Forecasting Monthly Airline Passenger Numbers with Small Datasets Using Feature Engineering and a Modified Principal Component Analysis

Sara Al-Ruzaiqi

A Doctoral Thesis

Submitted in partial fulfilment

of the requirements for the award of

Doctor of Philosophy

of

Loughborough University

2019

Dedication

I dedicate this Thesis to the two most important people in my life my father and my mother (my beloved parents) for their endless support and unconditional love. My parents always believed in me even in times when I was full of doubt in myself. They were always there cheering me up and stood by me through the good and bad times.

I hope both of you are proud of me.

Abstract

In this study, a machine learning approach based on time series models, different feature engineering, feature extraction, and feature derivation is proposed to improve air passenger forecasting. Different types of datasets were created to extract new features from the core data. An experiment was undertaken with artificial neural networks to test the performance of neurons in the hidden layer, to optimise the dimensions of all layers and to obtain an optimal choice of connection weights – thus the nonlinear optimisation problem could be solved directly. A method of tuning deep learning models using H2O (which is a feature-rich, open source machine learning platform known for its R and Spark integration and its ease of use) is also proposed, where the trained network model is built from samples of selected features from the dataset in order to ensure diversity of the samples and to improve training. A successful application of deep learning requires setting numerous parameters in order to achieve greater model accuracy. The number of hidden layers and the number of neurons, are key parameters in each layer of such a network. Hyper-parameter, grid search, and random hyper-parameter approaches aid in setting these important parameters. Moreover, a new ensemble strategy is suggested that shows potential to optimise parameter settings and hence save more computational resources throughout the tuning process of the models. The main objective, besides improving the performance metric, is to obtain a distribution on some hold-out datasets that resemble the original distribution of the training data. Particular attention is focused on creating a modified version of Principal Component Analysis (PCA) using a different correlation matrix – obtained by a different correlation coefficient based on kinetic energy to derive new features. The data were collected from several airline datasets to build a deep prediction model for forecasting airline passenger numbers. Preliminary experiments show that fine-tuning provides an efficient approach for tuning the ultimate number of hidden layers and the number of neurons in each layer when compared with the grid search method. Similarly, the results show that the modified version of PCA is more effective in data dimension reduction, classes reparability, and classification accuracy than using traditional PCA.

Keywords: Feature Engineering; Deep Learning; Principle Component Analysis (PCA); algorithm; prediction.

Acknowledgements

بِسْمِ ٱللَّهِ ٱلرَّحْمَٰنِ ٱلرَّحِيمِ

First of all, I would like to address my most precious appreciation to my supervisor Dr Christian Dawson for his guidance and encouragement through this work. His profound knowledge, positive attitude, patient guidance, and valuable suggestions on my work guaranteed the completion of this Thesis. One simply could not wish for a better or friendlier supervisor. He has set an example of excellence as an instructor and a mentor. I have been extremely lucky to have a supervisor who cared so much about my work.

My special acknowledgements go to all those people who provide me with data for my experiments. My warm appreciation is due to the Public Authority for Civil Aviation, Directorate General of Meteorology, and Ministry of Tourism in Oman.

My warmly acknowledge go to his majesty Sultan Qaboos government the Ministry of Higher Education (my sponsor) and Cultural Attaché (Embassy of Oman) for providing the financial assistance and support throughout the research period.

I cannot find proper words to express my deep gratitude to my family and friends for their sincere encouragement and inspiration during this period, which helped to bring me into this stage of my life.

And, last but not least, I would like to thank all who have knowingly and unknowingly helped me and been involved in the successful completion of this report.

Sara Al-Ruzaiqi

Loughborough, England

20.09.2019

Abbreviations

ACF	Auto correlation function
GDP	Gross Domestic Product
ANN	Artificial neural network model
AR	Autoregressive models
ARIMA	Autoregressive Integrated Moving Average
ARIMAX	Auto Regressive Integrated Moving Average with Exogenous Input
BSM	Basic Structural models
CRM	Customer relationship management
DGP	Data generating process
H2O	Open source machine learning platform
MA	Moving Average models
MAD	Mean Absolute Deviation
MAE	Mean Absolute Error
MAPD	Mean Absolute Percentage Error
MAPE	Mean Absolute Percentage Division
MASE	Mean Absolute Scaled Square Error
MECA	Ministry of Environmental and Climate Affairs
MSA	Mean Absolute Error
MSE	Mean Square Error
MSPE	Mean Squared Prediction Error
PCA	Principal Component Analysis
PACF	partial auto correlation function
RAE	Relative Absolute Error
RMSE	Root Mean Squared Error
RMSE	Root Mean Squared Error
RRSE	Root Relative Squared Error
SES	Simple exponential smoothing
SMAPE	Symmetric Absolute Percentage Error
S	Seasonal component of a time series
T	Trend component of a time series
α,β,γ	Smoothing parameters for smoothing based approaches

$\hat{y}$	Vector of time series forecasts
ω	Combination weight vector
e	Unity vector
ε	Forecast error
$\hat{y}$	Time series forecast
$\hat{y}c$	Combined time series forecast
ω	Weights for linear forecast combination
φ	Dampening factor

Table of Contents

List of Figures

List of Tables

Chapter 1

2.1 Introduction

Knowing the future trend of passengers travelling through air transportation is of immense importance in today's world (Adrangi et al., 2001). It is important to 'forecast' the number of airline passengers as accurately as possible as such predictions can be used in many contexts ranging from simple initial planning to complicated business decisions (Carson et al., 2011). The airport facility of a country indicates the economic standard of that country (Kincaid, 2016). A global rise in overseas travel has meant that the number of passengers using airport facilities has sharply increased. In 2017, the International Air Transport Association (IATA, 2017) published their 62nd annual travel statics report based on data from its 290 airline members. IATA reports that a record number of travellers flew in 2017 between more city pairs than ever, and that a record-breaking 4.1 billion passengers flew on scheduled airline services that year. That was 280 million more than in 2016, representing a 7.3% increase year on year.

A number of different airlines operate within an airport, and to maintain a supply of facilities to meet demand accurate business and economic decisions need to be made (Tsui et al., 2011). Accurate forecasts of air transport activity are essential in the planning processes of states, airports, airlines, and other relevant bodies (Riga et al., 2009). Accurate forecasts also assist aircraft manufacturers in planning future aircraft types (in terms of size and range) and when to develop them (Cho, 2003; Cuhadar, 2014; Kulendran & Witt, 2003). Since passenger transport demand forecasting greatly affects the effectiveness of investment efficiency by adequacy and accuracy of the performance estimation, it is seen as a critical criterion for investors as well as for airlines (Market Research.com, 2017).

Air traffic forecasts are a key input into an airline's fleet planning and route network development, and are also used in the preparation of the airline's annual operating plan (Coshall, 2006). Furthermore, analysing and forecasting air travel demand may also assist an airline in reducing its risk through an objective evaluation of the demand side of the airline business (Cho, 2003; Cuhadar, 2014; Kulendran & Witt, 2003).

Identifying the potential impact of the future trend of airline passengers, this study intends to thoroughly analyse the pattern of airline passengers travelling through Muscat International Airport in Oman. The study aims to establish a prediction mechanism for future values of airline passenger numbers of this airport, which has experienced significant growth over recent years, as passenger numbers have more than doubled since 2009, when the airport served 4,556,502 passengers for the calendar year (OAMC, 2017). Oman is heavily reliant upon its air transport industry (ONA, 2016), due to the vast distances across the country, as well as between its urban centres, and also its location at the centre of the Middle East – lying at the junction of key trade routes leading north/south and east/west. According to the report of National Centre for Statistics and Information, Oman tourism is expected to be one of the largest industries in the country, since the number of tourists increased to 1.96 million in 2013 from 1.36 million in 2007 (NCSI, 2017). Based on the data reported by the World Travel & Tourism Council, the direct contribution of Oman tourism to national GPD is 3.3 percent in 2014; it generated 37,000 job positions, which is 3.5 percent of total employment (WTTC, 2014). These data indicate that Oman is still behind the UAE and Bahrain, but ahead of Kuwait, Saudi Arabia and Qatar. The Oman government has invested heavily in tourism and is currently implementing a major project of expanding and upgrading Muscat International Airport.

Air transport stimulates the growth of local economies, contributing to the development of companies, and increasing the competitiveness of businesses. It generates jobs, which also translates into the society's wealth. This creates possibilities of boosting the ease and flow of goods and people. All of this favours higher living standards, increasingly greater travel comfort and a wider choice of services offered by the aviation market.

However, Omani aviation is faced with the challenge of effectively satisfying the society's demand for air transport. Such demand is not limited to the throughput of air infrastructure, but also involves fitting it effectively into both the Omani and, primarily, the Middle Eastern transport systems. The biggest changes affecting the size and structure of demand for transport are taking place in the technological and innovative aspects of transport, in the structure and technologies of manufacturing, and in the society's lifestyle.

Opening the market to new carriers, greater competition and decreasing ticket prices attracts more people to air travel. The trend is expected to continue in the next few years, provided new airports appear and old ones are modernised. Accordingly, there is a need to make predictions in the form of passenger flow forecasts at existing and new civil airports that would give, even to a limited extent, an overview of future scenarios. It is therefore important to stress the significance of airline forecasts as the basis for not only financial planning, but investment and infrastructure planning. For example, the number of passengers in various categories (e.g. arrivals, departures, in transit) determines requirements concerning a terminal's throughput. Passenger traffic is linked to many factors, and the inclusion of the time factor alone is a considerable simplification. It is a well-known fact that air passenger transportation will be influenced by various factors, including the population of the country, future amount of the Gross Domestic Product (GDP), consumption levels, the value and volume of foreign exchanges, etc. To a certain extent, such forecasts enable the right decisions on future activities in the analysed area to be taken. Thanks to the use of suitable forecasting methods, key decisions become more justified and substantiated with an appropriate analysis.

The traditional approach to generating long-term forecasts consists of statistical methods involving time series and econometric models in order to extrapolate observable growth patterns (gravity models, analyses and variants) (Gardner & Mckenzie, 1985; Gardner & Mckenzie, 1988; Gardner & Mckenzie, 1989). Forecasting airline passenger numbers intensity using (regional) air models involves determining the demand for air transport in the region. Another important aspect is the seasonality of air transport. Seasonal variations make it necessary to monitor passengers' intensity, which include: number of passengers (current traffic, traffic incoming from other airports, generated traffic), airfare (if applicable), classification of travel according to origin/destination (for origin-destination and connecting flights), in each month of the year, and on each day of the month. Currently, there are many applications of data mining in the aviation sector as research has been undertaken to forecast passenger flows on domestic or international flights in specific cities and airports (Cho, 2003; Kulendran & Witt, 2003; Chen, 2006; Feldhoff et al., 2012; Kim et al., 2003; Lu et al., 2009; Cuhadar, 2014; Boccaletti et al., 2014; Huang et al., 2015; Zou et al., 2014) (Wang et al., 2014).

Researchers are engaged in a long-term quest to predict airline passenger numbers through analysing past patterns (Van der Maaten & Hinton, 2008). The choice of method used largely depends on the research question and data available (Armstrong & Collopy, 1992). An important aspect to be addressed is the viability of measuring techniques. This Thesis investigated the process of finding practical knowledge from an immense amount of data saved on databases, data warehouses and different information repositories (Fayyad et al., 1996), a process known as "data mining", as an alternative to previously known mechanisms. Such approaches have been used in various application domains, such as in sentiment analysis, object recognition, online advertisement, and social marketing. The capability of data mining for machine learning is using combinations of different techniques from various fields, such as artificial intelligence, statistics, database systems, and pattern recognition (Riga et al., 2009). This ability to data mine can significantly improve the forecasting capabilities of current methods.

A vast amount of literature exists relating to approaches to modelling airline passenger movement, as this has important business implications in real life. Sometimes a whole group of factors is considered in order to model historical passenger movements, and sometimes these movements are modelled on their own. The data structure is complicated, and the historical observation period should be sufficiently long for any data-hungry machine learning technique (Yang & Wu, 2006). A large set of training data is required for this type of method to capture the relationship among factors that could be observed. In the case of those movements that are modelled on their own, the passenger data are modelled by several methods based on different feature engineering, feature extraction, feature derivation, multivariate and univariate time series to capture their movement over time, and from this the expected future movement can be predicted (Bose & Mahapatra, 2001; Opitz & Maclin, 1999).

In this study, the subject of the analysis is Muscat International Airport, with passenger flights being the focus. Four techniques are commonly used for forecast calculations: seasonal exponential smoothing, seasonal ARIMA, artificial neural networks (ANN) using deep learning for the optimisation issues and principal component analysis (PCA). It was decided that PCA would be used for this study.

PCA is an easy, classical multivariate data analysis technique, which is popular within linear feature extraction as well as the data dimension reduction of numerous uses (Bengio, 2013). It has been applied in numerous areas of information processing to prepare data due to its distinctive error reducing and correlating properties. PCA starts with compressing most of the information in the first data space with fewer features, then maximising the variances in a subspace (Timmerman, 2003) The PCA subspace is distributed through the corresponding top eigenvalues of the sample covariance matrix. PCA also can be applied in data preparation for both supervised and unsupervised learning and recognition processes (Turk & Pentland, 1991). Despite its simplicity compared with other techniques such as seasonal exponential smoothing, seasonal ARIMA, and artificial neural networks (ANN), PCA is flexible enough to process a wide range of factors, such as an airline's requirements, traffic flow, land uses, and meteorology. Although other algorithms do help, they often lack one or more functionalities that are fulfilled by PCA. This study tests the ability of several models for predicting the number of international airline passengers in Oman to a better understanding of forecast model selection and combination approaches. The choice of methods requires a detailed study on the research questions, data structure, availability of information, forecasting horizon, etc. The methods chosen for this study have greater flexibility in terms of data size, as PCA can be applied to both large and small datasets. The results are easy to interpret, and the model can be updated as soon as new data points are available. Moreover, it is a non-parametric and direct method of obtaining relevant information from unclear datasets. PCA provides a roadmap for reducing dimensions of a complex dataset and reveal the simplified structures behind it. However, most PCA methods are not able to realise the desired benefits when they handle real-world, nonlinear data. Here is where the challenge lies. Current implementations of PCA use a correlation matrix, which is obtained by the Pearson correlation coefficient. However, in some cases the Pearson correlation coefficient could be limited in the sense that it fails to capture other properties of the data except the linear relation. Therefore, in order to conduct further analysis, a modified version of PCA with kinetic correlation matrix using kinetic energy is proposed.

The research period covers the years 1998 to 2016. Data were presented in monthly cycles. The study consisted of 51,983 observations of numeric variables. Despite the significance of Oman's domestic airline market sector, little to no work has been done

to understand the behaviour of passengers travelling by air transportation in the Oman region. Moreover, there has been no previously reported study that has developed and empirically tested PCA algorithm-based models for forecasting airline passenger demand. The primary objective of this study is to address this apparent research gap in the literature. In order to address the research objective, various forms of mathematical expressions were proposed and tested. The study also sought to examine whether the combined approach based on PCA and ANN approaches are useful tools for this application.

2.2 Research Aims and Objectives

The study aims to develop a novel method, which will introduce a new modification version of PCA with kinetic correlation matrix using kinetic energy and forecast the number of airline passenger as accurately as possible as there are many forecasting methods discussed in the literature. The efficiency of the modified and traditional version of PCA is compared, by applying them to an airline passenger dataset. Again, the choice of competing methods should depend on the structure of the data, accuracy and reliability of the forecast values obtained from those methods, and ease of use (Kao et al. 2013; Jammazi. 2012; He et al. 2012; Pal & Mitra 2009; Goyal & Mehra 2017). A users' preference also plays a vital role while selecting one from a pool of closely competing models.

Numerous studies have applied machine learning and deep learning forecasting models to air passenger forecasting (Mueller & Chatterji 2002; Tu et al. 2008; Zonglei et al. 2008; Xu et al. 2005; Khanmohammadi et al. 2016), however, most published studies concentrate on three specific regions, USA, Europe and Asia Pacific. To our knowledge, no such studies have been conducted using Oman airport data. All previous studies used only the point forecasts while comparing the forecasting performance of methods. While choosing a forecasting model for future use it is very important to consider the confidence bands around the point forecasts value to evaluate the performance of the selected model on the unforeseen data (Gao et al. 2013; 2016; Deising et al. 2015). This study will reduce this gap in the literature by considering the prediction interval as well as point forecasts while propose forecasting models for Muscat airport.

It is understood that this study must consider a larger subset of forecasting models available in the literature due to the data available in hand is big and consists of several observations of airline passenger's data. However, a broader model class can be considered when more features are available. Armstrong and Collopy (1992), discussed that a good forecasting model ideally has less prediction error and quantifies the risk associated with the forecasted value in terms of prediction intervals. Other modelling approach other than feature engineering, time series, and univariate can be used if some variables related to the airline passenger data can be measured and available to model. That may range from using multiple regression models to advanced machine learning algorithms.

The airport authority may interested to know the forecast for airline passenger for some frequencies other than monthly, for example daily, weekly etc. The current models can be updated for data measured with various frequencies or some advanced model can be applied as the data length increases. Along with airline passenger movement, the authorities may be interested to know the future behavior of some other important variable, for example, predicting flight delays, as it is a tremendous economy cost and dissatisfaction can be brought to airline passengers.

Proper forecasts for air traffic passengers are of great matter for Oman as tourism is considered as one of the most profitable industries (UNWTO, 2016). Initial this study keeping in mind the great need of forecasting models for the airport of Oman and finding the gap in the literature about not using prediction interval around point forecasts. This report will help Muscat airport to build their inaugural forecasting models to track the future trend of air travellers and make intelligent and wise business decisions. It should be kept in mind that forecasting tasks can vary in many dimensions, the length of the forecast horizon, the size of the test set, forecast error measures, the frequency of data, etc.

It is unlikely that once selected, a forecasting method will be better than all other plausible methods all the time. Generally, the sensible likelihood coming from the forecast model should be frequently analyzed depending on the current task and when a new data set is available.

Hence, a summary of aims and objectives of the study are:

1. Review the literature to identify the potential forecasting models that can be applied on Airplanes data sets;
2. Propose forecasting models for the airline passenger numbers travelling through Muscat airport of Oman;
3. Quantify the uncertainty around the forecasted value in terms of the prediction interval.

2.3 Research Hypotheses

Given the above discussion, the research proposed and presented in this Thesis could immensely benefit the air transportation community in the future by providing the means to use the modified version of principal component analysis in forecasting airline passenger numbers, especially when coping with real-world nonlinear data.

Motivated by the above reasoning, the proposed research will focus on answering the following research questions:

1. Are current data mining techniques useful tools for measuring airline passenger numbers?
2. Is the modified version of PCA operative in data dimension reduction, class reparability and classification accuracy than traditional PCA?
3. Does a combined approach based on PCA and ANN enable better prediction in forecasting air passenger numbers?
4. Can machine learning and the modified version of PCA be effectively used, replacing existing statistical and conceptual analysis approaches, in forecasting air passenger numbers?

2.4 Research Contributions

The research conducted within the context of this Thesis has met all of the above objectives and has led to the following original contributions:

2.4.1 Conceptual Contribution

Previous research with respect to the efficiency of the principal component analysis multivariate technique has not been sufficiently detailed and rigorous enough in the field of air passenger forecasting. PCA has been chosen due to its flexibility in terms of data size, as PCA can be applied to both large and small datasets (Turk & Pentland, 1991). This lack of detailed investigation leads one to a number of open research

questions. Chapter 5 of this Thesis defines a novel method that concentrates largely on the research topic proposed, which involves feature engineering using PCA and its implementation to deep neural networks.

2.4.2 Technical Contributions

There are two main technical contributions to the state of the art from this Thesis:

a) A machine learning approach based on time series models, different feature engineering, feature extraction, and feature derivation is proposed to improve air passenger forecasting. Correspondingly a modified version of principal component analysis (PCA) using a different correlation matrix is proposed – obtained by a different correlation coefficient based on kinetic energy to derive new features (when carrying out this study). Chapter 5 proposes the use of a modified version of PCA, which is a statistical approach with kinetic correlation matrix using kinetic energy, to forecast the number of airline passengers as accurately as possible. The novel approaches of MVPCA supported by linear and non-linear regression in machine learning proposed in this chapter not only benefit the research community within the airline sector, but beyond, specifically those who carry out medical, material inspection and environmental monitoring. This contribution has resulted in the following conference paper.

S. Alruzaiqi, C.W. Dawson, (2018) Modification Version of Principal Component Analysis with Kinetic Correlation Matrix using Kinetic Energy, Future of Information and Communication Conference (FICC), Singapore, 5th -6th April 2018, and appeared in Volume 886 of the Advances in Intelligent Systems and Computing series in Springer. (Appendix 1).

b) Predicting airline passenger numbers is one of the main concerns of airline services. This Thesis presents a possible combined approach between PCA and artificial neural networks (ANN) for forecasting airline passenger numbers. Due to the small size of the available dataset, PCA is used to reduce the dimensions and the required principal

components (PCs) are selected based on the given rules. These PCs are then treated as inputs to an ANN to make forecasts of the airline passenger numbers. Data from the Oman Management Airport Company (OMAC) are used to compare the results of the proposed model with that of traditional regression models. To the best of our knowledge, there is no existing work that utilises artificial neural networks combined with PCA to predict airline passenger numbers in the Oman Region. Therefore, this study will discuss the concept of using ANN to examine several external features that enable better prediction and demand forecasting based on such approaches. These approaches are not only tedious but will also not be able to identify the presence of fine-detail discriminative features between data captured from different groups. In Chapter 4, an experiment is carried out to test the performance of neurons in the hidden layer of an ANN to solve the nonlinear optimisation problem. This approach is work-intensive; therefore, a method is also introduced for improving the prediction performance of metrics by ensemble method predictions made on different engineered feature spaces, replacing outliers that were not predicted correctly. The research outcomes prove the capability of machine learning algorithms using the combined approach based on PCA and ANN to carry out such discriminate tasks with a very high degree of accuracy. This contribution has resulted in the following conference paper.

S. Alruzaiqi, C.W. Dawson, (2019) Optimizing Deep Learning Model for Neural Network Topology, Computing Conference, London, 16th - 17th July 2019, and appeared in Volume 997 of the Advances in Intelligent Systems and Computing series in Springer. (Appendix 2).

2.4.3 Comprehensive Literature Investigation

Chapter 6 provides an insight into how the outcomes of the research conducted within the context of this Thesis were effectively used to improve the performance of the principal component analysis algorithm, enabling the algorithms to be re-designed using different feature engineering methods, showing improvements in the accuracy of the tasks carried out. This work proves the usefulness of the novel research work

carried out for this Thesis and its potential contributions to airline research. The research conducted within the remit of this Thesis has resulted in the following secondary contribution:

- A comprehensive review of the literature to investigate existing work on principal component analysis applied to forecasting airline passenger numbers (Chapter 2).

2.4.4 Data Collection

It was noted that a dataset of similar nature in which the forecast of the airline passenger numbers, based on feature engineering, and principal component analysis, has not been conducted prior to this research and hence no public database is available to support the original research presented in this Thesis. The recent dataset has thirteen years' worth of monthly data, eight years' worth of monthly tourism data, and more than twenty years' worth of other data. The main distinction of this study to the studies undertaken to forecast airline passenger for other airports is that this study has a good amount of airline passenger data – with each set such as population size represented in a different time frame, i.e. 1980-2015 – yet want to identify the future trend of airline passenger movement and to develop predictive models to forecast the number of airline passengers for Muscat airport situated in Oman. Therefore, the outcome from this research will have a profound impact on the knowledge by reducing the dimensionality of data space for airline forecasting field. Hence it was essential to carry out the tasks relevant to capturing this novel dataset. The primary impact of this work can be grouped into:

a. Analysing different feature extraction and selection strategies, as these are highly effective methods of feature extraction involving a mathematical process which transforms a selection of (possibly) correlated variables right into a (smaller) selection of uncorrelated variables known as principal components.

b. Implementing modified version of PCA: The dataset collected during this research will be made publicly available and this will be itself contribute to the future research committee (Chapter 5).

2.5 Thesis Overview

The remainder of this Thesis is organised as follows:

Chapter 2 provides a review of relevant studies that used the time series method of prediction; that explained the components that could be present in airline passenger data; and, since we are mostly interested in forecasting airline passenger numbers, a description on some of the widely used forecasting methods. Furthermore, a review of time series, feature engineering, and deep learning studies are discussed in this chapter.

Chapter 3 introduces the research background and covers details of the forecasting models used in the field of aviation and the theoretical background of various statistical and machine learning algorithms used in this Thesis for data analysis. Multiple evaluation metrics are used to measure the forecast accuracy. It provides details of the analysis design, the processes adopted in carrying out the subjective experiments, and data preparation for the analysis to be conducted in the contributory chapters that follow.

Chapter 4 proposes the use of machine learning algorithms based on time series models, different feature engineering, feature extraction, and feature derivation to improve air passenger forecasting. An experiment was undertaken with artificial neural networks to test the performance of neurons in the hidden layer to optimise the dimensions of all layers and to obtain an optimal choice of connection weights – thus the nonlinear optimisation problem could be solved directly. A method of tuning deep learning models using H2O is also proposed.

Chapter 5 proposes the novel use of PCA, involving the use of a modified version named Modified Version of Principle Component Analysis, for the prediction analysis when conducting the tasks assigned. Evaluation performance is also proposed.

Finally, Chapter 6 concludes with an insight into future work, including details of a separate project conducted to prove the concepts that are the outcomes of the research presented in this Thesis.

Chapter 2: Literature Review

3.1 Introduction

The aim of this chapter is to provide an extensive review of the research areas that are relevant to the work undertaken in this Thesis. To this end, the chapter presents previous research on passenger flow forecast using a number of methodologies and techniques with varying degrees of success. The methodologies used in this work include both "top-down" and "bottom-up" approaches. The top-down approach is based on a single aggregated forecast which is forecasting the total passenger number of a country. This method can be applied to individual airports using their historical passenger data. On the other hand, the bottom-up approach can be applied to individual airports to obtain an aggregate forecast. Both top-down or bottom-up methods are widely used to forecast passenger flows nowadays. The typical applications, attributes and limitations of these methods are discussed in this chapter.

3.2 Traditional Time Series Forecasting

Numerous studies have been attracted by the rapidly grown global air traffic. The annual growth of global air passenger traffic increased by 5.4% between 1970 and 2012 (IBRD-IDA, 2012). In the past four decades, many studies on air traffic demand have been performed, providing abundant literature dealing with determinants of air traffic demand. Prediction of future traffic demand based on different forecasting techniques has become an important factor in *airport development* planning (Suryan et al., 2016).

In order to determine the feasibility of an airport, it is necessary to forecast its demand over its design life (Hyndman & Athanasopoulos, 2013). Air traffic forecasting has a long history because of the importance of how future traffic demand has a direct effect on the airline industry (Profillidis, 2000). There are many methods available for improving the forecast result, two of which have been studied in the research works presented in this Thesis, applying and improving conventional time series methods and obtaining information from the dataset. A statistical model based on time series is a state-of-art method for predicting demand. It can be summarised in three approaches – the univariate time series smoothing and forecasting approach; the multivariate

approach based on Principal Component Analysis (PCA); and approaches based on econometric modelling (Box & Jenkins, 1976).

One advantage of a univariate method is that it is simple to explain and straightforward to apply compared with other data-greedy machine learning algorithms. Application to small dataset and ease of use in real-life scenarios are other important reasons that univariate time series forecasting methods are used (Tsui et al., 2011).

Forecasting using traditional time series is a well-researched area (Taneja, 1987; Nam & Schaefer, 1995; Weatherford et al., 2003; Hyndman & Koehler, 2006). Time series forecasting tries to find patterns and regularities which are used for applying future values from a sequence of data points. It has different degrees of flexibility and complexity so that many ways to generate forecasts are available and it is possible to come up with more than one forecasting result for the same problem. Therefore, it will come to a question of whether or not all or some of the individual forecasts can be used in combination for obtaining a superior forecast. The general forecasting methods and combinations thereof will be presented in Sections Two, Three and Four of this chapter, while Section Three also presents a popular and easy-to-use empirical evaluation approach.

This section continues with describing the selection method of popular time series forecasting techniques (Makridakis et al., 1998; Box et al., 2015). The background for this method will be introduced in the later sections of this Thesis.

3.3 Simple Forecasting Methods

Some forecasting methods are very simple yet provide accurate forecasts. Usually, they are used as benchmarks for more complicated models. If a complex model does not provide better forecasts than these simple models, then it is not worth considering. These methods are discussed in the following sections of this chapter.

3.3.1 Average Method

Setting the forecast to the average level of the observed data can produce the best prediction. If $y_1, y_2, \ldots, y_T$ is the observed series for $t = 1, \ldots, T$, then the forecast is set to the average value:

$$\hat{y}_{T+h|T} = \bar{y} = \frac{y_1 + y_2 + \ldots + y_T}{T} \quad 3.1$$

where, $h = 1, \ldots, H$ is the forecasting period (Cleman, 1989).

3.3.2 Naïve Method

In this method, all future values are set to the last observed values.

$$\hat{y}_{T+h|T} = y_T, \qquad \text{for all } h \quad 3.2$$

This method as stated by (Winkler & Cleman, 1992) often provides accurate forecasts for different economic and financial time series datasets in practice.

3.3.3 Seasonal Naïve Method

For seasonal data, the forecast is set to the same value observed in the same period of the previous year, for example, using the same quarter of the previous year in the case of quarterly data.

The forecast equation can be written as:

$$y_{T+h-km} \qquad where\, m = \text{seasonal period}, \qquad k = \lfloor \frac{(h-1)}{m} \rfloor + 1, \quad 3.3$$

and $\lfloor u \rfloor$ denotes the integer part of u. For monthly data, the forecast values for all Julys in the future are the same as the value observed in last July (see Makridakis, 1993)

3.3.4 Drift Method

The forecasts from the naïve method are controlled by allowing the forecasts to move upward or downward gradually.

The number of variations over a period of time, also known as drift, is equal to the average variation observed in the previous data. The forecasts can be calculated using:

$$y_T + \frac{h}{T-1} \sum_{t=2} T(y_t - y_{t-1}) = y_T + h(\frac{y_T - y_1}{T-1}) \quad 3.4$$

Even though these methods are very simple, they can be very effective for small datasets and short forecast horizons. These methods also serve as benchmarks for more sophisticated models (Clemen & Winkler, 1986;Graham, 1996; Zarnowitz, 1984).

3.4 Exponential Smoothing

Proposed in the late 1950s, exponential smoothing became one of the most successful forecasting methods for univariate time series. In this method, forecasts are weighted averages of historical observations, in which weights decrease exponentially as they get older. Consequently, the more recent observations attract higher weights (Holt, 1957). It is necessary to identify the intrinsic seasonal structure in the airline traffic data if one is interested in monthly or quarterly forecasts. This method generates reliable forecasts quickly and is quite useful if the data to hand is small. The exponential smoothing approach includes various types of forecasting models that can be applied to time series with various characteristics (additive or multiplicative manner). Using the statistical framework of these models, point forecasts as well as prediction intervals can be obtained. The prediction interval is a useful piece of information in planning services needed to provide for the expected passengers' travel through an airport (Gardner & Mckenzie, 1985; Gardner & Mckenzie, 1988; Gardner & Mckenzie, 1989). Various models under the exponential smoothing family are described in the following subsections.

3.4.1 Simple Exponential Smoothing

Simple exponential smoothing (SES) is the most elementary form of the exponential smoothing modelling approach. It is also known as single exponential smoothing in some literature. This model is applicable if the time series has a linear evolution (Hyndman & Athanasopoulos, 2013). In this case, the forecasts can be calculated through the weighted average of the historical data. The equation of a one-step-ahead forecast from this model is:

$$\hat{y}_{t+1|t} = \alpha y_t + \alpha(1-\alpha)y_{t-1} + \alpha(1-\alpha)^2 y_{t-2} + \ldots, \qquad 3.5$$

where $0 \le \alpha \le 1$ is the smoothing parameter. The parameter α regulates the rate of the weights' diminution. Hence, the one-step-ahead forecast for time $t+1$ shows a weighted average, considering the findings in the series $y_1, y_2, \ldots, y_t$.

Equation 2.5 can also be rewritten in component form. Having only one component, level (l_t) the forecast equation and the smoothing equation takes the form:

Forecast equation $\hat{y}_{t+1|t} = l_t$ **3.6**

Smoothing equation $l_t = \alpha y_t + (1-\alpha) l_{t-1}$, **3.7**

where l_t is the level (or the smoothed value) at time t. The forecast equation gives the value at time $t+1$ as nil but the calculated level at time t. The smoothing equation applies the smoothing factor on all previous historical observations and provides an approximate calculation of the level of the series at each time period t (McNees, 1992; Armstrong, 2001). Through this exponential process, it is possible to have a 'flat' forecast function (Hyndman & Athanasopoulos, 2013), and according to the period considered the model becomes:

$$\hat{y}_{t+h|t} = \hat{y}_{t+1|t}; \quad h = 2,3,\ldots \qquad \textbf{3.8}$$

3.4.2 Holt's Linear Trend Method

Holt extended simple exponential smoothing to apply to time series with trend (Holt, 1957). This method has two smoothing parameters and is also referred to as double exponential smoothing. This method requires a forecast equation, one smoothing equation for each level examined, and a second smoothing equation for the trend:

Forecast equation $\hat{y}_{t+h|t} = l_t + hb_t$ **3.9**

Level equation $l_t = \alpha y_t + (1-\alpha)(l_{t-1} + b_{t-1})$ **3.10**

Trend equation $b_t = \beta(l_t - l_{t-1}) + (1-\beta) b_{t-1}$ **3.11**

In this case:

- l_t indicates an approximate calculation of the level of the series at time t;

- b_t indicates an approximate calculation of the trend (i.e., slope) of the series at time t;
- α ($0 \le \alpha \le 1$) represents the smoothing parameter for the level; and
- β ($0 \le \beta \le 1$) indicates the smoothing parameter for the trend.

The forecast function is trending rather than being flat. It is growing linearly with h, when the h-step-ahead-forecast equals the last estimated value plus h times the last approximate trend value.

(Önder & Kuzu, 2014) used Holt's linear exponential smoothing method to forecast air traffic passenger numbers for 15 airports in Turkey for the years 2013 to 2023 using ten years' worth of historical data from 2002 to 2012. They found that Holt's linear exponential smoothing technique was a successful predicting method regarding to the monthly time series data, however, this study deals with the point forecast only and did not incorporate the risk associated with the forecasted value.

3.4.3 Exponential Trend Method

In Holt's linear trend model, the trend in the forecast function is linear and the estimated trend is added to the estimated level. A variation of this method is to multiply the estimated slope with the estimated level in order to have a linear growth rather than a constant slope, making the forecast function exponential. The method takes the form:

Forecast equation $$\hat{y}_{t+h|t} = l_t + b_t^h \quad \textbf{3.12}$$

Level equation $$l_t = \alpha y_t + (1-\alpha)(l_{t-1} b_{t-1}) \quad \textbf{3.13}$$

Trend equation $$b_t = \beta \frac{(l_t}{l_{t-1}} + (1-\beta) b_{t-1} \quad \textbf{3.14}$$

where b_t represents an estimated relative growth rate (Holt, 1957).

3.4.4 Damped Trend Methods

Both Holt's linear method and the exponential trend method tend to over-forecast, especially when the forecast horizon is large. This is because Holt's linear method has a constant trend (while the increase or the decrease is indefinite) and the exponential

trend method has an exponential growth rate, which can grow or decline indefinitely. To overcome this shortcoming of these useful methods, the additive damped trend method, which adds a new 'damping' parameter to Holt's linear method, has been proposed (Gardner & Mckenzie, 1985; Gardner & Mckenzie, 1988; Gardner & Mckenzie, 1989). The dampen parameter makes the trend flatter after a time when the forecast horizon is too long.

3.4.5 Additive Damped Trend

If the damping parameter is ϕ ($0 \le \phi \le 1$), the additive damped trend method has following specification as stated by (Makridakis & Hibon, 2000):

Forecast equation $$\hat{y}_{t+h|t} = l_t + (\varphi + \varphi^2 + \ldots + \varphi^h) b_t \quad \textbf{3.15}$$

Level equation $$l_t = \alpha y_t + (1-\alpha)(l_{t-1} + \varphi b_{t-1}) \quad \textbf{3.16}$$

Trend equation $$b_t = \beta(l_t - l_{t-1}) + (1-\beta)\varphi b_{t-1} \quad \textbf{3.17}$$

The additive damped trend model is identical to Holt's linear model if $\phi = 1$. For values between 0 and 1, ϕ dampens the trend so that it settles to a constant value sometime in the future. The forecasts converge to $l_t + \phi b_t/(1-\phi)$ as $h \to Inf$ for any value of $0 \le \phi \le 1$. The addition of a damped trend has been demonstrated advantageous if forecasts are required automatically for various series (Hyndman & Athanasopoulos, 2013).

3.4.6 Multiplicative Damped Trend

Taylor proposed a multiplicative damped trend model by adding a damping parameter to the exponential trend method (Taylor, 2003). This is driven by the forecast efficiency observed in the additive trend case. Moreover, if compared with Holt's linear method, the multiplicative damped trend method produces fewer conservative forecasts than the additive damped one. The model takes the form:

Forecast equation $$\hat{y}_{t+h|t} = l_t b_t^{(\varphi + \varphi^2 + \ldots + \varphi^h)} \quad \textbf{3.18}$$

Level equation $l_t = \alpha y_t + (1-\alpha) l_{t-1} b_{t-1}^{\varphi}$ **3.19**

Trend equation $b_t = \beta \frac{l_t}{l_{t-1}} + (1-\beta) b_{t-1}^{\varphi}$ **3.20**

3.4.7 Holt-Winters Seasonal Method

In Holt's linear model, only the trend or the trend-cycle component is modelled (Holt, 1957). It is extended to include seasonality, consisting of the forecast equation along with three more equations, and specifically: one indicating the level (l_t), the second indicating the trend (b_t) and the last one indicating the seasonal component (s_t). The corresponding smoothing parameters are α, β and γ, while m is used to indicate information about the seasonal period per year, for example, $m = 4$ for quarterly data, and $m = 12$ for monthly data. Like the exponential trend method, there are two variations of the Holt-Winters seasonal method: the additive method and the multiplicative method. The choice of the two variations depends on the nature of the seasonal component present in the dataset.

3.4.8 Holt-Winters Additive Seasonal Method

In the additive method:

- The scale of the analysed series shows the seasonal component in absolute terms;
- The series are set by subtracting the seasonal component in the level equation;
- The seasonal component will add up to nearly zero for each year.

Forecast equation $\hat{y}_{t+h|t} = l_t + hb_t + s_{t-m+h^+_m}$ **3.21**

Level equation $l_t = \alpha(y_t - s_{t-m}) + (1-\alpha)(l_{t-1} + b_{t-1})$ **3.22**

Trend equation $b_t = \beta(l_t - l_{t-1}) + (1-\beta) b_{t-1}$ **3.23**

Seasonal equation $s_t = \gamma(y_t - l_{t-1} - b_{t-1}) + (1-\gamma) s_{t-m}$, **3.24**

where $h_m^+ = \lfloor (h-1) \mod m \rfloor + 1$, which takes into account that the final year of the sample feeds the estimates of the seasonal indices needed to forecast. The notation $\lfloor u \rfloor$ is the largest integer number not greater than u. In the level equation, it is shown a weighted average coupling the non-seasonal forecast $(l_{t-1} + b_{t-1})$ and observations $(y_t - s_{t-m})$, which are the seasonally adjusted for time t. The trend equation is as same as Holt's linear model. The seasonal equation shows the weighted average coupling the current seasonal index of the season $(y_t - l_{t-1} - b_{t-1})$ and the index of the same season of the previous year (i.e., m time periods before). Therefore, if the seasonal variations are nearly constant within the series, the additive method is preferred (Clemen, 1989).

3.4.9 Holt-Winters Multiplicative Seasonal Method

The multiplicative version is suitable when the variations change seasonally with the level of the series. As opposed to the additive version, in multiplicative method:

- The seasonal component is given in percentage.
- The series is changed seasonally by dividing the seasonal component. The seasonal component will add up to nearly m per year.

The multiplicative method is represented in the following form:

Forecast equation $$\hat{y}_{t+h|t} = (l_t + hb_t)s_{t-m+h_m^+} \quad \textbf{3.25}$$

Level equation $$l_t = \alpha \frac{y_t}{s_{t-m}} + (1-\alpha)(l_{t-1} + b_{t-1}) \quad \textbf{3.26}$$

Trend equation $$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1} \quad \textbf{3.27}$$

Seasonal equation $$s_t = \gamma \frac{y_t}{(l_{t-1} + b_{t-1})} + (1-\gamma)s_{t-m}, \quad \textbf{3.28}$$

See (Holt, 1957).

3.4.10 Holt-Winters Damped Method

For many seasonal time series, the Holt-winter method with a damped trend and

Forecast equation $$\hat{y}_{t+h|t} = [l_t + ((\varphi + \varphi^2 + \ldots + \varphi^h))b_t]s_{t-m+h^+_m} \quad 3.29$$

Level equation $$l_t = \alpha \frac{y_t}{s_{t-m}} + (1-\alpha)(l_{t-1} + \varphi b_{t-1}) \quad 3.30$$

Trend equation $$b_t = \beta(l_t - l_{t-1}) + (1-\beta)\varphi b_{t-1} \quad 3.31$$

Seasonal equation $$s_t = \gamma \frac{y_t}{(l_{t-1} + \varphi b_{t-1})} + (1-\gamma)s_{t-m}, \quad 3.32$$

multiplicative seasonality performs better than any other competing forecasting methods (Holt, 1957).

3.5 Regression

In the regression approach, a forecast or dependent variable is expressed as one or more outcome related independent or explanatory variables. Formula (2.33) expresses a simple linear regression on a single variable, where *a* is the intercept, *b* the slope of the line and *ε* the error, which is caused from the actual observation on the deviation of the linear relationship.

$$y = a + bx + \varepsilon \quad 3.33$$

In Eq.(2.33), *x* represents the time index. The regression parameters can be estimated by using the standard least squares approach (Clemen & Winkler, 1986; Graham, 1996; Zarnowitz, 1984).

3.5.1 Decomposition and Theta-Model

The aim of decomposition is to separately project the isolated components of a time series into the future data, then produce a final forecast by recombining them. Traditionally, the components are:

- a cycle with a trend of long-term changes;

- like months or holiday times seasonality with a trend of shorter-term but constant length changes; and
- random or irregular error components.

The Theta-model has been proposed recently (Assimakopoulos & Nikolopoulos, 2000). In this model, the seasonally adjusted series are decomposed into long- and short-term components and the curvature of the time series is modified using a coefficient of θ (shown in Formula (2.34)) to the time series' second-order differences.

$$y''_{new}(\theta) = \theta * y''_{original} \quad \textbf{3.34}$$

The value of Theta is bigger than the one being dilated, and it amplifies its short-term behaviour while theta values between zero and one have the opposite effect (Croushore, 1993).

3.5.2 Autoregressive Integrated Moving Average Model

One of the most widely used univariate time series forecasting models is the Autoregressive Integrated Moving Average (ARIMA) process (Box et al., 2015). The ARIMA model is a combination of autoregressive, moving average and difference parameters.

In an observed series, autoregressive parameters control the effect of lagged observations, whereas moving average parameters control the effect of past innovations. The order of difference parameter indicates the needed amount of differences to render a series stationary. The structure of the model also varies, depending on the seasonal pattern of a process. A time series with trend and seasonal pattern can be hard to forecast when these components are present, as the trend and seasonality will affect the value of the times series at different time points. Sometimes the trend and seasonal components are removed or made stationary before fitting any forecasted model. In a stationary time series, the properties do not vary with the times at which the series is observed. Usually, there are no predictable durable patterns in a stationary time series. The time plot of a stationary series shows the results to be horizontal with constant variance. In the literature, several statistical tests exist to check the stationary of a time series. Detailed discussion on such topics is out of the scope of this report.

A time series can be rendered stationary by calculating the differences between two observations in sequence (McNees, 1992; Armstrong, 2001). This is known as difference and can be written as:

$$y'_t = y_t - y_{t-1} \qquad \textbf{3.35}$$

Differencing is used to make the mean of a time series stable by eliminating changes in the time series level, thereby removing the trend and seasonality too. Sometimes the observed series may not be stationary after taking the first difference, and a second-order difference may require making the data a stationary series (McNees, 1992; Armstrong, 2001):

$$y''_t = y'_t - y'_{t-1} \qquad \textbf{3.36}$$

$$= (y_t - y_{t-1}) - (y_{t-1} - y_{t-2}) \qquad \textbf{3.37}$$

$$= y_t - 2y_{t-1} + y_{t-2} \qquad \textbf{3.38}$$

Through calculating the difference between an observation and the same observation in the year before we get the seasonal difference (McNees, 1992; Armstrong, 2001).

$$y'_t = y_t - y_{t-m}, \quad \text{where} \quad m = \text{number of seasons.} \qquad \textbf{3.39}$$

To distinguish seasonal differences from ordinary differences, the latter is referred to as the "first difference" or "differences at lag 1". Therefore, it is necessary, sometimes, to apply both first difference and seasonal difference to obtain stationary data. The backward shift operator B is a very helpful notation when dealing with time series lags (McNees, 1992; Armstrong, 2001):

$$By = y_{t-1} \qquad \textbf{3.40}$$

In the literature, the "lag" operator L is also used to replace the "Backshift" B operator. So, if B operates on y_t it shifts the data back one period. Hence, two

applications of B to y_t shifts the data back two periods (McNees, 1992; Armstrong, 2001):

$$B(By_t) = B^2 y_t = y_{t-2} \qquad \textbf{3.41}$$

and application to the monthly data indicates:

$$B^{12} y = y_{t-12} \qquad \textbf{3.42}$$

The operator B is useful to describe the *difference* procedure, which is at the core of the ARIMA structure. When using the backward operator, the differences can be presented as below (Goodrich, 1984; Goodrich, 1986; McNees, 1992; Goodrich, 2000; Goodrich, 2001):

First order difference: $y'_t = y_t - y_{t-1} = 1 - By_t = (1-B)y_t$ **3.43**

Second order difference: $y''_t = y_t - 2y_{t-1} + y_{t-2} = (1-2B+B^2) = (1-B)^2 y_t$ **3.44**

Monthly seasonal difference: $y_t - y_{t-m} = (1-B^m)y_t$ **3.45**

An observed time series requires differencing until it becomes white noise, i.e. uncorrelated in time and distribution, with a mean equal to zero and variance constant. When the differences series is white noise, the random walk model can be adapted to the main series, which can be presented as below (Goodrich, 1984; Goodrich, 1986; McNees, 1992; Goodrich, 2000; Goodrich, 2001):

$$y_t - y_{t-1} = e_t \quad \text{or} \quad y_t = y_{t-1} + e_t \qquad \textbf{3.46}$$

The error e_t is usually specified as white noise. Another version of the random walk model allows the differences to have a non-zero mean and is written as:

$$y_t - y_{t-1} = c + e_t \quad \text{or} \quad y_t = c + y_{t-1} + e_t, \qquad \textbf{3.47}$$

where c is the average of the changes between consecutive observations (Goodrich, 1984; Goodrich, 1986; McNees, 1992; Goodrich, 2000; Goodrich, 2001).

After obtaining a stationary time series using appropriate difference and stabilising variance by taking log transformations, the ARIMA model can be fitted to the differences data. The ARIMA process is a combination of two basic linear models: the autoregressive (AR) process and the moving average (MA) process. One aspect of these models is that they offer an acceptable approximation of the data generating process (DGP) in a world of Gaussian distributions. In terms of forecasting, these models can provide accurate forecasts as long as the DGP is Gaussian and non-linear features are not strong. Even for some datasets with known non-linear dynamics, this modelling process generates close forecasts since the non-linearity can be not constant with time and is not so relevant to provide prediction improvements.

3.5.3 Autoregressive Models (AR)

Autoregressive approaches have proved to be one of the more accurate time series models, since it can be tested and fully estimated within the framework of least squares regression. The term *auto regression* denotes that it is a regression of the variable regress on itself. An autoregressive model of order p, is known as AR (p) (Hyndman & Athanasopoulos, 2013) and can be represented by the formula (2.48):

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + e_t, \qquad \textbf{3.48}$$

with c being a constant and e_t is the white noise. This appears as a multiple regression, except for its *lagged* values of y_t as predictors. Autoregressive models are very flexible at handling a variety of different time series patterns. Generally, we restrict AR models to stationary data and some limitations on the parameter values are needed.

For example:

- For an AR (1) model, $y_t = c + \phi_1 y_{t-1} + e_t$
    - The restriction is $-1 < \phi < 1$.

- For an AR (2) model, $y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + e_t$,
    - The restrictions are $-1 < \phi_2 < 1, \quad \phi_1 + \phi_2 < 1, \text{and} \phi_2 - \phi_1 < 1$.

More complicated constraints are imposed when $p \geq 3$ and care should be taken during the estimation process.

Stationary autoregressive processes have two characteristics:

- Autocorrelation function, $\text{ACF}(j) = \rho_j = \text{corr}(y_t, y_{t-j})$.
- Partial autocorrelation function, $\text{PACF}(j) = \text{corr}(y_t, t_{t-j} \mid y_{t-1}, \ldots, y_{t-j+1})$.

The existence of an AR (p) process in an observed time series is checked using ACF and PACF. It is believed that an AR (p) process exists in a time series if sample $\text{AR}(j)$ converges to zero as $j \to \infty$ at a geometric rate and sample $\text{PACF}(j)$ equals zero for values greater than p (Box et al., 2015; Makridakis et al., 1998).

For an AR (p) model the one-step-ahead point forecast equation is obtained, replacing the true coefficients by the in-sample estimates, and the error term e_t by 0 (Hyndman & Athanasopoulos, 2013).

$$\hat{y}_{t+1|t} = \hat{c} + \hat{\phi}_1 y_t + \hat{\phi}_2 y_{t-1} + \ldots + \hat{\phi}_p y_{t-p+1} \qquad \textbf{3.49}$$

The subsequent forecast function is the result of the replacement of unrecognised observation of y_t by its forecast (Hyndman & Athanasopoulos, 2013). The two-step-ahead forecast function could be represented as:

$$\hat{y}_{t+2|t} = \hat{c} + \hat{\phi}_1 \hat{y}_{t+1} + \hat{\phi}_2 y_t + \ldots + \hat{\phi}_p y_{t-p} \qquad \textbf{3.50}$$

The h -step-ahead forecasts can be acquired in the same manner.

3.5.4 Moving Average Models (MA)

Instead of past observations, the past forecast errors are used as predictor variables in a moving average model. A moving average model with q parameters, refereed as MA (q) (McNees, 1992; Armstrong, 2001), is represented as:

$$y_t = c + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \ldots + \theta_q e_{t-q} \quad \textbf{3.51}$$

where e_t is white noise. The values of e_t are unknown so 48 is not a regression model as commonly understood. Modifying the parameters $\theta_1, \ldots, \theta_q$ leads to other time sequences/series patterns. In a similar approach to autoregressive models, the difference of the error term e_t will modify the range of the series, but not the pattern. Any stationary AR (p) model can be presented as an infinite MA (∞) model by duplicated substitution of the parameters (Box et al., 2015). The reverse holds if some constraints are put on the MA parameters, i.e., an MA (q) model can be presented as an infinite AR (∞) process. Then MA model is called "invertible".

The invertibility limits are almost the same of the stationarity limits. For example,

- For an MA (1) model: $-1 < \theta < 1$.
- For an MA (2) model: $-1 < \theta_2 < 1, \quad \theta_1 + \theta_2 > -1, \text{and} \qquad \phi_1 - \phi_2 < 1$

Like AR model, the constraint imposed on MA parameters when $q \geq 3$ should be considered during the estimation process. The ACF for an MA process becomes 0 after $j = q$. In contrast the PACF converges to 0 geometrically. Forecasting from an MA requires an estimation of the coefficients θ_j and errors e_t. Since, the error term is unobservant at any time period, it is replaced by the previous periods' forecast-error $\hat{e}_t$. For an MA (q) model (McNees, 1992; Armstrong, 2001), the one-step-ahead forecast function is:

$$\hat{y}_{t+1|t} = \hat{c} + \hat{\theta}_1 \hat{e}_{t-1} + \hat{\theta}_2 \hat{e}_{t-2} + \ldots + \hat{\theta}_q \hat{e}_{t-q}, \quad \textbf{3.52}$$

The two-step-ahead forecast function is:

$$\hat{y}_{t+2|t} = \hat{c} + \hat{\theta}_2 \hat{e}_{t-1} + \hat{\theta}_3 \hat{e}_{t-2} + + \ldots + \hat{\theta}_q \hat{e}_{t-q+1} \quad \textbf{3.53}$$

This implies that forecasts will become trivially close to the mean of the series following some steps. The moving average model should not be confused with *moving*

average smoothing. The latter is widely adopted for estimating the trend-cycle of the past value whereas the former is a forecasting model.

3.5.5 Non-seasonal ARIMA Model

A non-seasonal univariate ARIMA model is generally expressed as ARIMA (p, d, q), with the following specifications:

- p : The number of autoregressive parameters.
- d : The number of non-seasonal differences to make the series stationary.
- q : The number of moving average parameters.

If the series $\{y_t\}, t = 1, \ldots, T$ follows an ARIMA (p , d , q) model (Hyndman & Khandakar, 2008), then it can be expressed as:

$$\phi_p(B)(\nabla^d y_t - \mu) = \theta_q(B) e_t, \quad 3.54$$

where B is the back-shift operator.

$$B y_t = y_{t-1} \quad 3.55$$

$$\nabla^d = (1 - B^d) \quad 3.56$$

is the number of non-seasonal differences with.

$$B^d y_t = y_{t-d} \quad 3.57$$

$$\phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 - \ldots - \phi_p B^p, \quad 3.58$$

$$\theta_q(B) = 1 - \theta_1 B - \theta_2 B^2 - \ldots - \theta_q B^q, \text{ and} \quad 3.59$$

$$e_t \text{ is id N}(0, \sigma_e^2). \quad 3.60$$

3.5.6 Seasonal ARIMA Model

The common structure of a seasonal univariate ARIMA model is $\text{ARIMA}(p,d,q)(P,D,Q)_s$, with the form:

$$\phi_p(B)\Phi_P(B^s)(\nabla^d\nabla_s^D y_t - \mu) = \theta_q(B)\Theta_Q(B^s)e_t, \quad 3.61$$

where s is the period of seasonality within a year (Hyndman & Khandakar, 2008).

$$\phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 - \ldots - \phi_p B^p \quad 3.62$$

$$\Phi_P(B^s) = 1 - \Phi_1 B^s - \Phi_2 B^{2s} - \ldots - \Phi_P B^{Ps} \quad 3.63$$

$$\theta_q(B) = 1 - \theta_1 B - \theta_2 B^2 - \ldots - \theta_q B^q \quad 3.64$$

$$\Theta_Q(B^s) = 1 - \Theta_1 B - \Theta_2 B^{2s} - \ldots - \Theta_Q B^{Qs} \quad 3.65$$

$$\nabla_s^D y_t = y_t - y_{t-D_s} \quad 3.66$$

and D is the number of seasonal differences (see (Hyndman & Khandakar, 2008).

Applying an ARIMA model to an observed series involves two steps: identification and estimation. In the identification stage, the value of the autoregressive parameters (p and P) and moving average orders (q and Q) are specified, along with the difference orders (d and D). After the orders of different parameters have been obtained, the coefficient values are estimated at the estimation stage.

When the ARIMA model was first introduced in the 1970s, it was assumed that the selection of an ARIMA model from the class of plausible models was 'more art than science' and should only be done by experienced professionals. However, the growth of economics, the development of business, and the advancement of computer technology have increased the use of time series models.

In some situations, tens of thousands of observed series need to be analysed at the same time, or frequent updating to the forecasting function is required to get an overall picture, and a detailed analysis of each series is not possible due to cost and time constraints.

Prominent among methods that provide automatic algorithms are *TSE-AX* (Mélard & Pasteels, 2000), AUTOBOX (Automatic Forecasting Systems, Inc., 1999), Hannan-Rissanen automatic algorithm (Hannan & Rissanen, 1982), Autoregressive Moving Average (ARARMA) (Parzen, 1982), FOREX and Forecast pro (Goodrich, 1984; Goodrich, 1986; Goodrich, 2000; Goodrich, 2001), *SCA-Expert* (Liu, 1989), *TRAMO* and *SEATS* (Gómez & Maravall, 1996; Gómez & Maravall, 2000), and R (Hyndman & Khandakar, 2008).

However, even though an automated algorithm provides model selection, it is important to understand the behaviour of the models rather than solely rely on the automated process.

3.5.7 Nonlinear Forecasting

Principal Component Analysis and Artificial Neural Networks models are widely used nonlinear methods for forecasting, which combine two or more sets of model coefficients into one and determine which is used for forecasting based on the regime or state the system is likely to be in. With the advantages of freely choosing a model for each problem, the artificial neural networks can be applied to data-driven forecasting.

In time series forecasting, the time indices and time lagged observations can be applied as input variables to obtain the forecast output. So far, the neural networks have been widely used for forecasting purposes with success, a summary of related work in this field can be found in the study of (Zhang et al., 1998).

3.6 Air Travel Demand Modelling and Forecasting

A number of studies have applied time series smoothing and univariate forecasting models to air passenger forecasting (Adrangi et al., 2001; Jonga et al., 2004; Hui et al., 2004; Matsumoto, 2004; Matthiessen, 2004; Mason, 2005; Lee, 2009; Önder & Hasgül, 2009; Hwang & Shiao, 2011; Sengupta et al., 2011). However, most reported studies concentrate on three specific areas: the USA, Europe and Asia Pacific. To date, no such studies have been undertaken using Oman airports data.

Autoregressive Integrated Moving Average (ARIMA) models are most widely used to forecast air passenger numbers. In fact, (Box & Jenkins, 1976) applied the ARIMA model to an airline passenger dataset when the ARIMA model was first introduced in the literature. Since then, ARIMA models and other univariate models are used to forecast airline passenger data as they can capture the complex seasonality, cyclical and trend structure in a single model. As pointed out by (Zhang, 2003), the popularity of ARIMA models lies in the fact that they are based on minimal assumptions.

ARIMA models are used in many studies for forecasting air passenger data. In one study case in Indonesia, (Faisal, 2002) used time series analysis ARIMA for international air traffic flow. He found that the growth rate for the international passenger flow was on average 7%. However, the influence of seasonal factors for international cargo and international aircraft movement using the decomposition method was not clearly shown.

(Mubarak, 2014), using Radial Basis Function Neural Networks, achieved an error rate below 1%. It indicates this method is appropriate for use with the Juanda airport. Meanwhile, (Lasmita, 2010) tried to predict the patterns of air traffic movements in *Adisucipto*, the principal airport serving the Yogyakarta area on the island of Java, Indonesia using the WEKA data-mining tool to simplify controlling apron movement.

In a study by (Castellani et al., 2010), passenger flows to Italy were used as a proxy to measure tourism demand. In order to study the monthly time series arrival data of Italian islands Sardinia and Sicily during 2003-2008, a set of ARIMA model specifications were constructed. Results showed that it is affected significantly by both meteorological variables, such as weather and temperature, and lagged values of the exchange rates USD/EUR and YEN/EUR. These factors were controlled for, which improved the explanatory and forecasting power of estimated ARIMA models (Castellani et al., 2010). (Andreon & Postorino, 2006) forecasted yearly inbound and outbound air transport demand at Reggio Calabria airport (Southern Italy). The authors found that the three models they used – two univariate ARIMA models and a multivariate ARIMAX model with two explanatory variables – were capable of generating accurate forecasts.

A study to forecast outbound air traffic flow from the United Kingdom to twenty destination was undertaken by (Coshall, 2006). In this study quarterly traffic flow was

modelled using naive models, Holt-Winters' model and a variety of ARIMA models. The ARIMA model outperformed the other models when forecasting performance was compared using Root Mean Square Error (RMSE).

Lim & McAleer (2001) forecasted monthly arrivals to Australia from three different destinations: Hong Kong, Singapore and Malaysia. They use the single exponential smoothing model, Brown's double exponential smoothing model, the additive and multiplicative seasonal Holt-Winters models and the non-seasonal exponential smoothing model. The models were evaluated using RMSE. They found that the multiplicative Holt-Winters model offered the best performance for Hong Kong and Singapore, whereas the additive Holt-Winters model was the most accurate for Malaysia.

A study undertaken by Kulendran & Witt (2003) forecasted international business passengers to Australia from four countries – New Zealand, Japan, United Kingdom and the United States – at one, four and six quarters ahead. The authors found ARIMA and Basic Structural Models (BSM) provided the most accurate models for short-term predictions (one quarter ahead) among the five other models they considered. The conclusion made by the authors indicates that the performance of the forecasting is influenced by the forecasting horizon and depends on sufficient seasonal unit roots detection.

Cho (2003) compared the forecasting performance of three models: ARIMA, the multiplicative exponential smoothing model, and the artificial neural network model (ANN) on the tourist arrival data to Hong Kong from Taiwan, Korea, Japan, Singapore, and the United Kingdom. He noted that ARIMA and Holt-Winters usually give a good forecasting performance. However, ANN outperforms the other two models for all but the United Kingdom, for which Holt-Winters offers the best performance.

Since the opening of the Hong Kong International Airport (HKIA) in 1998 the volumes of air passengers and cargo have grown steadily, except for the post 9/11 period and around the time of the SARS outbreak. A forecast of the airport passenger flows allows for short- and long-term planning for airport facilities and the flight network. Tsui et al. (2011) compared the performance of ARIMA and a multivariate ARIMAX model while forecast air passenger traffic at Hong Kong International Airport. They showed that the result of ARIMA forecasting is more accurate than ARIMAX in one over one

to three-month horizons during the period. Tsui et al. (2011) used the Box-Jenkins methodology to estimate a seasonal ARIMA and a seasonal ARIMAX to forecast the number of passengers using data from 1993-2011.

In contrast to the study by Carson et.al (2011), passengers were categorised into different groups depending on destinations. Eleven groups, such as Africa, Europe, Japan, etc., were identified and separate forecasts were computed for each region. The sum of these formed the aggregated airport forecast of the passenger throughput. The final model was a SARIMAX (p, d, q)(P, D, Q) $_{12}$, which is an extension to ARIMA that supports the direct modeling of the seasonal component of the series where the explanatory variables were significant. Instead of using the actual fuel price, a dummy variable set to 1 was used if the fuel price in month *t* was more than 80 dollars per barrel, and 0 otherwise. The authors used three different scenarios when producing the forecasts to account for changes in the explanatory variables:

1. The explanatory variables are assumed to take on values made by forecasts from external sources.
2. GDP is assumed to decrease at a 5 % annual rate while the oil price remains as in 1.
3. The oil price remains below 80 dollars per barrel.

The key finding of their work was that a SARIMAX model, taking into account GDP, oil price and connecting flights, made the best forecast with an average monthly deviation of 3,6 %.

Önder et al. (2009) used ARIMA models, traditional time series analysis, and the artificial neural network forecasting method for forecasting international arrivals to Turkey on monthly data between 1986 and 2007. They obtained forecasts for 2008-2010 and found that the two successful forecasting methods based on residual analysis and classical time series test were artificial neural networks and Winter's seasonal exponential smoothing technique.

Önder & Kuzu (2014) applied some classical time series smoothing and decomposition techniques to air traffic volume data from Turkey's airports between January 2007 and May 2013. They applied a simple moving average, simple exponential smoothing, linear moving average, linear exponential moving average with single parameter, Brown's quadratic exponential smoothing method and Holt's linear exponential

smoothing with two parameters to the passenger travel data and obtain yearly forecasted values for the year 2013-2023. They presented the analysis at aggregate levels on all airports in Turkey separated into two groups: domestic and international. They found that the Holt's linear exponential smoothing method with two parameters was the best method to be used for passenger traffic variable.

Cuhadar (2014) compared forecasting performance among various exponential smoothing and ARIMA models for monthly inbound tourism demand to Istanbul on data between January 2000 and December 2013. According to MAPE, a seasonal ARIMA model $(2, 0, 0)(1, 1, 0)_{12}$ showed the best performance among all other methods compared, and monthly forecasts were generated for the period January 2014 to December 2015.

Principal Component Analysis (PCA) is one of the multivariate analysis techniques usually used for correlation analysis, data reduction, data multidimensional and efficiency assessment (Taylor, 2003). Its main purpose is to simplify and make it easier to analyse the data by reducing the number of dimensions and compressing the data without much loss of information. PCA is able to use the first data space to compress most information into a couple of features. It attempts to search a subspace in which the variance is maximised (Timmerman, 2003). The PCA subspace is distributed through the corresponding top eigenvalues of the sample covariance matrix. PCA may be put onto both supervised and unsupervised machine learning. It's been used with good results in many applications and investigation areas. Different enhancements to PCA have been recommended to improve its efficiency or performance (Turk & Pentland, 1991). Most PCA methods might not produce desirable benefits when they handle real-world nonlinear data. As the nonlinear PCA and variants can effectively catch the nonlinear relation, they might provide much more effective power to cope with real-world nonlinear data (Van der Maaten & Hinton, 2008). Different feature extraction as well as selection strategies are recommended when using PCA, as it is a highly effective method of feature extraction involving a mathematical process in which a selection of correlated variables is transformed into a smaller selection of principal components consisting of uncorrelated variables. Numerous scientific studies have been done with this information dimension reduction technique (Coates & Ng, 2012). It is known that the main function of PCA is to figure out the most indicative vectors, i.e.,

the best eigenvalues corresponding to the given eigenvectors in a sample covariance matrix.

In a study by Carson et al. (2011) the multivariate approach based on Principal Component Analysis was found to produce more accurate forecasts of air travel demand than the univariate approach. In the study, data for 179 individual airports were used to compute a model where the aggregated demand was forecasted by the sum of all individual forecasts, taking into account the oil price and the unemployment rates in the regions served by the different airports. This approach was suitable for U.S. data where the economic structure of the country is very heterogeneous, i.e. there are large differences in the economic structure within the country.

The unemployment rate in New Jersey differs considerably from the unemployment rate in Washington, which is why it is reasonable to treat each region separately when computing forecasts. The main finding was that the multivariate approach outperforms the univariate approach in terms of forecast error measurements.

The previous studies suggested that the performance of PCA forecasting models is influenced by the forecasting horizon, the passengers' origins and destination countries, and the market segment. Besides, these models usually forecast in the short and medium term more accurately.

Econometric demand modelling is one of the most widely explored and active method in the transportation sector (Wadud, 2011). It involves building a causal relationship between the demands of passengers in independent explanatory variable sets. In the past decades, the econometrics model has undergone significant advances to include sophisticated activity-based models that use random utility theories (Hill et al., 2001).

Some factors that affect air travel demands include income or GDP, ticket price, fuel price, unemployment, travel time, frequency of flights, population size, exchange rates, and export and import factors (Wadud, 2011; Wadud, 2013; Profillidis, 2000). However, it was pointed out that the income or GDP is the most important factor in the econometric method for demand forecasting, since it reflects a country's economy directly (Wadud, 2013). Therefore, GDP per capita is an indicator to understand the average standard of living in a country. Attention must be paid to the aspects of the equitable distribution of income in a high GDP per capita country. There is also an argument that the opportunity to develop the economy of a country with a growing

population is higher than people who did not develop at all (Önder et al., 2009). For example, the economic rationale depends on the determination of balances between the population with the available natural resources, economic planning, the amount of income per capita, the amount of labour available for construction and how much manpower is available to manage industry, agriculture and natural resources.

The econometric factor is an important variable in air travel demand modelling. However, it has been ignored in many relevant studies. Spence (1975), cited by Ippolito (1981), considered that the econometric factor is the critical factor in product differentiation.

In a study by De Vany (1975), the omission of the econometric factor has been caught, and the flight frequency is integrated as an indicator of reflecting the quality of service. Although the empirical results of the study when the flight variable frequency was considered were unsatisfactory, it has the merit of using the service variable quality as a determinant of air travel demand.

These empirical studies show that the demand for air travel is proportional to the frequency of flights and inversely proportional to the load factor. However, the result of Ippolito (1981) is that only a small fraction of those studies that were carefully designed to eliminate the influence of empirical and theoretical problems. Therefore, this result is related to the assumptions in these studies.

In order to forecast the air traffic demand of Saudi Arabia, a stepwise regression technique is used to find the best econometric model (Ba-Fail et al., 2000). In this study, authors split the income into several sectors: oil-GDP, government non-oil GDP, private non-oil GDP, and total non-oil GDP. It was found that the model includes the explanatory variables as total spending and that the population size is an appropriate model for forecasting the domestic air traffic demand in Saudi Arabia. Either aggregate income or any of its components are not significantly explanatory of international air travel demand in Saudi Arabia.

Alperovich & Machnes (1994) used permanent instead of current income as a relevant variable in predicting the demand in international air travel. It was concluded that the main reason for misspecification in the previous studies was the exclusion of variables representing the consumer's wealth. The authors found that the air traffic demand for out of Israel is income-elastic and price-inelastic, which supports the basic assumption

indicating that the key determinant for air travel demand is the wealth of consumers. The consumers' wealth is divided into two categories, financial and non-financial, and the authors argued that the propensity to travel probably is different based on the source of income, and so an aggregate income variable that only combines incomes is not adequate. The major innovation of this study is that it found the wealth variables have significantly high positive signs. Hence, the air travel demand depends not only on the current income but also the wealth of consumer.

Besides, the exchange rates have a significant effect on air travel demand, including airline decisions, consumer decisions, and financial accounts (IATA, 2017). Changes in foreign exchange affect the consumer demand, but the degree of effect varies based on routes. The degree of variance also relies on other factors, such as the balance of travel between degree of substitutability and specific routes. Moreover, exchange rates also affect the decisions of airlines in terms of supply. In order to keep the balance between the supply and demand, airlines have to undertake price adjustments. Otherwise, permanent changes in the exchange rates might have influence on the long-term investment and network planning decisions of the aviation industry. In other words, as assumed by (IATA, 2017), foreign exchange fluctuations also affect an airline's financial system through its balance sheet valuations and daily profitability activities.

3.7 Forecast Combination

In the last four decades, time series forecasting has been studied extensively, and numerous empirical studies have been conducted to compare various methods in out-of-sample accuracy. Among these studies, there are three biggest competitions, namely the M-competition in 1982, the M2-competition in 1993 and the M3-competition in 2000. The general results of these methods can be summarised as below (Makridakis & Hibon, 2000):

1. All of them are complex, like ARIMA; less sophisticated approaches like exponential smoothing are not necessary.
2. Forecasting performance requires accurate measurement.
3. Forecasting performance relies on different time horizons.
4. Forecast combination based on averages outperforms individual methods. There are arguments on whether or not empirical evaluations are properly applied to the model's performance.

Although plenty of literature on forecast combinations is available, a straightforward method for the right approach has not yet been found.

The relative performance of the models depends on the correlation between the estimated sample size and forecast errors, and the error variance of the individual forecasts (De Menezes et al., 2000). While multivariate approaches are more suitable for small sample sizes, regression and univariate methods are properly suited to bigger datasets.

Comparing univariate time series forecast combination studies, the research works on multivariate methods are small. Studies about forecast combinations of neural networks reported a significant improvement on linear forecasting and individual forecasting combination models (Shi & Liu, 1993; Shi et al., 1999). However, in these studies, only a very short time series is applied to forecasting, and the neural network architecture details are not specified. The performance of neural networks is found to be mixed dependent on measurement errors and forecasting horizons in a more extensive study (Donaldson & Kamstra, 1996). Deutsch et al. (1994) reported a notable improvement on one specific time series in out-of-sample forecast error; extensive empirical studies gave discouraging results (Terui & van Dijk, 2002; Stock & Watson, 2004).

In summary, generally, the forecast combinations are considered to be beneficial in most of the literature (Makridakis & Hibon, 2000; De Menezes et al., 2000). In time series forecast combination or combination with weight changes, the most suitable method is based on very small datasets, in other words, the adaptivity and nonlinearity are not beneficial for every time series based on extensive study reports with mixed results.

3.8 Integrating Uncertainty into Airline Passenger Forecasting

In the civil aviation sector, numerous studies have been undertaken to forecast airline passenger numbers either for a particular airline, or for an airport, or for a whole country. Simple to advanced quantitative, qualitative or decision analysis has been applied depending on the data structure and the complexity of problems (Taneja, 1987; Nam & Schaefer, 1995; Weatherford et al., 2003). The main purpose of these methods was to set up a procedure to forecast air traffic data. It is also important to consider the

uncertainties around these projected values during the aviation planning process about the intended use of these forecasting methods (or set of methods).

Kincaid (2006) pointed out three common procedures related to the air traffic forecasting that explained uncertainty:

- High and low forecasts. For example: in developing demand forecasts under this procedure, many of the forecasting assumptions are adjusted in one direction to create a positive forecast, and to produce a negative forecast in the other direction.
- What-if analysis. In this procedure, the influence of a single event, such as a dramatic rise in fuel prices, is estimated and reported in relation to the model forecast. Uncertainties are normally expected to be event-specific and are described as either threats or opportunities.
- Sensitivity analysis. In this procedure, forecasting assumptions are diverse, considered one at a time, and the variations in the expected results (e.g., a tourism demand forecast) are reported consequently.

It is also pointed out that these approaches provide a superficial understanding of the risks and uncertainties involved and are rarely used in any meaningful way in the planning process. There are analytical procedures available that have the potential to incorporate the underlying risk and uncertainties involved while using a forecasting model. Though not used widely in aviation data, the understanding of associated risks while using any forecasting models can play an important part in decision making, especially when choosing a model from a set of competing models with similar forecasting performance. When some potential forecasting models with similar performance are available, more weight should be given to the one with the highest accuracy and least risk involved.

The incorporation of associated risk and uncertainties into forecasting can be categorised in the following two ways:

- Objective probability: data-driven processes that involve analysing historical data.
- Subjective probability: judgment-driven processes including the judgement as well as experts' and stakeholders' points of view.

Intelligent use of these types of procedures in conjunction with the forecasting model applied can be used to advance the comprehension about mitigating risk and control uncertainty within the airport context. These could be further augmented by applying computational methodologies such as Bootstrap validation, as stated by Efron (Efron, 1979; Efron, 1983).

To our knowledge, no such method was applied to quantify the risk associated with the projected value by a forecasting model in the aviation sector. The availability of small data is another issue in our study since existing literature used large historical data to forecast airline passenger.

3.9 Forecasting Performance Evaluation

A good forecasting model should capture some or all of the statistical properties of the underlying data generating process. However, it is not required to believe that the used forecasting model did generate the observations. According to the accuracy of the forecasts obtained with a system, the system itself could be considered reliable. One approach to determine the level of accuracy is to compare the current results with those forecasted by the model. It is also possible to adapt the model to only part of the data (training set) and applying the model to the other available data (test set) to test its accuracy. Moreover, the second subset could be considered an ex-post testing to determine both the method (process) accuracy and the functional form of the model.

Forecasts can be obtained for a "sequence of forecast horizons" starting from a common origin, or in a "rolling" fashion where the historical data are updated at each iteration, but a consistent forecast window is maintained. Whatever way the forecasts are generated, it is very important to evaluate the accuracy of the forecasts generated before using the model in a real-life application. The forecast accuracy measure plays a more important role when a model needs to be selected from a class of potential forecasting models. For a given observed series, y_h, the forecasts $\hat{y}_h$ ($h = 1, \ldots, H$) can be obtained by applying a forecasting model. There are several forecast accuracy measures available in the literature, and can be separated into three main groups:

1. Scale-dependent errors.
2. Percentage errors.
3. Scaled errors.

3.9.1 Scale-Dependent Errors

The forecast error is given by the difference between the observed and the forecast value, $e_h = y_h - \hat{y}_h$, which has the same scale as the data. Accuracy measurements based on e_h depend on the scale and, consequently, cannot be used to evaluate a series based on diverse scales, for example, forecasts from the original scale and log scale of the same series.

The three more frequently adopted scaled dependent measures are:

$$\text{Mean Absolute Error(MAE)} = \frac{1}{H}\sum_{h=1}^{H}|e_h| \tag{3.67}$$

$$\text{Mean Squared Error(MSE)} = \frac{1}{H}\sum_{h=1}^{H}e_i^2 \tag{3.68}$$

$$\text{Root Mean Squared Error(RMSE)} = \sqrt{\frac{1}{H}\sum_{h=1}^{H}e_h^2} \tag{3.69}$$

When comparing different forecasting methods based on a single dataset, the MAE is popular because of its simplicity and easy to understand inference. In the literature, MAE stands for Mean Absolute Deviation (MAD), and MSE stands for Mean Squared Prediction Error (MSPE) (Armstrong, 1978; Hyndman & Koehler, 2006).

3.9.2 Percentage Errors

The percentage error is given by $p_h = 100e_h/y_h$. The common percentage-error-based measurements are:

$$\text{Mean Absolute Percentage Error(MAPE)} = \frac{100}{H}\sum_{h=1}^{H}\left|\frac{e_h}{y_h}\right| \tag{3.70}$$

$$\text{Mean Absolute Percentage Deviation(MAPD)} = \frac{\sum_{h=1}^{H}|e_h|}{\sum_{h=1}^{H}|y_h|} \tag{3.71}$$

$$\text{symmetric Mean Absolute Percentage Error(SMAPE)} = \frac{200}{H}\sum_{h=1}^{H}\frac{|y_h - \hat{y}_h|}{|y_h + \hat{y}_h|} \tag{3.72}$$

Being scale-dependent, the percentage-error-based measures are commonly utilised to compare the forecast performance of different datasets. However, the main drawbacks of these measures are that they can be infinite or undefined if any y_i equals zero or is near zero. Also, they influence the negative errors more than the positive ones (Armstrong, 1978; Hyndman & Koehler, 2006).

3.9.3 The Use of Scaled Errors

Hyndman & Koehler (2006) proposed how to scale the errors using the Mean Absolute Error from a simple forecast method instead of percentage errors when the forecast accuracy is compared across series on multiple scales. Moreover, naïve forecasts, which are considered an effective way of defining a scaled error for a non-seasonal time series, are given by:

$$q_h = \frac{e_h}{\frac{1}{T-1}\sum_{t=2} H\,|\,y_t - y_{t-1}\,|} \qquad \textbf{3.73}$$

For seasonal series with period m, a scaled error uses a seasonal naïve forecast:

$$q_h = \frac{e_h}{\frac{1}{T-m}\sum_{t=m+1} T\,|\,y_t - y_{t-m}\,|} \qquad \textbf{3.74}$$

Since both denominator and numerator involve values of original data, q_j is independent of the scale of the data. The following measure can be derived as:

$$\text{Mean Absolute Scaled Error(MASE)} = \frac{1}{H} q_h \qquad \textbf{3.75}$$

If a scaled error is less than one, it means a better forecast result is obtained compared to the average naïve forecast on the sample data. If it is greater than one, on the other hand, it indicates that the result is worse than the naïve forecasting result (Armstrong, 1978; Hyndman & Koehler, 2006).

3.10 Measuring the Accuracy of Forecasts Using Training and Test Sets

Consequently, Rob Hyndman pointed out that it is necessary to evaluate forecast accuracy using genuine and real forecasts. Hyndman & Koehler (2006) stated in the accuracy of forecasts that the accuracy is determined by only considering how a model

performs on a new set of data and not using fitting tools. It is useless to look at how well a model fits the previous data.

When selecting models, it is necessary to use some of the available data for fitting, and use the rest of the data for testing purposes, as was done in the above examples. Then the testing data can be used to measure how well the model is likely to forecast on new data. The size of the test set usually should be around 20% of the sample, while this also depends on the sample length and how far ahead the intention is to forecast. The size should be at least equal to the forecast timespan. The following points should be noted:

- A model that fits the data well won't necessarily fit the forecast well.
- It is always possible to find the perfect model by using enough parameters.
- Over-fitting a model on a set of data is as bad as failing to figure out the systematic pattern in the data.

In literature the "training set" is also known as the "in-sample data" and the "test set" is also known as "hold-out set" or "out-of-sample data".

3.11 Feature Engineering and Machine Learning

In a predictive model, the input vectors are easily conceivably transformed or augmented to promote predictive performance. This behaviour is generally referred to as feature modification, feature extraction, or feature engineering. In machine learning, feature engineering refers to the process of creating or selecting variables from a set of data to improve machine learning results (Domingos, 2012). Feature engineering is also known as feature extraction, feature construction, or feature generation. There are different explanations for the terms of feature engineering, extraction, construction, and generation. Some of them are quite close, however, the difference between these terms are: the building of features from raw data (Muhammad et al., 2015; Kanter & Veeramachaneni, 2015); the creation of new features from one or multiple features (Markovitch & Rosenstein, 2002); and the creation of a map of original to new features conversion (Motoda & Liu, 2002).

Unless otherwise specified, the term "feature engineering" and its synonyms used in this Thesis refer to the process of creating new features from one or multiple features. These methods are important for linear regression in that they change model input into

favourable forms (Freeman & Tukey, 1950). These changes are applied in order to reduce residual errors in linear regression. Box & Cox (1964) showed a method of finding how to helpfully transform several power capabilities into a linear regression input. The algorithm used in their study is known as the Box Cox transformation. Power transformations use exponents of the variables of a machine learning model. There are other ways to conduct a transformation by mathematical functions; for example, using logarithms is a good option. Linear regression, which benefits from feature engineering transformations, is not the only machine learning model. The single features are independent of one another with these simple transformations.

The Box Cox transformation depends on the stochastic sampling of the data and cannot guarantee an ideal set of transformations. Breiman & Friedman (1985) proposed an alternative conditional expectation (PCA) algorithm, which can ensure great transformations for linear regression. A set of maximum transformations for every one of the predictor features is used in a PCA algorithm for linear regression. Although, the initially designed transformations ensure linear regression, it also shares advantages with other models (Cheng & Titterington, 1994).

In almost all machine learning model types, an extra grid search for optimising parameters is a typical means of feature transformation. By using grid search matching on specific features, it's possible to clean the data and lower over-fitting. The knot numbers and its places within the grid search is the "hyperparameter", which is assigned for this specific transformation technique. Extra grid search is able to make a good approximation of the form of various other features. Brosse et al. (1999) used extra grid search for changing data in a neural network used to forecast the division of fish populations.

Machine vision is also a widely used program in feature engineering. An early type of computer vision feature engineering is the Scale Invariant Feature Transform (SIFT) (Lowe, 1999). In a machine learning model with the purpose of learning to identify figures, the model might not see the very same figures at different scales, which is problematic. SIFT pre-processes the data and provides them in a format in which the images are created in a scale of different identical features. These feature types can be generalised for different issues and used in machine vision, including object recognition stitching and panorama stitching (Brown & Lowe, 2003).

Another widely used machine learning algorithm is text classification. Scott & Matwin (1999) used feature engineering for text classification in order to improve the overall performance of understanding rules. It allows the structure and frequency of the text to be generalised to different features. A machine learning model represents textual data that creates a substantial number of dimensions. Feature engineering is used to decrease the dimensions in text classification.

Feature engineering has been found to be important in the Kaggle and KDD Cup data science rivalries. One group used feature engineering and an ensemble of machine learning models to win the KDD Cup 2010 rivalry (Yu et al., 2011). Histograms of arranged slopes with different features were exhibited in this opposition. Cheng et al (2011) developed an era of mechanised feature algorithms for data sorted by domain-specific knowledge. The finding had numerous applications. For example, Ildefons & Sugiyama (2013) won the Kaggle Algorithmic Trading Challenge with a feature engineering and model ensemble. The features engineered methods in these rivalries were made physically or with information about the particular rival issues. Such learning indicates domain-specific knowledge and human instinct that could not be re-generated by a machine. Feature engineering is also used for natural language processing (NLP). One engineered feature for NLP is the frequency-inverse document frequency (TF-IDF). Basically, this engineered feature estimates how important a word is to a document in a collection or corpus (Rajaraman & Ullman, 2011). TF-IDF is the mainstream tool for content classification, text mining, and NLP.

Machine learning algorithms can perform feature learning automatically. Frequently, these calculations are unsupervised. Coates et al. (2011) specified a neural network with an unsupervised single layer for feature engineering. PCA (Timmerman, 2003) along with t-distributed stochastic neighbour embedding (Van der Maaten & Hinton, 2008) are proven to achieve success for automatic feature engineering for dimension reduction algorithms.

Deep neural networks need a wide range of layers in order to learn complex collaborations in the data. Deep learning benefits from feature engineering despite its innovative learning ability. Bengio (2013) reported that feature engineering is beneficial for computer vision, speech recognition, signal processing and classification. Le (2013) developed high-level features using unsupervised procedures to engineer a

deep neural network for signal processing. Lloyd et al. (2014) used feature engineering to produce the Automatic Statistician project. This framework used an unexpected regression issue model and created readable reports. It can figure out the transformation forms that could improve particular features. Kanter & Veeramachaneni (2015) engineered a method known as "deep feature synthesis" that transforms relational database tables instantly into the feature vector that is used for the regular machine learning model. A neural network's feature vector input cannot encode directly the many-to-many and one-to-many relations that are easy to encode in RDBMSs. The deep feature synthesis algorithm uses SQL-like transformations, such as MAX, MIN, and COUNT, for summarising and encoding the relations into a feature vector. The deep feature synthesis was found on their algorithm's potential to outperform a couple of competitors in three data science competitions. Automated feature engineering was also implemented for particular domains in some studies. Davis & Foo (2016) applied an automated feature detection to a HTTP traffic and tunnel modelling task. Cuayáhuitl (2016) invented SimpleDS, which is a text document processing approach that does not make use of manual feature engineering, instead using a full reinforcement learning system. Manual feature engineering is still popular due to its value in various contexts. Bahnsen et al. (2016) showed guidelines for and examples of feature engineering for fraud detection relating to credit cards. It is a good example of specific knowledge domain application. Zhang et al. (2016) studied feature engineering for phishing detection. Although these methods are used successfully in some specific domains, they do not relate to the issue statement suggested by this study, which is producing a generic automatic feature engineering system.

Feature engineering might remove unnecessary or redundant features. This process requires assessing the relevance of the variable. It can be implemented by creating a model to test variable correlations to the dependent variable. Feature engineering involves creating new variables by combining multiple variables and modifying variables (Kern, 2014). In this Thesis, the first use of feature engineering is the selection of the relevant variables. The input data may contain too many variables, some of which do not improve the prediction performance; thus, the predictive model contains too many features and becomes overly complicated. Therefore, unnecessary variables must be removed in order to make the model more efficient. Variable removal

can be done manually, based on domain knowledge, or automatically (Domingos, 2012).

The main two goals of feature engineering are accuracy improvement and dimensionality reduction (Kern, 2014). When the goal is accuracy improvement, the resulting feature space will contain more features than the original feature space. When the goal is dimensionality reduction, while the resulting feature space will contain fewer features than the original.

In this Thesis, the main goal is to improve the accuracy of the predictor in feature generation methods. Dimensionality reduction has less priority, since the feature generation results are input to a feature selection phase with the aim of reducing the dimensionality of the feature space. However, dimensionality reduction has to be taken into account to avoid generating an extreme number of new features. The importance of feature generation is illustrated in Table 2.1, considering a specific example.

Table 3-1 Features are Date and Visitors; extracted feature is IsWeekendDay

Date	Visitors	IsWeekendDay
2019-04-13	18,325	Yes
2019-03-11	6,426	No
2019-01-26	18,845	Yes
2019-04-08	6,434	No
2019-02-18	6,924	No

In Table 2.1, the original feature is “Date” and the dependent feature is “Visitors”. It shows a date and the corresponding number of visitors for a theme park. It can be seen that there is no obvious pattern between the predicting and the dependent feature. By using feature engineering, we can extract what kind of day the date is, shown in the “IsWeekendDay” column. This indicates whether the date is a weekend day or not. There is a clear pattern that implies the number of visitors is significantly higher on weekend days than on weekdays. Another situation where feature generation can improve performance is when there is feature interaction. Then two (or more) features are not correlated or relevant to the dependent feature on their own, however, together they have a (high) influence on the dependent feature. For example, in the case of the

quality and price of a product, they will not give much indication of whether a product is purchased often when these features are separated. However, they have a high correlation to the purchase of the product when combined. If the quality is high and the price is low, then the product will be purchased often. However, without knowing both price and quality value, it cannot guarantee that the product will be purchased often. If both the price and quality are low, then the product will not be purchased by many customers, and vice versa.

In this Thesis, the feature selection was done by observing the output of the linear regression model to find how much correlation each variable has with the dependent variable. The other purpose of using feature engineering in the Thesis is modifying variables. Examples include combining multiple variables to create a new variable; calculating a variable differently so that it can be used better in classification; and categorising a variable so that it has a limited range of possible values.

3.12 Discussion

This chapter introduced and discussed the forecast combination approaches with a focus on their application to the airline industry. It should be kept in mind that forecasting tasks can vary by many dimensions, the length of the forecast horizon, the size of the test set, forecast error measures, the frequency of data, etc. It is unlikely that once selected, a forecasting method will be better than all other plausible methods all the time. Generally, the sensible likelihood coming from the forecast model should be frequently analysed depending on the current task and when a new data set is available. A short introduction to airline demand was presented, and the importance and need for effective forecasting procedures in the airline industry has been explained and justified.

The study concentrates largely on the research topic proposed, that is, regarding forecasting monthly airline passenger numbers using Principal Component Analysis and its implementation to deep neural networks. This provides the opportunity for building the ability to manipulate expressions. The concept of Principal Component Analysis will enable the dissertation algorithm to recommend engineered features. While choosing a forecasting model for future use, it is very important to consider the confidence bands around the point forecasts value to evaluate the performance of the selected model on the unforeseen data.

All previously mentioned studies in this chapter used only the point forecasts while comparing the forecasting performance among methods. This Thesis will reduce this gap in the literature by considering the prediction interval as well as point forecasts while proposing forecasting models for Muscat International airport.

Chapter 3: Air Demand in Oman and Predictive Modelling

4.1 Introduction

Oman is a fairly large country (310,000 km2), with a long coastline reaching down to the Indian Ocean in the south, and is strategically placed at the mouth of the Gulf at the southeast corner of the Arabian Peninsula – see Figure 3.1 (Calvin, 2016). It has an extensive mountain range spanning the country, creating a variety of climate zones. Moreover, Oman has numerous highlands within its geographical niche, with Muscat, Dhofar, and Nizwa being the highest tourist attraction sites (Cullen & Kusky, 2010). However, Oman's economic growth has not ultimately reflected the development of the infrastructure, and it is still considered a developing country, regardless of the nation's steady economic growth.

Figure 4-1 Location of Oman's four airports.

In Oman, the aviation industry is one transportation sector that has experienced slow development, which has primarily compromised the level of service quality of that industry in the country. However, as postulated by (Omanair, 2011), airline carriers in Oman have experienced numerous problems that can be traced to the planning and policy structure of Oman. For example, the significant growth of the country is

affecting the location of its airports. As a result, this eventually affects the effective planning of airports in Oman, which has a negative effect on its airline industry development. Additionally, with the projected increase in demand for the aviation industry, there is a dire need to develop a well-structured aviation planning and policy framework (Omanair, 2017). There are currently three domestic airports and one international airport in Oman (Oman Airports, 2018), as shown in Table 3.1.

Table 4-1 Domestic and international airports in Oman.

Domestic and international airports in Oman		
City Served	**IATA**	**Airport name**
Duqm	DQM	Duqm Airport
Muscat	MCT	Muscat International Airport
Salalah	SLL	Salalah Airport
Sohar	OHS	Sohar Airport

Muscat is the capital city and political and economic centre of Oman and, accordingly, Muscat International Airport is the most significant and busiest airport in the country. While Salalah and Sohar airports also cater for international flights, their share is relatively small. International flights through Salalah Airport stood at 574 in January 2019 compared to 55 flights during the same period through Sohar Airport. Table 3.2 shows the positive change in the annual passenger traffic for Muscat International Airport for the past four years. Figure 3.1 above presents the location of these three airports. Almost all domestic flights start or end at Muscat International Airport.

Table 4-2 Positive change in passenger traffic.

Muscat Airport Annual Passenger Traffic		
Year	**Passengers**	**% Change**
2018	15,392,580	9%
2017	14,034,865	28%
2016	10,314,449	18%
2015	8,709,505	5%

The airports are located far from each other, and domestic flight times are over 50 minutes from Muscat. Despite the distances, there are four viable airports in Oman due, in part, to the lack of significant road connections. Besides, the railways are not well developed, so the main cities and adjacent countries are not linked together (Oman Airports, 2018). Therefore, travelling by land is time-consuming and unpleasant. However, this situation is changing in recent years with the development of land transportation, including new roads and interconnected rail bridges (Oman Airports, 2018). The number of air passengers from Oman has increased steadily during the last decade, which is why it is of interest for the airports, airline companies and authorities to obtain forecasts to plan future production and logistics. Forecasting airline passenger numbers have attracted numerous researchers in recent years due to its essential role in people's daily life. The research works of this Thesis are focused on forecasting Oman air passenger volumes. The main objective of this study is to evaluate accurate monthly air passengers in Oman for international flights. In Oman, transportation has been considered an important economic sector of the country for a long time (UNWTO, 2016). It has been proven that air transportation depends significantly on the economic activity of the country (Oxford Economics, 2011). There was a remarkable growth of air passengers in Oman between 2006 and 2017. The handling capacity of Oman airports was 12 million passengers in 2016, which more than doubled to 4.7 million by 2006 (IATA, 2018). Predicting future air passenger volumes is essential, as it allows air transport authorities to construct and implement airport infrastructure facilities for future needs and offers airline companies the capacity to match the increasing passenger demand for air transportation. Furthermore, as an essential employer in Oman, the air transportation industry contributes to a prosperous Omani economy both directly and indirectly. For example, 15.9% of Omani employees were working on-site in the air transportation sector, compared with 6.2% in the road industry (NCSI, 2017).

The purpose of forecasting depends on the time duration of the forecast. Generally, short-term forecasting spans a period of a few months to one year and requires daily operations; the medium-term forecasting considers a period of one to five years and involves route-planning decisions; and long-term prediction spans a period of 5 to 10 years and decisions for airport and airline infrastructures have to be taken account. Also, air travel demands are the main factors for route development, fleet planning and annual operating planning of the airline companies. Analysing and forecasting air travel

demand helps reduce the airlines' and airports' risk by objectively evaluating the demand side of the air transport business. Accurate forecasting is essential for airport and airline managers to plan effectively and efficiently, as inaccurate prediction may lead to bad decisions and ineffectiveness in overall operations of management. Therefore, this study provides forecasts for air passenger numbers in Oman based on various models and techniques. Given that air transportation demand is highly linked to the economy (Adeniran & Stephens, 2018), which is typically characterised by business cycles, or alternations between periods of economic growth and downturns, the data series should exhibit cyclical patterns or seasonality. Any economy is highly susceptible to a variety of fundamental issues, such as political, economic, climate issues, etc., which are likely to modify the past trends and the volatility in the data. Both of these characteristics have been taken into account in the time series models presented in this study.

4.2 Datasets Overview

The research objectives were developed to address the shortcomings associated with demand forecasting methods. Firstly, to determine the factors that affect the number of airline passengers travelling by using econometric models of projection, with an emphasis on the use of panel data that comprise observations of multiple phenomena taken over various periods for the same subject. Secondly, the economic factors that affect airline passenger numbers have to be determined by an econometrics projection model with an emphasis on the use of time series data. There are several factors affecting air travel demand; each factor is composed of elements, which can stimulate or constrain air travel growth (Vasigh & Fleming, 2016). For the purpose of air travel demand analysis, these factors are more conveniently classified into two broad groups: those external to the airline industry, and those within the industry (Ba-Fail et al., 2000). The comprehensive search of the literature on previous air travel demand forecasting studies reported in the leading journals and literature, and presented in Chapter 2, showed that there is a range of socio-economic factors that influence air travel demand. Real GDP, real GDP per capita and airfares were the most common explanatory variables included in these studies (Ba-Fail et al., 2000). GDP is the factor that is directly proportional to air passenger demand: if the GDP is increasing, people can travel abroad, and air demand will increase. Similarly, if the GDP increases because of good infrastructure, tourist arrival will also increase. Other important factors

that influence air travel demand reported in the literature include unemployment (McKnight, 2010), tourism demand (Koo et al., 2018), world jet fuel prices (Gesell, 1993), and real interest rates (Cook, 2007; Wensveen, 2007). Similarly, short-term conditions such as inflation, interest rate and currency exchange rates can have a substantial effect on the growth potential of both individual airlines and the entire industry. The obtained insights could help airlines and managers of all airports in Oman. The explanatory variables available for this study are described in Table 3.3:

Table 4-3 Exploratory data analysis.

Data Cleaning and Exploratory DataAnalysis
Tourism statistics: General indicators of tourism statistics
Number of guests: Number of guests in different regions
Hotels: Hotel statistics
Climatological summary: Climatological information of Muscat and Salalah airports
Component IDL, which contain pax movements, including: Total freight; Mail Traffic; Aircraft; Passenger movement of Muscat and Salalah airports.
Other variables, such as: Oman's jet fuel price; Oman's real interest rates; Oman's unemployment size; Oman's real GDP; Oman's real GDP per capita; Oman's population size; Incoming passengers of Oman; Outgoing passengers of Oman; Total passengers of Oman; Incoming aircraft movement of Oman; Outgoing aircraft movement of Oman; Total aircraft movement of Oman.
Airline schedule
Pax: Passengers departing from Oman to any international destination
Distance: Distance between Oman and any international airport
Avg. Base Fare: Average base fare from Oman to any international destination
Public Holidays: Oman's Public holidays

4.2.1 Cleaning Process Description

The dataset contains 11 data files from multiple sources. To process data for the study, the following process was followed to ensure the data is internally consistent by having the same content and format of each data type:

1. Data Layout Standardisation;
2. Data Cleansing; and
3. Data Collation – appending similar data from the same sources.

Considering most of the data were present in the report layout instead of tabular form, a tabular format for each file was designed. To ensure that these files were easy to maintain and update, a standard table format was designed for all the files (wherever possible). i.e. information like month or year were kept in rows rather than creating a column for each month. Due to the traditional report level layout of the data files, data had to be cleansed and then transformed into the above-defined formats (Table 3.3). Data cleansing and transformation were undertaken automatically. Data from different files were then moved/appended to the required files one by one. The data were thoroughly checked to ensure that there was no loss or change in data. Given the data provided in Excel format as input, the desired output is a single table of data. There is a total of 11 tabs (or tables) of passengers' data and economically relevant data. The raw data are not in a form suitable for data analysis. The desired format should have each variable (or predictor) as a column and each observation as a row. There was inconsistency in the data format that had to be normalised and there was (5%) missing data to be removed. R was used to clean, tidy, and join data to form a single view of all available information ready for exploratory data analysis. Data were checked for official statistics, explored for trends and anomalies, and documented with initial findings and analysis into a single R notebook file. Data pre-processing includes loading data from the Excel file, removing spaces from column names, and tidying data. The initial data cleaning and investigation revealed the following variables and data types. All processed data are listed in Table 3.4 below:

Table 4-4 All processed data.

Data	Period	Daily/Monthly/Yearly
Passengers departing from Oman to any international destination	1998 – 2016	Monthly
General indicators of tourism statistics	2009 – 2015	Yearly
Number of guests in different regions	2009 – 2014	Yearly
Hotel statistics	2009 – 2015	Yearly
Climatological information of Muscat airport	2005 – 2016	Daily
Total freight / mail traffic / aircraft / passenger movement of Muscat airport	2004 – 2016	Yearly
Oman's jet fuel price	1997 – 2016	Monthly
Oman's real interest rates	1986 – 2016	Yearly
Oman's unemployment size	1991 – 2016	Yearly
Oman's real GDP	1970 – 2016	Yearly
Oman's real GDP per capita	1970 – 2016	Yearly
Oman's population size	1961 – 2016	Yearly
Incoming passengers of Oman	2014 – 2015	Monthly
Outgoing passengers of Oman	2014 – 2015	Monthly
Total passengers of Oman	2014 – 2015	Monthly
Incoming aircraft movement of Oman	2014 – 2015	Monthly
Outgoing aircraft movement of Oman	2014 – 2015	Monthly
Total aircraft movement of Oman	2014 – 2015	Monthly
Airline schedule	2013 – 2016	Daily
Distance between Oman and any international airport	1998 – 2016	Monthly
Average base fare from Oman to any international destination	1998 – 2016	Monthly
Oman's Public holidays	1998 – 2016	Yearly

4.2.2 Proposed Time Series Predictive Modelling

Since there are various factors affecting air passenger volume prediction, in-depth study is needed of the internal and external environment of the aviation industry. The economic forecasting processes, such as the Benchmark model, the ARIMA model (i.e., autoregressive moving average model), the Random Forest model and the Casual model, are not only taking the economic phenomena into account in time series dependence, but also consider the interference of random fluctuations. These models, for short-term forecasting of economic operations, have a very high accuracy rate and have been widely used in different studies (Box & Jenkins, 1976; Cryer & Chan, 2008; Castellani et al., 2010; Tsui et al., 2014; Findley et al., 1998). The passenger volume can be regarded as a stochastic time series formed with the passage of time. By analysing whether the time series is stationary, and stochastic and seasonal factors, these models can be used in relation to civil aviation passenger transport. Therefore, this section will use the time series analysis of the ARIMA, Benchmark, Casual and Random Forest models using R software to fit the data to achieve airline passenger numbers forecast. A number of forecast algorithms have been described in Chapter 2. Many of the empirical studies mentioned have one thing in common: Forecasters have spent a lot of time and applied a lot of knowledge in designing more or less sophisticated methods tuned for specific time series.

This section describes an empirical experiment comparing the performance of forecast combination methods that have not been heavily tuned and are likely to be employed by users who are not forecasting experts. In practical applications with a vast dataset, this method is not feasible often, because often a significant number of forecasts has to be calculated every second. This section aims to provide some empirical evidence on the effectiveness of seasonal and trend time series on modelling and forecasting airline passenger numbers. Therefore, the main question is:

- Are time series methods able to directly model seasonal and trend variations and produce satisfactory forecasts?

Different levels of prediction will be investigated, depending on whether there are enough training examples. Time will be spent on finding the best way to include historical statistics (months lag data) for each observation. Several statistical models will be explored in an attempt to fit the data.

A short review of previous research will be presented in Subsections 3.2.3, 3.2.4, and 3.2.5. This is followed by a methodology part where the estimated models are described. Section 3.3 contains a description of the data, and then evaluated and compared considering forecast performance in the following sections. Analysis and discussion are presented in Section 3.4.

4.2.3 Benchmark Model

When modelling monthly time series with recurring patterns every year, Box & Jenkins (2015) established a two-coefficient time series model of factored form, which is nowadays known as the airline model (Findley et al., 1998). The model is often used to forecast passenger flow since it is both accurate and straightforward to estimate. The model has the form:

$$(1-B)(1-B^s)Y_t = (1-\theta B)(1-\Theta B^s)e_t \qquad \textbf{4.1}$$

It is a $SARIMA(0,1,1)x(0,1,1)_s$, where B is the backshift operator, Y_t is the number of passengers in time t, θ is a moving average (1) term, and Θ is a seasonal moving average term. $e_t \sim WN(0,\sigma^2)$. Thus, for monthly data (Findley et al., 1998), s=12. To forecast a $SARIMA(0,1,1)x(0,1,1)_{12}k$, the following representation is used:

$$(1-B)(1-B^s)Y_{t+k} = (1-\theta B)(1-\Theta B^s)e_{t+k} \qquad \textbf{4.2}$$

The airline model serves as a suitable benchmark model that is to be improved. The Box-Jenkins method proceeds in four steps: identification, estimation, diagnostic testing and forecasting. The first step in the Box-Jenkins method is determining whether the time series is stationary and whether it is necessary to be modelled for any significant seasonality. The stationarity of the time series can be evaluated by running a sequence plot where it shows the constant location and scale. The stationarity can also be detected by an autocorrelation plot with prolonged decay. Box and Jenkins recommend the differencing approach to achieve stationarity. However, fitting a curve and subtracting the fitted values from the original data can also be used in the context of the Box-Jenkins model. Typically, for monthly data, one would include either a seasonal MA 12 term or a seasonal AR 12 term. In Box-Jenkins models, removing seasonality before fitting the model is not necessary. Instead, the seasonal terms in the

model specification can be included in the ARIMA estimation software. Once the model has been specified, its moving and autoregressive average parameters, as well as seasonal equivalents in the case of the SARIMA model, have to be estimated. The non-linear least squares are one of the most used methods. The estimation is carried out in R with the ARIMA function of the forecast package. Model diagnostics for Box-Jenkins models are similar to model validation for non-linear least-squares fitting. That is, the error term A_t is assumed to follow the assumptions for a stationary univariate process. The residuals are supposed to be white noise drawing from a fixed distribution with a constant mean and variance, or independent when their distributions are normal. If the Box-Jenkins model is appropriately used, the residuals should satisfy these assumptions. If these assumptions are not met, a more appropriate model should be used for fitting. In this case, one should go back to the model identification step and try to develop a better model.

4.2.4 Causal Model

To develop forecasts for airline passenger numbers for Muscat International Airport, this study attempted to create a causal model. To establish a causal model, the following steps must be followed:

1. Data examination;
2. Searching for main causes;
3. Statistical analysing for short-range forecast development; and
4. Creating long-range forecasts by scenarios.

It is recommended that a detailed data examination should be done first, since the available data are often full of errors, inconsistencies, and other factors that introduce spurious fluctuations into past trends (Hyndman & Athanasopoulos, 2013). Searching leading causes of changes in passenger flows is the next appropriate step (Taylor, 2003). It is critical to determine at this stage if there are any particular factors over other factors that should be entered into the model. At this point, the investigations determine which elements of the model are correlated significantly with passenger flows and whether the overall model fits the data (See Figure 3.2).

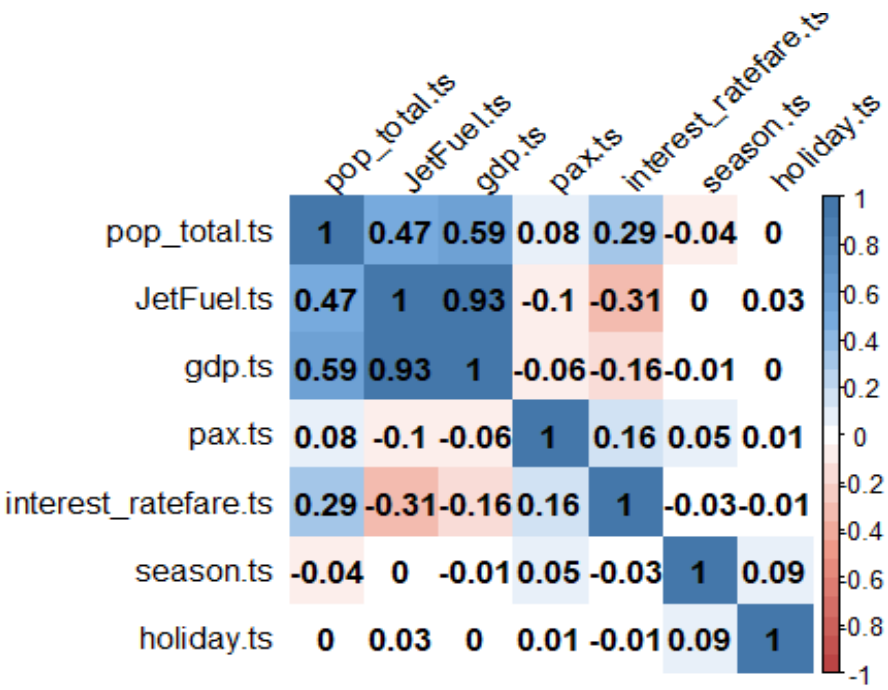

Figure 4-2 Correlations between Total Passenger and the other variables.

If it fits, the model may not be very accurate but at least plausible. This justification of the model is sufficient for starting preparation for short-range forecasting. However, it is probably not reasonable to expect that this statistical model will produce reasonable forecasts for long-range forecasting, where the situation may change dramatically. Our approach, then, is to develop scenarios that describe how the significant causes of airline passenger numbers might change and use them to prepare order-of-magnitude forecasts of flows. It probably is not very elegant; however, it provides an honest projection of low and high levels of flow.

The causal model, which aims to improve the forecasting performance of air passengers, is an expansion of the airline model with explanatory variables (Hyndman & Athanasopoulos, 2013). The model is defined as $SARIMAX(0,1,1)x(0,1,1)_{12}$, and it is outlined as:

$$(1-B)(1-B^s)Y_t = (1-\theta B)(1-\Theta B^s)e_t + \sum_{i=1}^{b} \beta_i X_{i,t-v} \qquad 4.3$$

where B is the number of exogenous variables included in the model and v is the lag. The explanatory variables can be lagged or not lagged, or, possibly a mixture of the two depending on the delay of the effect in the dependent variable given a change in the

specific explanatory variable (Hyndman & Athanasopoulos, 2013). The SARIMAX forecasting model is obtained from:

$$(1-B)(1-B^s)Y_t = (1-\theta B)(1-\Theta B^s)e_{t+k} + \sum_{i=1}^{b} \beta_i X_{i,t-v+k} \quad \textbf{4.4}$$

To make a forecast, the causal model uses unknown or available values of the explanatory variables based on the lag structure. According to the trends in air passenger numbers, the causal model must explain steady growth in both international and national flows in a time period. Since there is a rapid increase in domestic air flows while the global flows are gradually decreased after 1970, when searching for causes, plausible reasons must be identified both for the shift in trends around 1970 and for the subsequent divergence between national and international travel. The basic structure of the change in trends is self-explanatory. Oman discovered vast quantities of oil, and this domestic prosperity increased the ability of Omanis to travel. At the same time, the economy of the United States is depressed by the oil crisis and thus dampened the international airflows that represent the influx of American tourists. The preceding period of steady growth in air travel, on the other hand, is the expected pattern associated with constant demographic and economic expansion. Any of several variables would represent this effect. In general, many variables might represent the underlying causes been described in the following section.

4.2.5 Explanatory Variables

Several variables might affect the number of monthly passengers. Latent factors essential to incorporate in the model are determinants of indicators of economic development and variations in occurrences of holidays. Since there are no close substitutes to international air travel, cross-price elasticities are not considered. Possible explanatory variables, for which data are available during the investigated time period, are listed below.

4.2.5.1 Price of Crude Oil

It was found that the price of jet fuel is correlated with the price of crude oil (Carter et al., 2006; Morgan, 2001). It is not possible to compare one ticket to another since ticket prices depend on factors such as destination, number of stopovers, etc. However, the fuel price is an operating cost for all companies and thus comparable. The jet fuel expenditure accounted for 29% of the total operating expenses, which had increased by

15% by 2007 (IATA, 2014). The airline companies' operating cost increases with the increase in oil price. The oil price might serve as a proxy for the ticket price and therefore reflected in the ticket price. Data on the variable comes from the World Bank and is the monthly average price of Brent crude oil, which originates from the North Sea and is the most commonly used benchmark for crude oil prices (EIA , 2014). However, the time-consuming refining process of crude oil into jet fuel introduces a delay on ticket price, so its reflectance on the lag structure of the variable is not apparent. Additionally, if the crude oil price rises, ticket prices are adjusted by airlines, and these changes immediately come into force. Airlines set the prices of the tickets far in advance, sometimes up to a year ahead, and forecasts of the oil price are taken into account.

4.2.5.2 Foreign Exchange Rate

The most significant share of the international flights departing from Oman is to Europe, accounting for 42% or 2,445 weekly seats (Oman Airports, 2018). Thus, from the passengers' point of view, if the OMR/GBP/EURO exchange rate depreciates, the cost of holidays decreases. These data are obtained from the Central Bank of Oman and represent the monthly average. Since airline tickets are bought in advance, variations in the exchange rate should reasonably affect air travel demand with a delay. Castellani et al. (2010) found that the exchange rate of GBP/EUR must have a lag of three months to best explain variation in arrivals of tourists from the UK to Sicily.

4.2.5.3 Unemployment Rate

According to basic economic theory, consumption is likely to be postponed when a large portion of the population is lacking high income. Problems may arise since the unemployment rate does not frequently fluctuate much every month, whereas the passenger flow does. The data used as a variable, provided by the World Bank, is based on monthly average values.

4.2.5.4 Holiday Dummy Variables

The occurrence of Eid (A public holiday in Muslim countries) or Easter had a tremendous impact on passenger flow in specific months, which led to an underestimation of the passengers in that month and overestimation of the passengers in other months. Since other holidays occur in particular months, Eid is the only holiday that needs to be accounted for. To control for this, a holiday dummy variable is

constructed which takes on the value one if Eid occurs in April and zero if it occurs in another month, i.e. March. This is done since ARIMA(1,0,2)x(2,0,0) only remembers the values in the previous month and the value at the same month in the previous years.

4.2.6 Random Forest Regression

The random forest model is one of the most effective methods in machine learning for predictive analytics, making it an industrial workhorse for machine learning. The random forest model is an additive model that makes predictions by combining decisions from a sequence of base models (Cheng & Titterington, 1994). Typically, this class of model can be described as:

$$g(x) = f_0(x) + f_1(x) + f_2(x) + \ldots \quad 4.5$$

where g is the final model and it is the sum of simple base models f_i. Each base classifier here represents a simple decision tree. This technique uses the model ensemble method for obtaining better predictive performance using multiple models. In this model, all the base models are constructed independently using a different subsample of the data. The random forest model can capture non-linear interactions between the features and the target.

4.2.7 Construction of Models: Data Source and Description

This Thesis investigates air passenger numbers; to that end, this study analyses 33 airlines from the International Air Transportation Association (IATA, 2017). Insights from the model and accuracies of its predictions are presented here. The data represent eight central regions and consist of 51,983 monthly observations. Monthly time series are more difficult to predict because they have more seasons to deal with than other types of seasonal data such as quarterly time series. The time series data regression model employed the use of GDP, population, the exchange rate, and a dummy variable obtained from the World Bank data (World Bank, 2016). The dependent variable is the number of passengers per month. The forecasts were made for the total passengers departing from Oman to any international destination. Arriving passengers will not be included since the aim is to estimate the number of international air travel passengers. Therefore, the monthly observations in the year 2016 were treated as unknown and used to compare the point predictions with the actual values in order to evaluate forecast accuracy. In 1998 the economic crisis shook the economy of Oman, and to

account for this a “dummy variable” is added to the model. A dummy variable (also known as an “indicator variable” (Gujarati & Porter, 2009)) is one that takes the value 0 or 1 to indicate the absence or presence of some categorical effect that may be expected to shift the outcome. Dummy variables are used as devices to sort data into mutually exclusive categories (Gujarati & Porter, 2009), such as holiday/not holiday or, in econometric time series analysis, to indicate the occurrence of major strikes (Gujarati & Porter, 2009). A dummy variable can thus be thought of as a truth-value represented as a numerical value 0 or 1. When the dummy takes on a value of 0, that variable's coefficient will have no role in influencing the dependent variable; when takes on a value of 1 its coefficient acts to alter the intercept (see Section 3.3.4). In this section, the proposed working algorithm of the study is presented. Firstly, a benchmark model is built. It is a time series forecast, which only takes the previous values of the passenger data into account. This benchmark model was then expanded by exogenous explanatory variables, referred to as a “causal model”. The criteria for model selection used include standardised residuals, ACF of residuals, Q-Q plot, and Ljung-Box statistic. Finally, a random forest model was built in an attempt to obtain more accurate forecasts.

3.3.1. Benchmark Model

The purpose of this data exploration was to find exogenous variables that affect monthly airline passengers departing from Muscat to any international destination between January 1998 and December 2016. Time series models used for forecasting airline passenger numbers, which use only information on the variable to be forecast and makes no attempt to discover the factors, affect its behaviour. Therefore, this study will extrapolate any trend and seasonal patterns and ignore all other information such as competitor activity, marketing initiatives, and changes in economic conditions. In this data exploration, this study focused on finding latent factors that affect passenger flow. The Box-Jenkins method was applied to determine the fit (meaning that the properties of the process do not change over time) of the airline model to monthly Oman data. This approach is a crucial assumption in the Box-Jenkins method, and why it is often necessary to transform the data until this condition is fulfilled (Cryer & Chan, 2008). In Figure 3.3, only observations until December 2015 are included since the remaining ones are saved to evaluate the forecasts. R software was used to draw the timing diagram, as shown in Figure 3.3 below. It can be seen from Figure 3.3 that the

passenger flows include a positive trend and seasonal variation. There is a sharp decline in the number of passengers in 2000/2001 as a result of the terrorist attacks on the World Trade Center. The decline in 2008 and 2014/2015 is due to the financial crisis (Frankel and Saravelos, 2010; Feldkircher, 2014). The data were aggregated across all destination regions for a given month, and it can be seen that the number of passengers was increasing following a relatively stable trend year by year. The following general linear regression models were used in this study.

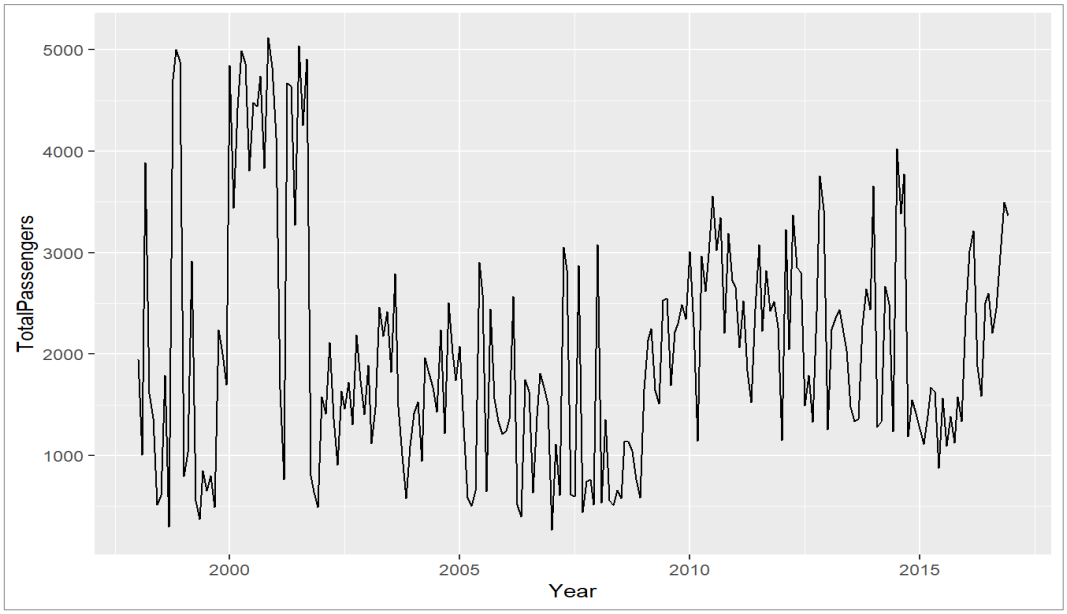

Figure 4-3 Monthly departing passengers (in 1000s).

As expected, there is a clear seasonal pattern (see Figure 3.3) where the summer months are the busiest. Since the pattern is identical regardless of year, this indicates that the behaviour of passengers does not change over time. There is a positive trend, and the variance increases with time. The series follows the expected pattern and, not surprisingly, it resembles an airline passenger process. The point forecasts can be quite off for long-term forecasting, as it seems not to follow the overall pattern. The "overall pattern" in the plot of monthly passengers (Figure 3.3) is the visible seasonal pattern in this data spanning from the beginning of 2000 to mid-2006 (about 90 data points or months). The seasonality observed is not a typical calendrical pattern but is very clearly associated with the business cycle or, more specifically, with market downturns in 2001 and 2008. This study would not be able to model the business cycle, per se, since it has

a much longer time frame. However, it can be "controlled" with dummy variables or by factoring in a macroeconomic index such as quarterly GDP and holidays.

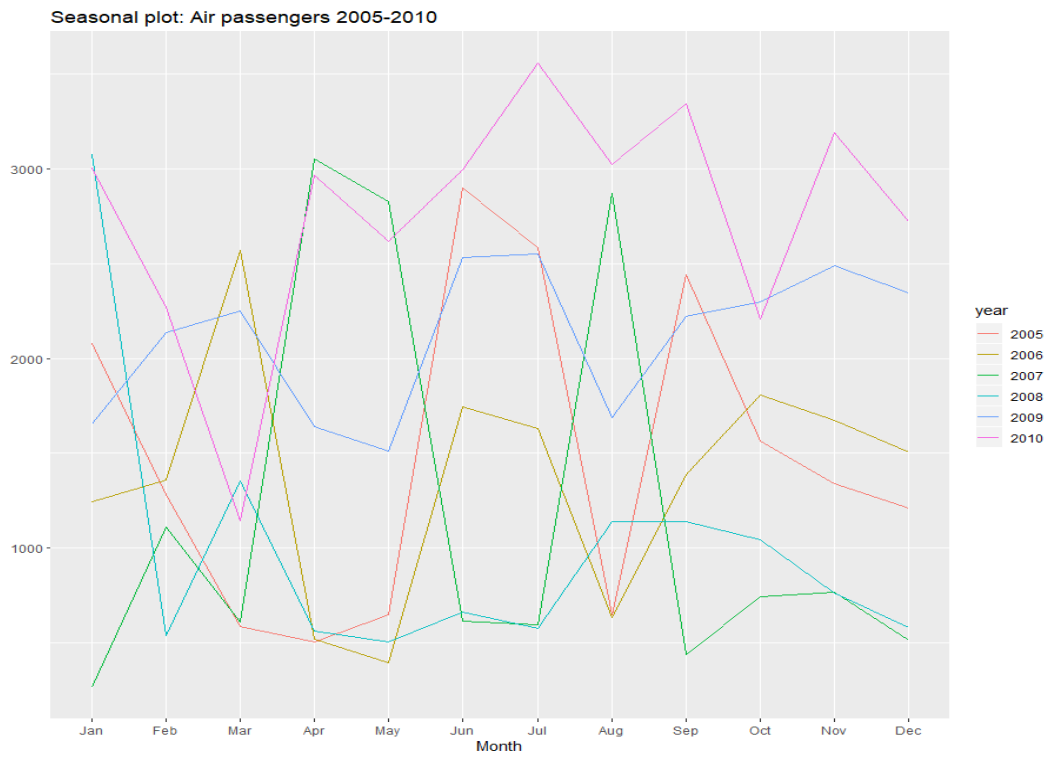

Figure 4-4 Seasonal plot: Air passengers.

Figures 3.4 shows that each year in January is the trough of passenger numbers, in the mid is the peak of passenger flows. If observed closely in Figure 3.4, you can see the long-term seasonal in this data. The change of seasons can make the time series of air passenger numbers change regularly, which is the seasonal periodicity seen in Figure 3.3. There are many reasons for the seasonal changes, such as climate, holidays and so on.

By eliminating seasonal effects in the sequence by seasonal adjustment, the trend of the time series can be seen clearly (Liu et al., 2008). In order to reflect the essential attribute of the real economy more accurately, more methods need to be used to eliminate and adjust the seasonal variation factors in the time series. The original data displayed in Figure 3.4 show a positive trend and increasing seasonal variation. According to the Loess method, stationarity is required for the data log transformation to smooth out the seasonal variation with no loss of information. This trend decomposition method is an STL decomposition method, which is a particular variant

of the time series decomposition method with increased robustness and generality (Cleveland & Cleveland, 1990). At this stage, more understanding is gained about the difference between seasonality, trend, and cycle. The STL decomposition explains it thoroughly since it is a commonly used method to smooth two-dimensional scatter points. It combines the flexibility of the nonlinear regression and simplicity of the traditional linear regression. R's stl() function is applied ("seasonal and trend decomposition using Loess") to the dataset:

```
decomposed <- stl(time.series, s.window="periodic")
plot(decomposed)
```

This decomposes the data as the sum of a seasonal, a trend and a noise/remainder time series:

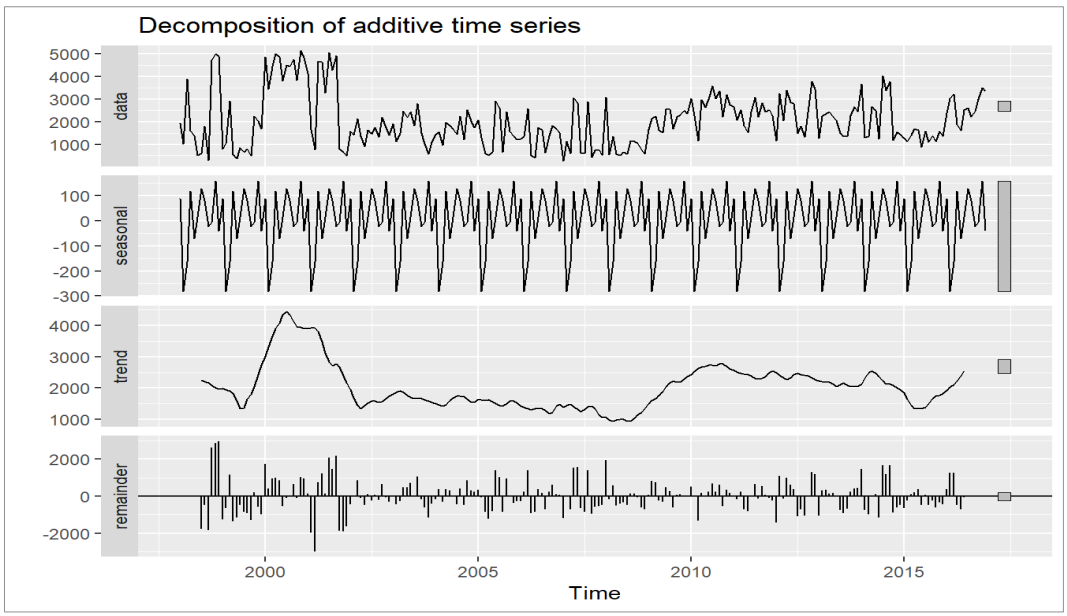

Figure 4-5 The transformed data series.

When estimating a response variable value, firstly a subset of data from the forecast of nearby variables must be obtained (Variables that may or may not have any effects on the prediction), then quadratic regression or linear regression is used on the subset regression. The weighted least squares method used in this study is close to the estimated value of the point of higher weight. The last step is to use a local regression model to estimate the response variable values. The whole fitting curve obtained by the point-by-point operation method has many advantages. For example, it can handle any seasonal data, and the quarter component can be changed over time. It can control the rate of change and has better robustness to outliers. However, the disadvantage of this

method is that it is only applicable to the additive model. The upward curve at the top of Figure 3.5 represents the development trend of passenger flows and the associated time sequence diagram information. As signified earlier, the plot in Figure 3.5 indicates a non-stationary process with a linear trend, but the log transformation made the seasonal variance constant over time. To make the series stationary, first and seasonal differences are applied. After the transformation, the mean is centred on zero. In order to effectively predict the future trend of air traffic, the air passenger numbers curve needed to be fit to make reasonable predictions. First, the seasonal changes in the time series will be removed, and then making the remaining part of the data. In order to forecast the monthly passenger flow, multiple models for forecasting the number of air passengers departing from Oman will be estimated and forecasted in the next sections.

4.2.8 ARIMA Model

A popular and widely used statistical method for time series forecasting is the ARIMA model. An ARIMA model is a class of statistical models for analysing and forecasting time series data. It explicitly caters to a suite of standard structures in time series data, and as such provides a simple yet powerful method for making skilful time series forecasts. It has an advantaged when handling time series datasets as it has a functionality of autoregression which most of the machine learning algorithms lack. ARIMA is usually limited to short seasonal data, e.g., 12 months (monthly) or 3 months (quarterly). Several ARIMA models have been proposed, and the data is split into training and test sets. Only observation data until December 2015 are included since the remaining ones are saved to evaluate the forecasts. This technique called cross-validation. It is a method that involves keeping a particular sample of the dataset which is not trained in the model. The model can be tested on this sample later before finalising it. The steps involved in cross-validation are listed as below:

1. Reserving a sample data set.
2. Training the model by the remaining dataset.
3. Using the reserved sample for validation test. It will help to gauge the effectiveness of the performance of the model. If the result is a positive result on the validation data, the current model can be used continuously.

There are various methods available for performing cross-validation, and the data split into train and test sets. The train set contains the data until 2015 and the test set

contains the rest. The next step is building the ARIMA model. A linear combination of past values of the same variable has been applied in this forecasting study using an autoregression model. The term autoregression indicates that it is a regression of the variable against its previous values (Box & Jenkins, 2015). Therefore, an autoregressive model of order ρ can be written as:

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \varepsilon_t \qquad \textbf{4.6}$$

where ε_t is white noise, and it is typically distributed. It is a regression with lagged values of y_t as predictors and ϕ_k is a coefficient for k^{th} lagged value. It can be referred to as AR (p) model. Moving average models are used if there are any random jumps in the time series data; these jumps are represented in the error that is calculated. A moving average model uses past forecast errors in a very regression-like model (Box & Jenkins, 1976; Box & Jenkins, 2015).

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} \qquad \textbf{4.7}$$

where ε_t is white noise, and it is typically distributed. This is referred to as an MA (q) model, a moving average model of order q.

4.2.9 Finding the Order of AR and MA Models

After a timeseries has been made stationary by transformations, the next step in fitting an ARIMA model is to determine whether MA or AR terms are necessary for correcting any autocorrelation that remains in the differenced series. By using software tools, such as Statgraphics, one can try to combine different terms and select the best combination.

Table 4-5 Consideration of model AR and/or MA model condition.

Model	ACF	Ljung-Box
AR(p)	Spikes decays towards zero	Spikes cut off at zero
MA(q)	Spikes cut off at zero	Spikes decays towards zero
ARMA(p, q)	Spikes decays towards zero	Spikes decays towards zero

Also, there is another systematic way to complete this: by checking the autocorrelation function (ACF) or the Ljung-Box plots of the differenced series, the required numbers of AR or MA terms can be identified temporarily. The value of *p* will be equal to the value of lag where the coefficient line is touching the upper boundary of the confidence interval in the Ljung-Box plot. The value of *q* will be equal to the value flag where the coefficient line is touching the upper boundary of the confidence interval in the ACF plot. The value of *(d)*, which is the differencing value will be equal to the number of times the data is differenced to convert into stationary. In the plots, the dotted lines on either side of 0 represent the confidence interval. These can be used to find the values of *p* and *q*, i.e., the order of AR and MA. In order to model monthly international air travel passengers and other seasonal time series, a two-coefficient time series model of factored form, which is also known as the airline model, a seasonal ARIMA(1,0,2)x(2,0,0), has been developed (Box & Jenkins, 1976). The model SARIMA(pax.ts),1,0,2,2,0,0,12) is often used to forecast passenger flow since it is both accurate and straightforward to estimate. The airline model will serve as a suitable benchmark model that is to be improved. Firstly, the airline model fits the Oman data, as illustrated in Figure 3.6.

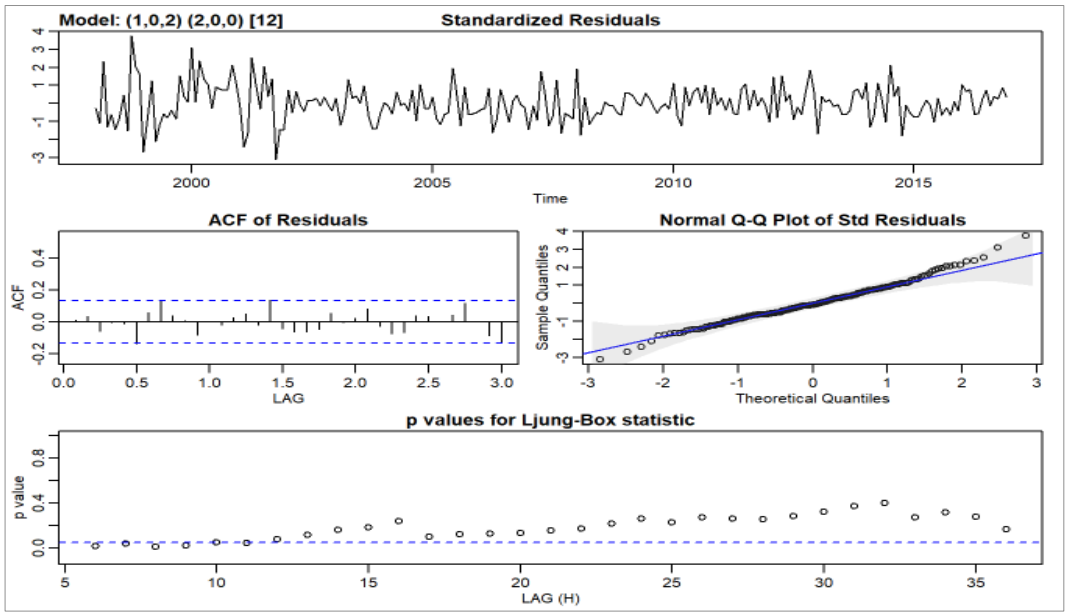

Figure 4-6 Benchmark model.

The fitting results of monthly airline passenger numbers obtained by the ARIMA model shown in Figure 3.6 have reached an ideal fitting effect. This method can be used in the

future for forecasting the change of passenger flows. High prediction accuracy of the model can be seen in Figure 3.6. The mean of the residuals is close to zero, and the residual series are not significantly correlated. Apart from the one outlier, the time plot of the residuals indicates that the variation of the residuals stays almost the same as the historical data and, therefore, the residual variance can be treated as a constant. The variance in the time series appears constant in the above Standardised Residuals plot, except in years 2000/2001, 2008 and 2014/2015 and the several months of the following years due to the differencing. The magnitude of these residuals is larger than 4, which is unusual in a standard normal distribution. This often occurs due to omitted variable bias, i.e. shocks to the series as a result of certain factors not being controlled. The right tail of the histogram seems a little bit longer, which indicates that the residuals may not be normal even when we ignore the outlier. Therefore, this method forecasted a good result, but there might be some inaccuracies of prediction intervals. Figure 3.6 shows the ACF and Ljung-Box plots. The regression coefficient for the ARIMA (1,0,2)x(2,0,0) model is illustrated in Table 3.6 and it shows that the estimated intercept of the model was 2047.2502.

Table 4-6 The regression coefficient for ARIMA (1,0,2)x(2,0,0).

Coefficients						
Auto regression (ar1)	Moving average (ma1)	Moving average (ma2)	Sarima (sar1)	Sarima (sar2)	Intercept	Regression coefficients (xreg)
0.8903	-0.4517	-0.1380	-0.0897	-0.1830	2047.2502	8.580

4.2.10 Causal Model

The causal model is defined as an extension of the airline model, to account for omitted factors with a possible effect on passenger flow. Based on the available data, the aim to find exogenous variables that affect monthly air passengers departing from Oman (Muscat International Airport) to any international destination. The factors explored include Oman's jet fuel price, Oman's real interest rates, Oman's unemployment size, Oman's real GDP, Oman's real GDP per capita, Oman's population size, average base fare from Oman to any international destination, and Oman's public holidays. The approach to finding the omitted factors is to compare the ARIMA model using one variable to the benchmark model using penalty function statistics, such as Akaike

Information Criterion [AIC] (Akaike, 1974) or Bayesian Information Criterion [BIC] (Schwarz, 1978) as a reference. AIC/BIC is used for assisting time series analysts to reconcile the need to minimise errors with the conflicting desire for model parsimony.

These statistics are in the format of minimising the sum of the residual sum of squares plus a "penalty" term, which incorporates the estimated parameter coefficient numbers into the factor in the model parsimony. If the AIC and BIC are similar, it is prudent to go for the model that uses fewer parameters. If there is an accurate ARMA time series model, BIC and AIC are the best candidates for it. The BIC is strongly consistent while AIC will usually result in an over-parameterised model, with too many MA and AR terms (Mills, 1993). Other researchers are also in favour of the BIC over the AIC (Gómez & Maravall, 1997). Thus, in practice, using the objective model selection criteria involves estimating a range of models and the one with the lowest information criterion is selected. This selection will bring many difficulties. First, using the penalty function criterion is computationally expensive. So, it is imperative to choose the maximum order to avoid high computational requirements. However, there is a lack of previous information, which can assist in selecting the maximum order of the ARIMA model.

One good way for determining the maximum order of the model is to select one less maximum for the regular seasonal span terms (for example, three for four sets of data) and one for the seasonal term. Second, the different objective model selection criteria can suggest different models. That is, the ranking order based on the BIC will usually not be the same as under the AIC. Comparisons of the top seven ranking models under the BIC and AIC for the monthly passengers departing from Muscat to any international destination during the period January 1998 to December 2016 are illustrated in Figures 3.7 – 3.20.

4.2.10.1 The Effect of Monthly Average Base Fare on Passenger Flow

A time series graph of the Average Base Fare forecasted values is displayed in Figure 3.7 and 3.8.

Figure 4-7 Monthly average base fare effect on passenger flow.

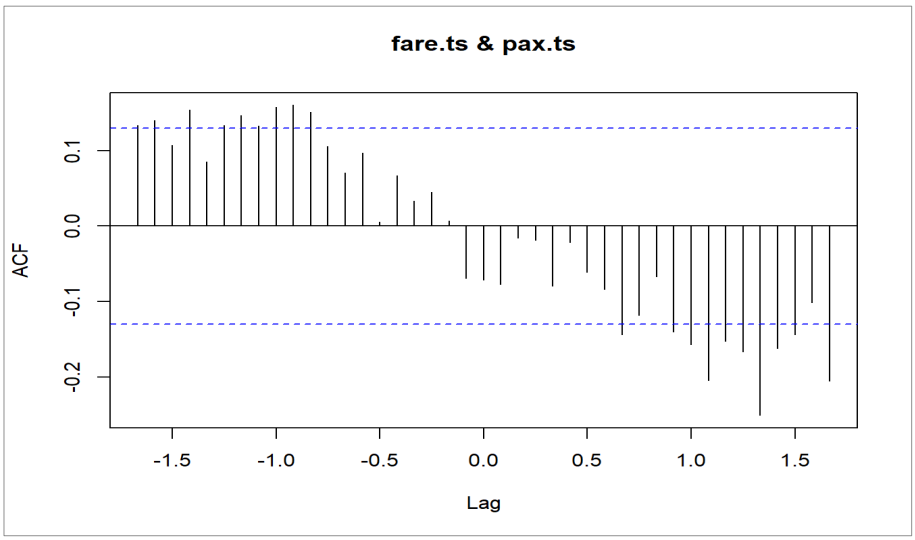

Figure 4-8 Model1: Arima((pax.ts.)=fare.ts).

As can be seen from Figures 3.7 and 3.8, Average Airfare has some effect on passenger flow. As price elasticity has a negative impact on airfare, an increased airline congestion cost will cause airfare impact to become more negative and decrease the air passenger demand. Airfare represents the commercial aeroplane fare for transportation. The impact of airfare on air passenger demand is determined by utilising the concept of price elasticity of demand. Price elasticity of demand is the percentage change in demand when the average airfare changes by 1% (Sabatelli, 2016). The time elasticity of demand can be defined as the percentage change of total travel demand, and it occurs at around 1% of travel time. In this study, the airfare impact is defined as a change in demand from a percentage change in average travel cost times to price elasticity. The

exchange rate has some effect on the airfare prices as well. It had the expected negative coefficient since it is measured in OMR/EUR/USD. The effect of an increase in the exchange rate goes two ways; when it becomes cheaper to travel to countries using euros or dollars as a currency, Omanis are more likely to travel to such countries. The overall vacation cost decreases, as hotel stays become cheaper. Meanwhile, it becomes more expensive for tourists to visit Oman. Since tourists are included in the data when they leave Oman, the negative coefficient indicates that the effect is dominant for Omanis travelling abroad.

4.2.10.2 The Effect of Jet Fuel Price Effect on Passenger Flow

The exchange rate has some effect also on jet fuel prices. Jet fuel price in Oman has some effect on passenger flow (see Figures 3.9 and 3.10).

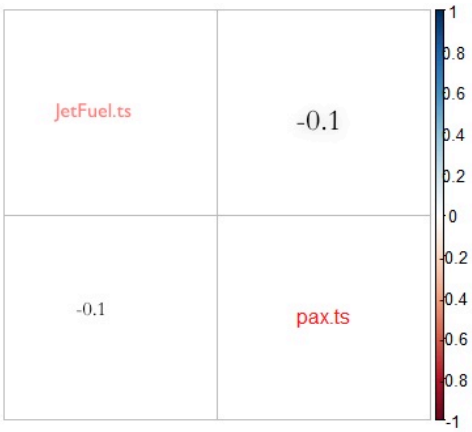

Figure 4-9 Jet fuel price effect on passenger flow.

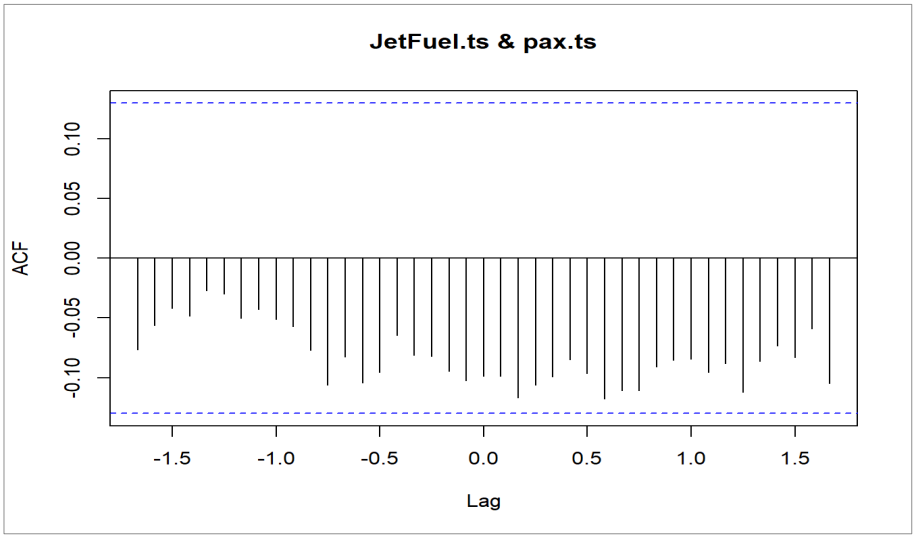

Figure 4-10 Model2: Arima((pax.ts.)=JetFuel.ts).

Since the operational costs of airlines increase with the increase of the oil price, the oil price can serve as a proxy for the ticket price. It was expected that the adverse effects of this relationship are reflected in the ticket prices; however, the positive coefficient we have obtained proved that it is not correct. It is reported that including oil price as an explanatory variable has led to the same problem, with the possible explanation that the oil price serves as an indicator for the overall state of the global market (Yu et al., 2009). When economies grow, demand for oil increases, which puts upward pressure on the price. In the estimate, the model in this study, the increases in oil price happened in global periods of economic prosperity. In this period, the passenger numbers also increased simultaneously with the oil price. Historical oil price data showed a peak in August 2008 and a decline in the following time window. Until this time, both monthly passenger numbers and oil price increased steadily. Both passenger numbers and oil price decreased dramatically after the financial crisis, and showed an even stronger positive correlation, although the decline in passengers was a result of other factors. The effect of oil price could be different if a pre- and post-2008 model was to be constructed. In general, the causal forecast lies between the naïve forecast and the actual number of passengers, indicating a forecast improvement. The causal model is used for forecasting the passenger flows 12 months ahead because the oil price is non-lagged; it requires predictions of the explanatory variables for each month.

4.2.10.3 The Effect of Oman Population Size on Passenger Flow

Air travel demand is affected by several factors, such as population, which relates directly to fundamental demographic and macroeconomic factors (Seraj et al., 2001). Population growth can generate more travel demand. In our model, population must be examined since it acts as a predicted variable because future populations are not guaranteed. Based on experience, the population models are notoriously unreliable. According to national figures and short period aggregates, the predictions of population models are reasonably accurate, but they are inaccurate in most countries when considering over twenty to thirty years' estimation. It requires significant adjustments over short periods of less than ten years due to the migration, fertility rate and death rate of the population. The disaggregate level, which can be applied to city or metropolitan levels or even the national reliability level, has not been attained in this study. The disaggregate population of urban areas in developing countries must be divided into a non-economic population who are subsumed by urban areas only because

they have no position in the rural economic structure and an economic population who has wage-earning or engaged in trade. The effects on air transport demand by the non-economic population are small, so the essential population factor is influenced by the economic growth factor. The higher the growth, the greater is the proportion of the population to enter into the economic sphere of activity. Figures 3.11 and 3.12 illustrates Oman's population size, which has little effect on passenger flow.

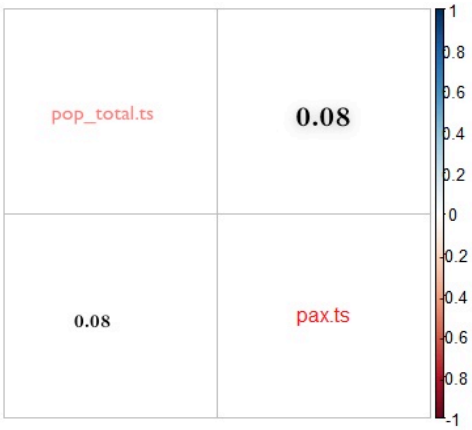

Figure 4-11 Oman population size effect on passenger flow.

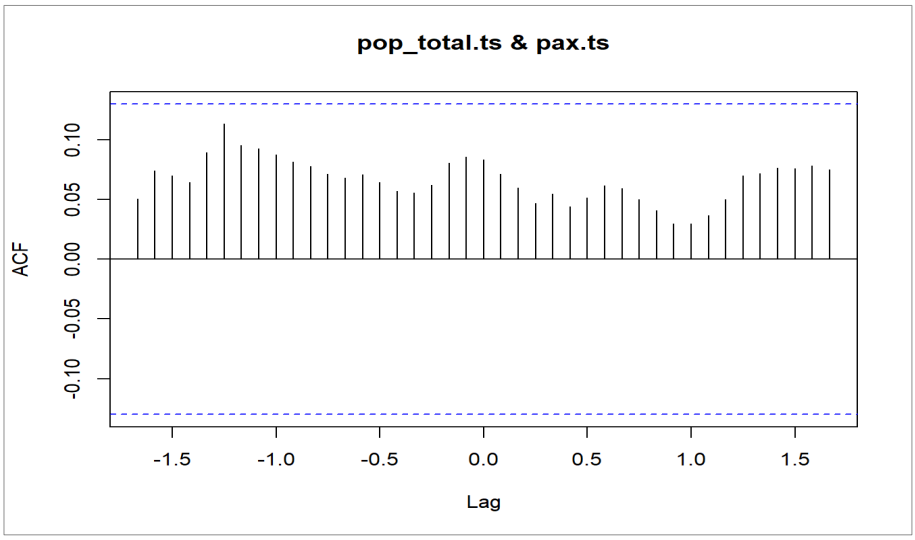

Figure 4-12 Model3: Arima((pax.ts.)=pop_total.ts).

4.2.10.4 The Effect of Oman's Real GDP on Passenger Flow

The air passenger numbers are influenced directly by the quality of people's life and income. One of the critical impacts on air traffic demand is attributed to GDP because it increases profit by generating an increase in travel (Pupavac, 2009). The research result of EU data indicated that there is a lag in growth rate in air passenger transport demand

compared with the GDP growth rate (Grosche et al., 2007). For example, passenger traffic increased by only 9.33% between 2000 and 2007, while the European GDP has increased by 16.61% in this period. Oman's real GDP is presented in Figures 3.13, and it shows some effect on passenger flow. Data from Figures 3.13 and 3.14 show a correlation between the actual GDP growth rates and realised growth rates of passenger numbers. This correlation is rather strong, as the number of passengers is growing at a higher rate when there is a growth in GDP; vice versa, when the rate of GDP growth is negative, the total number of air passengers shows greater negative rate. The regression model in this study was built on explanatory or independent variables like GDP. The disaggregated approach would probably increase the accuracy of the forecast and could be of interest to airline companies to forecast specific routes. It also can be used for computing aggregated forecasts of interest to Oman's transport agency and other stakeholders. However, such an approach is often time consuming and reliant on extensive data and is beyond the ambitions of this Thesis. The GDP on a per capita basis is also an indicator of how well off the population is. Due to air travel still being considered a luxury in developing countries, per capita GDP is an indicator of people's abilities to fly.

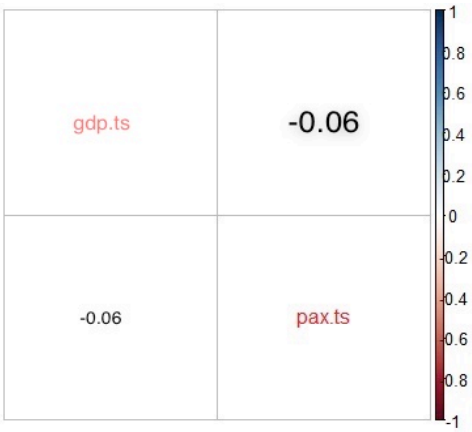

Figure 4-13 Oman's GDP effect on passenger flow.

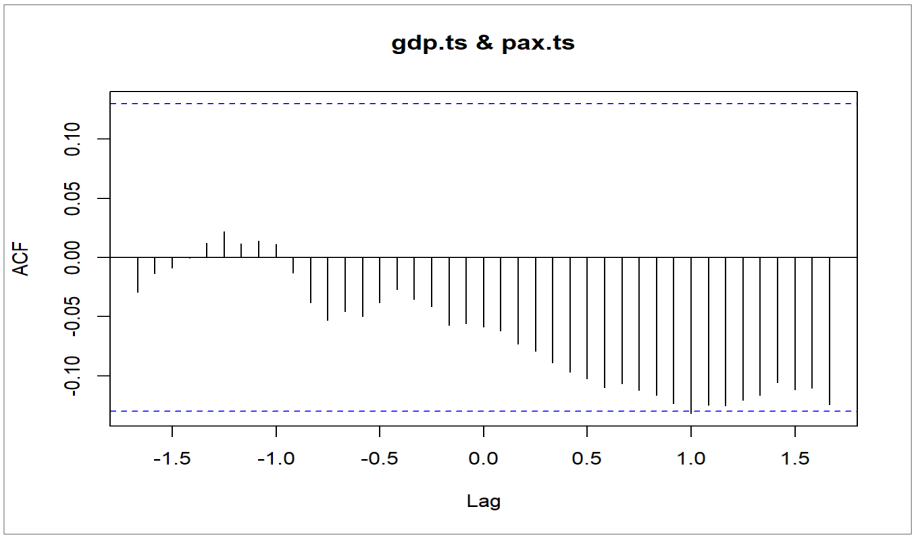

Figure 4-14 Model4: Arima((pax.ts.)=gdp.ts).

4.2.10.5 The Effect of Oman's Unemployment Size on Passenger Flow

The unemployment rate is also a determinant factor of air travel demand (Clark et al., 2009; Wensveen, 2011). It is a standard independent variable in regressions used to forecast air travel demand, even when income effects are also included (Carson et al., 2011). Unemployment in the airline industry will affect the quality of the services because new people will not be hired due to financial instability in the industry. To provide better services to its customers, it is necessary for the company to have skilled employees. However, the unemployment position will cause poor service and low demand in the industry, and it will force the industry towards a recession period. Figures 3.15 and 3.16 indicates that Oman’s unemployment size has some effect on passenger flow.

Figure 4-15 Oman's unemployment size effect on passenger flow.

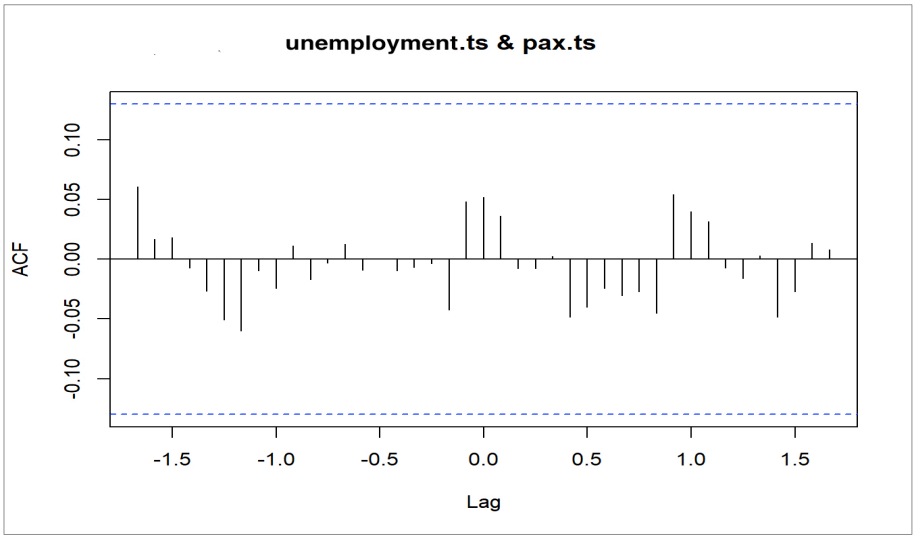

Figure 4-16 Model5: Arima((pax.ts.)=unemployment.ts).

4.2.10.6 The Effect of Oman's Real Interest on Passenger Flow

There are also a variety of other factors that might affect air travel demand, i.e. holidays and real interest rates (Gesell, 1993). Interest rates affect the balance between expenditure and saving (Cook, 2007). High-interest rates restrict economic activity and cause a dampening effect on airline traffic (Wensveen, 2011). Exchange rates and interest rates affect the growth of air passenger demand firmly but in the short-term (Omanair, 2011). Exchange rates can be strong determinants of air transport demand but mainly on international routes. If the currency of a country is stronger than others, then its citizens can obtain more foreign currency with the same amount of money than before. It will provide relatively lower prices and increase the desire to go to a country, which has a lower price. For example, it was reported that positive exchange rate coefficients are more attractive for UK citizens (Dargay & Hanly, 2001). It means UK citizens could get more local currency with the same number of pounds and would have more desire to travel to such countries.

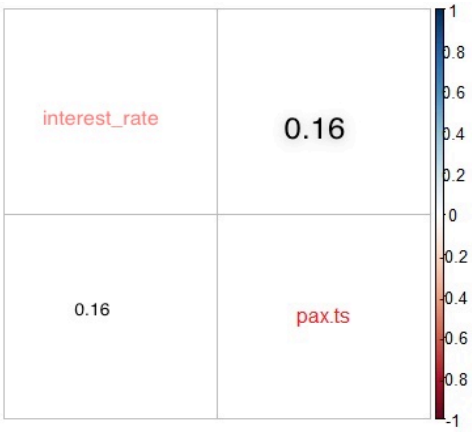

Figure 4-17 Oman's real interest effect on passenger flow.

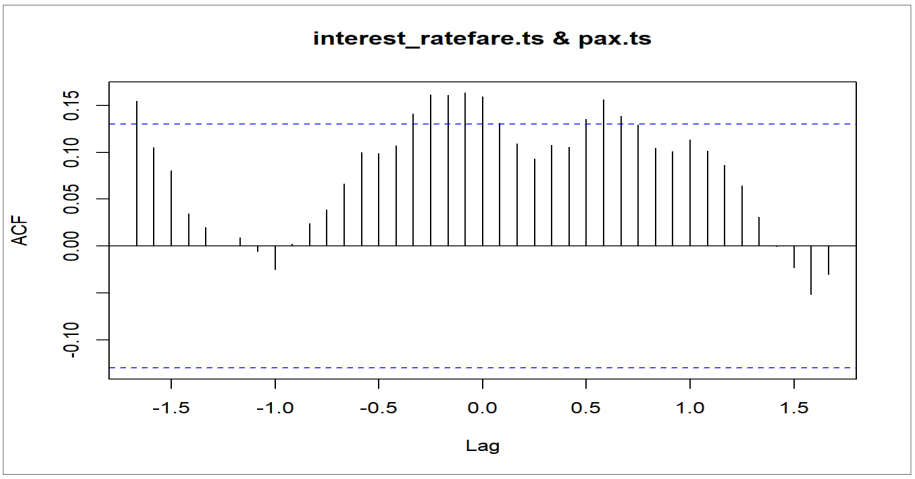

Figure 4-18 Model6: Arima((pax.ts.)=interestrate.ts).

4.2.10.7 The Effect of Oman's Monthly Public Holiday on Passenger Flow

Oman's monthly public holiday (Eid) in Figures 3.19 and 3.20 show some effect on passenger flow. The variable controlling for Eid also has the expected positive sign. If a year when Eid occurred in March, for example, is followed by a year when Eid occurred in April, this yields a positive effect on the number of passengers in April. When scrutinising the original data for monthly passengers, March is barely affected by the occurrence of Eid, while April is. A possible explanation might be that April is a warmer month than March, which is why holiday planners are more likely to adjust vacation plans and travel somewhere cooler, depending on which month Eid occurs in. A weeks' leisure in April makes people more likely to travel abroad than a week's leisure in March. During the period 1998 to 2015, only four Eids have occurred in March. The model captures the Eid effect but underestimates it. When making the 2016 forecast, all observations included in the model, hence another rare Eid observation. This results in an Eid effect significant on the 1% level, compared to the forecast for

2015, where the effect was significant on the 10% level. Since future Eid dates are known, this is a simple way to improve forecasts. The zero coefficients mean there is no linear relationship between passenger flow and the number of holidays.

Figure 4-19 Oman's monthly public holiday effect on passenger flow.

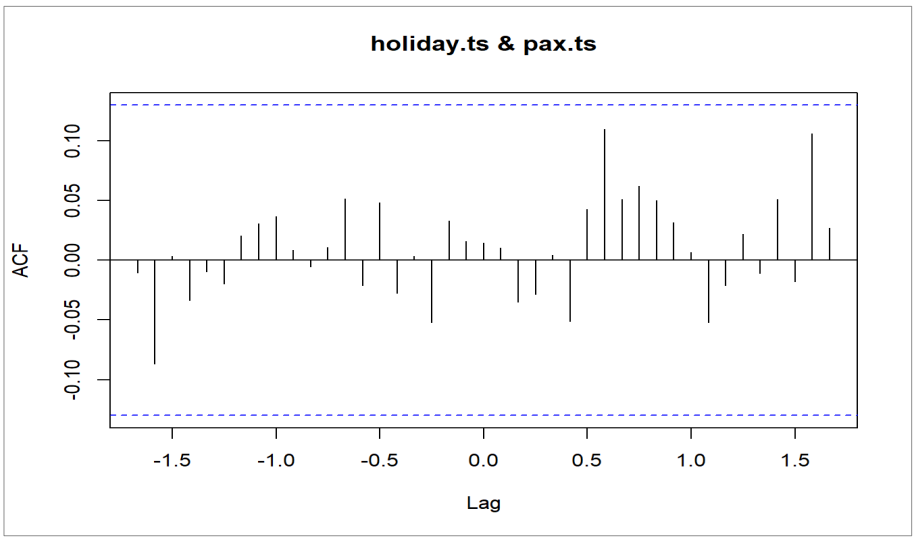

Figure 4-20 Model7: Arima((pax.ts.)=holiday.ts).

Point predictions of the causal model follow the actual values to quite a reasonable extent. The most significant deviations seem to occur during the summer months and in April. The model underestimates the number of passengers in the summer of 2014; hence, people appear to have travelled more during the summer of 2014 compared with previous summer months. The model continues to underestimate passenger numbers in April, even though Eid is accounted for. The model captures the effect of the 2014 Eid

holidays, which will occur on April 2015. The point prediction for this month is adjusted by the forecast of April 2015, in comparison to the 2014 forecast. The other point predictions lie within the trend of the series. An overall increase in passengers is predicted for August 2015 to November 2015, which might be due to increased weekend travelling, which is more popular during the fall because of more clement weather and public holiday periods. The point prediction for March is less than the 2014 forecast, but there is no way of determining the accuracy.

4.2.11 Random Forest Model

Random forest algorithm is a supervised classification and regression algorithm. As can be seen from the name, this algorithm randomly creates a forest with several trees. In general, when there are more trees in the forest, the forest looks more robust. Similarly, in the random forest classifier, the higher the number of trees in the forest, the more accurate results are produced. Random forest is an ensemble of decision trees; it randomly selects a set of parameters and creates a decision tree for each set of chosen parameters. The variables of the random forest model include the average base fare, numbers of holidays, MA(3) of total passengers, MA(1) of total passengers, and first difference of total passengers. Since the monthly passengers in the year 2016 are treated as unknown, unlike the ARIMA model, the random forest model can only predict the monthly passengers one by one: predict the passengers in January 2016, then take the prediction in January as known value and predict the passengers in February, and so on.

4.3 Performance Evaluation: Evaluation Metrics

Accurate forecasts can only be determined by considering how well a model performs on new data that were not used when fitting the model. Looking at how well a model fits the historical data is not effective. It is common to use a portion of the available data for fitting when choosing a model and use the rest of the data for testing the model. Then the testing data can be used to measure how well the model is likely to forecast one new data. This study uses multiple metrics to measure forecast accuracy. Let y_i denote the observation, and $\hat{y}_i$ denote a forecast of y_i. The forecast error can be simplified as $e_i = y_i - \hat{y}_i$, which is on the same scale as the data. Accuracy measures that are based on e_i are, therefore, scale-dependent and cannot be used to make comparisons between series that are on different scales.

MAE and RMSE are the two most commonly used scale-dependent measures based on absolute errors or squared errors (Armstrong, 1978; Hyndman & Koehler, 2006):

Mean absolute error: MAE = $mean(|e_i|)$, **4.8**

Root mean squared error: RMSE = $\sqrt{mean(e_i^2)}$ **4.9**

When comparing forecast methods by using a single data set, MAE is more favourable because it is easier for understanding and computing. The percentage error is given by $p_i = 100\, e_i/y_i$. It has the advantage of being scale-independent, so it is frequently used for comparing forecast performance between different data sets (Armstrong, 1978; Hyndman & Koehler, 2006).The most commonly used measures are:

Mean percentage error: MPE = $mean(p_i)$, **4.10**

Mean absolute percentage error: MAPE = $mean(|p_i|)$. **4.11**

The disadvantage of measures based on percentage errors is infinite or undefined if $y_i = 0$ for any i in the period of interest and having extreme values when any y_i is close to zero. Another problem with percentage errors is that it assumes a meaningful zero more often.

Theil's U statistic is a relatively accurate measure that compares the forecasted results with a naive forecast. It also squares the deviations to give more weight to significant errors and to exaggerate errors, which will help to eliminate significant errors. For cross-sectional data (Theil, 1958), a scaled error can be defined as:

$$q_j = \frac{e_j}{\frac{1}{N}\sum_{i=1}^{N}|y_i - \overline{y}|} \quad \textbf{4.12}$$

In this case, it is compared with the mean forecast (Armstrong, 1978; Hyndman & Koehler, 2006). The mean absolute scaled error can be simplified to:

MASE = mean($|q_j|$). **4.13**

When forecasting data with distinct seasonal patterns and an overall increasing trend, the causal models predict reasonably well. The RMSE gives a root of the squared average deviation of 54,515 passengers per month, or 9.5% in mean absolute percentage terms, compared with the benchmark model of 59,069 or 10.2%. Both models' point predictions are plotted with the actual series in Figure 3.21, where it can be seen that both forecasts underestimate the number of passengers in April, June and July while overestimating them in other months. In general, the causal forecast lies between the benchmark forecast and the actual number of passengers, indicating a forecast improvement. The following table lists the evaluation metrics of the models.

Table 4-7 The evaluation metrics of the model.

Model	RMSE	MAE	MAPE	MASE
Monthly passengers time series (benchmark model)	59,069.36	49,876.47	10.22179	1.039352
Monthly passengers + average base fare time series	54,515.8	46,412.12	9.479257	0.96716
Monthly passengers + jet fuel price time series	91,178.87	75,934.25	14.5154	1.582358
Monthly passengers + GDP time series	2,807,080	2,804,603	579.501	58.44379
Monthly passengers + GDP per capita time series	60,459.4	52,647.66	11.20872	1.0971
Monthly passengers + unemployment size time series	62,835.92	51,316.97	10.22283	1.06937
Monthly passengers + interest rates time series	59,752.49	49,829.1	10.12643	1.038365
Monthly passengers + holidays time series	58,877.53	49,867.34	10.21048	1.039162
Random forest	34,176.74	25,447.81	4.892305	0.5302949

As can be seen from the evaluation metrics table, it was found that the ARIMA model of monthly passengers with average base fare and random forest model performs better than the benchmark model. Figure 3.21 shows the actual monthly passengers in 2016 and the predictions of three models: the benchmark model, the benchmark + average base fare model and the random forest model.

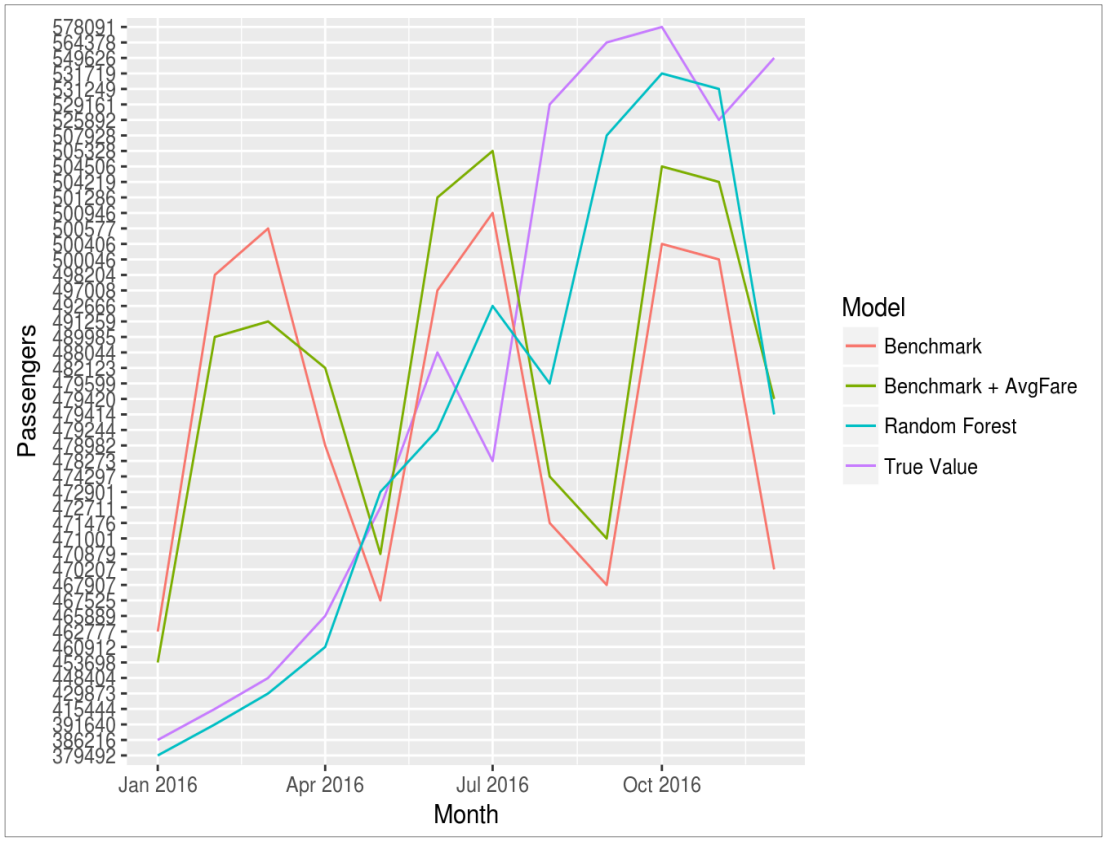

Figure 4-21 The actual monthly passengers in 2016 and the predictions of the benchmark model, benchmark + average base fare model and random forest model.

4.4 Discussion

This chapter has discussed the available data used for analysis and hypothesis testing throughout this study. The data source and some specific information, such as the structure of data tables and the number of records, are mentioned clearly to explain the reason for the proposed hypothesis in Chapter 1. The proposed cleaning algorithm of the dataset was presented. This algorithm proposes to collect data and pre-process them using R. A model validation process is required to help build confidence in the model. The objective of this Thesis is to obtain a deeper understanding of the model.

For this purpose, historical data during the time horizon of simulation of the base model (1998-2015) are needed. A model will be valid when the error rate is less than 5% (see Table 3.21) according to the previous research by (Barlas, 1994). The comparison between model and data of air passenger demand, Oman's unemployment size, Oman's real GDP, Oman's real GDP per capita, Oman's population size, and average base fare from Oman (Muscat) to any international destination are given respectively.

The forecast of departing passengers is improved by taking into account changes in oil price and the OMR/EUR/USD exchange rate while controlling for the occurrence of public holidays. The actual values of the explanatory variables were used to determine whether it is advantageous to use them as input. This should be considered when evaluating the forecast since it is impossible in out-of-sample forecasts. Based on the result of the ARIMA model, which is used for predicting the trend of passenger flows over the next 12 months, it was concluded that this model works better in our case. The following conclusions are made based on the result of the ARIMA model for passenger flow prediction:

1. Based on Oman's comprehensive national strength and people's higher living standards, air passenger volume will keep a steady increase and the air passenger market will become more significant.
2. There is an inevitable seasonal fluctuation in airline passenger numbers, the main reasons for which include public holidays, oil price, and other factors mentioned in this chapter. The summertime, i.e. August, is the peak in air passenger numbers.

Only by accurately predicting the trend of air passenger traffics can airlines reasonably adjust the workforce and material resources in the passenger transport market. These adjustments will improve the operational capacity and service quality of airlines.

Chapter 4: Optimising the Deep Learning Model for Neural Network Topology to Improve Classification Accuracy

5.1 Introduction

Machine learning is the science of instructing a computer to perform operations without telling it exactly how to perform the tasks involved in such an operation (Burkov, 2019). As such, a model is fed with large quantities of data as well as the anticipated responses. From the data input, the model comes up with a mathematical sequence of how to label the input and bind it with the anticipated response. This is done in such a way that it is possible for the model to generate the correct response prediction as well as input that has not yet been seen (Burkov, 2019). This form of machine learning that utilises labels is known as supervised learning. In the current world, machine learning is used to analyse enormous volumes of data, the complexity of which increases along with any increase in activities and specialisations.

In the last decade, the advent of deep learning technology facilitated the creation of more effective learning models, which have been applied in various contexts such as health and education. When optimising such models, they are calibrated with large datasets which consequently enable future predictions (Bengio, 2013). There are various optimisation techniques that are employed in deep learning to enhance performance. One commonly used technique is principle components analysis (PCA), which is mostly applied to enhance the calibration of deep neural networks. The main challenge remains of how to teach deep learning neural networks with large sets of data (Hornik, 1990). While trying to solve this problem, researchers have come up with deep learning methods capable of teaching deep network variants such as the conviction network presented by Fukushima (Hochreiter, 1991). This development played a critical role in re-establishing the enthusiasm of other researchers toward conducting more research on deep neural networks. It is, however, a common phenomenon for researchers investigating neural networks with numerous layers to encounter the

problem of some transfer functions approaching zero. It was Hochreiter who came up with a solution to this issue of disappearing gradient when conducting his PhD (Bishop, 1991)

Before deep learning came into use, the majority of neural networks employed simple quadratic error performance measures on the output layer (De Boer et al., 2005). The invention of the cross-entropy error feature facilitated the achievement of better outcomes than the quadratic function previously in use (Michalski et al., 2014). This is mainly because it tackled the vanishing gradient issue by allowing errors to change weight even when a neuron's gradient saturates (their results are close to zero). Additionally, it offers a more irregular form of error representation as opposed to the quadratic error performed for classification neural networks. As such, the analysis used in this study will utilise the cross-entropy error measure for classification as well as the more commonly used root mean square error (RMSE) measure for regression.

Random weights are the starting points for neural networks (Rzempoluck, 2012). Sampling of these random weights is often done within ranges such as (-1, 1). At times, range initialisation can create a set of weights that proves difficult for back-propagation training (for example, being caught in a local minima). To counter this challenge, Bastien et al. (2012) unveiled the rectified linear unit (ReLU) transfer function. Contrary to sigmoidal transfer functions, ReLU transfer functions achieve better training results for deep neural networks. This is because the hidden layers of neural networks make use of the unique ReLU transfer feature. As stated in past studies, it is essential to identify the type of transfer function that is to be used for each layer in a deep neural network (Glorot & Bengio, 2010; Bengio, 2013). Most deep neural networks make use of a linear transfer function for regression in combination with softmax transfer function for classification in their output layers. No transfer function is required for input layers.

Overfitting has been identified as a common hindrance for neural networks. Overfitting is said to occur when a trained neural network starts mastering the outliers in a dataset (Bastien et al., 2012). Due to the availability of many variables, as opposed to the total number of training examples, the overfitting problem is countered through feature selection and regularisation. To begin with, extraction of data from the input data is conducted. This is a way of creating a new representation specifically for the current

task (Grus, 2015). For the task to be achieved, a classification system is then learned on top of the extracted features. Once training has been undertaken, it should be possible for the system to be employed on data that have not yet been seen during the training stage and then accurately predict the response, giving rise to the class labels (Kelleher et al., 2015). Until recently, the features extracted from the input were mostly handcrafted features. This means that the features are designed specifically for the input and task at hand. Generally, they are not only tied to the type of data, for instance prediction of airline passenger numbers, but to a specific subset, such as the prediction of Oman airline passenger numbers using Muscat airport data. In most instances, the majority of these features are not robust to change.

Another way to extract features from data is to develop a feature extractor using machine learning. Rather than building a system to classify some numbers, a learning system is built to extract features from the input (Han et al., 2016). In such a case where numbers are being used, the network is able to learn higher-level features directly from the input figures. This approach is deemed superior to the use of handcrafted features, for various reasons. First, a model that has been trained on each dataset can be adapted to many types of input, whereas handcrafted features may require hand-tuning for each dataset (Han et al., 2016). Additionally, this method does not require expert knowledge of the numbers being analysed.

This Thesis is focused on in-depth analysis of techniques used to extract features from data. As outlined in the next section, attention is given to the extraction of features from numbers, in this case, airline passenger figures. All machine-learning workflows depend on feature engineering and feature selection. These two actions are considered by various data science and machine learning communities to be equal. Although they share some overlaps, they differ by having diverse objectives (Girolami, 2011). Knowing the distinct goals for each can significantly improve workflows and processes in data science. The overfitting problem emanating from the small number of variables in comparison to the total number of training samples will be countered through feature selection and regularisation techniques.

For regression purposes, evaluation criteria for effective models are defined. Data is partitioned into training, validation, and testing sets to ensure robust statistics (Dangeti, 2017). In order to address the prediction problem, influential variables will be listed to

facilitate better understanding of the factors that underlie the predictions. All results and executable codes from experiments and detailed analysis will be included in the R notebook file as well as in Python. As such, the purpose of this chapter is to therefore determine the model accountable for exogenous variables and, by doing so, consequently improve a regular time series forecast of monthly airline passengers departing from Oman to any international destination. It is also in this chapter that a description of feature engineering will be outlined as well as its significance, problems solvable through it, engineering of such problems and how to go deep into the process as well. First, features and feature matrices will be explained and then an outline of the differences between feature engineering and feature selection will be provided.

5.2 Brief Overview of Features

Machine learning is the process of generalising from a set of training data to predict or infer an output. Normally, a collection of numeric examples is taken as input by a machine learning algorithm in a process commonly referred to as 'training' or 'fitting'. Once the process is complete, the final result is a model that can be used by the algorithms to predict outputs in the future (Balasubramanian et al., 2014). A two-dimensional feature matrix is created by stacking the numeric examples on top of each other. The rows in the matrix represent examples while columns represent features. Table 4.1 below illustrates.

Table 5-1 Feature matrix diagram. Each row represents an example, and each column represents a feature describing that example.

	Features			
Examples	**Destination.Airport.Name**	**Destination.City.Name**	**Destination.Region.Name**	**….**
	71	69	83	
	153	148	46	
	159	153	33	
	….	….	….	….

It is important to know the roles of all features in order to gain a clear understanding of machine learning. The main role of the features is to transform input data from its current form into a format that is suitable and readable by the algorithms. At times, if

the input contains single numeric values for each value – for example, the pound amount in a credit card transaction – transformation may not be necessary. For deep learning in particular, features are usually simple since the algorithms generate their own internal transformations (Rahman, 2019). This approach requires large amounts of data that on the other side tends to be less interpretable. These trade-offs however are worthwhile when using image processing or natural language processing. In some cases, feature engineering is essential to convert data in formats suitable for machine learning. Both interpretability and performance are largely determined by the features chosen.

It is worth noting that without feature engineering, the accurate machine learning systems that are currently deployed by major companies today would not be existent. Feature engineering is therefore the process of using domain knowledge to extract new variables from raw data that make machine learning algorithms work (Zheng & Casari, 2018). In a typical machine learning implementation, quantity prediction is achieved using information obtained from the data sources of the company in question. Such sources could be various databases as well as log files. Generally, they contain many tables connected by certain columns. An ecommerce airline website's database could, for example, have a table called 'members' containing a single row for every client that visited the site. It could also have a table called 'interactions' containing a row for each interaction (click or page visit) that the member made on the site. This table could also have information about when the interaction took place and the type of event that the interaction represented (For example, is it a "Ticket Purchase" event, a "Search" event, or an "Add to Cart" event?). The two tables are then interrelated by the 'member ID' column, as illustrated in Table 4.2 and Table 4.3 below.

Member ID	Email	Sign up Date
WY0	Sara@gmail.com	7/11/2018
WY1	Chris@gmail.com	21/10/2018

Table 5-2 Members table for airline clients.

Table 5-3 Interaction table with an e-commerce airline website.

Interaction ID	Member ID	Date	Type	Amount
X0	WY0	30/12/2018 08:22:00	Add to Cart	N/A
X1	WY0	30/12/2018 08:22:40	PurchaseTicket	400
X2	WY0	31/12/2018 10:12:00	PurchaseWeight	100
X3	WY1	01/01/2019 09:02:00	Add to Cart	N/A
X4	WY1	01/01/2019 13:00:00	PurchaseTicket	1200

**Note the `member ID column' which is uses to interrelate the tables*

To come up with an accurate prediction of when a member will next purchase an item, it would be necessary to have a single numeric feature matrix with a row for every member, which would then be used in the machine learning algorithm. The members' table does not contain much essential information. However, a few features such as the number of days since a specific member signed up can be constructed, though the options are limited at this point. To improve the predictive power, therefore, the historical data in the 'interactions' table should be considered, an operation that is possible through feature engineering. Aggregate statistics for each member can be computed using all values in the Interactions table with reference to the Members' ID. Below are possible aggregate features that can be derived from the members' historical behaviour.

- Average time between past purchases.
- Average amount of past purchases.
- Maximum amount of past purchases.
- Time elapsed since last purchase.
- Total number of past purchases.

To compute all these features, it would be vital to first find all interactions related to a particular member. Filtering out interactions whose "Type" is not "Purchase" would be conducted and subsequently it would be possible to compute a function that returns a single value using the available data. Generally, this process is unique for each case and

dataset. These features are highly specific and wouldn't make much sense for a dataset from a different industry, such as one describing network outages. However, in a network outage dataset, features using similar functions can still be built. To achieve this, it would require transformation of the Location column into one with a True/False value that indicates whether the data centre is in the Arctic Circle, for example (Zheng & Casari, 2018). Features can also be computed by utilising aggregation functions similar to the ones used for e-commerce, such as the following:

- Average time between past outages.
- Average number of affected servers in past outages.
- Maximum number of affected servers in past outages.
- Time elapsed since last outage.
- Total number of past outages.

To effectively use machine learning algorithms and consequently build predictive models, this type of feature engineering is essential. Deducing the right set of features to create leads to the biggest gains in performance (Kuhn & Johnson, 2019). This is usually the primary focal point of a scientist on most occasions. The right transformations depend on many factors such as; the type/structure of the data, the size of the data, and the objectives of the data scientist. In practice, these transformations run the gamut time series aggregations (average of past data points), image filters (blurring an image), and turning text into numbers (using advanced natural language processing that maps words to a vector space), as illustrated above.

Feature tools were also developed in this study to relieve some of the implementation burden on data scientists and reduce the total time spent on this process by enabling feature engineering automation. Feature engineering transformations can be "unsupervised", meaning that computing them does not require access to the outputs, or labels of the problem at hand. Additionally, the interpretability of such data is usually of a high level. In the examples mentioned above, the historical aggregations of member data or network outages are interpretable. On the other hand, the image filter is not, since each feature would represent a pixel of data. It should be noted that feature engineering methods are less common than feature selection methods. This is because feature selection has been in use for a long time, whereas feature engineering has only been designed whenever a need arose to solve the problems of a specific situation.

5.2.1 Feature Selection

With any given dataset, it is possible to select many features. The crucial consideration point is which features are to be used for a specific model to achieve the desired results. There also exists an infinite number of possible transformations. Even when restrictions are maintained to the space of common transformations for a given type of dataset, there are still thousands of possible features left (Kuhn & Johnson, 2019). It would be more admirable when all these features are plugged in to determine the ones that worked. However, the algorithms in reality do not function well when overloaded with too many features. This is especially true when the number of features is greater than the number of data points.

There exist numerous feature selection algorithms capable of converting a set with too many features into several subsets of the desirable size. These algorithms are also ideal for various data types, as with feature engineering algorithms. The choice of a specific feature selection algorithm should be driven by the goals of the data scientist. Feature engineering should however come prior to all of the aforementioned points. It is quite a challenge to select the most predictive features from a large space. As such, the more training examples one has, the better the performance, though the computation time will definitely increase (Dong & Liu, 2018). There are several overarching methods exist which fall into one of two categories:

- Supervised selection.
- Unsupervised selection.

5.2.1.1 Supervised Selection

This method entails the examination of features together with a trained model from which computation of performance is possible. This strategy tends to work well because of its interpretability as well as features being selected based on a model's actual performance (Guyon et al., 2008). The drawback is that these strategies require a lot of time to run. Although it is a guarantee that this method will deliver the expected results, it is quite expensive and thus should be limited to small feature sets. This strategy attempts to combine features in every possible way, and consequently chooses only the best combination. If, for example, there are 1,000 features and only 10 are required, then 2.6 x 10^23 different combinations will be tried. This would be

extremely time consuming. However, there are more complex but less ideal algorithms that can perform the operation in less time.

5.2.1.2 Unsupervised Selection

This method entails a technique by which machine learning algorithms are fed with unlabelled data in an operation popularly known as a clustering algorithm. This operation entails the grouping of similar objects based on a given criterion, such as density or distance. The objects present in any grouping therefore tend to be more similar than the outliers. This technique proves helpful when selecting significant features from a training dataset. Each unsupervised algorithm has its strengths and drawbacks that determine its applicability. In this study, the principle component analysis (PCA) technique has been used to select essential features. This has been achieved through the identification of independent features and consequently the removal of the redundant features from the training dataset. It can therefore be stated that the primary objective of these algorithms is to study the intrinsic and hidden data structures to get a thorough understanding, facilitate segmentation of similar datasets into groups and, as a result, simplify them.

5.2.1.3 Fundamental Tools of Data Science

In a bid to improve the performance of their models, feature engineering and feature selection are often the first areas that data scientists seek to improve. Vivid understanding of these areas is of importance to any data science task. Through feature engineering, it is possible to come up with complex models that could only otherwise be developed from raw data (Kuhn & Johnson, 2019). It also allows you to build interpretable models from any amount of data. Feature selection enables limitation of these features to a manageable number. Consequently, the risks of overwhelming the algorithms are greatly reduced, as well as the computation time.

5.2.2 Feature Extraction Process

As previously mentioned, feature engineering is the process through which domain knowledge and data science skillsets are combined to come up with features that speed up a model's training as well as providing more accurate predictions. Its main aim is the means of deriving a measurable model from data, which can be used to accurately anticipate future conduct. Recently, the measure of accessible data has expanded exponentially and "big data analysis" is required to be at the centre of most future

advancements (Breiman & Friedman, 1985). Due to rapid improvements in the field of data analysis, there still lacks an agreement on how new features should be extracted from a small dataset.

This study therefore undertakes a variety of experiments on feature engineering by utilising existing sources of learned or engineered knowledge. This is not a very familiar approach in the study of airline passenger numbers, as numerous scientists don't consider it productive. Such scientists tend to focus more on new learning algorithms or feature selection techniques with the main aim of enhancing performance. However, studies have proved that performance can also be improved by various feature engineering techniques (Freeman & Tukey, 1950). Building machine learning classification models, feature selection and extraction are vital phases due to the critical roles they play in enhancing the accuracy and the overall performance of the model. These phases are relatively expensive with no guarantee that features that have been extracted manually will fit appropriately in different data modalities. Through deep learning models, the phases of feature extraction, selection and classification are integrated into a single optimised process. However, computing them is quite expensive compared to traditional machine learning methods. Additionally, for good classification performance to be achieved, large training datasets are required.

The process of feature engineering in a focused structure assumes the procedure below:

1. Investigate the data.
2. Clean/process the data.
3. Construct a statistical model (repeat) in order to derive features
4. Select the best features.

In this chapter, several feature engineering methods for air passenger numbers are investigated in the context of a symbolic rule-based learning algorithm. Many new features are explored in an attempt to find extra information and features that were not present in the original dataset by creating and extracting new helpful features. For a machine learning model to be able to classify some objects, they should initially be represented as sets of characteristics (Nixon, 2013). In the training set, transformation of each object into a vector of feature values explaining a spot in a feature area is essential. The aforementioned efforts have been applied in this study to examine the

feature engineering methods for airline passenger numbers within the context of symbolic machine learning, as illustrated in Figure 4.1.

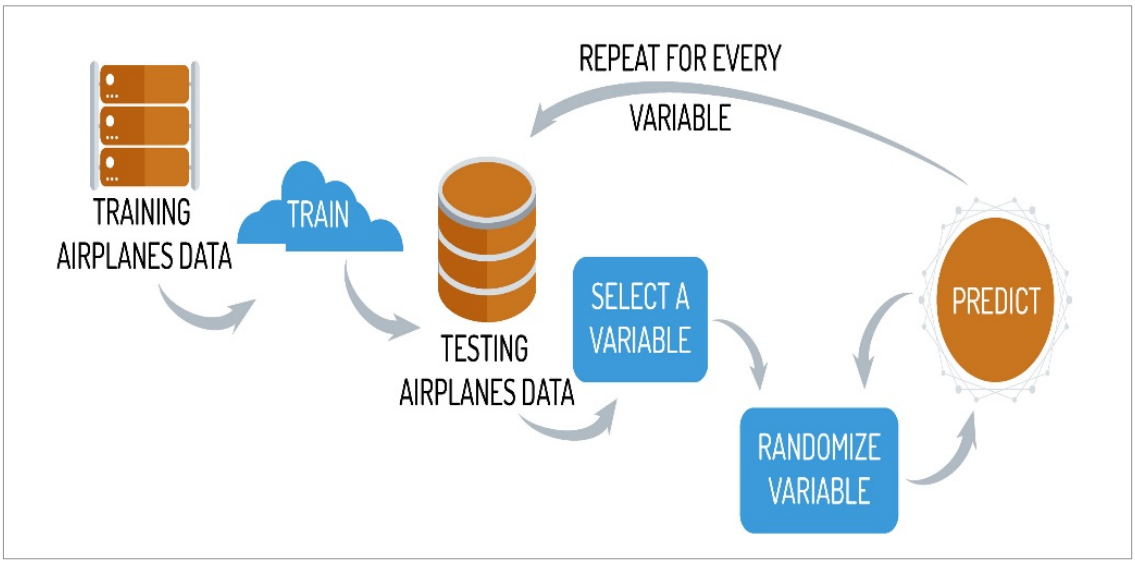

Figure 5-1 Feature engineering techniques for aeroplanes dataset.

The main idea is that modelling and the prediction will each be performed once to obtain a benchmark score. Subsequently, prediction will be done hundreds of times for every variable in the product while randomising that adjustable. If the variable being randomised affects the model's benchmark score, then it is marked as an essential variable. Alternatively, if nothing changes, or should the variable beat the benchmark, then it is marked as useless. Running each variable's predictions hundreds of times gives a clear image of the variables that are impacting the model and the extent of their implication. This method is advantageous due to its uncertainty towards any specific model. It therefore gives the possibility of achieving any kind of result after the modelling stage.

5.2.3 Feature Generation and Extraction

An effective approach to boosting the number of features is to physically incorporate several of them based on an educated guess about their importance. At times, some features are rather small, and it's not often intuitive which features hold predictive power (Domingos, 2012). In such a case, it is possible to use automatic methods to add valuable features. Past studies have proved that feature engineering, or creating new input features for machine learning models, is a key part of building an accurate predictive model (Domingos, 2012). Each problem is usually domain-specific and, as a result, better features are often the deciding factor of a model's performance. At the

commencement of this study, the main dataset, which was the starter file, contained the following seven feature spaces relying on Muscat International Airport experts' knowledge of the aviation domain and the associated passengers' journeys with respect to the target variable:

```
FeatureSpace={'Year','Month','DestinationAirport',Destin
ationAirportName,'DestinationCityName',     'Destination
Region Name','Destination Country Name'}
```

Based on these original feature space, this study aimed to train a simple model on this feature space in order to have some starting point from which we could observe the importance of variables and some performance measures. In this first approach, the main interest was to improve the "explained variance" measure. To commence with the random forest process, airline dataset for flights between Oman and other cities in different countries became the primary focal point. The final goal was to establish the main factors/variables affecting the overall figures of passengers travelling. The first data were read from *Pax_data* and the training dataset was created through the removal of the first column. To simplify the results, all fields were converted to numeric with the use of fix Sapply functions. As such, it will be easy for the random forest calculation to create random trees:

```
pax_data=read.csv('E:/pax_data.csv',header=T,sep=',')tra
in=pax_data[,2:ncol(pax_data)]train[,1:ncol(train)]=sapp
ly(train[,1:ncol(train)],as.numeric)
```

Since machine learning will be conducted using the feature engineering technique, which requires numerical data, it became necessary to transform our text data into numbers. Otherwise, a lot of critical information from the table would have been omitted. As explained in Section 4.2, text data, which is one of the most challenging forms of data, can be easily converted into numeric data with the help of feature engineering techniques. Such techniques have been derived from natural language processing as explained in the previous section. Upon successful transformation of text data into a continuous numerical scale, the simple *randomForest* library of R was used to create a random forest with *Passenger.numbers.in.thousands..000* as the predictor value on the basis of the number of variables as the response. It is important to not only have an accurate, but also an interpretable, model. Other than wanting to know what this study model's air passenger figures predictions are, it is also desirable to know why the predictions are high or low and the most significant features in determining the forecast as well. Identification of the variables can facilitate timely predictions and better still enable enhancement of the services rendered. There are various benefits

emanating from knowing feature importance indicated by machine learning (Kelleher et al., 2015):;

1. Clearly understanding the model makes it possible to not only verify its correctness but also to come up with improvements for the model by highlighting the important models that can be focused on.
2. It gives room for feature selection whereby insignificant x-variables can be eliminated and consequently produce similar or better results within a shorter training time.
3. It makes it possible to forego accuracy for the sake of interpretability in a business case whenever the need arises. When an airline website, for example, rejects a client's purchase request, the client can learn the reason behind it.

For the above reasons, this Thesis will therefore explore different approaches of interpreting feature importance in a random forest model. The majority of these approaches are applicable to other models, such as linear regression and black boxes such as XGBoost, as explained in Chapter 5.

5.2.4 Variable Importance from Machine Learning Algorithms

As an important tool for interpretation, feature selection is regularly used by scientists to examine model parameters such as coefficients in a linear model. Through it, measures of feature importance are also provided by various random forest (RF) implementations. Although few machine learning practitioners are aware, the RF importance technique – to be precise, the permutation importance – is applicable to a wide variety of models. This technique is common, efficient and reliable as it observes how the accuracy of a model is impacted by random sampling of predictors and consequently measures the variable importance from the observations. Unlike linear regression coefficients, which are poor proxies for feature importance, the RF technique does not rely on internal model parameters and thus makes it broadly applicable.

Use of permutation importance is recommended on all models, including linear ones. This is because all issues associated with the interpretation of model parameters can be successfully avoided. In Section 4.2.2 it is described how Breiman & Friedman (1985) identified accuracy as an issue of great concern. It is therefore worth noting that the more accurate the model in this study is, the more the importance measures and other

interpretations can be trusted. The difference between predicted and expected outcomes, also known as residual analysis, has to be measured to determine the goodness-of-fit in a linear model. It is however not possible to tell when a model is biased when conducting residual analysis. Breiman & Friedman (1985) quote William Cleveland, one of the fathers of residual analysis, as having said that "residual analysis is an unreliable goodness-of-fit measure beyond four or five variables."

5.2.4.1 Variable Importance on Original Features

Figure 4.2 below employs *varImpPlot(rf)* to illustrate the observed next feature importance as well as explained variance for the random forest library obtained from the original data. An expounded fraction of the model translates to be the explained variance. This is the R^2 value in a simple linear model, which is equal to the squared correlation coefficient. When the learned parameters of a flexible model, for example, the structure of a decision tree, have considerable variations with the training data, such a model is said to have a high variance. The measure of how the target variance of the training set can be explained by out-of-bag predictions is known as the percentage explained variance. Unexplained variance would be to due true random behaviour or lack of fit. Running *randomForest:::print.randomForest* as the last element in *rf.fit$rsq* and subsequently multiplying it by 100 retrieves that percentage explained variance. Figure 4.2 is an illustration of the above.

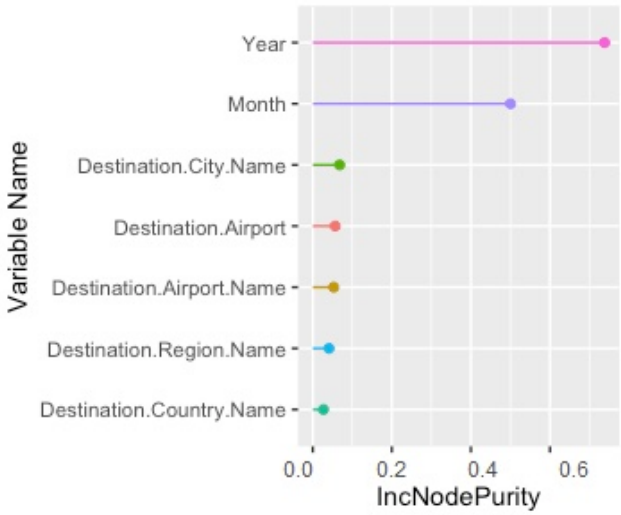

Figure 5-2 Model 1: Variable importance on original features (% Var Explained: 33.43).

Seven features were used to train a regression to predict passenger numbers using seven features. For a better explanation of feature selection, a column of random numbers was added. All features of less importance in the random column were considered junk and

thus tossed out. The *varImpPlot* which described parameters with high responsiveness at the top and those with low responsiveness at the bottom was plotted after the training.

From Figure 4.2**,** it can be stated that the "Year" and "Month" parameters are more responsive (strongest predictor) respective to other parameters that predict the values of the number of passengers with a 33.43% variance. It is worth noting that as the accuracy of the model increases, feature importance measures as well as other interpretations can be trusted more. As datasets get larger, it is more challenging to build reliable and efficient predictive models and at the same time understand what is happening with the data. For example, it is the desire of all data analysts to know the predictors that have a significant influence on the predicted results in a fitted model. At times, a variable that makes business sense may be available but there lacks assurance as to whether it can help in predicting the variable Y. It is worth noting that a feature that may be quite useful in one machine learning algorithm – for example, a decision tree – could end up under-represented or even unused in another algorithm, for example, in a regression model. However, it should be kept in mind that a variable with poor signs of offering help in the explanation of the response variable Y could be helpful if combined with other predictors. By this, it is meant that a variable might have a low correlation value of (~0.2) with Y, but in the presence of other variables, it can help to explain certain patterns/phenomena that other variables can't explain when alone. In such scenarios, therefore, the decision as to whether such variables should be included or excluded is hard to make. Outlined later in this topic are strategies that can greatly aid in eliminating the challenge. From the strategies, a firm understanding will also be gained on the importance of a particular variable and how much it contributes to the model.

Moving on with the first step of this research, two new features were created even though the original space feature did not look promising. In many cases, the given features are not directly useful or are much correlated. One example of features that are not directly useful is the one described in this section, where it was intended to predict the airline passenger numbers using specific features such as, month, holiday, GDP, jet fuel prices, total population, interest rate, average base fair and distance. In simple models, direct usefulness of the original features does not prevail. In other cases, the features may be too many and as also display high levels of correlation and thus fitting

robust statistical models becomes a challenge. In the case of this study, replacement of the features with a new and smaller set of features that contain similar information as the original ones was tried. Since the month is influential, it was decided to make a new feature based on the four seasons derived from the month column. From this action, it was observed that there is a possibility of extracting useful information from the month, which can subsequently help in determining whether month1, month2, etc. are summer, winter, etc.

5.2.4.2 Variable Importance with Season on Original Features

The techniques discussed above are also referred to as "binning" and provide insights during descriptive data analysis. Binning is a way of converting numerical continuous variables into discrete variables by categorising them on the basis of the range of values of the column in which they fall. For example, if the 'seasons' feature is added in the transaction level data while rolling the data up on the "seasons" column, it is possible to directly compare air traffic across the four seasons. A loop will be set up where data is extracted from the month and assigned to a season. Since cut was used to determine whether, in this regression task, there is any influence of the season corresponding with any particular month, the solution uses the base of R only. As such, the prediction of seasons and the plotting of importances was conducted using the other seven features on a random column with a newly built RF classifier. 'Month' was used to develop a new feature as follows:

If months January, February, and March are grouped as season 1, months April, May, June grouped as season 2, and so on until four seasons have been attained from all months, then 'month' will be captured by the RF and it will be reasonable to bin it into four groups based on the four seasons developed. With this column newly added to the training dataset, a random forest procedure was run and a new graph printed. Figure 4.3 demonstrates.

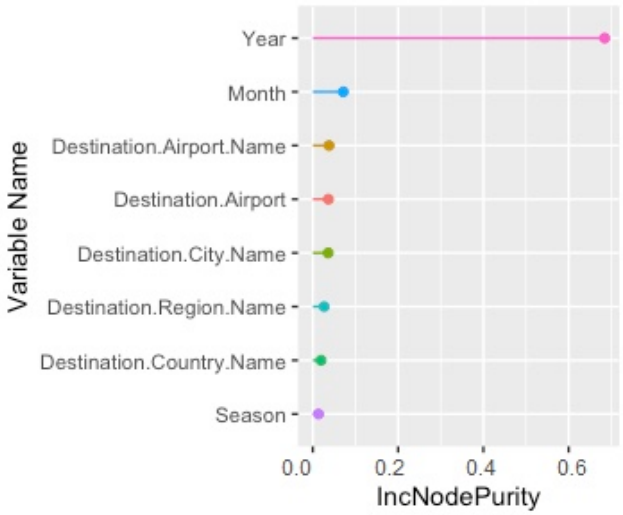

Figure 5-3 Model2: Variable importance with season on original features (% Var Explained: 37.13).

From Figure 4.3, it is clear that a vital role is played by the newly added column in the prediction of airline passenger figures. The variables 'year', 'month', and 'season' are the most essential, with 'year' playing a key role as the principle variable. It is therefore clear that the number of passengers from Oman to various destinations is largely influenced by the seasons. Additionally, the random column is considered more predictive of the season than other features, as per the RF classifier. Learning in machine learning models is greatly influenced by the features present in the data frame columns. As such, it is evident that more accurate predictions and faster training will occur if the features are better. Figure 4.3, for example, shows that around Christmas, and generally during winter, airline passenger figures tend to be relatively high.

Further on, the algorithm becomes more advanced and reliable when a feature is added specifying the exact number of days left before Christmas, rather than by depending on the date alone. As such, it is possible to employ domain knowledge and perception to engineer similar simple features that will largely increase the accuracy of the models. It is clear that passenger figures during the weekends are relatively higher than during the weekdays. This information can be conveyed to the machine learning model by creating a 1/0 flag in the dataset row that corresponds to the weekday/weekend respectively. Similarly, monthly statistics show that passenger figures are higher during public holidays. The addition of 1/0 flag can therefore play a significant role in relaying this information. However, seasonality associated with moving months cannot be

captured with ease. Even with monthly statistics, this is still challenging as festivities can occur in any month of the Islamic calendar. There is no provision for this in the usual seasonal models and even the general assumption of the complex seasonality is that festivity patterns occur at the same time each year. Vectors for all events can therefore be used to successfully counter this challenge.

5.2.4.3 Variable Importance with Season and Holiday on Original Features

In this study, therefore, a decision was made to derive a new feature based on Oman's major holidays. Owing to the fact that there are multiple public holidays per month, a better magnitude of holidays in each month was obtained by establishing the sum of all holidays. Creating a vector of all major holidays and matching their respective dates in the model's data frame was pinpointed as the most effective way of getting a general holiday indicator variable. A common magnitude of all holidays was achieved by marking a vector with 1 for months with holidays or with 0 for months with no holidays. As a result, a significant influence on the passenger figures in any particular month was witnessed. Figure 4.4 is an illustration of the above information.

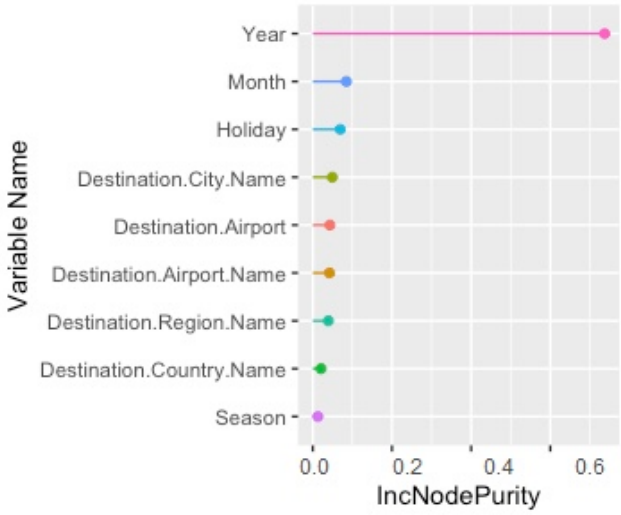

Figure 5-4 Model3: Variable importance with season and holiday on original features (% var explained: 49.61).

By adding one more column, 'holiday', as illustrated in the figure above, it would be possible to find out whether the number of holidays in a particular month are influential to the seasons or not. According to Figure 4.6, it is clear that the seasonal prediction power is weakened by the addition of holidays. Only the first three variables can therefore be considered for prediction. The variance explained increased from 37.13%

to 49.61% creating these additional features that are also captured by decision tree feature importance.

Numerical data consists of both decimals and integers, also known as real numbers. In any machine learning model, it is essential to have numerical data since only numbers can be understood and interpreted by machines. It is however common to have underlying assumptions on data in most ML models. As such, for machine learning algorithms to give optimal results, it is important to identify the assumptions and transform the related data into the most suitable format. In a linear regression, for example, there is the assumption that residuals are normally distributed and that a linear relationship exists between the variables. To eradicate the assumptions, it should be ensured that the Y variable is normally distributed. According to Kuhn and Johnson (2013), symmetrical or unimodal distributions of features facilitates a better functionality of machine learning algorithms.

Machine learning cannot be successful when data is input in its raw form. The success of machine learning is more heavily dependent on the quality of features that have been extracted than on the ML algorithms. Manual intervention and domain knowledge are two essential features that significantly impact the engineering process of meaningful and descriptive features. The possibility of the feature engineering process to generate large number of figures is another complication. If a high ratio of features with respect to the total number of points presents itself, then the curse of the dimensionality challenge occurs. As this ratio rises, the likelihood of the ML algorithm uncovering spurious patterns which are less related to the reality also increases. This has a negative impact on the model, which is likely to exhibit low accuracy levels upon deployment. To avoid this problem, effective strategies will have to be implemented by only selecting the important features. Based on the above information, a strategic test was conducted as part of this study to determine whether the addition of new features such as jet fuel price, gross domestic product, distance, interest rate and population would help improve the current predictor's accuracy.

5.2.4.4 Variable Importance with Season, Holiday and GDP on Original Features

GDP is an illustration of the market value of all finished goods and services produced in a nation within a specific year. It is a global standard measure used to determine the economic growth of a nation as well as comparing economies of various states

internationally. The basic assumption is that the performance of a nation/society is reflected adequately by an economy's value of income, expenditure or production. A strong economy is signified by a high GDP. In such a nation, the rate of unemployment is relatively low and the demands of labour and production are on the rise. When the economy of a nation is growing, it is likely to attract more investors and consequently lead to the development of the state. It is a complicated task to calculate the gross domestic product. So as to come up with meaningful insights for use in this study, some features from the available data were constructed manually due to format inconsistency. A feature derived from Oman's GDP for the period 1998-2016 was used to test the impact of GDP on the model's prediction power. The training dataset was first kept in temp variable, that is, *temp=train,* after which GDP columns were added based on years. The RF code used to check the influence of this feature on the prediction capacity of this model is as follows:

```
table(train$Year)gdp=c(6330.73,7059.21,8710.99,8559.57,8
670.26,9070.49,10115.04,12398.59,14575.16,16225.66,22963
.38,17518.83,17518.83,21164.34,21631.89,20204.93,19129.8
4,15550.68,16329.00)gdp_year=rep(1998:2016)gdp_df=data.f
rame(c(as.data.frame(gdp_year),as.data.frame(gdp)))
```

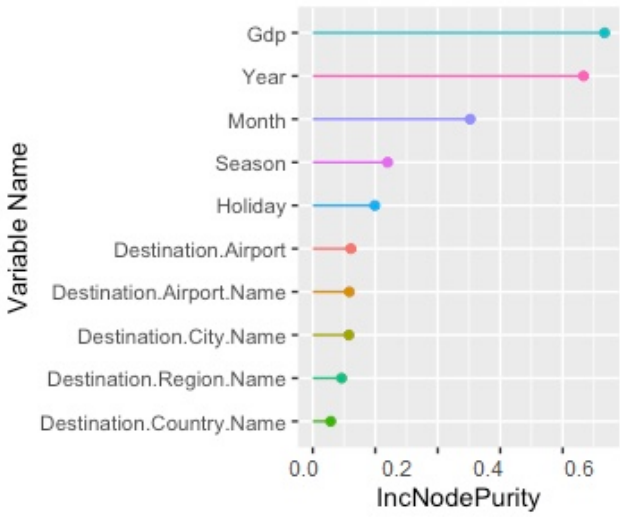

Figure 5-5 Variable importance with season, holiday and GDP on original features (% Var Explained: 67.06).

From Figure 4.5, it is clear that the annual GDP has more influence than other parameters. An increase in the explained variance from 49.61% to 67.06% is realised, leading to the creation of additional features that are also captured by decision tree

feature importance. The target data of the total passengers were not aggregated with other variables. If such an aggregation could have been conducted, the train data could have been contaminated, leading to interference with the outcome.

5.2.4.5 Variable Importance with Season, Holiday, GDP, and Jet Fuel on Original Features

Later, an RF classifier was built to employ the jet fuel price feature in predicting airline passenger figures. As previously done, a random column was used to plot the importance. Between July 2004 and July 2008, the cost of jet fuel hiked up to 244%. This consequently marked it as the operating item with the highest cost (IATA, 2010). The air transport system is affected by changes in jet fuel on two main sides, which are:

1. Networks and fleets, as well as scheduling and pricing, affect the supply side.
2. The general economy affects the demand side.

The price of crude oil plays a vital role in determining the effective cost of jet fuel. The general expectation is that when the effective cost of fuel increases, an imbalance between demand and supply occurs, leading to changes in the supply of airline items to such areas as network and fleet. That said, the significance of determining how an increase in fuel prices affects airline passenger figures in this study became evident.

Data on jet fuel was therefore added to the model to determine their impact on airline passenger figure predictions. For jet fuel, data were copied and a csv file derived from it. It therefore became possible to determine whether a new feature, 'Jet Fuel', would have an impact on prediction.

The new column was added into the PAX and train data, and again, the RF was run as follows:

```
jet_fuel=read.csv('E:/jet_fuel.csv',header=T,sep=',')
```

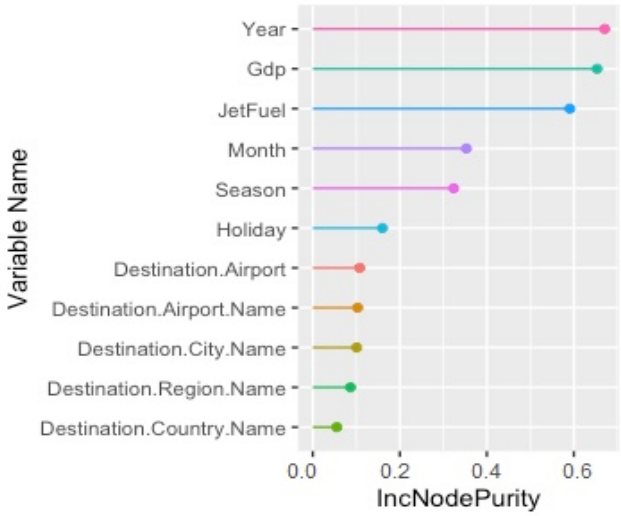

Figure 5-6 Variable Importance with season, holiday, GDP, and jet fuel on original features (% Var Explained: 69.18).

Figure 4.6 illustrates the prediction power of this study's random forest model. If the top variable is dropped from the model, the prediction power will significantly reduce. On the other hand, introduction of one of the bottom variables might not have much impact on the model's prediction power. The explained variance increased from 67.06% to 69.18%.

5.2.4.6 Variable Importance with Season, Holiday, GDP, Jet Fuel, Population, and Interest rate on Original Features

Population size is another factor with the potential to influence air travel demand. With a higher population, the demand for mobility increases. As such, it is essential to consider the population of a country in the estimation and differentiation of air travel demand. Nevertheless, population was added as one more parameter, subsequently deriving a new column in the train database. The influence was again determined:

```
pop_total=c(2225481,2239403,2272547,2323203,2385075,2448
194,2506891,255337,2593750,2652281,2762073,2943747,32100
03,3545192,3906912,4236057,4726413,4490541,5119745)year=
rep(1998:2016)
```

According to Richard E. Quandt (2006), social, economic and demographic factors affect peoples' rational decision-making capacity and in turn affect the demand for mobility. The author also explicitly states that different modes of travel and various destinations are commodities, all which have varying prices from which consumers

make choices with the main aim of maximising satisfaction. This statement has a broad perspective and is backed by various philosophical explanations such as the utility theory, economic theory as well as consumer theory, all which have been considered in the selection of variables used in this study's model. A solid understanding of the model's theoretical foundations is essential as it will enable the conducting of evaluations based on accidental or causal interrelations of the variables.

Global interest rates are constantly on the rise and more so in the US. This has a detrimental impact as a high US dollar exchange rate results in losses for airlines whose revenues are in local currencies. In India and Indonesia, for example, exchange margins are sliced down significantly as the currencies operate at a 20-year low against the United States dollar. Consequently, all airlines that use the US dollar to pay for costs such as lease rentals and jet fuel, but on the other hand, they also book their revenues using their local currency, incur huge losses. Escalations in finance contracts as well as aircraft orders as a result of rising interest rates commonly negatively impact airlines.

According to a banker, Winston Yin from Korea Development Bank, the cost of borrowing for airlines with a floating rate debt could be significantly impacted even by a 50-point base increase on interest rates. Yin further states, "Some airlines have been requesting interest rate hedges for new aircraft deliveries." He also says that nowadays, based on requests for proposals (RFPs), airlines are beginning to prefer fixed rate funding. Craig Fraser, an official from Fitch Ratings backs Winston's view by stating, "By some measures, this is the longest upturn we've had in the aviation sector, but we need to keep in mind that this is a sector with a high structural risk profile. Performance can turn very quickly, and there are a lot of signs of complacency if you look around … They are certainly insisting on the optional to fix funding rates in the future."

As such, an interest rate according to the 'years' column was added to the model and the RF was again employed to check its impact on prediction, as follows:

```
interest_rate=c(27.05,-1.21,-6.58,14.46,3.67,-1.92,-
4.81,-12.57,-5.48,-0.90,-19.93,43.50,-7.61,-
9.34,0.68,6.9,3.35,26.24,25.24)year=rep(1998:2016)
```

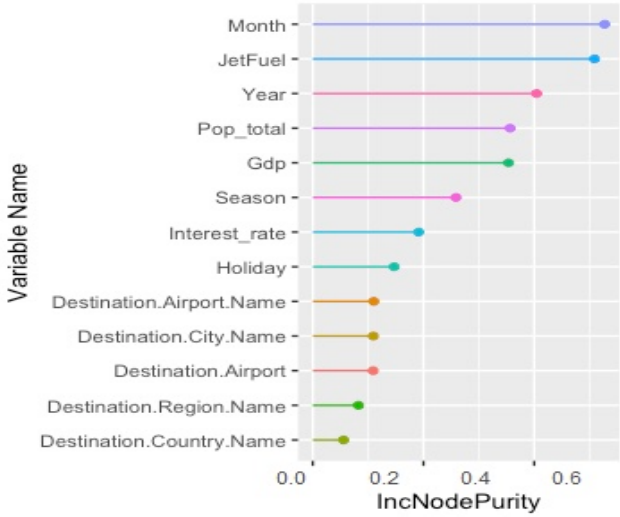

Figure 5-7 Variable importance with season, holiday, GDP, jet fuel, population, and interest rate on original features (% Var Explained: 68.55).

From Figure 4.7 it is clear that the two new parameters (population and interest rate) are influential, with an explained variance of 68.55%. There are several interrelated methodological decisions that a researcher must make when performing a factor analysis. One key decision is that an appropriate variable for factor analysis must be determined. Peterson (2000) states that, on average, 56% of variance is usually accounted for. It is not until the last factor accounts for less than 5% of the variance or until the extracted factors account for at least 95% of the variance that the factoring procedure can be stopped in natural sciences. For social sciences, on the other hand, a solution accounting for 60% of the variance should be considered as satisfactory in cases where information is less precise (Hair et al., 2014). According to Nunnally and Bernstein (1994), ‘‘the goal [of a factor analysis] is to explain the most variance (or related property) with the smallest number of factors.” Tinsley and Tinsley (1987) stated that ‘‘less than 50% of the total variance is explained by a factor solution.’’

5.2.4.7 Variable Importance with Season, Holiday, GDP, Jet Fuel, Population, Interest rate, and Distance on Original Features

It therefore became necessary to add a ‘distance’ parameter as another variable in the model to investigate its impact on the demand for air travel. The usage of a particular aircraft on any route is usually well explained by the total distance covered. With an increase in distance between two destinations, larger and longer-range aircraft are required. Although tickets often become more expensive with an increase in distance,

this is not always the case. A flight between Los Angeles and Portland, for example, was booked at the end of the month of January at $160 for a two-way trip. On the other hand, a person flying from the same departure place but to as far as Chicago, which is more than twice the distance, would not need to pay double. The extra airfare was recorded to be only an extra $28 for a two-way trip. It should be noted that there are various factors that affect the prices of air tickets and distance is just one of them.

```
distances=read.csv('E:/distances.csv',header=T,sep=',')
```

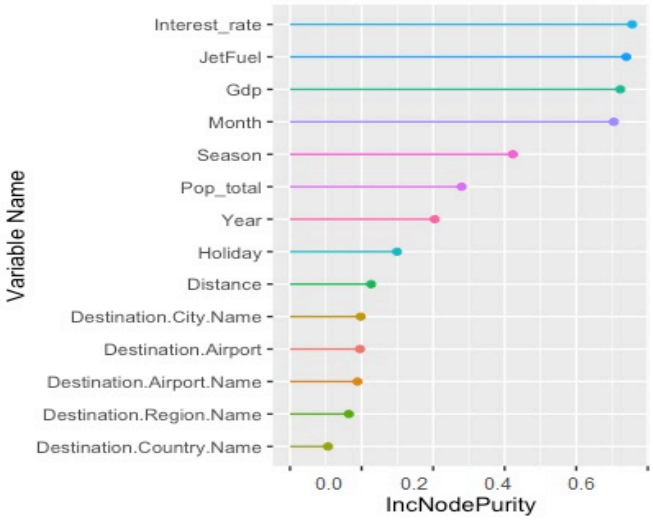

Figure 5-8 Variable importance with season, holiday, GDP, jet fuel, population, interest rate, and distance on original features (% Var Explained: 67.74).

From Figure 4.8 it is clear that distance does not play a big role in the prediction of the total number of passengers, with an explained variance of 67.74%.

5.2.4.8 Variable Importance with all Features

The pricing variables for airfare feature tend to be far more than for the majority of other features. Within the aviation industry, the issue of how effective measurement of the fare average base, also known as price on demand, has been debated for a long time. Another issue of concern has been how significant commercial value can be gained by airlines proactively managing it. Price on demand explains the percentage increase in airline passenger figures when the prices drop by a certain percentage. It is important to determine how demand is impacted by price changes, as it guides airlines when making aircraft orders worth billions of dollars. Airfare pricing removes a significant risk from any airline's business plan. From Oman's air data, this study is able to measure the price of various routes which have received significant attention as

evidenced by the airport's operations as well as the Transatlantic low cost. The average base fare was added into the model, as follows:

```
fair=read.csv('E:/avg_base_fair.csv',header=T,sep=',')tr
ain["fare"]=fair$Avg.Base.Fare..OMR.
```

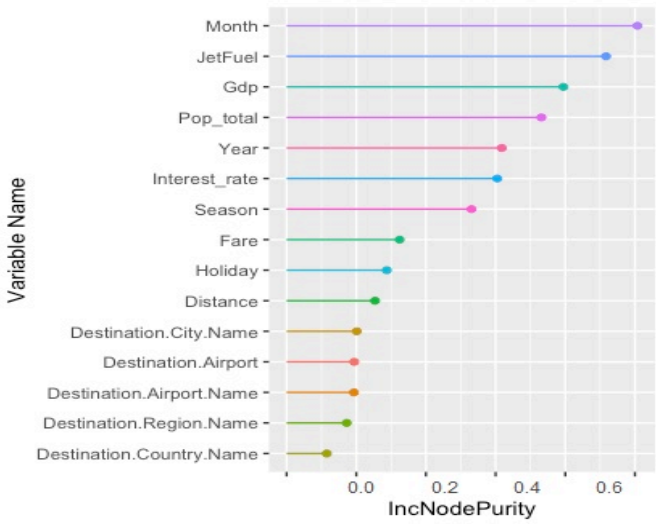

Figure 5-9 Variable importance with all features (% Var Explained: 69.04).

From Figure 4.9 it is clear that fare rates have a significant impact on airline passenger figures. The explained variance raised from 67.74% to 69.04%. Although price elasticity is a valuable measure, many airlines have not recognised it and incorporated it in their planning. When determining fare rates, it is essential to understand price elasticity measures so as to come up with the most effective rates. The urge to understand the factors behind airfare pricing is also driven by the constantly growing air carriers, with each seeking to expand their services.

5.3 Optimising a Deep Learning Model to come up with a Robust Neural Network Topology

Deep machine learning, also known as deep learning, is a kind of machine learning algorithm model that inputs data to higher-level abstraction by using a deep architecture with many hidden layers composed of linear and non-linear transformations (McGregor et al., 2004; Friedman & Popescu, 2008; Rizescu & Avram, 2014). In other terms, deep learning is the process through which a neural network, in which the input and output are separated by several layers of nodes, is used. The numerous layers of nodes are composed of multiple levels of non-linear operations such as neural nets with numerous

hidden layers. There are several areas in which deep learning entailing feature representation and predictive modelling has been successfully explored and applied. Such areas include computer vision, natural language processing, remote sensing and bioinformatics. Deep learning has been applied extensively due to the numerous benefits it provides. Such benefits include the ability to model the processes of complex systems, the ability to generate high level feature representation, high levels of prediction accuracy, and higher robustness in modelling (Chen & Wasserman, 2016).

Prediction of airport usage is an important aspect of air passengers' prediction research.

According to Gosavii & Bandla (2002) and Riedel & Gabrys (2003), original forecasting models don't consider the influential factors comprehensively, and less so when dealing with changeable state or complex airport usage states. This Thesis, however, recognises a variety of attributes such as classification vectors and designs new time series algorithms through the calculation of representatives in all kinds of clusters between observing time series and representatives. A new approach of feature selection is investigated and features that are important in prediction of the number of passengers traveling in different seasons are subsequently demonstrated. Higher probabilities of surviving among the more fit combination of features guides feature extraction in deep learning. In this study, therefore, as explained in Section 4.2.3, features were extracted by Feature-Space = {'Year', 'Month', 'Destination Airport', 'Destination Airport Name', 'Destination City Name', 'Destination Region Name', 'Destination Country Name'}. In Figure 4.4, the impacts of various parameters from a variety of techniques are illustrated. This is achieved through classification accuracy and subsequently comparing the results with those obtained through the addition of features step by step. Additionally, discussing the importance of correlation among different features gives a vivid understanding of the impacts of such correlations, as further explained by Riedel & Gabrys (2007).

In this study, deep attention is given to features by experimenting on new features obtained from optimised neural network hidden layers. The capabilities of the existing machine learning models were used to visualise the importance of feature selection. After choosing the set of features, adding new features in different steps and visualising the correlation among all the features, the methodology of incorporation of deep neural network was chosen. All modelling best practices are maintained by using this

methodology. Strong arguments against machine learning techniques are constantly emanating from various scholars. The majority of the scholars argue that the entire variable selection process lacks a clear definition. As such, they have studied deep neural networks with the chief aim of achieving more accurate forecasting results (Friedman & Popescu, 2008; Essa & Ayad, 2012). By focusing on the above-mentioned methodology in this study, desirable results were achieved, as demonstrated in the following sections of this chapter.

5.4 Illustration: Optimisation Techniques (Finding A Good Neural Network Topology)

5.4.1 Techniques Description

Among the various optimisation challenges encountered in deep learning, such as, it requires so much time, and consuming data input, neural network training is the most robust of all. A heuristic method based on computing saliency, also known as sensitivity, has therefore been proposed to counter this challenge. As such, H2O library, which runs under the R environment, was chosen for implementation in order to get better accuracy, and later on, find a good neural network topology, given the code explanations line by line. There exists a wide variety of neural network libraries such as Python and Tensor Flow. However, the main reason that led to the selection and implementation of H2O is its nature as an open source machine learning platform. Additionally, it enables companies to build models based on large datasets with no need of conducting any sampling and consequently achieve more accurate predictions. It is easy to implement at any level, fast, and with high accessibility. As such, companies can perform quicker data computations through its GUI-driven platform. As of now, the H2O platform operates with both the basic and advanced levels of algorithms, which include but are not limited to; bagging, deep learning, boosting, principle component analysis, k-means, time series, naïve Bayes, and generalised linear models. Additionally, H2O has facilitated operations for Hadoop, Python, R and Spark users by releasing APIs suitable for them all. As such, it becomes possible for researchers, like in this study, to build models at an individual level.

Furthermore, H2O is free for use to any interested researcher and enables quicker computation. It also consists of a safe feature through which a tool, such as Python or

R, can be directly connected with a CPU. With this rare feature, it is possible to channel more processing power and memory into the tools and consequently increase the overall speed of computations. As a result, it becomes possible for computations to be conducted at 100% CPU capacity. Computations can also be performed by connecting the tool with clusters using various cloud platforms. Even when operating with a small cluster, H2O handles large datasets using in-memory compression in addition to having provisions that enable the implementation of parallel distributed network training. As stated by Landset & Khoshgoftaar (2015) and supported by McGregor et al. (2004), the model has a straightforward hyperparameter tuning that entails Java back-end development as well as an option to run on some number of clusters k where k represents the number of cores on the processor

5.4.2 Solving a Problem (Dataset)

Principally, cross-validation splits the dataset into two: the training and the test sets. The expected error for the whole model is then determined by averaging the prediction errors for each of the two sets. In this study, therefore, data were split into five partitions of equal sizes. In the first fitting, 20% of the data is taken as the test set and accounts for the first fold. The remaining 80% is taken as the training set and accounts for the other four folds. Test/training data is then fitted into the model K times, where K is the number of folds. An average of the prediction errors from all fittings is then calculated to determine the overall prediction error for the model. Although five and ten splits/folds are commonly used in various studies, it is up to the researcher to determine the number of folds to have in a study depending on the data size. In other cases, when both the number of folds and the number of cases in a dataset are equal, that is, K=N, the leave-one-out cross-validation can be conducted. The error and variance are however affected by the number of splits. It should therefore be noted that the fewer the splits, the higher the bias/error and the lower the variance, and vice versa. After cross-validation was successfully conducted, a double check was run using a 25% dropout from the original train file. A dropout is a 5-fold cross-validation taken to its extreme, whereby, the test set is 9,840 observations while the training set is composed of all the remaining observations. It should be noted that in a dropout, 1=number of observations in the dataset. Furthermore, a more frequent and efficient running of cross-validation can be conducted with the use of various built-in functions in R. in this study, therefore, the customised codes were used and, as a result, a better understanding of the actions of

the algorithms and how they were undertaking the actions was achieved. Sacrificing some efficiency percentage would have enabled the creation of a 'cross-validator' but, on the other hand, a well customised process was achieved by personalising and tweaking most of the parameters. The 'Train' and 'Test' were the two main parts of the airline dataset selected at the beginning of the deep learning process. 51,983 observations were recorded for the 'Train' dataset while 9,840 observations were recorded for the 'Test' dataset. To capture the observations and record all relevant data, the author of this study made a request to, and was granted permission by, the Public Authority for Civil Aviation. However, restrictions applicable to the availability of such data were enacted in this research and thus limits the publication of the data. All the data were sourced from Oman Management Air Company (OMAC).

The dataset was in the form of large numeric variables that were used for predicting airline passenger figures for Muscat Airport. The data were not scaled and thus in raw form and contained binary columns representing quantitative independent variables such as unemployment rate, real GDP per-capita, GDP, and interest rates. The 'TARGET' column indicated the variable to be predicted. Such predictions will play a vital role in enabling OMAC to take the necessary steps based on real-time forecasts to improve service provision and consequently generate more revenue. However, the main focus of this study is not the dataset but rather the analytical models used for prediction. H2O and R features are also at the epicentre of the discussion due to their importance in deep learning. As stated in Chapter 1, the main aim of this study is to successfully build and train a deep learning prediction model. As such, this chapter has put more emphasis on feed-forward neural networks that will help to achieve the study's principle goal. The chapter will first explore the available data; then discuss its simplification; then describe the application of the model to obtain the prediction results. R language was used in the aforementioned steps, and in the H2O modelling process as well.

5.4.3 Import and Set-up Model (The H2O Package Implementation)

Deep learning in H2O is based on a multi-layer feed-forward artificial neural network that is trained with stochastic gradient descent using backpropagation. It is possible for the network to contain a large number of hidden layers consisting of neurons with tanh, rectifier and maxout activation functions. High predictive accuracy is enabled by advanced features such as adaptive learning rate, rate annealing, momentum training, dropout, L1 or L2 regularisation, check pointing and grid search. Each compute node

embarks on the multi-threading technique that works asynchronously to train a copy of the global model parameters, which in turn periodically contributes to the overall model by the use of model averaging across the network. A feed-forward artificial neural network (ANN) model, is the most common type of deep neural network and the only type that is supported natively by H2O. With the use of H2O, therefore, it is possible to build predictive models with programming environments such as R, Python, Scala and a web-based UI known as Flow.

In this section, a description of how a good neural network topology can be obtained using the H2O package in R is outlined. Below are the requirements and implementation steps.

1. The datasets need to be linked to the H2O cluster before the H2O engine is used to train the model. This can be achieved in many different ways, and for this study, a passenger numbers (PAX) data frame is linked and imported using the code [train=read.csv('finalTrain.csv',header=T , sep=','),]
2. The finalTrain is then split into a validation set, a training set, and a test set data. This provides the initial insight into the performance of the tuned deep learning model. To achieve this, the predicted values on the unseen test will be plotted against the real ground values. This is as illustrated in Figure 4.11, which is the original distribution plot. As explained in Section 4.4.3, the model parameters will be optimised using the validation dataset in the course of the training process, while the test dataset will be used to test the performance of the model.
3. The last step will entail predicting passenger numbers using classes with a grid dataset on the airlines' dataset.

With the aim being able to predict the PAX column, the original dataset copied in the *train_copy* variable for future use will be employed, as illustrated in the code below:

```
train_copy=train
target="Passenger.numbers.in.thousands..000."
```

The data will then be imported into R normally and will assume the shape below. It is the variable to predict. Figure 4.10 is a drawn scatter plot showing data pattern in the passenger flight list. It portrays the possibility of having enough variance to come up with a good predictor. Multiple runs could be conducted to determine the variation in

the prediction performance and to investigate the impact of model regularisation by tuning the 'Dropout' parameter in the H2O deep learning *(...)* function.

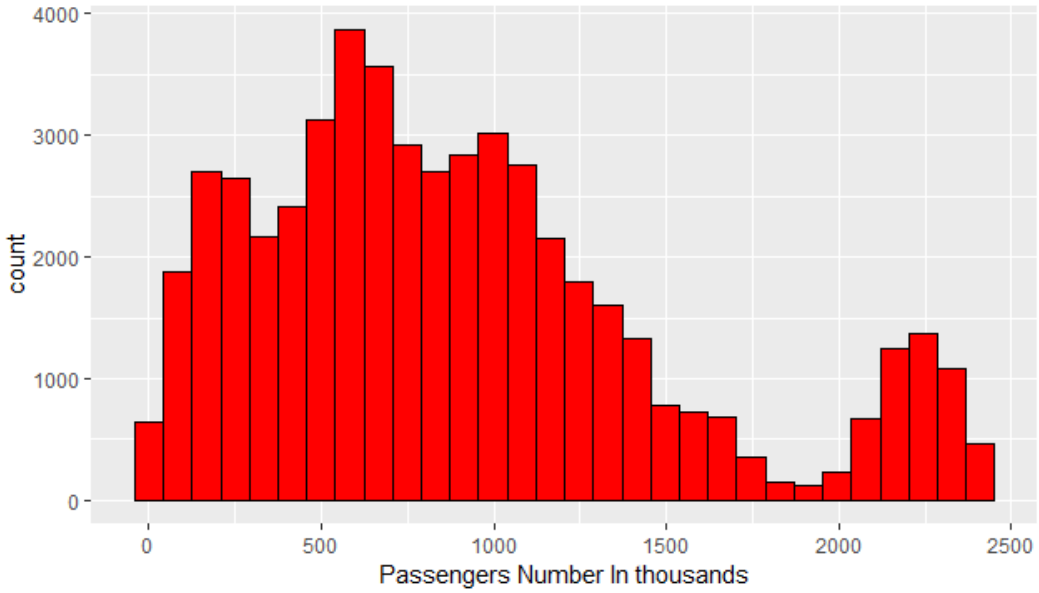

Figure 5-10 The original distribution of target.

For the next step, it is essential to consider memory usage to ensure that memory utilisation is enough over the lengthy period of the model training process. Below is the code used to build a local cluster with 4GB memory allowance.

h2o.init(nthreads=-1, max_mem_size="4G").

Deep learning in this study has been initiated by the creation of a single thread, shown above. To avoid excess or entire use of memory, only 4GB of the total RAM has been allocated to the experiment. However, memory usage can be adjusted depending on the prevailing situation. A memory problem affecting one node in a cluster occurred during the experiment and thus highlighted that allocating a suitable and efficient memory size is always an issue in experiments like this one. To manage memory usage and eradicate the problem, the following solution was devised. After running each code, the machine ran *h2o.ls()* to pinpoint all temporary H2O objects that had been created when running the codes. As such, *h2o.ls()* could be used to delete the temporary objects and thus free up some memory. It however emerged that *h2o.ls()* required more clear specifications of the objects to be deleted. The code below was therefore applied to solve the problem.

h2o.removeAll()

The code was used to remove any old memory in case the cluster in question had been previously run. After running this, all temporary objects were cleared out, and checked with *h2o.ls()*. However, this didn't solve the root of the problem. When the data process was run on H2O cluster, java.exe took up about 4GB in memory, which is about the size of the dataset being uploaded onto the H2O cluster. There was an assumption that this memory would be freed up with *h2o.removeAll()*; but that was not the case. Java.exe still took up about 4GB in memory after all temporary objects were removed. All un-used objects in R were removed, by making sure that the unused object was no longer available, from R's perspective, without shutting down the session all together. There could be several factors which needed to be addressed to create a relation between removed objects in R, and instantly seeing memory going down during the Java process.

5.4.4 Training a Deep Neural Network Model and Creating some Base Scenarios (Default Models)

The syntax is quite similar to other machine learning algorithms within R. The main differences are the inputs for x as well as y that are essential in utilising the column numbers as identifiers. Nowadays, the classifier is fitted and run for several preliminary tests to get a grasp on how the model is actually doing when predicting creditability. This consequently brings out the need to run a variety of cross-validation methods. Cross-validation is a model evaluation approach that doesn't use traditional fitting measures (such as R2 of linear regression) during assessment of the model. Cross-validation is centred on the predictive ability of the model. The process follows the steps below:

1. Splitting the dataset into training and test sets.
2. Training the model using the training set.
3. Using the model on the test set.

It should be noted that running the operation only once cannot give reliable estimation of the model's overall performance. This is because the estimate has a non-neglectable variance. Consequently, a variant of the n-fold cross-validation, which gives better results in cases where the size of the dataset is limited, was used to better estimate the model's performance. If the datasets would have been large enough, a sizeable portion would have been set aside to be used to validate the model after it has been run on the

larger portion of the dataset by examining the resulting prediction error. Unfortunately, in most cases, more so in social sciences research, large datasets are not always available. The original dataset was saved in the temporary variable *temp=train*. The training dataset was distributed in three parts randomly; 0.75 into train, and the rest into test and predict. From the created *smp_size* (sample size), the range of data in the datasets was specified as follows:

```
smp_size <- floor(0.75 * nrow(train))
train_ind   <-   sample(seq_len(nrow(train)),   size   =
smp_size)
train <- train[train_ind, ] This is used for training
test <- train[-train_ind, ] This is used for test data
train=as.h2o(train)
```

Using the library, we created our training data as follows:

```
1: annmodel <- h2o.deeplearning(
2: x=predictors,
3: y=target,
4: training_frame=train,
5: hidden=c(100,80,45),
6: epochs=20,
7: nfolds=5,
8:  fold_assignment="Modulo"  # can  be  "AUTO",  "Modulo",
"Random" or "Stratified"
9: )
```

*Result: Neural Network -> 292.5256 errors (errors before 308.0929)

The above topology means that there are 100 neurons in the first hidden layer, 80 neurons in the second hidden layer and 45 neurons in the third hidden layer. Data have been fed to the deep learning module ANN with predictors, target and train data. Deep learning is based on a multi-layer feed forward artificial neural network that is trained with stochastic gradient descent using backpropagation. From the above result, it appears as if the data really lends itself to the Neural Network. The next section will seek to unveil whether this is truly the situation or whether the neural network default parameters were achieved by luck.

5.4.5 Testing the Model: Model Evaluation

The test is commenced by the addition of L1 and L2 as well as boosting the number of training rounds/epochs from 1 to 100. Consequently, the errors tend to reduce to 292.5256, which is quite sensible as the study's model has been made less prone to catching 'noise' and to facilitate better generalisation. Experimentation is conducted with a wide range of hidden layers and number of neurons to construct a list of potentials so that the chosen parameter was somewhere in the middle. The best classification obtained was from [128, 63, 32] neurons, which is a fairly complex

network. The model ended up having fewer errors, which were fewer still than on the validation. The *h2o.predict (...)* function will return the predicted label with the probabilities of all possible outcomes (or numeric outputs for regression problems), which is quite useful if more models are to be trained and an ensemble is to be built. In the codes below, data were predicted with predict function in H2O and *as.numeric* used to find impurity in the output. A data frame of the given prediction vector was created and a scatter plot of the data was drawn to show the pattern in passenger flight list (see Figure 4.11). In Figure 4.11 below, the actual total value of passengers is shown on the Y-axis while the X-axis shows the predicted values from the predictor. A mismatch of both the actual and the predicted values is shown at points 2000-2500 on the Y-axis. This can be confirmed by reviewing the last points at 0-500 on the Y-axis. Hence, the ANN needs to be trained more. Regarding the number of epochs, the neural network should iterate the best number from the possible states of the greedy search100 epochs. A dropout of 25% of data from original train file was performed in order to have a 5-fold cross validation with 75% data for train and 25% data for evaluation to have some initial intuition about how the tuned deep learning model is perform by plotting predicted values on the unseen test set against ground truth values. The predicted values on the unseen test set were then plotted against the true ground values and thus giving an insight on performance of the tuned deep learning model. The information above is illustrated in plot bellow Figure 4.11 followed by plot of the original distribution and the predicted distribution Figure 4.12 and Figure 4.13.

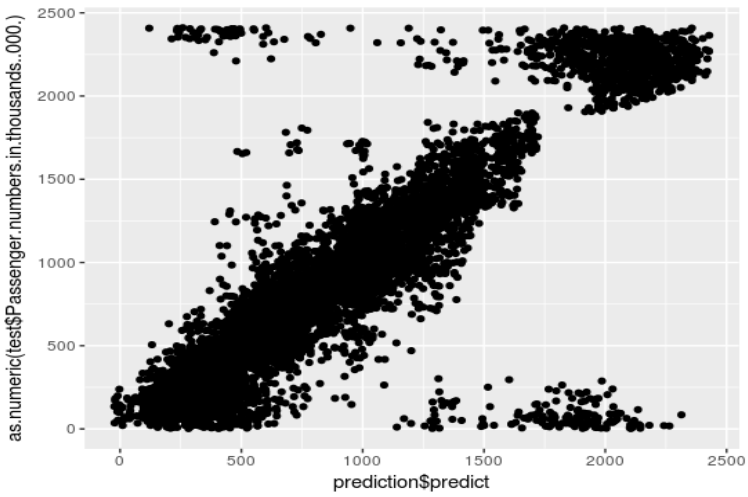

Figure 5-11 Predicted Values on the Unseen Test Set Against Ground Truth-Values.

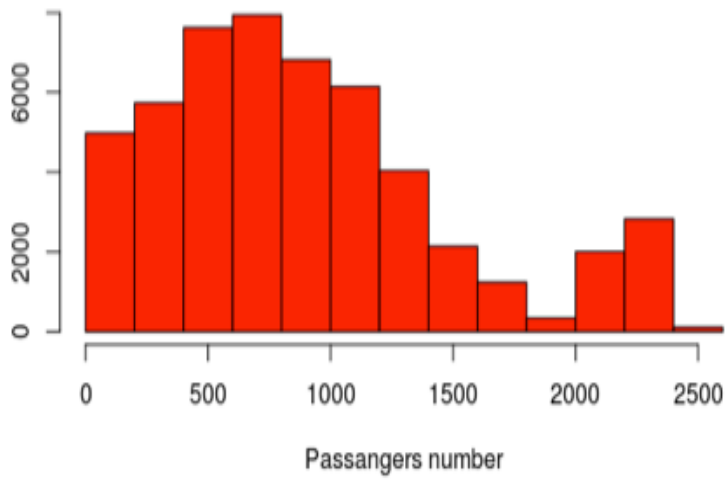

Figure 5-12 The original distribution of target.

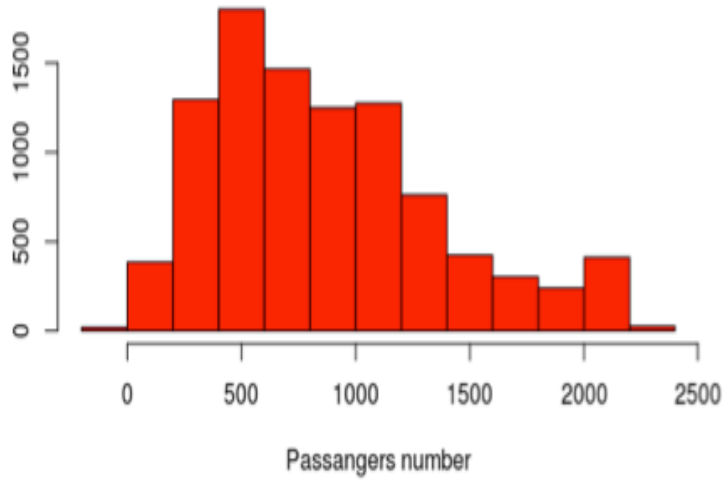

Figure 5-13 The distribution of predicted values.

5.5 Hyperparameter Tuning: Tuning with Grid Search and Random Hyperparameter Search

Fine tuning strategies primarily aim to extract essential features from unlabelled details, identifying and eliminating input redundancies, and protecting the vital data sections in discriminative and robust representations. In neural network architectures, deep hierarchies are built by stacking unsupervised layers on top of each other (Le, 2013). Upon giving input layer activations to the first layer, they are automatically fed to the other layers. The initial deep features training is conducted within an unsupervised layer and then fine-tuned into classifiers though back-propagation (Lloyd et al., 2014). These actions significantly improve the performance stability of the network (Bengio,

2013). A test set error of less than 10% can be obtained within one minute when fine tuning is conducted. With larger models, an error rate of even less than 5% can be achieved. For the dataset used in this study, deep learning methods are more effective than any other methods. This is because they directly partition the space into sectors which are highly required in the experiment. With the aim of this study being to identify parameters that can positively influence the accuracy of the model, hyperparameter tuning is an effective action. The first 10,000 features of the training dataset will be trained in the case of speed. In most occasions, random parameter search tends to be more productive than use of rigid search when searching for over four parameters. This therefore gives high possibilities of locating one of several excellent models. In this study, emphasis was put on tuning the network structure as well as the regularisation of parameters. The grid search was stopped as soon as the overall performance at the roof of the leader board ceased displaying any more changes, i.e., after the search has converged. The activation models used in the next model are rectifier, tanh, and maxout. For logistic performance, the Tanh function is shifted and rescaled. Its almost zero symmetry facilitates a quicker converging of the training algorithm. *library(metrics)* is used to fine tune implementation in this study; metrics is a set of evaluations that unsupervised machine learning employs on a large scale. In the case of artificial neural networks, the activation part is the rectifier which is considered the positive part of the network's argument (Lloyd et al., 2014) and is as follows:

$$f(x) = x^{+} = \max(0, x) \tag{5.1}$$

with X as an input to a neuron. Hahnloser et. al. (2000), driven by strong mathematical justifications as well as biological motivations, were the first researchers to introduce the activation network in a dynamical network. Its first demonstration was conducted in 2011 and it proved to facilitate better training in deep networks in comparison to other activation functions that were in use before then. As of 2017, the rectifier had boomed to become the most used activation function for deep neural networks due to its two main benefits, which are:

1. Its operations are relatively fast.
2. The gradient vanishing condition cannot affect it.

The rectifier is quick to compose as it does not involve any normalisation, nor does it require exponential computations. Nonetheless, a rectifier can be eliminated by a bigger

gradient, and in such a case, it can never be reactivated again. Application of maxout generalisation solves this problem by capping the weight data dot product and thus preventing it from being eliminated (Glorot & Bengio, 2011). The most recent studies on batch normalisation show that when normalisation is done with batchwise whitening as well as by re-scaling, most of the rectifier-related issues can be solved. Both re-scaling and normalisation with batchwise whitening are achievable through the inclusion of a linear layer before the activation functions (Sutskever et al., 2013). As such, any hyperparameters that are not critical are either turned off completely or set to zero in the early stages. In the event that the loss does not drop, tuning of the learning rate is initiated. Typically, the learning rate ranges between 1-1e-7. Each time, the rate should be dropped by a factor of 10 and then tested in short iterations while at the same time monitoring the loss closely. A consistent rise in the loss signifies a relatively high learning rate while failure of the loss to go down means that the learning rate is too low. The learning rate should therefore be adjusted accordingly until it flattens prematurely. Once stabilisation has been achieved in the model, further tuning can be conducted. From the experiment, the following hyperparameters turned out to be the most tuned:

- Learning rate.
- Mini-sized batch.
- All factors of regularisation.
- Hyperparameters that are layer-specific, such as the dropout.

Upon successful debugging of the models in this study, focus was shifted to model capacity and tuning. The next section is a discussion of how to improve the performance of a deep learning network and to tune deep learning hyperparameters.

5.5.1 Improving Deep Neural Network Model Performance using Hyperparameter Tuning

Gradual addition of layers and nodes into the deep network is conducted to increase its capacity. The tuning process is more empirical and less theoretical owing to the fact that more complex models are produced by deep layers. The main aim of gradual addition of layers and nodes is to overfit the model as regularisations can tune it down. The iterations are done repeatedly until the accuracy improvement is diminishing and thus the drops in the training and computation performance can no longer be justified. However, the maximum number of hidden nodes between any two affine layers is

restricted by the memory size. In very deep networks, it is observed that the gradient diminishing problem is more recurrent. Close monitoring should be conducted on the activation histogram after implementation of the activation functions. The gradient descent will be ineffective in the functions are in a relatively different scale. It will also be possible to further trace the problem if there is a high number of dead nodes present in the network. High prevalence of dead nodes can be caused by diminishing gradients, bugs, or weight initialisations. If none of these causes is evident in the deep network, advanced experimentation of the ReLU functions should be conducted.

Later, the network was trained to avoid any overfitting. This is achievable so long as the data in the study and the data in the initial dataset are kept relatively similar in context. As such, the pre-trained model will have learnt the features that are applicable to the classification condition of this study. Since the dataset in this study has a relatively similar context to the initial dataset that was used to train the pre-trained model, fine tuning must then be conducted. More features are captured in the first layer of a network that has been pre-trained with a diverse and large dataset. All the features captured are relevant and useful to all classification operations. By nature, if the dataset of the study belongs to some distinct domain, for example, health data, and if no pre-trained networks on that domain are available, the network will have to be trained afresh from the beginning. In situations where the dataset is relatively small, overfitting might occur when the pre-trained network is fine-tuned. This is more so the case when several layers in the network are entirely connected. As such, it should be noted that fine-tuning gives much better results when the dataset is relatively large.

The value of the raining error is based on the parameter *training_frame=train* which is a specification of the randomly selected training points that have already been sampled and are to be used for scoring. This study utilises a default 10,000 points. The validation error is based on the parameter *validation_frame=valid_frame*, from which the identical value on the validation set is regulated and subsequently set to be the whole validation set by default. When either of the parameters is set to zero, the whole corresponding dataset is instantly used for scoring. 'Target' and 'Train' sets with L1 and L2 for regularisation had already been fed into the deep learning module ANN with predictors. The foundation of Deep Learning is a multi-layer feed forward artificial neural network already trained with stochastic gradient descent with the use of back-propagation. Typically, the size of the batch is either 8, 16, 32, or 64. A small batch

size will not give a smooth gradient descent. On the other hand, a very high batch size will result in a longer completion time for one training iteration with relatively smaller returns as well.

In this study, each training iteration was taking too long and thus leading to lowering of the batch size. Close monitoring of the overall learning speed was done and when the oscillation became too much, it became clear that the prices have been pushed to the extreme. Below is the fine-tuning of the study with the resultant topology following a grid search hyperparameter optimisation.

```
Tuned Model of Deep Neural
          1: function Tuned Model(H2O)
          2: x=predictors,
          3: y=target,
          4: training_frame=train,
          5: hidden=c(128,63,32),
          6: epochs=100,
          7: nfolds=5,
          8: fold_assignment="Modulo",
          9:l1=5.6e-05,
          10:=7.4e-05,
          11:input_dropout_ratio=0.05
          12:) end function
```

The topology above illustrates a neural network consisting of 128 neurons in the first hidden layer, 63 neurons in next hidden layer and 32 neurons on third hidden layer. The total number of epochs, which show passes with data, per iteration on N nodes, is 100. To achieve a higher prediction accuracy, additional epochs will be utilised. This will, however, be subject to the affordability of the arising computation cost. A dropout rate of 20% to 50% was observed. Changing input dropout ratio rate improved the training dataset to 0.8560758. Changing it definitely changes the results. Once again start predicting with the help of data, which have already trained. This as.numeric function is used for finding impurity in output. The basic steps of improving deep neural network model performance using hyperparameter tuning are as follows:

5.5.1.1 Splitting the Data According to Initial Train Test Split (1-Fold Cross-Validation)

Half of the outlier labels in the training set were removed and subsequently considered as un-labelled data. In the set-up, these removals were also considered as contaminated in the normal class. However, no regularisation was applied to any of the methods up to that point. This plays a huge role in easing comparability of the results. It was critical that various techniques of unsupervised outlier detection be applied in feature

transformation. This was achieved by choosing an arbitrary subset of options and also the k-NN outlier, whereby the sum of distances of K from the nearest neighbours was computed. The basis of these functions is on a rough search for unsupervised algorithms, and to determine which function moderately affected the training set.

5.5.1.2 Getting the Best Three Features

Numerous decision trees are created in the random forest approach. With the use of the function *randomForest()* in the random forest package, random forests were created and analysed. In turn, they were used for prediction. As illustrated in Figures 4.14, a reduction in errors was realised when different features with ensemble functions were used. As such, using this technique proved to be a significant improvement over the ANN.

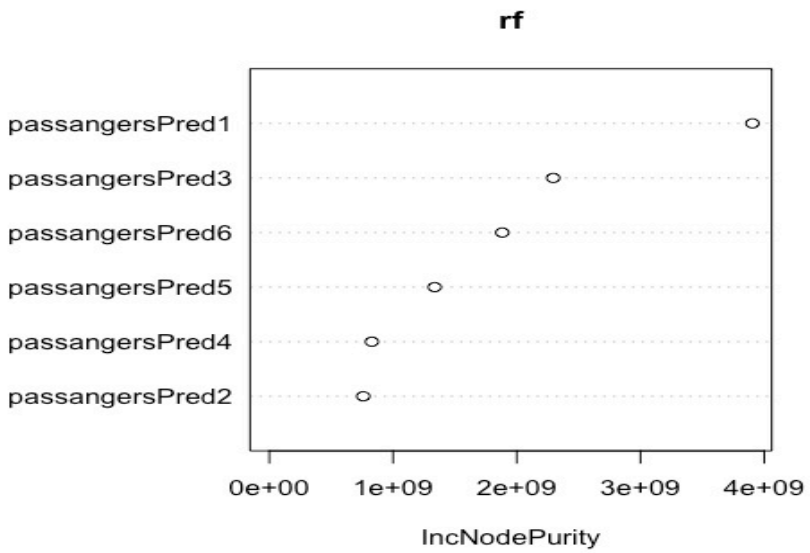

Figure 5-14 Variable importance on generated features.

The error is related to random forest importance error. Variable importance refers to how much a given model "uses" that variable to make accurate predictions. The more a model relies on a variable to make predictions, the more important it is for the model.

5.5.1.3 Features Obtained from Deep Learning Hidden Layers

The number of neurons in each hidden layer was similar to the number of features extracted from the deep learning model. To ensure that only the important non-linear features were selected, a correlation of all features corresponding to a neuron in the hidden layer containing the target variable was computed. Only one feature per neuron in the newly computed correlations was kept. Subsequently, the final four non-linear

features were achieved by comparing the maximum correlations with other features from the same hidden layer. Figure 4.15 shows a hierarchy of importance of features that play a vital role in the prediction of passenger figures. It is clearly shown that non-linear features attained through the method outlined above are *deepf1*, *deepf2*, *deepf3*, and *deepf4*, and they have the highest influence in the prediction, as captured by XGBOOST features importance.

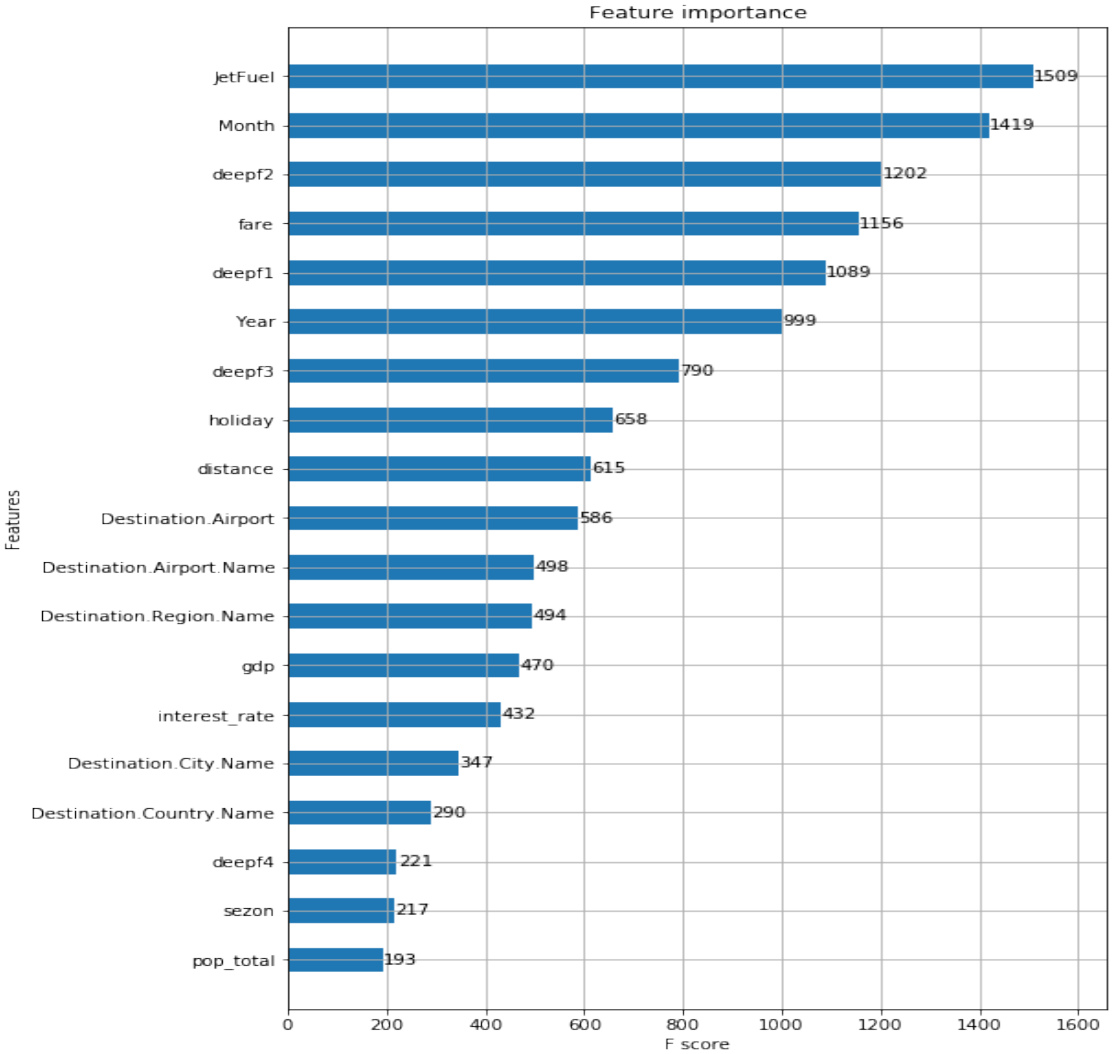

Figure 5-15 Features importance obtained from deep learning hidden layers.

5.5.2 Extra Grid Search to Optimise Parameters

To enable comparison of all models' performances, and consequently tune the hyperparameter values, all possible models attained through combination of hyperparameter sets were trained using a grid search model. During the process, a strong interrelation of some of the hyperparameters was observed. As such, tuning

should therefore be conducted using a mesh of possible combinations on a logarithmic scale. For two hyperparameters, λ and γ, for example, the initial values can first be corresponded and then dropped by a factor of 10 in each step. A random slight shift of the points is done rather than to use the exact cross points. As a result of this random shifting, properties that could have remained hidden were discovered. The optimal point was further retested within the border region if it was found to be lying in the border of the mesh. Due to the computational intensity of a grid search, it should be used sporadically in smaller projects. The results were fine-tuned by lengthening iterations and dropping the values by a factor of 10 or lower. Poor performance on held-out test data is realised when a large-sized feed-forward neural network is trained on a small training set. To lower this 'overfitting', half of the characteristic detectors were randomly omitted during each training session. This consequently halts the complex co-adaptation whereby a function is only useful in the presence of other particular element detectors. Each neuron therefore learns how to identify a feature, an act that significantly aids in coming up with the suitable answer. This was performed in this study and maintained as long as the inner combinatorial large variety remains operational. Random grid searches are reliable when locating the greatest parameters. As such, another grid search was run on the model, as it had given perfect scores at the beginning. This new grid search would unveil its relevance and whether it can overtake the NN. The randomly created frame was run on the codes below.

```
valid_frame=as.h2o(test)
dl_random_grid <- h2o.grid(
algorithm="deeplearning",
grid_id = "dl_grid_random",
training_frame=train,
validation_frame=valid_frame,
x=predictors,
y=target,
epochs=1,
stopping_metric="RMSE",
stopping_tolerance=1e-2,
```

Above process stops when log loss does not improve by >=1% for 2 scoring events

```
stopping_rounds=2,
score_validation_samples=10000,
```

Now we have down sample validation set for faster scoring

```
score_duty_cycle=0.025,
```

This is for not score more than 2.5% of the wall time max_w2=10

Both the accuracy and training speed of the model were significantly impacted by the values picked for the hyperparameters. A variety of hyperparameters, such as attempting different amounts of hidden layers as well as alternatively selecting layers, was tried, as *hyper_params = hyper_params, search_criteria = search_criteria*. The fact that a grid search gives a high predictive accuracy led to its selection in this study. Additionally, it allows specification of a variety of hyperparameter values as well as enabling a trial of all possible combinations. The *getrid()* feature can be successfully utilised to make every possible combination of variables in R. As such, the incremental outcomes become visible as the models are now being constructed by fetching the grid with the *h2o.getGrid* feature. The first element in *grid@summary_table* is taken into consideration.

4.5.2 Improving Deep Neural Network Model Performance Using Ensemble Learning

There are various types of learning paradigms. One that is commonly used is neural network ensemble learning, whereby higher generalisation abilities are a representation of several neural networks and thus outshine individual networks. Its applicability therefore explains its relevance for deep learning in multi-layer neural networks. Additionally, the overall performance of a conventional neural network ensemble can be successfully enhanced through various qualities of deep neural networks. Due to their efficiency in increasing the percentage of accuracy in a variety of operations, model ensembles are commonly used in DL competitions. In this Thesis, a new ensemble strategy is suggested, with which the overall performance of neural network classifiers will be enhanced. Predictions in ensemble models are based on the following:

1. Single vote for every model.
2. Weighted voted on the basis of models' prediction confidence level.

The suggested solution brings together a variety of neural network classifiers as well as another group of characteristics which are possibly feature engineered. In this study, it has been demonstrated that co-adapting features is more advantageous than individual optimisation of features for every distance metric. As such, enhancement of each feature set in the context of the whole ensemble is achieved. The solution also outlines how the ideal degrees of an ensemble can be obtained, and consequently how the model can be enhanced and optimised. When compared with other commonly used classifiers,

the results of a DF-TS-INN classifier indicate a significant improvement in performance.

The aim of this Thesis was to evaluate and retain the method used herein. Getting rid of the intense values benefited each training session on the model and also facilitated the datasets' validation. It should be noted that a big leap in the development of the model would be realised even by conducting some metric enhancements.

5.6 Discussion

Predicting airline passenger figures accurately is an integral part of successful and efficient management practices in the airline industry. This chapter has therefore presented an overview of all the features that impact the forecasting of airline passenger figures. As demonstrated by the experiments herein, is important to select closely related features so as to achieve an accurate forecast of airline passenger figures. It has also been demonstrated that the addition of new features to the old ones alters the variance results in the new output, which improves the forecasting results. The results were also significantly improved when the deep neural network was optimised. This study also proposes the H2O method of tuning the deep learning models. In this method, diversification of samples is achieved by the building of a trained network from the samples of features selected from the dataset.

As manifested in the results, the final predictions obtained from the test set are significantly better than the predictions of the training dataset. This became clearly visible when the training set histogram was reassembled into a better histogram of the test set final predictions. It is noted that regardless of the combination method for different machine learning models, the accuracy of results from a deep neural network model is higher than that of results from individual machine learning models.

Feature importance of the original feature space formed the basis of parameterisation of the predictions. A fixed rule was employed to combine the predictions emanating from different time series models. Calculations of weighing parameters that are to be adopted while forecasting on holiday and season were done using the fixed rule. The calculations are, however, dependent on the number of months, destination, seasonality, and holidays. Although the rules have significantly optimised the performance, the study intends to scrap the fixed rules and instead adopt a dynamic combination

approach. Through such a combination, the basic features used will be adjusted while at the same time adapted to any newly introduced features.

When building machine learning classification models, feature selection and extraction are essential and inevitable phases. They also significantly impact the model's overall accuracy and performance. However, these phases are expensive and there is no guarantee that features extracted manually will generalise effectively in various data modalities. Later, the building of a hybrid classification scheme facilitated exploration of the advantages derivable from combining traditional machine learning models with deep learning models. For feature selection and extraction, the hybrid classification scheme utilises the first few layers of a convolutional neural network. Classification was then performed by feeding the extracted features into a traditional learning algorithm that was supervised. From the experimental results derived, it is clear that the hybrid approach outshines both the deep learning and traditional machine learning algorithms in the isolation of small data.

Several approaches suitable for the derivation of feature importance from various machine learning models such as random forest have been demonstrated in this chapter. It is as important to understand the results as it is to have good results. As such, every scientist should strive to understand the variables that are important in their models and why. This can not only give a better understanding of business operations but also pave the way for further improvements to their models.

All codes used in this chapter are shown in the Appendices section.

Chapter 5: Creating a Modified Version of Principal Component Analysis (PCA) to improve the Forecasting Performance Using a Different Correlation Matrix.

6.1 Introduction

In time series analysis, forecasting future observations are a key focus for economists and statisticians (Koopman & Ooms 2006; Heij et al. 2006). Over the past decade, the increased use of a variety of predictors in forecasting air passenger figures has in turn led to increased accuracy in the resulting forecasts. There are various variables that are used to forecast air traffic numbers, such as price index and employment rate. Such variables usually entail datasets with large numbers of time series observations that have been made over a long period of time (Vasigh & Fleming, 2016). When conducting the actual analysis, all variables play a significant role. As such, considering them or ignoring them would have an impact on the forecast accuracy and could result to suboptimal results. It is therefore a matter of great significance to determine the right variables to be considered in an analysis. Researchers, therefore, employ various data analysis statistical methods to determine the most suitable information from the available predictors to ensure that both theoretical and empirical performance of the forecasts have been improved (Hassanien & Gaber, 2017).

Principal Component Analysis (PCA), a commonly used method, is one of the many techniques have been coined to counter samples' errors. One of the primary goals of PCA is identifying patterns in data, an act that is only achievable when variables show a strong correlation and consequently making it possible for the dimensionality of data to be reduced (Mirkin, 2011). While reducing the dataset dimensionality, PCA ensures that as much statistical information or variability is preserved. In other words, PCA enables derivation of a smaller dimensional subspace by determining maximum variance directions in high dimensional data while at time retaining much of the original information (Mirkin, 2011). As such, it is certain that the overall prediction

accuracy for air passenger variables and figures can be greatly improved by utilizing factor based forecasting with PCA estimation to extract small numbers of latent factors (Anderson, 1984).

Further on, regression analysis tends to be easier when PCA is incorporated into it (Sapankevych & Sankar 2009). Basically, the training dataset is used to learn an appropriate estimator function that is to be used in generating new outputs from the corresponding inputs (Adhikari & Agrawal 2013). Although there are various disciplines where PCA application exists, a correlation matrix is currently used to implement PCA. Such a matrix is derived using the Pearson Correlation Coefficient to correlate variables in a pairwise orientation. At times, however, data properties outside the linear relation may not be captured by the Pearson correlation coefficient (Jackson, 2012). This chapter therefore aims to bring forth a new, more reliable and effective forecasting method based on PCA. A Nonlinear Principle Component Analysis (MPCA) modified version that will use kinetic energy will be introduced and applied to the study's dataset with samples of selected features being used to compute the transformational matrix. The MPCA simplifies the PCA by not computing a multivariable quadratic optimization problem displaying any linear constraints (Adhikari & Agrawal 2013).

The main aim of this chapter is to present the most appropriate approach for long term forecasting at Muscat Airport. As such, the performance of three forecasting approaches are evaluated. The first approach is the PCA model linear forecasting method. The second approach entails application of machine learning in the form of MPCA as a nonlinear forecasting technique. In the third approach, the outputs of the hybrid MPCA are leveraged when a curve is fitted into the trend component and extrapolated to the future. The final forecast is then obtained by superimposing the MPCA-forecasted seasonal component into the trend component. Rather than repetitive fitting of a model, this approach entails projecting a fitted curve into the future. In this chapter, therefore, an analysis of the implications of MPCA on forecasting accuracy, as well as, a combined Regression-PCA analysis for airline passenger figures are presented. Monthly statistics of passengers traveling via Muscat Airport are used in this study. Since aged data might exhibit unwanted characteristics, data used in the study was for the period January 1998 to December 2016. In total, 51,983 observations were collected and divided into two; the Training set and the Test set. Model parameters

were obtained from the Training set while performance evaluation as well as determination of forecasting accuracy was done using the Test set.

Univariate autoregressive models and multivariate vector autoregressive models are some of the commonly used analysis methods (Lütkepohl, 2013). When handling many time series, most of the models have many limitations. In addition to the developmental challenge as discussed in Chapter 4, it is also difficult to measure performance and draw conclusions for such models. Since there are numerous and varying modelling assumptions on which derivations and relative results are based on, it becomes difficult to compare the models theoretically (Lütkepohl, 2013). It is, for example, hard to determine whether the entire processes or algorithms are affected by assumptions. Empirically, applicability and variability of datasets could be another challenge. As such, the data requirements, specifications, implementation, forecast horizons and areas among other aspects of a research are always faced by frailness and lack of clarity (Lütkepohl, 2013).

Many studies have been conducted and consequently provide a variety of methods to counter the challenges of low dimensional data and, as a result, improve forecasting performance. There are three classes of methods that have been derived (Stock & Watson, 2004). First, there is the class that combines a large cluster of forecasts from relatively simple models, whereby; multivariate models form the foundation of computing individual forecasts from which a combination of the forecasts is produced entailing reasonable weights that are based on some historical measures. The Theory of forecast combinations was first proposed by Bates and Granger, (1969) and over time, its application has successfully borne improvements in forecasting results (Timmerman, 2003; Stock & Watson, 2004). In the second class, several latent factors employ factor analyses to cover information in the predictors subsequently providing factor augmented forecasts. In the third class, forecast accuracy is improved by reducing the samples' errors.

Referring back to Chapter 4, this chapter employed traditional tools with a variety of features such as monthly national accounts to assess autoregressive models in the short run, forecasting, as well as, economic analysis. It was shown clearly that the variance results of the new outputs were significantly altered by addition of new features to the old outputs and thus a better forecasting performance was obtained in comparison to

more complicated econometric models. With the aim of analysing the performance of complicated models, MPCA is applied in this chapter is based on Chapter 4's results. Consequently, a forecast of the monthly year-on-year airline passengers' figures' growth in Oman is conducted. Based on Principle component Analysis, autoregressive and MPCA estimations are used as they will enable extraction of only the common trend by removal of noise factor from variable shocks. A comparison of the forecasts from the old PCA and the newly obtained ones is then made.

In the next section, PCA is deeply reviewed and a definition given. Additionally, the basic outline of the MPCA framework is given and a comparison made with the old PCA. Section 5.2 will bring forth the proposed methodology while section 5.3 will show an implementation of the model. The experimental results derived from the model will be reported in section 5.4. A new ensample method will be discussed in section 5.5 while section 5.6 will bear the conclusion.

6.1.1 Presentation of Principle Component Analysis: Review of the PCA

First introduced by Karl Pearson, Principal Component Analysis, commonly known as PCA, is a classical multivariate technique of conducting data analysis. It is commonly used to extract features linearly as well as, a wide range of data dimension reduction (Bengio, 2013). It calculates the eigenvectors of the covariance matrix in high dimensional original inputs to come up with smaller numbers of uncorrelated inputs. Due to its high ability to correlate properties and reduce errors, PCA has been used to prepare data in various areas of information processing. Literature has it that much attention has been drawn to PCA. The first contribution of PCA to probability, for example, is attributed to Onicescu and Mihoc, in 1996 and consequently became a viable solution to the forecasting challenge of having too many predictors all which are potentially useful. In the past, PCA has been applied to dimensionality decrease in cases of enormous data, where the data has been correlated by numerous correlation metrics, or when deriving new features that were non-existent e.g. by addition of 2, 3…5 features to the initial CSV file to consequently achieve Eigenvalue or Eigen-features. Fewer features are achieved by compressing much of the information in the initial data space. PCA achieves this by maximizing the variance in any potential subspace (Timmerman, 2003) whereby spanning of the PCA subspace through eigenvectors in line with the sample covariance matrix's top eigenvalues occurs.

It is also possible for both unsupervised and supervised learning and recognition processes to adopt PCA in their data preparation phases (Turk & Pentland, 1991). However, since in most times data available is usually in nonlinear form, PCA strategies might not give desirable classification benefits. On the other side, it is possible for MPCA and its variants to work effectively with nonlinear relations and thus provide the advantage of working efficiently with nonlinear data (Maaten, & Hinton 2008). It should be noted that PCA operates by finding vectors that are highly indicative, that is, eigenvectors that corresponds to the best eigenvalues in a covariance matrix sample. As such, PCA is a mathematical process with high levels of effectiveness in transforming selected variables with a correlation possibility into principal components which are a smaller selection of variables that are not correlated (Wensveen 2007; Coates & Ng 2012; Babu & Reddy 2014; Phyoe et al. 2016).

In the modern age, data sets are often too large and thus pose a challenge to computer hardware in addition to slowing down the overall performance of various machine learning algorithms (Hastie et al., 2013). It should be kept in mind that the main aim of PCA is to conduct a procedure to reduce the variables and in turn attain a small number of factors that take into account variations from a large set of observation variables. The attained small number of factors consequently elevate the accuracy in entire prediction models. A modified PCA version is proposed in this study to make airline passenger figures more accurate. This is done having it in mind that investment decisions can be significantly influenced even by the slightest performance improvement. Input features in form of economic indicators have been calculated using time series analysis, given its crucial role in forecasting activities. PCA is then applied to input features to extract influential components which upon filtration, they facilitate transformation of high dimensional inputs into low dimensional features that are easily applicable. Later on, the reduced features are converted into principal components using MPCA which then constructs the forecasting model using the low dimensional input variables obtained earlier on to predict the final airline passenger figures.

6.1.2 Modified Principal Component Analysis Framework

Application of various feature extraction methods in data mining has been witnessed over the past few decades. The performance of machine learning algorithm on data is therefore largely impacted by the efficiency and effectiveness of the data extraction method used (Perner, 2012). Even an advanced and highly complex machine learning

algorithm is likely to exhibit poor performance with the application of poor feature extraction. On the other hand, a simple machine learner will show high performance levels when a good feature extraction method is applied. At high levels of abstraction, PCA has shown the possibilities of giving better results in automated extraction of data features. Machine learning would achieve a major breakthrough upon successful performance of feature engineering in a more automated manner. As such researchers would be able to extract features automatically without having to exercise direct human input. In a study conducted by Yang, et al. (2004), 2-D PCA was proposed with which there would be no need of converting images into vectors.

According to Plataniotis & Venetsanopoulos (2008), the effectiveness of 2-D PCA is relatively high as compared to PCA, and it can therefore be considered as a special multi-linear PCA. Recently, L1_norm-based PCA that are robust to outliers have been suggested by researchers. They include; Sparse PCA by Meng et al. (2012), 2DPCA-LI by Pang & Yuan (2010) (Li, 2010), PCA-LI by Kwak, (2008) (Kwak, 2008) and Robust PCA by Candes, (2011). However, desirable classification results may not be achieved from most of the aforementioned approaches when dealing with nonlinear data. However, nonlinear PCA models and their variants with the ability to deal with nonlinear data available in the real world have been proposed in various studies. They are; Nonlinear PCA by Wang & Tang (2004); Schölkopf et al. (1998) and the Kernel PCA (KPCA) by Huang et al. (2009). Other extension of the PCA that are based on measurement of probabilistic similarity are Probabilistic PCA (PPCA) by Tipping &Bishop (1999), Bilinear PPCA by Zhao (2012), Independent Component Analysis (ICA) by Koldovsky et al. (2006), and the Half-Quadratic PCA by Zheng et al. (2011).

PCA is a key method of reducing dimensionality and extracting features. When investigating PCA based methods, the main focus point of researchers is on vectors that are highly representative. Such researchers barely show interest on discriminative vectors. With a chief aim of improving PCA classification performance, this study will propose a novel PCA with the use of different correlation matrix and as well modify the Pearson correlation PCA. While doing so, Information correlation Coefficients as well as Kinetic Energy methods will be employed. This framework will have a more powerful data representation ability and at the same time give better classification results. Owing to the fact that each of above mentioned PCA methods are based on only one measurement, i.e., the distance or probability measurement. Although it is easy to

implement these PCA algorithms, the classification results of them are usually not very good (Phyoe et al. 2016). As such, the non-linear relations from various classes may not be captured by these methods. The modified PCA should therefore provide a solution to this challenge. Consequently, the following ideal properties emanate from the application of Kinetic energy metrics by MPCA;

1. Contrary to use of only one measurement by other PCA methods, MPCA employs multiple measurements and thus gaining a high potential to capture sufficient non-linear relations from the data at hand. Additionally, it has a high capability of data representation.
2. In cases where data dimensionality is larger than number of training samples, MPCA is highly applicable.
3. It is possible to theoretically apply MPCA to both un-supervised and supervised learning models.

Moreover, other forms of MPCA similarity measurements also exists. The most common one is Mahalanobis distance which is a highly effective and efficient multivariate distance metric used to measure the distance between a distribution and a given point. Its applicability in detecting multivariate anomaly, classifying datasets that are highly imbalanced and one-class classification makes it highly useful (Hillel et al. 2005).

In a training set with N centred training samples $xi \in RM(i = 1,2,\ldots,N)$, the covariance matrix would be determined by (Hoffman et al. 2013);

$$C = \frac{1}{N}\sum_{i=1}^{N} x_i x_i^T = \frac{1}{N} X X^T \qquad \textbf{6.1}$$

Whereby; $X = [x_1, x_2 \ldots x_N]$. in the event where the dimensionality of the above covariant matrix C displays high substantiality, which in most cases is usually $M\ N$, the Eigen decomposition of C becomes hard and most likely infeasible and thus giving rise of the need to explain a new matrix (Hoffman et al. 2013).

$$D = \frac{X^T X}{N} \qquad \textbf{6.2}$$

It is clear that matrices C and D have a similar non-zero Eigenvalues that are denoted by $\lambda_i (i = 1,2,\ldots,r)$. The Eigenvectors affiliated to the matrix D nonzero eigenvalues are

denoted by $v_i(i = 1,2,...,r)$. As such, covariance matrix C Eigenvalues corresponds to the following Eigenvectors;

$$u_i = \frac{Xv_i}{\sqrt{\lambda_i}}, \qquad (i = 1,2, \dots, r) \tag{6.3}$$

When a pattern recognition is fed into the PCA, ui(i = 1,2,...,r) are labelled as Eigen-pattern while the subspace spanned by ui(i = 1,2,...,r) is termed as an Eigen-space. The modified version in this study takes into account the fact that the PCA method has employed a different correlation matrix. The main limitation of its implementation is that data properties beyond linear relations are not captured. If, for example, a researcher is correlating two random vectors using Pearson Coefficient; $x = \{-4,-3,-2,-1,0\}, y = x^2 \Rightarrow Cor(x,y) = 0$, this result would be wrong since the non-linear relation existing between the two vectors, resulting from the functional transformation $(x)^2 \rightarrow (y)$ is not captured. The correlation is therefore not Zero and a modified PCA has therefore been introduced in this study to improve this (Hoffman et al. 2013).

6.2 Methodology Framework

6.2.1 Features Based on Information Energy (Kinetic Energy)

Information Energy was first introduced in in the early 18th Century by Gottfried Leibniz and Johann Bernoulli, and a study was later conducted by Gustave Coriolis in 1829 to unveil its potential applicability. It is basically an application of the Kinetic Energy used in Physics to probability, and its definition is as follows with a countable sets of states (Nowikow et al. 2001);

$$X_1, x_2 \dots x_n, . \tag{6.4}$$

The corresponding probabilities are as follows;

$$p = (p_1, p_2, \dots, p_n), IE(p_1, p_2, \dots, p_n) = \sum_{i=1}^{n} p^{i^2} \tag{6.5}$$

From the above, in an experiment with **n** outcomes with the probability 1/n being common among all outcomes, the information energy IE=1/n. the Information Energy will have a maximum value of IE=1 and the probability for every outcome will be 1

whenever the experiment gives similar results. This means that a decrease in randomness will automatically cause an increase in Information Energy.

It is like reverse of Shannon entropy used to determine uncertainty by measuring information bits. Although it is also an Entropy, it should best be considered as 1/2∗m∗v2 of a random vector. The kinetic energy method is simple but powerful and thus provides quite reliable accuracy enhancement as well as in improving methods of machine learning on raw data. This is more so the case in the event where categorical data exists in groups and such data could be possibly discretized even when it is in in a continuous form. Additionally, if the corresponding probabilities of a random vector X are $p_1, p_2, ..., p_n$, the kinetic energy would be equal to (pi^2). This approach in probability is similar to physics only that (1/2)∗ does not prevail here. The probabilities are assumed to be floating, either because their total masses sums to 1, which is quite un-credible, or because they have a mass of 0 by them being considered as floating inertial masses with no mass. Based on Quantum physics, this can be explained as follows; if for example a random variable X=1,1,1,3,5,3, computation of kinetic energy will be;

The probabilities of the three random vector categories (Nowikow et al. 2001), which are 1, 3 5 are

a) Prob (1) = 3/6 =1/2
b) Prob (3) = 2/6 =1/3
c) Prob (5) =1/6

$$KineticEnergy(X) = "Sumofsquaredprobabilities" = prob(1)^2 + prob(3)^2 + prob(5)^2 = (1/2)^2 + (1/3)^2 + (1/6)^2 = 0.3888.$$

It should also be noted that;

If $$X = \{1,1,1,...,1\}, KineticEnergy(X) = Sum(prob(1)^2) = 1 \quad \textbf{6.6}$$

It therefore means that at a maximum value of 1, kinetic energy remains perfect so long as there is no diversity (Nowikow et al. 2001). On the other hand, an analogy made with atomic nuclei causes release of energy in large amounts in an occurrence known as nuclear fusion as shown below;

$$X = 1,2,3,4,.... + X_n KineticEnergy(X) = 0 \quad 6.7$$

As such, Kinetic energy decreases to zero or near zero whenever a high level of uncertainty or maximum level of diversity prevails. In such a case, random vector categories are said to be emanating from the atomic nuclei as shown in the previous example. This results to high energy that later on translates into low atomic numbers but an eventual low energy. Kinetic energy is bound between 0 and 1 as illustrated in the two examples above. Pattern recognition in MPCA will be improved if a new factor with the ability to measure the properties of random vectors is introduced.

6.2.2 Features Based on Information Correlation Coefficient

First introduced by Linfoot (1957), the Informational Correlation Coefficient (ICC) is also referred to as Onicescu's correlation coefficient (Iosifescu, 1986). At the bivariate normal distribution, it is equal to the classical Pearson's Correlation Coefficient. It is, additionally, a jont probability density distribution function of vectors X and Y. The ICC can be described as follows in a case with random vectors x and y;

$$O(x, y) = \frac{\sum_k p^{(Pk)*p(Qk)}}{\sqrt{IE(P) * IE(Q)}} \quad 6.8$$

In this study, the above function only applies for discrete data.

Only the linear properties of the manifold in which this study's data lies are captured by the Pearson Correlation. Taking a random vector in R, for example, x=c(-4,-3,-2,-1,0,1,2,3,4) and y=x^2, a 0 correlation will be yielded by Spearman or Pearson while in the real sense, the actual correlation is 0.5 due to the transformation function x->^2. In this study therefore, a new correlation coefficient has been applied in the modified PCA instead of fitness functions in genetic algorithms or cross entropy in neural networks. Formerly, decreasing large datasets that are correlated by a number of correlation metrics or deriving new features by addition were the main uses of PCA. On the other hand, determination of eigenvectors or eigenvalues was achieved through covariance matrix or Pearson Correlation.

With the new correlation matrix, a modified version of the original algorithm has been created by implementing the totally different correlation metric possessing the ability to capture the kinetic properties of two random vectors. The new correlation metrics were

hence implemented with the main aim being modification of the original PCA to derive a version that could obtain eigenvalues and eigenvectors that could in turn enable reduction of dimensionality by application of a correlation matrix possessing the Kinetic Correlation coefficients. The modified PCA was implemented and tested using the airline dataset by addition of two new features in the neural network. As it will be described later on in this chapter, the results of the implementation showed a boost of the final score by several hundred positions.

6.3 The Working Algorithm of the Study: The Modified PCA Implementation

First, a new correlation Coefficient known as "O" was developed. This correlation uses kinetic energy to select features. With the modified PCA developed in this study, the new Octave correlation in the PCA can have positive impacts to science when compared to the Pearson Correlation that is commonly used and only measures linear relationship. Additionally, measures linear relationship. There is also distance correlation that first introduced in 2005 by (Gábor J Székely, 2005) and is used to measure dependence levels between random vectors capturing non-linear relations. As such, there are multiple kinetic properties that could exist in a data such as number of 1s in a row, non-linear shape or linear shape. Based on the Kinetic properties, features are derived from the modified PCA by use of kinetic correlation metrics rather than by use of Pearson Correlation Coefficient. "Pandas" and "NumPy" which are the two most essential libraries are imported to enable numerical analysis as well as data manipulation.

Feature selection is subsequently conducted using the correlation that have been explained using two random vectors which are $X=x1,x2,..xN$ and $Y=y1,y2,...yN$. There are two main coefficients as explained below.

The Informational Correlation (IC). This type of correlation measures the merits of a predicted value and does not use kinetic energy. The IC improves the predictive skills of a forecasting analyst when used as a performance metric to forecast airline passenger figures. The IC is similar to correlation in the fact that it can measure the linear relationship between two random variables which in this case are the actualized numbers and the predicted figures. IC coefficients ranges from 0 to 1 where a perfect linear relationship is denoted by 1 and thus

showing good forecasting skills whereas inexistence of a linear relationship between the actual values and the predicted values is denoted by 0 and thus showing poor forecasting skills (Kent & Williams, 1995). Below is the IC function;

$$C(x,y)=Cx1,x2,\ldots xN,y1,y2,\ldots,yN=\sum_{i=0}^{N} p(xi)*p(yi) \quad 6.9$$

The Informational Correlation Coefficient (ICC). ICC measures ratings or measurement reliability for data that has already been sorted into groups or collected as groups through a process known as clustering. Pearson Correlation also known as interclass correlation is a term related to ICC and is commonly used unlike other statistics such as the Cohen's kappa that is rarely used. Pearson is commonly used in cases where there are only one or two meaningful pairs, from one or two systems to improve inter-rater reliability. The ICC could as well suffer from kinetic energy usage in information just like other statistical correlation coefficients (Kent & Williams, 1995) would and consequently resulting to the following;

$$O(X,Y)=(\sum_{i=0}^{N} p(xi)*p(yi))/(\sum_{i=0}^{N} xi2*\sum_{i=0}^{N} yi2) \quad 6.10$$

It should be noted that kinetic energy forms the denominator and thus the ICC coefficient (Kent & Williams, 1995) could be presented as;

$$O(X,Y)=IC/kinetic(X)*kinetic(Y) \quad 6.11$$

In order to master the dot product computation techniques explained here in a mathematical field, it should be ensured that the cardinality of unique events also known as classes of the two random vectors is similar as that of the set. It should be noted that existence of different shapes will disable the functionality of the correlation coefficient. IC for two random variables referred to as the dot product of probabilities corresponding to each class is returned as illustrated in code 5.3 below.

```
def ic(vector1,vector2):
a=vector1
b=vector2
prob1=np.unique(a,return_counts=True)[1]/a.shape[0]
prob2=np.unique(b,return_counts=True)[1]/b.shape[0]
```

```
p1=list(prob1)
p2=list(prob2)
diff=len(p1)-len(p2)
if diff>0:
for elem in range(diff):
p2.append(0)
if diff<0:
for elem in range((diff*-1)):
p1.append(0)
ic=np.dot(np.array(p1),np.array(p2))
return ic
```

With IC and kinetic energy of a vector functions available, a new function with the ability to compute kinetic correlation is defined. Consequently, kinetic energy-based correlation will be returned as illustrated below;

```
def o(vector1,vector2):
i_c=ic(vector1,vector2)

o=i_c/np.sqrt(kin_energy(vector1)*kin_energy(vector2))
return o
```

An update of the formula is done to add square root to the denominator and consequently bound the probabilities between 0 and 1. The number of items in the NumPy array will be returned by SHAPE function in form of a tuple. The function will additionally create a new matrix whereby the number of rows will have been initialized with zero values as shown below.

```
rows=data.shape[1]
rows
matrix= np.zeros((rows,rows))
```

Further on, the o() function that had been previously defined will facilitate creation of the correlation matrix as illustrated in Table 5.1. For comparison purposes, Table 5.2 illustrates the correlation matrix derived through the Pearson method.

Table 6-1 Correlation matrix with the function o().

	0	1	2	3	4	5
0	1.000000	1.000000	0.974678	0.326396	0.184695	0.229420
1	1.000000	1.000000	0.974678	0.326402	0.184695	0.229420
2	0.974678	0.974678	1.000000	0.320928	0.180018	0.223611
3	0.326396	0.326402	0.320928	1.000000	0.070861	0.131047
4	0.184695	0.184695	0.180018	0.070861	1.000000	0.490760
5	0.229420	0.229420	0.223611	0.131047	0.490760	1.000000

Subsequently, the 'scipy' library is used to derive correlation matrices on the basis of Pearson r' models which is illustrated below;

Table 6-2 Correlation matrix correlation matrix on basis of 'Pearson r' model.

	0	1	2	3	4	5
0	1.000000	0.751037	0.770805	-0.041212	-2.751721e-02	-1.111287e-05
1	0.751037	1.000000	0.959409	-0.013875	-3.120134e-02	-1.200190e-05
2	0.770805	0.959409	1.000000	-0.020539	-2.392666e-02	-1.213799e-05
3	-0.041212	-0.013875	-0.020539	1.000000	2.169034e-01	2.223767e-05
4	-0.027517	-0.031201	-0.023927	0.216903	1.000000e+00	8,876347e-07
5	-0.000011	-0.000012	-0.000012	0.000022	8.876374e-07	1.000000e+00

6.4 Experimental Analysis / Performance Evaluation

This section aims to evaluate the PCA algorithm by removing all features that are redundant and irrelevant. In this study's experiments, comparisons were made between kinetic correlation matrix derived from kinetic energy and the modified PCA as well as between Pearson R Correlation and PCA. Comparison of these methods gave the results explained henceforth;

6.4.1 Comparison of Modified PCA with Kinetic Correlation Matrix from Kinetic Energy and PCA with Pearson R Correlation

The main activity undertaken here involved replacing the Pearson r-based correlation matrix, or even the covariance in some instances, with the Onicescu correlation coefficient-based correlation matrix. Figures 5.1 and 5.2 clearly illustrate the results obtained after Pearson Coefficient was used to test the kinetic correlation of the study's data.

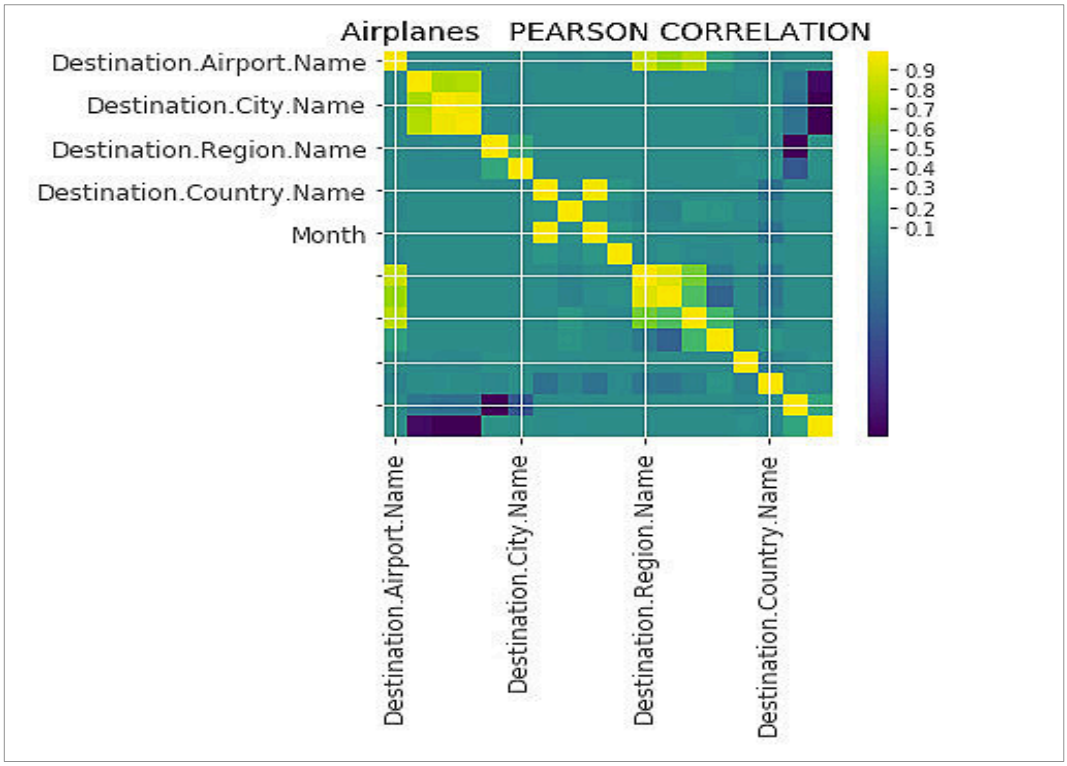

Figure 6-1 Air passenger numbers data with Pearson Correlation.

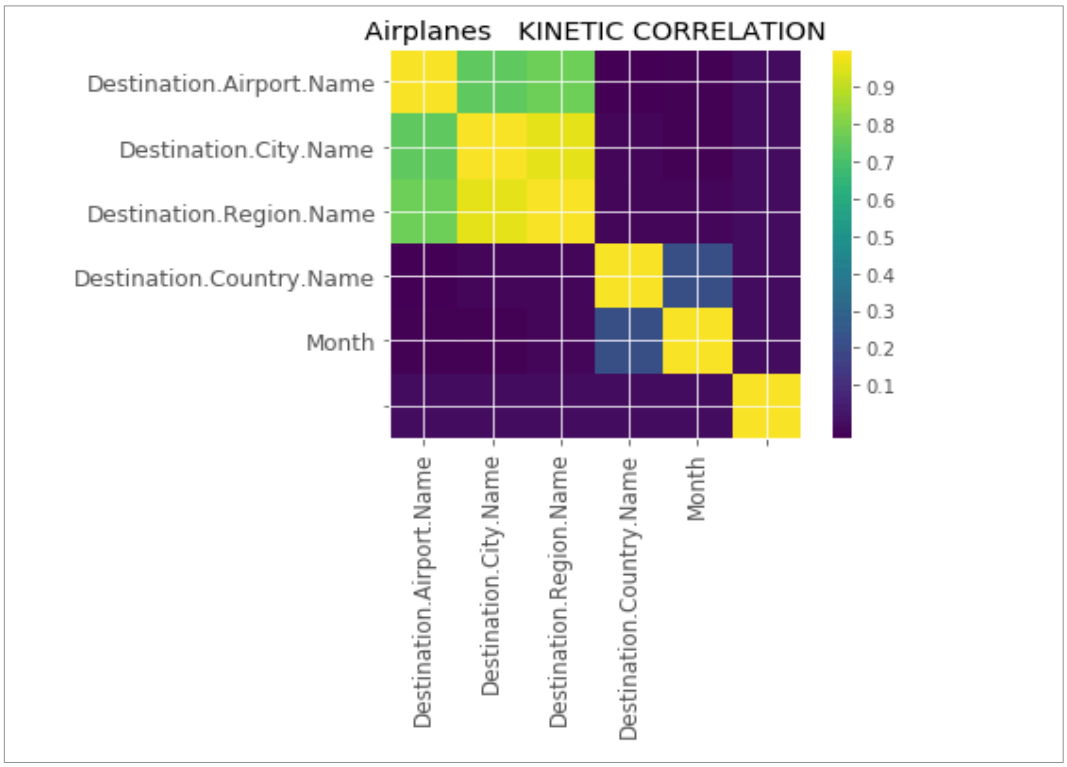

Figure 6-2 Train passenger numbers data with Kinetic Correlation.

The above results were obtained with the main goal being to determine whether more space could be brought in the distance between data clusters when kinetic energy correlation is applied. The following plots were obtained after the data's kinetic correlation was tested against the Pearson Coefficient;

As it had been anticipated, the kinetic correlation matrix of the kinetic correlation is far much higher when obtained from kinetic energy in comparison to the one obtained from the Pearson. Only linear relations in a dataset were detected by Pearson's R. both the X and Y axis have 7 columns each in the graphs. The correlation between columns on the scale 0 to 1.0 is shown by colouring on each square. A light colouring shows high correlation while a dark colouring indicates low correlation. It was also an aim of the experiments to figure out the reaction of plugging in the kinetic correlation into the PCA algorithm. Two new columns were created using the kinetic energy PCA method to achieve this. Two new columns are added on the kinetic model to enable airplane data training for further analysis to be conducted. The code below illustrates these actions.

```
train['kineticPCA1']=new_features[:,0]
train['kineticPCA2']=new_features[:,1]
train.to_csv('airplanes5.csv',index=False)
```

With the two columns newly added from the training data set, all passenger number values were added to another variable and the same column removed from the actual input data as explained further in Section 5.4.2.

6.4.2 Features Obtained from Kinetic Energy PCS Components

Complex interrelationships between variables give rise to a wide range of commonly applied predictive analytics. The process of determining the most relevant inputs to be used in a predictive model is called feature selection. With feature selection techniques, redundant, irrelevant and unwanted features with little or no contribution to, or those that lower, the accuracy of the models, can be identified and removed. From a mathematical perspective, formulation of input selection is done as a combinatorial optimization problem. Here, the error on dataset selection instances represents the generalized performance of the predictive model which is the function to be optimized. Inclusion (1) or the exclusion (0) of the input variables forms the design variables in the neural network. A wide variety of combinations, such as, 2N with N being the number of characteristics, would be evaluated through an exhaustive feature selection. As such,

requires application of intelligent methods that will facilitate efficient selection of features.

This section has demonstrated the application of eXtreme Gradient Boosting package, commonly known as XGBoost. This package facilitates the gradient boosting framework to be implemented in a scalable and efficient manner (Friedman, 2001; Friedman et al., 2000). It additionally supports a variety of objective functions such as ranking, classification and regression. The package is designed in such a way that it is expandable and thus giving room for users to define individual objectives with ease. Due to its high speed and good performance, the XGBoost algorithm has dominated in applicability for machine learning with either tabular or structured data (Chen and Guestrin 2016). In this study, it has been applied in a passenger figures training dataset with 51,983 observations and 9 variables. A variant of the famous 1-fold cross validation was used to ensure that a better performance estimate of the model has been achieved. The dataset was split with 75% of the dataset being allocated for the training set and 25% being allocated to the test set. The interrelation between features or target variables is shown by correlation. A positive correlation indicates that the value of the target variable is increased by an increase in one feature value. On the other hand, a decrease in the target variable value due to an increase in one feature value is an indicator of a negative correlation. With XGBoost, features highly related to the target variable can be easily identified. As such, the Table 5.3 below shows the obtained mean values of features obtained from the PCA components.

Table 6-3 The Mean Values of Features Obtained from Kinetic Energy PCA Components.

train-rmse-mean	test-rmse-mean
1031.46	1031.48
374.967	378.627
332.24	338.45
315.669	324.328
306.623	318.784

Table 5.3 shows that the calculated mean values are closer to the actual one.xgb.train values when the XGBoost model was run with python ML modules. The one.xgb.train is an advanced XGBoost model training interface. It additionally indicates that the

number of total mean values equals the number of boost rounds. The model's feature importance property can be used to derive individual feature importance of each feature in the dataset. Each feature in the data is given a score, whereby, a higher score shows that the particular feature is more relevant or important to the output variable. Increase in the model's prediction error depicts a feature's importance which is achieved following permutation of feature's values and consequently breaking the relationship between the true outcome and the feature. Extraction of the top ten dataset features will be conducted using the extra tree classifier. It should be noted that Tree Based Classifiers comes with the feature importance which is an inbuilt class. Figure 5.3 below is a hierarchical illustration of features used in airline passenger figures in their order of importance. It is clearly illustrated that the most predictive values are month, jet fuel, and fare. It is also to be noted that running the XGBoost model and inspecting the feature importance had great influence on capturing kineticPCA1 and kineticPCA2 on the plot. These are the features obtained from the modified PCA.

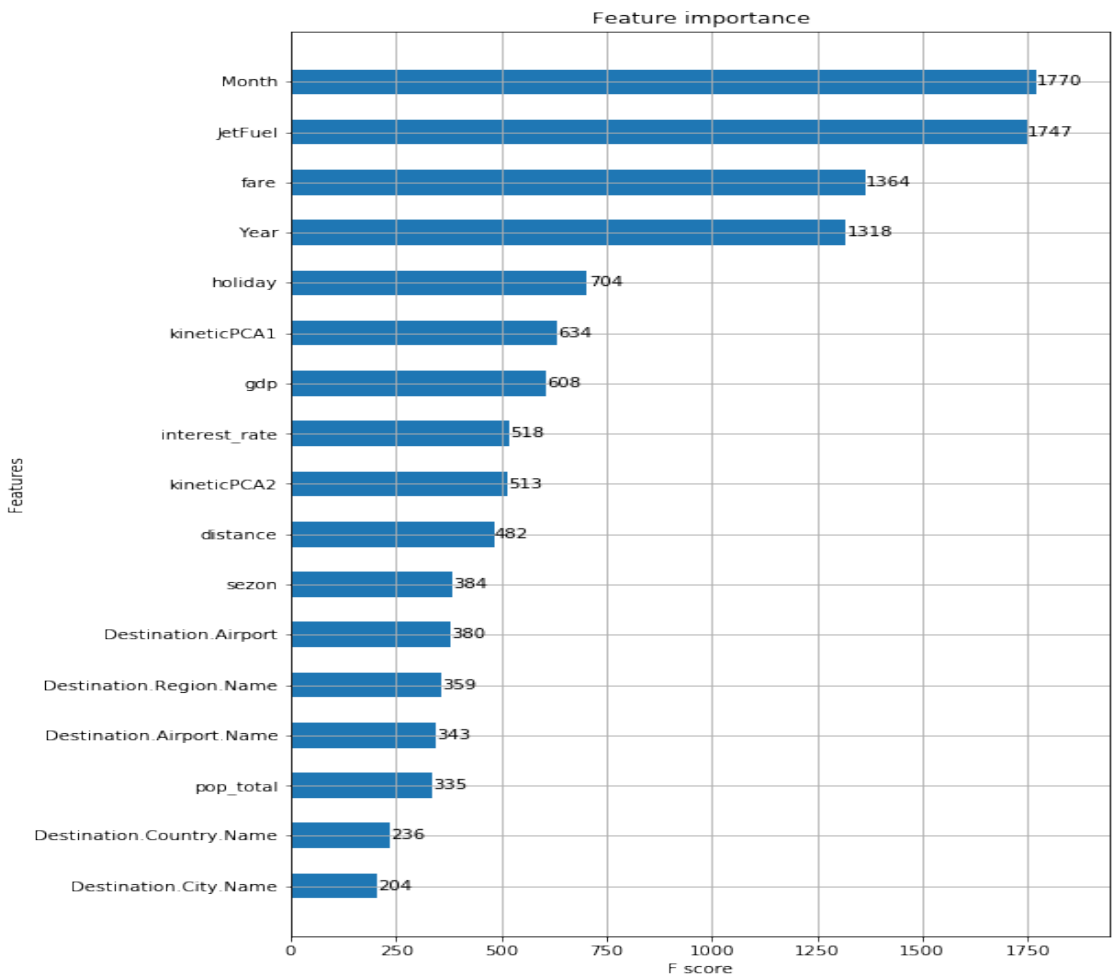

Figure 6-3 Principal Component Analysis Features (KineticPCA1 and KineticPCA2).

Table 5.4 is an outline of the predicted airline passenger figures using the prediction model. Additionally, the two newly created features have the second highest importance when compared to other methods of feature engineering that are to be later introduced in the study's experiments.

Table 6-4 Prediction Model using KineticPCA1 and KineticPCA2).

passangersPred5
515.111389
636.188843
621.036011
624.570862
607.825500

6.4.3 Features Obtained from Training Data Only

The XGBoost model ran in this section employed features obtained from the first experiment's training data. The whole activity commenced by deriving airline data from all columns of the training set. Nevertheless, all passenger number values were added to another variable and the entire column was then removed from the actual input data set. The behaviour of the model based on the newly created dataset was then determined by running the XGBoost model.

Jet fuel turned out to be the most important feature from the training dataset. A partial dependence plot based on the jet fuel feature shows clearly how changes of in the feature results to changes in the output regardless of the generalization error.

Table 6-5 The Mean Values of Features Obtained from Training Data Only.

train-rmse-mean	test-rmse-mean
1031.48	1031.5
375.129	378.55
331.191	337.084
314.235	322.608
305.394	317.117

From the Table 5.5, it is clear that the means calculated herein are much closer to the actual means. According to Table 5.5, the number of mean total values is equal to the number of boost rounds. It is also clear that the MPCA has learnt to use the 'JetFuel' feature to make predictions. It is therefore possible to unveil the important features from the training data by checking on the features that are utilized by the model to make predictions.

There is a reason as to why the training dataset was used and not the test data. Practically, it is the intention of this study to get the best possible final model by using all train model data. As such, there is no test data that is available to be used in feature importance computation. The same problem arises when there is the desire to estimate the model's generalization error. When cross-validation is nested for estimation of feature importance, the challenge of mot calculating the feature importance on the final model containing all data would arise. However, the behaviour would be different for models that have several data subsets.

Eventually, it would be necessary to determine whether there is the need to establish the level of the model's reliance on each feature, and in this case the training data set, to make predictions or the overall contribution of the features to the model's performance based on unseen data, which is the test data in this case. As of now, there is no published study or research with a detailed comparison of training data vs. test data on their applicability in computation of feature importance. Such an examination would need a more intensive examination exceeding the use of MPCA applied in this study.

Further on, an example has been outlined in which feature importance computation was based on training data. The choice for training data was largely influenced by the fact that it required a code with relatively fewer lines.

Figure 5.4 below is a hierarchical representation of feature importance in prediction of airline passenger figures from the most important to the least important. As such, it is clear that jet fuel, moth and fare have the largest influence on prediction. It is also to be noted that running the XGBoost model and inspecting the feature importance had great influence on capturing features that had been obtained from the training dataset such as holiday and season.

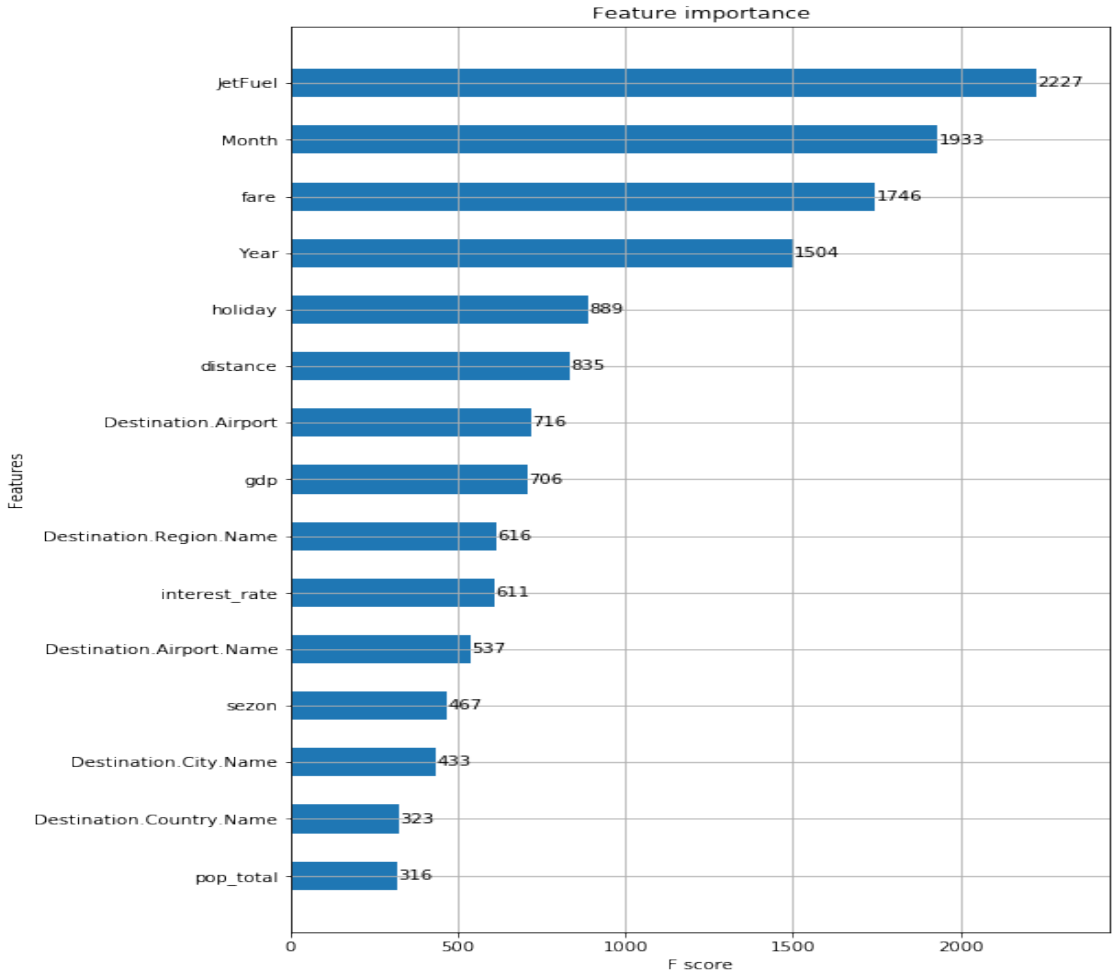

Figure 6-4 Features Importance Obtained from Training Data Only

Table 5.6 is an outline of the prediction model has been used for given values in passenger figures prediction and the prediction values have been kept in a new CSV file.

Table 6-6 Prediction Model using Training Data Only.

passangersPred1
540.558472
630.196777
617.746277
621.555908
627.247681

6.4.4 Features Obtained from Deep Learning Hidden Layers

In an artificial neural network, a hidden layer exists between input and output layers. It is in this layer that the activation function is utilized for the artificial neurons to take in a set of weighted inputs and consequently produce outputs. In almost all neural networks, this is the area where engineers typically simulate the human mental activities. There are various ways of setting up hidden neural network layers. Weighted inputs are randomly assigned in some instances while in others, back-propagation is conducted to calibrate and fine-tune the inputs. Either way, the functionality of the hidden layer artificial neuron is similar to that of the biological neuron in the brain. Construction of neural network hidden layers is the main focus point of many ML models analytics. Varying results are generated from different ways of setting up the hidden layers. In this study, for example, diversity in multiple datasets was achieved by creation of a differently engineered dataset. At this juncture, it was decided that addition of non-linear features emanating from implementation of R in the deep Learning model in Chapter 4 was necessary. The used model had the topology illustrated below;

```
model<-
h2o.deeplearning(x=predictors,y=target,training_frame=tr
ain,
hidden=c(100,63,30,15),epochs=30,
nfolds=5,fold_assignment="Modulo"   ,#  can  be  "AUTO",
"Modulo", "Random" or "Stratified",l1=5.6e-05, l2=7.4e-
05,input_dropout_ratio=0.05)
```

100 neurons in first hidden layer, 63 neurons in second hidden layer and 30 neurons in the third hidden layer and 15 neurons in last hidden layer were used to train the deep learning neural network. Each hidden layer had the same number of neurons as the number of features extracted from the deep learning model. The correlations of each feature corresponding to a neuron in the hidden layer were computed with the target variable, which is the passenger number in 1000/month, to ensure that the most important non-linear features are selected effectively.

In the course of correlation computation, a single feature from each neuron was kept, this being a depiction of the comparison between maximum correlation and other features in the same hidden layer. Subsequently, the final four non-linear features are obtained. The behaviour of the model based on the new dataset was then determined by running the XGBoost model as illustrated in Table 5.7 below.

Table 6-7 The Mean Values of Features Obtained from Deep Learning Hidden Layers.

train-rmse-mean	test-rmse-mean
1031.7	1031.72
376.89	381.492
333.428	341.489
314.42	326.202
303.514	319.026

In Table 5.7, clearly illustrates that the calculated mean values are closer to the actual ones though a significant difference was recorded on the last set of inputs.

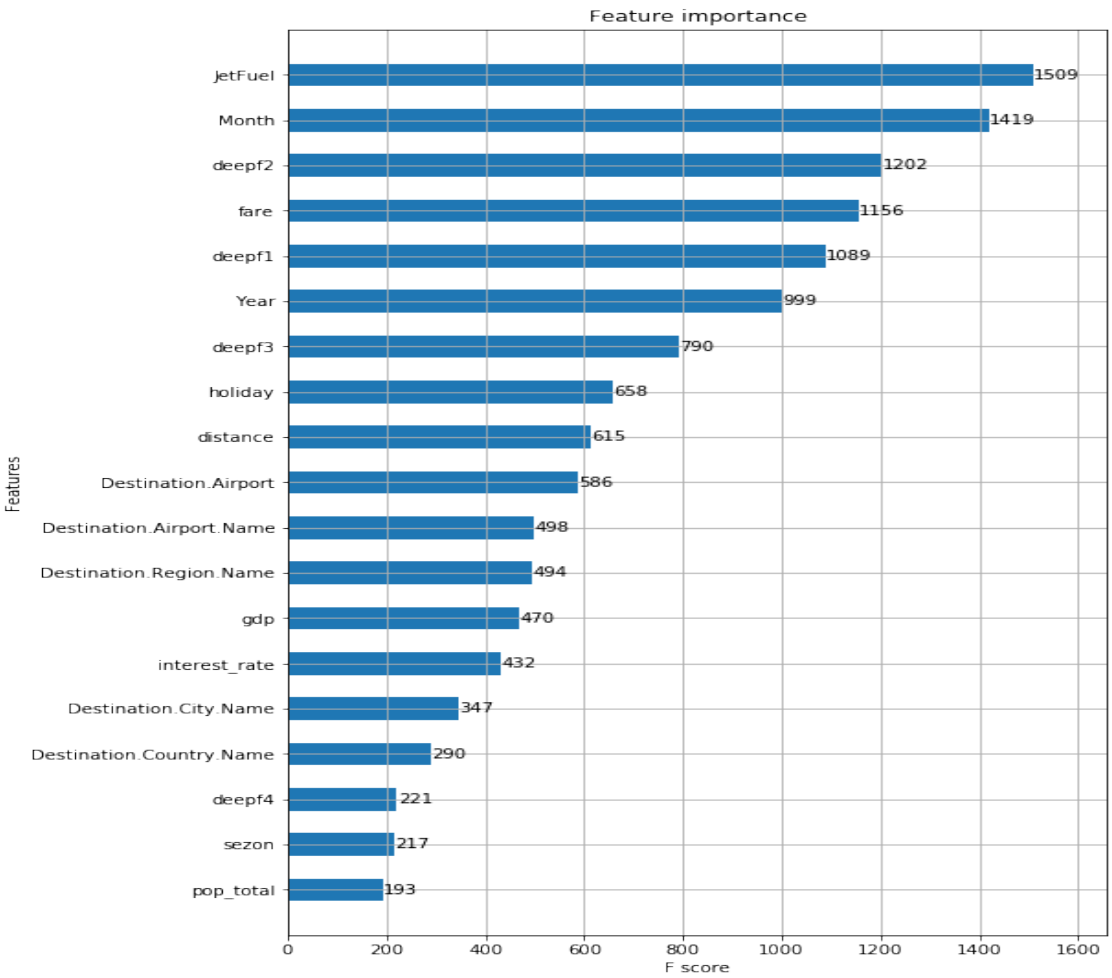

Figure 6-5 Features Importance Obtained from Deep Learning Hidden Layers.

Figure 5.5 above is a hierarchical representation of feature importance in prediction of airline passenger figures from the most important to the least important. As such, it is clear that jet fuel, moth and fare have the largest influence on prediction. It is also to be noted that running the XGBoost model and inspecting the feature importance had great influence on capturing non-linear features deepf1, deepf2, deepf3, and deepf4 on the plot.

Table 6-8 Prediction Model using Deep Learning Hidden Layers.

passangersPred4
582.059143
545.397705
546.236450
552.925049
627.696899

Table 5.8 above is an outline of the predicted airline passenger figures with the use of Deep Learning Hidden Layers.

6.4.5 Features Obtained from Genetic Algorithm

Genetic algorithm is one of the most advanced feature selection algorithms. It is a stochastic method founded on biological evolution and natural genetics mechanics for function optimization. This section will demonstrate application of genetic algorithms in selection of the most relevant features leading to performance optimization in a predictive model. The symbolic transformer is a genetic algorithm from which the features were extracted. This algorithm is an estimator that commences with building a relationship representative from a naïve random formula's population (Lowe, 1999). Being a heuristic optimization method, genetic algorithm is inspired by natural evolution procedures.

The generalization performance of a predictive model is usually the function to be optimized in feature selection. In the first step, the individuals in a population are created and initialized. Random initialization of the individuals' genes occurs due to the nature of this algorithm as a stochastic optimization method. Next on, a fitness must be assigned to each individual. The predictive model must therefore be trained with the

train data and subsequently use selection data to evaluate is selection data to facilitate successful evaluation of fitness. Low fitness will be indicated by a high selection error. As such, the probability of individuals displaying the greatest fitness to be selected for recombination will be high.

After successful performance of the fitness activities, individuals to be recombined in the next generation are chosen by the selection operator based on their fitness levels. At this stage, the individuals that are more fitted to the environment have a high survival possibility. With N/2 being the number of selected individuals, N is the population size. Direct survival of the fittest individuals for the next generation is enabled by elitism selection. The number of directly selected individuals is controlled by elitism which is commonly set to a small value such as 1,2… onwards. Further on, stochastic sampling with replacement or the roulette wheel is one of the most commonly applied selection methods. In this method, all individuals are placed on a roulette with areas proportional to their fitness. The roulette is then turned and random selection of individuals is done. The individual that corresponds is then selected for recombination.

Once the selection operator has chosen the population halfway, a new population is generated by recombining the selected individuals using the crossover operator. This operator randomly picks two individuals and their features combined to give rise to four new ones for the new population. This process is done repeatedly until the new population achieves a similar size as the old one. The decision of which parent will bear the off springs' features is made by the crossover operator. As such, it is possible for the operator to generate off springs that are identical to the parents and thus resulting to lowly diverse new generation. To counter this problem, the mutation operator randomly changes some feature values in the off springs. A random number between 0 and 1 was generated to determine whether a feature is to be mutated or not. From this, the variable is flipped whenever this number turns out to be lower than the mutation rate value. Usually, the mutation rate is 1/m with m being the number of features. With the mutation rate values, one feature for each individual must be mutated statistically. The entire process is the conducted repeatedly until a stopping criterion is met. Prom the process, it is more likely that a generation will be well adapted to the environment than its predecessor.

Below are advantages derivable from use of genetic algorithm methods;

1. They have improved performance in comparison to traditional feature selection techniques.
2. They can handle datasets containing many features.
3. It is not a must to have specific knowledge of the problem being studied in order to apply them.
4. It is easy to parallelize algorithms clusters in computer clusters.

Their main disadvantages are;

1. They may be computationally expensive since a predictive model must be built to evaluate each individual.
2. Due to their stochastic nature, they take relatively longer to converge.

The formulas assume a tree-like structure representation while at the same time recursively applying mathematical functions to constants and variables. The fittest individuals in a population are selected and evolved into a successive generation of programs. Such individuals are taken through genetic operations such as reproduction, mutation and crossover. Conclusively, in this study's model, selection of the best variable's subset can be achieved by genetic algorithms. This process however requires enormous computations. Figure 5.6 below illustrates the results explained above'

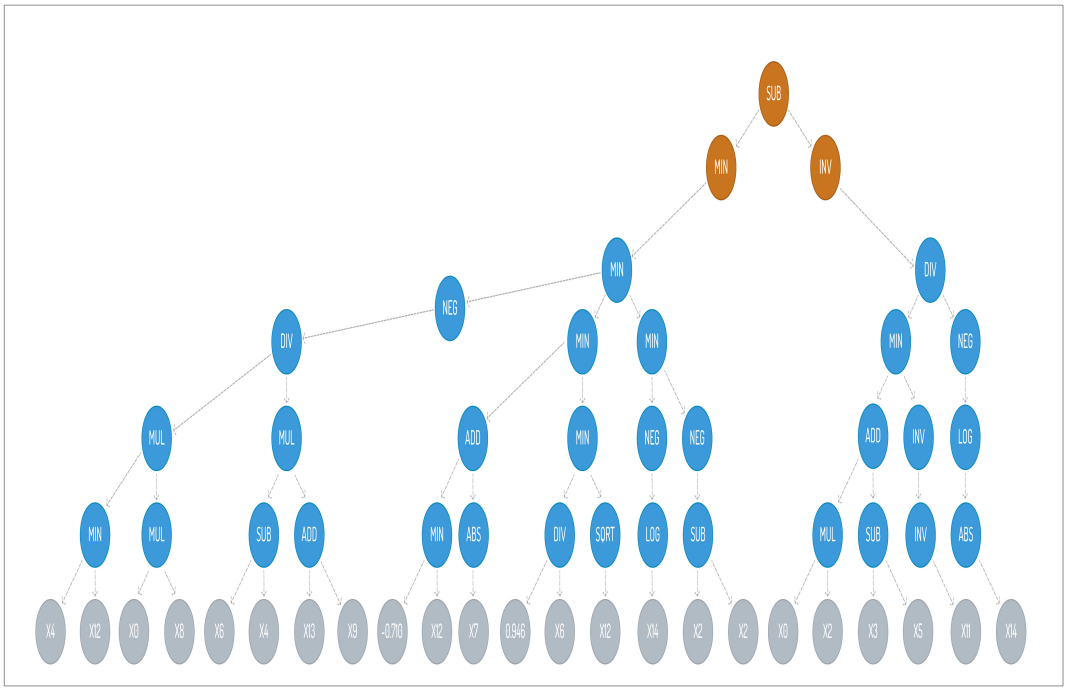

Figure 6-6 Tree-like Structures of the Genetic Algorithm.

The different kinds of operations produced by genetic algorithm can be easily observed in a genetic program. After running the XGBoost model, two new features were obtained from the genetic transformer and added to the algorithm.

Table 6-9 The Mean Values of Features Obtained from Genetic Algorithm.

train-rmse-mean	test-rmse-mean
1031.46	1031.48
375.332	378.553
333.665	339.378
317.021	325.23
307.186	318.731

In Table 5.9, clearly illustrates that the calculated mean values are closer to the actual ones though a significant difference was recorded on the last set of inputs. Moreover, Figure 5.7 below is a hierarchical representation of feature importance in prediction of airline passenger figures from the most important to the least important. As such, it is clear that jet fuel, month and fare have the largest influence on prediction. Additionally, it is clear that a relatively low influence below researcher's expectation emanated from capturing the genetic features 'genetic_feat1' and 'genetic_feat2' in the plot.

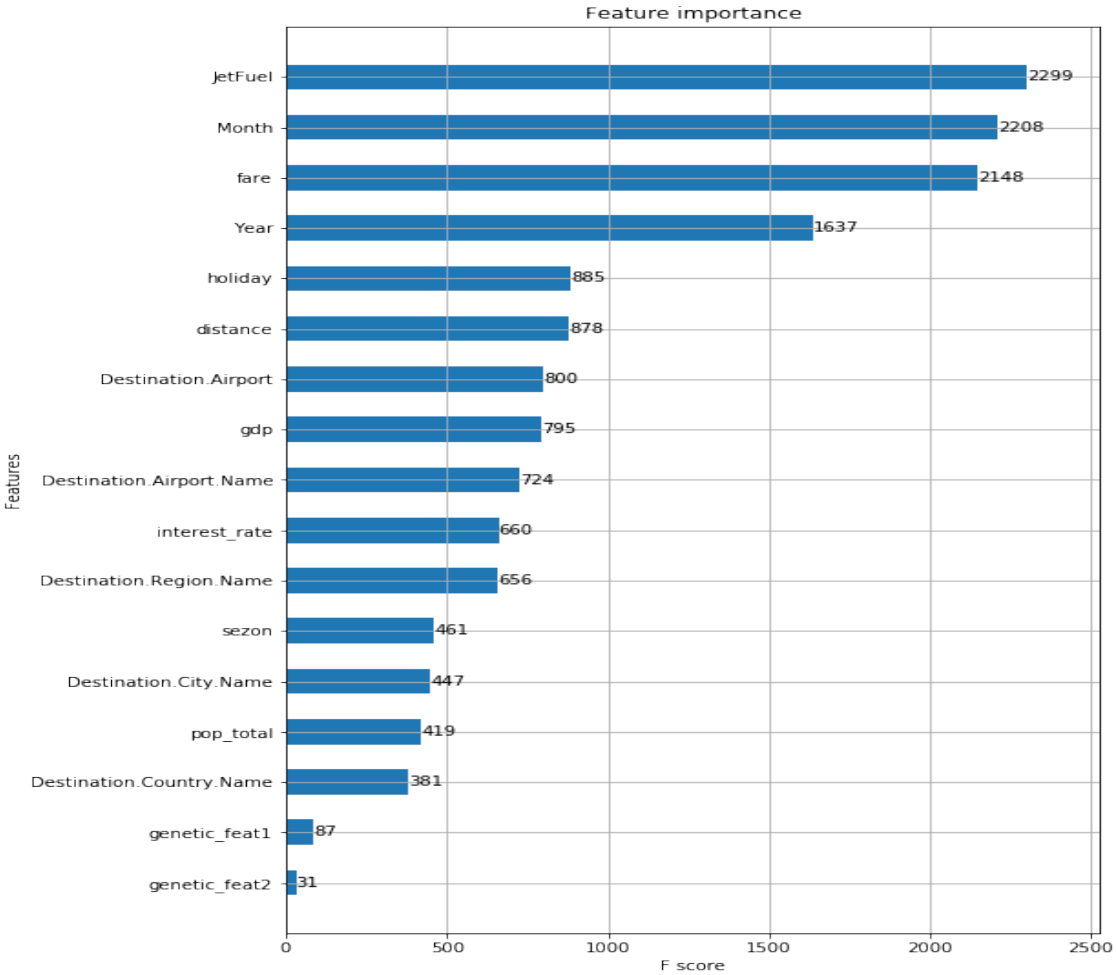

Figure 6-7 Features Importance Obtained from Genetic Algorithm.

Table 5.10 below is an outline of the predicted airline passenger figures with the use of Genetic Algorithm.

Table 6-10 Prediction Model using Genetic Algorithm.

passangersPred2
529.164917
611.098145
600.075256
609.790955
607.040894

6.4.6 Features Obtained from One-Hot Encoding

One-hot-encoding is method commonly used in data science to deal with categorical features. It is a process through which categorical variables are converted into a form

that can be easily fed to ML algorithms and consequently enable them produce improved prediction results. In this study, the 'month' feature consisting of 12 categories was transposed into 12 dimensional binary spaces using this method. As such, the categorical variables are represented as binary vectors. It is a requirement of the process that the categorical values be first mapped into integer values. Each integer value is subsequently represented as binary vector that have all values as zero. Only the index of the integer is marked with 1. When the XGBoost model is trained on this new feature importance dataset, 12 newly added columns are obtained. All the passenger number values were then added to another variable and the same column removed from the actual input Dataset see Table 5.11.

Table 6-11 The Mean Values of Features Obtained from One-Hot Encoding.

train-rmse-mean	test-rmse-mean
1032.74	1032.78
370.113	373.86
327.627	333.901
311.213	320.576
301.409	314.091

Table 5.11, clearly illustrates that the calculated mean values are closer to the actual values of one.xgb.train, though a significant difference was recorded on the last set of inputs.

Figure 5.8 below is a hierarchical representation of feature importance in prediction of airline passenger figures from the most important to the least important. As such, it is clear that jet fuel, month and fare have the largest influence on prediction. The Figure also shows that the number of mean total values is equal to the number of boost rounds. Additionally, 12 new features and their corresponding influences have been obtained rather than a single feature and thus showing that the influential magnitude is not very high.

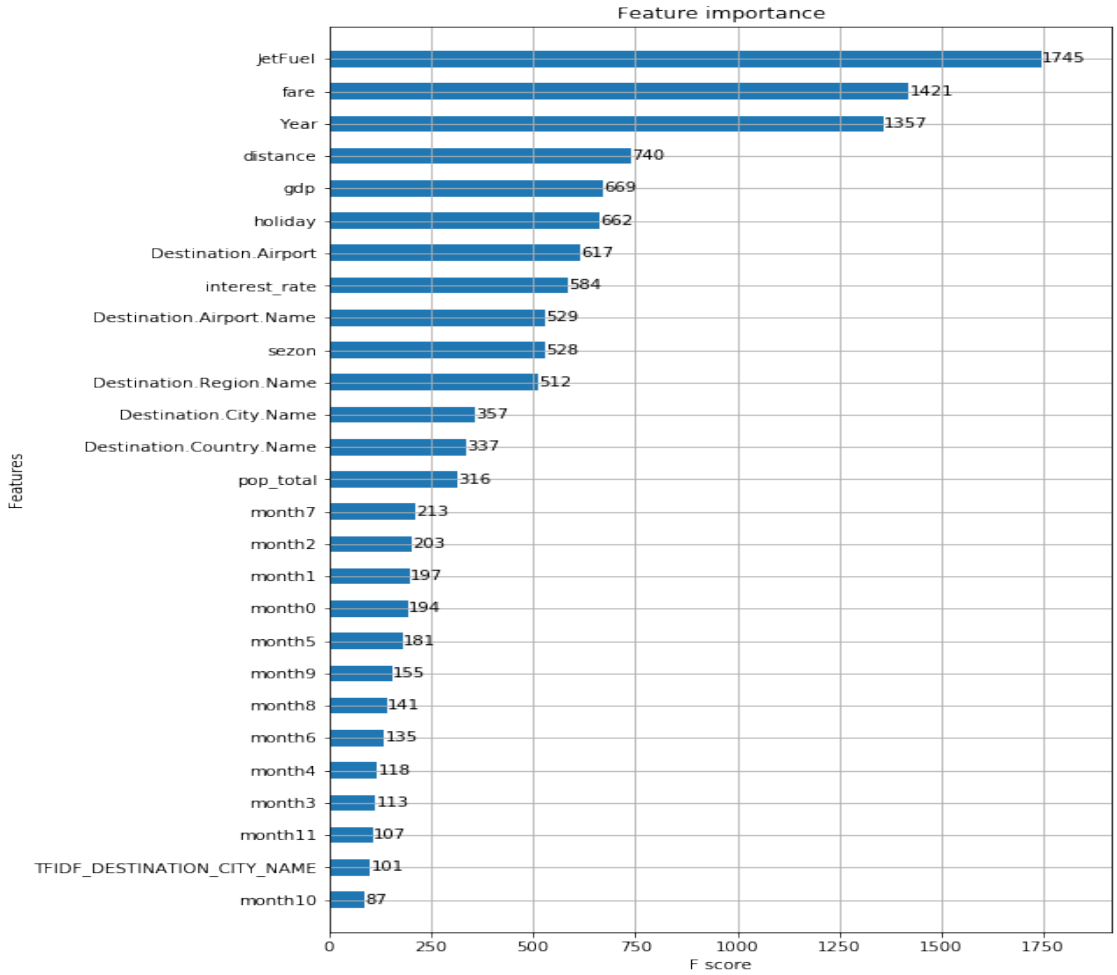

Figure 6-8 Features Importance Obtained from One-Hot Encoding.

At first, all possible one variable models were fit and then the variable model with the highest performance was chosen. Later on, all possible two variable models were fit and thus adding a second variable to the one variable model with the highest performance previously attained. A two-variable model with the highest performance is chosen and if it displays superiority than the previous one variable model, then the experiment is proceeded with it. All three variable models are fit, then four variable models, then five and so on until the process achieves the best k features or there are no more improvements in the model. However, the first impressions of this approach are; the main challenge is that there is a high possibility of not finding a most optimal solution. More so, variables from future steps that are yet to be included in the process may work quite well with the candidate variable. Also, if the algorithm terminates beforehand, the candidate variable may not have the chance of utilizing potentially significant predictors.

Table 6-12 Prediction Model using One-Hot Encoding.

passangersPred3
586.558441
620.629944
619.795837
614.935303
612.234863

Based on one-hot-encoding, Table 5.12 above is an outline of the predicted airline passenger figures and the prediction values have been kept in a new CSV file.

6.4.7 Feature Obtained from Conditional Probability

After determining that categorical nature cannot be used to exploit more features, it was decided that a conditional feature obtained from joint probability distribution be created. In their analysis of the theory of probability, Gut, (2013) defines conditional probability as the measure of how much an occasion is likely to occur given that another particular event has already occurred. In a case where A is the event of interest and B is a known event that has already, or is assumed to have, occurred, the probability of A occurring under the condition of B is written as P(A | B), or sometimes $P_B(A)$ or P(A / B). In this study therefore, the conditional probability P (City | Month, Year) was created and designed to answer the following question;

"What is the probability that on a specific row in particular city giving the evidence that month has some particular value and year has some particular year?"

To, therefore, experiment on creating a conditional feature obtained from joint probability distribution, the airline dataset with a single newly added column was read.

Table 6-13 The Mean Values of Features Obtained from Conditional Probability.

train-rmse-mean	test-rmse-mean
1031.46	1031.48
373.877	377.486
331.002	336.737
315.453	323.695
306.327	317.887

All the passenger figures values were put in another variable and the same column removed from the actual input dataset. In Table 5.13 above, the plot clearly indicates that the mean values calculated are much closer to the actual values. A significant difference was however recorded after the points were increased. After the XGBoost was ran on the dataset entailing the new feature, the conditional feature was determined to be the least influential as shown in the figure below. This feature was however kept with the main aim of increasing the model's diversity as shown in below Figure 5.9.

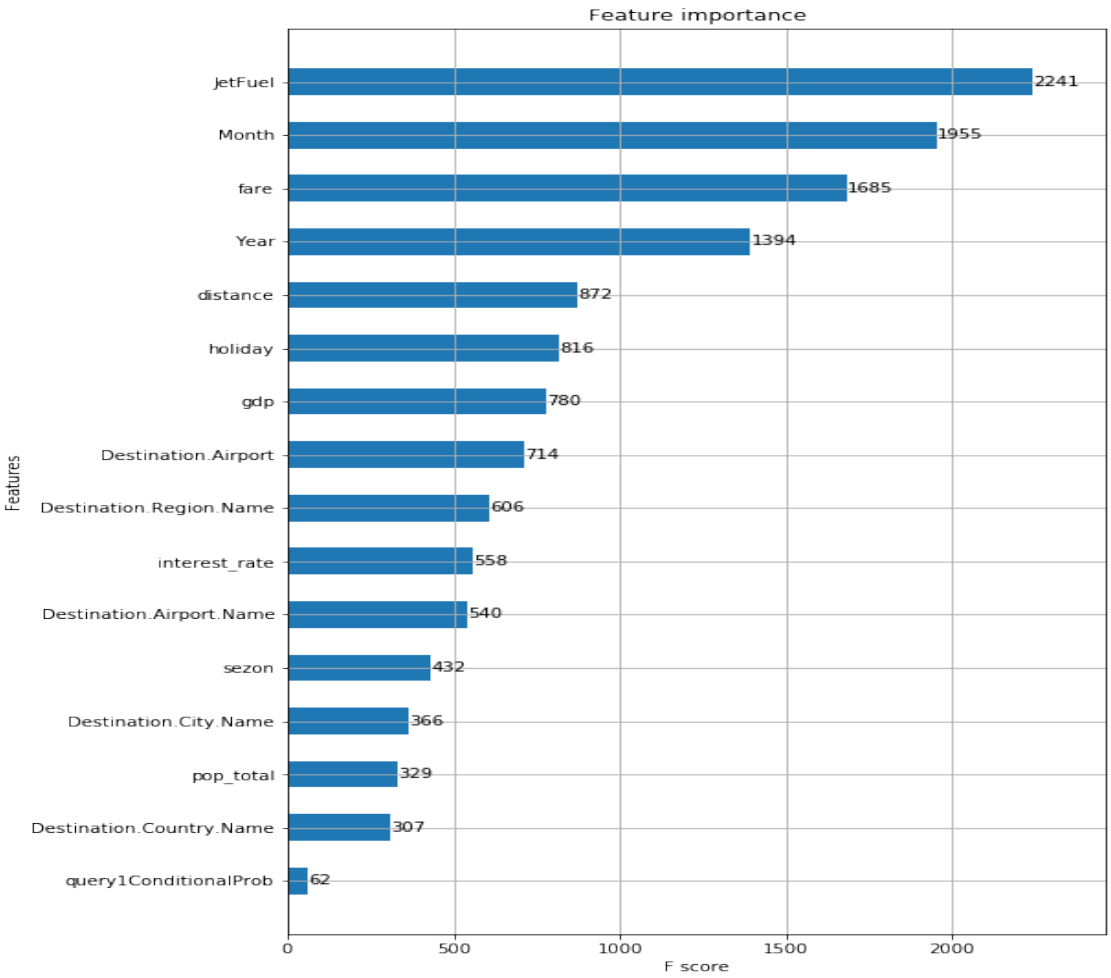

Figure 6-9 Features Importance from Conditional Probability.

Figure 5.9 above is a hierarchical representation of feature importance in prediction of airline passenger figures from the most important to the least important. As such, it is clear that jet fuel, month and fare have the largest influence on prediction. The figure also shows that the number of mean total values is equal to the number of boost rounds.

Additionally, the influential magnitude is not quite high as one new feature has been obtained with the corresponding influences rather than having a zero feature.

Table 6-14 Prediction Model using Conditional Probability.

passangersPred6
621.677002
628.961792
629.928589
644.625793
650.002625

Table 5.14 above is an illustration of the passenger figure values predicted with the use of conditional probability. As aforementioned, the conditional probability P (City Name |month) was intended to answer the question; what is the probability of achieving particular passenger number value for a given destination city for a particular given month in a particular year?

Nevertheless, the probability space was defined by loping in for the total number of keys after the list counts of each tuple were taken. At each time, an entry at the same key position was added by a similar count as well as the actual shape count. Consequently, an empty table with an equal number of tuples and with zero as the initialization was created and the values then added into the joint probability table for each probability. In conclusion, a list of conditional probability of city was created and added as a new column in the train dataset.

6.5 Ensemble Stage and Outlier Detection

Ensemble analysis is a method that has been widely reviewed by literature with its main function being reducing a model's dependence on a specific data set or data locality. Clustering and classification are some of the areas where the ensemble technique is commonly applied with each of these meta-algorithm areas being considered as a vibrant and active subfield in its own right. This technique creates a more robust model by combining the results from a variety of models. An outlier is defined as a particular data point maintaining a significance distance from other similar points. This is an occurrence resulting from measurement variability or at times indicating experimental errors. The outliers should not be included in the dataset if possible. It is, however, a

difficult undertaking to determine these anomalous instances and in most scenarios, it is practically impossible. ML algorithms have a high sensitivity to distribution and ranges of attribute values. It should therefore be noted that the entire training process can be misled or completely spoilt by data outliers resulting to longer training durations, models with low accuracy levels and definitely poor results.

Generally, outlier detection makes ensemble analysis a complicated process and thus being focused on in literature concerning outlier analysis. The case of high dimensional data is an instance where ensemble analysis is commonly used. One of the earliest feature bagging approaches used in detection of high dimensional outliers is Formalization of outlier ensemble analysis (Lazarevic et al 2005). Currently, ensemble analysis is designed for application on high dimensional cases though there is a potential broader applicability. The certainty and robustness of results obtained from the process of subspace discovery are greatly improved through ensemble analysis.

As such, with the aforementioned in mind, this part of the study has focused on detecting the outliers for the monthly airline passenger figures which is the target feature. The following steps formed the foundation of the outlier detection algorithm;

1. After the XGBoost model was used to make predictions on all datasets obtained through various feature engineering processes as illustrated in sections 5.3.7, 5.4.2, 5.4.3, 5.4.4, 5.4.5, 5.4.6, as well as on the initial dataset with the original features, the following set of predictions were obtained;
2. Prediction1, Prediction2, Prediction3, Prediction4, Prediction5, Prediction6. Prediction 2 was a key point of interest and referred to predictions obtained after the dataset was ran using the modified PCA method.
3. The data frame containing the predictions was randomly split into two sets with the first one being 75% of the initial size and representing the training set while the second one was the remainder 25% and represented the validation set.
4. The “o” training set outliers in the training set were flagged with ‘1’ whenever a particular value was determined to be an outlier and with ‘0’ if the value was not an outlier. Following multiple experiments, a good definition of outliers for the case of this study was said to be; y=outlier, where y>=2.5*standard deviation of y +mean of Y, where Y=target /passenger numbers.

5. Outliers are then predicted in the test set after a random forest classifier was fitted in the training set
6. The maximum values from all predictions were used to replace the predicted target feature outlier values as shown; replaced_outlier=MAX((Prediction1, Prediction2, Prediction3, Prediction4, Prediction5, Prediction6)

The above steps were repeated in the absence of **prediction2** which had been previously mentioned to be from the modified PCA version and the results compared with those from the above steps. The following conclusions were made;

a. The histogram of the training set prediction is reassembled into a better histogram by the final test set predictions.
b. A better replacement of the outliers is achieved.
c. A decrease of the root mean squared error is recorded.

The above a. b, and c observations were made only in the presence of **prediction2** from the modified PCA. Refer to Appendix 3 for more explanation.

6.6 Discussion

With the continuously arising challenges and trends in the field of airline passenger figures predictions, it is consequently become a key focus point of researchers in various studies. Successful and smooth operations as well as a boost in profitability can be achieved through accurate prediction of airline passenger figures. Forecasting of future passenger numbers is mainly based on past data. In a case where passenger data for the past 90 days is given and an estimate of the next 10 days is required from it, the most suitable estimator to use is the multivariate conditional mean due to its ability to minimize the mean square error of estimation. However, it is not at all times that numerical results can be relied on since use of this method to estimate future figures is not a well-conditioned process.

Long-term forecasting has been achieved using a variety of methods. Principle Component Analysis is commonly applied method that employs linear techniques. The main aim of this chapter was to therefore propose method with similar forecasting power but with the ability to improve the reliability of the numerical results obtained. As such, a modified PCA version modified with kinetic correlation matrix using kinetic energy has been introduced. Different sets of airline passenger data have been used to assess the modified PCA and the results compared with those from the traditional PCA.

The modified PCA results shows that Pearson correlation is much lower than Kinetic correlation. This is relatively sensible since Pearson's R can only detect linear relations in a given dataset.

It also turned out that classification accuracy, classes' reparability and data dimension reduction were achieved more efficiently and effectively with the modified PCA than with the traditional PCA. Based on the results obtained in this chapter, clustering in hyper-dimensional space using kinetic correlation as a distance can be achieved with application of the modified PCA and consequently make it run in real time when doing future work. Outliers have also been highlighted as a major hindrance in the construction of predictive models. They have been attributed to poor results in various studies. To counter the challenge, this study proposed an ensemble method that effectively detected outliers.

Chapter 6: Conclusions and Future Work

7.1 Overview

The foundation framework for civil aviation was set up in 1944 after the signing of the Chicago Convention by Franklin Roosevelt. Since then, tremendous transformations have been witnessed, subsequently leading to the global economisation of the industry to its current state (ICAO, 2005). For the past 30 years, the aviation industry has showcased an annual average growth rate of 5% (MIT, 2015). Aeronautical Information Services (Eurocontrol, 2014), as well as Air Traffic Control (ATC), are key specialties for Air Traffic Management (ATM). Through such ATM activities, safe aircraft flights are facilitated, and improvements are sought on the existing operations to increase effectiveness and efficiency in the industry's service provision (Zhong et al., 2015).

Air Traffic plays an important role in facilitating a smooth flow of people and capital as the per capita income grows in the Middle East (Duval, 2008; Zhong et al., 2016) (Duval, 2008). Various airlines have been able to intensify their operations to provide services to almost all corners of the globe (Duval, 2008; Zhong et al., 2016). Airbus, for example (Airbus, 2014), one of the largest aircraft manufacturers, forecasts an average annual growth of 4.6% in air traffic operation in the period 2014-2024 (Airbus, 2014). The forecast also shows that Asia and the Middle East will eventually become the largest markets due to their being major global destinations for various business activities (Airbus, 2014). Similar estimates are also brought forth by the International Air Transport Association, which stated that the number of air passengers per year is expected to rise to 7.3 billion by 2034. This figure will be more than twice the number of passengers served in 2014 (IATA, 2014).

Despite the anticipated future growth in the industry, a survey by the European Organisation for the Safety of Air Navigation disclosed some rather disturbing facts. The results of the survey showed it was critical that more than 16 factors needed to be reviewed and addressed, as they would otherwise hinder the anticipated growth in one way or another (Eurocontrol, 2009). As such, when conducting future forecasts, the

relevant authorities also should come up with appropriate strategic planning and development policies to counter all the hindrances that would constrain the anticipated growth. The survey result highlights the significance of assessing air travel forecasts to develop viable strategic plans for the future of the aviation industry; (Wensveen, 2007; Phyoe et al., 2016). Forecasting also serves to provide a foundation of environmental assessment for carbon emissions emanating from aviation activities (ICAO, 2010). Originally, influencing factors were not considered comprehensively in forecast models, especially with regards to passenger numbers. (Gosavii et al., 2002; Riedel & Gabrys, 2003).

In this study, however, different attributes are recognised as classification vectors, and new time series algorithms were designed by calculating representatives in all clusters between observing time series and representatives. The study additionally embarks on a new approach based on feature selection and subsequently demonstrates the features that are significant in facilitating the prediction of passengers travelling to various destinations at different seasons. The deep learning and genetic algorithm-based feature extraction is guided by higher probabilities of survival for fitter combinations of features, where the features are extracted by Feature-Space = {'Year', 'Month', 'Destination Airport', 'Destination Airport Name', 'Destination City Name', 'Destination Region Name', 'Destination Country Name'}. The effects of different parameters for a variety of techniques were also studied by classifying accuracy and comparing the results with those obtained through stepwise addition of features. An in-depth discussion is also conducted to determine how different features correlate and the significance of such correlations.

7.2 Study Approach

The available data was relatively large, consisting of 51,983 observations. Consequently, a larger subset of forecasting models as available in the literature had to be considered. However, a broader or narrower model can be considered to match the size of the dataset. If more airline passenger data variables are available and can be measured, other modelling approaches can be used. Such models include advanced machine learning algorithms and multiple regression models.

Chapter 2 described how various methods were employed to forecast airline passenger figures. Principle component analysis (PCA) is a commonly used forecasting method

with its applicability traceable to the forecast of airport passenger traffic in Hong Kong (Tsui et al., 2014). Nonlinear forecasting methods were also used to forecast the air passenger flow in Russia (Blinova, 2007).

Chapter 3 described how various hybrid methods were employed to conduct data pre-processing prior to the actual forecasting. When forecasting air passengers passing through Muscat International Airport, the STL decomposition method was predeceased by seasonal decomposition pre-processing. This was in a non-linear forecast that showed more accuracy than a variety of standalone methods. Seasonal decomposition allows observation of the underlying distinctive characteristics in a particular time series by removing any systematic seasonal variations. In this study, the application of STL decomposition increased the accuracy of time series' components (Hyndman & Athanasopoulos, 2013).

In Chapter 4, hybrid methods were proposed for instances where more than one forecasting method is used. This will consequently enable compensation for the drawbacks emanating from individual methods. When the ARIMA model was employed, followed by the ANN model, for example, high levels of accuracy were obtained in comparison to application of single standalone models (Zhang, 2003). This chapter additionally presents an overview of various factors affecting the forecasting of airline passenger figures. A clear depiction emanates from the examples used therein revealing the significance of using interrelated features to increase the accuracy of forecasted airline passenger figures. It also came out clearly that the addition of new features to the old inputs significantly altered the variance results of new outputs.

Optimisation of the deep neural network model significantly improved the results. A general observation made was that more accurate results were obtained from the deep neural network than from any individual machine learning model or even a combination of different machine learning models (Brzezinski & Stefanowsk, 2014; Zhang et al., 2009). Being the new model in the predictive world, the deep neural network prediction parameterisation is based on important features of the original space. The basic predictions produced by different time series models are combined by a fixed rule depending on the booking level. The weighting parameters to adopt the forecasts to the season and holiday are calculated by fixed rules too, depending on the number of months to a destination as well as information relating to seasonality and

public holidays. Although performance has been used to optimise all these rules, the ultimate goal was to eradicate all fixed rules and adopt a dynamic approach to the combination. The combination should, therefore, be able to adjust the basic methods and also adopt new features.

This study pays deep attention to the experimentation of new features obtained from optimised neural network layers. The significance of feature selection was visualised using existing machine learning models' strongpoints. The methodology of deep neural network incorporation is selected after the selection of features, addition of new features at various steps and visualisation of how all features correlate. It is through this methodology that best modelling practices are obtained and maintained. Since various scholars are raising concerns against machine learning and criticising it for poor definition of the variable selection process, such scholars have consequently embarked on the use of deep neural network with the desire to achieve accurate forecasting results (Essa & Ayad, 2012; Friedman & Popescu, 2008).

The figure segments should deliver reliable forecasts for each flight date and admission level. The ticket request at a charge level (e.g. per take-off date) can be displayed as a period arrangement. However, only a few strategies have been able to deliver figures with an acceptable accuracy level as a result of the nature and structure of the available information. In our case, rapid global changes are being witnessed and as a result, only a little information is available and various pertinent estimates are often lacking. Deep research on this theme has shown that straightforward and vigorous time arrangements impacting models such as basic normal, diverse renditions of exponential smoothing or relapse models portray superiority over more refined techniques. Additionally, academic investigations have proved that movements in airline flows are not random. To the contrary, they behave in a highly non-linear, dynamic manner and thus make predictability of airline passenger figures quite challenging (Huang et al., 2013). As discussed in Chapter 3, many factors impact air traffic, among them prevailing economic conditions. It is therefore worth noting that economic indicators used in this analysis are calculated from historical data.

Various machine learning approaches have been used to analyse these economic indicators and consequently predict future trends. The majority of the approaches come with various drawbacks such as over-fitting or under-fitting, initialising a large number

of control parameters, and finding the optimum solutions. As a solution to these shortcomings, Chapter 5 details how the modified principal component analysis (MPCA) method is suggested. The main reason for this choice is that MPCA uses the structural risk minimisation principle for function estimation while the traditional methods implement the empirical risk minimisation principle. This is preferable because, in order to make predictions, which are as accurate as possible on an actual unknown distribution, it is necessary to minimise the error of the predictions on the data that obtained from that distribution.

Attention is drawn to the results obtained from a modified principal component analysis (PCA) using a different correlation matrix sourced from kinetic information energy properties of the features as represented by random vectors. Additionally, experimentation on the new features obtained from an optimised neural network's hidden layers has been conducted.

With the main aims being to obtain distribution on some hold-out dataset that resembles the original distribution of the training dataset and improving prediction performance metric, predictions made on different engineered feature spaces were assembled while replacing all outliers of incorrect predictions. The first important steps in developing MPCA-based forecasting model are feature extraction (transforming the original features into new ones) and feature selection (choosing the most influential set of features). To improve this model, however, two-stage methodologies named PCA-MPCA have been proposed in this research. It was also a proposal of authors from previous studies to take the average of the data to fill the missing values of continuous features and the most common value to fill the missing value of the discrete feature. However, it was noted that making these changes could result in a high misclassification rate which would, in turn, affect the classification accuracy.

This study, therefore, proposed two new methods to solve the issue of missing values. In the first method, all features with a high frequency of missing values are removed. Such features would not contribute any significant value to the final results. In the second method, the finite difference method is used to determine the missing values and consequently result in an efficient selection of a feature set. Now with the desired feature set, PCA was used in the first stage to extract features which were then reduced into a low-dimensional feature space. MPCA was then applied to the reduced feature

space in the second stage to extract principal components, from which a forecasting model was finally constructed.

7.3 Research Outcomes

The results of this study show that the random forest model is the most powerful of all models in predicting monthly airline passenger flow. The variables used in the model are base fare, number of public holidays, MA(3) of total passengers, MA(1) of total passengers, and the first difference of total passengers. The travel behaviour of air passengers from Oman to various destinations around the globe is influenced by base fare and holidays, both of which are exogenous variables. When the random forest model is fitted with new features, a 69% variance increase was observed. The deep learning model was optimised by conducting a hyperparameter search to determine the best topology from thousands of topologies in the neural hidden layers by determining the root mean square errors. The best hyperspaces of parameters were determined through a grid search of some validation frames (split them in training and test validation).

Outliers of incorrect predictions were successfully replaced with feature engineering file ensembles obtained from a variety of engineering processes. The best model obtained from hyperparameter optimisation was used to obtain the deep feature, which is non-linear, as well as to derive new features and thus gave rise to multiple files based on the original features to have diversity for the ensemble. This was done in the form of data frames with all vectors and correlations with layers 1 and 2 being added. Numerous deep features were obtained, from which correlations are made with target features, giving rise to the maximum correlation heights. Only columns with height index correlations were kept. As such, those correlating with our targets were obtained.

Later, a modified version of PCA with kinetic correlation matrix using kinetic energy was presented. The features of this modified PCA were assessed with different sets of air passenger data and compared to traditional PCA. It emerged clearly that the modified version of PCA is more effective in data dimension reduction, classes reparability and classification accuracy. Satisfactory results were obtained from the modified PCA version as the predictions histogram in the training set was reassembled into a better histogram in the final predictions set. Moreover, there was a better replacement of the outliers while at the same time a decrease in the root mean squared

errors was marked. The results of the modified PCA version also showed that the kinetic correlation is much higher than the Pearson correlation. This is much more sensible owing to the fact that Pearson's R is only able to detect linear relations in any data.

This study also highlighted the possibility of airport authorities forecasting airline traffic figures on other frequencies other than on a monthly basis. This is because the models presented in this study can be successfully loaded with data measured at various frequencies or some advanced model can be applied as the data length increases. Additionally, the models described in this study can enable air authorities to predict future occurrences that could impact operations such as flight delays. This will help improve the management of operations to avoid losses and customer dissatisfaction.

7.4 Overall Contribution of the Study to the Knowledge of the Field

The major contribution of this Thesis is the advancement in the forecasting approaches and models that have been developed and tested for forecasting Oman air passenger demand. It goes without saying that the modified principal component analysis approach and optimised method of robust neural network technique are the newest and unique modelling paradigms. As such, these models have been successfully piloted in the Oman air passenger demand forecast. Additionally, this is the first reported study to employ MPCA models for forecasting airline passenger demand.

The models in this study have shown a high level of accuracy, reliability and a greater predictive capability in comparison to the traditional principal component analysis models (PCA), which are currently the recommended approaches of the International Civil Aviation Organisation and other key government agencies around the world. Therefore, this study has contributed to the current knowledge by:

1. Proposing an optimised method of the robust neural network technique with existing machine learning models;
2. Enabling extraction of new features from the original feature space;
3. Proving the significance of new features extracted on existing features through different experiments, i.e. deep neural network; and
4. Proposing the modified version of PCA, which is a statistical approach with kinetic correlation matrix using kinetic energy and forecasts the

number of airline passengers with a high level of accuracy in comparison to existing methods.

7.5 Suggestions for Future Research

With the primary outcome of this research being that models based on MPCA approaches are more effective than Pearson's traditional linear PCA models, it is essential for further research to be conducted to validate this work. This being the pioneering study, future researchers will know the challenges faced while undertaking similar studies and thus more easily mitigate them. To eradicate the challenge of dataset size limitation, future studies can base their case studies on larger airports with intensified operations. Additionally, based on these results, the modified PCA can be applied to make clustering in hyper-dimensional space using kinetic correlation as a distance (increasing performance), to make it run in real-time in future work. When handling various clustering categories such as a clustering algorithm, clustering K-means or in hierarchical clustering, future researchers should be advised that it requires a for-loop at every point to get the nearest point from row vector. This is because n rows of data complexity will be of the order n^n, which is impossible to finish unless using the said method. In a two-dimensional space, there is a trick to hasten implementation using a divide and conquer method, which has complexity n or log n. All these challenges can be mitigated with a modified PCA version containing all the latest features. Since this study employed only one dataset to study limited features of the modified PCA model, future studies should employ large sets and sub-sets of data with a wide variety of features to gain a full understanding of the model.

It is worth noting that forecasting tasks entail numerous dimensions such as the length of the forecast horizon, the size of the test set, forecast error measures, and the frequency of data. It is therefore unlikely that once selected, a particular forecasting method will be better than all other plausible methods all the time. As such, the probabilities of any forecast models should be frequently analysed depending on the current task as well as the availability of new datasets. This study was initiated keeping in mind the great need of forecasting models for the airports in Oman and finding the gaps in the literature related to non-usage of prediction interval around point forecasts. It is therefore believed this report will help Oman airports to build their inaugural forecasting models to track the future trends of air travellers and make intelligent and informed business decisions.

References

Önder, A., Candemir, A. & Kumral, N., 2009. An Empirical Analysis of the Determinants of International Tourism Demand: The Case of Izmir. *European Planning Studies*, pp.1525-33.

Önder, E. & Hasgül, O., 2009. Time Series Analysis with Using Box Jenkins Models and Artificial Neural Network for Forecasting Number of Foreign Visitors. *Journal of Institute of Business Administration*, pp.62-83.

Önder, E. & Kuzu, S., 2014. Forecasting Air Traffic Volumes using Smoothing Techniques. *Journal of Aeronautics and Space Technologies*, pp.65-85.

Adeniran, A.O. & Stephens, M.S., 2018. The Dynamics for Evaluating Forecasting Methods for International Air Passenger Demand in Nigeria. *Journal of Tourism & Hospitality*, pp.1-11.

Adhikari, R.&.A.R., 2013. An Introductory Study on Time series Modeling and Forecasting.. 10.13140/2.1.2771.8084..

Adrangi, B., Chatrath, A. & Raffiee, K., 2001. The demand of US air transport service: a chaos and nonlinearity investigation. *Transportation Research, Part E*, pp.337-53.

Airbus, 2014. *Flying by numbers.* [Online] Available at: http://www.airbus.com/company/market/forecast/.

Akaike, H., 1974. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, pp.716–23.

Alperovich, G. & Machnes, Y., 1994. The role of wealth in the demand for international air travel. *Journal of Transport Economics and Policy*.

Anderson, J.C..&.G.D.W., 1984. The effect of sampling error on convergence, improper solutions, and goodness-of-fit indices for maximum likelihood confirmatory factor analysis. *Psychometrika*, 49, pp.155-173..

Andreon, A. & Postorino, M.N., 2006. *A Multivariate ARIMA Model to Forecast Air Transport Demand*. Association for European Transport.

Armstrong, J.S. & Collopy, F., 1992. Error Measures for Generalizing About Forecasting Methods: Empirical Comparisons. *International Journal of Forecasting*, pp.69-80.

Assimakopoulos, V. & Nikolopoulos, K., 2000. The theta model: a decomposition approach to forecasting. *International Journal of Forecasting*, pp.521-30.

Automatic Forecasting Systems, Inc., 1999. *Autobox 5.0 Reference Guide 1999*. [Online] Available at: http://www.autobox.com/cms/.

Ba-Fail, , Abed, & Jasimuddin, S., 2000. The determinants of domestic air travel demand in the Kingdom of Saudi Arabia. *Journal of Air Transportation World Wide*.

Bahnsen, A., Stojanovic, A., Aouada, D. & Ottersten, B., 2016. Improving credit card fraud detection with calibrated probabilities. Philadelphia, USA, 2016. Proceedings of the fourteenth siam international conference on data mining.

Balasubramanian, V., Ho, S.-S. & Vovk, , 2014. *Conformal Prediction for Reliable Machine Learning: Theory, Adaptations and Applications*. Illustrated ed. Newnes Publishers.

Bar-Hillel, A., Hertz, T., Shental, N. & and Weinshall, D., 2005. Learning a mahalanobis metric from equivalenceconstraints. *Journal of Machine Learning Research*, 6, pp.937–965.

Barlas, Y., 1994. *Model Validation in SD , in SD: Exploring the Boundaries*. Scotland, UK, SD Society: Methodological and Technical Issues, Proceedings of International SD Conference, University of Stirling.

Bastien, F..P.L.R.P.B.I.J.G.A.B.-e.N.B.a.Y.B., 2012. *Theano: new features and speed improvements.Deep Learning and Unsupervised Feature LearningNIPS 2012 Workshop.*

Bates, J.&.G.C.J., 1969. The Combination of Forecasts. *Journal of the Operational Research Society*, 20(4), pp.451–68. https://doi.org/10.1057/jors.1969.103.

Bengio, Y.a.G.X., 2010. Understanding the dif-ficulty of training deep feedforward neural networks.. *InProceedings of AISTATS 2010*, 9, pp.249–256.

Bengio, Y., 2013. Representation learning: a review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp.1798-828.

Bishop, C.M., 1991. A fast procedure for retrainingthe multilayer perceptron. *International Journalof Neural Systems*, 2(3), pp.229–36.

Blinova, T.O., 2007. Analysis of possibility of using neural network to forecast passenger traffic flows in Russia.. *Aviation*, 11(1), pp.28-34.

Boccaletti, S. et al., 2014. The structure and dynamics of multilayer networks. *Phys. Rep.*, pp.1-122.

Bose, I. & Mahapatra, R.K., 2001. Business data mining—A machine learning perspective. *Information and Management,39(3)*, pp.211 – 225.

Box, G.E.P. & Cox, D.R., 1964. An analysis of transformations. *Journal of the Royal Statistical Society. Series B (Methodological)*, pp.211-52.

Box, G.E.P. & Jenkins, G.M., 1976. *Time Series Analysis: Forecasting and Control*. Revised Edition ed. San Francisco: Holden Day.

Box, G.E.P., Jenkins, M.G., Reinsel, & Ljung, G., 2015. *Time Series Analysis: Forecasting and Control*. 5th ed. John Wiley & Sons, Inc.

Breiman, L. & Friedman, J.H., 1985. Estimating optimal transformations for multiple regression and correlation. *Journal of the American Statistical Association*, pp.580-98.

Brosse, S., Lek, S. & Dauba, F., 1999. Predicting fish distribution in a mesotrophic lake by hydroacoustic survey and artificial neural networks. *Limnology and Oceanography*, pp.1293-303.

Brown, M. & Lowe, D.G., 2003. Recognising panoramas., 2003. Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003).

Burkov, A., 2019. *The Hundred-page Machine Learning Book*. Illustrated ed. Andriy Burkov.

Calvin, A.H.J., 2016. *Oman: the Modernization of the Sultanate*. Abingdon, New York: Routledge.

Carson, R.T., Cenesizolu, T. & Parker, R., 2011. Aggregate demand for USA commercial air travel. *Int. J. Forst.*, pp.923-41.

Carter, D.A., Daniel, A.R. & Betty, J.S., 2006. Does Hedging Affect Firm Value? Evidence from the US Airline Industry. *Financial Management*, pp.53-86.

Castellani, M., Mussoni, M. & Pattitoni, P., 2010. Air Passenger Flows – Evidence from Sicily and Sardinia. *Journal of Tourism, Culture and Territorial Development*, pp.16-28.

Chen, L., 2006. The Application of Wavelet analysis and neural network in Air Pollution Forecasting. *Int. J. Wirel. Mob. Comput.*, pp.608-14.

Chen, T.&.G.C., 2016. XGBoost: A Scalable Tree Boosting System.. pp.785-794.. 10.1145/2939672.2939785..

Cheng, W. et al., 2011. Automated feature generation from structured knowledge., 2011. Proceedings of the 20th ACM International Conference on Information and Knowledge Management.

Cheng, B. & Titterington, D.M., 1994. Neural Networks: A Review from a Statistical Perspective. *Statistical Science*, pp.2-30.

Chindanur, n.b.&.B.E., 2014. A moving-average filter based hybrid ARIMA–ANN model for forecasting time series data.. *Applied Soft Computing.* , 23, pp.27–38. 10.1016/j.asoc.2014.05.028..

Cho, V., 2003. A comparison of three different approaches to tourist arrival forecasting. *Tourism Management*, pp.323-30.

Cho, V., 2003. A comparison of three different approaches to tourist arrival forecasting. *Tourism Management*, pp.323-30.

Clark, A.E., Knabe, A. & Rätzel, S., 2009. Unemployment as a social norm in Germany. *Schmollers Jahr*, pp.251–60.

Cleveland, R.B. & Cleveland, W.S., 1990. STL: A Seasonal-Trend Decomposition Procedure Based on Loess. *Journal of Official Statistics*, pp.3-33.

Coates, A., Lee, H. & Ng, A.Y., 2011. An analysis of single-layer networks in unsupervised feature learning., 2011. Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics.

Coates, A. & Ng, A.Y., 2012. *Learning feature representations with k-means Neural Networks*. Berlin: Heidelberg: Springer.

Cook, A., 2007. *European Air Traffic Management: Principles, Practice and Research*. Aldershot: Ashgate.

Coshall, J., 2006. Time Series Analyses of UK Outbound Travel by Air. *Journal of Travel Research*, pp.335-47.

Cryer, J.D. & Chan, K.-S., 2008. *Time Series Analysis*. 1st ed. Springer.

Cryer, J.D. & Chan, K.S., 2008. *Time Series Analysis*. 1st ed. Springer.

Cuayáhuitl, H., 2016. impleds: A simple deep reinforcement learning dialogue system. *arXiv preprint arXiv:1601.04574*.

Cuhadar, M., 2014. Modelling and Forecasting Inbound Tourism Demand to Istanbul: A Comparative Analysis. *European Journal of Business and Social Sciences*, pp.101-19.

Cullen, K.E. & Kusky, T.M., 2010. *Arabian geology*. Encyclopedia of Earth and Space Science. New York City: Infobase Publishing.

D Brzezinski, J.S., 2014. "Reacting to different types of concept drift: The accuracy updated ensemble algorithm[J]", Neural Networks and Learning Systems IEEE Transactions on. 25(1), pp.81-94.

Dangeti, P., 2017. *Statistics for Machine Learning*. Packt Publishing Ltd.

Dargay, J. & Hanly, M., 2001. The determinants of the demand for international air travel to and from UK. *ESRC Transport Studies Unit, Centre for Transport Studies, University College London*, pp.1-14.

Davis, J.J. & Foo, E., 2016. Automated feature engineering for HTTP tunnel detection. *Computers & Security*, pp.166-85.

Davis, J.J. & Foo, E., 2016. Automated feature engineering for HTTP tunnel detection. *Computers & Security*, pp.166-85.

De Boer, P.-T..D.P.K..M..a.R.Y.R., 2005. A tutorial onthe cross-entropy method. *Annals of operations research*, 134(1), pp.19–67.

De Menezes, L.M., Bunn, D.W. & Taylor, J.W., 2000. Review of guidelines for the use of combined forecasts. *European Journal of Operational Research*, pp.190-204.

De Vany, A., 1975. The Effect of Price and Entry Regulation on Airline Output, Capacity and Efficiency. *Bell Journal of Economics*, pp.327-45.

Deutsch, M., Granger, C.W.J. & Teräsvirta, T., 1994. The combination of forecasts using changing weights. *International Journal of Forecasting*, pp.47-57.

Domingos, P., 2012. A few useful things to know about machine learning. *Communications of the ACM*, pp.78-87.

Donaldson, R.G. & Kamstra, M., 1996. Forecast combining with neural networks. *Journal of Forecasting*, pp.49-61.

Dong, G. & Liu, H., 2018. *Feature Engineering for Machine Learning and Data Analytics*. illustrated ed. CRC Press.

Duval, D.T., 2008. Regulation, competition and the politics of air access across the pacific. *Journal of Air Transport Management*, 14, pp.237-42. Available at: https://doi.org/10.1016/j.jairtraman.2008.04.009.

Efron, B., 1979. Bootstrap Methods: Another Look at the Jackknife. *Ann. Statist.*, pp.1-26.

Efron, B., 1983. Estimating the Error Rate of A prediction Rule: Improvements in Cross-validation. *J. Amer. Statist. Assoc.*, pp.316-31.

EIA , 2014. *U.S. Energy Information Administration, Benchmarks play an important role in pricing crude oil*. [Online] Available at: http://www.eia.gov/todayinenergy/detail.cfm?id=18571 [Accessed 7 May 2019].

Emmanuel J. Candès, X.L.Y.M.a.J.W., 2011. Robust principal component analysis. *J. ACM* , 58(3), pp.1-37. DOI=http://dx.doi.org/10.1145/1970392.1970395.

Essa, A. & Ayad, H., 2012. Student success system: Risk analytics and data visualization using ensembles of predictive models. *in Proc. 2nd International Conference on Learning Analytics and Knowledge*, pp.158–61.

Eurocontrol, 2009. *Challenges of Air Transport 2030: Survey of Experts' Views*. [Online] Available at: https://www.eurocontrol.int/eec/gallery/content/public/document/eec/other_document/2009/003_Challenges_of_air_transport_2030_experts_view.pdf.

Eurocontrol, 2014. *Airspace modelling.*. [Online] Available at: https://www.eurocontrol.int/articles/airspace-modelling [Accessed 21 December 2018].

Faisal, 2002. *Analisis Time Series Lalulintas Angkutan Udara Internasional di Indonesia*.

Fayyad, U., Piatetsky-Shapiro, G. & Smyth, P., 1996. "From Data Mining to Knowledge Discovery: An Overview". In U. Fayyad, G.P.-S.P.S.R.U.e. *Advances in Knowledge Discovery and Data Mining*. MIT Press, Cambridge, Mass. 1996. pp.1-36.

Feldhoff , J.F. et al., 2012. Comparative System Analysis of Direct Steam Generation and Synthetic Oil Parabolic Trough Power Plants With Integrated Thermal Storage. *Solar Energy*, pp.520-30.

Findley, et al., 1998. Topographic diversity of fungal and bacterial communities in human skin. *Nature*, pp.367-70.

Findley, et al., 2014. Topographic diversity of fungal and bacterial communities in human skin. *Nature*, pp.367-70.

Freeman, M.F. & Tukey, J., 1950. Transformations Related to the Angular and the Square Root. *The Annals of Mathematical Statistics*, pp.607-11.

Friedman, J., 2000. Greedy Function Approximation: A Gradient Boosting Machine.. *The Annals of Statistics.*, 29. 10.1214/aos/1013203451..

Friedman, J.&.T.R.&.H.T., 2001. Additive Logistic Regression: A Statistical View of Boosting (With Discussion and a Rejoinder by the Authors). *Annals of Statistics - ANN STATIST.*, 28, pp.337-407. 10.1214/aos/1016120463.

Friedman, J.H. & Popescu, B.E., 2008. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, pp.916–54.

Gábor J Székely, M.L.R., 2005. A new test for multivariate normality. *Journal of Multivariate Analysis*, 1(93), pp.58-80.

Gómez, V. & Maravall, A., 1996. *Programs TRAMO and SEATS, Instruction for User*. Banco de España.

Gómez, V. & Maravall, A., 1997. *Programs TRAMO and SEATS, Instruction for User*. Banco de España.

Gómez, V. & Maravall, A., 1997. *Programs TRAMO and SEATS, Instruction for User*. Banco de España.

Gómez, V. & Maravall, A., 2000. *Automatic Modeling Methods for Univariate Series*. Banco de España.

Gardner, E.S.J. & Mckenzie, E., 1985. Forecasting Trends in Time Series. *Management Science*, pp.1237-46.

Gardner, E.S.J. & Mckenzie, E., 1985. Forecasting Trends in Time Series. *Management Science*, pp.1237-46.

Gardner, E.S.J. & Mckenzie, E., 1988. Model Identification in Exponential Smoothing. *Journal of Operational Research Society*, pp.863-67.

Gardner, E.S.J. & Mckenzie, E., 1988. Model Identification in Exponential Smoothing. *Journal of Operational Research Society*, pp.863-67.

Gardner, E.S.J. & Mckenzie, E., 1989. Seasonal Exponential Smoothing with Damped Trends. *Management Science*, pp.372-76.

Gardner, E.S.J. & Mckenzie, E., 1989. Seasonal Exponential Smoothing with Damped Trends. *Management Science*, pp.372-76.

Gesell, L.E., 1993. *The administration of public airports*. 3rd ed. Chandler, AZ: Coast Aire Publications1993.

Girolami, M., 2011. *A First Course in Machine Learning*. Illustrated ed. CRC Press.

Glorot, X. & Bengio, Y., 2011. Deep sparse rectifier neural networks., 2011. International Conference on Artificial Intelligence and Statistics.

Gomez, V. & Maravall, A., 1997. *Programs TRAMO and SEATS: instructions for the user*. Mimeo, Banco de España.

Goodrich, R.L., 1984. FOREX: A time Series Forecasting Expert System., 1984. Fouth International Symposium on Forecasting.

Goodrich, R.L., 1986. FOREX: A time Series Forecasting Expert System., 1986. Sixth International Symposium on Forecasting.

Goodrich, R.L., 2000. The Forecast Pro methodology. *International Journal of Forecasting*, pp.533-35.

Goodrich, R.L., 2001. Commertial Software in the M 3-Competition. *International Journal of Forecasting*, pp.537-84.

Gosavi, A.a.B.N.a.D.T.K., 2002. A reinforcement learning approach to a single leg airline revenue management problem with multiple fare classes and overbooking. *IIE Transactions* , 34(9), pp.729-742.

Gosavii, A., Bandla, N. & Das, T.K., 2002. A reinforcement learning approach to a single leg airline revenue management problem with multiple fare classes and overbooking. *IIE transactions*, pp.729–42.

Grosche, T., Rothlauf, F. & Heinzl, A., 2007. Gravity models for airline passenger volume estimation. *Journal of Air Transport Management*, pp.175-83.

Grus, J., 2015. *Data Science from Scratch: First Principles with Python*. re-print ed. "O'Reilly Media, Inc.

Gujarati, D.N. & Porter, D.C., 2009. *"Panel Data Regression Models". Basic Econometrics (Fifth international ed.)*.. Boston: McGraw-Hill. ISBN 978-007-127625-2.

Gut, A., 2009. An Intermediate Course in Probability. *Springer*.

Guyon, I., Gunn, S., Nikravesh, M. & Zadeh, L., 2008. *Feature Extraction: Foundations and Applications*. Illustrated ed. Springer.

Hahnloser, R.&.S.R.&.M.M.&.D.R., 2000. Digital selection and analog amplification co-exist in an electronic circuit inspired by neocortex.. *Nature.* .

Hair, J.&.S.M.&.H.L.&.K.V., 2014. Partial Least Squares Structural Equation Modeling (PLS-SEM): An Emerging Tool for Business Research. European Business Review. 26, pp.106-21. 10.1108/EBR-10-2013-0128..

Han, D., Al-Jawad, N. & Du, H., 2016. Facial Expression Identification Using 3D Geometric Features from Microsoft Kinect Device. Baltimore, Maryland, United States, 2016. SPIE Commercial + Scientific Sensing and Imaging.

Hannan, E.J. & Rissanen, J., 1982. Recursive Estimation of Mixed Autoregressive-Moving Average Order. *Biometrika*, pp.81-94.

Hassanien, A.E. & Gaber, T., 2017. *Handbook of Research on Machine Learning Innovations and Trends*. Illustrated ed. IGI Global.

Hastie, T., Tibshirani, R. & Friedman, J., 2013. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. illustrated ed. Springer Science & Business Media.

Heij, C..P.J.F.G.a.D.v.D., 2006. Time series forecasting by principalcovariate regression 2006-37. *Econometric Institute Report*.

Hill, R.C., Griffiths, W.E. & Judge, G.C., 2001. *Undergraduate Econometrics*. 2nd ed. New Jersey: Jhon Wiley & Sons, Inc.

Hochreiter, S., 1991. Untersuchungen zu dynamischen neuronalen Netzen. Diploma thesis, Institut fuer Informatik, Lehrstuhl Prof. Brauer, Tech. Univ. Munich. Advisor: J. Schmidhuber.

Holt, C.C., 1957. Forecasting Seasonals and Trends by Exponentially Weighted Moving Averages. *ONR Memorandum*.

Hornik, K..S.M..&.W.H., 1990. Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks.. *Neural Networks*, 2, pp.359-66.

Huang S.Y., Y.Y.R.a.d.E.g.S., 2009. Robust kernel principal component analysis. Neural Computation. 21(11), pp.3179-213.

Huang, Z..W.X..G.A.J..F.T.J..T.A.J.., 2013. An open-access modeled passen-gerflow matrix for the global air network in 2010.. *PLoS ONE*, 8.

Huang, C. et al., 2015. On using smoothing spline and residual correction to fuse rain gauge observations and remote sensing data. *Journal of Hydrology*, pp.410-17.

Hui, C., Lee, C. & Rousseau, D.M., 2004. Psychological Contract and Organizational Citizenship Behavior in China: Investigating Generalizability and Instrumentality. *The Journal of applied psychology*, pp.311-21.

Hwang, C.C. & Shiao, G.C., 2011. Analyzing air cargo flows of international routes: an empirical study of Taiwan Taoyuan International Airport. *Journal of Transport Geography*, pp.738–44.

Hyndman, R.J. & Athanasopoulos, G., 2013. *Forecasting: principles and practice*. OTexts.org/fpp/.

Hyndman, R.J. & Khandakar, Y., 2008. Automatic Time Series Forecasting: The forecast Package for R. *Journal of Statistical Software*, pp.1-22.

Hyndman, R.J. & Koehler, A., 2006. Another Look at Measures of Forecast Accuracy. *International Journal of Forecasting*, pp.679-88.

IATA, 2010. *International Air Transport Association. IATA Monthly Jet Fuel Cost and Consumption Report.*. [Online] Available at: http://www.airlines.org/Energy/FuelCost/Pages/MonthlyJetFuelCostandConsumptionReport.aspx [Accessed 13 May 2018].

IATA, 2014. *Fact Sheet: Fuel, Report published December 2014*. Montreal, Canada: The International Air Transport Association.

IATA, 2017. *62nd Annual Report on World Air Transport Statistics*. [Online] Available at: https://www.iata.org/publications/store/Pages/world-air-transport-statistics.aspx [Accessed 5 May 2019].

IATA, 2017. *62nd Annual Report on World Air Transport Statistics*. [Online] Available at: https://www.iata.org/publications/store/Pages/world-air-transport-statistics.aspx [Accessed 5 May 2019].

IATA, 2018. *Traveler Numbers Reach New Heights*. [Online] Available at: https://www.iata.org/pressroom/pr/Pages/2018-09-06-01.aspx [Accessed 7 May 2019].

IBRD-IDA, 2012. *The World Bank IBRD-IDA: World Development Indicators*. [Online] Available at: http://data.worldbank.org/data-catalog/world-development-indicators.

ICAO, 2005. *The economic & social benefits of air transport. Montreal, Quebec, Canada.*. Montreal: Air Transport Action Group.

ICAO, 2010. *Environmental Report 2010. Aviation and Climate Change*. International Civil Aviation Organization. https://www.icao.int/environmental-protection/Documents/Publications/ENV_Report_2010.pdf.

Ildefons, M.D.A. & Sugiyama, M., 2013. Winning the Kaggle Algorithmic Trading Challenge with the Composition of Many Models and Feature Engineering. *IEICE transactions on information and systems*, pp.742-45.

Iosifescu, M., 1986. Octav Onicescu, 1892-1983. International Statistical Review / Revue Internationale De Statistique. 54(1), pp.97-108.. Retrieved from http://www.jstor.org/stable/1403261.

Ippolito, R.A., 1981. Estimating airline demand with quality of service variables. *J. Transp. Econo. Poli.*, pp.7-15.

J. Zhao, P.L.H.Y.a.J.T.K., 2012. Bilinearprobabilistic principal component analysis.IEEE Trans. on Neural Networks and Learning Systems.. 23(3), pp.492–503.

Jackson, S.L., 2012. *Research Methods and Statistics: A Critical Thinking Approach*. 4th ed. Cengage Learning.

Jonga, G., Gunnc, H. & Akiva, M.B., 2004. A meta-model for passenger and freight transport in Europe. *Transport Policy*, pp.329–44.

Kanter, J.M. & Veeramachaneni, , 2015. Deep feature synthesis: towards automating data science endeavors., 2015. IEEE International Conference on Data Science and Advanced Analytics (DSAA).

Kelleher, J.D., Mac-Namee, B. & D'Arcy, , 2015. *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*. illustrated ed. MIT Press.

Kern, R., 2014. *Feature Engineering*. [Online] Available at: http://kti.tugraz.at/staff/denis/courses/kddm1/featureengineering.pdf.

Kim, K.W., Seo, H.Y. & Kim, Y., 2003. Forecast of domestic air travel demand change by opening the high speed rail. *KSCE J. Civil Eng.*, pp.603-09.

Kincaid, I.S., 2016. *Addressing Uncertainty about Future Airport Activity Levels in Airport Decision Making (TRB's Airport Cooperative Research Program (ACRP) Report 76)*. Transportation Research Board.

Koldovský, Z.&.T.P.&.O.E., 2006. Efficient Variant of Algorithm FastICA for Independent Component Analysis Attaining the CramÉr-Rao Lower Bound. IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council. 17. 1265-77.. 10.1109/TNN.2006.875991..

Koo, T..T.D.a.D.D., 2018. The Effect of Levels of Air Service Availability on Inbound tourism demand from Asia to Australia", Airline Economics in Asia.. *Advances in Airline Economics*, 7, pp.45-167. https://doi.org/10.1108/S2212-160920180000007009.

Koopman, S.J.&.O.M., 2006. Forecasting daily time series using periodic unobserved components time series models.. *Computational Statistics & Data Analysis.*, 51, pp.885-903.. 10.1016/j.csda.2005.09.009..

Kruger, U.a.Z.J.a.X.L., 2008. *Developments and Applications of Nonlinear Principal Component Analysis –a Review (inproceedings)*. Berlin, Heidelberg: Springer Berlin Heidelberg.

Kuhn, M. & Johnson, K., 2019. *Feature Engineering and Selection: A Practical Approach for Predictive Models*. CRC Press.

Kulendran, N. & Witt, S., 2003. Forecasting the Demand for International Business Tourism. *Journal of Travel Research*, pp.265-71.

Kwak, N., 2008. Principal component analysis based on L1-normmaximization.IEEE Transactions on Pattern Analysis and Ma-chine Intelligence,. pp.1672–80. doi:10.1109/TPAMI.2008.114.

Lütkepohl, H., 2013. *Introduction to Multiple Time Series Analysis*. Illustrated ed. Springer Science & Business Media.

Landset, S.&.K.T.&.R.A.&.H.T., 2015. A survey of open source tools for machine learning with big data in the Hadoop ecosystem.. *Journal of Big Data.*. 2. 10.1186/s40537-015-0032-1..

Landset, S. & Khoshgoftaar, T.M., 2015. A survey of open source tools for machine learning with big data in the Hadoop ecosystem. *Journal of Big Data*.

Lasmita, C.Y., 2010. *The Patterns of Air Traffic Movements in Adi Sutjipto Airport*.

Lazarevic, A.&.K.V., 2005. Feature bagging for outlier detection. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 21, pp.157-166. 10.1145/1081870.1081891.

Le, Q.V., 2013. Building high-level features using large scale unsupervised learning., 2013. Speech and Signal Processing (ICASSP), IEEE International Conference on Acoustics.

Lee, S.H., 2009. The networkability of cities in the international air passenger flows 1992–2004. *Journal of Transport Geography*, pp.166-75.

Lemke, C.&.R.S.&.G.B., 2009. Dynamic combination of forecasts generated by diversification procedures applied to forecasting of airline cancellations.. *IEEE/IAFE Conference on Computational Intelligence for Financial Engineering, Proceedings (CIFEr).* , pp.85 - 91. 10.1109/CIFER.2009.4937507..

Li, X..P.Y..&.Y.Y., 2010. L1-Norm-Based 2DPCA. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics). 40, pp.1170-1175.

Li, Y., Chen, C.Y. & Wasserman, W., 2016. Deep Feature Selection: TheoryAnd Application To Identify Enhancers And Promoters. *Journal ofComputational Biology*, pp.322-36.

Lim, C. & McAleer, M., 2001. Forecasting Tourist Arrival to Australia Using Smoothing Methods. *Annals of Tourism Research*, pp.965-77.

Linfoot, E., 1957. "An Informational Measure of Correlation."Information and Control. 1, pp.85–89.

Liu, L.M., 1989. Identification of Seasonal Arima Models Using A filtering Method. *Communications in Statistics - Theory and Methods*, pp.2279-88.

Liu, X., Lin, X. & Wang, H., 2008. Novel online methods for time series segmentation. *TKDE*, pp.1616-26.

Lloyd, J.R. et al., 2014. Automatic construction and natural-language description of nonparametric regression models. *arXiv:1402.4304.*.

Lowe, D.G., 1999. Object recognition from local scale-invariant features., 1999. The Proceedings of the Seventh IEEE International Conference on Computer Vision.

Lu, H..P.K.N..&.V.A.N., 2008. MPCA: Multilinear Principal Component Analysis of Tensor Objects.. *IEEE Transactions on Neural Networks.*, 19, pp.18-39.

Lu, C.-J., Lee, T.-S. & Chiu, C.-C., 2009. Financial time series forecasting using independent component analysis and support vector regression. *Decision Support Systems*, pp.115-25.

Mélard, G. & Pasteels, J., 2000. Automatic ARIMA Modeling Including Interventions, Using Time Series Expert Software. *International Journal of Forecasting*, pp.497-508.

Makridakis, S. & Hibon, M., 2000. The M3-Competition: results, conclusions and implications. *International Journal of Forecasting*, pp.451-76.

Makridakis, S.G., Wheelwright, S.C. & Hyndman, R., 1998. *Forecasting: Methods and Applications*. 3rd ed. John Wiley & Sons, Inc.

Market Research.com, 2017. *Travel & Leisure Market Research Reports & Industry Analysis*. [Online] Available at: https://www.marketresearch.com/Consumer-Goods-c1596/Travel-Leisure-c90/.

Markovitch, & Rosenstein, D., 2002. Feature Generation Using General Constructor Functions. *Machine Learning*, pp.59-98.

Mason, K.J., 2005. Observations of fundamental changes in the demand for aviation services. *Journal of Air Transport Management*, pp.19–25.

Matsumoto, H., 2004. International urban systems and air passenger and cargo flows: some calculations. *Journal of Transport Geography*, pp.197–206.

Matthiessen, C.W., 2004. International air traffic in the Baltic Sea Area: Hub-gateway status and prospects. Copenhagen in focus. *Journal of Transport Geography*, pp.197-206.

McGregor, A., Hall, M., Lorier, P. & Brunskill, J., 2004. *Flow clustering using machine learning techniques*. Passive and Active Network.

McKnight, P., 2010. *Airline economics. Introduction to Aviation Management, LIT Verlag, Germany: 25-53.* In book (ed. Wald A., Fay C., Gleich R.).

Meng, D.&.Z.Q.&.X.Z., 2011. Improve robustness of sparse PCA by L1-norm maximization. Pattern Recognition.. 45, pp.487-497.. 10.1016/j.patcog.2011.07.009..

Michalski, R.S., Carbonell, J.G. & Mitchell, T.M., 2014. *Machine Learning: An Artificial Intelligence Approach, Volume 1*. Edited ed. Elsevier.

Mills, T., 1993. *The Econometric Modelling of Financial Time Series*. Cambridge: CambridgeUniversity Press.

Mirkin, B., 2011. *Core Concepts in Data Analysis: Summarization, Correlation and Visualization*. Illustrated ed. Springer Science & Business Media.

MIT, 2015. *Airline industry overview.* [Online] Available at: http://web.mit.edu/airlines/analysis/analysis_airline_industry.html [Accessed 21 December 2018].

Morgan, J.P., 2001. *Low-fare airline industry*. Industry analysis from J.P.Morgan Securities Ltd.

Motoda, H. & Liu, H., 2002. Data reduction: feature selection. *Data Mining and Knowledge Discovery - DATAMINE*, pp.208-13.

Mubarak, T., 2014. *Airport Passenger Demand Forecasting Using Radial Basic Function Neural Networks*.

Muhammad, A.M., Haswadi, H., Dewi, N. & Habibollah, H., 2015. A review on feature extraction and feature selection for handwritten character recognition. *International Journal of Advanced Computer Science & Applications*, pp.204-12.

Nam, K. & Schaefer, , 1995. Forecasting international airline passenger traffic using neural networks. *Logistics and Transportation Review*, pp.239-51.

NCSI, 2017. *Foreign Investment Survey in Oman*. [Online] Available at: https://www.ncsi.gov.om/Pages/AllIndicators.aspx.

NCSI, 2017. *Oman National Center for Statistics and Information 2017*. [Online] Available at: https://www.ncsi.gov.om/Elibrary/LibraryContentDoc/bar_Statistical%20Year%20Book%202017_c2111831-e13a-4075-bf7b-c4b5516e1028.pdf [Accessed 7 May 20195].

Nixon, M., 2013. *Feature Extraction and Image Processing*. Elsevier publishers.

Nunnally, J..B.I., 1994. Psychometric Theory (3rd ed.). New York:McGraw-Hill.

OAMC, 2017. *Oman Airport Management Company: The History of Oman Aviation*. [Online] Available at: https://www.omanairports.co.om/content/oman-airports-history.

Oman Airports, 2018. *ABOUT OMAN AIRPORTS*. [Online] Available at: Available at:https://www.omanairports.co.om/en [Accessed 7 May 2019].

Omanair, 2011. *Annual report 2011*. [Online] Available at: https://www.omanair.com/sites/default/files/content/about_us/pdf/2011annualreport_eng.pdf [Accessed 7 May 2017].

Omanair, 2017. *Annual Report 2017: Going the Extra Mile to Build the Hub*. [Online] Available at: https://www.omanair.com/sites/default/files/content/about_us/pdf/2017annualreport_eng.pdf [Accessed 7 May 2019].

ONA, 2016. *Oman Air Offers 23,000 More Seats to Salalah*. [Online] Available at: http://omannews.gov.om/ona_en/description.jsp?newsId=287281.

Opitz, D. & Maclin, R., 1999. Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research, Vol. 11*, pp.169–98.

Oxford Economics, 2011. *An Alternative APD regime*. [Online] Available at: https://www.iata.org/about/Documents/iata-annual-review-2014.pdf [Accessed 7 May 2019].

Parzen, E., 1982. ARARMA Models for Time Series Analysis and Forecasting. *J. Forecast.*, pp.67-82.

Perner, P., 2012. *Machine Learning and Data Mining in Pattern Recognition: 8th International Conference, MLDM 2012, Berlin, Germany, July 13-20, 2012, Proceedings*. Illustrated ed. Petra Perner.

Peterson, C., 2000. The Future of Optimism. The American psychologist. 55, pp.44-55. 10.1037/0003-066X.55.1.44..

Phyoe, S..L.Y..&.Z.Z., 2016. Determining future demand: studies for air traffic forecasting. In 11th International Conference on Engineering & Technology, (ECBA-2016). *Computer, Basic & Applied Sciences* . Retrieved from http://kkgpublications.com/ijtes-volume2-issue3-article1-4/.

Profillidis, V.A., 2000. Econometric and Fuzzy Models for the Forecast of Demand in the Airport of Rhodes. *Journal of Air Transport Management*, pp.95-100.

Pupavac, D., 2009. *Načela ekonomike prometa, [Principles of Transport Economics]*. Rijeka, Veleučilište u Rijeci, [Polytechnics of Rijeka].

Quandt, R.E., 2006. Measurement and inference in wine tasting. *ournal of Wine Economics*, 1, pp.7–30.

Rahman, R., 2019. *Applications of Deep Learning Models for Traffic Prediction Problems*. University of Central Florida.

Rajaraman, A. & Ullman, J.D., 2011. *Mining of massive datasets*. 2nd ed. Cambridge University Press.

Riedel, S.&.G.B.J., 2007. VLSI Sign Process Syst Sign Im 49: 265.. https://doi.org/10.1007/s11265-007-0076-3.

Riedel, S. & Gabrys, B., 2003. Adaptive mechanisms in an airline ticket demand forecasting system. Oulu, Finland, 2003. in Proc. EUNITE'2003 Conference: European Symposium on Intelligent Technologies, Hybrid Systems and their Implementation on Smart Adaptive Systems.

Riga, M., Tzima, F.A., Karatzas, K. & Mitkas, P.A., 2009. Development and Evaluation of Data Mining Models for Air Quality Prediction in Athens, Greece. *Environmental Science and Engineering*, pp.331-44.

Rizescu, D. & Avram, V., 2014. *Using Onicescu's informational energy to approximate social entropy*. Procedia-Social and Behavioral Sciences.

Rzempoluck, E.J., 2012. *Neural Network Data Analysis Using SimulnetTM*. Illustrated ed. Springer Science & Business Media.

S. M. Phyoe, N.Y.N.S.A.a.Z.W.Z., 2016. The impact of population growth on the future air traffic demand in Singapore.. In P. A. Kowalski, S.L.a.P.K., ed. *International Conference on Computer Networks and Communication Technology, ACSR-Advances in Computer Science Research*., 2016.

Sabatelli, L., 2016. Relationship between the Uncompensated Price Elasticity and the Income Elasticity of Demand under Conditions of Additive Preferences. *PLoS ONE*, p.e0151390.

Sapankevych, N.&.S.R., 2009. Time Series Prediction Using Support Vector Machines: A Survey.. *Computational Intelligence Magazine, IEEE.* , 4, pp.24 - 38.. 10.1109/MCI.2009.932254..

Schölkopf, B.&.S.A.&.M.K.-R., 1998. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. Neural Computation.. 10. 1299-1319. 10.1162/089976698300017467.

Schwarz, G., 1978. Estimating the Dimension of a Model. *Annal of Statistics*, pp.461-64.

Scott, S. & Matwin, S., 1999. Feature engineering for text classification., 1999. Proceedings of the Sixteenth International Conference on Machine Learning (ICML 1999).

Sengupta, P., Tandale, M., Cheng, V. & Menon, P., 2011. Air Traffic Estimation and Decision Support for Stochastic Flow Management. In *American Institute of Aeronautics and Astronautics Guidance, Navigation, and Control Conference*. Portland, Oregon , 2011.

Seraj, Y.A., Abdullah, O.B. & Sajjad, M.J., 2001. An econometric analysis ofinternational air travel demand in Saudi Arabia. *Journal of Air TransportManagement*, pp.143–48.

Shi, S., Da, X.L. & Liu, B., 1999. Improving the accuracy of nonlinear combined forecasting using neural networks. *Exp. Systems with Applications*, pp.49-54.

Shi, S. & Liu, B., 1993. Nonlinear combination of forecasts with neural networks., 1993. Proc.of 1993 International Joint Conference on Neural Networks.

Spence, A.M., 1975. Monopoly,QualityandRegulation. *Bell Journal of Economics*, pp.417-29.

Stock, J.H. & Watson, M.W., 2004. Combination forecasts of output growth in a seven-country data set. *Journal of Forecasting*, pp.405-30.

Suryan, V., Sinha, A., Malo, P. & Deb, K., 2016. *Handling inverse optimal control problems using evolutionary bilevel optimization*. Vancouver, BC: IEEE Congress on Evolutionary Computation (CEC).

Sutskever, I., Martens, J., Dahl, G. & Hinton, G., 2013. On the importance of initialization and momentum in deep learning., 2013. Proceedings of the 30th International Conference on Machine Learning (ICML-13).

Taneja, K., 1987. *Airline Traffic Forecasting.D.C.* Heath and Company.

Taylor, J.W., 2003. Exponential Smoothing with A Damped Multiplicative Trend. *International Journal of Forecasting*, pp.715-25.

Terui, N. & van Dijk, H.K., 2002. Combined forecasts from linear and nonlinear time series models. *International Journal of Forecasting*, pp.421-38.

Timmerman, M.E., 2003. Principal component analysis. *Journal of the American Statistical Association*, pp.1082-108.

Timmerman, M.E., 2003. Principal component analysis. *Journal of the American Statistical Association*, pp.1082-108.

Tinsley, H.E..&.T.D.J., 1987. Uses of factor analysis in counseling psychology research. Journal of Counseling Psychology. 34(4), pp.414-424.. http://dx.doi.org/10.1037/0022-0167.34.4.414.

Tipping, M.E.a.B.C.M., 1999. Probabilistic principal com-ponent analysis.. *Journal of the Royal Statistical Society*, 61, pp.611–22.

Tsui, W.H.K., Balli, O. & Gower, H., 2011. Forecasting airport passenger traffic: the case of Hong Kong International Airport., 2011. Aviation Education and Research Proceedings.

Tsui, W.H.K., Balli, O.H. & Gower, H., 2011. Forecasting airport passenger traffic: the case of Hong Kong International Airport s.l., 2011. Aviation Education and Research Proceedings.

Tsui, W.H.K., Gilbey, A. & Balli, H., 2014. Estimating airport efficiency of New Zealand airports. *Journal of Air Transport Management*, pp.78-86.

Turk, & Pentland, , 1991. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, pp.71-86.

UNWTO, 2016. *International tourist arrivals up 4% in the first half of 2016*. [Online] Available at: http://media.unwto.org/press-release/2016-09-26/international-tourist-arrivals-4-first-half-2016.

van der Maaten L, H.G., 2008. Visualizing Data using t-SNE.. *Journal of Machine Learning Research*, 9, pp.2579–605.

Van der Maaten, L. & Hinton, G., 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, pp.2579-605.

Vasigh, B. & Fleming, K., 2016. *Introduction to Air Transport Economics: From Theory to Applications*. Revised ed. Routledge Publishers.

Wadud, Z., 2011. Modelling and Forecasting Passenger Demand for a New Domestic Airport with Limited Data. Transportation Research Record. *Journal of the Transportation Research Board*, pp.59-68.

Wadud, Z., 2013. Simultaneous Modelling of Passenger and Cargo Demand at an Airport. Transportation Research Record. *Journal of the Transportation Research Board*, pp.63-74.

Wang, B. et al., 2014. Future Change of Asian-Australian Monsoon Under RCP 4.5 Anthropogenic Warming Scenario. *Climate Dynamics*, pp.83-100.

Weatherford, L.R., Gentry, T.W. & Wilamowski, , 2003. Neural network forecasting for airlines: A comparative analysis. *Journal of Revenue and Pricing Management*, pp.319-31.

Wensveen, J.G., 2007. *Air transportation: A management perspective, 6th ed.* 6th ed. Farnham, Ashgate, Surrey, United Kingdom.

Wensveen, J.G., 2011. *Air transportation: a management perspective*. 7th ed. Farnham: Ashgate Publishin.

World Bank, 2016. *Monthly Oman World Bank Data crude oil prices 1989 –2016*. [Online] Available at: https://data.worldbank.org/country/oman [Accessed 7 May 2019].

WTTC, 2014. *Economy of Oman*. [Online] Available at: https://www.wttc.org/datagateway.

Yang, J..Z.D..F.A.F.&.y.Y.J., 2004. Two-dimensional PCA: A new approachto appearance-based face representation and recognition,IEEE Trans. *Pattern Anal.and Mach. Intell.*, pp.131–137.

Yang, Q. & Wu, X., 2006. 10 challenging problems in data mining research. *Int J Inform Technol Decision Making, vol. 5, no.4*, pp.597–604.

Yifeng, L..C.-Y.C..a.W.W.W., 2016. *Deep Feature Selection: Theory and Applicationto Identify Enhancers and Promoters*. Centre for Molecular Medicine and TherapeuticsUniversity of British Columbia.

Yu, H.-F. et al., 2011. Feature engineering and classifier ensemble for KDD cup 2010., 2011. JMLR: Workshop and Confrence Proceedings.

Yu, L., Wang, S. & Lai, K., 2009. A Neural-Network-based Nonlinear Metamodeling Approach to Financial Time Series Forecasting. *Applied Soft Computing*, pp.563-74.

Z. W. Zhong, Y.Y.T.a.Y.J.L., 2015. Studies of air transport management issues for the airport and region.. *In ISPE CE*, 2, pp.533-40.

Z. W. Zhong, R.S.S.W.X.C.a.Z.M.O., 2016. Studies of air traffic forecasts, airspace load and the effect of ADS-B via satellites on flight times," in Advanced Free-Space Optical Communication Techniques and Applications II. *SPIE*, 9991(2). Article Number: UNSP 99910B.

Z. W. Zhong, R.S.S.W.X.C.a.Z.M.O., 2016. Studies of air traffic forecasts, airspace load and the effect of ADS-B via satellites on flight times,in Advanced Free-Space Optical Communication Techniques and Applications II, Proceedings of SPIE.. 9991. Article Number: UNSP 99910B.

Zhang, P.G., 2003. Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, pp.159-75.

Zhang, Z..X.G..&.L.L., 2009. Latent Variable Models for Dimensionality Reduction. Proceedings Track. 5.. *Journal of Machine Learning Research* , pp.655-662..

Zhang, G., Patuwo, B.E. & Hu, M.Y., 1998. Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, pp.35-62.

Zhang, et al., 2016. Nonlinear coupling of flexural mode and extensional bulk mode in micromechanical resonators. *Appl. Phys. Lett.*, p.224102.

Zheng C, e.a., 2011. Identification and quantification of metabolites in 1H NMR spectra by Bayesian model selection. Bioinformatics.27:1637.

Zheng, A. & Casari, , 2018. *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*. illustrated ed. O'Reilly.

Zou, D.-l. et al., 2014. Study of 0~3 Hour Short-Term Forecasting Algorithm for Rainfall. *Journal of Tropical Meteorology*, pp.249-60.

Appendices

9.1 Appendix 1

This paper presented in Future of Information and Communication Conference (FICC), in Singapore, in the 5th -6th April 2018, and appeared in Volume 886 of the Advances in Intelligent Systems and Computing series in Springer.

New Modification Version of Principal Component Analysis with Kinetic Correlation Matrix using Kinetic Energy

Sara K Al-Ruzaiqi
Computer Science Department
Loughborough University
Muscat, Oman
s.k.s.al-ruseiqi@lboro.ac.uk

Dr Christian W Dawson
Computer Science Department
Loughborough University
Loughborough, United Kingdom
c.w.dawson1@lboro.ac.uk

***Abstract*—Principle Component Analysis (PCA) is a direct, non-parametric method for extracting pertinent information from confusing data sets. It presents a roadmap for how to reduce a complex data set to a lower dimension to disclose the hidden, simplified structures that often underlie it. However, most PCA methods are not able to realize the desired benefits when they handle real world, and nonlinear data. In this work, a modified version of PCA with kinetic correlation matrix using kinetic energy is proposed. The features of this modified PCA have been assessed on different data sets of air passenger numbers. The results show that the modified version of PCA is more effective in data compression, classes reparability and classification accuracy than using traditional PCA.**

Keywords— Principle Component Analysis (PCA); kinetic correlation matrix; kinetic energy; algorithm; prediction

I. INTRODUCTION

Principal Component Analysis (PCA) is a classical multivariate data analysis technique, which is popular within linear feature extraction as well as the data compression of numerous uses [1]. PCA has been applied in numerous areas of information processing to prepare data due to its distinctive result of error reducing and correlating properties. PCA compresses most of the information in the first data space into a fewer features. It attempts to look for a subspace in which the variance is maximized [2]. The PCA subspace is spanned through the eigenvectors corresponding to the top eigenvalues of the sample covariance matrix. PCA also can be applied in data preparation for both supervised and un-supervised learning and recognition processes [3].

However, most PCA strategies might not result in desirable classification benefits when they cope with real world, nonlinear data. As nonlinear PCA and its variants can effectively capture the nonlinear relations, they might provide more effective power to cope with the real world, nonlinear data [4]. It is recognized that PCA is designed to find the most indicative vectors, i.e., the eigenvectors corresponding to the best eigenvalues of the sample covariance matrix.

As data with good spectral resolution results in unwanted data for classification, a proven way to conquer this issue is reducing the dimensionality of data space. Different feature extractions, as well as selection strategies, recommend using PCA, as it is highly effective and involves a mathematical process which transforms a selection of (possibly) correlated variables into a (smaller) selection of uncorrelated variables known as principal components [5].

The sheer size of data in the modern age is not only a challenge for computer hardware but also a bottleneck for the performance of many machine learning algorithms. Identifying patterns in data is one of the main goals of a PCA analysis, and it only works by reducing the data dimensionality only when there is strong correlation between the variables. In brief, PCA is a data analysis technique which finds directions of maximum variance in high-dimensional data and projects them onto a smaller dimensional subspace while retaining most of the information.

In this work, a modified version of PCA with kinetic correlation matrix using kinetic energy is proposed, where the transformed matrix is computed from samples of selected features only. The efficiency of the modified and traditional versions of PCA is compared by applying them to an air passenger dataset. The results show that the modified version of PCA is more effective in data compression, class reparability and classification accuracy than using traditional PCA.

II. MODIFICATION OF PCA

Since the original definition of PCA via approximating multivariate distributions by planes and lines [2], scientists have defined PCA from various elements [2], [3]. Among the definitions, utilizing the covariance matrix of the training sets to explain PCA is extremely well known in pattern recognition as well as the machine learning community.

Current implementations of PCA use a correlation matrix, the matrix obtained by pairwise correlation using Pearson correlation coefficient. However, in some cases the Pearson correlation coefficient could be limited in the sense that it fails to capture other properties of the data outside of the linear relation. For example, the correlation of two random vectors: $x=\{-4,-3,-2,-1,0\}$, $y=x^2 \Rightarrow Cor(x,y)=0$, using Pearson coefficient. However, this result is not capturing the non-linear relation between the two random vectors given by the functional transformation $(x)^2 \rightarrow (y)$ which means the

978-1-5386-2056-4/18/$31.00©2018 IEEE

correlation is not zero (just non-linear). In order to improve this, the following two features have been introduced into traditional PCA in this work.

- *Information energy*: first introduced in 1966, is an analogy of the kinetic energy from physics to probability, which can be defined as follows:

  $x_1, x_2, \ldots, x_n$ and corresponding probabilities:

  $P=(P_1, P_2, \ldots, Pn)$

  $IE(p_1, p_2, \ldots, p_n) = \sum_{i=1}^{n} pi^2$

If the experiment has *n* outcomes, and every outcome has the same probability 1/*n*, then the information energy IE=1/*n*. If the experiment results in same outcome, then the probability for every outcome is 1 and the information energy has maximum value of IE=1.

The information energy increases when the randomness decreases. It is like reverse of Shannon entropy, for measuring bits of information to determine uncertainty. It is also an entropy, but the correct way to think about it is as 1/2∗m∗v2 of a random vector. Simple, but very powerful, the kinetic energy method works very well to improve the accuracy or improve some machine learning methods on row data especially if there are groups of categorical data, even if they are continuous they could be discretized.

- *Informational Correlation Coefficient,* also known as Onicescu's correlation coefficient, is a function of the joint probability density distribution of the two vectors x and y. Assume we have two random vectors x and y, the information correlation coefficient can be described as:

$$O(x,y) = \frac{\sum_k p^{(Pk)*p(Qk)}}{\sqrt{IE(P)*IE(Q)}} \quad (1)$$

This is only applicable for the discrete data that we have dealt with in this research.

The Pearson correlation captures only linear properties of the manifold on which our raw data lives. For instance: if we take a random vector in R x=c(-4,-3,-2,-1,0,1,2,3,4) and y=x^2, Pearson or Spearman, will yield 0 correlation when in fact it is 0.5 because of the functional transformation x->^2. In this work, a new correlation coefficient, as a performance metric, instead of cross entropy as in the case of neural networks, or, in the case of genetic algorithms, as fitness functions, has been applied in the modified PCA.

Previously, PCA was utilized to decrease large data sets, correlated by a number of correlation metrics, or used in addition to deriving new features. Consequently, Pearson correlation or the covariance matrix is used to determine eigenvalues and eigenvectors. Having a completely different correlation metric that captures kinetic properties of two random vectors against one another has also been used in creating a modified version of original algorithm with this new correlation matrix.

Hence, we implemented new correlation metrics, and the new idea was to modify the original PCA for obtaining eigenvectors and eigenvalues for dimensionality reduction using a correlation matrix with our kinetic correlation coefficient.

III. Implementing Modified PCA

In this work, a new correlation coefficient method called Octave has been introduced. The correlation is used as a method for feature selection (calculated between two features) using Kinetic Energy. The new Octave correlation makes a useful contribution as it provides a new measure of dependence between random vectors that capture non-linear relationships as well.

The modified version of PCA was assessed using a data set of air passenger numbers, from where the features of the modified PCA were derived, using kinetic correlation metrics instead of Pearson correlation coefficient based on kinetic correlation theory.

A. Implementation Setup

This function returns information coefficient IC for two random variables defined as the dot product of probabilities corresponding to each class:

```
def ic(vector1,vector2):
    a=vector1
  b=vector2
 prob1=np.unique(a,return_counts=True)[1]/a.shape[0]
    prob2=np.unique(b,return_counts=True)[1]/b.shape[0]
    p1=list(prob1)
    p2=list(prob2)
    diff=len(p1)-len(p2)
    if diff>0:
      for elem in range(diff):
        p2.append(0)
    if diff<0:
      for elem in range((diff*-1)):
        p1.append(0)
    ic=np.dot(np.array(p1),np.array(p2))
    return ic
```

And, having functions for kinetic energy of a vector and for information correlation, we can define a new function that computes kinetic correlation. This function will return correlation based on kinetic energy as illustrated below:

```
def o(vector1,vector2):
    i_c=ic(vector1,vector2)
  o=i_c/np.sqrt(kin_energy(vector1)*kin_energy(vector2))
  return o
```

The formula is updated such that the denominator contains sqrt in order to have probabilities bounded between 0 and 1.

SHAPE will return the number of items in the numpy array in the form of a tuple, then creates a matrix with the number of rows initialized with zero values.

```
rows=data.shape[1]
rows
matrix= np.zeros((rows,rows))
```

Then the correlation matrix is created with the function o() that was defined previously, as shown in Table 1. The correlation matrix obtained by the Pearson method is also listed in Table 2 for comparison.

978-1-5386-2056-4/18/$31.00©2018 IEEE

TABLE I. CORRELATION MATRIX WITH THE FUNCTION O()

	0	1	2	3	4	5
0	1.000	1.000	0.974	0.326	0.184	0.229
1	1.000	1.000	0.974	0.326	0.184	0.229
2	0.974	0.974	1.000	0.320	0.180	0.223
3	0.326	0.326	0.320	1.000	0.071	0.131
4	0.184	0.184	0.180	0.070	1.000	0.490
5	0.229	0.229	0.223	0.131	0.490	1.000

TABLE II. CORRELATION MATRIX ON BASIS OF 'PEARSON R' MODEL

	0	1	2	3	4	5
0	1.000	0.751	0.770	-0.041	-0.027	0.000
1	0.751	1.000	0.959	-0.013	-0.031	0.000
2	0.770	0.959	1.000	-0.020	-0.023	0.000
3	-0.041	-0.013	-0.020	1.000	0.216	0.000
4	-0.027	-0.031	-0.023	0.216	1.000	0.000
5	-0.000	-0.000	-0.000	0.000	0.000	1.000

B. Comparison of modified PCA with Kinetic Correlation Matrix from Kinetic Energy and PCA with Pearson R correlation

Our contribution is based on changing the correlation matrix that uses Pearson R correlation or, in some cases, the covariance, with a correlation matrix based on the Onicescu correlation coefficient. The results of testing the kinetic correlation of our data sets using the Pearson coefficient are shown in Fig. 1 and 2.

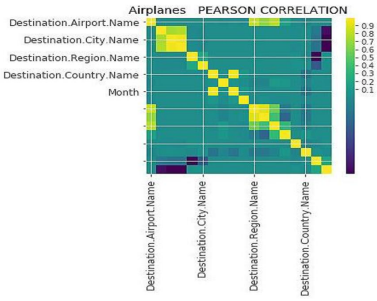

Fig. 1. Air passenger numbers data with Pearson Correlation.

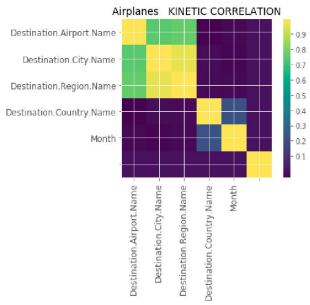

Fig. 2. A train passenger numbers data with Kinetic Correlation.

As expected the kinetic correlation has a much higher kinetic correlation matrix from kinetic energy than the Pearson one. Pearson's R is able to detect only linear relations in data. The graphs have the same list of seven columns on both x and y axis. The colouring of each particular square shows the actual correlation between the columns on the scale of 0 to 1.0. So, if the color is dark, there is low correlation and vice-versa.

C. Features Obtained from Kinetic Energy PCA Components

In this section, we implemented XGBoost [6]. This is an algorithm that has recently been dominating applied machine learning for structured or tabular data and it is designed for speed and performance. It has been applied here to a training data set of passenger numbers with a dataset of 51,983 observations with 9 variables. In order to get a better estimate of model performance, we used a variant of the famous 1-fold cross validation. We split dataset into a training set (75% of the data) and a test set (25% of the data) randomly for 1 different time and measure accuracy, false positive rate and false negative rate.

The XGBoost model was run within Python machine learning modules and the calculated mean values (Fig. 3) are very much nearer to the actual values of one.xgb.train, which is an advanced interface for training an XGBoost model.

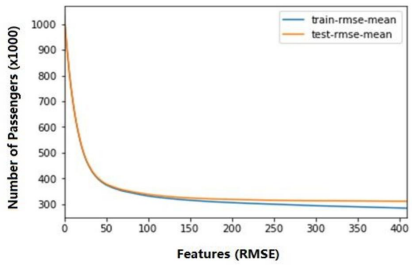

Fig. 3. The mean values of features obtained from Kinetic Energy PCA Components.

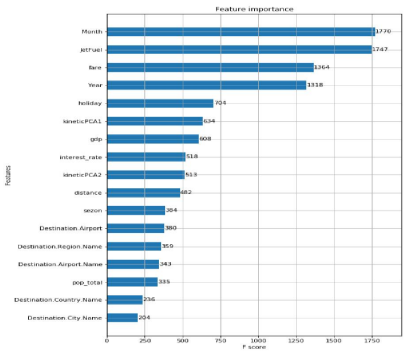

Fig. 4. Principal component analysis features (KineticPCA1 and KineticPCA2).

978-1-5386-2056-4/18/$31.00©2018 IEEE

TABLE III. PREDICTION MODEL USING KINETICPCA1 AND KINETIC PCA2

	passangersPred5
0	515
1	636
2	621
3	624
4	607

Fig. 4 shows the features for predicting the number of passengers from most important to least important. Here it shows that JetFuel, Month, and fare are the most predictive values. As is noted in the plot below the features obtained from the modified PCA, called kineticPCA1 and kineticPCA2, are captured with reasonable influence after running the XGBOOST model and inspecting feature importance. The number of passenger predictions using the Prediction model is given in Table 3.

D. Features Obtained from Deep Learning Hidden Layers

In this step, we created a different engineered dataset in order to have diversity in multiple datasets. We have chosen at this step to add non-linear features that were extracted from an R implementation of a Deep Learning model.

We trained a deep learning neural network with 100 neurons in first hidden layer, 63 neurons in second hidden layer and 30 neurons in the third hidden layer and 15 neurons in last hidden layer. The number of features extracted from the deep learning model was the same number of neurons in each hidden layer. For a better selection of only important non-linear features, we computed correlations of each feature that was corresponding to each neuron in the hidden layer with our target variable. During the computing the correlations, we kept only one feature from each neuron, where is the maximum correlation compared with other features in the same hidden layer, and obtained final four non-linear features. An XGBOOST model was then run to see the behavior of that particular model on the newly created data set.

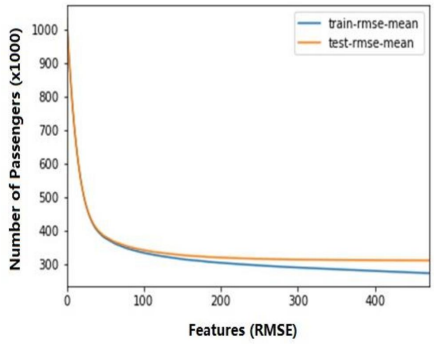

Fig. 5. The mean values of features obtained from deep learning hidden layers.

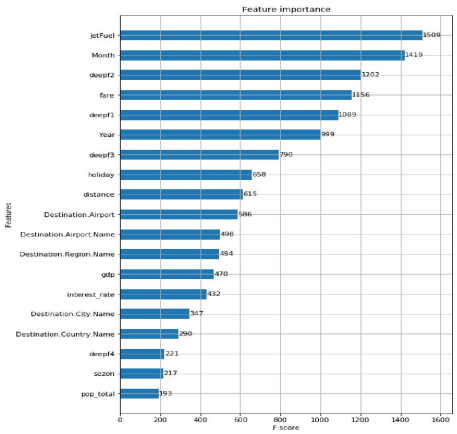

Fig. 6. Prediction model using deep learning hidden layers.

TABLE IV. PREDICTION MODEL USING DEEP LEARNING HIDDEN LAYERS

	passangersPred4
0	582
1	545
2	546
3	552
4	627

The plot in Fig. 5 shows that the mean values calculated are much nearer to one but differed more on the last set of inputs. Fig. 6 shows the features that are important for the number of passengers predicted from most important to least important. Here it shows JetFuel, Month, and fares have the highest predictive values. As observed from the plot the nonlinear features deepf1, deepf2, deep3, and deepf4 (obtained by the method described above) are very influential and are the ones with the highest influential impact captured by XGBOOST feature importance. The number of passengers predicted by using the Deep Learning Hidden Layers is given in Table 4.

E. Features Obtained from Genetic Algorithm

This feature was extracted from a genetic algorithm called *symbolic transformer*, which is an estimator that begins by building a population of naive random formulas to represent a relationship [7]. The formulas are represented as tree-like structures with mathematical functions being recursively applied to variables and constants. Each successive generation of programs is then evolved from the one that came before it by selecting the fittest individuals from the population to undergo genetic operations such as crossover, mutation or reproduction. The results are presented in Fig. 7.

978-1-5386-2056-4/18/$31.00©2018 IEEE

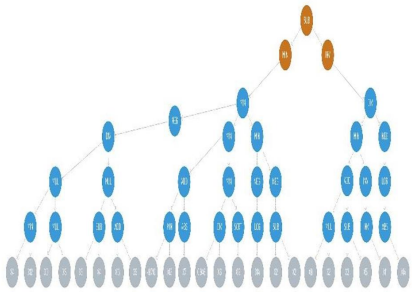

Fig. 7. Tree-like structures of the Genetic Algorithm.

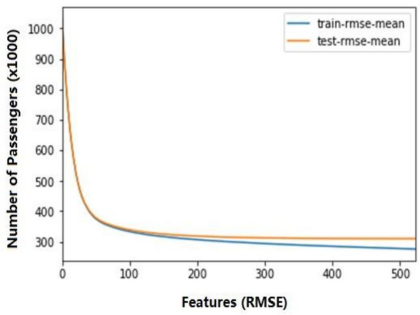

Fig. 8. The mean values of features obtained from Genetic Algorithm.

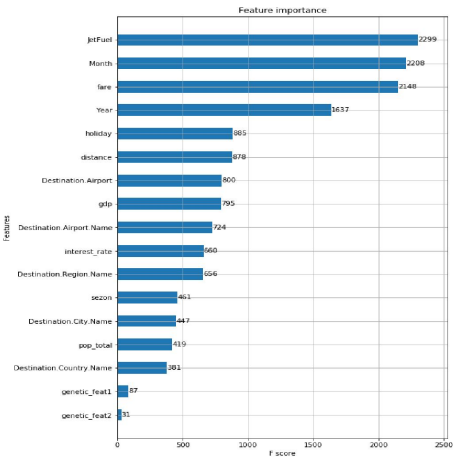

Fig. 9. Features importance obtained from Genetic Algorithm.

TABLE V. PREDICTION MODEL USING GENETIC ALGORITHM

	passangersPred2
0	529
1	611
2	600
3	609
4	607

In the genetic program, it is easy to observe different kinds of operations that the genetic algorithm produced. Two new features obtained from genetic transformer after running an XGBOOST model have been added into this algorithm. Fig. 8 shows that the mean values calculated are very near one, but differed more on last set of inputs. Fig. 9 shows that the features that are important for the number of passengers predicted from most important to least important. Here it shows that JetFuel, Month, and fare are the most predictive values. From the plot below the genetic features called genetic_feat1 and genetic_feat2, where captured with very small influence in contrast with our expectation when conducting the experiment. The number of passengers predicted by using the Deep Learning Hidden Layers is given in Table 5.

IV. CONCLUSION

In this work, a new modified version of PCA with kinetic correlation matrix using kinetic energy is presented. The features of this modified PCA have been assessed with different sets of air passenger data and compared to traditional PCA. The results of the modified version of PCA show that the kinetic correlation is much higher than that of the Pearson one, which makes lot of sense since Pearson's R is able to detect only linear relations in data. It turned out that the modified version of PCA is more effective in data compression, classes reparability and classification accuracy than those form traditional PCA.

Based on these results, the modified PCA can be applied to make clustering in hyper-dimensional space using kinetic correlation as a distance (increase performance) to make it run in real time in a future work. When coping with clustering, such as clustering algorithm, clustering K-means or in hierarchical clustering, it requires a for-loop at every point to get the nearest point from row vector. For n rows of data complexity will be of the order n^n, which is impossible to finish using this method. In two-dimensional space, there is a trick to fast implementation using divide and conquer, which has complexity n or log n. However, these problems can be solved by using modified PCA with properly added features.

In this work, only limited features of the modified PCA method were studied with one set of data. To fully understand and investigate the features of modified PCA, large subsets of data with more features should be considered.

ACKNOWLEDGMENT

I would like to address my special acknowledgements to all those people who provide me with data for my experiments. My warm appreciation is due to the Public Authority for Civil Aviation, Directorate General of Meteorology, and Ministry of Tourism in Oman.

978-1-5386-2056-4/18/$31.00©2018 IEEE

REFERENCES

[1] Bengio, Y. (2013). Representation learning: a review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(8), 1798-1828

[2] Timmerman, M. E. (2003). Principal component analysis (2nd Ed.). I. T. Jolliffe. Journal of the American Statistical Association, 98, 1082-108

[3] F. M. Palechor et al. (2017), "Cardiovascular Disease Analysis Using Supervised and Unsupervised Data Mining Techniques", Journal of Software, vol. 12, no.2, pp. 81-90

[4] Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. Journal of Machine Learning Research, 9 (2579-2605), 85.

[5] Coates, A., & Ng, A. Y. (2012). Learning feature representations with k-means *Neural Networks:*

[6] Chen T. Q. and Guestrin C. (2016). XGBoost: A Scalable Tree Boosting System, KDD'16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Pages 785-794

[7] Lowe, D. G. (1999). *Object recognition from local scale-invariant features.* Paper presented at the The Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999.

978-1-5386-2056-4/18/$31.00©2018 IEEE

9.2 Appendix 2

This paper presented in Computing Conference in London in the 16th -17th July 2019, and appeared in Volume 997 of the Advances in Intelligent Systems and Computing series in Springer.

Optimizing Deep Learning Model for Neural Network Topology

Sara K. Al-Ruzaiqi[1] and Christian W. Dawson[2]

[1] Computer science Department, Loughborough University, Muscat, Oman
s.k.s.al-ruseiqi@lboro.ac.uk
[2] Computer Science Department, Loughborough University, Loughborough, UK

Abstract. In this work a method of tuning deep learning models using h2o is proposed, where the trained network model is built from samples of selected features from the dataset, in order to ensure diversity of the samples and to improve training. A successful application of deep learning requires setting its parameters in order to get better accuracy. The number of hidden layers and the number of neurons are the key parameters in each layer of a deep machine-learning network, which have great control on the performance of the algorithm. Hyper-parameter, grid search and random hyper-parameter approaches aid in setting these important parameters. In this paper, a new ensemble strategy is suggested that shows potential to optimize parameter settings and hence save more computational resources throughout the tuning process of the models. The data are collected from several airline datasets to build a deep prediction model to forecast airline passenger numbers. The preliminary experiments show that fine-tuning provides an efficient approach for tuning the ultimate number of hidden layers and the number of neurons in each layer when compared with the grid search method.

Keywords: Deep Learning, H2O, and Optimizing.

1 Introduction

Deep learning has been applied in many contexts – for example, health (pattern recognition), education (machine translation) and vision interpretation. The optimization of such models involves calibration with large data sets, which are then used to make future predictions [1]. We frequently make use of different optimization techniques in deep learning to improve performance. For instance, grid search is one such technique that is used to enhance the calibration of deep neural network models. The challenge still remains to train deep learning neural networks with large data sets in an efficient manner [2].

Researchers have applied deep neural networks in a number of studies with promising outcomes [3]. Learning methods have been developed that can teach deep network variants such as conviction networks presented by Fukushima [4]. This development reestablished enthusiasm for deep neural networks. Neural networks with numerous layers will frequently encounter an issue of some transfer functions

approaching zero. Hochreiter was the first researcher to summarize this disappearing gradient issue within his Ph.D. thesis [5]. Prior to deep learning, the majority of neural networks used a simple quadratic error performance measure on the output layer [6]. De Boer et al. [6] introduced the cross-entropy error feature, and it frequently achieves much better outcomes than the quadratic function previously used as it deals with the vanishing gradient issue by permitting errors to change weights even if a neuron's gradient saturates (their results are close to zero).

Additionally, it offers a more irregular way of error representation as opposed to the quadratic error performed for classification neural networks. Thus, the analysis provided in this paper is going to utilize the cross-entropy error measure for classification as well as the more commonly used root mean square error (RMSE) for regression. Neural networks usually begin with random weights [9]. These random weights are often sampled within a range, such as (-1,1). This range initialization can sometimes create a set of weights that prove difficult for the back propagation training (for example, being caught in a local minima). [8] unveiled the rectified linear unit (ReLU) transfer function to deal with this issue. The ReLU transfer function typically achieves better training benefits for deep neural networks as opposed to the sigmoidal transfer functions, which are often used. Based on recent studies [1,7], the type of transfer function to use for deep neural networks is identified for each layer type.

The hidden layers of deep neural networks make use of the ReLU transfer function. For their output layer, the majority of deep neural networks employ a linear transfer function for regression, along with a softmax transfer function for classification. No transfer function is required for the input layer. Moreover, over-fitting is a regular issue for neural networks [10]. A neural network is believed to be over-fit when it has been trained to a stage such that the network starts to master the outliers in the data set. This neural network is learning how to commit to memory, not generalize [8]. Since there are many variables compared with the total number of training examples, feature selection, and regularization techniques are used to combat the over-fitting problem. Data are partitioned into training, validation, and testing sets to ensure robust statistics.

2 Optimizing Deep Prediction Model

The objective of this study is to train and build a deep prediction model. The focus is on feed-forward neural networks. We start by exploring the data, simplifying them and after that, applying the model and investigating the results. In these steps the R language is used along with the h2o R package. Monthly data (1989 to 2016) are sourced from the Oman Management Airport Company (OMAC) which contains 51,983 observations with 9 variables (2 are categorical explanatory variables; and the variable to predict passenger data (Pax) are also categorical with 13 levels).

The focus of this work is not on the data set as such, but on the analysis that should be done before making predictions and the features of h2o in R; and why it is important in deep learning. The primary reason behind selecting the h2o implementation over various other neural networks libraries is that it has plain hyper parameter tuning which makes it rapid. In addition, h2o possesses Java as a backend, which makes it easier to run over several cores of the processor.

In order to train models with the h2o engine, the datasets should be linked to the h2o cluster first, then run the Deep Learning model on the dataset, and the Deep Learning model will be tasked to perform (multi-class) classification. The data are

imported into R in the conventional way. Fig. 1 presents the monthly distribution of the passenger data set.

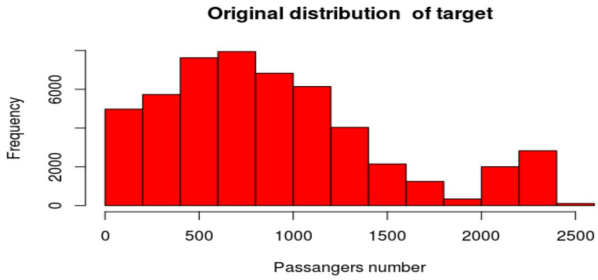

Fig. 1. The Original Distribution of Target

Because of the size of the data set it was possible to carry out multiple runs to observe the variation in prediction performance and to investigate the impact of model regularization by tuning the 'Dropout' parameter in the *h2o.deeplearning(...)* function. This was undertaken using the following steps:

- Set-up and connect to a local h2o cluster from R.
- Train a deep neural networks model.
- Use the model for predictions.
- Ensemble models.
- Consider the memory usage.

A local cluster with 4GB memory allowance was built to ensure that the memory utilization was sufficient over the period of the model training process. Only one thread was created to initiate deep learning in this project.

3 Methods

3.1 Training a Deep Neural Network Model

The dataset was split randomly into training and test sets (75% and 25% respectively). In addition, after this we double-checked using a dropout of 25% data from the original training file. A dropout is a 1-fold cross validation taken to its extreme: the testing set contained 9840 observations while the training set was composed of all the remaining observations. Note that in a dropout 1 = number of observations in the dataset. It appears as the data lends itself to the Neural Network. Later analyse will determine if this is, in fact, the situation or perhaps if the default parameters for the Neural Network have been a lucky-hit.

3.2 Testing the Model by Using the Model for Prediction

The experiments were started by adding L1 and L2 regularization (and boosting the number of training rounds/epochs from 1 to 100). This seems to reduce the errors down to 292.5256 (errors before 308.0929), which makes sense as we have made our model less prone to catching 'noise' and to generalize better. We experiment with a **wide range of hidden-layers and number of neurons** to construct a list of potentials so that the chosen parameter was somewhere in the middle. The model results in less error, which was even lower than on the validation. RMSE before dropout is 308.09 passengers; after 292.525 – so dropout has led to improvement.

The graph in Figure 2 on Y-axis shows the actual total values of passengers along with the predicted values from our predictor on x-axis. Some of the points near 2000 to 2500 on the y-axis show there is a miss match on actual and prediction count. The value of it can be confirmed as 0- 500 from y-axis at last points. Hence, there was a need to train the ANN further. Regarding the number of epochs, the neural network should iterate the best number from the possible states of the greedy search 100 epochs.

A dropout of 25% of data from original train file was performed in order to have a 1-fold cross validation with 75% data for train and 25% data for evaluation to have some initial intuition about how the tuned deep learning model is perform by plotting predicted values on the unseen test set against ground truth values. This result is presented as the plot bellow in Figure 2 of the predicted distribution followed by plot Figure 3 of the predicted values.

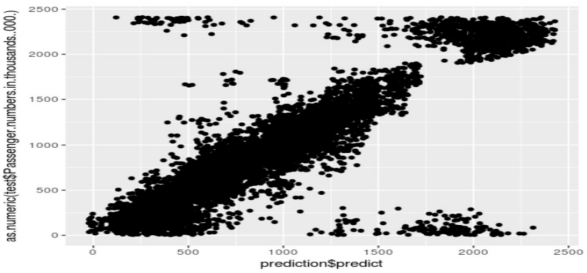

Fig. 2. Predicted Values on the Unseen Test Set Against Ground Truth-Values

Figure 2, shows a scatter plot showing patterns in passenger flight list. It looks like there is enough variation in the data to make this a good predictor.

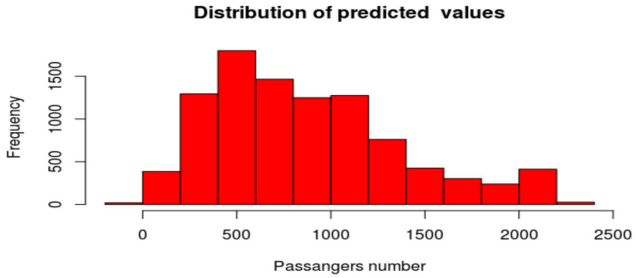

Fig. 3. The Distribution of Predicted Values

3.3 Fine Tuning using Hyper-Parameter, Grid Search and Random Hyper-Parameter Search

It is possible to achieve less than 10% test set error rate in few seconds using some tuning. Hyper-parameter tuning is very important for Deep Learning, as it could influence model accuracy. The very first 10,000 rows of the training dataset will be trained. Hyper parameter optimisation can usually be accomplished better with random parameter search than with grid search.

Within the following model, Maxout, Tanh, and Rectifier, are utilized as activation operators. The Tanh function is a rescaled and shifted logistic function - with symmetry roughly zero it enables the training algorithm to converge more quickly. Rectifier has two benefits: it is quick and does not suffer through the vanishing gradient condition.

3.4 Extra Grid-Search to Optimise Parameters

When a sizable feed-forward neural network is trained on a tiny training set, it usually performs badly on the held out test data. This "over-fitting" is significantly decreased by randomly omitting one half of the characteristic detectors on each training situation.

This stops complex co-adaptations in which a function detector is just useful in the context of other certain element detectors. Rather, each neuron learns to identify a feature, which is usually of great help for creating the appropriate answer provided the combinatorial large variety of inner contexts in which it must operate.

Random grid searches a favorite option to locate the greatest parameters. Another grid search on the model was run since it scored perfectly, in the beginning, to find out if it will be better and overtake the NN.

3.5 Fine Tuning the Hyper-Parameters

Since the dataset is not considerably different in context from the initial dataset, which the pre-trained model is trained on, it must go for fine-tuning. As a result,

preparing to apply a bit of topology, meaning dealing with neural network with128 neurons in very first hidden layer, 63 neurons in next hidden layer as well as 32 neurons on third hidden level. Epochs (passes with the data) per iteration on N nodes are 100. Additional epochs were utilized for higher predictive accuracy, but just when in the position to afford the computational cost.

The training error value was based on the parameter ***training_frame=train***, which specifies the selection of randomly sampled training points to be utilized for scoring; the default utilizes 10,000 points. The validation error is based on the parameter ***validation_frame=valid_frame***, which regulates the identical value on the validation set and it is set by default to become the whole validation set. Setting either of the parameters to zero instantly uses the whole corresponding dataset for scoring.

Here we have fed the data into our deep learning module ANN with predictors, target and train data as well as into the deep learning module with L1 and L2 for regularization. Deep Learning is based on a multi-layer feed forward artificial neural network that was trained with stochastic gradient descent using back-propagation. Below is the fine-tuning implementation, after performing hyper-parameter optimization using grid search the best neural network topology found is the following:

```
tuned_model <- H2O.deeplearning(
  x=predictors,
  y=target,
  training_frame=train,
  hidden=c(128,63,32),
  epochs=100,
  nfolds=5,
  fold_assignment="Modulo" ,
  l1=5.6e-05,
  l2=7.4e-05,
  input_dropout_ratio=0.05
)
```

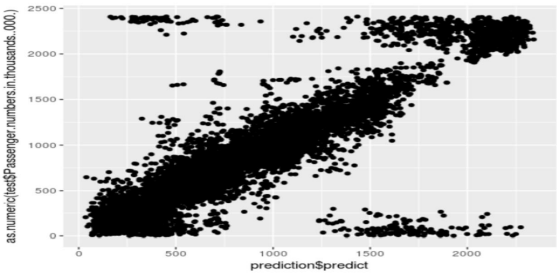

Fig. 4. A Newly Predicted Values on the Unseen Test Set Against Ground Truth-Values.

Changing input dropout ratio rate improved the training dataset to 0.8560758. Changing it definitely changes the results. Once again start predicting with the help of data, which already trained.

The plot in Figure 4 is of newly generated predicted data with the help of gplot. The graph above on Y-axis shows the actual total values of passengers along with the predicted values from our predictor on x-axis. Some of the points near 2000 to 2500 on y-axis show there is a miss match on actual and prediction count. It can be confirmed from 0- 500 from y-axis at last points. These are fewer as compared with the last plot, as the hidden layers increased in the ANN.

4 Improving Deep Neural Network Model Performance

A new ensemble strategy is suggested in this paper to enhance the overall performance of neural network classifiers. The suggested solution brings together a number of neural network classifiers, exactly where every classifier utilizes a unique distance function as well as likely another group of characteristics (feature engineered). These features engineered wish consuming a mix of grid the search engines (at the amount of this ensemble). Demonstrating that instead of optimizing the feature establish on their own for every distance metric, it's better to co adapt them, such that every feature set is enhanced within the context of this ensemble as entire.

Ensemble learning of neural network is a learning paradigm in which ensembles of a number of neural networks indicate increased generalization abilities, which outshine individuals of individual networks. For deep learning of multi-layer neural networks, ensemble learning remains relevant. Additionally, qualities of deep neural networks are able to offer possible chances to enhance the overall performance of conventional neural network ensembles. Within this paper, looking at this method having the best performing model and then trained the model. Furthermore, getting rid of the intense values has somewhat beneficial impact on the model effect on each training as well as validation datasets. A particular need to remember that even some enhancement within the metrics would result in a big leap of the model.

4.1 Split the Data According to Initial Train Test Split (1-Fold Cross Validation)

Half of the outlier labels were removed from the training set and considered unlabeled data (which treated as a contaminated normal class in the setup). At this point, no regularization has been applied for any method so that the results are more easily comparable. For feature transformation, trying to make use of a variety of established unsupervised outlier detection techniques: choosing arbitrary subsets of options, as well as k-NN outlier (compute sum of distances to k nearest neighbors). These functions are based upon a rough search for unsupervised algorithms, which function moderately effectively on the training set. It is a simple starting point along with options ought to be investigated.

4.2 Getting the Best Three Features

In the random forest approach, a large number of decision trees were created. The package "randomForest" has the function ***randomForest()*** which is used to create and analyze random forests, used for prediction. The below Figure 5 shows that when use different features with ensemble functions, errors get reduced. So, using same technique to improve the ANN.

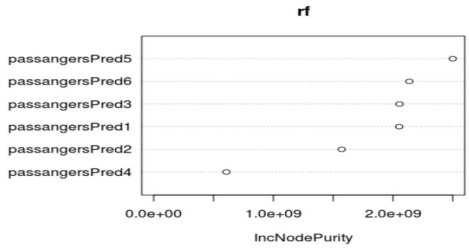

Fig. 5. Variable Importance on Generated Features.

The outlier threshold was calculated by multiplying standard deviation (SD) passenger of tempTr with 1.5 and added to Passenger.numbers.in.thousands..000.of tempTr, after that take mean of that entire data for prediction of tr. If the prediction entry of tree not in between 0 and 1 that entry will be eliminated.

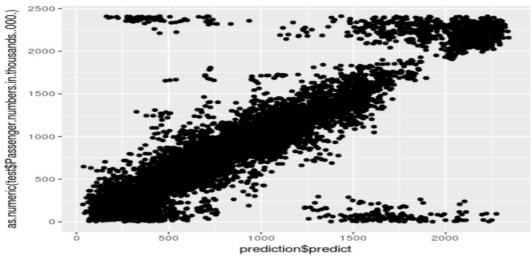

Fig. 6. A Last Predicted Values on the Unseen Test Set Against Ground Truth-Values.

Above plot in Figure 6, is newly generated predicted data with the help of ***gplot***. The graph above on Y-axis shows the actual total values of passengers along with the predicted values from our predictor on x-axis. Some of the points near 2000 to 2500 on y-axis show there is a miss match on actual and prediction count. Similar can be confirmed from 0- 500 from y-axis at last points. These are very less as compare to our last plot, as we have increased the hidden layers along with MSE i.e. mean square error methods in our ANN.

4.3 Features Obtained from Deep Learning Hidden Layers

The number of features extracted from the deep learning model was the same number of neurons in each hidden layer. For a better selection of only important non-linear features, the correlations were computed of each feature that was corresponding to each neuron in the hidden layer with the target variable (passenger number in thousands per month). Having computed the correlations, we then kept only one

3. Fukushima, K., 1980. Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, Volume 36, pp. 193-202.

4. Hochreiter, S., 1991. *Untersuchungen zu dynamischen neuronalen Netzen,* s.l.: Diploma, Technische Universität München.

5. Bishop, C. M., 1995. *Neural networks for pattern recognition.* 1st ed. s.l.:Oxford University Press.

6. De Boer, P.-T., Kroese, D. P., Mannor, S. & Rubinstein, R. Y., 2005. A tutorial on the cross-entropy method. *Annals of operations research*, Volume 134, pp. 19-67.

7. Glorot, X. & Bengio, Y., 2011. *Deep sparse rectifier neural networks.* s.l., International Conference on Artificial Intelligence and Statistics.

8. Bastien, F. et al., 2012. Theano: new features and speed improvements. *arXiv preprint arXiv:1211.5590.*

9. Russell, S. & Norvig, P., 1995. *Artificial intelligence: a modern approach.* 3rd ed. s.l.:Artificial Intelligence.

10. Masters, T., 1993. *Practical neural network recipes in C++.* 1st ed. s.l.:Morgan Kaufmann.

9.3 Appendix 3

9.3.1.1 Detecting the Outlier for the Monthly Airline Passenger Numbers

Outlined below is a line by line code explanation. It will be seen that first, "Pandas" and "NumPy," which are the two most significant libraries, were imported to facilitate numerical analysis and data manipulation. The train data set was then read and the first five rows printed as shown below.

Let Month and passenger numbers be represented by variables X and Y respectively. A function is then defined to create X and Y distributions with the use of ***NumPy*** co-variance function.

```
x=train[['Month']]
y=train[['Passenger.numbers.in.thousands..000.']]
def MahalanobisDist(x, y):
    covariance_xy = np.cov(x,y, rowvar=0)
    inv_covariance_xy = np.linalg.inv(covariance_xy)
    xy_mean = np.mean(x),np.mean(y)
    x_diff = np.array([x_i - xy_mean[0] for x_i in x])
    y_diff = np.array([y_i - xy_mean[1] for y_i in y])
    diff_xy = np.transpose([x_diff, y_diff])
    md = []
    for i in range(len(diff_xy)):

md.append(np.sqrt(np.dot(np.dot(np.transpose(diff_xy[i])
,

inv_covariance_xy),diff_xy[i])))
    return md
```

The dimensions of X and Y are then changed to find the outliers as shown below;

```
x=np.array(x)
x=x.reshape(x.shape[0],)
y=np.array(y)
y=y.reshape(y.shape[0],)
md = MahalanobisDist(x,y)
```

The previous function was then utilized to come up with the outlier for monthly passenger figures as shown henceforth;

```
def FindOutliers(x, y, p):
    MD = MahalanobisDist(x, y)
```

```
    nx, ny, outliers = [], [], []
    threshold = -2*np.log(1-p)
    for i in range(len(MD)):
        if MD[i]*MD[i] < threshold:
            nx.append(x[i])
            ny.append(y[i])
            outliers.append(i)  #  position  of  removed
pair
    return          (np.array(nx),          np.array(ny),
np.array(outliers))
```

The actual passenger figures data with outliers 1 is then printed as;

```
print(train[['Passenger.numbers.in.thousands..000.']]==O
utliers[1][1])
```

The actual data on number of months with outliers 2 was printed and the outlier calculations produced as follows;

```
print(train[['Month']]==Outliers[0][1])
train[train[['Passenger.numbers.in.thousands..000.']]==O
utliers[1][1] &train[['Month']]==Outliers[0][1]]
np.mean(y)+2*np.std(y)
```

Table 9-1 First Five Rows of Target Feature (Passenger Number)

	Year	Destination.Airport	Destination.Airport.Name	Destination.City.Name	Destination.Region.Name	Destination.Country.Name	Month	Passe
0	1998	76	71	69	83	10	5	439
1	1998	163	153	148	46	15	5	825
2	1998	36	159	153	33	3	5	861
3	1998	55	116	112	33	3	5	862
4	1998	207	208	204	33	3	5	865

Table 9-2 Actual Data of Number of Passengers with Outliers 1

	Passenger.numbers.in.thousands..000.
0	False
1	False
2	False
3	False
4	False
5	False
6	False
7	False
8	False
9	False
10	False
11	False
12	False
13	False
14	False
15	False
16	False
17	False
18	False
19	False
20	False
21	False
22	False
23	False
24	False
25	False
26	False
27	False
28	False

Table 9-3 Actual Data of Number of Passengers with Outliers 2

	Month
0	False
1	False
2	False
3	False
4	False
5	False
6	False
7	False
8	False
9	False
10	False
11	False
12	False
13	False
14	False
15	False
16	False
17	False
18	False
19	False
20	False
21	False
22	False
23	False
24	False
25	False
26	False
27	False
28	False
29	False