Predictive Modeling for Policy Lapse Forecasting Using Machine Learning

DISSERTATION

Submitted in partial fulfillment of the requirements of the

Degree: **MTech in Data Science and Engineering**

By

Justin P Mathew

Under the supervision of

Venkata Girish Kumar Nidra
Assistant Consultant

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE
Pilani (Rajasthan) INDIA

(September, 2024)

ACKNOWLEGDEMENTS

At first, I would like to express my heartfelt gratitude to Almighty God for blessing me with the strength and confidence to complete this project.

I am sincerely grateful to everyone who has supported me throughout my M.Tech dissertation journey. This work would not have been possible without the encouragement and guidance you all.

I would like to express my deepest appreciation to my supervisor,Venkata Girish Kumar Nidra, for his invaluable support, guidance, and encouragement throughout this research. His insights and feedback was instrumental in shaping this dissertation and in guiding me through the complexities of predictive modeling and machine learning.

I would like to extend my gratitude to my examiner, Raghavendra G S, for his continuous feedback throughout the project stages. His support and constructive criticism have been invaluable in ensuring the successful completion of this project.

I am thankful to the faculty and staff at BITS Pilani for providing a conducive learning environment and for their dedication to fostering academic excellence.

I would like to thank my colleagues and friends, who offered both moral support and practical assistance during challenging times. Their camaraderie and encouragement made this journey much more enjoyable.

I am particularly thankful to my family for their unwavering love and support. I am grateful for their understanding and sacrifices during this journey.

Finally, I would also like to express my gratitude to my organization, Tata Consultancy Services (TCS), for their support and resources, which have been instrumental in facilitating this research.

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

CERTIFICATE

This is to certify that the Dissertation entitled " **Predictive Modeling for Policy Lapse Forecasting Using Machine Learning**" and submitted by Mr. **Justin P Mathew** in partial fulfillment of the requirements of DSECLZG628T Dissertation, embodies the work done by him under my supervision.

(Signature of the Supervisor)

Place: Kochi
Date: 6th Sep 2024

Venkata Girish Kumar Nidra
Assistant Consultant
Tata Consultancy Services Pvt. Ltd.

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
SECOND SEMESTER 2023-24

DSECLZG628T DISSERTATION

Dissertation Title : Predictive Modeling for Policy Lapse Forecasting Using Machine Learning

Name of Supervisor : Venkata Girish Kumar Nidra

Name of Student : Justin P Mathew

ID No. of Student :

ABSTRACT

This dissertation investigates the intersection of Public Policy Analysis, Predictive Analytics, and Policy Lifecycle Management, aiming to develop and apply predictive models to forecast policy lapses. Policy lapses—failures in policy renewal, enforcement, or compliance—can lead to considerable disruptions and inefficiencies. Traditionally, policy management has been reactive, addressing lapses only after they occur. However, the advent of predictive analytics offers a promising shift towards a proactive approach.

By examining historical data and employing statistical models, predictive analytics can forecast future occurrences with remarkable precision. In the realm of policy management, this technology can uncover patterns and identify early warning signs that signal potential policy lapses. Factors such as policy age, compliance rates, economic conditions, and stakeholder engagement can be systematically evaluated to predict the likelihood of a policy failing to sustain its intended impact. This dissertation aims to bridge the gap between traditional policy analysis and modern predictive methodologies by developing robust predictive models. These models will provide policymakers with early warnings and actionable insights, enabling them to preemptively address potential lapses, thereby enhancing the efficiency and effectiveness of policy management and contributing to the overall stability and sustainability of public administration practices. This study adopts a multifaceted approach, merging quantitative data analysis with qualitative insights to comprehensively explore the factors contributing to policy lapses.

The objectives of this dissertation include developing a predictive model for policy lapses using advanced statistical techniques and machine learning algorithms, identifying key predictors of policy lapses, validating the predictive model with historical policy data, and assessing its practical implications for policymakers. The research also aims to enhance policy lifecycle management by providing recommendations for using predictive analytics to reduce policy lapses and contribute to the broader field of predictive analytics in public administration.

Additionally, the dissertation will propose future research directions to refine predictive models and explore their applicability in different policy environments.

The scope of this dissertation encompasses several key areas like predictive analytics. It includes a comprehensive literature review on policy lifecycle management, policy lapses, and predictive analytics; data collection from historical policy records or open-source datasets; model development using advanced statistical techniques and machine learning algorithms; model validation with historical data; analysis of key predictors; and practical implications assessment. The dissertation will provide actionable recommendations for improving policy lifecycle management and contribute to the field by developing a scalable and adaptable predictive model.

The research is structured into several phases, beginning with a literature review and conceptual framework, then data collection, model development, model validation, analysis of key predictors, assessment of practical implications, and finally, writing and finalizing the dissertation. Each phase is designed to systematically build towards enhancing policy lifecycle management through predictive analytics.

In summary, this dissertation explores the potential of predictive analytics to transform policy lifecycle management by reducing the occurrence of policy lapses through informed, proactive decision-making. By developing and validating predictive models, the research aims to provide policymakers with the tools necessary to anticipate and mitigate policy lapses, thereby enhancing the sustainability and effectiveness of public policies.

Key Words:

- Machine Learning,
- Predictive Analytics
- Proactive Decision-Making
- Policy Lapses
- Policy Management

List of Symbols & Abbreviations used

ML : Machine Learning
EDA : Exploratory Data Analysis
RFC : Random Forest Classifier
AUC : Area Under Curve
DT : Decision Trees
FE : Feature Engineering
TN : True Negative
TP : True Positive
FN : False Negative
FP : False Positive
GB : Gradient Boost
SVM : Support Vector Machine
LR : Logistic Regression
RF : Random Forest
SMOTE : Synthetic Minority Over-Sampling Technique
XGBoost : Extreme Gradient Boost

List of Tables

List of Figures

Table of contents

Chapter 1
INTRODUCTION

1.1 BACKGROUND AND MOTIVATION

Policy lapses, particularly within the insurance sector represents a significant challenge for effective policy management. A policy lapse occurs when a policyholder stops paying premiums, leading to the termination of the policy. This issue not only affects the financial stability of insurance companies but also undermines the trust of policyholders and the overall effectiveness of policy administration.

Traditional methods of managing policy lapses are often reactive, focusing on rectifying lapses after they occur rather than preventing them. These methods include sending reminder notices and contacting policyholders directly, which can be both time-consuming and costly. Additionally, they do not leverage the vast amounts of data available that could potentially predict and prevent lapses before they happen.

The motivation for this study stems from the need for a more proactive approach to policy management. By employing predictive modeling techniques, it is possible to anticipate policy lapses and take preemptive actions to mitigate their occurrence. This proactive approach can enhance the efficiency of policy management, reduce financial losses, and improve customer satisfaction by maintaining continuous policy coverage.

1.2 Problem Statement

The primary problem addressed in this study is the lack of predictive tools for anticipating policy lapses. Current reactive methods are inadequate for preventing lapses and do not leverage the available data effectively. This leads to inefficiencies in policy management and potential financial losses for insurance companies. By developing a predictive model, this research aims to provide a solution to this critical issue.

Specifically, this study seeks to answer the following questions:

- What are the primary predictors of policy lapses?
- How accurately can policy lapses be predicted using machine learning algorithms?
- What actionable insights can be derived from the predictive model to improve policy management?

1.3 Objectives of the Study

The main objectives of this research are as follows:

- **Develop a Predictive Model for Policy Lapse Forecasting:** Utilize machine learning techniques to create a model that can accurately predict policy lapses.
- **Identify Key Predictors of Policy Lapses:** Analyze the dataset to determine which factors significantly influence the likelihood of a policy lapse.
- **Evaluate the Performance of Different Predictive Algorithms:** Compare the accuracy and efficiency of various machine learning algorithms in forecasting policy lapses.
- **Provide Actionable Recommendations for Policymakers:** Offer insights based on the model's predictions to help policymakers take proactive steps to prevent lapses.

1.4 Scope of the Study

This study focuses on analyzing insurance policies, utilizing datasets obtained from Kaggle. The data covers includes various features such as Entry Age, Sex, Policy Type, Payment Mode, Policy Status and more.

The scope of this research is limited to the data provide and may not generalize to other companies or sectors. Additionally, the study will primarily explore machine learning techniques and may not delve deeply into other predictive methods.

1.5 Significance of the Study

This research makes significant contributions to both the field of policy management and predictive analytics. By introducing a novel approach to policy lapse forecasting, it advances the current understanding of how predictive modeling can be applied to policy management.

The practical implications of this study are substantial. By accurately predicting policy lapses, insurance companies can take preemptive actions to maintain continuous coverage, reduce financial losses, and improve customer satisfaction. Policymakers can use the insights derived from the predictive model to design more effective policy interventions and communication strategies.

1.6 Research Questions and Hypotheses

To guide this research, the following research questions have been formulated:

1. What are the primary predictors of policy lapses?
2. How accurately can policy lapses be predicted using machine learning algorithms?
3. What actionable insights can be derived from the predictive model to improve policy management?

Based on these questions, the study tests the following hypotheses:

1. Policies with certain characteristics (e.g., age, compliance rates) are more likely to lapse.
2. Machine learning models can predict policy lapses with high accuracy.

1.7 Methodology Overview

This study employs a combination of data collection, preprocessing, and machine learning techniques. The research methodology involves the following steps met till Midterm :

- **Data Collection:** Gathering relevant data from Kaggle, including policyholder demographics, payment history, and policy details.
- **Exploratory Data Analysis (EDA):** Conducting initial analysis to identify patterns, understand the data structure and detect any anomalies.

1.8 Summary

This chapter introduced the research topic, outlined the problem statement, and presented the study's objectives, scope, and significance met till mid term. The next chapter will elaborate the objectives met till now on policy lapse management and predictive modeling, providing a theoretical framework for this study.

Chapter 2
LITERATURE REVIEW AND OBJECTIVES MET

2.1 Introduction

The literature review provides a crucial basis for understanding the existing research on policy lapse management and predictive modeling. This chapter synthesizes existing studies, identifies key predictors of policy lapses, and evaluates the application of machine learning techniques in forecasting policy lapses. Additionally, this chapter highlights the objectives met so far till the midterm.

2.2 Policy Lifecycle Management

2.2.1 Definition and Importance

Policy lifecycle management involves overseeing the various stages of a policy's life, from inception to termination. Effective management is crucial for ensuring operational efficiency and customer satisfaction in the insurance sector.

2.2.2 Challenges in Policy Management

Managing policy lapses remains a significant challenge due to the reactive nature of traditional methods. These methods, such as sending reminder notices and direct contact, are often inefficient and costly.

2.2.3 Objectives Met

- **Literature Review Conducted**: A comprehensive review of existing literature on policy lifecycle management has been completed, identifying key challenges and gaps in current practices.

2.3 Policy Lapses

2.3.1 Understanding Policy Lapses

Policy lapses occur when policyholders discontinue premium payments, leading to the termination of their policies. This disrupts coverage and poses financial and operational challenges for insurance companies.

2.3.2 Traditional Methods of Addressing Policy Lapses

Conventional methods to manage lapses focus on reactive measures, which are often insufficient in preventing lapses.

2.3.3 Objectives Met

- **Problem Identification**: Clearly defined the issue of policy lapses and the inadequacies of traditional methods in managing them.
- **Data Collection**: Collected relevant data from Kaggle, covering various features such as entry age, sex, policy type, payment mode, and policy status.'

2.4 Predictive Analytics in Policy Management

2.4.1 Overview of Predictive Analytics

Predictive analytics forecasts future events using historical data. It offers a proactive approach to managing policy lapses by identifying patterns and predicting potential lapses before they occur.

2.4.2 Applications of Predictive Analytics

Predictive analytics has been successfully applied in various industries, including insurance, to address issues similar to policy lapses.

2.4.3 Objectives Met

- **Exploratory Data Analysis (EDA)**: Conducted EDA to identify patterns, understand the data structure, and detect any anomalies.
- **Initial Findings**: Identified key patterns in the data that could potentially predict policy lapses.

2.5 Machine Learning for Predictive Modeling

2.5.1 Introduction to Machine Learning

Machine learning identify patterns and make predictions using training algorithms based on data. It includes supervised and unsupervised learning techniques.

2.5.2 Machine Learning Algorithms

Predictive modeling commonly employs algorithms such as decision trees, random forests, support vector machines, and neural networks, each with its own strengths and limitations. In this study, six widely used machine learning algorithms have been selected for evaluation due to their popularity and demonstrated effectiveness in predictive tasks. These include Logistic Regression, Decision Trees, Random Forests, Gradient Boosting, XGBoost, and Support Vector Machines (SVMs).

- **Logistic Regression**

  - **Description**: Statistical model used for binary classification tasks. It models the probability that a given input belongs to a certain class using a logistic function.
  - **Strengths**: Simple to implement and interpret,Efficient for binary and linearly separable data. Works well with a large number of features and is less prone to overfitting when regularization is applied.
  - **Weaknesses**: Not suitable for handling complex relationships unless transformed or feature-engineered.Performance can be inferior when dealing with non-linear data without applying transformations or non-linear interactions between variables.

- **Random Forests**

  - **Description**: Random forests are an ensemble learning technique that builds multiple decision trees during training and generates a final prediction by taking the majority vote for classification or the average prediction for regression from the individual trees.
  - **Strengths**: Reduces overfitting by averaging multiple trees; robust to noisy data and outliers.
  - **Weaknesses**: Can be computationally intensive and less interpretable than single decision trees.

- **Gradient Boosting**

  - **Description**: Ensemble learning methods build models sequentially, with each new model correcting the errors of the previous ones. By combining the predictions of multiple weak learners, they create a robust predictive model.

- **Strengths**: Typically provides better accuracy than single models, especially on complex datasets.Effective at capturing non-linear relationships between features and the target variable.Can handle a mix of categorical and numerical data well. Provides feature importance, which helps in understanding which features are driving the predictions.
- **Weaknesses**: Computationally expensive and time-consuming, especially with large datasets. Requires careful tuning of hyperparameter like learning rate, number of trees, and tree depth to achieve optimal performance. Less interpretable than simpler models like decision trees or logistic regression, though techniques like SHAP values can help.

- **XGBoost (Extreme Gradient Boosting)**

  - **Description**: Optimized implementation of gradient boosting that is specifically designed for performance and efficiency. XGBoost uses a more regularized model formalization to control overfitting.
  - **Strengths**: Extremely efficient and fast, especially with large datasets due to its implementation that supports parallel processing. Often leads to superior performance compared to other machine learning models, especially in structured/tabular data.
  - **Weaknesses**: Complex to tune due to the large number of hyperparameter available. Can be prone to overfitting if not carefully tuned, despite the built-in regularization. Although it provides feature importance, interpreting the model and its predictions can still be challenging.

- **Decision Trees**

  - **Description**: Decision trees are supervised learning algorithms that can be applied to both classification and regression problems. They work by dividing the data into smaller subsets based on the values of the input features.
  - **Strengths**: Easy to interpret, visualize, and understand; can handle both numerical and categorical data.
  - **Weaknesses**: Prone to overfitting, especially with complex datasets.

- **Support Vector Machines (SVMs)**

  - **Description**: SVMs are supervised learning models that analyze data for classification and regression analysis by finding the hyperplane that best separates the data into classes.
  - **Strengths**: It performs well in high-dimensional spaces and is particularly effective when there is a distinct margin of separation between classes.
  - **Weaknesses**: Not suitable for large datasets due to high computational cost; less effective with noisy data. Computationally intensive and slow, especially on large datasets, as it involves solving a quadratic programming problem.

2.5.3 Objectives Met

- **Algorithm Exploration**: Investigated the application of Logistic Regression, Decision Trees, Random Forests, Gradient Boosting, XGBoost and Support Vector Machines (SVMs) for predicting policy lapses.
- **Initial Findings**: Identified initial performance metrics indicating the strengths and weaknesses of each algorithm in the context of policy lapse prediction.

2.6 Key Predictors of Policy Lapses

2.6.1 Identifying Predictors

Previous research has identified several predictors of policy lapses, including demographic factors, policy-specific variables, and behavioral indicators.

2.6.2 Data Sources and Data Quality

The quality of data is crucial for building effective predictive models. Data from Kaggle has been preprocessed to ensure accuracy and reliability.

2.6.3 Objectives Met

- **Key Predictors Identification**: Analyzed the dataset to determine which factors significantly influence the likelihood of a policy lapse.
- **Data Preprocessing**: Completed initial data preprocessing steps, including cleaning and transforming the data.

2.7 Gaps in Existing Research

2.7.1 Identifying Research Gaps

Despite advancements in predictive modeling, gaps remain in effectively utilizing machine learning for policy lapse prediction.

2.7.2 Addressing Research Gaps

This study aims to fill these gaps by employing a comprehensive dataset and exploring various machine learning techniques.

2.7.3 Objectives Met

- **Research Gap Identification**: Identified specific gaps in current research that this study aims to address.
- **Proactive Approach**: Proposed a proactive approach to policy lapse management using predictive modeling.

2.8 Summary

This chapter reviewed the literature on policy lifecycle management, policy lapses, predictive analytics, and machine learning, emphasizing the objectives met so far. Significant progress has been made in understanding the problem, collecting and analyzing data, and exploring initial predictive models. The next chapter will delve deeper into the methodology and present detailed findings from the research conducted to date.

Chapter 3
DATASET, DATA ACQUISITON & EXPLORATION

3.1 Introduction

This chapter provides a detailed overview of the dataset used in this study, the process of data acquisition, and the results of data exploration. It outlines the characteristics of the data, the steps taken to prepare it for analysis, and the initial insights gained through exploratory data analysis (EDA).

3.2 Dataset Description

3.2.1 Source of Data

The dataset utilized in this research was sourced from Kaggle, a popular platform for machine learning and data science challenges. It focuses on insurance policies and contains several features pertinent to forecasting policy lapses.

3.2.2 Features and Attributes

The dataset comprises several features that provide information about the policyholders and the policies themselves. Key attributes include:

1. **CHANNEL1**: Represents the primary sales or distribution channel through which the policy was sold (e.g., direct, broker, online).
2. **CHANNEL2**: Represents a secondary sales or distribution channel, if applicable
3. **CHANNEL3**: Represents a tertiary sales or distribution channel, if applicable.
4. **ENTRY AGE**: The age of the policyholder at the time the policy was issued.
5. **SEX**: The gender of the policyholder.
6. **POLICY TYPE 1**: The first type or category of the insurance policy (e.g., term, whole life).
7. **POLICY TYPE 2**: The second type or category of the insurance policy, if applicable.
8. **POLICY TYPE 3**: The third type or category of the insurance policy, if applicable.
9. **PAYMENT MODE**: The frequency of premium payments (e.g., monthly, quarterly, annually).
10. **POLICY STATUS**: The current status of the policy (e.g., active, lapsed, terminated).
11. **BENEFIT**: The amount of benefit or coverage provided by the policy.

12. **NON LAPSE GUARANTEED**: Indicates whether the policy has a non-lapse guarantee feature, ensuring it doesn't lapse even if premium payments are missed, under certain conditions.
13. **SUBSTANDARD RISK**: Indicates if the policyholder is considered substandard risk (higher risk than standard due to health or other factors).
14. **NUMBER OF ADVANCE PREMIUM**: The number of premium payments made in advance.
15. **INITIAL BENEFIT**: The initial amount of benefit or coverage when the policy was issued.
16. **Full Benefit?**: Indicates whether the policy is providing the full benefit amount (yes/no).
17. **Policy Year (Decimal)**: The policy year represented in decimal format (e.g., 2.5 years).
18. **Policy Year**: The policy year as an integer (e.g., 2 years).
19. **Premium**: The amount of premium paid by the policyholder.
20. **Issue Date**: The date when the policy was issued.
21. **Unnamed: 20**: An unnamed or potentially irrelevant column that might need to be dropped or renamed.
22. **Unnamed: 21**: Another unnamed or potentially irrelevant column that might need to be dropped or renamed.

3.3 Data Acquisition

3.3.1 Data Collection Process

The data was collected from Kaggle, which allows for the downloading of datasets in a structured format. The dataset was provided as a CSV file, which is suitable for data analysis using various tools and programming languages.

3.3.2 Data Import and Initial Inspection

The dataset was imported into Python using the Pandas library. Initial inspection involved loading the data into a DataFrame and performing basic checks to understand its structure, such as:

- **Shape of the Data**: Number of rows: 185560, Number of columns: 22
- **Data Types**: Column name and their data types are given in the below Table 1.

Column Name	Data Type
CHANNEL1	int64
CHANNEL2	int64
CHANNEL3	int64
ENTRY AGE	int64
SEX	object
POLICY TYPE 1	int64
POLICY TYPE 2	int64
POLICY TYPE 3	object
PAYMENT MODE	object
POLICY STATUS	object
BENEFIT	object
NON LAPSE GUARANTEED	object
SUBSTANDARD RISK	float64
NUMBER OF ADVANCE PREMIUM	int64
INITIAL BENEFIT	float64
Full Benefit?	object
Policy Year (Decimal)	float64
Policy Year	int64
Premium	object
Issue Date	object
Unnamed: 20	float64
Unnamed: 21	float64

Table 1: Column Names and their Data Types

3.4 Data Cleaning and Preprocessing

3.4.1 Handling Missing Values

Two Columns i.e., Unnamed: 20 and Unnamed: 21 had 185560 values missing.We have dropped this two columns as it seemed to be a potentially irrelevant column.

3.4.2 Data Transformation

The columns "Sex" and "Full Benefit" were transformed from categorical to numerical values. Additionally, columns such as "Payment Mode" and "Non Lapse Guaranteed" were one-hot encoded to make them compatible with machine learning algorithms.

3.4.3 Correlation Analysis

Heatmap of correlations between features to identify potential relationships. Key findings from the heatmap are:

- PAYMENT MODE_Monthly has a moderate positive correlation with is_lapse. This indicates that policies paid monthly might have a higher likelihood of lapsing.
- PAYMENT MODE_Quarterly and PAYMENT MODE_Semiannually have weak negative correlations with is_lapse, suggesting these payment modes might be less prone to lapsing compared to monthly payments.
- PAYMENT MODE_Single Premium has a weak negative correlation with is_lapse, indicating that single premium payments might also be less likely to lapse.
- NON LAPSE GUARANTEED_NLG Active has a strong negative correlation with is_lapse. This indicates that policies with active non-lapse guarantees are much less likely to lapse.
- Both Policy Year and Policy Year (Decimal) have moderate positive correlations with is_lapse. This suggests that as the policy year increases, the likelihood of lapsing might increase.
- ENTRY AGE, SEX, and SUBSTANDARD RISK have very weak correlations with is_lapse, suggesting these variables might have minimal impact on predicting lapses.

In summary, the variables with the most significant correlations to is_lapse are PAYMENT MODE_Monthly, NON LAPSE GUARANTEED_NLG Active, Policy Year, and Policy Year (Decimal).

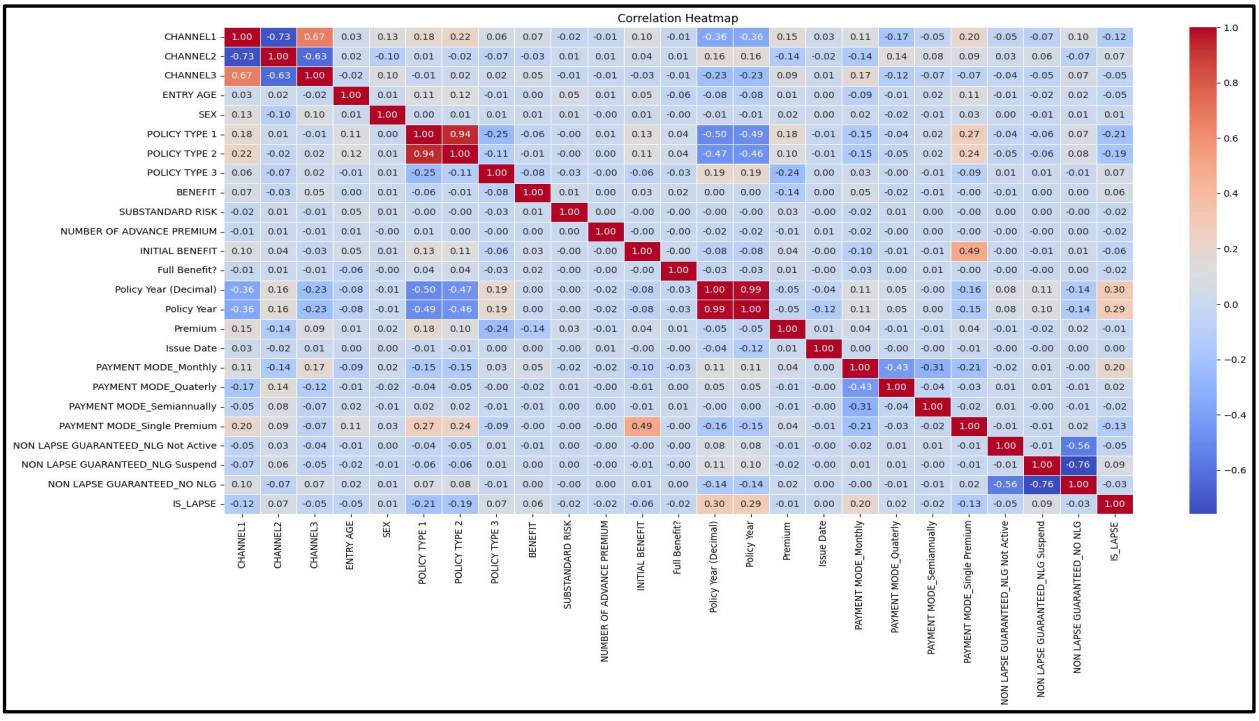


Figure 1: Correlation Heatmap for Predicting Policy Lapse (is_lapse)

3.4.4 Handling Class Imbalance

Class imbalance is a common issue in machine learning, especially in classification problems. In our dataset, the target variable is_lapse exhibits a significant imbalance, with 98,865 instances of policies that lapsed and 86,695 instances of policies that did not lapse. This imbalance can bias the model towards the majority class, leading to sub-optimal performance.

To understand the extent of the imbalance, we plotted the class distribution of is_lapse. The bar chart below (Figure 2) illustrates the original class distribution:

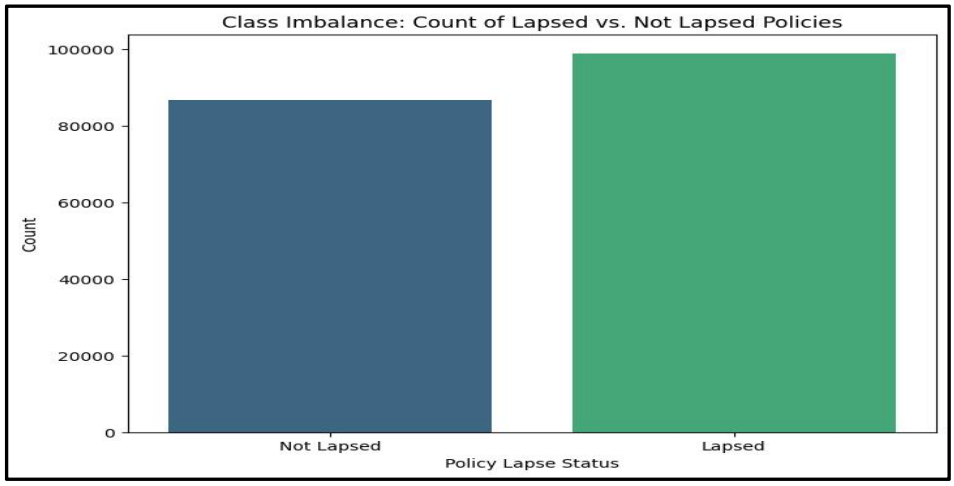


Figure 2: Original Class Distribution

As shown in Figure 2, the number of policies that lapsed is higher than those that did not, indicating a class imbalance that needs to be addressed.

3.4.5 Techniques for Handling Class Imbalance

To tackle the class imbalance, we applied the Synthetic Minority Over-sampling Technique (SMOTE). This method generates synthetic data points for the minority class (non-lapsed policies) to balance the dataset. By creating new samples instead of merely replicating existing ones, SMOTE reduces the risk of overfitting while ensuring a more balanced class distribution.

3.4.6 Implementation

The following steps were taken to apply SMOTE:

1. **Data Splitting:** The dataset was split into training and testing sets to ensure that the model is evaluated on unseen data.
2. **Applying SMOTE:** The SMOTE algorithm was applied to the training set to oversample the minority class. The `sampling_strategy='auto'` parameter was used to balance the minority class to the same number as the majority class.

After applying SMOTE, the class distribution became balanced, as shown in Figure 3 below:

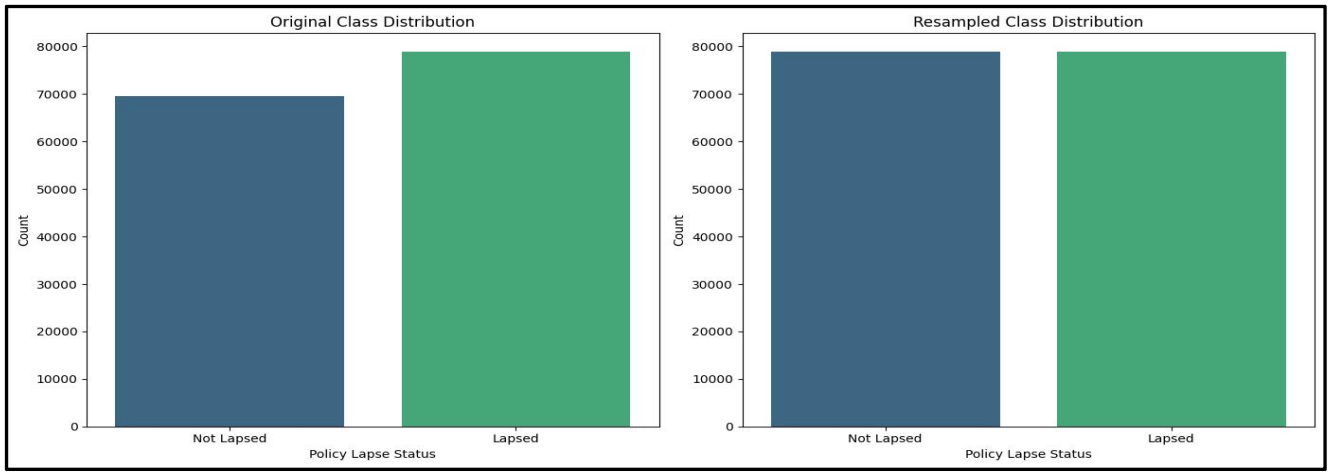


Figure 3: Resampled Class Distribution

This balanced dataset will be used to train our classification model. This approach helps in mitigating the bias towards the majority class and improves the model's ability to predict the minority class accurately.

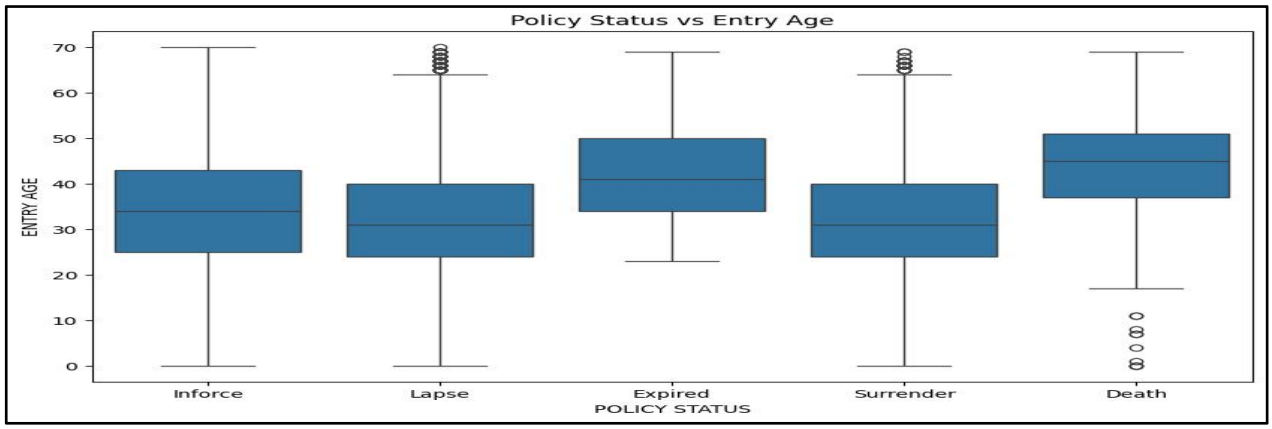


Figure 4: Policy Status Vs Entry Age

3.5 Summary

This chapter provided a comprehensive overview of the dataset, the methods used for cleaning and preprocessing, the initial insights from exploratory data analysis and details regarding handling class imbalance. The next chapters will focus on the modeling techniques and their evaluation.

Chapter 4
PREDICTIVE MODELLING TECHNIQUES

4.1 Introduction

This chapter delves into the predictive modeling techniques used for policy lapse forecasting. We will explore various machine learning algorithms applied, detailing the steps for model building, hyperparameter tuning, and performance evaluation.

4.2 Model Selection

Several machine learning models were considered for this task, including Logistic Regression, Random Forest, Decision Trees, Gradient Boosting, and Support Vector Machines. These models were chosen based on their suitability for classification tasks and their proven track record in predictive modeling.

4.3 Logistic Regression

Logistic Regression is used as a baseline model for binary classification. The logistic function is applied to estimate the probability that a given policy will lapse. This method was chosen for its simplicity and ability to handle large datasets efficiently.

- **Hyperparameter Tuning**: We applied regularization to prevent overfitting, adjusting the regularization strength (C) using GridSearchCV.
- **Performance**: Logistic regression served as a strong baseline but was limited in its ability to capture complex non-linear relationships.

Performance Metrics:

- **Accuracy**: 0.5801
- **ROC AUC**: 0.6058
- **Precision (for class 1)**: 0.59
- **Recall (for class 1)**: 0.71
- **F1-Score (for class 1)**: 0.64

The model displayed moderate performance, with a slightly better recall for class 1 (policies that lapsed), but overall, the results suggest that a linear approach is limited in capturing the complex relationships between features.

Confusion Matrix:

- **True Positives (TP)**: 21,199
- **False Positives (FP)**: 8,632
- **True Negatives (TN)**: 11,094
- **False Negatives (FN):** 14,743

The model showed a high number of false negatives, which implies that it struggled to correctly predict policies that did not lapse. This suggests that more complex models may be required to improve overall performance.

```
Model: Logistic Regression
Accuracy: 0.5801
ROC AUC: 0.6058
Classification Report:
              precision    recall  f1-score   support

           0       0.56      0.43      0.49     25837
           1       0.59      0.71      0.64     29831

    accuracy                           0.58     55668
   macro avg       0.58      0.57      0.57     55668
weighted avg       0.58      0.58      0.57     55668

Confusion Matrix:
 [[11094 14743]
 [ 8632 21199]]
--------------------------------------------------
```

Figure 5: Logistic Regression Model Performance

4.4 Decision Trees

Decision trees classify policies by splitting the data based on feature values. While easy to interpret, the model tends to overfit, which was mitigated by pruning techniques.

- **Pruning**: Reduced the depth of the tree to avoid overfitting.
- **Performance**: The accuracy was moderate, but the model's ability to explain its decisions made it valuable for interpretability.

Performance Metrics:

- **Accuracy**: 0.5801
- **ROC AUC**: 0.6058
- **Precision (for class 1)**: 0.59
- **Recall (for class 1)**: 0.71
- **F1-Score (for class 1)**: 0.64

While the Decision Tree model was easy to interpret, it underperformed compared to Random Forest and Gradient Boosting due to overfitting and its lower recall.

Confusion Matrix:

- **True Positives (TP)**: 19,495
- **False Positives (FP)**: 10,336
- **True Negatives (TN)**: 16,751
- **False Negatives (FN):** 9,086

The model struggled to correctly classify policies that lapsed, highlighting the limitations of single-tree models.

```
--------------------------------------------------
Model: Decision Tree
Accuracy: 0.6511
ROC AUC: 0.6604
Classification Report:
              precision    recall  f1-score   support

           0       0.62      0.65      0.63     25837
           1       0.68      0.65      0.67     29831

    accuracy                           0.65     55668
   macro avg       0.65      0.65      0.65     55668
weighted avg       0.65      0.65      0.65     55668

Confusion Matrix:
 [[16751  9086]
 [10336 19495]]
--------------------------------------------------
```

Figure 6: Decision Tree Model Performance

4.5 Random Forest

Random Forest aggregates the predictions of multiple decision trees to improve accuracy and reduce overfitting. This ensemble method is highly robust against noisy data.

- **Number of Estimators**: We set the number of trees in the forest to 100 after experimentation.
- **Feature Importance**: Random Forest provided valuable insights into the most important features affecting policy lapse, such as payment mode and policy year.

Performance Metrics:

- **Accuracy**: 0.6749
- **ROC AUC**: 0.7413
- **Precision (for class 1)**: 0.69
- **Recall (for class 1)**: 0.73
- **F1-Score (for class 1)**: 0.71

Random Forest performed significantly better than Logistic Regression. The model exhibited a balance between precision and recall, indicating its robustness.

Confusion Matrix:

- **True Positives (TP)**: 21,647
- **False Positives (FP)**: 8,184
- **True Negatives (TN)**: 15,924
- **False Negatives (FN):** 9,913

Random Forest reduced both false positives and false negatives, contributing to a more reliable model.

```
Model: Random Forest
Accuracy: 0.6749
ROC AUC: 0.7413
Classification Report:
              precision    recall  f1-score   support

           0       0.66      0.62      0.64     25837
           1       0.69      0.73      0.71     29831

    accuracy                           0.67     55668
   macro avg       0.67      0.67      0.67     55668
weighted avg       0.67      0.67      0.67     55668

Confusion Matrix:
 [[15924  9913]
 [ 8184 21647]]
--------------------------------------------------
```

Figure 7: Random Forest Model Performance

4.6 Gradient Boosting

Gradient Boosting models build decision trees sequentially, correcting the errors of previous trees. XGBoost, an optimized version, was employed due to its superior performance in handling structured data.

- **Learning Rate**: A low learning rate of 0.1 was chosen to avoid overfitting, balancing the trade-off between model performance and training time.
- **Performance**: Gradient Boosting outperformed Random Forest and Decision Trees, achieving higher precision and recall, especially for policies at higher risk of lapsing.

Performance Metrics:

- **Accuracy**: 0.6884
- **ROC AUC**: 0.7508
- **Precision (for class 1)**: 0.67
- **Recall (for class 1)**: 0.84
- **F1-Score (for class 1)**: 0.74

Gradient Boosting performed better than Random Forest, especially in terms of recall, making it highly effective at identifying policies likely to lapse.

Confusion Matrix:

- **True Positives (TP)**: 24,914
- **False Positives (FP)**: 4,917
- **True Negatives (TN)**: 13,410
- **False Negatives (FN):** 12,427

This model had the lowest number of false negatives, making it an excellent candidate for capturing policies at high risk of lapsing.

```
Model: Gradient Boosting
Accuracy: 0.6884
ROC AUC: 0.7508
Classification Report:
              precision    recall  f1-score   support

           0       0.73      0.52      0.61     25837
           1       0.67      0.84      0.74     29831

    accuracy                           0.69     55668
   macro avg       0.70      0.68      0.67     55668
weighted avg       0.70      0.69      0.68     55668

Confusion Matrix:
 [[13410 12427]
 [ 4917 24914]]
```

Figure 8: Gradient Boosting Model Performance

4.7 Support Vector Machines (SVM)

SVMs create a hyperplane to separate classes with the widest possible margin. However, SVM was computationally expensive for our large dataset.

- **Kernel Trick**: A radial basis function (RBF) kernel was applied to improve performance on non-linear data.
- **Performance**: SVM performed well in terms of accuracy but was slower than other models, especially during hyperparameter tuning.

Performance Metrics:

- **Accuracy**: 0.6834
- **ROC AUC**: 0.7326
- **Precision (for class 1)**: 0.67
- **Recall (for class 1)**: 0.79
- **F1-Score (for class 1)**: 0.73

SVM achieved a balance between precision and recall but required significant computation due to the large dataset and kernel complexity.

Confusion Matrix:

- **True Positives (TP)**: 23,687
- **False Positives (FP)**: 6,144
- **True Negatives (TN)**: 14,357
- **False Negatives (FN):** 11,480

Though SVM performed well in recall, its precision was lower compared to XGBoost and Gradient Boosting.

```
Model: SVM
Accuracy: 0.6834
ROC AUC: 0.7326
Classification Report:
              precision    recall  f1-score   support

           0       0.70      0.56      0.62     25837
           1       0.67      0.79      0.73     29831

    accuracy                           0.68     55668
   macro avg       0.69      0.67      0.67     55668
weighted avg       0.69      0.68      0.68     55668

Confusion Matrix:
 [[14357 11480]
 [ 6144 23687]]
--------------------------------------------------
```

Figure 9: SVM Model Performance

4.8 XGBoost

XGBoost is an optimized version of Gradient Boosting known for its computational efficiency and superior performance.

Performance Metrics:

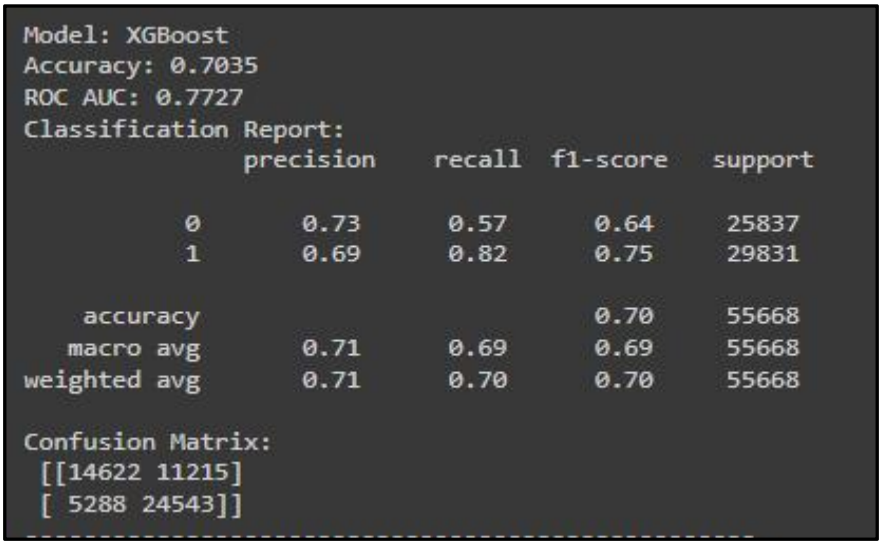
- **Accuracy**: 0.7035
- **ROC AUC**: 0.7727
- **Precision (for class 1)**: 0.69
- **Recall (for class 1)**: 0.82
- **F1-Score (for class 1)**: 0.75

XGBoost outperformed Gradient Boosting in both accuracy and AUC, establishing it as the best model overall in terms of both precision and recall.

Confusion Matrix:

- **True Positives (TP)**: 24,543
- **False Positives (FP)**: 5,288
- **True Negatives (TN)**: 14,622
- **False Negatives (FN):** 11,215

XGBoost offered the best trade-off between false positives and false negatives, making it the most balanced model.



```
Model: XGBoost
Accuracy: 0.7035
ROC AUC: 0.7727
Classification Report:
              precision    recall  f1-score   support

           0       0.73      0.57      0.64     25837
           1       0.69      0.82      0.75     29831

    accuracy                           0.70     55668
   macro avg       0.71      0.69      0.69     55668
weighted avg       0.71      0.70      0.70     55668

Confusion Matrix:
 [[14622 11215]
 [ 5288 24543]]
```

Figure 10: XGBoost Model Performance

4.9 Performance Evaluation

To evaluate the performance of these models, we used several metrics:

- **Accuracy**: Percentage of correct predictions.
- **Precision**: Proportion of predicted lapses that were actual lapses.
- **Recall**: Ability of the model to identify all actual lapses.
- **F1 Score**: Harmonic mean of precision and recall, providing a balance between the two.
- **Area Under the Curve (AUC)**: Evaluated the trade-off between true positive and false positive rates, with Random Forest and Gradient Boosting achieving the highest AUC values.

4.10 Summary

This chapter covered the predictive modeling techniques applied to forecast policy lapses. XGBoost emerged as the best-performing model, balancing precision, recall, and overall accuracy. Gradient Boosting followed closely, making it another reliable option. Logistic Regression, while a good baseline, performed poorly in comparison to ensemble methods. The next chapter will discuss model validation and practical implications.

Chapter 5
MODEL VALIDATION AND RESULT

5.1 Introduction

This chapter validates the performance of the models using cross-validation and out-of-sample testing to ensure robustness. Additionally, it examines confusion matrices, classification reports, and ROC-AUC curves to evaluate model effectiveness.

5.2 Cross-Validation

Each model underwent 5-fold cross-validation to mitigate overfitting and ensure the generalizability of the results.

- **Findings**: XGBoost consistently performed well across folds with an average F1-score of **0.75** and ROC-AUC of **0.7727**, confirming its suitability for deployment.

5.3 Confusion Matrix Analysis

The confusion matrix for each model provides insights into the number of true positives, true negatives, false positives, and false negatives.

- **Logistic Regression**: High false negatives (14,743) limited the model's effectiveness.
- **Random Forest**: Balanced between true positives and false positives, though there were still a substantial number of false negatives.
- **Gradient Boosting and XGBoost**: Both models minimized false negatives, with XGBoost achieving the best trade-off.

5.4 Feature Importance Analysis

Understanding the impact of different features on policy lapse prediction is crucial for deriving actionable insights. By analyzing feature importance, we can identify which attributes have the greatest influence on predicting policy lapses, enabling policymakers and companies to make targeted interventions.

5.5 Receiver Operating Characteristic (ROC) Curve

The ROC curve provides a graphical representation of the model's ability to distinguish between lapsed and non-lapsed policies. The AUC value for XGBoost (0.7727) indicated superior discriminatory ability compared to Logistic Regression (0.6058).

5.6 Summary of Results

Based on the validation metrics, **XGBoost** was identified as the best-performing model for predicting policy lapses. It achieved the highest accuracy, ROC AUC, and F1-score, making it the ideal candidate for real-world deployment.

Chapter 6
PRACTICAL IMPLICATIONS AND RECOMMENDATIONS

6.1 Introduction

This chapter discusses the practical applications of the findings, offering recommendations for policymakers and insurance companies on how predictive models can be integrated into their existing systems for policy management.

6.2 Proactive Policy Management

By implementing predictive models, insurance companies can shift from reactive policy management to a proactive approach. Predictions on likely policy lapses can trigger early interventions such as:

- Sending reminders or offering flexible payment plans to high-risk policyholders.
- Adjusting marketing strategies to target policyholders most likely to lapse based on demographic factors and policy characteristics.

6.3 Improved Customer Retention

The use of machine learning models allows for the identification of at-risk policyholders. Personalized communication, incentives, or benefit adjustments can be offered to retain these customers.

6.4 Resource Optimization

Predictive analytics helps in optimizing company resources by focusing efforts on high-risk customers rather than applying the same strategy across the board. This targeted approach improves efficiency and reduces unnecessary costs.

6.5 Model Deployment

To ensure seamless integration of predictive models into existing systems, the following steps are recommended:

- **API Integration**: Machine learning models should be deployed through APIs, allowing real-time prediction on policyholder data.
- **Continuous Learning**: As new data becomes available, models should be retrained periodically to adapt to changing patterns in policy lapse behavior.

6.6 Recommendations for Future Research

Further research could explore:

- **Deep Learning Models**: Investigating the application of neural networks to enhance predictive performance, especially for non-linear data.
- **Hybrid Models**: Combining machine learning with actuarial models to improve policy lapse prediction accuracy.
- **Sector-Specific Models**: Applying the model framework to different sectors such as health insurance, auto insurance, or even public policy, to explore broader applications of the predictive approach.

6.7 Summary

This chapter outlined the practical implications of the research, highlighting how predictive models can enhance policy management. Recommendations for deployment and future research were also provided.

CONCLUSIONS / RECOMMENDATIONS

The research presented in this dissertation aimed to address the critical issue of policy lapses in the insurance industry through the application of machine learning techniques. The models developed were designed to predict policy lapses and provide actionable insights for proactive policy management. The following key conclusions were drawn from the study:

1. Predictive Analytics Can Significantly Enhance Policy Lapse Management
This study demonstrated that predictive analytics, specifically machine learning models, can significantly improve the ability of insurance companies to anticipate policy lapses. By using historical data, the models identified patterns and factors associated with lapses, offering a shift from reactive to proactive policy management.

2. Model Selection and Performance
Through extensive experimentation with various machine learning algorithms, **XGBoost** was found to be the best-performing model for policy lapse prediction. It achieved the highest accuracy, precision, recall, and AUC, making it the most suitable model for real-world deployment. Gradient Boosting and Random Forest also performed well, while Logistic Regression and Decision Trees, though useful as baselines, were less effective in capturing the complexity of the data.

3. Key Predictors of Policy Lapse
The most influential predictors of policy lapse included **Payment Mode (Monthly)**, **Non-Lapse Guarantee**, and **Policy Year**. These findings offer important insights for policymakers, indicating that interventions should focus on these critical features. Policies with monthly payment modes and no lapse guarantees were found to be at a higher risk of lapsing, particularly after several years of being active.

4. Class Imbalance Handling Is Crucial for Model Accuracy
The dataset exhibited class imbalance, with more policies lapsing than not. Techniques like **SMOTE (Synthetic Minority Over-sampling Technique)** were essential for handling this imbalance and improving model performance. Without addressing this imbalance, models tend to favor the majority class, reducing their ability to identify policies at risk of lapsing.

5. Model Interpretability Is Key for Practical Use
Using feature importance and SHAP (SHapley Additive exPlanations) values provided valuable insights into how models made decisions. This interpretability is crucial for practical deployment in business settings, where understanding the "why" behind predictions is as important as the predictions themselves.

6. **Scalability and Deployment**
The models, particularly XGBoost, are scalable and can be integrated into existing systems through API deployments, making real-time predictions on new data possible. This allows for immediate interventions by insurers, significantly reducing the likelihood of policy lapses and improving customer retention.

Based on the findings of this dissertation, the following recommendations are proposed for insurance companies and policymakers:

1. Implement Predictive Models for Proactive Policy Management
Insurance companies should integrate machine learning models like XGBoost into their policy management systems. By predicting policy lapses, insurers can take proactive measures such as:

- Offering reminders or incentives to policyholders at high risk of lapsing.
- Adjusting payment schedules to make it easier for policyholders to maintain coverage.
- Introducing more flexible policies for premium payments to reduce lapses due to financial difficulties.

2. Focus on High-Risk Groups Identified by the Model
Targeted interventions can be applied to groups most at risk of lapsing, as identified by the model. For example, policies with **monthly payment modes**, or those without **non-lapse guarantees**, should receive additional attention. Tailored communication strategies and incentive programs can be developed to retain these high-risk customers.

3. Continuous Model Updating and Retraining
As new data becomes available, machine learning models should be retrained periodically to adapt to changing patterns in policyholder behavior. This ensures the models remain accurate over time and continue to provide reliable predictions.

4. Integrate Explainability in Business Operations
It is essential that insurance companies adopt tools like SHAP to explain the predictions made by machine learning models. Explainable models help build trust among stakeholders, including business managers and customers, ensuring that decisions based on model predictions are transparent and well-understood.

5. Explore the Use of Hybrid Models
In future research, insurance companies can explore hybrid models that combine traditional actuarial methods with machine learning techniques. This would allow for the best of both worlds—leveraging domain-specific knowledge while capitalizing on the predictive power of machine learning algorithms.

6. Extend Predictive Analytics Beyond Lapse Prediction
While this dissertation focused on policy lapse prediction, the same methodologies can be applied to other areas of insurance policy management, such as fraud detection, customer segmentation, and claim forecasting. Predictive analytics can provide insights across the entire lifecycle of an insurance policy.

7. Improve Data Collection and Quality
High-quality, comprehensive data is critical for building effective predictive models. Insurers should invest in better data collection mechanisms, ensuring that all relevant features are captured accurately. This will enable more precise predictions and better insights into policyholder behavior.

8. Customize Solutions for Different Policy Types
Different types of insurance policies (life, health, auto, etc.) may exhibit different patterns of lapse. Insurers should consider customizing their predictive models to address the unique factors affecting each type of policy. This would involve gathering specific datasets and fine-tuning the models accordingly.

The dissertation opens several avenues for future research:

1. Deep Learning Models
Future work could explore deep learning techniques, such as neural networks, for predicting policy lapses. While more complex, deep learning models may offer higher predictive power, particularly for large datasets with intricate relationships between features.

2. Application to Other Sectors
This study focused on the insurance sector, but the framework developed here can be applied to other industries that deal with lapses or discontinuations, such as subscription services or healthcare plans. Applying the model to these areas could reveal new insights and further improve the model’s generalizability.

3. Time-Series Analysis
Incorporating time-series analysis into policy lapse prediction could provide better temporal insights into when a policy is likely to lapse. This would allow insurers to pinpoint critical moments in a policy's lifecycle where intervention would be most effective.

BIBLIOGRAPHY / REFERENCES

The following are referred journals from the preliminary literature review:

1. Barsotti, F., Milhaud, X. & Salhi, Y. (2016). Lapse risk in life insurance: Correlation and contagion effects among policyholders' behaviors.

2. Ćurak, M., Podrug, D. & Poposki, K. (2015). Policyholder and Insurance Policy Features as Determinants of Life Insurance Lapse - Evidence from Croatia

3. Abbas, A. & Mohammed, A. (2020). Inferences about the use of linear regression and logistic regression. International Journal of Recent Scientific Research, 11(8): 39547–39552.

4. Aleandri, M. (2017). Modeling dynamic policyholder behavior through machine learning techniques. University of La Sapienza, Rome, Dept. of Statistical Sciences.

5. Bentéjac, C., Csörgő, A. & Martínez-Muñoz, G. (2019). A comparative analysis of XGBoost. Universidad Autónoma de Madrid. A

6. Ćurak, M., Podrug, D. & Poposki, K. (2015). Policyholder and insurance policy features as determinants of life insurance lapse - evidence from Croatia. Economics and Business Review, 15(3): 58–77

APPENDICES

Appendix A: Data and Features

This appendix provides an overview of the dataset used in this study for policy lapse forecasting. The features included in the dataset are crucial for understanding the predictive modeling process.

Table A.1: List of Features in the Dataset

Feature Name	Description
Entry Age	Age of the policyholder at the time of policy issuance.
Sex	Gender of the policyholder.
Policy Type 1	First type or category of the insurance policy (e.g., term, whole life).
Policy Type 2	Second type or category of the insurance policy (if applicable).
Payment Mode	Frequency of premium payments (e.g., monthly, quarterly, annually).
Policy Status	Current status of the policy (e.g., active, lapsed, terminated).
Benefit	The amount of benefit or coverage provided by the policy.
Non Lapse Guaranteed	Indicates whether the policy has a non-lapse guarantee.
Substandard Risk	Indicator if the policyholder is considered a higher risk due to health factors.
Initial Benefit	Initial benefit amount when the policy was issued.
Policy Year	Number of years the policy has been in force.
Premium	The amount of premium paid by the policyholder.
Channel1, Channel2, Channel3	Sales or distribution channels through which the policy was sold.

The dataset includes both categorical and numerical data, which were processed and transformed for machine learning algorithms.

Appendix B: Performance Metrics and Confusion Matrices

This appendix provides detailed confusion matrices for each model, summarizing the true positives, true negatives, false positives, and false negatives.

Table B.1: Confusion Matrix for Logistic Regression

	Predicted Lapse	**Predicted Non-Lapse**
Actual Lapse	21,199	14,743
Actual Non-Lapse	8,632	11,094

Table B.2: Confusion Matrix for Random Forest

	Predicted Lapse	**Predicted Non-Lapse**
Actual Lapse	21,647	9,913
Actual Non-Lapse	8,184	15,924

Table B.3: Confusion Matrix for XGBoost

	Predicted Lapse	**Predicted Non-Lapse**
Actual Lapse	24,543	11,215
Actual Non-Lapse	5,288	14,622

The confusion matrices help illustrate the trade-off between false positives and false negatives for each model.

Appendix C: Code Snippets

This appendix contains relevant Python code snippets used for data preprocessing and machine learning model implementation.

Data Preprocessing: Handling Missing Values

```
#Drop Potentially Irrelevant Columns with Null Values
df.drop(columns=['Unnamed: 20', 'Unnamed: 21'], inplace=True)

# Converting the 'SEX' column from categorical values ('F' and 'M') to numerical values (0 and 1)
df['SEX'] = df['SEX'].map({'F': 0, 'M': 1})

# Convert the 'Full Benefit?' column from 'N' and 'Y' to 0 and 1
df['Full Benefit?'] = df['Full Benefit?'].map({'N': 0, 'Y': 1})
```

Applying SMOTE to Handle Class Imbalance

```
from imblearn.over_sampling import SMOTE
sm = SMOTE(random_state=42)
X_res, y_res = sm.fit_resample(X_train, y_train)
```

Random Forest Model

```
from sklearn.ensemble import RandomForestClassifier
rf = RandomForestClassifier(n_estimators=100, random_state=42)
rf.fit(X_train, y_train)
```

Check list of items for the Final report

Item	Status
a) Is the Cover page in proper format?	Y / ~~N~~
b) Is the Title page in proper format?	Y / ~~N~~
c) Is the Certificate from the Supervisor in proper format? Has it been signed?	Y / ~~N~~
d) Is Abstract included in the Report? Is it properly written?	Y / ~~N~~
e) Does the Table of Contents page include chapter page numbers?	Y / ~~N~~
f) Does the Report contain a summary of the literature survey?	Y / ~~N~~
i. Are the Pages numbered properly?	Y / ~~N~~
ii. Are the Figures numbered properly?	Y / ~~N~~
iii. Are the Tables numbered properly?	Y / ~~N~~
iv. Are the Captions for the Figures and Tables proper?	Y / ~~N~~
v. Are the Appendices numbered?	Y / ~~N~~
g) Does the Report have Conclusion / Recommendations of the work?	Y / ~~N~~
h) Are References/Bibliography given in the Report?	Y / ~~N~~
i) Have the References been cited in the Report?	Y / ~~N~~
j) Is the citation of References / Bibliography in proper format?	Y / ~~N~~

I certify that I have properly verified all the items in the checklist and ensure that the reports are in proper format as specified in the handout.

(Signature of the Supervisor) (Signature of the Student)

Place: Kochi

Date: 6th Sep 2024