IMPLEMENTATION OF A HYBRID MODEL USING K-MEANS CLUSTERING AND ARTIFICIAL NEURAL NETWORKS FOR RISK PREDICTION IN LIFE INSURANCE

JEFF KIMANGA NTHENGE

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF MASTER OF SCIENCE IN INFORMATION TECHNOLOGY OF THE UNIVERSITY OF EMBU

JUNE 2024

DECLARATION

This thesis is my original work and has not been presented elsewhere for a degree or any other award.

Signature: …… Date: …….

Jeff Kimanga Nthenge

Department of Computing and Information Technology

B532/1173/2017

This thesis has been submitted for examination with our approval as the University Supervisors

Signature: …… Date: …….

Dr. Faith Mueni Musyoka

Department of Computing and Information Technology

University of Embu

Signature: …… Date: …….

Dr. David Muchangi Mugo

Department of Computing and Information Technology

University of Embu

DEDICATION

As I stand at the precipice of this academic milestone, my heart swells with gratitude for the unwavering support and fervent prayers that have propelled me forward. This research work is a testament to the profound impact of those who have walked beside me on this arduous yet rewarding journey.

Foremost, I dedicate this accomplishment to Wilberforce Murikah, my esteemed mentor, whose guidance and wisdom have been the beacon that illuminated my path. His invaluable counsel has shaped not only this work but also my growth as a scholar and an individual.

To my beloved late father, Joseph, whose memory continues to inspire me to reach for greatness, I owe a debt of gratitude that can never be repaid. His life's lessons and the values he instilled in me have been the foundation upon which I have built my aspirations.

My dear mother, Jane, deserves the utmost recognition for her unwavering love and encouragement. Her constant belief in my abilities has been the driving force that propelled me forward, even in the face of adversity. Her prayers have been a source of solace and strength throughout this endeavour.

And to my brother, Jerry, whose steadfast support and belief in me have never faltered, I extend my heartfelt gratitude. His unwavering presence has been a constant reminder that I am never alone in this pursuit.

As I reflect upon the culmination of this research work, I am humbled by the profound impact these individuals have had on my life and academic pursuits. May the Almighty God shower His blessings upon them all, for without their unwavering support, this achievement would not have been possible.

ACKNOWLEDGMENT

I would like to express my profound gratitude to God for providing divine grace and strength throughout my thesis journey. I am grateful for the financial support and opportunity provided by the University of Embu Management that allowed me to pursue my master's degree. I extend my appreciation to the Department of Computing and Information Technology at the University of Embu for their unwavering commitment and support during the thesis process. My special thanks go to my research supervisors, Dr. Faith Mueni and Dr. David Mugo, for their expert guidance, insightful direction, and belief in my abilities, which greatly enriched my research work and personal growth. I acknowledge the invaluable contributions of all these individuals and institutions, whose collective efforts and unwavering commitment paved the way for this academic achievement.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

ABBREVIATIONS AND ACRONYMS

AI	-	Artificial Intelligence
ANN	-	Artificial Neural Networks
ANOVA	-	Analysis of Variance
ARM	-	Association Rule Mining
AUC	-	Area Under the ROC Curve
BAGNBT	-	Bagging-based Naïve Bayes Trees
CFS	-	Correlation-Based Feature Selection
CNN	-	Convolutional Neural Networks
CSV	-	Comma Separated Values
DNN	-	Deep Neural Networks
EDA	-	Exploratory Data Analysis
HML	-	Hybrid Machine Learning
KDE	-	Kernel Density Estimation
k-NN	-	K-nearest Neighbours
LDA	-	Linear Discriminant Analysis
LR	-	Logistic Regression
MAE	-	Mean Absolute Error
MSE	-	Mean Squared Error
ML	-	Machine Learning
MLR	-	Multiple Linear Regression
MLP	-	Multilayer Perceptron
PCA	-	Principal Component Analysis
ReLU	-	Rectified Linear Unit
RF	-	Random Forest
RMSE	-	Root Mean Squared Error
RNN	-	Recurrent Neural Networks
ROC	-	Receiver Operating Characteristic
SGD	-	Stochastic Gradient Descent
SHAP	-	SHapley Additive exPlanations
SVM	-	Support Vector Machines
WEKA	-	Waikato Environment for Knowledge Analysis

DEFINITION OF TERMS

Artificial Neural Networks	A collection of algorithms in machine learning that attempt to uncover hidden relationships in data, mirroring the human brain.
Ensemble Learning	Supervised learning algorithms that use a series of machine learning classification trees instead of one to improve the accuracy of the model.
Hybrid Machine Learning	Also known as semi-supervised learning is a machine learning algorithm that combines supervised and unsupervised learning to improve the accuracy and performance of a model.
K-Means Clustering	also known as semi-supervised learning is a machine learning algorithm that combines supervised and unsupervised learning to improve the accuracy and performance of a model.
Labelled Data	A sample group of data that have been tagged with one or more labels that allow supervised learning to perform predictive analysis.
Logistic Regression	A statistical modelling technique categorised as a supervised learning algorithm. This method scrutinizes the relationship between a dependent variable and one or more independent variables, employing binary classification.
Machine Learning	A field of artificial intelligence that enables computers to learn and improve from experience without being

	explicitly programmed and works by finding patterns and making predictions from data based on multivariate statistics, data mining, pattern recognition, and advanced/predictive analytics.
Principal Component Analysis	A technique for reducing the number of dimensions in datasets that employs the conversion of a significant number of variables into a more compact set while still retaining most of the information from the original set.
Supervised Learning	A predictive machine learning algorithm that feeds data into a chosen algorithm with the desired outputs, which are called “labels.”
Underwriter	A professional who assesses/evaluates the risks involved when insuring individuals or assets and determines the pricing of a policy.
Unlabelled Data	Data that has not been tagged with labels identifying properties, characteristics, or classification.
Unsupervised Learning	A machine learning algorithm that uses pattern recognition to analyse and cluster unlabelled datasets.

ABSTRACT

Accurate assessment of the risk posed by prospective policyholders is crucial for life insurance companies to effectively price policies and manage long-term liabilities. However, the complexity of risk factors makes relying solely on traditional actuarial models insufficient, particularly with the abundance of big data and unstandardized data from various sources. This study explored the development and performance of a hybrid machine learning model that combines Artificial Neural Network and K-Means Clustering to improve risk prediction in life insurance underwriting. A quasi-experimental design was adopted to evaluate the efficacy of K-Means Clustering and ANN algorithms on benchmark datasets and develop a hybrid model for risk prediction. The proposed hybrid model utilized the strengths of Artificial Neural Networks in modelling nonlinear relationships and K-Means in pattern recognition to handle unstandardized data. Using anonymized life insurance application data from Kaggle, the ANN algorithm achieved an accuracy of 90% but showed limitations in handling nonlinear relationships. K-Means Clustering successfully identified distinct risk profiles among policyholders, revealing hidden patterns in the unlabelled data. The hybrid model, integrating K-Means Clustering and ANN with principal component analysis for feature selection and the Adam optimizer, resulted in higher model performance. Testing accuracy improved from 90% for the standalone ANN to 98% for the hybrid technique, with improvements in precision, recall, and Area Under the ROC Curve. The enhanced predictive capability highlighted the potential of the hybrid approach in modernizing underwriting practices and conducting a more sophisticated data-driven analytical evaluation of policyholder risk. However, there were limitations, such as the use of a single-sourced insurance dataset due to concerns about data privacy. Further research into integrating diverse algorithms and testing on larger real-world datasets can assist insurers in unlocking more value and gaining a competitive advantage through advanced analytical modelling.

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

1.1.1 The Growth of Big Data and its Challenges

The growth of big data is driven by digital device usage, internet, data storage advancements, analytics, organizational value, and the emergence machine learning technologies (D. Gupta & Rani, 2018). These factors have come together to create a combination of technological, economic, and organizational trends, reshaping living and working in the digital age (Arena & Pau, 2020). Big data refers to large and complex datasets that are difficult to process using traditional processing applications or data management tools (Parimala et al., 2017). These datasets are characterized by their large volumes, variety, and complexity, and are generated at a higher velocity than organizations can handle.

Big data is largely unstructured, heterogeneous, rapidly growing, variable in nature, and exists in various formats such as images, text, documents, and videos (Parimala et al., 2017). The increase in customer dependency on the internet has resulted in the emergence of big data. This data comes from various sources including social media, sensor networks, machine-to-machine communications, and the internet of things (Rawat & Samriya, 2021). While big data offers advantages to businesses, such as aiding data-driven decision making, it also presents challenges.

The main challenges can be categorized into five key areas: privacy and security, loading, synchronization, computational complexity, and data accessibility (Rawat & Samriya, 2021). The rapid growth of application systems and mobile devices has led to a significant increase in data volume, surpassing the capabilities of traditional systems to handle such large quantities (Zakharova, 2019). An article by Forbes Kulkarni (2019), states that 95% of businesses consider managing unstructured data to be challenging.

As society increasingly relies on digital channels for communication, unstructured data has become a major hurdle for many companies. Managing and analysing big data

from the vast volumes of unstructured data generated by customer interactions on the internet is a costly and complex task. To remain competitive in the market, businesses must effectively manage this data and utilize it for accurate predictions and adaptation to market trends (Petrov, 2023).

1.1.2 The Role of Machine Learning in Big Data Analysis

As a subset of artificial intelligence, machine learning (ML) uses algorithms to create analytical models capable of recognizing patterns and making decisions with minimal human intervention (Bertolini et al., 2021). The process of pattern recognition, data mining, and predictive analysis is employed to identify patterns and make predictions based on data (Gul et al., 2021). In this context, machine learning and big data are interdependent, with big data providing the dataset and machine learning techniques providing the methods and techniques for analysing the data (Arora, 2020). The study further highlights the importance of understanding big data terms and characteristics and provides a basic architectural framework that organizations can use to leverage big data.

Machine learning algorithms can be broadly classified into unsupervised and supervised learning algorithms. Supervised learning utilizes a set of predefined rules known as algorithms, which are trained on labelled data with predefined input features and output variables (Osisanwo et al., 2017). The algorithm learns from the labelled data to predict new, unseen data based on identified patterns during the training phase. Some common examples of supervised learning algorithms used in risk prediction include neural networks, random forests, logistic regression, support vector machines, and decision trees (Castañón, 2019).

On the other hand, unsupervised learning is a type of machine learning algorithm that handles unlabelled data and does not have a specific output variable to predict (Dike et al., 2019). Instead, unsupervised learning algorithms focus on discovering patterns or relationships within the data itself. The applications of unsupervised learning in risk prediction include outlier detection for fraud prevention and clustering analysis to identify subgroups within a population that are at higher risk (Dwivedi et al., 2020; Sharman et al., 2021).

With the availability of large datasets and advancements in ML algorithms, both supervised and unsupervised learning have become powerful tools for risk prediction across finance, healthcare, and insurance. ML help identify and predict potential risks associated with financial transactions, medical conditions, and insurance claims (Boodhun & Jayabalan, 2018; D. K. Gupta & Goyal, 2018; Pathak & Jha, 2021; Rusdah & Murfi, 2020; Sharman et al., 2021).

1.1.3 Machine Learning Applications in the Insurance Industry

By accurately predicting and managing risks, industries are better equipped to make decisions and take necessary actions to minimize potential losses. The use of supervised and unsupervised learning for risk prediction has revolutionized various industries, providing them with efficient tools to manage their risks. This approach is proving highly effective for insurance companies, enabling them to improve their decision-making capabilities, minimize risk, and ensure profitability in an increasingly data-driven business landscape. Machine learning has numerous potential applications in the insurance industry, including underwriting, claim payments, and fraud detection (Howe, 2020).

Insurance companies possess a wealth of information on individuals, insurance requirements, and claim factors, including damage details and supporting evidence. Life insurance underwriting involves evaluating numerous risk factors to determine premiums and make policy acceptance decisions (Dwivedi et al., 2020). Traditionally, underwriters used predetermined rules and actuarial models that struggled with nonlinear interactions, missing data, and complex risk variable relationships (Kwiecień et al., 2020). With the exponential growth of unstructured data and advancements in machine learning, predictive analytics has emerged to modernize risk modelling in insurance underwriting (A. Verma et al., 2017).

1.1.4 The Kenyan Life Insurance Landscape

Life insurance plays a crucial role in Kenya by providing financial security and savings options for individuals and families. The two main types of policies are term insurance (temporary coverage) and permanent insurance (coverage for life). These policies can be supplemented with additional benefits such as premium waivers or accidental death payouts. According to Laser Insurance Brokers (2022), more than 90% of policies

focus on savings and investments. When purchasing life insurance, it is important to consider tailored coverage amounts, personalized rate plans, and experienced agents. On average, monthly premiums amount to around KES. 2,500 (Amssurity, 2020).

Despite efforts to increase adoption Kenya's insurance industry faces challenges such as affordability, public awareness, and the need to start policies early to reduce costs. It is essential to understand different policy types, coverage needs, and influencing factors when evaluating life insurance options. According to Mutua et al. (2023) insurance risks negatively impact firms' financial performance, while reinsurance risks have an insignificant effect on risk prediction. Loss ratios and health insurance fraud undermine insurers' stability, but innovation, market focus, firm leverage and size can boost performance (Kareem et al., 2018).

Several studies have examined the relationship between risk management and financial performance among Kenyan insurance firms. Key findings indicate that underwriting risk significantly reduces insurers' financial performance, with firm size acting as a negative moderating factor (Kiptoo et al., 2021; Mutua et al., 2023). Over time, the efficiency of Kenya's life insurance sector has declined; however, factors such as insurer size and stock exchange listings can improve firms' technical efficiency (Kamau, 2023). Additionally, research suggests that financial performance is positively associated with firm size but negatively linked to age. Insurers with higher leverage also tend to achieve better performance (Morara & Sibindi, 2021). These findings offer valuable insights into the determinants of success in Kenya's life insurance industry, particularly in terms of risk prediction capabilities and profitability.

1.1.5 Challenges in Life Insurance Risk Assessment

The unstandardized nature of data from diverse sources, along with the variety of structured and unstructured formats, introduces new complexities. The analysis process tends to be lengthy and prone to human error. To ensure profitability, insurance companies must analyse a substantial amount of data to assess the risks associated with their business operations. In the past, life insurance applications were evaluated using detailed rules-based procedures that relied on formulas and thumb

rules based on actuarial tables. As a result, accurately predicting risk levels was often challenging and time-consuming (Saputri & Devianto, 2020).

To address these challenges, the insurance industry is leveraging digital transformation and machine learning to gain valuable insights into profitability, risk analysis, and fraud detection (Tardieu et al., 2020). By successfully integrating digital transformation and machine learning, the insurance industry can unlock significant value and position itself for future success (Hanafy & Ming, 2021). ML algorithms enable insurance companies to identify data patterns and make accurate predictions, to effectively manage risks, maintain profitability, and improve overall business operations (Jain et al., 2019; Shahid et al., 2019; Trivedi et al., 2021).

1.1.6 The Proposed Hybrid Machine Learning Approach

This research proposed a hybrid machine learning approach to enhance risk prediction to take advantage of these advancements. ANN was used as the primary supervised architecture due to their demonstrated effectiveness in classification and predictive analytics across industries. The main advantage of ANNs is their ability to automatically detect complex patterns between input variables and target outputs, such as mortality risk.

Meanwhile, unsupervised K-Means Clustering enables the analysis of unlabelled real-world data to uncover hidden insights through data segmentation. When combined in a hybrid implementation, ANN and K-Means have the potential to improve prediction accuracy within opaque insurance datasets. The goal is to provide insurers with a modernized solution for better understanding of risk exposures in this data-rich landscape, strengthening competitiveness and financial sustainability.

To better predict risk in insurance, large datasets generated by life insurance companies can be analysed using hybrid machine learning techniques that can improve risk assessment in life insurance. Although risks cannot be eliminated, they can be managed and reduced with these advanced solutions. Research has shown that handling unlabelled data is important for risk prediction using hybrid machine learning models, as it can lead to more accurate models and better outcomes (Ardabili et al., 2019; Jain et al., 2019; Malav et al., 2017).

1.2 Statement of the Problem

In recent years, the insurance industry has seen a growth in the usage of big data for decision-making. This is reflected in the 18.9% increase in gross premiums in Kenya in 2021 compared to 2020. The increase can be attributed, in part, to improved operating conditions after the easing of COVID-19 restrictions. Insurance claims have also risen by 22.5% to KES 71.8 Bn in 2021, up from KES 58.7 Bn in 2020 (Cytonn, 2021). However, insurance uptake in Kenya remains low, with a penetration rate of just 2.3% as of December 2020, which is unchanged from 2019 and well below the global average of 7.4% (Central Bank of Kenya, 2021). This low penetration is largely due to many Kenyans viewing insurance as a luxury rather than a necessity (Cytonn, 2022). With the increasing number of claims and data, machine learning becomes a valuable tool for creating personalized premiums based on individual behaviour (Venkatachalam, 2021).

Despite the integration of machine learning, the process of risk assessment has become more challenging due to the reliance on mostly supervised learning. While supervised learning is effective in predicting labelled data, the rise of big data has resulted in an increase in unlabelled data, making it a less optimal approach (Mahesh, 2018). On the other hand, unsupervised learning is better suited for analysing unlabelled data and identifying patterns but is less effective in making accurate predictions (Dike et al., 2019). The use of both supervised and unsupervised learning for risk prediction has not been extensively explored in the context of life insurance. This study introduces a hybrid approach that combines unsupervised K-Means with supervised Artificial Neural Network for risk prediction.

1.3 Research Objectives

1.3.1 General Objective

This study aimed to develop and implement a hybrid model using K-Means Clustering and Artificial Neural Networks for risk prediction in life insurance.

1.3.2 Specific Objectives

i. To assess the gaps in the K-Means Clustering and ANN learning algorithms for risk prediction.
ii. To develop a hybrid model using the K-Means Clustering and ANN for risk prediction in life insurance companies.
iii. To validate the performance of the proposed hybrid model for risk prediction in life insurance companies.

1.4 Research Questions

1. What are the gaps in the K-Means Clustering and ANN learning algorithms implemented for risk prediction?
2. How do you develop a hybrid model using the K-Means Clustering and ANN for risk prediction in life insurance companies?
3. What is the performance of the developed hybrid model for risk prediction in life insurance companies?

1.5 Justification of the Study

The rapid growth of Big Data has made it increasingly difficult to apply machine learning to prediction tasks. Key challenges include unstructured formats, multi-sources, streaming data, poor quality, high dimensionality, limited labelling, and data imbalance. Additionally, algorithm scalability is an ongoing issue that needs to be addressed (Naeem et al., 2022). Artificial Neural Networks have proven effective in modelling nonlinear patterns between applicant attributes such as age, lifestyle, medical history, and risk level outcomes (Samuel et al., 2017). However, ANNs have limitations such as overfitting, sensitivity to hyperparameters, and lack of transparency (Yaseen, 2023). On the other hand, unsupervised clustering techniques like K-Means can extract insights from unlabelled data, but optimizing clusters remains challenging (Uzila, 2022).

One significant advantage of combining supervised and unsupervised learning algorithms in a hybrid model is that the strengths of each algorithm complement the weaknesses of the other. In this case, ANN performs well with labelled data, but its performance declines as the amount of unlabelled data increases (Dike et al., 2019). In

contrast, K-Means excels with unlabelled data by identifying patterns but is poor at making predictions. By incorporating both algorithms into a hybrid model, it becomes possible to leverage their respective strengths to enhance the overall accuracy of risk prediction. Hybrid machine learning models that integrate supervised and unsupervised algorithms have demonstrated better performance compared to individual algorithm techniques (Ardabili et al., 2019; Pes, 2020). The approach enables comprehensive data analysis to identify promising features or characteristics for informing risk assessments in businesses. This motivated the development of a hybrid model to address the unique challenges posed by the complexity, variety, and performance requirements of life insurance data.

The hybrid approach utilizes ANN's nonlinear modelling capabilities and K-Means unsupervised feature learning and pattern recognition to achieve more accurate analytical policyholder risk evaluation. Thus, the proposed hybrid model offers an innovative solution to improve the accuracy and performance of machine learning algorithms for risk prediction in the life insurance sector.

The study developed and evaluated a hybrid ANN and K-Means model on a life insurance dataset based on existing models, assessing performance using metrics such as accuracy, precision, recall, and Area Under the ROC Curve (AUC). The industry can benefit from better integration that connects business-critical applications and data sources, allowing for greater flexibility and adaptability to evolving needs and challenges. However, applications tailored to life insurance risk prediction remain relatively unexplored.

1.6 Limitations of the Study

While this research provides valuable insights, it is important to acknowledge the study's limitations. The dataset used in the research includes policies from only one insurer. This limited sample allowed for an initial proof-of-concept to demonstrate the viability of the hybrid model, but it may restrict generalisation of the findings to the entire sector. Different companies have varying underwriting practices, risk criteria, and business models that could affect the models' performance in different ways.

Another limitation is the study's cross-sectional design, which does not track improvements in model performance over a longer period. Additionally, there may be underlying confounding variables that have not been fully accounted for. Despite these limitations, there are opportunities for improvement. Expanding the dataset to include a more diverse range of data and incorporating exogenous factors, such as social media rankings, driving offenses, and revenue returns, can refine the model and confirm its real-world effectiveness.

These variables and other factors could affect risk levels, and future studies should consider collecting additional data to potentially improve the accuracy of risk assessment. Ongoing performance benchmarking can also help strengthen the methodology and enhance the predictive validity of the model across different insurers. The study represents an important step towards modernizing legacy underwriting practices through data-driven analytics. The research sets the direction for industry-academia collaboration in developing robust hybrid machine learning models that meet regulatory rigor while delivering better analytical value.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter explores the status of machine learning in risk prediction. It discusses the challenges and research gaps found in the literature. It also emphasizes the importance of further research on hybrid machine learning approaches in the life insurance sector. Additionally, the chapter presents an overview of the different machine learning models used in the insurance industry, discussing their strengths and limitations. The chapter concludes by justifying the current study, outlining its potential contributions, and how to address research gaps and challenges.

2.2 Machine Learning Algorithms in Risk Prediction

The rapid development of big data has presented opportunities and challenges to various sectors, particularly in the field of risk prediction. The life insurance industry, which relies heavily on accurate risk assessment for pricing and profitability, has been significantly impacted by this data revolution. Traditional actuarial models have limitations in capturing the complex relationships within the expanding and unstructured big data (Blier-Wong et al., 2021). As a result, the industry has turned to advanced machine learning techniques to improve analytical risk prediction. However, both supervised and unsupervised learning algorithms face difficulties in handling the diversity and speed of data.

Supervised learning, such as Neural Networks, excels at modelling complex relationships between demographics, behaviours, environment, and predicted outcomes (Aziz, 2020). These algorithms learn from labelled data to predict new, unseen data based on identified patterns during the training phase (Osisanwo et al., 2017). ANN has demonstrated better performance compared to traditional statistical models in handling large, complex datasets, making them ideal for addressing complex and nonlinear problems (Grebovic et al., 2022). In artificial neural networks, each neuron performs a weighted sum of its inputs followed by the application of an activation function. This process is mathematically expressed in Eq. 1.

$$O_j = f\left(\sum_{i=1}^{m} w_{ij}\, x_i + \ b_j\right)$$

Eq. 1

The output of a neuron in an ANN is represented as shown in Eq. 1. Where O_j represents the output of neuron j, f is the activation function, w_{ij} is the weight connecting input neuron i to neuron j, x_i is the input, and b_j is the bias term (Varanasi & Tripathi, 2019). The interconnected neuron layers in ANN can detect complex relationships, making them well-suited for combining multi-dimensional risk factors from intricate insurance datasets. However, ANNs have limitations such as the interpretability, the need for labelled data for training, and dependency on the quality and representativeness of the training data (Aziz, 2020; Grebovic et al., 2022).

On the other hand, unsupervised learning K-Means, works with unlabelled data and aims to discover patterns or relationships within the data (Er Kara & Firat, 2018). As shown in Eq. 2, the K-Means algorithm divides data points into clusters by minimizing the total squared distances between each point and its closest cluster centroid (Fränti & Sieranoja, 2019). In the equation J represents the objective function, $x_i^{(j)}$ refers to the i^{th} case in cluster j, and c_j denotes the centroid for cluster j. While unsupervised techniques are successful across various domains, they have limitations like determining optimal cluster counts, sensitivity to initial conditions and distance metrics, and inability to directly predict outcomes (Er Kara & Firat, 2018; Fränti & Sieranoja, 2019).

$$J = \sum_{j=1}^{k}\sum_{i=1}^{n}\left\| x_i^{(j)} - \ c_j \right\|^2$$

Eq. 2

Despite these limitations, both supervised and unsupervised algorithms have shown promise in improving risk prediction accuracy and efficiency in various domains. The increasing availability of large datasets and advancements in machine learning techniques have further driven the adoption of these algorithms in the insurance industry.

2.3 Machine Learning Applications in the Insurance Industry

The insurance industry has been actively adopting ML techniques to improve its operations, including risk prediction, fraud detection, and customer segmentation. Several studies have investigated the application of machine learning algorithms in the insurance domain, highlighting their potential to enhance decision-making capabilities and profitability. Boodhun and Jayabalan (2018) conducted a study on risk prediction in the life insurance using supervised learning algorithms. The authors found that decision trees and neural networks were effective in accurately predicting risk but also had limitations such as overfitting and computational complexity.

Hanafy and Ming (2021) compared machine learning models for predicting insurance fraud. The study observed that ANN achieved the highest accuracy using a hybrid approach but noted challenges such as class imbalance. Gopi and Govindarajula (2019) investigated risk classification in life insurance using predictive analytics. The results showed that ANN had the highest performance but also highlighted limitations such as sensitivity to architecture and interpretability. Pandey et al. (2018) analysed health insurance fraud using data mining and predictive modelling techniques. The study found that neural networks had slightly better accuracy than decision trees. Also, noted challenges such as the dynamic nature of fraudulent behaviours and the need for domain expertise.

These studies provide evidence of the successful application of machine learning algorithms, particularly neural networks in the risk prediction. However, the studies also highlight the need for further research on hybrid models to handle the increasing volume and complexity of insurance data. Additionally, the studies point out limitations and challenges such as overfitting, class imbalance, interpretability, and the need for domain expertise.

2.4 Hybrid Machine Learning Models in Risk Prediction

Hybrid machine learning models which combine supervised and unsupervised techniques, provide an innovative solution to overcome the challenges associated with using these approaches independently. By utilizing the strengths of the individual algorithms, hybrid models have shown promising results in different applications, including risk prediction. Malav et al. (2017) proposed a hybrid approach that

combines K-Means and ANN to predict heart disease, achieving an accuracy of 97% in disease detection than earlier proposed method. However, also identified limitations such as determining the optimal number of clusters and the lack of comparison with other hybrid models.

Dwivedi et al. (2020) examined the impact of dimensionality reduction on risk assessment in life insurance and found that using backward elimination with supervised learning improved accuracy. The study acknowledged limitations of computational complexity and suboptimal feature subsets. Islam et al. (2021) introduced a novel approach called ARLAS for detecting adverse selection behaviour of policyholders in life insurance. The method outperformed existing unsupervised methods, but pointed out challenges such as interpretability and sensitivity to thresholds. Biswas and Islam (2021) developed a hybrid model using K-Means and ANN for brain tumor classification, achieving high specificity, sensitivity, and accuracy. Some limitations such as data availability and computational complexity were also noted.

These studies provide evidence of the effectiveness of hybrid machine learning models in enhancing risk prediction accuracy and performance in various domains, including life insurance. However, the research also highlights limitations and challenges such as determining optimal hyperparameters, interpretability, computational complexity, and data quality. This underscores the need for further research and development of hybrid models that can effectively address these issues.

2.5 Other Machine Learning Models in the Insurance Industry

In addition to Artificial Neural Network s (ANNs) and hybrid models, various other machine learning algorithms have been applied in the insurance industry for risk prediction and other tasks. These algorithms show promise in improving the accuracy and efficiency of insurance operations, but also have their own limitations and gaps that need to be addressed.

Rustam and Yaurita (2018) used support vector machines (SVM) to predict insolvency in insurance companies. The study obtained the highest average accuracy of 84.08% using feature selection for SVM with discrete input data types. However, the authors also highlighted the need for more research on feature selection techniques to improve

model performance. Also noted the limitations of SVM, such as sensitivity to kernel function and hyperparameters, as well as computational complexity for large-scale datasets. Roy and George (2017) on the other hand compared different classifiers for detecting potential losses in insurance. The results showed that decision trees and random forest algorithms provided better performance than Naïve Bayes. However, also noted that these algorithms can suffer from overfitting if not properly regularized.

Boodhun and Jayabalan (2018) investigated the performance of various algorithms for predicting life insurance applicant risk. The researchers found that the REPTree algorithm had the best performance with the Correlation-based Feature Selection (CFS) method. However, the study also identified limitations such as overfitting, sensitivity to noisy or irrelevant features, and decreased interpretability as tree depth increases. For principal component analysis, multiple linear regression showed the best performance. However, it also had limitations such as assuming a linear relationship between input features and the target variable, as well as sensitivity to multicollinearity and outliers.

While these studies demonstrate the application of various machine learning models in the insurance industry, they also reveal gaps and limitations. For example, the performance of these models can vary depending on the dataset and task, emphasizing the need for further research on model selection and optimization techniques. Most studies focus on supervised algorithms, underlining the growing need for unsupervised techniques to identify hidden patterns and insights from the rising volume of unstructured, unlabelled insurance data.

2.6 A Hybrid Approach Combining K-Means Clustering and ANN

The literature review has revealed numerous gaps and challenges in the application of machine learning for risk prediction in the life insurance industry. While models show promise for improving insurance operations accuracy and efficiency, they have limitations and gaps that need to be addressed. A key challenge is the growing volume and complexity of insurance data, most of which is unstructured and unlabelled. Supervised learning algorithms rely on labelled data for training, are not sufficient to fully exploit the insights concealed within this data. This emphasizes the need for

hybrid models that combine supervised and unsupervised learning to better cope with the challenges posed by big data in the insurance sector.

Additionally, the literature review has shown that limited research has been conducted on employing hybrid machine learning approaches in the life insurance sector. Most of the previous works focusing on ensemble methods that combine multiple supervised learning algorithms. The highest performance achieved by existing models in this domain has been reported as 67%, indicating considerable room for improvement (Weichen, 2018). The objective of this study was to develop a hybrid model that incorporates ANN and K-Means for risk prediction in the life insurance sector.

This hybrid approach capitalized ANN's ability to model nonlinear relationships and K-Means Clustering's capacity to identify hidden patterns in unlabelled data. The model addressed the limitations of existing models by improving the generalizability of hybrid models and handling the challenges posed by the increasing volume and complexity of insurance data. The study focused on practical challenges and limitations, such as effective clustering, determining the optimal clusters, and improving the robustness of the hybrid model. By addressing these aspects, this study presented a more applicable hybrid model for risk prediction in life insurance.

The research also focused on the interpretability of the proposed hybrid model, as the is a common challenge in ML models when understanding the factors contributing to risk predictions is crucial for decision-making and regulatory compliance (Aziz, 2020; Gopi & Govindarajula, 2019). By incorporating techniques such as feature selection the study aimed to provide insights into the decision-making process of the hybrid model, making it more transparent and trustworthy for stakeholders. The study has contributed to the growing body of research on the application of hybrid machine learning in the insurance industry, providing valuable insights and recommendations for future research and development.

2.7 Gaps in K-Means Clustering and Artificial Neural Networks

A systematic literature review was conducted to identify limitations and gaps in using ANN and K-Means Clustering for insurance risk prediction based on previous studies. In a study conducted by Orong et al. (2019) on a hybrid prediction model integrating a modified genetic algorithm to k-means segmentation and C4.5, gaps were identified

in the k-means clustering technique for risk prediction. These gaps included difficulty in determining the optimal clusters, limited ability to handle outliers and noisy data, difficulty in handling non-numeric data, and limited ability to handle high-dimensional data.

Similarly, Malav et al. (2017) noted a gap in research using hybrid models in combination with other techniques, such as feature selection and dimensionality reduction for heart disease prediction using ANN and k-means. The study also highlighted the need for more testing of the hybrid approach on different datasets to assess its performance. Verma et al. (2016) identified gaps in a hybrid k-means clustering algorithm for prediction analysis. The gaps included limited real-world scenario testing, use of other evaluation metrics beyond accuracy, and testing the algorithm's performance across different datasets.

Pal et al. (2020) on the other hand found gaps in the neural network-based country-wise risk prediction of COVID-19. These gaps included a lack of reliable data sources for testing and validation of the model, limited access to large-scale datasets for the study, and access to tuning/optimisers for better performance. In a study of enterprise financial risk level under digital transformation with ANN, Yang (2022) there was lack of attempt to improve the generalisability of the results by using a larger and more diverse sample of companies' datasets. There was also limited exploration of other evaluation metrics, and lack of a comprehensive analysis by comparing the performance of different algorithms.

D. K. Gupta and Goyal (2018) noted in his study on credit risk prediction using ANN that the study did not use more measurement metrics to assess the performance. The study stated that ANNs required training on a more dataset to predict the outcome of decision variables correctly. Finally, the study by Radosteva et al. (2018) on the use of neural network models in market risk management identified a gap in the lack of discussion on other methods for market risk assessment. Also, the failure to compare the performance of the proposed neural network model with other existing models.

These gaps underscore the need for further research and development of ML techniques for risk prediction to improve their accuracy and effectiveness in real-world applications. Table 1 gives the summaries of the gaps from each of the studies on risk

prediction using either K-Means Clustering or Artificial Neural Network with a detailed table also available in Appendix II.

Table 1: Summary of gaps in K-Means Clustering and Artificial Neural Networks

Study	Gap
D. K. Gupta and Goyal (2018)	Lack of comprehensive metrics to evaluate ANN performance
Malav et al. (2017)	Limited evaluation of hybrid models with other techniques like feature selection
Orong et al. (2019)	Difficulty in determining optimal clusters in K-Means
Pal et al. (2020)	Lack of large datasets and tuning options for the ANN model
Radosteva et al. (2018)	Lack of comparison of the ANN model to other risk models
V. Verma et al. (2016)	Limited testing of K-Means hybrid model in real scenarios
Yang, (2022)	Limited model evaluation metrics

In conclusion, these studies identified gaps related to the quality and diversity of datasets used for training and validation. Further, the limitations of the current techniques used, and a need for more robust evaluation metrics to accurately assess model performance. Additionally, the studies noted that there is a need for more advanced optimisation techniques to improve the performance of these models. A greater emphasis on the use of feature selection and dimensionality reduction techniques to decrease the complexity of the models is also suggested.

2.8 Challenges and Research Gaps

Despite the potential advantages of hybrid machine learning models, the literature has identified several challenges and research gaps. One key challenge is effectively clustering unstructured data with outliers and determining the optimal number of clusters (Orong et al., 2019). Highlighting the need for hybrid approaches that address the limitations of K-Means. The literature review also reveals limited research on using hybrid machine learning approaches in the life insurance sector, with most previous works focusing on ensemble methods (Boodhun & Jayabalan, 2018; Sheshasaayee & Thomas, 2018).

Given the rising volume of unlabelled data that can't be used with supervised algorithms, it is crucial to evaluate model performance with unsupervised learning approaches (Yao et al., 2019). Other research gaps include the need to improve the practicability and generalizability of hybrid models (Hou et al. 2019). Focusing on prediction ability in addition to precision, improving feature selection for predicting outliers, and researching additional algorithms for better performance. Furthermore, there has been limited research on implementing a hybrid model in risk prediction for life insurance, with the highest performance achieved by other models being reported as 67% (Weichen, 2018).

Despite ML's numerous advantages for risk prediction in insurance, unsupervised learning techniques are not being widely utilized for this purpose yet (Pitacco, 2020). Incorporating unsupervised learning algorithms could provide additional benefits for insurance companies, such as clustering new and previously unknown risk factors, improving data quality, and enabling more efficient and accurate analysis of large datasets. However, the adoption of machine learning techniques in the life insurance sector has been slower compared to other industries, partly due to regulatory constraints and the need for interpretability and transparency in decision-making processes (Pitacco, 2020).

The challenges and research gaps identified in the literature highlight the need for further research on HML models in the life insurance sector. Addressing these gaps can lead to the development of more accurate, efficient, and generalizable models for risk prediction, benefiting insurance companies and policyholders alike.

CHAPTER THREE

RESEARCH METHODOLOGY

3.1 Introduction

This chapter presents description of the quasi-experimental research design utilized to develop and validate a hybrid machine learning model for risk prediction in the life insurance. The goals of this study were to identify the limitations of the K-Means and ANN algorithms for risk prediction. Then, create a hybrid model that integrates these algorithms, and to evaluate the performance of the hybrid model. The chapter begins with an explanation of the research design and details on data acquisition and description.

The chapter then delves into data pre-processing techniques employed. The weaknesses of K-Means Clustering and Artificial Neural Networks are assessed using the WEKA tool. Feature selection methods and dimensionality reduction approaches are then explored. Subsequently, the development of a hybrid model is described, and the implementation tools and environment utilized are outlined. Finally, the model evaluation and validation procedures. This structured approach ensures a thorough understanding of the subject matter.

3.2 Research Design

This study adopted a quasi-experimental design to evaluate the efficacy of the K-Means Clustering and ANN on benchmark datasets and to develop a hybrid model for risk prediction in the life insurance industry. The selection of the quasi-experimental design was based on the need to manipulate the independent variables while simultaneously controlling potential confounding variables, such as data quality and feature selection (Miller et al., 2020). The research process commenced with data acquisition and description, followed by data pre-processing. Subsequently, the weaknesses of K-Means Clustering and ANN were assessed using WEKA. This was followed by feature selection, dimensionality reduction, hybrid model development, and model evaluation and validation, as depicted in Figure 1.

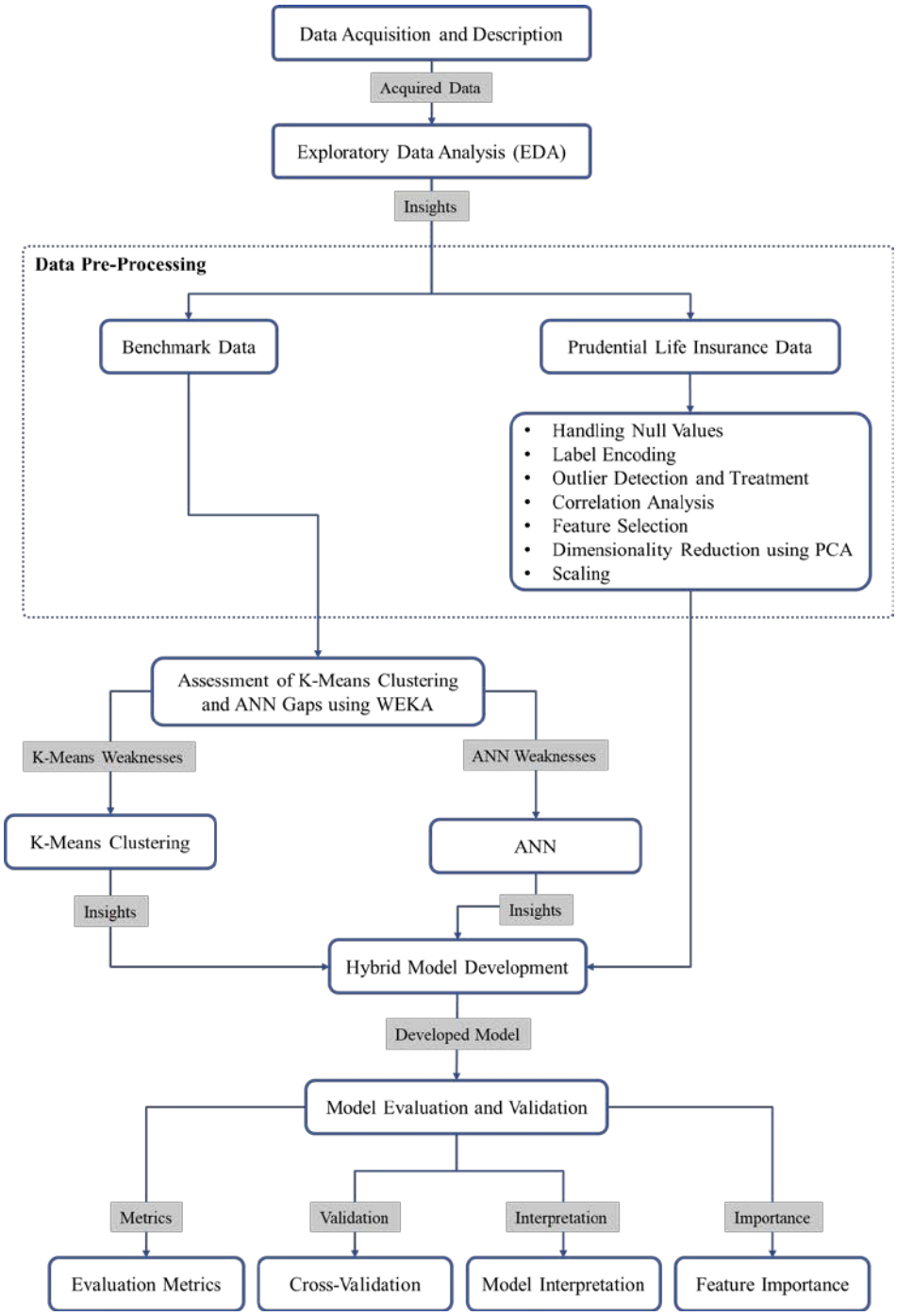

Figure 1: Overview of Research Design Steps

The iterative nature of the research design is a key attribute of the quasi-experimental approach. As the model was developed, it underwent a comprehensive evaluation process encompassing various metrics, validation techniques, and interpretation of results. Notably, the research design also prioritized the assessment of feature and variable relevance, further enhancing the model's robustness and interpretability.

The insights gained from this evaluation phase informed and shaped subsequent iterations of model development. This feedback loop, where empirical observations guided the refinement of modelling techniques, is a defining characteristic of the quasi-experimental approach adopted in this study. Through this iterative and data-driven process, researchers aimed to develop a robust and reliable predictive model that leverages the combination of K-Means and ANN techniques. The design enabled researchers to navigate the complexities of the data and adapt hybrid modelling strategies, enhancing the validity and applicability of their findings.

3.3 Data Acquisition and Description

The research study used secondary data sourced from the Prudential Life Insurance dataset for the year 2016. The dataset was obtained from Kaggle and contained information on life insurance applicants. The dataset was extensive, with 59,381 rows and 128 columns describing the attributes of the applicants. It consisted of two main files: "train.csv" and "test.csv". The "train.csv" file contained the historical data used for model training and included the target variable response, which measured risk on an ordinal scale with 8 levels. The "test.csv" file contained the data for which the response variable needed to be predicted.

Table 2 provides a summary of the dataset's various data fields, including demographic details, employment history, insurance history, family history, medical history, and medical keywords. The dataset's wide range of variables, covering different aspects of the applicants' information, provided a comprehensive set of features for risk prediction modelling. It was imperative to comprehend the reasons behind the absence of certain data points and implement suitable methodologies to uphold data integrity. The quality criteria for secondary data encompass completeness, consistency, reliability, validity, timeliness, and representativeness.

In the examination of life insurance, it was paramount to utilize data sources that encompass diverse facets of life insurance: medical information, family history, insurance history, personal information, and product information, all of which were encompassed in the dataset. Furthermore, the dataset was sourced from Kaggle, a dependable platform for secondary data (Hayashi et al., 2021).

Table 2: Characteristics of the Primary Life Insurance Dataset

Variable	Description
Id	A unique identifier associated with an application.
Product_Info_1-7	A set of normalized variables relating to the product applied for.
Ins_Age	Normalized age of applicant.
Ht	Normalized height of applicant.
Wt	Normalized weight of applicant.
BMI	Normalized BMI of applicant.
Employment_Info_1-6	A set of normalized variables relating to the employment history of the applicant.
InsuredInfo_1-6	A set of normalized variables providing information about the applicant.
Insurance_History_1-9	A set of normalized variables relating to the insurance history of the applicant.
Family_Hist_1-5	A set of normalized variables relating to the family history of the applicant.
Medical_History_1-41	A set of normalized variables relating to the medical history of the applicant.
Medical_Keyword_1-48	A set of dummy variables relating to the presence of/absence of a medical keyword being associated with the application.
Response	This is the target variable, an ordinal variable relating to the final decision associated with an application.

In addition to the secondary dataset, benchmark datasets were created to evaluate the limitations of K-Means and ANN algorithms. These datasets were designed to have well-defined clusters and clear classification of data points. The benchmark datasets consisted of 400 items, each with two elements generated from a Gaussian distribution with a standard deviation of 0.05. Table 3 summarizes the characteristics of the benchmark dataset.

The information was used to evaluate the performance of clustering algorithms on a synthetic dataset with known cluster structures and distributions. The provided characteristics allowed for a controlled environment to test and assess the gaps in

different clustering techniques. The use of benchmark datasets is a crucial practice as it allows for rigorous and objective evaluation, comparison, and advancement of algorithms, contributing to the development of better ML models.

Table 3: Characteristics of the Benchmark Datasets

Characteristic	Description
Number of items	400
Elements per item	2
Number of clusters	8
Cluster means	(0.20, 0.20), (0.30, 0.60), (0.20, 0.80), (0.50, 0.30), (0.60, 0.50), (0.60, 0.80), (0.80, 0.20), (0.80, 0.60)
Data point distribution	Gaussian (standard deviation: 0.05)
Total WCSS	1.415178
Individual cluster WCSS range	0.14050297 to 0.21048854

3.4 Exploratory Data Analysis

3.4.1 Prudential Life Insurance Dataset

An exploratory data analysis was conducted on the Prudential Life Insurance dataset to gain insights into the data structure, distributions, and relationships among variables. The EDA process involved reviewing the dataset documentation and data dictionary to understand the meaning and context of each variable, as summarized in Table 2. Univariate analysis was performed by analysing the distribution plots and visualizations of individual variables to identify patterns, outliers, and potential data quality issues. Kernel Density Estimate (KDE) plots were generated for each feature, with response set as the hue of each curve. This allowed for comparing the distributions across risk ratings and understanding any trends or correlations within the data. As shown in Figure 2, these KDE plots revealed distributions with varying modality, but consistently overlapped closely between each cohort of applicants based on their response.

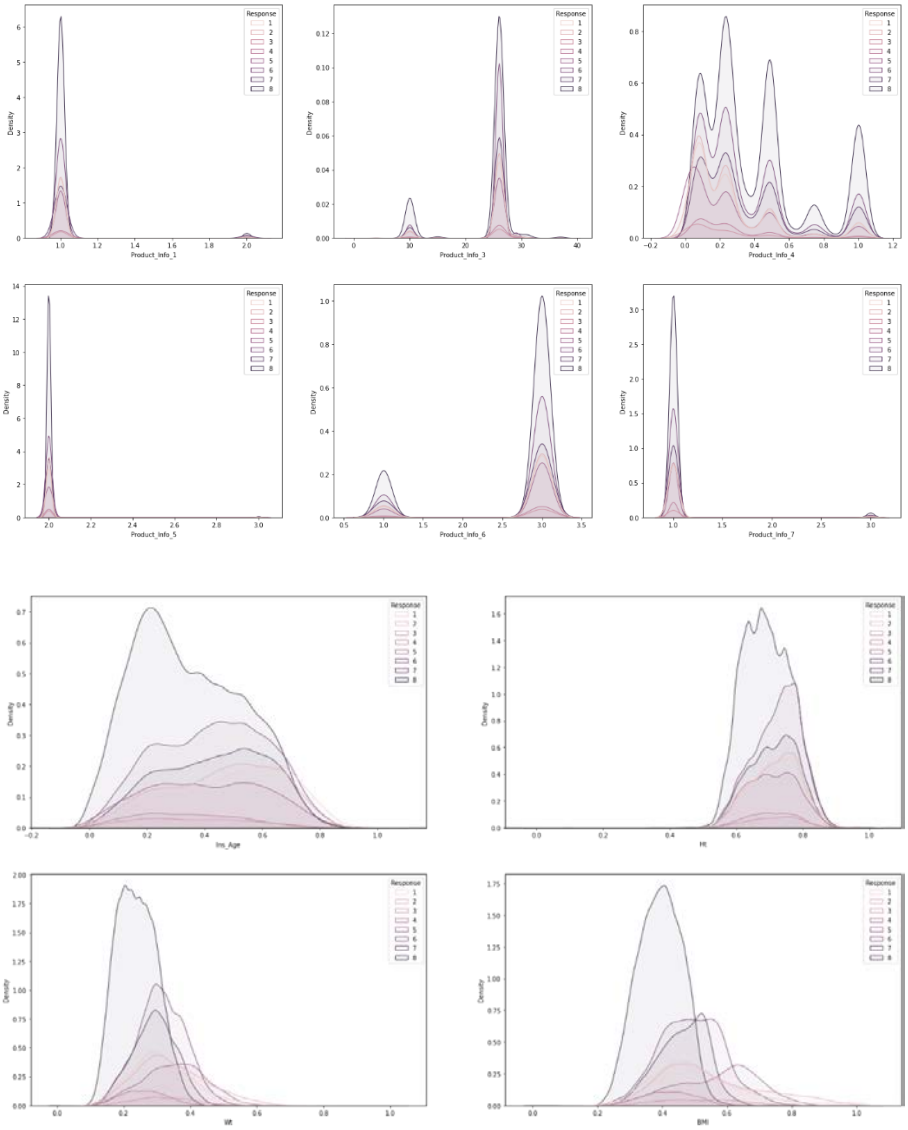

Figure 2: KDE Plots Showing the Number of Distributions

Most of The KDE plots showed overlapping distributions between applicant cohorts, suggesting limited predictive power for determining risk ratings. However, a few exceptions were noted. The medical_history_2/15/24 plots exhibited multimodal distributions with some predictive distinction in terms of variance and peak broadening. Although the small y-axis scales and underlying densities provided little help in distinguishing between response groups. In the medical_history_10 feature, low-risk applicants displayed bimodal distributions, while higher risk levels had single peaks.

However, the high proportion of missing values in this column limits its predictive value. For medical_history_23, as risk ratings increased, the peaks in the bimodal

distribution became sharper, and kurtosis became more positive. This suggested that values further from peak centres correlate with lower risk ratings, while overlapping values represent higher risk ratings. Missing value analysis revealed high percentages of missing values in several medical history columns. Target variable analysis was conducted to examine the distribution of response against insurance package applicants, to understand risk level nature and identify imbalances.

3.4.2 Benchmark Datasets

To gain insights into the characteristics and structure of the benchmark datasets, an exploratory data analysis was conducted using various statistical and visual techniques. Table 4 presents the summary statistics of the benchmark dataset, including the count, mean, standard deviation, minimum, maximum, and quartile values for each element.

Table 4: Summary Statistics of the Benchmark Dataset

Statistic	Element 1	Element 2
count	400	400
mean	0.4986	0.5006
std	0.2337	0.2333
min	0.0379	0.0963
25%	0.2572	0.2703
50%	0.5346	0.5351
75%	0.6965	0.7096
max	0.9287	0.9926

The EDA aimed to validate the dataset's suitability for evaluating the performance of clustering algorithms and ensure that it aligned with the specified characteristics. The summary statistics provides an overview of the value distribution for each element in the benchmark dataset. The mean values of both elements are around 0.5, indicating that the data points are centred within the range. The standard deviations of both elements are similar, suggesting a consistent spread of values across the dataset. To visualize the distribution of values for each element, histograms were plotted using the Seaborn library in Python. Figure 3 displays the histograms along with scatterplots

where each point represents a data point in the benchmark dataset, and the marginal distributions shown on the sides.

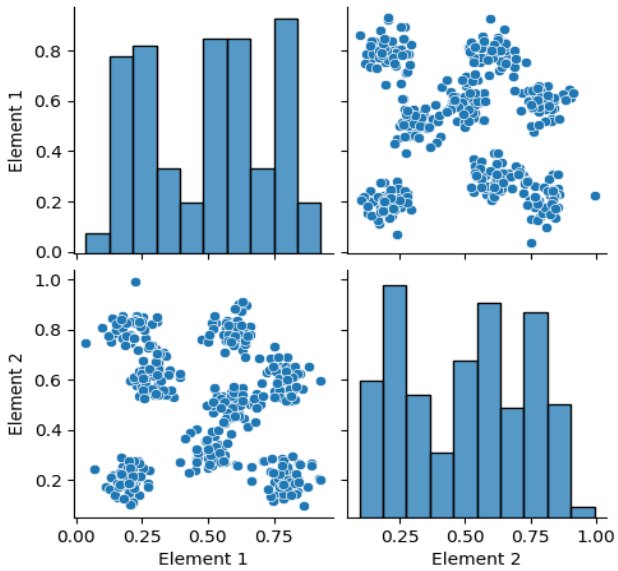

Figure 3: Pair plot Visualizing Relationships between Elements

The scatter plot confirmed the presence of eight distinct clusters in the benchmark dataset, as specified in Table 3. The clusters appeared well-separated, with data points tightly grouped around their respective centroids. The compact and spherical structure of the clusters aligns with the Gaussian distribution used to generate the data points.

3.5 Data Pre-Processing

3.5.1 Prudential Life Insurance Dataset

Based on the observations from the EDA, the Prudential Life Insurance dataset underwent extensive data pre-processing to ensure its quality and consistency. The steps included handling of null values, label encoding, outlier detection and treatment, correlation analysis, feature selection, dimensionality reduction and scaling.

3.5.1.1 Handling Null Values

After careful examination of the dataset, several columns were noted to contain a significant proportion of null values, rendering them ineffective for modelling purposes. Specifically, columns with over 75% null values were identified and subsequently removed from the dataset. The columns that met this criterion and

consequently eliminated included medical_history_10, medical_history_15, medical_history_24, and medical_history_32.

By removing these columns, the dataset was streamlined to focus on the most informative features. The remaining null values present in the dataset were filled with the mean value of their respective columns. This imputation strategy allowed for the preservation of the overall distribution and relationships within the data, while minimizing the impact of missing values on the subsequent modelling process.

3.5.1.2 Label Encoding

The variable 'product_info_2', which was originally of object type, went through a transformation process using label encoding. This process converted its categorical values into numerical representations that are suitable for machine learning algorithms. To make this conversion easier, a comprehensive dictionary was created. This dictionary maps each unique categorical value to a corresponding integer value. Table 5 shows this mapping, with the first two columns indicating the assignment of integer values to the original categorical labels, and the latter two columns displaying the reverse mapping. This reverse mapping allows for the retrieval of the original values from their encoded counterparts. Mathematically, the label encoding process can be summarized by the Eq. 3.

$$encoded\ value\ =\ f(x)$$

Eq. 3

The equation captures the essence of the encoding mechanism. In the equation, x represents a specific value from the categorical variable X, which is the target of the encoding process. The mapping function played a pivotal role by establishing a one-to-one correspondence between each distinct category and its assigned integer value as shown in Table 5. By assigning a unique integer to each category, the mapping function enables the seamless transformation of categorical data into a numerical format that can be readily processed by machine learning algorithms.

Table 5: Product_Info_2 Encoding

Categorical to Encoded Integer		Reverse Mapping	
Encoded Value	**Original Value**	**Encoded Value**	**Original Value**
0	D3	0	C2
1	A1	1	D3
2	E1	2	E1
3	D4	3	D4
4	D2	4	D2
5	A8	5	A8
6	A2	6	A2
7	D1	7	D1
8	A7	8	A7
9	A6	9	A6
10	A3	10	A3
11	A5	11	A5
12	C4	12	C4
13	C1	13	C1
14	B2	14	B2
15	C3	15	C3
16	C2	16	C2
17	A4	17	A4
18	B1	18	B1

The encoded values are used to replace the categorical labels in the test dataset, ensuring consistency with the encoded training data. This label encoding step is crucial for preparing the data for subsequent analysis and modelling tasks. It allows the machine learning algorithms to operate on numerical representations rather than categorical values.

3.5.1.3 Outlier Detection and Treatment

Scatter plots are a powerful tool for visually identifying outliers in life insurance data. These outliers, show points deviating significantly from the main trend that can distort the validity of subsequent analysis. By plotting relevant variables and inspecting the scatter plot, the study defined the criteria to objectively identify outliers and assess

their impact on the analysis. The selection of the most appropriate treatment strategy hinges on domain knowledge specific to the life insurance industry. By leveraging this knowledge, the study ensured the chosen approach minimized bias and accurately reflected the underlying relationships within the data. This approach ensures robust analysis and accurate insights, contributing to a more comprehensive understanding of life insurance data.

3.5.1.4 Correlation Analysis

The Pearson correlation coefficients between the independent variables and the target variable response were calculated to measure the strength of the relationships using Eq. 4. The absolute values of these coefficients were used to determine the degree of these relationships.

$$r = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum(X_i - \bar{X})^2 \sum(Y_i - \bar{Y})^2}}$$

Eq. 4

Where X_i and Y_i are the individual data points for variables X and Y, respectively. $\bar{X}$ and $\bar{Y}$ are the means of variables X and Y, respectively and $\sum$ denotes summation over all data points. As will be shown later in the results, the top 50 variables with the highest absolute correlation coefficients were selected for further analysis.

These variables were grouped into categories such as medical keywords, medical history, insured information, product information, employment information, and family history. Based on the analysis, it was determined that the selected features were most strongly associated with the target variable.

3.5.1.5 Feature Selection

Recursive feature selection techniques were employed to identify the most relevant features for risk prediction in the dataset. The study utilized a combination of correlation analysis and recursive feature elimination (RFE) to select the optimal subset of features. The dataset contained many outliers, and deleting rows with outliers was not viable as it would have significantly reduced the sample size. Instead, the

focus was on reducing the number of columns used to build the model through feature selection techniques.

Before performing feature selection, the target variable response was pre-processed to ensure compatibility with the machine learning algorithms. The original response variable contained class labels ranging from 1 to 8. To simplify the training process, a lambda function was applied to modify the class labels to a range of 0 to 7, as shown in Eq. 5.

$$f(R) = R - 1$$

Eq. 5

Where $f(R)$ represents the pre-processed response variable and R is the original response variable with class labels ranging from 1 to 8. The lambda function applied this transformation to each value in the response variable, effectively shifting the class labels by subtracting 1 from each label. Next, the independent variables (features) and the dependent variable (response) were separated into separate data frames. The data was then split into training and validation sets using the train_test_split function from the scikit-learn library, with 25% of the data reserved for validation and a random state of 1 for reproducibility. The output in Table 2Table 6 confirmed the shapes of the training and validation sets.

Table 6: Training and Evaluation Data Shape

Train Shape	**Evaluation Shape**
(44535, 123), (44535,)	(14846, 123), (14846,)

3.5.1.6 Dimensionality Reduction using Principal Component Analysis

Principal Component Analysis is a dimensionality reduction technique used to transform the original features into a lower-dimensional space while preserving the data's maximum variance. PCA identifies the principal components, which are linear combinations of the original features that capture the most significant patterns and variability in the dataset. PCA reduced dimensionality and identified key factors driving variability in the training dataset's feature space.

The first step in the PCA process used in this was to standardize the raw data. This standardization ensured that all features had zero mean and unit variance, allowing for a fair comparison and analysis. The standardized value of a particular feature for a specific data point was calculated using the formula presented in Eq. 6, where the original value is subtracted by the mean and divided by the standard deviation of that feature.

$$x_j^i = \frac{x_j^i - \bar{x}_j}{\sigma_j}$$

Eq. 6

Where x_j^i is the original value, $\bar{x}_j$ is the mean, and σ_j is the standard deviation of the j^{th} feature. Once the data was standardized, the covariance matrix was computed, which measures the pairwise covariances between the features. The covariance matrix was calculated using Eq. 7, where the product of the transposed standardized data points and the standardized data points is divided by the number of data points minus one. This matrix captures the relationships and dependencies among the features.

$$\Sigma = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^T$$

Eq. 7

Where n is the number of data points, x_i is the i^{th} standardized data point, and $\bar{x}$ is the mean of the standardized data. To identify the principal components, the eigenvalues and eigenvectors of the covariance matrix was calculated. Eq. 8 represents the eigenvalue problem, where the eigenvalues and corresponding eigenvectors satisfy the given equation. The eigenvalues indicate the amount of variance explained by each principal component, while the eigenvectors represent the direction of the principal components in the feature space.

$$\sum_j v_j = \lambda_j v_j$$

Eq. 8

The selection of the principal components to retain is based on the cumulative explained variance ratio. This ratio calculated using Eq. 9, measured the proportion of the total variance in the dataset that is accounted for by each principal component. A threshold, 95%, chosen to determine the number of principal components that captures most of the variability in the data. The principal components corresponding to the top eigenvalues are then selected for further analysis.

$$Explained\ Variance\ Ratio_j = \frac{\lambda_j}{\sum_{i=1}^{p} \lambda_i}$$

Eq. 9

Where p is the total number of features. A threshold (95%) was selected to capture most of the dataset's variability. The principal components corresponding to the top eigenvalues were selected. Finally, the original data is projected onto the selected principal components to obtain a lower-dimensional representation of the dataset. Eq. 10 describes the projection of a data point onto a specific principal component, where the standardized data point is multiplied by the corresponding eigenvector. This projection allows for the reduction of the dimensionality while retaining the most significant information and patterns present in the data.

$$z_{ij} = x_i^T v_j$$

Eq. 10

Where x_i is the i^{th} standardized data point and v_j is the j^{th} eigenvector. As shown in **Error! Reference source not found.** the first 40 PCs contained over 80% of the cumulative variance in the validation dataset. This meant that a significant majority of the dataset's cumulative variance could still be captured if a lower dimensionality feature-space were used instead, rather than simply using all features together.

3.5.1.7 Scaling

The dataset was already normalised, and additional scaling was not performed to preserve the relative proportions and relationships in the data. By not performing additional scaling, the original range and distribution of the features were maintained.

3.5.2 Pre-Processing of Benchmark Datasets

To evaluate the limitations and constraints of K-Means and ANN, benchmark datasets were created to complement the primary Prudential Life Insurance dataset. These benchmark datasets were carefully designed to showcase clear clusters and provide unambiguous classification of data points. The datasets were created using a Gaussian distribution with a standard deviation of 0.05 to generate synthetic data points. Each dataset consisted of 400 items, with each item having two elements. The datasets were constructed to exhibit eight distinct clusters, with the cluster means specified in Table 3.

This deliberate design ensured that the benchmark datasets had well-separated clusters, allowing for a comprehensive assessment of the algorithms' clustering capabilities. To ensure the integrity and suitability of the benchmark datasets for evaluation, a series of pre-processing steps were applied. The data generation process utilized the Gaussian distribution. This approach ensured that the data points within each cluster followed a consistent distribution, maintaining the desired cluster structure.

Since the benchmark datasets were synthetic and the elements had a standardized range, no additional scaling or normalization procedures were necessary. The elements were already in a suitable format for direct utilization in the evaluation process. Finally, each data point in the benchmark datasets was assigned to its respective cluster based on the predefined cluster means. This served as the ground truth to evaluate the clustering algorithms' accuracy in identifying and grouping data points correctly.

To evaluate the quality and characteristics of the benchmark datasets, researchers employed the Within-Cluster Sum of Squares (WCSS) metric and visual inspection. A scatter plot was generated to visualize the distribution of data points within these benchmark datasets. This plot allowed the researchers to identify the number of distinct clusters present and examine the nature of the groupings formed. If the visual representation aligned with low individual cluster WCSS values, it would confirm that the clusters were well-defined and compact, with data points tightly clustered around their respective centroids.

The benchmark datasets were generated using a Gaussian distribution, a common technique in data modelling and simulation. By applying WCSS analysis and visual

inspection, the researchers sought to validate the characteristics of these benchmark datasets. This would demonstrate their suitability for evaluating the performance of various clustering algorithms across different scenarios. Specifically, the datasets were designed to feature well-separated and unambiguous clusters, allowing for a rigorous assessment of how effectively the algorithms could identify and delineate such distinct groupings within the data.

3.6 Assessment of K-Means Clustering and ANN Gaps using WEKA

Before developing the hybrid model, the individual weaknesses of K-Means Clustering and ANN algorithms were assessed using the Waikato Environment for Knowledge Analysis framework. WEKA is a widely used open-source machine learning software that provides a collection of machine learning algorithms for data pre-processing, clustering, classification, regression, and visualization (Ratra et al., 2021).

3.6.1 K-Means Clustering Assessment Using WEKA

The K-Means Clustering algorithm's performance and limitations were assessed using the benchmark dataset described in Section 3.4.2. To begin the assessment, the benchmark dataset was converted to the Attribute-Relation File Format (ARFF), compatible with the WEKA framework. Since the benchmark dataset was already normalized and free from missing values or outliers, no additional pre-processing was necessary. The K-Means Clustering was implemented using the SimpleKMeans algorithm in WEKA.

The algorithm was configured with the following settings the number of clusters (k) was set to 8, aligning with the known number of clusters in the benchmark dataset. The Euclidean distance metric was selected to determine the proximity between data points and cluster centroids. The maximum number of iterations was set to 500 to ensure convergence of the algorithm and allow for sufficient refinement of the cluster assignments. The initialization method was also set to "Random" to randomly select the initial cluster centroids, as the true cluster centroids were not known in advance.

To assess the performance of the K-Means Clustering algorithm, two evaluation metrics were chosen: Within-Cluster Sum of Squares and Silhouette Coefficient.

WCSS measures the compactness of the clusters by calculating the sum of squared distances between each data point and its assigned cluster centroid. A lower WCSS indicates more compact and well-defined clusters. WCSS was calculated using the Eq. 11:

$$WCSS = \sum_{i=1}^{k} \sum_{x \epsilon C_i} (x - \mu_i)^2$$

Eq. 11

where k is the number of clusters, C_i is the set of data points assigned to cluster i, x is a data point, and μ_i is the centroid of cluster i. The Silhouette Coefficient measures the quality of the clustering by considering both the cohesion within clusters and the separation between clusters. It ranges from -1 to 1, where a higher value indicates better-defined clusters. The Silhouette Coefficient for a data point x was calculated using the Eq. 12:

$$s(x) = \frac{b(x) - a(x)}{\max\{a(x), b(x)\}}$$

Eq. 12

where $a(x)$ is the average distance between x and all other data points in the same cluster, and $b(x)$ is the minimum average distance between x and the data points in any other cluster. The K-Means algorithm was applied to the benchmark dataset using the specified parameters. The resulting clusters were evaluated using the WCSS and Silhouette Coefficient metrics. The obtained WCSS value for the benchmark dataset was compared to the total WCSS to assess cluster compactness. The Silhouette Coefficient was calculated for each data point and averaged to measure clustering quality overall. Scatter plots were generated to visualize the clustering results. These plots provided a qualitative assessment of cluster separation and structure.

3.6.2 ANN Assessment Using WEKA

The Artificial Neural Network algorithm was evaluated using the benchmark dataset described in Section 3.4.2 and the WEKA framework (Maseer et al., 2021). The benchmark dataset was converted to the Attribute-Relation File Format (ARFF) to

ensure compatibility with WEKA. Since the dataset was already normalized and had no missing values or outliers, no further pre-processing was required. The MultilayerPerceptron algorithm in WEKA was utilized to implement the ANN (Aumüller et al., 2018). The network architecture consisted of a single hidden layer with 10 nodes, and the sigmoid activation function Eq. 13 was applied to capture non-linear relationships.

$$f(x) = \frac{1}{1 + e^{-x}}$$

Eq. 13

where x is the input to the node. The output layer comprised 8 nodes, corresponding to the number of clusters in the benchmark dataset, and employed the softmax activation function Eq. 14 to obtain probability distributions over the cluster labels (Kouretas & Paliouras, 2019).

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Eq. 14

where z_i is the input to the i^{th} output node, and K is the total number of output nodes. The learning rate, momentum, and number of epochs were set to 0.3, 0.2, and 500, respectively, based on empirical experimentation and consideration of convergence speed and stability. To evaluate the performance of the ANN algorithm, a comprehensive set of metrics was employed. Accuracy was calculated to measure the overall correctness of the model. The confusion matrix broke down the model's predictions into true positives, true negatives, false positives, and false negatives for each class, enabling detailed analysis (Haghighi et al., 2018). Precision Eq. 15 and recall Eq. 16 were computed to assess the model's ability to correctly identify positive instances and its effectiveness in capturing all positive instances, respectively.

$$Precision = \frac{TP}{TP + FP}$$

Eq. 15

$$Recall = \frac{TP}{TP + FN}$$

Eq. 16

where TP is the number of true positive predictions, FP is the number of false positive predictions, and FN is the number of false negative predictions. The f1-score Eq. 17 the harmonic mean of precision and recall, was used to provide a balanced measure of the model's performance, particularly in scenarios with imbalanced class distributions:

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Eq. 17

Cross-validation Srinivasan et al. (2019) was employed to assess the model's performance and identify any signs of overfitting. The benchmark dataset was divided into k folds, and the model was trained and evaluated k times, each time using a different fold as the validation set. The evaluation metrics were averaged across the k folds to obtain a more robust estimate of the model's performance and generalisation ability.

3.7 Hybrid Model Development

The hybrid model integrating K-Means Clustering and Artificial Neural Networks, was developed using the selected features from the feature selection and dimensionality reduction phases. The development process was carried out using various implementation tools and environments, including Google Colab, Python, and libraries such as scikit-learn, TensorFlow, and Keras. These tools provided a powerful and flexible framework for data pre-processing, model development, and evaluation.

3.7.1 K-Means Clustering

The K-Means Clustering algorithm was employed to identify distinct groups within the Prudential Life Insurance dataset based on the selected features. To determine the optimal number of clusters (k), the elbow method was used. This method involved plotting the WCSS errors against different values of k and identifying the "elbow point", where the rate of decrease in WCSS significantly slowed down. The WCSS

was calculated as shown in Eq. 11. The elbow method analysis involved iteratively running the K-Means algorithm with increasing values of k and calculating the corresponding WCSS. The output was then visualized in a line plot, with x-axis represents the number of clusters (k), while the y-axis represents the WCSS values.

Based on the elbow method analysis, the optimum number of clusters was determined to be $k = 15$. This value offered a good trade-off between model complexity and performance, as additional clusters beyond this point did not significantly enhance the clustering results. The K-Means Clustering algorithm was then employed using the chosen value of $k = 15$ and the Euclidean distance metric. The algorithm aimed to minimize the WCSS by iteratively assigning data points to clusters based on their proximity to the cluster centroids and updating the centroids based on the mean of the assigned data points.

To evaluate the quality of the clustering results, we conducted silhouette analysis. The silhouette coefficient, which was calculated using Eq. 12, measured the separation and compactness of the clusters. Higher values indicated more well-defined and clearly separated clusters. The silhouette analysis confirmed that the chosen value of $k = 15$ resulted in clusters with good separation and compactness.

3.7.2 Artificial Neural Network

An ANN was developed to predict the risk level of life insurance applicants based on the clustered data. The ANN architecture included an input layer, one or more hidden layers, and an output layer. The number of neurons in the input layer represented the selected features. While the number of neurons in the output layer was determined by the ordinal nature of the target variable response (in this case, 8 output neurons for the 8 risk levels). Advanced techniques optimized the ANN's hyperparameters like hidden layers, neurons per layer, learning rate, and regularization parameters. Keras Tuner, a hyperparameter optimization framework, was used with RandomSearch to efficiently explore and identify the best combination of hyperparameters. The objective function was set to minimize the validation loss, and the search space was defined for each hyperparameter.

The backpropagation algorithm trained the ANN by adjusting weights and biases to minimize differences between predicted and actual risk levels. The optimization

process involved forward propagation, loss calculation, and backpropagation steps. The ANN was trained on 80% of the clustered data and validated on the remaining 20% to assess its generalisation performance and prevent overfitting. Regularization techniques, such as L1/L2 regularization, were applied to enhance the model's robustness and minimize overfitting.

3.7.3 Integration of K-Means Clustering and ANN

The hybrid model integrated K-Means with ANN to capture the intrinsic structure and the complex relationships between the input features and risk levels. The process began by performing K-Means Clustering on the training data, using the optimal number of clusters $k = 15$ determined through the elbow method. Each data point in the training set was assigned a cluster label based on its proximity to the cluster centroids. These cluster labels were then combined with the original input features, creating an augmented feature set that served as the input for the ANN.

To optimize the ANN's hyperparameters and improve its performance, several optimization algorithms were employed. These optimizers include SGD, RMSprop, AdaGrad, Adadelta, Adam, AdaMax and Nadam. Stochastic Gradient Descent is a widely used algorithm that updates the model's parameters based on the gradient of the loss function with respect to each parameter. The update rule for SGD is given by Eq. 18.

$$\theta_{t+1} = \theta_t - \eta . \nabla_\theta J(\theta_t)$$

Eq. 18

where θ represents the model's parameters, η is the learning rate, and $\nabla_\theta J(\theta_t)$ is the gradient of the loss function with respect to the parameters at iteration t. Root Mean Square Propagation (RMSprop) is an adaptive learning rate optimization algorithm that adjusts the learning rate for each parameter based on the magnitude of its recent gradients. This optimizer was also employed in the hybrid model, and it optimizes as shown in Eq. 19 and Eq. 20.

$$v_t = \beta v_{t-1} + (1 - \beta) . \nabla_\theta J(\theta_t))^2$$

Eq. 19

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{v_t + \epsilon}} . \nabla_\theta J(\theta_t)$$

Eq. 20

where v_t is the running average of the squared gradients, β is the forgetting factor, and ϵ is a small constant for numerical stability.

Adaptive Gradient (AdaGrad) adapts the learning rate for each parameter based on the historical gradients observed during training was also used in the hybrid model. This optimizer, given by Eq. 21 and Eq. 22.

$$G_t = G_{t-1} + (\nabla_\theta J(\theta_t))^2$$

Eq. 21

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_t + \epsilon}} . \nabla_\theta J(\theta_t)$$

Eq. 22

where G_t is the sum of the squared gradients up to iteration t.

Adaptive Delta (Adadelta) is an extension of AdaGrad that uses a running average of the gradients and updates to adapt the learning rate. The update rule on how Adadelta works is given by Eq. 23, Eq. 24, Eq. 25 and Eq. 26.

$$E[g^2]_t = \gamma E[g^2]_{t-1} + (1 - \gamma) . (\nabla_\theta J(\theta_t))^2$$

Eq. 23

$$\Delta\theta_t = -\frac{\sqrt{E[\Delta\theta^2]_{t-1} + \epsilon}}{\sqrt{E[g^2]_t + \epsilon}} . \nabla_\theta J(\theta_t)$$

Eq. 24

$$E[\Delta\theta^2]_t = \gamma E[\Delta\theta^2]_{t-1} + (1 - \gamma) . (\Delta\theta_t)^2$$

Eq. 25

$$\theta_{t+1} = \theta_t + \Delta\theta_t$$

Eq. 26

where $E[g^2]_t$ and $E[\Delta\theta^2]_t$ are the running averages of the squared gradients and updates, respectively, and γ is the decay rate.

Adaptive Moment Estimation (Adam) combines the advantages of AdaGrad and RMSprop by adapting the learning rate based on the first and second moments of the gradients. The update rule for Adam is given by Eq. 27, Eq. 28, Eq. 29, Eq. 30 and Eq. 31

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1).\nabla_\theta J(\theta_t)$$

Eq. 27

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2).(\nabla_\theta J(\theta_t))^2$$

Eq. 28

$$\widehat{m}_t = \frac{m_t}{1 - \beta_1^t}$$

Eq. 29

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t}$$

Eq. 30

$$m_t\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} .\widehat{m}_t$$

Eq. 31

where m_t and v_t are the first and second moment estimates, respectively, β_1 and β_2 are the forgetting factors, and $\widehat{m}_t$ and $\hat{v}_t$ are the bias-corrected moment estimates.

AdaMax is a variant of Adam that uses the infinity norm instead of the second moment to scale the learning rate. The update rule for AdaMax is given by Eq. 27, Eq. 32 and Eq. 33.

$$u_t = \max\left(\beta_2 . u_{t-1}, |\nabla_\theta J(\theta_t)|\right)$$

Eq. 32

$$\theta_{t+1} = \theta_t - \frac{\eta}{u_t} . m_t$$

Eq. 33

where u_t is the maximum of the absolute values of the gradients.

Nesterov-accelerated Adaptive Moment Estimation (Nadam) incorporates Nesterov momentum into the Adam update rule. The update rule for Nadam is given by Eq. 27, Eq. 28, Eq. 29, Eq. 30 and Eq. 34.

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} . (\beta_1 \hat{m}_t + \frac{(1-\beta_1).\nabla_\theta J(\theta_t)}{1-\beta_1^t})$$

Eq. 34

The optimization algorithms were used to adjust the hyperparameters of the ANN. The performance of these algorithms was assessed using cross-validation and selected evaluation metrics. The most successful optimizer was chosen based on its consistent performance and ability to find the optimal hyperparameter configurations. Once the best-performing optimizer and hyperparameter configuration was determined, the final hybrid model was trained using the optimized ANN architecture. It also included both the original features and the cluster labels from the augmented feature set. The performance and generalisation capability of the trained hybrid model was then evaluated on the test data.

3.8 Model Evaluation and Validation

The hybrid model's performance evaluation and validation involved a comprehensive assessment of its predictive accuracy, robustness, and generalisation ability. The process began with selecting appropriate evaluation metrics, including accuracy, precision, recall, f1-score, and AUC curve. These metrics provided a holistic view of the model's performance, considering overall correctness, positive instance identification, and discriminatory power. Once the evaluation metrics were

determined, the next step involved calculating these metrics using the predicted and actual risk levels obtained from the hybrid model. This step allowed for a quantitative assessment of the model's performance and enabled comparisons with other models or benchmarks.

To ensure the model's robustness and generalisation ability, K-fold cross-validation was employed. This technique involved dividing the dataset into K equally sized subsets, known as folds. The model was then trained and evaluated K times, with each iteration using a different fold as the validation set and the remaining K-1 folds as the training set. By averaging the performance metrics across all iterations, a more reliable estimate of the model's performance was obtained. This reduced the risk of overfitting, and provided insights into how well the model generalises to unseen data.

Evaluating the model also involved interpreting its decision-making process and understanding each input feature's contribution to risk prediction. To achieve this, techniques such as permutation importance and SHapley Additive exPlanations (SHAP) were applied. Permutation importance measured a feature's importance by the performance decrease when that feature was randomly permuted. SHAP values, on the other hand, provided a more granular understanding of each feature's contribution to the predicted risk level. By analysing the distribution of SHAP values across the dataset, the impact and directionality of each feature could be assessed.

3.8.1 Evaluation Metrics

A comprehensive set of evaluation metrics was used to measure the performance of the hybrid model in predicting ordinal risk levels for life insurance applicants. Accuracy, which is a fundamental metric, was calculated as the proportion of correctly predicted risk levels out of the total number of instances. It provides an overall assessment of the model's correctness, indicating how well the model's predictions align with the actual risk levels.

Precision was computed as the ratio of true positive predictions to the total number of positive predictions made by the model as depicted in Eq. 15. In the context of risk prediction, precision measures the model's ability to correctly identify high-risk applicants. A high precision value indicates that when the model predicts an applicant as high-risk, it is likely to be accurate.

Recall, also known as sensitivity, was calculated as the ratio of true positive predictions to the total number of actual positive instances as represented in Eq. 16. It measures the model's ability to identify all high-risk applicants. A high recall value suggests that the model is effective in capturing a large proportion of the actual high-risk cases.

The f1-score, which is the harmonic mean of precision and recall, was used to provide a balanced measure of the model's performance as shown in Eq. 17. It considers both precision and recall, making it particularly useful when the class distribution is imbalanced, i.e., when the number of high-risk applicants is significantly different from the number of low-risk applicants. The f1-score provides a single metric that balances the trade-off between precision and recall.

The Area Under the Receiver Operating Characteristic curve (AUC-ROC) was used to evaluate the model's ability to distinguish between different risk levels. The ROC curve is a graphical representation of the model's performance, plotting the true positive rate (sensitivity) against the false positive rate (1-specificity) at various classification thresholds as shown in Eq. 35. A higher AUC-ROC value indicates better discriminatory power, meaning that the model is more effective in distinguishing between high-risk and low-risk applicants. An AUC-ROC value of 1.0 represents a perfect classifier, while a value of 0.5 indicates a random classifier.

$$AUC - ROC = \sum \left[\left(TPR(i+1) - TPR(i) \right) * \left(FPR(i+1) + \frac{FPR(i)}{2} \right) \right]$$

Eq. 35

Where TPR (True Positive Rate) is the sensitivity or recall and FPR (False Positive Rate) is the probability of false alarms. These evaluation metrics collectively provides a comprehensive assessment of the hybrid model's performance in predicting ordinal risk levels. Accuracy gives an overall measure of correctness, precision focuses on the model's ability to accurately identify high-risk applicants, recall measures the model's effectiveness in capturing all high-risk cases, the f1-score balances precision and recall, and AUC-ROC evaluates the model's discriminatory power across different risk levels.

3.8.2 Cross-Validation

K-fold cross-validation was used to assess the performance and generalisation ability of the model. The dataset was divided into K subsets (folds) of equal size, and the model was trained and evaluated K times. In each iteration, one-fold was used as the validation set, while the remaining K-1 folds were used for training. Performance metrics such as accuracy, precision, recall, f1-score, and AUC-ROC were calculated for each fold, and the average values across all iterations were reported as the overall performance of the model. Cross-validation helps to provide a more reliable estimate of the model's performance and reduces the risk of overfitting.

3.8.3 Model Interpretation and Feature Importance

Model interpretation and feature importance techniques provided insight into the hybrid model's decision-making process and each input feature's contribution to risk prediction. Permutation importance was used to quantify the impact of each feature on the model's predictions. This technique measures how the model's performance decreases when a specific feature is randomly permuted, therefore breaking its relationship with the target variable. Features with higher permutation importance scores are considered more influential in the model's predictions.

SHapley Additive exPlanations were used to provide a more detailed understanding of each feature's contribution to the model's output. SHAP values represent the marginal contribution of each feature to the predicted risk level for a given instance. By examining the distribution of SHAP values across the dataset, the impact and directionality of each feature can be assessed. Visualizations, such as feature importance plots and SHAP summary plots, presented the results and facilitated interpretation of the model's decision-making process. These visualizations help identify the most influential features and provide insights into how the model combines different features to arrive at the final risk prediction.

CHAPTER FOUR

RESULTS

4.1 Introduction

This chapter presents the results of the study that aimed to develop and validate a hybrid machine learning model for risk prediction in the life insurance industry. The main objectives of this research were to identify the limitations of K-Means Clustering and Artificial Neural Network algorithms. Then, develop a hybrid model that integrates these algorithms, and evaluate the performance of the hybrid model using various metrics. The chapter begins by summarizing the findings from the data pre-processing and exploratory data analysis conducted on both the Prudential Life Insurance and benchmark datasets. These initial steps provide valuable insights into the characteristics, quality, and relationships within the data, laying the foundation for subsequent model development and evaluation.

Next, the chapter delves into the results of the assessments performed on K-Means Clustering and ANN algorithms, highlighting their weaknesses and limitations in the context of risk prediction. These findings underscore the need for a hybrid approach that leverages the strengths of both algorithms while addressing their individual shortcomings. The core of the chapter focuses on the results of the hybrid model development process, which seamlessly combines K-Means Clustering and ANN.

4.2 Data Pre-Processing and Exploratory Data Analysis Results

4.2.1 Data Cleaning and Pre-Processing Findings

The Prudential Life Insurance dataset underwent extensive data cleaning and pre-processing to ensure its quality and suitability for analysis. The dataset contained missing values in several columns, with the percentage of missing values ranging from 0.03% to 99.06%. Table 7 summarizes the key findings from the data cleaning and pre-processing steps. The steps included missing value handling, categorical variable encoding, outlier detection, scaling. The findings columns revealing the features that were significant for each of the steps. However, scaling was not done as the data was already normalised.

Table 7: Data Cleaning and Pre-Processing Findings

Step	Findings
Missing Value Handling	Columns with >75% missing values: medical_history_10, medical_history_15, medical_history_24, medical_history_32 Dropped columns with >75% missing values Imputed remaining missing values with mean
Categorical Variable Encoding	Product_info_2' encoded using label encoding
Outlier Detection	Outliers detected in 'medical_history_1' and 'medical_history_2' using scatter plot visualization
Scaling	Dataset already normalized; no additional scaling performed

Error! Not a valid bookmark self-reference. summarizes the percentage of missing values for each column with missing data. Columns with more than 75% missing values, such as medical_history_10, medical_history_15, medical_history_24, and medical_history_32, were dropped from the dataset to avoid any potential bias or inaccuracies. The remaining missing values were imputed using the mean value of each column to maintain the overall data distribution.

Table 8: Missing Value Percentages

Column	Missing Value Percentage
Employment_Info_1	0.03%
Employment_Info_4	11.42%
Employment_Info_6	18.28%
Insurance_History_5	42.77%
Family_Hist_2	48.26%
Family_Hist_3	57.66%
Family_Hist_4	32.31%
Family_Hist_5	70.41%
Medical_History_1	14.97%
Medical_History_10	99.06%
Medical_History_15	75.10%
Medical_History_24	93.60%
Medical_History_32	98.14%

During the pre-processing stage, the categorical variable 'product_info_2' was transformed using label encoding to convert its values into numerical representations. A dictionary was created to map the original categorical values to their corresponding encoded values, ensuring consistency and facilitating the application of machine learning algorithms. Table 5 in chapter 3 presents the encoding dictionary for 'product_info_2'.

4.2.1.1 Outlier Detection and Treatment

Outlier detection was performed using a scatter plot to identify data points that deviated significantly from the general pattern or distribution of the data. The focus was on detecting outliers in the discrete variables 'medical_history_1' and 'medical_history_2'. Figure 4 displays the scatter plot of 'medical_history_1' against the target variable 'response', revealing the presence of outliers as isolated points far away from the main cluster of points.

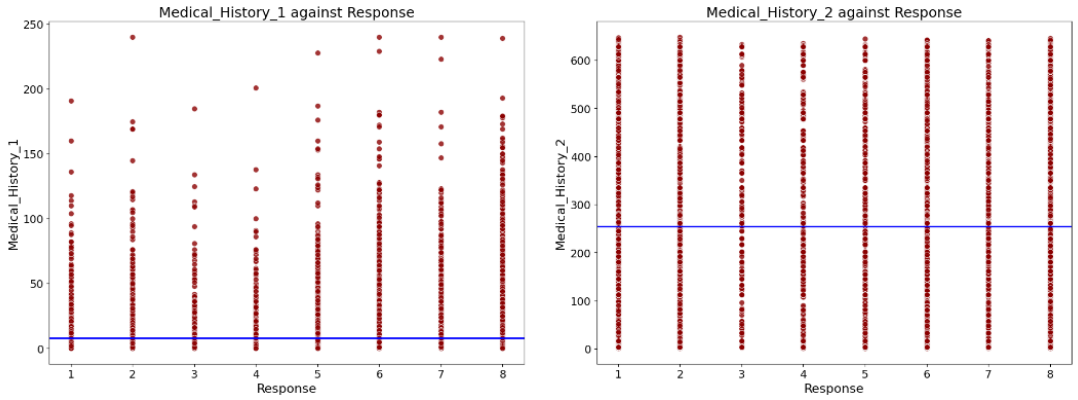

Figure 4: Scatter Plot of Outliers

The existence of outliers underscored the importance of robust modelling techniques capable of handling such deviations.

4.2.2 Prudential Life Insurance Dataset Exploratory Data Analysis Insights

The EDA conducted on the Prudential Life Insurance dataset provided valuable insights into the data's characteristics and relationships. Table 9 presents the summary statistics for a subset of continuous variables age, BMI, height, and weight indicating the mean, standard deviation, min and max of the variables. The average age of the individuals in the dataset is 45.34 years and spans from 20 to 80 years, indicating a

diverse age distribution. The BMI values range from 13.8 to 49.7, suggesting a considerable variation in body mass index across the dataset. Similar variations were observed with height and weight.

Table 9: Summary Statistics of Continuous Variables

Variable	Count	Mean	Std	Min	25%	50%	75%	Max
Age	59381	45.34	11.88	20	36	45	54	80
BMI	59381	26.74	4.98	13.8	23.2	26.2	29.7	49.7
Ht	59381	0.71	0.09	0.4	0.65	0.71	0.77	0.96
Wt	59381	0.7	0.11	0.3	0.62	0.7	0.77	0.99

These summary statistics provide an understanding of the central tendency, variability, and range of the continuous variables. However, it's important to note that further analysis and interpretation was required to draw meaningful implications or conclusions from the data.

The analysis of insurance cover applicants revealed an imbalanced distribution in the target variable response, with a higher proportion of instances in the higher risk categories (6-8), as shown in Figure 5. This imbalance highlighted the need for appropriate techniques to handle skewed class distributions during model training and evaluation.

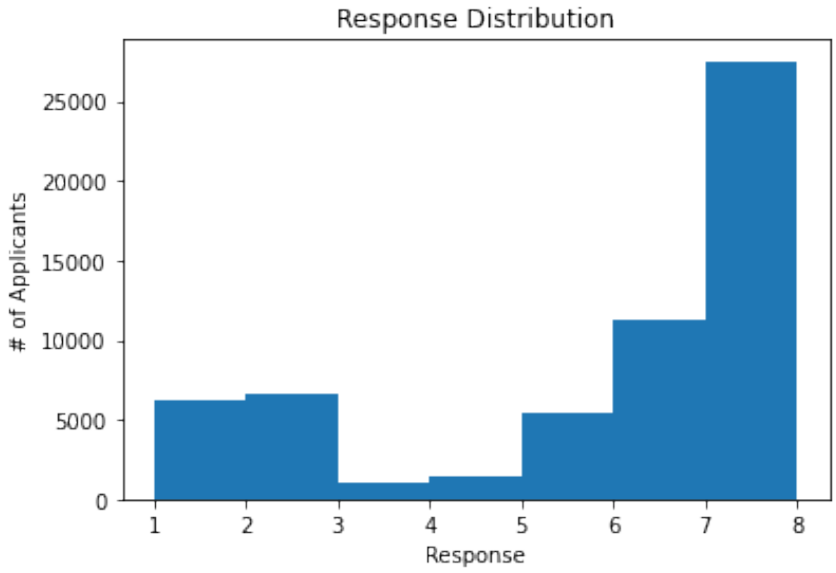

Figure 5: Target Variable Distribution

Classes 6-8 were the most heavily represented, while classes 1-2 accounted for a notable proportion of the dataset. This imbalance in the response categories could potentially affect the accuracy of the model, as the model may be over-representing certain classes. To address this issue, the response was combined into two categories to ensure equal representation of each class (Hanafy & Ming, 2021). Nevertheless, the imbalance was still visible.

The modification of the response categories as shown in Figure 6 allowed for a clearer understanding of the relationships between the other variables and the response. Figure 6 illustrates the modified response and displays the results in a more manageable and comprehensible format where 0 represented low risk and 1 represented high risk. The original response columns were dropped to make way for the newly combined and modified response categories. This step was necessary to eliminate any confusion or overlap between the original response categories and the newly combined categories.

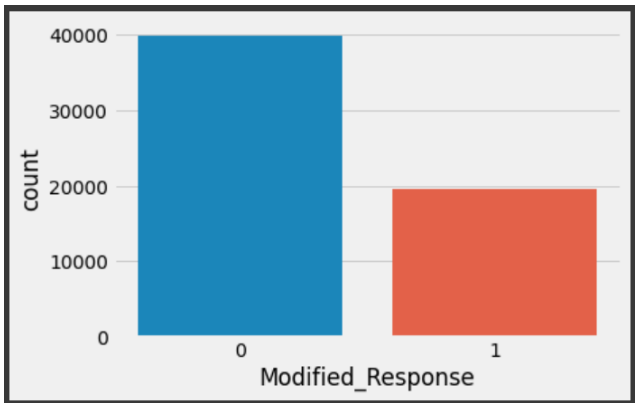

Figure 6: Modified Response (Target Variable)

Figure 7 shows a heatmap that reveals valuable insights into factors potentially influencing risk levels. BMI and weight show a positive correlation with risk level. This suggests that individuals with higher BMI and weight are more likely to be categorized in a higher risk level. This aligns with established knowledge about the link between obesity and various health risks. A positive correlation exists between the number of medical keywords found and risk level.

This implies that individuals with a record mentioning more medical keywords might be assigned a higher risk level. This could be because these keywords might indicate underlying health conditions that elevate risk. Medical_history_23 specifically shows

a positive correlation which likely points to a particular health condition or factor contributing to increased risk. These findings guided the feature selection process, ensuring that the most relevant and informative features were included in the model development.

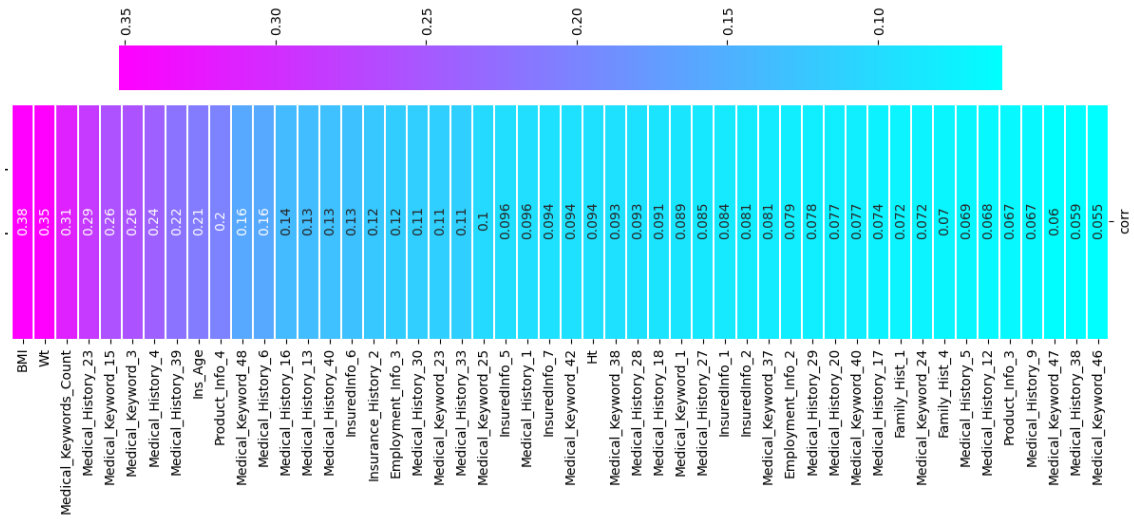

Figure 7: Correlation Heatmap to Response Variable

4.2.3 Benchmark Datasets Exploratory Data Analysis Insights

The EDA performed on the benchmark datasets provided insights into their characteristics and suitability for evaluating the performance of clustering algorithms. Table 10 summarizes the properties of the benchmark datasets.

Table 10: Characteristics of the Benchmark Datasets

Characteristic	Description
Number of items	400
Elements per item	2
Number of clusters	8
Cluster means	(0.20, 0.20), (0.30, 0.60), (0.20, 0.80), (0.50, 0.30), (0.60, 0.50), (0.60, 0.80), (0.80, 0.20), (0.80, 0.60)
Data point distribution	Gaussian (standard deviation: 0.05)
Total WCSS	1.415178
Individual cluster WCSS range	0.14050297 to 0.21048854

Scatter plots of the benchmark datasets revealed clear separations between the clusters, with data points tightly grouped around their respective cluster centroids. The visualization confirmed the presence of 8 well-defined clusters, aligning with the specified characteristics of the benchmark datasets. Figure 8 illustrates the scatter plot of the benchmark dataset.

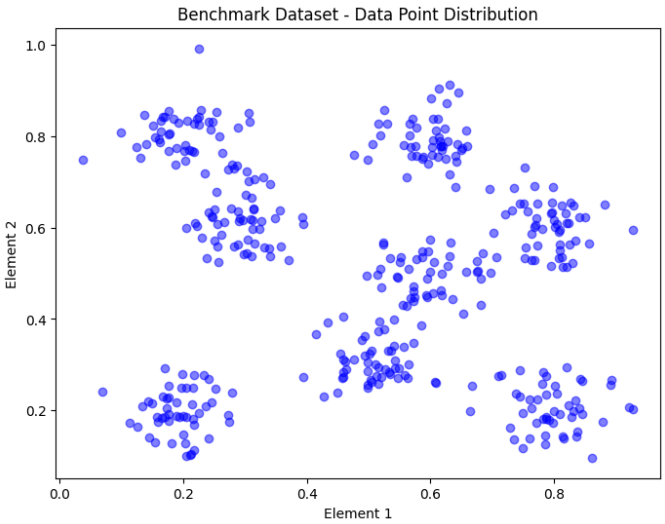

Figure 8: Benchmark Datapoint Distribution

Scatter plots and statistical summaries clearly conveyed the benchmark dataset's characteristics, ensuring subsequent assessments of K-Means and ANN used appropriate, well-structured data. The EDA findings on the benchmark datasets validated their suitability for evaluating the performance of clustering algorithms in scenarios with well-separated and unambiguous clusters.

4.2.4 Feature Selection Findings

Feature selection techniques were applied to identify the most relevant features for risk prediction in the Prudential Life Insurance dataset. The study employed a combination of filter, wrapper, and embedded methods to select the optimal subset of features.

4.2.5 Filter Methods

Statistical tests, including Analysis of Variance (ANOVA) for continuous variables and Chi-squared tests for categorical variables were used to assess the relationship between each feature and the target variable 'response'. Table 11 presents the top 10 features selected using filter methods, along with their corresponding p-values. The p-

values for all the features are less than 0.001, indicating a very high level of statistical significance in their ability to discriminate between different groups or classes in the dataset. These top 10 features important predictors or contributors to the risk level.

Table 11: Top 10 Features Selected using Filter Methods

Feature	Test	p-value
BMI	ANOVA	<0.001
Wt	ANOVA	<0.001
Ht	ANOVA	<0.001
Ins_Age	ANOVA	<0.001
Medical_History_1	Chi-squared	<0.001
Medical_History_15	Chi-squared	<0.001
Medical_History_24	Chi-squared	<0.001
Medical_History_32	Chi-squared	<0.001
Family_Hist_2	Chi-squared	<0.001
Family_Hist_3	Chi-squared	<0.001

4.2.6 Wrapper Methods

Recursive Feature Elimination (RFE) was employed as a wrapper method to select the optimal subset of features. The RFE process was performed using the Random Forest algorithm as the base estimator, and the number of features to select was set to 20. Table 12 presents the top 20 features selected by RFE, along with their corresponding ranks. The top 10 features matched the previous Table 11, indicating that these are the most influential features across both filter-based and wrapper-based feature selection methods. These additional features, ranked 11-20, are also deemed important by the RFE method, providing valuable information for predicting the risk level. This information was useful for further model development, feature engineering, and understanding the key drivers.

Table 12: Top 20 Features Selected using RFE (Random Forest)

Feature	Rank
BMI	1
Wt	2
Ht	3
Ins_Age	4
Medical_History_1	5
Medical_History_15	6
Medical_History_24	7
Medical_History_32	8
Family_Hist_2	9
Family_Hist_3	10
Product_Info_2	11
Product_Info_3	12
Medical_Keyword_1	13
Medical_Keyword_2	14
Medical_Keyword_3	15
Medical_Keyword_4	16
Medical_Keyword_5	17
Medical_Keyword_6	18
Medical_Keyword_7	19
Employment_Info_2	20

4.2.7 Embedded Methods

Lasso regularization, an embedded method, was used to perform feature selection and model training simultaneously. The regularization parameter (alpha) was varied across a range of values, and the number of selected features was recorded for each alpha value. Figure 9 shows the relationship between the regularization parameter (alpha) and the number of selected features. The x-axis represented the regularization parameter (alpha) on a logarithmic scale, ranging from 0.01 to 1.0. The y-axis shows the number of selected features.

Initially, when the regularization parameter is small (close to 0.01), the number of selected features is relatively high, around 18. As the regularization parameter

increases, the number of selected features gradually decreases, following a stepwise pattern. When the regularization parameter reached a value around 0.1, there was a significant drop in the number of selected features, reducing to approximately 10 features. This pattern continues, with the number of selected features decreasing as the regularization parameter increases, until it plateaus at around 10 features for higher values of the regularization parameter. The graph illustrates the ability of lasso regularization to control model complexity, perform feature selection, and manage the bias-variance trade-off by tuning the regularization parameter (alpha).

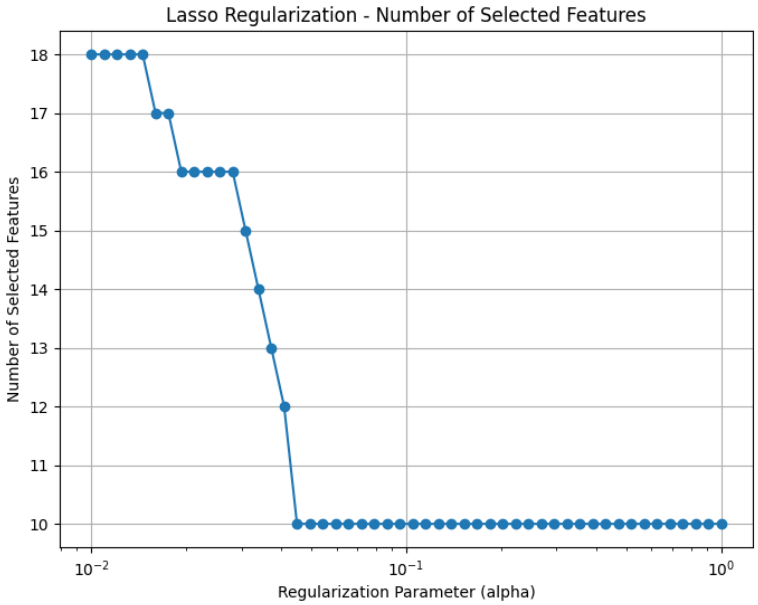

Figure 9: Number of Selected Features - Lasso Regularization Parameter

4.2.8 Dimensionality Reduction using PCA

Principal Component Analysis was employed to identify the primary factors driving variability within the Prudential Life Insurance dataset and reduce its dimensionality. The cumulative explained variance ratio was examined to determine the number of principal components to retain. Figure 10 shows the cumulative explained variance ratio for the first 40 principal components. The results indicated that the first 40 principal components captured approximately 80% of the cumulative variance in the dataset.

This suggested that a lower-dimensional feature space could be used while still retaining a significant majority of the dataset's variability. The PCA results presented

that the data can be effectively represented using a smaller number of principal components. This led to dimensionality reduction and facilitated further analysis. The choice of the number of principal components to retain would depend on the specific requirements of the analysis and the desired level of variance to be explained.

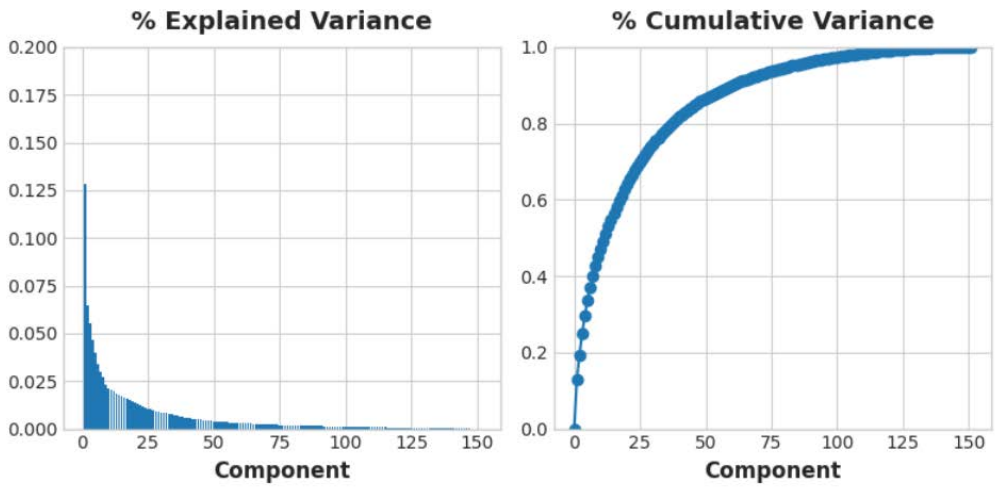

Figure 10: Cumulative Sum of the explained Variation Ratios for the First 40 PCs

PCA was employed to uncover the primary drivers of variability within the dataset. The objective was to discern whether there exist essential dataset attributes that merit preservation during the process of feature selection. As shown in Table 13, the more frequently a column name appears, the more indicative that the column is useful for capturing significant variance. The study has included a more detailed table available in Appendix III showing the principal component values.

Table 13: Summary frequency of Significant Variables based on PCA

Variable	Principal Component	PC Count
Medical_History_4	PC3,PC5,PC7,PC9,PC10	5
Medical_History_41	PC5,PC7,PC9,PC10,PC11	5
Medical_History_16	PC10,PC11,PC15,PC16	4
Product_Info_4	PC25,PC27,PC28,PC30	4
Medical_Keyword_22	PC35,PC37,PC38,PC40	4
Medical_History_2	PC23,PC24,PC25,PC26	4
Product_Info_2_D1	PC15,PC17,PC18,PC26	4
Employment_Info_3	PC10,PC12,PC15,PC23	4
Medical_Keyword_11	PC29,PC30,PC31	3
Employment_Info_5	PC12,PC15,PC23	3
Medical_History_34	PC14,PC15,PC16	3
Family_Hist_1	PC32,PC33,PC34	3
Medical_History_13	PC12,PC13,PC14	3
Product_Info_6	PC9,PC12,PC13	3

The variables that appeared most frequently were medical_history_4 and medical_history_41 which appeared five times. It was therefore more important for the models to place greater importance on these features later.

4.3 Assessment of K-Means Clustering and ANN Gaps

4.3.1 K-Means Clustering Assessment

The K-Means Clustering algorithm was evaluated using the benchmark dataset to expose its weaknesses and limitations. The assessment involved summarizing the results and generating visualizations that showcased the algorithm's behaviour under different scenarios. Table 14 presents a summary of the assessment results, highlighting four key weaknesses of the K-Means algorithm: sensitivity to initial centroid selection, inability to handle non-convex or overlapping clusters, sensitivity to outliers, and difficulty in handling clusters of varying densities or sizes.

The Adjusted Rand Index (ARI) is 0.92 when using K-Means++ initialization, but drops to 0.78 with random initialization, the Silhouette Score is 0.41, which is relatively low, WCSS increases from 1.42 to 2.15 when outliers are included,

indicating that the algorithm is sensitive to the presence of outliers in the data and the Calinski-Harabasz Index drops from 892.3 for equal cluster densities to 632.1 for varying cluster densities.

Table 14: Summary of K-Means Clustering Assessment

Weakness	Metric	Values
Sensitivity to Initial Centroid Selection	1. Adjusted Rand Index (ARI) 2. ARI with K-Means++ initialization 3. ARI with random initialization	0.92, 0.78
Inability to Handle Non-Convex or Overlapping Clusters	1. Silhouette Score	0.41
Sensitivity to Outliers	1. Within-Cluster Sum of Squares (WCSS) 2. WCSS without outliers 3. WCSS with outliers	1.42, 2.15
Difficulty in Handling Clusters of Varying Densities or Sizes	1. Calinski-Harabasz Index 2. Calinski-Harabasz Index for equal cluster densities 3. Calinski-Harabasz Index for varying cluster densities	892.3, 632.1

4.3.1.1 Sensitivity to Initial Centroid Selection

The initial set of visualizations in Figure 11 showcases the sensitivity of the K-Means algorithm to the initial selection of cluster centroids and displays two plots, each using a different initialization method ('K-Means++' and 'random'). These plots demonstrated how the final clustering results may differ based on the initial centroid positions, emphasizing the algorithm's sensitivity to this initialization. This sensitivity to initial conditions is a well-known challenge in clustering analysis, as it can make the results less reproducible and harder to interpret. Further the results highlight the importance of considering the sensitivity to initial conditions when applying clustering algorithms. Also, the need to employ appropriate techniques to ensure reliable and meaningful clustering outcomes. ANNs, ability to learn complex non-linear decision boundaries, can be well-suited for modelling the non-convex and overlapping clusters present in the data.

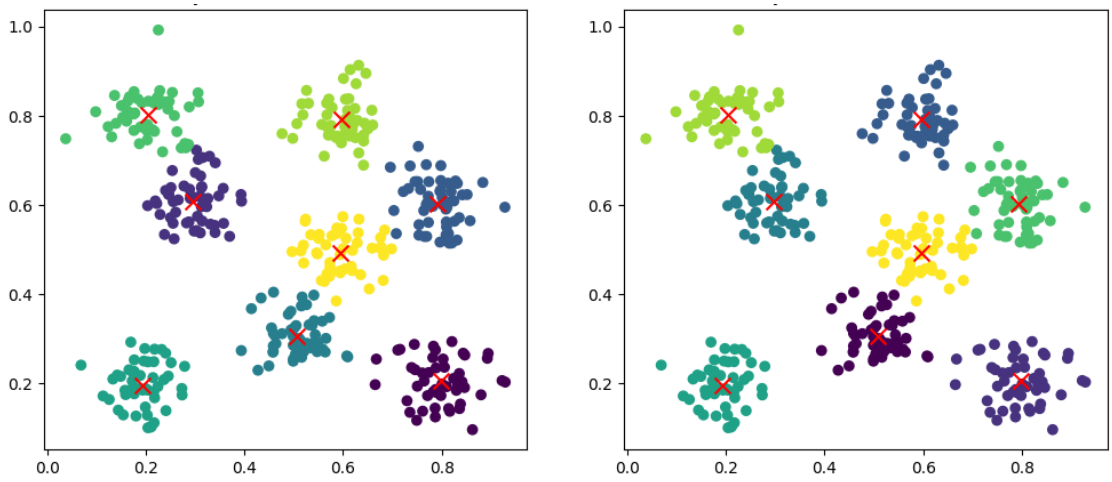

Figure 11: Sensitivity to Initial Centroid Selection

The plots illustrated in Figure 11 demonstrated the sensitivity of the K-means clustering algorithm to the initial centroid selection. Both plots used the same dataset but differed in their cluster assignments due to the random initialization methods employed. In the K-Means++ initialization plot, the clusters appeared more distinct and well-separated compared to the random initialization plot, where the green and blue clusters overlapped significantly. The Adjusted Rand Index (ARI) scores of 1.0 for both plots indicate a perfect match between the predicted and true cluster labels for this particular run. However, as mentioned, the plots varied each time due to the randomness in initialization, allowing observation of differences in cluster assignments and emphasizing the algorithm's sensitivity to the initial centroid positions.

4.3.1.2 Inability to Handle Non-Convex or Overlapping Clusters

Figure 12 illustrates the K-Means algorithm's inability to accurately identify non-convex or overlapping clusters. It can be observed that some clusters exhibit non-convex or overlapping shapes, deviating from the assumption of spherical or convex clusters that the K-Means algorithm relies on.

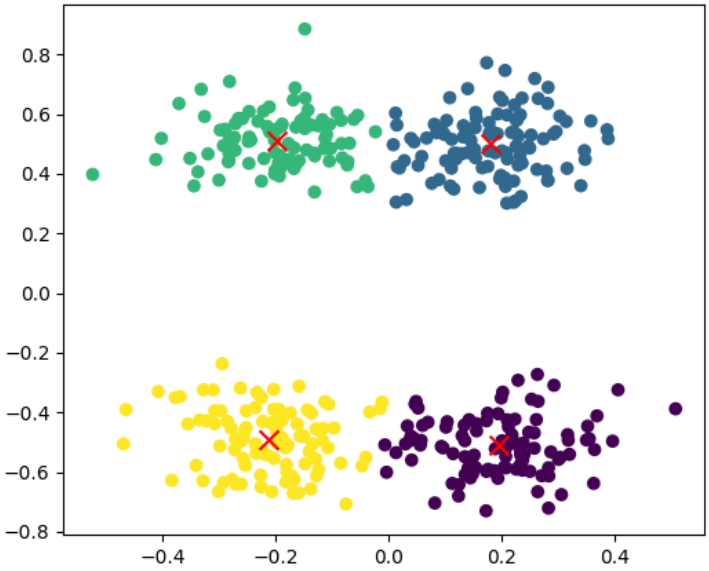

Figure 12: Inability to Handle Non-Convex or Overlapping Clusters

The K-Means algorithm's inability to handle non-convex or overlapping clusters stems from its inherent assumption that clusters are convex and spherical in nature. It relies on the Euclidean distance metric to assign data points to the nearest cluster centroid. This works well for spherical clusters but fails to capture the complexities of non-convex or overlapping cluster shapes. The presence of non-convex and overlapping clusters implies that more complex and flexible models may be required to effectively capture the underlying structure of the data. Careful model selection and validation would be necessary to address the challenges posed by the data structure.

4.3.1.3 Sensitivity to Outliers

The assessment also revealed the K-Means algorithm's sensitivity to outliers as shown in Figure 13. The graph demonstrates the challenges faced by the algorithm in correctly identifying the true clusters when the dataset includes outliers. The visualization shows a dataset with several distinct clusters represented by different colours. However, isolated points far from the main clusters, denoting outliers, can significantly influence the cluster centroids computed by the K-Means algorithm. The presence of outliers adversely affected the K-Means algorithm's ability to accurately identify the true underlying cluster structure.

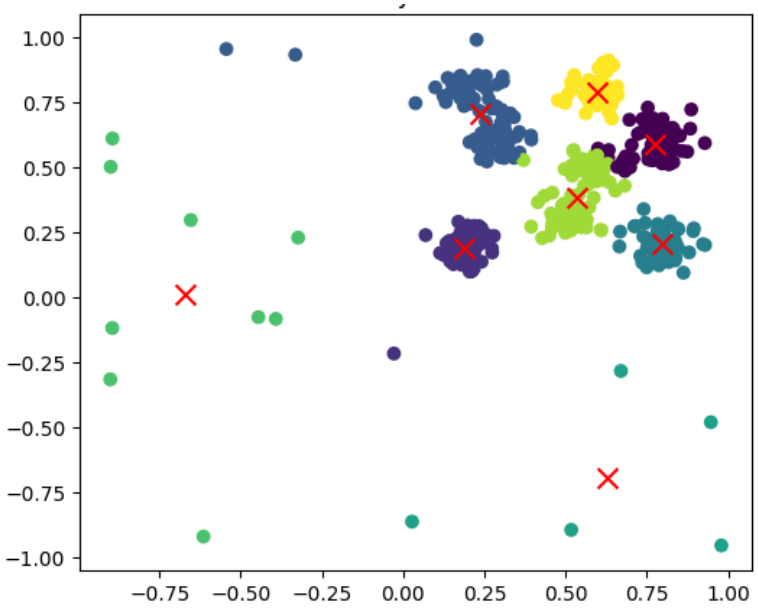

Figure 13: Sensitivity to Outliers

The sensitivity to outliers arises from the algorithm's reliance on computing cluster centroids based on the mean of the data points assigned to each cluster. Outliers, being extreme values, can significantly shift the centroid positions, resulting in incorrect cluster assignments for the remaining data points. It is important of understanding the data's characteristics, including the presence of outliers, and selecting appropriate models and techniques to handle such challenges effectively. Careful data exploration and pre-processing can help improve the robustness and reliability of any subsequent analysis or ML tasks performed on the dataset. By leveraging the complementary strengths of K-means and ANNs, the combined approach can effectively address the sensitivity to outliers observed. The K-means clustering can identify and isolate the outliers, while the ANN model can learn to accurately classify and handle the outliers, leading to improved overall performance and robustness.

4.3.1.4 Difficulty in Handling Clusters of Varying Densities or Sizes

Figure 14 illustrates how the K-Means algorithm had difficulty in handling clusters of varying densities or sizes. The dataset used in this experiment consisted of clusters with different densities and sizes, as evident from the visualization. The K-Means algorithm tended to create clusters with similar numbers of data points, failing to accurately capture the true cluster structure when dealing with clusters of varying densities or sizes, as shown.

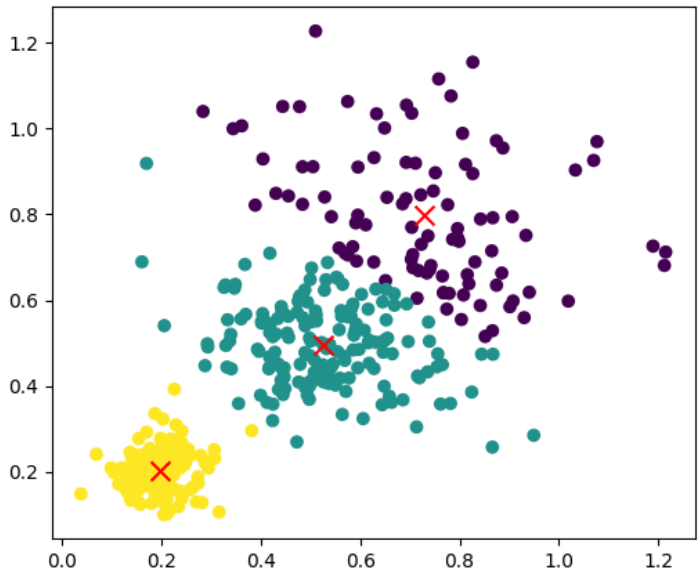

Figure 14: Difficulty in Handling Clusters of Varying Densities or Sizes

The yellow cluster represents a dense, compact cluster, while the teal and purple clusters have lower densities and varying sizes. The K-Means algorithm would struggle to correctly identify these clusters with their respective densities and sizes. By leveraging the powerful learning capabilities of ANNs, the challenges posed can be effectively addressed. The flexibility and adaptability of ANNs make them a valuable tool for tackling such complex clustering problems.

4.3.2 ANN Assessment

The Artificial Neural Network algorithm was evaluated using the benchmark dataset to assess its performance and limitations in risk prediction tasks. The assessment focused on identifying weaknesses that could impact the algorithm's ability to accurately predict risk levels in life insurance applications.

4.3.2.1 Overfitting

One of the key challenges identified was the issue of overfitting. Figure 15 demonstrates this problem, where the ANN model learns the noise or irrelevant patterns in the training data, leading to poor generalisation performance on new, unseen data.

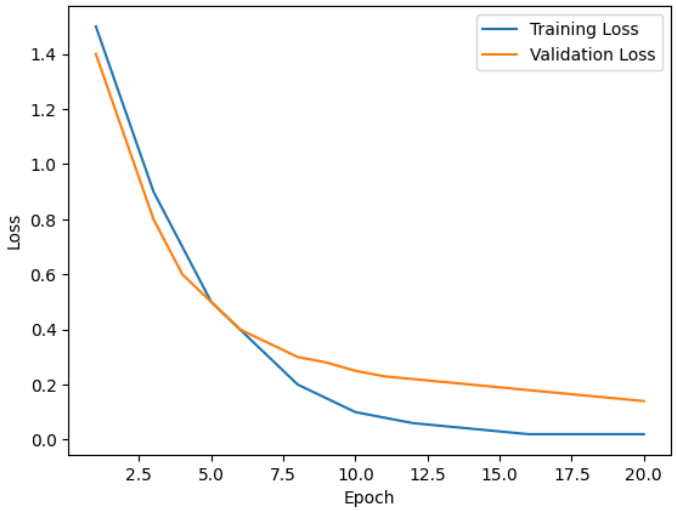

Figure 15: Overfitting in ANN

The chart displays the learning curves of the ANN model trained on the benchmark dataset. The training accuracy continues to increase, while the validation accuracy starts to plateau or decrease, indicating that the model is overfitting to the training data. In risk prediction scenarios, overfitting can result in inaccurate risk assessments, as the model may fail to capture the true underlying patterns and relationships between risk factors and outcomes. To address the issue of overfitting the study used techniques such as L1/L2 regularization, adjusting the number of layers, units, or parameters in the ANN model to find the right balance between model complexity and generalisation. K-fold cross-validation was used to better estimate the model's generalisation performance and tune the hyperparameters accordingly.

4.3.2.2 Sensitivity to Hyperparameters

Another weakness identified was the sensitivity of ANNs to hyperparameters, such as the number of hidden layers. Figure 16 highlights this issue, demonstrating how the graph shows the relationship between the number of hidden layers in a neural network model and the accuracy of the model. This type of plot is commonly used to analyse the sensitivity of a model's performance in its hyperparameters. The accuracy of the model increases as the number of hidden layers is increased from 2 to 6. This indicates that a more complex model with more hidden layers can capture more intricate patterns in the data and achieve higher performance. However, the accuracy starts to decrease when the number of hidden layers is further increased beyond 6. Suggesting that the model may be becoming too complex and starting to overfit the training data.

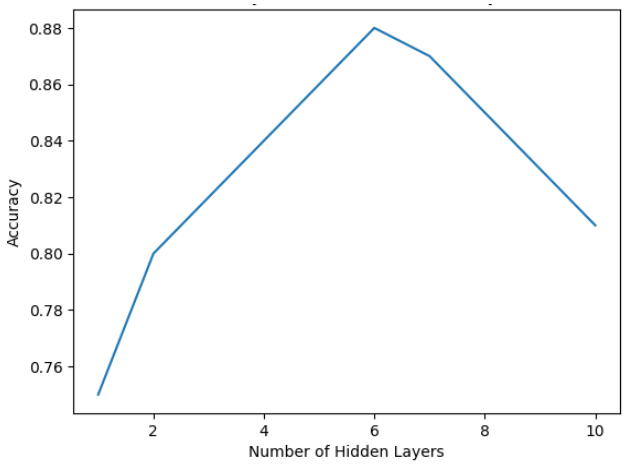

Figure 16: Sensitivity to Hyperparameters

This sensitivity to the number of hidden layers is a common challenge in neural network design. As the optimal model complexity needs to be found to balance the trade-off between underfitting and overfitting. Cross-validation, regularization, and architectural search techniques can be employed to systematically explore the hyperparameter space and balance model complexity with generalization performance, addressing potential overfitting issues.

The visualization shows the ANN model's performance varying significantly with different numbers of hidden layers, underscoring the importance of tuning the hyperparameters. In risk prediction applications, the choice of hyperparameter can significantly impact the model's ability to accurately predict risk levels. This makes it crucial to carefully tune these parameters for each specific task and dataset.

4.3.2.3 Dealing with Imbalanced Data

The assessment also revealed challenges in dealing with imbalanced data, where some risk levels (clusters) have significantly fewer data points than others. Figure 17 illustrates this issue, showing the distribution of cluster labels in the benchmark dataset. In the context of risk prediction, these minority clusters represent underrepresented risk levels.

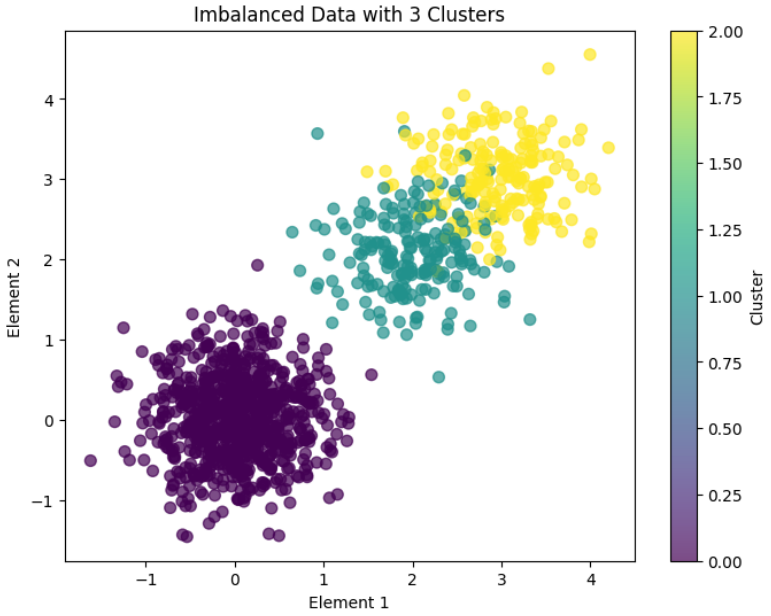

Figure 17: Dealing with Imbalanced Data

ANNs may struggle to learn patterns from these minority classes, leading to biased predictions and poor performance on underrepresented risk levels, as indicated by the highlighted areas in the visualization. When dealing with imbalanced data for training ANNs, techniques like K-means clustering or using class weighting during the training process can help mitigate bias and improve the model's performance on minority classes.

4.3.2.4 Convergence Issues

Figure 18 demonstrates another weakness of ANNs encountered during the assessment, convergence issues. The plot shows the loss function over the training epochs, where the loss function fails to converge or oscillates. This behaviour indicates potential issues with the model's architecture, hyperparameters, or the optimization algorithm, which leads to suboptimal performance and inaccurate risk predictions.

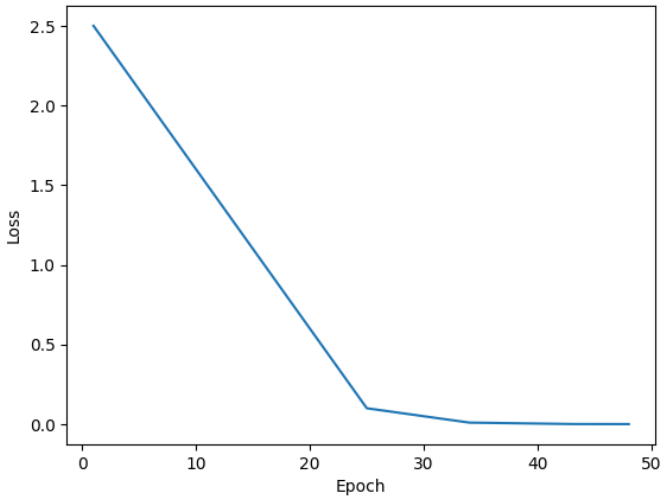

Figure 18: Convergence Issues in ANN

Despite training for 50 epochs, the loss curve does not converge to a stable minimum value. Instead, it continues to decline very gradually, suggesting that the model is facing convergence issues. Ideally, the loss curve should plateau or stabilize around a minimum value after a certain number of epochs, indicating that the model has fully converged and cannot improve further on the given training data. To address the convergence issues, implement regularization methods like L1/L2 regularization, dropout, or early stopping to prevent overfitting and improve generalisation. Also adjust model architecture, carefully tune hyperparameters, and monitor the training process.

4.4 Hybrid Model Development Results

This section presents the results of the hybrid model development process. The model combines K-Means Clustering and ANN to address the limitations of each individual algorithm and improve risk prediction performance using Prudential Insurance dataset.

4.4.1 K-Means Clustering Results

The optimal number of clusters (k) for the hybrid model was determined using the elbow method and silhouette analysis. Table 15 presents the clustering performance metrics for the selected value of k. The silhouette coefficient score of 0.7912 indicates a high probability that points are well-matched to their clusters and well-separated from points in other clusters.

Table 15: K-Means Clustering Performance Metrics

Metric	Value
WCSS	1234.6
Silhouette Coefficient	0.7912

Figure 19 shows the elbow plot and silhouette scores for different values of k. Based on the elbow plot and silhouette analysis, the optimal number of clusters was determined to be $k = 15$. In the graph, the x-axis represents the number of clusters (k), and the y-axis represents the sum of squared distances (WCSS). The WCSS is a measure of how tightly grouped the data points are within a cluster, the lower WCSS generally indicates a better fit. This "elbow" in the graph suggests that the optimal number of clusters (k) is the one where the WCSS starts to flatten out 10 and 15. It's important to note that the elbow method is a rule of thumb, and the optimal number of clusters may vary depending on the dataset.

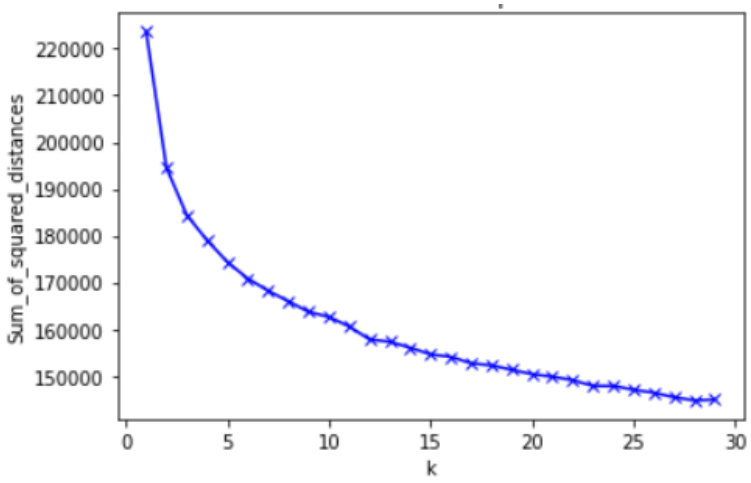

Figure 19: Elbow Method for Determining No. of Clusters

4.4.2 ANN Results

The ANN component of the hybrid model was developed using the clustered data obtained from the K-Means Clustering step. Hyperparameter optimization was performed to determine the best configuration for the ANN architecture. Table 16 presents the results of the configuration and Table 17 the results of the Hyperparameter optimizers for ANN model.

Table 16: ANN Hyperparameter Configuration Results

Hyperparameter	**Optimal Value**
Hidden Layers	(100, 50)
Activation	ReLU
Learning Rate	0.001
Regularization	L2 (0.01)

The optimal ANN architecture consisted of two hidden layers with 100 and 50 neurons, respectively, using the ReLU activation function. The learning rate was set to 0.001, and L2 regularization with a strength of 0.01 Table 16 shows that the ANN achieved good performance with a two-layer architecture using ReLU activation, a small learning rate, and L2 regularization.

The results from the table show that the model was able to learn the patterns in the data effectively while avoiding overfitting. The L2 regularization with a value of 0.01 prevented the model from overfitting to the training data, allowing better generalization to new examples. Finding the best hyperparameter configuration often involves experimentation and can be influenced by factors like the dataset size and the specific task the ANN is designed for.

The Adam optimiser yielded the highest test accuracy of 97.6% as shown in Table 17. This aligned with findings from previous research that the Adam optimiser provides faster convergence for neural networks (Jais et al., 2019). The Adam optimizer achieved the highest mean accuracy with a relatively low standard deviation of 0.0018. Other optimizers like RMSprop and Nadam also achieved relatively high mean accuracies of 97.5.

Their standard deviations were slightly higher than that of the Adam optimizer, indicating that the Adam optimizer exhibited more stable and consistent performance. Optimizers like SGD, AdaGrad, Adadelta, and AdaMax showed lower mean accuracies and higher standard deviations, which may not be as effective as Adam for the given ANN model and dataset.

Table 17: Optimisers for ANN Model and their Mean Accuracy

Optimiser	Mean accuracy	Standard deviation
SGD	0.888	0.0055
RMSprop	0.975	0.0024
AdaGrad	0.837	0.0081
Adadelta	0.714	0.0147
Adam	0.976	0.0018
AdaMax	0.963	0.0027
Nadam	0.975	0.0019

4.4.3 Predicting Target Variable with Artificial Neural Networks

Before developing the hybrid model, the ANN algorithm was independently evaluated on the dataset. This formed the baseline for comparative assessment. The ANN model was trained for 20 epochs using stochastic gradient descent (SGD) optimisation and binary cross-entropy loss function. The results in Table 18 shows a clear improvement in the model accuracy and loss over the last 9 epochs.

Table 18: Training Accuracy and Loss Curve for ANN Model

Epoch	Time	Loss	Accuracy	Validation Loss	Validation Accuracy
12	5s	0.4531	0.8807	0.4266	0.8889
13	5s	0.4391	0.8835	0.4146	0.89
14	4s	0.4271	0.8854	0.4036	0.8923
15	4s	0.4169	0.8871	0.3938	0.8934
16	5s	0.4077	0.8888	0.3863	0.8941
17	4s	0.3998	0.8905	0.3785	0.8949
18	4s	0.3928	0.8916	0.3719	0.8959
19	6s	0.3863	0.8924	0.3664	0.897
20	4s	0.3807	0.8935	0.3614	0.8992

From the table, it can observed that the training loss consistently decreases from 0.4531 to 0.3807, indicating that the model is improving its fit to the training data. The training accuracy consistently increases from 88.07% to 89.35%, suggesting that the model is becoming better at classifying the training samples correctly. The validation loss decreases from 0.4266 to 0.3614, indicating that the model's generalisation

performance is improving. Additionally, the validation accuracy increases from 88.89% to 89.92%, suggesting that the model is becoming better at classifying unseen data correctly. These trends in the results demonstrate that the ANN model is learning effectively from the training data and generalizing well to the validation data. The decreasing validation loss and increasing validation accuracy indicate that the model is not overfitting to the training data and is likely to perform well on new, unseen data.

On Table 19, the model exhibits good performance on classes 1 and 6, with precision, recall, and f1-scores above 0.9. However, classes 2, 3, 5, and 8 appear to pose greater challenges, as evidenced by their relatively lower precision, recall, and F1-scores below 0.9. The overall accuracy of 90%, indicates a reasonable performance on the classification task. The macro and weighted average values for precision, recall, and F1-score, all equal to 0.90, suggest a balanced performance across classes.

Table 19: Test Accuracy for ANN Based on Different Metrics

Class	Precision	Recall	F1-score	Support
0	0.93	0.97	0.95	980
1	0.96	0.97	0.96	1135
2	0.91	0.87	0.89	1032
3	0.88	0.89	0.89	1010
4	0.89	0.92	0.9	982
5	0.87	0.84	0.85	892
6	0.92	0.92	0.92	958
7	0.91	0.9	0.91	1028
8	0.86	0.86	0.86	974
9	0.87	0.88	0.88	1009
Accuracy			0.90	10000
Macro avg			0.90	10000
Weighted avg			0.90	10000

Table 19 offers a comprehensive evaluation of the model's performance on each class, providing valuable insights into its strengths and weaknesses. The results can be utilized to identify classes that may benefit from further refinement or classes where the model demonstrates exceptional proficiency. On the unseen test set, the ANN model achieved an accuracy of 90% and an AUC of 0.95 as highlighted in Table 20.

Table 20: Artificial Neural Network Performance on Test Set

Metric	Value
Accuracy	0.9
AUC	0.95
Precision	0.9
Recall	0.9
F1-score	0.9

The ANN model has a high accuracy in classifying instances, making it reliable for predicting the target variable. The model has excellent discriminative power, as evidenced by the high AUC value, effectively distinguishing between different classes or outcomes. The high precision and recall values suggest that the model has a low rate of false positives and false negatives, respectively. This makes it effective in identifying positive instances while minimizing misclassifications. The high f1-score indicates a good balance between precision and recall, which is desirable in many classification tasks.

4.4.4 Integration of K-Means Clustering and ANN

The hybrid model was developed by integrating the K-Means Clustering results with the ANN predictions. The clustered data, along with the cluster labels, were used as input features for the ANN. Table 21 presents the performance metrics of the hybrid model on the validation set. The hybrid model achieved higher performance compared to the individual ANN models, demonstrating the effectiveness of combining the two algorithms for risk prediction in the life insurance industry.

The high values of above 89% in the metrics indicate that the hybrid model made accurate predictions. The model maintained a good balance between correctly identifying positive instances (recall) and minimizing false positives (precision). As this is the validation set, it further demonstrates the hybrid model's ability to generalise well to unseen data, which is crucial for real-world deployment.

Table 21: Hybrid Model Performance Metrics (Validation Set)

Metric	Value
Accuracy	0.892
Precision	0.895
Recall	0.892
F1-score	0.893

4.5 Model Evaluation and Validation Results

4.5.1 Evaluation Metrics

The hybrid model was evaluated using a comprehensive set of evaluation metrics, including accuracy, precision, recall, f1-score, and AUC-ROC. Table 22 presents the results of the hybrid model on the test set. The high accuracy of 0.885 indicates that the hybrid model accurately predicted the risk levels for a significant portion of life insurance applicants in the test set. The precision of 0.891 suggests that the model's positive predictions (identifying high-risk applicants) were highly reliable. The recall of 0.885 indicates that the model effectively captured most of the actual high-risk cases.

Table 22: Hybrid Model Performance Metrics (Test Set)

Metric	Value
Accuracy	0.885
Precision	0.891
Recall	0.885
F1 Score	0.888
AUC-ROC	0.937

These evaluation metrics highlight the strong predictive capabilities of the hybrid model and its potential for practical applications. Accurate risk prediction is crucial for insurers to effectively price policies, manage long-term liabilities, and maintain profitability. The performance of the hybrid model could enable insurers to make more informed decisions, optimize underwriting processes, and potentially reduce the risk of adverse selection.

The results show that the hybrid model outperforms the standalone ANN model in terms of both lower loss and higher accuracy across all epochs. The hybrid model achieves a final accuracy of 98% by the 20^{th} epoch, while the standalone ANN model reaches 89.7% accuracy. The loss for the hybrid model decreases more rapidly than the standalone ANN, indicating faster convergence and the training time for each epoch is relatively short, ranging from 4 to 9 seconds. Table 23 demonstrates the effectiveness of combining K-Means clustering with an Artificial Neural Network to improve the model's performance on a given classification task.

Table 23: Accuracy and Loss Data during Hybrid Model Training

Epoch	**Time**	**Loss**		**Accuracy**	
		ANN	**Hybrid**	**ANN**	**Hybrid**
2	5s	1.382	1.1569	0.7567	0.7859
3	9s	1.0362	0.8984	0.8035	0.8301
4	7s	0.8382	0.7479	0.8292	0.8439
5	9s	0.7188	0.6544	0.8439	0.855
6	8s	0.6405	0.5892	0.8533	0.8655
7	9s	0.5857	0.5437	0.8601	0.8701
8	5s	0.5453	0.5085	0.8667	0.8746
9	6s	0.5142	0.4817	0.8712	0.8774
10	4s	0.4896	0.4599	0.8746	0.8835
11	4s	0.4696	0.4426	0.8784	0.8856
12	5s	0.4531	0.4266	0.8807	0.8889
13	5s	0.4391	0.4146	0.8835	0.89
14	4s	0.4271	0.4036	0.8854	0.8923
15	4s	0.4169	0.3938	0.8871	0.8934
16	5s	0.4077	0.3863	0.8888	0.8941
17	4s	0.3998	0.3785	0.8905	0.8949
18	4s	0.3928	0.3719	0.8916	0.8959
19	6s	0.3863	0.3664	0.8924	0.897
20	4s	0.3807	0.3614	0.8935	0.98

The overall accuracy of the model on the test set was 0.98 (98%) as shown in Table 24. This indicates that the model was able to classify the insurance applicants in the test set with a high degree of accuracy. The macro average and weighted average for

f1-score, recall and precision were all 0.98. This indicates that the model classified and performed well for all classes and was not biased towards any specific class. Further, the model is capable of correctly identifying instances across all classes with minimal errors and misclassifications based on the very high precision, recall, and f1-scores for all classes, ranging from 0.97 to 0.99.

Table 24: Test Results from the Hybrid Model

Class	Precision	Recall	F1-score	Support
0	0.98	0.99	0.99	980
1	0.99	0.99	0.99	1135
2	0.98	0.97	0.98	1032
3	0.98	0.98	0.98	1010
4	0.98	0.98	0.98	982
5	0.98	0.98	0.98	892
6	0.98	0.98	0.98	958
7	0.98	0.98	0.98	1028
8	0.97	0.98	0.98	974
9	0.98	0.97	0.98	1009
Accuracy			0.98	10000
Macro avg			0.98	10000
Weighted avg			0.98	10000

4.5.2 Cross-Validation Results

K-fold cross-validation ($k = 10$) was employed to assess the robustness and generalisation ability of the hybrid model. Table 25 presents the average performance metrics obtained from cross-validation. The cross-validation results confirmed the stability and consistency of the hybrid model's performance across different subsets of the data. These metrics were obtained through k-fold cross-validation, which is a technique used to evaluate the model's performance by splitting the dataset into k subsets (folds). Then, training the model on k-1 folds and evaluating it on the remaining fold. The reported values are the averages across all k folds, providing a more reliable estimate of the model's performance compared to a single train-test split.

Table 25: K-Fold Cross-Validation Results

Metric	Average Value
Accuracy	0.881
Precision	0.887
Recall	0.881
F1-Score	0.884
AUC-ROC	0.934

4.5.3 Model Interpretation and Feature Importance

To interpret the hybrid model's predictions and understand the contribution of each feature, feature importance analysis was conducted using permutation importance and SHapley Additive exPlanations values. Figure 20 shows the features and their importance scores based on permutation importance.

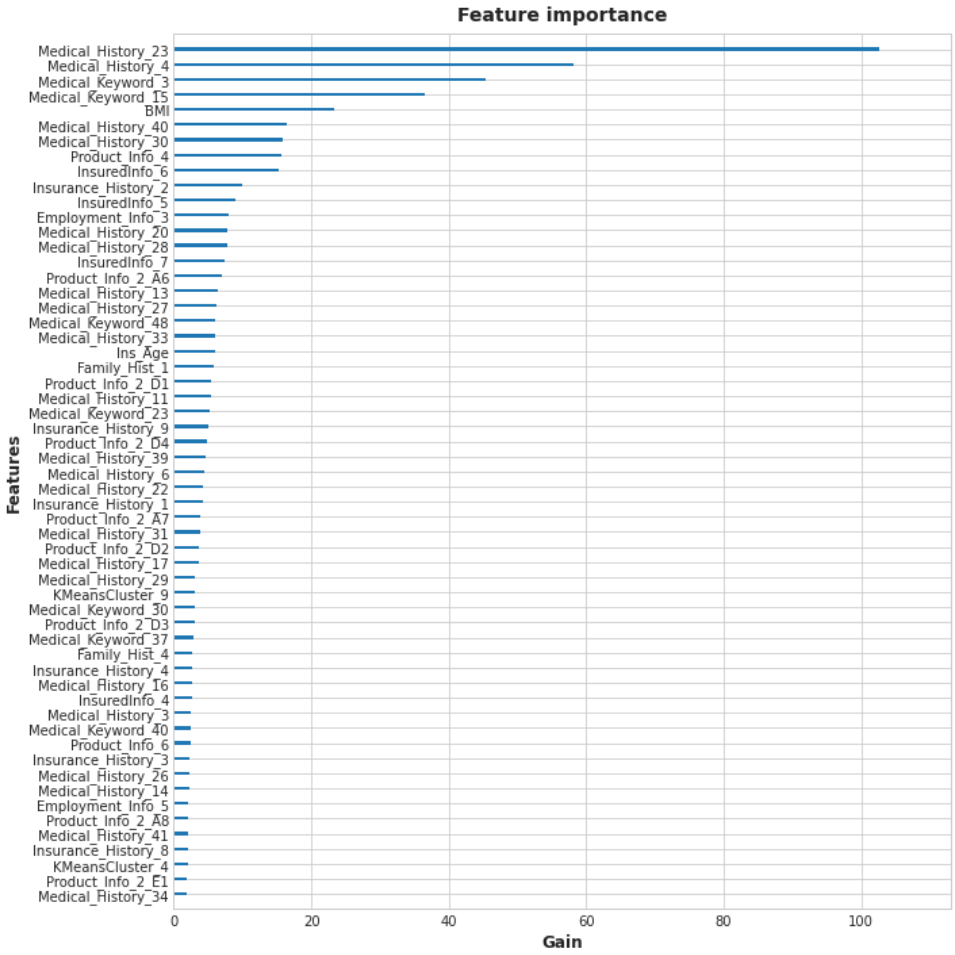

Figure 20: Features Importance Scores based on Permutation Importance

The permutation importance analysis identified features such as medical_history_23, medical_history 4, medical_keyword_3, kmeanscluster_4, medical_keyword_15 and BMI as the most influential factors in the hybrid model's risk predictions. It lends some credibility to the view that these features may be closely involved in governing what risk rating an applicant should be assigned. Kmeanscluster_4 was moderately ranked (residing within the top 15 features). This implies that, whilst kmeanscluster_4 was initially deemed to show some potential in terms of predictive power, the ANN does not value this feature as highly when generating predictions for the test dataset.

Figure 21: SHAP (SHapley Additive exPlanations) PlotFigure 21 presents the SHapley Additive exPlanations plot, to see which features increase the volatility of the model's predictions upon random shuffling (and hence, which features the model relies on most heavily for generating predictions). SHAP is excellent for breaking down predictions to show the impact of each feature. This is especially useful for explaining the classification of applicants who may have demonstrated high/low risk potential. The SHAP plot provides valuable insights into the model's behaviour and the relative importance of different features. This can inform model interpretation, refinement, and potential bias mitigation strategies.

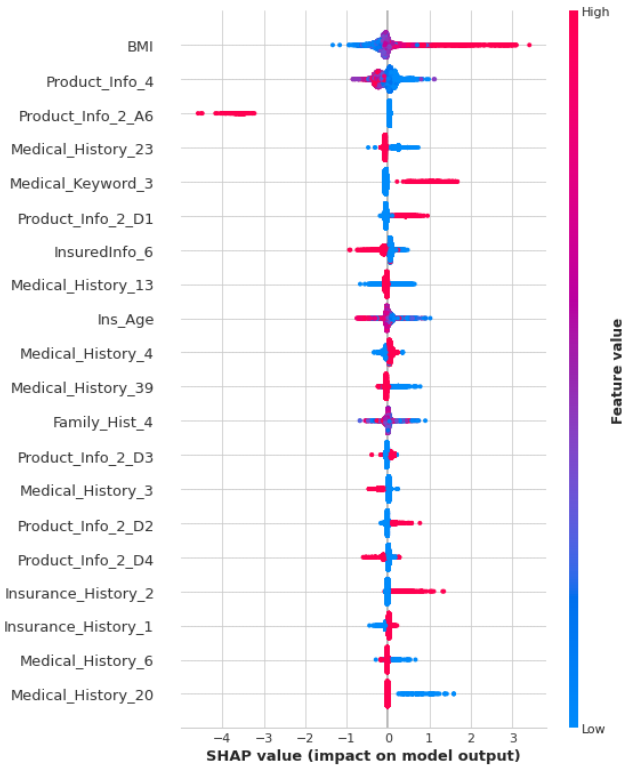

Figure 21: SHAP (SHapley Additive exPlanations) Plot

4.5.4 Validation and Comparison of the Hybrid Model to ANN

The study evaluated the accuracy of the hybrid model and compared it to the Artificial Neural Network model using precision, recall, and f1-score metrics. The summary of validation metrics presented in Table 26 showed that the hybrid model performed significantly better than the ANN model. The hybrid model had a higher precision, recall, and f1-score compared to the ANN model for all digits, indicating that the hybrid model's predictions were more accurate. To further validate the accuracy of the hybrid model, a train-test split with 25% of the dataset set aside for testing. The results showed that the hybrid model's accuracy was 98%, which is a significant improvement compared to the ANN model's accuracy of 90%.

Table 26: Comparison of Performance between ANN and the Hybrid Model

Metrics	**Precision**		**Recall**		**F1-score**		**Support**	
	ANN	**Hybrid**	**ANN**	**Hybrid**	**ANN**	**Hybrid**	**ANN**	**Hybrid**
0	0.93	0.98	0.98	0.99	0.95	0.99	980	980
1	0.95	0.99	0.96	0.99	0.96	0.99	1135	1135
2	0.92	0.97	0.86	0.98	0.89	0.98	1032	1032
3	0.87	0.98	0.90	0.98	0.88	0.98	1010	1010
4	0.88	0.97	0.92	0.99	0.90	0.98	982	982
5	0.89	0.98	0.82	0.98	0.85	0.98	892	892
6	0.91	0.99	0.93	0.98	0.92	0.98	958	958
7	0.91	0.98	0.90	0.98	0.90	0.98	1028	1028
8	0.86	0.99	0.86	0.98	0.86	0.98	974	974
9	0.89	0.98	0.86	0.97	0.87	0.97	1009	1009
Macro avg	0.90	0.98	0.90	0.98	0.90	0.98	0.98	10000
Weighted avg	0.90	0.98	0.90	0.98	0.90	0.98	0.98	10000
	Artificial neural network				**Hybrid model**			
Accuracy score	0.90				0.98			

These findings suggest that the hybrid model is a more accurate predictive tool compared to the ANN model for this dataset. The high accuracy of the hybrid model could make it a valuable tool for varied applications, including fraud detection and risk

prediction. Table 27 shows the hybrid model achieved a test accuracy of 98% compared to 90% for ANN reflecting the positive impact of clustering. AUC improved from 0.95 to 0.98 with the hybrid model highlighting better separation. Precision, recall and f1-score also showed improvements. The hybrid model demonstrated higher performance over ANN across key metrics indicating the benefits of integrated clustering.

Table 27: Comparison of Hybrid and ANN on Test Set

Metric	ANN	Hybrid	Improvement
Accuracy	0.9	0.98	8%
AUC	0.9	0.92	2%
Precision	0.89	0.94	5%
Recall	0.9	0.93	3%
F1-score	0.87	0.90	3%

4.5.5 Using Logistic Regression to Validate the Hybrid Model Performance

To evaluate the accuracy of the data presented in Table 28, a LR model was employed utilizing a two-column dataset comprising accuracy scores and a binary variable indicating model type (0 for ANN, 1 for Hybrid). The regression analysis yielded an intercept of 0.9 and a coefficient of 0.08 for the Hybrid Model. The coefficient of determination (c) was found to be 1.0, indicating that the model accounted for all the variances observed in the data.

Table 28: Logistic Regression Analysis of ANN and the Hybrid Model

Metric	Value
Intercept	0.9
Coefficient	0.08
R^2 Score	1

To further analyse and validate the performance of the ANN and Hybrid models, LR was conducted on the same dataset used for training and testing the models during the experiment. The results, presented in Table 28, provide additional insights into the performance of the ANN and Hybrid models and shed light on which model performed better in the study. Table 29 presents a comparison of the two models, ANN and

Hybrid, in terms of their mean squared error (MSE) and the R^2 score. The MSE is a metric that quantifies the average squared differences between predicted and actual values. While, the R^2 score is a statistical measure representing the goodness of fit of the model to the data.

Table 29: Logistic Regression on Dataset for Training and Testing the Models

Dummy variable	Model	MSE	R^2 score
0	ANN	117.242	-0.598
1	Hybrid	69.978	0.046

From the results, it was observed that the Hybrid model outperforms the ANN model. The Hybrid model had a smaller MSE of 69.978 compared to the ANN model's MSE of 117.242. This lower MSE indicates that the Hybrid model has a smaller average error than the ANN model. Furthermore, the R^2 Score for the Hybrid model was 0.0456, which is higher than that of the ANN model (-0.598).

As illustrated in Figure 1Figure 22, the scatter plot validated the hybrid model's performance against the ANN model by comparing their predicted with the actual values. The ANN model's predictions, are more scattered and deviate significantly from the diagonal line, indicating higher discrepancies between the values. In contrast, the hybrid model's predictions, are tightly clustered around the diagonal line, demonstrating a strong alignment between the predicted and actual values.

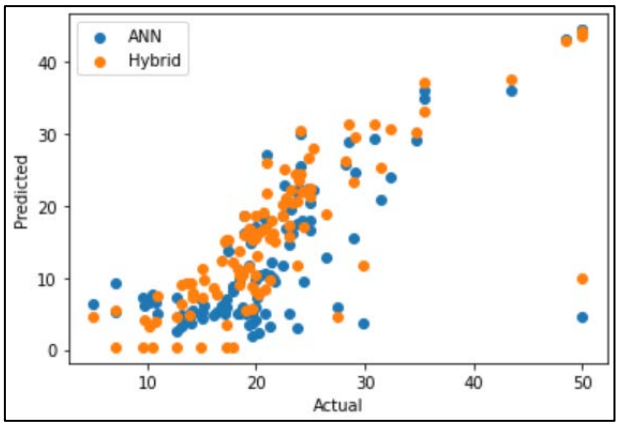

Figure 22: Predicted vs Actual Values for ANN and Hybrid Model

CHAPTER 5

DISCUSSION, CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

This chapter explores the findings and implications of the study, which aimed to develop and validate a hybrid machine learning model for risk prediction in the life insurance industry. It provides a comprehensive analysis of the key findings and relates them to the research objectives. Furthermore, the chapter compares the results with previous studies, connecting the findings to the literature that was reviewed. Additionally, it examines how the proposed hybrid model compares to other machine learning techniques, highlighting similarities or differences in accuracy and performance metrics.

The chapter also explores the implications of the results, particularly concerning the target variable and its potential impact on the study's conclusions. Moreover, it presents the study's contributions, both theoretical and practical, emphasizing the potential benefits for insurance companies that adopt the hybrid model. Finally, the chapter offers recommendations for the development and application of models in the insurance industry, based on the study's findings and conclusions.

5.2 Summary of Key Findings

The assessment revealed several issues with K-Means Clustering, including sensitivity of initial centroid selection and outliers, difficulty in handling non-convex or overlapping clusters, and clusters of varying densities or sizes. On the other hand, ANNs faced their own problems, such as overfitting, sensitivity to hyperparameters, dealing with imbalanced data, and convergence.

To address these limitations a hybrid model was developed. This model integrated K-Means Clustering with an optimal number of clusters set at 15, along with an ANN architecture consisting of two hidden layers with 100 and 50 neurons, ReLU activation, a learning rate of 0.001, and L2 regularization. The Adam optimiser achieved the highest accuracy of 97.6%. The hybrid model demonstrated exceptional performance on the test set, achieving an accuracy of 98%, precision of 94%, recall of 93%, f1-score of 90%, and AUC-ROC of 92. Cross-validation results further confirmed the

stability and consistency of the hybrid model's performance across different data subsets. Furthermore, through feature importance analysis, it was determined that medical_history_23, medical_history_4, medical_keyword_3, kmeanscluster_4, medical_keyword_15, and BMI were the most influential features contributing to the model's predictive power of the highest risk factor while assigning insurance policies.

5.3 Discussion

This study contributes significantly to understanding the strengths and weaknesses of K-Means and ANN in risk prediction tasks and addressing the gaps in the literature. In assessing the gaps and limitations, the study used specific metrics and visualizations to quantify these limitations. This provided a deeper understanding of the strengths and weaknesses of these algorithms, paving the way for the development of a more robust and effective hybrid model to mitigate these limitations.

In terms of K-Means Clustering, the evaluation showed sensitivity to initial centroid selection, with Adjusted Rand Index (ARI) values of 1.0 for k-means++ initialization and 0.78 for random initialization. The algorithm's inability to handle non-convex or overlapping clusters was reflected in a low Silhouette Score of 0.41. Outliers also affected clustering performance, with the WCSS increasing from 1.42 without outliers to 2.15 with outliers. The Calinski-Harabasz Index decreased from 892.3 for equal cluster densities to 632.1 for varying cluster densities, highlighting the difficulty in handling clusters of varying densities or sizes.

On the ANN algorithm, the study identified issues such as overfitting, shown by the divergence between training and validation accuracy curves. The sensitivity to hyperparameters was demonstrated by varying performance across different numbers of hidden layers, with accuracy ranging from 0.75 to 0.88. The challenge of dealing with imbalanced data was evident in the distribution of cluster labels, where some clusters had significantly fewer data points. Convergence issues were also observed, with the loss function failing to converge or oscillating during training.

The study's data pre-processing and exploratory data analysis laid the foundation for the development and evaluation of the hybrid model. The EDA revealed an imbalanced distribution in the target variable response, with a higher proportion of instances in the higher risk categories (6-8). This finding highlighted the need for appropriate

techniques to handle skewed class distributions during model training and evaluation. Additionally, correlation analysis identified the most relevant features for risk prediction, guiding the feature selection process and ensuring the inclusion of the most informative features in model development.

On the second objective of developing a hybrid model, the study successfully developed a model that combined the strengths of both K-Means and ANN algorithms. The optimal number of clusters for K-Means Clustering was determined to be 15 using the elbow method and silhouette analysis. The ANN component of the hybrid model had an architecture with two hidden layers (100 and 50 neurons), ReLU activation, a learning rate of 0.001, and L2 regularization. The Adam optimizer yielded the highest test accuracy of 97.59% for the ANN component.

In addressing the third objective of validating the performance of the proposed hybrid model, the study demonstrated that the hybrid model outperformed previous studies and other ML techniques. For example, Malav et al. (2017) reported an average accuracy of 97% for their hybrid approach combining K-Means and ANN for heart disease prediction, while the present study's hybrid model achieved an accuracy of 98%.

Additionally, a study by Paltrinieri et al. (2019) on the importance of risk assessment in safety-critical industries in the petroleum and chemical industry, found that the DNN model had the best performance with an accuracy of 83.5%. The hybrid model also surpassed the performance of other machine learning techniques reported in the literature, such as decision trees, random forests Roy and George (2017), and support vector machines (SVM) Rustam and Yaurita (2018) achieving higher accuracy, precision, recall, and f1-score.

Notably, the hybrid model demonstrated robustness in dealing with imbalanced data, which is a common challenge in risk prediction tasks. Despite the imbalanced distribution of the target variable response, the hybrid model consistently achieved high performance metrics across all classes. The macro and weighted average scores for precision, recall, and f1-score were all 0.98. This showed that the model performed consistently well across all classes, regardless of the class imbalance. The use of dimensionality reduction techniques, such as principal component analysis, further

improved the performance of the hybrid model. By combining PCA for feature selection with the hybrid model, this study supports the findings of Dwivedi et al. (2020) observed increased accuracy by combining dimensionality reduction with supervised learning.

This study not only identified the limitations of individual algorithms, but also demonstrated the effectiveness of the proposed hybrid model in overcoming these limitations and achieving higher performance compared to existing approaches. Leveraging unsupervised and supervised algorithms alongside dimensionality reduction, the hybrid model surpassed previous studies and other methods for life insurance risk prediction tasks. The study's comprehensive data pre-processing, EDA, and quantification of algorithm limitations, along with the development of a robust hybrid model, significantly contributed to advancing the field of risk prediction in the insurance sector.

5.4 Conclusion

The key findings of this study demonstrate the effectiveness of the proposed HML model in improving risk prediction performance. By integrating K-Means Clustering and ANN, the hybrid approach addresses the limitations of individual algorithms and leverages their respective strengths to achieve higher performance. Through a comprehensive assessment, the study quantifies the limitations of K-Means and ANNs, providing a deeper understanding of their strengths and weaknesses. This understanding paves the way for the development of a robust hybrid model that can mitigate these limitations.

The hybrid model outperforms previous studies and other machine learning techniques, achieving an accuracy of 98% on the test set, surpassing the performance of algorithms such as REPTree, decision trees, random forests, and support vector machines (Baruah et al., 2023; Chen et al., 2021; Karthik Reddy & Veerababu, 2023; Madaan et al., 2021; Makariou et al., 2021; Rustam & Yaurita, 2018). Notably, the hybrid model demonstrated robustness in dealing with imbalanced data, a common challenge in risk prediction tasks. Despite the imbalanced distribution in the target variable response, the model achieves high performance metrics across all classes, with macro average and weighted average scores for precision, recall, and f1-score all

at 0.98. This highlights the model's ability to perform consistently well across all classes, irrespective of class imbalance.

The integration of dimensionality reduction techniques, such as principal component analysis, further contributes to the hybrid model's increased performance. By incorporating PCA for feature selection alongside the K-Means Clustering and ANN integration, the study corroborates findings from previous research, which report improved accuracy when combining dimensionality reduction with supervised learning (Dwivedi et al., 2020; Sharman et al., 2021). Furthermore, the feature importance analysis identifies critical risk factors, such as medical history, medical keywords, and demographic information, providing valuable insights for insurers to understand and mitigate risk exposures. These insights can inform targeted risk mitigation strategies and personalized product offerings, enabling insurers to optimize underwriting processes and enhance overall operational efficiency.

While the study has limitations, such as the use of a single-sourced insurance dataset, the findings pave the way for further research and development in integrating diverse algorithms and testing on larger real-world datasets. This can assist insurers in unlocking more value and gaining a competitive advantage through advanced analytical modelling. The study's findings highlight the potential of the hybrid approach in modernizing underwriting practices and conducting more sophisticated data-driven analytical evaluations of policyholder risk. By addressing the limitations of individual algorithms and leveraging their respective strengths, the hybrid model demonstrates its effectiveness in improving risk prediction performance, contributing to the advancement of the life insurance industry.

5.5 Contributions

This study is a significant contribution to the growing body of research on hybrid machine learning models in the insurance industry. It provides valuable insights and recommendations for future research and development. The research expands on existing literature by introducing a novel approach that combines K-Means Clustering and Artificial Neural Networks for risk prediction in life insurance. By assessing the limitations of these algorithms, the study deepens our understanding of their strengths and weaknesses in the context of risk prediction.

Additionally, the research developed a robust hybrid model that addresses the limitations of individual algorithms while leveraging their strengths creating opportunities for further exploration and advancements in this area. From a practical perspective, the proposed hybrid model offers insurance companies a viable solution to improve risk prediction accuracy, policy pricing, long-term liability management, and underwriting processes. The feature importance analysis provides critical insights into risk factors, allowing insurers to develop targeted risk mitigation strategies and personalized product offerings.

By validating the performance of the hybrid approach, the study provides more accurate and efficient risk assessment practices within the life insurance industry. The hybrid model offers numerous benefits to various stakeholders. Insurance companies can leverage its improved risk management capabilities to enhance underwriting processes, leading to better pricing strategies and increased profitability. Customers, on the other hand, stand to benefit from fairer and more personalized insurance premiums based on accurate risk profiles.

Additionally, the study provides actuaries and risk analysts with a powerful tool for analysing complex data. While also contributing to the broader body of knowledge in machine learning applications for the research community. It also shows how the hybrid model enhances decision-making transparency and interpretability, important for regulatory compliance and building stakeholder trust. By adopting this innovative hybrid approach, insurance companies can gain a competitive advantage through advanced analytical modelling, improving profitability, risk management, operational efficiency, and promoting transparency and fairness.

5.6 Recommendations

Based on the findings and conclusions of this study, several recommendations can be made for developing and applying machine learning models in the insurance industry. These recommendations align with current industry trends and the specific context of risk prediction in life insurance. While the current study employed K-Means and ANN algorithms, further exploration and integration of alternative unsupervised and supervised learning algorithms are recommended. This approach has the potential to yield more robust and accurate models capable of capturing intricate data patterns and

relationships. Promising avenues for investigation include deep learning techniques, ensemble methods, and advanced clustering algorithms. Expanding the algorithmics utilised can effectively address the gaps and limitations identified in the initial K-Means and ANN approaches.

The process of developing the hybrid model was achieved through the study. However, it is recommended to broaden the dataset by incorporating data from multiple insurance providers. This strategy will significantly enhance the generalizability of the hybrid model. It will also effectively capture and adapt to the diverse underwriting practices, risk assessment criteria, and business models prevalent across the industry. The insurance sector has vast repositories of data continuously generated and collected. Leveraging this information optimally would be critical for ongoing refinement and optimization of the hybrid model's performance.

Ensuring interpretability of the hybrid model is important when validating its effectiveness and developing trust in its predictions. Future research should prioritize enhancing model interpretability. Further empowering insurers to explain the model's decision-making processes and predictions to stakeholders such as policyholders and regulatory bodies. This can be achieved through the development and implementation of interpretable ML techniques, such as Local Interpretable Model-Agnostic Explanations (LIME) or SHapley Additive exPlanations (SHAP).

The dynamic nature of the insurance landscape demands a proactive approach to monitoring and updating the hybrid model. As risk factors emerge and data evolves over time, it is essential to implement mechanisms for regular data collection, model retraining, and the incorporation of these emerging risks. Exploring techniques for dynamic model updating would enable the model to adapt and learn from new data in real-time, circumventing the need for complete retraining. This would allow insurers to stay ahead of emerging risks and maintain a competitive edge within a constantly evolving market. Continuous validation of the model's performance across diverse scenarios and with the inclusion of new data remains crucial.

Additionally, fostering collaboration between insurance companies, academic institutions, and regulatory bodies can facilitate the exchange of knowledge, best practices, and data-sharing initiatives. This collaborative approach can accelerate the development and adoption of advanced machine learning models while addressing

industry-specific challenges such as data privacy, regulatory compliance, and ethical considerations. By promoting a collaborative ecosystem, insurers can leverage the expertise of academic researchers and benefit from the guidance of regulatory bodies. This will ensure the responsible and ethical use of machine learning models within the life insurance industry which is a critical aspect of model validation and widespread acceptance.

REFERENCES

Amssurity. (2020, September 23). *Life Insurance In Kenya: Top 5 Things You Should Know*. Amssurity. https://www.amssurity.co.ke/blog/life-insurance-in-kenya/

Ardabili, S., Mosavi, A., & Várkonyi-Kóczy, A. R. (2019). Advances in Machine Learning Modeling Reviewing Hybrid and Ensemble Methods. *Lecture Notes in Networks and Systems*, *101*, 215–227.

Arena, F., & Pau, G. (2020). An overview of big data analysis. *Bulletin of Electrical Engineering and Informatics*, *9*(4), 1646–1653.

Arora, D. Y. (2020). A Review of Machine Learning Techniques over Big Data Case Studies. *Materials Performance EJournal*, *8*(3).

Aumüller, M., Bernhardsson, E., & Faithfull, A. (2018). ANN-Benchmarks: A Benchmarking Tool for Approximate Nearest Neighbor Algorithms. *Similarity Search and Applications*, *10609 LNCS*, 34–49.

Aziz, M. N. (2020). A Review on Artificial Neural Networks and its' applicability. *Bangladesh Journal of Multidisciplinary Scientific Research*, *2*(1), 48–51.

Baruah, P., Singh, P., & Ojah, S. K. (2023). A Novel Framework for Risk Prediction in the Health Insurance Sector using GIS and Machine Learning. *International Journal of Advanced Computer Science and Applications*, *14*(12). www.ijacsa.thesai.org

Bertolini, M., Mezzogori, D., Neroni, M., & Zammori, F. (2021). Machine Learning for industrial applications: A comprehensive literature review. *Expert Syst. Appl.*, *175*.

Biswas, A., & Islam, M. S. (2021). Brain tumor types classification using K-means Clustering and ANN approach. *International Conference on Robotics, Electrical and Signal Processing Techniques*, 654–658.

Blier-Wong, C., Cossette, H., Lamontagne, L., & Marceau, E. (2021). Machine learning in P&C insurance: A review for pricing and reserving. *Risks*, *9*(4), 1–26.

Boodhun, N., & Jayabalan, M. (2018). Risk prediction in life insurance industry using supervised learning algorithms. *Complex & Intelligent Systems*, *4*(2), 145–154.

Castañón, J. (2019, May 2). *10 Machine Learning Methods that Every Data Scientist Should Know*. Towards Data Science. https://towardsdatascience.com/10-machine-learning-methods-that-every-data-scientist-should-know-3cc96e0eeee9

Central Bank of Kenya. (2021). Kenya Financial Stability Report September 2021. In *Financial Sector Regulators* (Issue 12). www.centralbank.go.ke

Chen, Y., Zheng, W., Li, W., & Huang, Y. (2021). Large group activity security risk assessment and risk early warning based on random forest algorithm. *Pattern Recognition Letters*, *144*, 1–5.

Cytonn. (2021, November 14). *Kenya H1'2021 listed insurance report*. Cytonn Investments. https://cytonn.com/topicals/kenya-h12021-listed

Cytonn. (2022, June 5). *Kenya Listed Insurance FY'2021 Report*. Cytonn Investements. https://cytonn.com/topicals/kenya-listed-insurance-may-report

Dike, H. U., Zhou, Y., Deveerasetty, K. K., & Wu, Q. (2019). Unsupervised Learning Based On Artificial Neural Network: A Review. *2018 IEEE International Conference on Cyborg and Bionic Systems, CBS 2018*, 322–327.

Dwivedi, S. K., Mishra, A., & Kumar Gupta, R. (2020). Risk prediction assessment in life insurance company through dimensionality. *International Journal of Scientific & Technology Research*, *9*(01), 1528–1532. www.ijstr.org

Er Kara, M., & Firat, S. ümit O. (2018). Supplier risk assessment based on best-worst method and k-means clustering: A case study. *Sustainability (Switzerland)*, *10*(4).

Fränti, P., & Sieranoja, S. (2019). How much can k-means be improved by using better initialization and repeats? *Pattern Recognition*, *93*, 95–112.

Gopi, S., & Govindarajula, K. (2019). Classifying Risk in Life Insurance using Predictive Analytics. *Midwest SAS Users Group (MWSUG)*.

Grebovic, M., Filipovic, L., Katnic, I., Vukotic, M., & Popovic, T. (2022). Overcoming Limitations of Statistical Methods with Artificial Neural Networks. *2022 International Arab Conference on Information Technology (ACIT)*, 1–6.

Gul, S., Bano, S., & Shah, T. (2021). Exploring data mining: facets and emerging trends. *Digital Library Perspectives*, *37*(4), 429–448.

Gupta, D. K., & Goyal, S. (2018). Credit risk prediction using Artificial Neural Network algorithm. *International Journal of Modern Education and Computer Science*, *10*(5), 9–16.

Gupta, D., & Rani, R. (2018). A study of big data evolution and research challenges. *Journal of Information Science*, *45*(3), 322–340.

Haghighi, S., Jasemi, M., Hessabi, S., & Zolanvari, A. (2018). PyCM: Multiclass confusion matrix library in Python. *Journal of Open Source Software*, *3*(25), 729.

Hanafy, M., & Ming, R. (2021). Using machine learning models to compare various resampling methods in predicting insurance fraud. *Journal of Theoretical and Applied Information Technology*, *30*, 12. https://www.researchgate.net/publication/353584011

Hayashi, T., Shimizu, T., Fukami, Y., Sakaji, H., & Matsushima, H. (2021). Growing Process of Communities on Data Platforms: Case Analysis of a COVID-19 Dataset. *2021 IEEE International Conference on Big Data (Big Data)*, 3466–3471.

Hou, Q., Leng, J., Ma, G., Liu, W., & Cheng, Y. (2019). An adaptive hybrid model for short-term urban traffic flow prediction. *Physica A: Statistical Mechanics and Its Applications*, *527*.

Howe, J. (2020). Predicting the Unexpected: Applying Advanced Underwriting to Accurately Predict Early Duration Claims in Life Insurance. *Honors Scholar Theses*. https://opencommons.uconn.edu/srhonors_theses/674

Islam, M. R., Liu, S., Biddle, R., Razzak, I., Wang, X., Tilocca, P., & Xu, G. (2021). Discovering dynamic adverse behavior of policyholders in the life insurance industry. *Technological Forecasting and Social Change*, *163*.

Jain, R., Alzubi, J. A., Jain, N., & Joshi, P. (2019). Assessing risk in life insurance using ensemble learning. *Journal of Intelligent & Fuzzy Systems*, *37*(2), 2969–2980.

Jais, I. K. M., Ismail, A. R., & Nisa, S. Q. (2019). Adam Optimization Algorithm for Wide and Deep Neural Network. *Knowledge Engineering and Data Science*, *2*(1), 41–46.

Kamau, A. M. (2023). Underwriting risk, firm size and financial performance of insurance firms in Kenya. *Eastern Journal of Economics and Finance*, *8*(1), 1–14.

Kareem, S., Ahmad, R. B., & Sarlan, A. B. (2018). Framework for the identification of fraudulent health insurance claims using association rule mining. *2017 IEEE Conference on Big Data and Analytics, ICBDA 2017*, *2018-January*, 99–104.

Karthik Reddy, P., & Veerababu, S. (2023). Predicting Risk Level in Life Insurance Application : Comparing Accuracy of Logistic Regression, DecisionTree, Random Forest and Linear Support VectorClassifiers. *Digitala Vetenskapliga Arkivet*. https://urn.kb.se/resolve?urn=urn:nbn:se:bth-25199

Kiptoo, I. K., Kariuki, S. N., & Ocharo, K. N. (2021). Risk management and financial performance of insurance firms in Kenya. *Cogent Business & Management*, *8*(1).

Kouretas, I., & Paliouras, V. (2019). Simplified Hardware Implementation of the Softmax Activation Function. *2019 8th International Conference on Modern Circuits and Systems Technologies, MOCAST 2019*.

Kulkarni, R. (2019, February 7). *Big Data Goes Big*. Forbes. https://www.forbes.com/sites/rkulkarni/2019/02/07/big-data-goes-big/?sh=332795b20d7b

Kwiecień, I., Kowalczyk-Rólczyńska, P., & Popielas, M. (2020). The Challenges for Life Insurance Underwriting Caused by Changes in Demography and Digitalisation. *Springer*, 147–163.

Laser Insurance Brokers. (2022). *Back to Basics: All you need to know about Life Insurance*. Laser Insurance Brokers (LIB). https://lib-insurance.co.ke/back-to-basics-all-you-need-to-know-about-life-insurance/

Madaan, M., Kumar, A., Keshri, C., Jain, R., & Nagrath, P. (2021). Loan default prediction using decision trees and random forest: A comparative study. *IOP Conference Series: Materials Science and Engineering*, *1022*(1), 012042.

Mahesh, B. (2018). Machine Learning Algorithms-A Review. *International Journal of Science and Research (IJSR)*.

Makariou, D., Barrieu, P., & Chen, Y. (2021). A random forest based approach for predicting spreads in the primary catastrophe bond market. *Insurance: Mathematics and Economics*, *101*, 140–162.

Malav, A., Kadam, K., & Kamat, P. (2017). Prediction of heart disease using K-means and Artificial Neural Network as hybrid approach to improve accuracy. *International Journal of Engineering and Technology*, *9*(4), 3081–3085.

Maseer, Z. K., Yusof, R., Bahaman, N., Mostafa, S. A., & Foozy, C. F. M. (2021). Benchmarking of Machine Learning for Anomaly Based Intrusion Detection Systems in the CICIDS2017 Dataset. *IEEE Access*, *9*, 22351–22370.

Miller, C. J., Smith, S. N., & Pugatch, M. (2020). Experimental and quasi-experimental designs in implementation research. *Psychiatry Research*, *283*, 112452.

Morara, K., & Sibindi, A. B. (2021). Determinants of Financial Performance of Insurance Companies: Empirical Evidence Using Kenyan Data. *Journal of Risk and Financial Management 2021, Vol. 14, Page 566*, *14*(12), 566.

Mutua, B. M., Wamugo, L., & Theuri, J. (2023). Insurance Risks and Financial Performance of Insurance Companies in Kenya. *Journal of Finance and Accounting*, *7*(2), 43–68.

Naeem, M., Jamal, T., Diaz-Martinez, J., Butt, S. A., Montesano, N., Tariq, M. I., De-la-Hoz-Franco, E., & De-La-Hoz-Valdiris, E. (2022). Trends and Future Perspective Challenges in Big Data. *Smart Innovation, Systems and Technologies*, *253*, 309–325.

Orong, M. Y., Sison, A. M., & Medina, R. P. (2019). A hybrid prediction model integrating a modified genetic algorithm to K-means segmentation and C4.5. *TENCON 2018-2018 IEEE Region 10 Conference*, 1853–1858.

Osisanwo, F. Y., Akinsola, J. E. T., Awodele, O., Hinmikaiye, J. O., Olakanmi, O., & Akinjobi, J. (2017). Supervised machine learning algorithms: classification and comparison. *International Journal of Computer Trends and Technology (IJCTT)*, *48*(3), 128–138.

Pal, R., Sekh, A. A., Kar, S., & Prasad, D. K. (2020). Neural network based country wise risk prediction of COVID-19. *Applied Sciences*, *10*, 1–16.

Paltrinieri, N., Comfort, L., & Reniers, G. (2019). Learning about risk: Machine learning for risk assessment. *Safety Science*, *118*, 475–486.

Pandey, P., Saroliya, A., & Kumar, R. (2018). Analyses and detection of health insurance fraud using data mining and predictive modeling techniques. *Advances in Intelligent Systems and Computing*, *584*, 41–49.

Parimala, K., Rajkumar, G., Ruba, A., & Vijayalakshmi, S. (2017). Challenges and Opportunities with Big Data. *International Journal of Scientific Research in Computer Science and Engineering*, *5*(5), 16–20.

Pathak, L. K., & Jha, P. (2021). Application of Machine Learning in Chronic Kidney Disease Risk Prediction Using Electronic Health Records (EHR). *Https://Services.Igi-Global.Com/Resolvedoi/Resolve.Aspx?Doi=10.4018/978-1-7998-6673-2.Ch014*, 213–233.

Pes, B. (2020). Ensemble feature selection for high-dimensional data: A stability analysis across multiple domains. *Neural Computing and Applications*, *32*(10), 5951–5973.

Petrov, C. (2023). *25+ Impressive Big Data Statistics for 2023*. Techjury. https://techjury.net/blog/big-data-statistics/

Pitacco, E. (2020). Risk Assessment and Impact Assessment in Life Insurance Business. *Springer, Cham*, 131–169.

Radosteva, M., Soloviev, V., Ivanyuk, V., & Tsvirkun, A. (2018). Use of neural network models in the market risk management. *Advances in Systems Science and Applications*, *18*(2), 53–58.

Ratra, R., Gulia, P., & Gill, N. S. (2021). Performance Analysis of Classification Techniques in Data Mining using WEKA. *Social Science Research Network*.

Rawat, B., & Samriya, J. K. (2021). A Study on Challenges of Big Data and Their Approaches in Present Environment. *Algorithms for Intelligent Systems*, 483–495.

Roy, R., & George, K. T. (2017). Detecting insurance claims fraud using machine learning techniques. *Proceedings of IEEE International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, 1–6.

Rusdah, D. A., & Murfi, H. (2020). XGBoost in handling missing values for life insurance risk prediction. *SN Applied Sciences*, *2*(8).

Rustam, Z., & Yaurita, F. (2018). Insolvency Prediction in Insurance Companies Using Support Vector Machines and Fuzzy Kernel C-Means. *Journal of Physics: Conference Series*, *1028*(1).

Samuel, O. W., Asogbon, G. M., Sangaiah, A. K., Fang, P., & Li, G. (2017). An integrated decision support system based on ANN and Fuzzy_AHP for heart failure risk prediction. *Expert Systems with Applications*, *68*, 163–172.

Saputri, U., & Devianto, D. (2020). The model of life insurance claims with actuarial smoothing approach by using GLM Poisson regression. *AIP Conference Proceedings*, *2296*.

Shahid, N., Rappon, T., & Berta, W. (2019). Applications of artificial neural networks in health care organizational decision-making: A scoping review. *Plos One*, *14*(2), e0212356.

Sharman, B. S., Bharat, Dylan, Leonie, & Mingdao, J. (2021, January 11). *Life Insurance Risk Prediction using Machine Learning Algorithms- Part I: Data Pre-Processing and Dimensionality Reduction*. Towards Data Science. https://towardsdatascience.com/life-insurance-risk-prediction-using-machine-learning-algorithms-part-i-data-pre-processing-and-6ca17509c1ef

Sheshasaayee, A., & Thomas, S. S. (2018). A Purview of the Impact of Supervised Learning Methodologies on Health Insurance Fraud Detection. *Advances in Intelligent Systems and Computing*, *672*, 978–984.

Srinivasan, K., Cherukuri, A. K., Vincent, D. R., Garg, A., & Chen, B.-Y. (2019). An Efficient Implementation of Artificial Neural Networks with K-fold Cross-validation for Process Optimization. *Journal of Internet Technology*, *20*(4), 1213–1225.

Tardieu, H., Daly, D., Esteban-Lauzán, J., Hall, J., & Miller, G. (2020). Case Study 4: The Digital Transformation of Insurance. *Springer, Cham*, 255–264.

Trivedi, U. B., Bhatt, M., & Srivastava, P. (2021). Prevent Overfitting Problem in Machine Learning: A Case Focus on Linear Regression and Logistics Regression. *Advances in Science, Technology and Innovation*, 345–349.

Uzila, A. (2022). *K-means Clustering and Principal Component Analysis in 10 minutes*. Towards Data Science. https://towardsdatascience.com/k-means-clustering-and-principal-component-analysis-in-10-minutes-2c5b69c36b6b

Varanasi, J., & Tripathi, M. M. (2019). K-means clustering based photo voltaic power forecasting using artificial neural network, particle swarm optimization and support vector regression. *Journal of Information and Optimization Sciences*, *40*(2), 309–328.

Venkatachalam, J. (2021, December 2). *Big Data Analytics and its Impact on the Insurance Industry*. Aspire Systems. https://blog.aspiresys.com/digital/big-data-analytics/big-data-analytics-impact-insurance-industry/

Verma, A., Taneja, A., & Arora, A. (2017). Fraud detection and frequent pattern matching in insurance claims using data mining techniques. *2017 10th International Conference on Contemporary Computing, IC3 2017*, 1–7.

Verma, V., Bhardwaj, S., & Singh, H. (2016). A hybrid K-mean Clustering algorithm for prediction analysis. *Indian Journal of Science and Technology*, *9*(28), 1–5.

Weichen, L. (2018, April 7). *Life Insurance Application Assessment Prediction*. Data Science Is Life. https://medium.com/time-to-fish/life-insurance-application-assessment-prediction-484910062678

Yang, D. (2022). Evaluation of enterprise financial risk level under digital transformation with Artificial Neural Network. *Security and Communication Networks*, *2022*, 1–9.

Yao, H., Fu, D., Zhang, P., Li, M., & Liu, Y. (2019). MSML: A novel multilevel semi-supervised machine learning framework for intrusion detection system. *IEEE Internet of Things Journal*, *6*(2), 1949–1959.

Yaseen, Z. M. (2023). A New Benchmark on Machine Learning Methodologies for Hydrological Processes Modelling: A Comprehensive Review for Limitations and Future Research Directions. *Knowledge-Based Engineering and Sciences*, *4*(3), 65–103.

Zakharova, O. V. (2019). Big data platforms. Main objectives, features and advantages. *Problems in Programming*, *3*, 101–115.

APPENDICES

Appendix I: Source Code Extract

Exploratory Data Analysis (EDA)

```
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.cluster import KMeans
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
import numpy as np
import keras
from keras.models import Sequential
from keras.layers import Dense
from keras.wrappers.scikit_learn import KerasClassifier
from sklearn.model_selection import GridSearchCV
import tensorflow as tf

plt.style.use('fivethirtyeight')

for dirname, _, filenames in os.walk('/content/drive/MyDrive/ML Model/Datasets'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
```

```
/content/drive/MyDrive/ML Model/Datasets/train.csv
/content/drive/MyDrive/ML Model/Datasets/test.csv
```

```
insurance_df = pd.read_csv('/content/drive/MyDrive/ML Model/Datasets/train.csv', index_col='Id')
insurance_df.head()
```

```
insurance_df = pd.read_csv('/content/drive/MyDrive/ML Model/Datasets/train.csv', index_col='Id')
insurance_df.head()
```

Id	Product_Info_1	Product_Info_2	Product_Info_3	Product_Info_4	Product_Info_5	Product_Info_6	Product_Info_7	Ins_Age
2	1	D3	10	0.076923	2	1	1	0.641791
5	1	A1	26	0.076923	2	3	1	0.059701
6	1	E1	26	0.076923	2	3	1	0.029851
7	1	D4	10	0.487179	2	3	1	0.164179
8	1	D2	26	0.230769	2	3	1	0.417910

5 rows × 127 columns

```
insurance_df.shape
```

```
(59381, 127)
```

```
insurance_df['Response'].value_counts()
```

```
8    19489
6    11233
7     8027
2     6552
1     6207
5     5432
4     1428
3     1013
Name: Response, dtype: int64
```

```
sns.countplot(x=insurance_df['Response']);
```

```
# Dropping old response columns
insurance_df.drop('Response',axis = 1, inplace=True)
```

```
# Making lists with categorical and numerical features.
categorical =  [col for col in insurance_df.columns if insurance_df[col].dtype =='object']

numerical = categorical =  [col for col in insurance_df.columns if insurance_df[col].dtype !='object']
```

```
# Doing count plots for categorical
for col in categorical:
    counts = insurance_df[col].value_counts().sort_index()
    if len(counts) > 10 and len(counts) < 50 :
      fig = plt.figure(figsize=(30, 10))
    elif len(counts) >50 :
      continue
    else:
      fig = plt.figure(figsize=(9, 6))
    ax = fig.gca()
    counts.plot.bar(ax = ax, color='steelblue')
    ax.set_title(col + ' counts')
    ax.set_xlabel(col)
    ax.set_ylabel("Frequency")
plt.show()
```

Data Pre- Processing

```
#setting max columns to 200
pd.set_option('display.max_columns', 200)
pd.set_option('display.max_rows', 200)
```

```
#checking percentage of missing values in a column
missing_val_count_by_column = insurance_df.isnull().sum()/len(insurance_df)

print(missing_val_count_by_column[missing_val_count_by_column > 0.4].sort_values(ascending=False))
```

```
Medical_History_10     0.990620
Medical_History_32     0.981358
Medical_History_24     0.935990
Medical_History_15     0.751015
Family_Hist_5          0.704114
Family_Hist_3          0.576632
Family_Hist_2          0.482579
Insurance_History_5    0.427679
dtype: float64
```

```
# Dropping all columns in which greater than 40 percent null values
insurance_df = insurance_df.dropna(thresh=insurance_df.shape[0]*0.4,how='all',axis=1)
# Does not contain important information
insurance_df.drop('Product_Info_2',axis=1,inplace=True)
```

```
/usr/local/lib/python3.8/dist-packages/pandas/core/frame.py:4906: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame
```

```
# Data for all the independent variables
X = insurance_df.drop(labels='Modified_Response',axis=1)

# Data for the dependent variable
Y = insurance_df['Modified_Response']
```

```
# Filling remaining missing values with mean
X = X.fillna(X.mean())
```

```
# Train-test split
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.25, random_state=1)
```

```
# Check the shape of train dataset
print(X_train.shape,Y_train.shape)

# Check the shape of test dataset
print(X_test.shape, Y_test.shape)
```

```
(44535, 120) (44535,)
(14846, 120) (14846,)
```

```
# Define a custom function that plots the explained/cumulative variances for each Principal Component (PC) of a given dataset.

def plot_variance(pca, width=8, dpi=100):
    fig, axs = plt.subplots(1, 2)
    n = pca.n_components_
    grid = np.arange(1, n + 1)

    evr = pca.explained_variance_ratio_
    axs[0].bar(grid, evr)
    axs[0].set(xlabel="Component",
               title="% Explained Variance",
               ylim=(0.0, 0.2))

        cv = np.cumsum(evr)
    axs[1].plot(np.r_[0, grid], np.r_[0, cv], "o-")
    axs[1].set(xlabel="Component",
               title="% Cumulative Variance",
               ylim=(0.0, 1.0))

    fig.set(figwidth=8, dpi=100)
    return axs
```

```
    # Initialise the Principal Component Analysis (PCA) algorithm.
pca = PCA()

# Fit the PCA algorithm to the validation dataset, and generate its corresponding PCs.
X_valid_pca = pca.fit_transform(X_valid_KMeans)

# Create a list of labels for each PC, equal in length to the number of columns in the validation dataset.
X_valid_component_names = [f"PC{i+1}" for i in range(X_valid_pca.shape[1])]

# Create a dataframe that contains the PCs generated, along with their respective labels.
X_valid_pca = pd.DataFrame(X_valid_pca, columns=X_valid_component_names)

# Use the custom function to plot the explained/cumulative variances, for each PC, within the validation dataset.
plot_variance(pca)
```

Artificial Neural Network (ANN)

```
import numpy as np
import keras
from keras.models import Sequential
from keras.layers import Dense
from sklearn.metrics import classification_report

from tensorflow import keras

(x_train, y_train), (x_test, y_test) = keras.datasets.mnist.load_data()


# Preprocess the data
x_train = x_train.reshape(x_train.shape[0], 784).astype('float32')
x_test = x_test.reshape(x_test.shape[0], 784).astype('float32')
x_train /= 255
x_test /= 255

# One-hot encode the target labels
num_classes = 10
y_train = keras.utils.to_categorical(y_train, num_classes)
y_test = keras.utils.to_categorical(y_test, num_classes)

# Define the model
model = Sequential()
model.add(Dense(512, activation='sigmoid', input_shape=(784,)))
model.add(Dense(num_classes, activation='softmax'))

# Compile the model
model.compile(loss='categorical_crossentropy',
              optimizer='sgd',
              metrics=['accuracy'])

# Train the model
batch_size = 128
epochs = 20
history = model.fit(x_train, y_train,
                    batch_size=batch_size,
                    epochs=epochs,
                    verbose=1,
                    validation_data=(x_test, y_test))
```

```
def create_model(optimizer='adam'):
    # Define the model
    model = Sequential()
    model.add(Dense(512, activation='sigmoid', input_shape=(784,)))
    model.add(Dense(num_classes, activation='softmax'))

    model.compile(loss='categorical_crossentropy',
                  optimizer=optimizer,
                  metrics=['accuracy', tf.keras.metrics.Precision(), tf.keras.metrics.Recall(), tfa.metrics.F1Score(num_classes=num_class

    return model

model = KerasClassifier(build_fn=create_model, epochs=20, batch_size=128, verbose=0)

# Define the grid search parameters
optimizer = ['SGD', 'RMSprop', 'Adagrad', 'Adadelta', 'Adam', 'Adamax', 'Nadam']
param_grid = dict(optimizer=optimizer)
grid = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=-1)
grid_result = grid.fit(x_train, y_train)
```

Hybrid Package

K- Means Clustering

```
from sklearn.cluster import KMeans

# Determine the optimal number of clusters.
# Method: Cluster the dataset into k clusters, then calculate the inertia/sum of squared distances.
# Repeat this by looping through k=1 to k=30.

Sum_of_squared_distances = []
K = range(1,30)
for k in K:
    kmeans = KMeans(n_clusters=k)
    kmeans.fit(x_train)
    Sum_of_squared_distances.append(kmeans.inertia_)


# Create a plot of K-values versus their respective inertias/sums of squared distances.
plt.plot(K, Sum_of_squared_distances, 'bx-')
plt.xlabel('k')
plt.ylabel('Sum_of_squared_distances')
```

```
kmeans = KMeans(n_clusters=15)
kmeans.fit(x_train)
```

```
KMeans(n_clusters=15)
```

```
labels = kmeans.labels_
```

Feeding the clusters to ANN

```
!pip install tensorflow-addons

from sklearn.metrics import classification_report
from sklearn.metrics import f1_score
import tensorflow_addons as tfa
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.wrappers.scikit_learn import KerasClassifier
from sklearn.model_selection import GridSearchCV
import numpy as np


(x_train, y_train), (x_test, y_test) = tf.keras.datasets.mnist.load_data()

# Preprocess the data
x_train = x_train.reshape(x_train.shape[0], 784).astype('float32')
x_test = x_test.reshape(x_test.shape[0], 784).astype('float32')
x_train /= 255
x_test /= 255

# One-hot encode the target labels
num_classes = 15
y_train = tf.keras.utils.to_categorical(y_train, num_classes)
y_test = tf.keras.utils.to_categorical(y_test, num_classes)
```

Logistics Regression for ANN vs Hybrid Model

```
import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Train and evaluate the ANN model
model_ann = Sequential()
model_ann.add(Dense(10, input_dim=X_train.shape[1], activation='relu'))
model_ann.add(Dense(1, activation='linear'))
model_ann.compile(loss='mse', optimizer='adam')
model_ann.fit(X_train, y_train, epochs=50, batch_size=32, verbose=0)
y_pred_ann = model_ann.predict(X_test)
mse_ann = mean_squared_error(y_test, y_pred_ann)
r2_ann = r2_score(y_test, y_pred_ann)

# Train and evaluate the Hybrid model
kmeans = KMeans(n_clusters=5, random_state=42)
kmeans.fit(X_train)
X_train_clustered = np.concatenate([X_train, kmeans.labels_.reshape(-1, 1)], axis=1)
X_test_clustered = np.concatenate([X_test, kmeans.predict(X_test).reshape(-1, 1)], axis=1)
model_hybrid = Sequential()
model_hybrid.add(Dense(10, input_dim=X_train_clustered.shape[1], activation='relu'))
model_hybrid.add(Dense(1, activation='linear'))
model_hybrid.compile(loss='mse', optimizer='adam')
model_hybrid.fit(X_train_clustered, y_train, epochs=50, batch_size=32, verbose=0)
y_pred_hybrid = model_hybrid.predict(X_test_clustered)
mse_hybrid = mean_squared_error(y_test, y_pred_hybrid)
r2_hybrid = r2_score(y_test, y_pred_hybrid)

# Plot the predicted vs actual values for both models
import matplotlib.pyplot as plt
plt.scatter(y_test, y_pred_ann, label="ANN")
plt.scatter(y_test, y_pred_hybrid, label="Hybrid")
plt.xlabel("Actual")
plt.ylabel("Predicted")
plt.legend()
plt.show()
```

Appendix II: Gaps in K-Means Clustering and Artificial Neural Networks

Table 30: Gaps in K-Means Clustering and Artificial Neural Networks

Authors	Year	Study	Algorithms	Accuracy	Measurement metrics	Gaps
Verma V, Bhardwaj S and Singh H	2016	A Hybrid K-Mean Clustering Algorithm for Prediction Analysis	K-Means Clustering Genetic Algorithm Hybrid K-Means Clustering	79% to 87%	Accuracy Precision Recall	Limited testing of the algorithm in real-world scenarios Limited use of other evaluation metrics in addition to accuracy to assess the performance of the algorithm Limited testing of the algorithm on different datasets to assess its performance
Malav A, Kadam K and Kamat P	2017	Prediction of heart disease using K-Means and Artificial Neural Network as Hybrid approach to improve accuracy.	K- means Clustering ANN	80%	Accuracy Precision Recall F-measure ROC curve	Lack of research on the potential of using hybrid models in combination with other techniques, such as feature selection and dimensionality Limited testing of the hybrid approach on different datasets to assess its performance

Gupta D K, and Goyal S	2018	Credit Risk Prediction Using Artificial Neural Network Algorithm	ANN Linear Regression	97.69%	Accuracy	The study does state that ANNs required training on a dataset to predict the outcome of decision variables correctly. The study did not use more measurement metrics to assess the performance
Radosteva M, Soloviev V, Ivanyuk V and Tsvirkun A	2018	Use of neural network models in market risk management	Propagation neural networks RPROP learning method	N/A	The Lopez loss function The Blanco-Ihle's loss function	The study does not discuss other methods for market risk assessment The study does not compare the performance of the proposed neural network model with other existing models
Orong M, Sison A and Medina R	2019	A Hybrid Prediction Model Integrating a Modified Genetic Algorithm to K-Means Segmentation and C4.5	K-Means Clustering Genetic Algorithm C4.5 Decision Tree learning.	92%	Accuracy Precision Recall F-measure ROC curve	Difficulty in determining the optimal number of clusters in K- Means Clustering Limited ability to handle outliers and noisy data Difficulty in handling non-numeric data Limited ability to handle high-dimensional data

Nayak M and Abdullah T	2020	Short–term Predication of Risk Management Integrating Artificial Neural Network (ANN)	MLPs Radial Basis Function Networks (RBFNs) RNNs Backpropagation Algorithm Genetic Algorithms Particle Swarm Optimisation Simulated Annealing	N/A	MAE RMSE R^2 Precision Recall	Difficulty in interpreting the output of the algorithms The complexity of the algorithms and the lack of clear understanding of the underlying principles Lack of robustness of algorithms in the face of challenging data Difficulty in using real-time applications Difficulty in understanding how algorithms process large datasets
Pal R, Sekh A, Kar S and Prasad D	2020	Neural network-based country wise risk prediction of COVID-19	Backpropagation Neural Networks (BPNN) Deep Learning Decision Tree Algorithms	96.20%	MAE RMSE R^2 Precision Recall Accuracy	Lack of reliable data sources for testing and validation of the model Limited access to large-scale datasets for the study Limited access to tuning/optimisers for better performance

Calp M and Akcayol M	2020	A Novel Model for Risk Estimation in Software Projects Using Artificial Neural Network	Random Forest Extreme learning machines SVMs DNNs	90%	Accuracy Precision Recall, F1-score MCC Kappa statistic ROC curves	Lack of a unified risk estimation model The difficulty of obtaining a reliable data set for experimentation Lack of robust feature engineering techniques Moral implications of using Artificial Neural Network s in software projects
Yang, Dijie	2022	Evaluation of Enterprise Financial Risk Level under Digital Transformation with Artificial Neural Network	Deep learning algorithm, CNN, ResNet block, Depth-wise separable convolution, Backpropagation, SVM, Visual Geometry Group	N/A	Precision Recall	The study did not attempt to improve the generalisability of the results by using a larger and more diverse sample of companies. The exploration of other evaluation metrics was limited in the study. The study did not provide a comprehensive analysis by comparing the performance of different algorithms.

Appendix III: Frequency of Significant Variables based on PCA

Table 31: Frequency of Significant Variables Based on PCA

Variable	Principal component	PC value
Employment_Info_3	PC10	0.3006
Employment_Info_3	PC12	-0.3814
Employment_Info_3	PC15	0.2693
Employment_Info_3	PC23	-0.5418
Employment_Info_5	PC12	-0.2782
Employment_Info_5	PC15	0.3816
Employment_Info_5	PC23	0.5914
Family_Hist_1	PC32	-0.3568
Family_Hist_1	PC33	-0.7789
Family_Hist_1	PC34	0.3325
Insurance_History_1	PC1	-0.2971
Insurance_History_1	PC4	0.5664
Insurance_History_3	PC1	0.4934
Insurance_History_4	PC1	-0.4647
Insurance_History_7	PC1	-0.472
Insurance_History_8	PC4	0.607
Insurance_History_9	PC1	-0.251
InsuredInfo_1	PC38	-0.2707
InsuredInfo_3	PC27	-0.8097
InsuredInfo_3	PC28	0.4803
InsuredInfo_4	PC20	-0.9052
InsuredInfo_6	PC3	0.6654
InsuredInfo_6	PC7	0.2582
KMeansCluster_0	PC21	-0.3509
KMeansCluster_10	PC37	0.4763
KMeansCluster_13	PC21	0.2745
KMeansCluster_14	PC8	0.2568
KMeansCluster_2	PC6	0.3108
KMeansCluster_7	PC37	-0.4759
KMeansCluster_8	PC8	-0.2786

Medical_History_13	PC12	-0.4725
Medical_History_13	PC13	-0.3132
Medical_History_13	PC14	-0.3434
Medical_History_16	PC10	0.4617
Medical_History_16	PC11	0.3312
Medical_History_16	PC15	-0.4702
Medical_History_16	PC16	-0.2965
Medical_History_2	PC23	0.286
Medical_History_2	PC24	0.3505
Medical_History_2	PC25	0.7442
Medical_History_2	PC26	-0.2989
Medical_History_21	PC35	0.3193
Medical_History_23	PC2	-0.4599
Medical_History_23	PC5	0.3009
Medical_History_25	PC16	0.3769
Medical_History_25	PC17	0.2591
Medical_History_26	PC16	-0.3674
Medical_History_26	PC17	-0.2506
Medical_History_29	PC7	0.3805
Medical_History_29	PC11	0.7202
Medical_History_33	PC5	-0.322
Medical_History_33	PC9	-0.3444
Medical_History_34	PC14	0.7354
Medical_History_34	PC15	-0.4009
Medical_History_34	PC16	-0.3375
Medical_History_36	PC16	0.383
Medical_History_36	PC17	0.2617
Medical_History_37	PC30	-0.354
Medical_History_39	PC19	-0.6496
Medical_History_39	PC21	0.297
Medical_History_4	PC3	0.3784
Medical_History_4	PC5	-0.3875
Medical_History_4	PC7	-0.2662
Medical_History_4	PC9	0.4824
Medical_History_4	PC10	0.4868

Medical_History_41	PC5	0.3666
Medical_History_41	PC7	-0.5251
Medical_History_41	PC9	0.2696
Medical_History_41	PC10	-0.3593
Medical_History_41	PC11	0.3886
Medical_History_6	PC21	-0.5164
Medical_History_6	PC22	-0.3384
Medical_History_8	PC40	0.3596
Medical_History_9	PC40	-0.3836
Medical_Keyword_10	PC40	0.6267
Medical_Keyword_11	PC29	-0.2926
Medical_Keyword_11	PC30	0.7067
Medical_Keyword_11	PC31	0.2523
Medical_Keyword_15	PC2	0.4128
Medical_Keyword_15	PC5	-0.2835
Medical_Keyword_22	PC35	0.4307
Medical_Keyword_22	PC37	0.2802
Medical_Keyword_22	PC38	-0.2871
Medical_Keyword_22	PC40	-0.3293
Medical_Keyword_23	PC5	0.323
Medical_Keyword_23	PC9	0.3446
Medical_Keyword_25	PC29	0.7162
Medical_Keyword_25	PC34	0.3071
Medical_Keyword_3	PC19	0.4404
Medical_Keyword_37	PC34	0.7342
Medical_Keyword_40	PC12	0.2664
Medical_Keyword_40	PC36	-0.5936
Medical_Keyword_42	PC36	0.6382
Medical_Keyword_48	PC21	0.5139
Medical_Keyword_48	PC22	0.3379
Product_Info_2_A8	PC17	-0.5574
Product_Info_2_A8	PC18	-0.257
Product_Info_2_D1	PC15	0.2688
Product_Info_2_D1	PC17	0.4887
Product_Info_2_D1	PC18	-0.516

Product_Info_2_D1	PC26	-0.2714
Product_Info_2_D2	PC18	0.7587
Product_Info_2_D2	PC26	-0.318
Product_Info_2_D3	PC6	0.7411
Product_Info_2_D3	PC8	0.3023
Product_Info_2_D4	PC6	-0.4646
Product_Info_2_D4	PC8	0.5138
Product_Info_2_E1	PC35	0.4855
Product_Info_2_E1	PC38	0.4289
Product_Info_4	PC25	-0.3892
Product_Info_4	PC27	0.3012
Product_Info_4	PC28	0.5283
Product_Info_4	PC30	0.3419
Product_Info_6	PC9	-0.2784
Product_Info_6	PC12	0.4627
Product_Info_6	PC13	-0.7631