A DEEP-LEARNING INTEGRATED LEE-CARTER MODEL: APPLICATION TO THE MAURITIAN POPULATION

WOODUN DHANDEVI

A thesis submitted in fulfilment of the
requirements for the award of the degree of
MASTER OF SCIENCE IN DATA SCIENCE AND BUSINESS ANALYTICS

ASIA PACIFIC UNIVERSITY OF TECHNOLOGY & INNOVATION (APU)

DECEMBER 2019

ACKNOWLEDGEMENT

First and foremost, I would like to thank the Almighty God for giving me the knowledge and the direction to complete this research. Likewise, I would like to convey my deepest appreciation to my respected lecturer, Dr Manoj Jayabalan for his constant monitoring and encouragement throughout the whole process in preparing this assignment, right from its initial development, his unwavering support and exemplary guidance in providing the constructive comments, direction, expertise and kindness, all of which contributed immensely towards the efficacy in accomplishing this study within the period given. Furthermore, I would like to express my gratitude to my second marker Miss Seetha Letchumy A/P M. Belaidan and all my classmates who have been great sources of encouragement and motivation during the course of this research.

ABSTRACT

Since the beginning of 20th century, Mauritius has been experiencing continuous growth in its life expectancy, thereby having impending effects on insurance companies and pension providers. Intrinsically, the Lee-Carter model was introduced to model and forecast mortality rate in the aim of resolving issues related to ageing population. The Lee-Carter model comprises of three parameters namely the average log-mortality rate ($\mathrm{a_x}$), the response of mortality ($\mathrm{b_x}$) and the mortality index (k_t). To estimate the parameters $\mathrm{a_x}$ and $\mathrm{b_x}$, the Singular Value Decomposition (SVD) is generally used, however, the Non-Negative Matrix Factorization (NMF) caters only for positive values and can work well with missing values as compared to SVD. Additionally, to project the mortality index parameter k_t, the Auto Regressive Integrated Moving Average (ARIMA) is traditionally utilized but it was found that this model could not accurately identify the non-linear mortality patterns. In addition to this, the ARIMA model lacked the ability to outline the future mortality shape as well. Therefore, the main goal of this research was to use an alternative approach based on a deep learning technique in the aim of enhancing the predictive ability of the Lee-Carter model using the Mauritian mortality data. More precisely, the Recurrent Neural Network with Long Short-Term Memory (RNN-LSTM) architecture was proposed to project the mortality index parameter $\mathrm{k_t}$. But, since existing research fitted the ARIMA model to mortality data from 1984 to 2016 only, this study applied death data obtained from the Statistics Department of Mauritius, from 1984 to 2018 to fit the ARIMA model and compared the results with the RNN-LSTM one. Furthermore, a comparison was conducted between SVD and NMF to determine the best technique to estimate $\mathrm{a_x}$ and $\mathrm{b_x}$. Findings revealed that the SVD was the most suitable method to be used owing to its low error value of Mean Square Error (MSE) as compared to that of NMF. Furthermore, the RNN-LSTM model was found to outperform the ARIMA model. Using the estimated parameters, the death rates of the Mauritian population were projected for the next 30 years.

TABLE OF CONTENTS

Contents

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Research Background

Over the last decades, the rapid advancement in technology and industrial revolution have resulted in remarkable alterations in demographic tendencies. Since its independence in 1968, Mauritius has experienced drastic improvement in the health and living conditions of its population, whereby the life expectancy at birth as at 2018 is 71.65 years for males and 78.40 years for females. However, back then in 1962, the Statistics Department of Mauritius recorded that a baby girl was anticipated to survive till 62 years while a new born boy was expected to live to 59 years only. Additionally, Martial (2018) highlighted that the proportion of the population aged 65 years and above has increased from 10.5% in 2017 to 11.0% in 2018, thus indicating a rise in the number of pensioners and retirees which can ultimately prove to be a significant challenge to pension systems and have extensive consequences on the stability of public finances and future economic growth.

This growing life expectancy is owing to the drop-in mortality caused by infectious and chronic ailments at older ages (Hainaut, 2018). Therefore, it is highly significant that the Mauritian government anticipates that the ageing population will prove to be a looming and economic burden in the next decades and measures are to be taken to alleviate the effects of longevity risk. As such, mortality rate is considered to play an imperative role in the government pension policy formulation and pension modelling. In view of this, various models in the preceding centuries have been recommended such as Lee-Carter model. It has been comprehensively accepted by the actuarial community for its simplicity due to a smaller number of parameters and robustness in order to project death rates.

However, in recent years, many researchers have introduced machine learning techniques such as Neural Networks to predict mortality rate, which consequently proved to generate more accurate results than the extrapolative original Lee-Carter model (Richman and Wuthrich, 2018). Deprez, Shevchenko and Wuthrich (2017) accentuated that with the aid of these techniques, the latter could scrutinize the adequacy of the projected death rates.

Furthermore, machine learning algorithms proved to improve the goodness of fit of the Lee-Carter model as well as identify hidden patterns in the mortality data. In addition to this, Nigri et al. (2019) emphasized that applying deep-learning techniques to the Lee-Carter model not only preserved its parsimony and robustness but also generated much more precise and reliable mortality forecasts for actuaries, insurance and pension plans consultants, when it comes to handling future cash flows which are associated to longevity dynamics.

The main purpose of this research is to improve the predictive ability of the Lee-Carter model and forecast the mortality rate of the Mauritian population. Additionally, this study will enable insurers as well as pension providers to refine their pricing techniques as well as enhance their product development strategies, based on the death rate forecasts.

1.2 Problem Statement

Mauritius is prominent for its rapid ageing population whereby in 2011, the whole population comprised of 12.7% of elderly (aged $\geq$ 60 years) while in 2017, this value increased to 16.1% (SDM, 2018). Additionally, the old age dependency ratio which is defined as the ratio of people aged above 65 years to those between age 15 and 64 years, decreased drastically by 48.2% from 78.7% in 1972 to 30.5% in 2018. These rates are irrefutably upsetting and highlight the need for the seniors to be gradually self-supportive and financially strong in the impending years. Consequently, there will be lower revenues from social security contributions as well as higher expenditures on pensions and healthcare services on behalf of the government. The Statistics Department of Mauritius in a report in 2017 underlined that the government expenditure on social security and welfare for the year 2016 represented 29% of the entire government expenditure and 7.4% of the Gross Domestic Products (GDP) at market prices. Therefore, it is crucial to project longevity risk in order for the government to cater for the needs of the impending ageing population.

In view of the above, many studies are being conducted in the aim of finding solutions to these ageing population challenges such as modelling and forecasting death rate. As underscored earlier, Lee-Carter model has been widely accepted by the actuarial community and has proved to be the most reliable method in projecting mortality rate of populations together with its extensions. However, it does have certain shortcomings whereby according to Booth and Tickle (2008), the Lee-Carter model undertakes that the proportion of the rates of the mortality variation at different ages stays constant over time, yet evidence of substantial age-time interaction has been professed. Coupled to this, Girosi and King (2006) underlined that the Lee-Carter model projects mortality rates that are less smooth across-age and become progressively jagged over time which is non-intuitive and unreliable.

Additionally, Nigri et al (2019) highlighted that the Autoregressive Integrated Moving Average (ARIMA) method which is commonly used to predict the mortality index[1], was found not being able to capture the long-term trend inside the sequential mortality data and suggested the Recurrent Neural Network (RNN) with the Long Short-Term memory (LSTM) instead. Moreover, Woodun, Ho and Raja (2019) forecasted the Mauritian population mortality rate using the ARIMA model to project the mortality index parameter. But, the latter applied only mortality data from 1984 to 2016 and fitted the ARIMA model. In addition to this, the Non-Negative Matrix Factorization (NMF) method which is correspondingly used to decompose non-negative matrices can also be used as a substitute for Singular Value Decomposition (SVD) technique used traditionally to estimate the parameters of the Lee-Carter model. The NMF matrix decomposition method caters only for positive values and can work well with missing values as compared to SVD. Furthermore, the deep-learning techniques are still not prevalent in Mauritius and to the best of my knowledge, no researcher has attempted to forecast the mortality rate of the population based on machine learning or deep-learning algorithms.

1.3 Research Questions

This research attempts at modelling and forecasting the mortality rate of the Mauritian population using deep-learning techniques and the Lee-Carter model, while fulfilling the following research questions.

1) What is the most suitable method to forecast the mortality index?
2) What is the best possible technique to estimate the parameters of the Lee-Carter model?
3) How accurate are the deep-learning integrated Lee-Carter model and the forecasted death rates?

[1] *Overall level of mortality in a specific year*

1.4 Research Aim and Objectives

The aim of this research is to propose an approach to enhance the projecting capability of the Lee-Carter model and fit the model to the Mauritian mortality data from 1984 to 2018[2]. The goal of this study is to forecast the mortality rate of Mauritius and provide solutions to insurance companies and pension providers to alleviate the effects of ageing population.

The objectives of the research are outlined as follows.

1) To investigate state-of-the-art approaches to the Lee-Carter model used in modelling and forecasting mortality rate.
2) To determine the optimum technique to estimate the parameters of the Lee-Carter model.
3) To propose a deep-learning model to forecast the mortality index parameter.
4) To evaluate the performance of the Lee-Carter model.

1.5 Scope of Study

This research focuses mainly on the population of Mauritius which is around 1,265,637 as at 2018. As mentioned earlier, the data to be used, comprises of the death rates for both males and females of ages 0 to 85+ for the years 1984 to 2018. The time period of the historical data is 34 years. This study targets at applying the deep-learning integrated Lee-Carter model to the provided mortality data in the aim of predicting the future death rates. The model will be tested and explored whether it fits the data or not. The forecasting period usually varies and depends on the period of the data available. For example, if the data comprises of a period of 100 years, then based on the 100-year historical trends, 90 to 100 years forecasting period can be used. As such, since a 34-year period is available, a 30-year period is to be considered during projection. Additionally, the study emphasizes mainly on using the ARIMA and LSTM models to forecast the overall level of mortality parameter of the Lee-Carter.

However, the implementation of this proposed framework is indeed challenging, owing to the fact that to the best of my knowledge, no such study has been carried out yet on the Mauritian death data and it is highly uncertain that this innovative model will fit the data or

[2] *Obtained from Statistics Department of Mauritius (Source: http://statsmauritius.govmu.org/)*

not. Additionally, the data provided by the Statistics Department of Mauritius, has only a period of 34 years which specifies that the inconsistency in the mortality data must be meticulously substantiated in the aim of obtaining effective death forecasts for the population of Mauritius. Furthermore, the death data found might not be reliable enough and further tests will have to be conducted to fit the data to the deep-learning integrated Lee-Carter model.

1.6 Significance of Study

The projection of future mortality rates plays an imperative role in life insurance companies which aim at accomplishing suitable pricing of their life products. In addition to this, governments and pension providers rely significantly on upright death rate forecasts for effective administration of their pension obligations and to curtail their old-age pension expenditure. Therefore, from an economic perspective, this study will prove to be important to the insurance and pension industries in enhancing their pricing techniques and managing their pension plans correspondingly in the aim of reducing supplementary costs and maximize profits while establishing their future liabilities. This will ultimately boost the economic condition of the country since insurance companies and the pension providers contribute enormously to the financial situation of a specific nation. In other words, this research will empower insurance corporations to advance their business operations.

Although there are numerous methods and techniques to predict mortality rates, this research focuses more on integrating deep-learning to the Lee-Carter model, thereby a new model to be applied to the Mauritian population. From an academic point of view, this study will not only contribute to the knowledge of actuarial students but other researchers as well in familiarizing them with the emerging machine learning and artificial intelligence methods. To the best of my knowledge, no research has been conducted yet in projecting the mortality rates of Mauritius using deep-learning techniques along with the Non-Negative Matrix Factorization (NMF) method has never been applied.

1.7 Research Design

This section provides an outline of the research approach of how this study will be carried out with appropriate explanations in order to cope with the research problem. In other words, the research design focuses mainly on the methods to be used to collect and pre-process the data in the aim of obtaining accurate results as illustrated in Figure 1.1.

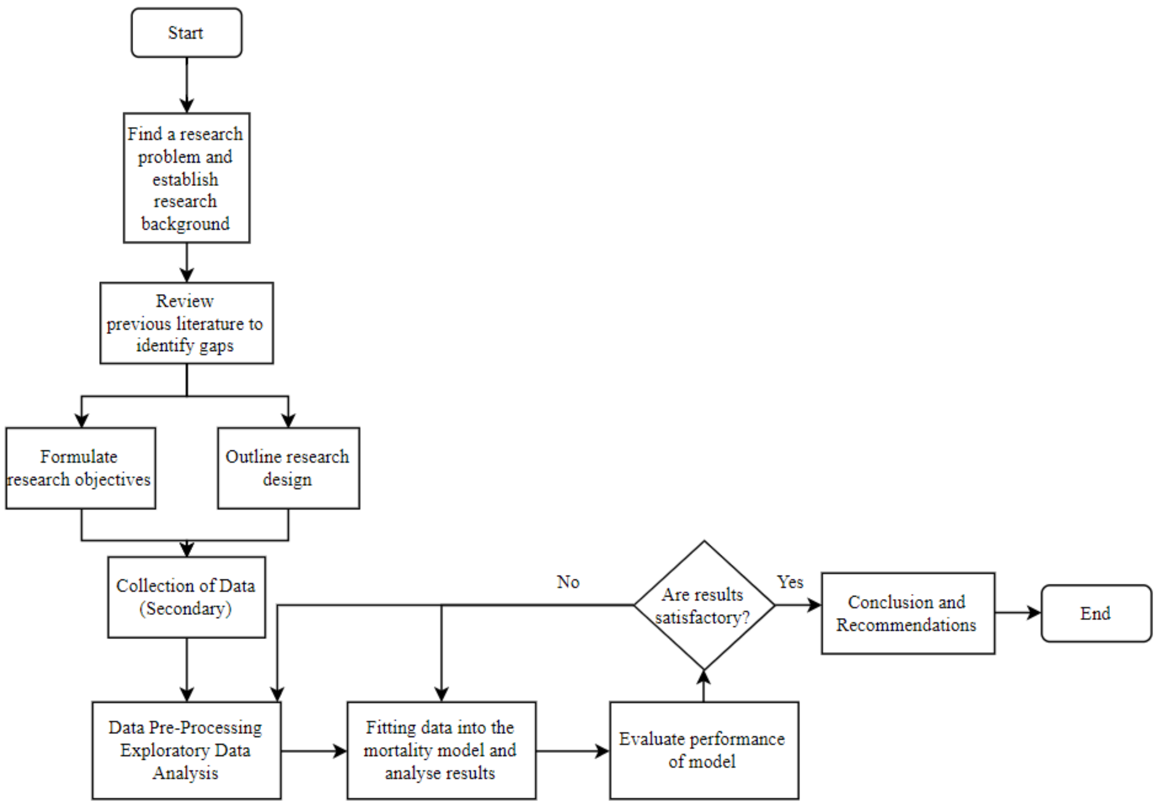

Figure 1.1: Research Design

In view of the Figure 1.1 above, it can be noticed that the research design starts with identifying a research problem and establishing the research background. In addition to this, previous works of researchers are reviewed in order to identify their findings and research gaps which will then be tackled in the study. In other words, the literature review survey is performed to further support the problems identified and to formulate the objectives. Moreover, once the secondary data is collected i.e. from online databases, data pre-processing and data exploration techniques are identified and applied to the dataset.

Furthermore, once the model is built, the cleaned data is fitted in the aim of achieving the objectives. Measure of errors or model selection criteria tools are used in order to evaluate the performance of the model. For instance, if the measures of errors are very high, it implies that the researcher will have to perform the data pre-processing and data exploration stage so that better accurate results are generated. Finally, once the model is tested and formulated objectives are accomplished, the findings are discussed as well as documented and recommendations are suggested in the final report.

1.8 Structure of Study

In the aim of predicting the mortality rate of the Mauritian population using the deep-learning integrated Lee-Carter model, this study is structured as follows: Chapter 1 delivers the preliminary idea of the project by giving an overview on the background of study. The problem identified related to this study is discoursed and the objectives to be accomplished are briefly stated. Furthermore, an insight of the implication and limitations of the research. Finally, an outline of the research methodology is provided for further in-depth understanding of this investigation.

Chapter 2 comprises of an ephemeral review of the related literature, with the aim of providing a theoretical analysis on the application of deep-learning techniques to the Lee-Carter model by assessing past researches that investigated on this mortality forecasting model on other populations.

Chapter 3 focuses mainly on the research methodology whereby the data pre-processing methods to be applied to the Mauritian mortality data will be outlined and explained. The Lee-Carter model will be explored together with the Recurrent Neural Network with the Long Short-Term Memory architecture. Coupled to this, the R programme software will be used.

Chapter 4 will progress with the real application of the model to the mortality data of Mauritius obtained from the Statistics Department of Mauritius, and exploratory data analysis will be performed on the training set of the original dataset. Moreover, based on the methods and software to be used as outlined in Chapter 3, the objectives of this research are to be achieved. The training dataset will be used to learn and train the model. In addition to this, the model selection criterions will be analysed, and mortality test dataset will be fitted to the trained model in order to obtain the death rate forecasts.

Chapter 5 will concentrate highly on the results after evaluating the trained model as well as forecasting the mortality rate for the Mauritian population while considering the estimated parameters of the deep-learning integrated Lee-Carter model. A comparison between the suggested techniques in estimating the parameters will be conducted as well.

Chapter 6 will highlight the limitations faced in conducting this study along with recommendations to improve the validity of the findings. This chapter concludes the study by summarizing the procedures undertaken and emphasizing the relevance of its findings.

Finally, it also focuses on the future effort that can be taken into consideration, which could not be accomplished owing to time-constraint during the course of the study.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter focuses mainly on the key concepts of deep-learning techniques along with its theoretical and practical application to the Lee-Carter model in projecting future mortality rates. Moreover, an overview of the demographic patterns of Mauritius is outlined. Past researchers' works will be analysed and reviewed in the aim of elaborating their findings and identifying their challenges as well as their limitations. The gaps in the previous works will be acknowledged in order to further conduct this study.

2.2 Demographic Trends in Mauritius

A study conducted by the Statistics Department of Mauritius in 2018 highlighted that the population growth rate of the country has been experiencing a decreasing trend since the beginning of the 21st century as exhibited in Figure 2.1, owing to the rapid industrial development, economic progress along with a rise in medical advancements.

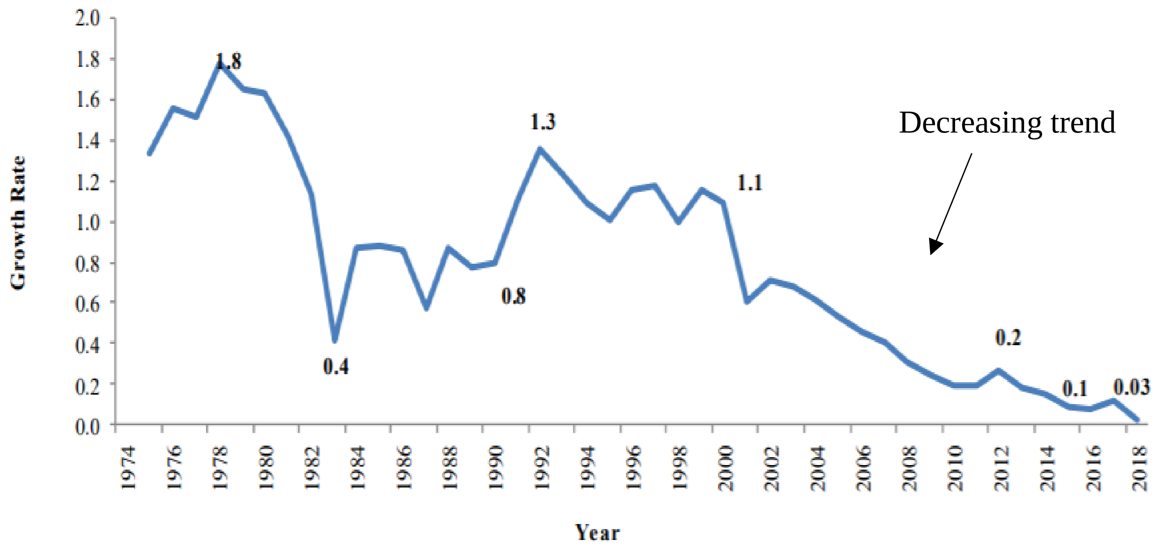

Figure 2.1: Population growth rate from 1974 to 2018 of the Mauritian Population (Source: SDM, 2018)

In addition to the above, the sex ratio of the Mauritian population was perceived to have had an opposite trend whereby the number of females was greater than the number of males as from the nineties as compared to before the eighties, as illustrated in Figure 2.2, due to the fact that females tend to live longer than males. As observed from the graph, there were 97.9 males per 100 females in the year 2018 while 103.8 males per 100 females in 1974. Besides, as at 2018, the female population was higher than the male population by 13,115 (SDM, 2018).

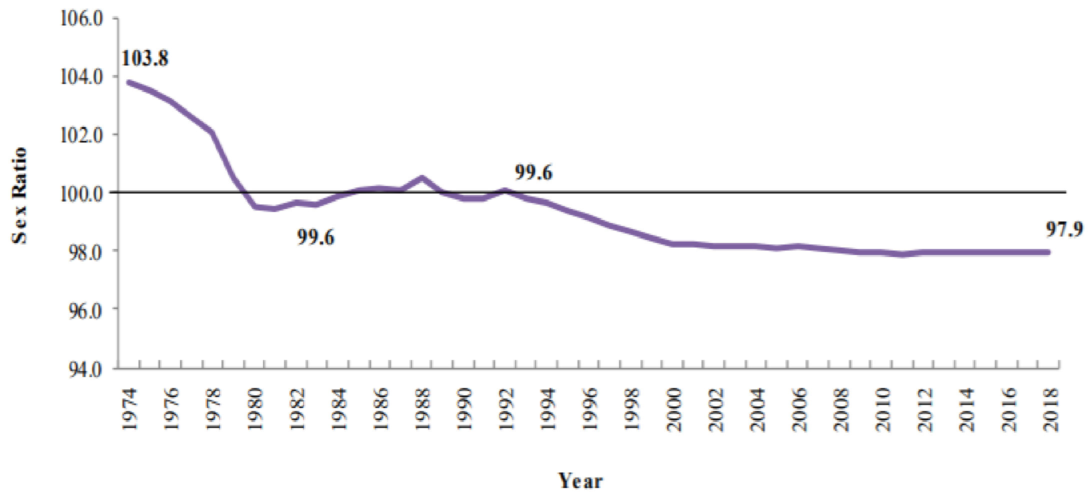

Figure 2.2: Sex ratio of the Mauritian population from 1974 to 2018 (Source: SDM, 2018)

Moreover, the incidence of mortality among females is lower than among males at all ages. From the Figure 2.3, it can be observed that mortality varies with age whereby there is a drastic drop in the mortality curve between ages 5 to 9 and then surges with increasing age for both males and females. The probabilities of dying for each individual vary between 0 and 1 and are presented in the Figure 2.3 per 1000 population. The high death rate between ages 0 to 4 is due to the fact that infants are highly prone to diseases and infections, thus eventually leading to death if not treated effectively.

Additionally, the sudden drop in death rate from ages 5 to 9 years is mainly because of building up of the immune systems of the infants, thus they are more resistible to illnesses. However, the probability of death among adolescents i.e. as from age 15 years, starts to increase owing to causes of death such as juvenile delinquency, cigarette and alcohol intake. From the age 30 years onwards, the mortality rate increases exponentially with growing age.

It can also be perceived that the probability of death for males is higher than that of females, most specifically between ages 20 to 30 years.

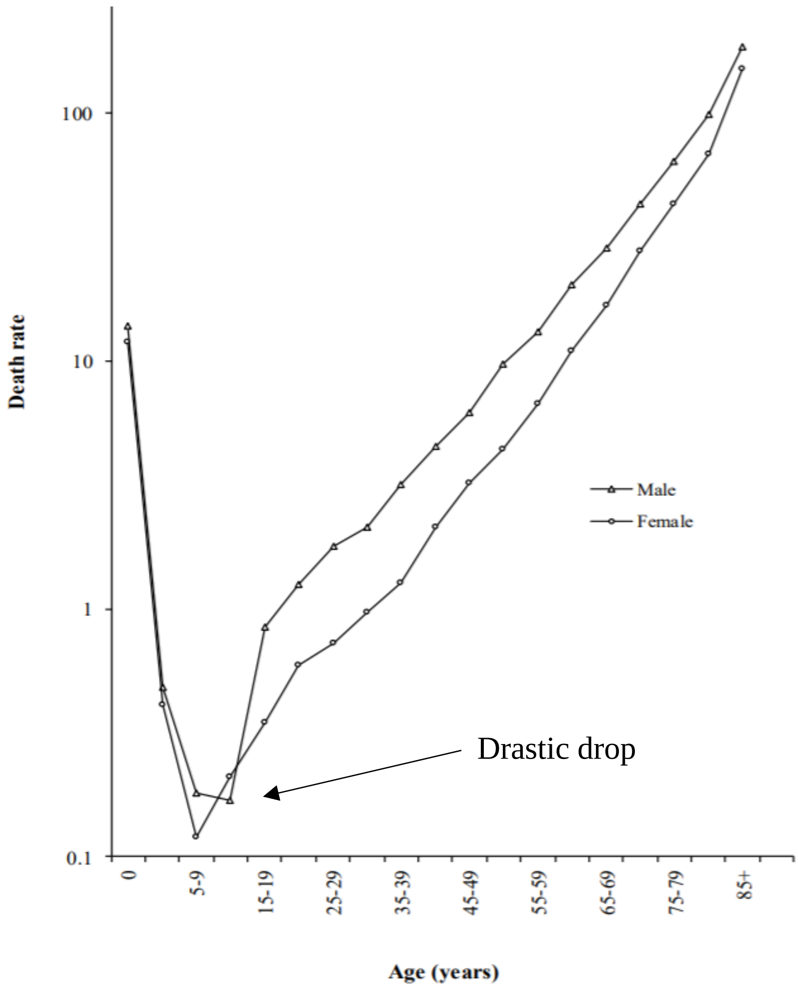

Figure 2.3: Age-Specific Death Rates for the years 2015 to 2017 (Source: SDM, 2018)

2.3 Theories and Models on Mortality

Mortality modelling dates back to when Gompertz introduced the law of mortality in 1825, and since then several models have been proposed in the aim of forecasting mortality rate. For instance, Booth and Tickle (2008) highlighted that most of these models are usually extrapolative and parsimonious, in other words, extrapolating historical mortality trends into the future and having less parameters respectively. However, the latter also added that actuaries and demographers also rely on two other approaches when it comes to forecasting the death rate of a particular population namely expectation and explanation. For example, the expectation method refers to considering expert opinion but is not reliable since the forecasted values are commonly subjective and bias (Lee and Miller, 2001). Besides, the explanatory approach implies taking into consideration the effect of morbidity and other death risk factors while modelling and predicting the death rate.

As such, Girosi and King (2006) accentuated that this method can only be used when forecasting for short-term period only. This is because that the latter encountered issues like non-linear interactions in the time-series data as well as large number of parameters to be estimated. These eventually affected the robustness of the model and reduced the accuracy of the projected death rates. Furthermore, in view of the extrapolation technique, time series methods such as Autoregressive Integrated Moving Average (ARIMA) and two-factor (age-period) models are normally used since they are stochastic and enable the computation of probabilistic prediction interval for the forecast value. For instance, the Lee-Carter model is a two-factor one since it comprises of a single time parameter and is the most popular mortality forecasting model applied by the actuarial community.

2.3.1 Gompertz Law of Mortality

Parametric models of human mortality are generally based on the Gompertz law of mortality which states that the force of mortality represented by μ_x increases with age x at a constant exponential rate, whereby the two parameters a and b are greater than 0, vary with the level of mortality along with the rate of increase in mortality with age as explained by the Equation 1, where a denotes the level of mortality at the initial age i.e. at x=0 and b is the rate of mortality increase over age.

$$\boldsymbol{\mu_x = ae^{bx}} \qquad \textbf{[1]}$$

However, the Gompertz model was modified by Makeham in 1867 who added a constant c to the above equation which can cater for the causes of death unrelated to age as shown in Equation 2.

$$\boldsymbol{\mu_x = ae^{bx} + c} \qquad \textbf{[2]}$$

Moreover, this Gompertz-Makeham model was further enhanced and it was found by Perks in 1932 that the force of mortality μ_x can be fitted in a logistic model for small values of α in order to project mortality rates. This model in Equation 3 was eventually modified by

Lee-Carter since computation of death rates was difficult since there were many unknown parameters to be estimated to be elaborated in Section 2.3.2.

$$\boldsymbol{\mu_x = \frac{ae^{bx}}{1+ \alpha e^{bx}} + c} \qquad \textbf{[3]}$$

2.3.2 Lee-Carter Model

To comply with the Gompertz law of mortality as discussed in Section 2.3.1, Lee and Carter established the stochastic method to forecast death rate with the least number of parameters in 1992. The mortality model has extensively been cited by many researchers when modelling and forecasting long-run age-specific mortality rates for any particular nation. The main advantage of this model is that it is simple, and parameters are effortlessly estimated and interpreted. Additionally, Booth and Tickle (2008) emphasized that the Lee-Carter model encompasses insignificant subjective judgement and generates stochastic forecasts with probabilistic prediction intervals. In the aim of improving the Gompertz law of mortality, Lee and Carter (1992) added a parameter k_t which ensures that the fitted deaths are equal to the total number of observed deaths in each year as established by the Equation 4.

$$\boldsymbol{\ln(m_{x,t}) = a_x + b_x k_t + \varepsilon_{x,t}} \qquad \textbf{[4]}$$

where:

$\boldsymbol{m_{x,t}}$ is the central mortality rate at age x in year t.

$\boldsymbol{a_x}$ is the average (over time) log-mortality at age x.

$\boldsymbol{b_x}$ measures the response at age x to change in the overall level of mortality over time.

$\boldsymbol{k_t}$ represents the overall level of mortality in year t.

$\boldsymbol{\varepsilon_{x,t}}$ is the residual.

In regard with the Lee-Carter model in Equation 4, the parameters $\boldsymbol{a_x}$, $\boldsymbol{b_x}$ and $\boldsymbol{k_t}$ are to be estimated and in general the Singular Value Decomposition (SVD) method is used, while abiding by the following constraints: $\boldsymbol{\sum_t k_t = 0}$ and $\boldsymbol{\sum_x b_x^2 = 1}$. When it comes to forecasting the death rates, the mortality index parameter i.e. $\boldsymbol{k_t}$ needs to be projected for future time points and most commonly the Box-Jenkins time series approach is applied. In other words, the parameter $\boldsymbol{k_t}$ is modelled and predicted using the random walk with drift such as ARIMA (0,1,0) model, assuming that the parameters $\boldsymbol{a_x}$ and $\boldsymbol{b_x}$ are invariant with time (Lee and Carter, 1992). However, enhancements to the Lee-Carter estimation basis have been proposed by many other researchers. For instance, techniques such as Weighted Least Squares (WLS) along with Maximum Likelihood Estimation (MLE) have been implemented by Brouhns et al. (2002), Czado et al. (2005) as well as Koissi and Shapiro (2006) when finding the parameters of the Lee-Carter, while assuming that the distribution of deaths follows a Poisson distribution.

2.4 Extension of the Lee-Carter Model

The increase in life expectancy is a significant contributing factor to ageing population. Thus, it is extremely prominent that demographers as well as actuaries model and forecast mortality rate. As a result, Nigri et al. (2019) emphasized that although the original Lee-Carter model and its extensions have produced accurate mortality forecasts for different populations, implementing machine learning techniques further enhanced the predictive ability of the mortality model. This section focuses mainly on the empirical evidence regarding the effectiveness of integrating machine learning and deep-learning techniques to the Lee-Carter model.

For instance, Richman and Wuthrich (2018) extended the Lee-Carter model to several populations using neural networks. The latter first built up various versions of the mortality model and applied to all countries' mortality data from the Human Mortality Database (HMD) for the years 1950 onwards. Moreover, they tried to implement the methods such as the Common Age Effect (CAE) and Augmented Common Factor (ACF) extensions of the Lee-Carter model (Li and Lee, 2005; Danesi, Haberman and Millossovich, 2015; Chen and Millossovich, 2018). According to Kleinow (2015), the Common Age Effect (CAE) refers to keeping the component of the Lee-Carter model that defines the change in mortality with time constant but having different period indices fitted for each population. Besides, Augmented Common Factor (ACF) follows the model as shown in Equation 5 where b_x^i and k_t^i are the rate of change of the log mortality with time and the time index respectively, both for population i. The population specific average mortality a_x^i is computed and subtracted from the matrix of mortality rates. Additionally, the original Lee-Carter, Augmented Common Factor (ACF) and Common Age Effect (CAE) models were fitted to the data and the forecasted mortality rates were analysed before comparing them to those of the deep neural network approach.

$$\log\ (u_{x,t}) = a_x^i + b_x k_t + b_x^i k_t^i \qquad [5]$$

Consequently, it was deduced that the deep neural network extension of the Lee-Carter model outperformed all the other models since it attained the smallest residuals with the model fitting the females' mortality data better than that of males, yet researchers still have to verify whether the same results are obtained for the Mauritian population. The deep neural network method efficaciously learnt the relationships of the inputs to the model in determining mortality and predicted the death rates for all the countries with high accuracy (Richman and Wuthrich, 2018). Nevertheless, it was recommended that model selection process such as Akaike Information Criterion (AIC) as well as the Bayesian Information Criterion (BIC) to be scrutinised more deeply in the aim of determining the optimal neural network architecture. Furthermore, the uncertainty in the evaluations of the neural network model was not addressed.

On the other hand, to deal with the limitations in the work of Richman and Wuthrich (2018) such as uncertainty in the model time processes, a neural network analyser to be implemented to the Lee-Carter model in order to recognise latent time processes for better mortality forecasts was suggested (Hainaut, 2018). In other words, the neural network analyser will help to spot non-linearities in the log-forces of mortality and it consists of two feed-forward neural networks. The proposed model was a non-linear and semi-parametric one and allowed the researchers to track the evolution of mortality curves while considering hidden determinants that affect mortality. In addition to this, the Markov Chain Monte Carlo (MCMC) method was introduced to the model to be fitted to the death data from 1946 to 2014 of France, United Kingdom and the United States in the aim of predicting the future mortality rates of multiple populations (Antonio et al., 2015). The results showed that the neural analyser projected accurate log-mortality rates over a short period of time and proved to enhance the predictive power of the original Lee-Carter model. For instance, the average relative errors for neural networks and their standard deviations were lower than those obtained with the actual mortality model i.e. the original Lee-Carter model.

To improve the estimation of the log-forces of mortality as discussed above, Deprez, Shevchenko and Wuthrich (2017) suggested some machine learning techniques to be applied to mortality models such as Lee-Carter and Renshaw-Haberman in the aim of analysing their downsides. As such, the regression tree boosting machine was used to examine the accuracy of the projected mortality rates of the Swiss population. In addition to this, the latter also enhanced the models by considering feature components of individuals such as age or birth cohort. Moreover, this non-parametric regression also enabled the researchers to investigate cause-of-death mortality, i.e. given the death of an individual with a specific feature, the conditional probability of its cause is studied. It was found that the main weakness of Lee-Carter and Renshaw-Haberman was that the mortality models never capture the effects of events that cause demise such as epidemics or any cohort effects, instead they only take into account the age of the individual. Therefore, the framework of the Poisson regression tree boosting was applied to the Swiss mortality data in order to estimate the probability of death due to causes of death namely health issues, accidents and others. This technique was proven to be an effective method in the aim of detecting patterns in the probabilities of death as well as to estimate cause-of-death mortality rates from real data, yet feature components such as income, marital status or education of individuals were not catered.

Levantesi and Pizzorusso (2019) examined the capability of machine learning to enhance the precision of some standard stochastic mortality models, both in estimating and forecasting death rates of the Italian population using mortality data from 1915 to 2014. The latter followed the approach of Deprez, Shevchenko and Wuthrich (2017). For instance, the latter applied the tree-based machine learning techniques such as decision tree, random forest and gradient boosting to calibrate a parameter i.e. the machine learning estimator which was then employed to mortality rates fitted by mortality models namely Lee-Carter, Renshaw-Haberman as well as the Plat model. It was perceived that the application of these machine learning methods, grounded on components such as age, sex, calendar year, and birth cohort, led to a better fit of the past data, with respect to the estimates generated by the Lee-Carter, Renshaw-Haberman and Plat models. As a consequence, it was found that the machine learning algorithms applied not only supported the mortality models but also enhanced their predictive ability. Coupled to this, results obtained showed that machine learning can actually be used to improve both fitting the mentioned mortality models and projecting death rates, while considering the framework of the Lee-Carter technique.

However, when estimating the parameters of the Lee-Carter model, the traditional Autoregressive Integrated Moving Average (ARIMA) approach lacks consistency since it does not capture the non-linear patterns in projected mortality rates (Nigri et al., 2019). Therefore, the latter proposed an alternative technique to ARIMA based on deep-learning method. For instance, in the aim of determining the patterns in the mortality index series over time more accurately, the Recurrent Neural Network (RNN) along with the architecture of Long Short-Term Memory (LSTM) were adopted.

Nigri et al. (2019) also accentuated that the application of LSTM not only allowed them to attain mortality forecasts more comprehensible with the observed mortality changing aspects but also in situations of non-linear mortality trends. In other words, the LSTM network is designed in such a way so that long sequences of data are elaborated, establishing a memory able to preserve the substantial correlations between data. In this way, the Long Short-Term Memory (LSTM) approach enables researchers to project future mortality over time while taking into account the crucial effects of historical mortality pattern and eventually effectively replicating it into the forecasted trend.

Nevertheless, Nigri et al. (2019) applied the same parameter estimation method i.e. the Singular Value Decomposition (SVD) when fitting the mortality model using the death data of six countries namely Australia, Denmark, Italy, Spain, Japan and the USA from the Human Mortality Database. When the ARIMA and LSTM processes were compared based on the measures of errors such as Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), it was perceived that LSTM method outperformed the ARIMA approach, with minimal values of MAE and RMSE for LSTM. In addition to this, Nigri et al. (2019) emphasized that their model i.e. the Recurrent Neural Network together with the Long Short-Term Memory architecture, generated more accurate predictions of mortality rates with negligible discrepancy between the real value of the time series and the forecasted one.

However, further research is required in order to test the robustness of the Long Short-Term Memory (LSTM) model when it comes to modelling and forecasting future death rate. In the aim of addressing the limitations faced in the works of Hainaut (2018) as well as Levantesi and Pizzorusso (2019) when using the Lee-Carter model, different techniques have been implemented and practised, and so far with the emergence of machine learning and artificial intelligence, the need and usefulness of the deep-learning integration of the Lee-Carter model as proposed in the paper of Nigri et al. (2019) was acknowledged as imperative.

To evaluate the performance of mortality models, the Bayesian Information Criterion (BIC) and Akaike Information Criterion (AIC) measures are the most commonly used model selection criteria, specifically when using the Lee-Carter model (Lee and Miller, 2001). For instance, lower the values of AIC and BIC, better the performance of the model. For example, Yasungnoen and Sattayatham, (2016) forecasted the mortality rates of Thailand with the aid of the Lee-Carter method. When estimating the parameters of the model mainly the mortality index k_t, the latter determined the most appropriate Autoregressive Integrated Moving Average (ARIMA) models using the values of AIC and BIC. However, Duolao and Pengjun (2005) investigated that measures of errors such as the Mean Square Error (MSE) and Mean Average Percentage Error (MAPE) are also highly significant to consider when determining the best technique between Singular Value Decomposition (SVD), Weighted Least Square (WLS) and Maximum Likelihood Estimation (MLE) in order to estimate the parameters of the Lee-Carter model such as a_x and b_x. For instance, Nigri et al. (2019) applied the Mean Square Error (MSE) to determine the optimal number of neurons in the neural network architecture when improving the accuracy of the Lee-Carter model. In addition to this, Castellani et al. (2018) applied the information criteria namely AIC and BIC in order to evaluate the deep-learning integrated Lee-Carter model when projecting the mortality rates for the populations of Japan, United Kingdom as well as Italy for the years 1994 to 2013. The findings showed that the deep-learning integrated Lee-Carter model not only proved to have better and accurate death rates as compared to the original Lee-Carter, but also is one of the most stable mortality models so far, which can be applied to any population mortality data (Deprez et al., 2017).

Hordri, Yuhaniz and Shamsuddin (2016) underscored that deep learning emphasises mainly on applying a set of algorithms to datasets in the aim of modelling high-level abstractions present in the data. In addition to this, it is also referred to as hierarchical learning whereby layers are arranged in such a way in order to classify the data as the most useful to least important. Vargas, Mosavi and Ruiz (2017) also added that deep learning consists of multiple hidden layers of artificial neural networks and that the recurrent neural network is one of the most practised deep learning concepts. Richman (2018) highlighted that deep learning forms part of the machine learning approach to Artificial Intelligence (AI) whereby systems are trained to distinguish patterns within data to attain insights.

Although actuaries are still new to deep learning research, there are successful existing works where the latter tried to apply this method in their traditional models. For instance, Kuo (2019) applied the deep learning approach to loss reserving, in other words, modelling and forecasting the future payments due to claims for property and casualty insurers. However, Goodfellow, Bengio and Courville (2016) accentuated that these machine learning algorithms such as deep learning technique can only substitute the solve actuarial problems to some extent, but when it comes to the insurance domain, the prior knowledge of actuaries and discussions with underwriters along with claim handlers are highly significant when computing reserves, since the reserving models are extensively dependent on actuarial assumptions. Similarly, further research is recommended for modelling and forecasting mortality rate using deep learning techniques implemented to models such as extensions of the Lee-Carter model (Nigri et al., 2019).

When it comes to the Lee-Carter model, the Singular Value Decomposition (SVD) method is used to determine the parameters of the mortality model. However, there are other techniques such as the Non-Negative Matrix Factorization (NMF) which is referred to as an unsupervised learning method that decomposes high-dimensional non-negative data into two non-negative matrices, which has not been explored yet in the field of the mortality rate forecasting, thus being one of the research goals of this study. According to Cheung et al. (2017), the NMF was originally formulated as an algorithm with rules that decrease the squared error iteratively between the data matrix and which is reconstructed by the decomposed matrices. Additionally, this algorithm was evaluated using the information criterion namely the AIC in the aim of determining a model order which is more functionally interpretable. The Non-Negative Matrix Factorization (NMF) method will be explored in depth in the next chapter.

2.5 Discussion

There have been many research contributions related to mortality modelling and the applications of machine learning and deep learning. The work of Deprez, Shevchenko and Wuthrich (2017) demonstrated that machine learning algorithms are beneficial to evaluate the goodness of fit of the mortality estimates generated by standard stochastic mortality models such as the Lee-Carter and Renshaw-Haberman. Moreover, Hainaut (2018) applied neural networks to investigate the latent factors of mortality and predict them consistent with a random walk with drift. Furthermore, Richman and Wuthrich (2018) expanded the Lee-Carter model to multiple populations with the aid of neural networks. But, Levantesi and Pizzorusso (2019) applied three machine learning algorithms namely decision tree, random forest and gradient boosting in order to standardize a machine learning estimator to be employed to mortality rates fitted by the mortality models Lee-Carter, Renshaw-Haberman and Plat.

When it comes to estimating the parameters of the Lee-Carter model, as illustrated in literatures, the Singular Value Decomposition is the most prominent technique used, however there are other approaches such as the Non-Negative Matrix Factorization. Thus, this study will consider this approach as well when forecasting the mortality rate of the Mauritian population.

Therefore, in regard with the previous studies, this study will focus mainly on applying the deep-learning technique to the Lee-Carter model. Moreover, as mentioned earlier, the techniques Autoregressive Integrated Moving Average (ARIMA) and Long Short-Term Memory (LSTM) will be compared based on the model selection criteria Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) together with the Mean Square Error (MSE). According to Woodun, Ho and Raja (2019), although extensions to the Lee-Carter model such as Lee-Miller (2001), Renshaw-Haberman (2006) and Cairns et al. (2007) were perceived to provide accurate mortality forecasts, in some studies, they were found not to fit the mortality data of some countries. But, the Lee-Carter fitted the Mauritian mortality data effectively and deep-learning integrated Lee-Carter model is yet to be applied to the Mauritian mortality data so that a comparison can be made with the previous death rates forecasts obtained using the original Lee-Carter model. Furthermore, the SVD and the NMF methods will be compared in order to determine the best method in estimating the parameters of the Lee-Carter.

2.6 Summary

Chapter 2 focused mainly on past literature on the application of machine and deep-learning algorithms to the mortality model i.e. the Lee-Carter model in the aim of modelling and forecasting future death rate. The previous works of researchers have shown that implementing deep-learning techniques to the Lee-Carter model did not only enhance the predictive capability of the method, but also much more accurate results were obtained which provided a helping-hand to actuaries and demographers when it comes to developing life insurance or pension products. Additionally, the shortcomings and extensions of the

Lee-Carter model were elaborated. Moreover, some of the model selection criteria were discussed in order to test the performance of the model before performing any forecasting.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

This chapter will emphasize mainly on the proposed research methodology of this study. In addition to this, the research framework will highlight on how the project will proceed and based on the dataset provided by the Statistics Department of Mauritius, the data analysis techniques will be implemented. An in-depth exploration of the techniques Long-Short Term Memory Recurrent Neural Network as well as matrices decomposition methods such as Singular Value Decomposition along with Non-Negative Matrix Factorization will be provided. The ethical considerations of this study will be addressed together with the data pre-processing as well as data analysis stages.

3.2 Research Approach

This study focuses mainly on quantitative approach whereby continuous secondary data will be used, obtained from the online database of Statistics Department of Mauritius (Martial, 2018). The positivist approach will be taken into consideration since the research involves the prediction of mortality rates for the Mauritian population while supporting the aim and objectives of the research.

This research will apply the Recurrent Neural Network with the architecture Long Short-Term Memory to forecast the mortality index parameter. A comparison between the Singular Value Decomposition (SVD) and Non-Negative Matrix Factorization methods will be conducted in order to determine the best technique to find the three mentioned parameters of the Lee-Carter model. Also, the results obtained for the mortality index parameter through ARIMA also will be derived in order to compare with those from the Recurrent Neural Network with Long Short-Term Memory. Based on model selection tools such as measures of errors namely Mean Square Error (MSE) along with Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC), the best method during the comparison mentioned earlier, will be adopted. The flow charts in Figure 3.1 and Figure 3.2 illustrate the framework and mechanism of the whole study.

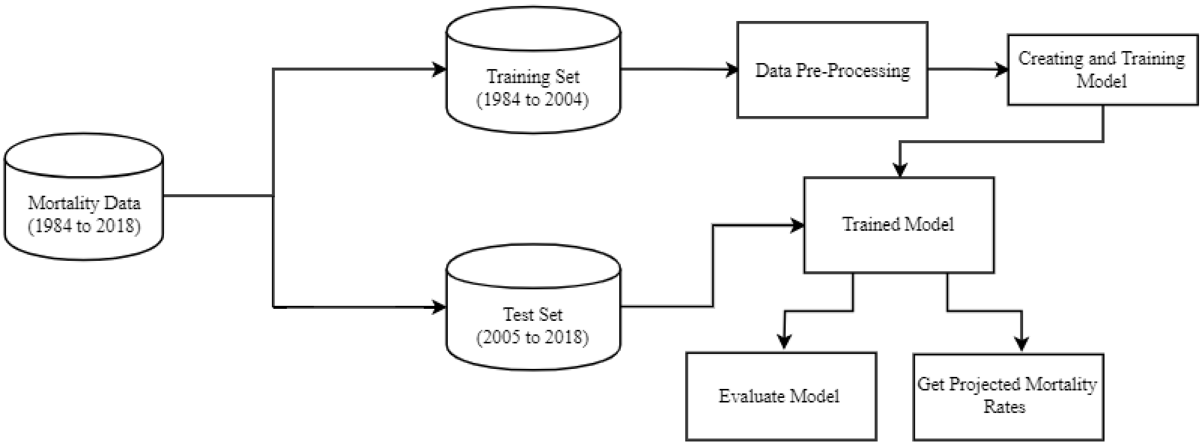

Figure 3.1: Research Mechanism

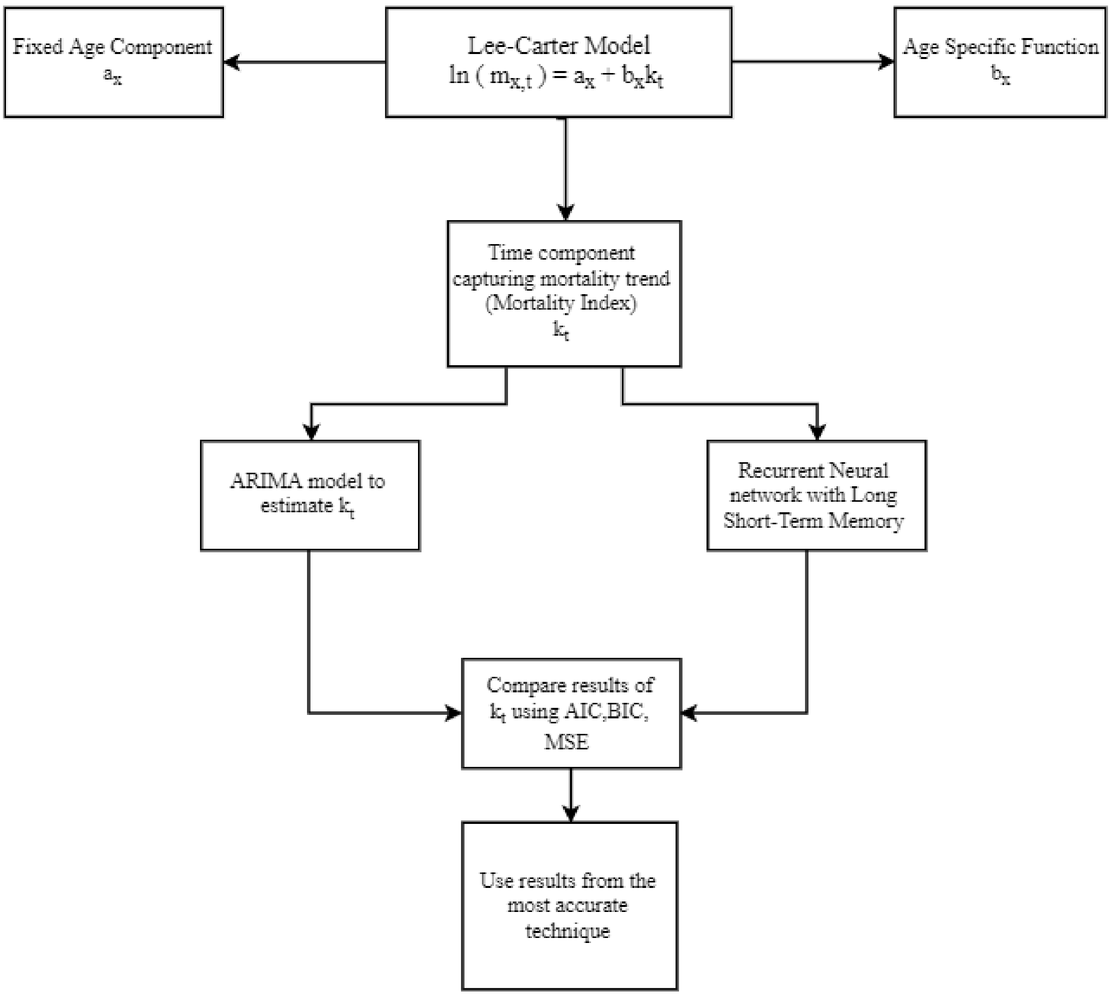

Figure 3.2: Proposed Methodology

3.2.1 Description of Data Set

The dataset comprises of death rates for males, females as well as for the total population for a period of 1984 to 2018 for each age from 0 to 85+ years. The mortality data summarizes the rate at which deaths occurs for a person of age *x* during the year *t* over a total population during that year *t* of that particular age x. The mortality rate is usually computed using the following formula.

$$m_{x,t} = \frac{D_{x,t}}{E_{x,t}} = \frac{deaths\ during\ year\ t\ aged\ x\ last\ birthday}{average\ population\ during\ year\ t\ aged\ x\ last\ birthday}$$

$m_{x,t}$ is defined as the crude death rate for age *x* in calendar year *t*

When carrying out the research, it is essential to abide by some ethical rules and regulations. The main aim of research ethics refers to the responsibility of the researcher to ensure that the secondary data collected for analysis will be used only for the research purpose and will not be disclosed to unauthorized parties. Besides, proper citations and resources used will be provided to avoid any cases of plagiarism.

Table 3.1: Description of Data

Attributes	**Description**	**Example**
Year	Calendar Year from 1984 to 2018	‘1984’, ‘1994’…
Age	Last birthday Age	‘0’, ‘10’,…, ‘85+’
Male	Death rate for male	‘0.000234’, ‘0.000300’
Female	Death rate for female	‘0.000100’, ‘0.000200’
Total	Death rate for overall population	‘0.011900’, ‘0.000620’

3.2.2 Lee-Carter Model

The Lee-Carter model has extensively been cited by many researchers when modelling and forecasting long-run age-specific mortality rates for any particular nation. The main advantage of this model is that it is simple, and parameters are effortlessly estimated and interpreted. Additionally, Booth and Tickle (2008) emphasized that the Lee-Carter model encompasses insignificant subjective judgement and generates stochastic forecasts with probabilistic prediction intervals. In the aim of improving the Gompertz law of mortality, Lee and Carter (1992) added a parameter k_t which ensures that the fitted deaths are equal to the total number of observed deaths in each year as established by the Equation 8.

$$\ln(\boldsymbol{m_{x,t}}) = \boldsymbol{a_x} + \boldsymbol{b_x k_t} + \boldsymbol{\varepsilon_{x,t}} \qquad \textbf{[8]}$$

where:

$\boldsymbol{m_{x,t}}$ is the central mortality rate at age x in year t.

$\boldsymbol{a_x}$ is the average (over time) log-mortality at age x.

$\boldsymbol{b_x}$ measures the response at age x to change in the overall level of mortality over time.

$\boldsymbol{k_t}$ represents the overall level of mortality in year t.

$\boldsymbol{\varepsilon_{x,t}}$ is the residual.

3.2.3 Techniques to estimate $\boldsymbol{a_x}$ and $\boldsymbol{b_x}$

In regard with the Lee-Carter model in Equation 4, the parameters a_x and b_x are to be estimated whereby a_x denotes the average log-mortality at age *x* while b_x is the response at age *x* to change in the overall level of mortality over time. In general the Singular Value Decomposition (SVD) method is used to find the values of these parameters, while abiding by the following constraints: $\sum_t \boldsymbol{k_t} = \boldsymbol{0}$ and $\sum_x \boldsymbol{b_x^2} = \boldsymbol{1}$. However, this study proposes to compare between Singular Value Decomposition (SVD) and Non-Negative Matrix Factorization (NMF), whereby the best technique will be applied to estimate a_x and b_x. The following discussion gives an overview of the SVD and NMF.

3.2.3.1 Singular Value Decomposition

The SVD technique is one which produces matrices when the matrix $Z = \ln(m_{x,t}) - \hat{a}_x$ is applied in order to estimate the parameters of the Lee-Carter model. The following matrices are obtained:

$$PdQ^1 = SVD\ (Z_{xt}) = d_1 P_{x1} Q_{t1} + \cdots + d_X P_{xX} Q_{tX} \qquad \textbf{[9]}$$

Approximation to the first term gives the estimates $\widehat{\boldsymbol{b_x}} = \boldsymbol{P_{x1}}$ and $\widehat{\boldsymbol{k_t}} = \boldsymbol{d_1 Q_{t1}}$

The above can be illustrated as shown in Equation 10:

$$Z = \ln(m_{x,t}) - \hat{a}_x \quad \textbf{[10]}$$

That is;

$$Z = \begin{pmatrix} \ln(m_{0,1984-1985}) - \hat{a}_0 & \cdots & \ln(m_{0,2015-2017}) - \hat{a}_0 \\ \vdots & \ddots & \vdots \\ \ln(m_{85+,1984-1985}) - \hat{a}_{85+} & \cdots & \ln(m_{85+,2015-2017}) - \hat{a}_{85+} \end{pmatrix} \quad \textbf{[11]}$$

Applying SVD to matrix Z will give the following decomposition:

$$SVD\ (Z) = d_1 P_{x1} Q_{t1} + d_2 P_{x2} Q_{t2} + d_3 P_{x3} Q_{t3} \ldots + d_X P_{xX} Q_{tX} \quad \textbf{[12]}$$

Where X = rank (Z), $d_i(i = 1,2, \ldots X)$ are singular values in inclining order with $P_{x,i}$ and $Q_{t,i}$ (i = 1,2,3,…X) as the conforming left and right singular vectors.

$$\widehat{\boldsymbol{b_x}} = \boldsymbol{P_{x1}} \text{ and } \widehat{\boldsymbol{k_t}} = \boldsymbol{d_1 Q_{t1}}$$

The SVD method is actually computed in the R program itself and is renowned for its simplicity for the parameter estimation.

3.2.3.2 Non-Negative Matrix Factorization

As discussed earlier, the Non-Negative Matrix Factorization method is an unsupervised learning one and it assumes that both the data samples and data dimensions are independently distributed. The following equations explain the NMF algorithm:

Let D = [$d_1 \ldots d_j \ldots d_T$] be a series of M-dimensional non-negative vectors. The NMF algorithm models the data as a linear combination of N M-dimensional non-negative basis vectors for N ≤ M, such that

$$d_j = \sum_{k=1}^{N} C_{kj}\, \boldsymbol{W}_k + \varepsilon \quad \textbf{[13]}$$

Where $\boldsymbol{W}_k$ is the kth basis vector, C_{kj} is the non-negative coefficient for the kth basis vector at data point j, and ε is any noise unexplained by the model.

Let W = [$w_1 \ldots w_k \ldots w_N$], so that $D \approx WC$, where C is an NxT matrix with C_{kj} being the matrix entry at row k and column j. When ε follows a Gaussian distribution with constant variance, W and C matrices may be estimated by using the following equations:

$$C_{kj}^{i+1} = C_{kj}^{i} \frac{[(W^i)^T D]_{kj}}{[(W^i)^T W^i C^i]_{kj}} \quad \textbf{[14]}$$

Based on the above equation, i represents the iteration number and T denotes the matrix transpose.

3.2.4 Models to estimate mortality index parameter k_t

When it comes to forecasting the death rates, the mortality index parameter i.e. k_t needs to be projected for future time points. The Box-Jenkins times series approach, also known as Autoregressive Integrated Moving Average (ARIMA) is commonly used, however, it has some shortcomings such as its inability to describe the future mortality shape as well as its inaccurate detection of the pattern in the values of the level of mortality. Thus, in this study, a comparison between the ARIMA and the Recurrent Neural Network with Long-Short-Term Memory (RNN-LSTM) architecture will be conducted to show whether the LSTM is the best model to estimate the mortality index parameter or not. The following discussion outlines the ARIMA and RNN-LSTM models.

3.2.4.1 ARIMA Model

The Auto Regressive Integrated Moving Average (ARIMA) model is usually regarded as a class of models which explains a given time series data based on its own past trends and is used to forecast future values. It is commonly characterised by three terms ARIMA (p,d,q), where p is the order of the Auto Regressive term, q is the order of the Moving Average term while di is the number of differencing required to make the time series data into stationary. A stationary data is one whereby the variance stays constant over time. As discussed earlier, the parameter k_t is generally modelled and predicted using the random walk with drift i.e. ARIMA (0,1,0) model, where p is 0, d is 1 and q is 0, while assuming that the parameters a_x and b_x are invariant with time, as shown in Equation 15 (Lee and Carter, 1992).

$$\boldsymbol{k_t = k_{t-1} + d + \varepsilon_t} \qquad \textbf{[15]}$$

Where d is the drift parameter and $\boldsymbol{\varepsilon_t}$ is the error term normally distributed with mean 0 and variance 1.

However, this study will apply mortality data from 1984 to 2018 to again fit the ARIMA model since existing research conducted by Woodun, Ho and Raja (2019) used Mauritian population death data from 1984 to 2016 only.

3.2.4.2 Long-Short Term Memory Architecture

Recurrent Neural Networks (RNN) are usually defined as artificial neural network designed to recognise patterns in sequences of data. In addition to this, Nigri et al. (2019) accentuated that these algorithms take into consideration time and sequence and, thus have a time-based dimension. The latter also added that although the feed-forward neural networks are considered to be a powerful analysis tool, they were perceived to be ineffective when it comes to time series and sequential data. Both feed-forward and recurrent neural networks direct information through a sequence of mathematical operations computed at the nodes of the network but the feed-forward one feeds information straight through while the recurrent one cycles it through a loop. Salehinejad et al. (2018) underscored that the main objective of recurrent neural network is to curtail the difference between the output and target pairs by adjusting the weights of the network. Moreover, the latter added that the gradient descent is a simple and eminent optimisation technique in deep learning, whereby the weights of the model are enhanced by finding the error function derivatives with respect to each member of the weight matrices in the model. But, calculating error-derivatives through time is problematic, owing to the relationship among the parameters and the dynamics of the recurrent neural network.

Additionally, Hochreiter and Schmidhuber (1997) highlighted that it is possible to capture complex patterns of data in real-life by applying a robust non-linearity, but this may eventually cause the recurrent neural network model to suffer from the vanishing gradient problem during the training phase. This issue implies that the gradient magnitudes shrink exponentially as they are propagated back through time. As a result, this phenomenon causes the memory of the network to overlook long-term dependencies and to barely learn the relationship between temporally distant proceedings. As such, to solve this problem, Hochreiter and Schmidhuber (1997) established the Long Short-Term Memory (LSTM) architecture that considers the correlation between the sequential data to be discoursed in the next section.

Salehinejad et al. (2018) accentuated that the Long Short-Term Memory (LSTM) is one of the most effective approaches for plummeting the effects of vanishing and exploding gradients. In addition to this, LSTM method modifies the structure of hidden units from activation functions such as “sigmoid” or “tanh” to memory cells, in which their inputs and outputs are monitored by gates. In other words, these gates regulate the movement of information to hidden neurons and preserve extracted features from previous time-steps (Le, Jaitly and Hinton, 2015). The “tanh” and “sigmoid” activation functions are defined as shown in the Equation 4 and Equation 5 respectively. Moreover, a range of [0,1] is usually allocated to the “sigmoid” function.

$$\mathbf{tanh}(\boldsymbol{x}) = \frac{\boldsymbol{e^{2x}-1}}{\boldsymbol{e^{2x}+1}} \qquad \textbf{[16]}$$

$$\boldsymbol{\sigma}(\boldsymbol{x}) = \frac{\mathbf{1}}{\mathbf{1}+\boldsymbol{e^{-x}}} \qquad \textbf{[17]}$$

Where $\boldsymbol{\sigma}(\boldsymbol{x})$ is the “sigmoid” function

In this way, the recurrent neural networks gain both long-term and short-term memories, thus producing high performance models and manage time-series data effectively. The gates mentioned earlier act on the signals they receive, and comparable to the neural network’s nodes, they restrict or pass information based on its strength. Consequently, the network is able to remember information for longer periods. The LSTM has undergone numerous enhancements as demonstrated in Gers and Schmidhuber (2000) and is commonly referred to as Vanilla LSTM.

3.2.5 Performance Evaluation Tools

To assess the prediction power of the deep-learning techniques discussed above, it is highly significant to use performance evaluation tools namely Mean Square Error (MSE) as well as information criteria such as Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC). These metrics not only help to evaluate the accuracy of the models but also ensure that better results are obtained, and the objectives of the research are achieved.

3.2.5.1 Mean Square Error (MSE)

The mean square error (MSE) measures the average of the squared difference between the estimated values and what is estimated. In other words, MSE refers to the expected value of the squared error loss. The formula in Equation 18 is usually used to compute the MSE value.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widetilde{y_i})^2 \qquad \textbf{[18]}$$

3.2.5.2 Akaike Information Criterion (AIC)

AIC is a method to choose between the additive and multiplicative error models. It is grounded on the likelihood rather than one-step forecasts and is calculated using the given formula for n being the number of parameters in the Equation 8 and n is between 0 and 85.

$$AIC = n\ \ln(MSE) + 2k \qquad \textbf{[19]}$$

3.2.5.3 Bayesian Information Criterion (BIC)

Bayesian Information Criterion (BIC) is a criterion for model selection and commonly used in time series and linear regression. Lower value of BIC indicates lower error, hence a better model. The value for BIC can be computed using the R programming Software. It is also regarded as the Schwarz BIC and given by the Equation 20.

$$BIC = n\ \ln(MSE) + k\ \ln n \qquad \textbf{[20]}$$

3.3 Summary

Chapter 3 presented the research methodology with all the deep-learning techniques in the aim of achieving the main aim of this study which is to forecast the mortality rate of the Mauritian population using the deep-learning integrated Lee-Carter model. In addition to this, a description of the data was provided as well as the performance evaluation tools to be used to assess the models and to be able to choose the best one when estimating the parameters of the Lee-Carter model. The theory of the methods has been thoroughly described in this chapter and the experimental results will be discussed in the following chapters.

CHAPTER 4

DATA ANALYSIS AND IMPLEMENTATION

4.1 Introduction

This chapter emphasizes mainly on the data analysis of the dataset whereby techniques applied to pre-process the data will be accentuated. In addition to this, during the exploratory data analysis process, visualizations will be provided in order to infer better insights. Moreover, based on the discussion in Chapter 3, a comparison between the Singular Value Decomposition and Non-Negative Matrix Factorization methods will be conducted and based on the measures of errors, the best technique will be chosen for further prediction. Furthermore, different models of the Recurrent Neural Network with Long Short-Term memory architecture will be built and used to forecast the death rate of the Mauritian population with the aid of R programming.

4.2 Data Pre-processing

The dataset was obtained from the Statistics Department of Mauritius, it comprised of mortality rate named ‘death’ and population size named ‘exposure’ for ages 0 to 85+ for the years 1984 to 2018 (SDM, 2018) . As such, the data was divided into training and test set with 1806 and 1204 observations respectively. The training set consists of mortality data from 1984 to 2004 while the test set entails death rate from 2005 to 2018. There are 5 attributes in the dataset and the mortality rate for the Mauritian population will be forecasted for the next 30 years, i.e. from 2019 to 2048.

4.2.1 Initial Data Exploration

Table 4.1 labelled as ‘Exposure Data’ summarizes the number of living individuals recorded for each age for male, female and total population of the country for the year 1984 while the Table 4.2 denoted as ‘Death Data’ gives an overview of the mortality rate of each age for male, female and total population for the year 1984.

Table 4.1: Exposure Data

Year	Age	Male	Female	Total
1984	0	9719	9368	19087
1984	1	18623	17949	36572
1984	2	27527	26530	54057
1984	3	36430	35111	71541
1984	4	45334	43692	89026
1984	5	47218	45450	92669
1984	6	49103	47209	96312
1984	7	50987	48967	99954
1984	8	52872	50726	103597
1984	9	54756	52484	107240
1984	10	53435	51323	104758
1984	11	52114	50162	102276

Table 4.2: Death Data

Year	Age	Male	Female	Total
1984	0	0.0263402	0.0200683	0.0232619
1984	1	0.0200253	0.0152858	0.0176992
1984	2	0.0137105	0.0105034	0.0121364
1984	3	0.0073957	0.0057209	0.0065737
1984	4	0.0010809	0.0009384	0.0010109
1984	5	0.0009779	0.0008612	0.0009207
1984	6	0.0008750	0.0007841	0.0008304
1984	7	0.0007720	0.0007069	0.0007401
1984	8	0.0006691	0.0006297	0.0006498
1984	9	0.0005661	0.0005525	0.0005595
1984	10	0.0005318	0.0005449	0.0005383
1984	11	0.0004975	0.0005372	0.0005171

Using the datasets imported in R Studio, the descriptive statistics summary was derived as shown in Table 4.3.

Table 4.3: Descriptive Statistics of Death Data

	Death Data			
Statistics	**Age**	**Male**	**Female**	**Total**
Mean	43.5	0.023109	0.014465	0.018102
Median	43.5	0.006214	0.002852	0.004534
1st Quartile	22	0.001287	0.000682	0.001003
3rd Quartile	65	0.027269	0.015423	0.021142
Standard Deviation	24.83	0.037930	0.026461	0.03029
Skewness		2.79024	3.17929	2.79559
Kurtosis		13.15915	15.61817	12.5929

Table 4.4: Descriptive Statistics of Exposure Data

	Exposure Data			
Statistics	**Age**	**Male**	**Female**	**Total**
Mean	43.5	32174	32403	64577
Median	43.5	39205	38941	78169
1st Quartile	22	15239	16415	31433
3rd Quartile	65	47601	46375	93954
Standard Deviation	24.83	17215.59	15830.32	33025.26
Skewness		-0.39228	-0.41240	-0.40176
Kurtosis		1.57870	1.61601	1.59463

From the descriptive statistics of the death and exposure datasets, it can be noticed that the mean and median age are 43.5 years, while the standard deviation is 24.83 years. Since the mean is greater than the standard deviation, there is low probability of having outliers in this attribute. However, it can be observed that the mean values of the mortality rates for Male, Female and Total are lower than their standard deviation, implying that outliers might be present. In general, the kurtosis coefficient is considered when determining the presence of outliers in any dataset, whereby a value greater than 3 denotes that there are outliers and vice-versa. Thus, from the kurtosis values of the death data, it can be seen that they are all greater than 3, but those of the exposure data are lower than 3.

Additionally, the coefficient of skewness is given in the summary statistics, which summarize the spread of the data for each attribute. For example, the positive values of the skewness for the mortality data imply positively skewed distribution while the negative ones for the population data signify a negatively skewed distribution for male, female and total population. For instance, the mortality rates for male, female as well as total have skewness values such as 2.790, 3.179 and 2.796 respectively, which are greater than 0. In addition to this, the mean values are greater than the median death rates, thus implying skewed distribution to the right. As such, in the population data, the skewness values for male, female and total are -0.392, -0.412 and -0.402 respectively while the mean values are lower than the median ones. Therefore, the population data for these variables are mostly skewed to the left. The Figure 4.1 and Figure 4.2 illustrate clearly the spread in the data.

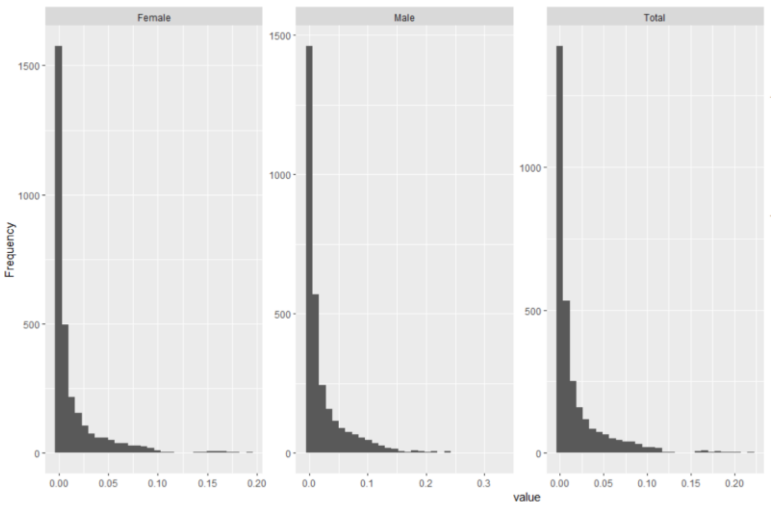

Figure 4.1: Histogram for Death Data

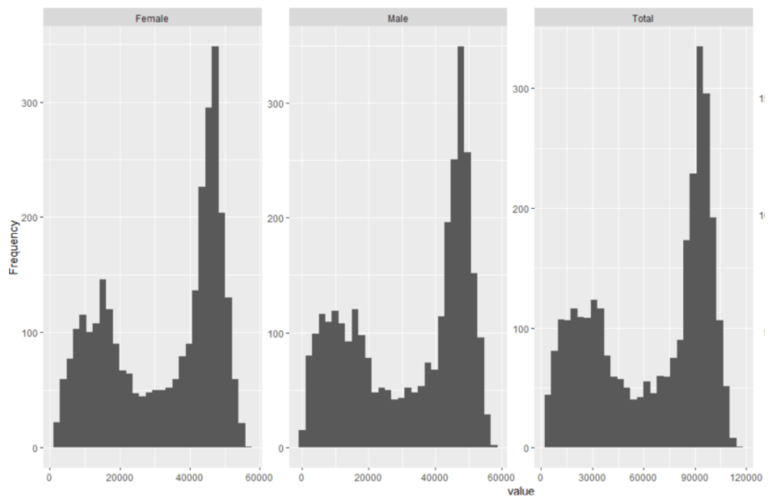

Figure 4.2: Histogram for Exposure Data

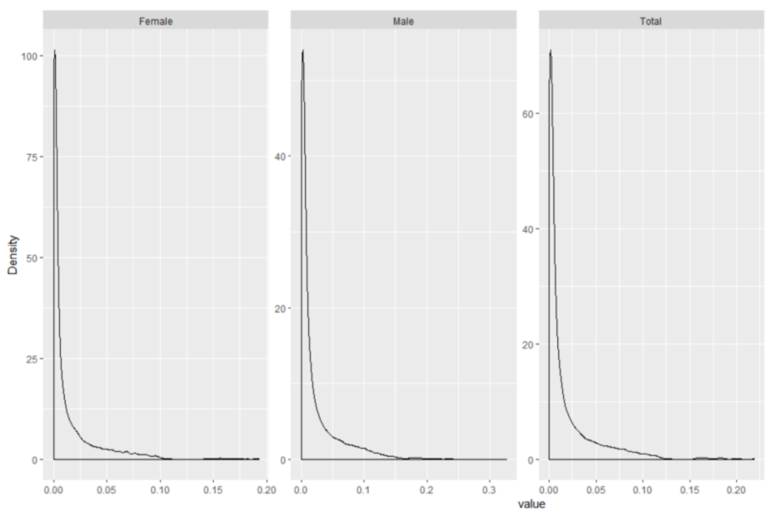

Figure 4.3: Plot density for Death Data

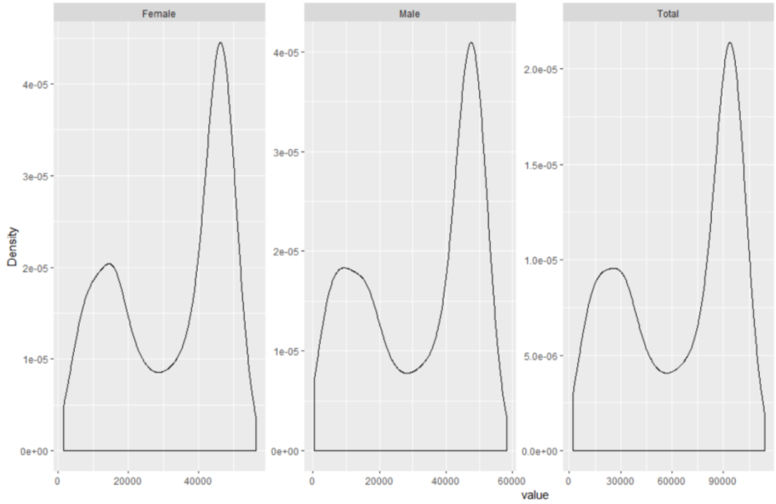

Figure 4.4: Plot density for Exposure Data

In view of the Figure 4.1 and Figure 4.3, it can be deduced that the mortality rate for female, male and total population are positively skewed while considering the Figure 4.2 and Figure 4.4, the data distribution for the population size is almost bimodal for the attributes male, female and total population.

4.2.1.1 Missing Value Treatment

Missing data are a common issue in analysis and can give rise to biases if not treated properly. Therefore, in order to verify whether the datasets had missing values the missing value plot was created as shown in Figure 4.5.

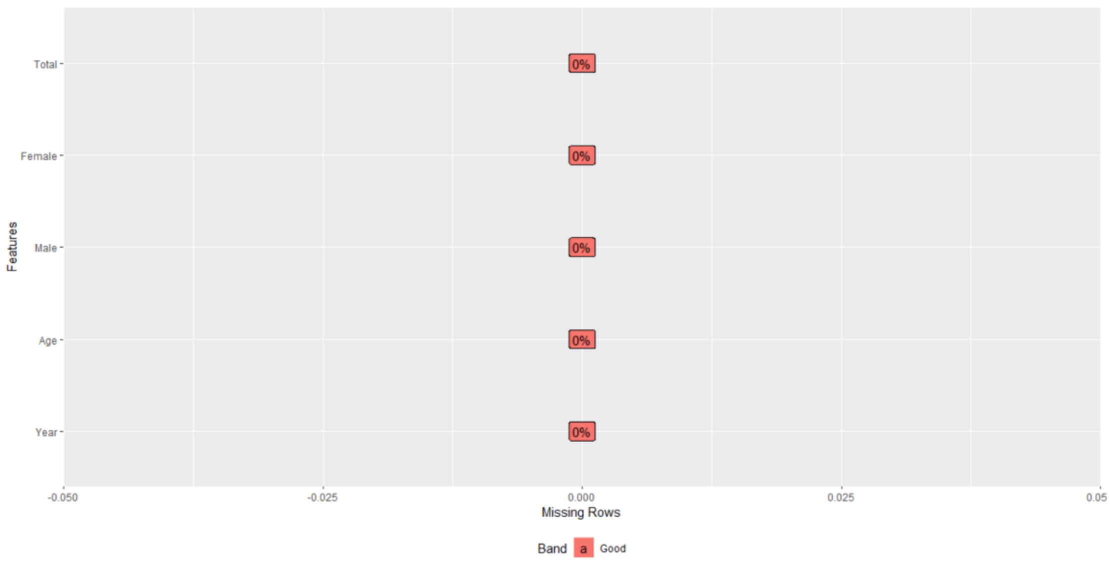

Figure 4.5: Missing value plot showing no missing values

It can be observed from the Figure 4.5 that there are no missing values present in the death and exposure data sets, thus no missing value treatment is required.

4.2.1.2 Outliers Treatment

It is highly significant to identify the presence of outliers in any data set which usually impact the goodness of fit of any predictive models. As such, the following boxplots were plotted to determine whether there is any outlier in the death and exposure datasets. As discussed earlier, the kurtosis values usually help to determine the presence of outliers in any dataset. Even though the kurtosis coefficients of the death data for male, female and total are greater than 3, the box plots in Figure 4.6 and Figure 4.7 do not demonstrate any outliers for these attributes.

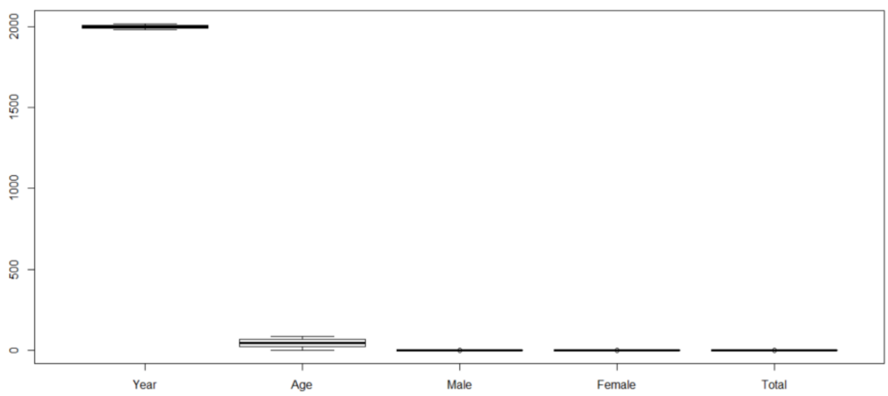

Figure 4.6: Box and Whisker for Death Data

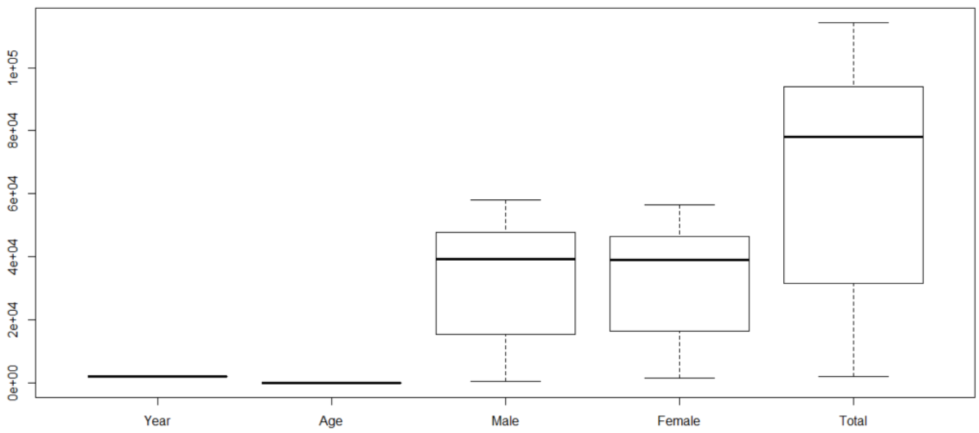

Figure 4.7: Box and Whisker for Exposure Data

The Figure 4.6 and Figure 4.7 illustrates that there are no outliers in the dataset of both mortality and population size.

4.2.1.3 Multicollinearity Treatment

Before modelling any predictive models, it is highly significant to verify whether the variables in the dataset are correlated or not. The presence of multicollinearity in general affects the goodness of fit of models as well as reduces their reliability. Therefore, the correlation matrices for the mortality and exposure data were plotted as demonstrated in Figure 4.8 and Figure 4.9. It can be observed that the correlation is high between the attributes, which makes sense since the mortality rate and size of population depends highly on the male and female. Thus, no treatment is required.

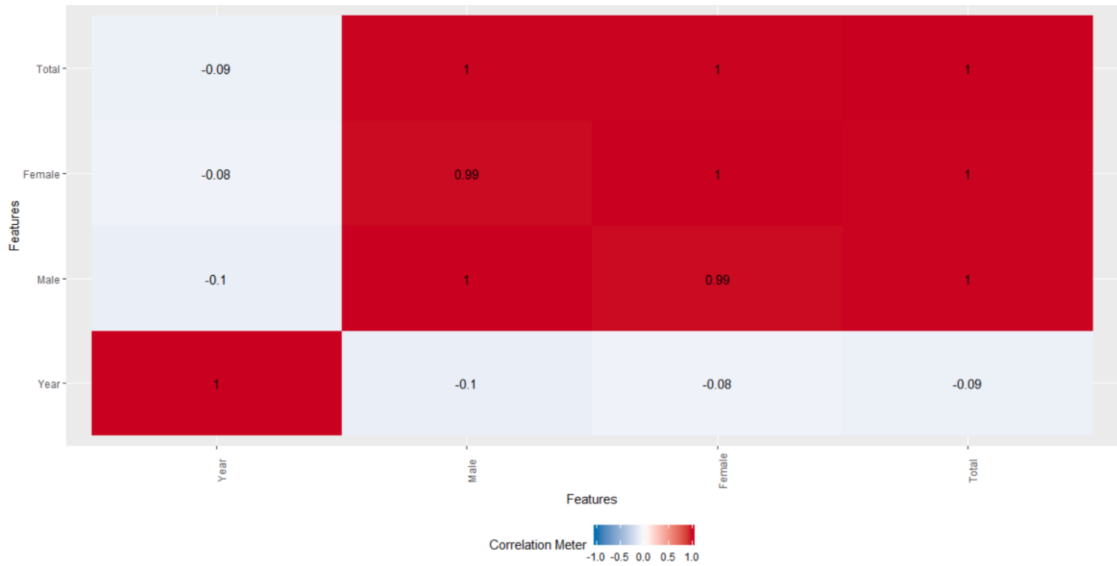

Figure 4.8: Correlation Matrix of Death Data

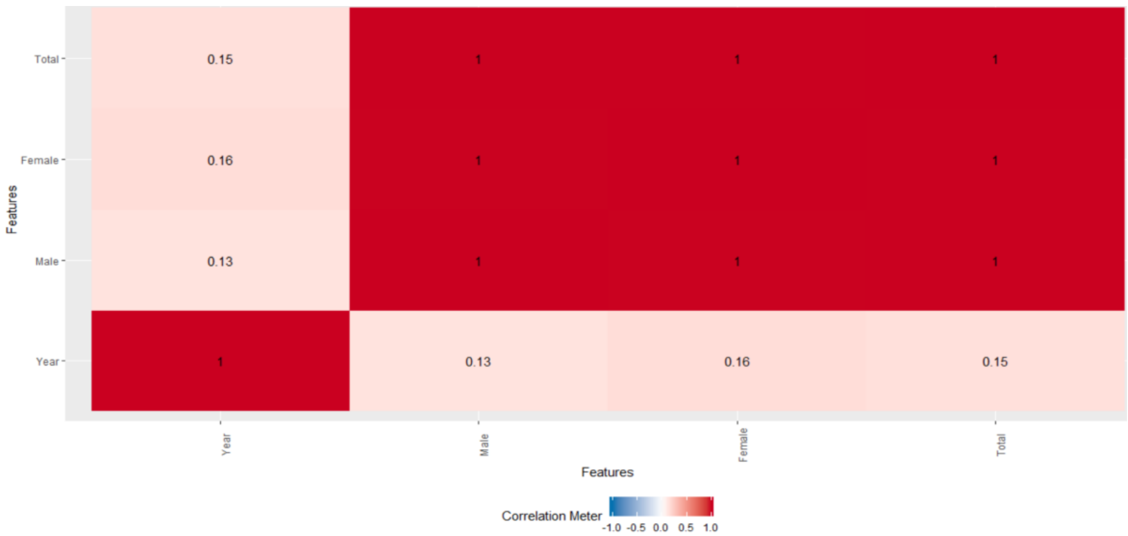

Figure 4.9: Correlation Matrix for Exposure Data

4.3 Exploratory Data Analysis (EDA)

This section focuses mainly on univariate analysis of the attributes in the datasets in the aim of gaining insights from the features of the data. In addition, a dashboard was created with the aid of Tableau to give an overview of the variables in the datasets.

4.3.1 Mortality Data Analysis

The Figure 4.10 illustrates the trend in the mortality rate for the Mauritian population for male, female as well as total. It can be observed that there is a decreasing trend in the mortality rate against years as shown in Figure 4.10.

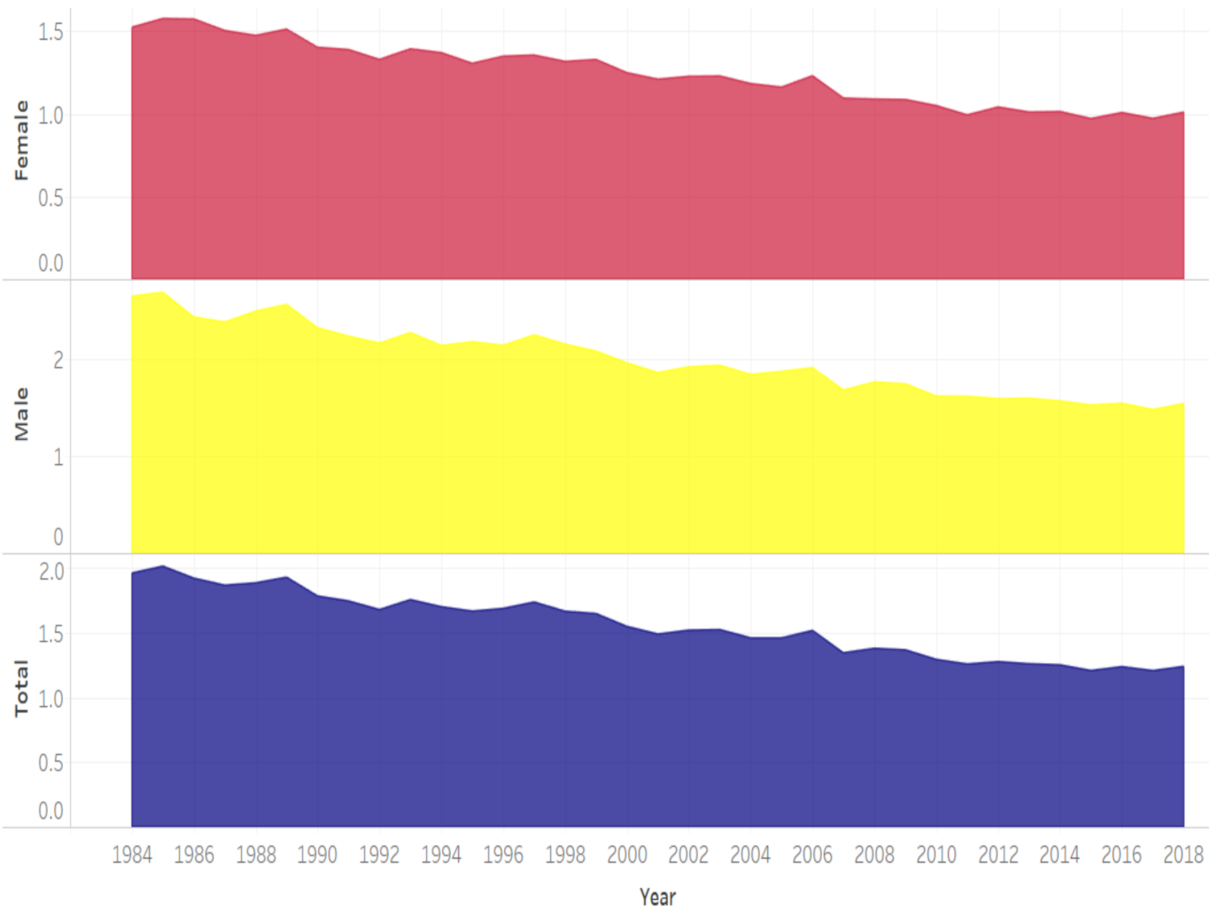

Figure 4.10: Graph showing trend in mortality rate

Figure 4.10 also illustrates that the death rates of both male and female have been fluctuating since 1984.

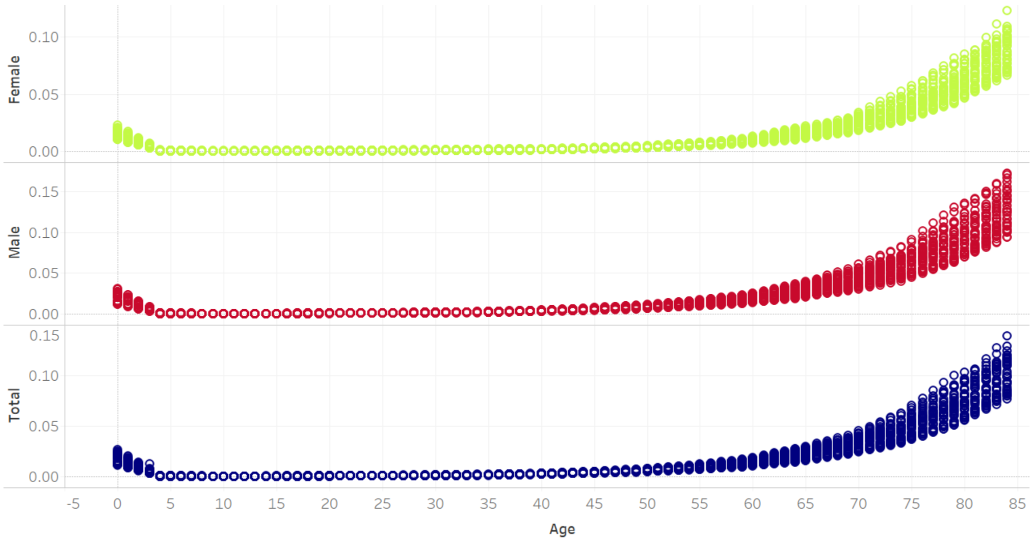

Figure 4.11: Mortality rate against Age

Figure 4.11 illustrates that the death rate increases as the age increases. In other words, there is high probability of death as the individual gets older.

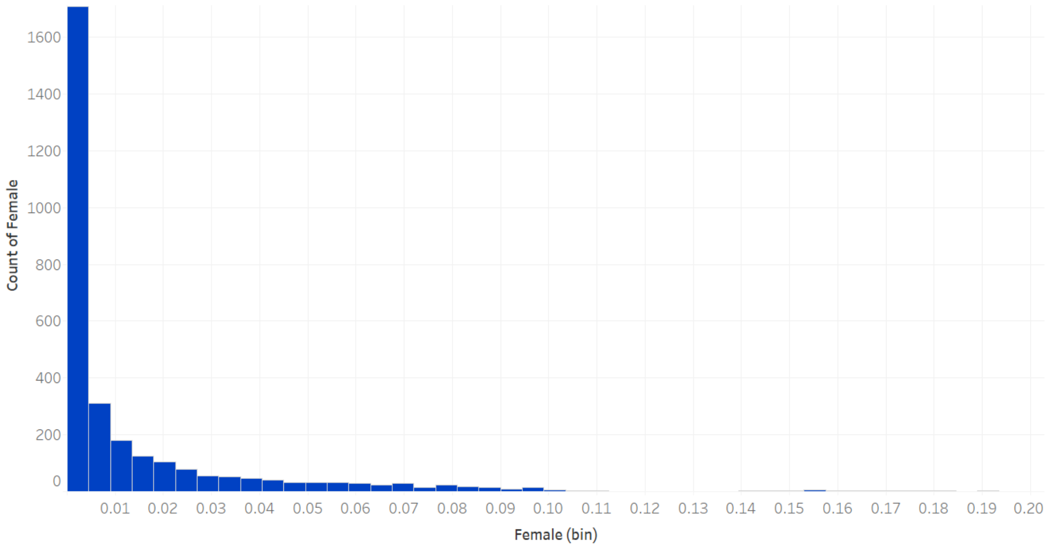

Figure 4.12: Histogram showing distribution in female mortality rate

Figure 4.12 shows that the spread of the data for the female variable is mostly skewed to the right. This can also be explained using the skewness coefficient which was found to be 3.179 from the summary statistics in Section 4.2.1. It can also be noticed that majority of the female in Mauritius low mortality rate between the range 0 and 0.01. Furthermore, the kurtosis value is 15.618, signifying that the distribution is leptokurtic i.e. its tails are longer and central peak is higher.

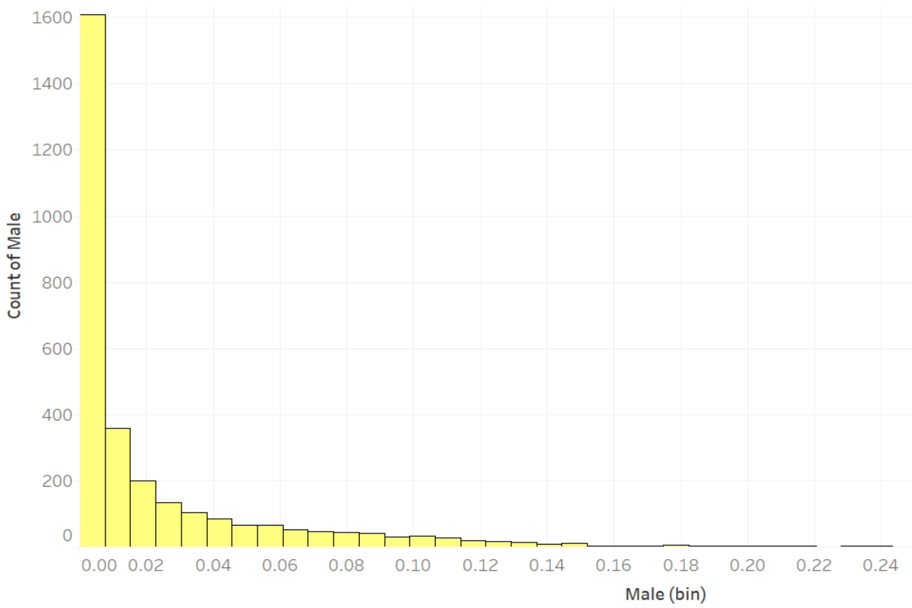

Figure 4.13: Histogram showing distribution in mortality rate of male

Figure 4.13 portrays that mortality rate for male also is distributed mostly between the range of 0 and 0.02. In addition to this, the coefficient of skewness which is 2.790 and kurtosis value of 13.159 illustrate that the data is positively skewed with a leptokurtic distribution. Furthermore, the tail is longer and mostly spread to the right.

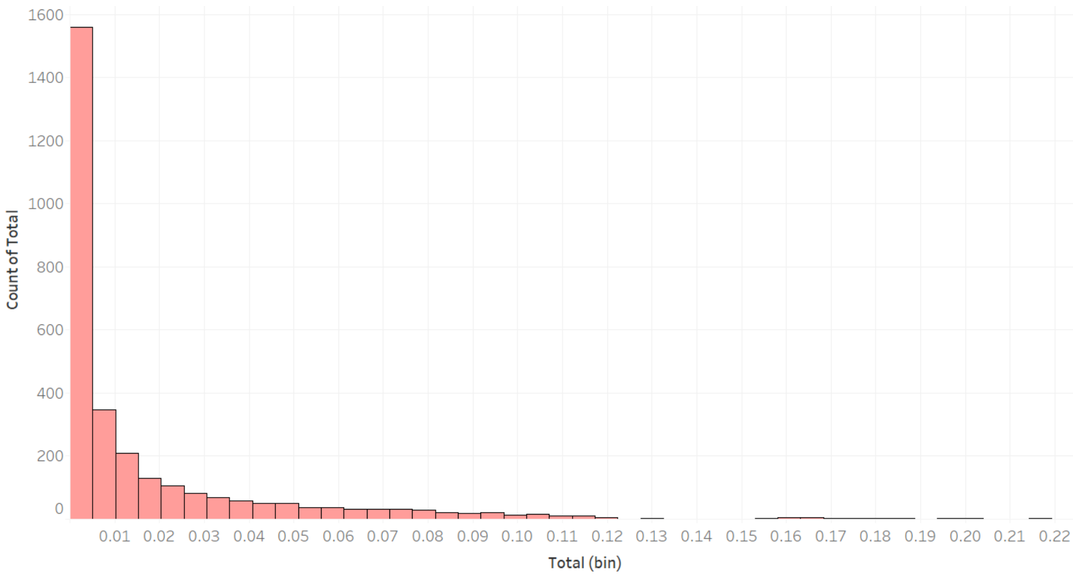

Figure 4.14: Histogram showing distribution in mortality for total population

The histogram in Figure 4.14 exhibits that the total population of Mauritius has a positively skewed distribution for its mortality rate with a value of skewness of 2.796 and kurtosis of 12.593.

4.3.2 Population Data Analysis

The area graph in Figure 4.15 portrays the trend in the population size whereby the upward trend is clearly observed over the years for each gender category.

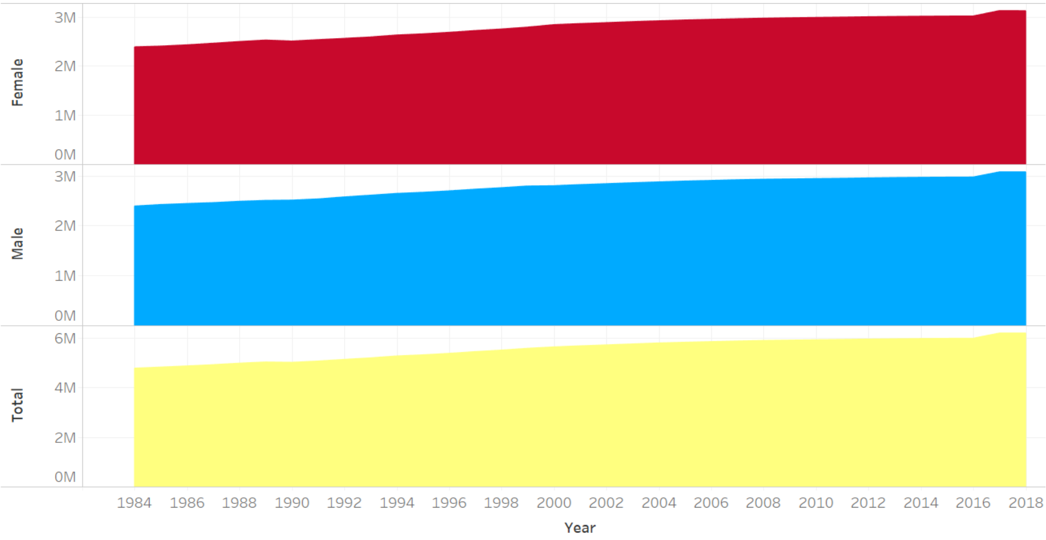

Figure 4.15:Area graph showing upward trend in population size

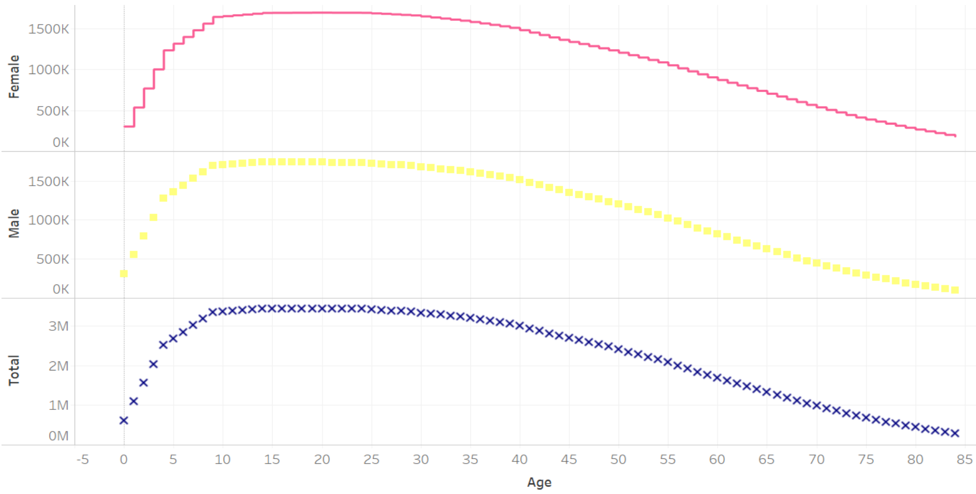

Figure 4.16: Population Size against Age

Figure 4.16 illustrates that the current population of Mauritius comprises mostly of people between the age group of 10 and 40, whereby the number of individuals aged more than 80 years is diminishing over the years. Although the overall population size is increasing, the number of people for each age category varies significantly. The histograms in Figure 4.17, Figure 4.18 and Figure 4.19 portray the spread in the population data, i.e. a bimodal distribution for male, female and total.

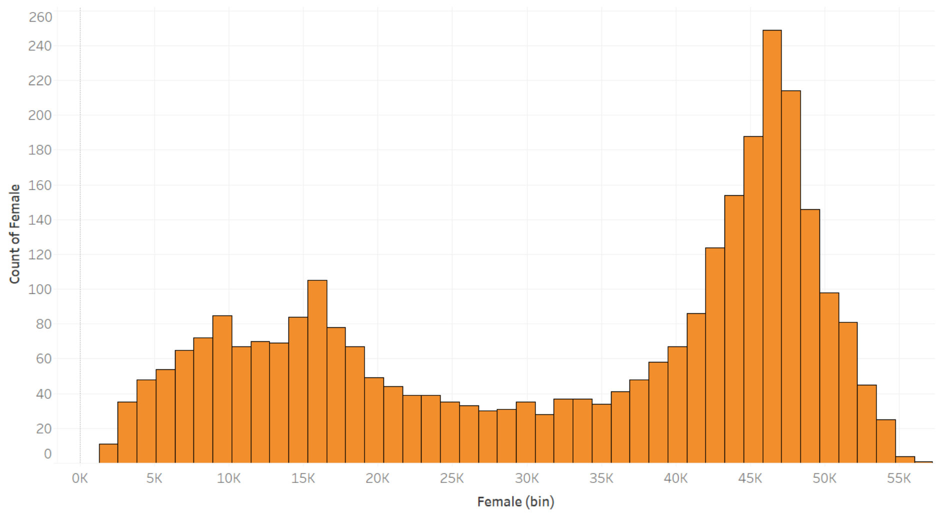

Figure 4.17: Histogram showing distribution in population size for female

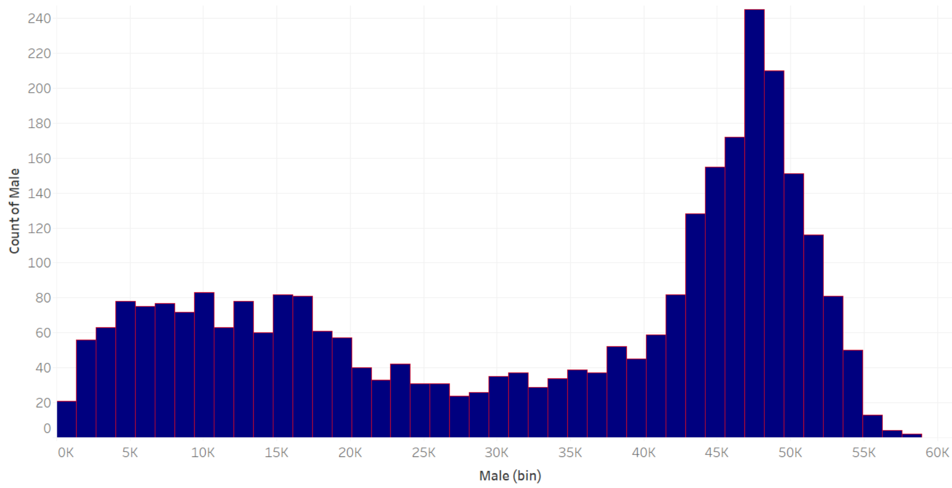

Figure 4.18: Histogram showing distribution in population size for male

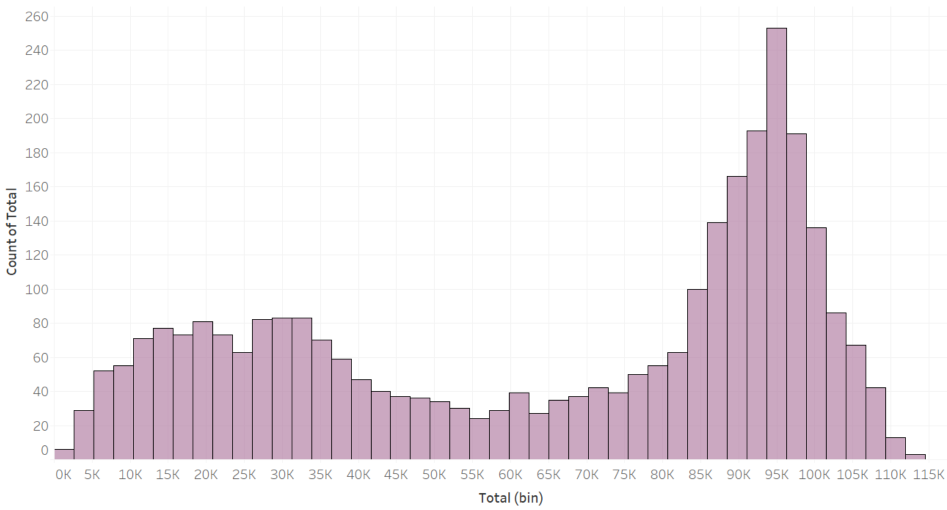

Figure 4.19: Histogram showing distribution in population size for total

4.3.3 Dashboard

The dashboards in Figure 4.20 and Figure 4.21 summarize the exploratory data analysis of both the mortality and population data. These dashboards will help actuaries and demographers to gain insights on the current trends of the demographics of the country. The latter will also be able to develop strategies in order to cater for the effects of the downward trends in the mortality rate of the Mauritian population. They will be able to enhance pricing methods of insurance policies and develop appropriate pension plans.

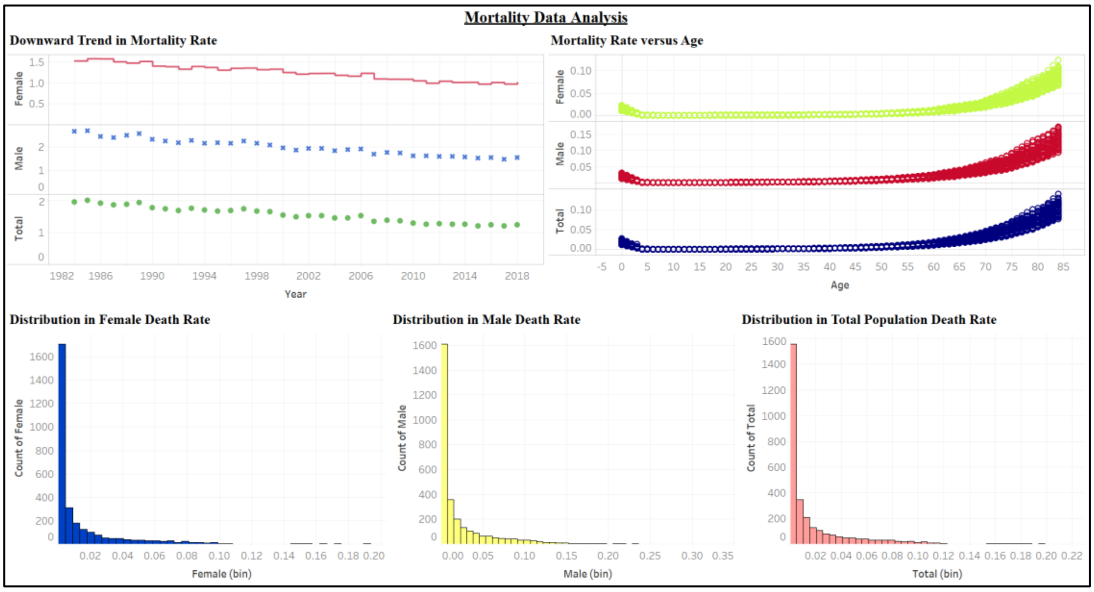

Figure 4.20: Dashboard 1-Mortality Data Analysis

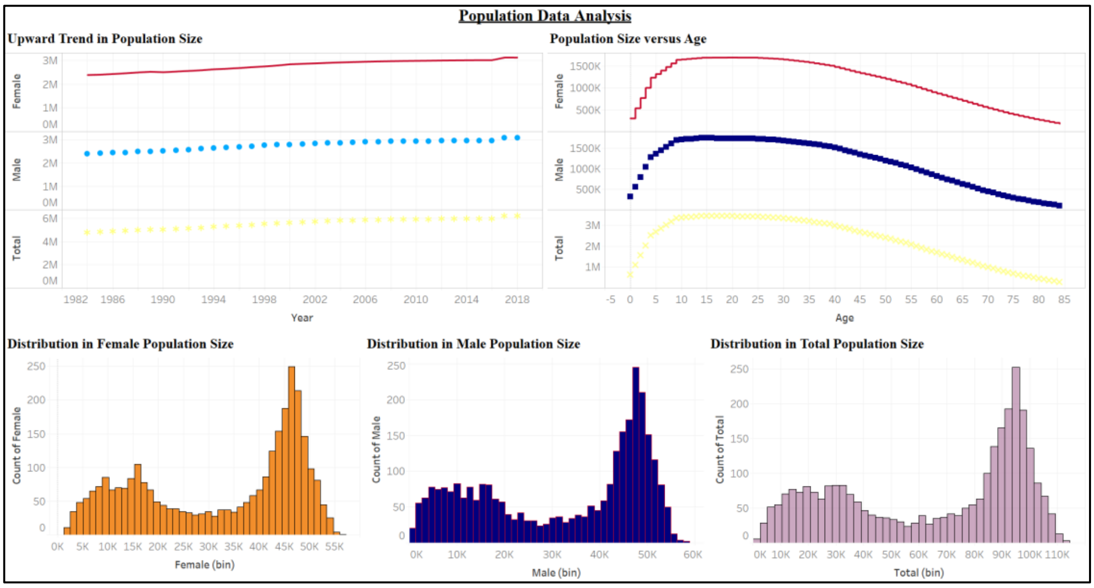

Figure 4.21: Dashboard 2- Population Data Analysis

4.4 Predictive Modelling

As discussed in Section 1.4, the main goal of this study is to forecast the mortality rate of the Mauritian population using the deep-learning integrated Lee-Carter model. This section highlights mainly the modelling of the mortality forecasting technique. A comparison between the techniques namely SVD and NMF as well as between the models ARIMA and LSTM will be conducted to improve the accuracy of the original Lee-Carter model by choosing the best one, based on MSE, AIC and BIC.

4.4.1 Creation of Demogdata

In the aim of fitting the Lee-Carter model, it is highly significant to create a demogdata object which implies the use of demographic data such as death rate, life expectancies and population size. As such the following R codes were run using the demography and forecast packages.

<u>R Codes</u>

```
> maledata<-demogdata(data=mortalityMale,pop=populationMale,ages=AGE,years=YEAR,type="mortality",label="Mauritius",name="Male",lambda=1)
> maledata
Mortality data for Mauritius
    Series: Male
    Years: 1984 - 2018
    Ages:  0 - 85
>
> femaledata<-demogdata(data=mortalityFemale,pop=populationFemale,ages=AGE,years=YEAR,type="mortality",label="Mauritius",name="Female",lambda=1)
> femaledata
Mortality data for Mauritius
    Series: Female
    Years: 1984 - 2018
    Ages:  0 - 85
>
> totaldata<-demogdata(data=mortalityTotal,pop=populationTotal,ages=AGE,years=YEAR,type="mortality",label="Mauritius",name="Total",lambda=1)
> totaldata
Mortality data for Mauritius
    Series: Total
    Years: 1984 - 2018
    Ages:  0 - 85
```

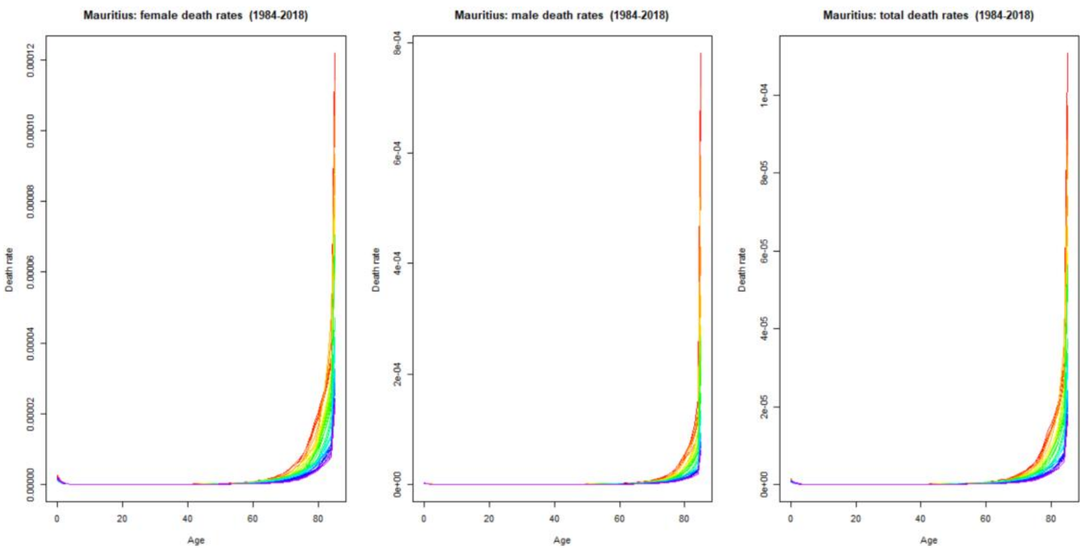

Figure 4.22: Death rates versus Age

Figure 4.22 plots the death rates of the male, female and total population against age of individuals. It can be noticed that there is high chance of death for people over the age of 65 years old. However, to better observe the trend in the mortality, it is crucial to plot the mortality curve in regard with the Gompertz Law of Mortality i.e. the Logarithmic value of death rate against Age, as discussed in Section 2.3.1.

<u>R Codes</u>

```
> demogdata=read.demogdata("Mortality.txt","Population.txt",type="mortality",label="Mauritius",max.mx=10,skip=0,scale=1)
> demogdata
Mortality data for Mauritius
    Series: male female total
    Years: 1984 - 2018
    Ages:  0 - 85
```

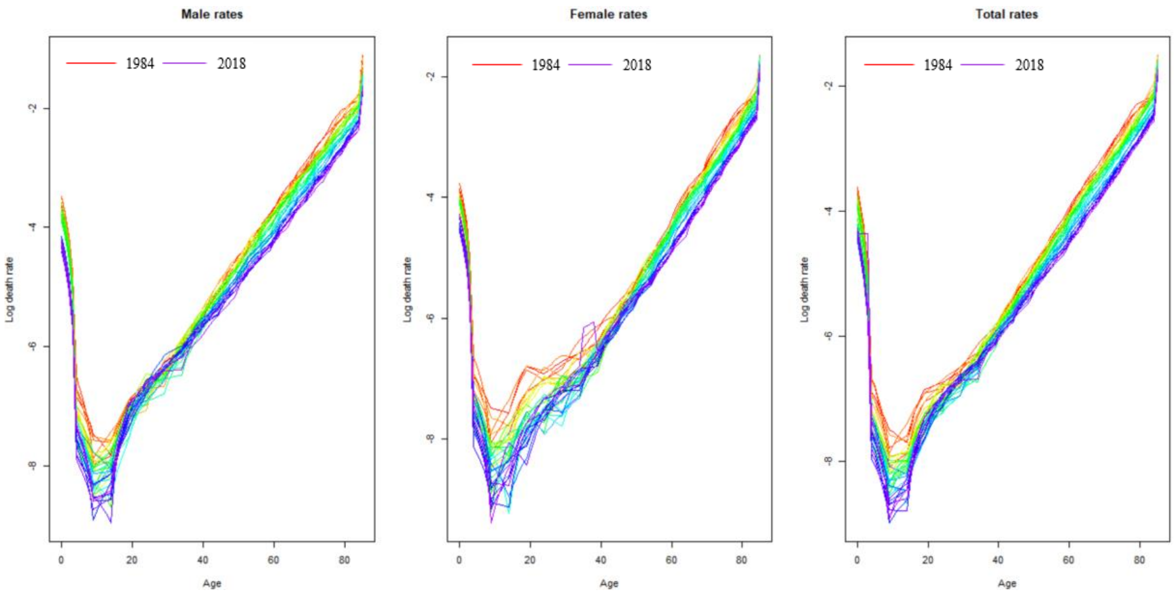

Figure 4.23: Log Death Rates against Age

Figure 4.23 portrays the Log death rates of male, female and total population against age of individuals, whereby the colourful lines represent the different mortality trends for the years 1984 to 2018. The rainbow package in R was used to plot the graph above. It can be deduced that males tend to have higher risk of death as compared to females, since the mortality curve for males shows that the probability of death increases faster as from age 17 years onwards while that of females only starts rising at a fast pace as from the age 50 years and above. This shows that females have a higher life expectancy than males. Additionally, it can be observed that there is a dip in the risk of death for male, female as well as overall population around the age of 10 to 15 years. In other words, infants of age 0 to 2 years old have lower risk of death, but as soon as they reach the age of adolescence, the mortality risk rises sharply. Furthermore, the mortality bump in male mortality curve is usually explained by the subsequent deaths of young men due to accidents, health-related diseases caused by cigarette and alcohol intake, while the dip in female mortality curve is most commonly due to the high risk of maternal mortality in young women. As it can be seen, after the age of 30 years, the risk of mortality increases exponentially for male, female as well as for the overall total population. In the aim of further analysing the probability of death for the Mauritian population over the years, Figure 4.24 was plotted, i.e. Log death rates against Years.

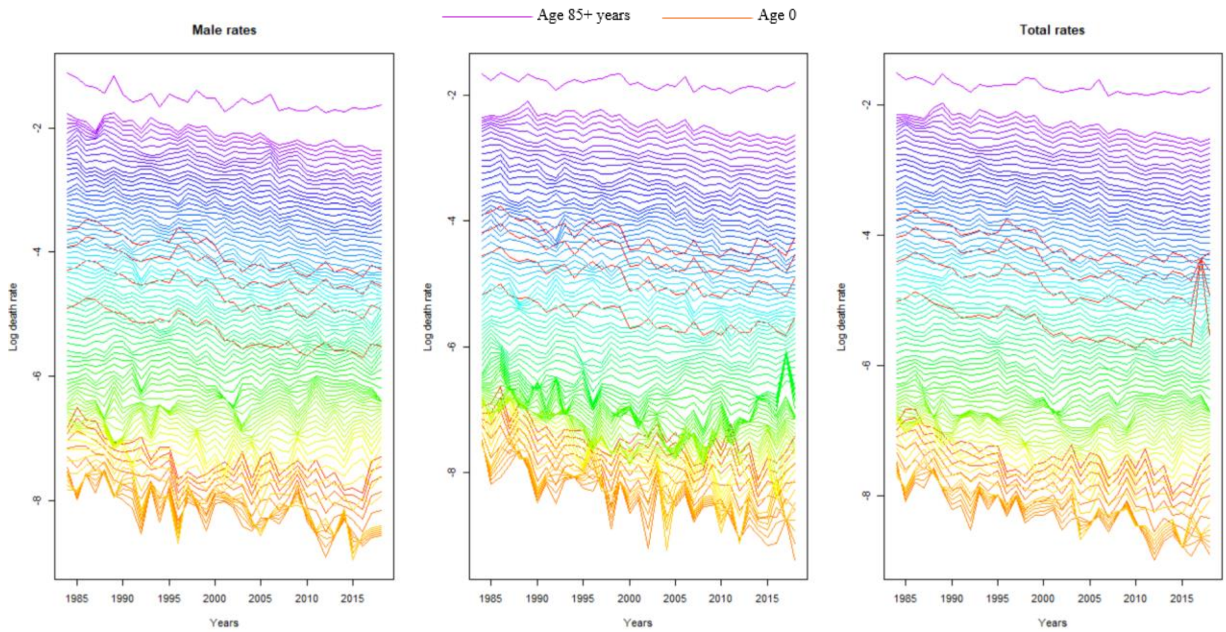

Figure 4.24: Log Death Rates against Years

In Figure 4.24, the colourful lines represent each age from 0 to 85+ years, whereby it can be observed that the first decreasing purple trend line denotes the risk of mortality for the individuals of age 85 years and above, i.e. the latter have high chance of dying.

4.4.2 Lee-Carter and Singular Value Decomposition

As accentuated in Section 2.3.2, the original Lee-Carter model applies the Singular Value Decomposition technique in order to estimate the parameters $\boldsymbol{a_x}$, $\boldsymbol{b_x}$ and $\boldsymbol{k_t}$ of the mortality forecasting model as shown in the Equation 18.

$$\ln(\boldsymbol{m_{x,t}}) = \boldsymbol{a_x} + \boldsymbol{b_x k_t} + \boldsymbol{\varepsilon_{x,t}} \qquad \textbf{[21]}$$

The following steps are generally considered when performing the Singular Value Decomposition technique:

1. $\widehat{a_x} = \frac{1}{n}\sum_1^n \ln(m_{x,t})$
2. A matrix $Z_{x,t}$ is generated for estimating b_x and k_t.
3. Singular Value Decomposition is applied to the matrix $Z_{x,t}$ to decompose itself into the product of three other matrices:

$$SVD\ (Z) = d_1 P_{x1} Q_{t1} + d_2 P_{x2} Q_{t2} + d_3 P_{x3} Q_{t3} \dots + d_X P_{xX} Q_{tX}$$

   Where d is the age element, Q is the Singular Value and P is the time component
4. The expected values of $\widehat{k_t}$ is given by $d_1 Q_{t1}$ and $\widehat{b_x} = P_{x1}$
5. Estimation of new matrix $\widehat{Z_{x,t}}$ is obtained through the product of $\widehat{k_t}$ and $\widehat{b_x}$ to get $\widehat{Z_{x,t}} = \widehat{b_x}\,\widehat{k_t}$
6. The natural logarithm is then computed $\ln(m_{x,t}) = \widehat{a_x} + \widehat{Z_{x,t}} = \widehat{a_x} + \widehat{b_x}\,\widehat{k_t}$

Using the SVD method, the mean values of the parameters denoted as $\widehat{a_x}$ and $\widehat{b_x}$ were estimated for all the ages from 0 to 85+ for male, female and total population. The outputs of the values are provided in the appendix.

Figure 4.25 was obtained whereby it can be observed that the Lee-Carter model with SVD fitted the mortality data effectively.

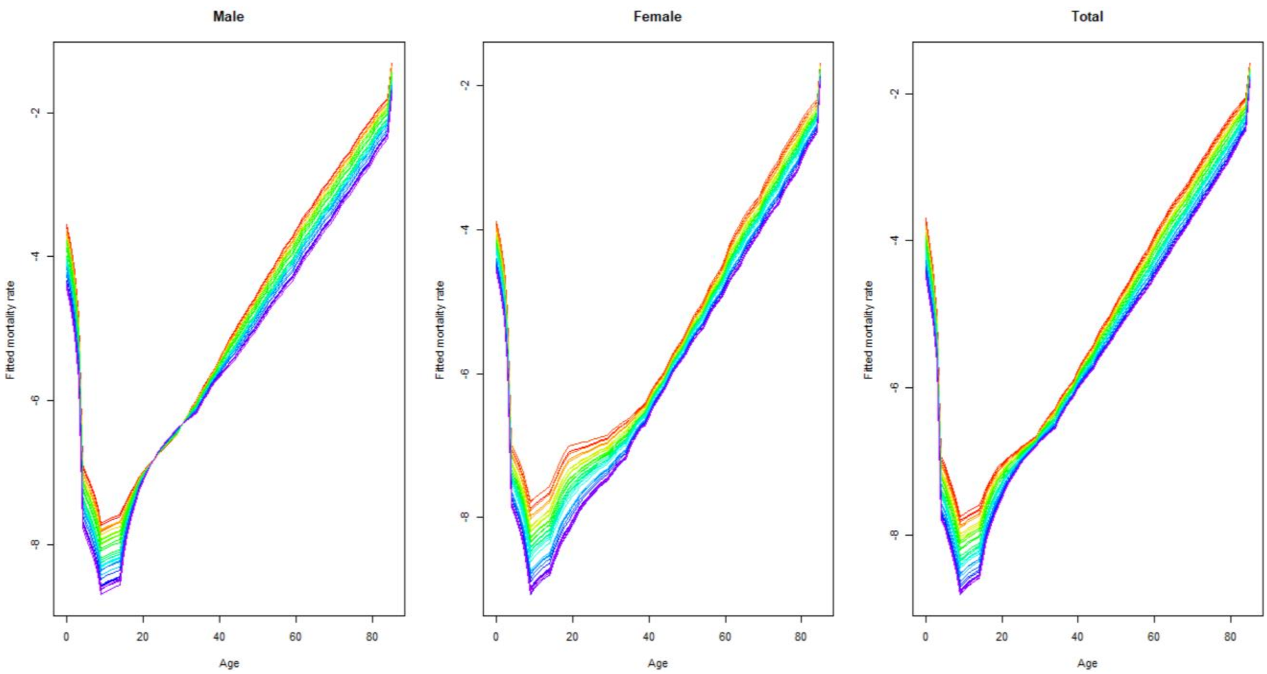

Figure 4.25: Lee-Carter model with SVD

Table 4.5: Percentage Variation Explained by model using SVD

Lee-Carter Analysis for:	**Percentage Variation explained**
Male	73.9%
Female	72%
Total	79%

The Table 4.5 shows that the Lee-Carter with SVD fitted the male mortality data better than the female one with a higher percentage value of 73.9%.

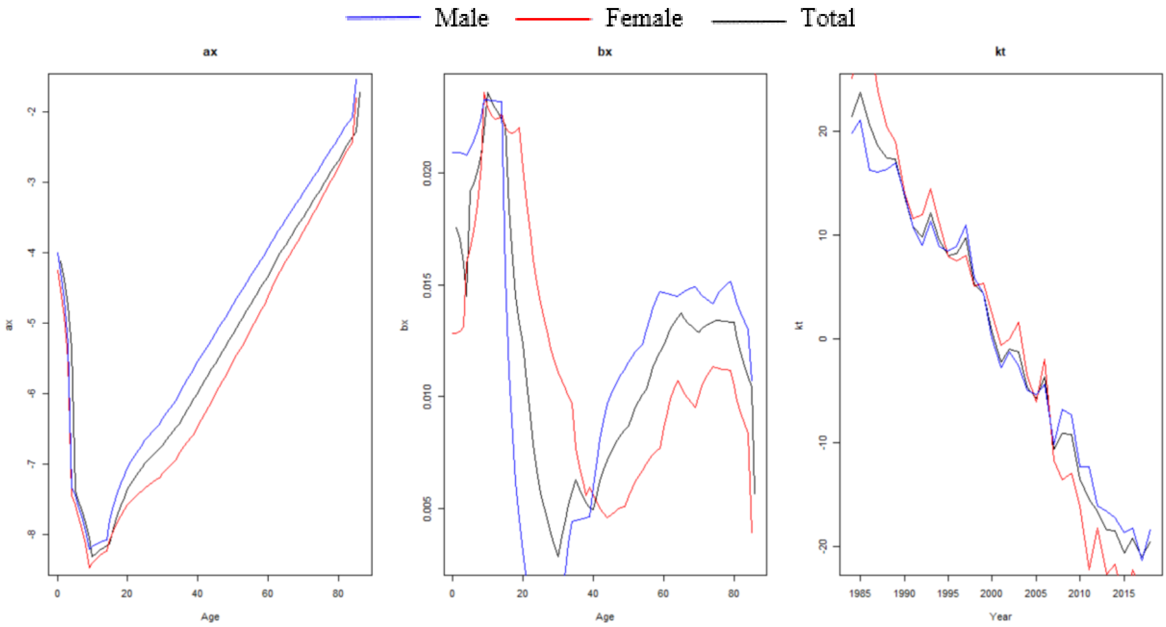

Figure 4.26: Parameter estimates of the Lee-Carter model using SVD

The Figure 4.26 exhibits the plots of each parameter of the Lee-Carter model estimated using the Singular Value Decomposition technique. The a_x parameter usually describes the general shape in the age-specific death rates i.e. the average mortality observed as the age increases. In addition to this, it can be seen that the values of a_x for male are higher than those of female, which means that males are more inclined to die earlier than female. The parameter b_x outlines the tendency of death at age x to vary when the overall level of mortality i.e. k_t changes. For instance, when the value of b_x is large, the mortality rate at that age x alters considerably when the value of k_t changes. Furthermore, the Table 4.6 summarizes the measures of error of the goodness of fit of the Lee-Carter model with SVD for male, female and total population.

Table 4.6: Measures of errors of the model goodness of fit using SVD

Model Goodness of fit for:	Measures of Error
Male	ME = 0.00066 MSE = 0.01141 MAPE = 0.01376
Female	ME = 0.00020 MSE = 0.01940 MAPE = 0.01583
Total	ME = 0.00013 MSE = 0.00886 MAPE = 0.01120

4.4.3 Lee-Carter and Non-Negative Matrix Factorization

The Non-Negative Matrix Factorization method is commonly applied to decompose non-negative matrices which can also be used as a substitute for Singular Value Decomposition technique used traditionally to estimate the parameters of the Lee-Carter model as discussed in Section 4.4.2. The NMF matrix decomposition method caters only for positive values and can work well with missing values as compared to Singular Value Decomposition (SVD). This section focuses in using the NMF to fit the Lee-Carter model to the Mauritian mortality data with the aid of the NMF package.

R Codes

```
#Fitting Lee-Carter with NMF
library(NMF)
demogdatanmf<-nmfObject(demogdata)
demogdatanmf

NMFLcaM<-lca(demogdatanmf,series="Male",max.age=85)
NMFLcaF<-lca(demogdatanmf,series="Female",max.age=85)
NMFLcaT<-lca(demogdatanmf,series="Total",max.age=85)

par(mfrow=c(1,3))
plot(fitted(NMFLcaM),main="Male")
plot(fitted(NMFLcaF),main="Female")
plot(fitted(NMFLcaT),main="Total")
```

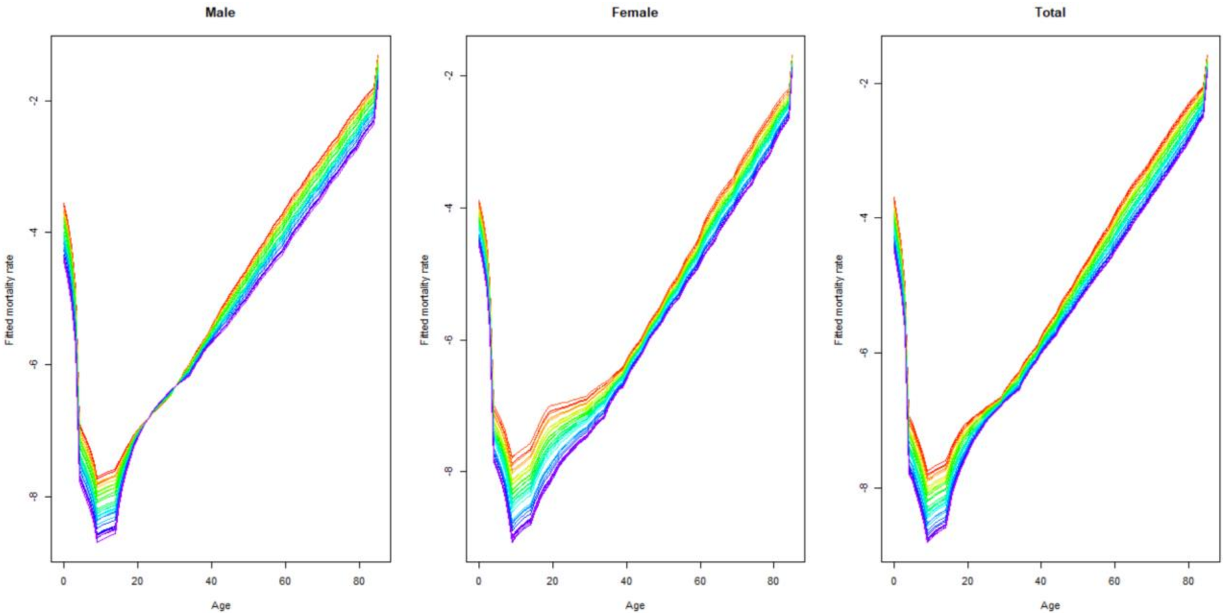

Figure 4.27: Lee-Carter with NMF

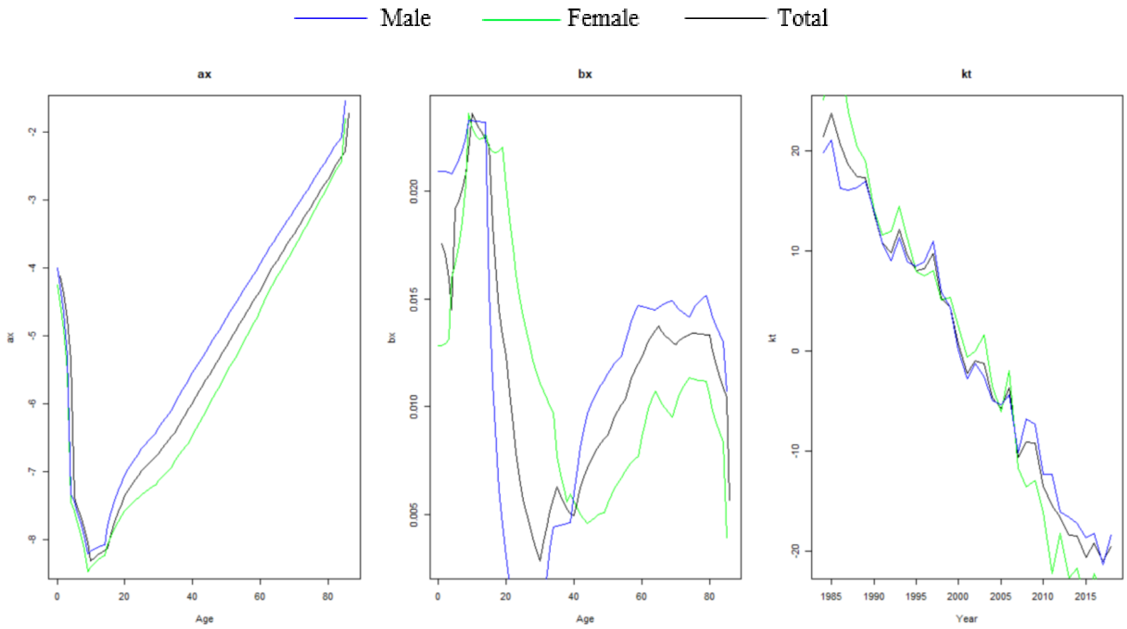

Figure 4.28: Parameter estimates of the Lee-Carter model using NMF

Table 4.7: Percentage Variation Explained by model using NMF

Lee-Carter Analysis for:	**Percentage Variation explained**
Male	71.3%
Female	70.8%
Total	77.2%

Table 4.8: Measures of errors of the model goodness of fit using NMF

Model Goodness of fit for:	**Measures of Error**
Male	ME = 0.00062 MSE = 0.01362 MAPE = 0.01330
Female	ME = -0.00024 MSE = 0.02426 MAPE = 0.01668
Total	ME = -0.00003 MSE = 0.01083 MAPE = 0.01124

The results from the Table 4.6 and Table 4.8 show that the NMF performed weakly as compared to SVD with lower percentage variation explained and higher measures of error. The SVD method is found to be the best technique to be used in estimating the parameters, which will be chosen to proceed further in predicting the parameter k_t .

4.4.4 ARIMA to forecast k_t

This section focuses on applying the ARIMA model to predict the mortality index parameter using data from 1984 to 2018 since existing research performed by Woodun, Ho and Raja (2019) fitted the ARIMA model only to death data from 1984 to 2016 only. The results will then be compared to the LSTM model, to be discussed in the next chapter.

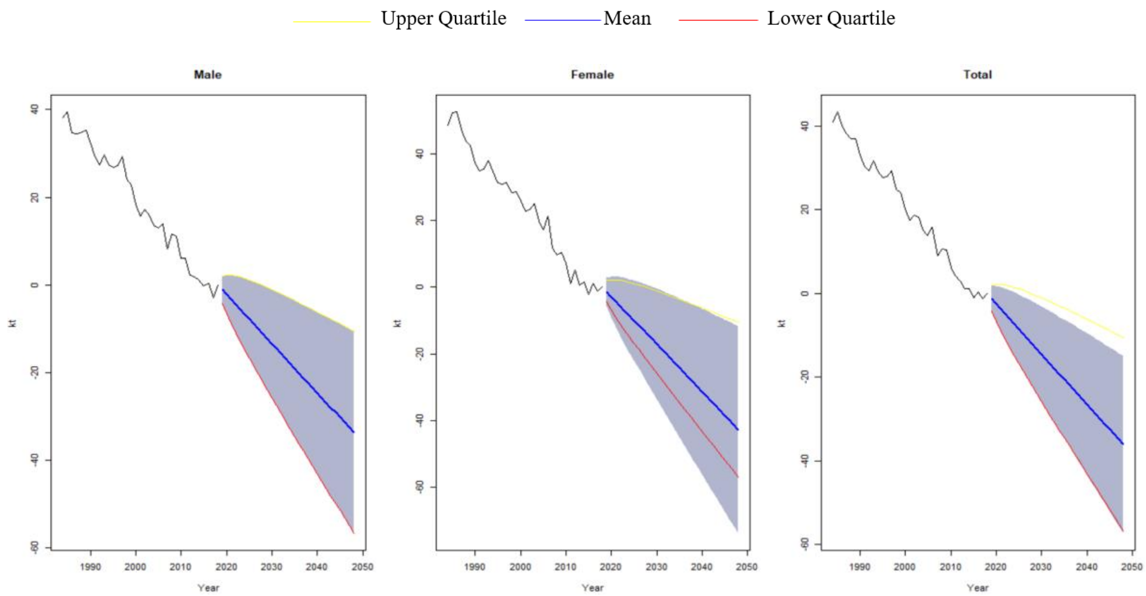

Figure 4.29: Projected values of kt using ARIMA(0,1,0)

Figure 4.29 was plotted using the forecasted values for the parameter k_t for the next 30 years as summarized in the Table 4.9.

Table 4.9: Forecasted kt values using ARIMA

	kt		
Year	**Male**	**Female**	**Total**
2019	-1.123	-1.423	-1.205
2020	-2.247	-2.845	-2.410
2021	-3.369	-4.267	-3.616
2022	-4.493	-5.690	-4.822
2023	-5.617	-7.113	-6.027
2024	-6.739	-8.535	-7.233
2025	-7.863	-9.958	-8.438
2026	-8.986	-11.380	-9.643
2027	-10.109	-12.803	-10.849
2028	-11.233	-14.225	-12.054
2029	-12.356	-15.648	-13.259
2030	-13.479	-17.070	-14.465
2031	-14.603	-18.493	-15.670
2032	-15.726	-19.916	-16.876
2033	-16.849	-21.338	-18.081
2034	-17.973	-22.761	-19.287
2035	-19.096	-24.183	-20.492
2036	-20.219	-25.606	-21.697
2037	-21.343	-27.028	-22.902
2038	-22.466	-28.451	-24.108
2039	-23.589	-29.874	-25.313
2040	-24.713	-31.296	-26.519
2041	-25.836	-32.719	-27.725
2042	-26.959	-34.141	-28.930
2043	-28.082	-35.564	-30.135
2044	-29.206	-36.986	-31.340
2045	-30.329	-38.409	-32.546
2046	-31.452	-39.831	-33.751
2047	-32.576	-41.254	-34.957
2048	-33.699	-42.676	-36.163

Table 4.9 shows that in the next 30 years, there will be enormous improvement in the life expectancy of the Mauritian population whereby it can be seen from the Figure 4.29 that the mortality rate is continuously decreasing for the coming 30 years. This impending reduction in death rate raises the apprehension of insurance companies and pension providers where the latter will have to charge higher premiums and prices for pension annuities. As mentioned in Section 3.2.5, model performance evaluation tools namely AIC, BIC and MSE will be used to compare between the models. As such, the Table 4.10 summarises the mentioned metrics.

Table 4.10: Performance Evaluation Metrics for ARIMA model

Performance Evaluation Tools	Male	Female	Total
MSE	0.01141	0.01940	0.00886
AIC	-374.23	-329.11	-395.73
BIC	-366.90	-321.78	-388.40

Having low values of MSE signifies that the mortality rates of the Mauritian population had almost insignificant outliers. It can also be perceived that the actual mortality rates are approximately equivalent to the fitted values. In addition to this, the AIC and BIC values tend to the negative infinity, owing to the fact that the MSE values are approaching zero.

4.4.5 Long Short-Term Memory to forecast $\boldsymbol{k_t}$

As discussed in Section 3.2.4, the RNN-LSTM will be used to forecast the mortality index parameter, whereby the results will be compared to those of the ARIMA model, after which the most suitable model will be used to forecast the future death rates. The following equation describes the LSTM model to be implemented.

$$\boldsymbol{k_t = f\left(k_{t-1}, k_{t-2}, \dots k_{t-J}\right) + \varepsilon_t} \quad \textbf{[23]}$$

Where $J \in \mathbb{N}$ is the number of time lags considered and $\boldsymbol{\varepsilon_t}$ is the homoschedastic (equal variance) error term. The following steps are considered in order to implement the LSTM model.

1. To use the estimated values of a_x and b_x as obtained in Section 3.7.2 i.e. using the SVD method.
2. The data is split into training and test according to the 60% and 40% rule respectively, as in most research papers this was found to be the ideal split ratio (Nigri et al., 2019; Deprez et al., 2017).
3. In order to select the best architecture for the Recurrent Neural Network with LSTM, a preliminary round of fine-tuning is carried out as shown in Section 4.4.5.2.

4.4.5.1 Data Partition

Using the 60% and 40% split ratio, the training set comprises of observations from year 1984 to 2004 while the test set consists of those from year 2005 to 2018. Thus, making a set of training with 1806 observations and test set with 1204.

4.4.5.2 Fine-Tuning

In the aim of determining the best RNN-LSTM model to forecast the values of the mortality index parameter k_t, the following experiments were performed to determine the best combination of hyperparameters for the network, where during training of model, a validation split of 0.2 was used since most researchers adopted similar value (Nigri et al., 2019; Deprez et al., 2017). After determining the best model, the test set will be used to predict the mortality index parameter.

1) Male taken as target variable

Experiment 1 : Optimizer= ‘rmsprop’, Epochs=50, activation = ‘linear’, 2 LSTM Hidden layers.

```
#50 Epochs
model <- keras_model_sequential()
model %>%
  layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
  layer_lstm(units = 1806, return_sequences = F) %>%
  # using linear activation on last layer, as output is needed in real number
  layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'rmsprop',metrics='accuracy')
summary(model)

model %>% fit(train_reshape, traintarget, epochs=50,validation_split=0.2)

plot(model)
```

```
Model: "sequential"
________________________________________________________________________________
Layer (type)                     Output Shape                  Param #
================================================================================
lstm (LSTM)                      (None, 1, 1806)               13068216
________________________________________________________________________________
lstm_1 (LSTM)                    (None, 1806)                  26100312
________________________________________________________________________________
dense (Dense)                    (None, 1)                     1807
================================================================================
Total params: 39,170,335
Trainable params: 39,170,335
Non-trainable params: 0
```

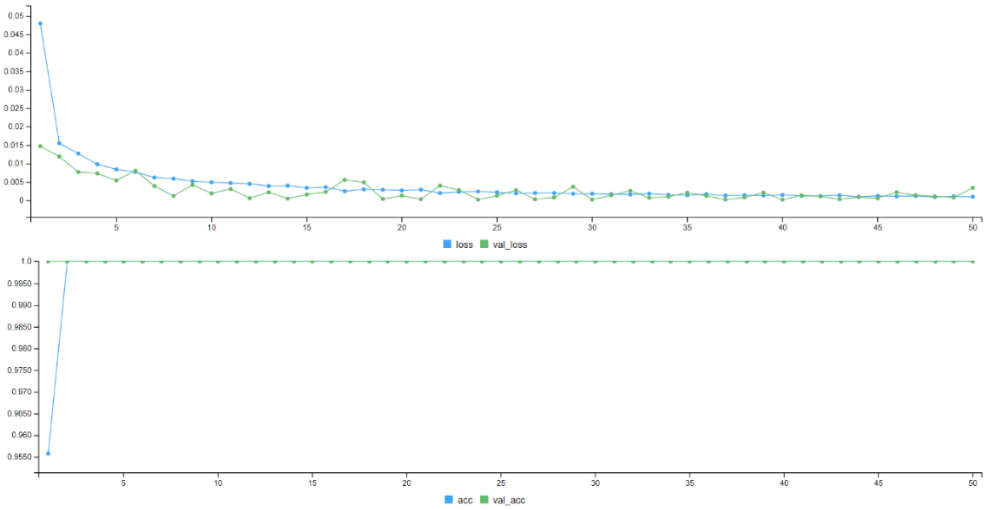

Figure 4.30: Model plot for 50 epochs for male

Using the Linear activation function together with the Root Mean Square Propagation (RMSProp) and training the model at 50 epochs, an MSE of 0.0005 was obtained on the validation set, implying that the model learnt both the training as well as the validation set effectively well. The Linear activation function was used owing to the fact that the target variable is a continuous one.

<u>Experiment 2 :</u> Optimizer= 'rmsprop', Epochs=100, activation = 'linear', 2 LSTM Hidden layers.

```
#100 Epochs
model <- keras_model_sequential()
model %>%
  layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
  layer_lstm(units = 1806, return_sequences = F) %>%
  # using linear activation on last layer, as output is needed in real number
  layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'rmsprop',metrics='accuracy')
summary(model)

model %>% fit(train_reshape, traintarget, epochs=100,validation_split=0.2)

plot(model)
```

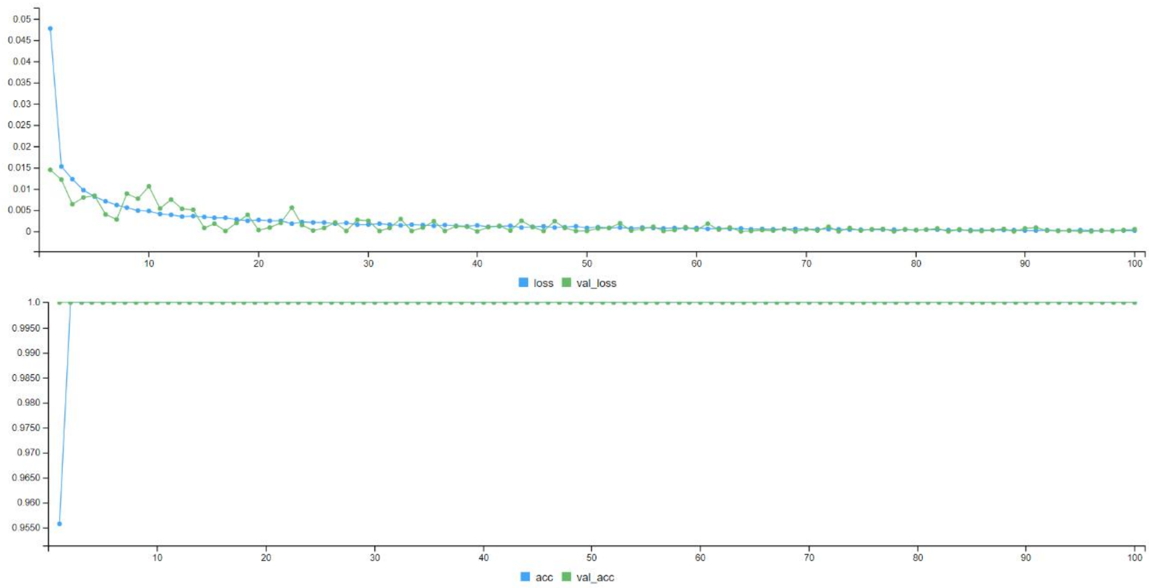

Figure 4.31: Model plot for 100 epochs for male

Figure 4.31 shows that increasing the epochs to 100 did not change the MSE value (0.0005). Thus, the number of epochs does not affect the loss function of any network. However, in order to investigate whether changing the optimizer which helps to reduce the loss function, really impacts the neural network model. As such, the rmsprop optimizer was altered to adam one as shown in experiment 3.

Experiment 3 : Optimizer= 'adam', Epochs=50, activation = 'linear', 2 LSTM Hidden layers.

```
#Double LSTM Hidden Layers with adam optimizer
model <- keras_model_sequential()
model %>%
  layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
  layer_lstm(units = 1806, return_sequences = F) %>%
  # using linear activation on last layer, as output is needed in real number
  layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(train_reshape, traintarget, epochs=50,validation_split=0.2)

plot(model)
```

```
Model: "sequential_1"
________________________________________________________________________________
Layer (type)                     Output Shape                      Param #
================================================================================
lstm_2 (LSTM)                    (None, 1, 1806)                   13068216
________________________________________________________________________________
lstm_3 (LSTM)                    (None, 1806)                      26100312
________________________________________________________________________________
dense_1 (Dense)                  (None, 1)                         1807
================================================================================
Total params: 39,170,335
Trainable params: 39,170,335
Non-trainable params: 0
```

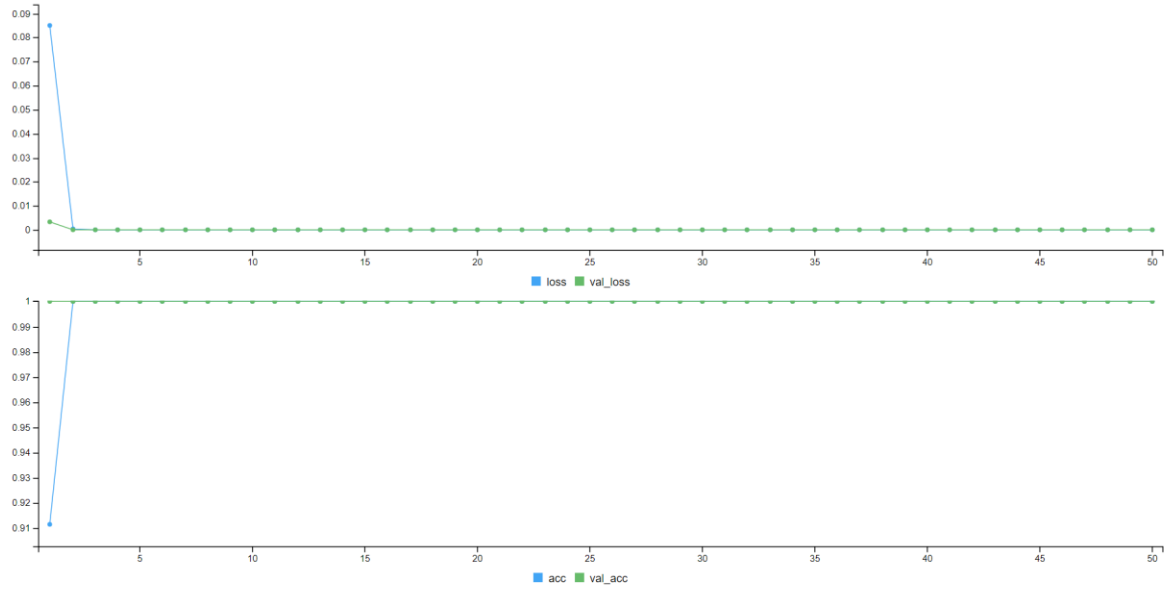

Figure 4.32: Model plot using Adam optimizer for male

With the aid of the Adaptive Moment Estimation (Adam) optimizer, the MSE value reduced further to 0 as compared to that obtained when applying the RMSProp optimizer i.e. 0.005. Having the MSE loss function approaching 0, implies that the model fits both training and validation sets very well.

2) Female taken as target variable

```
#Female as target
train<-training[,(1:4)]
test_set<-test[,(1:4)]

training_reshape<-array(dim=c(1806,1,2))
testing_reshape<-array(dim=c(1204,1,2))
dim(training_reshape)
trainingtarget<-training$Female
trainingtarget<-array(dim=c(1806))
dim(trainingtarget)
```

Experiment 1 : Optimizer= ‘rmsprop’, Epochs=50, activation = ‘linear’, 2 LSTM Hidden layers.

```
#Double LSTM Hidden Layers with rmsprop optimizer
model <- keras_model_sequential()
model %>%
  layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
  layer_lstm(units = 1806, return_sequences = F) %>%
  # using linear activation on last layer, as output is needed in real number
  layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'rmsprop', metrics='accuracy')
summary(model)

model %>% fit(training_reshape, trainingtarget, epochs=50,validation_split=0.2)

plot(model)
```

```
Model: "sequential_2"
________________________________________________________________________________
Layer (type)                        Output Shape                     Param #
================================================================================
lstm_4 (LSTM)                       (None, 1, 1806)                  13068216
________________________________________________________________________________
lstm_5 (LSTM)                       (None, 1806)                     26100312
________________________________________________________________________________
dense_2 (Dense)                     (None, 1)                        1807
================================================================================
Total params: 39,170,335
Trainable params: 39,170,335
Non-trainable params: 0
```

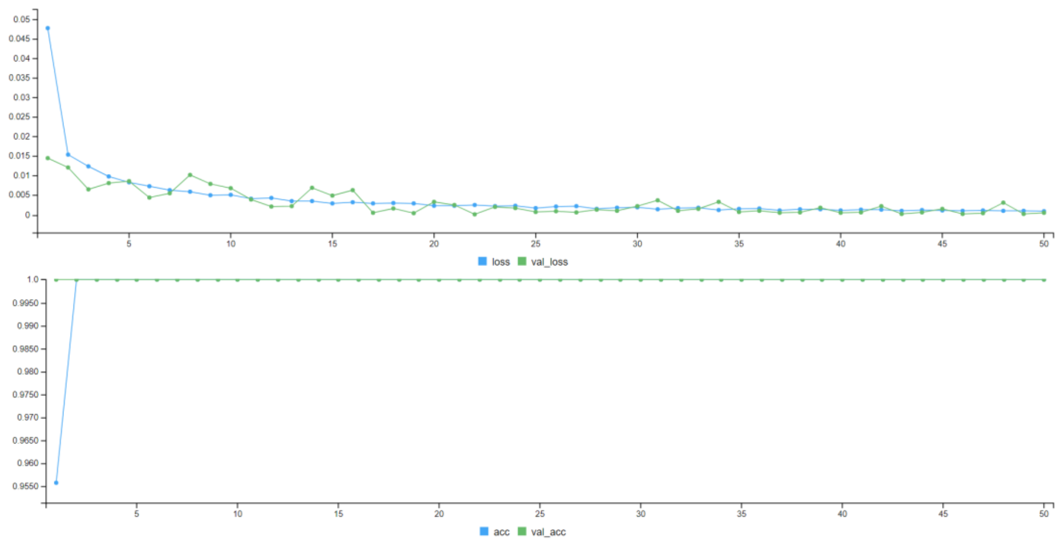

Figure 4.33: Model plot using rmsprop optimizer for female at 50 epochs

From the Figure 4.33, it can be observed that the loss function MSE of the training set decreased gradually from 0.0048 to 0.0002 while that of the validation set dropped from 0.0149 to 0.0002 when 50 epochs was applied. In addition to this, it can be deduced that the model fits the female dataset better than the male death data since a lower MSE of 0.0002 was obtained as compared to that of male (MSE=0.0005). Moreover, to investigate the effect of optimizer on the female data, the following experiment was conducted.

<u>Experiment 2 :</u> Optimizer= ‘adam’, Epochs=50, activation = ‘linear’, 2 LSTM Hidden layers.

```
#Double LSTM Hidden Layers with adam optimizer
model <- keras_model_sequential()
model %>%
  layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
  layer_lstm(units = 1806, return_sequences = F) %>%
  # using linear activation on last layer, as output is needed in real number
  layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(training_reshape, trainingtarget, epochs=50,validation_split=0.2)

plot(model)
```

```
Model: "sequential"
________________________________________________________________________________
Layer (type)                    Output Shape                    Param #
================================================================================
lstm (LSTM)                     (None, 1, 1806)                 13068216
________________________________________________________________________________
lstm_1 (LSTM)                   (None, 1806)                    26100312
________________________________________________________________________________
dense (Dense)                   (None, 1)                       1807
================================================================================
Total params: 39,170,335
Trainable params: 39,170,335
Non-trainable params: 0
```

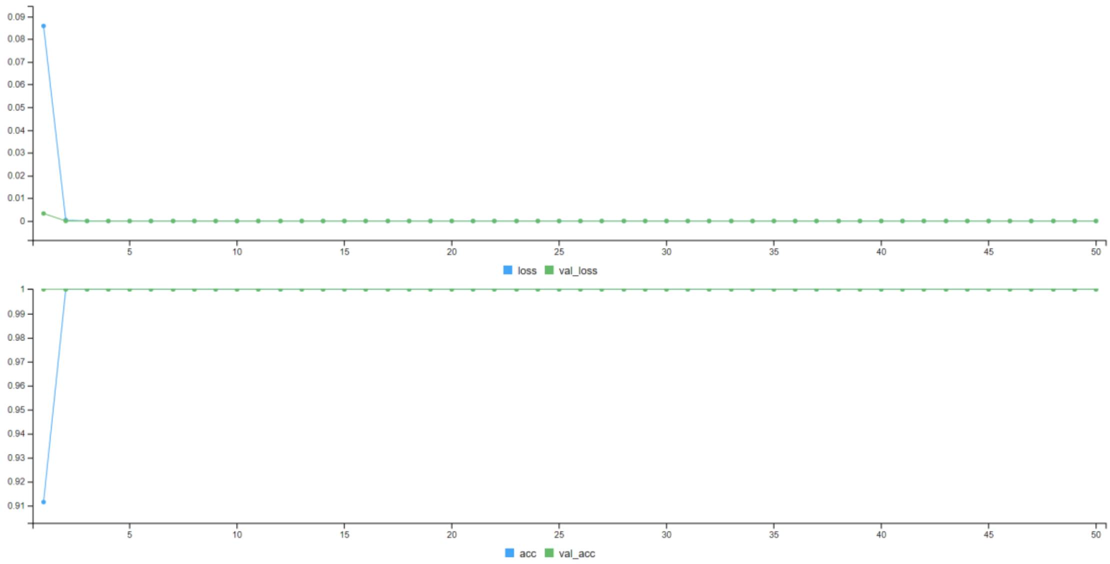

Figure 4.34: Model plot using adam optimizer for female at 50 epochs

Figure 4.34 portrays that changing the optimizer from ‘rsmprop’ to ‘adam’, in fact reduced the MSE value of the validation set to 0 from 0.0002.

3) <u>Total taken as target variable</u>

```
#Total as target
trainT<-training[,(1:5)]
test_setT<-test[,(1:5)]
train_set_reshapeT<-array(dim=c(1806,1,2))
test_set_reshapeT<-array(dim=c(1204,1,2))
dim(train_set_reshapeT)
train_settargetT<-training$Total
train_settargetT<-array(dim=c(1806))
dim(train_settargetT)
```

<u>Experiment 1 :</u> Optimizer= ‘rmsprop’, Epochs=50, activation = ‘linear’, 2 LSTM Hidden layers.

```
#Double LSTM Hidden Layers with rmsprop optimizer
model <- keras_model_sequential()
model %>%
  layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
  layer_lstm(units = 1806, return_sequences = F) %>%
  # using linear activation on last layer, as output is needed in real number
  layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'rmsprop', metrics='accuracy')
summary(model)

model %>% fit(train_set_reshape, train_settarget, epochs=50,validation_split=0.2)

plot(model)
```

```
Model: "sequential"
________________________________________________________________________________
Layer (type)                        Output Shape                      Param #
================================================================================
lstm (LSTM)                         (None, 1, 1806)                   13068216
________________________________________________________________________________
lstm_1 (LSTM)                       (None, 1806)                      26100312
________________________________________________________________________________
dense (Dense)                       (None, 1)                         1807
================================================================================
Total params: 39,170,335
Trainable params: 39,170,335
Non-trainable params: 0
```

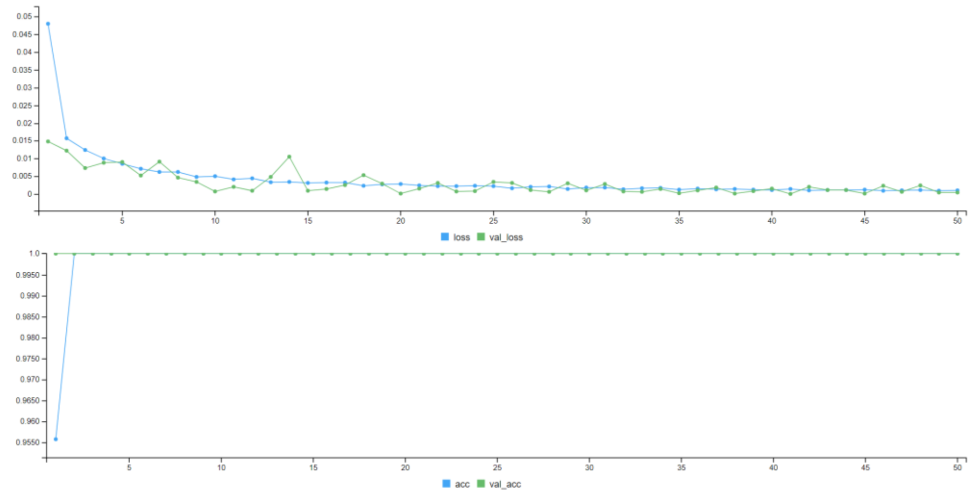

Figure 4.35: Model plot using rmsprop optimizer for total at 50 epochs

Applying the ‘rmsprop’ optimizer as well as the ‘linear’ activation to the total population death data, a value of MSE of 0.0004 was obtained on the validation set. In addition to this, the training set had MSE of 0.001.

Experiment 2 : Optimizer= ‘adam’, Epochs=50, activation = ‘linear’, 2 LSTM Hidden layers.

```
#Double LSTM Hidden Layers with adam optimizer
model <- keras_model_sequential()
model %>%
  layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
  layer_lstm(units = 1806, return_sequences = F) %>%
  # using linear activation on last layer, as output is needed in real number
  layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(train_set_reshape, train_settarget, epochs=50,validation_split=0.2)

plot(model)
```

```
Model: "sequential_1"
________________________________________________________________________________________
Layer (type)                          Output Shape                         Param #
========================================================================================
lstm_2 (LSTM)                         (None, 1, 1806)                      13068216
________________________________________________________________________________________
lstm_3 (LSTM)                         (None, 1806)                         26100312
________________________________________________________________________________________
dense_1 (Dense)                       (None, 1)                            1807
========================================================================================
Total params: 39,170,335
Trainable params: 39,170,335
Non-trainable params: 0
```

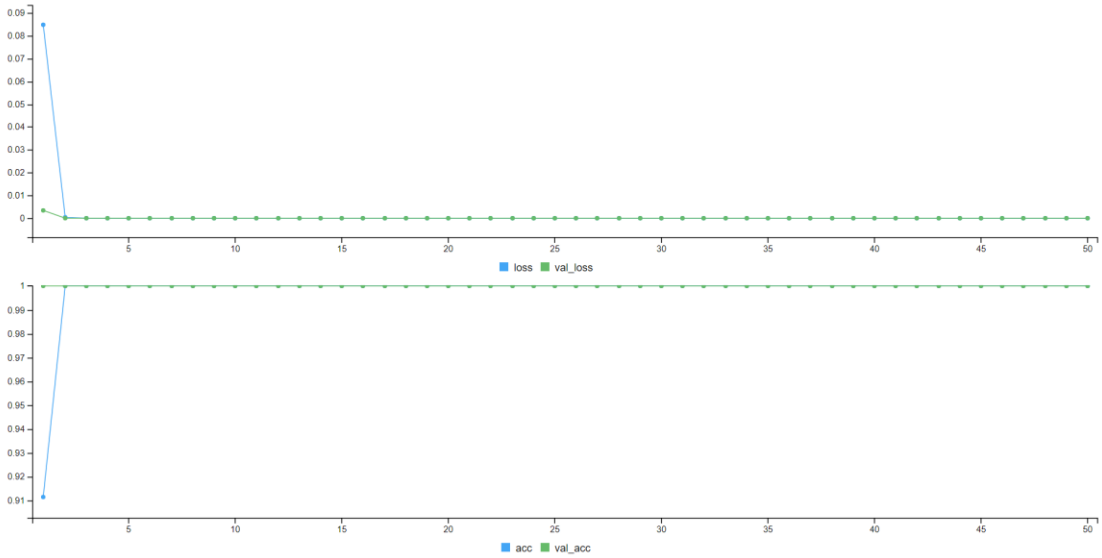

Figure 4.36: Model plot using adam optimizer for total at 50 epochs

The validation set gave a value of MSE of 0 while that of the training set decreased gradually from 0.0848 to 0, implying that the model fitted the mortality data effectively well.

Nigri et al. (2019) had found that using one LSTM hidden layer together with the Rectified Linear Unit (ReLU) activation and 'adam' optimizer proved to be the best model. Therefore, in aim of determining the best LSTM model, the following experiments were also performed.

1) <u>Male as target variable</u>

<u>Optimizer= 'adam', Activation= 'ReLU', 1 LSTM Hidden Layer</u>

```
#Double LSTM Hidden Layers with relu activation
model <- keras_model_sequential()
model %>%
  layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T, activation = "relu" ) %>%
  #layer_lstm(units = 1806, return_sequences = F) %>%
  # using linear activation on last layer, as output is needed in real number
  layer_dense(units = 1)

model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(train_reshape, traintarget, epochs=50,validation_split=0.2)

plot(model)
```

```
Model: "sequential_3"
________________________________________________________________________________
Layer (type)                        Output Shape                      Param #
================================================================================
lstm_5 (LSTM)                       (None, 1, 1806)                   13068216
________________________________________________________________________________
dense_3 (Dense)                     (None, 1, 1)                      1807
================================================================================
Total params: 13,070,023
Trainable params: 13,070,023
Non-trainable params: 0
```

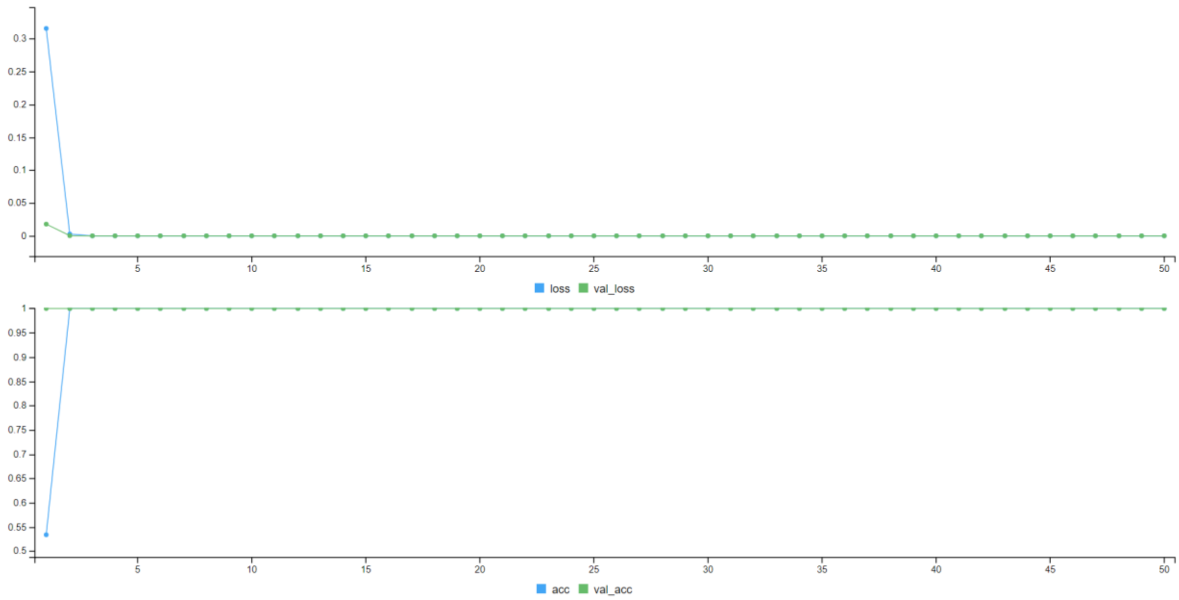

Figure 4.37: Model plot using ReLU optimizer for male at 50 epochs

The Figure 4.37 shows that similar results were obtained as when two LSTM hidden layers were applied. In other words, both validation and training sets gave a value of 0 for the loss function MSE. The loss function of validation decreased from 0.0178 to 0 while that of the training set dropped from 0.3152 to 0.

2) Female as target variable

Optimizer= ‘adam’, Activation= ‘ReLU’, 1 LSTM Hidden Layer

```
model <- keras_model_sequential()
model %>%
  layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T, activation = "relu" ) %>%
  #layer_lstm(units = 1806, return_sequences = F) %>%
  # using linear activation on last layer, as output is needed in real number
  layer_dense(units = 1)

model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(training_reshape, trainingtarget, epochs=50,validation_split=0.2)

plot(model)
```

```
Model: "sequential_4"
________________________________________________________________________________________________
Layer (type)                          Output Shape                             Param #
================================================================================================
lstm_6 (LSTM)                         (None, 1, 1806)                          13068216
________________________________________________________________________________________________
dense_4 (Dense)                       (None, 1, 1)                             1807
================================================================================================
Total params: 13,070,023
Trainable params: 13,070,023
Non-trainable params: 0
```

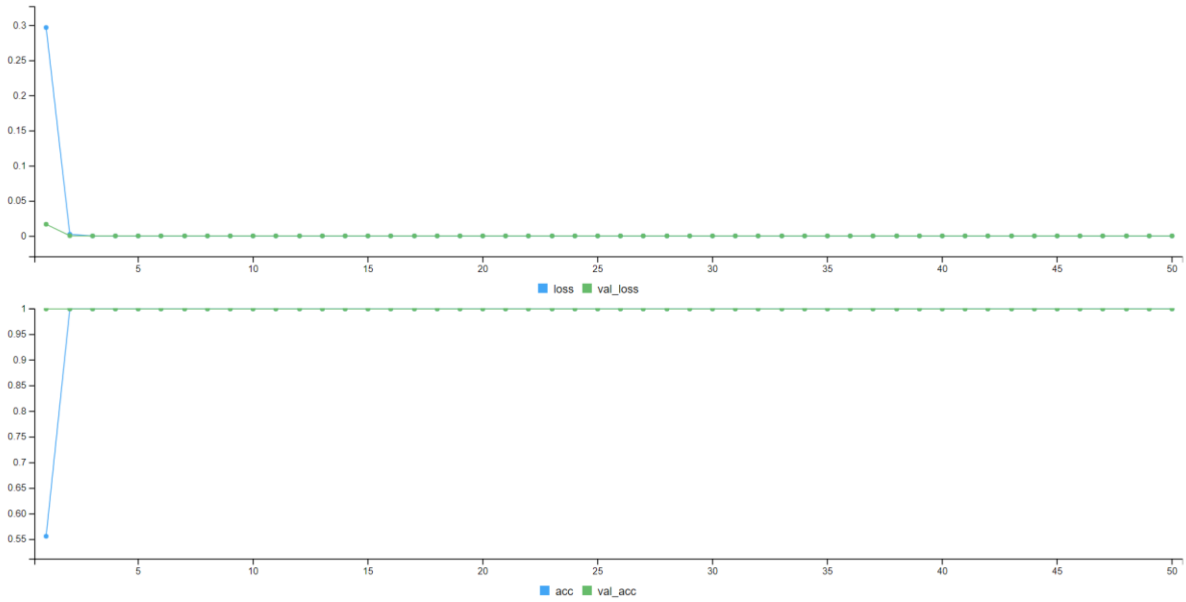

Figure 4.38: Model plot using ReLU optimizer for female at 50 epochs

Figure 4.38 illustrates that the training set had a decreasing MSE loss function value from 0.2971 to 0, while that of the validation set dropped from 0.0166 to 0 after 50 epochs. But, when looking at the 1st epoch, the validation loss function value was 0.0023, with a training set accuracy of 55.57%.

3) <u>Total as target variable</u>

<u>Optimizer= 'adam', Activation= 'ReLU', 1 LSTM Hidden Layer</u>

```
#Double LSTM Hidden Layers with relu activation
model <- keras_model_sequential()
model %>%
  layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T, activation = "relu" ) %>%
  #layer_lstm(units = 1806, return_sequences = F) %>%
  # using linear activation on last layer, as output is needed in real number
  layer_dense(units = 1)

model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(train_set_reshapeT, train_settargetT, epochs=50,validation_split=0.2)

plot(model)
```

```
Model: "sequential_5"
________________________________________________________________________________________________
Layer (type)                          Output Shape                          Param #
================================================================================================
lstm_7 (LSTM)                         (None, 1, 1806)                       13068216
________________________________________________________________________________________________
dense_5 (Dense)                       (None, 1, 1)                          1807
================================================================================================
Total params: 13,070,023
Trainable params: 13,070,023
Non-trainable params: 0
```

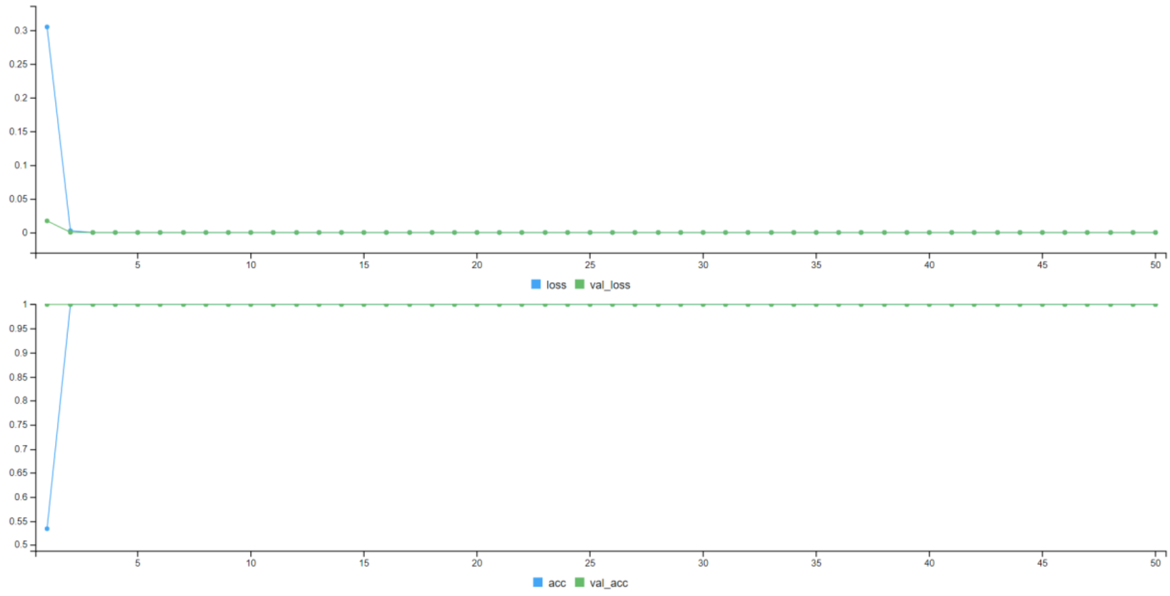

Figure 4.39: Model plot using ReLU optimizer for total at 50 epochs

Figure 4.39 portrays that the loss function MSE of the training set reduced to 0 from 0.3050 and that of the validation decreased from 0.0172 to 0. At the 1st epoch, the training set had an accuracy of 53.35%, implying that the model fitted the Mauritian mortality data successfully well.

4.4.5.3 Comparison between LSTM models

In view of the experiments, the Table 4.11 summarizes the measures of errors of the LSTM models, whereby the best model will be chosen to forecast the mortality index parameter, k_t.

Table 4.11: Performance Evaluation Metrics for LSTM models

			Measures of Errors		
No	**LSTM Models**	**Target Variable**	**MSE**	**AIC**	**BIC**
1	Optimizer= 'rmsprop', Epochs=50, activation = 'linear', 2 LSTM Hidden layers.	Male	0.0005	-640	-633
		Female	0.0002	-718	-711
		Total	0.0004	-659	-652
2	Optimizer= 'adam', Epochs=50, activation = 'linear', 2 LSTM Hidden layers.	Male	0.0000	-∞	-∞
		Female	0.0000	-∞	-∞
		Total	0.0000	-∞	-∞
3	Optimizer= 'adam', Activation= 'ReLU', 1 LSTM Hidden Layer, Epochs=50	Male	0.0000	-∞	-∞
		Female	0.0000	-∞	-∞
		Total	0.0000	-∞	-∞

The Table 4.11 shows the LSTM model in Experiment 2 performed better where the adam optimizer reduced the MSE loss function significantly. Even though, the model in Experiment 3 also gave low MSE values, where AIC and BIC are approaching negative infinity, the model in Experiment 2 will be used to forecast k_t. This is because as more hidden layers are added, better the performance of the model (Hasim, Andrew and Francoise, 2014). As such, using the LSTM model with adam optimizer, 2 hidden layers and linear activation, the following values of the mean forecasted mortality index parameter, $\widehat{k_t}$ were obtained.

Table 4.12: Forecasted values of $\widehat{k_t}$ using LSTM model

Variables	$\boldsymbol{\widehat{k_t}}$
Male	-17.411
Female	-22.049
Total	-18.684

4.5 Summary

Chapter 4 focused mainly on analysing the datasets. In addition to this, the exploratory data analysis revealed that no missing values and multicollinearity treatment were needed. Furthermore, the spread of the attributes in the datasets were explored and their distributions of the variables were exhibited with the aid of histograms. Moreover, dashboards were created to illustrate interactive visualizations relating to the variables in the datasets. After data analysis, the SVD and NMF methods were compared to determine the best technique to estimate the parameters of a_x and b_x. Additionally, the ARIMA and the RNN-LSTM models were fitted to the mortality data of Mauritius and using the measures of errors such as MSE, AIC and BIC. This chapter discussed mainly the results of the parameters of the Lee-Carter model. However, the next chapter will highlight primarily the forecasted death rates of the Mauritian population for the next 30 years with the aid of the mean of the estimated parameters i.e. $\widehat{a_x}$, $\widehat{b_x}$ and $\widehat{k_t}$.

CHAPTER 5

FINDINGS AND DISCUSSION

5.1 Introduction

This chapter will concentrate highly on forecasting the mortality rates of the Mauritian population using the test sets for the upcoming 30 years using the estimated parameters found in the previous chapter. Additionally, this section will provide in-depth discussion of the main findings of the Chapter 4 and highlight all the results obtained.

5.2 Singular Value Decomposition against Non-Negative Matrix Factorization

As accentuated in the previous chapter, the Lee-Carter model comprises of the following parameters a_x, b_x and k_t and in order to fit the model, the SVD model is usually applied but, in this study the NMF method was compared to the original technique, to determine the best method to estimate these parameters. As such, based on the results obtained as shown in the Table 5.1, it was deduced that the NMF method was insignificant and that the SVD one was still the most effective one. In other words, with the aid of the measures of errors such as MSE, the SVD was found to have produced lower error values.

Table 5.1: Comparison between SVD and NMF

	Measure of Error- Mean Square Error (MSE)	
Model Goodness of fit for:	**SVD**	**NMF**
Male	0.01141	0.01362
Female	0.01940	0.02426
Total	0.00886	0.01083

As discussed earlier that NMF method usually deals effectively with missing values in any dataset. Since the mortality and population data did not have any missing values, the SVD technique turned out to be the most suitable one in order to estimate the parameters of the

Lee-Carter model. The Figure 5.1 portrays a trend analysis of the parameters estimated by SVD.

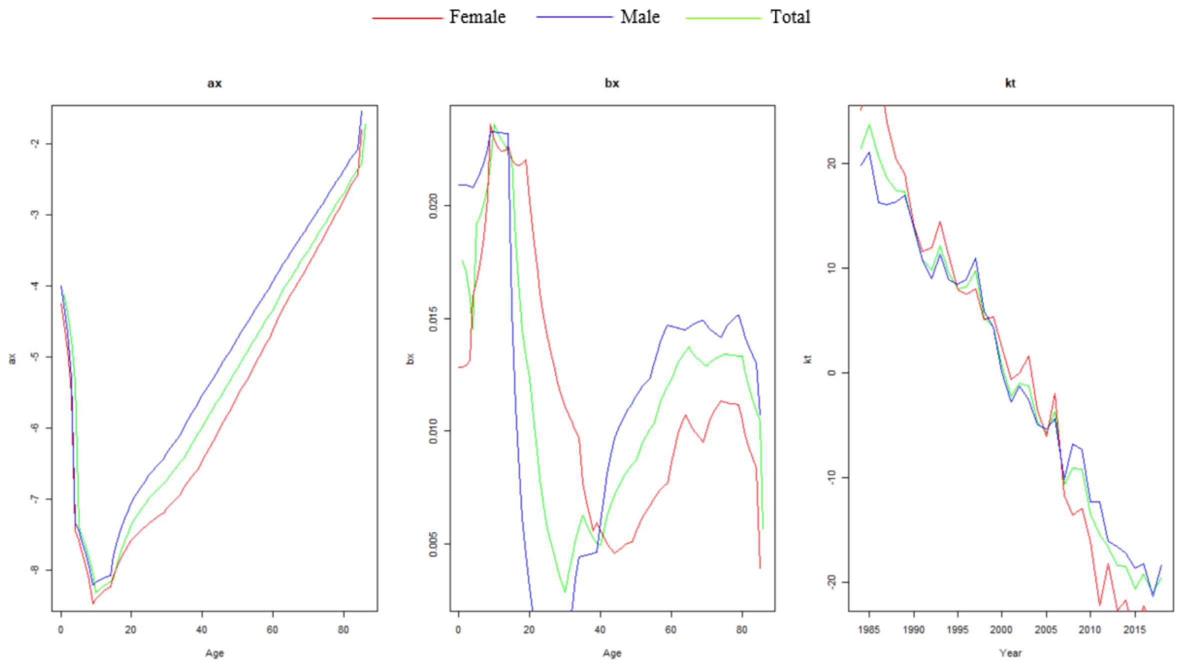

Figure 5.1: Trend analysis of a_x, b_x and k_t

Figure 5.1 shows that there is a decreasing trend in the mortality index parameter, k_t for both male and female. In addition to this, when looking at the graph of a_x, it can be observed that the shape of the curve follows that of the mortality curve. There is a hump at around the age 15 years which indicates that the mortality risk of adolescents increases sharply at this age. This dip is due to being prone to accidents, diseases caused by cigarette and alcohol intake. Furthermore, it can be deduced that males have a high mortality risk than females. Moreover, the graph of the parameter b_x which denotes the response at age x to change in the overall level of mortality over time, portrays that death rate decreases more significantly for females as compared to those of males. It can also be noticed that the values of b_x between the ages 60 and 80 years are higher for males than females, thus showing improvement in the death rates of males within that age range.

5.3 ARIMA versus RNN-LSTM

The main aim of using the RNN-LSTM model was to overcome the shortcomings of the ARIMA model i.e. the inability of the latter to determine the trend in the mortality index parameter k_t. In addition to this, Nigri et al. (2019) highlighted that the LSTM model not only caters effectively for non-linear mortality patterns, but also is able to take into consideration the effect of historical death rate trend and replicate it on the future mortality rate. As such, based on the number of experiments conducted while building the LSTM models, it was found that the model with the linear activation function, adam optimizer as well as 2 LSTM hidden layers was the most suitable one to find the forecasted values of mortality index parameter. However, when comparing against the ARIMA model of order (0,1,0), it was deduced that the LSTM model was better, owing to its low values of measures of errors such as MSE, AIC and BIC, as shown in the Table 5.2.

Table 5.2: Comparison between ARIMA and LSTM

	MSE		AIC		BIC	
Model Goodness of fit for:	**ARIMA**	**LSTM**	**ARIMA**	**LSTM**	**ARIMA**	**LSTM**
Male	0.01141	0	-374.23	-∞	-366.90	-∞
Female	0.01940	0	-329.11	-∞	-321.78	-∞
Total	0.00886	0	-395.73	-∞	-388.40	-∞

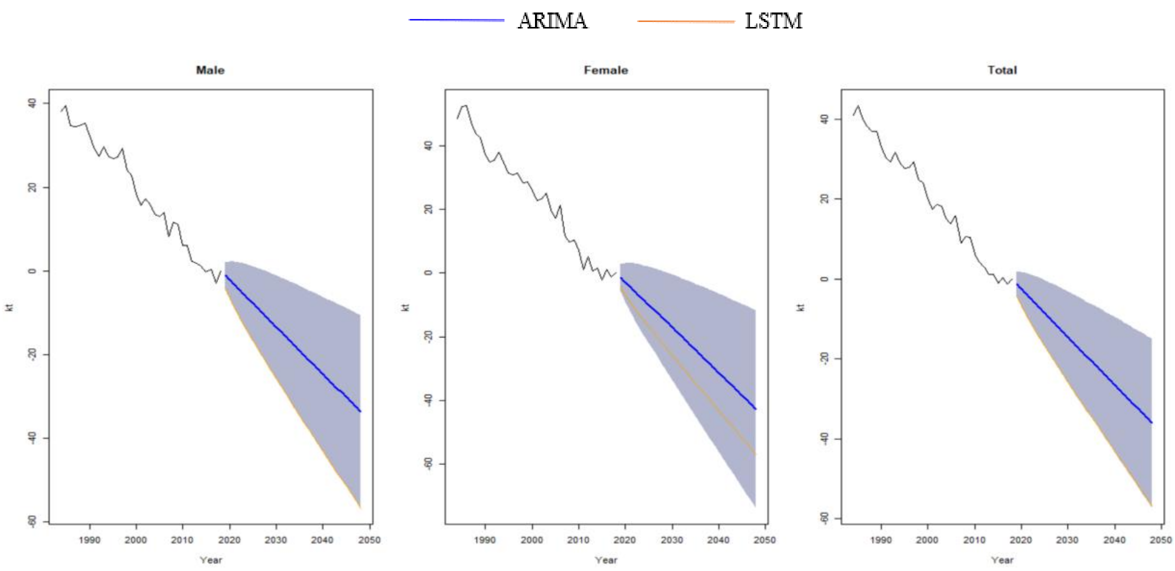

Figure 5.2: Trend analysis of forecasted k_t of both ARIMA and LSTM

Table 5.2 shows clearly that the LSTM is the most suitable one to be used to forecast the values of k_t, whereby the AIC and BIC are approaching negative infinity and the MSE value is 0. Using the LSTM model as well as the projected mortality index parameter values, the Figure 5.2 was plotted. It can be seen that there is a decreasing trend in the mortality index for the next 30 years for both ARIMA and LSTM models. However, it can be observed that the LSTM seems to catch very well the non-linearity of the future mortality trend, thus being the best model to represent the diminishing dynamics of mortality, with respect to the ARIMA model.

5.4 Deep-Learning Integrated Lee-Carter mortality forecasting

This section emphasizes mainly on forecasting the mortality rates of the Mauritian population using the deep-learning integrated Lee-Carter model. In other words, the Lee-Carter model whereby its parameters a_x and b_x were estimated using the Singular Value Decomposition method and the mortality index parameter k_t was projected using the Recurrent Neural Network with the Long Short-Term Memory architecture.

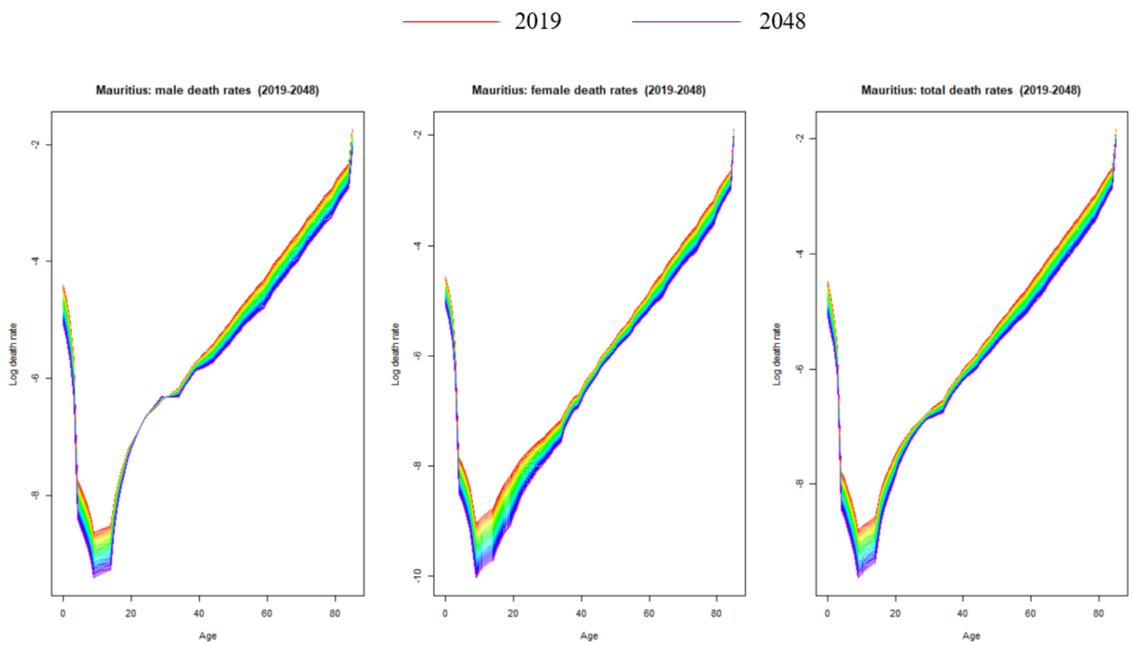

Figure 5.3: Forecasted mortality rates from 2019 to 2048

Figure 5.3 exhibits the forecasted death rates for male, female as well as total population from 2019 to 2048. It can be noticed that the mortality rates for the year 2048 are lower than those of the year 2019, implying significant improvement in the death rate of the Mauritian population. When looking at the male forecasted death rates, it can be observed that the trend is similar for all the years around the age of 25 years to 40 years. In addition to this, the female projected mortality rates illustrate that the patterns in the mortality through the years 2019 to 2048 are dissimilar, except between the age range of 1 year and 5 years, likewise for the total population since the overall population of Mauritius comprises of more females than males. Moreover, it can also be deduced that youths between the ages of 5 and 20 years will experience early death in the forthcoming years. A snapshot of the forecasted death rates are shown in the Table 5.3, Table 5.4 and Table 5.5, while the whole output is attached in the appendix.

Table 5.3: Forecasted mortality rates from 2019 to 2023 for male

Age	2019	2020	2021	2022	2023
0	0.0121905	0.0119077	0.0116315	0.0113616	0.0110981
1	0.0092557	0.0090410	0.0088312	0.0086264	0.0084263
2	0.0063207	0.0061741	0.0060308	0.0058910	0.0057543
3	0.0033851	0.0033066	0.0032299	0.0031550	0.0030818
4	0.0004371	0.0004270	0.0004171	0.0004075	0.0003981
5	0.0003856	0.0003765	0.0003677	0.0003591	0.0003507
6	0.0003338	0.0003258	0.0003181	0.0003105	0.0003031
7	0.0002815	0.0002747	0.0002680	0.0002615	0.0002551
8	0.0002286	0.0002229	0.0002173	0.0002119	0.0002066
9	0.0001743	0.0001698	0.0001654	0.0001612	0.0001570
10	0.0001803	0.0001756	0.0001711	0.0001667	0.0001624

Table 5.4: Forecasted mortality rates from 2019 to 2023 for female

Age	2019	2020	2021	2022	2023
0	0.0103856	0.0101979	0.0100136	0.0098326	0.0096549
1	0.0078928	0.0077498	0.0076094	0.0074714	0.0073360
2	0.0053997	0.0053013	0.0052047	0.0051099	0.0050168
3	0.0029059	0.0028521	0.0027994	0.0027477	0.0026969
4	0.0003954	0.0003865	0.0003778	0.0003693	0.0003610
5	0.0003408	0.0003329	0.0003251	0.0003175	0.0003101
6	0.0002860	0.0002790	0.0002722	0.0002655	0.0002590
7	0.0002310	0.0002250	0.0002191	0.0002134	0.0002078
8	0.0001751	0.0001701	0.0001653	0.0001606	0.0001560
9	0.0001172	0.0001134	0.0001096	0.0001060	0.0001025
10	0.0001275	0.0001234	0.0001194	0.0001156	0.0001119

Table 5.5: Forecasted mortality rates from 2019 to 2023 for total

Age	2019	2020	2021	2022	2023
0	0.0113545	0.0111167	0.0108838	0.0106558	0.0104326
1	0.0087612	0.0085831	0.0084085	0.0082376	0.0080701
2	0.0062152	0.0060968	0.0059806	0.0058667	0.0057550
3	0.0035071	0.0034464	0.0033868	0.0033282	0.0032706
4	0.0004193	0.0004097	0.0004003	0.0003912	0.0003823
5	0.0003663	0.0003578	0.0003495	0.0003413	0.0003334
6	0.0003133	0.0003058	0.0002985	0.0002914	0.0002844
7	0.0002598	0.0002534	0.0002471	0.0002410	0.0002350
8	0.0002059	0.0002006	0.0001953	0.0001903	0.0001853
9	0.0001507	0.0001465	0.0001424	0.0001384	0.0001345
10	0.0001587	0.0001543	0.0001501	0.0001459	0.0001419

Furthermore, in the aim of visualizing the future trend against the historical one of the death rates, the following graphs were plotted while considering the mortality patterns ages 20, 65 and 80 years.

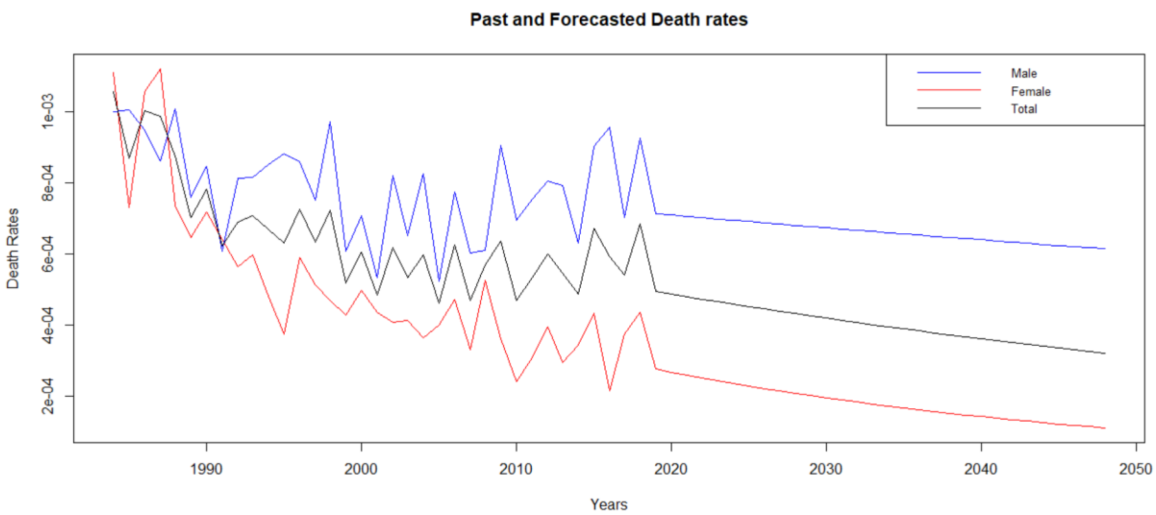

Figure 5.4: Past and Forecasted Death rates for the age 20 years

Figure 5.4 portrays that the mortality rate has been fluctuating for the previous years but as from 2020 onwards, a downward trend can be observed. In addition to this, it can be noticed that the mortality rate for individuals of age 20 years decreases slowly over the next 30 years. Furthermore, it can also be deduced that males tend to have higher death rates than females. When looking at the total population, the death rate pattern is almost similar to that of the females.

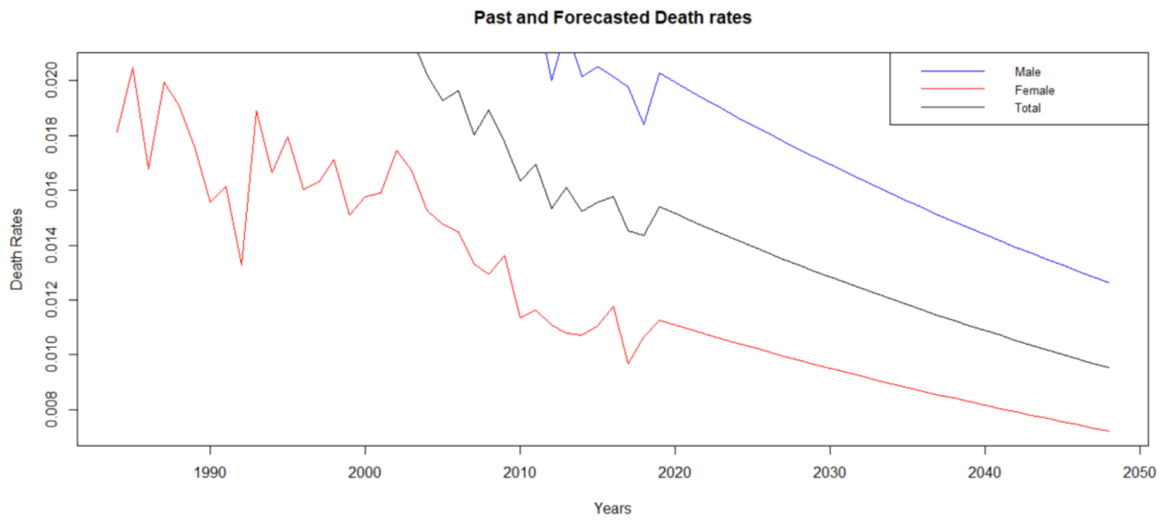

Figure 5.5: Past and Forecasted Death rates for the age 65 years

When considering the projected mortality rates for individuals of age 65 years, it can be perceived that the pace at which the death rate is decreasing, is faster as compared to that of 20 years. Furthermore, the downward trend is shaper than that obtained for the individuals of age 20 years. This implies that people of 65 years have higher probability of dying sooner than 20 years youths. Moreover, males have higher mortality rate than females.

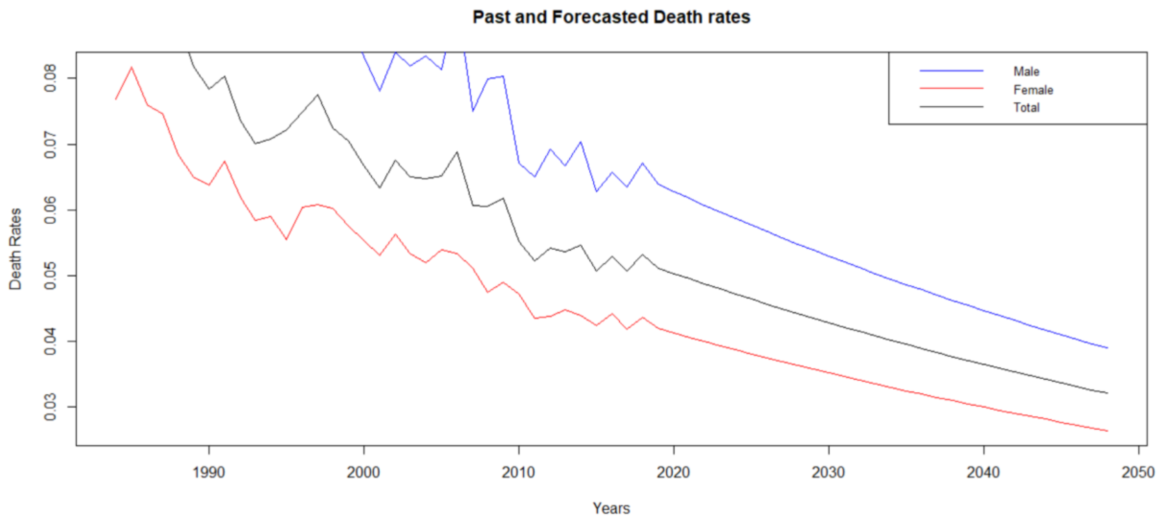

Figure 5.6: Past and Forecasted Death rates for the age 80 years

Similarly, when the historical and forecasted death rates of individuals of 80 years were plotted, the same decreasing trend was seen in the mortality rate, but as compared to people of age 65 years, those who are 80 years are more prone to die faster. Additionally, older males have higher risk of dying than females of age 80 years.

5.5 Evaluation of Forecasted Death Rates

After forecasting the mortality rates of the Mauritian population, it is highly significant that these values are assessed using performance evaluation tools namely MSE, AIC and BIC. As such, the following table summarizes the measures of errors whereby it can be seen that the MSE values are very low, almost close to 0, implying that the deep-learning integrated Lee-Carter model fitted the Mauritian mortality data effectively well. From the AIC and BIC values, it can be perceived that the values are approaching negative infinity, thus showing that the model fits the data properly.

Table 5.6: Performance metrics of forecasted death rates

Performance Evaluation Tools	**Male**	**Female**	**Total**
MSE	0.0000007	0.0000001	0.0000003
AIC	-1202.07	-1353.75	-1273.93
BIC	-1194.74	-1346.42	-1266.60

Additionally, to investigate the goodness of fit of the deep-learning integrated Lee-Carter model proposed in this study, it is also crucial to study the residuals of the fitted model, as shown in the heatmaps of Figure 5.7, Figure 5.8 and Figure 5.9.

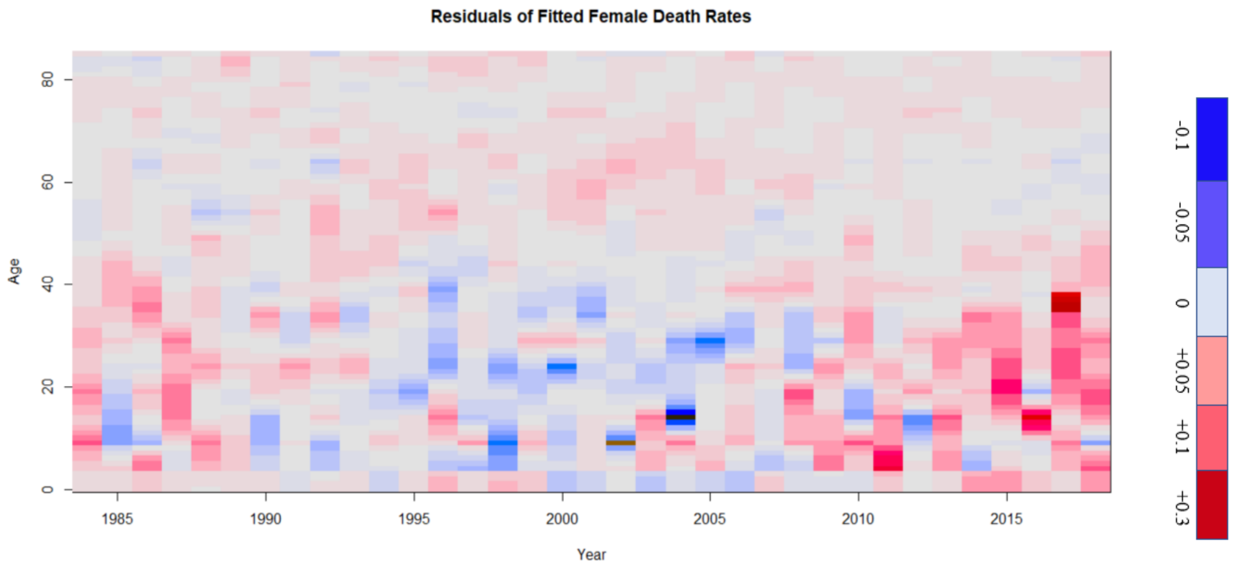

Figure 5.7: Heatmap of residuals of fitted female death rates

The Figure 5.7 exhibits the residuals of the fitted female death rates whereby the y-axis represents age of female and x-axis is the year. Based on the colour key, it can be seen that most of the death rates fitted the model effectively well with the lowest errors, whereby the white colour cells show a residual value of 0. Additionally, between ages 5 and 20 years, fluctuations in the residuals of the fitted death rates can be noticed, whereby when looking at the year 2004 for the age 15 years, the error value was in the range of -0.05 and -0.1, implying that fitted mortality rate was greater than the actual ones. In addition to this, the dark red cells which fall under the range of 0.1 to 0.3 can be seen for the age 35 years during the year of 2017. But overall, it can be perceived that the residuals are not distributed symmetrically throughout the years for all ages.

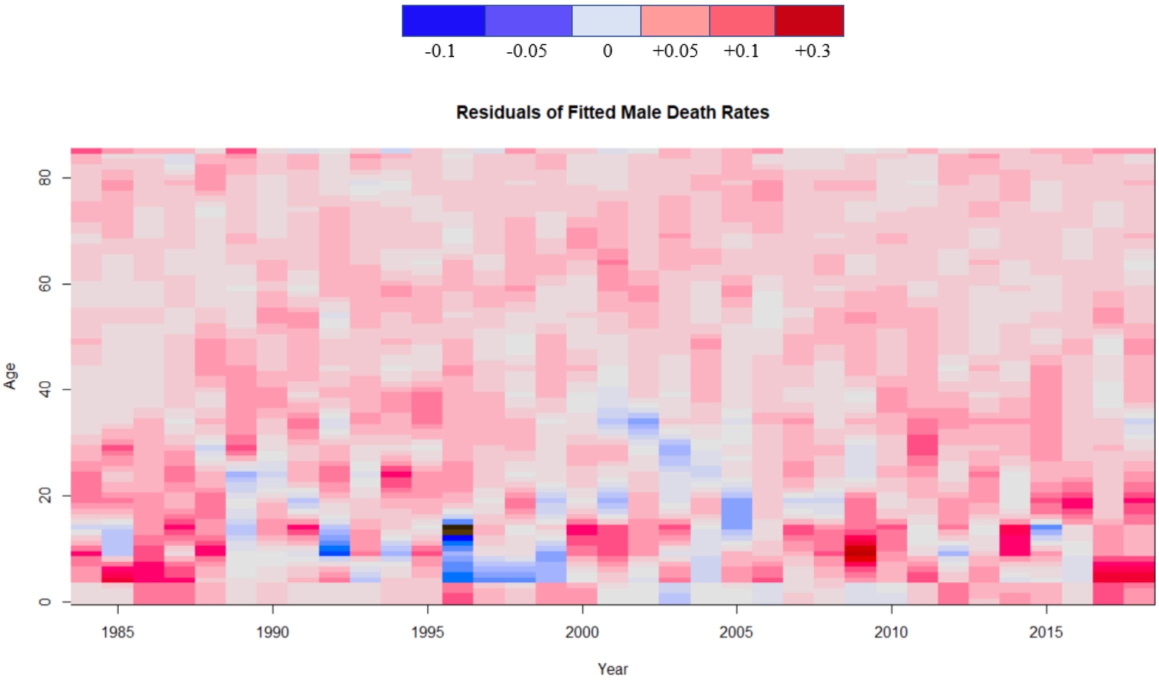

Figure 5.8: Heatmap of residuals of fitted male death rates

Likewise, the heatmap for male fitted death rates was plotted, whereby the differences between the actual and fitted death rates for males can be observed. It can be seen that for the ages 40 years and above, the residuals are mostly similar through the years 1984 to 2018. However, most of the residuals can be noticed to be significant between the ages 5 and 20 years from 1984 to 2018. For instance, for the year 1996, the errors between the actual and fitted mortality rates for males can be observed to be within the range of -0.1 and -0.5 for the ages 5 to 18 years. In addition to this, it can also be noticed that for the year 2009 for the ages 10 to 15 years, the residual values fall under the range of +0.1 and +0.3, implying that there are some discrepancies in the mortality data for that particular year.

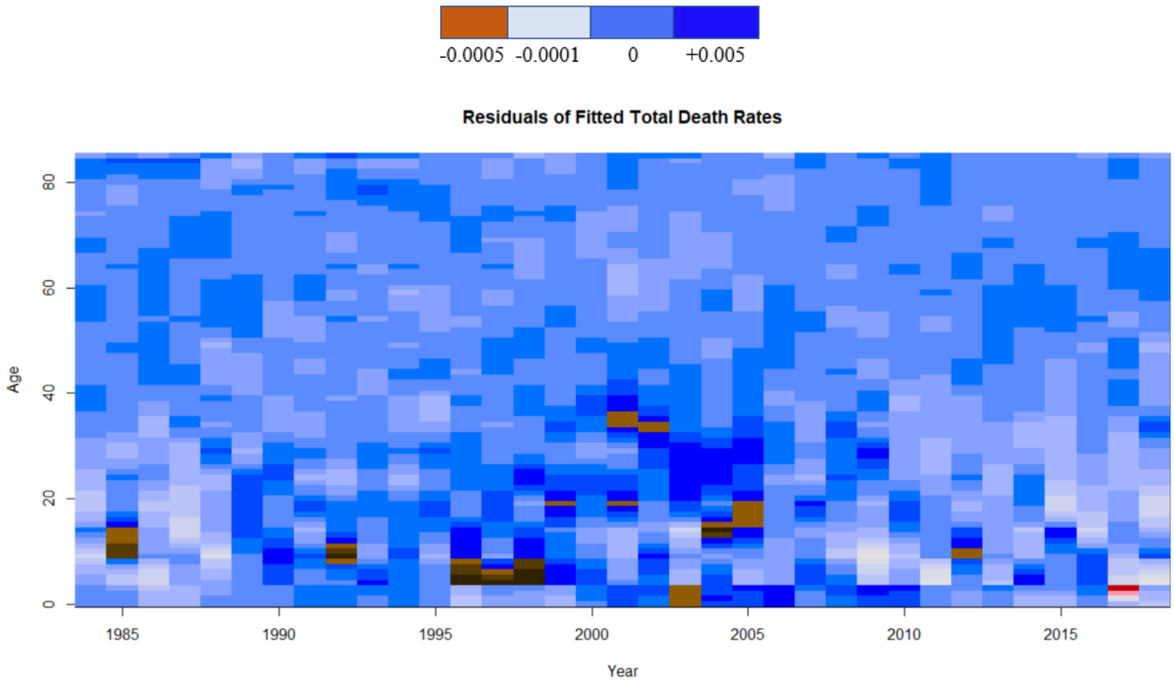

Figure 5.9: Heatmap of residuals of fitted total death rates

Figure 5.9 illustrates the residuals for the total population whereby it can be seen that the model fitted the death data of the total Mauritian population effectively well, since most of the residual values are 0. However, there are few anomalies where the errors are negative and fall under the range of -0.0005 and -0.0001. For instance, from 1996 to 1998 for the ages 5 to 15 years, the fitted death rates are greater than the actual values, thus implying that the model does not fit the data well at those instances.

5.6 Discussion

In regard with the findings of the study, a decreasing trend in the forecasted mortality rates was observed for the next 30 years, whereby males seem to have higher probability of death as compared to females. In addition to this, the results obtained highlighted the inadequacy of the NMF method as well as the ARIMA model. As such, the parameters of the Lee-Carter model such as a_x and b_x were estimated using the SVD technique while the mortality index parameter k_t was projected with the aid of the RNN-LSTM model. In view of the performance evaluation metrics namely MSE, AIC and BIC, the LSTM model was found to outperform the ARIMA model of order (0,1,0) which is generally used when the original Lee-Carter model is applied for modelling and forecasting future death rates. Although the ARIMA model is widely used in exhibiting time series data, but when it comes to demographical data, it cannot work well since there are huge volatility changes in the datasets. Owing to this, the Recurrent Neural Network with the Long Short-Term Memory architecture was proposed in this study since the neural network algorithm is a powerful learning one. Additionally, the LSTM network demonstrated to have produced better forecasts for the mortality index parameter than those obtained using the ARIMA model, this is because the LSTM architecture learnt well the trend in the historical Mauritian mortality data and substituted it with high accuracy for forthcoming years.

Coupled to the above, another noteworthy aspect of the RNN-LSTM model observed was that during the experiments in order to determine the most suitable LSTM model, there was not the need for prior selection of time steps. However, it was highly crucial to satisfy the input shape dimension of the LSTM model which is generally a 3D one, instead of 2D like the usual neural networks. In addition to this, the LSTM model is commonly structured in the aim of explaining long sequences of data, therefore forming a memory which can preserve the significant relationships between data. In other words, the LSTM architecture helped to keep information over time during processing, ensuring that the patterns in the past mortality data do not gradually disappear. These features of the LSTM model contributed to better and more accurate forecasted values for the mortality index parameter unlike the ARIMA model.

With the aid of the projected death rates, not only actuaries but demographers as well will be able to apprehend the imminent improvement in mortality in the upcoming years. As such, using this deep-learning integrated Lee-Carter model, insurance companies along with pension providers will be capable of eradicating the effects of longevity risk on their products as well as enhancing their pricing techniques of pension annuities and life insurance policies. In other words, this proposed model will not only help the latter to reduce supplementary costs due to longevity risk but will also be able to maximize their profits while founding their future liabilities.

5.7 Summary

Chapter 5 emphasized primarily on the results obtained during modelling and forecasting the future death rates of the Mauritian population using the deep-learning integrated Lee-Carter model. The techniques proposed were compared effectively using the performance evaluation metrics namely MSE, AIC and BIC. It was found that the SVD method was better than the NMF one when it comes to estimating the parameters $\mathrm{a_x}$ and $\mathrm{b_x}$ of the Lee-Carter model. In addition to this, the RNN-LSTM model outperformed the ARIMA model and was used to project the mortality index parameter k_t. The forecasted death rates were also assessed based on their measures of errors.

CHAPTER 6

CONCLUSION AND RECOMMENDATION

This chapter will focus primarily on concluding the whole project whereby a summary of the results will be provided. Moreover, it will also highlight the limitations faced in conducting this study as well as the recommendations to improve the validity of the findings. Further area of focus for future advanced researches will also be outlined.

6.1 Summary

Mortality is a substantial contributing factor to population dynamics and is vital in many fields namely economy, demography and social sciences. Intrinsically, the Lee-Carter model has been an enormous milestone in the field of mortality, whereby modelling and forecasting mortality rates were made easier with the aid of this approach. In the Lee-Carter model, the k_t parameter which describes the mortality trend over time, plays a crucial role regarding the future mortality behaviour. The traditional way of estimating this parameter is by applying the ARIMA model of order (0,1,0). However, this technique did not only show evident limitations to outline future mortality shape but also is unable to deal with non-linear mortality patterns. Therefore, the main goal of this research was to use an alternative approach based on a deep learning technique in the aim of enhancing the predictive ability of the Lee-Carter model. More precisely, the Recurrent Neural Network with Long Short-Term Memory architecture was proposed to project the mortality index parameter and to compare the results with those obtained using the ARIMA model, based on performance evaluation metrics such as MSE, AIC and BIC.

This research demonstrated the application of the RNN-LSTM on the death data whereby after preliminary round of fine-tuning that the most suitable LSTM model with 2 hidden layers, adam as optimizer and linear as activation function was determined. Moreover, the NMF method which was a new method to be applied to mortality data for matrix decomposition, proved not to be as efficient as the SVD technique. When the ARIMA performance was compared to that of the LSTM one, it was deduced that the LSTM seemed to have captured the long-term trend inside the sequential data better than the ARIMA model, which was also observed using the k_t plots and results of goodness of fit measures.

The findings showed that ARIMA had MSE of 0.00886 for the total population while LSTM gave 0 value.

Using the SVD technique the parameters a_x and b_x were estimated while the LSTM model was used to forecast the mortality index parameter k_t. Once the parameters of the Lee-Carter model were found, the death rates for the next 30 years were forecasted and evaluated using the performance evaluation tools. It was perceived that that the projected mortality rates were reliable owing to their low MSE values as well as their AIC and BIC approaching the negative infinity.

However, this study has certain implications regarding the LSTM model. For instance, the Recurrent Neural Network is a flexible tool and it can aid to overcome the shortcomings of the ARIMA model and produce accurate forecasts for death rates but coming up with the best LSTM model is an actual challenge. As such, only few experiments were performed in order to choose the hyperparameters, owing to the continuous run time of the neural network models, which will be further elaborated in the next section.

6.2 Limitations of Study

During the course of this study, few difficulties were faced, hampering some facets of the project. To the best of my knowledge, no such research has been conducted before on the Mauritian mortality data i.e. no one has ever applied this deep-learning integrated Lee-Carter model, thus, there is no benchmark to the level of accuracy that should have been met in this study. Furthermore, as discussed earlier, in the aim of determining the most suitable LSTM model, it is highly significant to conduct a number of experiments so that the best hyperparameters are chosen. As such, in this study, owing to time-constraint and the duration of each run time of the mode, only few fine-tuning experiments were performed. Additionally, the robustness of the LSTM model is yet to be investigated before concluding the final model.

Coupled to the above, since the deep-learning integrated Lee-Carter model is still new to actuaries and demographers, it will take the latter time to adopt this new model when it comes to modelling and forecasting the death rates of any particular country. Furthermore, even though the LSTM model produced more accurate forecasts than the ARIMA model, its usefulness still needs to be researched further.

6.3 Future Work

This project not only contributed a new mortality forecasting model to help actuaries and demographers to deal with the imminent problems related to ageing population but also added on to previous literatures, whereby to my knowledge, no one has applied this deep-learning integrated Lee-Carter to the Mauritian population before. However, this study also revealed numerous lines of investigation which can be pursued as future work. As such, the following techniques to analyse the problem more deeply as well as to deal with the proposed methodology are suggested.

6.3.1 Gated Recurrent Unit (GRU)

Another popular Recurrent Neural Network architecture is the gated recurrent unit (GRU) which can also be used as an alternative to the LSTM model. The GRU is usually regarded as less complex as compared to the LSTM architecture. The GRU can be implemented to forecast the mortality index parameter and the results can be compared against those of the LSTM, to determine which architecture is the best.

6.3.2 Fine-Tune Model Parameters

To determine the best LSTM model, it is highly crucial to obtain the best hyperparameters. Therefore, it is recommended that future researchers experiment by either increasing or decreasing the number of hidden layers in the neural network, while assessing the optimizers and activation functions, so that better outcomes are attained. Furthermore, one may also explore by adding drop-out layers to the neural network for building more robust models. Additionally, the cross-validation technique can be used when it comes to splitting the dataset into training and test, instead of applying only the 60%-40% split ratio. This will further enhance the accuracy of the model performance.

6.3.3 Other approaches

Coupled to the above recommendations, it is also advised that while modelling and forecasting mortality rate of any population, to consider the approach of individual information such as health condition, socio-economic factors. With the aid of cause of death information, researchers will gain a better understanding of observed and predicted death rates. Finally, extending the application of this deep-learning integrated Lee-Carter model to other countries can help in exploring the potentiality of the LSTM architecture in other stochastic mortality models.

REFERENCES

Booth, H. and Tickle, L. (2008) Mortality Modelling and Forecasting: a Review of Methods. Annals of Actuarial Science. [Online] 3(1-2). pp.3-43. Available at: http://www.ressources-actuarielles.net/EXT/ISFA/1226.nsf/0/8e4362531df5fbcdc125785d005906d6/$FILE/Booth_Tickle_2008.pdf [Accessed 17 May 2019].

Carter, L. and Lee, R. (1992) Modelling and forecasting US sex differentials in mortality. *International Journal of Forecasting.* [Online] 8(3). pp.393-411. Available at: https://www.jstor.org/stable/2290201 [Accessed 28 May 2019].

Castellani, C., Duff, A., Bell, S., Heijerman, H., Munck, A., Ratjen, F., Sermet-Gaudelus, I., Southern, K., Barben, J., Flume, P., Hodková, P., Kashirskaya, N., Kirszenbaum, M., Madge, S., Oxley, H., Plant, B., Schwarzenberg, S., Smyth, A., Taccetti, G., Wagner, T., Wolfe, S. and Drevinek, P. (2018) ECFS best practice guidelines: the 2018 revision. *Journal of Cystic Fibrosis* [Online] 17(2). pp.153-178. Available at: https://www.ncbi.nlm.nih.gov/pubmed/29506920 [Accessed 15 Jun. 2019].

Chen, R. and Millossovich, P. (2018) Sex-specific mortality forecasting for UK countries: a coherent approach. *European Actuarial Journal* [Online] 8(1). pp.69-95. Available at: https://www.researchgate.net/publication/322896118_Sex-specific_mortality_forecasting_for_UK_countries_a_coherent_approach [Accessed 4 Jun. 2019].

Cheung, V., Devarajan, K., Severini, G., Turolla, A. and Bonato, P. (2017) Decomposing time series data by a non-negative matrix factorization algorithm with temporally constrained coefficients. *Medicine and Biology Society (MBS)* [Online] 12(2). pp.15-19. Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5593271/ [Accessed 26 Jun. 2019].

Deprez, P., Shevchenko, P. and Wuthrich, M. (2017) Machine Learning Techniques for Mortality Modelling. *SSRN Electronic Journal*. [Online] 12(2). pp.34-38. Available at: https://arxiv.org/pdf/1705.03396.pdf [Accessed 10 May 2019].

Duolao, W. and Pengjun, L. (2005) Modelling and forecasting mortality distributions in England and Wales using the Lee–Carter model. *Journal of Applied Statistics* [Online] 32(9). pp.873-885. Available at:

https://econpapers.repec.org/article/tafjapsta/v_3a32_3ay_3a2005_3ai_3a9_3ap_3a873-885.htm [Accessed 14 Jun. 2019].

Filip, R. (2017) Forecasting the mortality rates using Lee-Carter model and Heligman-Pollard model. *Journal of Physics*. [Online] 890(12). p.012128. Available at: https://iopscience.iop.org/article/10.1088/1742-6596/890/1/012128 [Accessed 10 May 2019]

Girosi, F. and King, G. (2006) Extending Lee–Carter Mortality Forecasting. *Mathematical Population Studies.* [Online] 13(1). pp.1-18. Available at: https://gking.harvard.edu/files/gking/files/lc.pdf [Accessed 11 May 2019].

Goodfellow, I., Bengio, Y. and Courville, A. (2016) Deep Learning. *Artificial Intelligence and Computing Journal* [Online] 11(4). pp.14-17. Available at: https://www.deeplearningbook.org/front_matter.pdf [Accessed 22 Jul. 2019].

Hainaut, D. (2018) A neural-network analyser for mortality forecast. *ASTIN Bulletin*. [Online] 48(02). pp.481-508. Available at: https://www.cambridge.org/core/journals/astin-bulletin-journal-of-the-iaa/article/neuralnetwork-analyzer-for-mortality-forecast/9045C2A616EF9E9B063560704DC399AD [Accessed 10 May 2019].

Hasim, S., Andrew, S. and Francoise, B. (2014) Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modelling. *INTERSPEECH 2014* [Online] 3(2). pp.13-18. Available at: https://wiki.inf.ed.ac.uk/twiki/pub/CSTR/ListenTerm1201415/sak2.pdf [Accessed 8 Dec. 2019].

Heligman, L. and Pollard, J. (1980) The age pattern of mortality. *Journal of the Institute of Actuaries* [Online] 107(1). pp.49-80. Available at: https://www.actuaries.org.uk/system/files/documents/pdf/0049-0080.pdf [Accessed 23 May 2019].

Hochreiter, S. and Schmidhuber, J. (1997) The Long Short-Term Memory. *Neural Computation* [Online] 9(8). pp.1735-1780. Available at: https://www.bioinf.jku.at/publications/older/2604.pdf [Accessed 8 Aug. 2019].

Hordri, N., Yuhaniz, S. and Shamsuddin, S. (2016) A Systematic Literature Review on Features of Deep Learning in Big Data Analytics. *International Journal of Advances in Soft*

Computing and its Applications [Online] 9(1). pp.32-49. Available at: https://www.researchgate.net/publication/316716909_A_Systematic_Literature_Review_on_Features_of_Deep_Learning_in_Big_Data_Analytics [Accessed 14 Jul. 2019].

Kleinow, T. (2015) A common age effect model for the mortality of multiple populations. *Insurance: Mathematics and Economics* [Online] 63(3). pp.147-152. Available at: https://www.researchgate.net/publication/274572687_A_common_age_effect_model_for_the_mortality_of_multiple_populations [Accessed 6 Jun. 2019].

Kuo, K. (2019) Deep Triangle: A Deep Learning Approach to Loss Reserving. Applied Stochastic Models in Business and Industry. [Online] 12(2). pp.13-17. Available at: https://arxiv.org/pdf/1804.09253.pdf [Accessed 31 May 2019].

Le, Q., Jaitly, N. and Hinton, G. (2015) A Simple Way to Initialize Recurrent Networks of Rectified Linear Units. *Neural and Evolutionary Computing* [Online] 11(2). pp.14-19. Available at: https://arxiv.org/abs/1504.00941 [Accessed 17 Aug. 2019].

Lee, R. and Carter, L. (1992) Modelling and Forecasting U. S. Mortality. *Journal of the American Statistical Association*. [Online] 87(419). p.659. Available at: https://www.jstor.org/stable/2290201?seq=1#page_scan_tab_contents [Accessed 9 May 2019].

Lee, R. and Miller, T. (2001) Evaluating the Performance of the Lee-Carter Method for Forecasting Mortality. *Demography* [Online] 38(4). p.537. Available at: http://pages.stern.nyu.edu/~dbackus/BCH/demography/LeeMiller_Demo_01.pdf [Accessed 22 May 2019].

Levantesi, S. and Pizzorusso, V. (2019) Application of Machine Learning to Mortality Modelling and Forecasting. *Risks* [Online] 7(1). p.26. Available at: https://www.mdpi.com/2227-9091/7/1/26 [Accessed 9 May 2019].

Lin, L. and Huang, N. (2008) Credit Risk Assessment Using BP Neural Network with Dempster-Shafer Theory. *Artificial Intelligence and Computational Intelligence* [Online] 11(2). pp.12-18. Available at: https://www.semanticscholar.org/paper/Credit-Risk-Assessment-Using-BP-Neural-Network-with-Lin-Huang/810ce74ec99488ae25197001d17cb9f26cae4649 [Accessed 22 Jul. 2019].

Martial, C. (2018) *Digest of Demographic Statistics*. [Online] Available at: http://statsmauritius.govmu.org/English/StatsbySubj/Documents/Digest/Demography/Digest_Demo_Yr17.pdf [Accessed 10 May 2019].

Martino, L., Elvira, V., Luengo, D., Artes-Rodriguez, A. and Corander, J. (2015) Orthogonal MCMC algorithms. *IEEE Workshop on Statistical Signal Processing (SSP)* [Online] 10(5). pp.12-13. Available at: https://www.researchgate.net/publication/264466025_Orthogonal_MCMC_algorithms [Accessed 8 Jun. 2019].

Nigri, A., Levantesi, S., Marino, M., Scognamiglio, S. and Perla, F. (2019) A Deep-learning Integrated Lee–Carter Model. *Risks.* [Online] 7(1). p.33. Available at: https://www.mdpi.com/2227-9091/7/1/33 [Accessed 9 May 2019].

Richman, R. and Wuthrich, M. (2018) A Neural Network Extension of the Lee-Carter Model to Multiple Populations. *SSRN Electronic Journal.* [Online] 21(2). pp.23-34. Available at: https://www.colloquium2019.org.za/wp-content/uploads/2019/04/Richman-Wuthrich-final.pdf [Accessed 9 May 2019].

Safitri, L., Mardiyati, S. and Rahim, H. (2018) Forecasting the mortality rates of Indonesian population by using neural network. *Journal of Physics: Conference Series* [Online] 974(1). pp.234-246. Available at: https://www.researchgate.net/publication/323926156_Forecasting_the_mortality_rates_of_Indonesian_population_by_using_neural_network [Accessed 5 Aug. 2019].

Salehinejad, H., Sankar, S., Barfett, J., Colak, E. and Valaee, S. (2018) Recent Advances in Recurrent Neural Networks. *Neural and Evolutionary Computing.* [Online] 10(3). pp.13-18. Available at: https://arxiv.org/abs/1504.00941 [Accessed 13 Aug. 2019].

SDM (2018) *Population and Vital Statistics Republic of Mauritius*. [Online] Available at: http://statsmauritius.govmu.org/English/Publications/Documents/EI1334/Pop_Vital_JanJunYr18.pdf [Accessed 11 May 2019]

Skymind (2018). *A Beginner's Guide to Neural Networks and Deep Learning*. [Online] Available at: https://skymind.ai/wiki/neural-network [Accessed 30 Jul. 2019].

Vargas, R., Mosavi, A. and Ruiz, R. (2017) Deep Learning: A Review. *Advances in Intelligent Systems and Computing* [Online] 5(2). pp.10-14. Available at:

https://www.researchgate.net/publication/318447392_DEEP_LEARNING_A_REVIEW [Accessed 30 May 2019].

Woodun, D., Ho, M. and Raja, R. (2019) Lee-Carter Mortality Forecasting: Application to Mauritian Population. *International Journal of Recent Technology and Engineering (IJRTE)* [Online] 7(5S). pp.169-175. Available at: https://www.ijrte.org/wp-content/uploads/papers/v7i5s/ES2143017519.pdf [Accessed 5 Jul. 2019].

Yasungnoen, N. and Sattayatham, P. (2016) Forecasting Thai Mortality by Using the Lee-Carter Model. *Asia-Pacific Journal of Risk and Insurance* [Online] 10(1). pp.91-105. Available at: https://econpapers.repec.org/article/bpjapjrin/v_3a10_3ay_3a2016_3ai_3a1_3ap_3a91-105_3an_3a2.htm [Accessed 12 Jun. 2019].

APPENDIX A: R CODES

```
#installing packages
install.packages("DataExplorer")
install.packages("demography")
install.packages("demogR")
install.packages("rainbow")
install.packages("forecast")
install.packages("NMF")
install.packages("keras")
install.packages("Metrics")
install.packages("mlr")
install.packages("kerasR")
update.packages()
install.packages("devtools")
install.packages("rnn")
install.packages("tensorflow")
install.packages("dplyr")
install.packages("neuralnet")
install.packages("magrittr")
install.packages("moments")
install.packages("quadprog")
install.packages("TTR")
install.packages("quantreg")
install.packages("geometry")
install.packages("purrr")
install.packages("rminer")


library(devtools)
library(demography)
library(forecast)
library(demogR)
library(rainbow)
library(Metrics)
library(moments)
library(quadprog)
library(TTR)
library(quantreg)
library(geometry)


#importing dataset
setwd("C:/Users/USER/Desktop")
death <- read.table("Mortality.txt",h=T,sep='\t')
exposure <- read.table("Population.txt",h=T,sep='\t')

View(death)
View(exposure)

#data preprocessing for RNN
dim(death)
```

```
dim(exposure)
head(death)
head(exposure)
str(death)
sum(is.na(death))
str(exposure)
sum(is.na(exposure))
exposure$Age<-as.numeric(exposure$Age)
death$Age<-as.numeric(death$Age)

summary(death)
sd(death$Age)
sd(death$Male)
sd(death$Female)
sd(death$Total)
skewness(death$Male)
skewness(death$Female)
skewness(death$Total)
kurtosis(death$Male)
kurtosis(death$Female)
kurtosis(death$Total)


summary(exposure)
sd(exposure$Age)
sd(exposure$Male)
sd(exposure$Female)
sd(exposure$Total)
skewness(exposure$Male)
skewness(exposure$Female)
skewness(exposure$Total)
kurtosis(exposure$Male)
kurtosis(exposure$Female)
kurtosis(exposure$Total)


library(DataExplorer)
plot_missing(death)
plot_missing(exposure)
boxplot(death)
boxplot(exposure)
plot_histogram(death)
plot_histogram(exposure)
plot_density(death)
plot_density(exposure)
plot_correlation(death)
plot_correlation(exposure)


#Creating demogdata object for Male,Female and Total
```

```
death$Age<-as.numeric(as.character(death$Age))
death$Age[is.na(death$Age)]<-85
exposure$Age<-as.numeric(as.character(exposure$Age))
exposure$Age[is.na(exposure$Age)]<-85

YEAR<-unique(death$Year);nC=length(YEAR)
AGE<-unique(death$Age);nL=length(AGE)
mortalityMale<-matrix(death$Male/exposure$Male,nL,nC)
populationMale<-matrix(exposure$Male,nL,nC)
mortalityFemale<-matrix(death$Female/exposure$Female,nL,nC)
populationFemale<-matrix(exposure$Female,nL,nC)
mortalityTotal<-matrix(death$Total/exposure$Total,nL,nC)
populationTotal<-matrix(exposure$Total,nL,nC)

maledata<-
demogdata(data=mortalityMale,pop=populationMale,ages=AGE,years=YEAR,type="mort
ality",label="Mauritius",name="Male",lambda=1)
maledata

femaledata<-
demogdata(data=mortalityFemale,pop=populationFemale,ages=AGE,years=YEAR,type="
mortality",label="Mauritius",name="Female",lambda=1)
femaledata

totaldata<-
demogdata(data=mortalityTotal,pop=populationTotal,ages=AGE,years=YEAR,type="mor
tality",label="Mauritius",name="Total",lambda=1)
totaldata


#plotting death rates against age
par(mfrow=c(1,3))
#par(mfrow=c(1,1))
plot(femaledata)
plot(maledata)
plot(totaldata)


#read.demogdata suitable for fitting lee-carter
demogdata=read.demogdata("Mortality.txt","Population.txt",type="mortality",label="Mau
ritius",max.mx=10,skip=0,scale=1)
demogdata

par(mfrow=c(1,3))
plot(demogdata,series="Male",datatype="rate",main="Male rates")
plot(demogdata,series="Female",datatype="rate",main="Female rates")
plot(demogdata,series="Total",datatype="rate",main="Total rates")


par(mfrow=c(1,3))
```

```
plot(demogdata,series="Male",datatype="rate",plot.type="time",main="Male
rates",xlab="Years")
plot(demogdata,series="Female",datatype="rate",plot.type="time",main="Female
rates",xlab="Years")
plot(demogdata,series="Total",datatype="rate",plot.type="time",main="Total
rates",xlab="Years")


#Fitting Lee-Carter with SVD
mydataLcaM<-lca(demogdata,series="Male",max.age=85)
mydataLcaF<-lca(demogdata,series="Female",max.age=85)
mydataLcaT<-lca(demogdata,series="Total",max.age=85)

mydataLcaM
mydataLcaF
mydataLcaT

mydataLcaF$ax
mydataLcaF$bx
mydataLcaM$ax
mydataLcaM$bx
mydataLcaT$ax
mydataLcaT$bx

par(mfrow=c(1,3))
plot(fitted(mydataLcaM), main="Male")
plot(fitted(mydataLcaF),main="Female")
plot(fitted(mydataLcaT),main="Total")

par(mfrow=c(1,3))
plot(mydataLcaT$ax,main="ax",xlab="Age",ylab="ax",type="l")
lines(x=mydataLcaF$age,y=mydataLcaF$ax,main="ax",col="red")
lines(x=mydataLcaM$age,y=mydataLcaM$ax,main="ax",col="blue")
legend("topleft",c("Male","Female","Total"),cex=0.6,col=c("blue","red","black"),lty=1)

plot(mydataLcaT$bx,main="bx",xlab="Age",ylab="bx",type="l")
lines(x=mydataLcaF$age,y=mydataLcaF$bx,main="bx",col="red")
lines(x=mydataLcaM$age,y=mydataLcaM$bx,main="bx",col="blue")
legend("topleft",c("Male","Female","Total"),cex=0.6,col=c("blue","red","black"),lty=1)

plot(mydataLcaT$kt,main="kt",xlab="Year",ylab="kt",type="l")
lines(x=mydataLcaF$year,y=mydataLcaF$kt,main="kt",col="red")
lines(x=mydataLcaM$year,y=mydataLcaM$kt,main="kt",col="blue")
legend("topleft",c("Male","Female","Total"),cex=0.6,col=c("blue","red","black"),lty=1)

summary(mydataLcaF)
summary(mydataLcaM)
summary(mydataLcaT)
```

```
plot(residuals(mydataLcaF),main="Residuals of Fitted Female Death Rates")
plot(residuals(mydataLcaM),main="Residuals of Fitted Male Death Rates")
plot(residuals(mydataLcaT),main="Residuals of Fitted Total Death Rates")

par(mfrow=c(1,3))
plot(forecast(mydataLcaF,h=30),plot.type="component")
plot(forecast(mydataLcaM,h=30),plot.type="component")
plot(forecast(mydataLcaT,h=30),plot.type="component")

fM<-forecast(mydataLcaM,h=30)
fF<-forecast(mydataLcaF,h=30)
fT<-forecast(mydataLcaT,h=30)

ratesM<-cbind(demogdata$rate$Male[1:85,],fM$rate$Male[1:85,])
ratesF<-cbind(demogdata$rate$Female[1:85,],fF$rate$Female[1:85,])
ratesT<-cbind(demogdata$rate$Total[1:85,],fT$rate$Total[1:85,])


fMrates<-fM$rate$Male[1:85,]
fFrates<-fF$rate$Female[1:85,]
fTrates<-fT$rate$Total[1:85,]

par(mfrow=c(1,3))
plot(fM)
plot(fF)
plot(fT)

#Age 65
par(mfrow=c(1,1))
plot(seq(min(demogdata$year),max(demogdata$year)+30),ratesF[65,],col="red",xlab="Ye
ars",ylab="Death Rates",type="l",main="Past and Forecasted Death rates")
lines(seq(min(demogdata$year),max(demogdata$year)+30),ratesM[65,],col="blue",xlab="
Years",ylab="Death Rates")
lines(seq(min(demogdata$year),max(demogdata$year)+30),ratesT[65,],col="black",xlab="
Years",ylab="Death Rates")
legend("topright",c("Male","Female","Total"),cex=0.8,col=c("blue","red","black"),lty=1)

#Age 20
par(mfrow=c(1,1))
plot(seq(min(demogdata$year),max(demogdata$year)+30),ratesF[20,],col="red",xlab="Ye
ars",ylab="Death Rates",type="l",main="Past and Forecasted Death rates")
lines(seq(min(demogdata$year),max(demogdata$year)+30),ratesM[20,],col="blue",xlab="
Years",ylab="Death Rates")
lines(seq(min(demogdata$year),max(demogdata$year)+30),ratesT[20,],col="black",xlab="
Years",ylab="Death Rates")
legend("topright",c("Male","Female","Total"),cex=0.8,col=c("blue","red","black"),lty=1)

#Age 80
par(mfrow=c(1,1))
```

```
plot(seq(min(demogdata$year),max(demogdata$year)+30),ratesF[80,],col="red",xlab="Years",ylab="Death Rates",type="l",main="Past and Forecasted Death rates")
lines(seq(min(demogdata$year),max(demogdata$year)+30),ratesM[80,],col="blue",xlab="Years",ylab="Death Rates")
lines(seq(min(demogdata$year),max(demogdata$year)+30),ratesT[80,],col="black",xlab="Years",ylab="Death Rates")
legend("topright",c("Male","Female","Total"),cex=0.8,col=c("blue","red","black"),lty=1)


fT
fM
fF

summary(fM)
summary(fF)
summary(fT)


print(fMrates)
print(fFrates)
print(fTrates)


#ARIMA to forecast kt
fM$kt.f$lower
fM$kt.f$upper
fF$kt.f$lower
fF$kt.f$upper
fT$kt.f$lower
fT$kt.f$upper

par(mfrow=c(1,3))
plot(fM$kt.f,main="Male",xlab="Year",ylab="kt")
lines(fM$kt.f$lower,col="red")
lines(fM$kt.f$upper,col="yellow")
legend("topleft",c("Upper Quartile","Mean","Lower Quartile"),col=c("yellow","blue","red"),lty=1,cex=0.4)

plot(fF$kt.f,main="Female",xlab="Year",ylab="kt")
lines(fM$kt.f$lower,col="red")
lines(fM$kt.f$upper,col="yellow")
legend("topleft",c("Upper Quartile","Mean","Lower Quartile"),col=c("yellow","blue","red"),lty=1,cex=0.4)

plot(fT$kt.f,main="Total",xlab="Year",ylab="kt")
lines(fM$kt.f$lower,col="red")
lines(fM$kt.f$upper,col="yellow")
legend("topleft",c("Upper Quartile","Mean","Lower Quartile"),col=c("yellow","blue","red"),lty=1,cex=0.4)
```

```
fM$kt.f$lower
fM$kt.f$upper
fF$kt.f$lower
fF$kt.f$upper
fT$kt.f$lower
fT$kt.f$upper

fF$kt.f
fM$kt.f
fT$kt.f


#Fitting Lee-Carter with NMF
library(NMF)
demogdatanmf<-nmfObject(demogdata)
demogdatanmf

NMFLcaM<-lca(demogdatanmf,series="Male",max.age=85)
NMFLcaF<-lca(demogdatanmf,series="Female",max.age=85)
NMFLcaT<-lca(demogdatanmf,series="Total",max.age=85)

summary(NMFLcaM)
summary(NMFLcaF)
summary(NMFLcaT)

par(mfrow=c(1,3))
plot(fitted(NMFLcaM),main="Male")
plot(fitted(NMFLcaF),main="Female")
plot(fitted(NMFLcaT),main="Total")

par(mfrow=c(1,3))
plot(NMFLcaT$ax,main="ax",xlab="Age",ylab="ax",type="l")
lines(x=NMFLcaF$age,y=NMFLcaF$ax,main="ax",col="green")
lines(x=NMFLcaM$age,y=NMFLcaM$ax,main="ax",col="blue")
legend("topleft",c("Male","Female","Total"),cex=0.4,col=c("blue","green","black"),lty=1)

plot(NMFLcaT$bx,main="bx",xlab="Age",ylab="bx",type="l")
lines(x=NMFLcaF$age,y=NMFLcaF$bx,main="bx",col="green")
lines(x=NMFLcaM$age,y=NMFLcaM$bx,main="bx",col="blue")
legend("topright",c("Male","Female","Total"),cex=0.4,col=c("blue","green","black"),lty=1
)

plot(NMFLcaT$kt,main="kt",xlab="Year",ylab="kt",type="l")
lines(x=NMFLcaF$year,y=NMFLcaF$kt,main="kt",col="green")
lines(x=NMFLcaM$year,y=NMFLcaM$kt,main="kt",col="blue")
legend("topright",c("Male","Female","Total"),cex=0.4,col=c("blue","green","black"),lty=1
)

nmffM<-forecast(NMFLcaM,h=30)
nmffF<-forecast(NMFLcaF,h=30)
```

```
nmffT<-forecast(NMFLcaT,h=30)

nmffT
nmffM
nmffF

summary(nmffM)
summary(nmffF)
summary(nmffT)



#Fitting the Recurrent Neural Networks with LSTM
#Splitting the demogdata into training and test set
#Using SVD method, since no difference in NMF and SVD
#Training set:1984-2004
#Test set:2005-2018

library(keras)
library(kerasR)
library(tensorflow)
library(neuralnet)
library(magrittr)
library(mlr)
library(dplyr)
library(rnn)
library(purrr)
library(rminer)

training<-death[1:1806,]
test<-death[1807:nrow(death),]

View(training)
View(test)

str(training)
str(test)


#Male as target
train<-training[,(1:3)]
test_set<-test[,(1:3)]
train_reshape<-array(dim=c(1806,1,2))
test_reshape<-array(dim=c(1204,1,2))
traintarget<-train$Male
traintarget<-array(dim=c(1806))

dim(train_reshape)
dim(traintarget)
```

```
#Double LSTM Hidden Layers with rmsprop optimizer
#50 Epochs
model <- keras_model_sequential()
model %>%
layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
layer_lstm(units = 1806, return_sequences = F) %>%
# using linear activation on last layer, as output is needed in real number
layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'rmsprop',metrics='accuracy')
summary(model)

model %>% fit(train_reshape, traintarget, epochs=50,validation_split=0.2)

plot(model)

#predict
forecast<-predict(model,test_reshape)
forecast


#100 Epochs
model <- keras_model_sequential()
model %>%
layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
layer_lstm(units = 1806, return_sequences = F) %>%
# using linear activation on last layer, as output is needed in real number
layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'rmsprop',metrics='accuracy')
summary(model)

model %>% fit(train_reshape, traintarget, epochs=100,validation_split=0.2)

plot(model)

#predict
forecast<-predict(model,test_reshape)
forecast


#Double LSTM Hidden Layers with adam optimizer
model <- keras_model_sequential()
model %>%
layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
layer_lstm(units = 1806, return_sequences = F) %>%
# using linear activation on last layer, as output is needed in real number
layer_dense(units = 1, activation = "linear")
```

```
model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(train_reshape, traintarget, epochs=50,validation_split=0.2)

plot(model)

#predict
forecast<-predict(model,test_reshape)
forecast


#Double LSTM Hidden Layers with relu activation
model <- keras_model_sequential()
model %>%
layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T, activation = "relu" )
%>%
#layer_lstm(units = 1806, return_sequences = F) %>%
# using linear activation on last layer, as output is needed in real number
layer_dense(units = 1)

model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(train_reshape, traintarget, epochs=50,validation_split=0.2)

plot(model)

#predict
forecast<-predict(model,test_reshape)
forecast



#Female as target
train<-training[,(1:4)]
test_set<-test[,(1:4)]

training_reshape<-array(dim=c(1806,1,2))
testing_reshape<-array(dim=c(1204,1,2))
dim(training_reshape)
trainingtarget<-training$Female
trainingtarget<-array(dim=c(1806))
dim(trainingtarget)


#Double LSTM Hidden Layers with rmsprop optimizer
model <- keras_model_sequential()
model %>%
layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
```

```
layer_lstm(units = 1806, return_sequences = F) %>%
# using linear activation on last layer, as output is needed in real number
layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'rmsprop', metrics='accuracy')
summary(model)

model %>% fit(training_reshape, trainingtarget, epochs=50,validation_split=0.2)

plot(model)

#predict
forecast<-predict(model,testing_reshape)


#Double LSTM layers with adam optimizer
model <- keras_model_sequential()
model %>%
layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
layer_lstm(units = 1806, return_sequences = F) %>%
# using linear activation on last layer, as output is needed in real number
layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(training_reshape, trainingtarget, epochs=50,validation_split=0.2)

plot(model)

#predict
forecast<-predict(model,testing_reshape)
forecast


#Double LSTM Hidden Layers with relu activation
model <- keras_model_sequential()
model %>%
layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T, activation = "relu" )
%>%
#layer_lstm(units = 1806, return_sequences = F) %>%
# using linear activation on last layer, as output is needed in real number
layer_dense(units = 1)

model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(training_reshape, trainingtarget, epochs=50,validation_split=0.2)

plot(model)
```

```
#predict
forecast<-predict(model,testing_reshape)
forecast


#Total as target
trainT<-training[,(1:5)]
test_setT<-test[,(1:5)]
train_set_reshapeT<-array(dim=c(1806,1,2))
test_set_reshapeT<-array(dim=c(1204,1,2))
dim(train_set_reshapeT)
train_settargetT<-training$Total
train_settargetT<-array(dim=c(1806))
dim(train_settargetT)


#Double LSTM Hidden Layers with rmsprop optimizer
model <- keras_model_sequential()
model %>%
layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
layer_lstm(units = 1806, return_sequences = F) %>%
# using linear activation on last layer, as output is needed in real number
layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'rmsprop', metrics='accuracy')
summary(model)

model %>% fit(train_set_reshapeT, train_settargetT, epochs=50,validation_split=0.2)

plot(model)

#predict
forecast<-predict(model,test_set_reshapeT)


#Double LSTM Hidden Layers with adam optimizer
model <- keras_model_sequential()
model %>%
layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T ) %>%
layer_lstm(units = 1806, return_sequences = F) %>%
# using linear activation on last layer, as output is needed in real number
layer_dense(units = 1, activation = "linear")

model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(train_set_reshapeT, train_settargetT, epochs=50,validation_split=0.2)

plot(model)
```

```
#predict
forecast<-predict(model,test_set_reshapeT)
forecast

#Double LSTM Hidden Layers with relu activation
model <- keras_model_sequential()
model %>%
layer_lstm(units = 1806, input_shape = c(1, 2), return_sequences = T, activation = "relu" )
%>%
#layer_lstm(units = 1806, return_sequences = F) %>%
# using linear activation on last layer, as output is needed in real number
layer_dense(units = 1)

model %>% compile(loss = 'mse', optimizer = 'adam',metrics='accuracy')
summary(model)

model %>% fit(train_set_reshapeT, train_settargetT, epochs=50,validation_split=0.2)

plot(model)

#predict
forecast<-predict(model,test_set_reshapeT)
forecast
```

APPENDIX B: VALUES OF PARAMETERS AND FORECASTED DEATH RATES

Female Forecasted Death Rates 2019-2048

Age	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031
0	0.01039	0.01020	0.01001	0.00983	0.00965	0.00948	0.00931	0.00914	0.00898	0.00881	0.00865	0.00850	0.00834
1	0.00789	0.00775	0.00761	0.00747	0.00734	0.00720	0.00707	0.00694	0.00682	0.00669	0.00657	0.00645	0.00634
2	0.00540	0.00530	0.00520	0.00511	0.00502	0.00493	0.00484	0.00475	0.00466	0.00458	0.00449	0.00441	0.00433
3	0.00291	0.00285	0.00280	0.00275	0.00270	0.00265	0.00260	0.00255	0.00250	0.00246	0.00241	0.00237	0.00232
4	0.00040	0.00039	0.00038	0.00037	0.00036	0.00035	0.00034	0.00034	0.00033	0.00032	0.00031	0.00031	0.00030
5	0.00034	0.00033	0.00033	0.00032	0.00031	0.00030	0.00030	0.00029	0.00028	0.00028	0.00027	0.00026	0.00026
6	0.00029	0.00028	0.00027	0.00027	0.00026	0.00025	0.00025	0.00024	0.00023	0.00023	0.00022	0.00022	0.00021
7	0.00023	0.00022	0.00022	0.00021	0.00021	0.00020	0.00020	0.00019	0.00019	0.00018	0.00018	0.00017	0.00017
8	0.00018	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00014	0.00014	0.00013	0.00013	0.00013	0.00012
9	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010	0.00009	0.00009	0.00009	0.00008	0.00008	0.00008
10	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00010	0.00010	0.00010	0.00010	0.00009	0.00009	0.00009
11	0.00014	0.00013	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010	0.00009
12	0.00014	0.00014	0.00013	0.00013	0.00013	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010
13	0.00015	0.00014	0.00014	0.00014	0.00013	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00010	0.00010
14	0.00015	0.00015	0.00014	0.00014	0.00013	0.00013	0.00013	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010
15	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00014	0.00014	0.00013	0.00013	0.00012
16	0.00021	0.00020	0.00020	0.00019	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00014
17	0.00023	0.00022	0.00022	0.00021	0.00020	0.00020	0.00019	0.00019	0.00018	0.00018	0.00017	0.00017	0.00016
18	0.00026	0.00025	0.00024	0.00023	0.00023	0.00022	0.00021	0.00021	0.00020	0.00019	0.00019	0.00018	0.00018
19	0.00028	0.00027	0.00026	0.00025	0.00024	0.00024	0.00023	0.00022	0.00022	0.00021	0.00020	0.00020	0.00019
20	0.00031	0.00030	0.00029	0.00028	0.00028	0.00027	0.00026	0.00025	0.00025	0.00024	0.00023	0.00023	0.00022
21	0.00034	0.00033	0.00032	0.00031	0.00031	0.00030	0.00029	0.00028	0.00028	0.00027	0.00026	0.00025	0.00025
22	0.00037	0.00036	0.00035	0.00034	0.00034	0.00033	0.00032	0.00031	0.00030	0.00030	0.00029	0.00028	0.00028

23	0.00040	0.00039	0.00038	0.00037	0.00037	0.00036	0.00035	0.00034	0.00033	0.00033	0.00032	0.00031	0.00030
24	0.00043	0.00042	0.00041	0.00040	0.00039	0.00039	0.00038	0.00037	0.00036	0.00035	0.00035	0.00034	0.00033
25	0.00046	0.00045	0.00044	0.00043	0.00042	0.00042	0.00041	0.00040	0.00039	0.00038	0.00038	0.00037	0.00036
26	0.00049	0.00048	0.00047	0.00046	0.00045	0.00044	0.00043	0.00043	0.00042	0.00041	0.00040	0.00040	0.00039
27	0.00051	0.00051	0.00050	0.00049	0.00048	0.00047	0.00046	0.00045	0.00044	0.00044	0.00043	0.00042	0.00041
28	0.00054	0.00053	0.00052	0.00051	0.00050	0.00050	0.00049	0.00048	0.00047	0.00046	0.00045	0.00045	0.00044
29	0.00056	0.00055	0.00055	0.00054	0.00053	0.00052	0.00051	0.00050	0.00049	0.00049	0.00048	0.00047	0.00046
30	0.00061	0.00060	0.00059	0.00058	0.00057	0.00056	0.00055	0.00054	0.00054	0.00053	0.00052	0.00051	0.00050
31	0.00065	0.00064	0.00063	0.00062	0.00061	0.00060	0.00059	0.00058	0.00058	0.00057	0.00056	0.00055	0.00054
32	0.00069	0.00068	0.00067	0.00066	0.00065	0.00064	0.00063	0.00062	0.00061	0.00061	0.00060	0.00059	0.00058
33	0.00073	0.00072	0.00071	0.00070	0.00069	0.00068	0.00067	0.00066	0.00065	0.00064	0.00063	0.00062	0.00062
34	0.00077	0.00076	0.00075	0.00074	0.00073	0.00072	0.00071	0.00070	0.00069	0.00068	0.00067	0.00066	0.00065
35	0.00090	0.00089	0.00088	0.00087	0.00086	0.00085	0.00084	0.00083	0.00082	0.00081	0.00080	0.00079	0.00079
36	0.00099	0.00098	0.00097	0.00096	0.00095	0.00094	0.00093	0.00092	0.00091	0.00090	0.00090	0.00089	0.00088
37	0.00108	0.00107	0.00106	0.00105	0.00104	0.00103	0.00102	0.00101	0.00100	0.00099	0.00099	0.00098	0.00097
38	0.00116	0.00115	0.00114	0.00114	0.00113	0.00112	0.00111	0.00110	0.00109	0.00108	0.00107	0.00107	0.00106
39	0.00121	0.00120	0.00118	0.00117	0.00117	0.00116	0.00115	0.00114	0.00113	0.00112	0.00111	0.00110	0.00109
40	0.00136	0.00135	0.00134	0.00133	0.00132	0.00131	0.00130	0.00129	0.00128	0.00127	0.00126	0.00125	0.00124
41	0.00152	0.00151	0.00150	0.00148	0.00147	0.00146	0.00145	0.00144	0.00143	0.00142	0.00141	0.00140	0.00139
42	0.00167	0.00166	0.00165	0.00164	0.00163	0.00161	0.00160	0.00159	0.00158	0.00157	0.00156	0.00155	0.00154
43	0.00183	0.00182	0.00180	0.00179	0.00178	0.00177	0.00175	0.00174	0.00173	0.00172	0.00171	0.00170	0.00168
44	0.00198	0.00197	0.00196	0.00194	0.00193	0.00192	0.00191	0.00189	0.00188	0.00187	0.00186	0.00184	0.00183
45	0.00221	0.00219	0.00218	0.00216	0.00215	0.00213	0.00212	0.00210	0.00209	0.00208	0.00206	0.00205	0.00203
46	0.00243	0.00241	0.00239	0.00238	0.00236	0.00235	0.00233	0.00231	0.00230	0.00228	0.00227	0.00225	0.00223
47	0.00265	0.00263	0.00261	0.00259	0.00258	0.00256	0.00254	0.00252	0.00250	0.00249	0.00247	0.00245	0.00243
48	0.00287	0.00285	0.00283	0.00281	0.00279	0.00277	0.00275	0.00273	0.00271	0.00269	0.00267	0.00265	0.00263
49	0.00309	0.00307	0.00305	0.00302	0.00300	0.00298	0.00296	0.00294	0.00292	0.00289	0.00287	0.00285	0.00283

50	0.00340	0.00337	0.00335	0.00332	0.00330	0.00327	0.00324	0.00322	0.00319	0.00317	0.00314	0.00312	0.00309
51	0.00371	0.00368	0.00365	0.00362	0.00359	0.00356	0.00353	0.00350	0.00347	0.00344	0.00341	0.00338	0.00335
52	0.00402	0.00398	0.00395	0.00391	0.00388	0.00384	0.00381	0.00377	0.00374	0.00371	0.00368	0.00364	0.00361
53	0.00432	0.00428	0.00424	0.00420	0.00416	0.00413	0.00409	0.00405	0.00401	0.00398	0.00394	0.00390	0.00387
54	0.00462	0.00458	0.00454	0.00449	0.00445	0.00441	0.00437	0.00432	0.00428	0.00424	0.00420	0.00416	0.00412
55	0.00514	0.00509	0.00504	0.00499	0.00494	0.00489	0.00484	0.00480	0.00475	0.00470	0.00465	0.00461	0.00456
56	0.00566	0.00560	0.00554	0.00549	0.00543	0.00538	0.00532	0.00527	0.00521	0.00516	0.00511	0.00505	0.00500
57	0.00617	0.00611	0.00605	0.00598	0.00592	0.00586	0.00580	0.00573	0.00567	0.00561	0.00556	0.00550	0.00544
58	0.00669	0.00662	0.00655	0.00648	0.00641	0.00634	0.00627	0.00620	0.00614	0.00607	0.00600	0.00594	0.00588
59	0.00720	0.00712	0.00705	0.00697	0.00689	0.00682	0.00674	0.00667	0.00660	0.00652	0.00645	0.00638	0.00631
60	0.00801	0.00792	0.00782	0.00772	0.00763	0.00754	0.00744	0.00735	0.00726	0.00718	0.00709	0.00700	0.00692
61	0.00882	0.00871	0.00859	0.00848	0.00837	0.00826	0.00815	0.00804	0.00794	0.00783	0.00773	0.00763	0.00752
62	0.00963	0.00950	0.00937	0.00924	0.00911	0.00898	0.00885	0.00873	0.00861	0.00849	0.00837	0.00825	0.00814
63	0.01044	0.01029	0.01014	0.00999	0.00985	0.00970	0.00956	0.00942	0.00928	0.00915	0.00902	0.00888	0.00875
64	0.01125	0.01108	0.01092	0.01075	0.01059	0.01043	0.01027	0.01011	0.00996	0.00981	0.00966	0.00952	0.00937
65	0.01252	0.01233	0.01215	0.01197	0.01180	0.01162	0.01145	0.01128	0.01112	0.01095	0.01079	0.01063	0.01048
66	0.01378	0.01358	0.01339	0.01319	0.01300	0.01282	0.01263	0.01245	0.01227	0.01210	0.01192	0.01175	0.01158
67	0.01504	0.01483	0.01462	0.01442	0.01421	0.01401	0.01382	0.01362	0.01343	0.01324	0.01306	0.01287	0.01269
68	0.01630	0.01607	0.01585	0.01564	0.01542	0.01521	0.01500	0.01479	0.01459	0.01439	0.01419	0.01400	0.01381
69	0.01756	0.01732	0.01709	0.01686	0.01663	0.01640	0.01618	0.01597	0.01575	0.01554	0.01533	0.01512	0.01492
70	0.01933	0.01906	0.01879	0.01852	0.01826	0.01800	0.01774	0.01749	0.01724	0.01699	0.01675	0.01651	0.01628
71	0.02111	0.02080	0.02049	0.02018	0.01988	0.01959	0.01930	0.01901	0.01873	0.01845	0.01818	0.01791	0.01765
72	0.02288	0.02253	0.02219	0.02185	0.02152	0.02119	0.02086	0.02054	0.02023	0.01992	0.01962	0.01932	0.01902
73	0.02466	0.02427	0.02389	0.02352	0.02315	0.02279	0.02243	0.02208	0.02173	0.02139	0.02105	0.02072	0.02040
74	0.02644	0.02601	0.02560	0.02519	0.02478	0.02439	0.02399	0.02361	0.02323	0.02286	0.02249	0.02213	0.02178
75	0.02954	0.02907	0.02861	0.02815	0.02770	0.02726	0.02683	0.02640	0.02598	0.02556	0.02515	0.02475	0.02436
76	0.03265	0.03213	0.03162	0.03111	0.03062	0.03013	0.02965	0.02918	0.02871	0.02826	0.02781	0.02737	0.02693

77	0.03574	0.03518	0.03462	0.03407	0.03353	0.03300	0.03247	0.03196	0.03145	0.03095	0.03046	0.02998	0.02950
78	0.03884	0.03822	0.03762	0.03702	0.03644	0.03586	0.03529	0.03473	0.03418	0.03364	0.03311	0.03259	0.03207
79	0.04193	0.04126	0.04061	0.03997	0.03934	0.03872	0.03811	0.03751	0.03692	0.03633	0.03576	0.03520	0.03464
80	0.04780	0.04709	0.04639	0.04571	0.04503	0.04437	0.04371	0.04307	0.04243	0.04181	0.04119	0.04058	0.03998
81	0.05371	0.05296	0.05223	0.05150	0.05079	0.05008	0.04938	0.04870	0.04802	0.04735	0.04670	0.04605	0.04541
82	0.05965	0.05887	0.05810	0.05734	0.05658	0.05584	0.05511	0.05439	0.05367	0.05297	0.05227	0.05159	0.05091
83	0.06563	0.06481	0.06401	0.06321	0.06242	0.06165	0.06088	0.06012	0.05938	0.05864	0.05791	0.05719	0.05648
84	0.07163	0.07078	0.06994	0.06912	0.06830	0.06749	0.06669	0.06590	0.06512	0.06435	0.06359	0.06284	0.06210

Age	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044
0	0.00819	0.00805	0.00790	0.00776	0.00762	0.00748	0.00734	0.00721	0.00708	0.00695	0.00683	0.00670	0.00658
1	0.00622	0.00611	0.00600	0.00589	0.00578	0.00568	0.00558	0.00547	0.00538	0.00528	0.00518	0.00509	0.00500
2	0.00425	0.00417	0.00410	0.00402	0.00395	0.00388	0.00381	0.00374	0.00367	0.00360	0.00354	0.00347	0.00341
3	0.00228	0.00224	0.00220	0.00216	0.00212	0.00208	0.00204	0.00200	0.00196	0.00193	0.00189	0.00186	0.00182
4	0.00029	0.00029	0.00028	0.00027	0.00027	0.00026	0.00026	0.00025	0.00025	0.00024	0.00023	0.00023	0.00022
5	0.00025	0.00024	0.00024	0.00023	0.00023	0.00022	0.00022	0.00021	0.00021	0.00020	0.00020	0.00019	0.00019
6	0.00021	0.00020	0.00020	0.00019	0.00019	0.00018	0.00018	0.00017	0.00017	0.00017	0.00016	0.00016	0.00015
7	0.00016	0.00016	0.00016	0.00015	0.00015	0.00014	0.00014	0.00014	0.00013	0.00013	0.00013	0.00012	0.00012
8	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010	0.00010	0.00009	0.00009	0.00009	0.00008
9	0.00008	0.00007	0.00007	0.00007	0.00007	0.00006	0.00006	0.00006	0.00006	0.00006	0.00005	0.00005	0.00005
10	0.00008	0.00008	0.00008	0.00008	0.00007	0.00007	0.00007	0.00007	0.00006	0.00006	0.00006	0.00006	0.00006
11	0.00009	0.00009	0.00008	0.00008	0.00008	0.00008	0.00007	0.00007	0.00007	0.00007	0.00006	0.00006	0.00006
12	0.00009	0.00009	0.00009	0.00009	0.00008	0.00008	0.00008	0.00008	0.00007	0.00007	0.00007	0.00007	0.00006
13	0.00010	0.00010	0.00009	0.00009	0.00009	0.00008	0.00008	0.00008	0.00008	0.00007	0.00007	0.00007	0.00007
14	0.00010	0.00010	0.00009	0.00009	0.00009	0.00009	0.00008	0.00008	0.00008	0.00008	0.00007	0.00007	0.00007
15	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010	0.00009	0.00009	0.00009	0.00009	0.00008

16	0.00014	0.00013	0.00013	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00010	0.00010	0.00010	0.00010
17	0.00016	0.00015	0.00015	0.00014	0.00014	0.00013	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011
18	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00014	0.00014	0.00013	0.00013	0.00012	0.00012	0.00012
19	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00014	0.00014	0.00013	0.00013	0.00013
20	0.00021	0.00021	0.00020	0.00020	0.00019	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00016	0.00015
21	0.00024	0.00024	0.00023	0.00022	0.00022	0.00021	0.00021	0.00020	0.00020	0.00019	0.00019	0.00018	0.00018
22	0.00027	0.00026	0.00026	0.00025	0.00024	0.00024	0.00023	0.00023	0.00022	0.00022	0.00021	0.00021	0.00020
23	0.00030	0.00029	0.00028	0.00028	0.00027	0.00027	0.00026	0.00025	0.00025	0.00024	0.00024	0.00023	0.00023
24	0.00032	0.00032	0.00031	0.00030	0.00030	0.00029	0.00029	0.00028	0.00027	0.00027	0.00026	0.00026	0.00025
25	0.00035	0.00035	0.00034	0.00033	0.00033	0.00032	0.00031	0.00031	0.00030	0.00029	0.00029	0.00028	0.00028
26	0.00038	0.00037	0.00037	0.00036	0.00035	0.00035	0.00034	0.00033	0.00033	0.00032	0.00031	0.00031	0.00030
27	0.00041	0.00040	0.00039	0.00038	0.00038	0.00037	0.00036	0.00036	0.00035	0.00034	0.00034	0.00033	0.00033
28	0.00043	0.00042	0.00042	0.00041	0.00040	0.00040	0.00039	0.00038	0.00038	0.00037	0.00036	0.00036	0.00035
29	0.00046	0.00045	0.00044	0.00043	0.00043	0.00042	0.00041	0.00041	0.00040	0.00039	0.00039	0.00038	0.00037
30	0.00050	0.00049	0.00048	0.00047	0.00046	0.00046	0.00045	0.00044	0.00044	0.00043	0.00042	0.00042	0.00041
31	0.00053	0.00053	0.00052	0.00051	0.00050	0.00049	0.00049	0.00048	0.00047	0.00047	0.00046	0.00045	0.00044
32	0.00057	0.00056	0.00055	0.00055	0.00054	0.00053	0.00052	0.00051	0.00051	0.00050	0.00049	0.00049	0.00048
33	0.00061	0.00060	0.00059	0.00058	0.00057	0.00056	0.00056	0.00055	0.00054	0.00053	0.00053	0.00052	0.00051
34	0.00064	0.00063	0.00062	0.00062	0.00061	0.00060	0.00059	0.00058	0.00057	0.00057	0.00056	0.00055	0.00054
35	0.00078	0.00077	0.00076	0.00075	0.00074	0.00074	0.00073	0.00072	0.00071	0.00070	0.00070	0.00069	0.00068
36	0.00087	0.00086	0.00085	0.00084	0.00084	0.00083	0.00082	0.00081	0.00080	0.00080	0.00079	0.00078	0.00077
37	0.00096	0.00095	0.00094	0.00093	0.00093	0.00092	0.00091	0.00090	0.00089	0.00089	0.00088	0.00087	0.00086
38	0.00105	0.00104	0.00103	0.00102	0.00102	0.00101	0.00100	0.00099	0.00098	0.00098	0.00097	0.00096	0.00095
39	0.00108	0.00107	0.00106	0.00105	0.00104	0.00103	0.00103	0.00102	0.00101	0.00100	0.00099	0.00098	0.00098
40	0.00123	0.00122	0.00121	0.00120	0.00119	0.00118	0.00117	0.00116	0.00115	0.00114	0.00113	0.00113	0.00112
41	0.00138	0.00137	0.00136	0.00135	0.00134	0.00133	0.00132	0.00131	0.00130	0.00129	0.00128	0.00127	0.00126
42	0.00152	0.00151	0.00150	0.00149	0.00148	0.00147	0.00146	0.00145	0.00144	0.00143	0.00142	0.00141	0.00140

43	0.00167	0.00166	0.00165	0.00164	0.00163	0.00162	0.00161	0.00159	0.00158	0.00157	0.00156	0.00155	0.00154
44	0.00182	0.00181	0.00180	0.00179	0.00177	0.00176	0.00175	0.00174	0.00173	0.00172	0.00171	0.00169	0.00168
45	0.00202	0.00201	0.00199	0.00198	0.00197	0.00195	0.00194	0.00193	0.00191	0.00190	0.00189	0.00188	0.00186
46	0.00222	0.00220	0.00219	0.00217	0.00216	0.00214	0.00213	0.00211	0.00210	0.00208	0.00207	0.00206	0.00204
47	0.00242	0.00240	0.00238	0.00237	0.00235	0.00233	0.00232	0.00230	0.00228	0.00227	0.00225	0.00224	0.00222
48	0.00261	0.00260	0.00258	0.00256	0.00254	0.00252	0.00250	0.00249	0.00247	0.00245	0.00243	0.00242	0.00240
49	0.00281	0.00279	0.00277	0.00275	0.00273	0.00271	0.00269	0.00267	0.00265	0.00263	0.00261	0.00259	0.00258
50	0.00307	0.00304	0.00302	0.00300	0.00297	0.00295	0.00293	0.00290	0.00288	0.00286	0.00284	0.00281	0.00279
51	0.00332	0.00330	0.00327	0.00324	0.00321	0.00319	0.00316	0.00313	0.00311	0.00308	0.00306	0.00303	0.00301
52	0.00358	0.00355	0.00352	0.00349	0.00345	0.00342	0.00339	0.00336	0.00333	0.00330	0.00328	0.00325	0.00322
53	0.00383	0.00380	0.00376	0.00373	0.00369	0.00366	0.00363	0.00359	0.00356	0.00353	0.00349	0.00346	0.00343
54	0.00408	0.00405	0.00401	0.00397	0.00393	0.00389	0.00386	0.00382	0.00378	0.00375	0.00371	0.00368	0.00364
55	0.00452	0.00447	0.00443	0.00438	0.00434	0.00430	0.00426	0.00421	0.00417	0.00413	0.00409	0.00405	0.00401
56	0.00495	0.00490	0.00485	0.00480	0.00475	0.00470	0.00465	0.00461	0.00456	0.00451	0.00447	0.00442	0.00437
57	0.00538	0.00533	0.00527	0.00521	0.00516	0.00510	0.00505	0.00500	0.00495	0.00489	0.00484	0.00479	0.00474
58	0.00581	0.00575	0.00569	0.00563	0.00557	0.00551	0.00545	0.00539	0.00533	0.00527	0.00522	0.00516	0.00511
59	0.00624	0.00618	0.00611	0.00604	0.00598	0.00591	0.00585	0.00578	0.00572	0.00566	0.00559	0.00553	0.00547
60	0.00683	0.00675	0.00667	0.00658	0.00650	0.00642	0.00635	0.00627	0.00619	0.00612	0.00604	0.00597	0.00590
61	0.00743	0.00733	0.00723	0.00714	0.00704	0.00695	0.00686	0.00677	0.00668	0.00659	0.00650	0.00642	0.00633
62	0.00802	0.00791	0.00780	0.00769	0.00759	0.00748	0.00737	0.00727	0.00717	0.00707	0.00697	0.00687	0.00678
63	0.00863	0.00850	0.00838	0.00825	0.00813	0.00801	0.00790	0.00778	0.00767	0.00756	0.00745	0.00734	0.00723
64	0.00923	0.00909	0.00895	0.00882	0.00868	0.00855	0.00842	0.00829	0.00817	0.00805	0.00792	0.00780	0.00769
65	0.01032	0.01017	0.01002	0.00987	0.00973	0.00958	0.00944	0.00930	0.00917	0.00903	0.00890	0.00877	0.00864
66	0.01142	0.01125	0.01109	0.01093	0.01078	0.01062	0.01047	0.01032	0.01017	0.01002	0.00988	0.00974	0.00960
67	0.01252	0.01234	0.01217	0.01200	0.01183	0.01166	0.01150	0.01134	0.01118	0.01102	0.01087	0.01072	0.01057
68	0.01362	0.01343	0.01325	0.01306	0.01288	0.01271	0.01253	0.01236	0.01219	0.01202	0.01186	0.01170	0.01154
69	0.01472	0.01452	0.01433	0.01413	0.01394	0.01375	0.01357	0.01339	0.01321	0.01303	0.01285	0.01268	0.01251

70	0.01605	0.01582	0.01559	0.01537	0.01515	0.01494	0.01473	0.01452	0.01431	0.01411	0.01391	0.01371	0.01351
71	0.01738	0.01713	0.01687	0.01662	0.01638	0.01613	0.01590	0.01566	0.01543	0.01520	0.01497	0.01475	0.01453
72	0.01873	0.01844	0.01816	0.01788	0.01761	0.01734	0.01707	0.01681	0.01656	0.01630	0.01605	0.01581	0.01557
73	0.02008	0.01976	0.01945	0.01915	0.01885	0.01855	0.01826	0.01797	0.01769	0.01741	0.01714	0.01687	0.01661
74	0.02143	0.02109	0.02075	0.02042	0.02009	0.01977	0.01945	0.01914	0.01883	0.01853	0.01823	0.01794	0.01766
75	0.02397	0.02359	0.02321	0.02284	0.02247	0.02211	0.02176	0.02141	0.02107	0.02074	0.02041	0.02008	0.01976
76	0.02650	0.02608	0.02567	0.02526	0.02486	0.02446	0.02407	0.02369	0.02331	0.02294	0.02257	0.02222	0.02186
77	0.02903	0.02857	0.02812	0.02767	0.02724	0.02680	0.02638	0.02596	0.02555	0.02514	0.02474	0.02435	0.02397
78	0.03156	0.03107	0.03057	0.03009	0.02961	0.02915	0.02868	0.02823	0.02778	0.02734	0.02691	0.02649	0.02607
79	0.03409	0.03356	0.03303	0.03251	0.03199	0.03149	0.03099	0.03050	0.03002	0.02955	0.02908	0.02862	0.02817
80	0.03939	0.03881	0.03823	0.03767	0.03711	0.03657	0.03603	0.03549	0.03497	0.03445	0.03394	0.03344	0.03295
81	0.04478	0.04416	0.04354	0.04294	0.04234	0.04175	0.04117	0.04060	0.04004	0.03948	0.03893	0.03839	0.03786
82	0.05024	0.04958	0.04893	0.04829	0.04766	0.04703	0.04641	0.04581	0.04520	0.04461	0.04403	0.04345	0.04288
83	0.05577	0.05508	0.05439	0.05372	0.05305	0.05239	0.05174	0.05109	0.05046	0.04983	0.04921	0.04860	0.04800
84	0.06136	0.06063	0.05992	0.05921	0.05851	0.05782	0.05713	0.05646	0.05579	0.05513	0.05448	0.05383	0.05319

Age	2045	2046	2047	2048
0	0.00646	0.00635	0.00623	0.00612
1	0.00491	0.00482	0.00473	0.00464
2	0.00335	0.00329	0.00323	0.00317
3	0.00179	0.00176	0.00172	0.00169
4	0.00022	0.00021	0.00021	0.00020
5	0.00018	0.00018	0.00018	0.00017
6	0.00015	0.00015	0.00014	0.00014
7	0.00012	0.00011	0.00011	0.00011
8	0.00008	0.00008	0.00008	0.00008
9	0.00005	0.00005	0.00005	0.00004

10	0.00005	0.00005	0.00005	0.00005
11	0.00006	0.00006	0.00006	0.00005
12	0.00006	0.00006	0.00006	0.00006
13	0.00006	0.00006	0.00006	0.00006
14	0.00007	0.00006	0.00006	0.00006
15	0.00008	0.00008	0.00008	0.00007
16	0.00009	0.00009	0.00009	0.00008
17	0.00010	0.00010	0.00010	0.00009
18	0.00011	0.00011	0.00011	0.00010
19	0.00012	0.00012	0.00012	0.00011
20	0.00015	0.00014	0.00014	0.00013
21	0.00017	0.00017	0.00016	0.00016
22	0.00020	0.00019	0.00019	0.00018
23	0.00022	0.00022	0.00021	0.00021
24	0.00025	0.00024	0.00024	0.00023
25	0.00027	0.00027	0.00026	0.00026
26	0.00030	0.00029	0.00029	0.00028
27	0.00032	0.00031	0.00031	0.00030
28	0.00034	0.00034	0.00033	0.00033
29	0.00037	0.00036	0.00036	0.00035
30	0.00040	0.00040	0.00039	0.00038
31	0.00044	0.00043	0.00042	0.00042
32	0.00047	0.00046	0.00046	0.00045
33	0.00050	0.00050	0.00049	0.00048
34	0.00054	0.00053	0.00052	0.00051
35	0.00067	0.00067	0.00066	0.00065
36	0.00077	0.00076	0.00075	0.00074
37	0.00086	0.00085	0.00084	0.00083
38	0.00095	0.00094	0.00093	0.00092
39	0.00097	0.00096	0.00095	0.00094

40	0.00111	0.00110	0.00109	0.00108
41	0.00125	0.00124	0.00123	0.00122
42	0.00139	0.00138	0.00137	0.00136
43	0.00153	0.00152	0.00151	0.00150
44	0.00167	0.00166	0.00165	0.00164
45	0.00185	0.00184	0.00183	0.00181
46	0.00203	0.00201	0.00200	0.00199
47	0.00220	0.00219	0.00217	0.00216
48	0.00238	0.00236	0.00235	0.00233
49	0.00256	0.00254	0.00252	0.00250
50	0.00277	0.00275	0.00273	0.00270
51	0.00298	0.00295	0.00293	0.00291
52	0.00319	0.00316	0.00313	0.00311
53	0.00340	0.00337	0.00334	0.00331
54	0.00361	0.00357	0.00354	0.00351
55	0.00397	0.00393	0.00389	0.00385
56	0.00433	0.00429	0.00424	0.00420
57	0.00469	0.00464	0.00459	0.00454
58	0.00505	0.00500	0.00494	0.00489
59	0.00541	0.00535	0.00530	0.00524
60	0.00582	0.00575	0.00568	0.00561
61	0.00625	0.00617	0.00608	0.00600
62	0.00668	0.00659	0.00650	0.00641
63	0.00712	0.00702	0.00692	0.00682
64	0.00757	0.00745	0.00734	0.00723
65	0.00851	0.00839	0.00826	0.00814
66	0.00946	0.00933	0.00919	0.00906
67	0.01042	0.01027	0.01013	0.00998
68	0.01138	0.01122	0.01107	0.01091
69	0.01234	0.01217	0.01201	0.01185

70	0.01332	0.01313	0.01294	0.01276
71	0.01432	0.01411	0.01390	0.01369
72	0.01533	0.01509	0.01486	0.01464
73	0.01635	0.01609	0.01584	0.01559
74	0.01737	0.01709	0.01682	0.01655
75	0.01944	0.01913	0.01883	0.01853
76	0.02151	0.02117	0.02084	0.02050
77	0.02358	0.02321	0.02284	0.02248
78	0.02566	0.02525	0.02485	0.02446
79	0.02773	0.02729	0.02686	0.02643
80	0.03246	0.03198	0.03151	0.03104
81	0.03733	0.03681	0.03630	0.03580
82	0.04232	0.04176	0.04121	0.04067
83	0.04740	0.04681	0.04623	0.04565
84	0.05257	0.05194	0.05133	0.05072

Male Forecasted Death Rates 2019-2048

Age	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031
0	0.01219	0.01191	0.01163	0.01136	0.01110	0.01084	0.01059	0.01034	0.01010	0.00987	0.00964	0.00942	0.00920
1	0.00926	0.00904	0.00883	0.00863	0.00843	0.00823	0.00804	0.00785	0.00767	0.00749	0.00732	0.00715	0.00698
2	0.00632	0.00617	0.00603	0.00589	0.00575	0.00562	0.00549	0.00536	0.00524	0.00512	0.00500	0.00488	0.00477
3	0.00339	0.00331	0.00323	0.00315	0.00308	0.00301	0.00294	0.00287	0.00281	0.00274	0.00268	0.00261	0.00255
4	0.00044	0.00043	0.00042	0.00041	0.00040	0.00039	0.00038	0.00037	0.00036	0.00035	0.00035	0.00034	0.00033
5	0.00039	0.00038	0.00037	0.00036	0.00035	0.00034	0.00033	0.00033	0.00032	0.00031	0.00030	0.00030	0.00029
6	0.00033	0.00033	0.00032	0.00031	0.00030	0.00030	0.00029	0.00028	0.00028	0.00027	0.00026	0.00026	0.00025
7	0.00028	0.00027	0.00027	0.00026	0.00026	0.00025	0.00024	0.00024	0.00023	0.00023	0.00022	0.00021	0.00021

8	0.00023	0.00022	0.00022	0.00021	0.00021	0.00020	0.00020	0.00019	0.00019	0.00018	0.00018	0.00017	0.00017
9	0.00017	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00015	0.00014	0.00014	0.00013	0.00013	0.00013
10	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00015	0.00014	0.00014	0.00014	0.00013
11	0.00019	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00015	0.00014	0.00014	0.00014
12	0.00019	0.00019	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00015	0.00014	0.00014
13	0.00019	0.00019	0.00019	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00015	0.00014
14	0.00020	0.00019	0.00019	0.00018	0.00018	0.00017	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00015
15	0.00031	0.00030	0.00030	0.00029	0.00029	0.00028	0.00028	0.00027	0.00027	0.00026	0.00026	0.00025	0.00025
16	0.00041	0.00040	0.00040	0.00039	0.00039	0.00038	0.00038	0.00038	0.00037	0.00037	0.00036	0.00036	0.00035
17	0.00051	0.00051	0.00050	0.00050	0.00049	0.00049	0.00048	0.00048	0.00047	0.00047	0.00047	0.00046	0.00046
18	0.00061	0.00061	0.00060	0.00060	0.00060	0.00059	0.00059	0.00058	0.00058	0.00058	0.00057	0.00057	0.00056
19	0.00071	0.00071	0.00071	0.00070	0.00070	0.00070	0.00069	0.00069	0.00068	0.00068	0.00068	0.00067	0.00067
20	0.00082	0.00081	0.00081	0.00081	0.00080	0.00080	0.00080	0.00080	0.00079	0.00079	0.00079	0.00079	0.00078
21	0.00092	0.00092	0.00091	0.00091	0.00091	0.00091	0.00091	0.00091	0.00090	0.00090	0.00090	0.00090	0.00090
22	0.00102	0.00102	0.00102	0.00102	0.00102	0.00101	0.00101	0.00101	0.00101	0.00101	0.00101	0.00101	0.00101
23	0.00112	0.00112	0.00112	0.00112	0.00112	0.00112	0.00112	0.00112	0.00112	0.00112	0.00112	0.00112	0.00112
24	0.00122	0.00122	0.00122	0.00122	0.00123	0.00123	0.00123	0.00123	0.00123	0.00123	0.00123	0.00124	0.00124
25	0.00131	0.00132	0.00132	0.00132	0.00132	0.00132	0.00133	0.00133	0.00133	0.00133	0.00133	0.00134	0.00134
26	0.00141	0.00141	0.00141	0.00142	0.00142	0.00142	0.00142	0.00143	0.00143	0.00143	0.00143	0.00144	0.00144
27	0.00150	0.00151	0.00151	0.00151	0.00152	0.00152	0.00152	0.00152	0.00153	0.00153	0.00153	0.00154	0.00154
28	0.00160	0.00160	0.00161	0.00161	0.00161	0.00162	0.00162	0.00162	0.00163	0.00163	0.00164	0.00164	0.00164
29	0.00169	0.00170	0.00170	0.00171	0.00171	0.00171	0.00172	0.00172	0.00173	0.00173	0.00174	0.00174	0.00175
30	0.00177	0.00177	0.00177	0.00177	0.00177	0.00178	0.00178	0.00178	0.00178	0.00178	0.00178	0.00178	0.00178
31	0.00185	0.00185	0.00184	0.00184	0.00184	0.00184	0.00184	0.00183	0.00183	0.00183	0.00183	0.00183	0.00182
32	0.00193	0.00192	0.00192	0.00191	0.00191	0.00190	0.00190	0.00189	0.00189	0.00188	0.00188	0.00187	0.00187
33	0.00200	0.00200	0.00199	0.00198	0.00197	0.00197	0.00196	0.00195	0.00194	0.00194	0.00193	0.00192	0.00191
34	0.00208	0.00207	0.00206	0.00205	0.00204	0.00203	0.00202	0.00201	0.00200	0.00199	0.00198	0.00197	0.00196

35	0.00232	0.00231	0.00230	0.00228	0.00227	0.00226	0.00225	0.00224	0.00223	0.00222	0.00221	0.00219	0.00218
36	0.00255	0.00254	0.00253	0.00252	0.00250	0.00249	0.00248	0.00247	0.00245	0.00244	0.00243	0.00242	0.00240
37	0.00279	0.00277	0.00276	0.00275	0.00273	0.00272	0.00270	0.00269	0.00268	0.00266	0.00265	0.00264	0.00262
38	0.00302	0.00301	0.00299	0.00297	0.00296	0.00294	0.00293	0.00291	0.00290	0.00288	0.00287	0.00285	0.00284
39	0.00325	0.00323	0.00322	0.00320	0.00318	0.00317	0.00315	0.00314	0.00312	0.00310	0.00309	0.00307	0.00306
40	0.00348	0.00346	0.00344	0.00341	0.00339	0.00337	0.00334	0.00332	0.00330	0.00328	0.00325	0.00323	0.00321
41	0.00371	0.00368	0.00365	0.00362	0.00359	0.00357	0.00354	0.00351	0.00348	0.00345	0.00343	0.00340	0.00337
42	0.00394	0.00391	0.00387	0.00384	0.00380	0.00377	0.00373	0.00370	0.00367	0.00363	0.00360	0.00357	0.00353
43	0.00417	0.00413	0.00409	0.00405	0.00401	0.00397	0.00393	0.00389	0.00385	0.00381	0.00377	0.00374	0.00370
44	0.00440	0.00436	0.00431	0.00426	0.00422	0.00417	0.00413	0.00408	0.00404	0.00399	0.00395	0.00391	0.00387
45	0.00481	0.00475	0.00470	0.00465	0.00459	0.00454	0.00449	0.00444	0.00439	0.00434	0.00429	0.00424	0.00419
46	0.00521	0.00515	0.00509	0.00503	0.00497	0.00491	0.00485	0.00480	0.00474	0.00468	0.00463	0.00458	0.00452
47	0.00560	0.00554	0.00547	0.00541	0.00534	0.00528	0.00521	0.00515	0.00509	0.00503	0.00497	0.00491	0.00485
48	0.00600	0.00593	0.00586	0.00578	0.00571	0.00564	0.00557	0.00550	0.00544	0.00537	0.00530	0.00524	0.00517
49	0.00640	0.00632	0.00624	0.00616	0.00608	0.00601	0.00593	0.00586	0.00578	0.00571	0.00564	0.00557	0.00550
50	0.00702	0.00693	0.00684	0.00676	0.00667	0.00658	0.00650	0.00641	0.00633	0.00625	0.00617	0.00609	0.00601
51	0.00764	0.00754	0.00744	0.00735	0.00725	0.00715	0.00706	0.00697	0.00688	0.00679	0.00670	0.00661	0.00652
52	0.00826	0.00815	0.00804	0.00794	0.00783	0.00772	0.00762	0.00752	0.00742	0.00732	0.00722	0.00712	0.00703
53	0.00888	0.00876	0.00864	0.00852	0.00841	0.00829	0.00818	0.00807	0.00796	0.00785	0.00774	0.00764	0.00753
54	0.00949	0.00936	0.00923	0.00911	0.00898	0.00886	0.00874	0.00861	0.00850	0.00838	0.00826	0.00815	0.00804
55	0.01025	0.01010	0.00995	0.00981	0.00967	0.00953	0.00939	0.00925	0.00912	0.00899	0.00886	0.00873	0.00860
56	0.01100	0.01083	0.01067	0.01051	0.01035	0.01020	0.01004	0.00989	0.00974	0.00960	0.00945	0.00931	0.00917
57	0.01175	0.01157	0.01139	0.01121	0.01103	0.01086	0.01069	0.01053	0.01036	0.01020	0.01004	0.00989	0.00973
58	0.01249	0.01229	0.01210	0.01190	0.01171	0.01153	0.01134	0.01116	0.01098	0.01081	0.01063	0.01046	0.01030
59	0.01324	0.01302	0.01281	0.01260	0.01239	0.01219	0.01199	0.01179	0.01160	0.01141	0.01122	0.01104	0.01086
60	0.01465	0.01441	0.01418	0.01395	0.01372	0.01349	0.01327	0.01306	0.01284	0.01263	0.01243	0.01223	0.01203
61	0.01606	0.01580	0.01554	0.01529	0.01504	0.01479	0.01455	0.01432	0.01408	0.01385	0.01363	0.01341	0.01319

62	0.01746	0.01718	0.01690	0.01663	0.01636	0.01609	0.01583	0.01557	0.01532	0.01507	0.01483	0.01459	0.01435
63	0.01886	0.01856	0.01826	0.01796	0.01767	0.01738	0.01710	0.01682	0.01655	0.01628	0.01602	0.01576	0.01551
64	0.02026	0.01993	0.01961	0.01929	0.01898	0.01867	0.01837	0.01808	0.01778	0.01750	0.01721	0.01693	0.01666
65	0.02211	0.02175	0.02139	0.02104	0.02070	0.02036	0.02003	0.01971	0.01939	0.01907	0.01876	0.01845	0.01815
66	0.02395	0.02356	0.02317	0.02279	0.02242	0.02205	0.02169	0.02133	0.02098	0.02064	0.02030	0.01997	0.01964
67	0.02578	0.02536	0.02494	0.02453	0.02413	0.02373	0.02334	0.02296	0.02258	0.02221	0.02184	0.02148	0.02113
68	0.02762	0.02716	0.02671	0.02627	0.02584	0.02541	0.02499	0.02457	0.02417	0.02377	0.02338	0.02299	0.02261
69	0.02945	0.02896	0.02848	0.02800	0.02754	0.02708	0.02663	0.02619	0.02576	0.02533	0.02491	0.02449	0.02409
70	0.03235	0.03182	0.03130	0.03079	0.03029	0.02979	0.02930	0.02882	0.02835	0.02789	0.02743	0.02698	0.02654
71	0.03526	0.03468	0.03412	0.03357	0.03303	0.03249	0.03196	0.03145	0.03094	0.03044	0.02994	0.02946	0.02898
72	0.03815	0.03754	0.03694	0.03634	0.03576	0.03519	0.03462	0.03407	0.03352	0.03298	0.03245	0.03193	0.03142
73	0.04104	0.04039	0.03975	0.03912	0.03849	0.03788	0.03728	0.03669	0.03610	0.03553	0.03496	0.03441	0.03386
74	0.04393	0.04324	0.04256	0.04189	0.04122	0.04057	0.03993	0.03930	0.03868	0.03807	0.03747	0.03688	0.03630
75	0.04792	0.04715	0.04639	0.04564	0.04490	0.04418	0.04347	0.04277	0.04208	0.04140	0.04073	0.04008	0.03943
76	0.05190	0.05105	0.05022	0.04940	0.04859	0.04779	0.04701	0.04624	0.04549	0.04474	0.04401	0.04329	0.04258
77	0.05589	0.05496	0.05405	0.05316	0.05228	0.05141	0.05056	0.04972	0.04890	0.04809	0.04729	0.04651	0.04574
78	0.05988	0.05888	0.05789	0.05692	0.05597	0.05503	0.05411	0.05321	0.05232	0.05144	0.05058	0.04974	0.04891
79	0.06386	0.06279	0.06173	0.06069	0.05966	0.05866	0.05767	0.05670	0.05574	0.05480	0.05388	0.05297	0.05208
80	0.07048	0.06933	0.06820	0.06709	0.06600	0.06492	0.06387	0.06283	0.06180	0.06080	0.05981	0.05883	0.05787
81	0.07709	0.07588	0.07468	0.07350	0.07234	0.07120	0.07008	0.06897	0.06788	0.06681	0.06576	0.06472	0.06370
82	0.08371	0.08243	0.08116	0.07992	0.07869	0.07749	0.07630	0.07513	0.07398	0.07284	0.07173	0.07063	0.06955
83	0.09032	0.08898	0.08765	0.08634	0.08505	0.08378	0.08253	0.08130	0.08009	0.07889	0.07772	0.07656	0.07541
84	0.09694	0.09553	0.09414	0.09277	0.09142	0.09009	0.08878	0.08748	0.08621	0.08495	0.08372	0.08250	0.08130

Age	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044
0	0.00898	0.00878	0.00857	0.00837	0.00818	0.00799	0.00780	0.00762	0.00745	0.00727	0.00711	0.00694	0.00678

1	0.00682	0.00666	0.00651	0.00636	0.00621	0.00607	0.00593	0.00579	0.00565	0.00552	0.00539	0.00527	0.00515
2	0.00466	0.00455	0.00445	0.00434	0.00424	0.00414	0.00405	0.00395	0.00386	0.00377	0.00368	0.00360	0.00352
3	0.00250	0.00244	0.00238	0.00233	0.00227	0.00222	0.00217	0.00212	0.00207	0.00202	0.00197	0.00193	0.00188
4	0.00032	0.00032	0.00031	0.00030	0.00029	0.00029	0.00028	0.00027	0.00027	0.00026	0.00026	0.00025	0.00024
5	0.00028	0.00028	0.00027	0.00026	0.00026	0.00025	0.00025	0.00024	0.00023	0.00023	0.00022	0.00022	0.00021
6	0.00024	0.00024	0.00023	0.00023	0.00022	0.00022	0.00021	0.00021	0.00020	0.00020	0.00019	0.00019	0.00018
7	0.00020	0.00020	0.00019	0.00019	0.00019	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00016	0.00015
8	0.00016	0.00016	0.00016	0.00015	0.00015	0.00014	0.00014	0.00014	0.00013	0.00013	0.00013	0.00012	0.00012
9	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010	0.00010	0.00009	0.00009
10	0.00013	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010	0.00010	0.00009
11	0.00013	0.00013	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010	0.00010
12	0.00014	0.00013	0.00013	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010
13	0.00014	0.00014	0.00013	0.00013	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010
14	0.00014	0.00014	0.00013	0.00013	0.00013	0.00012	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010
15	0.00024	0.00024	0.00024	0.00023	0.00023	0.00022	0.00022	0.00022	0.00021	0.00021	0.00021	0.00020	0.00020
16	0.00035	0.00034	0.00034	0.00034	0.00033	0.00033	0.00032	0.00032	0.00032	0.00031	0.00031	0.00030	0.00030
17	0.00045	0.00045	0.00045	0.00044	0.00044	0.00043	0.00043	0.00043	0.00042	0.00042	0.00041	0.00041	0.00041
18	0.00056	0.00056	0.00055	0.00055	0.00054	0.00054	0.00054	0.00053	0.00053	0.00053	0.00052	0.00052	0.00052
19	0.00067	0.00066	0.00066	0.00066	0.00065	0.00065	0.00065	0.00064	0.00064	0.00064	0.00063	0.00063	0.00063
20	0.00078	0.00078	0.00078	0.00077	0.00077	0.00077	0.00076	0.00076	0.00076	0.00076	0.00075	0.00075	0.00075
21	0.00089	0.00089	0.00089	0.00089	0.00089	0.00089	0.00088	0.00088	0.00088	0.00088	0.00088	0.00087	0.00087
22	0.00101	0.00101	0.00101	0.00101	0.00101	0.00100	0.00100	0.00100	0.00100	0.00100	0.00100	0.00100	0.00100
23	0.00112	0.00112	0.00112	0.00112	0.00113	0.00113	0.00113	0.00113	0.00113	0.00113	0.00113	0.00113	0.00113
24	0.00124	0.00124	0.00124	0.00124	0.00125	0.00125	0.00125	0.00125	0.00125	0.00125	0.00125	0.00126	0.00126
25	0.00134	0.00134	0.00134	0.00134	0.00135	0.00135	0.00135	0.00135	0.00135	0.00136	0.00136	0.00136	0.00136
26	0.00144	0.00144	0.00145	0.00145	0.00145	0.00145	0.00146	0.00146	0.00146	0.00146	0.00147	0.00147	0.00147
27	0.00154	0.00155	0.00155	0.00155	0.00156	0.00156	0.00156	0.00157	0.00157	0.00157	0.00157	0.00158	0.00158

28	0.00165	0.00165	0.00165	0.00166	0.00166	0.00167	0.00167	0.00167	0.00168	0.00168	0.00169	0.00169	0.00169
29	0.00175	0.00176	0.00176	0.00176	0.00177	0.00177	0.00178	0.00178	0.00179	0.00179	0.00180	0.00180	0.00181
30	0.00178	0.00179	0.00179	0.00179	0.00179	0.00179	0.00179	0.00179	0.00179	0.00179	0.00180	0.00180	0.00180
31	0.00182	0.00182	0.00182	0.00182	0.00181	0.00181	0.00181	0.00181	0.00181	0.00180	0.00180	0.00180	0.00180
32	0.00186	0.00186	0.00185	0.00185	0.00184	0.00184	0.00183	0.00183	0.00182	0.00182	0.00182	0.00181	0.00181
33	0.00191	0.00190	0.00189	0.00188	0.00188	0.00187	0.00186	0.00186	0.00185	0.00184	0.00183	0.00183	0.00182
34	0.00195	0.00194	0.00193	0.00192	0.00191	0.00190	0.00189	0.00188	0.00187	0.00186	0.00186	0.00185	0.00184
35	0.00217	0.00216	0.00215	0.00214	0.00213	0.00212	0.00211	0.00210	0.00209	0.00208	0.00207	0.00206	0.00205
36	0.00239	0.00238	0.00237	0.00236	0.00234	0.00233	0.00232	0.00231	0.00230	0.00229	0.00227	0.00226	0.00225
37	0.00261	0.00260	0.00258	0.00257	0.00256	0.00254	0.00253	0.00252	0.00250	0.00249	0.00248	0.00247	0.00245
38	0.00282	0.00281	0.00280	0.00278	0.00277	0.00275	0.00274	0.00272	0.00271	0.00270	0.00268	0.00267	0.00266
39	0.00304	0.00302	0.00301	0.00299	0.00298	0.00296	0.00295	0.00293	0.00292	0.00290	0.00289	0.00287	0.00286
40	0.00319	0.00317	0.00315	0.00312	0.00310	0.00308	0.00306	0.00304	0.00302	0.00300	0.00298	0.00296	0.00294
41	0.00334	0.00332	0.00329	0.00326	0.00324	0.00321	0.00319	0.00316	0.00313	0.00311	0.00308	0.00306	0.00304
42	0.00350	0.00347	0.00344	0.00341	0.00338	0.00335	0.00331	0.00328	0.00325	0.00323	0.00320	0.00317	0.00314
43	0.00366	0.00363	0.00359	0.00355	0.00352	0.00348	0.00345	0.00341	0.00338	0.00334	0.00331	0.00328	0.00325
44	0.00382	0.00378	0.00374	0.00370	0.00366	0.00362	0.00358	0.00354	0.00351	0.00347	0.00343	0.00339	0.00336
45	0.00415	0.00410	0.00405	0.00401	0.00396	0.00392	0.00387	0.00383	0.00379	0.00375	0.00370	0.00366	0.00362
46	0.00447	0.00442	0.00437	0.00432	0.00426	0.00422	0.00417	0.00412	0.00407	0.00402	0.00397	0.00393	0.00388
47	0.00479	0.00473	0.00468	0.00462	0.00457	0.00451	0.00446	0.00440	0.00435	0.00430	0.00425	0.00420	0.00414
48	0.00511	0.00505	0.00499	0.00492	0.00486	0.00480	0.00475	0.00469	0.00463	0.00457	0.00452	0.00446	0.00441
49	0.00543	0.00536	0.00529	0.00523	0.00516	0.00510	0.00503	0.00497	0.00491	0.00485	0.00479	0.00473	0.00467
50	0.00593	0.00586	0.00578	0.00571	0.00563	0.00556	0.00549	0.00542	0.00535	0.00528	0.00521	0.00515	0.00508
51	0.00644	0.00635	0.00627	0.00618	0.00610	0.00602	0.00594	0.00587	0.00579	0.00571	0.00564	0.00556	0.00549
52	0.00693	0.00684	0.00675	0.00666	0.00657	0.00648	0.00640	0.00631	0.00623	0.00614	0.00606	0.00598	0.00590
53	0.00743	0.00733	0.00723	0.00713	0.00704	0.00694	0.00685	0.00675	0.00666	0.00657	0.00648	0.00639	0.00631
54	0.00793	0.00782	0.00771	0.00760	0.00750	0.00740	0.00729	0.00719	0.00709	0.00700	0.00690	0.00681	0.00671

55	0.00848	0.00836	0.00824	0.00812	0.00800	0.00788	0.00777	0.00766	0.00755	0.00744	0.00733	0.00722	0.00712
56	0.00903	0.00889	0.00876	0.00863	0.00850	0.00837	0.00824	0.00812	0.00800	0.00788	0.00776	0.00764	0.00753
57	0.00958	0.00943	0.00928	0.00914	0.00900	0.00886	0.00872	0.00858	0.00845	0.00832	0.00819	0.00806	0.00794
58	0.01013	0.00997	0.00981	0.00965	0.00950	0.00935	0.00920	0.00905	0.00890	0.00876	0.00862	0.00848	0.00835
59	0.01068	0.01051	0.01033	0.01016	0.01000	0.00983	0.00967	0.00951	0.00936	0.00921	0.00905	0.00891	0.00876
60	0.01183	0.01164	0.01145	0.01126	0.01108	0.01089	0.01072	0.01054	0.01037	0.01020	0.01003	0.00987	0.00971
61	0.01297	0.01276	0.01256	0.01235	0.01215	0.01195	0.01176	0.01157	0.01138	0.01119	0.01101	0.01083	0.01065
62	0.01412	0.01389	0.01366	0.01344	0.01322	0.01301	0.01280	0.01259	0.01238	0.01218	0.01198	0.01179	0.01160
63	0.01525	0.01501	0.01476	0.01453	0.01429	0.01406	0.01383	0.01361	0.01339	0.01317	0.01296	0.01275	0.01254
64	0.01639	0.01613	0.01587	0.01561	0.01536	0.01511	0.01486	0.01462	0.01439	0.01416	0.01393	0.01370	0.01348
65	0.01786	0.01757	0.01728	0.01700	0.01672	0.01645	0.01618	0.01592	0.01566	0.01541	0.01515	0.01491	0.01466
66	0.01932	0.01900	0.01869	0.01839	0.01809	0.01779	0.01750	0.01721	0.01693	0.01665	0.01638	0.01611	0.01585
67	0.02078	0.02044	0.02010	0.01977	0.01944	0.01912	0.01881	0.01850	0.01819	0.01789	0.01760	0.01731	0.01703
68	0.02223	0.02187	0.02151	0.02115	0.02080	0.02046	0.02012	0.01978	0.01946	0.01914	0.01882	0.01851	0.01820
69	0.02369	0.02329	0.02291	0.02253	0.02215	0.02179	0.02142	0.02107	0.02072	0.02038	0.02004	0.01970	0.01938
70	0.02610	0.02567	0.02525	0.02484	0.02443	0.02403	0.02364	0.02325	0.02287	0.02250	0.02213	0.02177	0.02141
71	0.02851	0.02805	0.02760	0.02715	0.02671	0.02628	0.02585	0.02543	0.02502	0.02462	0.02422	0.02382	0.02344
72	0.03092	0.03042	0.02993	0.02945	0.02898	0.02852	0.02806	0.02761	0.02717	0.02673	0.02630	0.02588	0.02546
73	0.03332	0.03279	0.03227	0.03176	0.03125	0.03075	0.03027	0.02978	0.02931	0.02884	0.02838	0.02793	0.02749
74	0.03572	0.03516	0.03460	0.03406	0.03352	0.03299	0.03247	0.03196	0.03145	0.03096	0.03047	0.02999	0.02951
75	0.03880	0.03817	0.03756	0.03695	0.03636	0.03577	0.03519	0.03463	0.03407	0.03352	0.03298	0.03245	0.03193
76	0.04188	0.04120	0.04053	0.03986	0.03921	0.03857	0.03794	0.03732	0.03671	0.03611	0.03551	0.03493	0.03436
77	0.04498	0.04424	0.04351	0.04278	0.04208	0.04138	0.04069	0.04002	0.03936	0.03871	0.03806	0.03743	0.03681
78	0.04809	0.04728	0.04649	0.04572	0.04495	0.04420	0.04346	0.04273	0.04202	0.04132	0.04063	0.03995	0.03928
79	0.05120	0.05034	0.04949	0.04865	0.04783	0.04703	0.04623	0.04545	0.04469	0.04393	0.04319	0.04247	0.04175
80	0.05693	0.05601	0.05509	0.05420	0.05331	0.05245	0.05159	0.05075	0.04992	0.04911	0.04831	0.04753	0.04675
81	0.06269	0.06170	0.06073	0.05977	0.05883	0.05790	0.05699	0.05609	0.05520	0.05433	0.05348	0.05263	0.05180

82	0.06848	0.06743	0.06640	0.06538	0.06438	0.06339	0.06242	0.06146	0.06052	0.05959	0.05868	0.05778	0.05689
83	0.07429	0.07318	0.07209	0.07101	0.06995	0.06891	0.06788	0.06687	0.06587	0.06489	0.06392	0.06296	0.06202
84	0.08011	0.07895	0.07780	0.07667	0.07555	0.07445	0.07337	0.07230	0.07125	0.07021	0.06919	0.06818	0.06719

Age	2045	2046	2047	2048
0	0.00662	0.00647	0.00632	0.00617
1	0.00503	0.00491	0.00480	0.00469
2	0.00343	0.00335	0.00328	0.00320
3	0.00184	0.00180	0.00175	0.00171
4	0.00024	0.00023	0.00023	0.00022
5	0.00021	0.00020	0.00020	0.00019
6	0.00018	0.00017	0.00017	0.00017
7	0.00015	0.00014	0.00014	0.00014
8	0.00012	0.00012	0.00011	0.00011
9	0.00009	0.00009	0.00008	0.00008
10	0.00009	0.00009	0.00009	0.00008
11	0.00009	0.00009	0.00009	0.00009
12	0.00010	0.00009	0.00009	0.00009
13	0.00010	0.00010	0.00009	0.00009
14	0.00010	0.00010	0.00010	0.00009
15	0.00020	0.00019	0.00019	0.00019
16	0.00030	0.00029	0.00029	0.00029
17	0.00040	0.00040	0.00039	0.00039
18	0.00051	0.00051	0.00050	0.00050
19	0.00062	0.00062	0.00062	0.00061
20	0.00075	0.00074	0.00074	0.00074
21	0.00087	0.00087	0.00087	0.00087
22	0.00100	0.00100	0.00100	0.00100
23	0.00113	0.00113	0.00113	0.00113

24	0.00126	0.00126	0.00126	0.00126
25	0.00136	0.00137	0.00137	0.00137
26	0.00147	0.00148	0.00148	0.00148
27	0.00158	0.00159	0.00159	0.00159
28	0.00170	0.00170	0.00171	0.00171
29	0.00181	0.00182	0.00182	0.00183
30	0.00180	0.00180	0.00180	0.00180
31	0.00180	0.00179	0.00179	0.00179
32	0.00180	0.00180	0.00179	0.00179
33	0.00181	0.00181	0.00180	0.00179
34	0.00183	0.00182	0.00181	0.00180
35	0.00204	0.00203	0.00202	0.00201
36	0.00224	0.00223	0.00222	0.00221
37	0.00244	0.00243	0.00242	0.00240
38	0.00264	0.00263	0.00261	0.00260
39	0.00284	0.00283	0.00281	0.00280
40	0.00292	0.00290	0.00288	0.00286
41	0.00301	0.00299	0.00296	0.00294
42	0.00311	0.00308	0.00305	0.00303
43	0.00321	0.00318	0.00315	0.00312
44	0.00332	0.00328	0.00325	0.00321
45	0.00358	0.00354	0.00350	0.00346
46	0.00384	0.00379	0.00375	0.00370
47	0.00410	0.00405	0.00400	0.00395
48	0.00435	0.00430	0.00425	0.00419
49	0.00461	0.00455	0.00449	0.00444
50	0.00501	0.00495	0.00489	0.00482
51	0.00542	0.00535	0.00528	0.00521
52	0.00582	0.00574	0.00566	0.00559
53	0.00622	0.00614	0.00605	0.00597

54	0.00662	0.00653	0.00644	0.00635
55	0.00702	0.00691	0.00681	0.00671
56	0.00741	0.00730	0.00719	0.00708
57	0.00781	0.00769	0.00757	0.00745
58	0.00821	0.00808	0.00795	0.00783
59	0.00862	0.00848	0.00834	0.00820
60	0.00955	0.00939	0.00924	0.00909
61	0.01048	0.01031	0.01014	0.00998
62	0.01141	0.01122	0.01104	0.01086
63	0.01234	0.01214	0.01194	0.01175
64	0.01326	0.01305	0.01284	0.01263
65	0.01443	0.01419	0.01396	0.01373
66	0.01559	0.01533	0.01508	0.01483
67	0.01674	0.01647	0.01620	0.01593
68	0.01790	0.01760	0.01731	0.01703
69	0.01906	0.01874	0.01843	0.01812
70	0.02106	0.02071	0.02037	0.02004
71	0.02306	0.02269	0.02232	0.02196
72	0.02506	0.02465	0.02426	0.02387
73	0.02705	0.02662	0.02620	0.02578
74	0.02905	0.02859	0.02814	0.02769
75	0.03141	0.03090	0.03041	0.02992
76	0.03380	0.03325	0.03270	0.03217
77	0.03620	0.03560	0.03501	0.03443
78	0.03862	0.03797	0.03734	0.03672
79	0.04105	0.04035	0.03967	0.03900
80	0.04599	0.04524	0.04450	0.04378
81	0.05098	0.05018	0.04939	0.04861
82	0.05602	0.05516	0.05432	0.05349
83	0.06110	0.06019	0.05929	0.05840

84	0.06621	0.06524	0.06429	0.06336

Total Forecasted Death Rates 2019-2048

Age	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031
0	0.01135	0.01112	0.01088	0.01066	0.01043	0.01021	0.01000	0.00979	0.00959	0.00938	0.00919	0.00900	0.00881
1	0.00876	0.00858	0.00841	0.00824	0.00807	0.00791	0.00775	0.00759	0.00743	0.00728	0.00713	0.00699	0.00685
2	0.00622	0.00610	0.00598	0.00587	0.00575	0.00565	0.00554	0.00543	0.00533	0.00523	0.00513	0.00503	0.00493
3	0.00351	0.00345	0.00339	0.00333	0.00327	0.00321	0.00316	0.00310	0.00305	0.00300	0.00295	0.00289	0.00284
4	0.00042	0.00041	0.00040	0.00039	0.00038	0.00037	0.00036	0.00036	0.00035	0.00034	0.00033	0.00033	0.00032
5	0.00037	0.00036	0.00035	0.00034	0.00033	0.00033	0.00032	0.00031	0.00030	0.00030	0.00029	0.00028	0.00028
6	0.00031	0.00031	0.00030	0.00029	0.00028	0.00028	0.00027	0.00026	0.00026	0.00025	0.00025	0.00024	0.00023
7	0.00026	0.00025	0.00025	0.00024	0.00024	0.00023	0.00022	0.00022	0.00021	0.00021	0.00020	0.00020	0.00019
8	0.00021	0.00020	0.00020	0.00019	0.00019	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015
9	0.00015	0.00015	0.00014	0.00014	0.00013	0.00013	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011
10	0.00016	0.00015	0.00015	0.00015	0.00014	0.00014	0.00013	0.00013	0.00013	0.00012	0.00012	0.00012	0.00011
11	0.00017	0.00016	0.00016	0.00015	0.00015	0.00014	0.00014	0.00014	0.00013	0.00013	0.00013	0.00012	0.00012
12	0.00017	0.00017	0.00016	0.00016	0.00016	0.00015	0.00015	0.00014	0.00014	0.00014	0.00013	0.00013	0.00012
13	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00015	0.00014	0.00014	0.00013	0.00013
14	0.00019	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00015	0.00014	0.00014	0.00014
15	0.00025	0.00024	0.00024	0.00023	0.00023	0.00022	0.00022	0.00021	0.00021	0.00021	0.00020	0.00020	0.00019
16	0.00031	0.00031	0.00030	0.00029	0.00029	0.00028	0.00028	0.00027	0.00027	0.00026	0.00026	0.00025	0.00025
17	0.00037	0.00037	0.00036	0.00035	0.00035	0.00034	0.00034	0.00033	0.00033	0.00032	0.00031	0.00031	0.00030
18	0.00043	0.00043	0.00042	0.00041	0.00041	0.00040	0.00039	0.00039	0.00038	0.00038	0.00037	0.00036	0.00036
19	0.00050	0.00049	0.00048	0.00047	0.00047	0.00046	0.00045	0.00045	0.00044	0.00043	0.00043	0.00042	0.00041

20	0.00056	0.00055	0.00055	0.00054	0.00053	0.00053	0.00052	0.00051	0.00051	0.00050	0.00049	0.00049	0.00048
21	0.00062	0.00062	0.00061	0.00060	0.00060	0.00059	0.00058	0.00058	0.00057	0.00057	0.00056	0.00055	0.00055
22	0.00069	0.00068	0.00068	0.00067	0.00066	0.00066	0.00065	0.00065	0.00064	0.00063	0.00063	0.00062	0.00062
23	0.00075	0.00075	0.00074	0.00074	0.00073	0.00072	0.00072	0.00071	0.00071	0.00070	0.00070	0.00069	0.00068
24	0.00082	0.00081	0.00081	0.00080	0.00080	0.00079	0.00079	0.00078	0.00077	0.00077	0.00076	0.00076	0.00075
25	0.00088	0.00088	0.00087	0.00087	0.00086	0.00085	0.00085	0.00084	0.00084	0.00083	0.00083	0.00082	0.00082
26	0.00094	0.00094	0.00093	0.00093	0.00092	0.00092	0.00091	0.00091	0.00090	0.00090	0.00089	0.00089	0.00088
27	0.00101	0.00100	0.00100	0.00099	0.00099	0.00098	0.00098	0.00097	0.00097	0.00096	0.00096	0.00095	0.00095
28	0.00107	0.00106	0.00106	0.00105	0.00105	0.00105	0.00104	0.00104	0.00103	0.00103	0.00102	0.00102	0.00102
29	0.00113	0.00112	0.00112	0.00112	0.00111	0.00111	0.00111	0.00110	0.00110	0.00109	0.00109	0.00109	0.00108
30	0.00119	0.00119	0.00118	0.00118	0.00117	0.00117	0.00116	0.00115	0.00115	0.00114	0.00114	0.00113	0.00113
31	0.00125	0.00125	0.00124	0.00123	0.00123	0.00122	0.00121	0.00121	0.00120	0.00119	0.00119	0.00118	0.00117
32	0.00132	0.00131	0.00130	0.00129	0.00128	0.00128	0.00127	0.00126	0.00125	0.00124	0.00124	0.00123	0.00122
33	0.00138	0.00137	0.00136	0.00135	0.00134	0.00133	0.00132	0.00131	0.00130	0.00129	0.00128	0.00128	0.00127
34	0.00144	0.00143	0.00142	0.00141	0.00139	0.00138	0.00137	0.00136	0.00135	0.00134	0.00133	0.00132	0.00131
35	0.00160	0.00159	0.00158	0.00157	0.00156	0.00155	0.00154	0.00153	0.00152	0.00151	0.00149	0.00148	0.00147
36	0.00177	0.00176	0.00175	0.00174	0.00172	0.00171	0.00170	0.00169	0.00168	0.00167	0.00166	0.00164	0.00163
37	0.00194	0.00192	0.00191	0.00190	0.00189	0.00187	0.00186	0.00185	0.00184	0.00183	0.00182	0.00180	0.00179
38	0.00210	0.00209	0.00207	0.00206	0.00205	0.00204	0.00202	0.00201	0.00200	0.00199	0.00198	0.00196	0.00195
39	0.00225	0.00224	0.00223	0.00221	0.00220	0.00219	0.00217	0.00216	0.00215	0.00213	0.00212	0.00211	0.00210
40	0.00245	0.00243	0.00242	0.00240	0.00238	0.00237	0.00235	0.00234	0.00232	0.00230	0.00229	0.00227	0.00226
41	0.00265	0.00263	0.00261	0.00259	0.00257	0.00255	0.00253	0.00251	0.00249	0.00247	0.00245	0.00244	0.00242
42	0.00284	0.00282	0.00280	0.00278	0.00275	0.00273	0.00271	0.00269	0.00267	0.00264	0.00262	0.00260	0.00258
43	0.00304	0.00301	0.00299	0.00296	0.00294	0.00291	0.00289	0.00286	0.00284	0.00281	0.00279	0.00277	0.00274
44	0.00324	0.00321	0.00318	0.00315	0.00312	0.00309	0.00307	0.00304	0.00301	0.00298	0.00296	0.00293	0.00290
45	0.00355	0.00351	0.00348	0.00345	0.00342	0.00338	0.00335	0.00332	0.00329	0.00326	0.00323	0.00320	0.00317
46	0.00386	0.00382	0.00378	0.00375	0.00371	0.00367	0.00364	0.00360	0.00357	0.00353	0.00350	0.00347	0.00343

47	0.00417	0.00413	0.00409	0.00404	0.00400	0.00396	0.00392	0.00389	0.00385	0.00381	0.00377	0.00373	0.00370
48	0.00448	0.00443	0.00439	0.00434	0.00430	0.00425	0.00421	0.00417	0.00412	0.00408	0.00404	0.00400	0.00396
49	0.00479	0.00474	0.00469	0.00464	0.00459	0.00454	0.00449	0.00445	0.00440	0.00436	0.00431	0.00426	0.00422
50	0.00524	0.00519	0.00513	0.00507	0.00502	0.00496	0.00491	0.00485	0.00480	0.00475	0.00470	0.00464	0.00459
51	0.00570	0.00563	0.00557	0.00551	0.00544	0.00538	0.00532	0.00526	0.00520	0.00514	0.00508	0.00502	0.00497
52	0.00615	0.00608	0.00601	0.00594	0.00587	0.00580	0.00573	0.00566	0.00559	0.00553	0.00546	0.00540	0.00534
53	0.00660	0.00652	0.00644	0.00637	0.00629	0.00621	0.00614	0.00606	0.00599	0.00592	0.00585	0.00578	0.00571
54	0.00705	0.00697	0.00688	0.00680	0.00671	0.00663	0.00655	0.00647	0.00639	0.00631	0.00623	0.00615	0.00608
55	0.00767	0.00757	0.00747	0.00737	0.00728	0.00718	0.00709	0.00700	0.00691	0.00682	0.00673	0.00664	0.00655
56	0.00828	0.00817	0.00806	0.00795	0.00784	0.00774	0.00763	0.00753	0.00743	0.00733	0.00723	0.00713	0.00703
57	0.00889	0.00877	0.00865	0.00853	0.00841	0.00829	0.00817	0.00806	0.00795	0.00783	0.00773	0.00762	0.00751
58	0.00951	0.00937	0.00923	0.00910	0.00897	0.00884	0.00871	0.00859	0.00846	0.00834	0.00822	0.00810	0.00799
59	0.01012	0.00997	0.00982	0.00967	0.00953	0.00939	0.00925	0.00912	0.00898	0.00885	0.00872	0.00859	0.00846
60	0.01118	0.01101	0.01084	0.01067	0.01051	0.01035	0.01019	0.01004	0.00989	0.00974	0.00959	0.00944	0.00930
61	0.01224	0.01204	0.01186	0.01167	0.01149	0.01131	0.01113	0.01096	0.01079	0.01062	0.01046	0.01029	0.01013
62	0.01329	0.01308	0.01287	0.01267	0.01247	0.01227	0.01207	0.01188	0.01169	0.01151	0.01132	0.01114	0.01097
63	0.01435	0.01412	0.01389	0.01366	0.01344	0.01323	0.01301	0.01280	0.01259	0.01239	0.01219	0.01199	0.01180
64	0.01541	0.01515	0.01490	0.01466	0.01442	0.01418	0.01395	0.01372	0.01350	0.01327	0.01306	0.01284	0.01263
65	0.01693	0.01665	0.01638	0.01612	0.01586	0.01560	0.01535	0.01510	0.01486	0.01462	0.01438	0.01415	0.01392
66	0.01844	0.01815	0.01786	0.01758	0.01730	0.01702	0.01675	0.01648	0.01622	0.01596	0.01571	0.01546	0.01521
67	0.01996	0.01964	0.01934	0.01903	0.01873	0.01844	0.01815	0.01786	0.01758	0.01731	0.01703	0.01677	0.01650
68	0.02147	0.02114	0.02081	0.02048	0.02017	0.01985	0.01954	0.01924	0.01894	0.01865	0.01836	0.01807	0.01779
69	0.02298	0.02263	0.02228	0.02194	0.02160	0.02127	0.02094	0.02062	0.02030	0.01998	0.01968	0.01937	0.01907
70	0.02520	0.02481	0.02442	0.02404	0.02367	0.02330	0.02294	0.02258	0.02223	0.02188	0.02154	0.02120	0.02087
71	0.02742	0.02699	0.02656	0.02615	0.02573	0.02533	0.02493	0.02454	0.02415	0.02377	0.02340	0.02303	0.02267
72	0.02963	0.02916	0.02870	0.02825	0.02780	0.02736	0.02692	0.02650	0.02608	0.02566	0.02526	0.02486	0.02446
73	0.03185	0.03134	0.03084	0.03035	0.02986	0.02939	0.02892	0.02846	0.02800	0.02756	0.02712	0.02668	0.02626

74	0.03406	0.03351	0.03298	0.03245	0.03193	0.03141	0.03091	0.03041	0.02993	0.02945	0.02897	0.02851	0.02805
75	0.03748	0.03688	0.03629	0.03571	0.03514	0.03457	0.03402	0.03347	0.03294	0.03241	0.03189	0.03138	0.03088
76	0.04090	0.04025	0.03960	0.03897	0.03835	0.03773	0.03713	0.03654	0.03595	0.03538	0.03481	0.03425	0.03370
77	0.04432	0.04361	0.04292	0.04223	0.04156	0.04089	0.04024	0.03960	0.03896	0.03834	0.03773	0.03712	0.03653
78	0.04774	0.04698	0.04623	0.04549	0.04476	0.04405	0.04335	0.04265	0.04197	0.04130	0.04064	0.04000	0.03936
79	0.05115	0.05034	0.04954	0.04875	0.04797	0.04721	0.04645	0.04571	0.04499	0.04427	0.04356	0.04287	0.04219
80	0.05716	0.05630	0.05545	0.05461	0.05379	0.05298	0.05218	0.05139	0.05062	0.04985	0.04910	0.04836	0.04763
81	0.06319	0.06228	0.06139	0.06051	0.05965	0.05879	0.05795	0.05712	0.05630	0.05550	0.05470	0.05392	0.05315
82	0.06925	0.06830	0.06737	0.06645	0.06554	0.06465	0.06377	0.06290	0.06204	0.06119	0.06036	0.05953	0.05872
83	0.07533	0.07434	0.07337	0.07242	0.07147	0.07054	0.06962	0.06871	0.06781	0.06693	0.06605	0.06519	0.06434
84	0.08143	0.08041	0.07940	0.07841	0.07743	0.07646	0.07550	0.07456	0.07362	0.07270	0.07179	0.07089	0.07001

Age	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044
0	0.00862	0.00844	0.00827	0.00809	0.00792	0.00776	0.00759	0.00743	0.00728	0.00713	0.00698	0.00683	0.00669
1	0.00671	0.00657	0.00644	0.00631	0.00618	0.00605	0.00593	0.00581	0.00569	0.00558	0.00546	0.00535	0.00524
2	0.00484	0.00475	0.00466	0.00457	0.00448	0.00440	0.00431	0.00423	0.00415	0.00407	0.00399	0.00392	0.00384
3	0.00280	0.00275	0.00270	0.00265	0.00261	0.00256	0.00252	0.00247	0.00243	0.00239	0.00235	0.00231	0.00227
4	0.00031	0.00030	0.00030	0.00029	0.00028	0.00028	0.00027	0.00026	0.00026	0.00025	0.00025	0.00024	0.00024
5	0.00027	0.00026	0.00026	0.00025	0.00025	0.00024	0.00023	0.00023	0.00022	0.00022	0.00021	0.00021	0.00020
6	0.00023	0.00022	0.00022	0.00021	0.00021	0.00020	0.00020	0.00019	0.00019	0.00018	0.00018	0.00018	0.00017
7	0.00019	0.00018	0.00018	0.00017	0.00017	0.00017	0.00016	0.00016	0.00015	0.00015	0.00015	0.00014	0.00014
8	0.00015	0.00014	0.00014	0.00014	0.00013	0.00013	0.00012	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011
9	0.00010	0.00010	0.00010	0.00010	0.00009	0.00009	0.00009	0.00009	0.00008	0.00008	0.00008	0.00008	0.00007
10	0.00011	0.00011	0.00010	0.00010	0.00010	0.00010	0.00009	0.00009	0.00009	0.00009	0.00008	0.00008	0.00008
11	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010	0.00010	0.00009	0.00009	0.00009	0.00009	0.00008
12	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010	0.00010	0.00009	0.00009	0.00009
13	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010	0.00009	0.00009
14	0.00013	0.00013	0.00013	0.00012	0.00012	0.00012	0.00011	0.00011	0.00011	0.00010	0.00010	0.00010	0.00010

15	0.00019	0.00018	0.00018	0.00018	0.00017	0.00017	0.00016	0.00016	0.00016	0.00015	0.00015	0.00015	0.00014
16	0.00024	0.00024	0.00023	0.00023	0.00022	0.00022	0.00022	0.00021	0.00021	0.00020	0.00020	0.00020	0.00019
17	0.00030	0.00029	0.00029	0.00028	0.00028	0.00027	0.00027	0.00026	0.00026	0.00025	0.00025	0.00025	0.00024
18	0.00035	0.00035	0.00034	0.00034	0.00033	0.00033	0.00032	0.00032	0.00031	0.00031	0.00030	0.00030	0.00029
19	0.00041	0.00040	0.00040	0.00039	0.00038	0.00038	0.00037	0.00037	0.00036	0.00036	0.00035	0.00035	0.00034
20	0.00047	0.00047	0.00046	0.00046	0.00045	0.00044	0.00044	0.00043	0.00043	0.00042	0.00042	0.00041	0.00041
21	0.00054	0.00054	0.00053	0.00052	0.00052	0.00051	0.00051	0.00050	0.00050	0.00049	0.00048	0.00048	0.00047
22	0.00061	0.00060	0.00060	0.00059	0.00059	0.00058	0.00058	0.00057	0.00056	0.00056	0.00055	0.00055	0.00054
23	0.00068	0.00067	0.00067	0.00066	0.00066	0.00065	0.00065	0.00064	0.00064	0.00063	0.00063	0.00062	0.00062
24	0.00075	0.00074	0.00074	0.00073	0.00073	0.00072	0.00072	0.00071	0.00071	0.00070	0.00070	0.00069	0.00069
25	0.00081	0.00081	0.00080	0.00080	0.00079	0.00079	0.00078	0.00078	0.00078	0.00077	0.00077	0.00076	0.00076
26	0.00088	0.00087	0.00087	0.00087	0.00086	0.00086	0.00085	0.00085	0.00084	0.00084	0.00083	0.00083	0.00082
27	0.00095	0.00094	0.00094	0.00093	0.00093	0.00092	0.00092	0.00092	0.00091	0.00091	0.00090	0.00090	0.00089
28	0.00101	0.00101	0.00100	0.00100	0.00100	0.00099	0.00099	0.00098	0.00098	0.00098	0.00097	0.00097	0.00096
29	0.00108	0.00108	0.00107	0.00107	0.00106	0.00106	0.00106	0.00105	0.00105	0.00105	0.00104	0.00104	0.00104
30	0.00112	0.00112	0.00111	0.00111	0.00110	0.00110	0.00109	0.00109	0.00108	0.00108	0.00107	0.00107	0.00106
31	0.00117	0.00116	0.00116	0.00115	0.00114	0.00114	0.00113	0.00112	0.00112	0.00111	0.00111	0.00110	0.00109
32	0.00121	0.00121	0.00120	0.00119	0.00118	0.00118	0.00117	0.00116	0.00115	0.00115	0.00114	0.00113	0.00113
33	0.00126	0.00125	0.00124	0.00123	0.00122	0.00121	0.00121	0.00120	0.00119	0.00118	0.00117	0.00117	0.00116
34	0.00130	0.00129	0.00128	0.00127	0.00126	0.00125	0.00124	0.00124	0.00123	0.00122	0.00121	0.00120	0.00119
35	0.00146	0.00145	0.00144	0.00143	0.00142	0.00141	0.00140	0.00139	0.00138	0.00137	0.00136	0.00135	0.00134
36	0.00162	0.00161	0.00160	0.00159	0.00158	0.00157	0.00156	0.00155	0.00154	0.00153	0.00152	0.00151	0.00150
37	0.00178	0.00177	0.00176	0.00175	0.00174	0.00173	0.00171	0.00170	0.00169	0.00168	0.00167	0.00166	0.00165
38	0.00194	0.00193	0.00192	0.00190	0.00189	0.00188	0.00187	0.00186	0.00185	0.00184	0.00183	0.00181	0.00180
39	0.00208	0.00207	0.00206	0.00205	0.00203	0.00202	0.00201	0.00200	0.00198	0.00197	0.00196	0.00195	0.00194
40	0.00224	0.00223	0.00221	0.00220	0.00218	0.00217	0.00215	0.00214	0.00212	0.00211	0.00209	0.00208	0.00206
41	0.00240	0.00238	0.00236	0.00235	0.00233	0.00231	0.00229	0.00228	0.00226	0.00224	0.00223	0.00221	0.00219
42	0.00256	0.00254	0.00252	0.00250	0.00248	0.00246	0.00244	0.00242	0.00240	0.00238	0.00236	0.00234	0.00232
43	0.00272	0.00270	0.00267	0.00265	0.00263	0.00260	0.00258	0.00256	0.00254	0.00252	0.00249	0.00247	0.00245
44	0.00288	0.00285	0.00283	0.00280	0.00278	0.00275	0.00273	0.00270	0.00268	0.00265	0.00263	0.00261	0.00258

45	0.00314	0.00311	0.00308	0.00305	0.00302	0.00299	0.00297	0.00294	0.00291	0.00288	0.00286	0.00283	0.00280
46	0.00340	0.00337	0.00333	0.00330	0.00327	0.00324	0.00321	0.00317	0.00314	0.00311	0.00308	0.00305	0.00302
47	0.00366	0.00362	0.00359	0.00355	0.00351	0.00348	0.00344	0.00341	0.00338	0.00334	0.00331	0.00328	0.00324
48	0.00392	0.00388	0.00384	0.00380	0.00376	0.00372	0.00368	0.00365	0.00361	0.00357	0.00353	0.00350	0.00346
49	0.00418	0.00413	0.00409	0.00405	0.00400	0.00396	0.00392	0.00388	0.00384	0.00380	0.00376	0.00372	0.00368
50	0.00454	0.00449	0.00444	0.00440	0.00435	0.00430	0.00425	0.00421	0.00416	0.00411	0.00407	0.00402	0.00398
51	0.00491	0.00485	0.00480	0.00474	0.00469	0.00464	0.00458	0.00453	0.00448	0.00443	0.00438	0.00433	0.00428
52	0.00527	0.00521	0.00515	0.00509	0.00503	0.00497	0.00491	0.00485	0.00480	0.00474	0.00468	0.00463	0.00457
53	0.00564	0.00557	0.00550	0.00544	0.00537	0.00530	0.00524	0.00518	0.00511	0.00505	0.00499	0.00493	0.00487
54	0.00600	0.00593	0.00585	0.00578	0.00571	0.00564	0.00557	0.00550	0.00543	0.00536	0.00530	0.00523	0.00517
55	0.00647	0.00639	0.00630	0.00622	0.00614	0.00606	0.00598	0.00590	0.00583	0.00575	0.00568	0.00560	0.00553
56	0.00694	0.00684	0.00675	0.00666	0.00657	0.00648	0.00639	0.00631	0.00622	0.00614	0.00605	0.00597	0.00589
57	0.00741	0.00730	0.00720	0.00710	0.00700	0.00690	0.00680	0.00671	0.00662	0.00652	0.00643	0.00634	0.00625
58	0.00787	0.00776	0.00765	0.00754	0.00743	0.00732	0.00722	0.00711	0.00701	0.00691	0.00681	0.00671	0.00662
59	0.00834	0.00822	0.00810	0.00798	0.00786	0.00774	0.00763	0.00752	0.00741	0.00730	0.00719	0.00708	0.00698
60	0.00916	0.00902	0.00888	0.00875	0.00861	0.00848	0.00835	0.00823	0.00810	0.00798	0.00786	0.00774	0.00762
61	0.00997	0.00982	0.00967	0.00952	0.00937	0.00922	0.00908	0.00894	0.00880	0.00866	0.00852	0.00839	0.00826
62	0.01079	0.01062	0.01045	0.01028	0.01012	0.00996	0.00980	0.00965	0.00949	0.00934	0.00919	0.00905	0.00890
63	0.01161	0.01142	0.01124	0.01105	0.01087	0.01070	0.01053	0.01036	0.01019	0.01002	0.00986	0.00970	0.00954
64	0.01242	0.01222	0.01202	0.01182	0.01163	0.01144	0.01125	0.01107	0.01088	0.01071	0.01053	0.01036	0.01019
65	0.01370	0.01348	0.01326	0.01305	0.01284	0.01263	0.01242	0.01222	0.01203	0.01183	0.01164	0.01145	0.01127
66	0.01497	0.01473	0.01450	0.01427	0.01404	0.01382	0.01360	0.01338	0.01317	0.01296	0.01275	0.01255	0.01235
67	0.01624	0.01599	0.01574	0.01549	0.01524	0.01500	0.01477	0.01454	0.01431	0.01408	0.01386	0.01364	0.01343
68	0.01751	0.01724	0.01697	0.01671	0.01645	0.01619	0.01594	0.01569	0.01545	0.01521	0.01497	0.01474	0.01451
69	0.01878	0.01849	0.01821	0.01793	0.01765	0.01738	0.01711	0.01685	0.01659	0.01633	0.01608	0.01583	0.01559
70	0.02055	0.02023	0.01991	0.01960	0.01929	0.01899	0.01870	0.01841	0.01812	0.01784	0.01756	0.01729	0.01702
71	0.02231	0.02196	0.02161	0.02127	0.02094	0.02061	0.02029	0.01997	0.01965	0.01934	0.01904	0.01874	0.01844
72	0.02407	0.02369	0.02332	0.02295	0.02258	0.02223	0.02187	0.02153	0.02119	0.02085	0.02052	0.02019	0.01987
73	0.02584	0.02543	0.02502	0.02462	0.02423	0.02384	0.02346	0.02309	0.02272	0.02236	0.02200	0.02165	0.02130
74	0.02760	0.02716	0.02672	0.02629	0.02587	0.02546	0.02505	0.02465	0.02425	0.02386	0.02348	0.02310	0.02273

75	0.03038	0.02990	0.02942	0.02895	0.02848	0.02803	0.02758	0.02713	0.02670	0.02627	0.02585	0.02544	0.02503
76	0.03316	0.03263	0.03211	0.03160	0.03109	0.03059	0.03011	0.02962	0.02915	0.02868	0.02822	0.02777	0.02733
77	0.03595	0.03537	0.03481	0.03425	0.03370	0.03317	0.03264	0.03211	0.03160	0.03110	0.03060	0.03011	0.02963
78	0.03873	0.03811	0.03750	0.03691	0.03632	0.03574	0.03517	0.03461	0.03405	0.03351	0.03297	0.03245	0.03193
79	0.04151	0.04085	0.04020	0.03956	0.03893	0.03831	0.03770	0.03710	0.03651	0.03592	0.03535	0.03479	0.03423
80	0.04691	0.04621	0.04551	0.04482	0.04415	0.04348	0.04283	0.04218	0.04155	0.04092	0.04030	0.03969	0.03910
81	0.05239	0.05164	0.05090	0.05017	0.04945	0.04874	0.04804	0.04735	0.04668	0.04601	0.04535	0.04470	0.04406
82	0.05792	0.05713	0.05635	0.05558	0.05482	0.05407	0.05333	0.05260	0.05189	0.05118	0.05048	0.04979	0.04911
83	0.06350	0.06267	0.06185	0.06105	0.06025	0.05946	0.05869	0.05792	0.05717	0.05642	0.05568	0.05496	0.05424
84	0.06913	0.06827	0.06741	0.06657	0.06574	0.06491	0.06410	0.06330	0.06251	0.06172	0.06095	0.06019	0.05944

Age	2045	2046	2047	2048
0	0.00655	0.00641	0.00628	0.00615
1	0.00514	0.00503	0.00493	0.00483
2	0.00377	0.00370	0.00363	0.00356
3	0.00223	0.00219	0.00215	0.00211
4	0.00023	0.00022	0.00022	0.00021
5	0.00020	0.00019	0.00019	0.00018
6	0.00017	0.00016	0.00016	0.00016
7	0.00014	0.00013	0.00013	0.00013
8	0.00010	0.00010	0.00010	0.00010
9	0.00007	0.00007	0.00007	0.00007
10	0.00008	0.00007	0.00007	0.00007
11	0.00008	0.00008	0.00008	0.00007
12	0.00009	0.00008	0.00008	0.00008
13	0.00009	0.00009	0.00008	0.00008
14	0.00009	0.00009	0.00009	0.00009
15	0.00014	0.00014	0.00013	0.00013
16	0.00019	0.00019	0.00018	0.00018

17	0.00024	0.00023	0.00023	0.00023
18	0.00029	0.00028	0.00028	0.00027
19	0.00034	0.00033	0.00033	0.00032
20	0.00040	0.00040	0.00039	0.00039
21	0.00047	0.00046	0.00046	0.00045
22	0.00054	0.00053	0.00053	0.00052
23	0.00061	0.00061	0.00060	0.00060
24	0.00068	0.00068	0.00068	0.00067
25	0.00075	0.00075	0.00074	0.00074
26	0.00082	0.00082	0.00081	0.00081
27	0.00089	0.00089	0.00088	0.00088
28	0.00096	0.00096	0.00095	0.00095
29	0.00103	0.00103	0.00102	0.00102
30	0.00106	0.00105	0.00105	0.00105
31	0.00109	0.00108	0.00108	0.00107
32	0.00112	0.00111	0.00110	0.00110
33	0.00115	0.00114	0.00113	0.00113
34	0.00118	0.00117	0.00116	0.00115
35	0.00133	0.00133	0.00132	0.00131
36	0.00149	0.00148	0.00147	0.00146
37	0.00164	0.00163	0.00162	0.00161
38	0.00179	0.00178	0.00177	0.00176
39	0.00193	0.00191	0.00190	0.00189
40	0.00205	0.00204	0.00202	0.00201
41	0.00218	0.00216	0.00214	0.00213
42	0.00230	0.00228	0.00227	0.00225
43	0.00243	0.00241	0.00239	0.00237
44	0.00256	0.00254	0.00251	0.00249
45	0.00278	0.00275	0.00272	0.00270
46	0.00299	0.00297	0.00294	0.00291

47	0.00321	0.00318	0.00315	0.00312
48	0.00343	0.00339	0.00336	0.00332
49	0.00364	0.00361	0.00357	0.00353
50	0.00394	0.00389	0.00385	0.00381
51	0.00423	0.00418	0.00413	0.00409
52	0.00452	0.00447	0.00441	0.00436
53	0.00481	0.00475	0.00470	0.00464
54	0.00510	0.00504	0.00498	0.00492
55	0.00546	0.00539	0.00532	0.00525
56	0.00581	0.00573	0.00565	0.00558
57	0.00617	0.00608	0.00599	0.00591
58	0.00652	0.00643	0.00633	0.00624
59	0.00688	0.00678	0.00668	0.00658
60	0.00750	0.00739	0.00728	0.00717
61	0.00813	0.00800	0.00788	0.00776
62	0.00876	0.00862	0.00848	0.00835
63	0.00939	0.00924	0.00909	0.00894
64	0.01002	0.00986	0.00969	0.00953
65	0.01109	0.01091	0.01073	0.01056
66	0.01215	0.01196	0.01177	0.01158
67	0.01322	0.01301	0.01281	0.01260
68	0.01428	0.01406	0.01384	0.01363
69	0.01535	0.01511	0.01488	0.01465
70	0.01675	0.01649	0.01623	0.01598
71	0.01815	0.01787	0.01759	0.01731
72	0.01956	0.01925	0.01894	0.01864
73	0.02096	0.02063	0.02030	0.01998
74	0.02237	0.02201	0.02166	0.02131
75	0.02463	0.02423	0.02384	0.02346
76	0.02689	0.02646	0.02604	0.02562

77	0.02916	0.02869	0.02823	0.02778
78	0.03142	0.03092	0.03043	0.02994
79	0.03369	0.03315	0.03262	0.03210
80	0.03851	0.03793	0.03735	0.03679
81	0.04343	0.04281	0.04219	0.04159
82	0.04844	0.04778	0.04713	0.04648
83	0.05353	0.05283	0.05214	0.05146
84	0.05869	0.05796	0.05723	0.05652

Values of $\widehat{a_x}$ and $\widehat{b_x}$

```
> mydataLcaF$ax
        0         1         2         3         4         5         6         7         8 
-4.250205 -4.523789 -4.901688 -5.516631 -7.440128 -7.573350 -7.728221 -7.913638 -8.146609 
        9        10        11        12        13        14        15        16        17 
-8.467130 -8.400093 -8.344678 -8.297946 -8.258692 -8.226643 -8.070105 -7.940948 -7.830134 
       18        19        20        21        22        23        24        25        26 
-7.733007 -7.646412 -7.579655 -7.521158 -7.469063 -7.422371 -7.380313 -7.333881 -7.292973 
       27        28        29        30        31        32        33        34        35 
-7.256426 -7.223477 -7.193681 -7.130435 -7.073540 -7.021727 -6.974140 -6.930049 -6.827064 
       36        37        38        39        40        41        42        43        44 
-6.750633 -6.681350 -6.617870 -6.573640 -6.460215 -6.359535 -6.268868 -6.186348 -6.110538 
       45        46        47        48        49        50        51        52        53 
-5.999684 -5.900575 -5.810849 -5.728823 -5.653254 -5.546063 -5.450283 -5.363591 -5.284330 
       54        55        56        57        58        59        60        61        62 
-5.211271 -5.097170 -4.995438 -4.903561 -4.819735 -4.742622 -4.613262 -4.499462 -4.397748 
       63        64        65        66        67        68        69        70        71 
-4.305733 -4.221685 -4.122649 -4.033199 -3.951544 -3.876370 -3.806683 -3.696916 -3.598509 
       72        73        74        75        76        77        78        79        80 
-3.509269 -3.427585 -3.352254 -3.242072 -3.143162 -3.053385 -2.971165 -2.895308 -2.782085 
       81        82        83        84        85 
-2.680974 -2.589569 -2.506127 -2.429341 -1.806180 
```

```
> mydataLcaF$bx
          0           1           2           3           4           5           6           7 
0.012821338 0.012856844 0.012926144 0.013115134 0.015989916 0.016607795 0.017430599 0.018579337 
          8           9          10          11          12          13          14          15 
0.020352484 0.023616818 0.022938969 0.022569477 0.022411607 0.022437847 0.022665248 0.022032692 
         16          17          18          19          20          21          22          23 
0.021807268 0.021775563 0.021862772 0.022022512 0.020216878 0.018685907 0.017344155 0.016150526 
         24          25          26          27          28          29          30          31 
0.015077297 0.014213813 0.013454143 0.012773065 0.012151328 0.011583235 0.011111165 0.010706514 
         32          33          34          35          36          37          38          39 
0.010347841 0.010027144 0.009732815 0.007685902 0.006873099 0.006190297 0.005603937 0.005960902 
         40          41          42          43          44          45          46          47 
0.005597556 0.005288211 0.005019803 0.004786737 0.004579249 0.004737438 0.004861911 0.004961081 
         48          49          50          51          52          53          54          55 
0.005042873 0.005111551 0.005559541 0.005923733 0.006224841 0.006477779 0.006692142 0.006998522 
         56          57          58          59          60          61          62          63 
0.007238172 0.007429851 0.007586536 0.007716278 0.008628075 0.009331117 0.009890150 0.010346393 
         64          65          66          67          68          69          70          71 
0.010725946 0.010428641 0.010163764 0.009928227 0.009717951 0.009529470 0.010070370 0.010494404 
         72          73          74          75          76          77          78          79 
0.010835901 0.011116607 0.011351538 0.011311634 0.011274392 0.011240316 0.011209282 0.011181082 
         80          81          82          83          84          85 
0.010460101 0.009833406 0.009283946 0.008798348 0.008366010 0.003938800 
```

```
> mydataLcaM$ax
        0         1         2         3         4         5         6         7         8 
-3.999399 -4.274825 -4.656245 -5.280719 -7.329956 -7.449624 -7.586794 -7.747948 -7.944494 
        9        10        11        12        13        14        15        16        17 
-8.200417 -8.166812 -8.137524 -8.112286 -8.090098 -8.071428 -7.791490 -7.584783 -7.418212 
       18        19        20        21        22        23        24        25        26 
-7.277818 -7.156148 -7.051358 -6.958929 -6.876060 -6.800756 -6.731693 -6.660579 -6.595563 
       27        28        29        30        31        32        33        34        35 
-6.535545 -6.479697 -6.427426 -6.346993 -6.274253 -6.207684 -6.146218 -6.089055 -5.979429 
       36        37        38        39        40        41        42        43        44 
-5.881699 -5.793340 -5.712600 -5.638225 -5.542685 -5.456152 -5.376996 -5.304006 -5.236259 
       45        46        47        48        49        50        51        52        53 
-5.140937 -5.054165 -4.974495 -4.900843 -4.832347 -4.733371 -4.643764 -4.561830 -4.486331 
       54        55        56        57        58        59        60        61        62 
-4.416298 -4.327441 -4.246300 -4.171575 -4.102285 -4.037670 -3.937233 -3.846367 -3.763355 
       63        64        65        66        67        68        69        70        71 
-3.686912 -3.616054 -3.526669 -3.444911 -3.369547 -3.299626 -3.234393 -3.144103 -3.061473 
       72        73        74        75        76        77        78        79        80 
-2.985281 -2.914577 -2.848616 -2.756120 -2.671795 -2.594270 -2.522498 -2.455665 -2.367230 
       81        82        83        84        85 
-2.286474 -2.212117 -2.143183 -2.078910 -1.545040 
```

```
> mydataLcaM$bx
           0            1            2            3            4            5 
 0.0208940382  0.0208936993  0.0208931679  0.0208912610  0.0207800648  0.0210724261 
           6            7            8            9           10           11 
 0.0214368747  0.0219034574  0.0225077632  0.0232746693  0.0232766272  0.0232496770 
          12           13           14           15           16           17 
 0.0232355174  0.0231859236  0.0231726628  0.0154095033  0.0110837739  0.0082200491 
          18           19           20           21           22           23 
 0.0061495008  0.0045696752  0.0030542738  0.0017760632  0.0006789883 -0.0002795190 
          24           25           26           27           28           29 
-0.0011249975 -0.0013106659 -0.0015463038 -0.0018045747 -0.0020711469 -0.0023370092 
          30           31           32           33           34           35 
-0.0005350687  0.0009830929  0.0022858706  0.0034210273  0.0044219637  0.0044655507 
          36           37           38           39           40           41 
 0.0045118058  0.0045551724  0.0045927569  0.0046289293  0.0060202946  0.0071686720 
          42           43           44           45           46           47 
 0.0081354093  0.0089622877  0.0096796853  0.0100922563  0.0104427876  0.0107441488 
          48           49           50           51           52           53 
 0.0110072658  0.0112391012  0.0115372245  0.0117883404  0.0120013775  0.0121851553 
          54           55           56           57           58           59 
 0.0123453060  0.0129774663  0.0135101768  0.0139656047  0.0143597174  0.0147049238 
          60           61           62           63           64           65 
 0.0146537450  0.0146099788  0.0145718501  0.0145379243  0.0145076883  0.0146157379 
          66           67           68           69           70           71 
 0.0147053559  0.0147811159  0.0148460757  0.0149020949  0.0147036750  0.0145374091 
          72           73           74           75           76           77 
 0.0143961110  0.0142742106  0.0141682249  0.0144600480  0.0146865592  0.0148671951 
          78           79           80           81           82           83 
 0.0150141142  0.0151358112  0.0146166781  0.0141581117  0.0137501487  0.0133846393 
          84           85 
 0.0130552079  0.0107265472 
```

```
> mydataLcaT$ax
        0         1         2         3         4         5         6         7         8 
-4.112885 -4.383009 -4.748968 -5.351907 -7.378517 -7.505038 -7.650625 -7.822685 -8.033588 
        9        10        11        12        13        14        15        16        17 
-8.309601 -8.265639 -8.225992 -8.189808 -8.156446 -8.125854 -7.909351 -7.736158 -7.590537 
       18        19        20        21        22        23        24        25        26 
-7.464600 -7.353383 -7.266365 -7.187876 -7.116289 -7.050344 -6.989174 -6.929364 -6.873872 
       27        28        29        30        31        32        33        34        35 
-6.822002 -6.773284 -6.727334 -6.654251 -6.587332 -6.525447 -6.467857 -6.414009 -6.312767 
       36        37        38        39        40        41        42        43        44 
-6.221362 -6.137979 -6.061269 -5.992274 -5.893627 -5.804278 -5.722551 -5.647216 -5.577329 
       45        46        47        48        49        50        51        52        53 
-5.478708 -5.389140 -5.307069 -5.231325 -5.160987 -5.060636 -4.969803 -4.886796 -4.810341 
       54        55        56        57        58        59        60        61        62 
-4.739460 -4.644755 -4.558481 -4.479225 -4.405912 -4.337703 -4.229640 -4.132324 -4.043788 
       63        64        65        66        67        68        69        70        71 
-3.962559 -3.887505 -3.798093 -3.716200 -3.640640 -3.570489 -3.505010 -3.409737 -3.322905 
       72        73        74        75        76        77        78        79        80 
-3.243123 -3.169317 -3.100644 -3.005203 -2.918246 -2.838367 -2.764486 -2.695756 -2.599870 
       81        82        83        84        85 
-2.512775 -2.432955 -2.359258 -2.290794 -1.729812 
```

```
> mydataLcaT$bx
          0           1           2           3           4           5           6           7 
0.017563711 0.017042456 0.015954906 0.014476815 0.019163216 0.019566935 0.020086691 0.020811633 
          8           9          10          11          12          13          14          15 
0.021852990 0.023586464 0.023220147 0.022911472 0.022636109 0.022368448 0.022135905 0.018406887 
         16          17          18          19          20          21          22          23 
0.016096305 0.014484620 0.013291144 0.012368216 0.010634610 0.009150113 0.007862136 0.006724267 
         24          25          26          27          28          29          30          31 
0.005709748 0.005053766 0.004448299 0.003883159 0.003355450 0.002860471 0.003754405 0.004523310 
         32          33          34          35          36          37          38          39 
0.005186499 0.005769898 0.006291376 0.005876630 0.005545417 0.005273396 0.005046103 0.004986573 
         40          41          42          43          44          45          46          47 
0.005678787 0.006251409 0.006732227 0.007141821 0.007495706 0.007820930 0.008093728 0.008326042 
         48          49          50          51          52          53          54          55 
0.008526837 0.008701603 0.009145907 0.009514607 0.009825732 0.010092007 0.010322519 0.010858006 
         56          57          58          59          60          61          62          63 
0.011308038 0.011691496 0.012022476 0.012311470 0.012711462 0.013036293 0.013305965 0.013533470 
         64          65          66          67          68          69          70          71 
0.013727567 0.013502152 0.013311983 0.013149520 0.013009288 0.012886555 0.013035053 0.013156434 
         72          73          74          75          76          77          78          79 
0.013257679 0.013343315 0.013416709 0.013400781 0.013382831 0.013364063 0.013345364 0.013327153 
         80          81          82          83          84          85 
0.012602744 0.011965944 0.011401876 0.010898609 0.010446840 0.005658303 
```

APPENDIX C: TURNITIN REPORT

CP2

ORIGINALITY REPORT

9%	1%	2%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	7%
2	www.mdpi.com Internet Source	1%
3	Submitted to The University of Manchester Student Paper	1%
4	Susanna Levantesi, Andrea Nigri. "A random forest algorithm to improve the Lee–Carter mortality forecasting: impact on q-forward", Soft Computing, 2019 Publication	1%
5	Submitted to UCSI University Student Paper	1%

Exclude quotes	Off	Exclude matches	< 1%
Exclude bibliography	On		

APPENDIX D: ETHICS FORMS

APPENDIX E: LOG SHEETS