DOKUZ EYLÜL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

MACHINE LEARNING MODELS FOR IDENTIFYING CAUSE EFFECT RELATIONSHIP IN MEDICAL TREATMENT DATA

by

Mohammed Abebe YIMER

June, 2021

IZMIR

MACHINE LEARNING MODELS FOR IDENTIFYING CAUSE EFFECT RELATIONSHIP IN MEDICAL TREATMENT DATA

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Computer Engineering

by
Mohammed Abebe YIMER

June, 2021
IZMIR

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"MACHINE LEARNING MODELS FOR IDENTIFYING CAUSE EFFECT RELATIONSHIP IN MEDICAL TREATMENT DATA"** completed by **MOHAMMED ABEBE YIMER** under the supervision of **ASSIST. PROF. DR. ÖZLEM AKTAŞ** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

Assist. Prof. Dr. Özlem AKTAŞ
Supervisor

Prof. Dr. Ali Rıza ŞİŞMAN
Thesis Committee Member

Prof. Dr. Alp KUT
Thesis Committee Member

Prof. Dr. Ayşegül ALAYBEYOĞLU
Examining Committee Member

Assoc. Prof. Dr. Deniz KILINÇ
Examining Committee Member

Prof. Dr. Özgür ÖZÇELİK
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

First and foremost I would like to thank and praise the almighty God (Allah). I want to pass my appreciation to the following people for their diverse contribution throughout my study. Nevertheless, it is hard to know where to start thinking and acknowledging people for their assistance they have shown me. The problem is that I am likely to omit some very important names and for that, I would like to apologize in advance. The following are those to whom I am particularly indebted.

I would like to offer my most profound praise to my research supervisor Prof. Dr. Süleyman Sevinç, for the advice, support, and kinship he was able to extend to me. Without his supervision, invaluable suggestions, dedicated advice, and remarks, the study wouldn't have been a reality. I really appreciate his encouragement, friendly approach, and fatherhood advice. Moreover, I would also like to pass my appreciation to my second supervisor Assist. Prof. Özlem AKTAŞ and my thesis committee members, Prof. Dr. Ali Rıza ŞİŞMAN and Prof. Dr. Alp KUT for the continuous follow-up and guidance they managed to provide to me.

My special thanks, gratitude, and love goes to my wife and my daughters for encouraging me and enduring the ups and downs in my absence. I am greatly obligated to forward my gratitude to my parents for their intensification of the whole spectrum of my life. Moreover, I would like to forward my heartfelt love and thanks to my brother and sisters and their families for their encouragement and moral support over all those times in my study.

Many colleagues and friends have influenced this thesis; I wish to convey my love and appreciation to every one of my companions and colleagues working at Arba Minch University, Labenko Bilişim, Computational Medicine Group, and to all my friends and classmates for their invaluable assistance and fortitude in every aspect for this thesis. Finally, I would like to thank every individual who has committed positive impacts to the successful realization of this thesis work, as well as expressing my apology that I could not mention individually one by one.

Mohammed Abebe YIMER

MACHINE LEARNING MODELS FOR IDENTIFYING CAUSE EFFECT RELATIONSHIP IN MEDICAL TREATMENT DATA

ABSTRACT

The healthcare industry is equipped with a treasure trove of data gathered from various sources. This massive amount of data can be used to work wonders. In this study, we used a carefully selected subset of data from the Medical Information Mart for Intensive Care database. We adopted principal component analysis as the main method, to observe and capture the daily medical changes in intensive care unit patients from their medical treatment data. The results can be used to inform the attending physicians, which laboratory tests are exhibiting variances after an intervention along with their associated epiphenomenon. This will be used to make decisions such as, which treatment or diagnosis to apply further or which prescriptions to give/avoid.

Experimental analysis results indicate that principal component analysis was able to capture daily patient changes after a certain treatment or intervention. Model validity and stability are evaluated using permutation and bootstrap tests. In both cases, the model exhibited an acceptable significance level. Moreover, causal impact analysis is conducted using Bayesian structural time-series models. Results showed that the approach provides promising results for interpreting large quantities of patient data for establishing a cause-effect relationship from medical interventions and be used as an early warning system. The study reflected the capability of principal component analysis to monitor and provide an alert to the clinicians about the patient's conditions, thereby providing opportunities for timely interventions. If combined with other AI models, the approach can be able to support medical decision making and enable effective patient-tailored care for better patient health outcomes.

Keywords: Causal inference, cause-effect relationships, causal impact, decision support, early-warning, principal component analysis, machine learning

TIBBİ TEDAVİ VERİLERİNDEN SEBEP SONUÇ İLİŞKİLERİNİN MAKİNA ÖĞRENİM YÖNTEMLERİ İLE BELİRLENMESİ

ÖZ

Sağlık sektörü, çeşitli kaynaklardan toplanan bir veri hazinesiyle donatılmıştır. Bu büyük miktardaki veri harikalar yaratmak için kullanılabilir. Bu çalışmada, tıbbi tedavi verilerini kullanarak yoğun bakım ünitesi hastalarında günlük tıbbi değişiklikleri gözlemlemek ve yakalamak için temel bileşen analizini ana yöntem olarak kabul edilmiştir. Sonuçlar, bir müdahaleden sonra hangi laboratuar testlerinin değişkenlik gösterdiğini ve ilişkili epifenomenleri ile ilgili hekimleri bilgilendirmek için kullanılabilecektir. Ayrıca, hangi tedavi veya tanının daha sonra uygulanacağı, hangi reçetelerin verilmesi veya kaçınılması gerektiği gibi kararlar verebilmek için bir ipucu olarak kullanılabilecektir.

Deneysel analiz sonuçları, temel bileşen analizinin belirli bir tedavi veya müdahaleden sonra hastanın günlük değişiklikler yakalayabildiğini göstermektedir. Model geçerliliği ve kararlılığı permütasyon ve önyükleme testi kullanılarak değerlendirilmiştir. Her iki durumda da, model hem tek kuyruklu hem de iki kuyruklu istatistiksel anlamlılık testleri için kabul edilebilir bir önem seviyesi sergilemektedir. Ayrıca nedensel etki analizi, Bayesci yapısal zaman serisi modelleri kullanılarak yapılmıştır. Sonuçlar, yaklaşımın tıbbi müdahalelerden bir neden-sonuç ilişkisi kurmak için büyük miktarlarda hasta verilerini yorumlamak için umut verici sonuçlar sağladığını ve bir erken uyarı sistemi olarak kullanılabileceğini göstermiştir. Bu çalışma, temel bileşen analizinin hastanın değişen koşulları hakkında klinisyenlere bir uyarı sağlama ve izleme kabiliyetlerini göstererek, hastalara zamanında müdahale fırsatları sağlama yeteneğini yansıtmaktadır. Diğer makine öğrenimi modelleriyle birleştirilirse, bu yaklaşım klinik karar vermeyi destekleyebilecek, daha iyi hasta sağlığı sonuçları için hastaya özel etkili bakımı sağlayabilir.

Anahtar kelimeler: Nedensel çıkarım, sebep-sonuç ilişkileri, nedensel etki, karar desteği, erken uyarı, temel bileşenler analizi, makine öğrenimi

CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER ONE

INTRODUCTION

1.1 Background

The application of technology for healthcare have become an exciting subject lately. Granted how extreme we have come with technological advancements, it is only right that we should utilize this information and advancement to make healthcare better, efficient, cost effective and patient tailored. We are now observing information technology (IT) moving forward the field, with the digitisation of medical records and healthcare service delivery. Also, the advancement in Artificial Intelligence (AI) is helping the healthcare industry deal with the apparent ever-increasing need for smart healthcare. The collection of significant patient data for many years by medical professionals has prompted a gigantic secret stash of data that we would now be able to take advantage of. This data is profoundly important for improving diagnosis and can assist with dissecting whole host of problems involving symptoms, drugs and dose. Without this, it would be notably all the more trying for clinical experts to come to the correct conclusions within a short period of time that healthcare delivery necessitates.

This work intends to capitalize on the aforementioned facts by applying multiple machine learning (ML) models in healthcare. In the past decade, the advancement in machine learning has presented us with multiple and notable inventions such as self-driving cars, speech and speaker recognition systems, advanced and efficient web search, and so forth. Machine learning is so inevitable nowadays that we possibly use it countless times each day yet without knowing it. ML explores the study and development of algorithmic models that can learn from data and make clustering, classification, and prediction on data. By implementing models from sample inputs these algorithms prevail over strictly following static program instructions. Machine learning is applied in a range of computing tasks where using plain algorithms with acceptable performance is problematic or unreal; example applications consist of e-mail filtering, network intrusions detection, optical character recognition (OCR), and computer vision.

Because of the advent of new computing machineries, today's machine learning is not like machine learning of the past (SAS, 2021). It was comprehended from pattern recognition and the theory that computers can learn without being programmed to perform explicit errands. Moreover, AI researchers wanted to see whether or not computers could gain insights from data. The monotonous part of ML is significant considering the fact that as models are provided with a new information, they can autonomously adjust themselves to the data. They learn from past experience to produce solid, reproducible decisions and results and reveal concealed insights through learning from longitudinal relationships and trends in the data. Generally, machine learning is favoured method for, speech recognition, natural language processing, computer vision, robot control, computational biology, finance, retail and travel. By applying machine learning in medical care, specialists and clinical experts will before long have the option to anticipate with precision on how long patients with fatal diseases will survive. Superfluous laboratory tests can also be avoided with the help and use of smart medical systems and tools that will learn from data. This in turn can help patients and the sector save money. ML can defiantly transform the healthcare domain. However, it is worth stating that any tool developed using ML in any way can't fully supplant specialists or medical practitioners. With the help and advancement of digital technology immense amount of patient medical information is being recorded by hospitals all over the world. These electronic medical records (EMR) assume an essential part in the automation and reshaping of the health care industry.

Electronic medical records what might be compared to paper records, or charts at a clinician's office. They normally encompass wide-ranging information such as medical treatment information and longitudinal patient medical data amassed from routine clinical practices. Until the advent of EMRs, patient medical charts were full of handwritten notes, chronological list of hospital visits and read in linear fashion, starting with the patient's descriptions of the symptoms, followed by physical examination reports of the patient, supporting impartial data, and finally, the physician's assessment and treatment plan. With the help of EMR, patient clinical data can be traced over a prolonged period by multiple healthcare providers from multiple location simultaneously. It can also be used to identify those patients due for

precautionary check-ups & inspections and monitor how each patient measures up-to particular requirements like vaccinations and blood pressure measurements. EMRs are designed to help medical organizations deliver competent and accurate care. Perhaps the most vital advantage is their universality, meaning that in lieu of keeping multiple charts at distinct healthcare places, a patient will have one electronic chart that can be retrieved from any healthcare facility using particular software.

Machine learning is a fast-growing trend in the healthcare sector, thanks to the advent of wearable devices and sensors that can use data to assess a patient's health in real-time (Ananth, 2020). The innovation can equally empower specialists to analyse data to recognize patterns or warnings that may urge better-quality judgments and treatment. Using machine-learning algorithms, we can analyse massive amount of structured and unstructured clinical observational data to expose concealed insights on indications, patient & provider concerns and other matters that can affect patient care. Then turn this acumen into evidence-based information that can aid us to predict and improve outcomes. Moreover, the study by Lavrač, Kononenko, Keravnou, Kukar, & Zupan (1998) emphasized that, healthcare professionals will be able to arrive to a correct decision or prompt an appropriate actions in a limited amount of time, if they are equipped with the right information at the right time. In view of this facts, it is obvious that users are in need of using smart tools in such a way that it can help simplify their life. This striking interest of users brings to the idea of EMRs and applying machine-learning models on theses EMRs. Randomized control trials (RCT) plays an important role in medical research for establishing causal-relationships between adverse drug events and their outcomes. However, we believe that the application of machine learning models could also play a vital role in this regards. Applying machine-learning models to identify cause-effect relationships in a medical treatment data plays an important role in saving human lives by providing relevant and timely information about the status of a patient to the attending physician.

1.2 Problem Statement

In the current digital age, an immense amount of data is being created and warehoused by private industries and governments alike. Nowadays any simple human

act can be recorded, traced and predicted. As a trend this simple act of recording, tracing and forecasting of tasks has come across the healthcare sector. However, although several success stories have been published, learning new actionable insights has not been as common in the healthcare area (Lay-Flurrie, 2016). The study Palaniappan & Awang (2008) indicates that, nowadays most hospitals use hospital information systems to manage the hospital as well as patient data. During this process the hospital gathers a wealth of data, however this data are seldom used for clinical decision support and for the betterment of the hospital services in terms of time, resources as well as monetary. Furthermore, Polat & Güneş (2007) describe the central problem of the information age is, handling the vast quantity of raw information collected. It also emphasizes that, as more and more data is recorded, the gap between producing the data and understanding the latent knowledge from the data is widening. Also, the study points out the use of data mining and knowledge discovery techniques to bridge this gap.

In the process of medical treatment, vast amount of data is accumulated in particular for inpatients or patients who are remotely monitored. These data includes laboratory data, demographic data, clinical data, radiological data and other types of data. While the physician provides diagnostic and treatment services for many patients, the credibility of the effectiveness and timeliness of the decisions falls in anticipation of analysing this pool of data in the face of the multitude of patient numbers. There is a need for informative machine learning tools and models to help boost the effectiveness, efficiency and timeliness of the medical treatments to be provided. In view of the aforementioned facts, the study aims to model the cause-effect relationships of medical treatment data and to bring this insight to the attention of the physician using machine-learning models.

In conclusion, clinically there are many studies conducted in this area. However, these studies do not suggest acceptable solutions in the clinical process in order to relate the causes of the medical treatment with its effects. In our study, the main goal will be to use an approach and develop a tool, which can be used by the physician as part of the daily routine clinical practice. In terms of the results to be obtained, the system is aimed as a data analysis tool and methods that can be directed by the

physician instead of being an autonomous system independent of the physician. The "most effective single method" will not be sought in this study. Instead, multiple options will be generated and presented for the physician to choose and use effectively in various situations, and the physician will be able to decide a certain type of treatment or diagnoses based up on the results presented.

1.3 Research Questions

A. Is this patient showing any changes after the treatment?

B. Is this drug going to be effective on this patient?

C. Does this treatment have a negative or positive effect on my patient?

1.4 Hypothesis

Doctors use normal reference ranges to decide what is normal or abnormal while diagnosing a patient for a disease. Any patient vital sign change is regarded as normal as long as it stays within the normal reference range. However, studies show that only using reference ranges for seriously ill patients in intensive care unit (ICU) setting is not the best sought out method. Also, we believe that this minor changes may reveal significant information if fused with other laboratory tests. Even if this changes are within the normal reference changes. In view of this facts, we hypothesize that principal component analysis (PCA) can be able to capture the changes after a medical intervention and cause-effect relationship can be established between the medical treatments and their outcomes by healthcare practitioners.

1.5 Objective of the Study

1.5.1 General Objective

The general objective of this research work is to explore the possibility of using machine-learning models on electronic medical records for the systematic pattern extraction and exploration of cause-effect relationships in medical treatment data.

1.5.2 Specific Objectives

In order to achieve the above general objective, the research will accomplish the following specific objectives:

- Review, analyse and study literatures on electronic medical record analysis and the structure and characteristics of EMR.
- Review, analyse and study patient electronic medical records to systematically extract and explore cause-effect relationships in medical treatment data.
- Asses the various machine-learning approaches and algorithms for the development of a model.
- Collect and organize required datasets for the proposed model.
- Design a machine learning model for identifying cause-effect relationships in medical treatment data.
- Develop a prototype application.
- Test and analyse the prototype system.
- Draw conclusion and recommendation based on results obtained.

1.6 Research Methodology

1.6.1 Data Collection

For the research, an acceptable amount of medical treatment data is collected from the publicly available Medical Information Mart for Intensive Care (MIMIC) v1.4 database. MIMIC-III is a large, single-center database encompassing information involving patients admitted to critical care units at a large tertiary care hospital. MIMIC-III contains deidentified, all-inclusive clinical data of patients admitted to the Beth Israel Deaconess Medical Center in Boston, Massachusetts, and is usually available to researchers internationally under a data-use agreement. The database includes information such as vital signs, medications, laboratory measurements, charted observations and notes, fluid balance, procedure and diagnostic codes, imaging reports, hospital length of stay, survival data, and more. The database supports applications comprising academic and industrial research, quality enhancement

initiatives, and higher education coursework (Goldberger et al., 2000; Johnson et al., 2016).

In order to gain access to MIMIC, one will need to finish the Collaborative Institutional Training Initiative (CITI) "Data or Specimens Only Research" course and be certified. Certificate of completion for the aforementioned course is presented under Appendix 2. The course covers nine modules on Data and Specimens Only Research including, History and Ethics of Human Subjects Research and Health Insurance Portability and Accountability Act (HIPAA) Privacy Protections. After completing the course, one needs to fill in a form to provide a data-use agreement and demand a restricted-access to the clinical databases hosted on PhysioNet along with the completion report from the CITI "Data or Specimens Only Research" training program (PDF or image file). After submitting the request application, we were approved and granted access to MIMIC-III v1.4 database. The database contains over 58,000 clinical admissions for 38,645 adults and 7,875 neonates. The data covers from June 2001 - October 2012 (Physionet, 2018). The database, although de-identified, yet contains comprehensive information about the clinical care of patients, due to this it is treated with appropriate care and respect.

1.6.2 Data Acquisition

The dataset is made up of 26 separate csv files. Which is converted into Postgress database for querying. For this research work, after downloading and converting the csv files into Postgress database, selected tables were queried for analysis. Table 1.1 contains list of tables used along with their descriptions for this work. Also, sample SQL queries used to extract data from Postgress database and the overall data extraction flow is presented.

Table 1.1 Queried list of tables

Table Name	Description	Rows
icustays	List of ICU stays.	61,532
patients	Patient's information admitted to the ICU.	46,520
diagnoses_icd	Diagnoses associated to hospital admission coded using the ICD9 system.	651,047
d_icd_diagnoses	Dictionary of the International Classification of Diseases, 9th Revision (Diagnoses).	14,710
labevents	Events relating to laboratory tests.	27,854,055
d_labitems	Dictionary of laboratory-related items.	753
microbiologyevents	Events relating to microbiology tests.	631,726
d_items	Dictionary of non-laboratory-related charted items.	12,487
noteevents	Notes associated with hospital stays.	2,083,180

The following is a sample query executed to return only patients admitted to ICU for further analysis and queries in accordance with this research work.

```
SELECT  icu.hadm_id, icu.subject_id, icu.icustay_id, icu.intime, icu.los, pat.gender, pat.dob, pat.dod, diag.icd9_code, diag.seq_num, icd9.long_title
INTO mimiciii.patient_data FROM mimiciii.icustays icu
  LEFT JOIN mimiciii.patients pat ON icu.subject_id = pat.subject_id
  LEFT JOIN mimiciii.diagnoses_icd diag ON icu.subject_id = diag.subject_id
  LEFT JOIN mimiciii.d_icd_diagnoses icd9 ON diag.icd9_code = icd9.icd9_code
```

After running first level queries such as the above one, further queries were executed to extract patients with ICU stay of more than 30. This is followed by python execution to prepare and format the data for analysis. The following workflow shows brief overview of the data extraction steps and Table 1.2 presents general statistical information on the dataset content prepared. Furthermore, the overall data extraction and preparation process from the original MIMIC-III dataset is depicted in Figure 1.1.

1. Convert the csv file to Postgress Database.
2. Query the aforementioned databases to prepare general dataset.
3. Query patients whose ICU length of stay (LOS) is >=30 from patients table.
4. Query ***labevents*** table for the aforementioned subjects.
5. Query ***microbiologyevents*** table for the aforementioned subjects.
6. Read the extracted labevents data into python.
7. Retrieve subject ids.
8. Loop through the subjects list:
    a. Extract subject specific data from dataset.
    b. Retrieve unique laboratory test dates.
    c. Retrieve unique laboratory test names.
    d. Reshape the data and create Pandas DataFrame for analysis.
    e. Save the result into a csv file with subject id as a file name.

For our intended research, a total of 1410 patients with a hospital LOS greater than or equal to 30 days is selected. Next, for each of this patients corresponding demographic and clinical data is extracted from the original database. After careful inspection of the extracted data 1306 (out of 1410) patients having at least 10 observations were selected for further analysis. This threshold is chosen so as to have more observations and to ascertain that the model really captures the intended relationship. The extracted dataset contains demographic information and numerous medical laboratory test results along with prescription information. Table 1.2 describes general information about the dataset used.

Table 1.2 Used dataset general information

	Count	**Average LOS (in days)**
M	732	55.2
F	574	57.78
Overall	1306	56.33

The above data is used for a preliminary investigation of the proposed method. PCA is applied on each patient data from the above selection in order to monitor patient

progress and changes after a certain treatment is applied. This is followed by a second round selection of specific dataset from the above group for a detailed investigation of patient monitoring and establishing cause-effect relationships. In the detailed investigation along with the daily change information medical prescriptions provided on those dates are also presented. The intuition behind this is by looking at the changes captured by the PCA results and the prescriptions on the same date, a healthcare professional can be able to establish cause-effect relationships from the data. For this task, after consulting with a healthcare expert, the top-ten diagnoses such as pneumonia and sepsis are selected out of the 1306 patients. This this followed by selecting 50 patients proportionally from these diagnoses. This threshold and method is adopted based on our consultation with a healthcare expert to incorporate medical treatments so that the user will be able to establish cause-effect relationships. The contents of the selection process is presented in Table 1.3. Table 1.4 presents sample dataset extracted with these steps for a single patient.

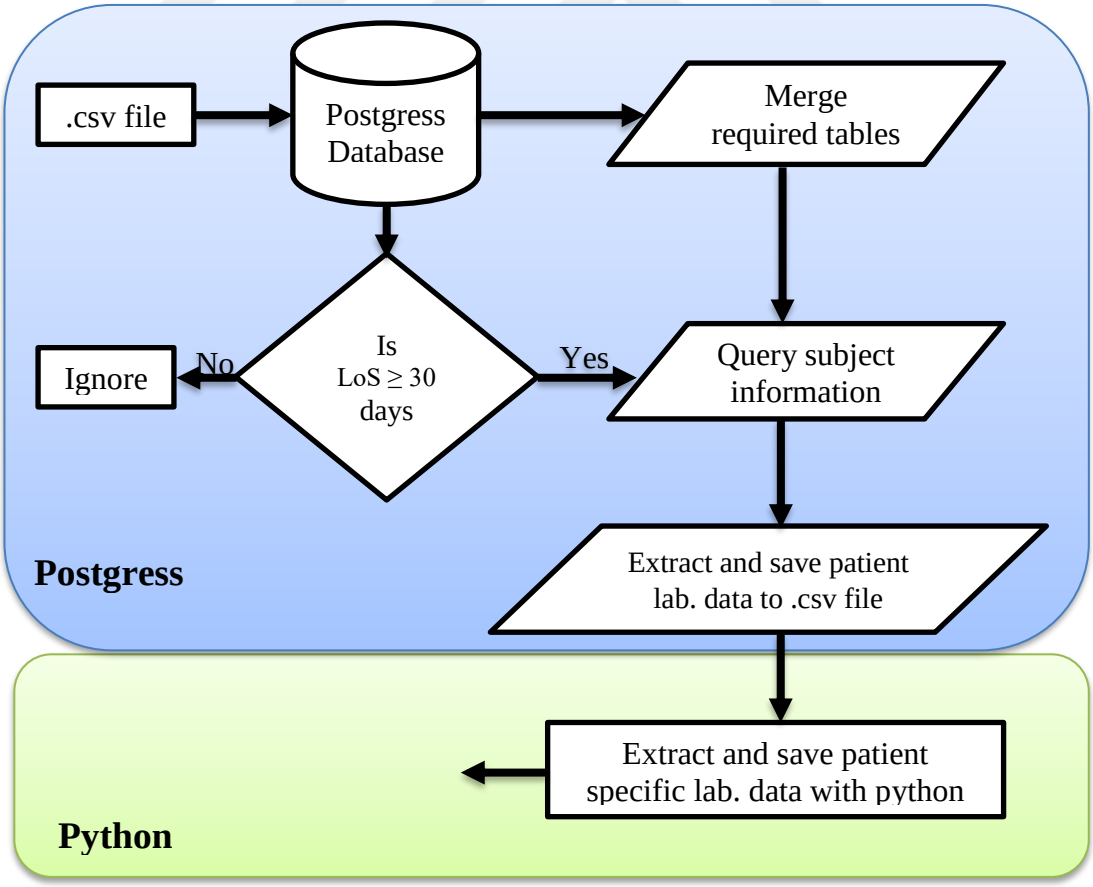


Figure 1.1 General data extraction flow from original database

Table 1.3 Results of subject selection process

Diagnosis	Count	Total Percentage (X)	Selected (X% of 50)
Sepsis	63	19	9
Pneumonia	55	16	8
Gastrointestinal Bleed	50	15	7
Fever	39	12	6
Congestive Heart Failure	26	8	4
Respiratory Failure	26	8	4
Coronary Artery Disease	25	7	4
Abdominal Pain	21	6	3
Chest Pain	16	5	2
Pancreatitis	15	4	2
Total	336	100	50

According to the information from Table 1.3 above, 50 patients with the listed diagnoses are selected. For instance, 9 and 8 patients diagnosed with sepsis and pneumonia respectively, are selected. In addition, their corresponding laboratory test results and the corresponding (available) medical treatment data such as the prescriptions provided are extracted from the original database. At this junction, it is worth mentioning that for some patients we were unable to find full prescription data throughout the ICU stay. In those circumstances analysis is conducted only for those dates where prescription data is available. The dataset contains laboratory tests conducted over a specified period of time. Numerous laboratory tests were conducted within the ICU stay, and in some cases, a test is conducted multiple times per day as presented in Table 1.4. Laboratory test dates are de-identified according to the HIPAA privacy rule. At the end of their hospital stay, the patients are discharged from the hospital alive or dead to home or another healthcare unit depending on their conditions.

Applying machine-learning techniques for a certain problem requires a pre-processing phase where the data is prepared and cleaned before it is actually used to ensure the quality and performance of the model (Abebe & Sevinç, 2020). Moreover, this pre-processing stage helps the models to overcome bias. This pre-processing task includes imputing missing values, removing outliers, and non-required

variables/features among others. In this study, after the dataset is extracted and prepared with the format in Table 1.4, data pre-processing is applied. First, non-numeric variables and variables with a single measured value (no variance) were excluded. Variables with only a single value are discarded from the analysis since they don't affect the results of the analysis. For example, Table 1.5 shows the contributions of variables under different principal components.

Table 1.4 Sample dataset content

Date	Anion Gap	Bicarbonate	Bilirubin, Direct	Bilirubin, Total	Chloride	…
2/26/2191 16:20	18	16	0.2	4.4	114	…
2/28/2191 0:30	nan	nan	0.3	4.3	115	…
3/1/2191 4:30	19	15	0.3	4.1	106	…
3/1/2191 13:55	19	13	nan	nan	110	…
3/2/2191 5:20	23	14	0.7	3.4	106	…
3/2/2191 12:45	23	13	nan	nan	105	…
3/3/2191 4:30	17	21	0.3	4.9	99	…
3/3/2191 23:35	21	20	0.3	6.3	97	…

Table 1.5 Variable contributions/loadings of a PCA analysis

PC No.	HbA1c%	Albumin	ALP	ALT	AaDO2	Amylase	APTT	AST
PC1	0	0.31	-0.33	0.62	0.00	0.26	0.00	0.59
PC2	0	-0.59	0.57	0.31	0.00	-0.25	0.00	0.42
PC3	0	0.00	0.00	0.00	1.00	0.00	0.00	0.00
PC4	0	0.00	0.00	0.00	0.00	0.00	1.00	0.00
PC5	0	-0.26	0.22	-0.13	0.00	0.93	0.00	0.00
PC6	0	0.70	0.72	-0.01	0.00	0.03	0.00	0.03
PC7	0	0.06	-0.09	-0.71	0.00	-0.06	0.00	0.69
PC8	0	0	0	0	0	0	0	0

HbA1c% has only one recorded value which is equal to 6.7 throughout the hospital stay for the selected patient. As a result, it can be seen that HbA1c% contributes nothing (the loading is zero) to the principal components. Consequently, removing this variable before analysis does not affect the results of the analysis. On the contrary,

removing it helps speedup execution time of the overall analysis. In addition, as part of this process imputing missing values is conducted. Generally, missing values arise where no data is recorded for a variable due to different reasons. Missing values are common and are inadequately handled in both observational and experimental researches. Failure to handle missing data issues properly will have an adverse effect on the model and can lead to biased parameter estimates and thereby degrading model performance and efficiency. Just like any other dataset, electronic medical records come with missing values due to a range of reasons. To produce a model with an acceptable performance and efficiency, this EMRs must be imputed properly before applying any machine-learning algorithm. Hence, a proper missing value imputation technique must be employed before applying any algorithm on any EMR (Abebe & Sevinç, 2020).

As a principal imputing method, for this work most_frequent strategy from Scikit-Learn is employed for imputing missing values. This method fills in the missing values with the most frequent value of each variable. That is for each laboratory test it searches for the most frequently occurring value and uses this value to impute the missing cell for that laboratory test. One of the main advantage of this strategy is that it can be used both for non-numeric and numeric data. For imputing this missing values python sklearn impute library is employed. This process is applied for each patient data selected for analysis separately. This strategy is selected due to the fact that the analysis is patient based.

Different medical laboratory tests use different unit of measurement. For instance, Albumin is commonly measured in g/dL whereas medical laboratory tests such as ALP, AST, ALT, and Amylase are measured in U/L. Mathematically, the difference in magnitude due to the difference in measurement unit affects the results of the principal component analysis. Tests with high values will allow the principal components (directions) to draw high proportion which shows a bias. To avoid such bias data standardization must be applied on the dataset. For example, Figure 1.2 depicts the scree plot of the difference of between PCA analysis results for standardized and non-standardized dataset. Figure 1.2 (a) show results of non-standardized dataset whereas Figure 1.2 (b) show results of standardized dataset. As it

can be seen from Figure 1.2 (a), the first principal component amounts about 71% percent of the overall variance and only the first two principal components present more than 90% of the overall variance in the data, this happened due to the difference in magnitude of the variables that created the bias. On the other hand, PC1in Figure 1.2 (b) explains about 30% of the total variance accounted for in the data, which is approximately half of what is explained by PC1 in Figure 1.2 (a). This happens due to the effect of data standardization.

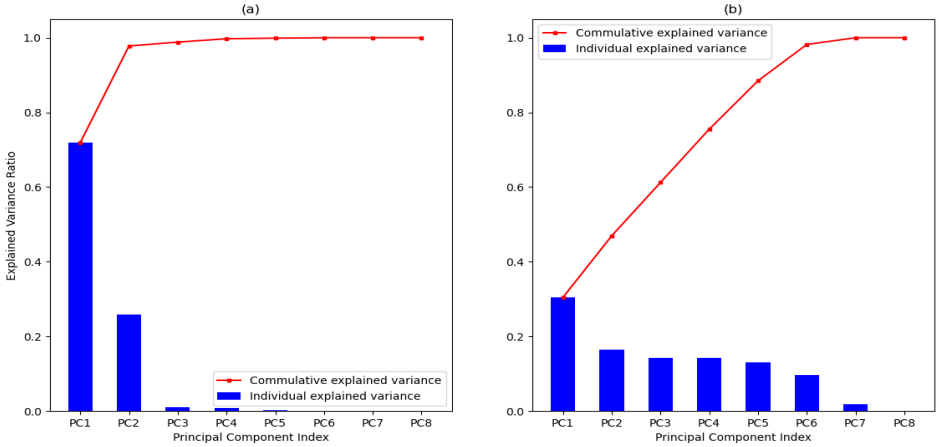


Figure 1.2 Standardized vs non-standardized data PCA analysis results difference

Generally, during standardization we remove the mean and scale the observation to unit variance. Mathematically, the standard score (z-score) of a sample x is calculated as in equation 1.1.

$$z = \frac{(x - \bar{x})}{s} \tag{1.1}$$

$\bar{x}$ is the mean and s is the standard deviation of the training samples. This process is applied on each variable by calculating the pertinent statistics on the samples in the dataset. Standardization is a general prerequisite for many machine learning models. Unless it is applied, most machine learning models perform poorly. After the dataset is extracted, imputed and standardized, patient daily records are sorted and grouped based on laboratory test date before they are fed to the model for analysis. A baseline

of at least four observation should be available on a specific date to start the analysis. Otherwise those observations are merged with the next day test results. This process is repeated iteratively until we achieve the minimum amount of observations for our analysis. Even though there is no universally agreed upon predetermined minimum number of observations to use for PCA, this is necessary so that we have ample number of observation to apply PCA effectively. Finally, the relevant patient data are fed into the model and principal components are computed and visualized. The overall data extraction and preparation process for the PCA analysis is described and depicted in Section 3.2.1 with more details.

1.6.3 Model

In terms of analysing patient data, no magic solution is available for all patients and all diseases types. Instead of designing a series of algorithms to help analyse and interpret patient data, in this study, models that can be able to provide an informative environment for the physicians are considered. The research intends to design explorative machine learning models for identifying cause-effect relationship from medical treatment data. With this intent, the research used PCA as a primary method for monitoring patient changes in intensive care unit settings. Moreover, machine learning models such as Gaussian processes regression (GPR), Support vector regression (SVR), and Long-short term memory (LSTM) approaches for predicting future values are also implemented. Based on the assessment and study of the medical records made, an appropriate machine-learning model for identifying cause-effect relationship in medical treatment data that is suitable and efficient is developed.

1.6.4 Tools and Implementations

In conducting the research on systematic pattern extraction and exploration of medical treatment data using machine-learning models for identifying cause-effect relationships, Python 3.6 specifically, libraries such as Scikit-Learn (Pedregosa et al., 2011b), Pandas, and Numpy are used as a main development tool. This tools are used mainly to conduct the PCA analysis and develop the predictive models mentioned in the document. Besides Python, CausalImpact R package is used for causal impact

analysis. For prototype application development, both python and bokeh library are mainly used. The prototype application is developed as a simple web-based application. Moreover, the overall task is carried out on a 64-bit, Intel Core i5 personal computer with an 8GB RAM and 2.60GHz CPU speed.

1.6.5 Experimental Analysis

The collected data is used both for training and evaluating the proposed models. Afterwards, for the performance analysis of the proposed methods, a prototype web based application is developed and the system evaluated. As a performance evaluation and model stability and validity of the PCA, non-parametric methods such as bootstrap and permutation testing are used. Subsequently, causal impact analysis is conducted and summary of the results is presented. Moreover, performance evaluation of the predictive models is computed using root mean squared error. After the completion of model evaluation processes, results are tabulated, plotted, and reported. Finally, based on the result of the evaluation, conclusion and recommendation are drawn and the results are reported.

1.7 Application of Results and Beneficiaries

Polat & Güneş (2007) the authors argue that assessment of data gathered from patients and verdicts of specialists are the most central features in medical analysis. Nevertheless, a knowledge based systems and various machine-learning practices for medical data analysis help specialists in a great deal. There are numerous advantages of developing machine-learning models for the healthcare industry. Specifically, the advantage of developing models for medical data analytics is paramount to the healthcare industry. Physicians are increasingly ineffective against the multitude of patient numbers and the multiplicity of data. The most important function that healthcare professionals need to make in patient diagnoses is correct evaluation of the patient conditions based upon the physical examination of the patient. Accurate evaluation of data provides a way for a timelier, effective treatment and more accurate diagnosis. In view of these facts, the beneficiaries of this study will be:

- Machine learning researchers for healthcare.

- Healthcare professionals who want to conduct research on medical treatment data analysis and for use by these.
- Machine-learning researchers who need to work on medical treatment data analysis.

1.8 Scope of the Study

The study did an exploratory data analysis using principal component analysis on electronic medical records collected from a single tertiary teaching hospital. There is no assurance that similar results would be achieved in a different site, clinical settings or specific patient groups. Nevertheless, it is worth mentioning that this tool is not intended in no way to replace or undermine the diagnostic skills and professional instinct of medical practitioners. In addition, medical interpretation of the results is consulted with selected medical professionals. However, any of this medical interpretation of the results is not included in this document.

1.9 Ethical and Data Acquisition Issues

Numerous obstacles exist that obstruct the development and application of machine learning systems in healthcare. Probably the greatest challenges for applying machine learning models in healthcare is acquiring patient electronic medical records with ample size and quality. Since electronic medical records are ensured by stringent privacy and security guidelines, the data isn't anything but difficult to gather, share, and circulate. Moreover, there are ultimatums with the format and worth of the records which requires vital energy to clean and make it ready for machine learning applications. Furthermore, to reflect ethical matters in this research, all information fields associated to patient identification are de-identified from the dataset before applying any machine learning processes according to the HIPAA guidelines. In order to request access to MIMIC, one will need to complete the CITI “Data or Specimens Only Research” course and be certified (see Appendix 2). By following and applying the required guidelines and restrictions put forward by the host of the MIMIC database, we are approved and granted a restricted access to the database for our use.

The article by Srikanth (2019) argues that the greatest machine learning apparatus for healthcare is the doctor's intelligence. However, it is said that there is a growing concern by the doctors that machine learning tools will replace them. There will never be a substitution for the art of medicine. Human touch and caring for patients is vital in healthcare. There is a possibility that the application of machine learning tools may eliminate this features to some extent. But it should be noted that this tools should be augmented in a daily routine clinical practice rather than a replacement for human involvement. This will assist the healthcare sector apply more analytics and AI-based tools into daily clinical practices. Similar methods used in medicine for treatment investigations should be applied and followed while designing machine learning tools for healthcare application to guarantee safety and efficiency. Furthermore, it is worth mentioning that we need to comprehend the ethics implicated in healthcare before applying any machine learning tool.

1.10 Ethics Committee Approval

The study used a publicly available, deidentified medical dataset with the proper procedures. Hence no ethics committee approval was required.

1.11 Organization of the Thesis

The whole thesis is organized into six chapters. Introduction in to the full research work including objectives and scope of the study are discussed under Chapter One. The Second Chapter is all about literature review and related works. It describes the approaches and methodologies used so far for causal inference and decision making in healthcare. It also focuses on the study and assessment of the nature and structure of electronic medical records and their analysis. Chapter Three discusses the architecture, structure, design, and characteristics of the proposed methods. Chapter Four mainly focuses on the experimental analysis conducted for establishing cause-effect relationships from medical treatment data. It also presents the results obtained using the prosed method. The development processes of a prototype application are discussed under Chapter Five. Finally, the last chapter, Chapter Six is all about drawing conclusion and recommendations that comprises both summary of the work done and recommendation for future work.

CHAPTER TWO
LITERATURE REVIEW AND RELATED WORKS

2.1 Conceptual Topics

2.1.1 Machine Learning

In recent days, machine learning has become an exciting field of study for numerous reasons. Various literatures define machine learning in different ways from different perspectives. Generally, machine learning can be defined as a collection of algorithms that can be used to teach computers how to perform a certain task by providing examples of how it is performed i.e. learn from examples/experience. For example, we want to write a program to discriminate between chronic kidney disease (CKD) and non-chronic kidney disease patients. To solve this problem we could write a set of instructions to flag CKD and non-CKD. Nevertheless, writing these set of instructions to discriminate CKD from non-CKD patients accurately can be quite problematic, resulting in many misclassified instances. Fortunately, machine learning provides a solution for these type of problems. A machine learning algorithm can be trained using training data that is manually labelled as CKD and non-CKD and the model learns to distinguish between the two. Finally, the performance of the algorithm can be tested later with test set (hold-out data). Generally, implementation of most machine learning models has two phases: training and testing phase. During the first phase, the model is trained or is allowed to learn from the collection of training set. Whereas in the second stage, model performance is tested using the test set by allowing it to make decision using some new test (hold-out) data.

Generally, there are two major categories of machine learning models: supervised and unsupervised learning. However, recent advancements in the field bring out two additional categories called semi-supervised learning and reinforcement learning. As the name implies in supervised learning, the model will be supplied with a labelled training data with the actual answers, e.g. CKD, non-CKD. Examples of this category include classification and regression. For unsupervised learning the model is supplied

with unlabelled dataset and is allowed to discover patterns within the data. Common examples of this group include dimensionality reduction and clustering. Semi-supervised learning is the amalgamation of supervised and unsupervised learning in which the model is supplied with a mixture of both labelled and unlabelled dataset. In reinforcement learning, the model is trained to make series of decisions. An agent such as a robot or controller depending on the outcomes of past actions, decides to learn the ideal action to take. There exists a misperceptions between artificial intelligence, machine learning, and deep learning by some users. Machine learning is built on the idea that computers can be able to learn and adapt through experience. Whereas artificial intelligence encompasses a broader area where computers can execute tasks intelligently. Generally, while AI encompasses broad area of application, machine learning can be categorized as a subset of AI, whereas deep learning (DL) can be categorized as a subset of machine learning as depicted in Figure 2.1.

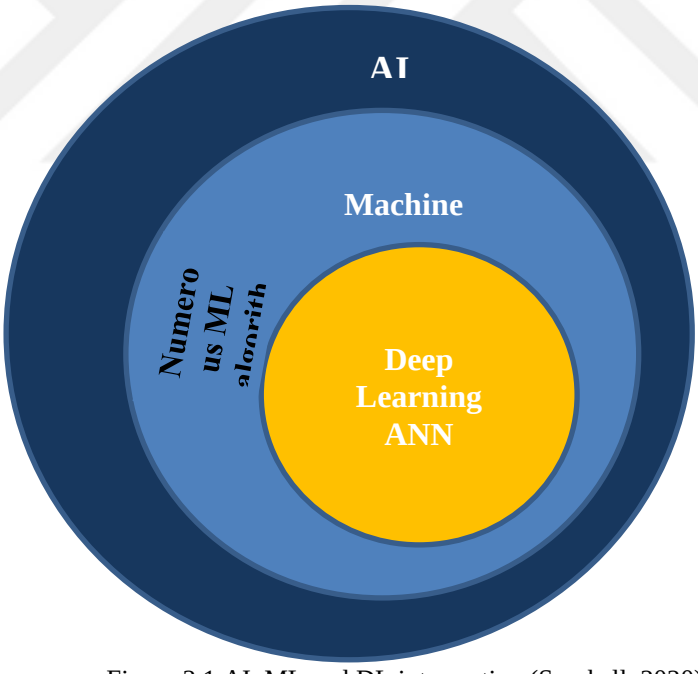


Figure 2.1 AI, ML and DL intersection (Serokell, 2020)

AI in particular, machine learning gives us a wide number of choices by which we can solve numerous problems, alongside the immense experience of the researchers in the field regarding which techniques will in general be effective on a specific class of data. Some advanced techniques give methods of computerizing a portion of these

decisions, for example, choosing between alternative models. Practically speaking, there is no single "silver bullet" solution for all learning problems. Utilizing machine learning practically requires that we utilize our own experience and experimentation to tackle problems. However, using machine learning models, we can do astounding things. It is also reasonable to mention that there are too many manual practices in medicine that can be improved by automation and the help of technology. Since the advancement of machine learning and its adaptation in the healthcare industry, developments in electronic medical records have been significant. If technology is to boost healthcare in the future, then the electronic healthcare records provided to doctors must be boosted by the use of data analytics and more machine learning applications. The worth of machine learning in healthcare is its capability to process vast amount of data afar the range of human ability, and then steadfastly translate analysis of that data into meaningful clinical understandings that can help healthcare authorities when planning and providing care. This ultimately leads to improved results, minimized care expenses, and increased patient contentment. The following are the main advantages of machine learning among others in healthcare:

- Personalized and precision medicine
- Reduce hospital readmissions
- Medical imaging and diagnosis
- Outbreak and chronic diseases prediction
- Smart healthcare records
- Reduces hospital length-of-stay
- Drug finding and production
- Clinical experiments and research
- Telemedicine and Robotic surgery

In the past two decades, technology for healthcare has become an emerging topic. Furthermore, the developments in the field especially AI and machine learning are what make it interesting for application in the healthcare sector. Recently, the application of AI-based tools is advancing the healthcare sector to a smarter healthcare delivery which started with the production of electronic medical records and the application of this tools for smart care delivery. Technology such as wearable devices,

applications, and AI are aiding overwrought hospital staff to manage the ever-increasing demand. The collection of vital patient data for years by medical professionals has led to a vast amount of data that we can now use. This data is extremely vital for advancing disease diagnosis and can help analyse medical issues including patient symptoms, drugs, and therapeutics. Without this information, it would be extensively all the more trying for clinical experts to reach at the correct conclusion at the right time.

2.1.2 Healthcare Data Analysis

Big data is the accumulation of vast amount of data with certain features that can be tapped to elucidate numerous tasks. Due to its great potential and application, it has garnered a special attention for the past decades. It is being generated, stored and analysed by various private and public industries to improve service delivery. Similar to other industries the healthcare industry also produces vast amount of data daily. There are various sources of big data in the healthcare industry. This includes hospital records, wearable sensors, medical records of patients, and other appliances that are part of internet-of-things. Besides, other areas of research such as biomedical researches also produce substantial amount of data relevant to public healthcare. The study by Dash, Shakyawar, Sharma, & Kaushik (2019) points out that proper management and analysis of this vast amount of data is vital to extract meaningful insight. However, there are several challenges accompanying the handling and management of big data that can only be handled by advanced computing solutions. In order to achieve this, healthcare providers need to be fortified with the suitable computing infrastructure. Scholars in the field agree that an efficient administration, scrutiny, and analysis of big data can revolutionize and open new opportunities for smart and modern healthcare. It is in view of these facts, the healthcare industry is taking the extra step to adapt this potential into a better service delivery mechanism and financial advantage.

Li, Zhang, & Tian (2016) also agrees that big data in healthcare holds great value for healthcare management, patient care and treatment, scientific research in the area and education in the healthcare sector. Due to the complexity and nature of medical

data, traditional data storage, access, processing and analysis capabilities are not sufficient. There is a need for new and advanced approaches that can be able to convert this wealth of data into meaningful insights for advancing the healthcare industry. In this regards, healthcare data analysts deliver assistance to enhance the effectiveness of healthcare service delivery. They also apply big-data analytics and data science in search of an acceptable resolution for the healthcare industry which is encompasses multiple units including doctors, medical materials, diagnostic tools, hospitals and much more. Providing suggestions to the hospital administration for increasing profit and effective and efficient service delivery without affecting treatment cost is the main goal of these data analysts. Additional roles of the data analyst includes (Patil, 2019):

- The collection and interpretation of data from various sources.
- Comprehending hospital functionalities and systems.
- Hospital database management.
- Reports and dashboards creation.

Patient care can be improved and accelerated with the proper use and application of advanced data analytics techniques. These techniques can help cut costs and improve the efficiency and effectiveness of healthcare delivery by analysing the wealth of data. Data analytics in this context just denotes the use of huge amounts of structured and unstructured data and analysing them for patterns to extract hidden knowledge from the data. Generally, the healthcare industry is data-reliant. In this regard, the application of data analytics is paramount. It can help derive patterns and hidden insights from the vast amount of data accumulated in electronic health records. It can help minimize wastage of resources, help track the progress of critically ill patients, and can even pursue the well-being of populations, and recognize people at risk for prolonged diseases. Using such type of inputs from data analytics, the healthcare industry can be more efficient in allocating resources to maximize income, improve population health, and patient care.

In recent days, the application and use of healthcare data analysis has become abundant, this presents optimistic changes in patient experience and quality of care. Furthermore, healthcare data analytics can provide a second dimension in which the performance and efficiency of the healthcare practitioners are evaluated. It can

frequently appraise healthcare practitioners in real-time, to trail and enhance the successful practices of physicians and improve healthcare service delivery. By providing detailed analysis results, it can also help cut costs by reducing unnecessary care and avoiding redundant examinations. It can also help the healthcare industry minimize waste and maximize efficiency by identifying patterns in population outcomes. Treatment of chronic diseases makes the healthcare industry spend large amount of money. However, the application of predictive models can help significantly lower expenses by forecasting which patients are at higher risk for disease and act proactively. With the use of data analytics, multiple patient-based variables can be modelled to predict the risk of chronic diseases.

Researches show that randomized controlled trials have for some time been acknowledged as the best quality standard for medical research. This process simply amasses a large number of subjects and arbitrarily split them into two groups, one subjected to the treatment under the study and the other one used as a control group. This process is performed repeatedly to prove exposing the patients to a certain treatment results in a better or alternate outcomes. However, as Ketchersid (2013) in his work, Big Data in Nephrology: Friend or Foe? points out, there is an abundant number of demerits to this approach including; the process is expensive, takes long time to complete, frequent exclusion of patient with some attributes, which in turn makes it hard to conclude the results of the study. Moreover, the study also reveals that, there is a strong urge for personalized healthcare these days, rather than applying clinical guidelines to every unique patient. In addition to the aforementioned drawbacks, the study by Angus (2015) reveals that the randomization process makes patients and physicians uncomfortable. And this is where the possibility of using big data analytics and machine learning models becomes interesting. In line with this, the study by Tahmasebian et al. (2017) states that, with the use of electronic devices and also software for patient databases in healthcare, there is a vast amount of information being generated daily. Nevertheless, these data is valuable only when meaningful information or new insight can be mined from them. When analysed, longitudinal medical data can assist us with understanding what occurred previously, what's going on this moment, or what is probably going to occur later on.

In line with this, utilization of AI systems in healthcare is a growing phenomenon (Babaoğlu, Findik, & Bayrak, 2010). Today, AI-based systems are used being embedded in many medical appliances such as electrocardiography, echography, electroencephalography and ultrasonography. Researches indicate that, there is a pressing need for collaborative machine learning solutions to be able to keep up with the progressively large and multifaceted datasets called Big Data (Holzinger & Jurisica, 2014). It is reasonable to say there are numerous manual tasks in healthcare that can be improved upon by automation and the help of technology. Since the advancement of machine learning and its adaptation in the healthcare industry, developments in electronic medical records have been significant. The value of machine learning in healthcare is its ability to handle tremendous amount of data more than the capacity of a human being, and subsequently reliably decipher analysis of that data into medical know-hows to assist physicians in planning and providing care, eventually instigating improved results, lower costs of care, and improved patient satisfaction. The majority of healthcare systems nowadays use a simple visual representation of historical data for analysis by the physicians. This type of presentation is important however, this becomes overwhelming to the physician with the increasing number of patients and vast amount of data. This necessitates the development of intelligent tools that can extract insight and semantics from this bulk of data for decision making.

2.1.3 Electronic Health Records

An electronic health record (EHR) is a digital equivalent of a patient's paper chart. EHRs are real-time, patient-centred records that make patient data accessible promptly and securely to permissible users (HealthIT.gov, 2020). The collection and preparation of EHR involves combining data that are interrelated with each other. These comprises medical history, diagnoses, medications, treatment plans, demographic data, vaccination dates, allergies, radiology images, and laboratory examination outcomes. One of the key features of an EHR is that, healthcare records can be generated and controlled by only accredited service providers in a digitally. Moreover, they can be used by multiple healthcare units – such as laboratories, specialists, medical imaging facilities, pharmacies, emergency units, schools, and clinics. EHRs present many

advantages for managing modern healthcare related data (Dash et al., 2019). These advantages include:

- Better access to the whole spectrum of the medical history of a patient.
- Timely and efficient treatment of medical conditions.
- The avoidance and exclusion of redundant and additional medical examinations.
- Removal of obscurities triggered by unreadable handwriting.
- Better cooperation amongst medical services providers.
- Improved medical practices with the help of automatic reminders and prompts.
- Faster data retrieval and improved public health surveillance.
- Aid in regulating the raising expenses of health insurance benefits.
- Lessen billing deferrals and disarray.
- Make available a vast amount of clinical data for researches.

2.1.4 Approaches to Causal Inference

Causality plays an important role in monitoring adverse drug events as well as risk factors for diseases with the help of electronic medical records. Causal inference is a process by which we draw a conclusion about a causal connection based on the conditions of the incidence of an effect. It is through causality we can be able to infer the behaviour of a medical treatment. This makes casual inference vital in medicine. In this study, causal inference states to the course of uncovering causal relationships from medical treatment data using machine learning models. Nevertheless, this does not mean that we remove the need for human judgment, but rather help healthcare professionals validate the results and make informed decisions. In medical sciences, understanding the characteristics of diseases and the associated treatments to minimize the effect of a disease is the starting point of causal analysis and assessment (Russo, 2017). However, it is worth mentioning that correlation does not imply causation. In addition to randomized control trials, most early studies applied graphical and probabilistic methods for casual inference in medicine. However, the review by Stern & Price (2020) advocate the development of ML-based medical devices to play vital role to encourage quality healthcare and the establishment of causal inference

techniques. Moreover, the study Rocca & Anjum (2020) supports the application and use of multiple methods to investigate health outcomes.

Kleinberg & Hripcsak (2011), presents graphical models and Granger causality as convenient frameworks for causal inference. Also it is pointed out that more recent approaches such as temporal logic, address some of the limitations observed in graphical models and Granger causality approaches. Besides, it is denoted that it is impossible to fully automate causal inference without human expert interventions from observational data. Numerous factors affect the cure or improvement of a disease (Monleon-Getino & Canela-Soler, 2017). This is why multiple experiments need to be conducted to make meaningful statements regarding cause-effect relationships. Little or no exhaustive study has been conducted using ML models for causal inference. For instance, the study Linden & Yarnold (2016) extends optimal discriminant analysis (ODA) for causal inference by reshaping the treatment-outcome relationship as a supervised learning problem. The study concludes that ODA offers several benefits. Furthermore, the work Pirracchio et al. (2019) presents patient-level ML method for causal inference for decision support for critically ill patients. In situations where randomized control trials are infeasible, other methods such as matching methods, propensity scores, regression discontinuity and the analysis of instrumental variables are considered for causal inference.

Advances in data technology opens new prospects for more targeted queries concerning patient treatment for better patient-centred outcomes. The use of machine learning models for causal inference has not been prevalent. It is also vital to emphasize that, given the risks involved, the application of machine learning models for causal inference in healthcare and biomedicine should not be considered as a sole solution by itself (Rose & Rizopoulos, 2019). In this paper, we explore the use of principal component analysis as a principal method for inferring causal relationships from longitudinal clinical laboratory data for critically ill patients. PCA is applied to capture daily changes from medical laboratory data and present results along with daily prescriptions so that cause-effect relationships can be established by human experts in the medical field.

2.2 Related Works

In recent years considerable efforts has been spent by researchers in solving the problems that exist in the healthcare industry and their automation thereof (Li et al., 2014; Luo, Szolovits, Dighe, & Baron, 2016; Pham, Tran, Phung, & Venkatesh, 2017; Werdiningsih, Hendradi, Nuqoba, & Ana, 2019; Zhang, Ren, Huang, Cheng, & Hu, 2019). All supervised, unsupervised, and probabilistic methods have been tried for their application and suitability in the healthcare sector. In this section, various studies in the field of machine learning for healthcare with special attention to causal inference at the end are presented. The order of presentation of the studies in this section is selected based on the similarities and relatedness of the study to this thesis work. All of the proposed approaches have their own merits, and preference for one over another depending on the settings they are developed in and the type of application they are developed for. The summary of reviewed literatures related to our work in some fashion or provide input to our work are presented hereafter.

Tahmasebian et al. (2017), puts forward general suggestions that should be kept in mind while designing machine learning models for medical data analysis with special attention to chronic kidney disease. It is suggested that the data mining results should be compared with expert system outputs. This suggestion is key as well as vital and can be applied for designing machine learning models for causal inference. The researchers applied association rule mining and suggested offering the resultant rules to postulate the role of CKD factors. Moreover, the authors propose the use of other data mining methods such as artificial neural networks and fuzzy models for the prediction of the status of a CKD patients. On the other hand, the work by Alzamora, Nguyen, Simoff, & Catchpoole (2012) describes a 3D interactive visualization of biomedical information of a large scale medical data. The model developed can be used either to prove an existing hypothesis or derive new scientific insights for childhood cancer patients. The work also argues that the explosion of medical data size requires techniques for uncovering less visible knowledge and extract unseen patterns. Also it strongly agrees that any medical data visualization tool, in addition to a simplified abstract view, should also provide detailed, interpretable, and navigable information at a particular focus point. Moreover, the study explains the importance

of domain knowledge and the use of dimensionality reduction techniques in large medical data analysis. Similarity space on biomedical data was applied, to calculate the (dis)similarity between patients. By applying this 3D visualization the physician has the ability to view the data's position from any preferable angle or distance, properties of patient populations, and identification of the particular patient.

Wang, Li, Ma, Huang, & Li (2014), also prove that due to the growing size of electronic medical records, there is a strong urge for a new set of instruments to extract knowledge of interest from this wealth of data. The work also emphasizes that the use of knowledge extraction from a large amount of patient data improves patient satisfaction and uncover new insights. In this research, a web-based visual mining system that supports explorative analysis of high dimensional categorical EMR for chronic kidney disease is developed. The proposed method uses Ochiai coefficients (a variety of cosine similarity) to compute the similarity between patients based on seventeen CKD factors selected in consultation with medical experts. A system for interactive visualization and exploration of patient progress overtime for decision making using hierarchical clustering and tracking graph is presented in (Widanagamaachchi, Livnat, Bremer, Duvall, & Pascucci, 2017). This type of exploring and viewing is important, however, it won't be enough to uncover latent information. In addition, the authors in Babič, Vadovský, Muchová, Paralič, & Majnarić (2017), also agree that EMRs have a vast potential to help medical practitioners with simple understandable results by applying suitable machine learning algorithms. To prove their hypothesis, the researchers performed experimental studies on data collected from a single hospital on mild cognitive impairments (MCI). The study by Mohamadlou et al. (2018), also presents the potential of using machine learning methods for predicting acute kidney injuries (AKI) for better assessment of existing and novel interventions for providing ultimate treatment. The proposed method used boosted ensembles of decision trees produced using the python XGBoost package. In addition to the application of machine learning for medical data exploration and visualization, other techniques such as predictive and clustering models have been explored.

Babaoğlu et al. (2010) explored principal component analysis as a dimensionality reduction technique before applying support vector machine (SVM) classification. An original dimension of 23 variables is reduced to 18 variables before applying SVM. The study explored the effectiveness of applying PCA before SVM on a dataset of 480 patients and compared the results against ordinary SVM classification without applying PCA. The work concluded that PCA shows effective usability by increasing accuracy. Moreover, it also decreases training error, helps minimize training and test time compared to the application of direct SVM on the dataset. The study Kara & Dirgenali (2007) used a dataset of 177 patients (82 atherosclerosis patients and 95 healthy volunteers) to explore the effectiveness of applying PCA before applying artificial neural networks (ANN). The study concludes that the application of PCA before applying ANN improves the performance of the ANN classifier. Moreover, the authors suggest the use of this approach as part of the ultrasonic Doppler device for non-invasive, inexpensive decision-making tool.

Early studies use generic vital signs in order to compute early warning scores to predict patient mortality for deteriorating patients (Mathukia, Fan, Vadyak, Biege, & Krishnamurthy, 2015; Prytherch, Smith, Schmidt, & Featherstone, 2010; Smith, Prytherch, Meredith, Schmidt, & Featherstone, 2013). These types of systems are solely based on biological factors of the patient's vital-signs such as heart rate, breathing rate, and systolic blood pressure. Moreover, these systems use the sum of assigned points of these vital signs from a subjectively fixed normal range to identify patients that are worsening but also can have their result altered by a well-timed medical intervention. However, recent advancements in machine learning and EMR paves the way for the development of intelligent tools for medical data analysis to help in decision making. In line with this, numerous predictive models have been presented. Clifton, Clifton, Pimentel, Watkinson, & Tarassenko (2014), with the aim of providing early warning of serious physiological deterioration, explored various machine learning methods for construing large amounts of continuously obtained, multivariate biological data using wearable patient monitors. The work also states that the job of doctors and healthcare workers hinders them from scrutinizing large amounts of patient data with a high degree of precision. The work conducted a retrospective

analysis of the data using four different types of novelty detection machine learning models. Results show that the approaches provide an effective early warning system compared to ordinary early warning systems (EWS), which uses a predefined threshold to activate a warning as part of the wearable devices. The system was also able to capture warning incidents missed by the built-in EWS. The study outlines that applying such type of machine learning models as part of clinical practice, serves as a vital tool such that preventive clinical actions may be taken to enhance patient results. Moreover, the application of predictive models in healthcare has become vital. In line with this, other highly investigated categories of machine learning models include disease-specific prediction models among others.

Ye et al. (2019) they applied a tree-based random forest algorithm on data collected from EMRs to forecast patients with great risk of intra-hospital death and achieved a c-statistics of 0.884. On the other hand, the study by Cai et al. (2016) presents a predictive paradigm for real-time predictions of the length of stay, death, and repatriation for inpatients from electronic health records. The study employed Bayesian network method to assess the likelihood of a subject being in one of the following states; at home, in the hospital or dead, and reached a mean daily accuracy of 80% and an AUROC of 0.82. The work in Babič et al. (2017), performed experimental studies on data collected on mild cognitive impairments. The work conducted a comparative study on methods such as decision trees, different statistical t-tests to predict the chance of a patient being positive or negative. The study claims the proposed approach provided satisfactory and superior results relative to traditional methods. Mohamadlou et al. (2018) also demonstrates the use of machine learning methods for predicting acute kidney injuries from EHR for better assessment of existing and novel interventions to provide vital treatment. On the other hand, the authors in Kara & Dirgenali (2007) suggest the use of their approach as part of medical devices for non-invasive, inexpensive decision-making tool. With respect to the application of machine learning for causal inference in healthcare, little or only a considerable amount of study can be found on the internet.

Numerous challenges remain in the advancement and improvement of machine learning tools in healthcare. Moreover, little or no observational study has been

conducted to design ML tools to establish cause-effect relationships from clinical laboratory data. The study by Stern & Price (2020), states that this is due to the generalizability of ML model results remain questionable mainly in circumstances where ML is unable to reveal causality due to the nature of the algorithms, ascertain predictive patterns instead of causal relationships. Moreover, an ML model developed in one hospital setting might not be appropriate in different hospital settings except causal inference means were applied in the development. The study Rose & Rizopoulos (2019) points out that multiple disciplines have benefitted from the advancement of machine learning, however, the application of these tools has not been widespread in some areas such as causal inference. This is because sample sizes in RCT pose major limitations. However, if appropriate control of confounding and other issues are handled, observational data may reveal hidden insights. Moreover, the study also advocates using the mixture of both RCT and observational data for establishing causal inference.

Causality plays an important role in monitoring adverse drug events as well as risk factors for diseases with the help of electronic medical records. A review study by Kleinberg & Hripcsak (2011), presents graphical models and Granger causality as convenient frameworks for causal inference. The study also points out that more recent approaches such as temporal logic methods address some of the limitations of the above models. Also, the study denotes that we cannot fully automate causal inference from observational data without human involvement in the process. Multiple experiments need to be conducted to make meaningful statements regarding cause-effect relationships. The study Linden & Yarnold (2016) extends optimal discriminant analysis for causal inference by converting the treatment-outcome relationship into a classification problem. The study concludes that ODA offers several benefits. Furthermore, the work Pirracchio et al. (2019) presents patient-level ML method for causal inference for decision support for critically ill patients. Advances in data technology opens new prospects for more directed queries vis-à-vis patient treatment for better patient-centred outcomes.

From recognizing disease threats and applying proactive steps, to performing diagnosis and patient-tailored medicine, big data, the application of high-performance

computing, and machine learning models are turning out to be vital for precision medicine (Prosperi et al., 2020). Precision medicine includes deliberating interventions among other tasks. Alternative scenarios and accurate specification of cause-effect relationship is vital in the application of clinical predictive models. Moreover, robust assumptions and prior domain knowledge are important for causal inference. The study by Richens, Lee, & Johri (2020) argues that machine learning has the capacity to transform clinical decision making and diagnosis. Yet, it mentions that most existing machine learning models are purely associative, they only identify diseases strongly correlated with patient symptoms which may result in erroneous diagnosis. To solve this issue of machine learning application in the field, the study reformulated diagnosis as a counterfactual extrapolation task and developed counterfactual diagnostic algorithm. That is in order to compute the probability that a disease is instigating the symptoms, counterfactual inference is applied. Finally, results were compared and the authors claim that the approach achieved expert clinical accuracy. Furthermore, the study stresses that causal reasoning is paramount for applying machine learning for healthcare diagnosis. Finally, in this study we explore the use of principal component analysis as a principal method for establishing cause-effect relationships from longitudinal clinical laboratory data in intensive care unit settings. PCA is applied to capture daily changes from medical laboratory data and present results along with daily prescriptions so that cause-effect relationships can be established by healthcare practitioners. Moreover, causal impact analysis of the changes is also conducted and statistical summary is also provided.

2.3 Summary

Spurred by the advancement and success of machine learning and data analysis algorithms in other fields of study, some analytic companies such as Google Health/DeepMind and IBM Watson Health, are turning their attention to problems in healthcare. Moreover, the healthcare industry is starting to adopt machine learning and data analysis tools in the effort to push the boundaries. This effort primarily involves data analysis using the vast amount of healthcare data. It is believed that, healthcare is an area that is extremely appropriate for the development of artificial intelligence based tools (Sarkar, 2020). Moreover AI and machine learning tools are believed to

add extra value to the healthcare industry. They are anticipated to enhance the quality and different aspects of the sector. Furthermore, researchers agree that machine learning is “making healthcare smarter” and has proved really life-impacting prospect in healthcare – predominantly in the field of medical diagnosis (Gharagyozyan, 2019). In the effort to come up with personalized or decision medicine vast amount of machine learning research is conducted and are being conducted by various researchers in the field. Yet, researches Deo (2015) point out that, it is worth mentioning that these vast amount of medical datasets and adequate learning algorithms have been available for many years. However, even with thousands of researches conducted so far in the field, very few have contributed meaningfully to clinical care.

Moreover, many researches indicate that the development of machine learning systems in healthcare are paramount both for the healthcare sector as well as for patients. However, most of the studies conducted so far focus on medical data classification and prediction only, focused on specific internal medicine subspecialties, and based on small sized datasets. This indicates that there are problem domains and subspecialties that need to be investigated and addressed. This brings the idea of developing a more full-fledged and robust machine-learning tool for medical treatment data analysis. Also the majorıty of the electronic healthcare systems nowadays use simple visual representation of historical data for analysis by the physicians. Limitation in the effective presentation of EMR to the user necessitates the development of intelligent tools that can extract insight from these bulk of data for decision making by the physicians. In this study, the main goal will be to use machine learning models and develop a tool, which can be used by the physicians as part of the daily routine clinical practice. In terms of the results to be obtained, the system is aimed as a data analysis tool and methods that can be directed by the physician instead of being an autonomous system independent of the physician. Multiple options will be generated and presented for the physician to choose and use effectively in various circumstances, and the physician will be able to decide or choose a certain type of treatment or therapeutic based up on the options and results are presented.

CHAPTER THREE

PROPOSED METHODS

One of the main purpose of this thesis is to find an appropriate machine learning model among the multitude of models for identifying cause-effect relationships from medical treatment data. The motivation behind this research undertaking is to develop a robust machine learning tool that could be applied in the healthcare sector and be used as supportive tool in daily clinical practices. To achieve this goal, multiple models have been researched with special attention to principal component analysis and predictive models such as Gaussian process regression, Support vector regression and Long-short term memory.

3.1 Design

Causal analysis and causal inference is an emerging area of biostatics (Yazdani & Boerwinkle, 2015). Causal inference is a process by which we draw a conclusion about a causal connection based on the conditions of the incidence of an effect. From a medical treatment perspective, we infer the causal effect of an intervention as the probable product of the treatment. In this chapter, a detail description of design issues and proposed causal inference technique are presented.

3.1.1 Approaches and Techniques

So far many causal inference researches have been done and different approaches have been used and researched, where the prominent ones include randomized control trials, probabilistic approaches, propensity scores, and others. Yet, only few machine learning studies and literatures can be found on the internet. All of the proposed methods have their own merits, and preference for one over another. In this thesis work, principal component analysis is examined as a non-autonomous primary method for establishing cause-effect relationships from medical treatment data. It is non-autonomous in the sense that medical expert input is vital to actually deduce the cause-effect relationship from the observed changes captured by the PCA analysis. Moreover, three prominent predictive models are implemented to predict future values

for each laboratory tests to anticipate future events which includes GPR, SVR, and LSTM.

Principal component analysis is the most commonly used linear dimensionality reduction technique, it creates a linear mapping of the data to a lower-dimensional space in a way that the variance of the data in the low-dimensional representation is maximized (Tomaszewski, Hipp, Tangrea, & Madabhushi, 2014). PCA creates new variables which are the linear combinations of the original variables, these new variables are orthogonal to each other (i.e. their correlation is equal to zero). These new variables are called principal components (PC). PCA can be seen as a rotation of the original space to find a more appropriate axis to convey the variability in the data. The covariance matrix of the data is created and the eigenvectors on this matrix are calculated (Jaadi, 2021). The eigenvectors that relate to the largest eigenvalues (the principal components) will be used to recreate a large portion of the variance of the data. The principal components are acquired so that the first principal component represents the majority of the possible variation of the original data after which each succeeding component has the next highest possible variance. The second principal component does its best to capture the variance in the data that is not captured by the first principal component (orthogonality). Once the new variables are created, we can choose the main ones (top principal components). The threshold is up-to-us and relies upon how much variance we need to keep. Figure 3.1 below shows a snapshot of a data and its first and second principal components axes.

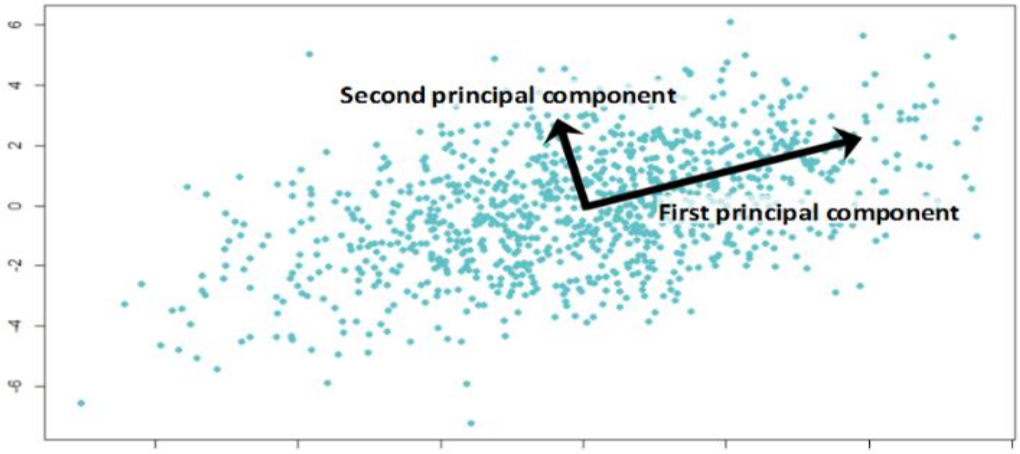


Figure 3.1 Principal Component Analysis (Minasyan, 2016)

Generally, principal component analysis involves the following major tasks:

- Standardize/center the scale of the data as per equation 1.1 in Section 1.6.2.
- Construct the covariance matrix of the data as in equation 3.1.

$$C = \frac{1}{N}\sum_{i} X_i X_i^T \tag{3.1}$$

- Compute the eigenvalues and correspondıng eigenvectors (Eigen decomposition) as in equation 3.2.

$\lambda v = Cv$, which can be written as,

$$\lambda X_i^T v = X_i^T C v \quad \forall i \in [1, N] \tag{3.2}$$

Where,

v is a non-zero vector of dimension N

λ is a scalar, termed as the eigenvalue corresponding to v

C is the eigenvector corresponding to v

- Sort the eigenvectors in descending order.
- Build a projection matrix W, where the k eigenvectors selected will be stored.
- Transform the original data onto the PCA space.

The new columns of this new, transformed space are the Principal Components we will use instead of our original variables. Those, as referenced above, are created in a way with the end goal that they store however much information as could be expected.

Advantage:

- Reduces complexity of data.
- Identifies most important features.

Disadvantage:

- Unless proper care is taken while choosing the number of eigenvalues to keep, it might prompt some measure of data loss.
- PCA will in general discover linear relationships between variables, which is on occasions unwanted.

- PCA fails in situations where mean and covariance are not able to represent the datasets adequately.
- We may not know the number of principal components to keep.

We hypothesize that PCA can be able to capture the changes that may happen due to the medical treatment or interventions on a data collected on a daily basis. PCA is commonly used for linear dimensionality reduction and exploratory purposes through variance maximization. The intuition behind PCA is to use a different coordinate system that is influenced by the observations is the data. The axes are placed in the direction of highest variance of the data to maximize the variance along that direction (Witten & Frank, 2006). This intuition can help us determine and show the patient's laboratory result exhibiting the highest variances (even though those changes might still be within the normal reference ranges) after having a certain medical intervention. Moreover, if applied on longitudinal data and along with medical treatment information, it may help us explore patient progress from time T_1 to T_2. In addition, this can also be used to track the changes which are the outcomes of the treatment applied and help us establish cause-effect relationships. So that an appropriate treatment or therapy can be prescribed or further diagnosis can be advised. In addition, each principal component is a linear product of the original individual feature. This can be used to see the effect and contribution of each laboratory test in that direction. This can help track the variable that changed and contributed the most to the PC which may have happened due to the introduction of the treatment. The succeeding PCs try to capture the next highest variances left out by the preceding PCs, though there might be redundancies. This may show epiphenomenon or parallel medical events or conditions happening. Most studies use PCA as pre-processing for classification tasks. However, based on the intuition how PCA works, we believe that it can be used as part of a tool for early warning of serous medical conditions by showing variables with the highest variances. To evaluate our primary hypothesis, a retrospective analysis was performed using PCA to monitor patient progress.

Moreover, as presented in the introduction to the chapter, three predictive models namely; GPR, SVR, and LSTM were also developed. The intention behind this is to provide a robust full-fledged tool to be used in daily clinical practices. These models

provide the user with predictive capability of future values for selected laboratory tests. This in turn will assist the user anticipate certain medical events and take measures proactively before they even occur. Three models are implemented so that the user will have the option to choose from based on the root mean square error (RMSE) of the models that is presented along with the prediction results. The following is a high-level descriptions of the aforementioned models.

A) Support Vector Regressions

A Support Vector Machine (SVM) is a classifier formally defined by a separating hyperplane (Sathyanarayana & Amarappa, 2014). Provided we have labelled training data, the algorithm outputs an ideal hyperplane which categorizes new samples. In 2-dimensional space this hyperplane is a line dividing a plane in two parts where in each class lay in either side. Also SVM can also be used as a regression method, maintaining all the main structures that typify the algorithm (maximal margin). In the Support Vector Regression, a margin of tolerance (epsilon) is set in approximation. In SVM the margin of tolerance is computed from the problem. The goal behind the algorithm is to decrease error, individualizing the hyperplane which capitalize on the margin, keeping in mind that part of the error is tolerated (Sayad, 2010).

One of the main characteristics of support vector regression is the use of kernels such as linear, radial basis function (RBF) and polynomial kernels. SVR is a supervised learning scheme and an effective method for real-value function estimation, however, it is less popular than SVM. SVR learns from training data by means of a symmetrical loss function, which uniformly penalizes high and low wrong predictions. A flexible plane/sphere of nominal radius is constructed equally nearby the projected function, such that the absolute values of errors less than a defined threshold are overlooked both above and below the estimate. Points outside the plane/sphere are penalized, however, there is no penalty for those within the tube, either above or below the function. The dimensionality of the input space does not affect the computational complexity of the model which is one of the merits of the SVR. Besides, the generalization capability of the approach with high prediction accuracy is outstanding (Awad, & Khanna, 2015).

B) Gaussian Process Regressions

Gaussian Processes (GP) are broad supervised learning technique that aims to solve regression and probabilistic classification tasks. Gaussian processes broaden multivariate Gaussian distributions to infinite dimension. A Gaussian process creates data situated throughout some domain such that any limited subset of the range adheres to a multivariate Gaussian distribution (Ebden, 2008). Gaussian process regression is a powerful, non-parametric Bayesian approach towards regression problems. Instead of asserting f(x) correlates to some specific regression models (e.g. $f(x) = mx + c$) as in linear regression, a Gaussian process can represent f(x) implicitly, but thoroughly, by allowing the data 'speak' more plainly for themselves. GPR is a form of supervised learning, but the training data are harnessed in a subtler way. A sample GPR prediction result is shown in Figure 3.2. According to Pedregosa et al. (2011a), the following are the advantages and disadvantages of this method.

The advantages include:

- For regular kernels the observations are interpolated by the prediction.
- The prediction is probabilistic with the goal that one can calculate realistic confidence intervals (CI) and decide depending on those on the off chance that one should refit the forecast in some locale of interest.
- Versatile: diverse and combinations of kernels can be applied.

The disadvantages include:

- The entire samples/features information are utilized for prediction (they are not sparse).
- Poor performance for high dimensional space with more number of variables in the data.

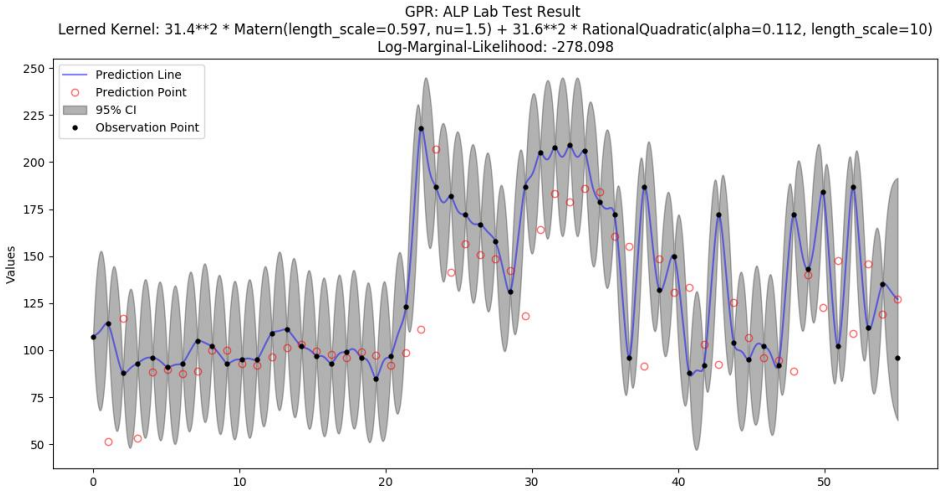


Figure 3.2 Gaussian process regression

C) Long-Short Term Memory

LSTM networks are specific category of recurrent neural network (RNN) skilled with learning order of dependence for sequence prediction problems. They function admirably on a huge assortment of issues and are currently commonly used. LSTMs are clearly designed to avoid the long-term memory dependency problem. Recollecting information for long periods is essentially their default behaviour (Olah, 2015). The problem of learning to keep information for a long period of time by recurrent backpropagation requires a very long time. The work in Hochreiter & Schmidhuber (1997) addressed this issue with the introduction of long-short term memory. In recent days, due to their success for a wide range of applications, LSTM models have garnered a lot of coverage. LSTM networks handle the exploding/vanishing gradient problem very well, this is one of the main reasons that makes them more successful. A higher level depiction of an LSTM network is presented in Figure 3.3. Generally, the gating mechanisms in LSTM networks which solves the “short-term memory” makes them different from ordinary recurrent neural networks. With this mechanism they are capable of preserving memory for a longer period which is an important feature for most applications such as time-series predictions, natural language processing (NLP) and sequence predictions.

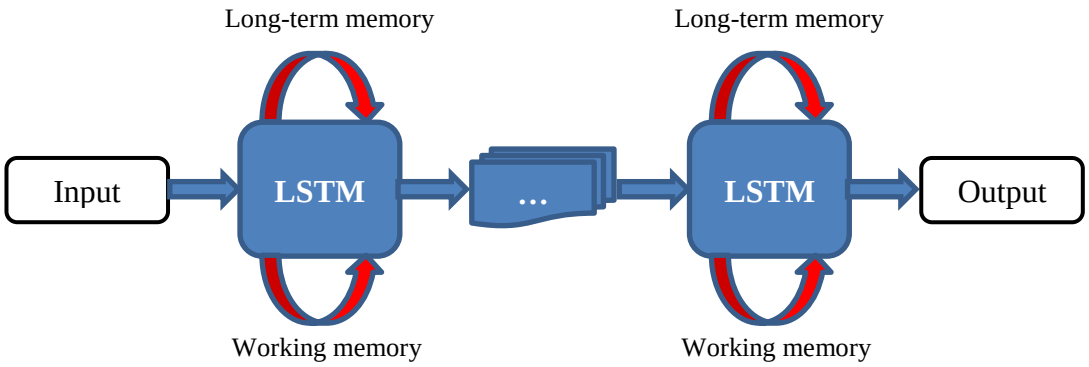


Figure 3.3 General architecture of LSTM networks

3.1.2 Design Goals

The main aim of this thesis work is to investigate machine learning models for systematic pattern extraction and analysis of longitudinal critically ill patients' medical data to establish cause-effect relationships. This approach can be used as part of a tool for early warning of life-threatening medical conditions as a daily routine clinical practices in intensive care unit settings.

3.1.3 Model Architecture

The overall architecture of the machine learning models for this thesis work are depicted in Figure 3.4 – 3.7 and described hereafter. A higher level illustration of the overall machine learning model design and implementation flow is depicted in Figure 3.4 below. As shown in the figure, at the first stage, the input dataset passes through a data preparation process. This step involves data extraction, cleaning and pre-processing tasks. This produces a feature vector ready to be used as an input for the implemented machine learning methods. After applying the implemented machine learning models, the results of the principal component analysis is subjected to permutation, bootstrap and causal impact analysis test, while the results of the predictive models passes through a testing phase. Finally, the results of the performance analysis step are tabulated and plotted for presentation.

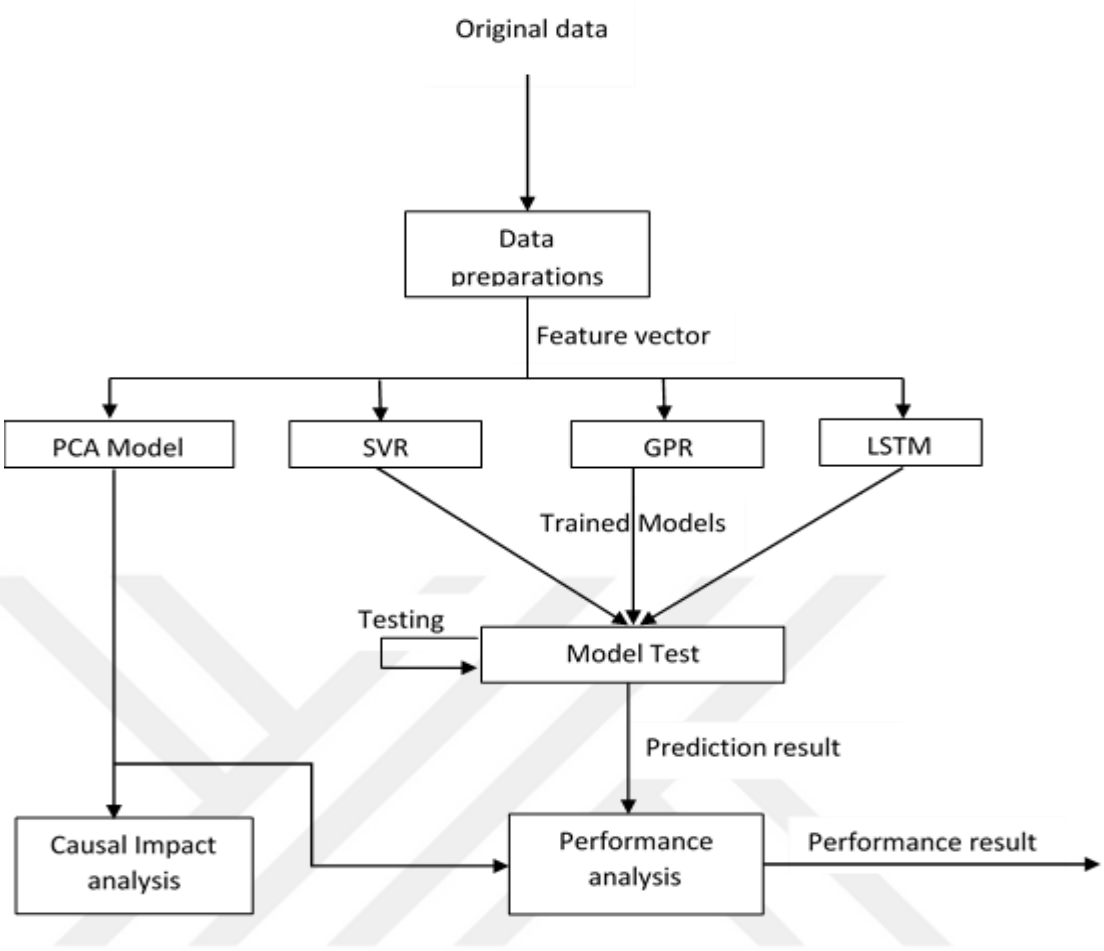


Figure 3.4 Machine learning model development flow chart

The following steps depict the overall stepwise analysis process depicted in Figure 3.4.

1. Read csv data.
2. Remove non-numeric and single value variables.
3. Impute missing data with *most frequent* value method.
4. Group data based on laboratory test date.
5. Get selection of analysis option from the user.
6. Get laboratory test dates from the dataset.
7. Loop through the dataset depending on the analysis option selected:
    a. Extract parts of the dataset pertinent to a specific date and the chosen analysis option.
    b. Perform PCA and predictive analysis on the extracted data.

c. Plot and return results.

8. Apply predictive models.
9. Perform performance analysis for both models.
10. Tabulate, plot and present results.

Figure 3.5 shows the overall machine learning models design and implementation process. The original dataset is used as input to the feature extractor module. The feature extraction process follows a two-step process. In the first step, the original .csv file is loaded to a Postgress database in which multiple queries are executed to extract the required data. In the second step, the results of the first step from Postgress querying is used as an input to a python module we prepared (see Figure 1.1). In this step, the input data is formatted and reshaped into an appropriate input format for PCA. The result of the feature extraction process is the production of a feature vector, which is used as an input for principal component analysis. After applying data pre-processing on the feature vector, the result is used both for PCA and predictive models implementations. Afterwards, performance analysis is conducted. The performance analysis and evaluation process for the principal component analysis is shown in Figure 3.6.

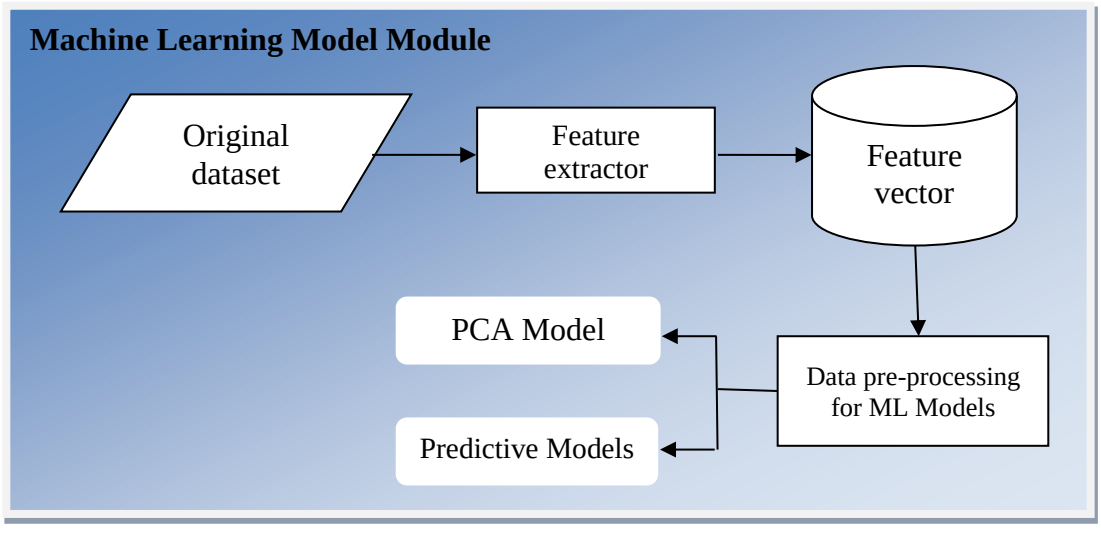


Figure 3.5 General machine learning model architecture

As depicted in Figure 3.6, the original dataset is given to the feature extractor to prepare a feature vector and make ready for the machine learning models. After

applying PCA, the results of the principal component analysis is subjected to permutation & bootstrap tests and causal impact analysis for model validity and stability tests. Causal impact analysis is conducted using CausalImpact R package (Brodersen, Gallusser, Koehler, Remy, & Scott, 2015; Brodersen & Hauser, 2015). The package constructs a Bayesian structural time-series model to estimate the causal impact of a treatment or an intervention based on a control group. The results of these tests is used to compute a significance level and compare the result with a predetermined significance level (alpha). This in turn can be used to make a decision on whether to accept or reject our hypothesis. This processes are elaborated in Section 4.4.1, 4.4.2 and 4.4.3 in a greater detail. Furthermore, the performance analysis and evaluation process for the predictive models is illustrated in Figure 3.7.

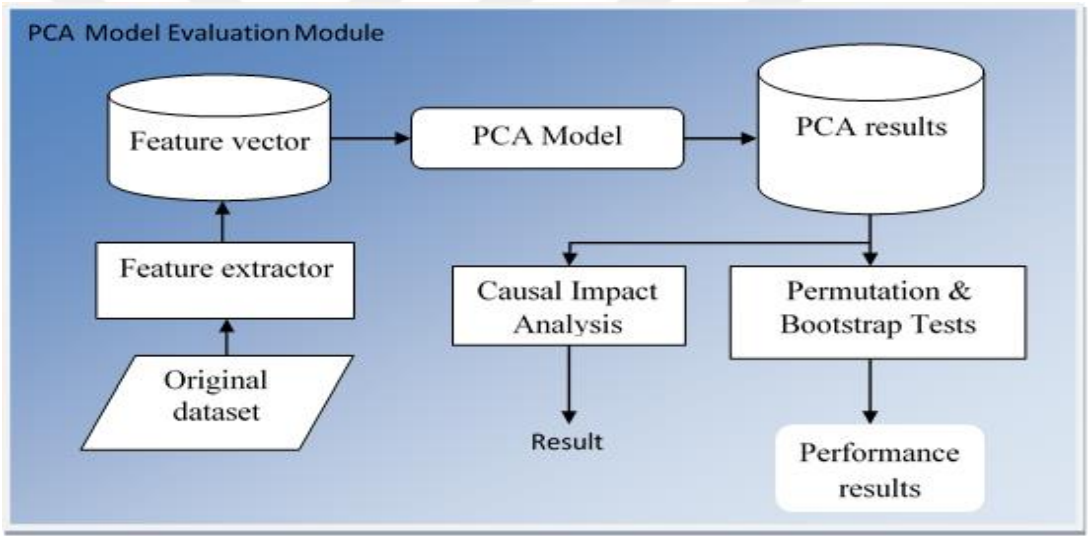


Figure 3.6 PCA model validity and stability test cycle

As depicted in Figure 3.7, the original dataset is given to the feature extractor to prepare a feature vector and make ready for the predictive models. Afterwards, the results of the three predictive models is subjected to performance analysis. For this task, root mean squared error is used for performance measurement. The results of this measurement are presented to the user along with the prediction results of these models. This is performed with the intention that the user will be able to select the best prediction with the lowest RMSE. This will give the user the flexibility of choosing an acceptable results depending on various circumstances.

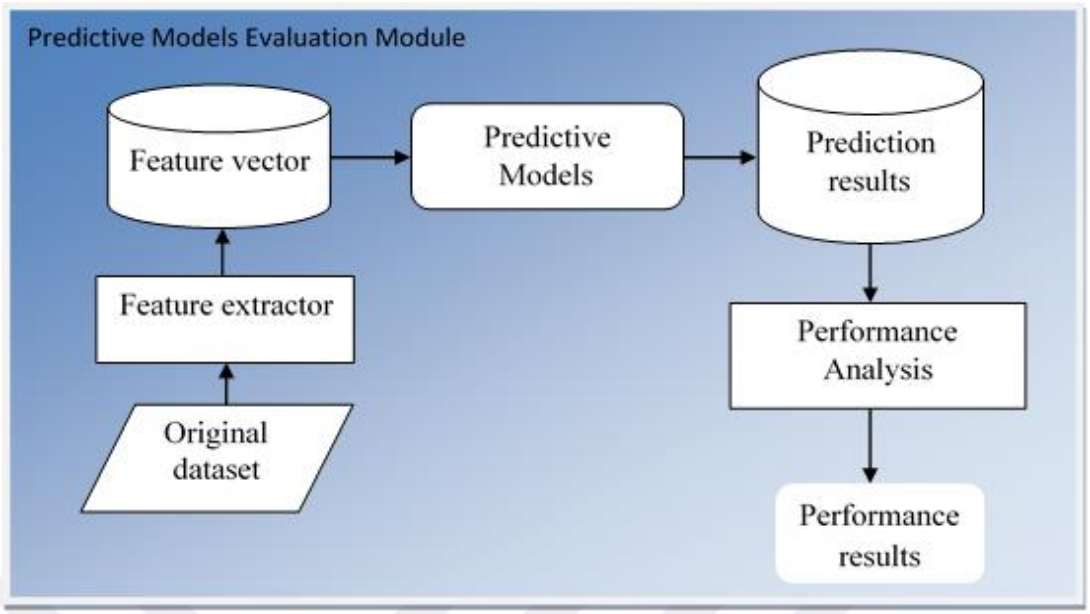


Figure 3.7 Predictive Models training and testing cycle

Overall, a principal component analysis method and three predictive models namely; support vector regression, Gaussian process regression and long-short term memory were implemented. The validity, stability and performance of the proposed models were tested using appropriate and applicable methods. For instance, model stability and validity is performed with both permutation and bootstrap testing for PCA since it is not possible to apply other performance measures. Moreover, the widely accepted pre-trained causal impact analysis model developed by google was used for further model validation. Finally, a conclusion about these models and the overall significance and application of the thesis work is derived based on the result of these tests. The objective here is to present a robust and full-fledge tool for the healthcare professionals that can assist them in their daily clinical practices with more flexibility and agility.

Causal inference in medical sciences is the process by which a conclusion is drawn about a causal link based on the circumstances of the manifestation of an outcome from a treatment. This problem can possibly be tackled using different approaches. One of such approaches is machine learning models among others. As far as establishing cause-effect relationships between the medical treatments and their

outcomes with possibly an interpretable results and a good and acceptable performance, there is a need to make an empirical selection and amalgamation of one or more than one approach. This can help as take an advantage from the underlying approaches thereby remedying the shortcomings of using only a single model. Moreover, it is worth mentioning that this models are not meant in no way to replace medical field experts. Rather they are intended to help and support those experts. Therefore, a high level of input and cooperation of medical experts is required and is vital to the success and successful implementation and deployment of this types of models.

3.2 Implementation

Here, implementation details of the proposed machine learning models and dataset preparation is presented. To begin with, python 3.6 to design and implement the machine learning models and Postgress 4 for original data acquisition and querying are used in the entire implementation of the models. The rationale behind the choice of these tools is, they are suitable for the selected problem space and application area. Python is well-suited and the most commonly used development environment for machine learning models. It can be applied for a wide range of machine learning applications, including signal and image processing, supervised and unsupervised learning, and deep learning. Moreover, there are numerous free libraries the can be plugged and used to design and implement the multitude of machine learning models that extend the python environment to solve various classes of problems. Python libraries such as Scikit-learn, Pandas, Numpy, Bokeh and Keras are widely used for this work. The second tool used is Postgress, a free, powerful, open source object-relational database management system. This tool is mainly used for accessing the original csv files, create relationships between related tables and finally query the database for creating the required dataset. For causal impact analysis of changes captured by the PCA model, CausalImpact R package developed by Google is used. In the following sections, the details starting from the dataset preparation for the machine learning models to the implementation of the models are discussed.

3.2.1 Dataset Preparation

Generally, the data preparation goes through a two-step process. In the first step, the original .csv file is loaded into Postgress SQL and the required patient medical history and prescription data is queried and extracted. The result is then saved to a csv file ready for the second step of python data pre-processing. The processes involved in the first step are described in Section 1.6.2 of this document. In the second step, the results from step one are used as an input and pass through various steps of data pre-processing tasks. This tasks mainly include imputing missing values, removing insignificant variables and standardizing the data. Then the result is used as an input for both the PCA and predictive models. The analysis part provides four different PCA analysis strategies for comparison as well as to provide multiple option for the user (when implemented as a tool). The options include; daily based separate analysis, daily based incremental analysis, sample based sliding window analysis and sample based two time frames range analysis. Depending on the option selected, we extract and loop through data, perform PCA and predictive analysis and finally plot and tabulate the results. The following steps show the data preparation, principal component analysis and predictive model workflow as depicted in Figure 3.8.

1. Acquire extracted data from Postgress SQL.
2. Remove non-numeric and single value (no variance) variables.
3. Impute missing values.
4. Standardize the data to have zero mean and unit standard deviation.
5. Group data by laboratory test date.
6. Choose analysis option: Daily based Separate analysis, Daily based Incremental analysis, Sample based sliding window analysis, and Sample based two time frames range analysis
7. Apply PCA and return principal components that explain up to 99% the total variance in the data.
8. Prepare and plot scree plots (individual and cumulative explained variance).
9. Prepare and plot principal component pie charts.
10. Prepare and plot 2D/3D projection plots.
11. Prepare and plot top n principal components pie charts.

12. Apply predictive models.
13. Perform performance analysis for both models.

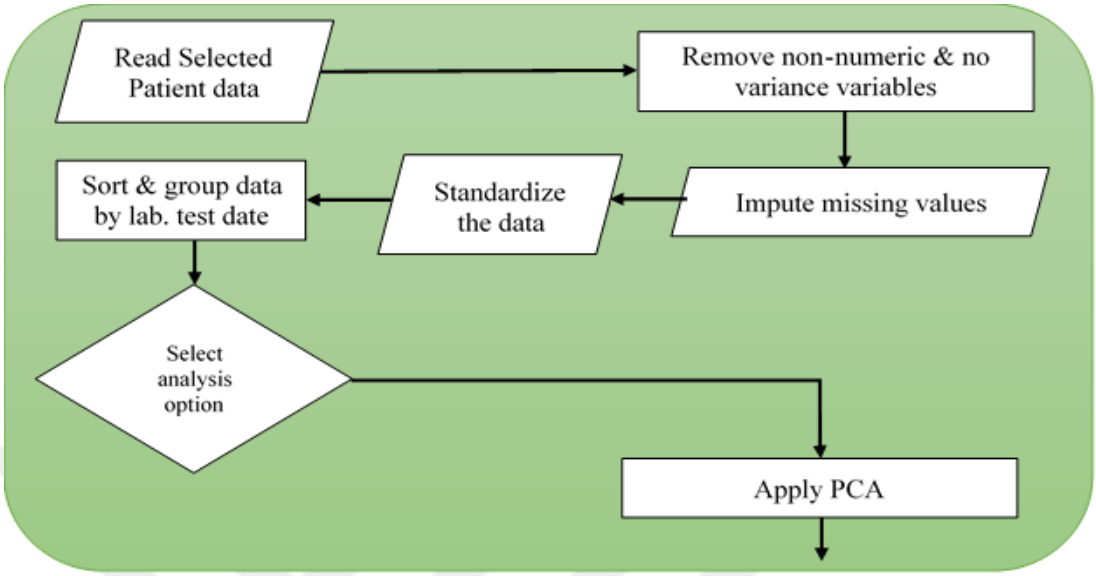


Figure 3.8 PCA data preparation process flow

Four different analysis strategies were implemented for principal component analysis for comparison as well as to provide multiple option for the user. In option one, the laboratory test results taken for each day and having at least four observations are separately fed into the algorithm and results were produced. Whereas in option two, laboratory test data collected for each day is analysed incrementally by adding data collected for each day on top of the data from the previous day/s, starting from day one. In option three, the user specifies a specific window size which is the number of samples or days to be analysed at a time, which should be greater than or equal to one and less than or equal to the total number of samples/dates in the dataset. Samples will be extracted by a sliding window of the specified size from the first to the last sample with no overlapping. Finally in option four, the user has the option to specify specific portion or section of the whole dataset to be analysed. For example by selecting the samples between the 5^{th} and 20^{th} observation (between day 5 and 20). Depending on the option selected, we extract and loop through the data, perform principal component analysis and plot and tabulate results. The intention behind implementing all the four methods is that the user can have flexibility to check patient progress from different perspectives while using this machine learning model as a

supporting tool. Due to the fact that the analysis is patient based missing values are imputed using most-frequent strategy along each column. This is followed by data standardization. Then the records are grouped based on the date the laboratory test is taken. A baseline of at least four observation should be available on a specific date to start the analysis. Otherwise those observations are merged with the next day test results. This process is repeated iteratively until we achieve the minimum amount of observations for analysis. This is due to the nature of the PCA algorithm that the more the number of samples, the better and reliable the PCA result is. Finally, the relevant patient data were fed into the model and principal components were computed and visualized.

3.2.2 Model Implementation

There are many variables that must be considered when interpreting the results of any laboratory test. In general health practice, health practitioners use normal reference ranges as guidelines of what is normal or abnormal. However, we believe that even if minor fluctuations of successive measurements are within normal reference ranges, when fused with other related laboratory tests these fluctuations may provide significant information on the affected areas of the body. This in turn can be used to make informed decisions. In view of these facts, an extensive search for an optimal machine learning model with a simple architecture and reasonably acceptable performance is explored in this thesis work. Principal component analysis and three types of predictive models are implemented in this research work. The working and prediction performance of each of these models is described in Chapter 4 of this document.

In the PCA analysis, principal components that capture 99% of the overall variance in the data are returned for each day. For each day, a varying number of principal components are captured. Negative values indicate a negative associations whereas positive value indicate a positive associations of the original variable for a principal component. The first PC explains the highest variance in the data whereas the second and successive PCs try to explain the variances missed by the preceding PC. The PCs are uncorrelated with the preceding PC, i.e. the correlation between these PCs is zero.

All succeeding PCs follow a similar fashion and try to capture the remaining variation without being correlated with the previous PC.

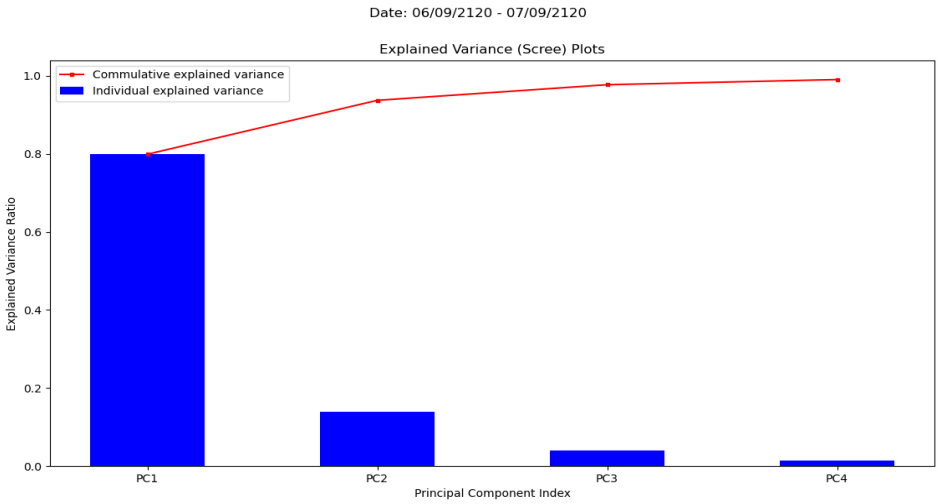


Figure 3.9 Principal component scree plot

In Figure 3.9, the blue bars denote individual explained variance whereas the red line shows the cumulative explained variance by the principal components. The first 4 principal components were able to explain 99% variance in the dataset in which the dataset had 56 original variables. The significance of the principal components reduces as we go from the first principal component to the last. The first component is the most important one, followed by the second, then the third, and so on.

With only minor changes, the support vector regression generally applies similar principles as the support vector machines for classification. Any python SVR model requires three major parameter to be tuned. These are, the kernel, regularization parameter (C) and gamma values. A kernel is a function to map lower-dimensional data into higher dimensional data. For this thesis work a radial basis function kernel is used. C or regularization parameter specifies how much we want to evade misestimating each training samples. For large values of C, the algorithm will pick a more modest edge hyperplane. For small values of C, the algorithm will search for a large margin separating the hyperplane regardless of whether that implies misclassifying some points. In our case, after much investigation, we chose our C value

to be 1e3 which is a large value for C which means our algorithm will choose a smaller-margin hyperplane. The gamma parameter characterizes how far the impact of a single training sample spans, with low values signifying 'far' and high values signifying 'close'. So as such, high gamma only points values close to the boundary lines are taken into consideration when deciding on the place of the hyperplane (Zoltan, 2018). With low gamma value sample points that are close and far from the boundary lines are considered when deciding the place of the hyperplane. After some trials a gamma value of 0.1 is selected for analysis in this work.

Gaussian process regression is a powerful, non-parametric Bayesian approach towards regression problems. GPR has few advantages, functioning admirably on little datasets and being able to provide uncertainty estimations on the predictions with CI. Instead of computing the likelihood distribution of parameters of a particular function, GPR computes the likelihood distribution over all allowable functions that fit the data. Nevertheless, like in a Bayesian approach, we need to postulate a prior, calculate the posterior using the training data, and compute the predictive posterior distribution on our points of interest (Sit, 2019). The GP prior can be specified using a mean function, m(x), and covariance function, k(x, x'). Within this GP prior, we can consolidate prior knowledge about the space of functions through the choice of the mean and covariance functions.

In python, one of the main parameters that need to be initialized by the programmer is the kernel. The kernel specifies the covariance function of the GP. The default kernel, "1.0 * RBF (1.0)" is used if none is passed. It is worth mentioning that initial values for kernel hyperparameters needs to specified, however, these kernel's hyperparameters are optimized and tuned during fitting automatically. For this research, after much deliberations a product of Constant and an RBF kernel are selected. A length_scale of 1.0 and length_scale_bounds of (1e-3, 1e3) are set for the constant kernel whereas, length_scale of 10 and length_scale_bounds (1e-1, 10.0) are chosen for the RBF kernel. Moreover, in the python implementation of GPR, there exists an option to specify number_of_restarts of the optimizer for computing the kernel's parameters that maximizes the log-marginal likelihood. In general, as specified in the sklearn GPR documentation, the first run of the optimizer is executed

from the kernel's initial parameters, the rest (if any) is executed from thetas sampled log-uniform randomly from the space of specified theta-values. For this thesis after some trial and errors the n_restarts_optimizer is set to 10 by keeping in mind the execution time it takes to finish an acceptable performance.

Long-Short Term Memory are a particular type of recurrent neural networks, specifically designed for learning long-term memory dependencies. There are multiple parameters and hyper-parameters that need to be set and tuned for an LSTM to work with an acceptable performance in python. This parameters include the number of hidden layers, the number of neurons on the hidden layers, activation function, optimizer and loss function to be used by the model. For this thesis, after a lot of trial and errors, an LSTM with a single hidden layer with 100 neurons and an activation function of rectified linear unit (ReLU) is used. The ReLU activation function has a simple computation complexity that returns the value provided as input as it is, or the value of 0 if the input is less than or equal to 0. Besides, an Adam optimizer and mean squared error (MSE) for loss calculation are used. The number of epochs is set to 50 and the model is trained with a look back of 5 time lags.

3.3 Design and Implementation Issues

In recent days, the healthcare sector has become a popular adopter and beneficiary of AI-based technological advancements. A specific class of AI called machine learning plays a vital role in many healthcare delivery units, from electronic health records to telemedicine. As long as it is used efficiently and properly, machine learning, presents a great deal of advantages for the healthcare sector. However, knowing the possible issues and problems in advance can help avoid the impacts that could have been lived.

Every machine learning algorithm requires a huge amount of data to train, however, not all data is appropriate and valuable. If the data is not well prepared and understood, the resultant product might be what we are not expecting. A considerable amount of time should be spent to research and find a tool for this task. Logistical difficulties and acceptance level of the healthcare field experts are other challenges faced for the

translation and implementation of AI-based systems in healthcare. On the other hand, standard performance metrics used in ML model evaluation test also pose a barrier. Using performance measurement metrics, that are intuitive to healthcare professionals preferably that aim to measure model performance beyond technical accuracy to encompass quality of care and acceptable patient outcomes, is essential. One should target to seize real clinical applicability and be understandable to intended users; the healthcare professionals. In addition, any ML engineer should also keep in mind issues such as dataset content changes, biases, encounters of generalization of results to new population, and the accidental side-effects of the ML models on healthcare outcomes. An extensive investigation should be conducted to minimize the instability of the models, improve model generalizability and interpretability in the healthcare sector.

Concluding, a considerable amount of time has been spent by the researchers not to fall into a trap for the aforementioned ML model design and implementation issues for this work. Moreover, if these issues are handled properly, the benefits of applying ML models in the healthcare sector is paramount both for the service providers, patients and all the stakeholder of the sector.

3.4 Summary

In this machine learning models for causal inference work, python 3.6 and Spyder 4 are used as primary implementation tools due to their ease of use and popularity in the field. They are easy to use to implement any machine learning models with different integrated libraries available. A fairly general framework using principal component analysis and selected predictive models is presented in this document. The selected approach provides an effective method for monitoring and presenting daily patient medical changes from patient laboratory test data. This in turn can be used by medical experts to make causal inference and informed decision. Otherwise, the situation might be difficult to handle manually and make decisions due to the size of the data at hand.

CHAPTER FOUR

EXPERIMENTAL ANALYSIS AND RESULTS

4.1 Introduction

Various trials and experiments have been conducted to design and implement a machine learning model for establishing cause-effect relationships from medical treatment data. The required dataset is extracted from the MIMIC-III database. A total of 1410 patients with a hospital length of stay of more than 30 days is selected. Out of 1410 only 1306 patient data is actually used for this research. 104 patients were removed from the analysis since we were unable to find enough amount (they have less than 10 observation) of data in the dataset. Overall, the dataset contains 732 male patients and 574 female patients with various diagnoses. An extensive, empirical search for an acceptable machine learning model is undertaken. Principal component analysis is selected as a primary method for monitoring patient changes and progress from the medical observational data. Using the PCA results and prescriptions data, a physician will be able see the effects of the treatment applied and thereby decide what to do next. Moreover, with the aim to make the model more full-fledged three prominent predictive models were implemented. With this the physician will have the capability to anticipate future values and act proactively. Besides, the three predictive models provide the flexibility for the user to select the best possible outcome in different circumstances. Finally, the results of PCA model stability and validity is conducted and discussed in Section 4.4.1, 4.4.2, and 4.4.3, whereas performance analysis results of the predictive models is discussed under Section 4.4.4.

4.2 Experiments

PCA relies upon the presumption that most information about the data is enclosed in the directions along which the variances are the highest (Kara & Dirgenali, 2007). In PCA, we pivot the axes of the data which is determined by the data itself to a different angle/perspective. The first axis is pivoted to cover the largest variance explained by the data which informs us what is more important. After selecting the

first axis, we choose the next axis, which explains the second most variability which is orthogonal to the first axis i.e., correlation with the first access is zero. Initial analysis for this research work is conducted on a single patient data. These data includes records encompassing personal and medical information of the patient. Initially, eight different laboratory tests were selected, namely, Hemoglobin A1c (%HbA1c), Albumin, Alkaline Phosphatase (ALP), Alanine Aminotransferase (ALT), Alveolar-arterial Oxygen Difference (AaDO2), Amylase, Activated Partial Thromboplastin Time (APTT), and Aspartate Aminotransferase (AST). These are real laboratory results of an ICU patient. After applying data pre-processing including imputing missing values a simple machine learning model is applied. Moreover, the results of the data analysis on selected machine learning algorithms were discussed with specialist physicians in terms of their medical implications and the correctness of the results. By looking at the results of the analysis, and confirmation from experts in the field, the selected approaches provided some promising results. Depending on this similar and extensive analysis and application of the proposed method is applied on a wider dataset collected over a long period.

A detailed analysis and experiment of selected ICU patient data was conducted. Datasets containing at least a minimum of 10 observations (as indicated in data preparation stage) are selected and used in these experiments. Dataset for each patient contain varying number of variables as well as observations. After much research and experimentation principal components that explain 99% of the total variance in the data are returned. This threshold is selected so that significant information will not be lost while selecting the principal components. For instance for a sample patient with patient id 96309, originally the dataset contains 353 observation and 83 variables. However, after data cleaning and pre-processing the data set is reduced to 353 observation and 56 variables. In the PCA analysis, principal components that make up 99% of the overall variance are returned for each day. For example analysis results for this patient (diagnosed for coronary artery disease) shows the first four PCs for the first day and the first ten PCs for the next day that explain 99% of the variances in the data. For the aforementioned patient a total of 229 principal components were generated for 37 unique dates. On some specific dates as high as 13 principal

components and as low as 2 principal component that explain a total of 99% variance in the data are returned. Figure 4.1 shows a sample scree plot and percentage of the principal components, for first day ICU stay of the patient. On the other hand, Figure 4.2 shows a two-dimensional plot of PC direction and original variable loadings/contributions.

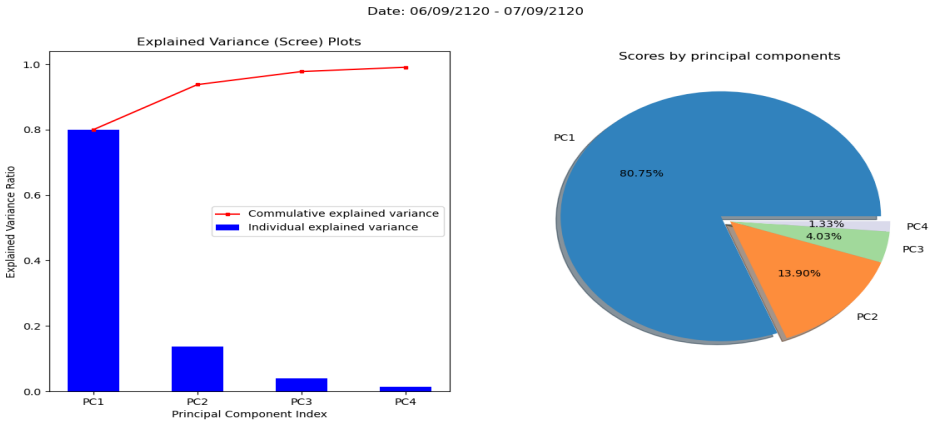


Figure 4.1 Selected patient scree plot and PC pie chart

As shown in Figure 4.1, the first 4 principal components explain 99% of the variance in the data for first day of ICU admission. Moreover, we can clearly see that the first principal component explains approximately 81% of the overall variance in the data. This shows that the physician can give emphasis for laboratory tests that contribute the highest under principal component 1. This information can be acquired from Figure 4.2 below. The x-axis represents principal component one whereas the y-axis represents principal component two from the above plot.

We can observe from Figure 4.2, that principal component one has a high negative association with Monocytes and a large positive association with Basophils where as principal component two has a large positive association with partial thromboplastin time (PTT) and a high negative association with Red cell distribution width (RDW). Basophils and Monocytes are types of white blood cells in the body. This two white blood cell types are shown as the highest associations for principal component one. On the other hand, PTT is a blood test to assess our body's ability to form blood clots. RDW measures how the sizes of our red blood cells vary. This two types of test are

shown as the highest associations for principal component two. Based on the experiments conducted, we can claim that PCA can provide a tool to observe and keep track of minor changes in the patient based on the data, even if those changes are within the normal reference ranges. If coupled with other machine learning models we believe that it will be paramount addition to the modernization of general healthcare services and the achievement of personalized healthcare. Sample results of the extensive analysis and summary are presented in Section 4.3. They show the comparison and progress of successive daily based contributions (increased or decreased variance) for selected variables under each principal component. A sample patient result is presented in this document since we cannot fit results for all the patients in one document.

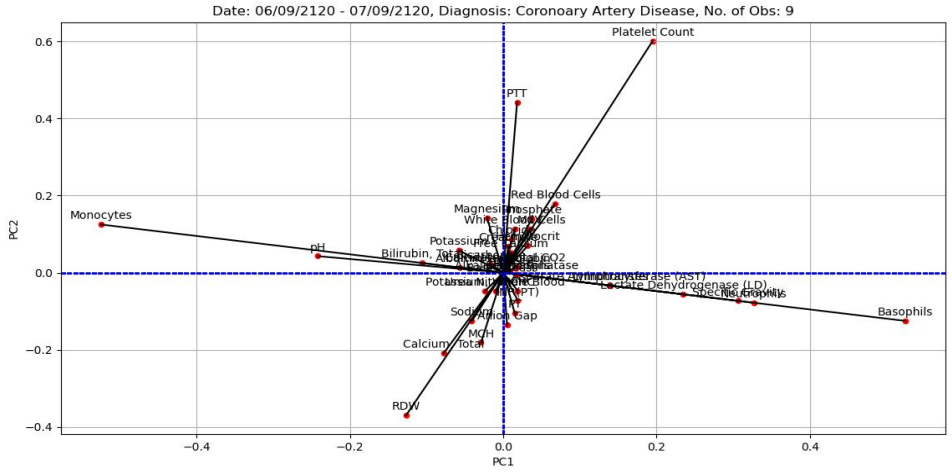


Figure 4.2 Two dimensional variable loading plot

Based on the promising results obtained from the preliminary experiment, we conducted further experiment and analysis using a selected subset of data. Our hypothesis is that, PCA can be applied on longitudinal data to monitor and capture changes in the data. Based on this a tool can be developed to bring this changes to the attention of the user. As explained before, PCA utilizes a particular coordinate system that relies upon the observations in the data. The axes are set in the direction of maximum variance in the data to maximize the variance in that direction (Witten & Frank, 2006). This intuition can help us determine and show the patient's laboratory result exhibiting the highest variances even though those changes might still be within the normal reference ranges. This process can be applied to track the changes after the

patient has been put through a certain medical intervention. Moreover, if applied on longitudinal data, it may help us explore patient progress from time T_1 to T_2. So that an appropriate treatment or therapy can be prescribed or further diagnosis can be advised.

In addition, each PC is a linear product of the original individual features. This can be used to see the effect and contribution of each laboratory test in that direction as it can be seen from Figure 4.2. The succeeding PCs try to capture the next highest variances left out by the preceding PCs. This may show epiphenomenon or parallel medical events or conditions happening. The plot shows the contribution of each original laboratory test in each direction (PCs) to indicate whether the variance is a negative or positive association. Similarly, successive daily based, computed results show the progress or changes for each variable on a daily basis. This can be used to track patient daily changes throughout the ICU stay. These results depict what changes happened on a specific day based on a treatment applied on the previous day. This can be used to decide what to perform next. Results to support the aforementioned intuition and hypothesis are presented under the Results section below.

4.3 Results

The study conducted a detailed retrospective analysis on 1,306 ICU patients with a minimum hospital LOS of 30 days. The selected target patients, were diagnosed with different diseases such as sepsis and pneumonia, with a mean hospital LOS of 56.33 days. At the end of their stay, the subjects were discharged alive or dead to home or another healthcare unit. In this section, we present sample PCA analysis results of a single patient (patient ID: 96309). This is due to the difficulty to include all patient analysis results in this document since the size of the analysis results for all the patients' is vast and cannot be fitted in one document. However, cumulative model performance analysis is presented in the next section for the overall analysis.

For example analysis results for the aforementioned sample patient (diagnosed for coronary artery disease) is presented. The results show the first four PCs for the first day, the first ten PCs for the second day and the first eight PCs for the third day that

amount approximately 99% of the variances explained in the data. These results are shown in Figure 4.3, Figure 4.4 and Figure 4.5, respectively. In addition, the pie charts on the figures describe the actual amount explained by each principal component on that specific day. For instance, on day one the analysis from Figure 4.3, the first principal component explained 80.75% of the total variance in the data whereas the second principal component explains 13.9% of the total variance in the data and so on. This shows the significance of each PC component on those dates and will help the user decide which direction to focus on or give more emphasis.

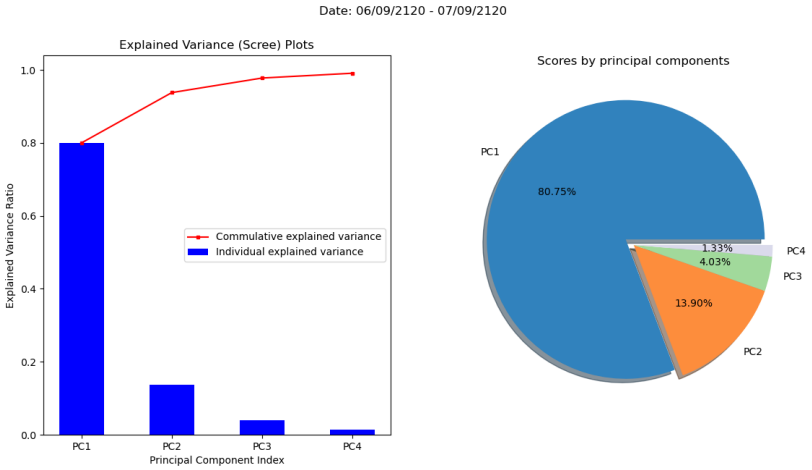


Figure 4.3 Day One Principal Component Scree Plot (PID: 96309)

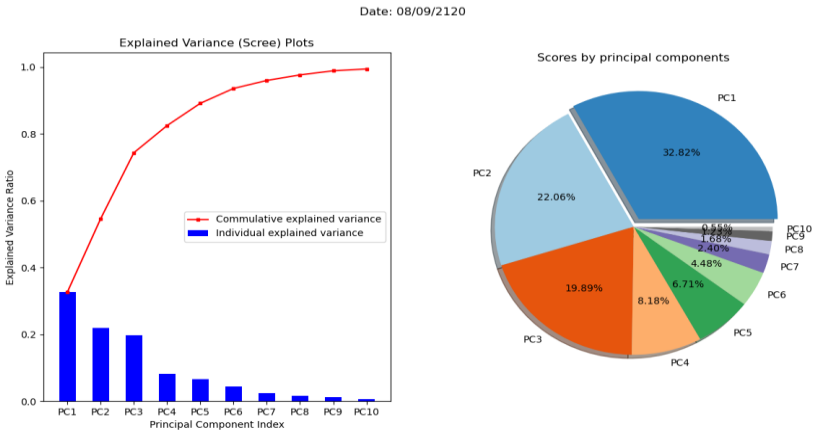


Figure 4.4 Day Two Principal Component Scree Plot (PID: 96309)

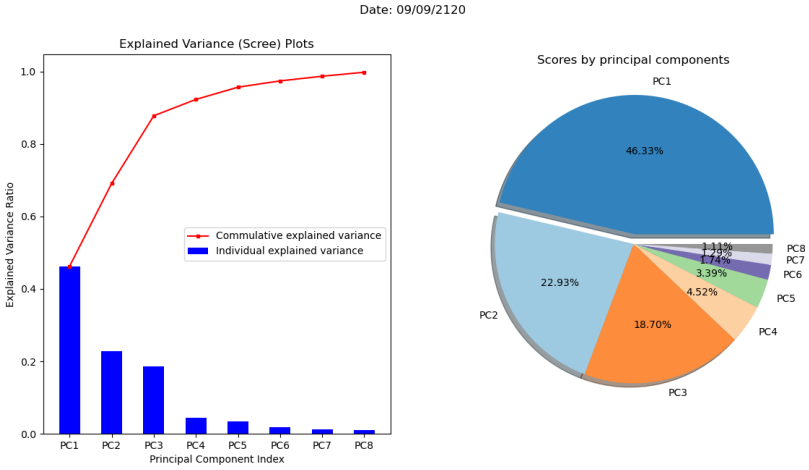


Figure 4.5 Day Three Principal Component Scree Plot (PID: 96309)

In addition the contribution of each variable to a PC can be tabulated as in Table 4.1 or presented using different formats such as heatmap or bar charts.

Table 4.1 Sample variable contributions/loadings

Day	PC	Amylase	AST	Chloride	Creatinine	Eosinophils	Hematocrit	Hemoglobin	INR(PT)	Lactate	pH	pO2
1	1	0.00	0.14	0.01	0.01	0.02	0.03	0.02	0.02	0.00	-0.24	0.00
	2	0.00	-0.03	0.09	0.07	-0.01	0.07	0.01	-0.07	0.00	0.04	0.01
	3	0.00	0.01	-0.07	-0.04	0.00	-0.05	-0.03	-0.26	0.00	-0.05	0.01
	4	0.00	0.01	0.09	-0.05	0.00	-0.06	-0.05	-0.13	0.00	-0.06	0.02
2	1	0.00	0.00	-0.05	0.01	0.00	0.07	-0.04	-0.18	-0.03	0.02	0.43
	2	0.00	0.00	0.05	-0.01	0.00	-0.13	0.21	0.29	-0.08	0.00	-0.15
	3	0.00	0.00	-0.05	0.00	0.00	-0.13	-0.60	0.20	0.03	0.01	0.40
	4	0.00	0.00	-0.11	0.01	0.00	-0.07	-0.12	0.02	0.59	-0.01	-0.29
	5	0.00	0.00	0.31	-0.04	0.00	0.19	-0.04	-0.05	0.22	-0.01	-0.02
	6	0.00	0.00	0.01	0.00	0.00	0.01	-0.15	-0.03	0.16	-0.06	-0.22
	7	0.00	0.00	0.15	0.02	0.00	-0.03	-0.11	-0.03	-0.03	-0.05	-0.20
	8	0.00	0.00	-0.14	-0.01	0.00	0.05	-0.11	-0.02	0.05	-0.11	-0.03
	9	0.00	0.00	-0.12	0.00	0.00	0.01	0.40	0.02	-0.19	-0.03	-0.14
	10	0.00	0.00	0.05	0.00	0.00	-0.04	-0.04	0.01	-0.10	-0.03	-0.47
3	1	0.00	0.00	-0.02	0.00	0.00	0.04	0.06	0.00	0.98	-0.01	0.02
	2	0.00	0.00	0.14	0.00	0.00	-0.23	-0.29	0.00	0.02	0.00	0.01
	3	0.00	0.00	0.07	0.00	0.00	-0.10	-0.12	0.00	0.11	0.00	0.02
	4	0.00	0.00	-0.09	0.00	0.00	-0.12	-0.20	0.00	-0.12	0.02	-0.08
	5	0.00	0.00	0.02	0.00	0.00	0.04	0.07	0.00	0.07	-0.04	-0.45
	6	0.00	0.00	-0.04	0.00	0.00	-0.09	-0.15	0.00	-0.01	-0.09	-0.12
	7	0.00	0.00	-0.03	0.00	0.00	-0.48	-0.41	0.00	0.01	0.00	0.00
	8	0.00	0.00	0.01	0.00	0.00	0.03	0.05	0.00	-0.01	-0.02	-0.82

Table 4.1 shows sample variable loadings per principal component for a three consecutive laboratory test dates. The bigger the absolute value of the coefficient is,

the more significant the corresponding variable is in ascertaining the principal component. For the most part, how big the absolute value of a coefficient has to be to think of it as critical is subjective. For instance, we can see from the above table that Amylase does not contribute to any of the PCs for three consecutive days, and this indicates the insignificance of these test on those days. On the other hand, pH contributes the highest under PC1 for day one, pO2 contributes the highest for day two and Lactate contributes the highest for day three of the PCA analysis. This trend shows us that there are daily changes happening, which can be attributed to a daily intervention administered such as medical prescriptions and/or some sort of therapy. Moreover, this tabular information can be presented using other high level representations for ease of use as depicted in Figure 4.6, Figure 4.7 and Figure 4.8 to show the daily changes after a certain treatment or intervention. Magnitude and directions of the coefficients of the original variables can be scrutinized to decipher every principal component. These plots visually show the component loading for day one, two and three of the analysis, respectively.

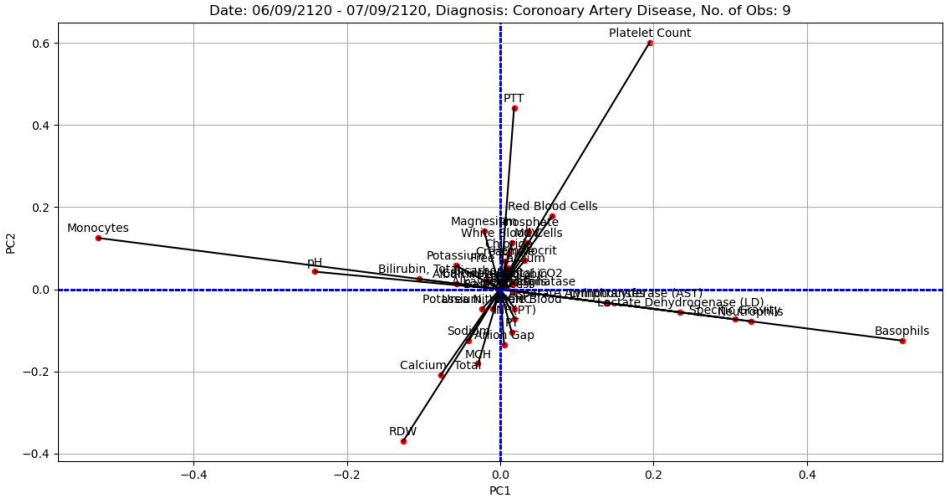


Figure 4.6 Day one original variable coefficient magnitude and direction

In these results, for day one of the ICU stay (Figure 4.6), PC1 has large positive associations with Basophils and a large negative association with Monocytes. PC2 has large positive associations with Platelet Count whereas the third PC component is driven by a large positive PTT association. Moreover, we can observe that there are

changes in magnitude and direction on the second and third days of the PCA analysis results (Figure 4.7 and Figure 4.8). This may mean that the condition of the patient is either getting better or worse. Or it may also show if a medical treatment is working or not. Based up on this, a trained physician can be able to easily infer the implication and make an informed decision. For instance, both PC1 and PC2 have a large positive association with Calculated Hematocrit on the second day whereas the third PC has a large negative association with Hemoglobin results. In addition, on the third day, PC1 is driven by a large positive association with Lactate, however, PC2 has a large positive association with Oxygen Saturation.

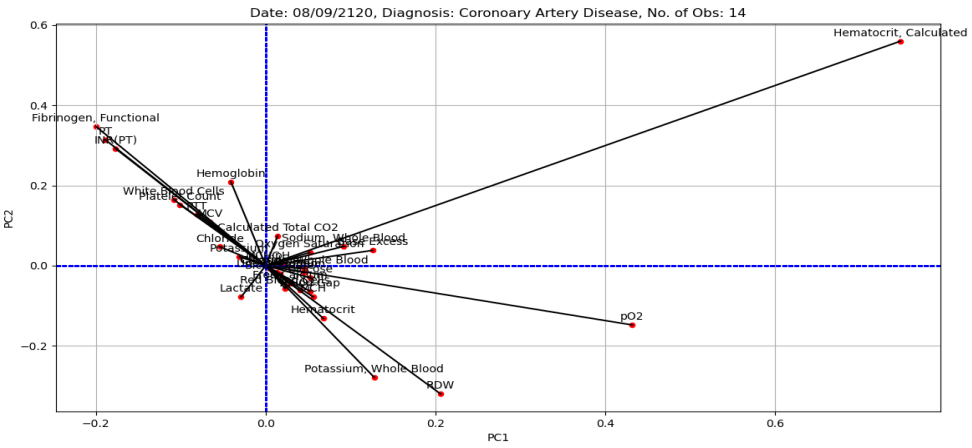


Figure 4.7 Day two original variable coefficient magnitude and direction

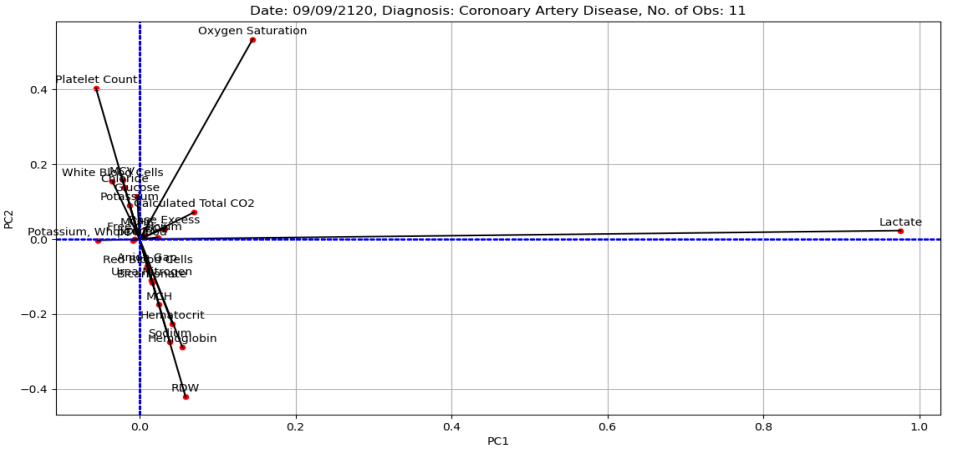


Figure 4.8 Day three original variable coefficient magnitude and direction

The plots only show comparison and progress of successive daily based contributions (negative or positive) of the variables under different PCs for a patient. They depict what changes happened on a specific day based on a treatment applied on the previous day. This can be used to decide on further steps that need to be carried out. Furthermore, this will allow the user to be able to establish cause-effect relationships. At this point, it is worth mentioning that, the patient has been given medical prescriptions on the specified dates (see Table 4.2). This may mean that the condition of the patient is either improving or worsening. Or it may also show if a medical treatment is working or not. Based upon this, a trained physician can be able to easily infer the implication and make a causal inference.

Table 4.2 Associated daily medical prescriptions.

Day One		**Day Two**		**Day Three**	
06/09/2120 - 07/09/2120		**06/09/2120 - 08/09/2120**		**08/09/2120 - 09/09/2120**	
Drug	**Drug Type**	**Drug**	**Drug Type**	**Drug**	**Drug Type**
Aspirin EC	MAIN	Magnesium Sulfate	MAIN	Ranitidine	MAIN
Heparin Sodium	MAIN	Albuterol-Ipratropium	MAIN	Meperidine	MAIN
5% Dextrose	BASE	Docusate Sodium	MAIN	Glycopyrrolate	MAIN
Potassium Chloride	MAIN	SW	BASE	Neostigmine	MAIN
Vancomycin	MAIN	NS	BASE	Propofol	MAIN
Iso-Osmotic Dextrose	BASE	0.9% Sodium Chloride	BASE	Vancomycin	MAIN
		Calcium Gluconate	MAIN	5% Dextrose	BASE
		Atorvastatin	MAIN	Chlorhexidine Gluconate 0.12% Oral Rinse	MAIN
		Heparin (IABP)	MAIN	Albumin 5% (25g / 500mL)	MAIN
		Lorazepam	MAIN	Ciprofloxacin HCl	MAIN
		Ciprofloxacin HCl	MAIN	Morphine Sulfate	MAIN
		Acetaminophen	MAIN	Docusate Sodium (Liquid)	MAIN
		Claritin	MAIN	Amiodarone	MAIN
		Heparin Sodium	MAIN	Oxycodone-Acetaminophen	MAIN
		Vancomycin	MAIN	Furosemide	MAIN
		Furosemide	MAIN	Vasopressin	MAIN
		Iso-Osmotic Dextrose	BASE	D5W	BASE
		5% Dextrose	BASE	Racepinephrine	MAIN
		Meperidine	MAIN	Milrinone	MAIN
		LR	BASE	Albumin 5% (12.5g / 250mL)	MAIN
				5% Dextrose (EXCEL BAG)	BASE
				Xopenex Neb	MAIN

It can be seen from Table 4.2 similar medicines have been used on different dates. The usage of some of the medicines listed also extends beyond the dates presented as a reference in the plots. The drugs were used as a base or main drug type as indicated in the table. Finally, the dimensionality suggested by the scree plots of the analysis are variant pursuant to the number of features and samples we have in the dataset, corresponding to 99% of the explained variance in the data. At the end, the stability

and validity of the model for the overall analysis was confirmed by bootstrap, permutation and causal impact analysis tests.

4.4 Performance Analysis

To weigh the proposed machine learning models, various and relevant validation tests have been conducted. Both permutation and bootstrap testing are employed for PCA model stability and validity tests. Moreover, causal impact analysis is also conducted using CausalImpact R library, which uses Bayesian networks for causal inference. In addition, the performance of the proposed predictive models is evaluated using RMSE errors. The results of these analyses is presented below under Section 4.4.1, 4.4.2 and 4.4.3.

4.1.1 Permutation Tests

Permutation test sometimes known as randomization test is employed without relying on a specific probability model. To assess the validity and stability of the proposed model, statistic of interest namely total variance accounted for (TVAF) is computed. TVAF is equal to the sum of the eigenvalues of the first n principal component. This is followed by r number of permutation (r=1000 in this case) and statistical estimates for each permutation is computed. The statistical significance between the observed TVAF and the estimated TVAF is determined by comparing the p-value to a significance level.

A statistical significance level of $\alpha=0.05$ is adopted as a rejection rule for this study. The alternative hypothesis that the estimated TVAF values does not deviate significantly from the observed TVAF value is tested against the null hypothesis that it does. If $p<\alpha$, the result is marked significant and H_0 is rejected. The stability and validity of the overall analysis was confirmed by the p-values of estimated TVAFs. The estimated statistics achieved a significance level of 0.001 which is much less than the predetermined α value which is 0.05. This proves that our alternative hypothesis is accepted which posits on our models' stability and validity. Finally we can generalize that the proposed model is stable and valid.

4.1.2 Bootstrap Tests

In addition to permutation testing, python code for bootstrap testing is implemented without relying on a specific probability model. Bootstrapping is a statistical technique that resamples a single dataset to create many simulated samples with replacement. This process allows us to compute standard errors, create confidence intervals, and do hypothesis testing for several types of sample statistics.

The null hypothesis that, there is no statistical difference among the observed total explained variance vs the estimated total explained variance is tested against the alternative hypothesis, there is a statistical difference between the observed total explained variance vs the estimated total explained variance. To assess the soundness and stability of the proposed model, statistic of interest namely total variance accounted for is computed. This is followed by r number of resampling/iteration (r=1000 in this case) and statistical estimates (estimated TVAF) for each permutation is computed. Four different sample sizes (resampling ratio) for resampling are experimented; 15%, 25%, 50% and 65% of the whole data. For every iteration the aforementioned sample size thresholds were tested. The statistical significance between the observed TVAF and the estimated TVAF is determined by comparing the p-value to a predefined alpha value. A statistical significance level α=0.05 is adopted as a rejection rule for this study. Results of a sample patient from the candidate subject data are presented below in Figure 4.9, Figure 4.10, Figure 4.11, and Figure 4.12.

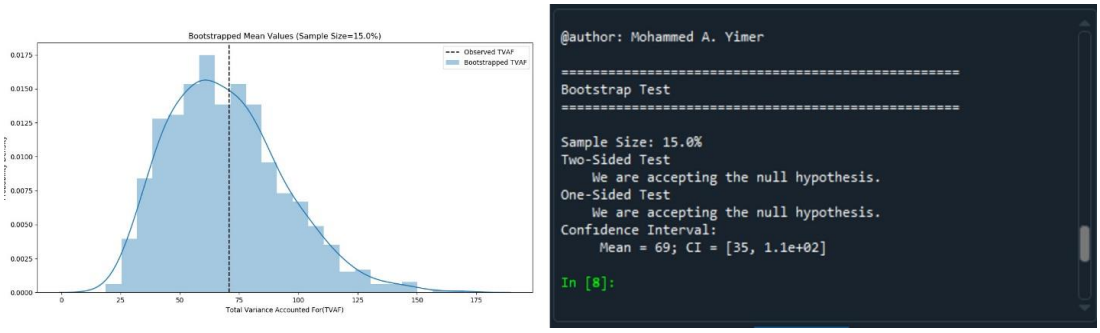


Figure 4.9 Bootstrap mean values and console outputs (Sample Size: 15%)

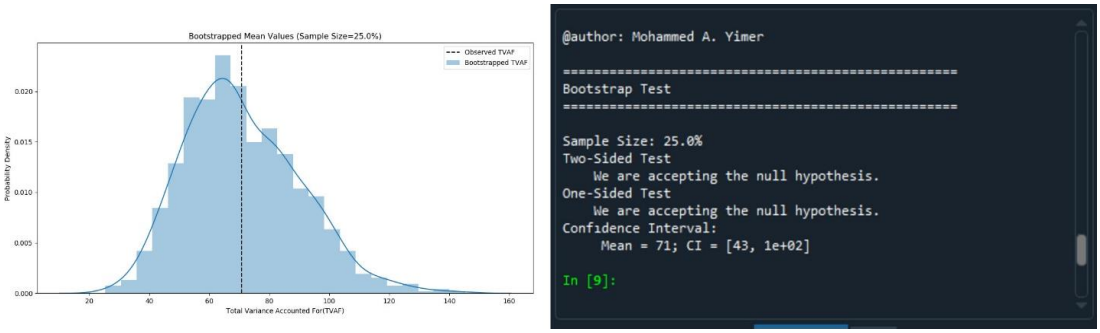


Figure 4.10 Bootstrap mean values and console outputs (Sample Size: 25%)

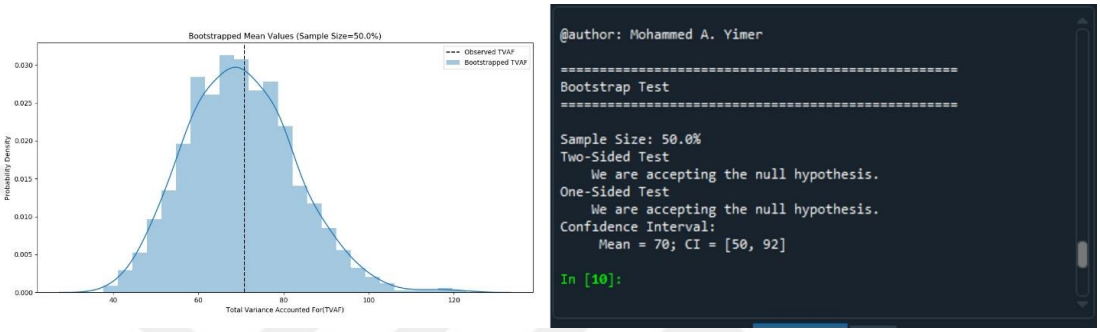


Figure 4.11 Bootstrap mean values and console outputs (Sample Size: 50%)

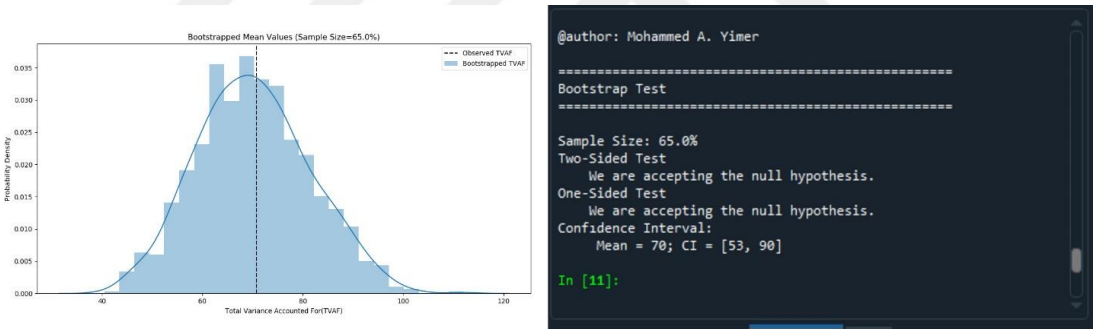


Figure 4.12 Bootstrap mean values and console outputs (Sample Size: 65%)

We used both one-sided and two-sided tests. In the two-tailed test, if the computed p-value is less than or equal to our expected significance level, our Null hypothesis will be rejected. The logic is that the p-value is the probability of getting outcome outside of our 95% confidence interval by chance. If the p-value is smaller than our alpha, it means it is improbable that any outcome outside of our 95% confidence interval is by chance, meaning that it is noteworthy and shouldn't be overlooked as an error. Whereas if the p-value is greater than our alpha, it means that it is likely that the result outside of our 95% confidence interval occurred by chance, so we shouldn't worry and will fail to reject our null hypothesis. For both the one-sided and two-sided

tests the model performed with an acceptable significance level. This means that, we are going to accept our null hypothesis and conclude that there is not a statistical difference between the observed explained variance vs the estimated explained variance. This posits the stability and validity of the proposed model.

4.1.3 Causal Impact Tests

In addition to model stability and validity tests using bootstrap and permutation testing for the PCA model, causal impact analysis is also conducted using CausalImpact R package by Google. The library implements a Bayesian structural time-series model to estimate the causal impact of a treatment. Generally, given a response time-series and a set of control (covariates) time-series, the model performs posterior inference and constructs a Bayesian structural time-series model, which in turn can be used to try and predict counterfactuals. These counterfactuals represent a time series that shows how the response would have progressed after the treatment if the treatment had never happened. It constructs a synthetic time series baseline after adjusting the size difference between the control group and the response. Finally, it returns a summary of the results as a table (Table 4.3, Table 4.4, and Table 4.5), verbal description, and a plot (Figure 4.14, Figure 4.16, and Figure 4.18).

For this task, covariate laboratory tests were selected for each consecutive day PCA analysis. The selection is carried out based on correlation analysis and consulting medical intern students. The algorithm requires control groups (covariates) that are not affected by the treatment or intervention themselves. In order to maintain this requirement, absolute care has been taken to select covariates with no contribution/loading in the PCA analysis. Since they are the ones that are not affected by the treatment, hence we don't expect to see any impact on the measure of these laboratory tests by the treatment. Below, we present sample results of causal impact analysis for a sample patient (PID: 96309) for consecutive three-day PCA analysis results. The causal impact plot comprises three panels. The top panel (original) displays the data (the black horizontal line) and a counterfactual prediction (a dashed horizontal line) for the post-treatment period. The treatment date is indicated by a dashed vertical line and the post-treatment period is the time-series following the

treatment time-point. The middle panel (pointwise) presents the pointwise dissimilarity among observed data and counterfactual predictions. This is the pointwise causal effect, that is estimated by the model. The bottom panel (cumulative) sums the pointwise contributions from the middle panel, ensuing in a plot of the cumulative effect of the treatment (Brodersen & Hauser, 2015). These inferences rely fundamentally upon the presumption that the covariates were not themselves influenced by the intervention.

For day one as depicted in Figure 4.6, Basophils contributes the highest for PC1. With this, control groups that are covariates of Basophils that are not impacted by the treatment (no contribution, no variance) are selected as depicted in Figure 4.13.

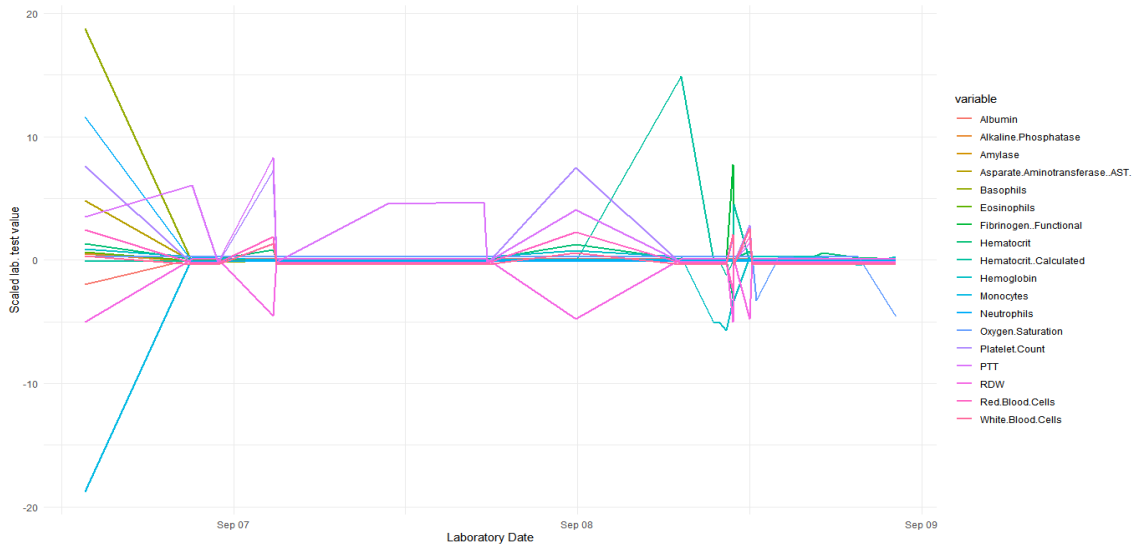


Figure 4.13 Basophils covariates time series plot

Results of the causal impact analysis for this laboratory test is depicted in Figure 4.14 below and followed by tabular and descriptive summary of the causal impact analysis results.

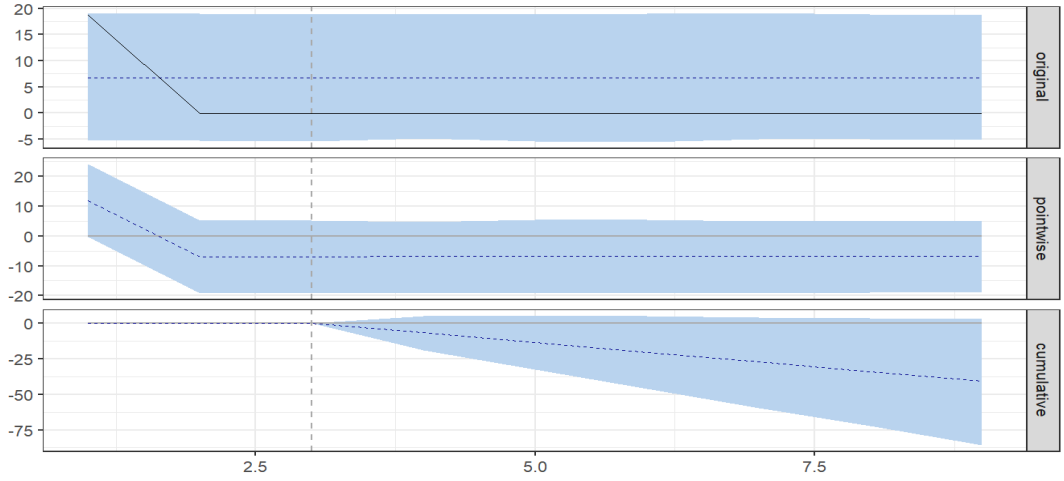


Figure 4.14 Basophils causal impact analysis results

Table 4.3 Day One causal impact analysis posterior inference

	Average	**Cumulative**
Actual	-0.05	-0.30
Prediction (s.d)	6.8 (3.7)	40.6 (22.4)
95% CI	[-0.51, 14]	[-3.05, 85]
Absolute effect (s.d.)	-6.8 (3.7)	-40.9 (22.4)
95% CI	[-14, 0.46]	[-85, 2.75]
Relative effect (s.d.)	-101% (55%)	-101% (55%)
95% CI	[-210%, 6.8%]	[-210%, 6.8%]
Posterior tail-area probability p: 0.03341		
Posterior prob. of a causal effect: 96.659%		

The average column discusses the average (across time) throughout the post-treatment period (time-points 4 through 9) whereas the cumulative column sums up individual time points. In our case the cumulative column bears no role for our analysis since the response variable is not a flow quantity. In the analysis, the projected mean causal effect of the treatment is -6.8. The 95% posterior interval of the mean influence is [-14, 0.46]. Since this includes 0, we can deduce that the treatment had a negative causal effect on the response variable. During the post-intervention time frame, the response variable had a mean value of roughly -0.05. Without an intervention, we

would have anticipated a mean value of 6.76. The 95% interval of this counterfactual prediction is [-0.51, 14.14]. Deducting this estimated values from the observed value yields an estimate of the causal impact the intervention had on the response variable. This impact is -6.81 with a 95% interval of [-14.19, 0.46]. Adding up the individual data points during the post-intervention period, the response variable had a total value of -0.30. Had the intervention not occurred, we would have anticipated an amount of 40.55. The 95% interval of this prediction is [-3.05, 84.86]. The above effects are given as absolute numbers. In relative terms, the response variable exhibited a drop off -101%. The 95% interval of this percentage is [-210%, +7%]. This implies that, in spite of the fact that it might look like the intervention has put forth a negative impact on the response variable when considering the intervention time frame in general, this impact isn't measurably critical. The clear impact could be the aftereffect of arbitrary changes that may be random to the intervention. This is often the case when the intervention period is too short to differentiate the signal from a noise. The likelihood of acquiring this impact by chance is little (Bayesian one-sided tail-area probability p = 0.033). This implies the causal impact can be considered genuinely significant.

For day two as depicted in Figure 4.7, Calculated Hematocrit contributes the highest for both PC1 and PC2. With this, control groups that are covariates of Calculated Hematocrit that are not impacted by the treatment are selected as depicted in Figure 4.15. Results of causal impact analysis for this laboratory test is depicted in Figure 4.16 below.

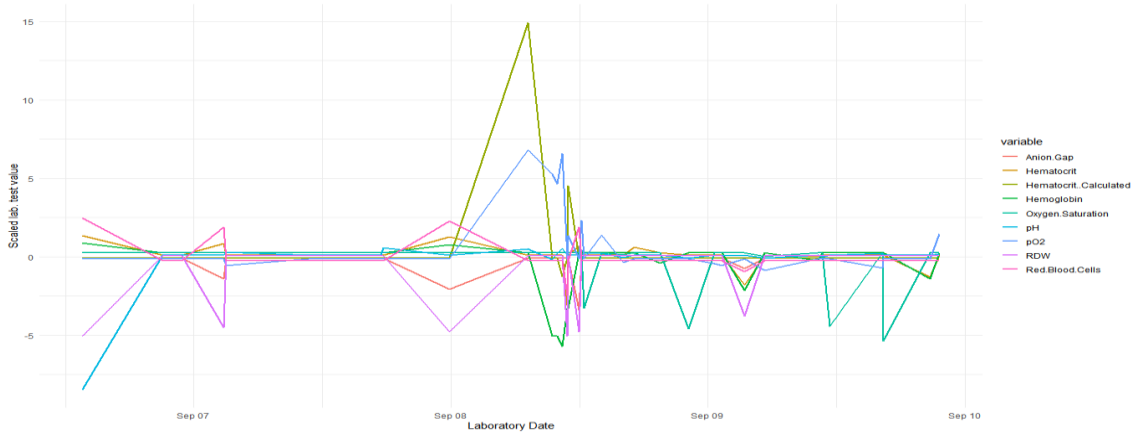


Figure 4.15 Calculated Hematocrit covariates time series plot

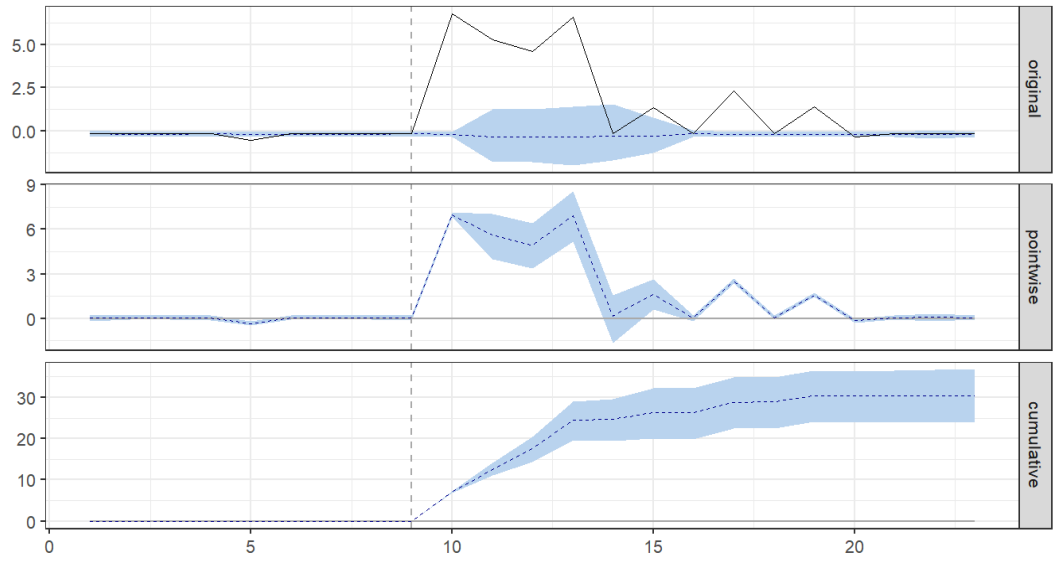


Figure 4.16 Calculated Hematocrit causal impact analysis results

Table 4.4 Day Two causal impact analysis posterior inference

	Average	**Cumulative**
Actual	1.9	27.1
Prediction (s.d)	-0.24 (0.21)	-3.37 (2.95)
95% CI	[-0.69, 0.23]	[-9.64, 3.19]
Absolute effect (s.d.)	2.2 (0.21)	30.5 (2.95)
95% CI	[1.7, 2.6]	[23.9, 36.8]
Relative effect (s.d.)	-905% (-88%)	-905% (-88%)
95% CI	[-711%, -1092%]	[-711%, -1092%]
Posterior tail-area probability p: 1e-04		
Posterior prob. of a causal effect: 99.98999%		

In the analysis, the projected mean causal effect of the treatment is 2.2. The 95% posterior interval of the average effect is [1.7, 2.63]. Since this excludes 0, we can deduce that the treatment had a causal effect on the response variable. During the post-intervention time frame, the response variable had a mean value of roughly 1.94. Without an intervention, we would have anticipated a mean value of -0.24. The 95% interval of this counterfactual prediction is [-0.69, 0.23]. Deducting this estimated values from the observed value yields an estimate of the causal impact the intervention

had on the response variable. This impact is 2.18 with a 95% interval of [1.71, 2.63]. Adding up the individual data points during the post-intervention period, the response variable had a total value of 27.12. On the other hand, had the intervention not occurred, we would have expected an amount of -3.37. The 95% interval of this prediction is [-9.64, 3.19]. The above effects are given as absolute numbers. In relative terms, the response variable exhibited a drop off -905%. The 95% interval of this percentage is [-711%, -1092%]. This implies that the negative impact seen during the intervention time frame is statistically significant. The likelihood of acquiring this impact by chance is little (Bayesian one-sided tail-area probability p = 0). This implies the causal impact can be considered genuinely significant.

Finally, for day three as depicted in Figure 4.8, Lactate contributes the highest for PC1. With this, control groups that are covariates of Lactate that are not impacted by the treatment are selected as depicted in Figure 4.17. Results of causal impact analysis for this laboratory test is depicted in Figure 4.18 below.

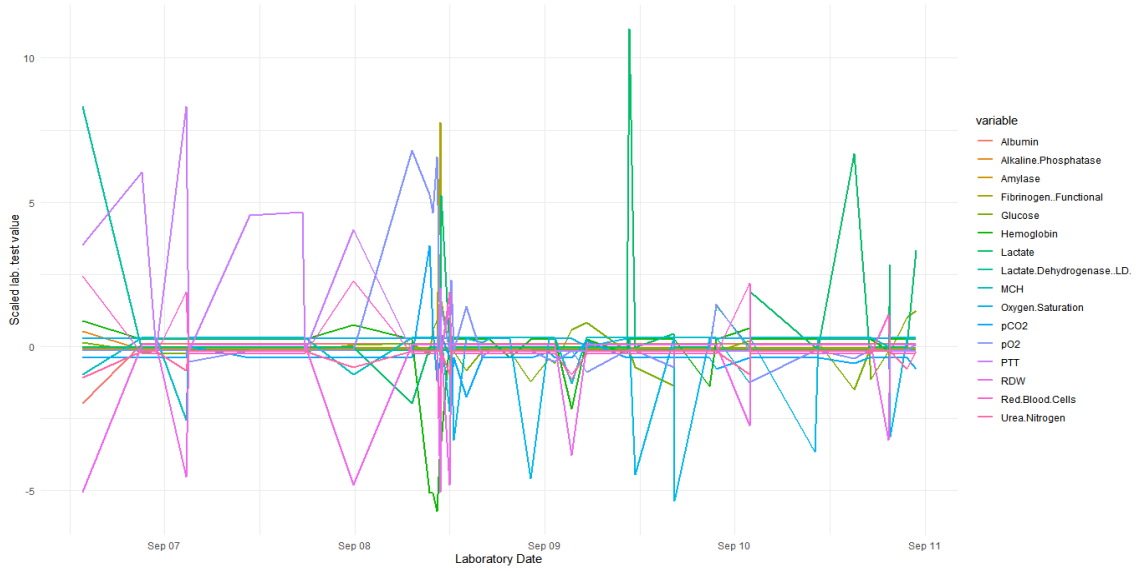


Figure 4.17 Lactate covariates time series plot

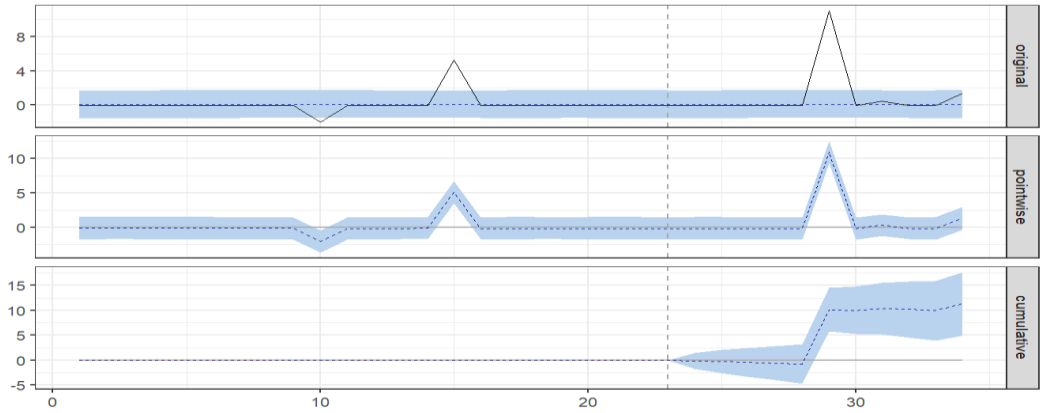


Figure 4.18 Lactate causal impact analysis results

Table 4.5 Day Three causal impact analysis posterior inference

	Average	**Cumulative**
Actual	1.1	12.5
Prediction (s.d)	0.11 (0.3)	1.20 (3.2)
95% CI	[-0.48, 0.68]	[-5.24, 7.53]
Absolute effect (s.d.)	1 (0.3)	11 (3.2)
95% CI	[0.45, 1.6]	[5.00, 17.8]
Relative effect (s.d.)	940% (270%)	940% (270%)
95% CI	[415%, 1475%]	[415%, 1475%]
Posterior tail-area probability p: 4e-04		
Posterior prob. of a causal effect: 99.96%		

In the analysis, the projected mean causal effect of the treatment is 1. The 95% posterior interval of the mean influence is [0.45, 1.6]. Since this excludes 0, we can deduce that the treatment had a causal effect on the response variable. During the post-intervention time frame, the response variable had a mean value of roughly 1.14. Without an intervention, we would have anticipated a mean value of 0.11. The 95% interval of this counterfactual prediction is [-0.48, 0.68]. Deducting this estimated values from the observed value yields an estimate of the causal impact the intervention had on the response variable. This impact is 1.03 with a 95% interval of [0.45, 1.62]. Adding up the individual data points during the post-intervention period, the response

variable had a total value of 12.53. Had the intervention not occurred, we would have anticipated an amount 1.20. The 95% interval of this prediction is [-5.24, 7.53]. The above effects are given as absolute numbers. In relative terms, the response variable exhibited an increase of +940%. The 95% interval of this percentage is [+415%, +1475%]. This implies that the positive effect seen during the intervention time frame is statistically significant and probably not going to be because of arbitrary variances. It ought to be noted, nonetheless, that whether or not this increase also bears considerable importance must be answered by comparing the absolute effect (1.03) to the original goal of the treatment. The likelihood of acquiring this impact by chance is little (Bayesian one-sided tail-area probability p = 0). This implies the causal impact can be considered genuinely significant. Moreover, in cases where there are spurious changes due to extra missing values and/or newly administered medical tests causing the change, the causal impact analysis was able to capture them as bogus.

4.1.4 Predictive Model Performances (Error)

Root mean square error is computed for performance measurement of the proposed predictive models. Generally, root-mean-square error is often used to compute the differences between predicted values by a model and the observed values. It is a degree of accuracy, to compare prediction errors of different models for a specific dataset and not amongst datasets. RMSE is always non-negative, and a value of 0 indicates a perfect fit to the data. RMSE is the square root of the average of squared errors as presented in equation 4.1 (Neill & Hashemi, 2018).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \hat{x}_i)^2}{N}} \tag{4.1}$$

Where,

i = the i^{th} observation,
N = the total number of non-missing observations,
x_i = observed values,
$\hat{x_i}$ = estimated (predicted) values

The prediction results along with the RMSE value of each model is presented for the user to see. The user will have the flexibility to choose the best result based on both

by looking at the line plots and the RMSE values. Sample performance results of a single patient are presented in Figure 4.19 and Table 4.6 below.

Figure 4.19 Sample predictive model performance results

As it can be seen from Figure 4.19, support vector regression seems to perform well for this particular laboratory test than the other models with an RMSE of 0.95. Additional prediction performance of the three predictive models for the same patient for other laboratory tests is presented in Table 4.6 below.

Table 4.6 Sample predictive model performance results

Laboratory Test	*SVR*	*GPR*	*LSTM*	*Average Error*
Albumin	**0.003**	**0.003**	0.020	0.009
Amylase	3.306	3.322	**2.155**	2.928
Anion Gap	0.369	**0.277**	1.632	0.759
Base Excess	0.532	**0.462**	1.486	0.826
Bicarbonate	0.342	**0.307**	2.610	1.086
Bilirubin, Total	**0.001**	**0.001**	0.014	0.005
Calcium, Total	**0.133**	0.229	0.440	0.267
Calculated Total CO2	**0.949**	1.013	3.976	1.979
Chloride	**2.047**	3.080	15.076	6.734
Chloride, Whole Blood	**0.387**	0.680	0.675	0.580
Creatinine	0.014	**0.009**	0.074	0.032
Eosinophils	0.139	**0.121**	0.175	0.145
Free Calcium	**0.000**	**0.000**	0.004	0.001
Hematocrit, Calculated	**0.045**	0.070	0.317	0.144
Hemoglobin	**0.063**	0.092	1.674	0.610
INR(PT)	**0.001**	**0.001**	0.006	0.003
Lactate	0.006	**0.002**	0.008	0.006
Lymphocytes	**0.068**	0.078	**0.068**	0.071
MCH	0.005	**0.004**	0.577	0.195
MCHC	**0.011**	0.012	2.862	0.962
MCV	0.085	**0.074**	1.511	0.557
Magnesium	0.017	**0.015**	0.103	0.045
Monocytes	0.010	**0.002**	0.005	0.006
Neutrophils	**0.867**	0.919	1.872	1.219
Oxygen Saturation	3.851	**1.926**	6.518	4.098
PT	**0.028**	0.039	0.188	0.085
Phosphate	**0.096**	0.108	0.155	0.120
Potassium	0.029	0.095	**0.028**	0.051
Potassium, Whole Blood	0.011	**0.007**	0.332	0.117
Protein	**2.275**	2.328	4.052	2.885
RDW	0.008	**0.005**	0.063	0.025
Red Blood Cells	0.005	**0.003**	0.124	0.044
Sodium, Whole Blood	**0.397**	0.643	5.195	2.078
White Blood Cells	**0.558**	0.752	1.548	0.953
pCO2	1.599	**0.927**	9.071	3.866
pH	**0.023**	0.025	0.449	0.166

As it can be seen from the above table for some laboratory test SVR performance is better while for some laboratory tests the other models perform well. The best model RMSE value is shown with a bold face.

4.2 Summary

Causal Inference in medical treatment data is all about determining the effect of a medical treatment or intervention on the health conditions of a patient. Presume that there was a treatment that took place, such as a medical prescription or therapeutics. By observing the results taken after the intervention, we know and have the outcome at our hand. What is additionally significant is to realize what might have occurred if the intervention didn't take place. By contrasting the two potential results we can surmise or quantify the effect of the clinical interventions. We can say that the patient's health improving or worsening is the effect of the treatment. Let there be two individuals, 1 and 2, and they are both being exposed to the treatment. By applying the treatment on both of them we can likewise notice the new results. However, at that point we don't have any clear idea what the potential results may have been. As it is not possible to know it, the next best thing we can do is estimate the other possible outcome. Causal Inference is a dubious issue that data science researchers infrequently talk about. While this may be because of the way that we by and large don't need to deal with it, still it is in every case great to realize how to take care of such an issue. It never hurts to have one more tool in our toolkit, one more weapon in our arsenal.

It is in view of these facts, numerous experiments are conducted for designing machine learning models for causal inference. Accordingly, different model performance measurement metrics have been applied depending on the models. For PCA, permutation and bootstrap testing have been applied and acceptable model performance with an acceptable significance level is achieved. Moreover, causal impact analysis is also conducted. On the other hand, RMSE computation is applied for performance measurement of the predictive models. Generally, the study did not intend to implement the three predictive models for the selection of one out of three with the best performance by the researchers. However, it is intended to present the results of the three models along with the performance values for the user and allow the user to choose from. This provides the user with the added flexibility to choose predictive model that is deemed relevant to certain problems and under different circumstances.

CHAPTER FIVE

PROTOTYPE APPLICATION DEVELOPMENT

5.1 Introduction

The prototype application is developed using python and bokeh server as a simple web-based application. The application performs PCA using patient laboratory test data. It also incorporates the medical prescriptions provided on specific days. This helps the medical practitioner to be able to establish a cause-effect relationship based on the PCA analysis results (the changes observed) and prescriptions provided on those dates. The application has three main modules presented in three separate tabs. The first module presents patient related demographic, general and statistical information based on patient data, the second module (the main module) presents PCA analysis results using tabular and other types of plots along with some controls deemed necessary for further analysis by the user. In addition, prescriptions given for the patient on those particular dates will also be presented on a separate pane. The controls include, options to change laboratory test dates and to select specific laboratory tests to include in the PCA analysis. Finally, the third module presents the three predictive models we implemented for the user to be able to see future values for anticipation. Similar to module two the user is presented with some controls to change laboratory test dates (to be able to see predictions based on specific date) and to select laboratory tests to see prediction for. This chapter discusses the development environment and presents results from the prototype application.

5.2 Development Environment

Python provides an interpreted, high-level and general-purpose programming language for application in different problem areas. It is a preferred programming language due to its readability, and has some similarities to the English language with influence from mathematics. Moreover, it is also the most preferred programming language for machine learning and deep learning due to the various benefits it offers. Bokeh Server provides python users the capability to create interactive web-based

applications that can connect frontend user-interface events to real, running python code (Bokeh Development Team, 2018). Model objects signifying things such as plots, ranges, axes, glyphs, etc. are created in python and converted to a JavaScript object notation (JSON) format used by the client library BokehJS. Generally, bokeh server aims to provide means to synchronize between Python and the browser. At the point when the controls are manipulated, their new values are consequently adjusted in the Bokeh Server. Callbacks are set off that additionally update the information for the plot in the server. These progressions are automatically adjusted back to the browser, and the plots are refreshed. One can easily develop an interactive web-based applications using python programming with help of bokeh server. They provides an easy to use, efficient and suitable development environment for web-based machine learning applications.

5.3 The Web-Based Application

For preliminary investigations on our first work, PCA is applied for observing daily changes based on laboratory test results without taking into consideration the medical prescriptions provided. However, later on we incorporated the prescription provided to the patient on specific dates along with the PCA analysis of the daily laboratory tests. By doing so, the medical practitioner can be able to establish causal relationships based on the PCA analysis results and prescriptions provided on those dates. Based on this, a simple web-based interactive application is developed. The overall task is implemented using bokeh server and python. Results of the completed prototype application are presented below. This prototype application is used as a base for analysis for this work however, with a little effort and some additions it can be turned into a full-fledged application for the problem area being investigated.

Figure 5.1 presents results of module one. The left side displays general demographic and basic patient information. The top right pane displays the number of times the patient has been admitted to the hospital along with admission date, admission location, admission type, initial diagnosis results, and hospital discharge time. On the other hand, the bottom right pane presents statistical data such as, mean and standard deviation for each laboratory test based on the patient data. At the bottom

of the page information such as the total number of unique laboratory tests conducted, total number of observations and most frequent laboratory tests applied on the patient are displayed.

Furthermore, Figures 5.1 – 5.9 present partial view of results from module two. Generally the left side pane presents user controls such as specific date range and laboratory test selection controls. The middle section presents principal component loadings (contributions) of each original variable with a table and principal component scree plot. In addition, daily prescriptions are presented on the right side pane separately. Additional plots are available under this module, these figures only present plots that we can fit on a single screen at a time. Some of these plots are presented separately below. As it can be seen in Figure 5.2 specific date range is specified in the FROM and TO dropdown boxes. Based on the dates specified the PCA analysis results is recomputed and results are updated and displayed. In addition, prescriptions given on those date are filtered and presented on the right side pane.

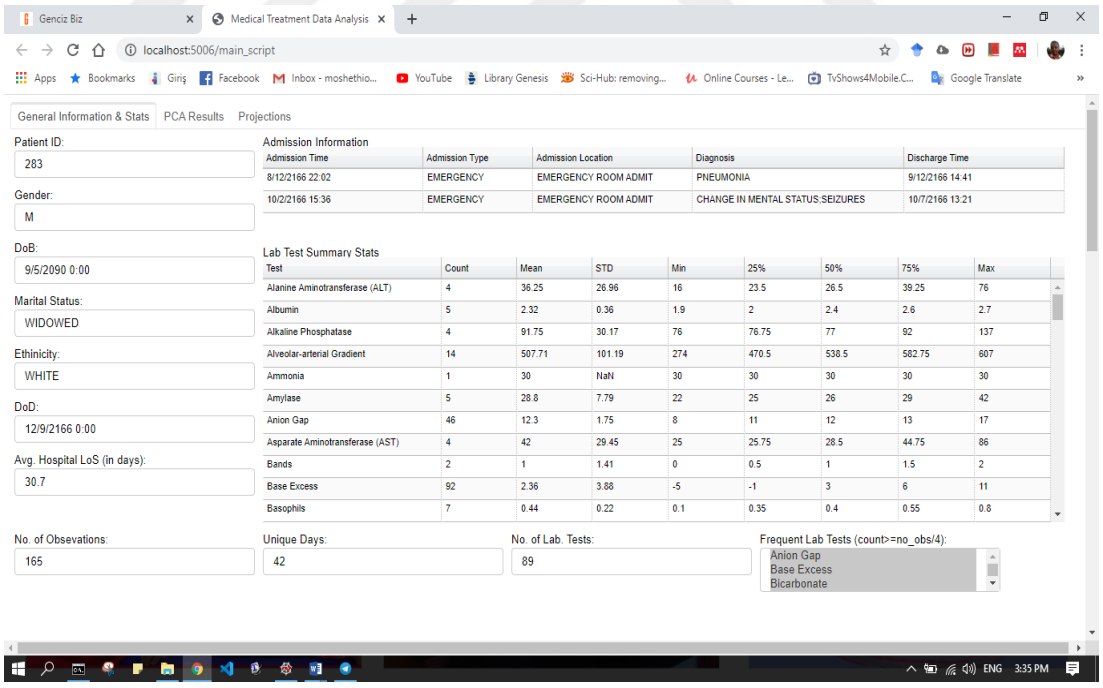


Figure 5.1 General and Stat. Information tab

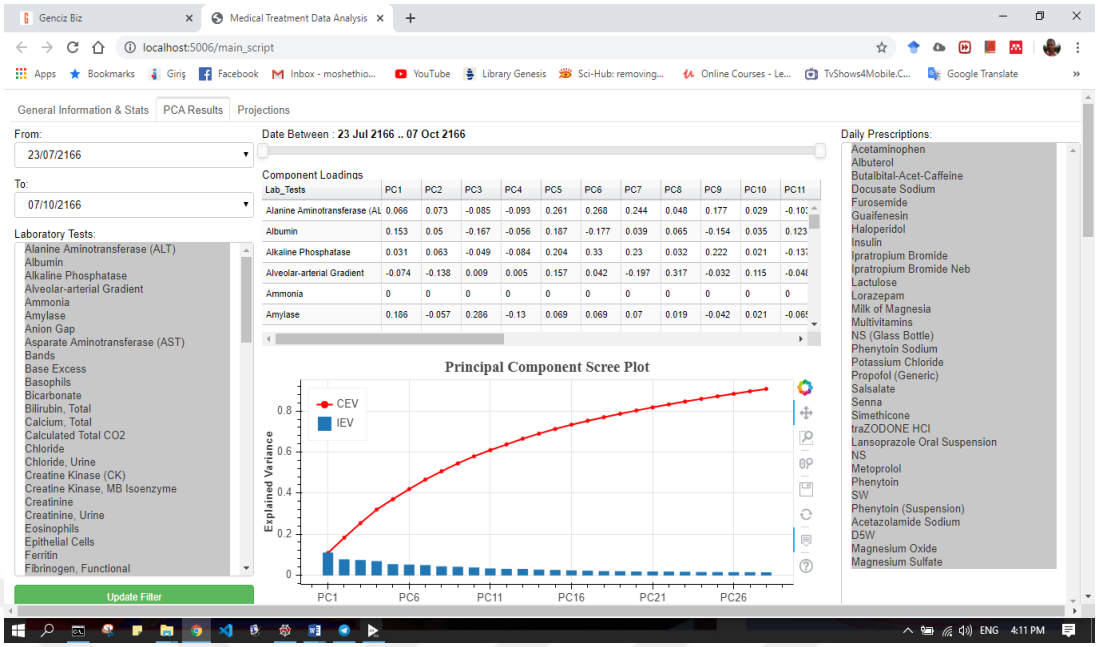


Figure 5.2 Full data PCA results

For instance, Figure 5.3 depicts a different result from Figure 5.2 which presents PCA results for full dataset by default after the user changed the date range using FROM and TO drop-boxes. These controls are populated with the list of dates from the laboratory test dates for that specific patient from the dataset.

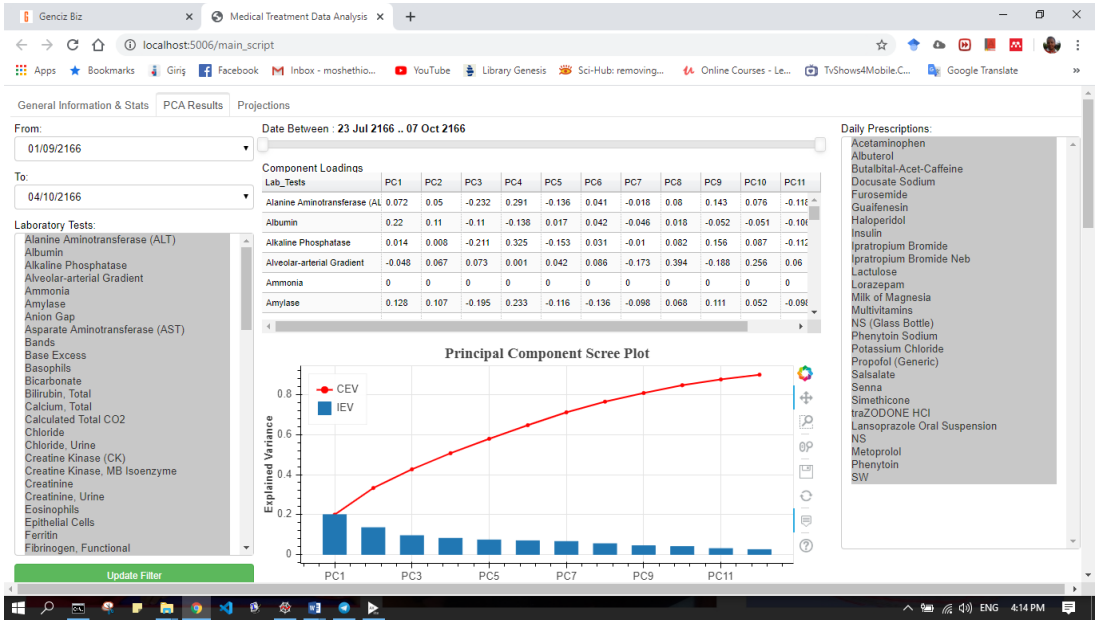


Figure 5.3 Selected dates PCA results

Moreover, Figure 5.4 presents results of the PCA analysis for specific laboratory tests selected in the Laboratory Tests list box, the shaded region that contains six laboratory test between Alanine Aminotransferase (ALT) to Amylase. Based on the selected laboratory tests the PCA analysis result is recomputed and plots are updated and displayed when the user clicks Update Filter button. Similarly, prescriptions given on those dates are filtered and presented on the right side pane automatically.

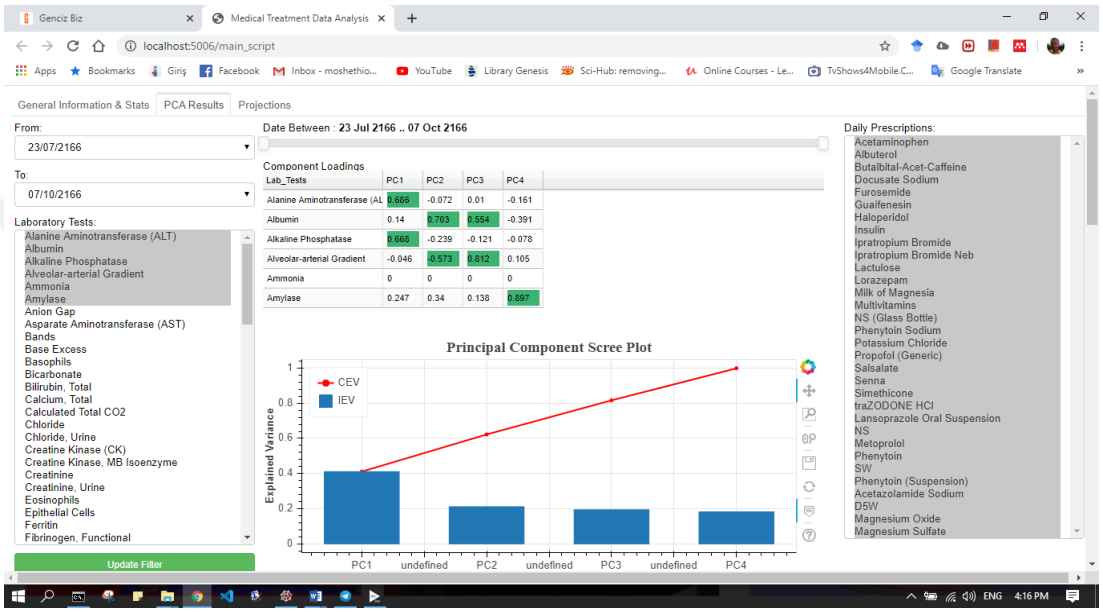


Figure 5.4 Selected laboratory tests PCA results

Figure 5.5 presents a 2D plot of laboratory test observation in principal component space, principal component 1 as the x-axis and principal component 2 as the y-axis. The arrows in the plot indicate the direction of the principal components. To which PC they belong to is displayed along with the leading variable and its associated loading in that direction when the user hovers the mouse on top of the arrows. Two side-by-side plot of this 2D plot is presented on the page with similar information by default. However, once the user selects specific date ranges, results are updated to show the difference between the selected date range and the rest of the data. The two plots show the difference in PC changes as per the selected date range which can be seen in Figure 5.8 and Figure 5.9. The leading (highest loading) variable for both PC1 and PC2 is written in bracket on the x- and y-axes labels respectively. For instance, PC1 has a

high positive association to Total Calcium where as PC2 is driven by a large positive association with PTT from Figure 5.5.

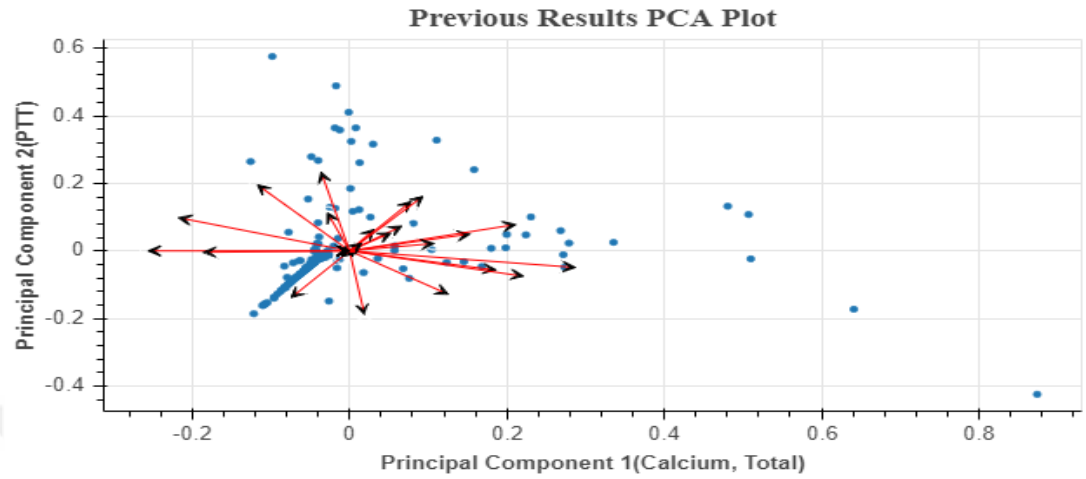


Figure 5.5 2D Principal component space plot

On the other hand, Figure 5.6 presents the top n principal component loadings for each original variable as a heatmap, in this case n=10. The colours represent the magnitude (contribution) of the loading of each variable in that direction as shown in the color legend on the right side. Figure 5.7 depicts the top 3 principal component variable loadings with pie charts. Similar to the other plots, for both figures, additional information such as laboratory test name, the loading and magnitude it contributes in that direction are displayed when the user hovers the mouse on each cell and slice, on the plots.

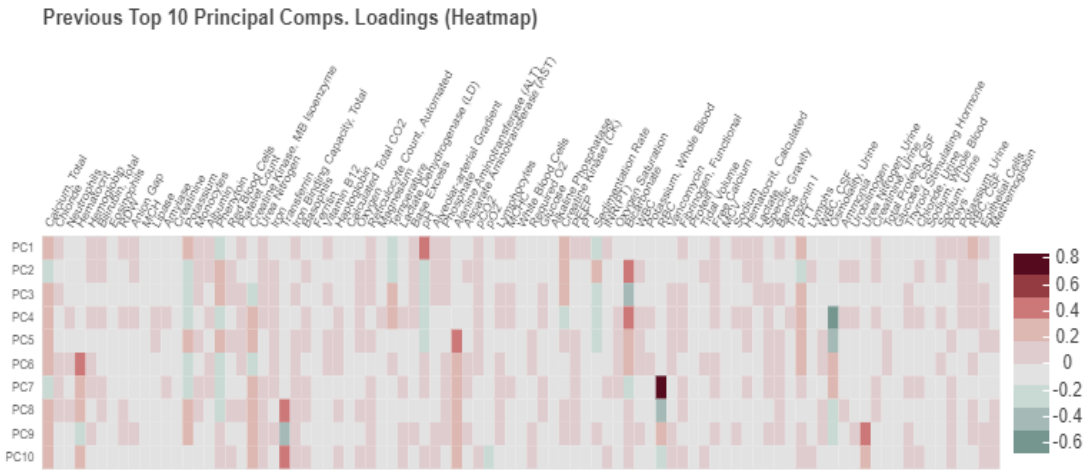


Figure 5.6 Top 10 PCs heatmap plot of variable loadings

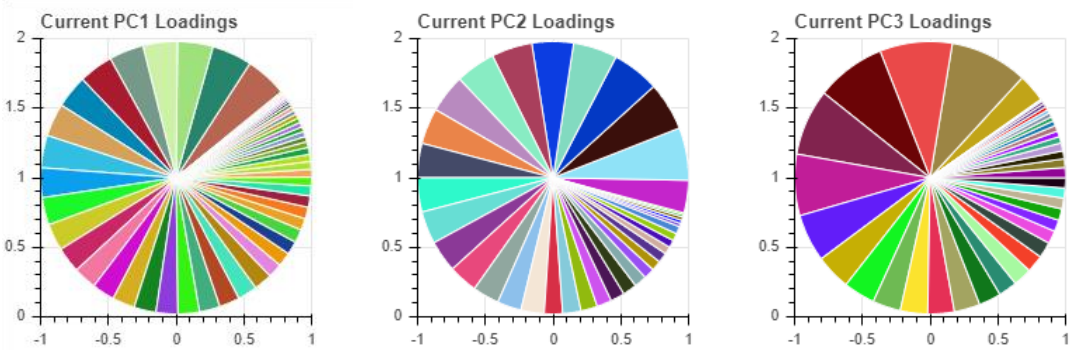


Figure 5.7 Top 3 PCs variable loadings

Sometimes the user might want to see analysis results of two date ranges for comparison. For instance, the user may want to see results from day 1 to day 3 data and compare this with results from day 4 to day 5 in order to see and scrutinize changes obtained based on the treatments applied. This may be necessary to see patient progress as well as establish cause-effect relationships from the data. Figure 5.8 presents a 2D plot of laboratory test observation in principal component space from previous day's laboratory test results, principal component 1 as the x-axis and principal component 2 as the y-axis based on user selected dates for comparison with current date results presented in Figure 5.9. The two plots are presented side by side when the user selects test result dates other than the initial test date.

For instance in this example, the user selected results between 18/08/20166 (actual dates are de-identified) and last date. Based on this PCA analysis of data from admission to 18/08/2166 (written on the figure title) are presented as previous results in Figure 5.8. Whereas the rest of the data starting from 18/08/2166 (written on the figure title) up-to the end are presented as current PCA results in Figure 5.9. With this the user will be able to contrast variances from previous results with the current one. By default the two figures present similar information, i.e. until the user selects specific dates and updates the results. Moreover, the user may want to see trends in change for some laboratory tests based on selected date ranges on a single plot. For example, the user wants to see change trends for laboratory test named ALT, for the selected date ranges in the scenario described above. This can be plotted as a time series plot so that the user will be able to see the trend. For this version of the prototype application, only

the top ten laboratory test with the highest contribution for the first principal component are plotted.

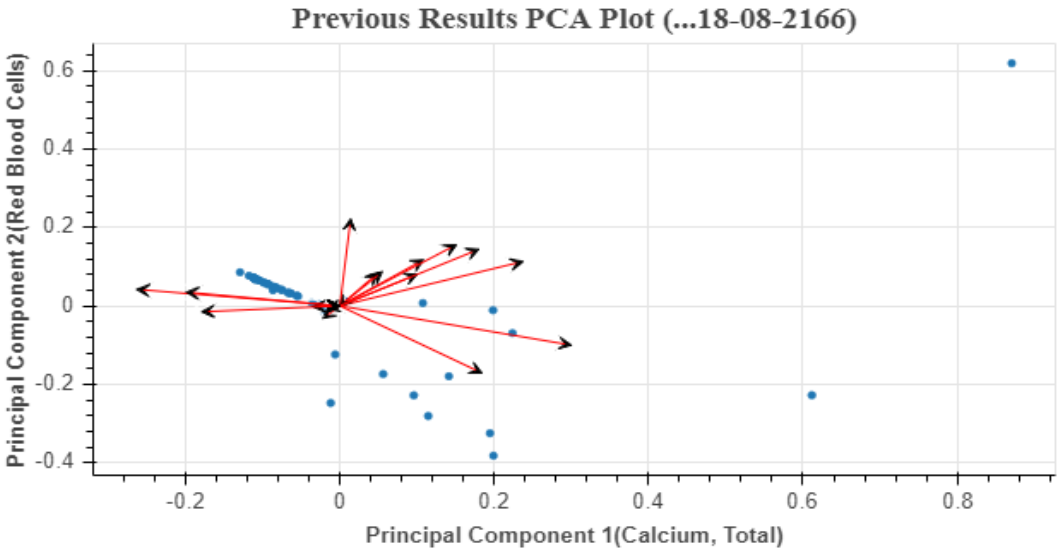


Figure 5.8 2D Principal component space plot (Previous Dates)

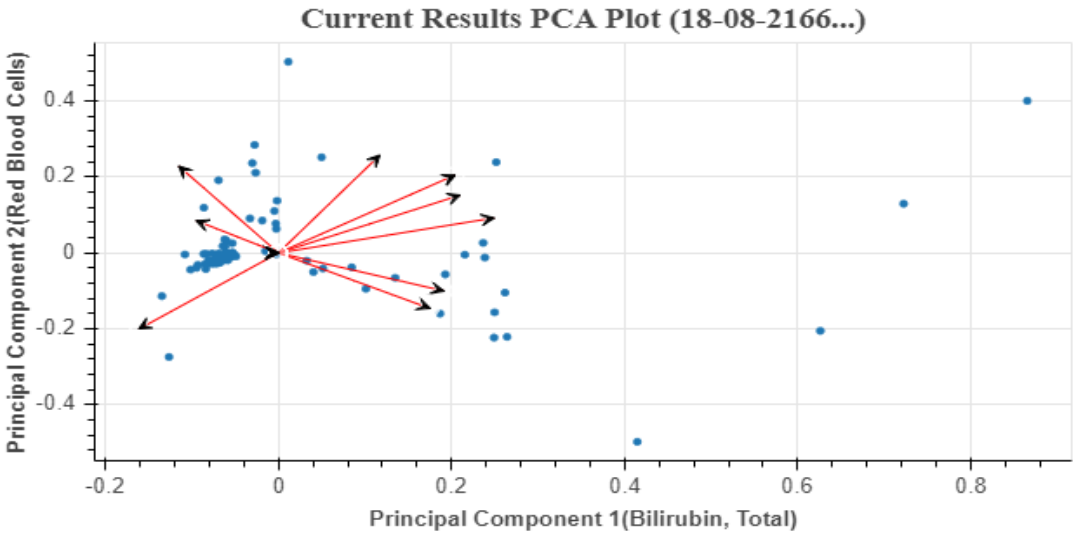


Figure 5.9 2D Principal component space plot (Current (Selected) Dates)

Finally, the last output from module 2 of the prototype application is presented in Figure 5.10. The figure presents top 10 laboratory tests under principal component 1 from current results analysis based on Figure 5.8 and Figure 5.9. The plot contains contributions (loadings) of each of the top 10 laboratory test under each principal component from both previous and current results as a trend. With this the user will be able to see trends in contribution for those top 10 laboratory tests.

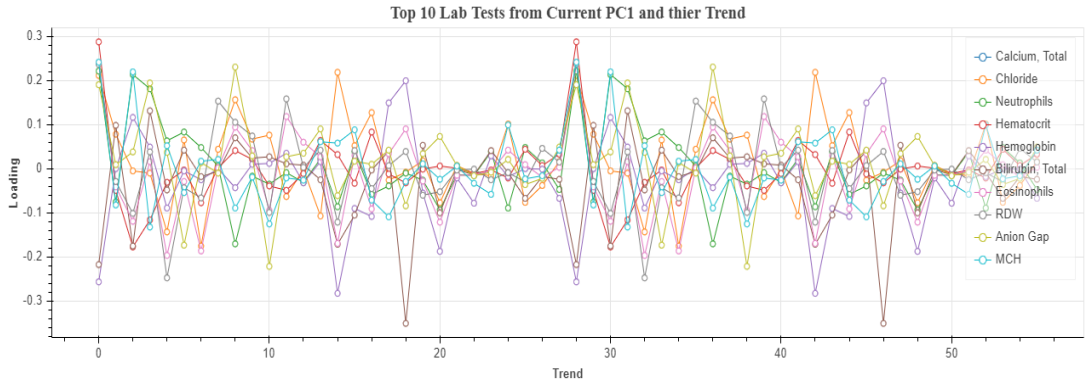


Figure 5.10 Top ten PCs variable loading trends

It is worth mentioning that every plot presented so far has a tooltip effect, which displays additional and required information while hovering a mouse over it. The prototype application is used as a main tool for feeding selected patient data for analysis and result presentation. However, direct python console outputs and matplotlib plots were used to prepare results required for these thesis document when necessary.

Three predictive models were implemented for future laboratory test value prediction as an additional tool under the third module. This predictive models can help the medical practitioner see future values and anticipate outcomes to act proactively. This, along with the PCA analysis described above will provide the medical practitioner a more robust system for decision making. This predictive models are also implemented as a web based application using bokeh and python. A sample result of these three models from the prototype application is presented in Figure 5.11 below for a single laboratory test (the user can be able to select many laboratory tests at the same time to see their predictions simultaneously). The models use past historical data to predict future three values. That is historical data will be used to train the model and predict three future time steps. A threshold of three future time-steps is used in this version of the application, however, it can be made to predict n future time-steps with minor modification. Similar to the other modules, module three also incorporates user controls. These controls include an option (date slider) for the user to select specific date ranges and see the predictions for the next three future time-steps. Furthermore, the user can also be able to select one or more than one laboratory test at a time for prediction.

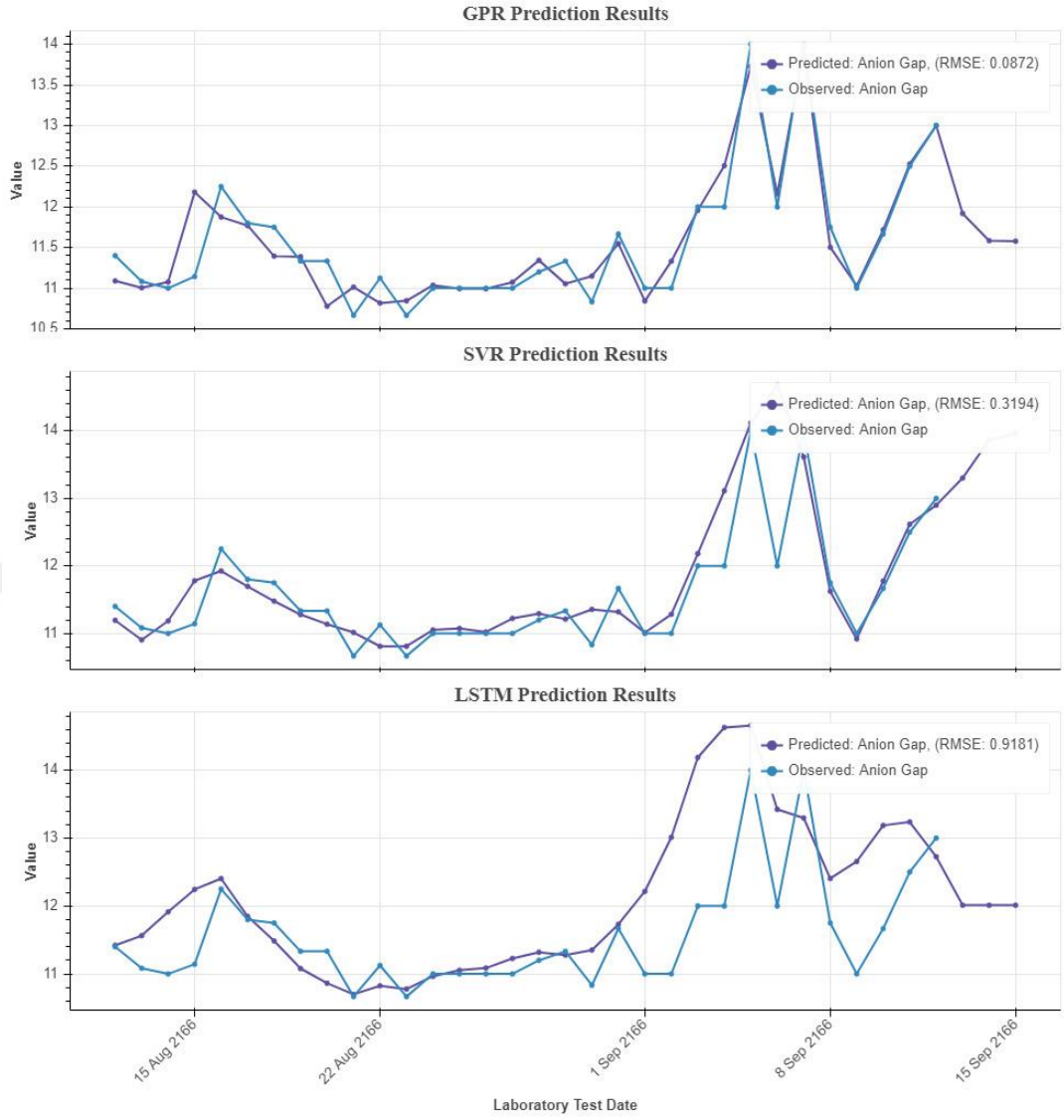


Figure 5.11 Prediction module results (only for Anion Gap laboratory test results)

The three predictive models present the flexibility to choose the best prediction result depending on the prevailing situations. However, technically we can see that GPR prediction has the least root mean square error of 0.087 compared to the other two models which is 0.32 and 0.92 for SVR and LSTM respectively for the selected laboratory test. This makes Gaussian process regression the best performer in the group for the selected laboratory test technically. However, the user might impose specific healthcare field related criteria for the selection of the best predicted values up-on seeing the results.

5.4 Summary

In this chapter, we presented a web-based prototype application for analysing and monitoring patient progress based on our hypothesis for establishing cause-effect relationships from medical treatment data. The application uses principal component analysis as a primary method for patient change monitoring from longitudinal data. The application is developed using Python and Bokeh Server. Moreover, the overall task is carried out on a 64-bit, Intel Core i5 personal computer with an 8GB RAM and 2.60GHz CPU speed. The application is used for experimental analysis for using the proposed model with the selected observational data. It can be used to apply the aforementioned model hypothesis testing method for every patient data and collect results.

The study, with the help of the prototype application, retrospectively demonstrated the capability of PCA as an early warning system to monitor and alert clinicians about patients, thereby providing opportunities for timely and proactive interventions. The approach provides the means to support clinical decision making and allow an efficient patient tailored care for improved outcomes.

CHAPTER SIX

CONCLUSION AND RECOMMENDATIONS

6.1 Conclusions

Many factors influence the success, failure and interpretation of medical treatments. Due to this, many issues must be considered when interpreting the results of any clinical laboratory test. Generally, normal reference ranges are used to decide what is normal and what is abnormal. However, it is believed that using only normal reference ranges for decision for critically ill patients in intensive care unit setting might not be enough. Moreover, minor changes that are with in normal reference ranges might reveal significant information, if they are presented with changes in other laboratory tests i.e. cumulative effect may reveal hidden insights.

The study presented a novel non-disease specific model that can observe patient daily clinical changes and provide non-disease specific analysis of patient progress over time. Moreover, by providing the daily prescriptions along with the analysis results a healthcare professional can be able to establish cause-effect relationships. Given these facts, the results can be used to decide what treatment or therapy to prescribe or which diagnosis to perform further. Fusing and presenting the daily changes with the daily prescriptions the patient had used, a healthcare professional can be able to establish cause-effect relationships. Besides it can also be used to decide which laboratory test to perform further and/or exclude from further analysis. It can also help decide which prescriptions to avoid and prescribe additional medicines. Furthermore, the effectiveness of a treatment and the improvement or cure of a disease can be influenced by multiple factors. The observation in a single patient may suggest the likelihood of a new property of a drug, or an adverse effect on the patient. This may not insure it happened due to causality, however, it can help rule out the possibility of coincidence between the clinical interventions and the outcomes. It is through causality we can be able to infer the behaviour of a medical treatment.

In this study causal inference implies the process of uncovering causal relationships from medical treatment data. Nonetheless, it does not mean that we remove the need for expert human input and judgment, but rather provide a tool to assist them make informed decisions. It is worth mentioning that no matter how detailed or clean the data is, machine learning models cannot avoid unknown or unprecedented aspects concomitant with a particular intervention that may explain an obvious outcome change. The study proposed a tool to assist healthcare professionals in daily clinical routine practices. Results showed that the approach, if fused with other machine learning models, presents a promising future for real-time patient monitoring in ICU settings. It can also help anticipate and avoid life-threatening conditions from happening proactively.

In any statistical model, where PCA is not an exception, model validation is imperative to generalize the results of a proposed model. Non-parametric model validation methods such as permutation and bootstrap tests are preferred way for non-parametric machine learning models such as PCA. By applying these non-parametric methods different matrices can be generated by permutation or resampling of the data, and their Eigenvalues and Eigenvectors will no longer be the same. The study applied both bootstrap and permutation testing for PCA model validation. Results show that the PCA model performs well with an acceptable significance level. In addition, causal impact analysis of the captured changes prove that the approach performs well in capturing daily changes. For predictive model validation, RMSE is used as a performance measurement metrics. Our results validate that the proposed method can be combined with other strategies to improve causal inference for critically ill patients. The proposed approach may help physicians feel confident about their decisions. Nevertheless, we would like to emphasize that any tool developed out of this approach is not meant to replace or undermine the skills and instincts of the medical practitioners. It is only meant to provide an alternative or a second eye for the users in presenting hidden insights. When fully realized, machine learning models could analyse longitudinal medical data to provide a second eye to the healthcare professionals. Principal component analysis is an interesting approach for patient

monitoring because it holds several advantages in observational and exploratory studies such as the ones discussed in this study.

Based on the analyses and experiments conducted on the selected data, PCA shows a promising future for monitoring patient progress in ICU setting. Coupled with other machine learning models, the researchers believe that it will be a vital addition to the modernization of general healthcare service delivery and the achievement of individualized healthcare. Moreover, the study by Holzinger, Schantl, Schroettner, Seifert, & Verspoor (2014) emphasizes the importance of integrative analysis that involves systematically merging various datasets and different algorithms together. It is in view of these facts, the researchers conducted multiple experiments on various machine learning models for establishing cause-effect relationships from medical treatment data on the MIMIC-III dataset. Finally, it is worth stating that the machine learning models implemented for the researched problem present a promising future for the advancement of effective healthcare delivery. As pointed out in this study, the application of machine learning methods to improve causal inference in observational studies is open to much additional investigations.

6.2 Recommendations

Machine learning models can identify hidden patterns and relationships in certain diseases from electronic health records. They act as a second pair of eyes in monitoring patient health and assist physicians. Moreover, they can also be able to help healthcare professionals make informed, timely, lifesaving, and effective decisions. In addition, the decision-making process in clinical medicine can be supported and facilitated with the appropriate selection and application of relevant machine learning models.

There are lots of research areas in in the healthcare sector that can be automated and made smarter with the application of machine learning models. With great care and caution, the same theory holds true for causal inference in healthcare. This study showed that PCA can be used as part of a tool for establishing cause-effect relationships from medical treatment data. Nevertheless, to assist researchers and stakeholders in the field, it will be of great paramount if the proposed method is fused

with other machine learning frameworks and models for a better and full-fledged application. The study took a great deal of time to come up with the selected machine learning models for establishing cause-effect relationships from medical treatment data. However, we believe that there are areas that still need a great deal of work and improved upon. Additional investigation need to be conducted to fuse this approach with other probabilistic and machine learning approaches to provide a better and robust tool for the medical practitioners.

Finally this research work suggests the following items as a future work:

- Prospective study of the application of the proposed model.
- Comparative study of different causal inference approaches such as Bayesian Networks and causal inference tools developed so far.
- Extending this work by incorporating additional and useful ML models in the area of causal inference.
- Comparison of hybrid approaches. For instance, ML models with RCT.

REFERENCES

Abebe, M., & Sevinç, S. (2020). Imputing Missing Values using Regression: the case of Electronic Medical Records. *In Fourth International Students Science Congress*, 46.

Alzamora, P., Nguyen, Q. V., Simoff, S., & Catchpoole, D. (2012). A novel 3D interactive visualization for medical data analysis. In *Proceedings of the 24th Australian Computer-Human Interaction Conference* (19–25). Melbourne, Australia: Association for Computing Machinery.

Ananth. (2020). *Machine Learning – Intelligent Decisions based on Data*. Retrieved May 5, 2021, from https://witanworld.com/article/2020/08/09/machinelearning/comment-page-18/

Angus, D. C. (2015). Fusing randomized trials with big data: The key to self-learning health care systems? *JAMA - Journal of the American Medical Association*, *314*(8), 767–768.

Awad, M., & Khanna, R. (2015). Support vector regression. In *Efficient Learning Machines* (67–80). Berkeley, CA: Apress.

Babaoğlu, I., Findik, O., & Bayrak, M. (2010). Effects of principle component analysis on assessment of coronary artery diseases using support vector machine. *Expert Systems with Applications*, *37*(3), 2182–2185.

Babič, F., Vadovský, M., Muchová, M., Paralič, J., & Majnarić, L. (2017). Simple understandable analysis of medical data to support the diagnostic process. In *SAMI 2017 - IEEE 15th International Symposium on Applied Machine Intelligence and Informatics*, 153–158.

Bokeh Development Team. (2018). *Bokeh: Python library for interactive visualization.* Retrieved October 20, 2020, from https://docs.bokeh.org/en/latest/docs/user_guide/server.html

Brodersen, K. H., Gallusser, F., Koehler, J., Remy, N., & Scott, S. L. (2015). Inferring causal impact using bayesian structural time-series models. *Annals of Applied Statistics*, *9*(1), 247–274.

Brodersen, K. H., & Hauser, A. (2015). *CausalImpact: An R package for causal inference using Bayesian structural time-series models*. Retrieved December 9, 2020, from https://google.github.io/CausalImpact/CausalImpact.html

Cai, X., Perez-Concha, O., Coiera, E., Martin-Sanchez, F., Day, R., Roffe, D., & Gallego, B. (2016). Real-time prediction of mortality, readmission, and length of stay using electronic health record data. *Journal of the American Medical Informatics Association*, *23*(3), 553–561.

Clifton, L., Clifton, D. A., Pimentel, M. A. F., Watkinson, P. J., & Tarassenko, L. (2014). Predictive monitoring of mobile patients by combining clinical observations with data from wearable sensors. *IEEE Journal of Biomedical and Health Informatics*, *18*(3), 722–730.

Dash, S., Shakyawar, S. K., Sharma, M., & Kaushik, S. (2019). Big data in healthcare: management, analysis and future prospects. *Journal of Big Data*, *6*(1), 1-25.

Deo, R. C. (2015). Basic science for clinicians. *American Heart Association*, 1920–1930.

Ebden, M. (2008). Gaussian processes: A quick introduction. *ArXiv,* 1-13.

Gharagyozyan, H. (2019). *A practical application of machine learning in medicine*. Retrieved June 8, 2020, from https://www.macadamian.com/learn/a-practical-application-of-machine-learning-in-medicine/

Goldberger, A. L., Amaral, L. A. N., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., … Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. *Journal of the American Heart Association*, *101*(23), e215-e220.

HealthIT.gov. (2020). *What is an electronic health record (EHR)?*. Retrieved October 2, 2020, from https://www.healthit.gov/faq/what-electronic-health-record-ehr

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Holzinger, A., & Jurisica, I. (2014). Knowledge discovery and data mining in biomedical informatics: The future is in integrative, interactive machine learning solutions. *Knowledge Discovery and Data Mining, LNCS, 8401*, 1–18.

Holzinger, A., Schantl, J., Schroettner, M., Seifert, C., & Verspoor, K. (2014). Biomedical text mining : state-of-the-art, open problems and future challenges. *Knowledge Discovery and Data Mining, LNCS, 8401*, 271–300.

Jaadi, Z. (2021). *A Step-by-Step Explanation of Principal Component Analysis (PCA)*. Retrieved May 5, 2021, from https://builtin.com/data-science/step-step-explanation-principal-component-analysis

Johnson, A. E. W., Pollard, T. J., Shen, L., Lehman, L. W. H., Feng, M., Ghassemi, M., … Mark, R. G. (2016). MIMIC-III, a freely accessible critical care database. *Nature Scientific Data*, *3*, 1–9.

Kara, S., & Dirgenali, F. (2007). A system to diagnose atherosclerosis via wavelet transform, principal component analysis and artificial neural networks. *ScienceDirect: Expert Systems with Applications*, *32*(2), 632–640.

Ketchersid, T. (2013). Big data in nephrology : friend or foe ? *Blood Purification*, *24592*, 160–164.

Kleinberg, S., & Hripcsak, G. (2011). A review of causal inference for biomedical informatics. *Journal of Biomedical Informatics*, *44*(6), 1102–1112.

Lavrač, N., Kononenko, I., Keravnou, E., Kukar, M., & Zupan, B. (1998). Intelligent data analysis for medical diagnosis: Using machine learning and temporal abstraction. *ACM AI Communications*, *11*(3), 191–218.

Lay-Flurrie, S. (2016). *Big data in healthcare: problems and potential.* Retrieved September 28, 2020, from https://www.phc.ox.ac.uk/news/blog/big-data-in-healthcare-problems-and-potential

Li, C. Y., Konomis, D., Neubig, G., Xie, P., Cheng, C., & Xing, E. (2014). Convolutional neural networks for medical diagnosis from admission notes. *ArXiv*, 1–16.

Li, J., Zhang, Y., & Tian, Y. (2016). Medical big data analysis in hospital information system. *Big Data on Real-World Applications*. Crotia:IntechOpen

Linden, A., & Yarnold, P. R. (2016). Combining machine learning and matching techniques to improve causal inference in program evaluation. *Journal of Evaluation in Clinical Practice*, *22*(6), 864–870.

Luo, Y., Szolovits, P., Dighe, A. S., & Baron, J. M. (2016). Using machine learning to predict laboratory test results. *American Journal of Clinical Pathology*, *145*(6), 778–788.

Mathukia, C., Fan, W., Vadyak, K., Biege, C., & Krishnamurthy, M. (2015). Modified early warning system improves patient safety and clinical outcomes in an academic community hospital. *Journal of Community Hospital Internal Medicine Perspectives*, *5*(2), 26716.

Minasyan, A. (2016). *What is the significance of eigenvectors in PCA (principal component analysis)?.* Retrieved October 13, 2020, from https://www.quora.com/What-is-the-significance-of-eigenvectors-in-PCA-principal-component-analysis

Mohamadlou, H., Lynn-palevsky, A., Barton, C., Chettipally, U., Shieh, L., Calvert, J., … Das, R. (2018). Prediction of acute kidney injury with a machine learning algorithm using electronic health record data. *Canadian Journal of Kidney Health and Disease*, *5*, 1–9.

Monleon-Getino, A., & Canela-Soler, J. (2017). Causality in medicine and its relationship with the role of statistics. *Biomedical Statistics and Informatics*, *2*(2), 61–68.

Neill, S. P., & Hashemi, M. R. (2018). Ocean Modelling for Resource Characterization. *In Fundamentals of Ocean Renewable Energy*, 193–235. Elsevier.

Olah, C. (2015). *Understanding LSTM Networks*. Retrieved May 5, 2021, from https://colah.github.io/posts/2015-08-Understanding-LSTMs/

Palaniappan, S., & Awang, R. (2008). Intelligent heart disease prediction system using data mining techniques. *AICCSA 08 - 6th IEEE/ACS International Conference on Computer Systems and Applications*, 108–115.

Patil, R. (2019). *How to become a successful healthcare data analyst.* Retrieved October 2, 2020, from https://www.kdnuggets.com/2019/11/become-successful-healthcare-data-analyst.html

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011a). *Scikit-learn: machine learning in python.* Retrieved October 13, 2020, from https://scikit-learn.org/stable/modules/gaussian_process.html

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, É. (2011b). Scikit-learn: machine learning in python. *Journal of Machine Learning Research*, *12*(85), 2825–2830.

Pham, T., Tran, T., Phung, D., & Venkatesh, S. (2017). Predicting healthcare trajectories from medical records: A deep learning approach. *Journal of Biomedical Informatics*, *69*, 218–229.

Physionet. (2018). *MIMIC:Requesting Access*. Retrieved May 5, 2021, from https://mimic.physionet.org/gettingstarted/access/

Pirracchio, R., Cohen, M. J., Malenica, I., Cohen, J., Chambaz, A., Cannesson, M., … Hubbard, A. (2019). Big data and targeted machine learning in action to assist medical decision in the ICU. *Anaesthesia Critical Care and Pain Medicine*, *38*(4), 377–384.

Polat, K., & Güneş, S. (2007). An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease. *Digital Signal Processing: A Review Journal*, *17*(4), 702–710.

Prosperi, M., Guo, Y., Sperrin, M., Koopman, J. S., Min, J. S., He, X., … Bian, J. (2020). Causal inference and counterfactual prediction in machine learning for actionable healthcare. *Nature Machine Intelligence*, 2(7), 369-375.

Prytherch, D. R., Smith, G. B., Schmidt, P. E., & Featherstone, P. I. (2010). ViEWS-Towards a national early warning score for detecting adult inpatient deterioration. *ScienceDirect: Resuscitation*, *81*(8), 932–937.

Richens, J. G., Lee, C. M., & Johri, S. (2020). Improving the accuracy of medical diagnosis with causal machine learning. *Nature Communications*, 1–9.

Rocca, E., & Anjum, R. L. (2020). Causal evidence and dispositions in medicine and public health. *International Journal of Environmental Research and Public Health*, *17*(6), 1–18.

Rose, S., & Rizopoulos, D. (2019). Machine learning for causal inference in biostatistics. *Biostatistics (2020)*, *21*(2), 336–338.

Russo, F. (2017). *Causation and correlation in medical science: theoretical problems*. (T. Schramme & S. Edwards, Eds.), *Handbook of the Philosophy of Medicine*. Dordrecht: Springer Science+Business Media.

Sarkar, T. (2020). *AI and machine learning for healthcare.* Retrieved June 10, 2020, from https://towardsdatascience.com/ai-and-machine-learning-for-healthcare-7a70fb3acb67

SAS. (2021). *Machine Learning: What it is and why it matters*. Retrieved May 5, 2021, from https://www.sas.com/en_be/insights/analytics/machine-learning.html

Sathyanarayana, S., & Amarappa, S. V. (2014). Data classification using Support vector Machine (SVM), a simplified approach. *International Journal of Electronics and Computer Science Engineering*, 3(4), 435–445.

Sayad, S. (2010). *Support vector regression.* Retrieved October 11, 2020, from https://www.saedsayad.com/support_vector_machine_reg.htm

Serokell. (2020, April 10). *Artificial intelligence vs. machine learning vs. deep learning: what's the difference?.* Retrieved March 24, 2021, from https://ai.plainenglish.io/artificial-intelligence-vs-machine-learning-vs-deep-learning-whats-the-difference-dccce18efe7f

Sit, H. (2019). *Quick Start to Gaussian Process Regression*. Retrieved May 5, 2021, from https://towardsdatascience.com/quick-start-to-gaussian-process-regression-36d838810319

Smith, G. B., Prytherch, D. R., Meredith, P., Schmidt, P. E., & Featherstone, P. I. (2013). The ability of the national early warning score (NEWS) to discriminate patients at risk of early cardiac arrest, unanticipated intensive care unit admission, and death. *ScienceDirect:Resuscitation*, *84*(4), 465–470.

Srikanth. (2019). *15 benefits of machine learning in health care*. Retrieved October 8, 2020, from https://www.techiexpert.com/benefits-of-machine-learning-in-health-care/

Stern, A. D., & Price, W. N. (2020). Regulatory oversight, causal inference, and safe and effective health care machine learning. *Biostatistics (Oxford, England)*, *21*(2), 363–367.

Tahmasebian, S., Ghazisaeedi, M., Langarizadeh, M., Mokhtaran, M., Mahdavi-Mazdeh, M., & Javadian, P. (2017). Applying data mining techniques to determine important parameters in chronic kidney disease and the relations of these parameters to each other. *Journal of Renal Injury Prevention*, *6*(2), 83–87.

Tomaszewski, J. E., Hipp, J., Tangrea, M., & Madabhushi, A. (2014). Machine vision and machine learning in digital pathology. In *Pathobiology of Human Disease: A Dynamic Encyclopedia of Disease Mechanisms* (3711–3722). San Diego: Academic Press.

Wang, C. F., Li, J., Ma, K. L., Huang, C. W., & Li, Y. C. (2014). A visual analysis approach to cohort study of electronic patient records. In *2014 IEEE International Conference on Bioinformatics and Biomedicine*, 521–528.

Werdiningsih, I., Hendradi, R., Nuqoba, B., & Ana, E. (2019). Identification of Risk Factors for Early Childhood Diseases Using Association Rules Algorithm with Feature Reduction. *Cybernetics and Information Technologies*, *19*(3), 154–167.

Widanagamaachchi, W., Livnat, Y., Bremer, P.-T., Duvall, S., & Pascucci, V. (2017). Interactive visualization and exploration of patient progression in a hospital setting. In *AMIA2017 - AMIA Annual Symposium proceedings*, 1773–1782.

Witten, I. H., & Frank, E. (2006). *Witten IH, Frank E: Data mining: practical machine learning tools and techniques*. *Morgan Kaufmann* (Second ed., Volume 5). San Francisco: Morgan Kaufmann.

Yazdani, A., & Boerwinkle, E. (2015). Causal inference in the age of decision medicine. *Journal of Data Mining Genomics Proteomics*, *6*(1), 139–148.

Ye, C., Wang, O., Liu, M., Zheng, L., Xia, M., Hao, S., … Ling, X. (2019). A real-time early warning system for monitoring inpatient mortality risk: Prospective study using electronic medical record data. *Journal of Medical Internet Research*, *21*(7), 1–13.

Zhang, B., Ren, H., Huang, G., Cheng, Y., & Hu, C. (2019). Predicting blood pressure from physiological index data using the SVR algorithm. *BMC Bioinformatics*, *20*(1), 1–15.

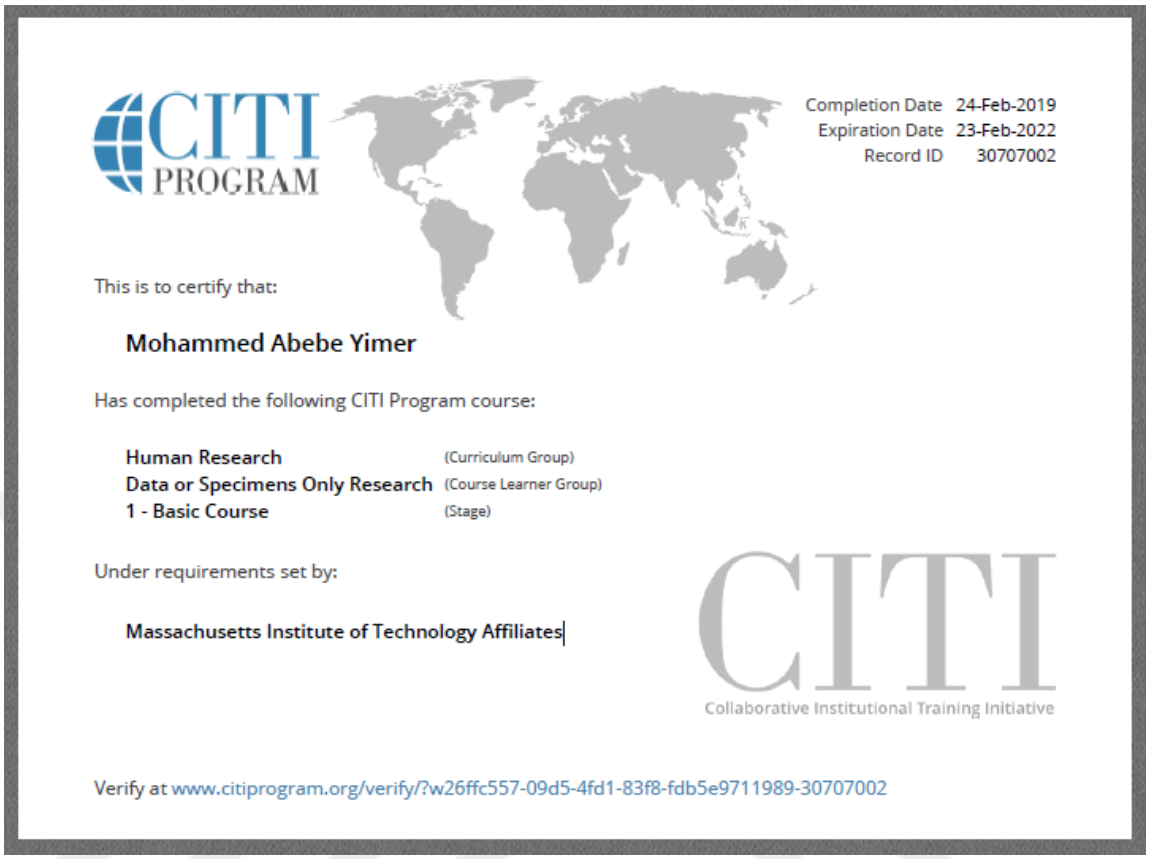
Zoltan, C. (2018). *SVM and Kernel SVM. Learn about SVM or Support Vector*. Retrieved May 5, 2021, from https://towardsdatascience.com/svm-and-kernel-svm-fed02bef1200

APPENDICES

Appendix 1: Acronyms and Abbreviations

AI	Artificial Intelligence
ANN	Artificial Neural Networks
CI	Confidence Interval
CITI	Collaborative Institutional Training Initiative
CKD	Chronic Kidney Disease
DL	Deep Learning
EHR	Electronic Health Records
EMR	Electronic Medical Records
EWS	Early Warning Systems
GP	Gaussian Processes
GPR	Gaussian Process Regression
GR	Gaussian Regression
HIPAA	Health Insurance Portability and Accountability Act
ICU	Intensive Care Unit
IT	Information Technology
JSON	JavaScript Object Notation
LOS	Length of Stay
LSTM	Long-Short Term Memory
MAE	Mean Absolute Error
MIMIC	Medical Information Mart for Intensive Care
ML	Machine Learning
MSE	Mean Squared Error
NLP	Natural Language Processing
OCR	Optical Character Recognition
ODA	Optimal Discriminant Analysis
PC	Principal Component
PCA	Principal Component Analysis
PID	Patient Identification
RBF	Radial Basis Function
RCT	Randomized Control Trials
ReLU	Rectified Linear Unit
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Networks
SQL	Structured Query Language
SVM	Support Vector Machines
SVR	Support Vector Regression
TVAF	Total Variance Accounted For

Appendix 2: CITI course completion certificate



CITI PROGRAM

Completion Date 24-Feb-2019
Expiration Date 23-Feb-2022
Record ID 30707002

This is to certify that:

Mohammed Abebe Yimer

Has completed the following CITI Program course:

Human Research	(Curriculum Group)
Data or Specimens Only Research	(Course Learner Group)
1 - Basic Course	(Stage)

Under requirements set by:

Massachusetts Institute of Technology Affiliates

CITI
Collaborative Institutional Training Initiative

Verify at www.citiprogram.org/verify/?w26ffc557-09d5-4fd1-83f8-fdb5e9711989-30707002

Appendix 3: Sample Python Code

```
"""
@author: Mohammed A.Yimer
============================================================
Establishing Causal-relationship from Medical Treatment Data
============================================================
"""
Import required libraries

#############################################################################
def doPCA(X, title, counter):
    # Applying PCA
    pca = PCA(n_components=0.99)
    pca.fit(X)
    X_transformed = pca.transform(X)
    # print(X_transformed)

    components = pca.components_
    exp_var = pca.explained_variance_
    exp_var_ratio = pca.explained_variance_ratio_
    cum_exp_var_ratio = np.cumsum(pca.explained_variance_ratio_)

    # Plotting results
    string = 'PC'
    PCs = np.empty(len(exp_var))
    PCs = [string+str(i+1) for i in range(len(exp_var))]
    comps = pd.DataFrame(data=np.round(components, 3), columns=X.columns, index=PCs)
    comps.index.name = 'PCNo'
    comps.to_csv('data\sample_results\PC data\\'+str(counter)+'.csv', index=True)

    # prepare 2d plots
    plot_2d(comps, title, counter)

    transformed_data = pd.DataFrame(data=X_transformed, columns=PCs)
    transformed_data.to_csv('data\sample_results\\transformed data\\'+str(counter)+'_transformed.csv')

    ''' SCREE PLOT '''
    fig = plt.figure(str(counter)+' Scree Plot & PCs Pie Chart')
    fig.suptitle(title)
    #plt.subplot(121)
    x = np.zeros(len(pca.explained_variance_ratio_))

    width = 1/2
    plt.plot(x,cum_exp_var_ratio, 's-', color='red', markersize=3, label='Commulative explained
variance')
    plt.bar(x, exp_var_ratio, width, color='blue', label='Individual explained variance')

#############################################################################
# Read data
df = pd.read_csv('data\lab_data\sub-96309.csv', parse_dates=[0], index_col=[0])
df_presc = pd.read_csv('data\prescription_data\96309_presc.csv', parse_dates=[4, 5], index_col=[0])

# remove a column with no variance and impute missing values
labData  = df[df.columns[df.isnull().sum() != df.shape[0]]]
temp = labData.iloc[:, :-1]
```

```
# Impute the missing values using scikit-learn SimpleImpute Class
imp_mean = SimpleImputer(strategy='most_frequent')
imp_mean.fit(temp)
labData = pd.DataFrame(data=imp_mean.transform(temp), columns=labData.columns[:-1],
index=labData.index)

# Remove Lab. Tests with only single values (no variance)
nunique = labData.apply(pd.Series.nunique)
cols_to_drop = nunique[nunique == 1].index
labData = labData.drop(cols_to_drop, axis=1, inplace=False)

# Apply Feature Scaling
sc = StandardScaler()
pcaData = sc.fit_transform(labData)
labData = pd.DataFrame(data=pcaData, columns=labData.columns, index=labData.index)
labData['Measurement_Date'] = labData.index

# Count daily measurements and select unique measurement date/year and month
daily_msrt = labData.groupby(labData['Measurement_Date'].dt.date)['Measurement_Date'].count()
unique_dates = daily_msrt.index.values

# loop through the data and extract based on date
step = 0
i = 0
counter=1
while i < len(unique_dates):
   temp_df = labData[labData['Measurement_Date'].dt.date==unique_dates[i]]

   if (temp_df.shape[0]<=3):
      count=i+1
      while count<len(unique_dates):
         temp = labData[labData['Measurement_Date'].dt.date==unique_dates[count]]
         frames = [temp_df, temp]
         temp_df  = pd.concat(frames)#, ignore_index=True)
         if temp_df.shape[0]>3:
            count += 1
            break
         count += 1
      size = len(temp_df["Measurement_Date"].map(lambda t: t.date()).unique())
      step=size
      callPCA_function(temp_df, size, counter)
      counter+=1
   else:
      step = 1
      temp_df = labData[labData['Measurement_Date'].dt.date==unique_dates[i]]
      size = len(temp_df["Measurement_Date"].map(lambda t: t.date()).unique())
      callPCA_function(temp_df, size, counter)
      counter+=1
   i = i + step

del cols_to_drop, imp_mean, nunique, pcaData, sc, temp, count, counter, daily_msrt, frames, i
del size, step, temp_df, unique_dates
```

Appendix 4: Sample Bokeh Code

```
# Make plot with histogram and return tab
def pca_tab(labData, prescData, subj_id):
    # update data based on slider values and call plot functions
    def update_data(attrname, old, new):
        start_date_frmtd = datetime.datetime.date(date_slider.value_as_datetime[0])
        end_date_frmtd = datetime.datetime.date(date_slider.value_as_datetime[1])
        # Get the current selected items (lab test) list
        selected_tests = lab_test_list.value
        df_new = labData[selected_tests]

        # extract data from the orginal dataframe based on the specified dates
        mask = (df_new.index.strftime('%Y-%m-%d') >= start_date_frmtd.
              strftime('%Y-%m-%d')) & (df_new.index.strftime('%Y-%m-%d') <=
                    end_date_frmtd.strftime('%Y-%m-%d'))
        new_src = df_new.loc[mask]
        X = new_src.iloc[:, :]

        # prescriptions
        mask_presc = (prescData.startdate >= start_date_frmtd.strftime('%Y-%m-%d')) & (
              prescData.startdate <= end_date_frmtd.strftime('%Y-%m-%d'))
        updated_presc = prescData.loc[mask_presc]

        # call function for update
        comps, transformed_data, exp_var, cum_exp_var_ratio = doPCA(X.astype(float), subj_id, title)
        top_middle.children[2],top_middle.children[3]  = comp_loading_tbl(comps, exp_var,
         cum_exp_var_ratio)
        top_right.children[1] = presc_data(updated_presc, 32)
        n = 10

        # prev. date results analysis
        initial_date = datetime.datetime.strptime(included_dates[0], '%d/%m/%Y')
        if(start_date_frmtd > initial_date):
            DD = datetime.timedelta(days=1)
            earlier = start_date_frmtd - DD
            # extract data from the orginal dataframe based on the specified dates
            new_mask = (df_new.index >= initial_date.strftime('%Y-%m-%d')) & (
                  df_new.index <= earlier.strftime('%Y-%m-%d'))
            src = df_new.loc[new_mask]
            X_new = src.iloc[:, :]
            comps_prev, transformed_data_prev, exp_var_prev, cum_exp_var_ratio_prev = doPCA(
                X_new.astype(float), subj_id, title)
            option = ['...'+str(start_date_frmtd.strftime('%d-%m-%Y')), str(start_date_frmtd.strftime('%d-
%m-%Y'))+'...']
            middle.children[0], bottom_middle.children[1], bottom.children[0], bottom_left.children[1],
bottom_right.children[1] = plot_pca_results(
                transformed_data_prev, transformed_data, comps, comps_prev, title, n, option)
        else:
            option = [str(start_date_frmtd.strftime('%d-%m-%Y'))+'...', str(start_date_frmtd.strftime('%d-
%m-%Y'))+'...']
            middle.children[0], bottom_middle.children[1], bottom.children[0], bottom_left.children[1],
bottom_right.children[1] = plot_pca_results(
                transformed_data, transformed_data, comps, comps, title, n, option)
        # end of update1 function

    def buttonClick_update_data():
```

```
# Get the current selected items (lab test) list
selected_tests = lab_test_list.value
df_new = labData[selected_tests]

# get the selected date range
start_date_frmtd = datetime.datetime.strptime(from_date.value, '%d/%m/%Y')
end_date_frmtd = datetime.datetime.strptime(to_date.value, '%d/%m/%Y')

# extract data from the orginal dataframe based on the specified dates
mask = (df_new.index >= start_date_frmtd.strftime('%Y-%m-%d')) & (
       df_new.index <= end_date_frmtd.strftime('%Y-%m-%d'))
src = df_new.loc[mask]
X = src.iloc[:, :]

# prescriptions
mask_presc = (prescData.startdate >= start_date_frmtd.strftime('%Y-%m-%d')) & (
       prescData.startdate <= end_date_frmtd.strftime('%Y-%m-%d'))
updated_presc = prescData.loc[mask_presc]

# call function for update
comps, transformed_data, exp_var, cum_exp_var_ratio = doPCA(X.astype(float), subj_id, title)
top_middle.children[1],top_middle.children[2]  = comp_loading_tbl(comps, exp_var,
 cum_exp_var_ratio)
top_right.children[1] = presc_data(updated_presc, 32)
n = 10

# prev. date results analysis
initial_date = datetime.datetime.strptime(included_dates[0], '%d/%m/%Y')
if(start_date_frmtd > initial_date):
    DD = datetime.timedelta(days=1)
    earlier = start_date_frmtd - DD
    # extract data from the orginal dataframe based on the specified dates
    new_mask = (df_new.index >= initial_date.strftime('%Y-%m-%d')) & (
          df_new.index <= earlier.strftime('%Y-%m-%d'))
    src = df_new.loc[new_mask]
    X_new = src.iloc[:, :]
    comps_prev, transformed_data_prev, exp_var_prev, cum_exp_var_ratio_prev = doPCA(
       X_new.astype(float), subj_id, title)
    option = ['...'+str(start_date_frmtd.strftime('%d-%m-%Y')), str(start_date_frmtd.strftime('%d-
   %m-%Y'))+'...']
    middle.children[0], bottom_middle.children[1], bottom.children[0], bottom_left.children[1],
   bottom_right.children[1] = plot_pca_results(
       transformed_data_prev, transformed_data, comps, comps_prev, title, n, option)
else:
    option = [str(start_date_frmtd.strftime('%d-%m-%Y'))+'...', str(start_date_frmtd.strftime('%d-
   %m-%Y'))+'...']
    middle.children[0], bottom_middle.children[1], bottom.children[0], bottom_left.children[1],
   bottom_right.children[1] = plot_pca_results(
       transformed_data, transformed_data, comps, comps, title, n, option)

# end of update2 function
```