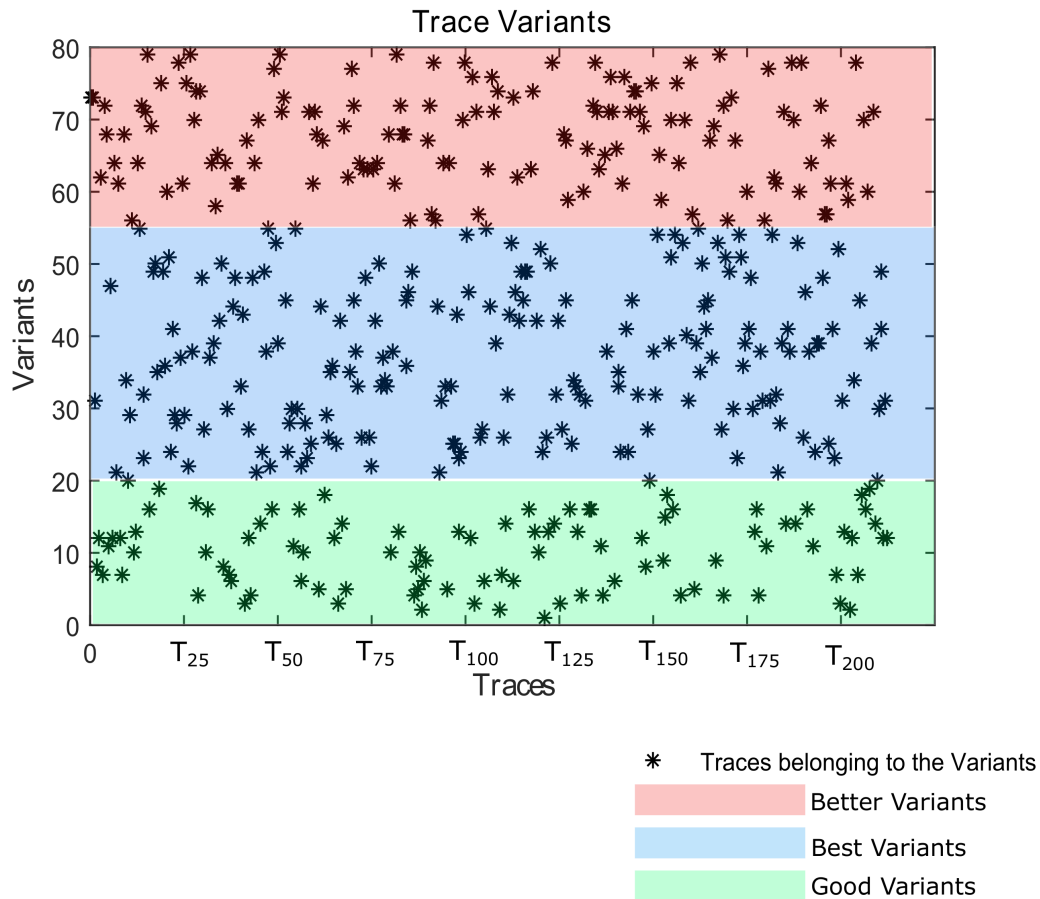


Figure 3.18: cluster of variants showing the traces belonging to different variants.

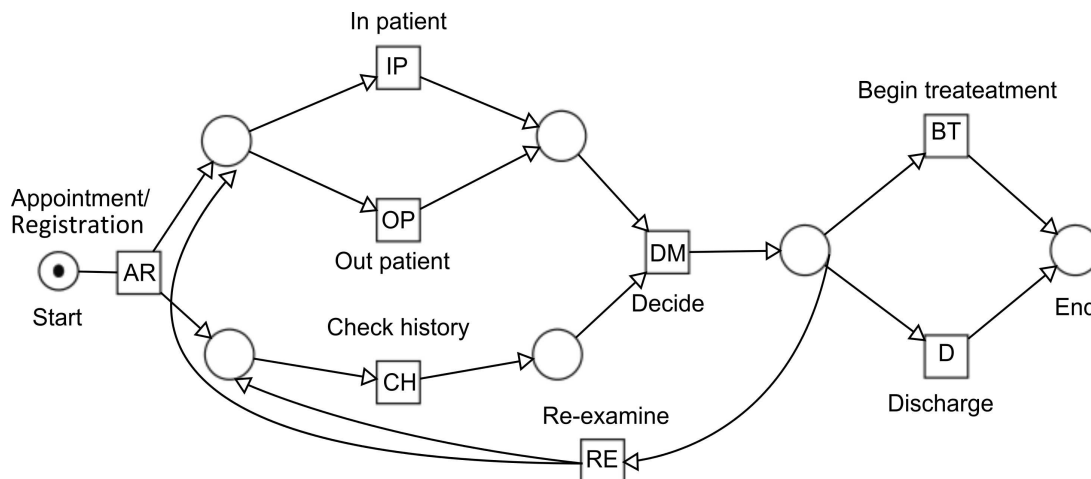


Figure 3.19: Petri-net model of hospital treatment process.

An MXML log starts and ends with the WorkflowLog tag. Each WorkflowLog can contain information related to one or more processes (represented by Process tag). Each process tag can have many cases (represented by ProcessInstance tag), and it can contain any number of events (represented by ProcessInstance tag). For example, hospital admission MXML file shown in the listing 3.1 consists of one event log with the information of



a single process. The process consists of a single case with six events. Each case (ProcessInstance) can be identified by a unique attribute called id. Each event under a case can have a various attribute such as (Id, EventType, TimeStamp, OriginatorId, etc.).

General structure of XES meta-model is shown in Figure 3.20. An XES document contains one event log with any number of cases and attributes which can be nested. XES permits the usage of five basic data types, namely, Boolean, Integer, String, Date, and Float for the standard built-in data types of XML `xs:boolean`, `xs:int`, `xs:string`, `xs:dateTime`, and `xs:float` respectively. The attribute that is mandatory should be declared as global. Listing 3.2 shows the part of event log related to hospital admission process in XES format. An XES document contains one event log with any number of cases and attributes which can be nested. In the example XES log, three extensions are declared: *Concept*, *Time*, and *Organizational*. For each of these extensions, a shorter prefix is given. These prefixes are used in the attribute names. For example, the *Time* extension defines an attribute `timestamp`. It also specifies two lists of global attributes. Traces have one global attribute: attribute `concept:name` is mandatory for all traces. Events have three global attributes: attributes `time:timestamp`, `concept:name` and `org:resource` are mandatory for all events. Further, classifiers classify the events based on the attributes. For example, `e = #resource(e)` classifies events based on the resource executing the event.

- *Concept*: defines the name attribute for traces and events. For traces, the attribute typically represents some identifier for the case. For events, the attribute typically represents the activity name.
- *Time*: defines the timestamp attribute for events.
- *Organization*: defines three standard attributes for events: `resource`, `role`, and `group`. The `resource` attribute refers to the resource that triggered or executed the event. The `role` and `group` attributes characterize the (required) capabilities of the resource and the resource's position in the organization.
- *Lifecycle*: defines the transition attribute for events, possible values of this attribute are `schedule`, `start`, `complete`, `auto skip`, etc.
- *Semantic*: defines the model reference attribute for all elements in the log. This is used for pointing to concepts in the ontology. For example, if there is any ontology

Listing 3.1: Event log of hospital treatment process in MXML format.

```

<WorkflowLog ... >
<Source program="Hospital process"/>
<Process ... >
<ProcessInstance id="xx12">
<AuditTrailEntry>
<Data>
<Attribute name="Costs">200</Attribute>
</Data>
<WorkflowModelElement>AR</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>10-10-2012T01:00</Timestamp>
<Originator>Pete</Originator>
</AuditTrailEntry>
<AuditTrailEntry>
<Data>
<Attribute name="Costs">130</Attribute>
</Data>
<WorkflowModelElement>IP</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>10-10-2012T01:02</Timestamp>
<Originator>Sean</Originator>
</AuditTrailEntry>
<AuditTrailEntry>
<Data>
<Attribute name="Costs">230</Attribute>
</Data>
<WorkflowModelElement>CH</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>10-10-2012T01:05</Timestamp>
<Originator>Sue</Originator>
</AuditTrailEntry>
<AuditTrailEntry>
<Data>
<Attribute name="Costs">340</Attribute>
</Data>
<WorkflowModelElement>Decide</WorkflowModelElement>
<EventType>DM</EventType>
<Timestamp>10-10-2012T01:11</Timestamp>
<Originator>Sara</Originator>
</AuditTrailEntry>
<AuditTrailEntry>
<Data>
<Attribute name="Costs">280</Attribute>
</Data>
<WorkflowModelElement>D</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>10-10-2012T01:20</Timestamp>
<Originator>Pete</Originator>
</AuditTrailEntry>
</ProcessInstance>
<ProcessInstance id="xx13">
<AuditTrailEntry>
<Data>
<Attribute name="Costs">170</Attribute>
</Data>
<WorkflowModelElement>AR</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>15-10-2012T01:00</Timestamp>
<Originator>Pete</Originator>
</AuditTrailEntry>
...
</ProcessInstance>
...
</Process>
</WorkflowLog>

```

Listing 3.2: Event log of hospital treatment process in XES format.

```

<?log xes.version="1.0" xes.features="nested-attributes">
<extension name="Concept" prefix="concept" uri="http://.../concept.xesext"/>
<extension name="Time" prefix="time" uri="http://.../time.xesext"/>
<extension name="Organizational" prefix="org" uri="http://.../org.xesext"/>
<global scope="trace">
<string key="concept:name" value="name"/>
</global>
<global scope="event">
<date key="time:timestamp" value="2012-09-16"/>
<string key="concept:name" value="name"/>
<string key="org:resource" value="resource"/>
</global>
<classifier name="Activity" keys="concept:name"/>
<classifier name="Resource" keys="org:resource"/>
<classifier name="Both" keys="concept:name org:resource"/>
<trace>
<string key="concept:name" value="xx12"/>
<event>
<string key="concept:name" value="AR"/>
<string key="org:resource" value="Pete"/>
<date key="time:timestamp" value="10-10-2012T01:00"/>
<string key="Event_ID" value="2342"/>
<string key="Costs" value="130"/>
</event>
<event>
<string key="concept:name" value="IP"/>
<string key="org:resource" value="Sean"/>
<date key="time:timestamp" value="10-10-2012T01:02"/>
<string key="Event_ID" value="2343"/>
<string key="Costs" value="230"/>
</event>
<event>
<string key="concept:name" value="CH"/>
<string key="org:resource" value="Sue"/>
<date key="time:timestamp" value="10-10-2012T01:05"/>
<string key="Event_ID" value="2344"/>
<string key="Costs" value="340"/>
</event>
<event>
<string key="concept:name" value="DM"/>
<string key="org:resource" value="Sara"/>
<date key="time:timestamp" value="10-10-2012T01:11"/>
<string key="Event_ID" value="2345"/>
<string key="Costs" value="280"/>
</event>
<event>
<string key="concept:name" value="D"/>
<string key="org:resource" value="Pete"/>
<date key="time:timestamp" value="10-10-2012T01:20"/>
<string key="Event_ID" value="2346"/>
<string key="Costs" value="170"/>
</event>
</trace>
<trace>
<string key="concept:name" value="xx13"/>
<event>
<string key="concept:name" value="AR"/>
<string key="org:resource" value="Pete"/>
<date key="time:timestamp" value="15-10-2012T01:00"/>
<string key="Event_ID" value="2347"/>
<string key="Costs" value="200"/>
</event>
...
</trace>
...
</log>

```

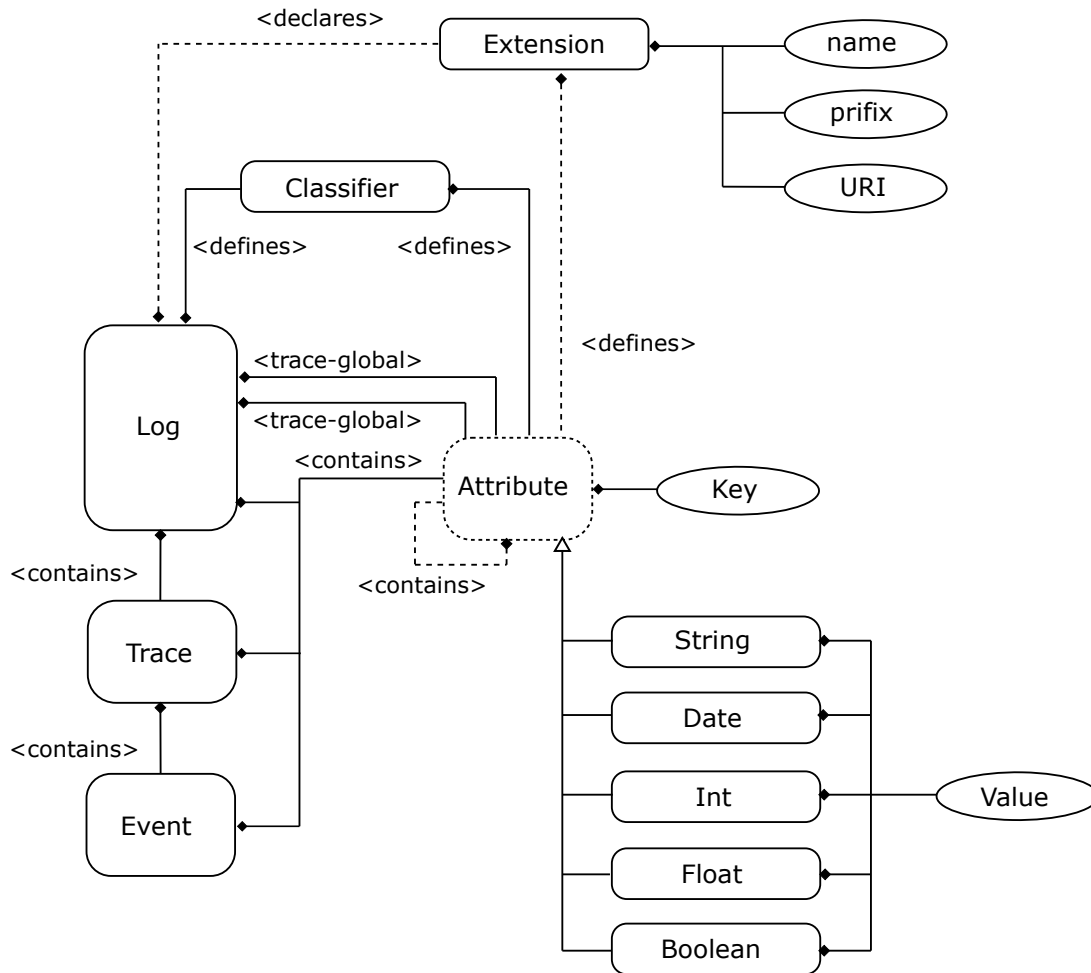


Figure 3.20: Event log structure in XES format.

explaining various classes of memberships, for example, Platinum, Gold, and Silver. Using semantic extension, any given trace can refer to the suitable element in the ontology for classifying the memberships.

# Chapter 4

## Modified Cascade Neural Network (ModCNN)

The proposed system is a stochastic learning tool to analyse the disease behaviour and predict its progression. For this, the significant risk factors associated with the disease were first identified in a supervised way. And, using these risk factors the model learns the disease behaviour and helps in prediction. The complete learning process is illustrated in the Table 4.1. In the Table 4.1,  $\{A, B, C, D, E\}$  are the features associated with the disease and  $\{1, 2, 3, 4, 5, 6, 7\}$  are the cases. Among the features,  $\textcircled{B}$ ,  $\textcircled{D}$ ,  $\textcircled{E}$  are identified as significant factors and the check mark ( $\checkmark$ ) indicates the presence of feature in the respective cases. The model classified the data set into  $\{Not\ Present, At\ the\ Edge, Partly\ Present\ and\ Present\}$ . Just identifying significant factors don't solve an issue of predicting the disease behaviour. For this, we need to categorize the data and then perform appropriate statistical analysis.

Table 4.1: Illustration of feature selection and classification

| Features $\rightarrow$ |   |                   |   |                   |                   |                |
|------------------------|---|-------------------|---|-------------------|-------------------|----------------|
| Cases $\downarrow$     | A | $\textcircled{B}$ | C | $\textcircled{D}$ | $\textcircled{E}$ | Result         |
| 1                      |   | $\checkmark$      |   |                   |                   | At the Edge    |
| 2                      |   |                   |   |                   |                   | Not Present    |
| 3                      |   | $\checkmark$      |   | $\checkmark$      |                   | Partly Present |
| 4                      |   |                   |   |                   |                   | Not Present    |
| 5                      |   | $\checkmark$      |   |                   | $\checkmark$      | Partly Present |
| 6                      |   |                   |   |                   |                   | Not Present    |
| 7                      |   | $\checkmark$      |   | $\checkmark$      | $\checkmark$      | Present        |

This chapter explains the architecture and functioning of ModCNN. A systematic analysis is conducted to understand and compare the efficiency of ANN, CCNN, and ModCNN in predicting the disease progression. On analysis and from the study, it was observed that ANN was well applied for disease evaluation. But, we understood that ANN is a slow learner when compared to CCNN. In this work, we modified CCNN and made it more efficient.

## 4.1 Introduction

The synaptic connections in the human brain are unique for every individual and are not entirely inherited. As a learning process, this synaptic connection iteratively trains itself (Fukushima, 1975). The Carnegie Mellon University experimented on neurons and studied that every feedback from different part of the human body is purely due to neurons in the brain, which are connected by this synaptic connections. Hence, as a learning process since birth, this feedback system trains the brain to interpret and understand sensory stimuli. It is impossible by any convention physiological experiment to understand the mechanism of pattern recognition inside the brain. But in 1943, the neurophysiologist Warren and McCulloch and a mathematician Walter Pitts (McCulloch and Pitts, 1943) understood this mechanism. Using this knowledge of neurons, they introduced its functionality and modeled simple neural network. Since then, a lot of research has been done. Due to the recent advancement in computational units, there is resurgence in neural network.

## 4.2 Artificial Neural Network

ANNs are statistical tool inspired by the functioning of nervous system. They are processed with a set of computational units known as input and output units. These units are interconnected by hidden units via a set of weights. The synaptic connection builds an electric circuit through which signals are processed from input to output. On obtaining the output, its weighted sum of signals is calculated and compared with the threshold. The threshold is a condition to break down processing of computation. If this threshold exceeds the defined limit, then nodes in unit fires, else remain inactive.

### 4.2.1 Flow of information in ANN

The structure of neural network is shown in Figure 4.1 and has three main units: *input unit* as *layer 1*, *hidden unit* as *layer 2* and *output unit* as *layer 3*. Activation of hidden unit is shown in equation 4.2, where the  $\theta_{11}$  is the weight from input unit 1 to hidden unit 1 and  $x_1$  is the input 1. The output of hidden unit is shown in equation (4.3), where  $\frac{1}{1+e^{-\lambda * I_H^1}}$  is a sigmoid function with  $\lambda$  being the learning rate and  $I_H^1$  is an activation of input to hidden unit. On obtaining the output from hidden unit, the input to output is activated by equation (4.4), where  $O_H^1$  is the output from hidden layer and  $\theta_1$  is the weight from hidden unit to output. On activating the input to output unit, the output is obtained by equation (4.5). Thus the output  $h_\theta(x)$  is obtained using  $O_o$ , from which the error  $\varepsilon = h_\theta(x) - y$  is calculated, where  $y$  is the desired output.

ANNs are trained with the cases of the known outcome, during which the weights interconnecting the neurons are adjusted using the backpropagation technique. Some hidden layers depends on the complexity of the problem. Addition of hidden layer may increase or decrease the accuracy. The computation process aims to train the neurons using backpropagation technique and find the optimal interconnecting weights. This entire process of training is known as learning. Learning is usually achieved by decreasing the error function between the input and actual targeted output. The process is iterated to identify the patterns associated with the outcome. Unlike other statistical analysis techniques as mentioned earlier, ANNs are not affected by the low frequencies in the input pattern (Lippmann and Shahian, 1997; Tu et al., 1998). Further, ANN works better as they use, in general, non linear functions of data (the activation function in ANN are typically non linear functions).

Weights from *input to hidden-neurons* and *hidden-neurons to output* are initialized as  $[W]_{k \times n}^1$  and  $[V]_{n \times p}^1$  respectfully, where  $k$  is number of inputs,  $n$  is number of neurons and  $p$  is number of outputs.  $[W]_{k \times n}^1$  and  $[V]_{n \times p}^1$  are matrices and are shown in equation 4.1.

$$[W]_{k \times n}^1 = \begin{bmatrix} w_1^1 & w_2^1 & \dots & w_n^1 \\ w_1^2 & w_2^2 & \dots & w_n^2 \\ \vdots & \vdots & \dots & \vdots \\ w_1^k & w_2^k & \dots & w_n^k \end{bmatrix} \quad [V]_{n \times p}^1 = \begin{bmatrix} v_1^1 & v_2^1 & \dots & v_p^1 \\ v_1^2 & v_2^2 & \dots & v_p^2 \\ \vdots & \vdots & \dots & \vdots \\ v_1^n & v_2^n & \dots & v_p^n \end{bmatrix} \quad (4.1)$$

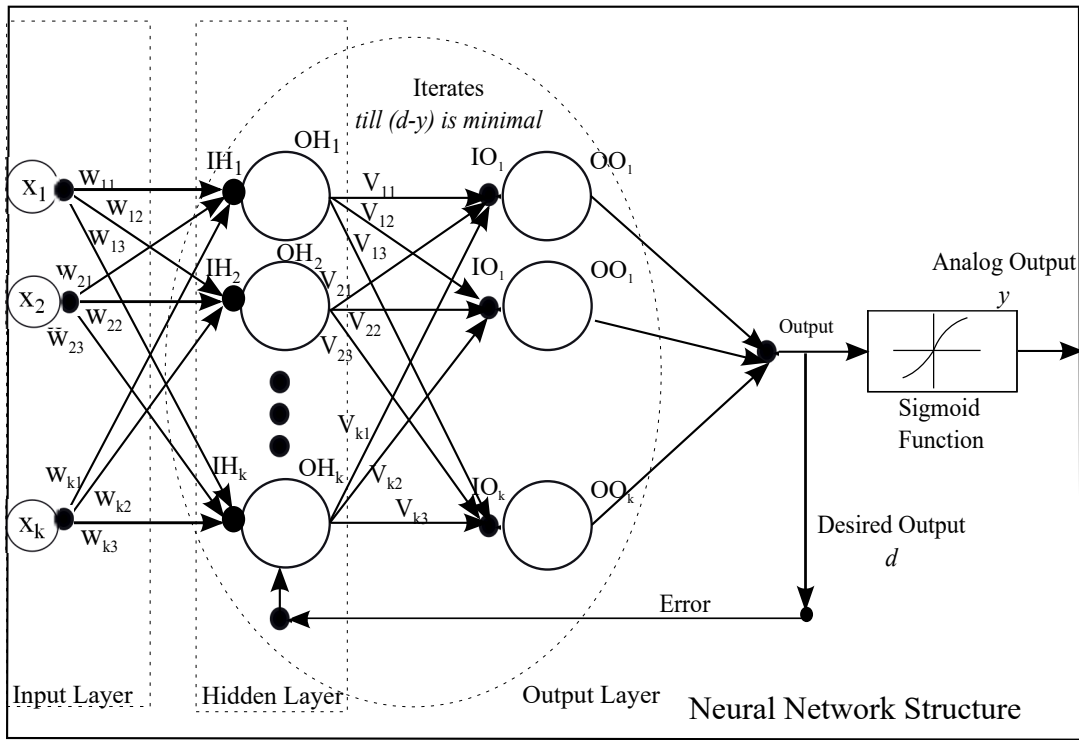


Figure 4.1: Neural Network model

Input to Hidden Layer

$$I_H^1 = x_1\theta_{11} + x_2\theta_{21} + \dots + x_k\theta_{k1}$$

$$I_H^2 = x_1\theta_{12} + x_2\theta_{22} + \dots + x_k\theta_{k2}$$

.....

$$I_H^k = x_1\theta_{1k} + x_2\theta_{2k} + \dots + x_k\theta_{kk} \quad (4.2)$$

Output of Hidden Layer

$$O_H^1 = \frac{1}{1 + e^{-\lambda * I_H^1}}$$

$$O_H^2 = \frac{1}{1 + e^{-\lambda * I_H^2}}$$

.....

$$O_H^k = \frac{1}{1 + e^{-\lambda * I_H^k}} \quad (4.3)$$

Input to Output Layer

$$I_O = O_H^1\theta_1 + O_H^2\theta_2 + \dots + O_H^k\theta_k \quad (4.4)$$

Output of Output Layer

$$O_o = \frac{1}{1 + e^{-\lambda * I_o}} \quad (4.5)$$



### 4.3 Cost Function $J(\theta_0, \theta_1)$

Cost function ( $J(\theta_0, \theta_1)$ ) is the cost for identifying the weights  $\theta$ , such that the predicted output  $h_\theta(x^i)$  is close to the desired output  $y^i$  Chu et al. (2007), where  $i$  is an index of the training example  $i = 1, 2, \dots, m$ . The hypothesis for linear regression is given in equation (6.14), where the  $\theta_0$  and  $\theta_1$  are the parameters of a linear regression model,  $x$  is the input and  $\theta^T$  represents the transpose of the weight vector  $\theta$ .

$$h_\theta(x) = \theta_0 + \theta_1 x_1 = \sum_{i=0}^n \theta_i x_i = \theta^T x \quad (4.6)$$

The objective of cost function is to identify the parameter  $\theta$ 's such that the predicted output  $h_\theta(x^i)$  is very close to the desired output  $y^i$ , i.e., the error  $\varepsilon^i = y^i - h_\theta(x^i)$  is minimum. Hence the objective is to *minimize*  $J(\theta_0, \theta_1)$ , i.e., to minimize the cost of identifying  $\theta_0, \theta_1$ .

*Theorem 1.* Let

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^i) - y^i)^2 \quad (4.7)$$

be a cost function of the linear regression model, from which cost function of logistic regression and the neural network is derived.

$$\text{Sigmoid function or Logistic regression model} = g(z) = \frac{1}{1 + e^{-z}}$$

$$\text{where } z = -\theta^T x$$

$$\therefore \text{Hypothesis } h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

Using equation 6.15

$$\text{Cost function } J(\theta_0, \theta_1) = \begin{cases} -\log(h_\theta(x)) & \text{if } y = 1 \\ -\log(1 - h_\theta(x)) & \text{if } y = 0 \end{cases} \quad (4.8)$$

Equation 4.8 rewritten as

Cost function of Logistic regression is =

$$J(\theta_0, \theta_1) = -\frac{1}{m} \sum_{i=1}^m (y \log(h_\theta(x)) + (1 - y) \log(1 - h_\theta(x))) + \frac{\lambda}{2m} \sum_{i=1}^n \theta_j^2 \quad (4.9)$$

On substituting  $y = (0, 1)$  in equation 4.9 the equation 4.8

is obtained and, where  $\frac{\lambda}{2m} \sum_{i=1}^n \theta_j^2$  is known as regularization term.

### 4.3.1 Cost function for ANN

The Cost function of Neural Network is obtained using the theorem 2. The cost function is the cost of identifying the optimal weights required to develop an optimal neural network model.

*Theorem 2.* Neural Network has multiple inputs ( $k$ ),  $L$  number of network layer,  $S_l$  number of units in layer  $l$  and  $m$  training set. Updating equation (4.9) yields the modified cost function for neural network:

*Let the cost function of Logistic regression be*

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m (y \log(h_{\theta}(x)) + (1 - y) \log(1 - h_{\theta}(x))) + \frac{\lambda}{2m} \sum_{i=1}^n \theta_j^2 \quad (4.10)$$

where  $\frac{\lambda}{2m} \sum_{i=1}^n \theta_j^2$  is known as regularization term theorem 1

Since the neural network has  $L$  number of network layers,  $S_l$  number of units in layer  $l$ ,  $k$  input and  $m$  training set

$$\begin{aligned} \therefore \text{Cost function} = J(\theta) = & -\frac{1}{m} \sum_{i=1}^m \sum_{k=1}^k (y_k^i \log(h_{\theta}(x^i))_k + \\ & (1 - y_k^i) \log(1 - h_{\theta}(x^i))_k) + \frac{\lambda}{2m} \sum_{l=1}^{L-1} \sum_{i=1}^{S_l} \sum_{j=1}^{S_{l+1}} (\theta_{ji}^l)^2 \end{aligned} \quad (4.11)$$

## 4.4 Cascade Correlation Neural Network (CCNN)

ANN showed good accuracy and efficiency in the field of medical research along with a lot of other scientific areas. But, as the data grew to become complicated, they became a slow learner. Their back-propagation computation made them slow to achieve their optimality. So (Fahlman and Lebiere, 1990) proposed CCNN. This has shown better accuracy and has overcome computational overlay. In CCNN, the process begins with a minimum number of hidden units and neurons in each hidden unit to maximize the correlation. Figure 4.2 shows the architecture of CCNN.

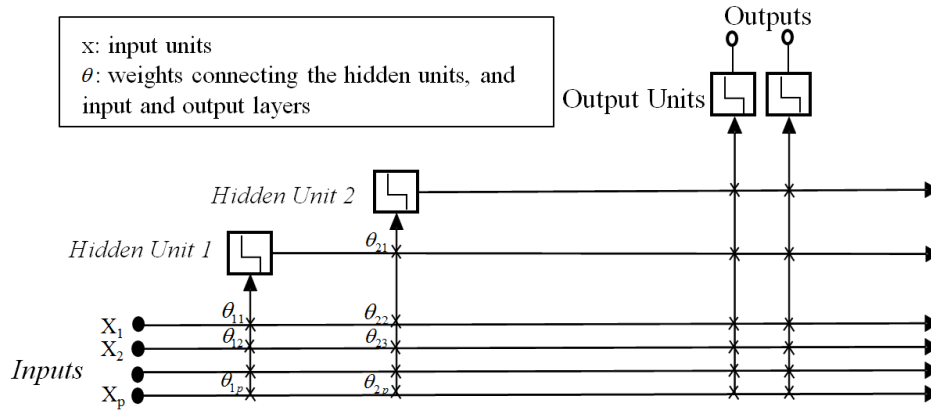


Figure 4.2: Schematic diagram of CCNN architecture .

### 4.4.1 Evolution of CCNN

The main reason for slow learning of ANN was "moving target problem". Instead of moving towards target directly to find the problem solution, it kept dancing as illustrated in Figure 4.3. In the Figure we can observe that CCNN moved towards the target directly with very less number of epochs while ANN took time to converge. As a solution for this, Fahlman and Lebiere (1990) proposed CCNN. CCNN was a cascade-correlation algorithm architecture with cascade correlation of network that learn by experience. In CCNN, weights of all the neurons are updated at once leading to a constant change in hidden units.

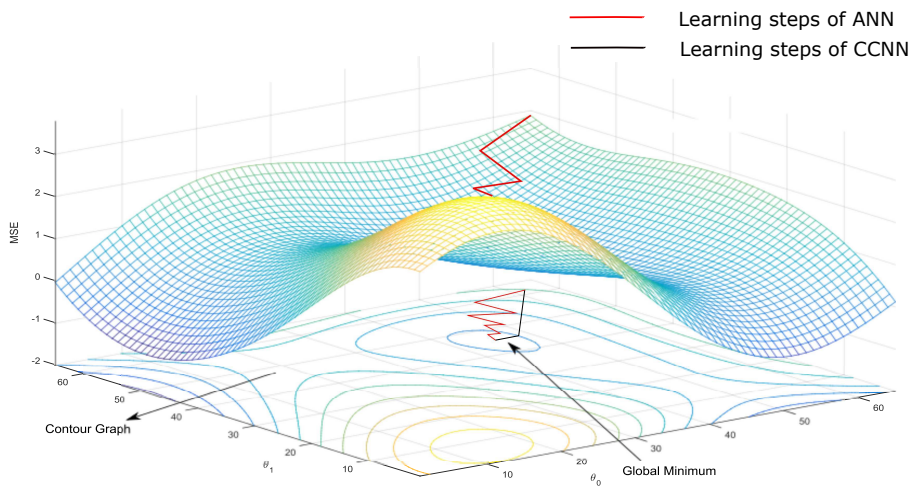


Figure 4.3: Moving target problem illustration of ANN, compared with CCNN

In CCNN, the weights are frozen as the hidden units are added to the network. CCNN proved to be quicker than traditional ANN as it did not use backpropagation algorithm

for adjusting the weights, instead, weights were adjusted to maximize the sum  $S$  over all output units  $O$ . CCNN receives the trainable inputs to its pre-existing hidden units, with no output units connected to it. The input units are then processed by adjusting the weights to maximize the correlation  $C$ . The output  $O$  is the magnitude of correlation between  $V$  (the candidate unit values) and  $\varepsilon_O$  (the residual output error).

$$C = \sum_o \left| \sum_p (V_p - \bar{V})(\varepsilon_{p,O} - \bar{\varepsilon}_O) \right| \quad (4.12)$$

In the equation 4.12,  $p$  is the training pattern, and  $\bar{V}$  and  $\bar{\varepsilon}$  are averaged over  $V$  and  $\varepsilon$  respectively. The objective here is to identify the best combinations of hidden units which maximize  $S$ . To maximize  $S$  the backpropagation rule of taking partial derivation of  $S$  concerning each combination of input weights  $\theta$  is applied.

$$\frac{\delta C}{\delta \theta} = \sum_{P,O} \sigma_O(E_{P,O} - E_O) f'_O I_{i,P} \quad (\text{where } i = 1, 2, \dots, n) \quad (4.13)$$

In the equation 4.13  $\sigma_O$  is the correlation between candidates values and output  $O$ ,  $f'_O$  is derivative of the pattern  $p$  concerning the sum of its inputs and  $X$  is the input. As the objective is to maximize the correlation, gradient ascent is used to identify the maximum of  $C$  after getting  $\frac{\delta C}{\delta \theta_i}$ .

#### 4.4.2 Architecture of CCNN

On comparison and testing, it was noted that CCNN learns faster than backpropagation. In CCNN the architecture is built upon two key ideas: *cascade* and *correlation*. In *cascade* (meaning: the medium of passing information) hidden units are added linearly or sequentially, one at a time and once added it is kept frozen. *Correlation* (meaning: building connection) helps in building up the relationship between the already existing hidden unit and newly added unit. This is achieved by maximizing the magnitude of correlation and by eliminating the error of prediction in the outcome.

Initially, there are no hidden units, building the model only with an input and an output unit. Each node in the input layer is connected to all the nodes in the output unit. The output unit generates a linear or non-linear sum of their input by employing sigmoid activation function. The hidden units are then added linearly. The newly added hidden unit will get a connection from network original input unit and already existing hidden unit (if any hidden unit is added). Addition of new hidden unit would freeze input

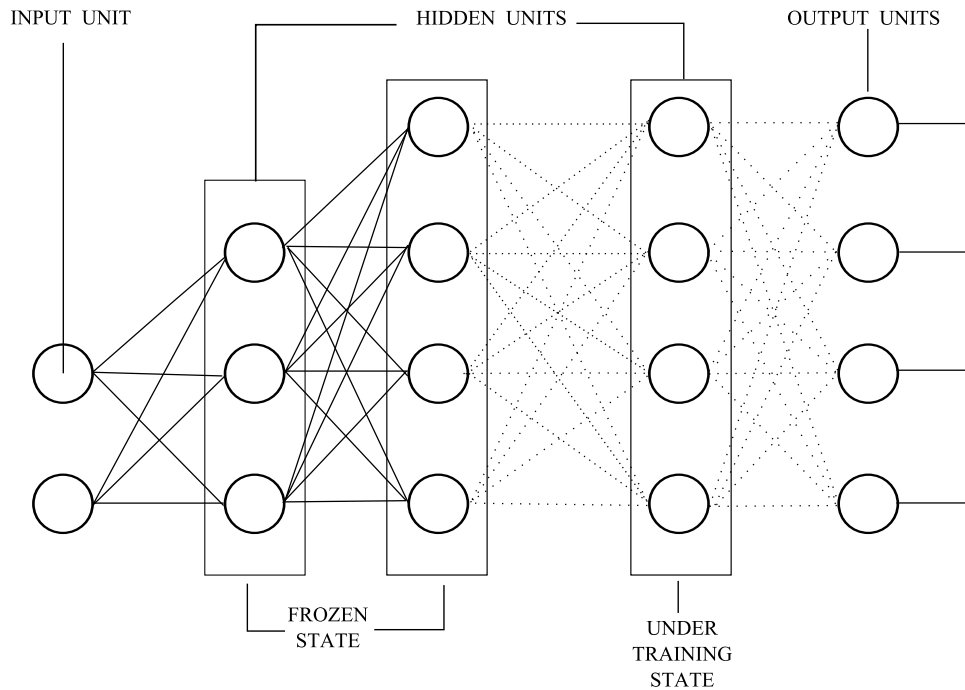


Figure 4.4: Illustration of frozen and training state in CCNN

weights, and output connection is trained now. This means that the model remains under consideration till hidden units optimality is obtained and don't need any more training. Unlike in ANN, at every iteration, all the combination of weights are trained for every epoch. The frozen and training state is illustrated in Figure 4.4. CCNN doesn't need backpropagation algorithm for learning. Instead, single layer network learning algorithm such as Widrow-Hoff or "delta" rule could be used. Fahlman and Lebiere (1990) applied "Quickprop" algorithm for training the weights.

Quickprop is an iterative function for finding the minimum of the cost function. The *optimization* problem in statistics such as GD assist in minimizing this cost/ loss function. That is, minimizing the error  $\varepsilon$  is an objective of the cost function. Similarly, in CCNN we reduce the error and maximize the correlation  $C$  between the units. Thus, developing a model of high quality and efficiency.

### 4.4.3 Training phase

Training of the model begins with one input and a output layer, and no hidden layer. At every iteration, a new neuron is added, and the decision about the neuron addition would be made by Gradient Descent (GD) technique. The added neuron is placed in the hidden layer and is connected to all preceding input units. On activating the neurons, the model with a connection between the unit of entry and neuron is kept frozen. In this state, the neurons are not connected to the output unit/ units. The neuron gets the connection to

the output layer only after training the weights. This process is iterated till an optimal weight is discovered. The architecture of CCNN is shown in Figure 4.2. Here, ■ illustrates that the network is in a frozen state and × shows it is in training state.

During the training of neurons, the objective was to adjust the weights to maximize the sum  $S$  over all output units  $O$ . This is obtained by equation 4.14. CCNN receives the trainable inputs to its pre-existing hidden units, with no output units connected to it. The input units are then processed by adjusting the weights to maximize the sum  $S$ . The output  $O$  is of the magnitude of the correlation between  $V$  (the candidate unit values) and  $E_O$  (the residual output error).

$$S = \sum_O \left| \sum_P (V_P - \bar{V})(E_{P,O} - \bar{E}_O) \right| \quad (4.14)$$

In the equation 4.14,  $P$  is the training pattern and are averaged over  $V$  and  $E$ . The objective here is to identify the best combinations of hidden units which maximizes  $S$ . In order to maximize  $S$  the backpropagation rule of taking partial derivation of  $S$  with respect to each combinations of input weights ( $\theta$ ) is applied. The partial differentiation of  $S$  with respect to  $\theta$  is shown in the equation 4.15. Here  $f'_O$  is derivative of pattern  $p$  with respect to sum of its inputs and  $X$  is the input. GD is used to identify the maximum of  $S$  after getting  $\frac{\partial S}{\partial \theta_i}$ .

$$\frac{\partial S}{\partial \theta_i} = \sum_{p,O} \sigma_O(E_{p,O} - E_O) f'_O I_{ip} \quad (4.15)$$

## 4.5 Modified Cascade-correlation Neural Network (Mod-CNN)

We advanced CCNN by optimizing the model to find an optimal number of neurons in each hidden unit along with some hidden units. In ModCNN, we are not freezing the hidden units until an optimal number of neurons are identified, and the hidden units are *not* added linearly but in parallel. For every added unit, an optimal number of neurons were identified. ModCNN was developed for analyzing the patterns of the clinical data and predicting the disease progression. The process begins with a minimum number of hidden units and neurons in each hidden units. After each iteration, error  $\varepsilon$  is calculated using Least Mean Square (LMS) algorithm. The objective here is to adjust the weights so that it minimizes MSE and maximize the correlation. After computing MSE, the GD

algorithm is applied on the set of MSE to find the minimum error  $\varepsilon$ . This would give us the optimal set of weights for each hidden unit.

### 4.5.1 Architecture of ModCNN

ModCNN works in two primary layers of architecture. *First layer* is to add new neurons to each hidden unit and to identify the optimal number of neurons. An attempt was made in every epoch to maximize the correlation between the neurons by minimizing the error ( $\varepsilon$ ). This minimization was done by finding its MSE, using LMS algorithm which is also known as Widrow-Hoff or delta algorithm (Widrow and Hoff, 1960). On obtained MSE, GD was applied to locate the combination of weights at which the  $\varepsilon$  was minimum. Thus, GD converges to discover optimal correlation of neurons to that hidden unit, where the error was minimum. On identifying the optimal number of neurons, a new hidden unit is added to the architecture. *Second layer* is for adding new hidden unit by freezing the number of neurons in the previous layer and computing the MSE with this new hidden unit. Obtained MSE of the current hidden unit, it is compared with the MSE of the previous hidden unit, and if the MSE has decreased, then that hidden unit is frozen else discarded. On observing the ascend in the magnitude of correlation, the newly added unit would be retained, else gets eliminated. The process of finding neurons to this new hidden units was continued, to find the optimal number of neurons. This process is repeated till an optimal combination of hidden units and neurons for ModCNN were discovered, for each set of inputs.

The architecture of ModCNN is shown in Figure 4.5. In the *first layer*, the optimal neurons are identified using the ADaptive LInear NEuron (ADALINE) network, and in the *second layer*, it freezes the hidden units on finding the optimal set of neurons. The ★ indicates that the units are under training and not frozen. ■ shows that nodes are frozen and are still under observation. The complete process is illustrated in Figure 4.6.

### 4.5.2 ADALINE circuit

ModCNN adopted ADALINE in developing a decision-making model. Bernard Widrow with Marcian E Hoff developed ADALINE and the training algorithm known as LMS in 1959, since then it has been expanded rapidly (Widrow and Hoff, 1960). The initial objective using ADALINE is to identify the weights such that cost function  $J$  is minimum.

LMS algorithm is applied to identify the set of weights ( $\theta$ ) generating MSE  $\xi$  and is

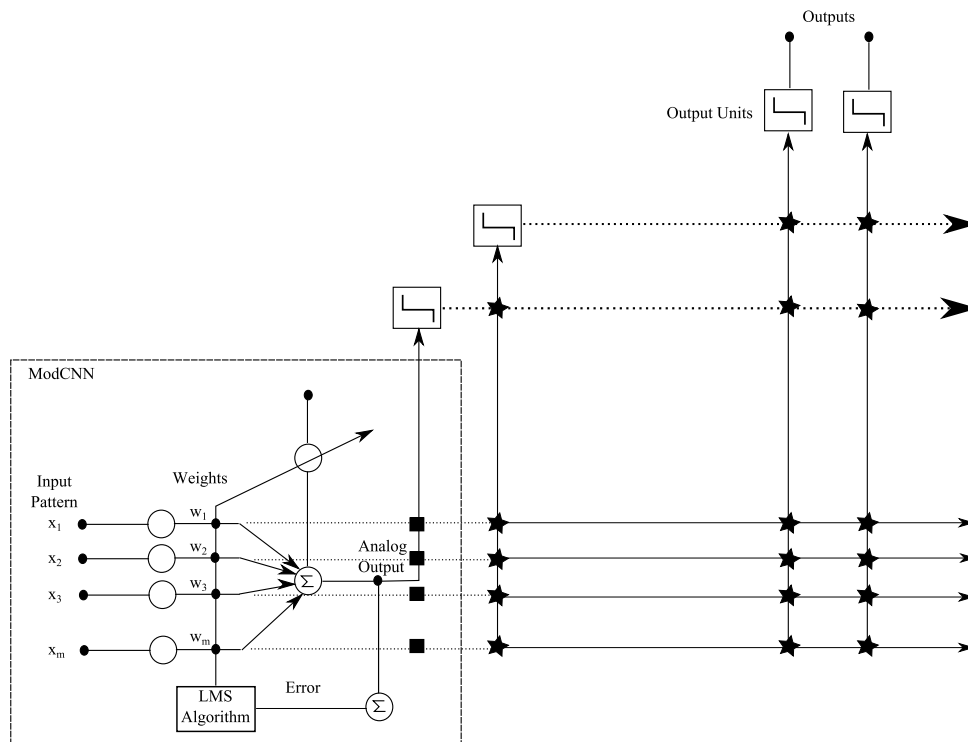
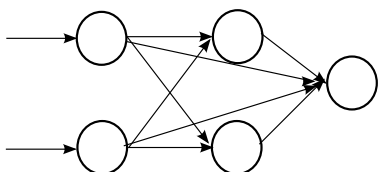
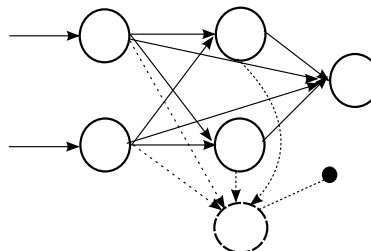


Figure 4.5: Schematic diagram of ModCNN architecture.

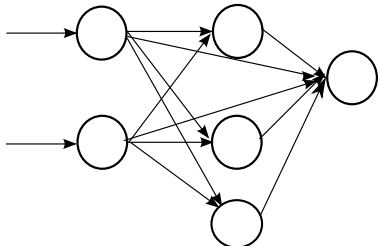
**Stage 1: Similar to ANN**



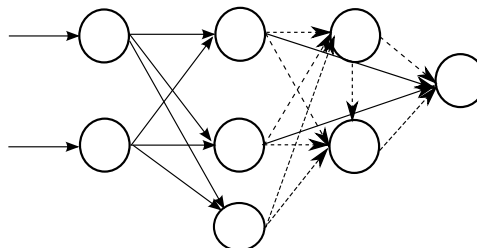
**Stage 2: New Node is added at hidden layer**



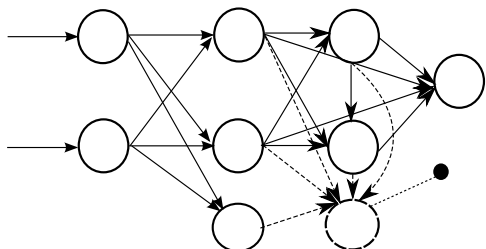
**Stage 3: Added node is optimized and frozen**



**Stage 4: New Hidden unit is added and is trained**



**Stage 5: Frozen hidden unit is added with node**



**Stage 6: Final ModCNN**

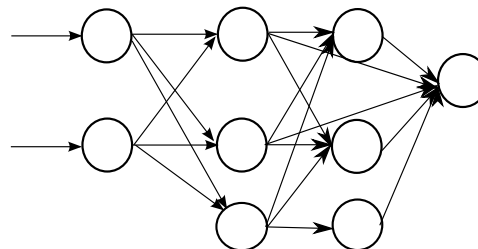


Figure 4.6: ModCNN training and frozen states

shown in Figure 4.7 (Widrow et al., 1994). It takes different patterns of input  $x$  and



gives analogue output  $h_{\theta}(x)$ . This analog output is compared with desired output  $y$  to calculate the error  $\varepsilon = h_{\theta}(x^i) - y^i$  at each iteration, till the minimum error is not achieved.

ADALINE circuit has a hidden unit with multiple combinations of weights. The process of finding error  $\varepsilon$  is repeated by adjusting the weights till minimum error is obtained. We applied the GD techniques to identify the optimal weights where the  $\varepsilon$  is minimum. This finally resulted in weights ( $\theta$ ) with less than a threshold is obtained and they were optimal.

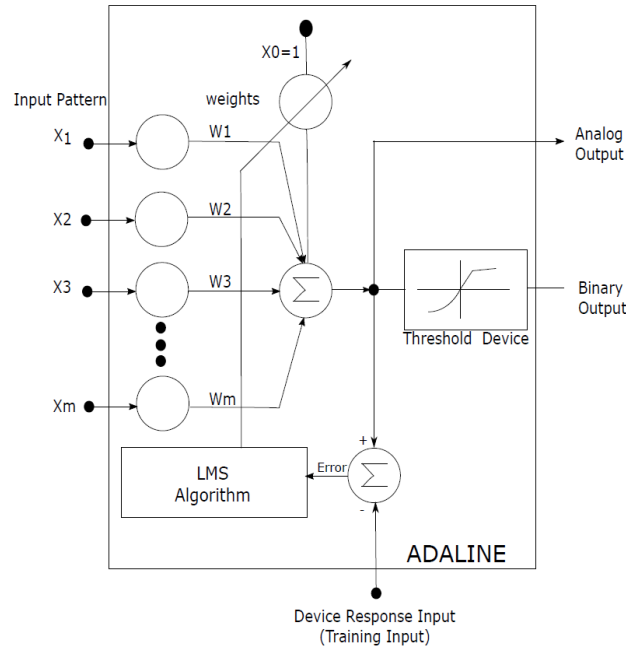


Figure 4.7: Architecture of ADALINE

### 4.5.3 Least Mean Square Algorithm: (LMS algorithm)

The LMS algorithm also known as Widrow-Hoff Delta learning rule was proposed by Widrow and Hoff (1960). Each input  $X = x_0, x_1, \dots, x_m$  goes to intermediate weights  $\theta = \theta_0, \theta_1, \dots, \theta_n$  to give the actual output  $h_{\theta}(x) = X^T \times \theta$ , which is the summation of  $(x_0 \times \theta_0), (x_1 \times \theta_1), \dots, (x_n \times \theta_n) = \sum_{i=0}^n X^T \times \theta$ . For each input pattern, there are *actual output* ( $h_{\theta}(x)$ ) and *desired output* ( $y$ ) and the error  $\varepsilon = h_{\theta}(x) - y = ((\sum_{i=0}^n X^T \times \theta) - y)$ . The error is used to adjust the weights, so that MSE is minimum.

We identified the optimal weights using GD algorithm. This is an iterative search algorithm, which starts randomly at some point known as *initial point*, from a vector space of weights. It moves towards an optimum position using gradient operator, thus

identifying the weights at which the costs could be minimum. The algorithm starts with initial weight = 0 (in most of the cases). On finding MSE for this weights using equation 4.18, their gradients are identified. The gradient is an inclined equation (either increasing or decreasing). Thus, if gradient on  $MSE > 0$  then, the error is increasing. This explains that weights are to decrease to reduce the error. The weight equation is shown in equation 4.16. In this equation  $\mu$  is the coefficient of convergence. To find the minimized error we go down the slope of error  $\varepsilon$ . So there is a subtraction from the current weight  $\theta_i$  and gradient coefficient.  $\nabla$  is gradient operator.

$$\theta_{i+1} = \theta_i - \mu \nabla \varepsilon_i \quad (4.16)$$

Let  $\bar{\theta} = [\theta_0, \theta_1, \dots, \theta_{N-1}]^T$  be the weight vector for the first hidden unit.  $\bar{x}(n) = [x(0), x(1), \dots, x(n)]^T$  be  $n$  input vector. The analogue output be  $h_{\theta}(x) = \bar{\theta}^T \bar{x}^i$  and error is calculated using  $e(n) = h_{\theta}(x^i) - y^i$ . The MSE ( $\xi$ ) is calculated from the equation 4.17.

$$\text{Output } h_{\theta}(x) = X^T * \theta$$

$$\text{Error } \varepsilon = h_{\theta}(x) - y = (X^T * \theta - y^i)$$

$$\varepsilon^2 = y^2 - 2yX^T\theta + \theta^T XX^T\theta$$

$$\text{Then the mean square error } \xi = E[\varepsilon^2]$$

$$= E[y^2] - 2E[yX^T]\theta + \theta^T E[XX^T]\theta \quad (4.17)$$

Let  $P$  be cross-correlation vector between  $\bar{x}(n)$  and  $y(n)$ ,  $R$  be an auto-correlation matrix of filtered inputs. The ADALINE circuit in equation 4.17 is generalized and the quadratic equation of LMS is shown in equation (4.18).

$$\text{let } P = E[yX^T]$$

$$R = E[XX^T]$$

*Quadratic function over weight/*

$$\text{mean square error } \xi = E[y^2] - 2P^T\theta + \theta^T R\theta \quad (4.18)$$

This MSE  $\xi$  is a quadratic equation and yields a bowl shaped plane as shown in Figure(4.8). Here only two weights are considered on the x-axis, y-axis and the MSE on

Z-axis. It has a global minimum and is achieved by GD as shown in the equation 4.19.

$$R\bar{\theta}_0 = \bar{P} \quad (4.19)$$

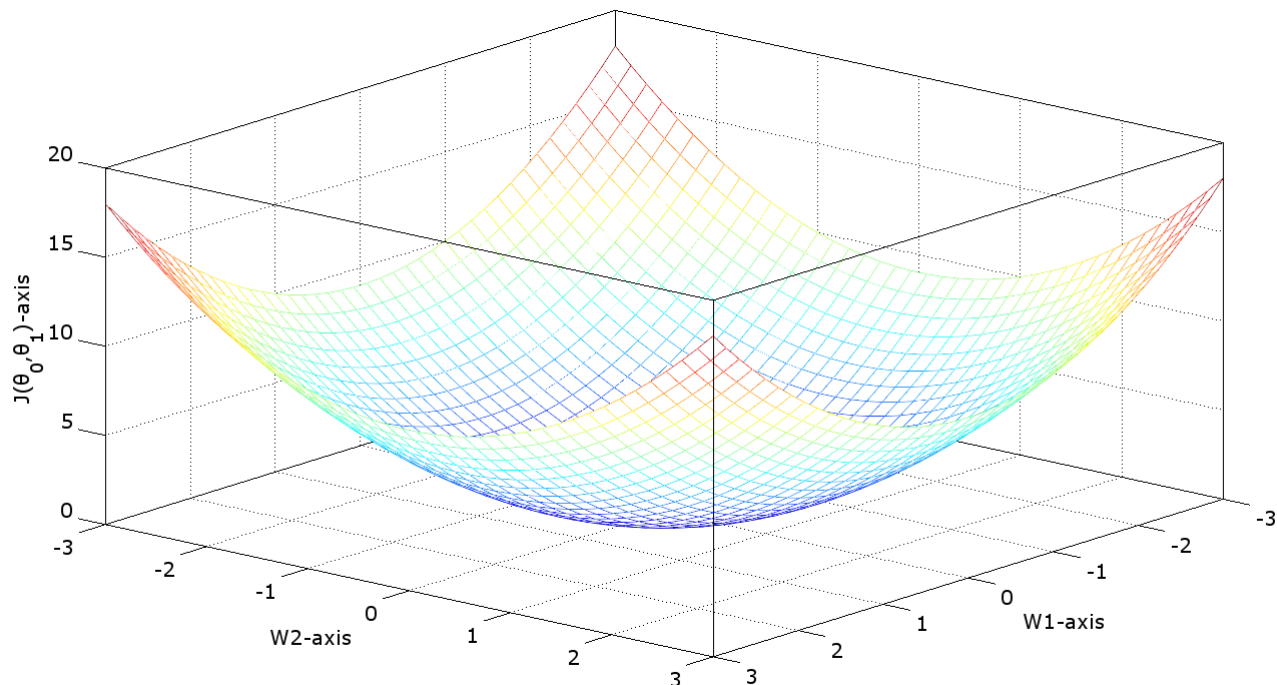


Figure 4.8: Paraboloid of the cost function

#### 4.5.4 Gradient descent

The objective is to minimize the cost function  $J$  for these weights. GD algorithm is applied to identify the global minimum of the cost function ( $J$ ) (Burges et al., 2005). Consider the graph in Figure (4.9), where the polynomial curve is obtained on plotting cost function  $J$  against one  $\theta$ , keeping  $\theta_0 = 0$ . The learning rates are the small baby steps ( $\alpha = \frac{\mu}{2}$ ) that converges to a global minimum to identify the optimal  $\theta$ . Consider a contour graph in Figure (6.17) plotted against  $\theta_1$ ,  $\theta_2$  and MSE. Using the GD algorithm the minimum of  $\theta_1$  &  $\theta_2$  is achieved (Bottou, 2010).

GD algorithm starts the iteration with some initial  $\theta$  such as  $J(\theta_0 = 0 \ \& \ \theta_1 = 0)$  and iteratively updates  $\theta$  as shown in equation (6.16) till the algorithm converges at a local minimum. The objective of gradient algorithm is to  $\underset{\theta_0 \theta_1 \dots \theta_n}{\text{minimize}} J(\theta_0, \theta_1 \dots \theta_n)$ . The

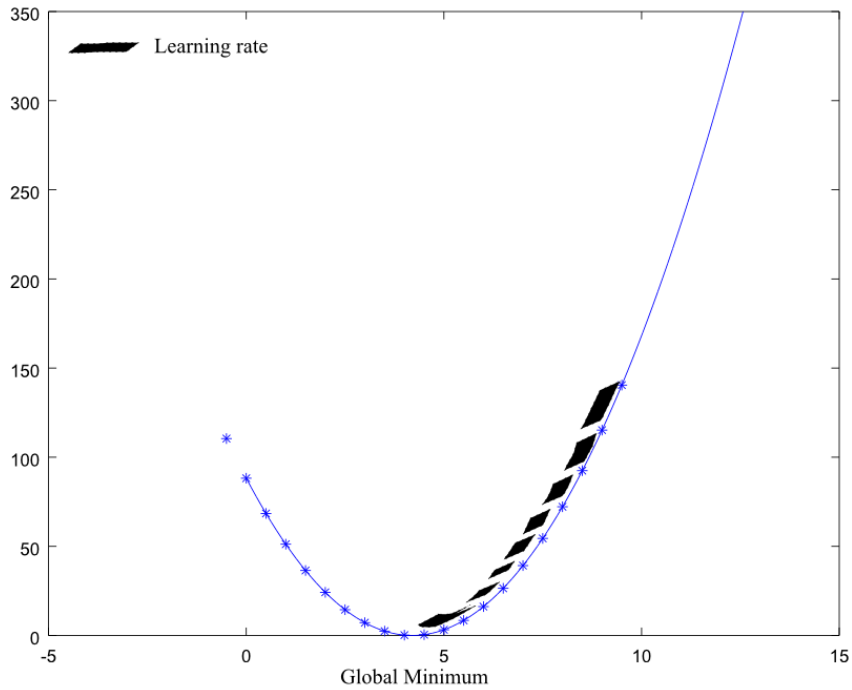


Figure 4.9: Learning Rate in gradient descent

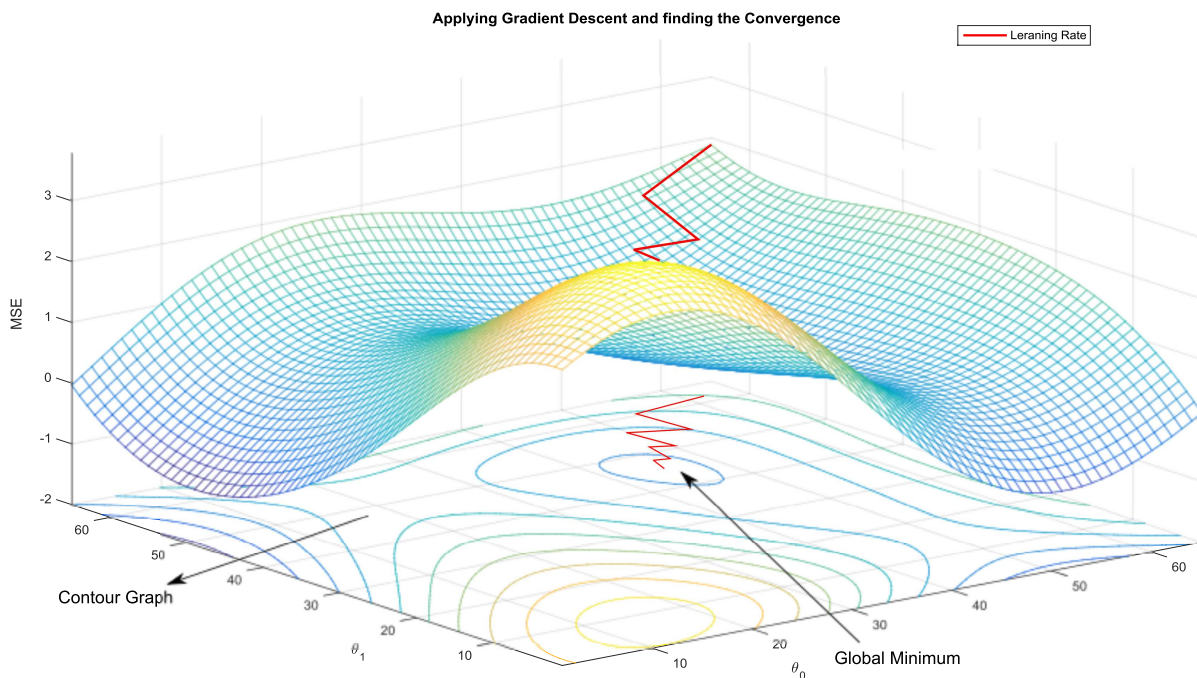


Figure 4.10: Learning Rate of gradient descent using contour graph

algorithm starts an iteration by moving with small baby step  $\alpha$  known as learning rate, in a direction that moves down to reach minimum  $\theta$ . After each iteration, the algorithm will check the direction of movement that converges to a local minimum. The GD repeats the equation (6.16) until it converges, by updating the  $\theta_j$  the value after every iteration.  $\frac{\partial}{\partial \theta_j}$  is measured by theorem (3). Convergence is the stopping condition:  $(\alpha \|\nabla J\|) > \varepsilon$ , where

$\|\nabla J\|$  is  $\sqrt{J(\theta_1^2) + J(\theta_2^2) + J(\theta_3^2) \dots}$  is the equation of normalization (Zhang, 2004).

$$\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) \quad \text{for all values of } j : 0, \dots, n \quad (4.20)$$

*Theorem 3.* Let  $J(\theta)$  be a cost function whose derivative exists, then  $\frac{\partial}{\partial \theta_j} J(\theta)$  is a continuous function.

Let  $J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^i) - y^i)^2$  from equation 6.15, then

$$\begin{aligned} & \frac{\partial}{\partial \theta_j} \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^i) - y^i)^2 \\ & \quad \text{Using product rule} \\ & \frac{\partial}{\partial \theta_j} \frac{1}{2m} \sum_{i=1}^m 2 \times (h_\theta(x^i) - y^i) = \frac{\partial}{\partial \theta_j} \frac{1}{m} \sum_{i=1}^m (h_\theta(x^i) - y^i) \times \frac{\partial}{\partial \theta_j} (h_\theta(x^i) - y^i) \end{aligned} \quad (4.21)$$

$$\begin{aligned} \text{Let } & \frac{\partial}{\partial \theta_j} (h_\theta(x^i) - y^i) = \frac{\partial}{\partial \theta_j} (h_\theta(x^i)) \\ & \quad \text{where } h_\theta(x^i) = \theta x \quad \text{if } \theta_0 = 0 \\ \therefore & \frac{\partial}{\partial \theta_j} (h_\theta(x^i) - y^i) = x^i \end{aligned} \quad (4.22)$$

Applying result of 4.22 in 4.21

$$\frac{\partial}{\partial \theta_j} J(\theta) = h_\theta((x) - y) \times x \quad (4.23)$$

*Theorem 4.* Let  $\frac{\partial}{\partial \theta_j} J(\theta) = h_\theta((x) - y) \times x$  be the partial derivative of the cost function. Using which GD algorithm for multiple parameters is proved

Let  $\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$  for all values of  $j : 0, \dots, n$  from equation 6.16, then

$$\begin{aligned} & \theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) \\ & \text{where } \frac{\partial}{\partial \theta_j} J(\theta) = h_\theta((x) - y) \times x \text{ from equation 4.23 of theorem 3} \\ & \text{Then } \theta_j = \theta_j - \alpha h_\theta((x) - y) \times x \\ & \text{where } h_\theta(x) = \theta_0 + \theta_1 x \\ & \text{Update } \theta_0 \text{ \& } \theta_1 \end{aligned} \quad (4.24)$$

$$\text{temp}_0 = \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$

$$\text{temp}_1 = \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$

$$\theta_0 = \text{temp}_0$$

$$\theta_1 = \text{temp}_1$$

Using theorem 4, the GD for multiple parameters can be achieved as shown in derivation of equation 4.26 and the GD algorithm is shown in algorithm 1.

$$\begin{aligned}\frac{\partial}{\partial\theta_j} J(\theta_0, \theta_1) &= \frac{\partial}{\partial\theta_j} \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^i) - y^i)^2 \\ &= \frac{\partial}{\partial\theta_j} \frac{1}{2m} \sum_{i=1}^m (\theta_0 + \theta_1 x^i - y^i)^2\end{aligned}$$

$$\text{let } \theta_0 \text{ i.e., } j = 0 : \frac{\partial}{\partial\theta_0} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^m (\theta_0 + \theta_1 x^i - y^i) \quad (4.25)$$

$$\text{let } \theta_1 \text{ i.e., } j = 1 : \frac{\partial}{\partial\theta_1} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^m (\theta_0 + \theta_1 x^i - y^i) \times x^i \quad (4.26)$$

---

**Algorithm 1:** GD algorithm

---

**Result:** Optimal  $\theta_0$  &  $\theta_1$

1 initialize  $\theta_0$  &  $\theta_1$  to random values;

2 **while** !converged **do**

3      $\theta_0 = \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^m (\theta_0 + \theta_1 x^i - y^i)$

4      $\theta_1 = \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^m (\theta_0 + \theta_1 x^i - y^i) \times x^i$

5 **end**

---

### 4.5.5 Finding correlation among the hidden units

The space or residual between analogue output could be decreased only by having an ideal set of  $\theta$ 's or the weights. By having so, the residual could be eliminated, and the error could be brought down, i.e.,  $(h_{\theta}x^{(i)} - y^{(i)})$ , where  $i = 0$  to  $m$  and  $m$  is number of training set. Thus the cost function  $J(\theta_0, \theta_1)$  would be the MSE about taking the mean of sum of all these errors as shown in equation 4.27.

$$J(\theta_0, \theta_1) = \frac{1}{2m} \left( \sum_{i=1}^n h_{\theta}(x^i) - y^i \right)^2 \quad (4.27)$$

The objective of optimization problem is to minimize this cost function  $J(\theta_0, \theta_1) = \underset{\theta_0, \theta_1}{\text{minimize}} J(\theta_0, \theta_1)$ . This is achieved by finding the cost function for different combinations of  $\theta_0$  and  $\theta_1$ . GD helps in finding the optimized values for  $\theta_0$  and  $\theta_1$ . The paraboloid curve shown in Figure 4.11 illustrates the operation of GD. Here the weights  $\theta_0$  is considered on X-axis,  $\theta_1$  on Y-axis, and MSE on Z-axis. GD is run on this to identify the optimal weights where the cost function is minimized.

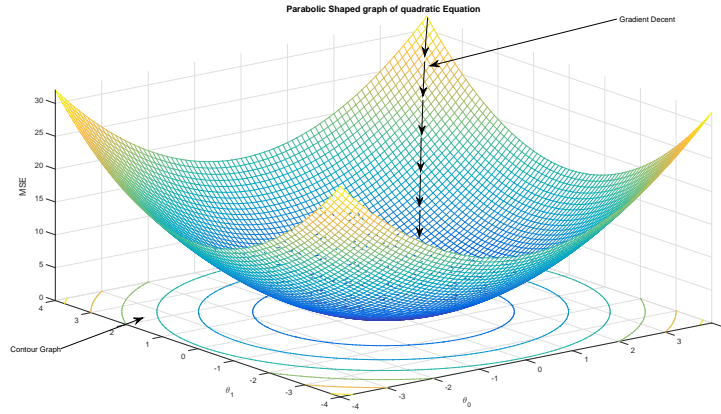


Figure 4.11: Illustration of gradient descent

Now the challenge is in finding the optimal  $\theta$  where  $J(\theta) = 0$ . For this, we applied GD algorithm to climb down the hill of a paraboloid curve and find the optimal combinations of  $\theta$ . The training of Quickprop would stop at a point where the distance between  $\theta_0$  &  $\theta_1$  and *error*  $\varepsilon$  are less than the threshold. Threshold is a user defined stopping condition and helps in stopping the iteration when minimized cost function is obtained (*minimize*  $J(\theta_0, \theta_1)$ ). The challenge here was in analysing the number of epochs the system would take to reach an optimization state. The optimization problem here is in training the output units to minimize the  $\xi$  and shown in equation 4.28. The equation is obtained by modifying basic MSE regression equation 4.27.

$$\varepsilon = \frac{1}{2} \sum_P (h_\theta(x^P) - y^P)^2 \quad (4.28)$$

In the equation 4.28,  $P$  is a pattern. The pattern is a set of activation across the network. Example: given the input pattern of a car, the model should be able to classify the type of the car.  $h_\theta(x^P)$  is the observed output, and  $y^P$  is the desired output. Since the model is trained in a supervised manner, we would train them using the desired output to check the error the model is making. Hence the objective is to minimize the error  $h_\theta(x^P) - y^P$ . This minimizing of error is performed with an assistance of GD 4.30.

$$E_P = (h_\theta(x^P) - y^P) f'_P(\text{net}_O) \quad (4.29)$$

$$\frac{\partial \varepsilon}{\partial \theta} = \sum_P \varepsilon_P, X_P \quad (4.30)$$

In the equation 4.30  $f'_P$  is the derivative of an activation function for the output

$h_\theta(x^P)$ .  $X_P$  is the input vector of the pattern  $P$ . After successful completion of training, the hidden units are frozen. Correlation between the  $h_\theta(x^P)$  and the residual output  $\varepsilon_P$  is maximized and is shown in 4.31. In the equation  $\bar{h}_\theta(x^P)$  and  $\bar{\varepsilon}_P$  are averaged over  $h_\theta(x^P)$  and  $\varepsilon_P$  respectively.

$$\begin{aligned}
\text{Correlation } C &= \sum_P \left| \sum_P (h_\theta(x^P) - \bar{h}_{\theta P})(\varepsilon_P - \bar{\varepsilon}_P) \right| \\
&= \sum_P \left| \sum_P h_\theta(x^P) \varepsilon_P - \bar{\varepsilon}_P \sum_P h_\theta(x^P) \right| \\
&= \sum_P \left| \sum_P h_\theta(x^P) (\varepsilon_P - \bar{\varepsilon}_P) \right| \tag{4.31}
\end{aligned}$$

Now, GD is applied to continue the maximization of correlation  $C$  and is shown in the equation 4.32. Here,  $\lambda_O$  is a sign of the correlation between the output unit  $h_\theta(x^P)$  and the residual error  $\varepsilon_P$ .

$$\begin{aligned}
\delta_P &= \sum_O \lambda_O (\varepsilon_P - \bar{\varepsilon}_P) f'_P \\
\frac{\partial C}{\partial \theta_i} &= \sum_P \delta_P X_{P,i} \tag{4.32}
\end{aligned}$$

### 4.5.6 Training phase

ModCNN used ADALINE circuit to identify the set of weights ( $\theta$  values) to reduce the error  $\varepsilon$  using LMS algorithm, instead of randomly selecting the  $\theta$ . On identifying the  $\theta$ , cost function  $J(\theta_0, \theta_1)$  is calculated. Objective is to identify optimal  $\theta$  and make the model more efficient. That is to find the system that can give minimum loss  $J(\theta_0, \theta_1) = \varepsilon(h_\theta(x), y)$ . The performance of training example data is measured by *empirical risk* function  $E_n(h_\theta(x))$  shown in the equation 4.34.

$$E_n(h_\theta(x)) = \int \xi(h_\theta(x), y) \mathcal{U} \mathcal{P}(x, y) \tag{4.33}$$

In the equation 4.33 we assume that there is a joint probability distribution  $\mathcal{P}(x, y)$  over  $X$  and  $Y$  subjected  $h_\theta : X \rightarrow y$  and  $\mathcal{U}$  is a symbol of uniform distribution of joint



probability. The objective is to reduce the *empirical risk* (4.34) and *expected risk* (4.33).

$$E_n(h_\theta(x)) = \frac{1}{n} \sum_{i=1}^n \xi(h_\theta(x^i), y_i) \quad (4.34)$$

### 4.5.7 Testing phase

The quadratic model developed in a learning phase of ModCNN and is shown in Figure (4.11). On running the GD on this optimal load is identified. In GD after each iteration  $\theta$  is updated as shown in equation 4.35. The objective here is to *reduce the empirical risk*  $E_n(h_\theta(x))$  shown in the equation (4.34), where  $\gamma$  is randomly chosen gain function,  $n$  is a number of training examples,  $\nabla$  is gradient  $(z_i, \theta_t)$ , where  $z_i = \{(x_0, y_0), (x_1, y_1) \dots (x_m, y_m)\}$  is the training pair and  $t$  is the iteration, hence *obtaining minimum cost function*.

$$\theta_{t+1} = \theta_t - \gamma \frac{1}{n} \sum_{i=1}^n \nabla_{\theta} h_{\theta}(x)(z_i, \theta_t) \quad (4.35)$$

Convergence is faster when convexity property of loss function  $\varepsilon$  is strong (Bottou, 2010). The Table 4.2 compares the asymptotic behaviour of ANN, CCNN, and ModCNN. First three rows show the cost of computation at each iteration, some minimum iteration required to reach optimized accuracy  $\rho$  and cost of similar computation. The fourth row gives the significant property of large scale machine learning, which indicates the computational cost of predefined error, the expected risk (4.33).

Although CCNN performed in less optimal way for optimizing and showing better accuracy  $\rho$  (*third row in Table 4.2*), they took exponentially lesser time in reaching the predefined Expected risk  $E(f)$ , when compared to ModCNN. Hence, when the importance is computational time rather some examples, CCNN performed better. Though the proposed ModCNN showed better accuracy than ANN and CCNN, they took a long time to find  $E(f)$ ; this was only due to an intense computation of ModCNN. So, we designed our system in a distributed manner using master-slave model and decreased this time to reach  $E(f)$ . In the Figure 4.12, the accuracy and time to achieve expected risk are evaluated. From the figure, we could analyse that ModCNN showed better accuracy than CCNN and even ANN but, till the feature size was around 100. Later on, we could see ascent in their accuracy. It is also clearly shown that, though proposed ModCNN performed better than ANN, CCNN outperformed ModCNN. For overcoming this drawback of ModCNN, we

distributed its computation using master-slave model of *Map-Reduce* (Ghemawat et al., 2003) and made it more efficient.

Table 4.2: Asymptotic equivalents for ANN, CCNN, and ModCNN

|  | ANN   | CCNN                       | ModCNN  |
|--|---|----------------------------|---|
| <i>Time per iteration</i>                            | 1   | n                          | n   |
| <i>Iteration to accuracy <math>\rho</math></i>       | $\log \frac{1}{\rho}$                                     | $\log \log \frac{1}{\rho}$ | $\frac{1}{\rho}$  |
| <i>Time to accuracy <math>\rho</math></i>            | $\frac{1}{\rho}$  | $n \log \frac{1}{\rho^2}$  | $n \log \log \frac{1}{\rho}$  |
| <i>Time to excess error <math>\varepsilon</math></i> | $\frac{1}{\varepsilon^\alpha} \log \frac{1}{\varepsilon}$ | $\frac{1}{\varepsilon}$    | $\frac{1}{\varepsilon^\alpha} \log \frac{1}{\varepsilon} \log \log \frac{1}{\varepsilon}$ |

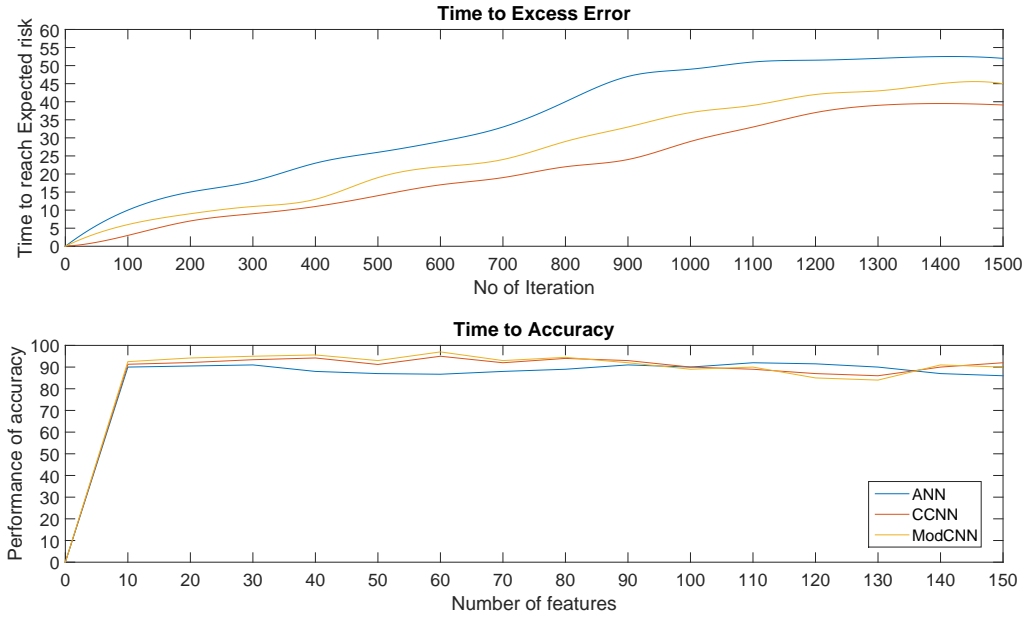


Figure 4.12: Empirical testing of ANN, CCNN, and ModCNN.

#### 4.5.8 Master-slave model

It was understood that complexity of the system was increased when compared with the performance of CCNN. This complexity was due to high computation when compared with CCNN. This ambiguity can be avoided by the application of the master-slave model. At the hidden unit, the feed-forward operation from one layer to another is done for different combinations of neurons, by each slave in parallel. The MSE was calculated by the slaves along with the number of iterations required to obtain  $MSE < threshold$ . The

master then runs the GD algorithm to identify an optimal number of neurons, where MSE and iterations are minimum. Hence the best optimal ModCNN model is discovered for the input data.

#### **A At slave**

Each slave receives information about number of neurons, an input data ( $X$ ) and an output data ( $Y$ ) from the master. The slaves compute an input to the hidden and output unit. The slaves also calculate output of hidden and output unit using the sigmoid function. Each slave using an information of actual output  $y$ , computed MSE. Finally, the slave would return the number of iterations it took to reach the threshold along with MSE to the master. This would help the master in identifying the better combination of number of neurons and input data.

#### **B At master**

The master on receiving the reply from all the slaves about some iterations and MSE runs the GD algorithm. Using this technique, it identifies an optimal set of neurons that yielded least MSE with a minimum number of iterations. The result of optimal neurons for different hidden units on running GD is shown in Figure 4.13.

After finding the optimal neurons for the first hidden unit, ModCNN adds a new hidden unit and records the change in MSE. A polynomial model is developed using the information of hidden unit and corresponding MSE. The GD algorithm is run on the polynomial curve and identifies the hidden unit where the overall MSE is less. This result is shown in Figure 4.14.

## **4.6 Summary**

ModCNN was able to identify the optimal combination of hidden units and neurons in each hidden unit. This was achieved by applying GD on MSE obtained at each iteration. The model was trained and tested for empirical risk factors. And it was observed that it performed better when compared to ANN and CCNN. By the application of master-slave model, it was made more optimised. Thus giving high prediction accuracy and assisting in taking critical clinical decisions.

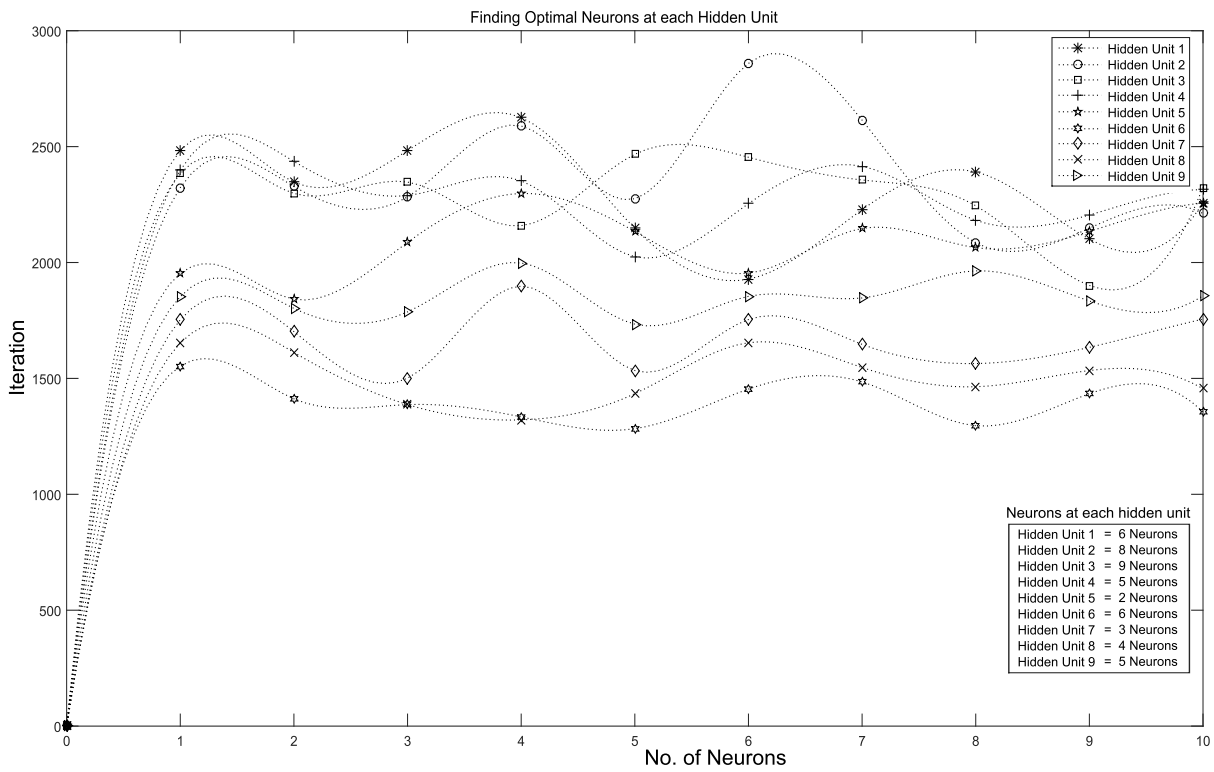


Figure 4.13: Gradient descent identifying the optimal neuron for different hidden units.

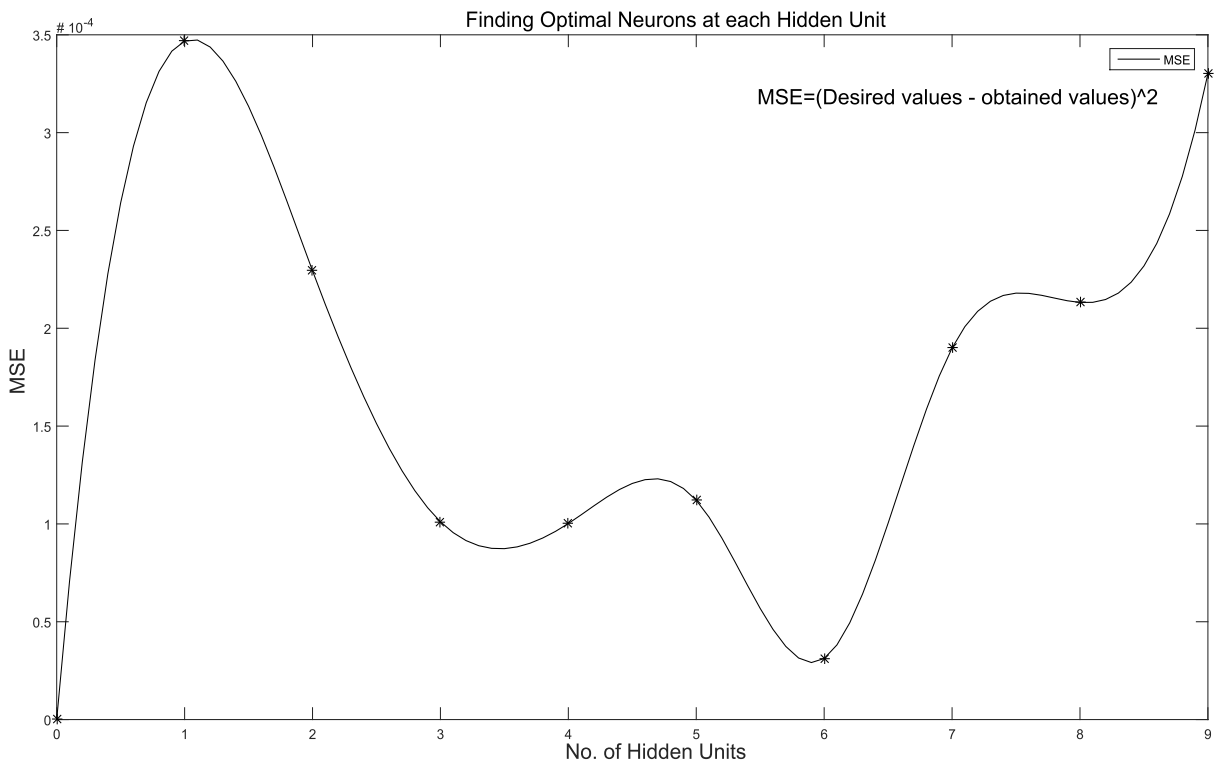


Figure 4.14: Gradient descent identifying the optimal hidden units using MSE.

# Chapter 5

## Experimental Result

### 5.1 Background

The current work focused on developing a *CDSS* for the treatment management of GSD. EHR data of treatment procedure for GSD patients were recorded from tertiary care centre in north Malabar, Kerala, India, during the period of 2014 to 2015. 530 cases of GSD were identified among 4800 patients, who came with the complaint of abdominal pain. Among the observed cases, 260 were complicated and rest were uncomplicated. Out of 260 cases, 143 (55%) presented with cholecystitis, 57 (22%) had choledocholithiasis, 44 (17%) had pancreatitis and remaining 16 (6%) cases were cholangitis. The uncomplicated GSD cases were more unlikely to undergo OC/ LC when compared to the complicated cases. The spectrum of GSD in this study was found to be comparable with the modes of presentations of California study (Glasgow et al., 2000), and the comparison is shown in Table 5.1. Glasgow et al. (2000) conducted a retrospective study on those who underwent cholecystectomy in 1996 in California, to identify the spectrum and cost of complicated GSD. They conducted study in two settings, *hospital based* on 248 patients and *community based* on 40571. This comparison shows that the incidence of choledocholithiasis and pancreatitis is increasing in the subjected region.

GSD was considered here due to its increasing prevalence in last few decades in India. Garg (2013) reported an epidemiological study by a group from All India Institute of Medical Science (AIIMS), Delhi on the prevalence of GSD in Kerala. The study showed prevalence of Acute Pancreatitis (AP):  $\widetilde{126}/100,000$  population and calcific pancreatitis:  $\widetilde{98}/100,000$ . This observation was very high when compared to  $\widetilde{27}/100,000$  in the western countries. Kumar Sangwan et al. (2016) conducted a retrospective analysis and observed the incidence of GSD is seven times in north India than in the south.

Table 5.1: Presentation of different spectrum of GSD in the current study, compared with California study (Glasgow et al., 2000)

| <b>Spectrum of GSD</b> | <b>Hosp. Based California Study</b> | <b>Community Based California Study</b> | <b>Proposed Study</b> |
|------------------------|-------------------------------------|---|-----------------------|
| Uncomplicated GSD      | 56%                                 | 56.5%                                   | 51%                   |
| Cholecystitis          | 29%                                 | 35.9%                                   | 27%                   |
| Pancreatitis           | 6%                                  | 4%                                      | 8%                    |
| Choledocholithiasis    | 5%                                  | 3.1%                                    | 11%                   |
| Cholangitis            | 0%                                  | 0.2%                                    | 3%                    |

## 5.2 Attribute Distribution

The description of the attribute used in this study is shown in Table 5.2. We used 32 features associated with GSD. In the Figures 3.1,3.2,3.3,3.4 and 3.5 the stratification of patients based on the different spectrum of GSD is shown. In these figures different attributes along with their prevalence percentage is shown on the top of the bar. The Figure 5.2 shows the feature distribution among the patients detected with complicated GSD. The need was to find the features in each patient along with their association towards the risk progression. Chi-squared test ( $(\chi)^2$ ) was applied for statistically analysing and conducting the hypothesis test about this feature distribution.

### 5.2.1 Chi-squared test ( $(\chi)^2$ )

Here the features are classified into mutually exclusive classes.  $\chi^2$  test evaluates the likelihood of association of the features in similar class using statistical testing.  $\chi^2$  test is conducted to analyse the distribution of the features and find the significantly associated features. The significant association is known as risk factors (red, yellow, green and blue), shown in Figure 5.2. Red being highly relevant and blue being least. The aim is to find the association between the significant factors and the different spectrum of GSD. Consider the Table 5.3 illustrating the values of observed risk factors for a different spectrum of GSD. Each cell value is observed as the significant value for different spectrum. Row total is the sum of the row values, column total is the sum of a column, and the total is either

Spectrum of GSD obtained by running ModCNN

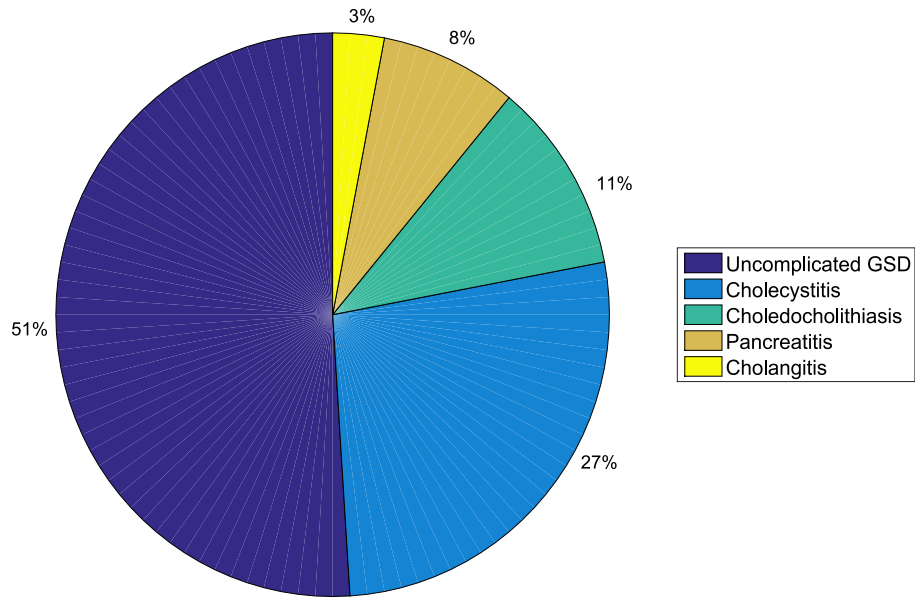


Figure 5.1: Classification result of GSD patients.

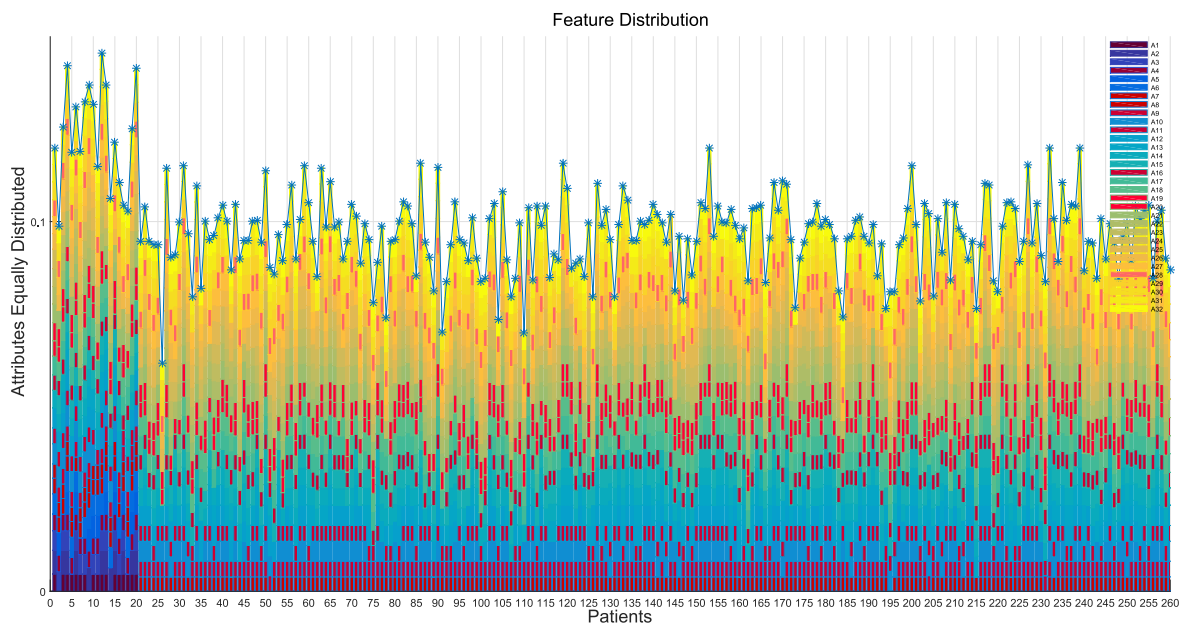


Figure 5.2: Clinical readings, showing the feature distribution of 260 patients.

the sum of row total or column total.

## A Hypothesis

Let the null hypothesis be:

- $H_0$ : Significant factors are not associated with spectrum.
- $H_1$ : Significant factors are associated with spectrum.

Table 5.2: Description of the attribute

| <b>Attributes</b>   |   |
|---|---|
| <b>Input Attributes:</b> (32 No.):<br>The list of input attributes are: | Age, Sex, Abdominal pain (A1), Dyspepsia (A7), Nausea & vomiting (A30), Fever (A8), Jaundice (A14), Pruritus (A20), Anorexia (A3), Tenderness (A26), Murphys sign (A16), Rebound tenderness (A24), Gallbladder palpable (A31), Hepatomegaly (A11), Obesity (A17), Diabetes Mellitus (A6), Hypertension (A29), IHD (A13), Previous Surgery (A19), Hereditary spherocytosis (A12), APD (A4), Total count >12000 (A28), Total bilirubin >2 (A27), Raised ALT (A22), Raised AST (A23), Raised ALP (A21), Amylase >three times upper limit (A2), Lipase >three times upper limit (A15), Dilated CBD (A5), Pancreas bulky and echotexture (A18), Gallbladder Thickened & pericholecystic fluid (A9) and USG features of cholangitis (A32). The attribute detail is shown in Figures 3.1,3.2,3.3,3.4 and 3.5   |
| <b>Significant Attributes:</b>  | <p><b>Cholangitis</b> (3): A1(<math>P = 0.1969</math>), A4(<math>P = 0.1190</math>), A7(<math>P &lt; 0.001</math>), A8(<math>P=0.2483</math>), A9(<math>P &lt; 0.001</math>), A10(<math>P = 0.8016</math>), A14(<math>P = 0.7571</math>), A19(<math>P = 0.2094</math>), A21(<math>P = 0.5802</math>), A26(<math>P &lt; 0.001</math>); <b>Pancreatitis</b> (6): A1(<math>P &lt; 0.001</math>), A4(<math>P = 0.6994</math>), A7(<math>P = 0.4563</math>), A8(<math>P &lt; 0.001</math>), A9(<math>P &lt; 0.001</math>), A10(<math>P &lt; 0.001</math>), A12(<math>P = 0.6040</math>), A14(<math>P = 0.8876</math>), A19(<math>P &lt; 0.001</math>), A21(<math>P = 0.4855</math>), A22(<math>P = 0.6040</math>), A26(<math>P = 0.5300</math>), A28(<math>P &lt; 0.001</math>); <b>Cholecystitis</b> (5): A1(<math>P &lt; 0.001</math>), A3(<math>P = 0.1568</math>), A4(<math>P &lt; 0.001</math>), A7(<math>P = 0.7350</math>), A8(<math>P &lt; 0.001</math>), A11(<math>P &lt; 0.001</math>), A16(<math>P &lt; 0.001</math>), A17(<math>P = 0.2145</math>), A24(<math>P = 0.0656</math>), A26(<math>P = 0.6537</math>), A29(<math>P = 0.9828</math>); <b>Choledocholithiasis</b> (2): A1(<math>P &lt; 0.001</math>), A3(<math>P = 0.9179</math>), A7(<math>P &lt; 0.001</math>), A11(<math>P = 0.2689</math>), A13(<math>P = 0.4730</math>), A14(<math>P = 0.2689</math>), A19(<math>P = 0.5364</math>) ()</p> |
| <b>Key attributes:</b>  | Age, Sex and Patients ID  |
| <b>Output:</b>  | Stratification of patients based on their risk level towards ERCP.  |



Table 5.3: Observed risk factors for a different spectrum of GSD

| Significance →<br>Spectrum ↓ | Red | Yellow | Green | Blue | Row Total |
|------------------------------|-----|--------|-------|------|-----------|
| Cholelithiasis               | 20  | 14     | 12    | 53   | 99        |
| Cholangitis                  | 18  | 13     | 18    | 32   | 81        |
| Pancreatitis                 | 12  | 48     | 39    | 33   | 132       |
| Cholecystitis                | 18  | 23     | 42    | 14   | 97        |
| Column Total                 | 68  | 98     | 111   | 132  | 409       |

### B Step 1: Finding the expected frequency

The expected frequency is calculated using the equation 5.1 and the observed values of Table 5.3. The result of this calculation is shown in Table 5.4.

$$\frac{Row_{Total} \times Column_{Total}}{Grand_{Total}} \quad (5.1)$$

Table 5.4: Expected frequency calculated for the observed values in Table 5.3

| Significance →<br>Spectrum ↓ | Red     | Yellow  | Green   | Blue    | Row Total |
|------------------------------|---------|---------|---------|---------|-----------|
| Cholelithiasis               | 16.4597 | 23.7213 | 26.8680 | 31.9511 | 99        |
| Cholangitis                  | 13.4670 | 19.4083 | 21.9829 | 26.1418 | 81        |
| Pancreatitis                 | 21.9462 | 31.6284 | 35.8240 | 42.6015 | 132       |
| Cholecystitis                | 16.1271 | 23.2421 | 35.8240 | 42.6015 | 97        |
| Column Total                 | 68      | 98      | 111     | 132     | 409       |

### C Step 2: Calculating $(\chi)^2$

$\chi^2$  is calculated using the equation 5.2. In the equation 5.2, the *Observed* and *Expected freq* are the values shown in Table 5.3 and 5.4 respectively, and *chi2cdf* is a *chi-square to cumulative distribution function*. The resulted values are tabulated in Table 5.5. If the  $\chi^2 < 0.05$  then that attribute is identified as significant. Significance of each attribute of the current study is shown in Table 5.6.

$$\begin{aligned} \chi^2 &= \frac{(Observed\ frequency - Expected\ frequency)^2}{Expected\ frequency} \\ &= 1 - chi2cdf(\chi - 1) \end{aligned} \quad (5.2)$$

Table 5.5: Result of  $(\chi)^2$ 

| Significance →<br>Spectrum ↓ | Red    | Yellow | Green  | Blue   |
|------------------------------|--------|--------|--------|--------|
| <b>Choledocholithiasis</b>   | 0.3829 | 0.0459 | 0.0041 | 0.0002 |
| <b>Cholangitis</b>           | 0.2167 | 0.1458 | 0.3956 | 0.2519 |
| <b>Pancreatitis</b>          | 0.0337 | 0.0036 | 0.5957 | 0.1413 |
| <b>Cholecystitis</b>         | 0.6410 | 0.9600 | 0.0023 | 0.0020 |

The relative importance of each feature along with their weight was calculated using *Relieff's algorithm* proposed by Kira and Rendell (1992). *Relieff*  $(X, Y, K)$  evaluates the rank and weight of input  $X$  and output  $Y$  with  $K$  nearest neighbours. Here the rank and weight give the importance of an attribute. Since the attributes were continuous, *Pearson correlation coefficients test* was conducted on the variables to represent the correlation between the input and output. Pearson (1896) introduced Pearson correlation coefficients test. This was needed to evaluate the data distribution before being used for statistical analysis. The result of ranking and correlation along with the weight calculation is shown in Table 5.6.

### 5.3 Testing Relative Risk of each Factor for Different Spectrum of GSD

On identifying the association among the features, we needed to find its risk towards the disease progression. Relative risk helps in measuring this association between exposed and non-exposed groups towards disease. Non-exposed are the group of people who are seen not to have any likelihood towards getting the disease. This measurement helps in finding the risk level they are in towards getting the disease. That is the probability of getting exposed. Relative risk is calculated using the equation shown in 5.3.

$$Relative\ risk = \frac{\frac{A}{(A+B)}}{\frac{C}{(C+D)}} \quad (5.3)$$

The values  $A, B, C, D$  and the illustrated values is shown in Table 5.7. Using the relative risk equation 5.3, we could interpret that the people with jaundice have 6.5 *times* higher risk towards GSD than who did not have GSD. On conducting the test for relative risk for each spectrum of GSD, we could analyze that pancreatitis was among the high-risk factors with relative risk of 98% followed by choledocholithiasis with 93% of the relative

Table 5.6: The statistical analysis of current study

| Attributes | Relieff Algorithms |           | Pearsons Correlation |          |             | $(\chi)^2$ |
|------------|--------------------|-----------|----------------------|----------|-------------|------------|
|            | Rank               | Weight    | Mean                 | SD       | Correlation |            |
| A1         | 2                  | 0.099978  | 0.627907             | 0.489083 | 0.36158     | <0.01      |
| A2         | 17                 | -0.002268 | 0.627907             | 0.489083 | -0.03204    | <0.01      |
| A3         | 22                 | -0.010773 | 0.883721             | 0.324353 | -0.14493    | 0.084385   |
| A4         | 6                  | 0.03987   | 0.255814             | 0.441481 | 0.256058    | <0.01      |
| A5         | 18                 | -0.003969 | 0.325581             | 0.474137 | -0.04721    | 0.429324   |
| A6         | 24                 | -0.014175 | 0.201551             | 0.67201  | -0.5929     | 0.395804   |
| A7         | 3                  | 0.072979  | 0.72093              | 0.45385  | 0.345259    | <0.01      |
| A8         | 4                  | 0.072226  | 0.488372             | 0.505781 | 0.314236    | <0.01      |
| A9         | 9                  | 0.028821  | 0.186047             | 0.39375  | 0.142128    | 0.020465   |
| A10        | 20                 | -0.007371 | 0.093023             | 0.293903 | -0.06855    | 0.760918   |
| A11        | 7                  | 0.039034  | 0.372093             | 0.489083 | 0.228848    | 0.290007   |
| A12        | 23                 | -0.012474 | 0.418605             | 0.499169 | -0.47087    | 0.239873   |
| A13        | 26                 | -0.017577 | -0.356588            | 1.01868  | -0.93905    | 0.668374   |
| A14        | 1                  | 0.595627  | 0.511628             | 0.505781 | 0.827636    | <0.01      |
| A15        | 19                 | -0.00567  | 0.046512             | 0.213083 | -0.04727    | 0.255014   |
| A16        | 5                  | 0.047601  | 0.790698             | 0.411625 | 0.285507    | <0.01      |
| A17        | 32                 | -0.027783 | -2.03100             | 2.05869  | -1.97735    | 0.337364   |
| A18        | 12                 | 0.006237  | 0.093023             | 0.293903 | 0.095206    | 0.454107   |
| A19        | 8                  | 0.031822  | 0.209302             | 0.411625 | 0.182181    | <0.01      |
| A20        | 30                 | -0.024381 | -1.472866            | 1.71202  | -1.63125    | 0.638959   |
| A21        | 16                 | -0.000567 | 0.162791             | 0.373544 | -0.02996    | 0.966678   |
| A22        | 25                 | -0.015876 | -0.0775185           | 0.84534  | -0.76600    | 0.431841   |
| A23        | 31                 | -0.012474 | -1.75193             | 1.88535  | -1.8043     | 0.429324   |
| A24        | 29                 | -0.02268  | -1.1937              | 1.5386   | -1.4582     | <0.01      |
| A25        | 15                 | 0.001134  | 0.883721             | 0.324353 | 0.003451    | 0.252252   |
| A26        | 11                 | 0.017634  | 0.162791             | 0.373544 | 0.098879    | 0.795061   |
| A27        | 21                 | -0.009072 | 0.976744             | 0.152499 | -0.12477    | 0.966678   |
| A28        | 10                 | 0.024863  | 0.813953             | 0.39375  | 0.102332    | 0.956645   |
| A29        | 27                 | -0.019278 | -0.635657            | 1.19201  | -1.1121     | <0.01      |
| A30        | 13                 | 0.004536  | 0.093023             | 0.293903 | 0.095206    | 0.239873   |
| A31        | 28                 | -0.020979 | -0.914727            | 1.36535  | -1.2851     | <0.01      |
| A32        | 14                 | 0.002835  | 0.139535             | 0.350605 | 0.051078    | 0.977675   |

risk. People with uncomplicated GSD has a very less relative risk of lesser than 50%. The Figure 5.3 shows the relative strength of each spectrum. The size of square gives the graphical idea of relative risk, as it keeps moving towards 10<sup>2</sup>%.

Table 5.7: Illustration of feature selection and classification

|                         | GSD ( <i>Yes</i> ) | GSD ( <i>No</i> ) | Row Total         |
|-------------------------|--------------------|-------------------|-------------------|
| Jaundice ( <i>Yes</i> ) | 240 (A)            | 150 (B)           | 390 (A+B)         |
| Jaundice ( <i>No</i> )  | 20 (C)             | 120 (D)           | 140 (C+D)         |
| <b>Total</b>            | 260 (A+C)          | 270 (B+D)         | 530 ((A+B)+(C+D)) |

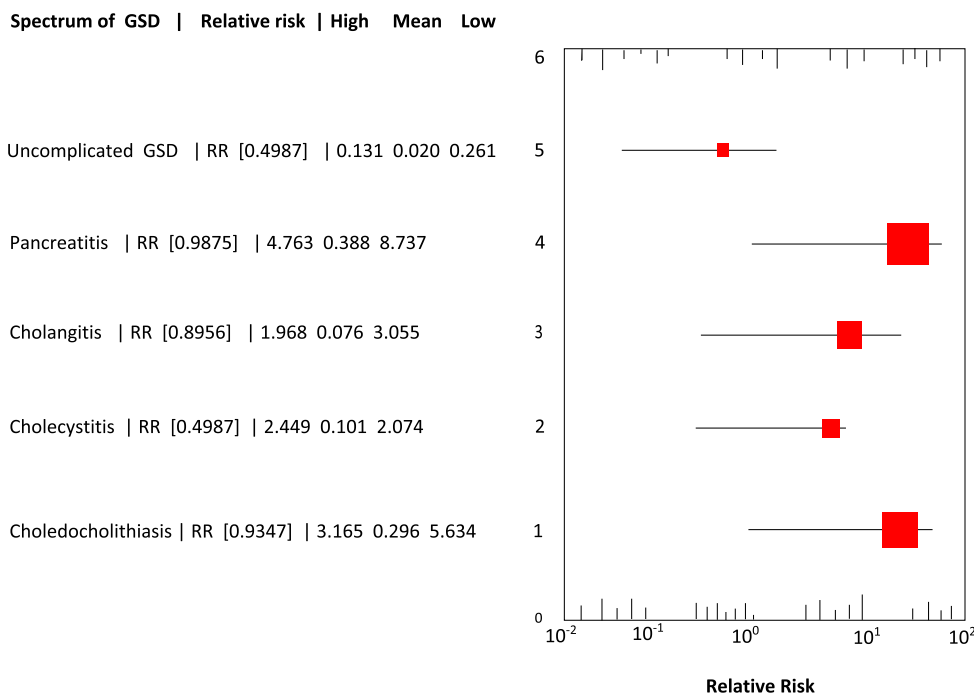


Figure 5.3: Relative strength of treatment effects in different spectrum of GSD.

The relative risk of each complicated cases of GSD was analysed and is shown in Figure 5.4. The relative risk is calculated in a retrospective way and is studied based on the disease progression for every hour, from the time of admission. On analysis, we could find that all the patients reached normal stage within two days of admission. But we could observe that the thirteen critical cases also descended towards normal as the initial treatment progressed. This would have been the reason the ERCP was conducted on them in the later stage of the disease progression.

On further risk analysis we could find that those thirteen critical cases had an incline showing high relative risk. The clinical decision support system (*CDSS*) developed by us aimed in identifying those thirteen cases and predicting the disease progression at the time of admission itself. Among the complicated cases of GSD, *CDSS* aimed in identifying the cases which may become critical as the disease progresses. Those cases which were not critical were found to become normal. This analysis will help in giving more attention to such critical cases and avoid any later stage complications.

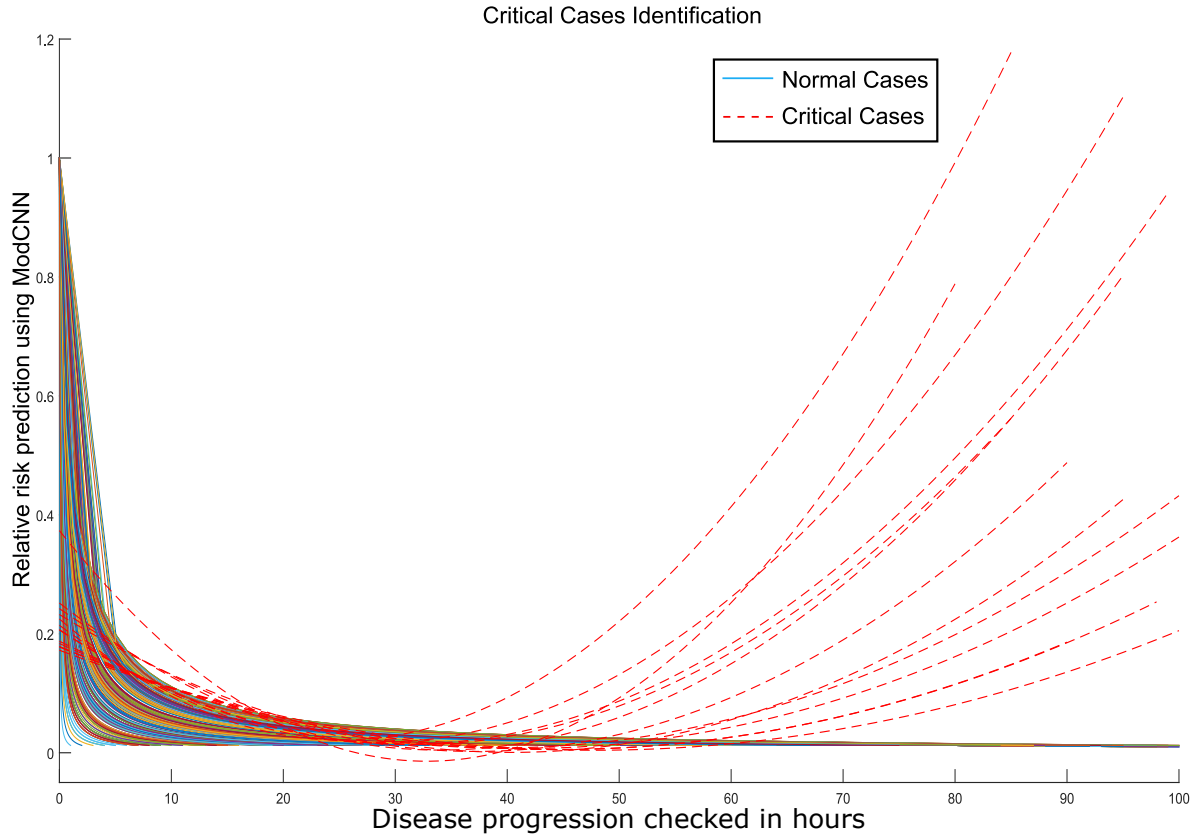


Figure 5.4: Analysing the disease progression and detecting the critical cases

## 5.4 Performance Comparison of ModCNN with ANN and CCNN

The spectrum of GSD was identified using ModCNN classification technique in-order to recognize the rare input pattern. This classification accuracy was evaluated by comparing it with ANN and CCNN. The result of this comparison is shown in Figure 5.5. For illustrating and showing the performance comparison of ModCNN, ANN and CCNN, we have considered ten patients data with equal distribution of different spectrum of GSD. Here, ten patients data is considered in order to show the performance of each model at different epochs, avoid spaghetti-like graphs and give better explanation.

The inputs are patients clinical data and output is the spectrum of GSD (“0” cholecystitis, “1” choledocholithiasis, “2” pancreatitis and “3” cholangitis). It was observed that, ModCNN achieved MSE=0.00 (classified output) at 1283 epochs, while CCNN and ANN still needed few more epochs to complete the classification process. Figure 5.6 shows the rate of error decrease and it is observed that, the convergence of ModCNN was faster than other models.

Model Comparison to Classify the Spectrun of GSD

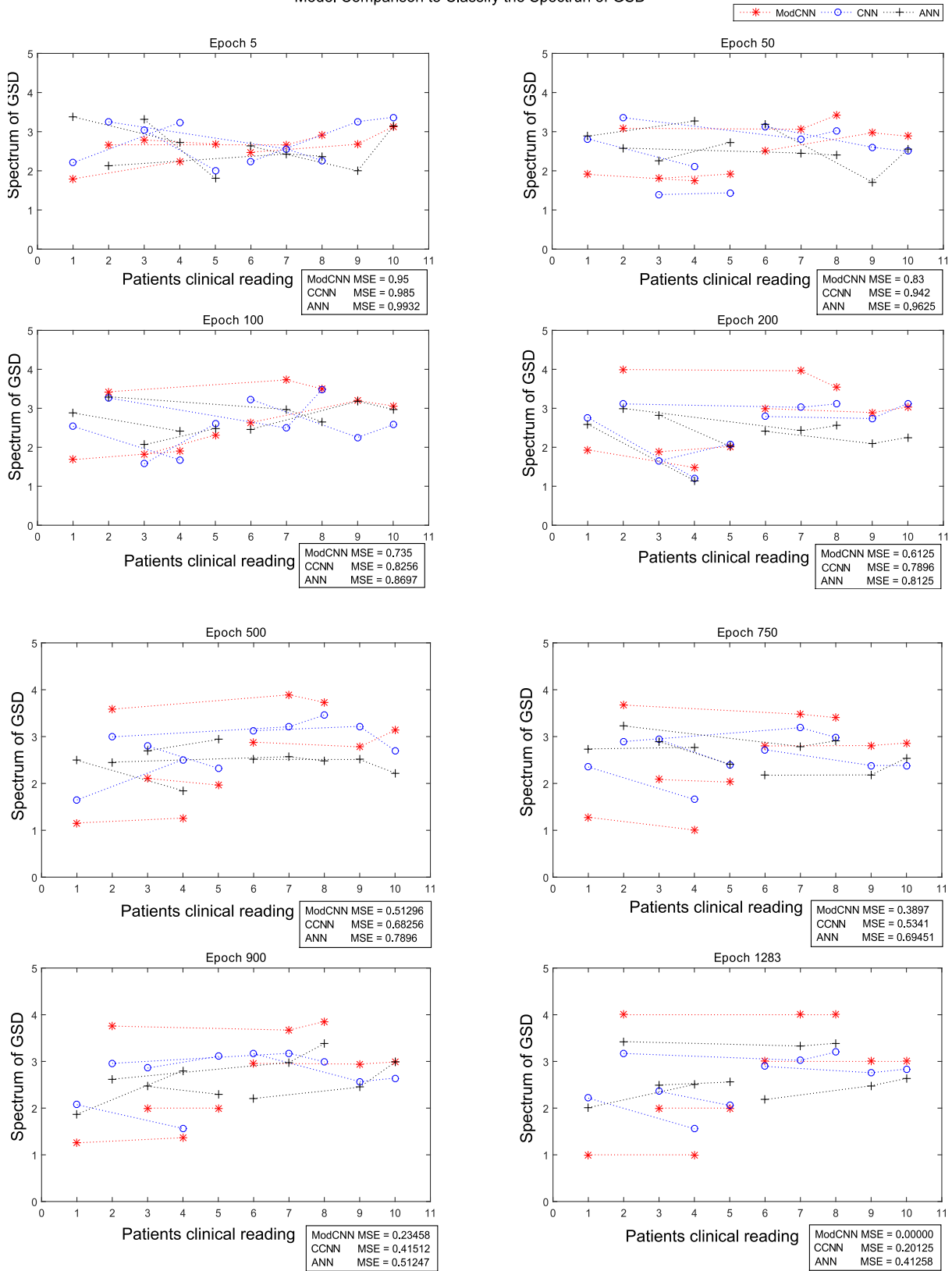


Figure 5.5: Performance comparison of ModCNN, CCNN and ANN for classifying spectrum of GSD .

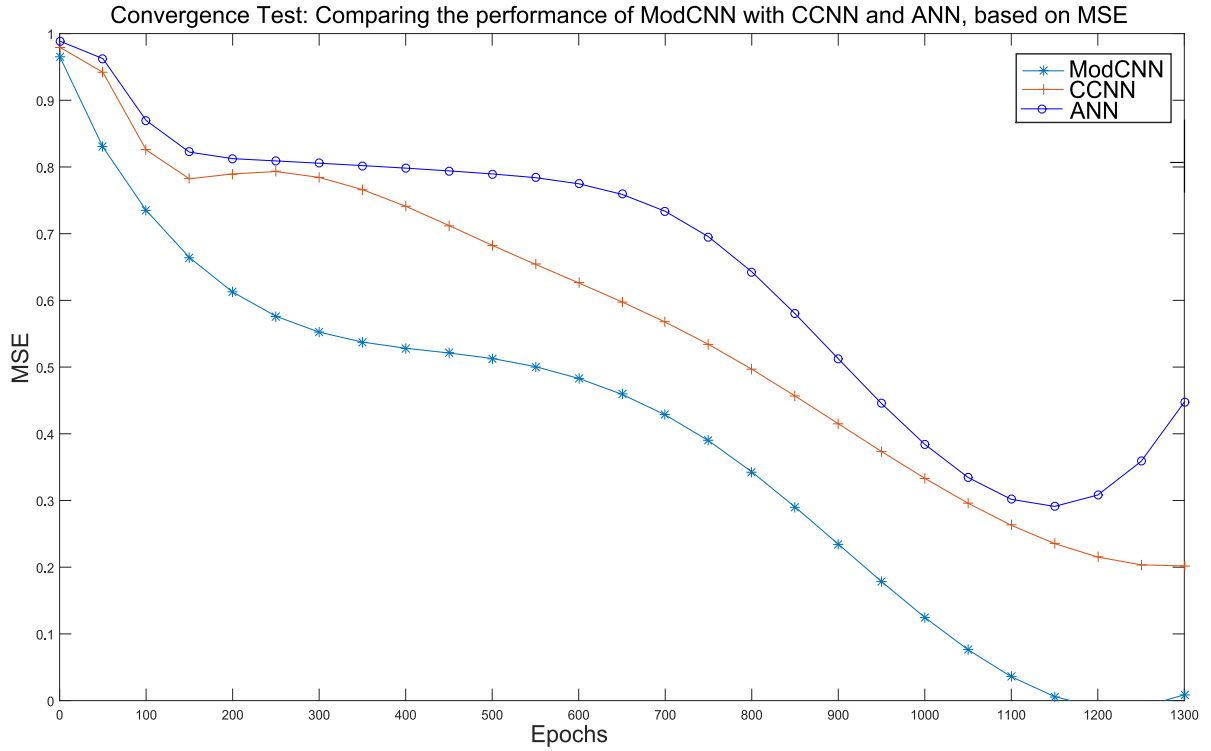


Figure 5.6: Classification performance of ModCNN, CCNN and ANN.

## 5.5 Validation of ModCNN

Along with the statistical analysis for finding the risk factors, we used ANN, CCNN, and ModCNN for discovering the significant factors associated with each spectrum of GSD. Each model identified a different set of factors. Figure 5.7 shows the factors identified along with performance comparison using  $A_Z$ . A1 to A32 are the clinical and USG findings and are shown in the Figures 3.1,3.2,3.3,3.4 and 3.5. Each stack in the stacked bar graph is the factor shown as the significant by different techniques. The value on top of the stacked bar is the value of  $A_Z$  for that model. Higher the  $A_Z$ , factors are more significant. On comparison, it was seen that ModCNN outperformed in the accuracy of prediction when compared with ANN and CCNN. The significant factors were validated by testing for accuracy of prediction using the concept of  $A_Z$  and are tabulated in Table 5.8. A Probability level of a random difference of  $P < 0.05$  was considered as significant independent predictors using  $\chi^2$ .  $A_Z$  is a ROC curve (plotted with sensitivity versus 1-specificity) that performs the comparison of different tests and chooses the best model.  $A_Z = 1$  is known as perfect discrimination, and 0.5 is referred to as absence of discrimination.  $A_Z$  was calculated using all variables (N=29) for each model.

Independent predictors identified by ANN, ModCNN and CCNN

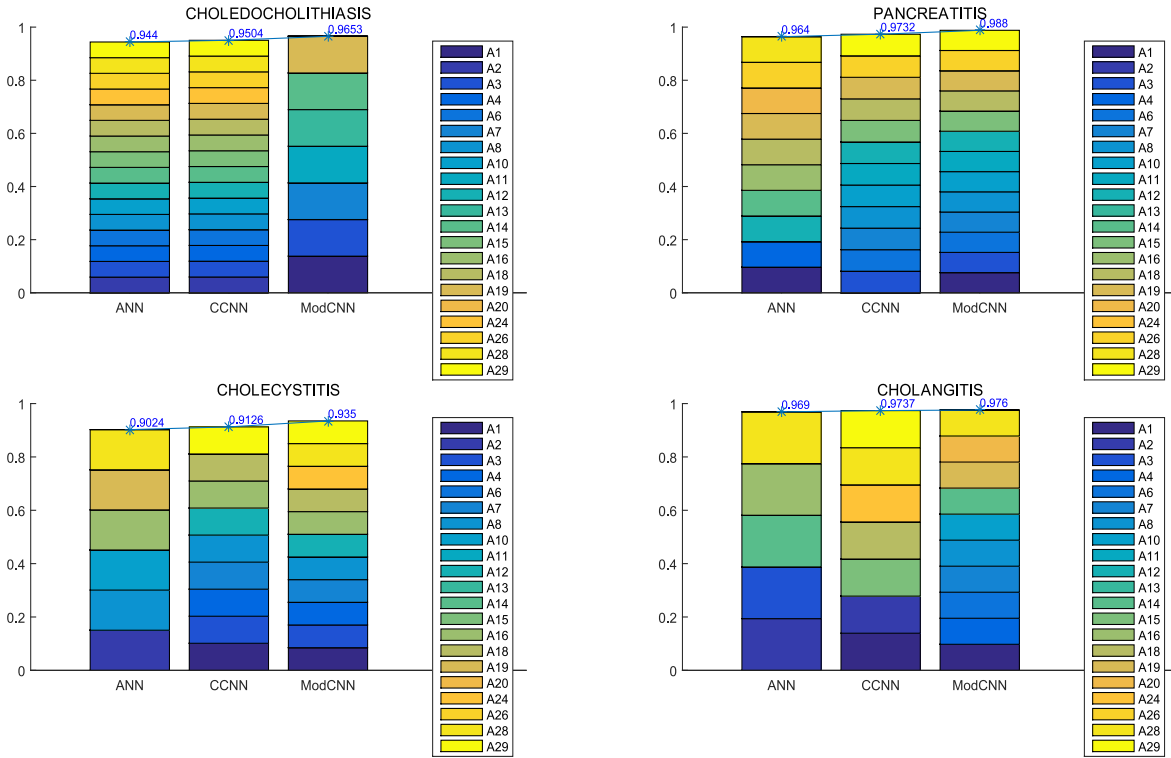


Figure 5.7: Comparison of ANN, CCNN and ModCNN for different spectrum of GSD

## 5.6 Testing for Accuracy in Prediction and Detection of Critical Cases

$A_Z$  is one of the well established statistical technique for evaluating the model performance. Higher the area under the curve more is the accuracy of prediction. The curve is obtained by plotting for *sensitivity* against  $(1 - \textit{specificity})$ . TP (True Positive) is when the people with the disease is classified as positive, and FN (False Negative) is when they are classified as negative. TN (True Negative) is when people with no disease are correctly classified as negative, and FP (False Positive) is when they are classified positive. Sensitivity and specificity can be defined using the Table 7.4. Sensitivity and Specificity is obtained using equation 7.3 and 6.1 respectively. On plotting the obtained values for each feature, we will be able to get  $A_Z$ .



Table 5.8: Factors associated with each spectrum of GSD and  $A_Z$  of ModCNN. Each factor is parenthesized with its  $P$  value.

| Spectrum of GSD     | Factors Associated   | $A_Z$  |
|---------------------|--|--------|
| Cholangitis         | A1( $P = 0.1969$ ), A4( $P = 0.1190$ ),<br>A7( $P < 0.001$ ), A8( $P=0.2483$ ),<br>A9( $P < 0.001$ ), A10( $P = 0.8016$ ),<br>A14( $P = 0.7571$ ), A19( $P = 0.2094$ ),<br>A21( $P = 0.5802$ ), A26( $P < 0.001$ )   | 0.9768 |
| Pancreatitis        | A1( $P < 0.001$ ), A4( $P = 0.6994$ ),<br>A7( $P = 0.4563$ ), A8( $P < 0.001$ ),<br>A9( $P < 0.001$ ), A10( $P < 0.001$ ),<br>A12( $P = 0.6040$ ), A14( $P = 0.8876$ ),<br>A19( $P < 0.001$ ), A21( $P = 0.4855$ ),<br>A22( $P = 0.6040$ ), A26( $P = 0.5300$ ),<br>A28( $P < 0.001$ ) | 0.9875 |
| Cholecystitis       | A1( $P < 0.001$ ), A3( $P = 0.1568$ ),<br>A4( $P < 0.001$ ), A7( $P = 0.7350$ ),<br>A8( $P < 0.001$ ), A11( $P < 0.001$ ),<br>A16( $P < 0.001$ ), A17( $P = 0.2145$ ),<br>A24( $P = 0.0656$ ), A26( $P = 0.6537$ ),<br>A29( $P = 0.9828$ )   | 0.9348 |
| Choledocholithiasis | A1( $P < 0.001$ ), A3( $P = 0.9179$ ),<br>A7( $P < 0.001$ ), A11( $P = 0.2689$ ),<br>A13( $P = 0.4730$ ), A14( $P = 0.2689$ ),<br>A19( $P = 0.5364$ )  | 0.9653 |

Table 5.9: Representation of TP (A), FN (B), FP (C) and TN (D)

| Test         | GSD (Yes)  | GSD (No)   | Row Total |
|--------------|------------|------------|-----------|
| Positive     | TP (A)     | FP (C)     | A + C     |
| Negative     | FN (B)     | TN (D)     | B + D     |
| <b>Total</b> | <b>A+B</b> | <b>C+D</b> |           |

$$Sensitivity = \frac{A}{A + B} \quad (5.4)$$

$$Specificity = \frac{D}{C + D} \quad (5.5)$$

## 5.7 Performance Measurement of ANN, CCNN and ModCNN

The current work aimed to build a model that could function more efficiently in identifying the cases that may become critical as the disease progress. This has to be done at the time of admission itself. On study, we came through many scoring systems for predicting the disease progression, but most of them can predict only after 48 hours of admission. On analysing the Figure 5.4, we can understand that all the 13 cases became complicated within 48 hours after admission. Hence applying APACHE (Knaus et al., 1985) or Balthazar (Balthazar et al., 1990) which are well-established tools would delay the process of treatment management. We needed a technique that could predict the disease behaviour at the time of admission and reduce the medical error.

The performance of proposed ModCNN was compared with ANN and CCNN. The Figure 5.8 shows the initial contour graph of ANN, CCNN and ModCNN, Figure 5.9 shows the intermediate stage and the Figure 5.10 shows that ModCNN has already converged, where in ANN and CCNN still needed more epochs to complete. This evaluation shows that proposed ModCNN is an efficient model when compared to ANN and CCNN. The efficiency was defined in terms of *the rate of convergence* as-well-as *the accuracy of prediction*.

## 5.8 Accuracy Measurement Using the Concept of $A_Z$

The performance of ModCNN was evaluated and compared with ANN and CCNN. ModCNN showed better accuracy when tested for  $A_Z$  with,  $A_Z = 0.9768, 0.9875, 0.9348$  and  $0.9653$  for cholangitis, pancreatitis, cholecystitis and choledocholithiasis respectively. The independent predictors associated with each spectrum of GSD along with their P-value is tabulated in Table 5.8 and the comparison of  $A_Z$  for each spectrum is shown in 5.11, 5.12, 5.13 and 5.14. The overall accuracy comparison shows that ModCNN had better accuracy of  $A_Z = 0.9642$  when compared to CCNN ( $A_Z = 0.9324$ ) and ANN ( $A_Z = 0.8965$ ). This is shown in Figure 5.15.

## 5.9 Summary

In this chapter the performance of ModCNN was evaluated and compared with ANN and CCNN. The patients data was recorded, who came with the complaint of abdominal pain

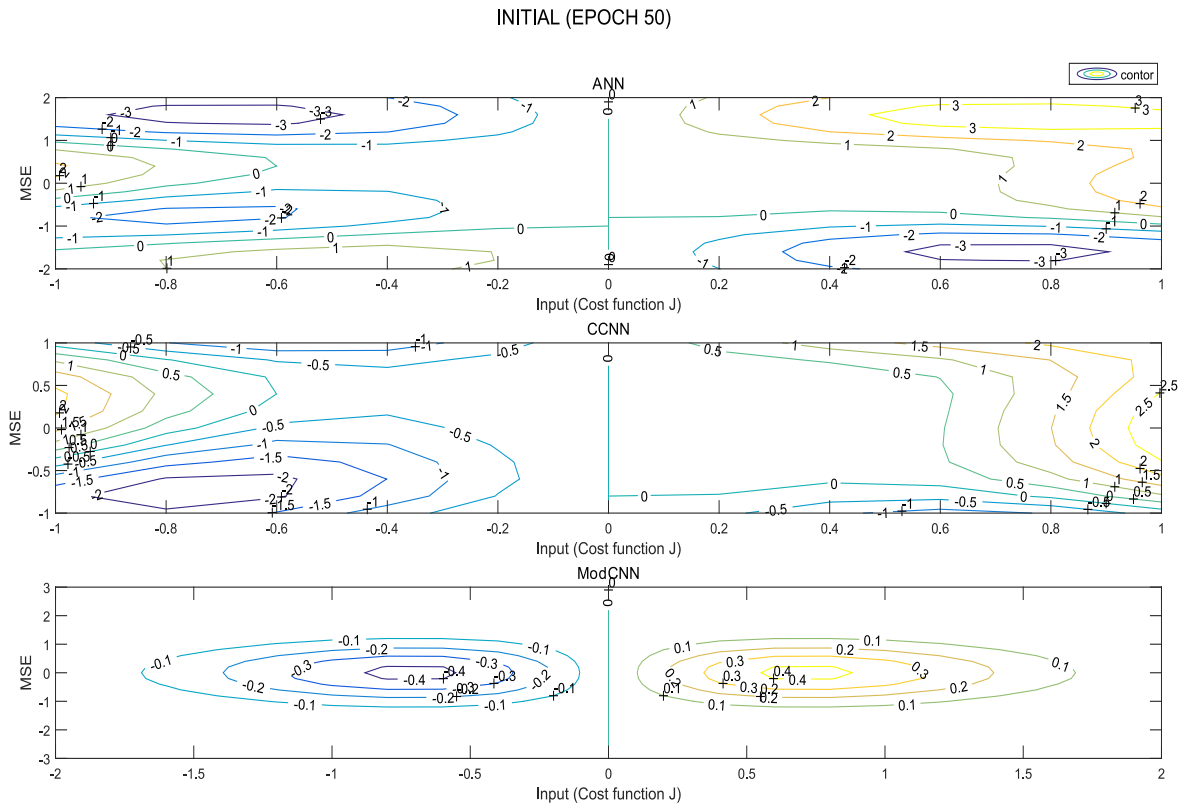


Figure 5.8: Initial Stage: Decrease in MSE showing classification performance

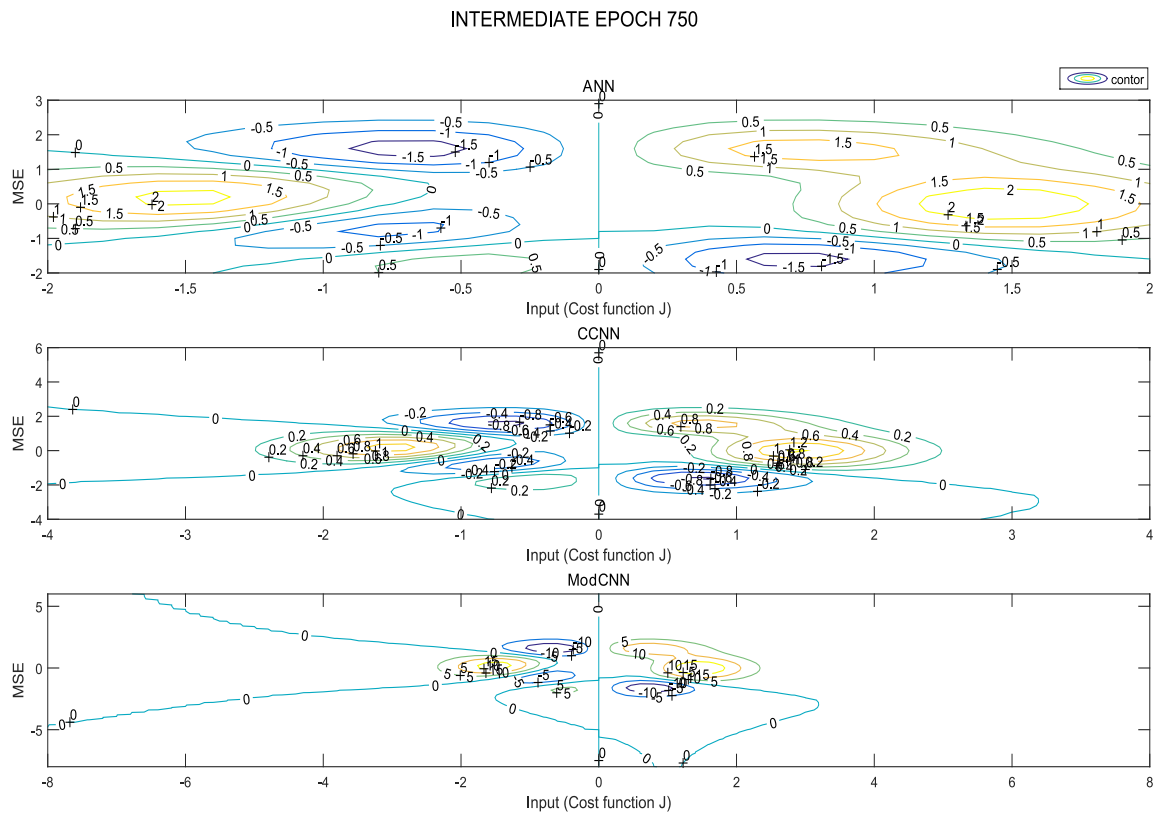


Figure 5.9: Intermediate Stage: Decrease in MSE showing classification performance

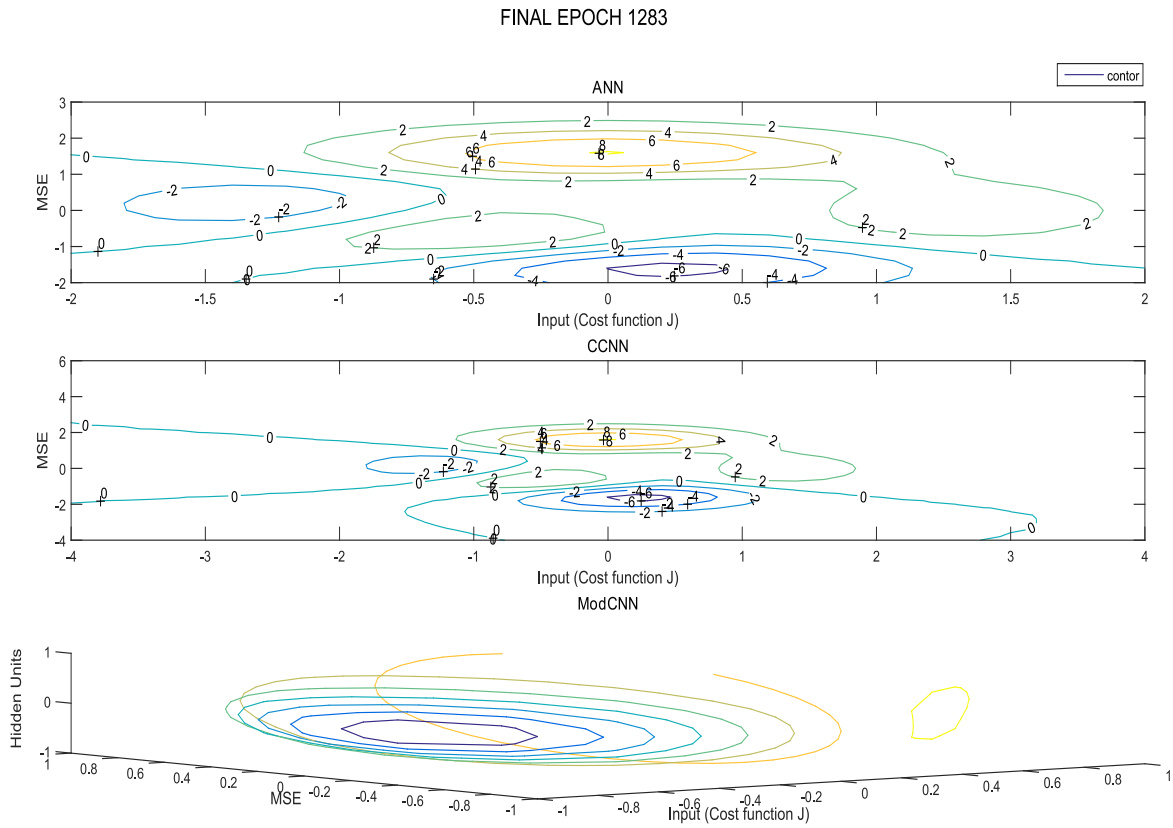


Figure 5.10: Final Stage: Decrease in MSE showing classification performance

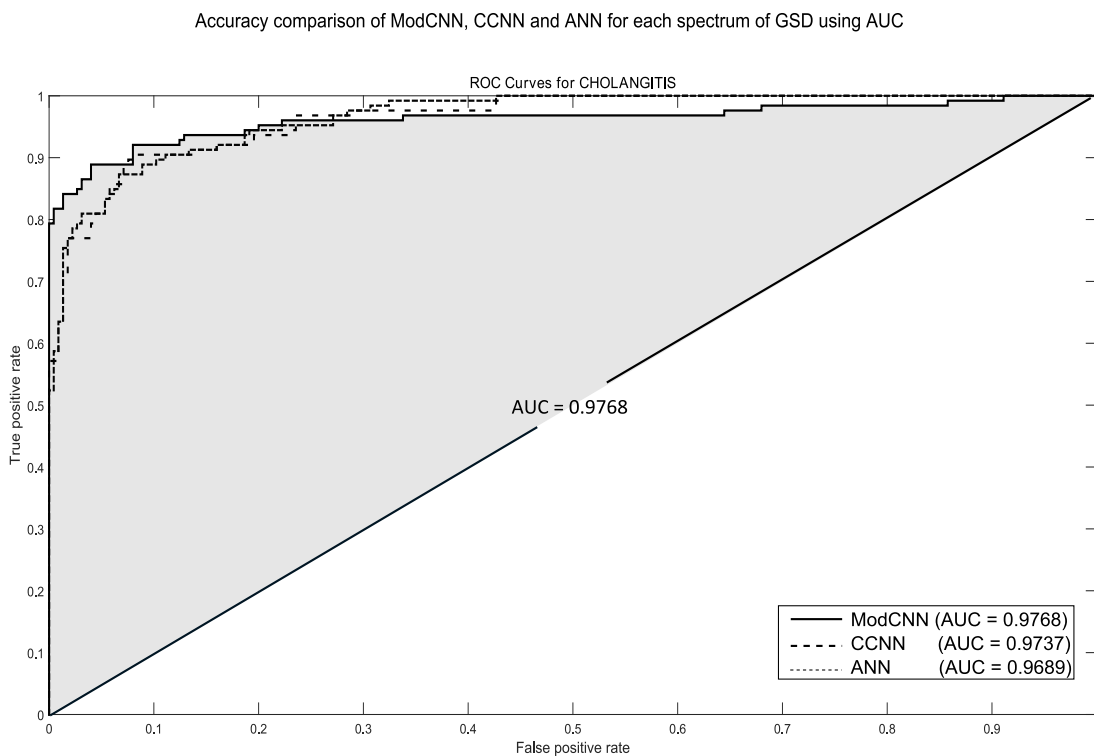


Figure 5.11: Comparison of accuracy of prediction for cholangitis using  $A_Z$

during the period of 2014 to 2015 at territory care centre in north malabar Kerala, India. The study focused on complicated cases of GSD. ModCNN was successful in stratifying

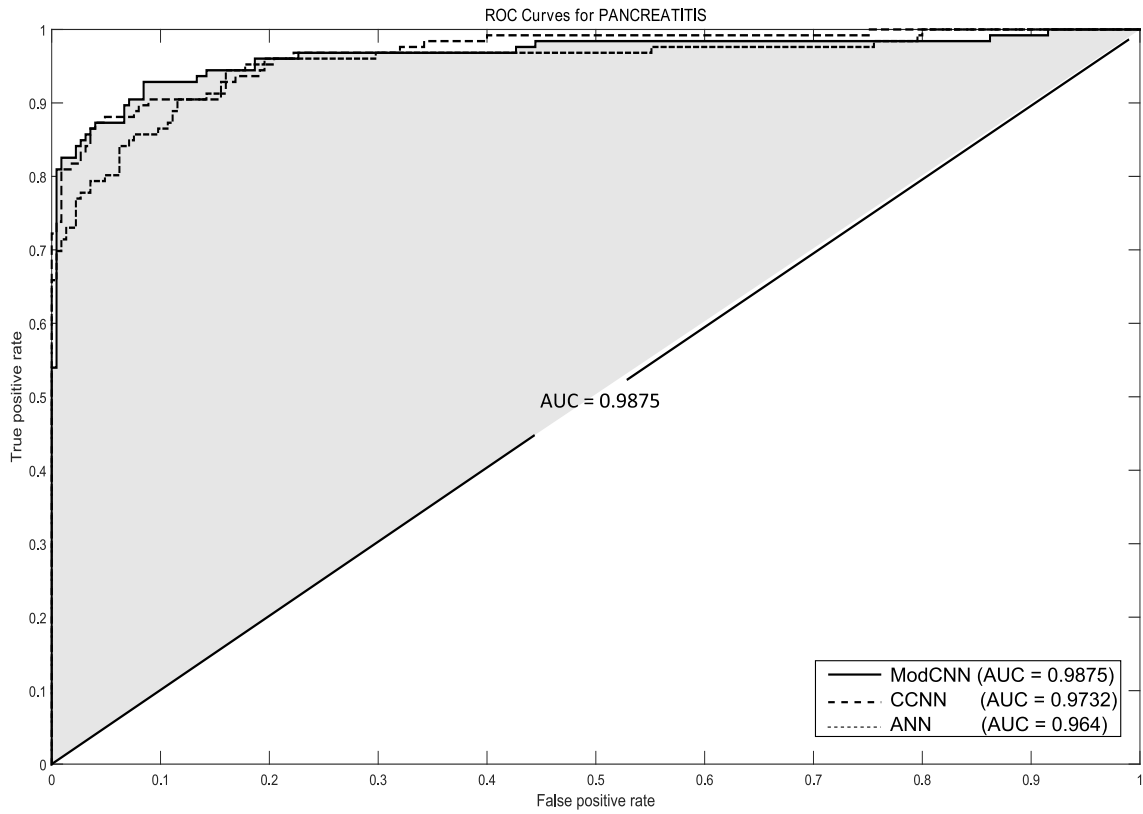


Figure 5.12: Comparison of accuracy of prediction for pancreatitis using  $A_Z$

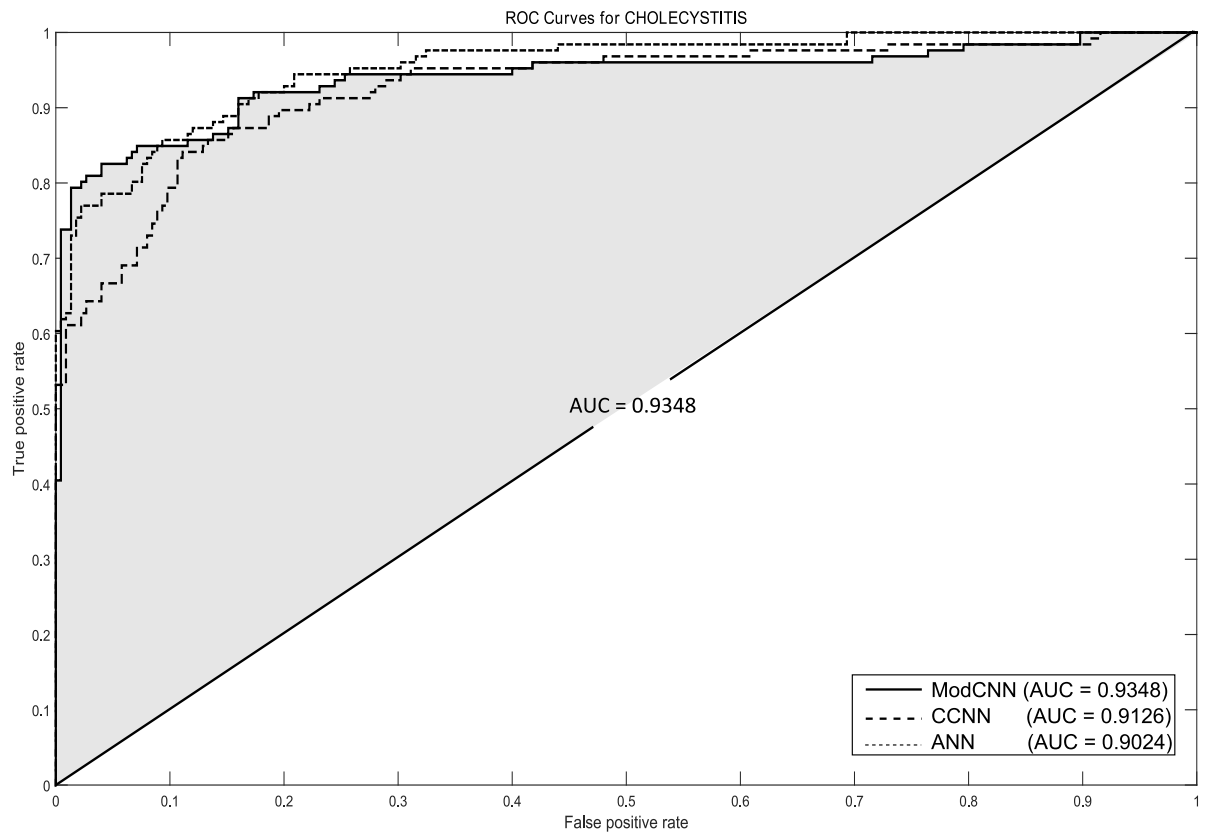


Figure 5.13: Comparison of accuracy of prediction for cholecystitis using  $A_Z$

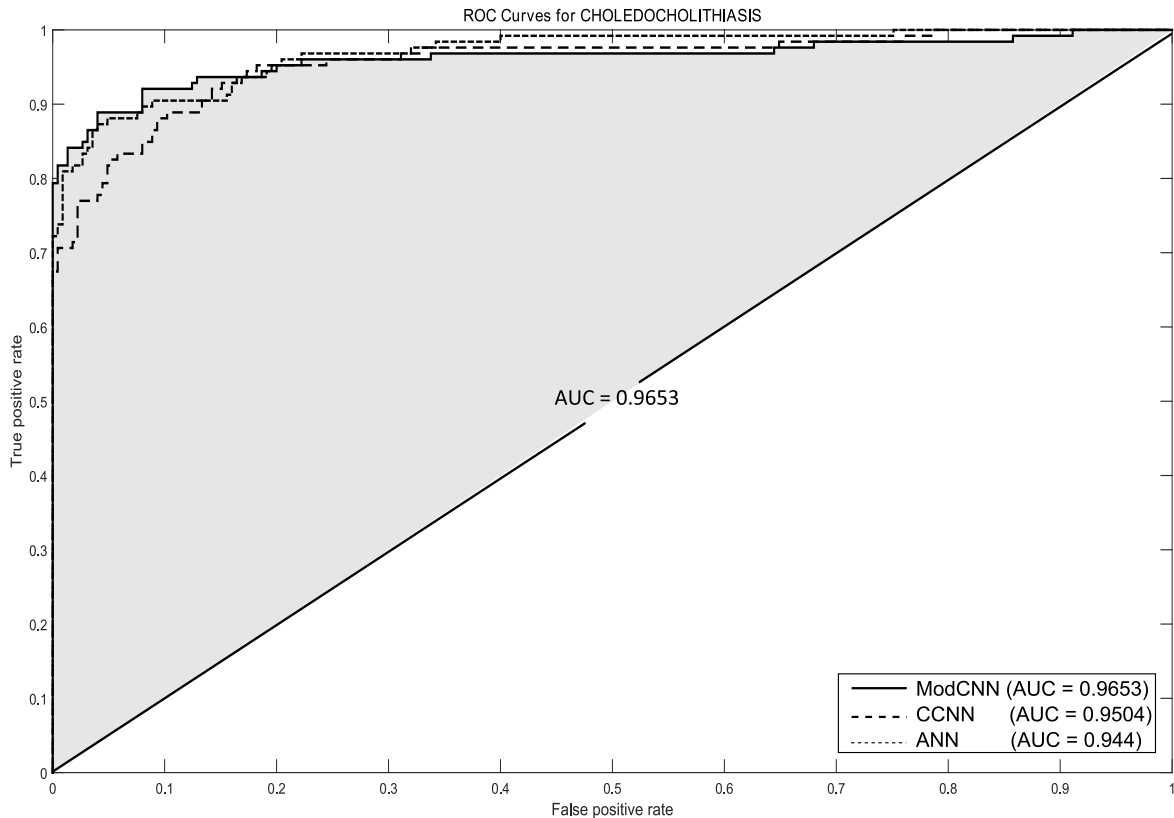


Figure 5.14: Comparison of accuracy of prediction for choledocholithiasis using  $A_Z$

different spectrum of GSD and also predicting the disease behaviour. Number of cases found in this current study comparable with the modes of presentation of California study (Glasgow et al., 2000). This comparison showed that the incidence of choledocholithiasis and pancreatitis is increasing in the subjected region.

Total of 32 features were observed, out of which the significant features were identified using  $\chi^2$  test, along with their relative risk factors. These significant factors were fed into ModCNN, ANN and CCNN to evaluate their performance. On validating, it was studied that ModCNN achieved MSE=0.00 at 1283 epochs, while CCNN and ANN still needed few more epochs to complete the classification process. Further the accuracy of prediction was evaluated using the concept of  $A_Z$ . The comparison showed that ModCNN was able to accurately identify thirteen complicated cases of GSD on whom emergency intervention was needed with an accuracy of 96.42%. Thus the experimental result showed that the proposed ModCNN was able to identify the complicated cases more accurately than ANN and CCNN.

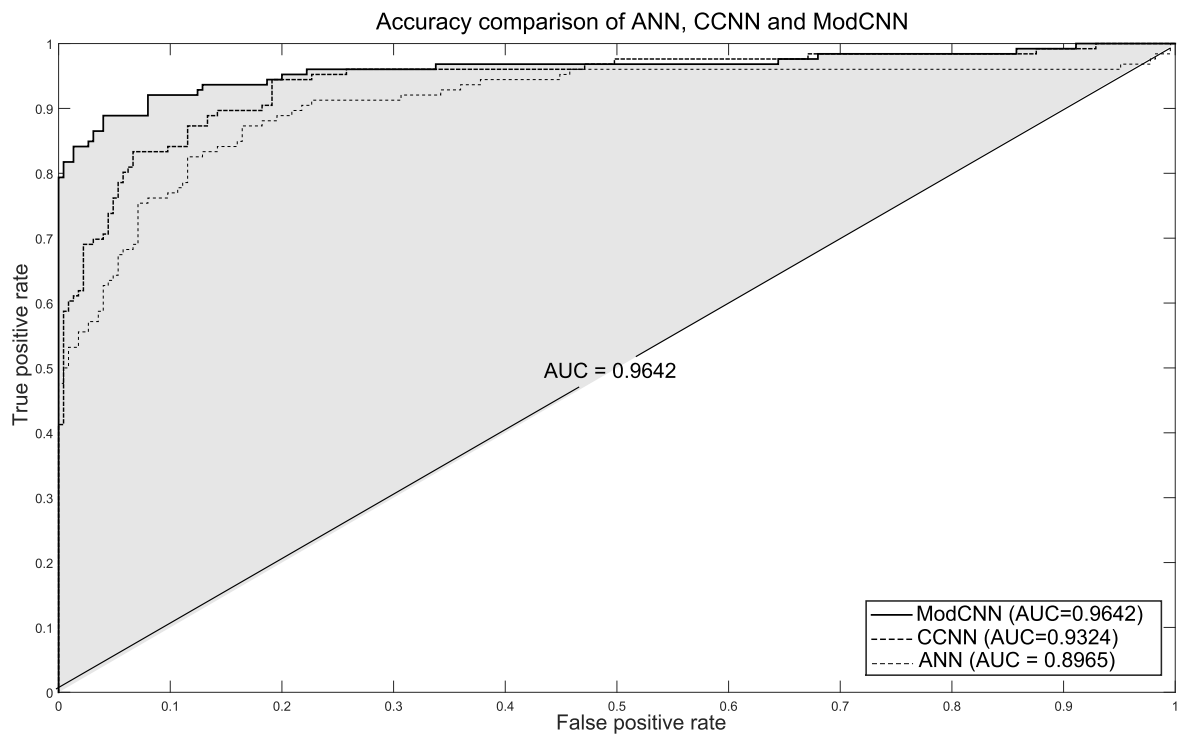


Figure 5.15: Accuracy comparison of ModCNN with ANN and CCNN  $A_Z$





# Chapter 6

## Process Mining in Healthcare System

In the healthcare system, patients have to go through many non-trivial processes which are time-consuming. The treatment procedures may vary from one patient to other based on their age, sex, and medical history. This procedure is multi-discipline, and each discipline follows their way of process execution. Hence, there must be a proper interaction between these discipline for successful process execution. This interaction is done through the messages as shown in Figure 6.1.

Process execution in a healthcare system needs proper monitoring and control (Dadam et al., 2000). Process mining supports in capturing the knowledge of organizational workflow. This would help in conducting proper coordination between healthcare professional and organizational units (Lenz and Reichert, 2007). Process mining has a huge class of techniques assisting in process evaluation and monitoring. Using this, we were able to manage execution of an individual activity at *the right time by the efficient resource* (Ter Hofstede et al., 2009; Weske, 2010; Dumas et al., 2005; Van der Aalst et al., 2004). The main advantage of applying process mining in healthcare domain is, it would reduce the resource cost along with waiting time of the patients (Adams et al., 1999). This chapter will discuss the application of process mining techniques for:

- Recommending the critical treatment path sequenced with critical activities.
- Identifying the adequate resources for conducting the critical activity.
- Clustering the traces and finding the Next Probable Activity (NPA).
- Assisting in load balancing of the resources.
- Predicting the length of stay in the hospital.

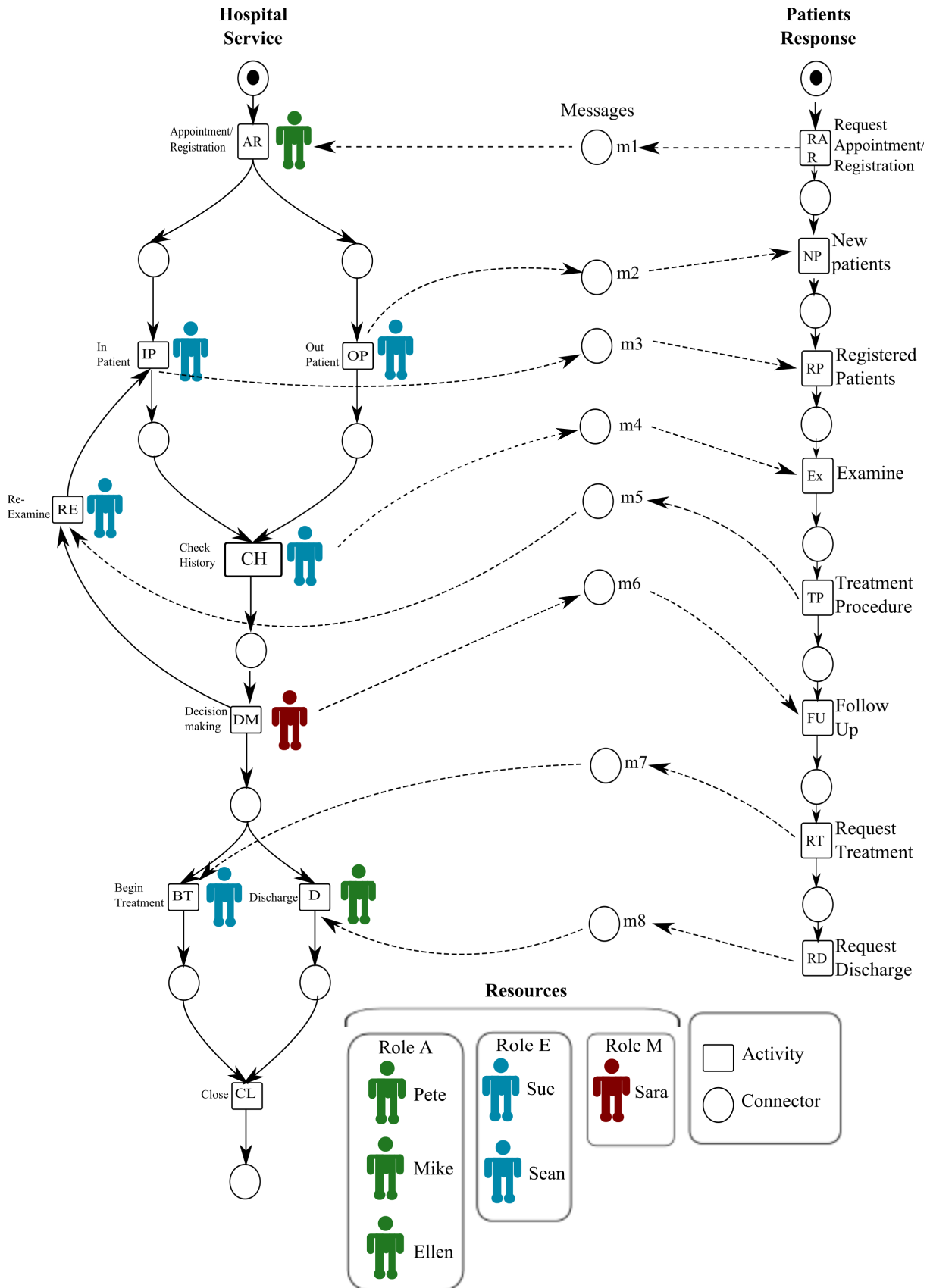


Figure 6.1: Typical process model of a healthcare system.

## 6.1 Introduction

ModCNN was applied for assisting the clinicians in taking appropriate clinical decisions by finding the critical cases at the time of admission. The critical path of treatment is recommended for those cases which were found to be critical. This path is the safest careflow, supported by adequate resources for completion of each activity. Thus, making the journey of a patient more safer and faster. Process mining was used here for providing the careflow to the patients. Process mining *discovers* the process models and *monitors* their behaviour. If any drift is observed in process execution, it would predict and suggest the changes to improve the process behaviour. Process mining is built upon three main pillars: process discovery, conformance, and enhancement. The architecture is shown in Figure 1.7. As event logs are the starting point for process mining, EHR serves as an input for process mining.

### 6.1.1 Electronic Health Record (EHR)

The illustration of EHR is shown in Figure 6.2. It is important to understand the structure of EHR along with the attributes needed for process analysis. In the current study, we focused on *patient record*, *medications* and *encounter history* for analysing EHR and recommending the right critical path. EHR can be referred as a multi-set of traces and *traces* are the sequence of activities shown in medication section of Figure 6.2. Let  $A_1 = \{AR, IP, OP, CH, DM, RE, BT, D\}$  and  $A_2 = \{RAR, NP, RP, Ex, TP, FU, RT, RD\}$  be set of activities shown in the Figure 6.1. Let  $\langle AR, IP, CH, DM, D \rangle$  and  $\langle RAR, NP, RP, Ex, TP, FU, RT, RD \rangle$  be the trace of  $A_1$  and  $A_2$  respectively. Set of all the traces executed through process execution forms an event log  $\mathcal{L}_1 = [\langle AR, IP, CH, DM, D \rangle^{21}, \langle AR, OP, CH, DM, D \rangle^{30}, \langle AR, IP, CH, DM, BT, D \rangle^{25}]$  and  $\mathcal{L}_2 = [\langle RAR, NP, RP, Ex, TP, FU, RT, RD \rangle^{85}]$ . In this event log ( $\mathcal{L}_1$  and  $\mathcal{L}_2$ ), there are three and one instances of traces respectively, and the superscripted values  $\{21, 30, 25 \text{ and } 85\}$  are the number of times those traces were executed. Figure 6.1 shows the interaction between two model using the messages  $\mathcal{M} = [\langle m1, m2, m3, m4, m5, m6, m7, m8 \rangle]$ . The goal of application of process mining in healthcare process is to discover an optimal model (critical path of treatment) with an efficient resource for handling each activity in the path of execution.


| Help   | Patients Details  |                   |  | Healthcare Service Providers                   |                    |                      |                   |                   |                    |  |  |
|--|---|-------------------|--|--|--------------------|----------------------|-------------------|-------------------|--------------------|--|--|
| Logout   |  | IME0011           |  |  |                    | <b>Name</b>          | <b>Dept.</b>      | <b>Last Visit</b> | <b>Next Visit</b>  |  |  |
|  |   | Shwetha           |  |  |                    | Laxman, Srinivas     | Cardiology        | 01/2006           | 07/2006            |  |  |
|  |   | <b>Aadhar No:</b> |  |  |                    | Meenaxi, Madhu       | RN                | 08/2005           | 11/2005            |  |  |
|  |   | 125678943652      |  |  |                    | Mohan, Jacob         | Dermatology       | 07/2005           | 12/2005            |  |  |
|  | <b>Sex:</b>   | <b>Phone:</b>     |  |  |                    |                      |                   |                   |                    |  |  |
|  | Female  | 91-0825-25369     |  |  |                    |                      |                   |                   |                    |  |  |
|  | <b>DOC:</b>   | <b>Address:</b>   |  |  |                    |                      |                   |                   |                    |  |  |
|  | 1940/01/01  | 19-Kotiabele      |  |  |                    |                      |                   |                   |                    |  |  |
|  |   | Mangalore         |  |  |                    |                      |                   |                   |                    |  |  |
|  |   | Karnataka         |  |  |                    |                      |                   |                   |                    |  |  |
| Patient Record   | Alerts  |                   |  | Medications of Chelecystectomy Done on 05/1981 |                    |                      |                   |                   |                    |  |  |
| >Summary<br>>Lab Result<br>>Diagnostic<br>>Images<br>>Details<br>>Notes or Comment | Allergies -Sulfa Drugs  |                   |  | <b>Name</b>                                    | <b>Dept.</b>       | <b>Activity Name</b> | <b>Start Time</b> | <b>End Time</b>   |                    |  |  |
|  | > Pap Smear Due   |                   |  | Ramu (green)                                   | Front office       | Registration         | 01/05/1981        | 01/05/1981        |                    |  |  |
|  | > Tp Due  |                   |  | Mohan (green)                                  | Front office       | Taken to doctor      | 01/05/1981        | 01/05/1981        |                    |  |  |
|  | > A1C above target  |                   |  | Sarita (blue)                                  | Nursing            | Preliminary check    | 10:30 AM          | 10:45AM           |                    |  |  |
|  |   |                   |  | Saxena (red)                                   | Surgery            | Doc. Consultation    | 01/05/1981        | 01/05/1981        |                    |  |  |
|  |   |                   |  |  |                    |                      | 01:00 PM          | 01:55 PM          |                    |  |  |
|  |   |                   |  | <b>Encounter History</b>                       |                    |                      |                   |                   |                    |  |  |
|  |   |                   |  | <b>Date</b>                                    | <b>Facility</b>    | <b>Speciality</b>    | <b>Clinicians</b> | <b>Reason</b>     | <b>Type</b>        |  |  |
|  |   |                   |  | 02/2006  | GP                 |                      |                   | Hypertension      | -                  |  |  |
|  |   |                   |  | 01/2006  | Cardio Assoc.      | Cardiology           | Laxman            | CAD               | OP                 |  |  |
|  |   |                   |  | 12/2005  | GP                 |                      |                   | Diabetes          | -                  |  |  |
|  |   |                   |  | 10/2005  | General Hosp.      | Dietician            | John              | Diab. Teaching    | OP                 |  |  |
|  |   |                   |  | 08/2005  | GP                 |                      |                   | Diabetes          | -                  |  |  |
|  |   |                   |  | 08/2005  | GP                 |                      | Rajesh            | Cellulitis        | -                  |  |  |
|  |   |                   |  | 08/2005  | Home Visit         | RN                   |                   | Cellulitis        | -                  |  |  |
|  |   |                   |  | <b>Immunization</b>                            |                    |                      |                   |                   |                    |  |  |
|  |   |                   |  | <b>Type</b>                                    | <b>Most Recent</b> | <b>No.</b>           | <b>Type</b>       | <b>Value</b>      | <b>Most Recent</b> |  |  |
|  |   |                   |  | Influenza                                      | 11/2005            | 7                    | A1C               | 0.071             | 12/2005            |  |  |
|  |   |                   |  | Preumovax                                      | 03/2005            | 1                    | LDL               | 2.41              | 12/2005            |  |  |
|  |   |                   |  | Twinrix  | 08/2005            | 1                    | BP                | 135/75            | 02/2006            |  |  |
|  |   |                   |  | Td   | 04/1996            | 1                    | Microalb          | 0.02              | 04/2006            |  |  |
|  |   |                   |  |  |                    |                      | Eye Exam          |                   | 05/2004            |  |  |

Figure 6.2: Example structure of EHR.

### 6.1.2 EHR process mining

As a preprocessing step, resources performing different activities were identified along with the different treatment execution paths known as variants. We also recorded information about the evidences and duration each resource took to handle it. Based on this information the resources were clustered as best, good and an average performer. The variants of treatment execution paths were identified using the technique of LCS. Variants are the frequently executed treatment paths. On discovering different possible variants, the NPA for the currently executing partial trace could be identified using the technique of trace matching proposed by Song et al. (2008). On discovered NPA, the set of resources capable of performing it were identified along with their availability and efficiency. Among the set of identified resource the best resource was chosen using the concept of *Yerkes-Dodson Law of Arousal*. This law was developed by psychologists Robert M. Yerkes and John Dillingham Dodson in 1908 Yerkes and Dodson (1908). It is also known as *Arousal Theory* and it states that by increasing arousal, the worker's performance can be improved. However, if the level of arousal increases too much, performance decreases, as shown in Figure 6.3. Using this information of NPA and resource

identified, we recommended the critical path of treatment. On conformance it was seen that applying the recommended path of execution, we could have avoided the thirteen cases on whom emergency intervention was needed.

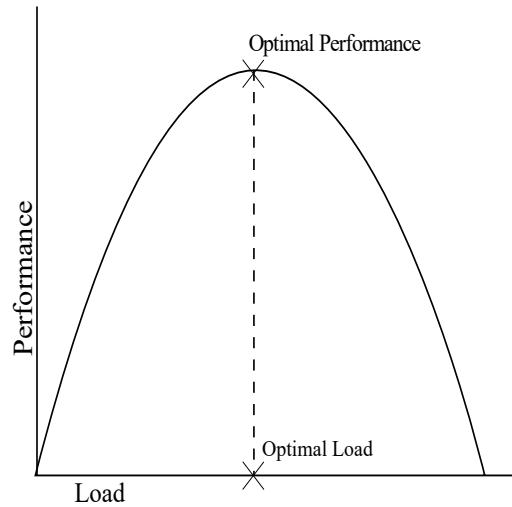


Figure 6.3: Illustration of Yerkes-Dodson Law of Arousal.

## 6.2 Preliminaries

The following notations and definitions will help in understanding process mining.

### Definition 6.1. Relation

A relation  $R$  is a subset ( $\subseteq$ ) of the Cartesian product of set  $A$  and  $B$ , and is denoted by  $aRb$ .

Let  $A$  be a set and  $R \subseteq A \times A$  be a relation, then,

- $R$  is *reflexive* if  $aRa$  for all  $a \in A$ .
- $R$  is *irreflexive* if  $\sim (aRa)$  for all  $a \in A$ .
- If  $aRb$  implies  $bRa$  for all  $a, b \in A$ , the relation is *symmetric*.
- If  $aRb$  and  $bRc$  implies  $aRc$  for all  $a, b, c \in A$ , the relation is *transitive*.
- Relation  $R$  is *antisymmetric* if  $aRb$  and  $bRa$  imply  $a = b$  for all  $a, b \in A$ .

### Definition 6.2. Functions

A function is a relation that uniquely associates members of one set with members of another set (denoted by  $f : A \rightarrow B$ ).

A function  $f$  from  $A$  to  $B$  is a relation such that for all  $a \in A$  is individually linked with an object  $f(a) \in B$ . This implies that a function can be many-to-one or one-to-one relation.

**Definition 6.3. Sequences**

Let  $S$  be a set. A sequence  $\sigma$  over  $S$  of length  $n \in \mathbb{N}$  is a function  $\sigma : \{1, \dots, n\} \rightarrow S$ .

Given a sequence  $s$  and set  $A$ ,

- Sequence  $s$  of length  $n$  are represented as  $\langle s_1, s_2, s_3, \dots, s_n \rangle$ .
- Length of sequence  $s$  is denoted by  $|s|$ .
- $i^{\text{th}}$  symbol of  $s$  is represented by  $s(i)$ .
- A subsequence of  $s$  that starts at position  $i$  and ends at position  $j$  of  $s$  is represented by  $s(i, j)$ .
- A *Head* prefix of length  $i$  of  $s$  represented by  $hd^i$ . The vector representation is  $s(1, i)$ .
- A *tail* suffix from the position  $i$  of  $s$  is represented by  $tl^i$ . The vector representation is  $s(i : |s|)$ .
- A new sequence  $p$  can be obtained by concatenating two sequences  $s = \langle s_1, s_2, \dots, s_m \rangle$  and  $t = \langle t_1, t_2, \dots, t_n \rangle$ . After concatenation the resulting sequence  $p$  can be denoted by  $s \diamond t$ , and it will be of length  $m + n$ .
- $A^*$  represents the set of all finite sequences over  $A$ .
- Set of all sequences of length  $n$  over set  $A$  is denoted by  $A^n$  (is  $\subseteq A^*$ ).

**Definition 6.4. Tuple**

Tuple (list) is a collection of ordered elements. Let  $A$  be a set and let  $t = \langle a_1, a_2, \dots, a_n \rangle \in A \times \dots \times A$  be a tuple of  $n$  elements (generally called as  $n$  tuple).  $t(i)$  refers to  $i^{\text{th}}$  element of tuple  $t$ . For example, let  $\langle a, b \rangle \in A \times A$  be a tuple of 2 elements  $t_1(\langle a, b \rangle) = a$  and  $t_2(\langle a, b \rangle) = b$ .

**Definition 6.5. Event and attribute** Let  $\mathcal{E}$  be the set of all event identifiers and  $AN$  be the set of all attributes.

- Let  $\mathcal{X}_x$  be a universal set (set of all possible values of  $x$ ), where  $x \in AN$ , and let  $e \in \mathcal{E}$ .
- $\#_x : \mathcal{E} \rightarrow \mathcal{X}_x \cup \{\perp\}$  is
  - the value for all attributes  $x$  not defined in  $e$ ,
  - $\#_x(e) = \perp$  denotes value of attribute  $x$  for any event  $e$ .

**Definition 6.6. Event log** A simple event log  $\mathcal{L}$  is a multi-set of traces over  $\mathcal{E}$ , i.e.,  $\mathcal{L} \in \mathbb{B}(\mathcal{A}^*)$ . It basically contains following fields

- **Case id** is used for distinctly identifying a particular instance of the process. For example, xx12, xx13, etc. represent the case ids in the hospital admission event log given in Table 6.1.
- **Event id** assigns a distinct identifier for every event related to a specific case. For example, event id for the case xx12 is 2346 and for the case xx13 is 3347.
- **Activity** assigns a readable name for every event of a case. For example, event 2346 of case xx12 and event of 2360 of case xx14 points to an activity named **Discharge** (D).
- **Resources** identify the individuals who are assigned and responsible for executing a specific activity. For example, Pete is assigned as a resource for executing the activity **Discharge** related to all the cases.
- **Timestamps** record the duration between the start and end of a particular activity.
- **Cost** is the expenditure incurred while executing a specific activity.

All cases in the event log  $\mathcal{L}$  can be converted into sequences of activity names using the classifier,  $\#_{activity}(e)$ . Applying this classifier to the cases shown in Figure 6.2, we get the simple event log:  $\mathcal{L}$  :

- $\mathcal{L}_1 = AR_{Pete}^{(0,3)}, IP_{Sean}^{(2,7)}, CH_{Sue}^{(5,15)}, DM_{Sara}^{(11,20)}, D_{Pete}^{(20,25)}$
- $\mathcal{L}_2 = AR_{Pete}^{(0,5)}, IP_{Sue}^{(3,11)}, CH_{Sean}^{(7,15)}, DM_{Sara}^{(12,18)}, BT_{Sean}^{(19,15)}$
- $\mathcal{L}_3 = AR_{Pete}^{(0,7)}, OP_{Sean}^{(2,12)}, CH_{Sue}^{(7,15)}, DM_{Sara}^{(11,20)}, RE_{Sue}^{(15,15)}, IP_{Sean}^{(21,20)}, CH_{Sue}^{(26,15)}, DM_{Sara}^{(30,22)}, D_{Pete}^{(108,93)}$
- $\mathcal{L}_4 = AR_{Pete}^{(0,5)}, OP_{Sue}^{(3,6)}, CH_{Sean}^{(6,11)}, DM_{Sara}^{(9,15)}, BT_{Sue}^{(15,30)}, DM_{Sara}^{(25,20)}, D_{Pete}^{(30,15)}$
- $\mathcal{L}_5 = AR_{Pete}^{(0,3)}, IP_{Sue}^{(2,9)}, CH_{Sean}^{(6,15)}, DM_{Sara}^{(10,15)}, RE_{Sue}^{(16,23)}, IP_{Sean}^{(21,19)}, CH_{Sue}^{(29,15)}, DM_{Sara}^{(38,15)}, BT_{Sean}^{(42,20)}, DM_{Sara}^{(50,15)}, D_{Pete}^{(52,19)}$

On classifying based on the resources using the classifier  $\#_{resource}(e)$  we get the sample event log as,

$\mathcal{L} = [\langle \text{Pete, Sean, Sue, Sara, Pete} \rangle, \langle \text{Pete, Sue, Sean, Sara, Sean} \rangle, \langle \text{Pete, Sean, Sue, Sara, Sue, Sean, Sue, Sara, Pete} \rangle, \langle \text{Pete, Sue, Sean, Sara, Sue, Sara, Pete} \rangle, \langle \text{Pete, Sue, Sean, Sara, Sue, Sean, Sue, Sara, Sean, Sara, Pete} \rangle, \dots]$

Table 6.1: Event log of hospital treatment process.

| Case id     | Event id         | Properties       |              |          |      |     |
|-------------|------------------|------------------|--------------|----------|------|-----|
|             |                  | Timestamp        | Activity     | Resource | Cost | ... |
| <b>xx12</b> | 2342             | 10-10-2012:01.00 | AR ( $P_1$ ) | Pete     | 130  | ... |
|             | 2343             | 10-10-2012:01.02 | IP ( $P_2$ ) | Sean     | 230  | ... |
|             | 2344             | 10-10-2012:01.05 | CH ( $P_4$ ) | Sue      | 340  | ... |
|             | 2345             | 10-10-2012:01.11 | DM ( $P_5$ ) | Sara     | 280  | ... |
|             | 2346             | 10-10-2012:01.20 | D ( $P_8$ )  | Pete     | 170  | ... |
| <b>xx13</b> | 2347             | 15-10-2012:01.00 | AR ( $P_1$ ) | Pete     | 200  | ... |
|             | 2348             | 15-10-2012:01.03 | IP ( $P_2$ ) | Sue      | 100  | ... |
|             | 2349             | 15-10-2012:01.07 | CH ( $P_4$ ) | Sean     | 400  | ... |
|             | 2350             | 15-10-2012:01.12 | DM ( $P_5$ ) | Sara     | 50   | ... |
|             | 2351             | 15-10-2012:01.19 | BT ( $P_7$ ) | sean     | 10   | ... |
| <b>xx14</b> | 2352             | 17-10-2012:01.00 | AR ( $P_1$ ) | Pete     | 200  | ... |
|             | 2353             | 17-10-2012:01.02 | OP ( $P_3$ ) | Sean     | 100  | ... |
|             | 2354             | 17-10-2012:01.07 | CH ( $P_4$ ) | Sue      | 400  | ... |
|             | 2355             | 17-10-2012:01.11 | DM ( $P_5$ ) | Sara     | 50   | ... |
|             | 2356             | 17-10-2012:01.15 | RE ( $P_6$ ) | Sue      | 10   | ... |
|             | 2357             | 17-10-2012:01.21 | IP ( $P_2$ ) | Sean     | 10   | ... |
|             | 2358             | 17-10-2012:01.26 | CH ( $P_4$ ) | Sue      | 400  | ... |
|             | 2359             | 17-10-2012:01.30 | DM ( $P_5$ ) | Sara     | 50   | ... |
| 2360        | 17-10-2012:01.33 | D ( $P_8$ )      | Pete         | 10       | ...  |     |
| <b>xx15</b> | 2361             | 19-10-2012:01.00 | AR ( $P_1$ ) | Pete     | 200  | ... |
|             | 2362             | 19-10-2012:01.03 | OP ( $P_3$ ) | Sue      | 100  | ... |
|             | 2363             | 19-10-2012:01.06 | CH ( $P_4$ ) | Sean     | 400  | ... |
|             | 2364             | 19-10-2012:01.09 | DM ( $P_5$ ) | Sara     | 50   | ... |
|             | 2365             | 19-10-2012:01.15 | BT ( $P_7$ ) | Sue      | 10   | ... |
|             | 2366             | 19-10-2012:01.25 | DM ( $P_5$ ) | Sean     | 10   | ... |
|             | 2367             | 19-10-2012:01.30 | D ( $P_8$ )  | Pete     | 10   | ... |
| <b>xx16</b> | 2368             | 21-10-2012:01.00 | AR ( $P_1$ ) | Pete     | 200  | ... |
|             | 2369             | 21-10-2012:01.02 | IP ( $P_2$ ) | Sue      | 100  | ... |
|             | 2370             | 21-10-2012:01.06 | CH ( $P_4$ ) | Sean     | 400  | ... |
|             | 2371             | 21-10-2012:01.10 | DM ( $P_5$ ) | Sara     | 50   | ... |
|             | 2372             | 21-10-2012:01.16 | RE ( $P_6$ ) | Sue      | 10   | ... |
|             | 2373             | 21-10-2012:01.21 | IP ( $P_2$ ) | Sean     | 10   | ... |
|             | 2374             | 21-10-2012:01.29 | CH ( $P_4$ ) | Sue      | 10   | ... |
|             | 2375             | 21-10-2012:01.38 | DM ( $P_5$ ) | Sara     | 50   | ... |
|             | 2376             | 21-10-2012:01.42 | BT ( $P_7$ ) | Sean     | 10   | ... |
|             | 2377             | 21-10-2012:01.50 | DM ( $P_5$ ) | Sara     | 10   | ... |
|             | 2378             | 21-10-2012:01.52 | D ( $P_8$ )  | Pete     | 10   | ... |
| ...         | ...              | ...              | ...          | ...      | ...  | ... |



### 6.3 Construction of Transition System

The event based transition system was constructed and is shown in Figure 6.4. This transition system assist in predicting the possible future state of the partial incomplete trace. The transition phase of any process has a *previous state*, *current state* and *future state*. The trace  $\langle A, B, C, D, E, F, G, H \rangle$  shown in Figure 6.4 is the sequence of activities that are already executed and known as partial trace ( $\sigma$ )(previous state) with  $\langle H \rangle$  being the current state of execution and  $\langle I, J, K, L \rangle$  being the possible future states.

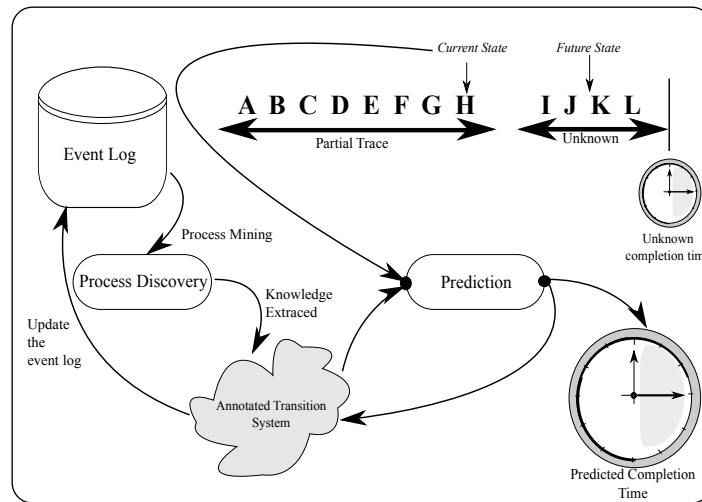


Figure 6.4: Illustration of design approach for predicting the future behaviour.

Consider an illustration of a process model shown in Figure 6.5, where  $\langle A, B, C \rangle$  are the partially executed trace with  $\langle C \rangle$  being the current state activity and  $\langle D, E, F \rangle$  being possible future state activities. The annotated transition system helps in predicting the future state i.e., whether  $\langle D \rangle$  or  $\langle E \rangle$  or  $\langle F \rangle$  to be followed after the current state activity. Annotated transition system having knowledge about the discovered process model identifies the resources capable for performing the future state activities.

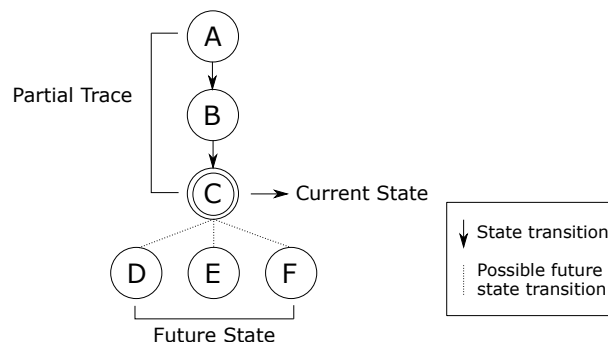


Figure 6.5: Illustration of current state to future state transition.

Annotated transition system with the help of activity, transition and causal metric, predicts the performance of each activity and resources before recommending the future state activity and resource.

- Activity metric helps in identifying the throughput/ processing time along with the waiting time at each activity. Using this information an activity with lesser processing and waiting time could be recommended as possible future state activity.
- Transition metric measures the performance of each activity and resource at different position based on their previous execution. This helps in finding the best position for each activity along with the resource for its execution.
- Causal metric assist in identifying the likelihood of occurrence of an event. This is measured by causal relationship between the preceding and succeeding pair of activity in the trace. This information helps in identifying the best position for each activity based on its preceding activity.

Further, the annotated transition system finds the cost of performing the activity using Time Driven Activity Based Costing (TDABC). Thus the system predicts the best future state which is measured in terms of time and cost incurred for the execution of that activity.

### 6.3.1 Initial design

Sample traces extracted from the process model shown in Figure 6.1 is shown in Table 6.1. The event log shown in the Table 6.1 will be used for the discussion and illustration in this chapter. To better understand the structure of event log, it is represented using a tree diagram and is shown in Figure 6.6. The activities in the process model shown in the Figure 6.1 is listed in Table 6.2

### 6.3.2 Current state

Here, process models discovered using process mining techniques are analysed for identifying their executable strength and weakness. We developed a position matrix for each activity along with its performance. Example: *suppose*, an activity is positioned at {2}, we identify its predecessor and successor, along with the duration it took for the completion of an assigned task, including the details about the resource performance is also

Table 6.2: Activities in the process model shown in the Figure 6.1

| Hospital Service |                           | Patients Response |                                   |
|------------------|---------------------------|-------------------|-----------------------------------|
| AR               | Appointment/ Registration | RAR               | Request Appointment/ Registration |
| IP               | In Patient                | NP                | New Patient                       |
| OP               | Out Patient               | RP                | Registered Patient                |
| CH               | Check History             | Ex                | Examine                           |
| DM               | Decision Making           | TP                | Treatment Procedure               |
| RE               | Re-Examine                | FU                | Follow Up                         |
| BT               | Begin Treatment           | RT                | Request Treatment                 |
| D                | Discharge                 | RD                | Request Discharge                 |
| CL               | Close                     |                   |                                   |

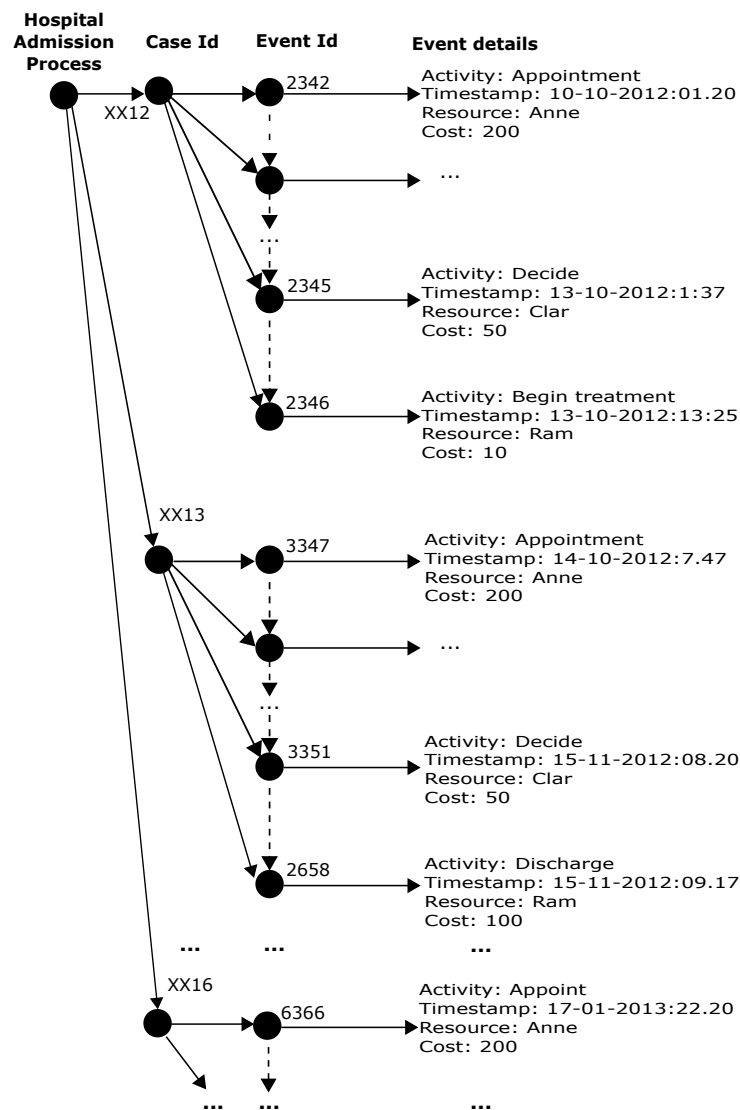


Figure 6.6: Tree structure of process log.

observed. Using this information, the NPA and an efficient resource for performing that NPA is identified for the current state.

### 6.3.3 Abstraction of event log for generating transition system

The transition system is a triplet of *State space* ( $S$ ), *Event labels* ( $E$ ) and *Transition relation* ( $T$ ): (S,E,T) (Mans, 2011). *State space* is the set of all possible states of the processes, in the discovered process model. *Event labels* are the labels defined for each event. *Transition relation* describes the transition of process from one state to another ( $T \subset S \times E \times S$ ). Consider:  $S_1$  and  $S_2$  as two states, then transition state is defined as  $S_1 \xrightarrow{e} S_2$ , where  $e$  is the event label.

**Definition 6.7. State representation** ( $l^{state}$ ) Let  $\mathcal{C}$  be a set of all the traces in a event log,  $\mathcal{R}$  be set of all the states then  $l^{state}$  is formally represented as  $l^{state} \in \mathcal{C} \rightarrow \mathcal{R}$

Example: Consider the partial trace  $\sigma = A, B, C..H$  in the Figure 6.4. Then  $l^{state}$  could be either  $l^{state} = H$  or  $l^{state} = Resource\ performing\ H$ . But, we considered complete trace execution as abstraction of  $l^{state}$ , i.e.,  $l^{state} = \sigma$ .

**Definition 6.8. Event representation:** ( $l^{event}$ )

Let  $\mathcal{E}$  be set of all events and  $\mathcal{R}$  be event representation using event labeling E, then  $l^{event}$  is formally represented as  $l^{event} \in \mathcal{E} \rightarrow \mathcal{R}$

With the knowledge of  $l^{state}$  and  $l^{event}$  we can now define transition system as:

**Definition 6.9. Transition System:** ( $l^{transition}$ )

Let,  $hd^k$  be first  $k$  elements in the sequence, where  $hd$  is head, and similarly  $tl^k$  be the tail of last  $k$  elements in the sequence, where  $k = 0 \leq k \leq |\sigma|$ . By the definition of the transition system (S,E,T).

$$\begin{aligned} S &= \left\{ l^{state} \left( hd^k(\sigma) \right) \mid \sigma \in Event\ Log(\mathcal{L}) \wedge 0 \leq k \leq |\sigma| \right\} \\ E &= \left\{ l^{event} \left( \sigma(k) \right) \mid \sigma \in Event\ Log(\mathcal{L}) \wedge 0 \leq k \leq |\sigma| \right\} \\ T &\subseteq S \times E \times S = \left\{ l^{state} \left( hd^k(\sigma) \right), l^{event} \left( \sigma(k+1) \right), l^{state} \left( hd^{k+1}(\sigma) \right) \right\} \end{aligned}$$

For identifying the state of transition, it is important to have proper information about current state transition system. For that, we need to have the performance information of each activity and resource. This would help in predicting the future state along with its completion time.

### 6.3.4 Performance information

The performance is measured by the help of process metrics (activity and transition metric), variant metrics and resource metrics. The *activity metric* provides the information of arrival and processing time of each activity. Using this, the throughput and waiting time at each activity is measured and helps in analysing each activity. The *transition metric* gives the possible state of transition for each partial trace. The *variant metric* clusters and finds different variants of process execution along with their performance. The *resource metric* measures the performance of each resource for a different set of activity they had performed.

### 6.3.5 Activity metric

The life cycle of an activity has an arrival time, waiting time and a processing time. It is important to trigger an activity when it is needed instead of making them wait. Delay in an execution of activity would delay the process execution and completion. The goal of this work is to bring down the waiting time and make the process more efficient and available.

#### Definition 6.10. Life cycle of an activity

- *Arrival time*: Time an activity is triggered/ arrived.
- *Waiting time*: It is the time an activity wait for its execution after being arrived.
- *Processing time*: Total throughput of an activity. It is the time from actual start to its completion.
- *Start time*: It is the time when the process begins its execution.

Consider the time line illustration shown in Figure 6.7 for the trace  $\mathcal{L}_1$  with case id xx12 in Table 6.1 . The Figure 6.7 clearly illustrate the turn-around time, waiting time and processing time. It can be seen from the Figure 6.7 that activity  $P_2$  arrived at time 2 and had to wait for 1 unit time for the completion of processing of activity  $P_1$ . The arrival and processing time for  $\mathcal{L}_1$  is shown in Table 6.3 and the subsequent Gantt chart is shown in Figure 6.8.

Using the formula shown in equation 6.2 and 6.3 the arrival metric is constructed for the trace  $\mathcal{L}_1$  and is shown in Table 6.4.

Table 6.3: *Initials*: Arrival and processing time of  $\mathcal{L}_1$

| Activity     | Arrival Time | Processing Time |
|--------------|--------------|-----------------|
| AR ( $P_1$ ) | 0            | 3               |
| IP ( $P_2$ ) | 2            | 7               |
| CH ( $P_4$ ) | 5            | 15              |
| DM ( $P_5$ ) | 11           | 20              |
| D ( $P_8$ )  | 20           | 25              |

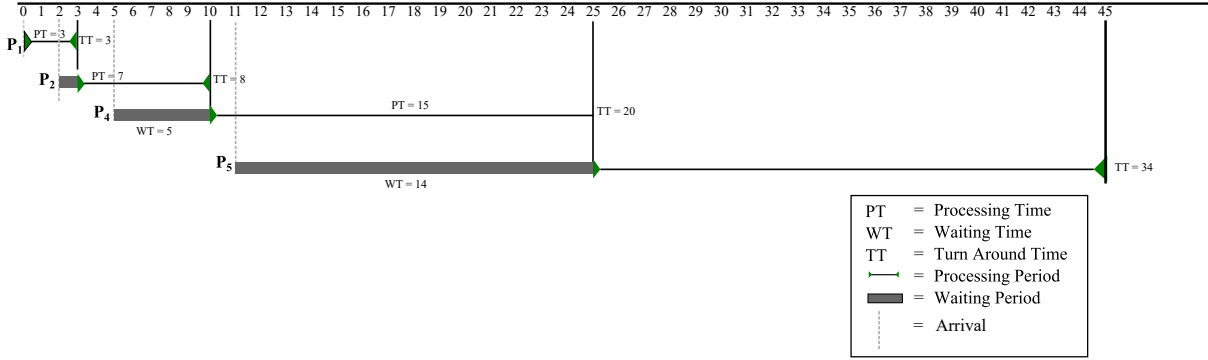


Figure 6.7: Illustration of different process life time.

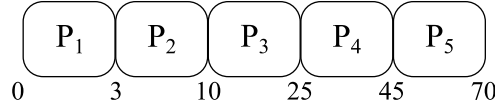


Figure 6.8: Gantt chart for the trace shown in Table 6.3.

$$\mathcal{T}_{Turn\ around}^{P_i} = \left( \mathcal{T}_{Processing}^{P_{i-1}} + \mathcal{T}_{Processing}^{P_i} \right) - \mathcal{T}_{Arrival}^{P_i} \quad (6.1)$$

$$\mathcal{T}_{Waiting}^{P_i} = \mathcal{T}_{Turn\ around}^{P_i} - \mathcal{T}_{Processing}^{P_i} \quad (6.2)$$

$$(6.3)$$

Suppose, if an allotted time for the completion of a process execution is 20, then the annotation is added with 20 unit time. For illustration, let us only consider the arrival time for each activity as  $\langle P_1^0, P_2^2, P_3^5, P_4^{11}, P_5^{20} \rangle$ . We add  $\langle P_1^0 \rangle$  with the time remaining in annotation being 20. On addition of  $\langle P_1^0, P_2^2 \rangle$ , the remaining time at annotation is  $20 - 2 = 18$ . Similarly, on adding  $\langle P_1^0, P_2^2, P_3^5 \rangle$ , the remaining time would become 13, on adding  $P_4^{11}$  the remaining time = 2, thus making no room for adding  $P_5$ . This situation arises only due to the waiting time at each activity. Hence it is important to reduce the

Table 6.4: Arrival Metric of  $\mathcal{L}_1$ 

| $\mathcal{T}$ | Process Completed | $\mathcal{T}_{Turn\ around}^{P_i}$ | $\mathcal{T}_{Waiting}^{P_i}$ |
|---------------|-------------------|------------------------------------|-------------------------------|
| 0             | –                 | –                                  | –                             |
| 03            | $P_1$             | 03 - 00 = 03                       | 03 - 03 = 00                  |
| 10            | $P_2$             | 10 - 02 = 08                       | 08 - 07 = 01                  |
| 25            | $P_4$             | 25 - 05 = 20                       | 20 - 15 = 05                  |
| 45            | $P_5$             | 45 - 25 = 34                       | 34 - 20 = 14                  |
| 70            | $P_8$             | 70 - 20 = 50                       | 50 - 25 = 25                  |

$\mathcal{T}_{Waiting}$ . For that we aggregated the  $\bar{\mathcal{T}}_{Waiting}$  of all the traces, along with preceding and succeeding activity and is shown in Table 6.5. This mean, activity  $\langle IP \rangle$  is preceded by  $AR$  and succeeded by  $CH$  with the waiting time period of 1. Aggregating the waiting time of all the succeeding activity along with its preceding activity and by constructing annotation system, the NPA for a partial trace  $\sigma$  could be decided.

Table 6.5: *Initials*: Succeeding and preceding activities  $\mathcal{L}_1$ 

| Activity     | $\mathcal{T}_{Waiting}$ |
|--------------|-------------------------|
| AR ( $P_1$ ) | 0                       |
| IP ( $P_2$ ) | 1                       |
| CH ( $P_4$ ) | 5                       |
| DM ( $P_5$ ) | 14                      |
| D ( $P_8$ )  | 25                      |

The Figure 6.9 shows behaviour of process execution with the error bar showing the waiting time at each activity. Each horizontal line in the plot represents traces 'xx12, xx13, xx14, xx15, xx16" in the Table 6.1 and vertical line in the trace is the waiting time for the execution of each activity in the trace. Trace length in the Figure 6.9 is measured by total throughput of process execution in the trace instance. Total throughput is the sum of *arrival, waiting and processing time*, and annotated transition system is intended to identify the trace with higher waiting time. By decreasing waiting time, throughput of process execution could be made faster. The error bar chart is plotted for the instance of traces shown in Table 6.1 with the arrival time and processing time shown in simple event log  $\mathcal{L}$  in the definition 6.6. On analysing the Figure 6.9 we observe that the trace

xx14 had highest throughput time when compared to other traces. This analysis assist in identifying drifted traces in terms of time taken and provide an alternative path of execution.

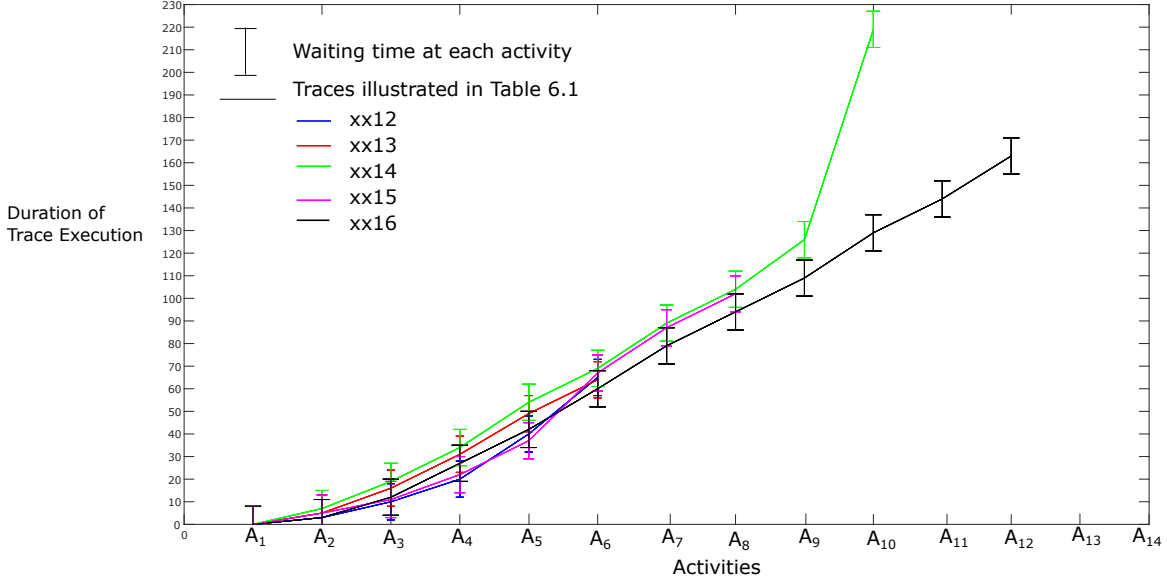


Figure 6.9: Analysis of process behaviour.

### 6.3.6 Transition metric

The transition metric function  $l^{transition}$  is used to measure the performance of an activity and resource at different position of its execution. Using this, for a partially executed trace  $\sigma_1$ , the NPA  $\sigma_2$  could be identified i.e.,  $l^{transition}(\sigma_1, \sigma_2)$ . Along with time function  $\mathcal{T}$  for each activity, the possible execution position  $\mathcal{P} \in P_1, P_2, P_3, \dots, P_n$  were also recorded. To identify  $\sigma_2$ , we first need to know the currently executing state  $l_{Current\ State}^{transition}$  along with information of its  $\mathcal{P} = l_{Current\ state}^{P_i}$ . Along with this information  $\overline{\mathcal{T}_{Waiting}}$  and  $\overline{\mathcal{T}_{Turn\ around}}$  for each traces were calculated. Based on the outcome of this, the best possible position of each activity was measured and is shown in equation 6.4.

$$l^{transition}(\sigma_1, \sigma_2) = \begin{cases} 0 & \text{if } \sigma_2 = \langle \ \rangle, \\ Max_{\mathcal{T}(\sigma_2)} - Min_{\mathcal{T}(\sigma_2)} & \text{if } \sigma_1 = \langle \ \rangle \text{ and } \sigma_2 \neq \langle \ \rangle \\ Max_{\mathcal{T}(\sigma_2)} - Max_{\mathcal{T}(\sigma_1)} & \text{if } \sigma_1 \neq \langle \ \rangle \text{ and } \sigma_2 \neq \langle \ \rangle \end{cases} \quad (6.4)$$

$$\text{where } Max_{\mathcal{T}(\sigma)} = max\{\overline{l_{Current\ State}^{transition}} | l_{Current\ State}^{transition} \in \sigma\}$$



Suppose the trace  $\mathcal{L} = \langle P^0, P^2, P^5, P^{11}, P^{20} \rangle$  is split into  $\sigma_1 = \langle P^0, P^2, P^5 \rangle$  and  $\sigma_2 = \langle P^{11}, P^{20} \rangle$ , then  $l^{transition}(\sigma_1, \sigma_2) = Max_{\mathcal{T}}(\langle P^0, P^2, P^5 \rangle) - Max_{\mathcal{T}}(\langle P^{11}, P^{20} \rangle)$  i.e.,  $20 - 5 = 15$ . Here, 15 is the remaining time in annotation from the completion of process execution, but the elapsed time could be calculated using equation 6.5.

$$Max_{\mathcal{T}(\sigma_1)} - Min_{\mathcal{T}(\sigma_1)} \quad if \quad \sigma_1 \neq \langle \quad \rangle \quad (6.5)$$

The equation 6.4 is modified to find the position of each activity and is shown in 6.6.

$$l_{\mathcal{P}_i}^{transition}(\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_n) = \begin{cases} 0 & if \quad \mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_n = \langle \quad \rangle, \\ Max_{\mathcal{T}(\sigma_1)} - Min_{\mathcal{T}(\sigma_1)} & if \quad \mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_n \neq \langle \quad \rangle \end{cases} \quad (6.6)$$

On identifying the possible set of activities at each position, the set of preceding and succeeding activities for the current state activity is discovered. This will help in predicting the future state. Preceding is required to match the currently executing partial trace  $\sigma$ . This is achieved by the help of *causal metric*.

### 6.3.7 Causal metric

In a process model, any two nodes (activities) are in causal relationship if there is a sequential order of execution between them. Suppose,  $A$  and  $B$  are two activities such that  $A \rightarrow B$ , then there exist causal relationship between  $A$  and  $B$ , where the symbol  $\rightarrow$  is the symbol of sequential execution. This means, in a trace any event get executed due to the execution of its preceding event and the reason of this occurrence is recorded in causal metric. Statistically, it gives the information about the likelihood of occurrence of an event based on the information about the preceding event. Let,  $X$  be GSD where  $\{x_1, x_2, \dots, x_n\} \in X$  are different stages in GSD and  $\{y_1, y_2, \dots, y_m\} \in Y$  be the factors associated with GSD. Then probability distribution function, we could find  $P(Y = y | Causing(X = x))$ , where  $y$  is the factor causing disease to reach stage  $x$ . Using this concept of causal relationship we developed the metric for *turn around time* and *waiting time* and is shown in Table 6.6 and 6.7 respectively. *Turn around time* and *Waiting time* is calculated using the equation 6.2 and 6.3 respectively and illustrated in Table 6.4. For illustration about the values in table 6.6, let us consider event log  $\mathcal{L}$  in

the definition 6.6. For finding the causal relationship between  $P_1$  and  $P_2$ .

- $\mathcal{L}_1 = (P_1)_{Pete}^{(0,3)}, (P_2)_{Sean}^{(2,7)}$
- $\mathcal{L}_2 = (P_1)_{Pete}^{(0,5)}, (P_2)_{Sue}^{(3,11)}$
- $\mathcal{L}_3 = (P_1)_{Pete}^{(0,7)}, (P_3)_{Sean}^{(2,12)}, (P_4)_{Sue}^{(7,15)}, (P_5)_{Sara}^{(11,20)}, (P_6)_{Sue}^{(15,15)}, (P_2)_{Sean}^{(21,20)}$
- $\mathcal{L}_4 =$  No causal relationship between  $P_1$  and  $P_2$
- $\mathcal{L}_5 = (P_1)_{Pete}^{(0,3)}, (P_2)_{Sue}^{(2,9)}, (P_4)_{Sean}^{(6,15)}, (P_5)_{Sara}^{(10,15)}, (P_6)_{Sue}^{(16,23)}, (P_2)_{Sean}^{(21,19)}$

Using the equation 6.2 =  $\mathcal{T}_{Turn\ around}^{P_i} = \left( \mathcal{T}_{Processing}^{P_{i-1}} + \mathcal{T}_{Processing}^{P_i} \right) - \mathcal{T}_{Arrival}^{P_i}$

- $\mathcal{L}_1 = (3 + 7) - 2 = 8$
- $\mathcal{L}_2 = (5 + 11) - 3 = 13$
- $\mathcal{L}_3 = (7 + 12 + 15 + 20 + 15 + 20) - 21 = 68$
- $\mathcal{L}_4 =$  No causal relationship between  $P_1$  and  $P_2$
- $\mathcal{L}_5 = (3 + 9) - 2 = 10$  and  $(3 + 9 + 15 + 15 + 23 + 19) - 21 = 63$ . Now taking the average of these =  $\frac{10+63}{2} = 36.5$

Now the  $\mathcal{T}_{Turn\ around}^{P_i} = \frac{8+13+68+36.5}{4} = 31.375$ . This is entered in the causal metric shown in the Table 6.6. Likewise all other values are entered along with for the Table 6.7.

Using the information of causal relation, the position metric showing the execution position of different activities is plotted and is shown in Figure 6.10. Figure 6.11, shows the successor and predecessor, along with the waiting and processing time at each position. This information helps in identifying the possible future state of currently executing trace  $\sigma$ . Before recommending the NPA to succeed the partial trace, we need to find the cost incurred for the delivery of the service. For the technique of Time Driven Activity Based Costing (TDABC) proposed by Kaplan and Anderson (2003) is used, for measuring the cost of activity and the resources.

Table 6.6: Causal relationship for all the activities based on turn around time for the event log  $\mathcal{L}$  shown in Table 6.1

|       | $P_1$ | $P_2$ | $P_3$ | $P_4$ | $P_5$ | $P_6$ | $P_7$ | $P_8$ |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| $P_1$ | 0     | 31.37 | 12.5  | 31.6  | 50.9  | 51.5  | 63    | 86.5  |
| $P_2$ | 0     | 0     | 0     | 41.87 | 59    | 49    | 68.5  | 91.33 |
| $P_3$ | 0     | 68    | 0     | 41    | 57.25 | 54    | 52    | 90    |
| $P_4$ | 0     | 65.5  | 0     | 0     | 50.9  | 51.5  | 63    | 86.5  |
| $P_5$ | 0     | 65.5  | 0     | 74    | 0     | 51.5  | 63    | 86.5  |
| $P_6$ | 0     | 65.5  | 0     | 74    | 91.75 | 0     | 92    | 112   |
| $P_7$ | 0     | 0     | 0     | 0     | 80.5  | 0     | 0     | 94    |
| $P_8$ | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

Table 6.7: Causal relationship for all the activities based on waiting time for the event log  $\mathcal{L}$  shown in Table 6.1

|       | $P_1$ | $P_2$ | $P_3$ | $P_4$ | $P_5$ | $P_6$ | $P_7$ | $P_8$ |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| $P_1$ | 0     | 18.37 | 3.5   | 18.4  | 32.6  | 32.5  | 41.33 | 68    |
| $P_2$ | 0     | 0     | 0     | 26.87 | 53.25 | 26    | 51    | 71.66 |
| $P_3$ | 0     | 48    | 0     | 8.5   | 38    | 39    | 22    | 75    |
| $P_4$ | 0     | 46    | 0     | 0     | 32.6  | 32.5  | 41.3  | 68    |
| $P_5$ | 0     | 46    | 0     | 59    | 0     | 32.5  | 41.3  | 68    |
| $P_6$ | 0     | 46    | 0     | 59    | 73.25 | 0     | 72    | 95    |
| $P_7$ | 0     | 0     | 0     | 0     | 63    | 0     | 0     | 77    |
| $P_8$ | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

### 6.3.8 Construction of annotated transition system for analysing the resource performance based on TDABC

TDABC starts by identifying the cost of all the activities contributing to the completion of a process. The cost of the activity is the amount spent on it for its completion, and it is the cost of the resources, assisting in performing that activity. For example, let's consider two traces:  $\langle A \rightarrow B \rightarrow C \rightarrow D \rangle$  and  $\langle A \rightarrow B \rightarrow E \rightarrow D \rangle$ , where  $\langle B \rangle$  is followed by  $\langle C \text{ and } E \rangle$ .  $\langle C \rangle$  took 10 unit time with resource cost of 200 for conducting it, where in  $\langle E \rangle$  took 5 unit time with resource cost of 400. Recommending NPA for  $\langle B \rangle$  with an adequate resource is a challenge and can be dealt with the technique of TDABC.

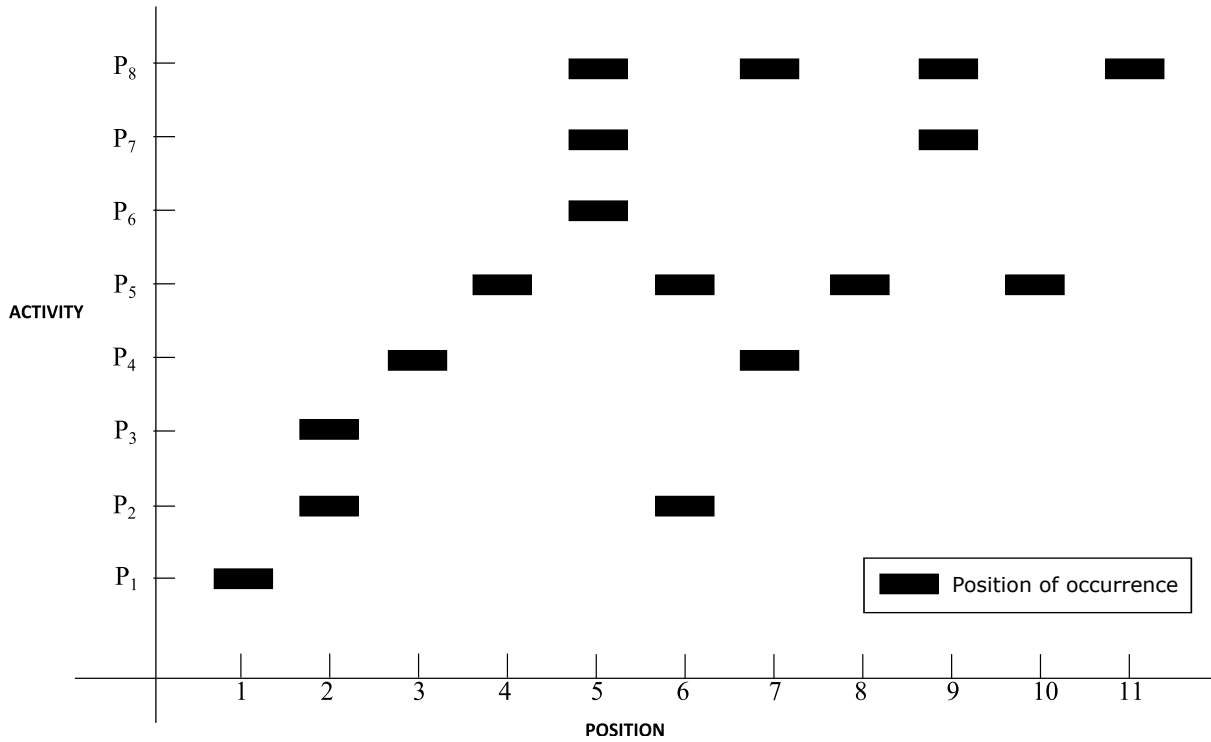


Figure 6.10: Position of different activity in the example log  $\mathcal{L}$ .

First, per unit cost is calculated by aggregating with activity and resource cost. Next, identify the priority of each activity. And finally, the resource cost per unit time of service delivered is measured, thus giving the cost/ activity. The complete process is listed below:

- List all the activities.
- Identify per unit cost of each resource based on service overhead.
- Aggregate the cost identified
- Prioritize each activity.
- Calculate the resource cost per unit of service.

### A Estimating cost per unit time

Here we first identify the load and capacity along with *Levels of arousal* of the performance using *Yerkes–Dodson law*. In the *level of arousal* the *Optimal Load* is the *maximum load* a resource can handle efficiently, along with its *performance*. Performance is a ratio of  $\frac{\text{Total time taken}}{\text{Load}}$ . The performance was analysed by increasing the load and observing the time taken. It was observed that, as the load was increased, there was a decrease in time taken for completing the assigned task. But at some point, there was a drift, and there was an increase in time taken for completion. That drifted point is known as Arousal

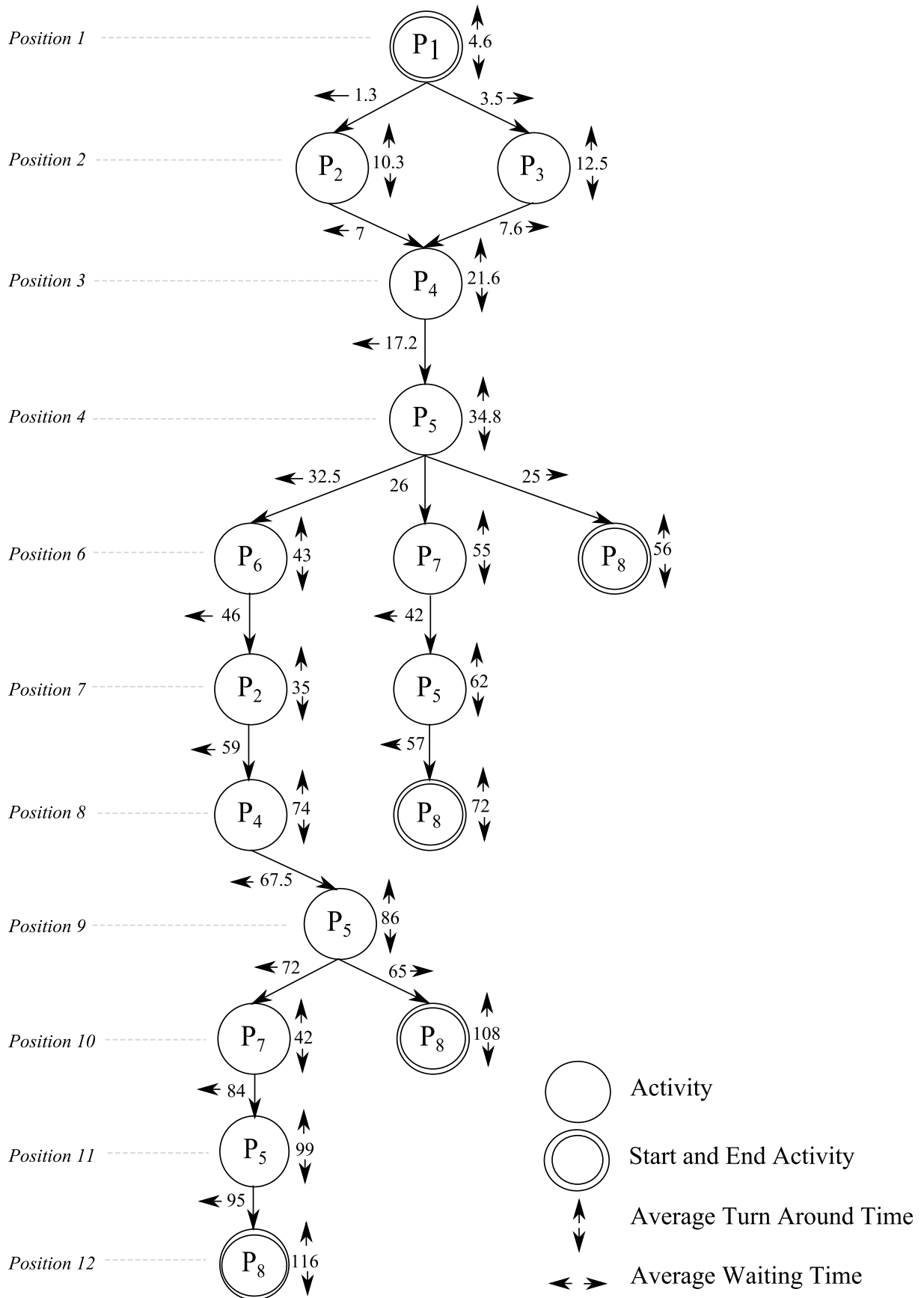


Figure 6.11: Process model with the information of processing time, waiting time, successor and predecessor for the example log  $\mathcal{L}$ .

(optimal load and performance of the resources). As a thumb rule, we assume the best capacity of any resource is between 80 – 85% of the estimated values. Suppose, if any resource’s Levels of arousal is 10 hours, we consider his optimal performance is about 8 hours per day and remaining to be his transit time.

## B Estimating duration of an activity

Knowing the cost per unit time, we now need to know the time each resource takes for completing one unit of activity. This is measured by the ratio of the total cost incurred by the resource performing the assigned activity, by total hours spend on it. Consider the traditional way of Activity Based Costing (ABC) for finding the cost per unit activity. The values extracted using tradition ABC technique, for the first case with case ID *xx12* (shown in Table 6.1), is shown in Table 6.8.

Table 6.8: Traditional activity based costing

| Activity                   | Resource | % of time spent | Assigned cost | Activity quantity | Cost driver rate  |
|----------------------------|----------|-----------------|---------------|-------------------|-------------------|
| AR                         | Pete     | 2.6%            | 130           | 5                 | $(130/5) = 26$    |
| IP                         | Sean     | 6.9%            | 230           | 4                 | $(230/4) = 57.5$  |
| CH                         | Sue      | 17.39%          | 340           | 7                 | $(340/7) = 48.57$ |
| DM                         | Sara     | 29.56%          | 280           | 9                 | $(280/9) = 31.11$ |
| D                          | Pete     | 43.47%          | 170           | 4                 | $(170/4) = 42.5$  |
| <b>Total Assigned Cost</b> |          |                 | <b>1150</b>   |                   |                   |

In the Table 6.8, *% of time spent* is identified by calculating the total-turn-around time of activity than finding its percentage in the total duration of the case. *Assigned cost* is the cost for running that activity. It is an expense incurred by the resource and is calculated for per unit time. *Activity quantity* is a number of time the activity is played in the entire process. *Cost driver rate* is calculated using the values of assigned cost and activity quantity.

*Let’s* consider the total allotted time be 1000 unit. The impact of practical capacity is now calculated and is tabulated in Table 6.9. Here the total time unit effectively used

is 576. Hence, it is 57.6% of the time which is allotted for the completion of the process, rest of 42.4% is wasted in transit time, and the resource is paid for that. Based on the % of time spent the effective time would be:  $AR=2.6\%$  (1000) which is 26, similarly it is 69, 173.9 295.6 and 434.7 percentage for  $IP$ ,  $CH$ ,  $DM$  and  $D$  respectively. This explains that, out of 26 time unit spent, only 15 (see Total Min. in Table 6.9) unit time was effectively used i.e., 57.69% was effectively used by the resource *Pete* for the activity  $AR$ . Similarly, 46.37%, 80.50%, 63.93% and 46% was used by resource *Sean*, *Sue*, *Sara* and *Pete*. Using the value of effective time usage and the assigned cost, the effective total cost can be calculated. The values measured is shown in Table 6.9.

Using the technique of TDABC, the effective time usage of each resource could be analysed and cost for completing the process could also be reduced. This method assist in analysing the behaviour of each resource before they are recommended for the completion of the partial trace  $\sigma$ .

Table 6.9: Impact of practical capacity

| Activity                   | Resource | Unit time in min. | Activity quantity | Total time in Min.  | Total Cost                |
|----------------------------|----------|-------------------|-------------------|---------------------|---------------------------|
| AR                         | Pete     | 3                 | 5                 | $3 \times 5 = 15$   | $(57.69\%(130)) = 75$     |
| IP                         | Sean     | 8                 | 4                 | $8 \times 4 = 32$   | $(46.37\%(230)) = 106.6$  |
| CH                         | Sue      | 20                | 7                 | $20 \times 7 = 140$ | $(80.5\%(340)) = 273.7$   |
| DM                         | Sara     | 21                | 9                 | $21 \times 9 = 189$ | $(63.93\%(280)) = 120.82$ |
| D                          | Pete     | 50                | 4                 | $50 \times 4 = 200$ | $(46\%(170)) = 78.20$     |
| <b>Total Min and Cost:</b> |          |                   |                   | <b>576</b>          | <b>654.32</b>             |

### 6.3.9 Construction of annotated transition system based on remaining turn-around time

Aim of constructing the transition system is to assist the prediction of future unknown state as shown in Figure 6.4. In any partial trace, it is important to understand already

executed trace and the time remaining for the completion of process. By the help of Mod-CNN, we can predict an average time a patient may become critical. This was achieved by observing the disease behaviour. Thus, by knowing the time elapsed by the partial trace, we can find the remaining time. Based on the remaining time, prediction of best possible future state is done. Along with this, we analysed the behaviour of each possible activity in the future state and resource performance as well, before recommending any possible path of execution. Consider the log file shown in the Table 6.10. The superscript values are turn-around time ( $\mathcal{T}_{Turn\ around}$ ) for each activity, measured by using the equation 6.2.

Table 6.10: Event log of Table 6.1 with the information of turn around time

| Case ID | Trace  |
|---------|--|
| xx12    | $\langle AR^3, IP^{11}, CH^{31}, DM^{65}, D^{115} \rangle$   |
| xx13    | $\langle AR^5, IP^{18}, CH^{42}, DM^{79}, BT^{124} \rangle$  |
| xx14    | $\langle AR^7, OP^{24}, CH^{51}, DM^{94}, RE^{148}, IP^{216}, CH^{294}, DM^{390}, D^{498} \rangle$                     |
| xx15    | $\langle AR^5, OP^{13}, CH^{29}, DM^{57}, BT^{109}, DM^{171}, D^{243} \rangle$   |
| xx16    | $\langle AR^3, IP^{13}, CH^{34}, DM^{66}, RE^{115}, IP^{178}, CH^{248}, DM^{324}, BT^{416}, DM^{515}, D^{631} \rangle$ |

The state transition metric function  $l^{transition}$  6.4 is used to identify the performance of an activity and the resources. They assist in identifying the NPA ( $\sigma_2$ ) for a partially executed trace  $\sigma_1$  i.e.,  $l^{transition}(\sigma_1, \sigma_2)$ . For constructing the annotated transition system, set of possible traces  $\{\sigma_2\}$  following the successor  $\sigma_1$  are identified. Thus using the definition of transition system 6.9 the annotated transition system could be defined as:

**Definition 6.11. (Annotated Transition System):** Annotated transition system is aimed in finding the future state activities for the partially executed trace and is shown in the equation 6.7.

$$A_T = \sum_{\sigma \in \mathcal{L}} \sum_{\substack{0 \leq k \leq |\sigma| \\ l^{state}(hd^k(\sigma))}} \left[ l^{transition} \left( hd^k(\sigma), tl^{|\sigma|-k}(\sigma) \right) \right] \quad (6.7)$$

The annotated transition system make use of transition metric function  $l^{transition}$  6.4 to measure the performance of an activity and resource at different position of its execution. Here  $hd^k$  (head) represents the prefix of length  $k$ ,  $tl^k$  (tail) is suffix from the position  $k$  and  $|\sigma|$  is the length of the partial trace  $\sigma$ .



Using this, for a partially executed trace  $\sigma_1$ , the NPA  $\sigma_2$  could be identified i.e.,  $l^{transition}(\sigma_1, \sigma_2)$ . Suppose the trace  $\mathcal{L} = \langle P^0, P^2, P^5, P^{11}, P^{20} \rangle$  is split into  $\sigma_1 = \langle P^0, P^2, P^5 \rangle$  and  $\sigma_2 = \langle P^{11}, P^{20} \rangle$ , then  $l^{transition}(\sigma_1, \sigma_2) = Max_{\mathcal{T}}(\langle P^0, P^2, P^5 \rangle) - Max_{\mathcal{T}}(\langle P^{11}, P^{20} \rangle)$  i.e.,  $20 - 5 = 15$ . Here, 15 is the remaining time in annotation from the completion of process execution, but the elapsed time could be calculated using equation 6.5. The annotated transition system ( $A_{\mathcal{T}}$ ) is constructed and is represented in Figure 6.12. The transition system depicts the process model shown in the Figure 6.1. Activity  $\langle \mathbf{AR} \rangle$  being the Starting activity in the event log shown in Table 6.10 the annotated transition system also starts with  $\langle \mathbf{AR} \rangle$ .

Now let us consider a trace instance  $\langle AR^3, IP^{11}, CH^{31}, DM^{65}, D^{115} \rangle$  shown in the first row of Table 6.10. In this instance,  $\langle AR \rangle$  has total turn-around time of 3 units. Similarly  $\langle IP \rangle$  took 11,  $\langle CH \rangle$  took 31,  $\langle DM \rangle$  took 65 and  $\langle D \rangle$  took 115 unit of time. The  $l^{transition}$  starts with empty successor  $l^{transition}(\sigma_1) = \langle \rangle$  and maps to state  $\emptyset$ . Thus remaining time is  $\mathcal{T}_{Remaining} = 115 - 3 = 112$  and is added to annotation state  $\emptyset$ . Now the successor is added to with  $l^{transition}(\sigma_1) = \langle AR^3 \rangle$  and is mapped to current state  $l^{transition}_{Current\ State}(\sigma_1) = \{AR\}$ . Now the remaining time is  $115 - 3 = 112$  and is added to the annotation of state  $\{AR\}$ . Successor  $\{IP\}$  is now added  $l^{transition}(\sigma_1) = \langle AR^3, IP^{11} \rangle$ . This is mapped to current state  $l^{transition}_{Current\ State}(\sigma_1) = \{AR, IP\}$ . Remaining time  $\mathcal{T}_{Remaining} = 115 - 11 = 104$  is added to annotation of state  $\{AR, IP\}$ . Activity  $\langle CH^{31} \rangle$  is added to its successor  $l^{transition}(\sigma_1) = \langle AR^3, IP^{11}, CH^{31} \rangle$  and is mapped to current state  $l^{transition}_{Current\ State}(\sigma_1) = \{AR, IP, CH\}$ . Remaining time  $\mathcal{T}_{Remaining} = 115 - 31 = 84$  is added to annotation of state  $\{AR, IP, CH\}$ . Activity  $\langle DM^{65} \rangle$  is added to its successor  $l^{transition}(\sigma_1) = \langle AR^3, IP^{11}, CH^{31}, DM^{65} \rangle$  and is mapped to current state  $l^{transition}_{Current\ State}(\sigma_1) = \{AR, IP, CH, DM\}$ . Remaining time  $\mathcal{T}_{Remaining} = 115 - 65 = 50$  is added to annotation of state  $\{AR, IP, CH, DM\}$ . Finally activity  $\langle D^{115} \rangle$  is added to its successor  $l^{transition}(\sigma_1) = \langle AR^3, IP^{11}, CH^{31}, DM^{65}, D^{115} \rangle$  and is mapped to current state  $l^{transition}_{Current\ State}(\sigma_1) = \{AR, IP, CH, DM, D\}$ . Remaining time  $\mathcal{T}_{Remaining} = 115 - 115 = 00$  is added to annotation of state  $\{AR, IP, CH, DM, D\}$ .

After this annotation is followed to all the process instance in the Table 6.10, we will have a state  $\emptyset$  annotated with bag containing five elements:  $\{112, 119, 491, 238, 628\}$ . State  $\{AR, IP\}$  is annotated with  $[(115 - 11 = 104), (124 - 18 = 106), (631 - 13 = 618)]$ . Similarly the state  $\{AR, IP, CH\}$  is annotated with  $[84, 82, 597]$ ,  $\{AR, IP, CH, DM\}$  is annotated with  $[50, 45, 565]$  and  $\{AR, IP, CH, DM, D\}$  is annotated with  $[0]$ .

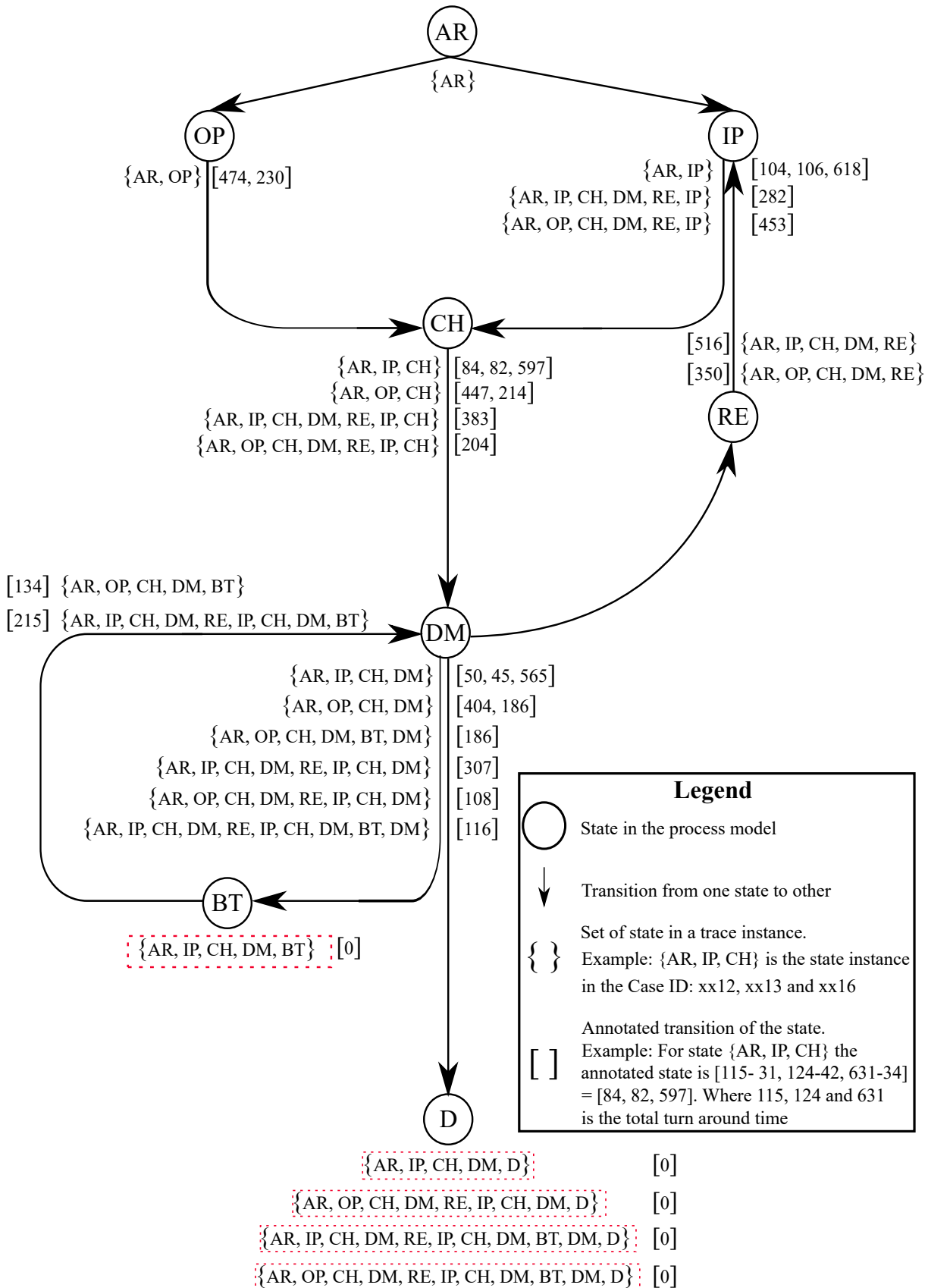


Figure 6.12: Annotated transition system based on log shown in Table 6.10.

## 6.4 Prediction Function

The annotation transition system is built with an information about the probable set of activities that precedes and succeeds the current state  $l_{Current\ State}^{transition}$  along with the information about the resource performance behaviour. It is now used for predicting the future state as shown in Figure 6.4. Let us consider the partial trace  $\sigma = \langle AR, IP, CH, DM \rangle$ . By the definition of annotation defined by Van der Aalst et al. (2011), the future state include the possibility of  $\langle BT \rangle$ ,  $\langle D \rangle$  and  $\langle RE \rangle$ . But, the prediction function that was defined in the annotation state earlier predicted the average time needed for the completion of a process. Suppose, the annotation state of the successor for  $\langle AR, IP, CH, DM \rangle$  be  $\{b_1, b_2, b_3, \dots, b_n\}$ . Here  $b_i$ ,  $i = 1, 2, 3, \dots, n$  is the start time of an event in the successor. Then by the definition of prediction function  $\bar{b} = \frac{\sum_{i=1}^n b_i}{n}$ . But in this work, we annotated the turn-around time and averaged it to find the  $\mathcal{T}_{Remaining}$ . Based on the annotation transition system  $A_T$ , the waiting time at each state is identified and it is reduced by recommending an efficient resource.

The annotated transition not only annotated the activities but also annotated the resource performance using the time they spend on completing the allotted task. The system assisted in identifying the load at a unit time and then measuring their arousal. Based on this observation an efficient resource for each activity is identified. Further for predicting the complete process we need to know the possible trace in the future state. For that we used the trace clustering and matching technique.

### 6.4.1 Trace clustering and trace matching using similarity check

Predicting the probability of the trace match for a partial trace with the cluster of traces is a crucial issue in many information retrieval and process mining applications. The traces are clustered based on the variants of process flow using the technique of Longest Common Subsequence (LCS). The aim is to match an partial trace ( $\sigma_1$ ) with the variants of traces known as annotated transition system  $A_T$ , to identify the matching traces, i.e.,  $LCS(\sigma_1, A_T)$  where  $\sigma_1$  is an incomplete partial trace.

Song et al. (2008) clustered the traces based on the feature and distance matrices. The feature matrix measures the number of features each trace has specifically for an individual event and distance matrix measures the distance between any two traces ( $\sigma_1, \sigma_2$ ). Based on the result of feature and distance matrix the traces were clustered. We applied this

technique of trace clustering and clustered the traces not only based on features but also including the duration each trace took for the completion of an assigned task. The trace matching for a partial trace was done by applying the technique of LCS with clustered traces.  $l^{length} = LCS(\sigma_1, A_T)$  measures the length of LCS between  $\sigma_1$  and  $A_T$ . Based on the matching length  $l^{length}$ , an efficiency of the process execution is predicted. This prediction avoids any emergency interventions by recommending an alternative path of execution.

On identifying the set of matched traces ( $A_T(sim)$ ) similarity test between  $\sigma$  and  $A_T(sim)$  i.e.,  $Sim(\sigma, A_T(sim))$  would estimate the similarity between them. The initial point of measurement for similarity check between  $\sigma$  and  $A_T(sim)$  is finding the  $l_p$  - distance between them, i.e.,  $l_p(\sigma, A_T(sim))$ . Here since the boundary is not defined, an infinite metric space equipped with a real space  $R^k$  known as Minkowski norms  $l_p(1 \leq p \leq \infty)$  is applied. The two points  $(\sigma, A_T(sim)) \in R^k$  and  $l_p$  - distance between them is measured by equation  $\|\sigma - A_T(sim)\|_p$  6.8.

$$\|\sigma - A_T(sim)\|_p = \left( \sum_{i=1}^k |\sigma_i|^p \right)^{\frac{1}{p}} \quad \text{for } 1 \leq p \leq \infty \quad (6.8)$$

Aim of the trace clustering and similarity checking using LCS technique is to identify and recommend the probable path of execution in future state. LCS is a well known application in the field of bio-informatics (Lin et al., 2006). It is a dynamic programming and assist in determining the maximum length of the match that can be obtained between the two strings (Ullman et al., 1976). Here two strings are the partially executed trace and the set of traces that are already been executed. The study showed that there exist lot of algorithms (Chvátal et al., 1972; Hirschberg, 1975; Wagner and Fischer, 1974) which identified the LCS, but they all had the best time complexity of  $O(n^2)$ . Hunt and Szymanski (1977) proposed an optimized method which took  $O((r+n)logn)$  time for the same problem of identifying LCS and they named it as  $O((r+n)logn)$  algorithm for LCS. Here  $r$  is the order paired position at which the sequence match. In the proposed method, identifying the positing of trace match is much important since, we are not intended in just matching the trace at any position but at the *current* position. Let A and B be two traces with sequence "babcdaa" and "abccbaaba" of length  $m$  and  $l$  respectively. LCS is "abc", but the trace actually don't match since they occur at unrelated positions. Hence for a trace to match it is very important for the activities to occur at the exact position.

By the rule of decision making using *decision tree*, let  $P_1$  and  $P_2$  be the relative position  $r$  of the string A and B (say  $a_i$  and  $b_j$ ). If  $i < j$  then  $r$  returns “less-than” else if  $i > j$  then  $r$  would return “greater-than” else it will return “equal”. If the relative position  $r$  return “equal” then only the string values are compared using LCS.

The distribution of activities based on the position of their occurrence is shown in the Figure 6.13. This information is extracted from the event logs recorded. On running LCS algorithm on the distributed set of activities, we identified the sequence of occurrence that repeatedly occurred and is shown in the Figure 6.14. The common sequence of events that repeatedly occur is known as variant. The traces occurring in the variants are clustered as good, better and best cluster and is shown in the Figure 6.15. The knowledge extracted from this information is useful for matching the sequence for a partial trace  $\sigma$ . If any delay in process execution was observed, an alternative path of execution was recommended using the discovered sequence of activities.

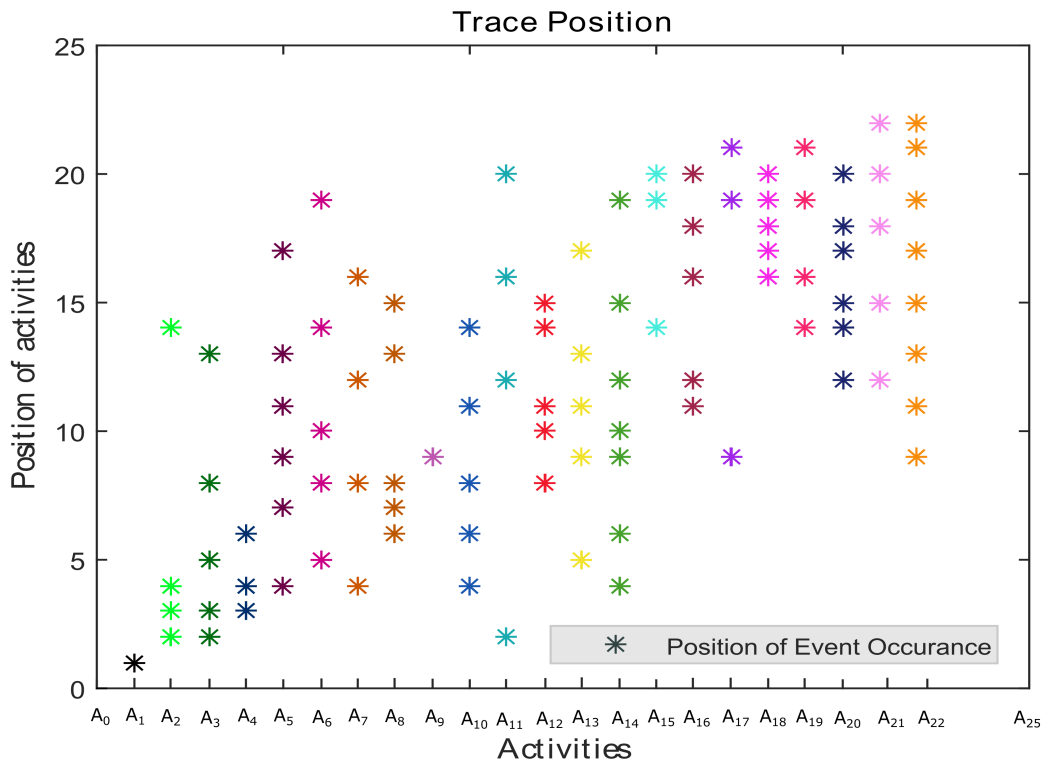


Figure 6.13: Activity position distribution showing the execution of activities.

Consider the partial trace  $\langle IP, CH, DM \rangle$  and a simple event log  $\mathcal{L}$  from the definition 6.6. By the definition of LCS the partial trace  $\langle IP, CH, DM \rangle$  was matched with the traces  $\mathcal{L}_1$ ,  $\mathcal{L}_2$ ,  $\mathcal{L}_3$  and  $\mathcal{L}_5$ . LCS would suggest the sequence of activities to be followed as  $\langle D \rangle$  from  $\mathcal{L}_1$  and  $\mathcal{L}_3$ ,  $\langle BT \rangle$  from  $\mathcal{L}_2$  and  $\mathcal{L}_5$  and  $\langle RE, IP, CH, DM, BT, DM, D \rangle$  from  $\mathcal{L}_5$ .

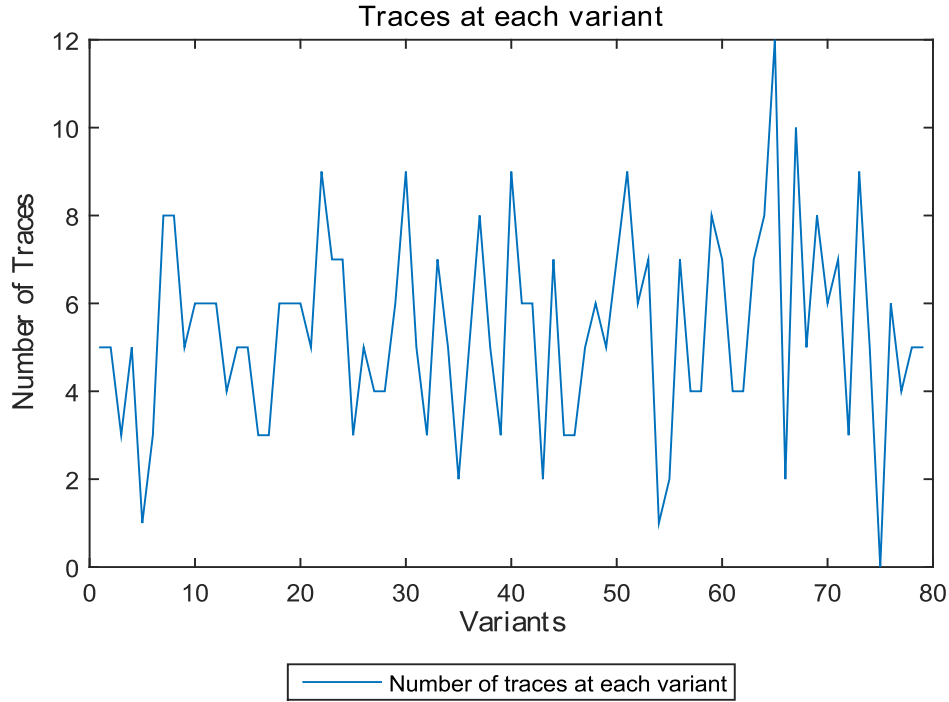


Figure 6.14: Traces at each variants.

But, based on the position  $r=right$  of the current state  $\langle DM \rangle$  of the partial trace, LCS would now recommend  $\langle BT, DM, D \rangle$  from  $\mathcal{L}_5$  as the sequence of traces to be executed in the future state.

The annotation system on matching the similarity between partial trace and the cluster of traces, identifies the possible successor. Along with, the adequate resources required for performing NPA is also identified. Thus avoiding any emergency interventions by recommending an alternative path of execution.

### 6.4.2 Identifying the resource load and their performance

The resources are classified based on their performance for an activity. The classification of best suitable resource is done for each performed activity in the process model. This analysis helps in recommending the suitable resources for performing an activity which will probably succeed the current state activity. On identifying the succeeding activity for the current state activity in the partial trace, the resources capable of performing those succeeding activities are discovered. Suitable resource is recommended using the concept of *Analytical Hierarchy Process (AHP)* and the *theory of Arousal*. Highly ranked resource with less arousal is recommended to perform the succeeding activity.

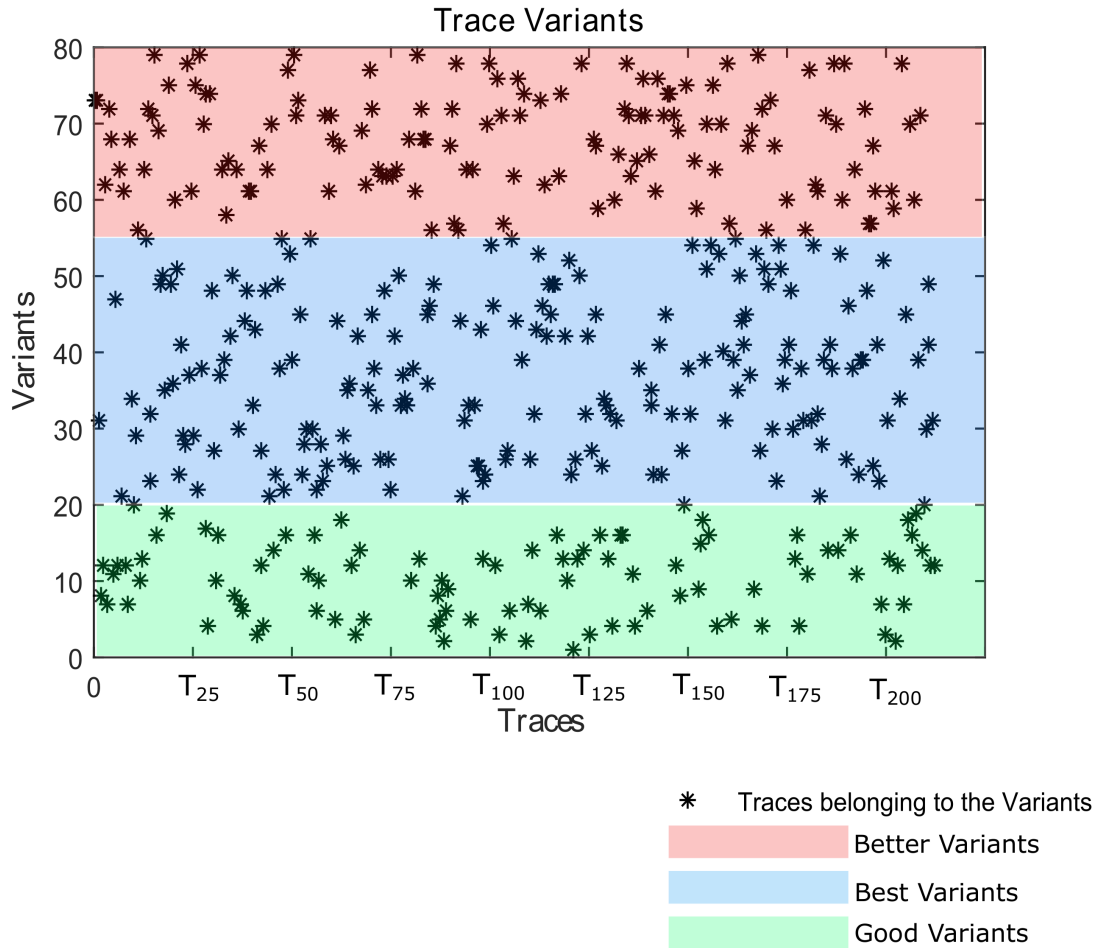


Figure 6.15: Cluster of variants showing the traces belonging to different variants.

### A Best resource for an activity: Analytical Hierarchy Process

AHP is applied for ranking the resources capable of performing the succeeding activity of the current state activity. AHP is a quantitative method for making decision and for selecting most efficient alternative. It was developed by Saaty (1970) and is extensively studied and refined since then. The AHP first decomposes the problem into a hierarchy of sub-problems where each of these sub-problems are analysed independently. Once the hierarchy is built, the decision makers evaluate systematically using pair-wise comparison. AHP converts this evaluation into numerical values that can be compared and processed on entire problem. The weight of priority ( $w_t$ ) is assigned to each element along the hierarchy and then compared in a rational manner (Saaty, 2008). Using the ranking outcome proper decisions can be taken about the resources.

Let  $A, B, \dots, G$  be the set of resources identified for performing an activity. The comparable judgement on the pair of resources  $A, B$  is represented by  $n \times n$  matrix where the entries  $a_{ij}$  in the matrix is defined by following rules

*Rule 1* if  $a_{ij} = a$  then  $a_{ji} = 1/a$

*Rule 2* If  $A = B$  then  $a_{ij} = 1, a_{ji} = 1; \Rightarrow a_{ii} = 1$  for all  $i$ .

*Rule 3* Weight and judgement relationship:

$$\frac{w_i}{w_j} = a_{ij} \text{ where } i, j = 1, \dots, n \quad (6.9)$$

*Rule 4* Multiple the first entry in  $i^{th}$  row by  $w_1$  and second entry by  $w_2$ . Therefore the general relation for  $i^{th}$  row is

$$w_i = a_{ij}w_j \text{ where } i, j = 1, \dots, n \quad (6.10)$$

More explicitly

$$w_i = \frac{1}{n} \sum_{j=1}^n a_{ij}w_j \quad (6.11)$$

*Rule 5* Principal Eigenvalue of a matrix.  $A$  is  $n \times n$  reciprocal matrix, let  $w$  be principal right eigenvector of  $A$ , let  $D = \text{diag}(w_1, \dots, w_n)$  be diagonal entries of  $w$  and set  $E/D^{-1}AD = [a_{ij}w_j/w_i] = [\gamma_{ij}]$ .  $E$  is equal to principal eigenvalue of  $A$

$$\sum_{j=1}^n a_{ij}w_j/w_i = [Aw]_i/w_i = \lambda_{max}w_i/w_i = \lambda_{max} \quad (6.12)$$

The computation reveals that  $\lambda_{max} \cong$  Eigenvalue

*Rule 6* Consistency index  $\mu$  is now chosen

$$\mu = \frac{\lambda_{max} - n}{n - 1} \quad (6.13)$$

$$\begin{pmatrix} 1 & \alpha \\ \alpha^{-1} & 1 \end{pmatrix} \begin{pmatrix} 1 + \alpha \\ (1 + \alpha)\alpha^{-1} \end{pmatrix} = 2 \begin{pmatrix} 1 + \alpha \\ (1 + \alpha)\alpha^{-1} \end{pmatrix}.$$

On getting the index  $\mu$  for each resource, the resource with higher index value is decided to be efficient to perform the succeeding activity.

## **B Resource load and performance analyser**

The *Yerkes-Dodson Law of Arousal* also known as *theory of Arousal* states that by increasing arousal, the workers performance can be improved. However, if the level of arousal



increases too much, performance decreases (Nakatumba and Van der Aalst, 2009). Here, the *level of arousal: optimal load* the resource can handle efficiently is identified using *ADALINE* model. For this, the performance of each resource at different load is recorded. The performance is then analysed by increasing the load and observing the time taken. It was observed that, as the load increases the time taken to complete the assigned task was decreasing, but at some point there was a drift where the time taken started increasing. That drifted point is known as Arousal.

Thus in pre-processing phase, information about resources performing the activities are identified. The resource set is then developed for each activity, which are then ranked. The higher ranked resources have high probability for performing that activity. The performance capability known as *level of arousal* for each resource is identified.

Consider the load and service time by a resource in the Table 6.11. *Load* is the size of task assigned on a resource per unit time and the *service time* is total time taken by that resource for executing that assigned task.

Table 6.11: Load and Service Time

|                 |    |    |    |    |    |    |    |    |    |    |
|-----------------|----|----|----|----|----|----|----|----|----|----|
| Load: x         | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10 |
| Service Time: y | 32 | 29 | 26 | 21 | 19 | 24 | 29 | 35 | 40 | 42 |

Let  $\mathcal{X}$  and  $\mathcal{Y}$  be input and output value space, then the hypothesis  $h(x)$  is  $\mathcal{X} \mapsto \mathcal{Y}$ . The hypothesis  $h$  is represented as in equation(6.14) and is known as univariate linear regression function, where  $\theta'_i$ s are the weight vector and vector  $\bar{x}_0$  is always = 1.

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1 = \sum_{i=0}^n \theta_i x_i = \theta^T x \quad (6.14)$$

The cost function of linear regression model is shown in equation 6.15.

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^i) - y^i)^2 \quad (6.15)$$

The cost function  $J(\theta)$  is now measured for each training example. The resultant cost function is shown in Table(6.12).

Curve plotted in the Figure6.16 is obtained by fixing  $\theta_0 = 0$ . On using both weight parameters ( $\theta_0$  &  $\theta_1$ ) the hyperplane is obtained and is shown in Figure 6.17 the corresponding cost function values are shown in Table 6.13. In the contour graph shown in Figure 6.17 each curve represents the cost function. Objective is to find an optimal  $\theta$

Table 6.12: Cost function with  $\theta_0$  fixed as 0

|               |        |      |       |      |       |      |       |      |       |       |        |
|---------------|--------|------|-------|------|-------|------|-------|------|-------|-------|--------|
| $\theta_1$    | -0.5   | 0    | 0.5   | 1    | 1.5   | 2    | 2.5   | 3    | 3.5   | 4     | cont.. |
| $J(\theta_1)$ | 110.45 | 88.2 | 68.45 | 51.2 | 36.45 | 24.2 | 14.45 | 7.2  | 2.45  | 0.2   | cont.. |
| $\theta_1$    | 4.5    | 5    | 5.5   | 6    | 6.5   | 7    | 7.5   | 8    | 8.5   | 9     | 9.5    |
| $J(\theta_1)$ | 0.45   | 3.2  | 8.45  | 16.2 | 26.45 | 39.2 | 54.45 | 72.2 | 92.45 | 115.2 | 140.45 |

value, so that cost function is minimum and this is achieved by using *gradient descent* algorithm.

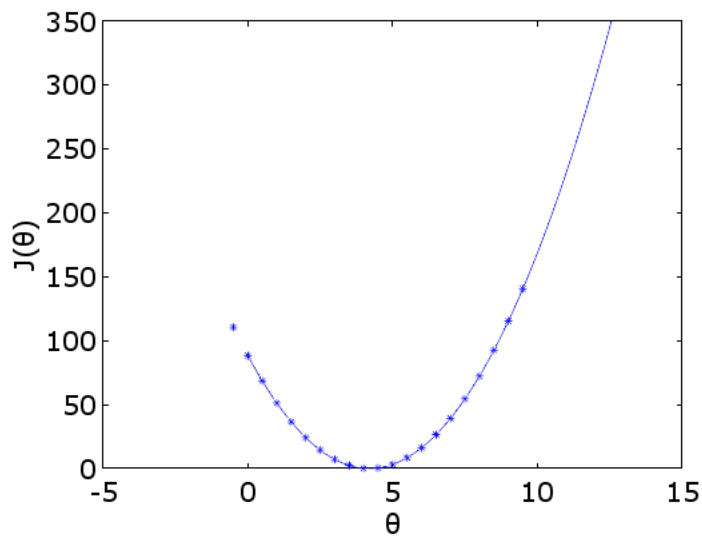


Figure 6.16: Cost function

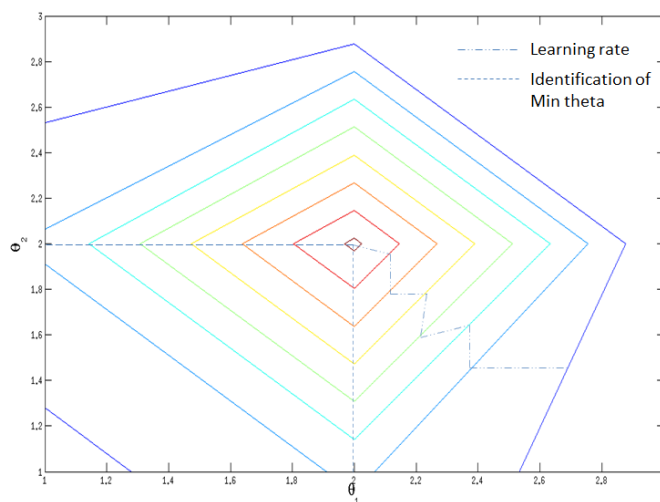


Figure 6.17: Learning rate of gradient descent using contour graph

Table 6.13: Cost function with  $\theta_1$  &  $\theta_2$ 

|                         |             |             |             |             |             |
|-------------------------|-------------|-------------|-------------|-------------|-------------|
| $\theta_1$              | -0.5        | 0           | 0.5         | 1           | 1.5         |
| $\theta_1$              | -100        | -50         | 0           | 50          | 100         |
| $J(\theta_1, \theta_2)$ | 5.4340e+004 | 1.4688e+004 | 8.6113e+001 | 1.0534e+004 | 4.6032e+004 |
| $\theta_1$              | 2           | 2.5         | 3           | 3.5         | 4           |
| $\theta_1$              | 150         | 200         | 250         | 300         | 350         |
| $J(\theta_1, \theta_2)$ | 1.0658e+005 | 1.9218e+005 | 3.0283e+005 | 4.3852e+005 | 5.9927e+005 |

### Gradient Descent

Gradient descent algorithm starts the iteration with some initial  $\theta$  such as  $J(\theta_0 = 0 \ \& \ \theta_1 = 0)$  and iteratively updates  $\theta$  as shown in equation 6.16 till the algorithm converges at local minimum. The objective of gradient algorithm is to *minimize*  $J(\theta_0, \theta_1, \dots, \theta_n)$ . The algorithm starts an iteration by moving with small baby step  $\alpha$  known as learning rate, in a direction that moves down to reach minimum  $\theta$ . After each iteration, the algorithm will check the direction of movement that converges to local minimum. The gradient descent repeats the equation 6.16 until it converges, by updating the  $\theta_j$  value after every iteration.  $\frac{\partial}{\partial \theta_j}$  is measured by theorem 3. Convergence is the stopping condition:  $(\alpha \|\nabla J\|) > \xi$ , where  $\|\nabla J\|$  is  $\sqrt{J(\theta_1^2) + J(\theta_2^2) + J(\theta_3^2) \dots}$  is equation of normalization Zhang (2004).

$$\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) \quad \text{for all values of } j : 0, \dots, n \quad (6.16)$$

Gradient decent finally converges at the optimal point which would be the optimal load a resource could handle. With this information and with the information about the current load on each resource, the annotated transition system would recommend the right resource to perform the succeeding activity in the future state. Thus, on identifying the succeeding activity and resource, next thing the annotated transition system had to do was to check the critical path of execution. The complete learning steps in gradient descent is shown in Figure 6.18. Here the learning stop when the algorithm converges giving the optimal load of the resource.

### 6.4.3 Identifying critical path activities

ModCNN helped in identifying the cases which were critical and needed interventions. Now, using the process mining techniques the critical path of execution for those identified critical cases were recommended. The critical path is a sequence of critical activities and

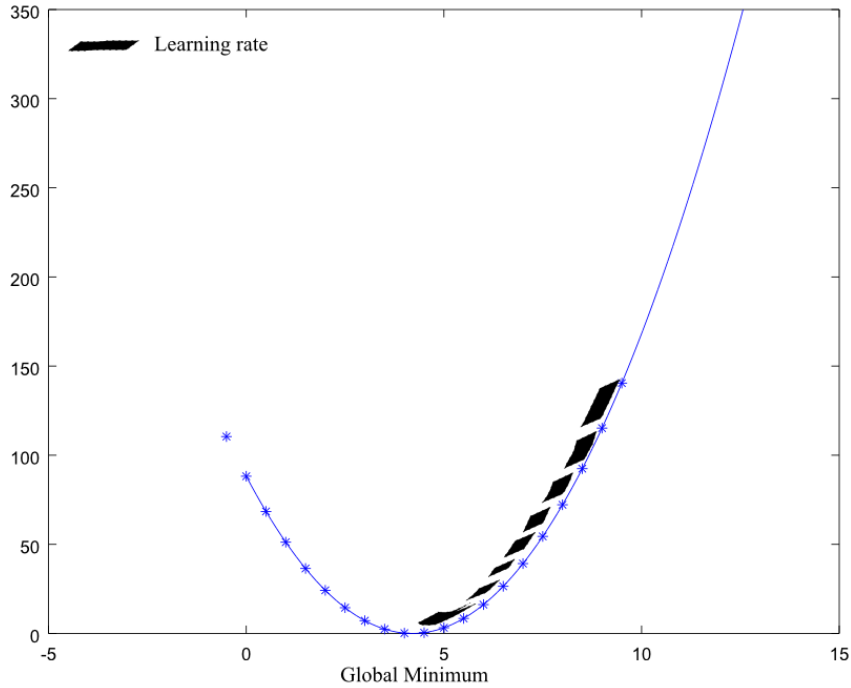


Figure 6.18: Learning rate in gradient descent

can be defined using:

- Earliest Start time (ES)** : *Earliest time an activity can be started (with waiting time  $\simeq$  zero).*
- Earliest Finish time (EF)** : *ES + processing time.*
- Latest Finish time (LF)** : *Latest time an activity could be started.*
- Latest Start time (LS)** : *LF – processing time.*

Consider the time line illustration shown in Figure 6.19 for the trace  $\langle AR^{0,15}, IP^{3,11}, CH^{7,15}, DM^{12,18}, BT^{19,15} \rangle$ . The figure clearly illustrate the turn-around time, waiting time and processing time. While recommending the critical path of execution we need to assure that all the critical activities are included. For that we need to first identify which are the significant activities. An activity is critical if the earliest and latest finish time is same, i.e., *waiting time  $\simeq$  zero*. If  $P_5$  with waiting time of 30 units is identified as critical, then we need to reduce its waiting time by employing an efficient resource for its execution.

#### 6.4.4 Steps involved in identifying critical path

The critical path was identified using following steps:

- **Identifying the specification of an activity:**

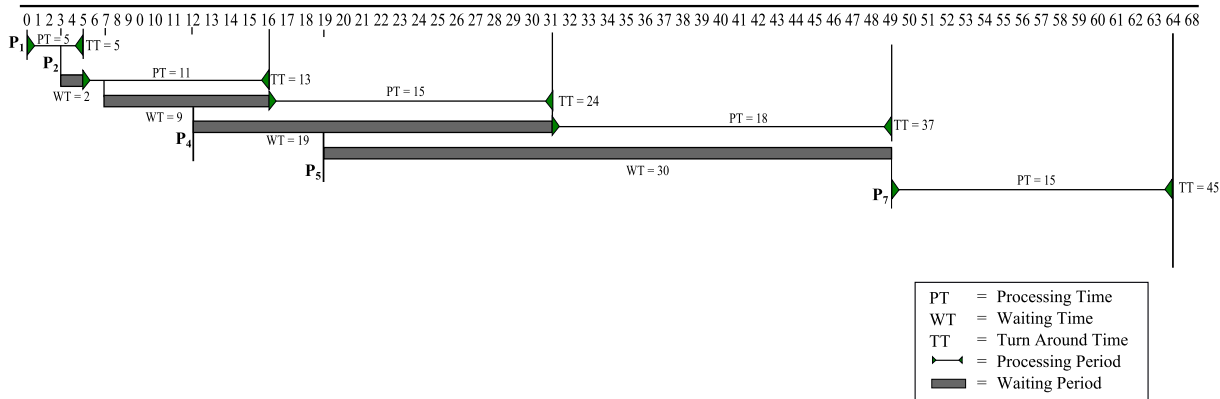


Figure 6.19: Illustration of process time execution.

Here, the  $l^{state}$  of an activity that includes  $l^{transition}$ ,  $\mathcal{T}_{Waiting}$ ,  $\mathcal{T}_{Turn-around}$ ,  $\mathcal{T}_{Remaining}$  is identified along with its successor and predecessor. Using causal metric we identified its waiting and turn-around time. Among them an activity with lesser waiting and turn-around time is given higher priority.

- **Establishing activity sequencing**

Using the transition metric  $l^{transition}$ , performance of each activity and resource at different position is identified. Along with the time function  $\mathcal{T}$  for each activity, their possible executable position  $\mathcal{P} \in P_1, P_2, P_3, \dots, P_n$  is also discovered. Based on the outcome of this, we measured the best possible position of each activity.

- **Construct the process model**

With the information of different possible positions of all the activities and the resource performance information, we could discover all the possible critical traces in a process model.

- **Identify the critical activity**

Among the specified activities, we identify the critical ones with higher priority and minimum waiting time.

- **Construct the critical path**

Based on the priority of each activity, its waiting time was decreased by assigning the best efficient resource who could complete the assigned task within the allotted time.

## 6.5 Summary

The current study aimed in identifying the complicated cases at the early stage of disease progression. This would prevent any complications in the later stage of disease. With the help of ModCNN critical cases were identified. Such identified critical cases needed immediate interventions and for that, critical treatment path was recommended. Critical treatment path is a sequence of critical activities, where each activity was performed by adequate resource. With the help of EHR process mining technique, we were able to build a annotation transition system and recommend this critical treatment path. The approach followed by us is explained as a flow chart in the Figure 6.20.

Complicated cases identified by ModCNN



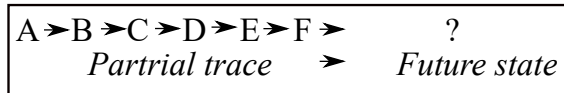
Recommend the critical treatment path



Construct the annotated transition system

*Aim of annotated transition system is to*

1. find and recommend the future state of execution



2. Identify the adequate resources



1. Identify the next succeeding activity in the future state.

This was achieved by:

1. **Activity metric:** To identify the waiting time at each activity
2. **Transition metric:** To identify the performance of activity and resource at different position of execution
3. **Causal metric:** Build a causal relationship between the succeeding and preceding activity. This is needed to find the reason of occurrence of an activity



2. On identifying the activity to succeed the current state, we measure the cost of occurrence of that activity using TDABC



3. On identifying the activity to succeed the current state, we need to find the efficient resource who is available and capable of performing the recommended activity.

1. **Theory of Arousal:** Using this theory proposed by Yerkes-dodson, we identified the optimal load of each resource where his performance is better. Based on this finding the recommendation was made.

2. **Analytic Hierarchy Process:** The concept of AHP was used to rank the resources. This means that higher the rank of the resource, his performance is higher.

3. Hence, Based on ranking and availability, the adequate resource was recommended to perform the activity in the future state.

Figure 6.20: Flow chart showing the chapter summary.





## Chapter 7

# Performance Evaluation of Annotated Transition System

The healthcare management are primarily faced with two uncertainties i.e., in managing the resources and the facilities. Due to this, efficient resource utilization could not be made, leading to high fluctuations between the occupancy and demand. The uncertainty of healthcare system could be reduced if the resource and disease behaviour is predicted well in advance along with the length of stay and journey the patients would make in the hospital. Journey of a patient is the path he follows to complete the treatment process. The path followed is known as a trace in process mining.

Aim of the study is to identify the cases that are critical at initial stage of observation and recommend a critical treatment path, to avoid any later stage complications. In this work, we applied the concept of EHR process mining to develop a annotated transition system. This system assisted in predicting the best possible future state for the partial incomplete trace. The system tried to identify the critical activities in the path along with the efficient resources capable of conducting those activities. The transition system conducted various conformation tests before recommended the path of execution. The goal was to recommend the critical treatment path having all the critical activities along with the probable time of completion. Thus the annotated transition system assisted not only in easing the journey of patients in the hospital but also reduced their stay in the hospital along with the cost. This chapter evaluates the performance of annotated transition system.

## 7.1 Length of Stay in the Hospital

EHR system was installed to record the journey of the patients who were suffering with complicated GSD. The length of stay for complicated GSD was higher along with their cost, when compared with uncomplicated GSD. This was because, the uncomplicated cases were those on whom OC/ LC was not performed. The length of stay is shown in Table 7.1. The Figure 7.1 explains that the frequency is very high for the complicated cases in the initial hour of treatment, which gradually decreases. But, for the uncomplicated cases there is a constant frequency with standard deviation ( $SD = 3.8$ ) when compared with complicated cases having  $SD = 8.3$ .

Table 7.1: Length of stay and its cost in the hospital

| Attributes                                    | Diagnosis     |               | Probability value          |
|---|---------------|---------------|----------------------------|
|   | Uncomplicated | Complicated   |                            |
| No. of Cases                                  | 270           | 260           | ...                        |
| Length of stay in hospital, mean and $\pm$ SD | $3.3 \pm 3.8$ | $7.8 \pm 8.3$ | $< 0.0001$ , $t$ test      |
| LC/ OC ration                                 | 170.5:1       | 38.5:1        | $< 0.0001$ , $\chi^2$ test |

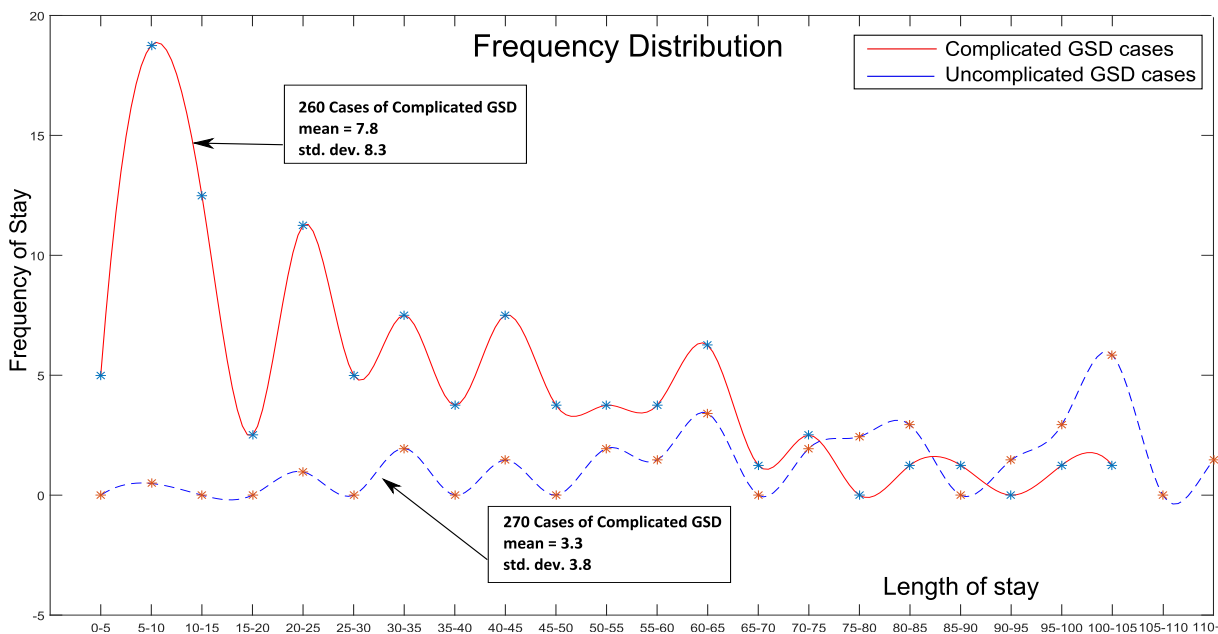


Figure 7.1: Frequency distribution of length of stay.

This is a priori knowledge representation and there is no optimization techniques being applied for reducing this length of stay distribution. Average length of stay is 65 – 70

hours with mean and  $SD = 4.8 \pm 3.6$ . The Figure 7.2 and 7.3 shows the corresponding frequency distribution along with the average length of stay for the identified spectrum of complicated GSD. On observing the  $SD$  of each spectrum, we understood that among the complicated cases though choledocholithiasis and cholecystitis had more mean value, their  $SD$  was lesser than pancreatitis.

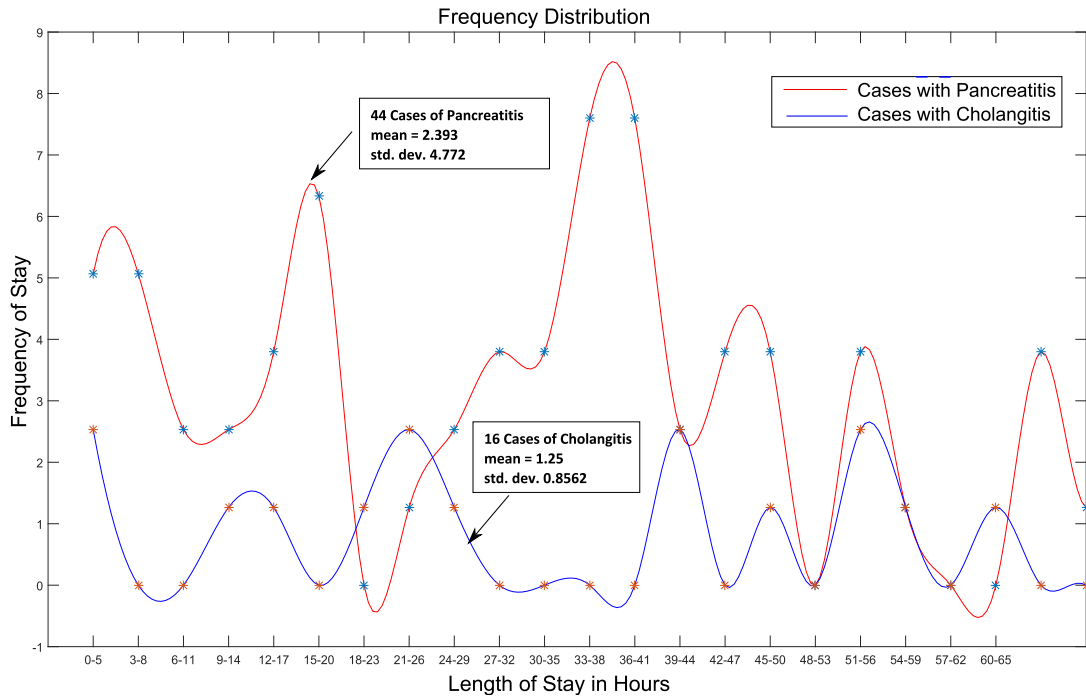


Figure 7.2: Frequency distribution of cholecystitis and choledocholithiasis.

### 7.1.1 Statistics

For the patients on whom OC/ LC was performed, the  $SD$  of length of stay from the time of admission to the procedural date was  $90 \pm 15$  hours. This delay was significantly less for the critical cases, as we identified their progression at the time of admission itself. Table 7.2 shows the mean delay of  $42 \pm 18$  hours between the duration of first USG where gallstone was confirmed to the date of OC/ LC. But, the delay for the critical cases between the USG and procedure of LC/ OC was significantly lesser.

This treatment delay was well assessed and overcome by identifying critical path of execution in future state and appointing efficient resources. This reduced delay rate for each critical activities in the critical path is shown in Table 7.3. The procedural diagram of LC and OC for different spectrum of GSD is illustrated in the Figure 7.4 7.5 7.6.

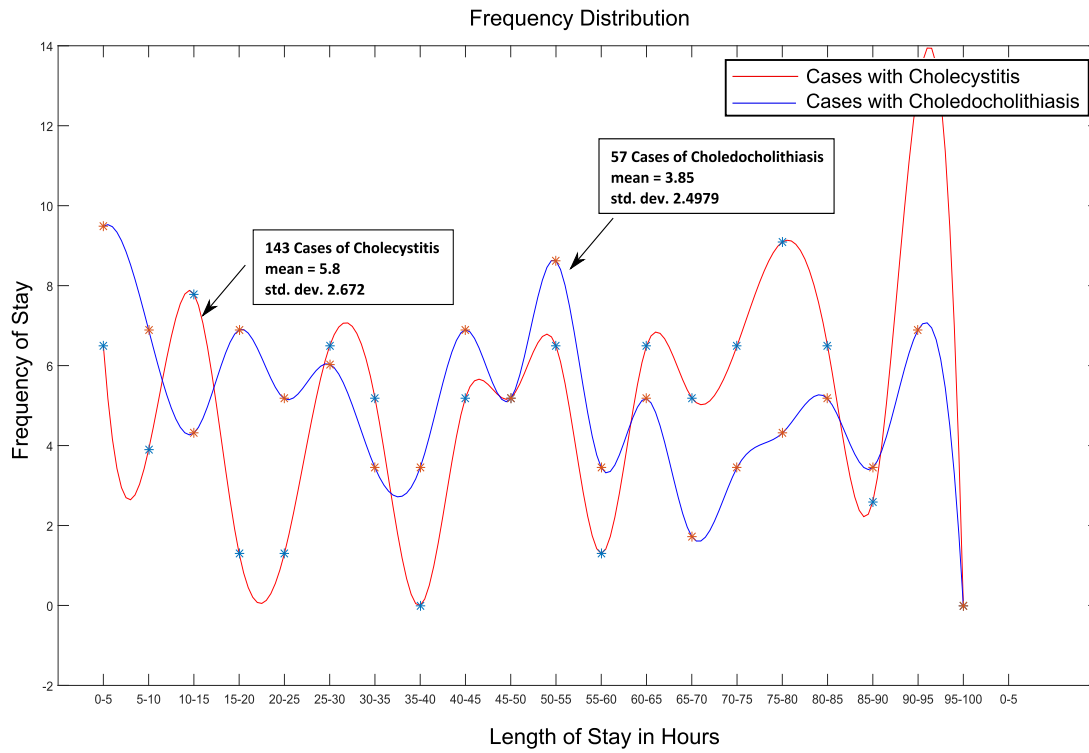


Figure 7.3: Frequency distribution of pancreatitis and cholangitis.

Table 7.2: Delay in length of stay

|  | Diagnosis            |                    | P                       |
|--|----------------------|--------------------|-------------------------|
|  | <i>Uncomplicated</i> | <i>Complicated</i> |                         |
| Delay: From time of admission to surgery mean and $\pm$ SD | 34 $\pm$ 12          | 90 $\pm$ 15        | < 0.0001, <i>t</i> test |
| Delay: From first USG to surgery mean and $\pm$ SD         | –                    | 42 $\pm$ 18        | < 0.56, <i>t</i> test   |

Table 7.3: Comparison in length of stay before and after applying the proposed system

|  | Complicated Cases          |                           | P                       |
|--|----------------------------|---------------------------|-------------------------|
|  | <i>Before</i> <sup>1</sup> | <i>After</i> <sup>2</sup> |                         |
| Delay: From time of admission to surgery mean and $\pm$ SD | 60 $\pm$ 15                | 28 $\pm$ 6                | < 0.0001, <i>t</i> test |
| Delay: From first USG to surgery mean and $\pm$ SD         | 42 $\pm$ 18                | 8 $\pm$ 3                 | < 0.001, <i>t</i> test  |

<sup>1</sup> Reading recorded in a retrospective way.

<sup>2</sup> Recorded after critical path was recommended with efficient resource for each critical activities.

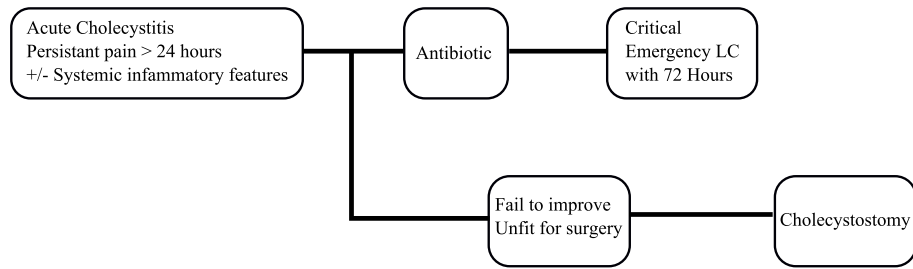


Figure 7.4: Gallstone management for ERCP of cholangitis.

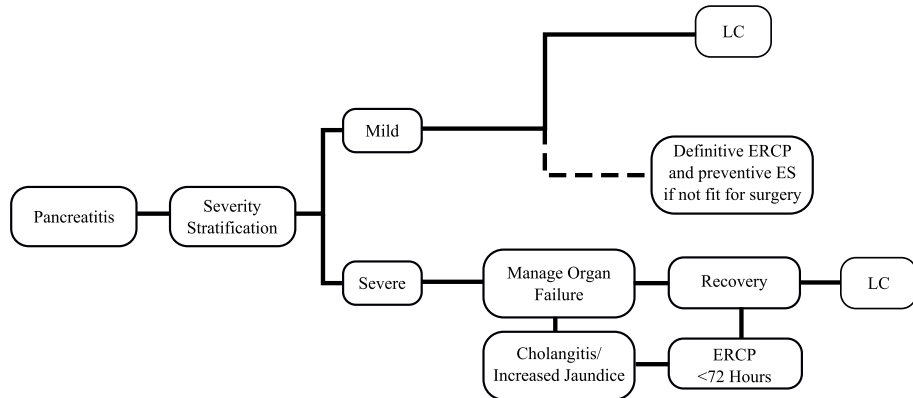


Figure 7.5: Gallstone management for ERCP of pancreatitis.

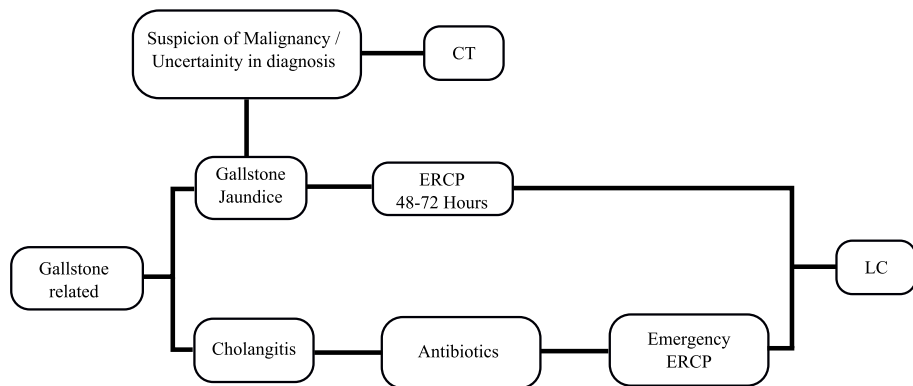


Figure 7.6: Gallstone management for ERCP of gallstone related jaundice and cholangitis.

## 7.2 An approach to develop a decision support system for GSD management

To build an established healthcare process model we need to reduce the waiting time at each event. This would in-turn decrease the length of stay in the hospital along with the journey of patient. Let us consider the GSD management process model followed and observed in the current study, shown in the Figure 7.7. The activities that were observed are *patient waits for consultation, History and physical examination, Lab investigations, MD consultation, Decision making, USG, CT, Surgical procedure*. On analysing the signifi-

cance of each activity, we could understand their behaviour. Figure 7.8 and 7.9 illustrates how the proceeding and succeeding activities waiting time is considered for analysis. Each scattered dots are the waiting time distribution and the bar is process execution. Aim is to reduce this scattered dots. The waiting time at each activity could be either because of unavailability of proper facilities or inefficiency of the resources.

By constructing annotated transition system, we were able to identify the waiting time and measure the resource performance. Using this system, we optimized the process model by discovering the critical path of execution and reducing the waiting time. Thus, by recommending/ replacing each identified resource by an efficient resource, we could optimize the process execution.

### 7.2.1 Resource performance

*Overall Equipment Effectiveness* (OEE) is used for measuring how effectively the resource available is performing. OEE is calculated using three main factors: *availability*, *performance* and *quality* and is shown in equation 7.1.

$$OEE = availability \times performance \times quality \quad (7.1)$$

#### A *Availability*

It is a state of an event where it has stopped execution before it was scheduled to be stopped. Suppose, a resource has to provide his service for 10 hours in a day and by some reason on a particular day he was not available for 2 hours, then his availability is 10 hours - 2 hours = 8 hours.

**Example 1** Let us consider that resource A is theoretically capable of consulting 25 cases in one hour. On a normal day he works for 12 hours i.e., he consults 300 cases in a normal day. On some particular day he was not available for 2 hours, then:

---

|  |                                       |
|--|---------------------------------------|
| <i>Scheduled consultation time</i>                 | 12 hours                              |
| <i>Down time (not available)</i>                   | 2 hours                               |
| <i>Available time</i>                              | 12 - 2 = 10 hours                     |
| <i>Available time/ scheduled consultation time</i> | 10 hours/ 12 hours = 83% availability |

---

Thus the availability of resource A is 83%

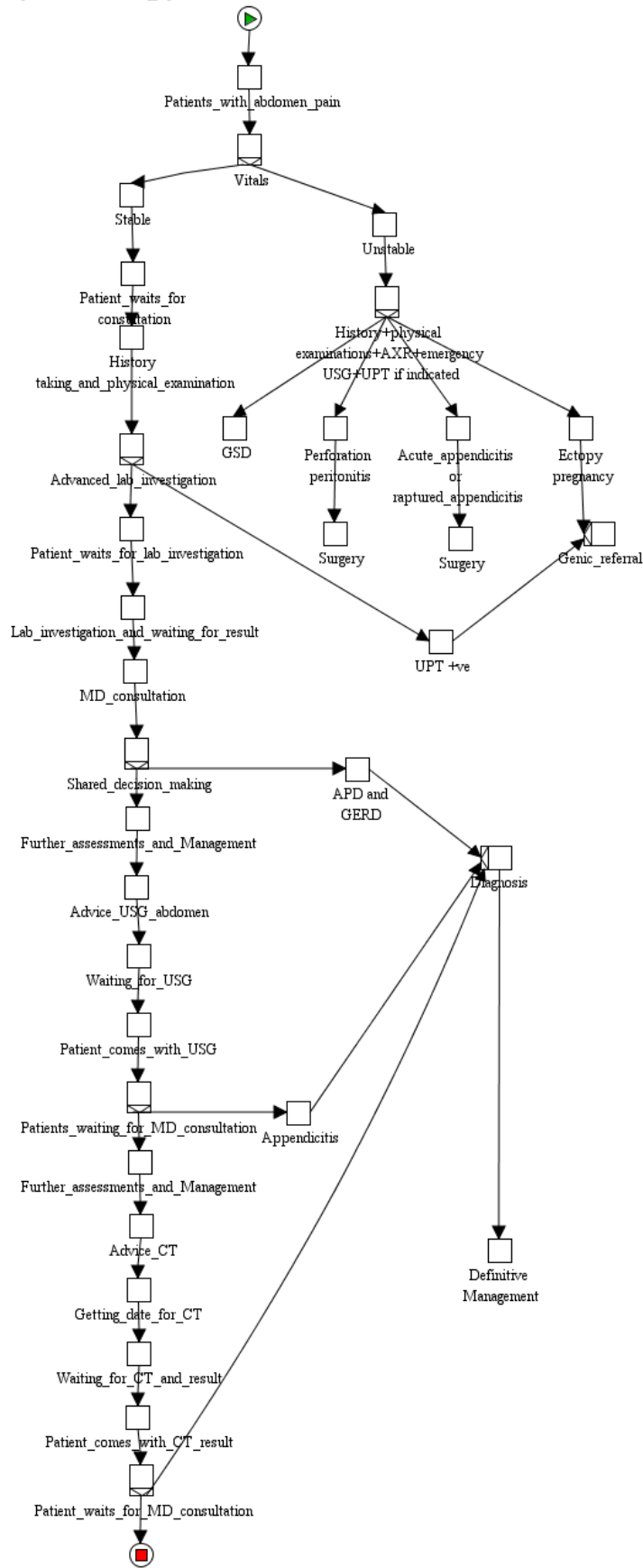


Figure 7.7: Process model of current system.

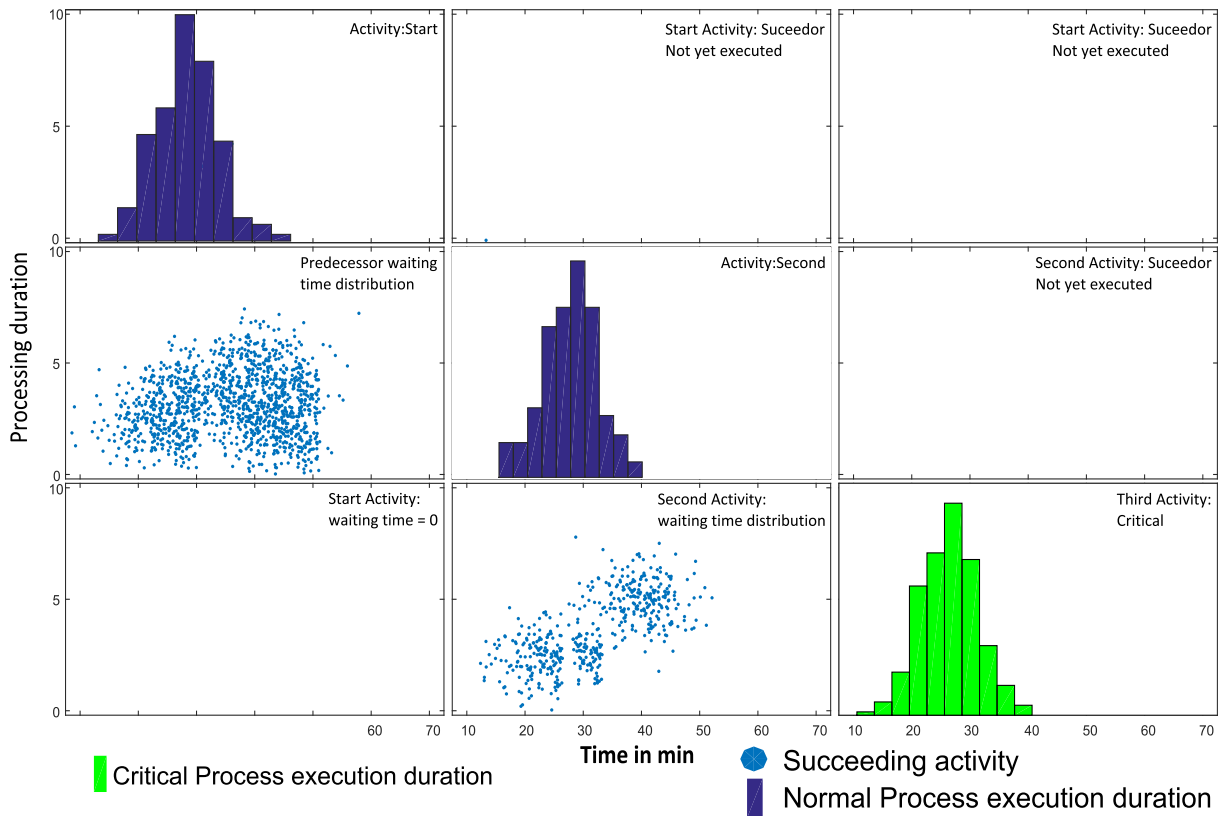


Figure 7.8: Analysing the activities for their processing and waiting time 1.

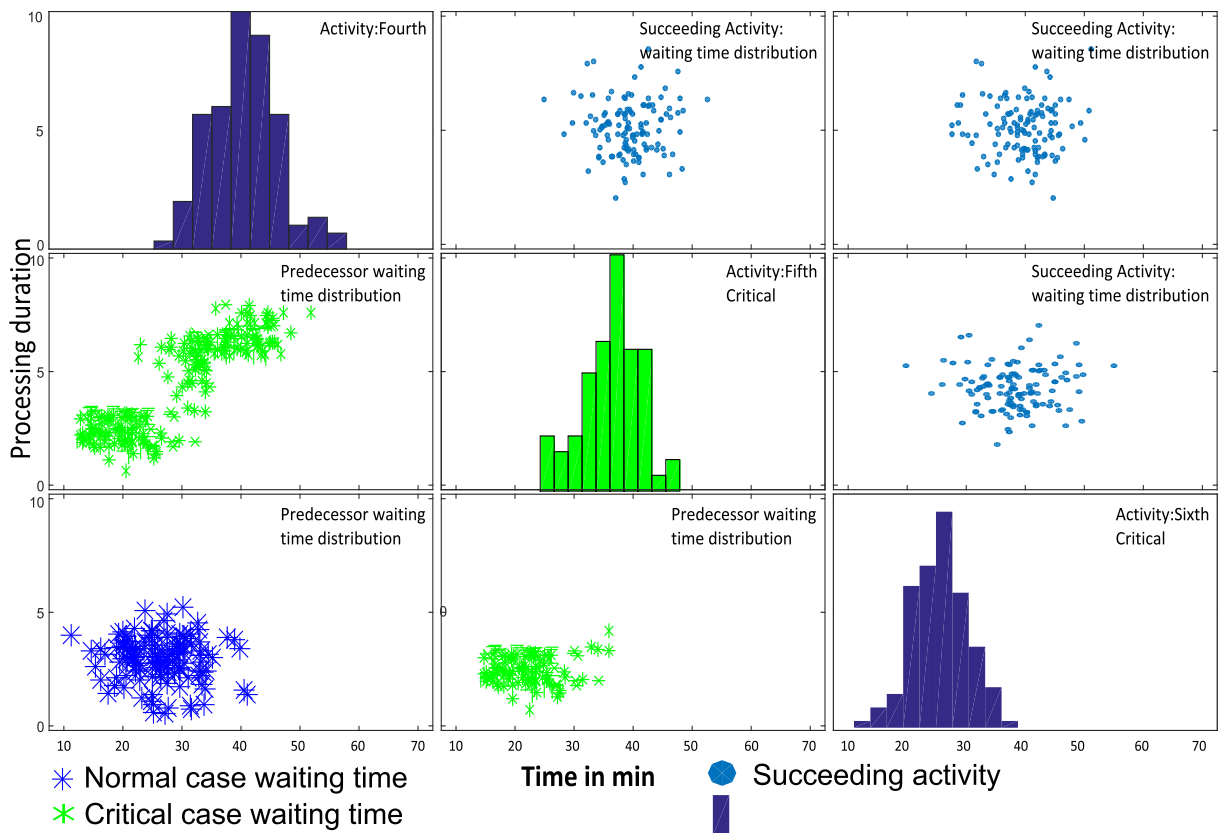


Figure 7.9: Analysing the activities for their processing and waiting time 2.



## B Performance

It is an estimation of how a resource has completed the assigned task. Performance finds the reason/ cause of *under-performance* of the resource. Actual performance is measured based on the average time he normally takes to serve the assigned task.

*Continuing the previous example*, let the recorded diagnosed cases be 225 for 10 hours of availability. It should be remembered that the resource A is capable of serving 25 cases per hour. Based on this observed information:

---

|  |                                       |
|--|---------------------------------------|
| <i>Available time</i>                                    | 10 hours                              |
| <i>Total consultation (throughput) in available time</i> | 225 cases in 10 hours of time         |
| <i>Theoretical time to serve 225 cases</i>               | 225 cases/25 cases per hour = 9 hours |
| <i>Theoretical time/ Available time</i>                  | 9 hours/ 10 hours = 90% performance   |

---

## C Quality

Quality measures how correctly the resource has diagnosed the cases. This is needed to find error the resource has made during that particular day of service.

*Continuing the previous example*, out of 225 cases consulted, it was observed that only 200 cases were properly diagnosed, and remaining 25 cases were wrongly diagnosed. Based on this information quality is checked as follows:

---

|   |  |
|---|--|
| <i>Consultation/ theoretical service time</i>         | 9 hours                                  |
| <i>Properly diagnosed cases in time</i>               | 200 cases/ 25 cases in 1 hours = 8 hours |
| <i>properly diagnosed (in time)/ performance time</i> | 8 hours/ 9 hour = 89% Quality            |

---

Hence  $OEE = availability \times performance \times quality = 83\% \times 90\% \times 89\% = 66\%$ . This means that, there is loss of 34% in a day. Since the resource ideally works for 12 hours in a day then 34% of 12 hours = 4 hours of service period is lost.

Further, using the concept of TDABC we were able to identify the cost and quality of each activity. This cost is the time taken for the completion of a task. As the cost of the resources was measured based on the time they spend on each allotted task, more the time they spend, more costly would be each activity. This, not only delays the completion of the process, but also become expensive. Hence it was highly needed to find the cost of each activity along with the load at each resources to provide better quality service.

Figure 7.10 and 7.11 shows the error rate, where the difference of cost driver rate was compared with the impact of practical capacity (*for illustration please refer Table 6.8 and 6.9*). In the Figure 7.10 and 7.11, length of error bar specify the duration of corresponding activity. In 7.11 we could see the length of error bar is significantly smaller than in 7.10 and the trace completion time is also averaging around 50 unit, when compared to 90 units in 7.10. This was because, values in the Figure 7.11 were obtained after optimizing the availability, performance and quality of the resources.

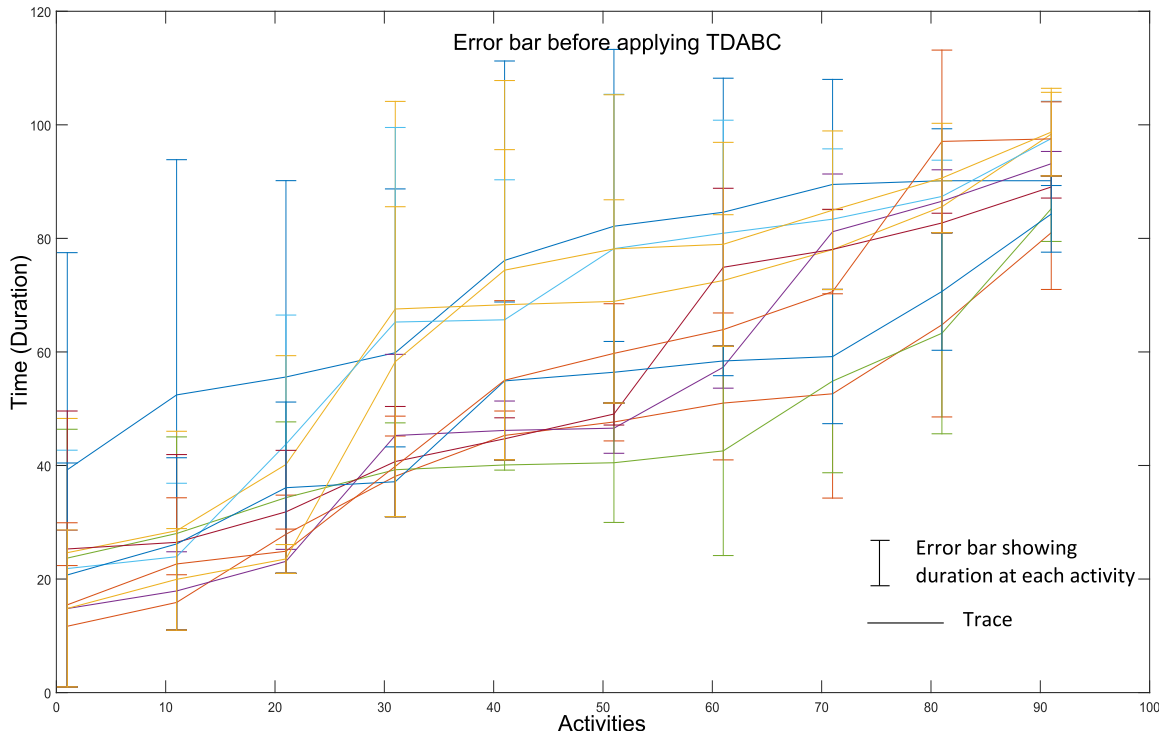


Figure 7.10: Time taken by resources for the completion of assigned task.

### 7.2.2 Trace execution

Further, the future state of partial trace  $\sigma$  was identified using the technique of trace clustering and matching. Here the traces were clustered using LCS. The partial trace  $\sigma$  was matched with the variant of traces (annotated transition system  $A_T$ ), to identify the matching traces, i.e.,  $LCS(\sigma, A_T)$ . The LCS technique was modified using the technique of trace clustering proposed by Song et al. (2008), to match  $\sigma$  with the cluster of traces based on the feature and distance matrices.

Thus, two major objectives: identifying the probable path of execution in the future state and finding the efficient resources available for executing each task in that path, was achieved in this work. For validation, we retrospectively ran the recommended path and

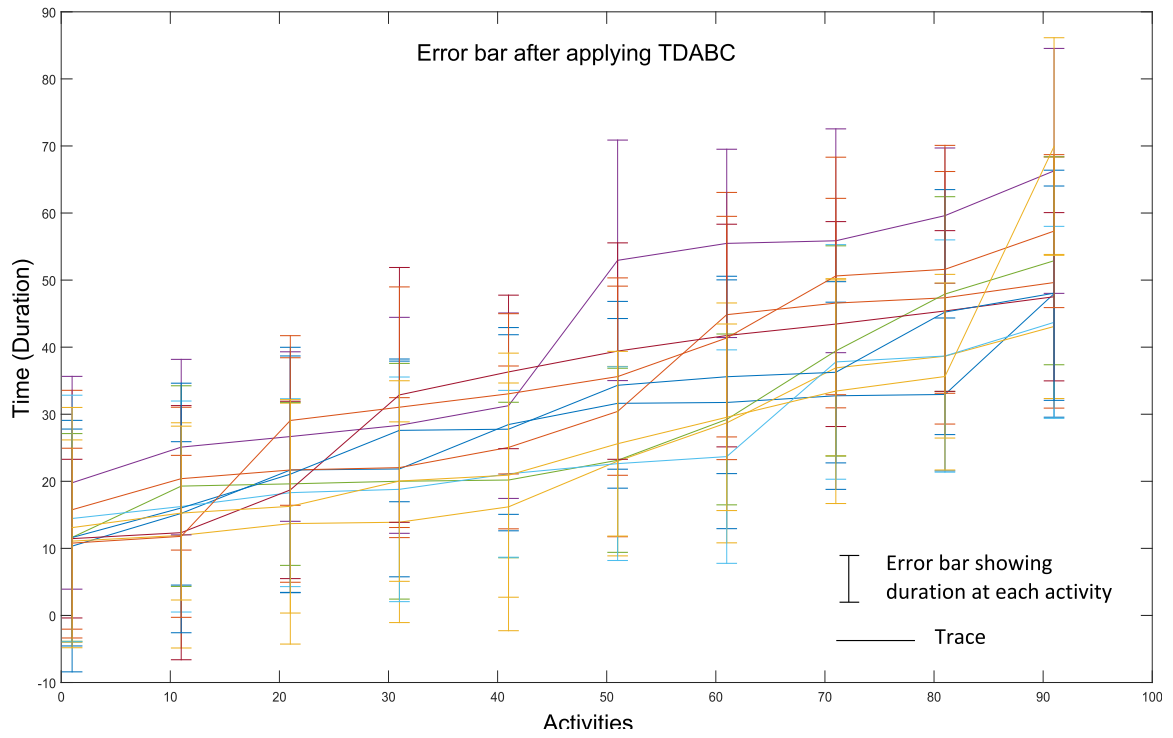


Figure 7.11: Time taken by resources for the completion of assigned task after application of TDABC.

compared its performance with that of conventionally executed path. We could observe that recommended path of execution took substantially lesser time for completing the task, when compared with conventional path of execution. This is well illustrated with a help of stem graph in Figure 7.12.

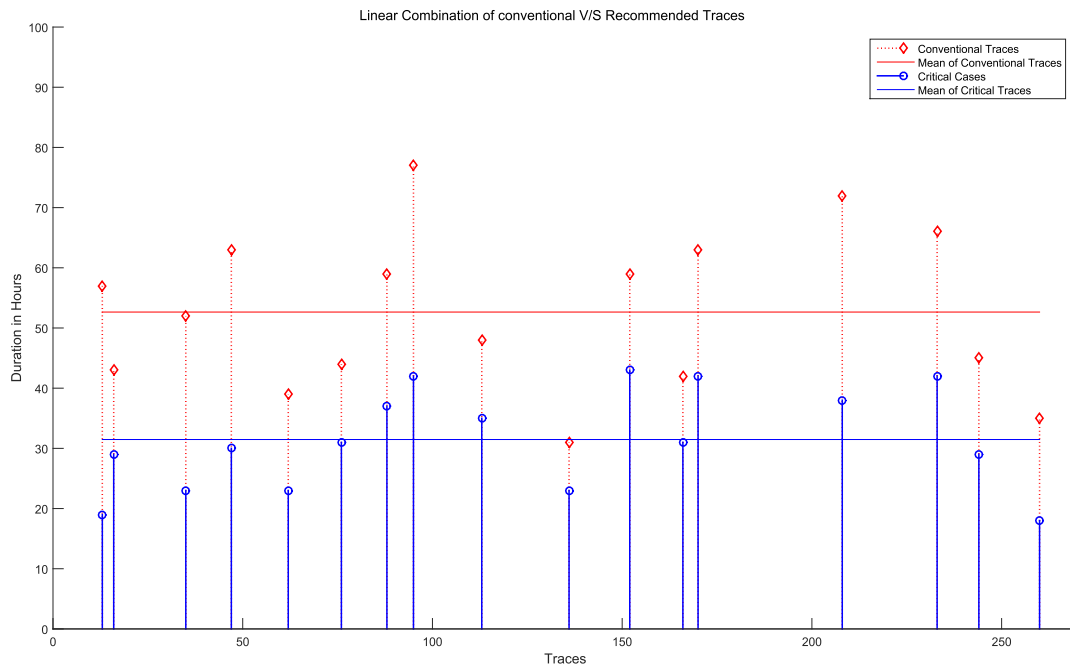


Figure 7.12: Conventional v/s recommended traces.

On optimization and recommendation, we checked the frequency rate and distribution of waiting time in predecessors and successors. On comparison, the earlier highest frequency of stay for cholecystitis was 14 which was reduced to 3.5–4, similarly for choledocholithiasis, it was 8–10, and was reduced to 2.5–3 (7.2). On comparing the highest frequency stay for pancreatitis, it was 8–9 which was reduced to 3.5–4, for cholangitis it was reduced from 2–3 to 1.5–2 (7.3). The resulted frequency distribution is shown in the Figure 7.13 and 7.14 respectively. The decrease in waiting time distribution is shown in the Figure 7.15.

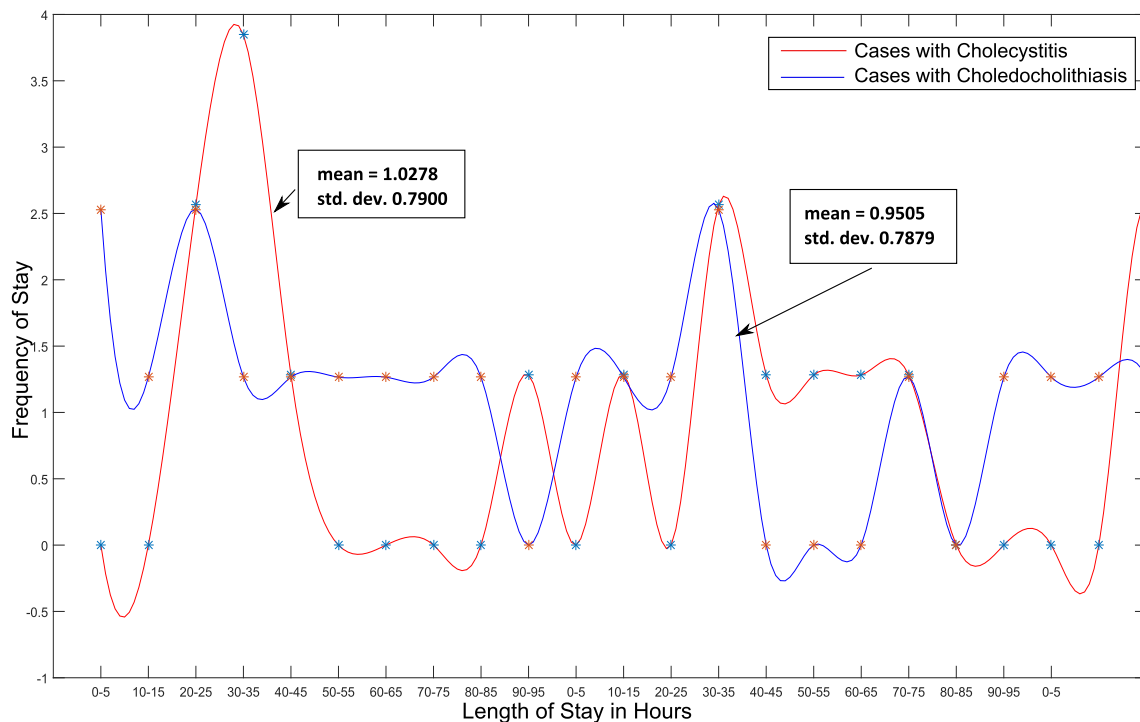


Figure 7.13: Frequency distribution of cholecystitis and choledocholithiasis after optimizing the resource performance.

### 7.3 Accuracy of prediction

The accuracy of prediction and recommendation about the future state and process behaviour was evaluated using the concept of  $A_Z$ . The accuracy of recommendation was  $A_Z = 93.562$  and is shown in Figure 7.16.  $A_Z$  is one of the well established statistical technique for evaluating the model performance. Higher the area under the curve more is the accuracy of prediction. The curve is obtained by plotting for *sensitivity* against  $(1 - specificity)$ . TP (True Positive) is when the people with the disease is classified as

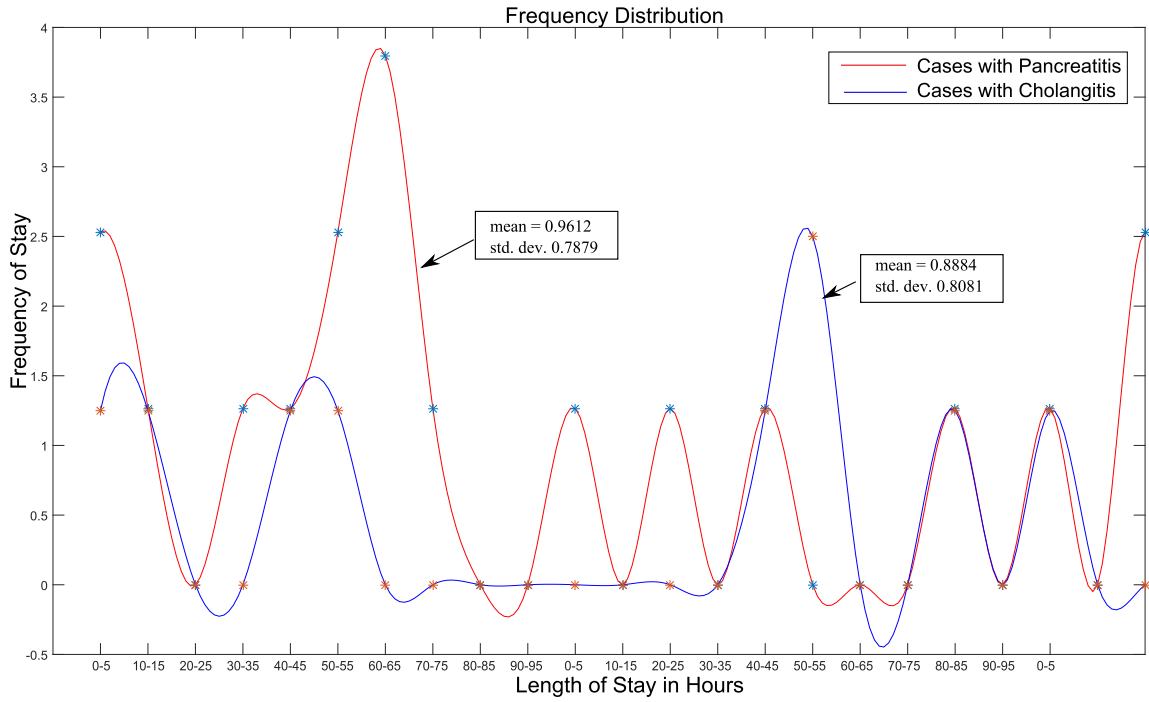


Figure 7.14: Frequency distribution of pancreatitis and cholangitis after optimizing the resource performance.

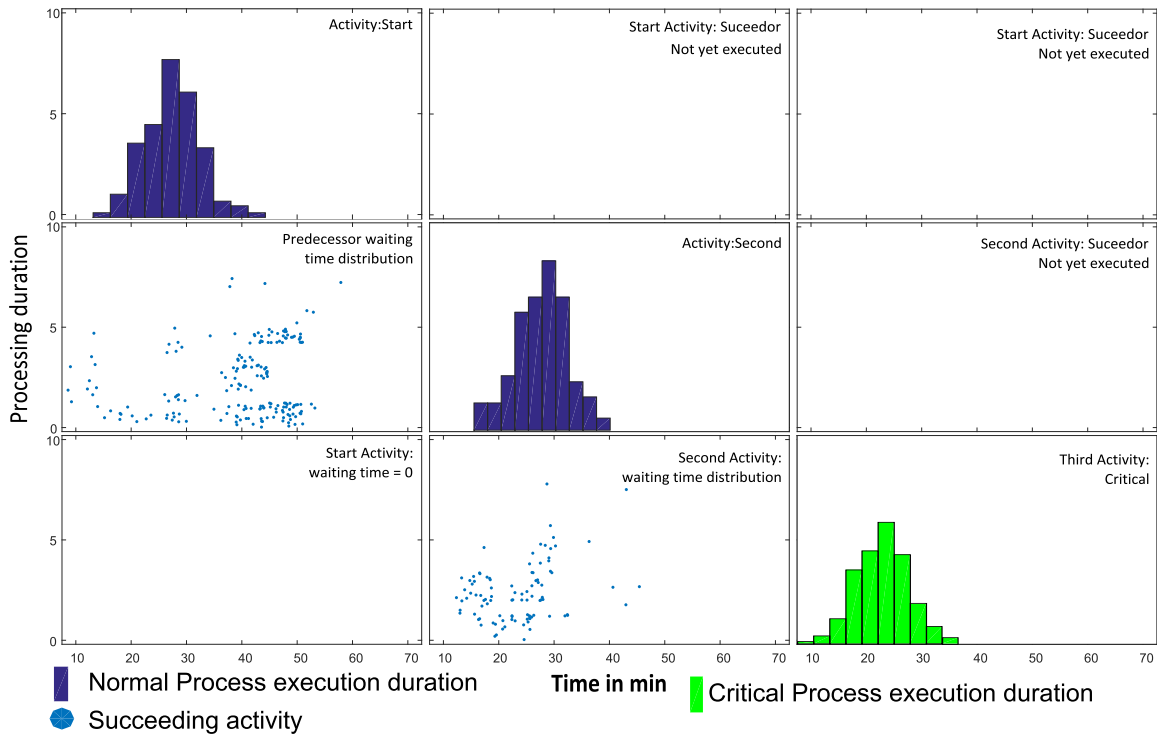


Figure 7.15: Analysing the activities for their processing and waiting time after optimizing the resource performance.

positive, and FN (False Negative) is when they are classified as negative. TN (True Negative) is when people with no disease are correctly classified as negative, and FP (False

Positive) is when they are classified positive. Sensitivity and specificity can be defined using the Table 7.4. Sensitivity and Specificity is obtained using equation 7.3 and 6.1 respectively. On plotting the obtained values for each feature, we will be able to get  $A_Z$ .

Table 7.4: Representation of TP (A), FN (B), FP (C) and TN (D)

| Test         | GSD (Yes)  | GSD (No)   | Row Total |
|--------------|------------|------------|-----------|
| Positive     | TP (A)     | FP (C)     | A + C     |
| Negative     | FN (B)     | TN (D)     | B + D     |
| <b>Total</b> | <b>A+B</b> | <b>C+D</b> |           |

$$Sensitivity = \frac{A}{A + B} \quad (7.2)$$

$$Specificity = \frac{D}{C + D} \quad (7.3)$$

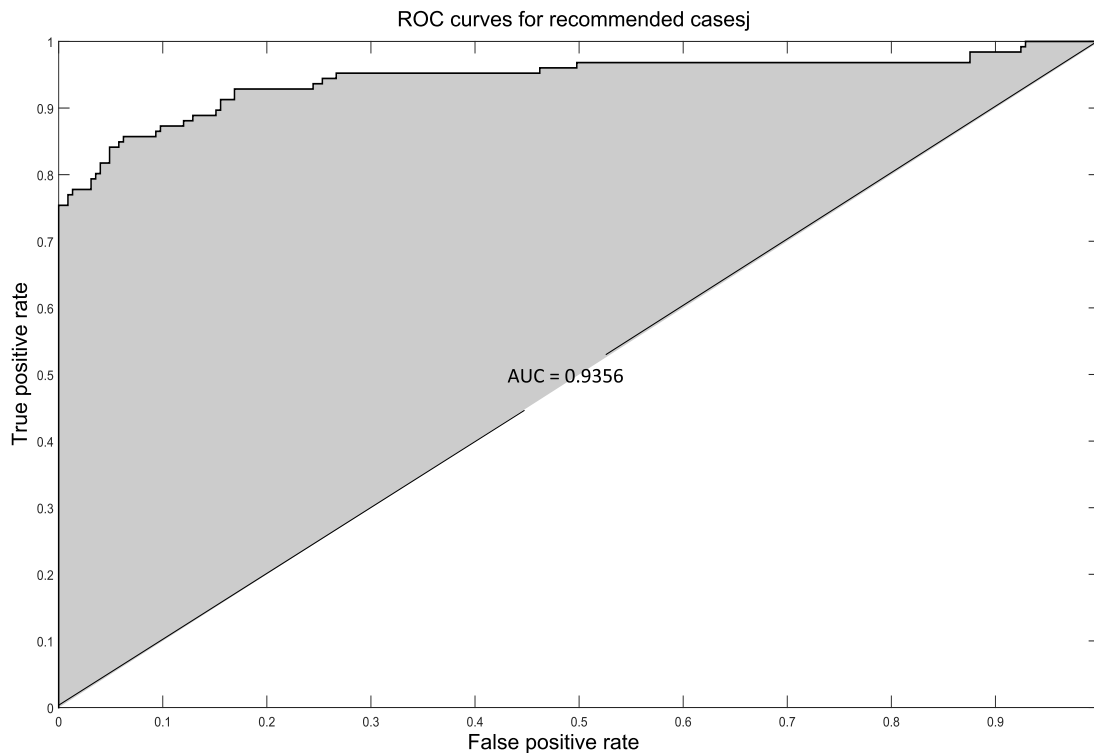


Figure 7.16: ROC showing the accuracy of recommendation.

## 7.4 Summary

The study aimed in recommending and providing proper care-flow to the patients who were found to be critical . This care-flow was achieved by providing proper treatment path and decreasing the length of stay in hospital along with the cost. The objective of whole work was to identify the critical case at the early stage and prevent them from becoming worse. With the help of EHR process mining we were able to achieve this . The experiment was tested and validated to find the accuracy of prediction.

On optimization and recommendation it was noted that the frequency rate and distribution of waiting time in predecessor and successor has come down. The detailed statistics is shown in Table 7.5.

Table 7.5: Frequency distribution comparison

|                     | Earlier     | After Optimization |
|---------------------|-------------|--------------------|
| Cholecystitis       | 14 days     | 3.5 to 4 days      |
| Choledocholithiasis | 8 - 10 days | 2.5 - 3 days       |
| Pancreatitis        | 8 - 9 days  | 3.5 - 4 days       |
| Cholangitis         | 2 - 3 days  | 1.5 - 2 days       |

Thus, by decreasing the waiting time, we were able to reduce the length of stay in hospital. By applying the technique of LCS proper recommendation was made. It was noted that recommended path of execution took 40-50% lesser time for completion when compared with conventional path of execution. The resource performance was evaluated with the concept of OEE. Accuracy of prediction and recommendation was evaluated using the concept of  $A_Z$  and it was noted that the proposed annotated transition system showed the accuracy of 93%. Hence by this approach we were able to successfully recommend proper path of execution, preventing the critical cases from becoming worse.





# Chapter 8

## Conclusions

The medical error is one of the leading cause of death, and faulty system is one of the reasons for medical error. Hence, it has become essential for building a better healthcare system. As a healthcare system is sequenced by series of clinical and non-clinical activities, it is important to streamline them and identify the critical activities. A treatment path filled with critical activities is known as a critical treatment path. A patient can be given proper care-flow in a healthcare system by analysing the disease progression and their treatment response. Care-flow in a healthcare system is treatment path sequenced by critical activities, and each activity performed by the efficient resource. The aim of this research work was to provide the quickest treatment and reduce the medical error. ModCNN was built for predicting the disease behaviour and assisting in finding the critical cases. It was designed to discover the optimal combination of hidden units and neurons. The performance of ModCNN was evaluated by comparing with ANN and CCNN and was validated using  $A_Z$ . It was noted that ModCNN outperformed other statistical models in finding the critical cases and showed the highest accuracy.

The study was focused on complicated GSD and was conducted in a retrospective way from territory care centre in north malabar, Kerala, India. 260 complicated cases were recorded during the study period and the spectrum of GSD was comparable with California study conducted by Glasgow et al. (2000). This shows that the prevalence of GSD is increasing in India.

On learning the performance of existing ANN and CCNN models, we developed ModCNN using the architecture of CCNN. In ModCNN, neurons and hidden units are adapted dynamically for giving better accuracy. ModCNN first identified the significant risk factors associated with each spectrum of GSD which were again fed into the system for predicting the disease progression. As this was a retrospective study, it was noted that ModCNN

accurately identified the 13 cases which were critical with an accuracy of  $A_Z = 0.9642$

The identified critical cases were further recommended with critical treatment path for providing proper care-flow, thereby decreasing the life threat, length of stay in the hospital, and avoiding any further complications. For achieving this, annotated transition system was built for predicting the future state and recommending the succeeding activity in the treatment path. Future state was identified using the concept of activity, transition and causal metric. They helped in reducing the waiting time and improving the performance. This shortened the patient's journey in the hospital and decreased the length of stay 40 to 50%. Further it was necessary to find the adequate resources for performing the future state activities. This was achieved by the theory of arousal and AHP. Overall, the recommendation of critical treatment path showed better accuracy when tested with the concept of  $A_Z = 93.56\%$

## 8.1 Summary of Contribution

The proposed *CDSS* can be used in routine clinical practice to predict the patients who may need immediate interventions at the time of admission and avoid any further complications. Following are the brief description of the contributions:

- *CDSS*, an information system was developed to classify the spectrum of GSD, identify the complicated cases among them and then to recommend critical treatment path to avoid any later stage complication.
- ModCNN was built for identifying the complicated cases. And to recommend the critical treatment path we developed a annotated transition system. This system identified the critical activities in the future state and the adequate resurces capable of performing those activities. The accuracy of prediction of ModCNN and recommendation of annotated transition system was evaluated using the concept of  $A_Z$ . Thus we were successful in developing a CDSS for assisting the clinicians in reducing the medical error.
- Modification to CCNN (ModCNN) was made and was further optimized by parallelizing it by using the master-slave model. In CCNN adopted the linear addition of neurons, while in ModCNN the pattern of neurons in hidden unit were identified parallelly. This made ModCNN's learning process much faster when compared to CCNN and ANN.

- ModCNN was used to identify the significant factors associated with each spectrum of GSD. This was needed since there were 32 features observed in each patient. Processing them would have taken a lot of time. Hence we first identified the significant factors associated with each spectrum and then fed them into the model to predict the disease progression. By doing so, we were able to get better accuracy in prediction.
- Thus, ModCNN accurately predicted the 13 critical cases at the time of admission itself. This would assist the clinicians in starting the right treatment and reducing the medical error.
- The identified 13 critical cases, further needed the recommendation of critical treatment path to provide proper care-flow. For this an annotated transition system was built to identify the future state of the partial trace and required resource for providing proper treatment management. The annotated transition system with the help of many techniques was able to evaluate the performance of each activity and reduce the waiting time along with the cost. Performance of the resources was evaluated to find the adequate resource capable of performing the future state critical activities.

## 8.2 Direction for Future Work

To make the system more optimal the following future works are recommended.

- The current work was a retrospective analysis. The model can be made more optimal by testing and validating with a prospective study. The challenge here is getting access to clinical data.
- The model was tested for GSD analysis. It can be generalized for taking routine clinical decisions.
- To make the system complete other parameters in healthcare system like: a number of available beds, availability of medicines, working condition of equipment and other resources which affect the treatment management along with the expertise level of the doctors can be considered.



# References

- Adams, M. A., Hosmer, A. E., Wamsteker, E. J., Anderson, M. A., Elta, G. H., Kubiliun, N. M., Kwon, R. S., Piraka, C. R., Scheiman, J. M., and Waljee, A. K. (2015). Predicting the likelihood of a persistent bile duct stone in patients with suspected cholelithiasis: accuracy of existing guidelines and the impact of laboratory trends. *Gastrointestinal endoscopy*, 82(1):88–93.
- Adams, W. T., Veale III, F. H., and Helmick, P. M. (1999). Computer imaging and workflow systems in the business office. *Healthcare Financial Management*, 53(5):48–51.
- Agrawal, P., Sachan, A., Singla, R. K., and Jain, P. (2012). Statistical analysis of medication errors in delhi, india. *Indo Global Journal of Pharmaceutical Sciences*, 2(1):88–97.
- Akande, K. O., Owolabi, T. O., Twaha, S., and Olatunji, S. O. (2014). Performance comparison of svm and ann in predicting compressive strength of concrete. *IOSR Journal of Computer Engineering*, 16(5):88–94.
- Almadi, M. A., Barkun, J. S., and Barkun, A. N. (2012). Management of suspected stones in the common bile duct. *Canadian Medical Association Journal*, 184(8):884–892.
- Almasalha, F., Xu, D., Keenan, G. M., Khokhar, A., Yao, Y., Chen, Y.-C., Johnson, A., Ansari, R., and Wilkie, D. J. (2013). Data mining nursing care plans of end-of-life patients: A study to improve healthcare decision making. *International journal of nursing knowledge*, 24(1):15–24.
- Altman, D. G. (1990). *Practical statistics for medical research*. CRC press.
- Alvi, A. R., Siddiqui, N. A., and Zafar, H. (2011). Risk factors of gallbladder cancer in karachi-a case-control study. *World journal of surgical oncology*, 9(1):164–168.

- Anderson, R. P., Jin, R., and Grunkemeier, G. L. (2003). Understanding logistic regression analysis in clinical reports: an introduction. *The Annals of thoracic surgery*, 75(3):753–757.
- Andersson, B. (2010). *Acute pancreatitis-severity classification, complications and outcome*, volume 2010. Department of Surgery, Clinical Sciences Lund, Lund University.
- Andersson, B., Andersson, R., Ohlsson, M., and Nilsson, J. (2011). Prediction of severe acute pancreatitis at admission to hospital using artificial neural networks. *Pancreatology*, 11(3):328–335.
- Anyanwu, K., Sheth, A. P., Cardoso, J., Miller, J. A., and Kochut, K. J. (2003). Healthcare enterprise process development and integration. 35(2):83–98.
- Ashizawa, K., Ishida, T., MacMahon, H., Vyborny, C. J., Katsuragawa, S., and Doi, K. (1999). Artificial neural networks in chest radiography: application to the differential diagnosis of interstitial lung disease. *Academic radiology*, 6(1):2–9.
- Attili, A., Carulli, N., Roda, E., Barbara, B., Capocaccia, L., Menotti, A., Okoliksanyi, L., Ricci, G., Capocaccia, R., Festi, D., and Lalloni, L. (1995). Epidemiology of gallstone disease in italy: prevalence data of the multicenter italian study on cholelithiasis (micol.). *American journal of epidemiology*, 141(2):158–165.
- Avery, A. A., Barber, N., Ghaleb, M., Dean Franklin, B., Armstrong, S., Crowe, S., Dhillon, S., Freyer, A., Howard, R., Pezzolesi, C., and Serumaga, B. (2012). Investigating the prevalence and causes of prescribing errors in general practice: the practice study.
- Baker, K., Dunwoodie, E., Jones, R. G., Newsham, A., Johnson, O., Price, C. P., Wolstenholme, J., Leal, J., McGinley, P., Twelves, C., and Hall, G. (2017). Process mining routinely collected electronic health records to define real-life clinical pathways during chemotherapy. *International Journal of Medical Informatics*, 103:32–41.
- Balázs, G. (2009). Cascade-correlation neural networks: A survey. *Department of Computing Science, University of Alberta, Edmonton, Canada*, pages 1–6.
- Balthazar, E. J., Robinson, D. L., Megibow, A. J., and Ranson, J. (1990). Acute pancreatitis: value of ct in establishing prognosis. *Radiology*, 174(2):331–336.

- Bartosch-Härlid, A., Andersson, B., Aho, U., Nilsson, J., and Andersson, R. (2008). Artificial neural networks in pancreatic disease. *British journal of surgery*, 95(7):817–826.
- Barwad, A., Dey, P., and Susheilia, S. (2012). Artificial neural network in diagnosis of metastatic carcinoma in effusion cytology. *Cytometry Part B: Clinical Cytometry*, 82(2):107–111.
- Basole, R. C., Braunstein, M. L., Kumar, V., Park, H., Kahng, M., Chau, D. H., Tamersoy, A., Hirsh, D. A., Serban, N., Bost, J., and Lesnick, B. (2015). Understanding variations in pediatric asthma care processes in the emergency department using visual analytics. *Journal of the American Medical Informatics Association*, 22(2):318–323.
- Bates, D. W., Cullen, D. J., Laird, N., Petersen, L. A., Small, S. D., Servi, D., Laffel, G., Sweitzer, B. J., Shea, B. F., Hallisey, R., and Vander Vliet, M. (1995). Incidence of adverse drug events and potential adverse drug events: implications for prevention. *Jama*, 274(1):29–34.
- Bathen, T. F., Jensen, L. R., Sitter, B., Fjösne, H. E., Halgunset, J., Axelson, D. E., Gribbestad, I. S., and Lundgren, S. (2007). Mr-determined metabolic phenotype of breast cancer in prediction of lymphatic spread, grade, and hormone status. *Breast cancer research and treatment*, 104(2):181–189.
- Beger, H. G. and Rau, B. M. (2007). Severe acute pancreatitis: clinical course and management. *World journal of gastroenterology: WJG*, 13(38):5043–5051.
- Benneyan, J. C. (2001). Number-between g-type statistical quality control charts for monitoring adverse events. *Health care management science*, 4(4):305–318.
- Black, A. D., Car, J., Pagliari, C., Anandan, C., Cresswell, K., Bokun, T., McKinstry, B., Procter, R., Majeed, A., and Sheikh, A. (2011). The impact of ehealth on the quality and safety of health care: a systematic overview. *PLoS medicine*, 8(1):1–16.
- Blackstone, S. and Taylor, A. (2012). These are the 36 countries that have better health-care systems than the us. <http://www.businessinsider.com/best-healthcare-systems-in-the-world-2012-6/op=1records/page=2>.(Accessed on Aug. 28, 2017).

- Blahuta, J., Soukup, T., Čermák, P., Rozsypal, J., and Večerek, M. (2012). Ultrasound medical image recognition with artificial intelligence for parkinson’s disease classification. In *MIPRO, 2012 Proceedings of the 35th International Convention*, pages 958–962. IEEE.
- Blaser, R., Schnabel, M., Biber, C., Bäumlein, M., Heger, O., Beyer, M., Opitz, E., Lenz, R., and Kuhn, K. A. (2007). Improving pathway compliance and clinician performance by using information technology. *International journal of medical informatics*, 76(2):151–156.
- Blum, T., Maisonneuve, P., Lowenfels, A. B., and Lankisch, P. G. (2001). Fatal outcome in acute pancreatitis: its occurrence and early prediction. *Pancreatology*, 1(3):237–241.
- Bose, R. J. C., van der Aalst, W. M., Žliobaitė, I., and Pechenizkiy, M. (2011). Handling concept drift in process mining. In *International Conference on Advanced Information Systems Engineering*, pages 391–405. Springer.
- Bostock, M., Ogievetsky, V., and Heer, J. (2011). D<sup>3</sup> data-driven documents. *IEEE transactions on visualization and computer graphics*, 17(12):2301–2309.
- Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *Proceedings of the 19th International Conference on Computational Statistics (COMPSTAT)*, pages 177–186. Springer.
- Bowers, S. (2013). Computer systems contractor csc set to pay shareholders \$97.5m. <https://www.theguardian.com/society/2013/sep/18/csc-courts-sign-off-payment-shareholders>.(Accessed on Aug. 28, 2017).
- Bruin, J. (2011). Newtest: command to compute new test @ONLINE. <http://www.ats.ucla.edu/stat/stata/ado/analysis/>.(Accessed on Sep. 01, 2017).
- Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., and Hullender, G. (2005). Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96. ACM.
- Burke, H. B., Rosen, D. B., and Goodman, P. H. (1994). Comparing artificial neural networks to other statistical methods for medical outcome prediction. In *Neural Networks, 1994. IEEE World Congress on Computational Intelligence., 1994 IEEE International Conference on*, volume 4, pages 2213–2216. IEEE.



- Catalogna, M., Cohen, E., Fishman, S., Halpern, Z., Nevo, U., and Ben-Jacob, E. (2012). Artificial neural networks based controller for glucose monitoring during clamp test. *PloS one*, 7(8):1–10.
- Cetta, F., Lombardo, F., Giubbolini, M., Baldi, C., Cariati, A., Diehl, A. K., Schwesinger, W. H., and Kurtin, W. E. (1995). Classification of gallstones and epidemiologic studies. *Digestive diseases and sciences*, 40(10):2189–2191.
- Chan, H.-P., Sahiner, B., Petrick, N., Helvie, M. A., Lam, K. L., Adler, D. D., and Goodsitt, M. M. (1997). Computerized classification of malignant and benign microcalcifications on mammograms: texture analysis using an artificial neural network. *Physics in medicine and biology*, 42(3):549–567.
- Chandra, B. and Varghese, P. P. (2007). Applications of cascade correlation neural networks for cipher system identification. *World Academy of Science, engineering and technology*, 26:312–314.
- Chang, C.-L. and Hsu, M.-Y. (2009). The study that applies artificial intelligence and logistic regression for assistance in differential diagnostic of pancreatic cancer. *Expert Systems with applications*, 36(7):10663–10672.
- Chazard, E., P. C. M. B. F. G. and Beuscart, R. (2009). Data-mining-based detection of adverse drug events. In *MIE*, pages 552–556.
- Chen, E. S. and Sarkar, I. N. (2014). Mining the electronic health record for disease knowledge. *Biomedical Literature Mining*, pages 269–286.
- Chu, C., Kim, S. K., Lin, Y.-A., Yu, Y., Bradski, G., Ng, A. Y., and Olukotun, K. (2007). Map-reduce for machine learning on multicore. *Advances in neural information processing systems*, 19:281–288.
- Chudova, D., Dolenko, S., Orlov, Y. V., Pavlov, D. Y., and Persiantsev, I. (1998). Benchmarking of different modifications of the cascade correlation algorithm. In *Adaptive Computing in Design and Manufacture*, pages 339–344. Springer.
- Chvátal, V., Klarner, D. A., and Knuth, D. E. (1972). *Selected combinatorial research problems*. Computer Science Department, Stanford University.

- Clayton, E., Connor, S., Alexakis, N., and Leandros, E. (2006). Meta-analysis of endoscopy and surgery versus surgery alone for common bile duct stones with the gallbladder in situ. *British Journal of Surgery*, 93(10):1185–1191.
- Creasy, J. M., Goldman, D. A., Gonen, M., Dudeja, V., Askan, G., Basturk, O., Balachandran, V. P., Allen, P. J., DeMatteo, R. P., D’Angelica, M. I., and Jarnagin, W. R. (2017). Predicting residual disease in incidental gallbladder cancer: Risk stratification for modified treatment strategies. *Journal of Gastrointestinal Surgery*, pages 1–8.
- Cummings, J. M., Boullier, J. A., Izenberg, S. D., Kitchens, D. M., and Kothandapani, R. V. (2000). Prediction of spontaneous ureteral calculous passage by an artificial neural network. *The Journal of urology*, 164(2):326–328.
- Cunningham, P., Carney, J., and Jacob, S. (2000). Stability problems with artificial neural networks and the ensemble solution. *Artificial Intelligence in medicine*, 20(3):217–225.
- Dadam, P., Reichert, M., and Kuhn, K. (2000). Clinical workflows—the killer application for process-oriented information systems? In *Proceedings of the 4th International Conference on Business Information Systems*, pages 36–59. Springer.
- Das, A., Nguyen, C. C., Li, F., and Li, B. (2008). Digital image analysis of eus images accurately differentiates pancreatic cancer from chronic pancreatitis and normal tissue. *Gastrointestinal endoscopy*, 67(6):861–867.
- Davidoff, F., Case, K., and Fried, P. W. (1995). Evidence-based medicine: why all the fuss? *Annals of Internal Medicine*, 122(9):727–727.
- Delias, P., Doumpos, M., Manolitzas, P., Grigoroudis, E., and Matsatsinis, N. (2013). Clustering healthcare processes with a robust approach. In *26th European Conference on Operational Research*.
- Dey, P., Lamba, A., Kumari, S., and Marwaha, N. (2011). Application of an artificial neural network in the prognosis of chronic myeloid leukemia. *Analytical and quantitative cytology and histology/the International Academy of Cytology [and] American Society of Cytology*, 33(6):335–339.
- Diamantopoulou, M. J., Antonopoulos, V. Z., and Papamichail, D. M. (2007). Cascade correlation artificial neural networks for estimating missing monthly values of water quality parameters in rivers. *Water resources management*, 21(3):649–662.

- Diamantopoulou, M. J., Papamichail, D. M., and Antonopoulos, V. Z. (2005). The use of a neural network technique for the prediction of water quality parameters. *Operational Research*, 5(1):115–125.
- Dobchev, D. and Karelson, M. (2016). Have artificial neural networks met expectations in drug discovery as implemented in qsar framework? *Expert opinion on drug discovery*, 11(7):627–639.
- Doering, A., Galicki, M., and Witte, H. (1997). Admissibility and optimality of the cascade-correlation algorithm. *Artificial Neural Networks—ICANN’97*, pages 505–510.
- Donaldson, M. S., Corrigan, J. M., Kohn, L. T., et al. (2000). *To err is human: building a safer health system*, volume 6. National Academies Press.
- Draper, N. R. and Smith, H. (2014). *Applied regression analysis*. John Wiley & Sons.
- Dumas, M., Van der Aalst, W. M., and Ter Hofstede, A. H. (2005). *Process-aware information systems: bridging people and software through process technology*. John Wiley & Sons.
- Eldar, R. (2002). Understanding and preventing adverse events. *Croatian medical journal*, 43(1):86–88.
- Eldar, S., Siegelmann, H. T., Buzaglo, D., Matter, I., Cohen, A., Sabo, E., and Abrahamson, J. (2002). Conversion of laparoscopic cholecystectomy to open cholecystectomy in acute cholecystitis: artificial neural networks improve the prediction of conversion. *World journal of surgery*, 26(1):79–85.
- Elveren, E. and Yumuşak, N. (2011). Tuberculosis disease diagnosis using artificial neural network trained with genetic algorithm. *Journal of medical systems*, 35(3):329–332.
- Fahlman, S. E. and Lebiere, C. (1990). The cascade-correlation learning architecture. In *Advances in neural information processing systems*, pages 524–532.
- Fanjiang, G., Grossman, J. H., Compton, W. D., and Reid, P. P. (2005). *Building a better delivery system: a new engineering/health care partnership*. National Academies Press.
- Fish, J. M. (2001). Human error in medicine: promise and pitfalls, part 2. *Annals of emergency medicine*, 37(4):419–420.

- Focsa, M. (2010). Knowledge-based ehr systems. In *Proceedings of the 31st Romanian National Conference on Medical Informatics "Solution-based Medical Informatics*, pages 64–68.
- Fukushima, K. (1975). Cognitron: A self-organizing multilayered neural network. *Biological cybernetics*, 20(3-4):121–136.
- Garg, P. K. (2013). *Chronic Pancreatitis-ECAB*. Elsevier Health Sciences.
- Gautham, M., sara George, P., and Mathew, A. (2011). Trends in incidence of gallbladder cancer–indian scenario. (9):1–9.
- Geng, L., Sun, C., and Bai, J. (2013). Single incision versus conventional laparoscopic cholecystectomy outcomes: a meta-analysis of randomized controlled trials. *PLoS One*, 8(10):1–10.
- Ghemawat, S., Gobioff, H., and Leung, S.-T. (2003). The google file system. In *ACM SIGOPS operating systems review*, volume 37, pages 29–43. ACM.
- Glasgow, R. E., Cho, M., Hutter, M. M., and Mulvihill, S. J. (2000). The spectrum and cost of complicated gallstone disease in california. *Archives of Surgery*, 135(9):1021–1025.
- Golub, R., Cantu, R., and Tan, M. (1998). The prediction of common bile duct stones using a neural network. *Journal of the American College of Surgeons*, 187(6):584–590.
- Government of India, Ministry of Health, F. W. (2013). Electronic health record standards for india. <http://snomedctnrc.in/downloads/EHR-Standards-for-India%20-August2013-32630521.pdf>.(Accessed on Sep. 02, 2017).
- Grana, M. and Jackowski, K. (2015). Electronic health record: a review. In *Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on*, pages 1375–1382. IEEE.
- Group, I. P. F. S. (2016). Prognostic factors in patients with metastatic germ cell tumors who experienced treatment failure with cisplatin-based first-line chemotherapy. *Journal of Clinical Oncology*, 28(33):4906–4911.
- Gupta, S. (2007). Workflow and process mining in healthcare. *Master's Thesis, Technische Universiteit Eindhoven*.

- Gurusamy, K., Samraj, K., Gluud, C., Wilson, E., and Davidson, B. (2010). Meta-analysis of randomized controlled trials on the safety and effectiveness of early versus delayed laparoscopic cholecystectomy for acute cholecystitis. *British journal of surgery*, 97(2):141–150.
- Güven, A. and Kara, S. (2006). Diagnosis of the macular diseases from pattern electroretinography signals using artificial neural networks. *Expert Systems with Applications*, 30(2):361–366.
- Halonen, K. I., Leppäniemi, A. K., Lundin, J. E., Puolakkainen, P. A., Kemppainen, E. A., and Haapiainen, R. K. (2003). Predicting fatal outcome in the early phase of severe acute pancreatitis by using novel prognostic models. *Pancreatology*, 3(4):309–315.
- Hatch, D. (2001). Incidence and acceptance of errors in medicine. *Bollettino dei medici svizzeri*, 82(25):1339–43.
- Helm, E. and Paster, F. (2015). First steps towards process mining in distributed health information systems. *International Journal of Electronics and Telecommunications*, 61(2):137–142.
- Hirayama, M., Kawato, M., and Jordan, M. I. (1993). The cascade neural network model and a speed-accuracy trade-off of arm movement. *Journal of motor behavior*, 25(3):162–174.
- Hirschberg, D. S. (1975). A linear space algorithm for computing maximal common subsequences. *Communications of the ACM*, 18(6):341–343.
- Hong, W.-d., Chen, X.-r., Jin, S.-q., Huang, Q.-k., Zhu, Q.-h., and Pan, J.-y. (2013). Use of an artificial neural network to predict persistent organ failure in patients with acute pancreatitis. *Clinics*, 68(1):27–31.
- Hosmer Jr, D. W., Lemeshow, S., and Sturdivant, R. X. (2013). *Applied logistic regression*, volume 398. John Wiley & Sons.
- Hundal, R. and Shaffer, E. A. (2014). Gallbladder cancer: epidemiology and outcome. *Clin Epidemiol*, 6(6):99–109.
- Hunt, J. W. and Szymanski, T. G. (1977). A fast algorithm for computing longest common subsequences. *Communications of the ACM*, 20(5):350–353.

- Hunter, B. and Segrott, J. (2008). Re-mapping client journeys and professional identities: A review of the literature on clinical pathways. *International journal of nursing studies*, 45(4):608–625.
- Ikeda, M., Ito, S., Ishigaki, T., and Yamauchi, K. (1997). Evaluation of a neural network classifier for pancreatic masses based on ct findings. *Computerized medical imaging and graphics*, 21(3):175–183.
- Imrie, C., Benjamin, I., Ferguson, J., McKay, A., Mackenzie, I., O’neill, J., and Blumgart, L. (1978). A single-centre double-blind trial of trasyolol therapy in primary acute pancreatitis. *British journal of surgery*, 65(5):337–341.
- Itchhaporia, D., Snow, P. B., Almassy, R. J., and Oetgen, W. J. (1996). Artificial neural networks: current status in cardiovascular medicine. *Journal of the American College of Cardiology*, 28(2):515–521.
- Jalloh, O. B. and Waitman, L. R. (2006). Improving computerized provider order entry (cpoe) usability by data mining users’ queries from access logs. In *AMIA Annual Symposium Proceedings*, volume 2006, page 379. American Medical Informatics Association.
- Jensen, P. B., Jensen, L. J., and Brunak, S. (2012). Mining electronic health records: towards better research applications and clinical care. *Nature Reviews Genetics*, 13(6):395–405.
- Jha, A. K., Larizgoitia, I., Audera-Lopez, C., Prasopa-Plaizier, N., Waters, H., and Bates, D. W. (2013). The global burden of unsafe medical care: analytic modelling of observational studies. *BMJ Qual Saf*, pages bmjqs–2012.
- Johnson, C. and Abu-Hilal, M. (2004). Persistent organ failure during the first week as a marker of fatal outcome in acute pancreatitis. *Gut*, 53(9):1340–1344.
- Jonnagaddala, J., Dai, H.-J., Ray, P., and Liaw, S.-T. (2017). Mining electronic health records to guide and support clinical decision support systems. In *Healthcare Ethics and Training: Concepts, Methodologies, Tools, and Applications*, pages 184–201. IGI Global.
- Jovanovic, P., Salkic, N. N., and Zerem, E. (2014). Artificial neural network predicts the need for therapeutic ercp in patients with suspected choledocholithiasis. *Gastrointestinal endoscopy*, 80(2):260–268.

- Jovanović, P., Salkić, N. N., Zerem, E., and Ljuca, F. (2011). Biochemical and ultrasound parameters may help predict the need for therapeutic endoscopic retrograde cholangiopancreatography (ercp) in patients with a firm clinical and biochemical suspicion for choledocholithiasis. *European journal of internal medicine*, 22(6):110–114.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45.
- Kaplan, R. S. and Anderson, S. R. (2003). Time-driven activity-based costing. *Journal of Cost Management*, 21(2):16–20.
- Kapoor, V. (2006). Cholecystectomy in patients with asymptomatic gallstones to prevent gall bladder cancer—the case against. *Indian J Gastroenterol*, 25:152–154.
- Karabulut, E. M. and İbrikçi, T. (2012). Effective diagnosis of coronary artery disease using the rotation forest ensemble method. *Journal of medical systems*, 36(5):3011–3018.
- Karthikeyan, N. and Sukanesh, R. (2012). Cloud based emergency health care information service in india. *Journal of medical systems*, 36(6):4031–4036.
- Kaymak, U., Mans, R., van de Steeg, T., and Dierks, M. (2012). On process mining in health care. In *Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on*, pages 1859–1864. IEEE.
- Keogan, M. T., Lo, J. Y., Freed, K. S., Raptopoulos, V., Blake, S., Kamel, I. R., Weisinger, K., Rosen, M. P., and Nelson, R. C. (2002). Outcome analysis of patients with acute pancreatitis by using an artificial neural network. *Academic radiology*, 9(4):410–419.
- Khoja, T., Neyaz, Y., Qureshi, N., Magzoub, M., Haycox, A., and Walley, T. (2011). Medication errors in primary care in riyadh city, saudi arabia. *Eastern Mediterranean Health Journal*, 17(2):156–159.
- Kim, E., Kim, S., Song, M., Kim, S., Yoo, D., Hwang, H., and Yoo, S. (2013). Discovery of outpatient care process of a tertiary university hospital using process mining. *Healthcare informatics research*, 19(1):42–49.

- Kim, J. A., Cho, I., and Kim, Y. (2008). Cdss (clinical decision support system) architecture in korea. In *Convergence and Hybrid Information Technology, 2008. ICHIT'08. International Conference on*, pages 700–703. IEEE.
- Kim, S. S., Lee, J. G., Kim, D. W., Kim, B. H., Jeon, Y. K., Kim, M. R., Huh, J. E., Mok, J. Y., Kim, S.-J., Kim, Y. K., , and Kim, J. (2011). Insulin resistance as a risk factor for gallbladder stone formation in korean postmenopausal women. *The Korean journal of internal medicine*, 26(3):285–293.
- Kira, K. and Rendell, L. A. (1992). The feature selection problem: Traditional methods and a new algorithm. In *Aaai*, volume 2, pages 129–134.
- Klösgen, W. and Zytchow, J. M. (2002). Knowledge discovery in databases: the purpose, necessity, and challenges. In *Handbook of data mining and knowledge discovery*, pages 1–9. Oxford University Press, Inc.
- Knaus, W. A., Draper, E. A., Wagner, D. P., and Zimmerman, J. E. (1985). Apache ii: a severity of disease classification system. *Critical care medicine*, 13(10):818–829.
- Kong, G., Xu, D.-L., and Yang, J.-B. (2008). Clinical decision support systems: a review on knowledge representation and inference under uncertainties. *International Journal of Computational Intelligence Systems*, 1(2):99.
- Kumar, V., Park, H., Basole, R. C., Braunstein, M., Kahng, M., Chau, D. H., Tamersoy, A., Hirsh, D. A., Serban, N., Bost, J., and B, L. (2014). Exploring clinical care processes using visual and data analytics: challenges and opportunities. In *Proceedings of the 20th ACM SIGKDD conference on knowledge discovery and data mining workshop on data science for social good*.
- Kumar Sangwan, M., Sangwan, V., kumar Garg, M., Singla, D., Thami, G., and Malik, P. (2016). Gallstone disease menacing rural population in north india: a retrospective study of 576 cases in a rural hospital. *International Surgery Journal*, 2(4):487–491.
- Kumarasinghe, G., Lavee, O., Parker, A., Nivison-Smith, I., Milliken, S., Dodds, A., Joseph, J., Fay, K., Ma, D. D., Malouf, M., and Plit, M. (2015). Post-transplant lymphoproliferative disease in heart and lung transplantation: defining risk and prognostic factors. *The Journal of Heart and Lung Transplantation*, 34(11):1406–1414.



- Lam, S. S. and Smith, A. E. (1998). Cascade-correlation neural network modeling of the abrasive flow machining process. *University of Pittsburgh*.
- Larsson, S. and Wolk, A. (2007). Obesity and the risk of gallbladder cancer: a meta-analysis. *British journal of cancer*, 96(9):1457–1461.
- Leape, L. L. (1994). Error in medicine. *Jama*, 272(23):1851–1857.
- Leape, L. L. (1997). A systems analysis approach to medical error. *Journal of evaluation in clinical practice*, 3(3):213–222.
- Leape, L. L., Bates, D. W., Cullen, D. J., Cooper, J., Demonaco, H. J., Gallivan, T., Hallisey, R., Ives, J., Laird, N., Laffel, G., and Nemeskal, R. (1995). Systems analysis of adverse drug events. *Jama*, 274(1):35–43.
- Leape, L. L., Brennan, T. A., Laird, N., Lawthers, A. G., Localio, A. R., Barnes, B. A., Hebert, L., Newhouse, J. P., Weiler, P. C., and Hiatt, H. (1991). The nature of adverse events in hospitalized patients: results of the harvard medical practice study ii. *New England journal of medicine*, 324(6):377–384.
- Leape, L. L., Lawthers, A. G., Brennan, T. A., and Johnson, W. G. (1993). Preventing medical injury. *QRB-Quality Review Bulletin*, 19(5):144–149.
- Leapfrog, G. (2013). Hospital errors are the third leading cause of death in u.s., and new hospital safety scores show improvements are too slow. <http://www.hospitalsafetygrade.org/newsroom/display/hospitalerrors-thirdleading-causeofdeathinus-improvementstooslow>. (Accessed on Aug. 28, 2017).
- LeLorier, J., Gregoire, G., Benhaddad, A., Lapierre, J., and Derderian, F. (1997). Discrepancies between meta-analyses and subsequent large randomized, controlled trials. *New England Journal of Medicine*, 337(8):536–542.
- Lenz, R., Elstner, T., Siegele, H., and Kuhn, K. A. (2002). A practical approach to process support in health information systems. *Journal of the American Medical Informatics Association*, 9(6):571–585.
- Lenz, R. and Reichert, M. (2007). It support for healthcare processes—premises, challenges, perspectives. *Data & Knowledge Engineering*, 61(1):39–58.

- Li, D., Simon, G., Chute, C. G., and Pathak, J. (2013). Using association rule mining for phenotype extraction from electronic health records. *AMIA Summits on Translational Science Proceedings*, 2013:142.
- Lian, J., Ma, Y., Ma, Y., Shi, B., Liu, J., Yang, Z., and Guo, Y. (2017). Automatic gallbladder and gallstone regions segmentation in ultrasound image. *International Journal of Computer Assisted Radiology and Surgery*, pages 1–16.
- Liang, B. A. and Storti, K. (1999). Creating problems as part of the "solution": the jcaho sentinel event policy, legal issues, and patient safety. *Journal of health law*, 33(2):263–285.
- Lin, Y., Jenness, J., and Huang, X. (2006). Parameterized computation of lcs for two sequences. In *BIOCOMP*, pages 31–40.
- Lippmann, R. P. and Shahian, D. M. (1997). Coronary artery bypass risk prediction using neural networks. *The Annals of thoracic surgery*, 63(6):1635–1643.
- Ma, H. (2007). Process-aware information systems: Bridging people and software through process technology. *Journal of the Association for Information Science and Technology*, 58(3):455–456.
- MacDougall, C. E., McGregor, C., and Percival, J. (2011). The fusion of clinical guidelines with technology: trends & challenges. *electronic Journal of Health Informatics*, 5(2):14.
- Mans, R. R. (2011). *Workflow support for the healthcare domain*. PhD thesis, Technische Universiteit Eindhoven.
- Mans, R. S., Schonenberg, M., Song, M., Aalst, W., and Bakker, P. J. (2009). Application of process mining in healthcare—a case study in a dutch hospital. *Biomedical Engineering Systems and Technologies*, pages 425–438.
- Mans, R. S., van der Aalst, W. M., Vanwersch, R. J., and Moleman, A. J. (2013). Process mining in healthcare: Data challenges when answering frequently posed questions. In *Process Support and Knowledge Representation in Health Care*, pages 140–153. Springer.

- Mantzaris, D., Anastassopoulos, G., and Adamopoulos, A. (2011). Genetic algorithm pruning of probabilistic neural networks in medical disease estimation. *Neural Networks*, 24(8):831–835.
- Mantzaris, D., Anastassopoulos, G., Iliadis, L., Tsalkidis, A., and Adamopoulos, A. (2010). A probabilistic neural network for assessment of the vesicoureteral reflux’s diagnostic factors validity. *Artificial Neural Networks–ICANN 2010*, pages 241–250.
- Marshall, J. C., Cook, D. J., Christou, N. V., Bernard, G. R., Sprung, C. L., and Sibbald, W. J. (1995). Multiple organ dysfunction score: a reliable descriptor of a complex clinical outcome. *Critical care medicine*, 23(10):1638–1652.
- Marshall, R. J. (2001). The use of classification and regression trees in clinical epidemiology. *Journal of clinical epidemiology*, 54(6):603–609.
- McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.
- McDonald, C. J. (1972). Regenstrief medical record system (rmrs). <https://lhncbc.nlm.nih.gov/director/>.(Accessed on Aug. 28, 2017).
- McGregor, C., Catley, C., and James, A. (2011). A process mining driven framework for clinical guideline improvement in critical care. In *Proceedings of the Learning from Medical Data Streams Workshop. Bled, Slovenia (July 2011)*.
- McMahon, M. J., Playforth, M. J., and Pickford, I. R. (1980). A comparative study of methods for the prediction of severity of attacks of acute pancreatitis. *British Journal of Surgery*, 67(1):22–25.
- Mertz, L. (2014). Saving lives and money with smarter hospitals: Streaming analytics, other new tech help to balance costs and benefits. *IEEE pulse*, 5(6):33–36.
- Mofidi, R., Duff, M. D., Madhavan, K. K., Garden, O. J., and Parks, R. W. (2007). Identification of severe acute pancreatitis using an artificial neural network. *Surgery*, 141(1):59–66.
- Mønsted Shabanzadeh, D., Tue Sørensen, L., and Jørgensen, T. (2016). Abdominal symptoms and incident gallstones in a population unaware of gallstone status. *Canadian Journal of Gastroenterology and Hepatology*, 2016:1–7.

- Montgomery, D. C., Peck, E. A., and Vining, G. G. (2015). *Introduction to linear regression analysis*. John Wiley & Sons.
- Moser, R. H. (1956). Diseases of medical progress. *New England Journal of Medicine*, 255(13):606–614.
- Muhammet, G. and Guneri, A. F. (2015). Forecasting patient length of stay in an emergency department by artificial neural networks. *Journal of Aeronautics and Space Technologies*, 8(2):43–48.
- Muszynska, C., Lundgren, L., Lindell, G., Andersson, R., Nilsson, J., Sandström, P., and Andersson, B. (2017). Predictors of incidental gallbladder cancer in patients undergoing cholecystectomy for benign gallbladder disease: Results from a population-based gallstone surgery registry. *Surgery*, 162(2):256–263.
- Nair, S. (2007). Clinical decision support systems. <https://www.slideshare.net/snair/clinical-decision-support-systems-presentation>.(Accessed on Aug. 30, 2017).
- Nakatumba, J. and Van der Aalst, W. M. (2009). Analyzing resource behavior using process mining. In *International Conference on Business Process Management*, pages 69–80. Springer.
- Nasiri, S., Dornhöfer, M., and Fathi, M. (2013). Improving ehr and patient empowerment based on dynamic knowledge assets. In *GI-Jahrestagung*, pages 402–413.
- Obenshain, M. K. (2004). Application of data mining techniques to healthcare data. *Infection Control & Hospital Epidemiology*, 25(8):690–695.
- Ohno-Machado, L. (1996). *Medical applications of artificial neural networks: connectionist models of survival*. PhD thesis, Stanford University.
- Olthoff, K. M., Emond, J. C., Shearon, T. H., Everson, G., Baker, T. B., Fisher, R. A., Freise, C. E., Gillespie, B. W., and Everhart, J. E. (2015). Liver regeneration after living donor transplantation: Adult-to-adult living donor liver transplantation cohort study. *Liver Transplantation*, 21(1):79–88.
- Opačić, D., Rustemović, N., Kalauz, M., Markoš, P., Ostojić, Z., Majerović, M., Ledinsky, I., Višnjić, A., Krznarić, J., and Opačić, M. (2015). Endoscopic ultrasound elastography

- strain histograms in the evaluation of patients with pancreatic masses. *World Journal of Gastroenterology: WJG*, 21(13):4014.
- Partington, A., Wynn, M., Suriadi, S., Ouyang, C., and Karnon, J. (2015). Process mining for clinical processes: a comparative analysis of four australian hospitals. *ACM Transactions on Management Information Systems (TMIS)*, 5(4):19.
- Pauwels, R. A., Buist, A. S., Calverley, P. M., Jenkins, C. R., and Hurd, S. S. (2001). Chronic obstructive pulmonary disease-global strategy for the diagnosis, management, and prevention of chronic obstructive pulmonary disease. *American Journal of Respiratory and Critical Care Medicine*, 163(5):1256.
- Pearson, K. (1896). Mathematical contributions to the theory of evolution.—on a form of spurious correlation which may arise when indices are used in the measurement of organs. *Proceedings of the royal society of london*, 60(359-367):489–498.
- Peleg, M. (2013). Computer-interpretable clinical guidelines: a methodological review. *Journal of biomedical informatics*, 46(4):744–763.
- Perimal-Lewis, L., Qin, S., Thompson, C., and Hakendorf, P. (2012). Gaining insight from patient journey data using a process-oriented analysis approach. In *Proceedings of the Fifth Australasian Workshop on Health Informatics and Knowledge Management-Volume 129*, pages 59–66. Australian Computer Society, Inc.
- Plackett, R. L. (1958). Studies in the history of probability and statistics: Vii. the principle of the arithmetic mean. *Biometrika*, pages 130–135.
- Poelmans, J., Dedene, G., Verheyden, G., Van Der Musselle, H., Viaene, S., and Peters, E. (2010). Combining business process and data discovery techniques for analyzing and improving integrated care pathways. In *Industrial Conference on Data Mining*, pages 505–517. Springer.
- Pronovost, P. and Vohr, E. (2010). *Safe patients, smart hospitals: how one doctor’s checklist can help us change health care from the inside out*. Penguin.
- Quaglioni, S. (2009). Process mining in healthcare: a contribution to change the culture of blame. In *Business Process Management Workshops*, pages 308–311. Springer.

- Ranson, J., Rifkind, K., Roses, D., Fink, S., Eng, K., and Spencer, F. (1974). Prognostic signs and the role of operative management in acute pancreatitis. *Surgery, gynecology & obstetrics*, 139(1):69–81.
- Reason, J. (2000). Human error: models and management. *BMJ: British Medical Journal*, 320(7237):768–770.
- Reason, J., Carthey, J., and De Leval, M. (2001a). Diagnosing “vulnerable system syndrome”: an essential prerequisite to effective risk management. *Quality and safety in health care*, 10(2):21–25.
- Reason, J., Carthey, J., and De Leval, M. (2001b). Diagnosing “vulnerable system syndrome”: an essential prerequisite to effective risk management. *Quality and safety in health care*, 10(2):21–25.
- Rebuge, Á. and Ferreira, D. R. (2012). Business process analysis in healthcare environments: A methodology based on process mining. *Information systems*, 37(2):99–116.
- Renner, P. (2009). Why most emr implementations fail: How to protect your practice and enjoy successfully implementation. [http://www.emrindustry.com/wp-content/uploads/2014/04/StreamlineMD\\_WhitePaper\\_1B.pdf](http://www.emrindustry.com/wp-content/uploads/2014/04/StreamlineMD_WhitePaper_1B.pdf). (Accessed on Aug. 28, 2017).
- Riha, A., Svátek, V., Nemeč, P., and Zvárová, J. Medical guideline as prior knowledge in electronic healthcare record mining. *WIT Transactions on Information and Communication Technologies*, 28:809–818.
- Rosenthal, D. I. (2013). Instant replay. In *Healthcare*, volume 1, pages 52–54. Elsevier.
- Rovani, M., Maggi, F. M., de Leoni, M., and van der Aalst, W. M. (2015). Declarative process mining in healthcare. *Expert Systems with Applications*, 42(23):9236–9251.
- Ryu, S., Chang, Y., Yun, K. E., Jung, H.-S., Shin, J. H., and Shin, H. (2016). Gallstones and the risk of gallbladder cancer mortality: A cohort study. *The American journal of gastroenterology*, 111(10):1476–1487.
- Saaty, T. L. (1970). *Optimization in integers and related extremal problems*. McGraw-Hill.
- Saaty, T. L. (2008). Decision making with the analytic hierarchy process. *International journal of services sciences*, 1(1):83–98.

- Saghiri, M., Asgar, K., Boukani, K., Lotfi, M., Aghili, H., Delvarani, A., Karamifar, K., Saghiri, A., Mehrvarzfar, P., and Garcia-Godoy, F. (2012). A new approach for locating the minor apical foramen using an artificial neural network. *International endodontic journal*, 45(3):257–265.
- Saxena, S. and Burse, K. (2012). A survey on neural network techniques for classification of breast cancer data. *International Journal of Engineering and Advanced Technology*, 2(1):234–237.
- Schuld, J., Schäfer, T., Nickel, S., Jacob, P., Schilling, M. K., and Richter, S. (2011). Impact of it-supported clinical pathways on medical staff satisfaction. a prospective longitudinal cohort study. *International journal of medical informatics*, 80(3):151–156.
- Sharma, M. and Aggarwal, H. (2016). Ehr adoption in india: Potential and the challenges. *Indian Journal of Science and Technology*, 9(34).
- Shavlik, J. W., Mooney, R. J., and Towell, G. G. (1991). Symbolic and neural learning algorithms: An experimental comparison. *Machine learning*, 6(2):111–143.
- Shortliffe, E. H. and Cimino, J. J. (2013). *Biomedical informatics: computer applications in health care and biomedicine*. Springer Science & Business Media.
- Simpson, R. and Pearson, K. (1904). Report on certain enteric fever inoculation statistics. *The British Medical Journal*, pages 1243–1246.
- Sittig, D. F. and Wright, A. (2015). What makes an ehr “open” or interoperable? *Journal of the American Medical Informatics Association*, 22(5):1099–1101.
- Slavin, R. E. (1986). Best-evidence synthesis: An alternative to meta-analytic and traditional reviews. *Educational researcher*, 15(9):5–11.
- Song, J., Zeng, W., Xu, Y., and Xu, W. (2011). The improvement of neural network cascade-correlation algorithm and its application in picking seismic first break. In *73rd EAGE Conference and Exhibition incorporating SPE EUROPEC 2011*.
- Song, M., Günther, C. W., and Aalst, W. M. (2009). Trace clustering in process mining. In *Business Process Management Workshops*, pages 109–120. Springer.

- Song, M., Günther, C. W., and Van der Aalst, W. M. (2008). Trace clustering in process mining. In *International Conference on Business Process Management*, pages 109–120. Springer.
- Soop, M., Fryksmark, U., Köster, M., and Haglund, B. (2009). The incidence of adverse events in swedish hospitals: a retrospective medical record review study. *International journal for quality in health care*, 21(4):285–291.
- Srivastava, K., Srivastava, A., Kumar, A., and Mittal, B. (2010). Significant association between toll-like receptor gene polymorphisms and gallbladder cancer. *Liver International*, 30(7):1067–1072.
- Steiner, C. A., Bass, E. B., Talamini, M. A., Pitt, H. A., and Steinberg, E. P. (1994). Surgical rates and operative mortality for open and laparoscopic cholecystectomy in maryland. *New England Journal of Medicine*, 330(6):403–408.
- Stone, C. P. (2014). A glimpse at ehr implementation around the world: The lessons the us can learn. *Journal of Electronic Healthcare*, 9(3):89–98.
- Stump, L. S. (2000). Re-engineering the medication error-reporting process: removing the blame and improving the system. *American Journal of Health-System Pharmacy*, 57(4):10–17.
- Suarez, A. L., LaBarre, N. T., Cotton, P. B., Payne, K. M., Coté, G. A., and Elmunzer, B. J. (2016). An assessment of existing risk stratification guidelines for the evaluation of patients with suspected choledocholithiasis. *Surgical endoscopy*, 30(10):4613–4618.
- Swets, J. A. (1988). Measuring the accuracy of diagnostic systems. *Science*, 240(4857):1285–1293.
- Tait Shanafelt, Christine A. Sinsky, S. S. (2017). Preventable deaths in american hospitals. <http://catalyst.nejm.org/medical-errors-preventable-deaths>.(Accessed on Aug. 28, 2017).
- Ter Hofstede, A. H., van der Aalst, W. M., Adams, M., and Russell, N. (2009). *Modern Business Process Automation: YAWL and its support environment*. Springer Science & Business Media.



- Thomas, E. J. and Brennan, T. A. (2001). Errors and adverse events in medicine: an overview. *Clinical Risk Management: Enhancing Patient Safety*. London: *BMJ Publishing*, pages 31–43.
- Thomas, E. J., Studdert, D. M., Burstin, H. R., Orav, E. J., Zeena, T., Williams, E. J., Howard, K. M., Weiler, P. C., and Brennan, T. A. (2000). Incidence and types of adverse events and negligent care in utah and colorado. *Medical care*, 38(3):261–271.
- Times, T. N. (2008). Gallstones in-depth report. <http://www.nytimes.com/health/guides/disease/acute-cholecystitis-gallstones/print.html>.(Accessed on Sep. 07, 2017).
- Tomar, D. and Agarwal, S. (2013). A survey on data mining approaches for healthcare. *International Journal of Bio-Science and Bio-Technology*, 5(5):241–266.
- Truett, J., Cornfield, J., and Kannel, W. (1967). A multivariate analysis of the risk of coronary heart disease in framingham. *Journal of chronic diseases*, 20(7):511–524.
- Tu, J. V., Weinstein, M. C., McNeil, B. J., and Naylor, C. D. (1998). Predicting mortality after coronary artery bypass surgery what do artificial neural networks learn? *Medical Decision Making*, 18(2):229–235.
- Ullman, J., Aho, A., and Hirschberg, D. (1976). Bounds on the complexity of the longest common subsequence problem. *Journal of the ACM (JACM)*, 23(1):1–12.
- van den Heever, M., Mittal, A., Haydock, M., and Windsor, J. (2014). The use of intelligent database systems in acute pancreatitis—a systematic review. *Pancreatology*, 14(1):9–16.
- Van der Aalst, W., Weijters, T., and Maruster, L. (2004). Workflow mining: Discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering*, 16(9):1128–1142.
- Van der Aalst, W. M. (2011). Data mining. In *Process Mining*, pages 59–91. Springer.
- Van Der Aalst, W. M., Reijers, H. A., Weijters, A. J., van Dongen, B. F., De Medeiros, A. A., Song, M., and Verbeek, H. (2007). Business process mining: An industrial application. *Information Systems*, 32(5):713–732.
- Van der Aalst, W. M., Schonenberg, M. H., and Song, M. (2011). Time prediction based on process mining. *Information systems*, 36(2):450–475.

- Van Oirschot, Y., van Dongen, B., Buijs, J., and Dijkman, R. (2014). *Using Trace Clustering for Configurable Process Discovery Explained by Event Log Data*. PhD thesis, Master's thesis.
- Vincent, J.-L., Moreno, R., Takala, J., Willatts, S., De Mendonça, A., Bruining, H., Reinhart, C., Suter, P., and Thijs, L. (1996). The sofa (sepsis-related organ failure assessment) score to describe organ dysfunction/failure. *Intensive care medicine*, 22(7):707–710.
- Vukicevic, A. M., Stojadinovic, M., Radovic, M., Djordjevic, M., Cirkovic, B. A., Pejovic, T., Jovicic, G., and Filipovic, N. (2016). Automated development of artificial neural networks for clinical purposes: Application for predicting the outcome of cholecholelithiasis surgery. *Computers in biology and medicine*, 75:80–89.
- Wagner, R. A. and Fischer, M. J. (1974). The string-to-string correction problem. *Journal of the ACM (JACM)*, 21(1):168–173.
- Walker, S. H. and Duncan, D. B. (1967). Estimation of the probability of an event as a function of several independent variables. *Biometrika*, 54(1-2):167–179.
- Wall, C. F.-T. L. A. (2013). 1 artificial neural networks predict survival from pancreatic cancer after radical surgery.
- Wang, R.-T., Xu, X.-S., Liu, J., and Liu, C. (2012). Gallbladder carcinoma: analysis of prognostic factors in 132 cases. *Asian Pacific Journal of Cancer Prevention*, 13(6):2511–2514.
- Wang, T., Luo, H., Yan, H.-t., Zhang, G.-h., Liu, W.-h., and Tang, L.-j. (2017). Risk factors for gallbladder contractility after cholecystolithotomy in elderly high-risk surgical patients. *Clinical interventions in aging*, 12:129–136.
- Wang, X., Sontag, D., and Wang, F. (2014). Unsupervised learning of disease progression models. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 85–94. ACM.
- Weed, L. (2017). History of ehr. <http://v2020eresource.org/home/newsletter/SM116>.(Accessed on Aug. 28, 2017).

- Weiland, D. E. (1997). Why use clinical pathways rather than practice guidelines? *The American journal of surgery*, 174(6):592–595.
- Weske, M. (2010). *Business process management: concepts, languages, architectures*. Springer Publishing Company, Incorporated.
- Widrow, B. and Hoff, M. E. (1960). Adaptive switching circuits. In *IRE WESCON convention record*, volume 4, pages 96–104. New York.
- Widrow, B., Rumelhart, D. E., and Lehr, M. A. (1994). Neural networks: applications in industry, business and science. *Communications of the ACM*, 37(3):93–106.
- Wilson, R. M., Harrison, B. T., Gibberd, R. W., and Hamilton, J. D. (1999). An analysis of the causes of adverse events from the quality in australian health care study. *The Medical Journal of Australia*, 170(9):411–415.
- Wu, J., Roy, J., and Stewart, W. F. (2010). Prediction modeling using ehr data: challenges, strategies, and a comparison of machine learning approaches. *Medical care*, 48(6):S106–S113.
- Yang, J. and Honavar, V. (1991). Experiments with the cascade-correlation algorithm. In *Neural Networks, 1991. 1991 IEEE International Joint Conference on*, pages 2428–2433. IEEE.
- Yang, W.-S. and Hwang, S.-Y. (2006). A process-mining framework for the detection of healthcare fraud and abuse. *Expert Systems with Applications*, 31(1):56–68.
- Yang, Y., Chen, H., Wang, D., Luo, W., Zhu, B., and Zhang, Z. (2013). Diagnosis of pancreatic carcinoma based on combined measurement of multiple serum tumor markers using artificial neural network analysis. *Chinese medical journal*, 127(10):1891–1896.
- Yerkes, R. M. and Dodson, J. D. (1908). The relation of strength of stimulus to rapidity of habit-formation. *Journal of comparative neurology*, 18(5):459–482.
- Yoldas, Ö., Koç, M., Karaköse, N., Klç, M., and Tez, M. (2008). Prediction of clinical outcomes using artificial neural networks for patients with acute biliary pancreatitis. *Pancreas*, 36(1):90–92.

- Zavaleta-Bustos, M., Castro-Pastrana, L. I., Reyes-Hernández, I., López-Luna, M. A., and Bermúdez-Camps, I. B. (2008). Prescription errors in a primary care university unit: urgency of pharmaceutical care in Mexico. *Revista Brasileira de Ciências Farmacêuticas*, 44(1):115–125.
- Zhang, L., Chen, Z., Fukuma, M., Lee, L. Y., and Wu, M. (2008). Prognostic significance of race and tumor size in carcinosarcoma of gallbladder: a meta-analysis of 68 cases. *Int J Clin Exp Pathol*, 1(1):75–83.
- Zhang, T. (2004). Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Proceedings of the twenty-first international conference on Machine learning*, page 116. ACM.
- Zhao, W., Wang, L., Hu, C., and Hou, J. (2011). Cascade-correlation neural network for sensor fault detection and data recovery with on-line learning. *Sensor Letters*, 9(5):2034–2037.

# List of Publications

## Journal Publications

- **Likewin Thomas**, Manoj Kumar, M. V. and Annappa, B. (2016), Best Resource Recommendation for a Stochastic Process, Information Journal, 19 (10B), 4617-4622.
- Manoj Kumar, M. V., **Likewin Thomas** and Annappa, B. (2016), Concept Drifts Detection and Localisation in Process Mining, Information Journal, 19(10B), 4611-4616.

## Conference Publications

- **Likewin Thomas**, Manoj Kumar, M. V. and Annappa, B. (2014), Efficient process mining through critical path network analysis. Proc., Advance Computing Conference (IACC), IEEE, Gurgaon, India, 511-516.
- **Likewin Thomas**, Manoj Kumar, M. V. and Annappa, B. (2015), An Optimal Process Model for a Real Time Process. Proc., 36<sup>th</sup> International Conference on Application and Theory of Petri Nets and Concurrency 15<sup>th</sup> International Conference on Application of Concurrency to System Design (Petri Nets 2015 // ACSD 2015), CEUR Proceedings, University libre de Bruxelles, Brussels, Belgium, 117-131.
- **Likewin Thomas**, Manoj Kumar, M. V. and Annappa, B. (2016), Optimized Path Recommendation for a Real Time Process, Proc., 18<sup>th</sup> International Conference on Management, Economics and Business Information, International Journal of Business and Economics Engineering, Tokyo, Japan, 3(5).
- **Likewin Thomas**, Manoj Kumar, M. V. and Annappa, B. (2016), Discovery of optimal neurons and hidden layers in feed-forward Neural Network, Proc., Emerging Technologies and Innovative Business Practices for the Transformation of Societies (EmergiTech), IEEE, Mauritius, 286-291.
- **Likewin Thomas**, Manoj Kumar, M. V. and Annappa, B. (2017), An online decision support system for recommending an alternative path of execution, Proc.,

2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), Coimbatore, IEEE.

- **Likewin Thomas**, Manoj Kumar, M. V. and Annappa, B. (2017), Recommending an alternative path of execution using an online decision support system, Proc., 2017 International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence (ISMSI'17), Hong Kong, ACM, 108-112.
- **Likewin Thomas**, Manoj Kumar, M. V. and Annappa, B. (2017), Prediction of Gallstone Disease Progression using Modified Cascade Neural Network, Proc., International Conference on Smart Systems, Innovation and Computing (SSIC), Springer LNCS, Manipal University, Jaipur.
- Manoj Kumar, M. V., **Likewin Thomas** and Annappa, B. (2014), Phenomenon of concept drift from process mining insight, Proc., 2014 IEEE Advance Computing Conference (IACC), IEEE , Gurgaon, India, 517-522.
- Manoj Kumar, M. V., **Likewin Thomas** and Annappa, B. (2015), Capturing the Sudden Concept Drift in Process Mining, Proc., 36<sup>th</sup> International Conference on Application and Theory of Petri Nets and Concurrency 15<sup>th</sup> International Conference on Application of Concurrency to System Design (Petri Nets 2015 // ACSD 2015), CEUR Proceedings, University libre de Bruxelles, Brussels, Belgium, 131-143.
- Manoj Kumar, M. V., **Likewin Thomas** and Annappa, B. (2016), Trace Logo: A Notation for Representing Control-Flow of Operational Process, Proc., 18<sup>th</sup> International Conference on Management, Economics and Business Information (ICMEBI), International Journal of Business and Economics Engineering, Tokyo, Japan, 3(5).
- Manoj Kumar, M. V., **Likewin Thomas** and Annappa, B. (2017), Predicting frequent Execution path in information systems, Proc., 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), Coimbatore, IEEE.
- Manoj Kumar, M. V., **Likewin Thomas** and Annappa, B. (2017), Distilling Lasagna from Spaghetti Processes, Proc., 2017 International Conference on Intel-

ligent Systems, Metaheuristics & Swarm Intelligence (ISMSI), Hong Kong, ACM, 157-161.

- Manoj Kumar, M. V., **Likewin Thomas** and Annappa, B. (2017), Simplifying Spaghetti Processes to Find the Frequent Execution Paths, Proc., International Conference on Smart Systems, Innovation and Computing (SSIC), Springer LNCS, Manipal University, Jaipur, April 2017.

## **Brief Bio-Data**

Likewin Thomas

Research Scholar

Department of Computer Science and Engineering

National Institute of Technology Karnataka, Surathkal

P.O. Srinivasnagar

Mangalore - 575025

Phone: +91 9483503377

Email: likewinthomas@gmail.com

## **Permanent Address**

Likewin Thomas

S/o Maj. (Retd.) T C Thomas

#43, Merry Cottage, Somaiha Layout

Shimoga - 577201

Shimoga (Dist.)

Karnataka, INDIA

## **Qualification**

M. Tech. in Computer Science and Engineering, National Institute of Technology Karnataka, Surathkal, Karnataka, 2011.

B. E. in Information Science and Engineering, Visvesvaraya Technological University, Belgaum, Karnataka, 2004.