NOVA IMS Information Management School

MAA

Mestrado em Métodos Analíticos Avançados
Master Program in Advanced Analytics

# A MACHINE LEARNING APPROACH TO PREDICT HEALTH INSURANCE CLAIMS

Miguel Filipe Martins Cordeiro

Internship report presented as partial requirement for obtaining the Master's degree in Advanced Analytics, with a Specialization in Business Analytics

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

**NOVA Information Management School**
**Instituto Superior de Estatística e Gestão de Informação**
Universidade Nova de Lisboa

# A MACHINE LEARNING APPROACH TO PREDICT HEALTH INSURANCE CLAIMS

by

Miguel Filipe Martins Cordeiro

Internship report presented as partial requirement for obtaining the Master's degree in Advanced Analytics, with a Specialization in Business Analytics

**Advisor:** Nuno Miguel da Conceição António

November 2022

# ACKNOWLEDGEMENTS

This thesis would not have possible without the support, knowledge, help and patience of all of those who were in some way involved with this project, therefore I want to thank and show my deepest gratitude to all of them.

First, I want to thank my advisor, professor Nuno António, for all his insights and readiness to help. His advice was very important to improve and enrich this project.

Second, I must thank my supervisors at Multicare, Maria do Carmo Ornelas and Filipa Marques, for giving me the opportunity to develop my thesis at a leading company in its field and for offering me the guidance and knowledge necessary to understand the insurance business. Also, the help given to me by my colleagues, particularly those who worked closest to me: Mariana, both Pedro's, and Catarina; cannot be undervalued. I want to thank them for always being available to discuss their ideas with me, allowing me to grow and to improve my work.

I also want to acknowledge my friends, for their support throughout the development of this project.

For always motivating and pushing me further, I have to thank my girlfriend, Carolina. Her words of encouragement always made me move closer to my goals.

Finally, I must thank my brother, for standing by my side during this process and offering to help anyway he could; and thank my parents, for all their work and sacrifice. I have them to thank for all that I have achieved so far. Their support and guidance was, and always will be, vital in the most important times.

# ABSTRACT

Renewing health insurance contracts is, usually, an annual process, in the Tailor Made policies branch. At Multicare, this process starts three months before the end of the clients' annuity, by estimating the costs of the last quarter using information from the first three. This estimation process is critical to the renewal of insurance contracts, since, if the estimation is too high, the client will overpay for their insurance and might seek more competitive alternatives. In contrast, if the predictions are too low, it will result in losses for the company. This part of the renewal process is currently performed by a time series algorithm, specifically an ARIMA model.

This project aims to build a machine learning-based model that will provide more accurate estimations of the claims' cost and frequency, in the Inpatient coverage, to Multicare. Several algorithms were tested: Linear and Logistic Regressions, Decision Trees, Random Forests, Gradient Boosting and XGBoost; and their results were then compared to the ones of the current ARIMA model. This study showed that a machine learning technique, the XGBoost, is more powerful than the ARIMA, as it projects 9% above the real costs, against the ARIMA's global error of -25%. These conclusions can lead to changes in Multicare's approach to predicting claim costs and, consequentially, its way of doing business.

# KEYWORDS

Claim Forecasting; Ensemble; Health Insurance; Machine Learning; Tailor Made Policies; XGBoost

# INDEX

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ABBREVIATIONS AND ACRONYMS

**ACES** Agrupamento de Centros de Saúde. A subdivision of the country in terms of the primary healthcare providers' influence areas, made by the SNS.

**ARIMA** Auto Regressive Integrated Moving Average. A model commonly used for time series problems to predict future points of the series.

**AUC** Area Under the Curve. A metric that calculates the entire area underneath the ROC curve.

**GAC** Gabinete de Actuariado e Controlo. Multicare's actuarial department, where this project was developed.

**GAM** Generalized Additive Models. These are GLMs in which the response variable linearly depends on unknown smooth functions of some predictive variables.

**GB** Gradient Boosting. An ensemble technique based on a sequential strategy of ensemble formation. In these models, the learning procedure consecutively fits new models to provide a more accurate estimate of the response variable, to compensate for the shortcomings of the existing learners.

**GLM** Generalized Linear Models. A generalization of linear models by relating them to the response variable through a link function.

**IBNR** Incurred But Not Reported. Represents the costs of claims that have already happened but have not yet been reported by the client. It is estimated using actuarial techniques, such as the Chain Ladder Method.

**INE** Instituto Nacional de Estatística. Portugal's National Statistics Institute.

**MAE** Mean Absolute Error. An error metric that represents the average of the absolute errors, which in turn are the difference between the predicted values and the observed ones.

**$R^2$** Coefficient of Determination. A statistical measure that represents the proportion of variance of a dependent variable that is explained by the independent variables, in a regression model.

**RFE** Recursive Feature Elimination. An algorithm that ranks the features by relevance regarding the prediction of a target variable. It is used to discover the ideal variables to build a model.

**RMSE** Root Mean Squared Error. Accuracy metric that measures the differences between the values predicted by a model and the observed values.

**ROC** Receiver Operative Characteristic. A graph that plots the True Positive Rate and False Positive Rate and shows the performance of a classification model at all thresholds.

**SAC** Suporte Actuarial Corporate. The team, inside GAC, where this project was developed.

**SNS** Serviço Nacional de Saúde. Portugal's National Health Service.

**XGBoost** Extreme Gradient Boosting. Like GB, XGBoost is an additive ensemble of decision trees optimized for fast parallel tree construction. It is more suited to handle larger amounts of data, as it is much faster than other gradient-boosting algorithms.

# 1. INTRODUCTION

This report was developed during a year-long internship at Multicare, from October 2020 to October 2021. Multicare is a company that specializes in health insurance and belongs to the Fidelidade group, the oldest insurance company in Portugal, with its roots going back to the beginning of the XIX century. Fidelidade operates in Europe, Africa, Asia, and South America through the various companies that belong to the group.

This internship was done in Multicare's Actuarial Department, known as GAC (*Gabinete de Actuariado e Controlo*), specifically in the Corporate Actuarial Support team (SAC – *Suporte Actuarial Corporate*). The main function of this team is to develop tools that the pricing teams can use to give Tailor Made clients more adjusted rates, thus helping them to be more competitive in the health insurance market. Even though this project is only a pilot study, it is directly linked to the team's objectives, as it is the beginning of the development of a new tool that will assist pricing teams in the contract renewal process.

## 1.1. PROBLEM STATEMENT

The insurance business at Multicare is divided into two branches: Standard and Tailor Made. Standard policies are accessible to everyone. They can be acquired at any agency, by talking to a mediator, or even online. These policies have fixed prices and the different plans are already established. Tailor Made policies, as the name suggests, are the ones companies buy for their employees as well as their families: spouses and children. Unlike the Standard business, Tailor Made policies are customizable: their prices change, as well as the coverages included, according to the client's choice. This customization gives clients more negotiation power, so it is crucial that Multicare presents an offer at an adequate rate for both the company and the client. It is on this type of policies that this report is focused.

When someone uses their health insurance, it is done on one of two systems. The first one is the Healthcare Provider Network, which consists of hospitals, clinics and other providers that have a contract with Multicare, and the cost of the claims can either be fixed or have a discount. It requires that the client use their Multicare card, so it is immediately known when someone uses their insurance. The other way clients can use their insurance is through the Reimbursement system. This happens when customers use a healthcare provider that is outside Multicare's network and then submit the claim to get a refund, usually an agreed-upon percentage of the claim's cost.

The claims that happened under the Reimbursement system are not immediately loaded in the company's system, they are only known when the client submits them. The IBNR (Incurred But Not Reported) estimation is used to deal with this problem. The IBNR problem is complex and, at this point, is still under development and outside the scope of this project. Therefore, this report will only take into consideration claims that happened under the Healthcare Provider Network.

There are two critical moments in which offers need to be presented: the signing moment, when the client purchases its first policy; and the renewal moment(s). This report is focused on the latter.

The renewal problem starts three months before the end of the client's annuity. Based on the cost of the previous nine months, the costs of the last three are estimated. Based on these estimations, the company calculates the rates that are going to be applied to the client in the following year.

Therefore, this estimation process is critical in the renewal process. If the estimation is too high, the client will have an over-expensive insurance and could be compelled to find cheaper alternatives. On the other hand, if the company underestimates the client's costs, this will result in losses and the client might not be receptive the negotiate a new and, in their perspective, worse deal.

The first part of the renewal process, the estimation of the claims' costs in the last three months of the annuity, is currently done using an ARIMA model. This type of model is commonly used in time series problems but is blind to the variables the company has available. It was out of the necessity to understand whether having more information available would improve the cost and frequency estimations that the need for this project arose and it is on this part of the renewal process that the project is focused on.

## 1.2. GOAL DEFINITION

The goal of this project is to build a model, using the clients' data and Machine Learning algorithms, that will provide the company more accurate estimations of the frequency and cost of claims to, consequentially, be able to offer clients the best possible deal for both parties. These techniques can help the company make sense of the available data and use it to make more informed decisions.

The renewals are evaluated on a coverage basis and a health insurance plan usually has multiple. Due to the complexity and specificity of each coverage, this report is focused on only one: Inpatient. Besides its complexity, the Inpatient coverage is the one that presents the biggest risk to the company and, alongside the Outpatient coverage, represents the biggest percentage of total costs.

Therefore, to predict the cost and frequency of Inpatient claims, this project is based on two models: one that calculates the cost of the claims, the Cost Model, and the other which predicts how many claims each customer will have, the Frequency Model. These two events are estimated separately, as is common practice in insurance businesses, as they are dissimilar and are often explained by different variables.

Additionally, and as was mentioned in the previous section, it is important to note that this project will only answer the first part of the renewal process: the estimation of the last quarter's costs of each client's annuity. To complete the renewal process, the costs of the next annuity need to be calculated afterwards, but this step is outside the scope of this project.

# 2. LITERATURE REVIEW

In this chapter, some techniques that are currently being used to predict claim frequency and severity are explored. It is important to note that there is little literature regarding health insurance, hence the majority of works explored in this section regard auto insurance. Nonetheless, the problem in hand is the same: how to correctly estimate claims to better price insurance policies.

Traditionally, actuaries have been using simple supervised learning statistical models, such as Generalized Linear Models (GLMs) and Generalized Additive Models (GAMs), to deal with the problems of claim estimation and loss reserving (Albrecher et al., 2019). However, with the increasing amount of data available and a number of more sophisticated algorithms appearing, the paradigm has been changing in favour of machine learning techniques.

The work of Guelman (2012) compared the performance of GLMs with Gradient Boosting Trees when used for auto insurance claim prediction. The author decided to explore Gradient Boosting (GB) due to its uniqueness in the sense of achieving both predictive accuracy and allowing for model interpretation, as this last feature is considered to be of high importance in a business environment where models are often approved by non-statistically trained decision makers who need to understand the models' outputs. In this paper, the author decided to split the frequency and severity problems, as is common practice, since some factors affect them differently, and the response variable of the frequency problem was coded as binary, given that only a few records had more than one claim. To guarantee that the model performance is an accurate approximation of the expected performance in future cases, the test set records' policy dates are posterior to the ones in the train dataset. Finally, the author used an undersampling technique to deal with the fact that the actual claim frequency is only 3.51%. The variables used in this study were divided into four categories: Driver characteristics, Policy characteristics, Accident/conviction history and Vehicle characteristics. The results showed the most important variable for the frequency model is related to the driver characteristics: the years licensed. As for the severity model, the most important variable is associated with vehicle characteristics: the vehicle age. To compare the performance of the two algorithms, the author used a ratio of the rate that would be charged based on the GB model to the rate that would be charged based on the GLM. Afterwards, the observations were grouped into five similarly sized buckets ranked by the ratio. Finally, for each bucket, the GLM-loss ratio was calculated. Based on the results shown in Figure 2.1, the author concluded that the upward trend in the GLM-loss ratio curve indicates the higher predictive performance of GB relative to GLM.

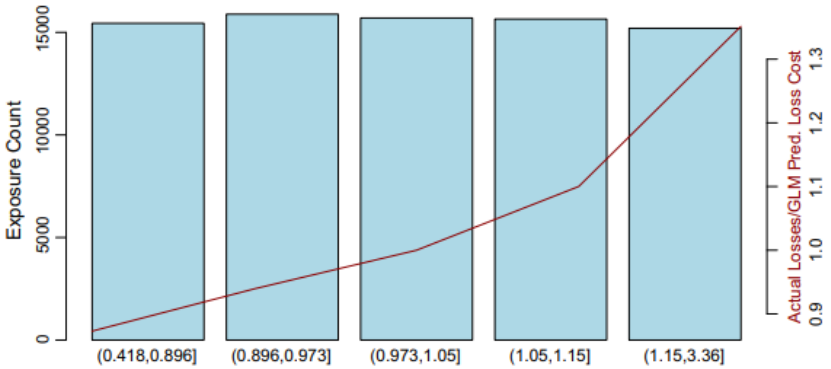


Figure 2.1 - Prediction accuracy of GB relative to GLM from Guelman (2012)

Pesantez-Narvaez, Guillen, and Alcañiz (2019) used Logistic Regression and Extreme Gradient Boosting (XGBoost), with both tree and linear boosters, to predict the occurrence of accident claims in motor insurance. They used a database from a Spanish insurance company which comprised information regarding the driver, the vehicle characteristics and driving patterns and habits. The authors' goal is to predict whether or not each driver will file accident claim with fault, during the observation period, so the response variable was coded as binary. Using sensitivity, specificity, accuracy and the root mean square error (RMSE), the authors evaluated the performance of the three approaches. It was possible to conclude that the Logistic Regression and the XGBoost with the linear booster provided similar results, and the XGBoost with the tree booster overfitted the data. To correct the overfitting problem, so that it can be possible to determine if the XGBoost with the tree booster outperforms the Linear Regression, the authors decided to apply a regularization technique, first the L2 (Ridge) and then L1 (Lasso). The L1 regularization method was considered to be the best solution for this problem, as the L2 did not improve the evaluation metrics whatsoever. After applying the regularization, the authors concluded that the XGBoost predictive performance was still very similar to the Logistic Regression. However, they do not claim that the XGBoost is not an improvement on traditional, simpler algorithms, given that parameter tuning was not implemented, and the authors suggest that applying these procedures might improve the model's predictive capacity.

Hanafy and Ming (2021) conducted a similar study to Pesantez-Narvaez et. al (2019): the goal is also to predict whether a driver will file an accident claim, yet the database used in this study belongs to a Brazilian insurance company and, in addition to the Logistic Regression and XGBoost, the authors also evaluated the performance of a Random Forest, Decision Trees, Naive Bayes and K-Nearest Neighbours, using accuracy, error rate, kappa statistic, sensitivity, specificity, precision, F1-score and the Area Under the ROC Curve (AUC). Before testing the proposed machine learning algorithms, the authors proceeded to treat the data for it to be ready for the modelling stage. The first step of the pre-processing stage was to oversample the minority class records, as there are only 3.7% instances of this class. Afterwards, missing values were dealt by deleting two variables, which had around 70% of missing values, and imputing the mode, for categorical variables, and the mean, for continuous variables. Using the Pearson Correlation, a set of variables was removed due to not being at all correlated with the target variable. Finally, the authors optimized some hyperparameters of the different models, using a grid search. After performing all transformations deemed necessary to the data, the different algorithms were evaluated according to the aforementioned metrics. The results

showed that the random forest outperformed all the other models in every metric with the exception of the error rate, specificity and recall. Regarding the error rate, the model struggled, but in the last two metrics, the random forest was the second-best algorithm (see Figure 2.2).

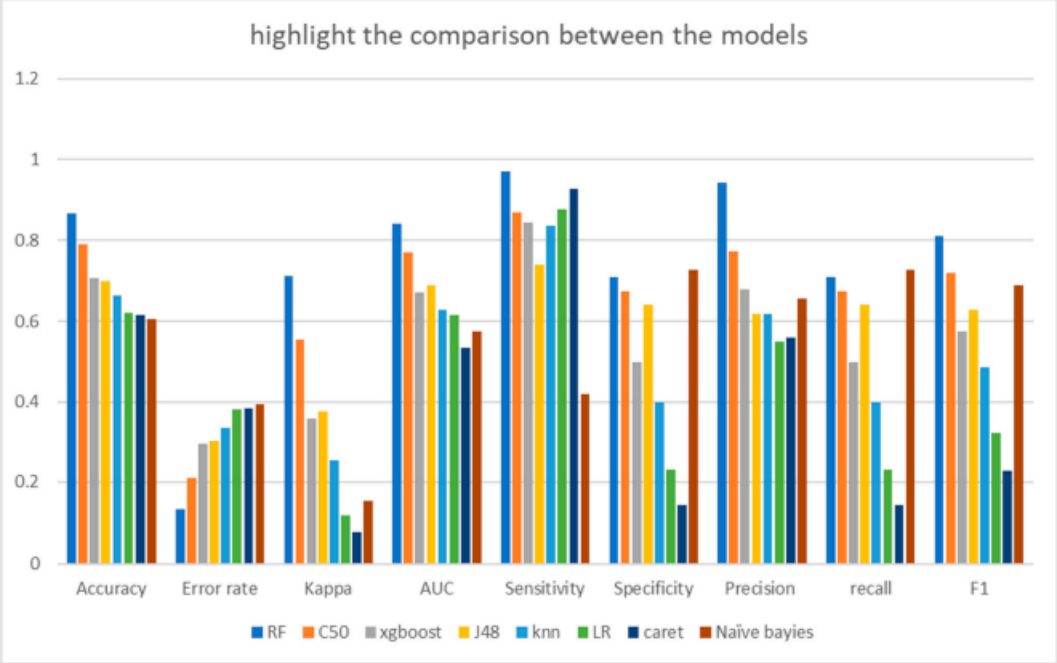


Figure 2.2 - Model Comparison from Hanafy and Ming (2021)

Fauzan and Murfi (2018) proposed using XGBoost to predict auto insurance claims and compared its performance to other ensemble learning algorithms, such as AdaBoost, Stochastic Gradient Boosting and Random Forest, and a Neural Network, using the normalized Gini Coefficient as the comparison metric. This study is a classification problem as its goal is to predict whether the policyholders file an insurance claim or not. In this paper, the authors reinforce the importance of claim prediction in insurance personalization. If the claims are correctly predicted, the insurance company can prepare the right type of policy for each potential client. As is customary with claim prediction problems, the dataset is imbalanced, with only 6% of clients filing accident claims. The dataset also has a significant number of missing values, however, the authors decided not to address these problems. After performing feature selection, reducing the number of variables from fifty-seven to twenty-four, the authors proceeded to tune the hyperparameters of the XGBoost algorithm using a six-stage grid search, where different parameters were optimized at each stage, updating the classifier with the result of the previous one. To compare the results of the XGBoost model with the previously mentioned ones, some changes needed to be made to the data, as those models are not prepared to deal with missing values: missing values in binary variables were filled with the median of the variable; missing values in integer features were filled by the rounded value of their mean; and the missing values in float features were filled with the value of their mean. Additionally, a similar grid search as the one used for the XGBoost was implemented, suited to the hyperparameters of each model. After comparing the performance of all models, the authors concluded that the Stochastic GB gave similar scores as the XGBoost. This is explained by the fact that Stochastic GB has the same underlying methods as the XGBoost. Also, Stochastic GB was trained using the imputed data whereas the data

used to train the XGBoost model had missing values. This suggests that the imputing strategy worked well and could be used in the XGBoost as well. Nevertheless, XGBoost still outperforms the other algorithms and is the authors' choice to answer the problem of claim prediction.

Abdelhadi, Elbahnasy, and Abdelsalam (2020) evaluated how Artificial Neural Networks, Decision Trees, Naive Bayes classifiers and XGBoost perform when predicting auto insurance claims. The authors used the mentioned algorithms to predict if the policyholder will file a claim during the observation period, using a dataset comprising 30.240 observations and twelve variables, including drivers' information, vehicle data and previous claim history. After dealing with the missing values problem, using a multiple imputation process, the authors decided to discretize some variables using the 25, 50 and 75 percentiles, binning the observations. Additionally, categorical data, such as gender or marital status, was encoded and, due to the different nature and scales of the features, the data was standardized. After fitting the treated data to the models in study, the authors concluded that the XGBoost and the Decision Tree performed significantly better than the Neural Network and Naïve Bayes classifier, in both accuracy and AUC, with the XGBoost being the best one.

As mentioned before, literature about the use of machine learning techniques to predict health insurance claims is scarce. However, this scenario changes if other types of insurance are considered, especially regarding motor insurance, where many studies have been conducted, as this section shows. The rationale behind predicting claims is the same, or at least similar, for all branches of insurance and that is the reason why this section is focused, almost exclusively, on motor insurance and the prediction of this type of claims. All the authors of the studies in analysis used at least one tree-based algorithm and compared them to other machine learning algorithms. In every case, one of the tree-based techniques was deemed as the most accurate model and, therefore, the best solution for the problem in study. The tree algorithms that were chosen the most were either Gradient Boosting (Guelman, 2012) or XGBoost (Abdelhadi et al, 2020; Fauzan & Murfi, 2018). The problem was divided into two parts: frequency and severity, also known as cost (Guelman, 2012), and the need for a resampling technique, due to the low claim frequency, emerged (Hanafy & Ming, 2021). The importance of proper data preparation (Abdelhadi et al, 2020), variable selection (Hanafy & Ming, 2021) and tuning of the models' hyperparameters (Fauzan & Murfi, 2018; Pesantez-Narvaez et al, 2019) was also highlighted. Learning from the authors of the aforementioned studies, their assumptions and conclusions were used and served as the basis for this project.

# 3. METHODOLOGY

As mentioned previously, the goal of this report is to build models that use the company's data on clients to predict claims and their costs more accurately, and to assist the subscribers during the policy renovation process. As is common in the insurance business, two main components need to be considered to estimate the clients' costs: the number of claims and the cost of those claims or, in other words, the frequency and the costs. These two events are estimated separately, as they are dissimilar and are often explained by different variables, thus there is a need to have two different data frames: the Cost Dataset and the Frequency Dataset.

This data-driven renovation process is still in its first stages in the company, so this report is detailing a pilot project which comprises only a few selected clients (companies) and the data spans over the course of three years, from 2017 to 2019.

Both datasets were extracted from the company servers, using SAS Enterprise Guide, by merging a panoply of different tables containing the information deemed necessary to build the proposed models, for example, customers' gender, age, area of residence, claim and diagnosis history, among others.

The data was properly handled, treated, and anonymised to protect the clients' identity and to comply with data protection rules.

The pre-processing and modelling stages of this project were performed using Python, namely the following packages: pandas, NumPy, csv, matplotlib, seaborn, plotly, category_encoders, scikit-learn, optuna, imblearn, collinearity and xgboost.

## 3.1. FREQUENCY DATASET

This dataset was used to build the frequency model and it has every information that is associated with each pilot client. It is composed of 106,609 insured people, however, given that the goal of this project is to calculate the cost of the last three months, the information needs to be displayed in a monthly view, so each person is going to be represented twelve times, one for every month of the annuity, resulting in a dataset with 1,279,308 observations.

Since the goal of this study is to estimate the frequency of inpatient claims, the target variable is the number of claims each client is going to file in that specific month. As there are almost no clients who file more than one claim in a month, it was decided to code the target variable as binary, indicating if the client filed an inpatient claim or not.

The remaining features can be divided into two main groups: the ones that are at the insured person level and characterize them individually; and others regarding social-economic and demographic indicators. This last group of variables was extracted from external sources such as the Portuguese National Statistics Institute, INE (*Instituto Nacional de Estatística*), or the Portuguese National Health Service, SNS (*Serviço Nacional de Saúde*).

The first group of variables features the target variable; clients' personal information, like age, gender and area of residence; and details about the client's plan, such as spending limits, co-pays and deductibles. Additionally, information regarding the clients' Inpatient claims in the previous six months was included, as well as a binary variable indicating if there is any history of oncologic diagnosis for each person in the database. Finally, information concerning the outpatient claims of the previous three months was included, as most, if not all, inpatient claims are scheduled and are, therefore, preceded by outpatient claims, mainly clinical analysis and medical appointments. Thus, it is expected that the increase in these claims can help predict the inpatient ones. This information is shown in Table 3.1.

| Variable | Type |
|---|---|
| Claim | Binary |
| Age | Numerical |
| Gender | Categorical |
| Spending Limit | Numerical |
| Copay | Numerical |
| Deductibles | Numerical |
| Oncologic Diagnosis | Binary |
| Amount of Previous Inpatient Claims | Numerical |
| Cost of Previous Inpatient Claims | Numerical |
| Amount of Previous Outpatient Claims | Numerical |
| Cost of Previous Outpatient Claims | Numerical |
| Days since last Outpatient Claim | Numerical |
| District | Categorical |
| Health Region | Categorical |

Table 3.1 – Frequency Database Internal Variables

The second group of variables, shown in Table 3.2, includes some social-economic and demographic indicators, extracted from external sources, as mentioned previously. These features include information about medical indicators, such as vaccination rates, medical screenings, or hospital admission rates; demographic indicators of the clients' area of residence; and, lastly, road networks information used to calculate the distance between the clients' residence and the closest healthcare providers, as well as the travel time. In this table, it is possible to see the description of the variables, their type, and sources.

The data extracted from INE is associated to each client through their statistical subsection, which corresponds to their block, in urban areas, the place or part of the place, in rural areas, or residual areas that may or may not contain statistical units.

The data retrieved from the SNS Transparency Portal is linked to the insured person by the closest public Hospital influence area or, in most cases, by their ACES (*Agrupamento de Centros de Saúde*) influence area. The ACES is how the SNS divides the country in terms of influence areas of the primary healthcare providers, the Health Centres.

| Variables | Type | Source |
|---|---|---|
| Residents by Education Level | Numerical | INE |
| Residents by Job Sector | Numerical | INE |
| Retired Residents | Numerical | INE |
| Responsiveness of Public Health Providers | Numerical | SNS Transparency Portal |
| Oncological Screenings | Numerical | SNS Transparency Portal |
| Number of Appointments | Numerical | SNS Transparency Portal |
| Number of Surgeries | Numerical | SNS Transparency Portal |
| Vaccination Rates | Numerical | SNS Transparency Portal |
| Responsiveness of Private Health Providers | Numerical | Private Entities |
| Distance to closest Public Hospital | Numerical | Road Network |
| Travel Time to closest Public Hospital | Numerical | Road Network |
| Distance to closest Private Hospital | Numerical | Road Network |
| Travel Time to closest Private Hospital | Numerical | Road Network |

Table 3.2 – Frequency and Cost Databases External Variables

### 3.1.1. Exploratory Data Analysis

By exploring the datasets, it is possible to gain a better sense of the data available for the project and to understand what adjustments are going to necessarily be done. This section is focused on performing an exploratory analysis of the data, to gain insights into the frequency dataset.

In Table 3.3, it is possible to see that Inpatient claims are not very common, and the dataset reflects it, with over 99% of records filing no claims at all, in the observed month. Additionally, only 0.01% of observations file two or more claims in a month, so the number of claims can be seen as a binary variable. This results in 99.7% of observations not filing claims and, oppositely, 0.3% file inpatient claims in that month, as is shown in Table 3.4.

As this is a rare event, the dataset is naturally imbalanced. Using the dataset in this condition to make predictions can lead to biased conclusions, thus this issue will need to be addressed.

| Number of Claims | Count | Frequency (%) |
|---|---|---|
| 0 | 1,275,745 | 99.7215 |
| 1 | 3,445 | 0.2693 |
| 2 | 114 | 0.0089 |
| 3 | 3 | 0.002 |
| 4 | 1 | 0.001 |

Table 3.3 – Claim Distribution by Number of Claims

| Has Claim | Count | Frequency (%) |
|---|---|---|
| No | 1,275,745 | 99.7 |
| Yes | 3,563 | 0.3 |

Table 3.4 – Claim Distribution by Binary Response

In terms of gender distribution, as Figure 3.1 displays, the dataset is relatively balanced with 55% of insured people being females. Figure 3.2 shows that the distribution is very similar for the number of claims, with 53% of claims associated to females. Given that both distributions are similar, this could mean that gender is possibly not an important variable to estimate claim frequency.

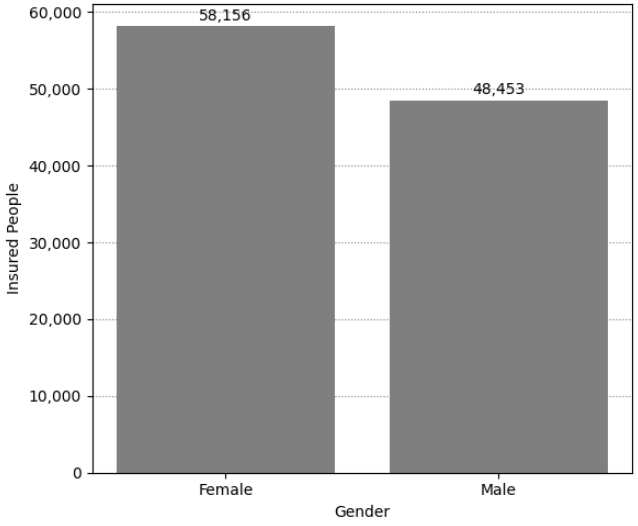


Figure 3.1 - Insured People by Gender

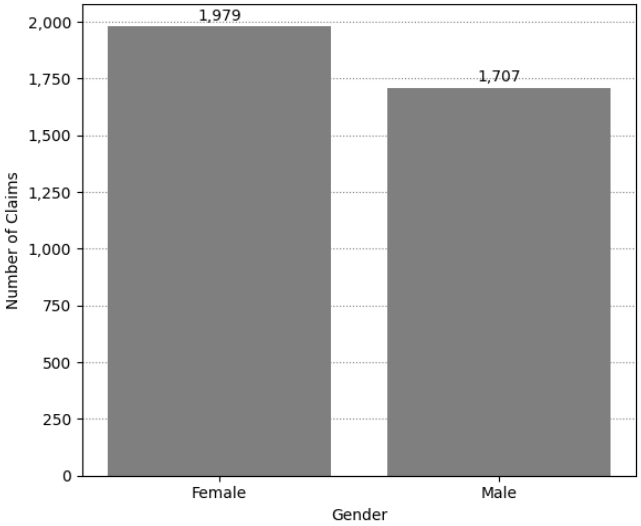


Figure 3.2 - Claims by Gender

In Figure 3.3, information regarding age distribution is on display. The median age is 35 years old and most insured people are between the ages of 20 and 60, which was expected given that this sample is based on the insurance plans companies purchase for their workers. The under 18-year-olds

correspond to the workers' children and the spike in the age 0 shows that there is a significant number of new-borns. Additionally, it is possible to see there are some insured people over the age of 65, mostly due to some insurance plans for retired people.

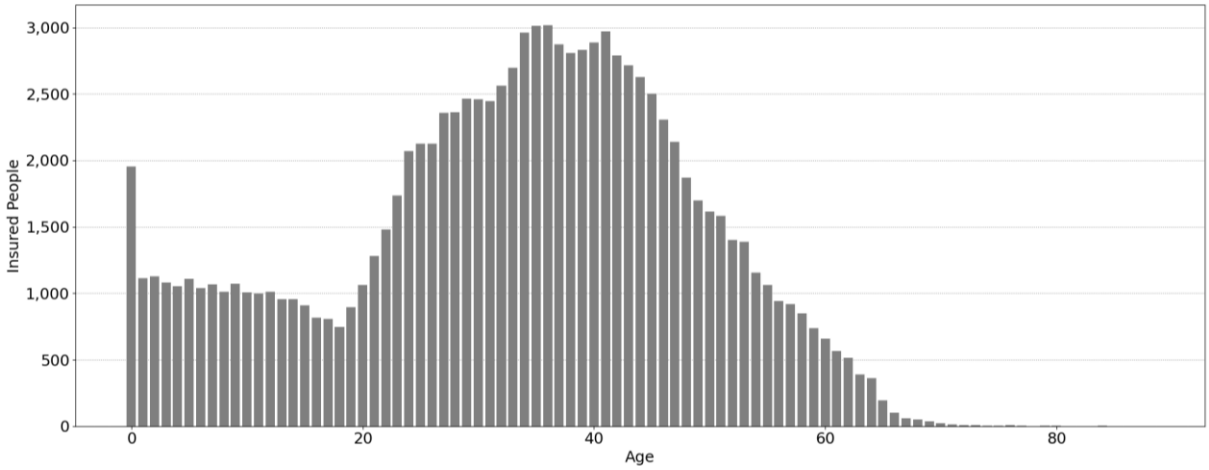


Figure 3.3 - Insured People by Age

Moreover, it is even easier to see the distribution mentioned above by looking at Figure 3.4, the insured people by age group. The predominance of individuals between the ages of 20 and 60 is evident and this can be explained, as well as the reduced number of people with later ages, by the fact that most individuals in this sample belong to the working population.

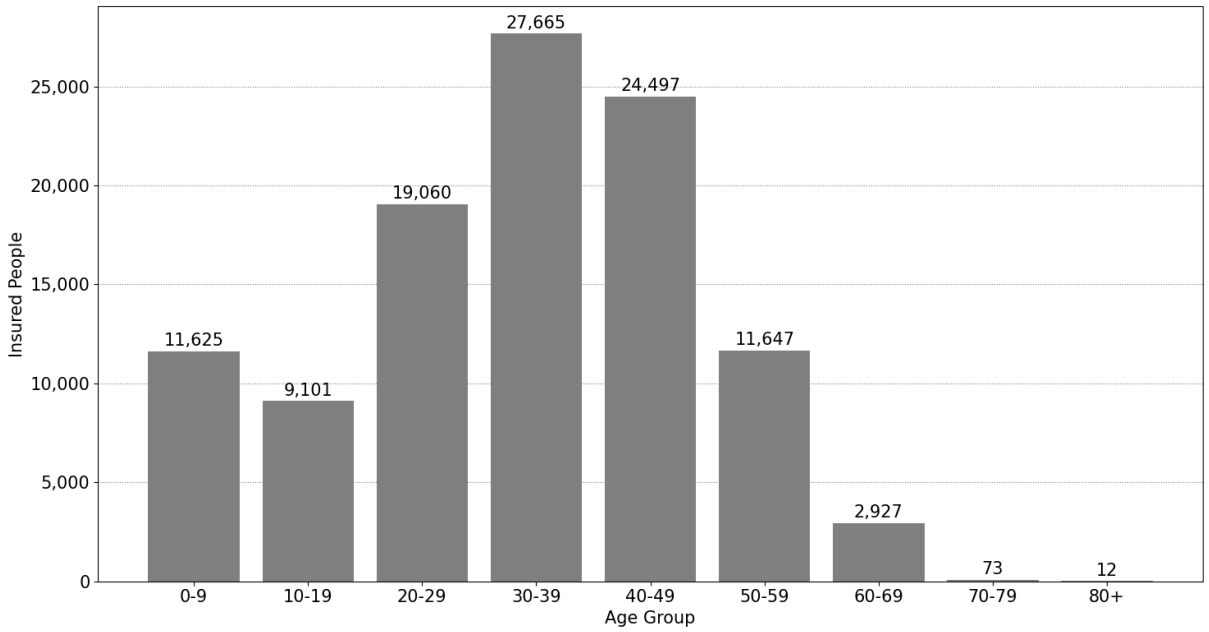


Figure 3.4 - Insured People by Age Group

Nevertheless, the scenario changes slightly when analysing the number of claims by age. The median age of the insured people who actually file inpatient claims is 42 years old and it is possible to see, in Figure 3.5 and Figure 3.6, that the most represented age group is 40-49, whereas the age group with most people is 30-39. Furthermore, the age groups 20-29 and 60-69 have roughly the same amount of claims, even though the first group has, more than six times, more people than the second.

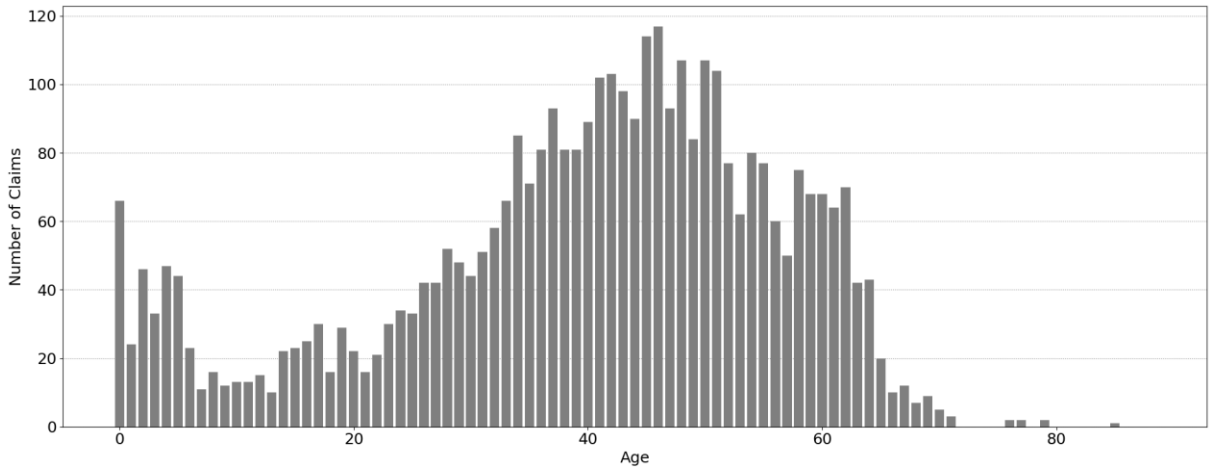


Figure 3.5 - Number of Claims by Age

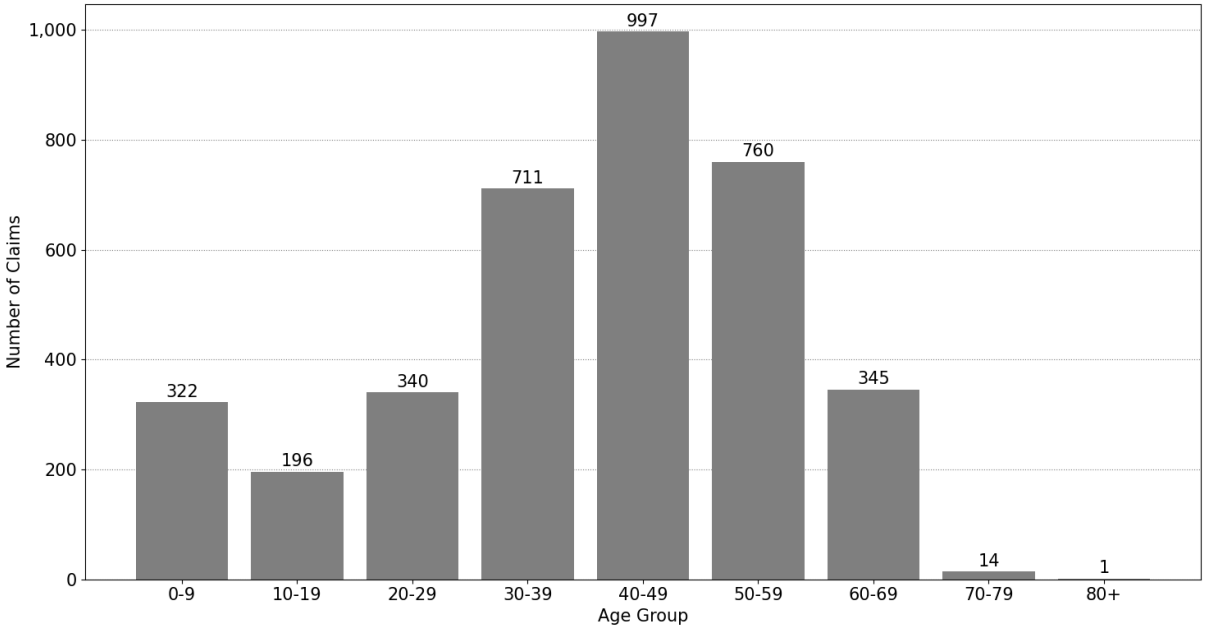


Figure 3.6 - Number of Claims by Age Group

Combining the information about both the number of people and the number of claims, it is possible to obtain the ratio of claims by insured person. Figure 3.7 evidences that this ratio goes up with the age of the insured person, therefore this information suggests that age can be an important factor to estimate the frequency of inpatient claims.

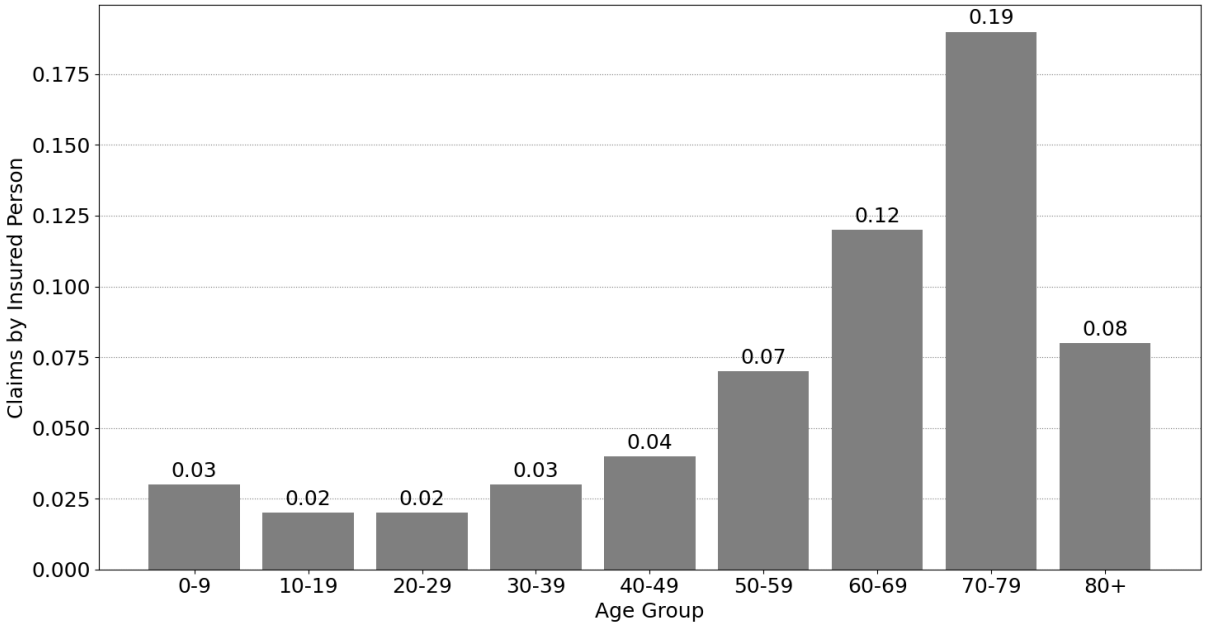


Figure 3.7 - Claims per Insured Person by Age Group

Figure 3.8 shows the distribution of insured people by district. When looking at it, three of them stand out as the most common: Lisbon, Porto and Setúbal. These three districts are the residence of roughly 75% of the insured people in this study and this distribution is similar for the number of claims, shown in Figure 3.9, with 83% of them being associated to clients from Lisbon, Porto or Setúbal.

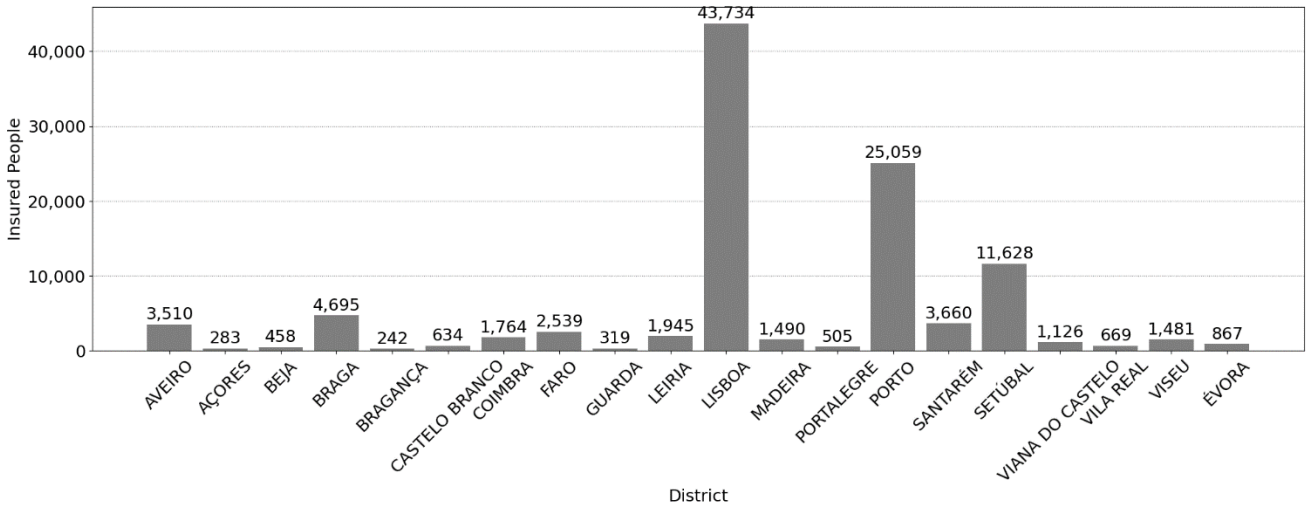


Figure 3.8 - Insured People by District

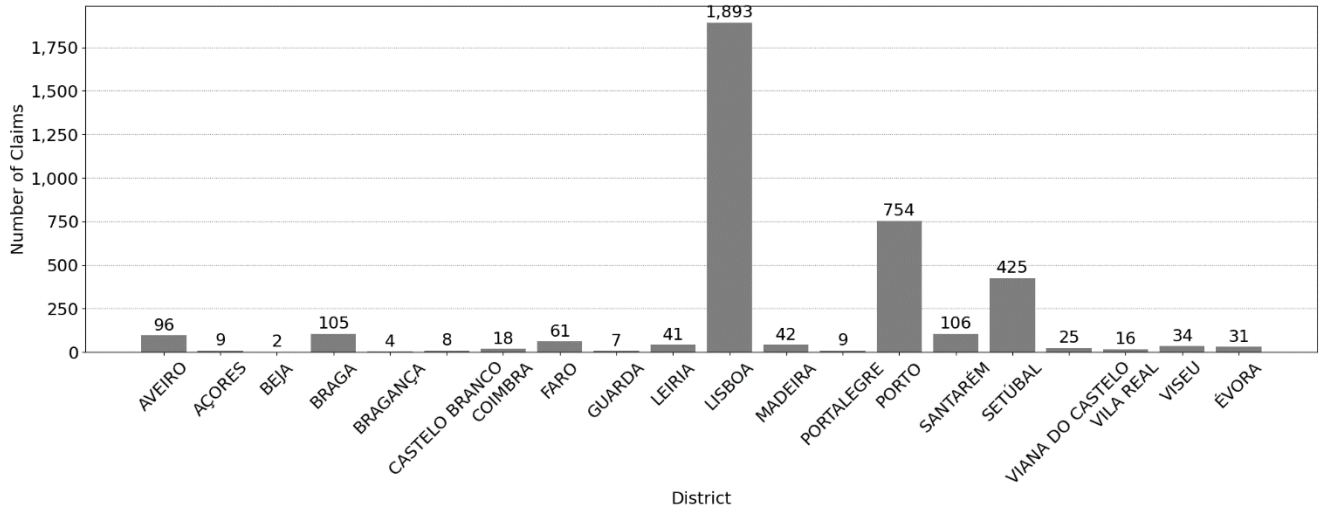


Figure 3.9 - Number of Claims by District

Given the overwhelming amount of people living in the districts mentioned previously, it was expected that the majority of claims were associated to insured people from those districts. For that reason, it can also be interesting to analyse the ratio of claims per insured person by district. Figure 3.10 shows that Lisbon, Porto and Setúbal still stand out, but are now accompanied by Évora and the Azores. However, it is important to remember that these districts have significantly fewer people living there.

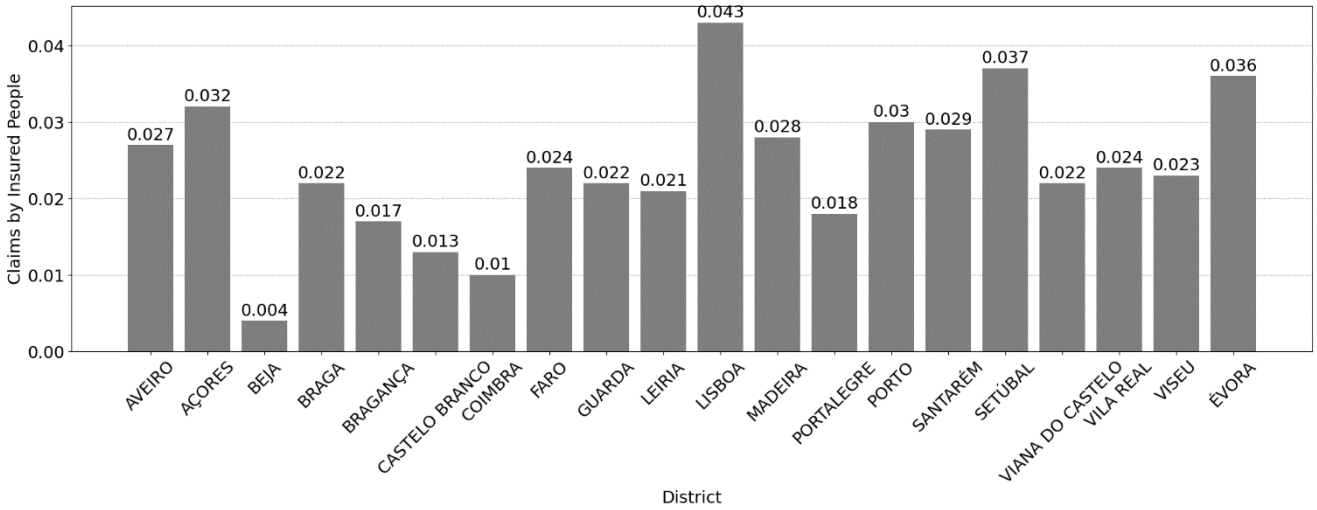


Figure 3.10 - Claims per Insured People by District

Another variable that can also be important to the process of estimating the frequency of inpatient claims is the spending limit of this coverage, as it dictates how much a person can spend. Additionally, it is important to note that the 500,000€ spending limit is fictitious and represents the unlimited spending limits, as it will be explained further ahead.

Figure 3.11 demonstrates that the majority of insured people have a spending limit of either 15,000€ or 17,500€, accounting for around 50% of the sample. There is also a significant number of people with an unlimited spending limit and 50,000€ is the fourth most common spending limit.

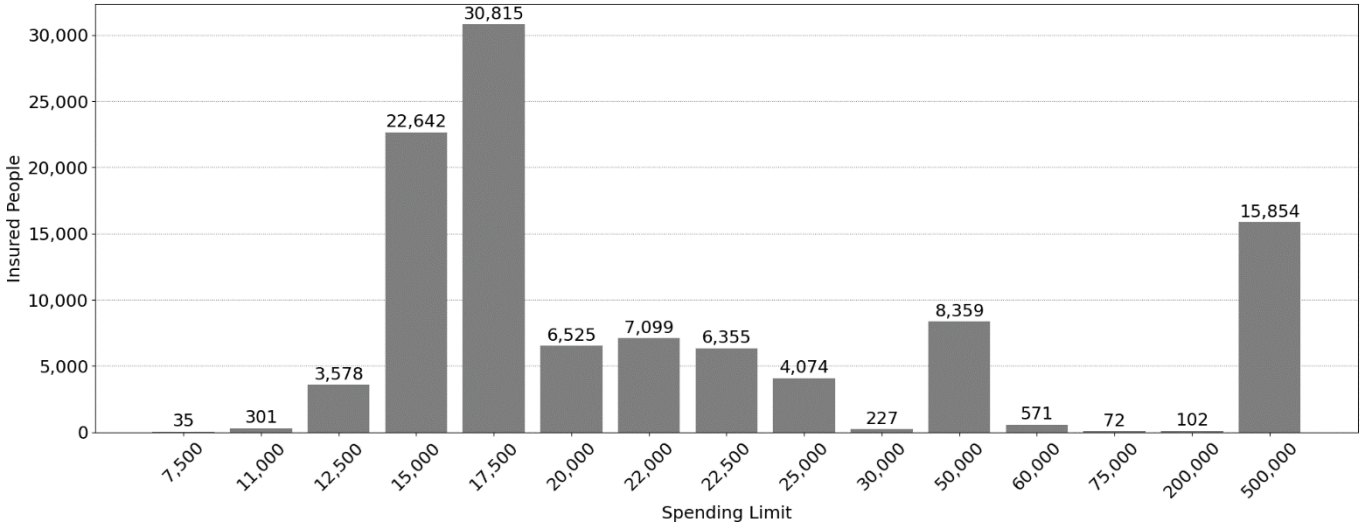


Figure 3.11 - Insured People by Spending Limit

The distribution of the claims by spending limit is similar to the one mentioned above, with the four most common spending limits being the same. However, the people with the most common spending limit, 17,500€, are only the third biggest spenders, as Figure 3.12 shows. Because of this unexpected behaviour, the ratio of claims per insured person by spending limit was also analysed.

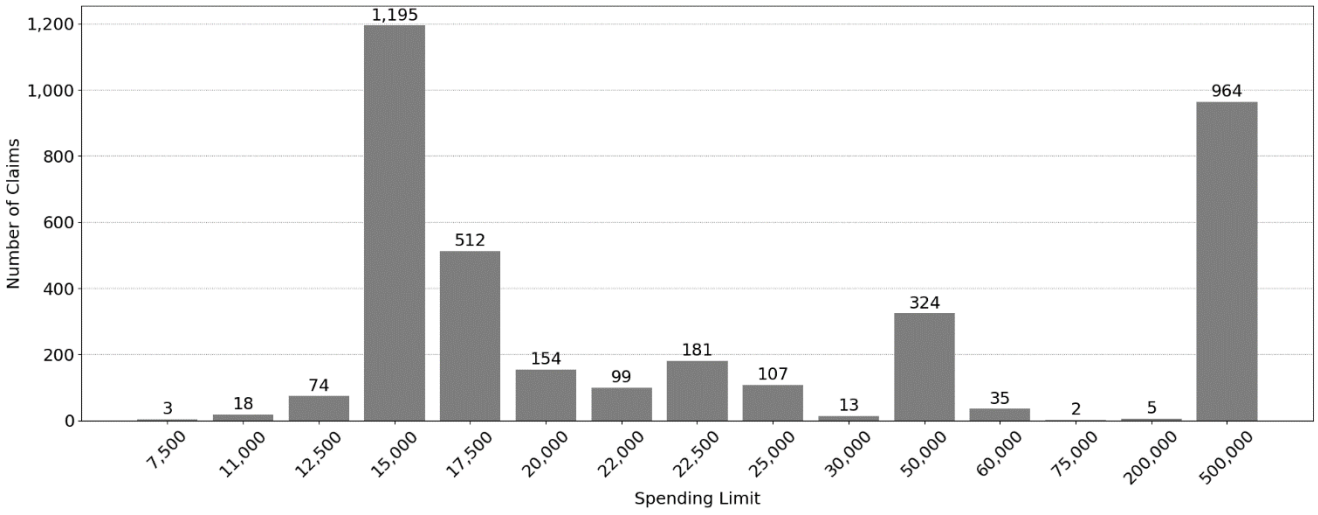


Figure 3.12 - Number of Claims by Spending Limit

As it is possible to see in Figure 3.13, the pattern of the ratio of claims per insured person is rather irregular. In other coverages, there is usually an ascending trend between this ratio and the spending

limits. In the case of the Inpatient coverage, this does not apply, most likely due to it not being a consumption coverage, unlike Outpatient, for example.

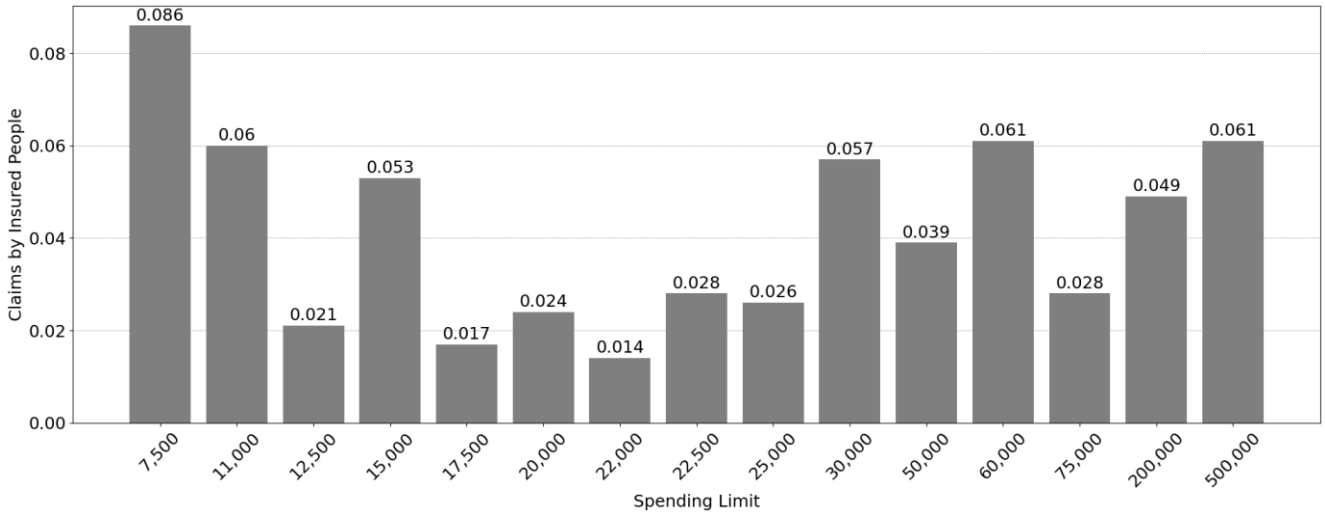


Figure 3.13 - Claims per Insured People by Spending Limit

Having an oncological diagnosis is considered to be something that can lead to more and more frequent inpatient claims. Figure 3.14 shows that the number of insured people who have an oncological diagnosis is reduced, amounting to only 0.2% of observations, and that is reflected in the number of claims, with 96% of claims being filed by people without an oncological diagnosis. However, when analysing the ratio of claims by insured person, it is possible to conclude that the oncological diagnosis has a big influence on the number of inpatient claims, as is reflected in Figure 3.15.

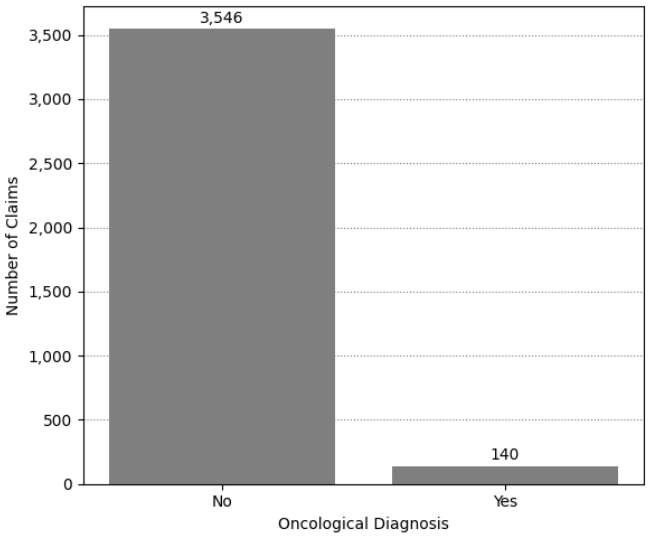


Figure 3.14 - Claims by Oncological Diagnosis

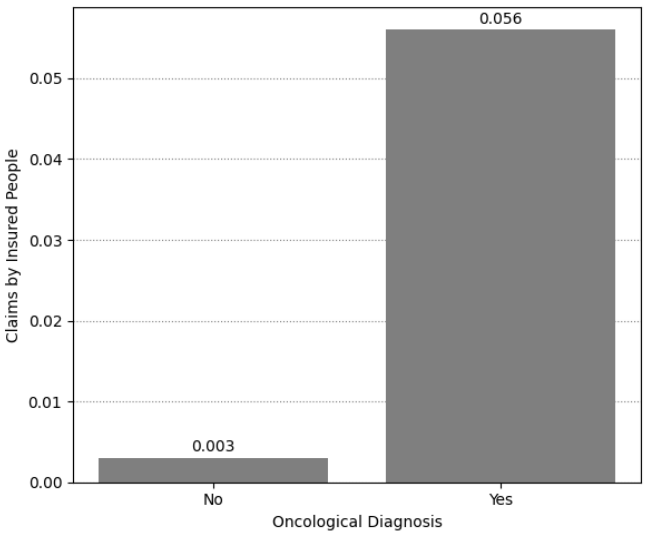


Figure 3.15 - Claims per Insured Person by Oncological Diagnosis

The amount of money spent on Outpatient claims can also be a factor that might affect Inpatient claims and it could be interesting to check the relation between them, as it is very rare to have an Inpatient claim without any medical appointment or auxiliary exams. With this goal in mind, the expenditure in Outpatient claims in the previous three months was plotted, in Figure 3.16, to see if there is a relationship between Inpatient and Outpatient claims.

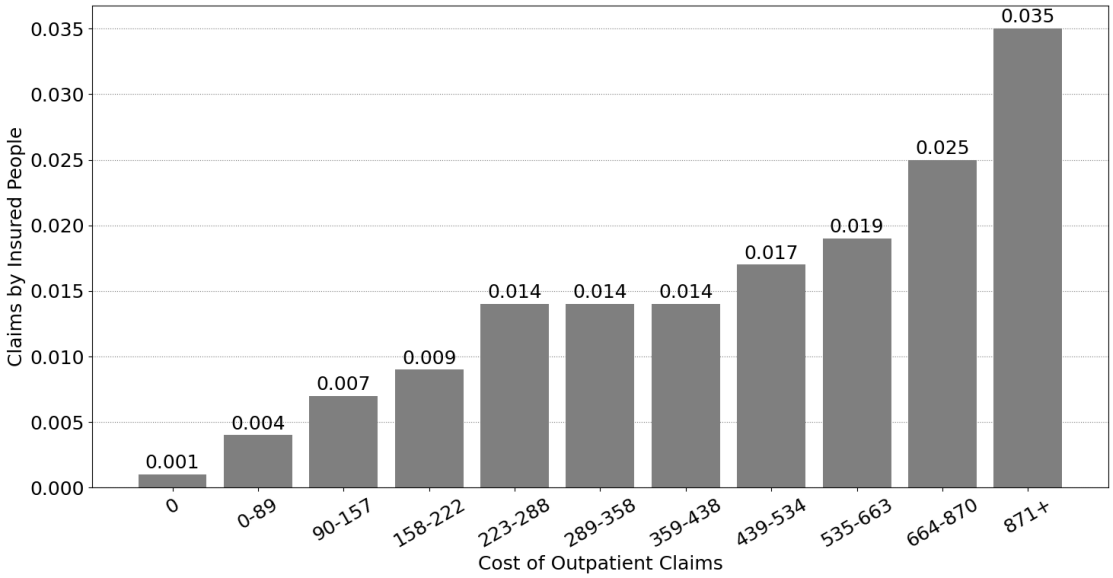


Figure 3.16 - Claims per Insured Person by Cost of Outpatient Claims

In the figure above, it is clear that the Claims per Insured Person ratio, the division between the number of Inpatient claims by the number of insured people, increases with the cost of the Outpatient claims. Therefore, it is possible to conclude that there is a relation between the expenditure on Outpatient claims and the number of Inpatient claims.

## 3.2. COST DATASET

This dataset was used to build the cost model and it has information regarding every claim belonging to the pilot clients, during the study period. It is composed by 3.686 claims.

Just like in the frequency problem, the variables present in this dataset can be divided into two groups: the ones at the insured person level and the others regarding social-economic and demographic indicators. The description of the first set of variables, as well as their types, can be seen in Table 3.4. Regarding the second group of features, it is the same as the one present in the frequency dataset and its information is presented in Table 3.2.

| Variables | Type |
|---|---|
| Claim Cost | Numerical |
| Age | Numerical |
| Gender | Categorical |
| Spending Limit | Numerical |
| Copay | Numerical |
| Deductibles | Numerical |
| Oncologic Diagnosis | Binary |
| Amount of Previous Inpatient Claims | Numerical |
| Cost of Previous Inpatient Claims | Numerical |
| District | Categorical |
| Health Region | Categorical |

Table 3.5 – Cost Database Internal Variables

### 3.2.1. Exploratory Data Analysis

Just like it was done for the frequency dataset, in this section, the cost dataset was explored to understand the data and what adjustments are going to have to be done.

The cost dataset is composed of every claim that each insured person filed in the period in study. This value amounts to 3,686 claims and the average value of a claim is around 3,000€.

As it was mentioned in section 3.1.1, 53% of claims are associated with female clients. In terms of cost, the distribution is almost identical: 54% of the total cost concerns females, as Figure 3.17 shows.

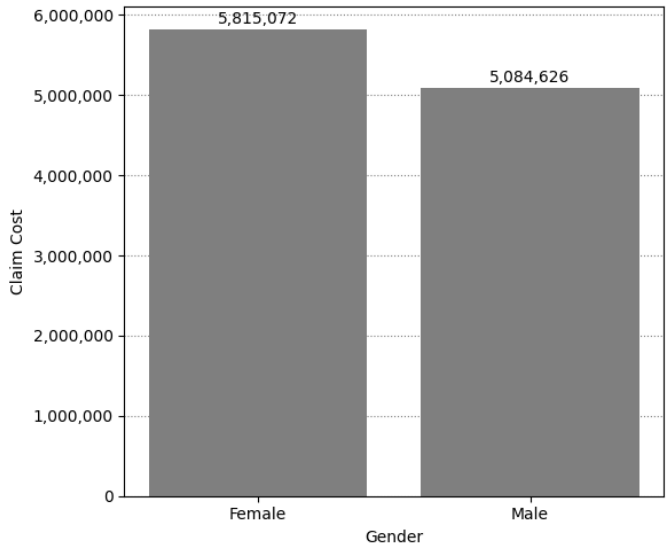


Figure 3.17 - Claim Cost by Gender

The median age of the people who file inpatient claims is 42 years old. However, the distribution of costs is not symmetrical, which indicates that older people tend to have more expensive claims.

In Figure 3.5 - Number of Claims by Age, it is possible to see that the number of claims is concentrated around the age of 40 and that there are not many claims of people above 60. Plus, there is a small spike of claims at the age of 0. Looking at Figure 3.18, it is visible that there are high costs associated with people between 60 and 65 years old, nevertheless the costs now tend to focus between the ages of 40 and 50. Also, even though newborns have a significant number of claims, their costs are not that high.

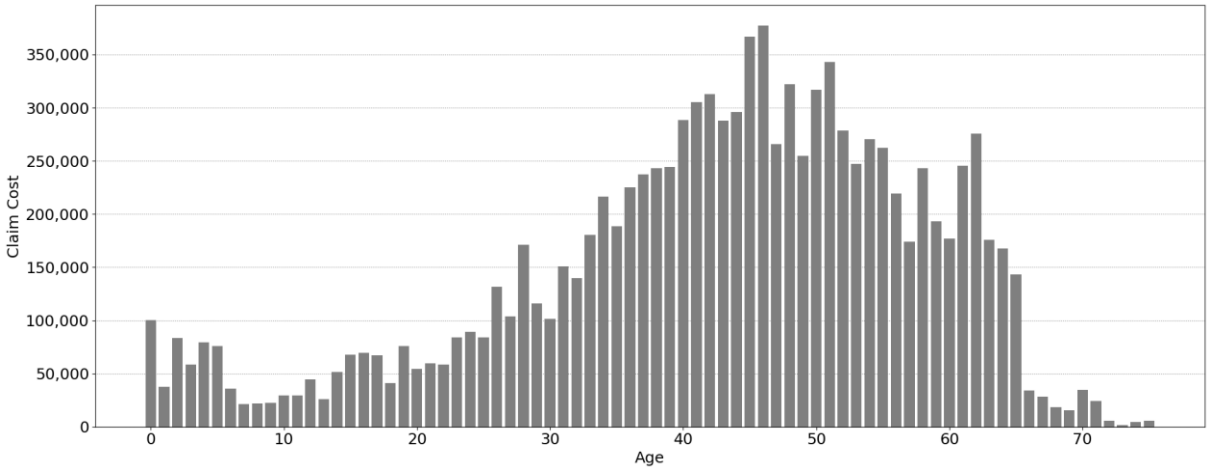


Figure 3.18 - Claim Cost by Age

In Figure 3.19, the cost per claim by age is displayed. It confirms the trend shown in the previous figure, as the cost per claim increases with age, although at a smaller pace. It shows a steady growth until the age of 65, where some spikes start to appear. Figure 3.7 - Claims per Insured Person by Age Group, had already shown that the older age groups have a higher claim by insured person ratio, however this can be explained by these age groups not having many people (see Figure 3.4) or many claims (see Figure 3.6).

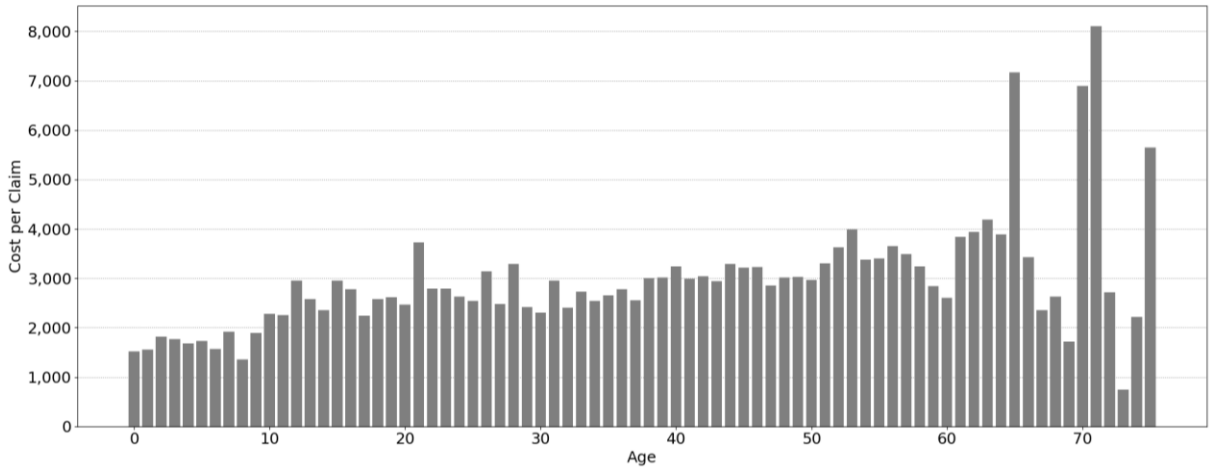


Figure 3.19 - Cost per Claim by Age

As it was mentioned previously, 83% of claims concern insured people who reside in either Lisbon, Porto or Setúbal. Thus, it is no surprise to see, in Figure 3.20, that 84% of the total costs are associated to these districts.

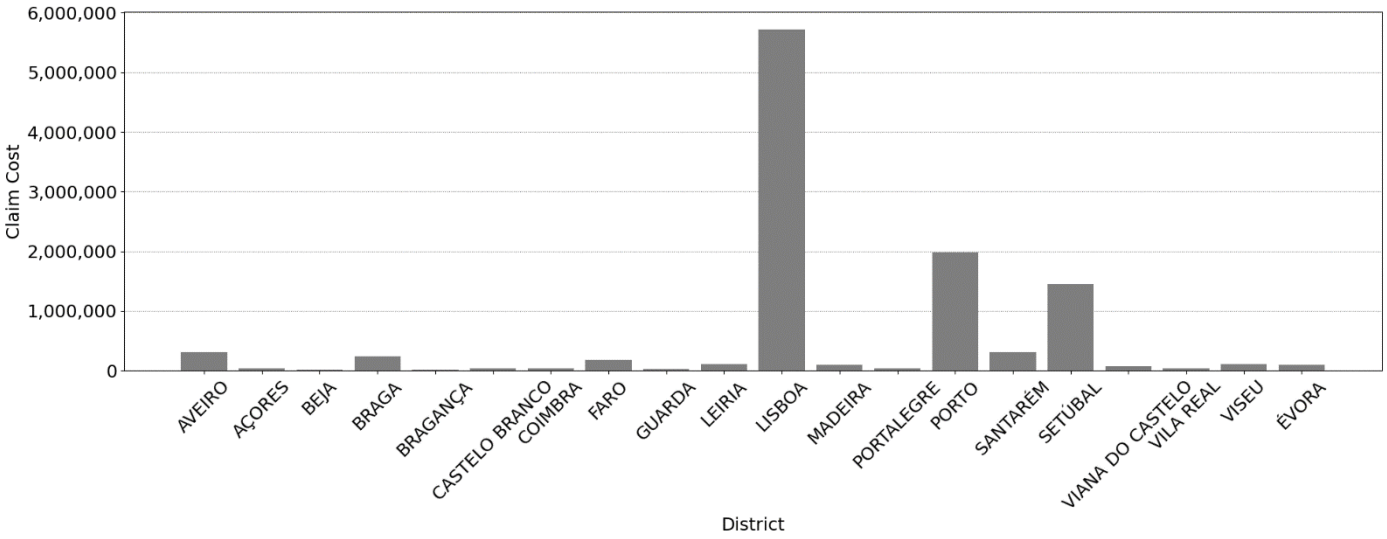


Figure 3.20 - Claim Cost by District

However, the scenario changes when the cost per claim is the metric in evaluation. As Figure 3.21 shows, there is not a significant pattern in the distribution of costs. Castelo Branco is the district with the highest cost per claim, and the only one above 4,000€, with a cost per claim of 4,971€. The remaining fluctuate between 2,162€ and 3,927€, the lowest and second highest cost per claim, respectively.

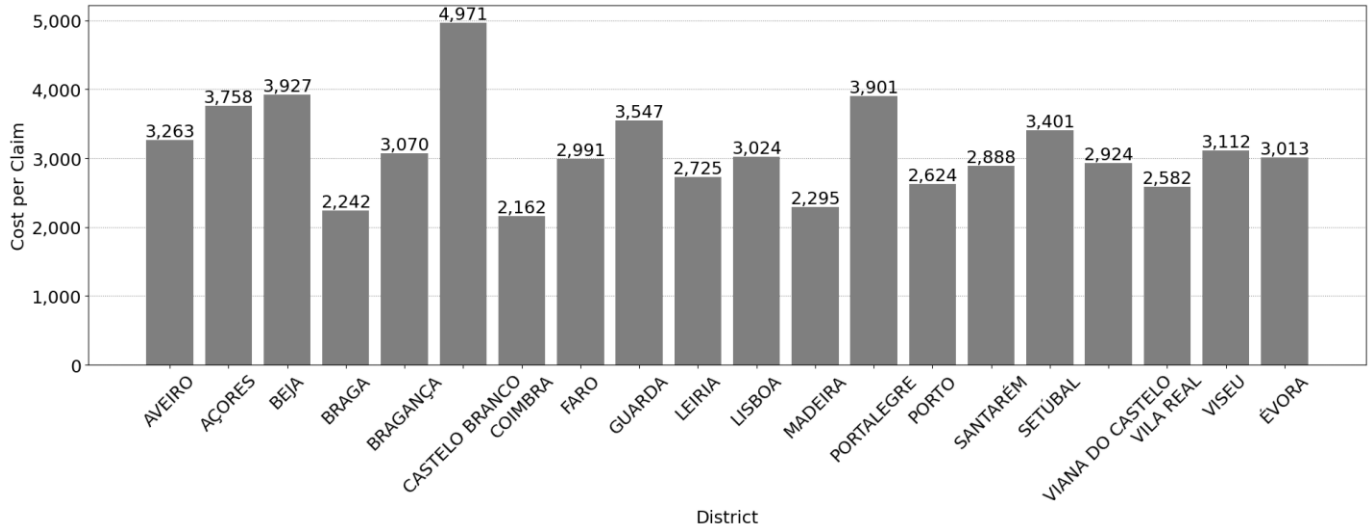


Figure 3.21 - Cost per Claim by District

The distribution of costs by spending limit, in Figure 3.22, is as expected, almost identical to the one shown in Figure 3.12 - Number of Claims by Spending Limit, given that more claims usually mean higher costs. For this reason, it can be more useful to look at the cost per claim, instead of the total costs.

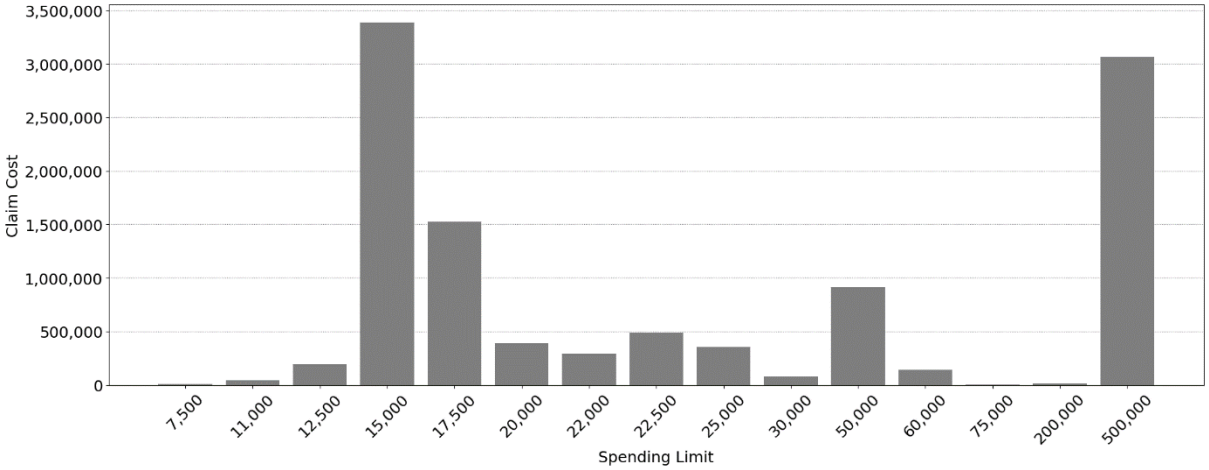


Figure 3.22 - Claim Cost by Spending Limit

The spending limits 30,000€ and 60,000€ have a cost per claim of 6,033€ and 4,010€, respectively, and these values are higher than the aforementioned average of 3,000€. While the spending limit 75,000€ has the lowest cost per claim, amounting to 1,995€ per claim, the remaining ones are very close to the average cost, as Figure 3.23 displays.

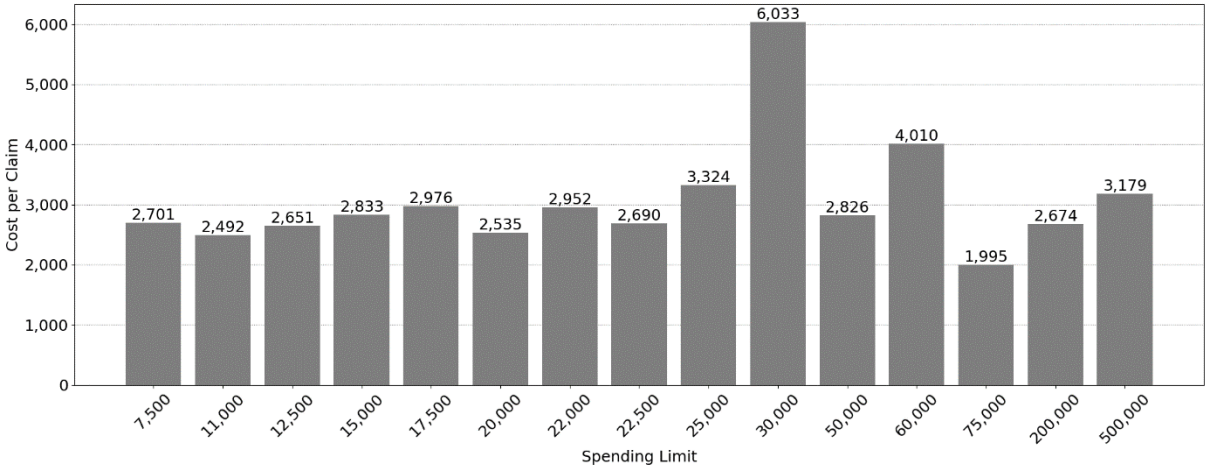


Figure 3.23 - Cost per Claim by Spending Limit

As it was mentioned previously, only 0.2% of insured people have an oncological diagnosis. This translates into having more claims associated with non-oncological patients and, consequently, the majority of costs, totalling around 95% of the total costs.

However, as is shown in Figure 3.15 - Claims per Insured Person by Oncological Diagnosis the number of claims per insured person is higher for oncological patients. The same analysis was done for the costs, in Figure 3.24, and is possible to conclude that the oncological diagnosis leads to bigger spendings, as these people have an average cost of 4,331€ per claim versus the 2,905€ per claim of non-oncological patients.

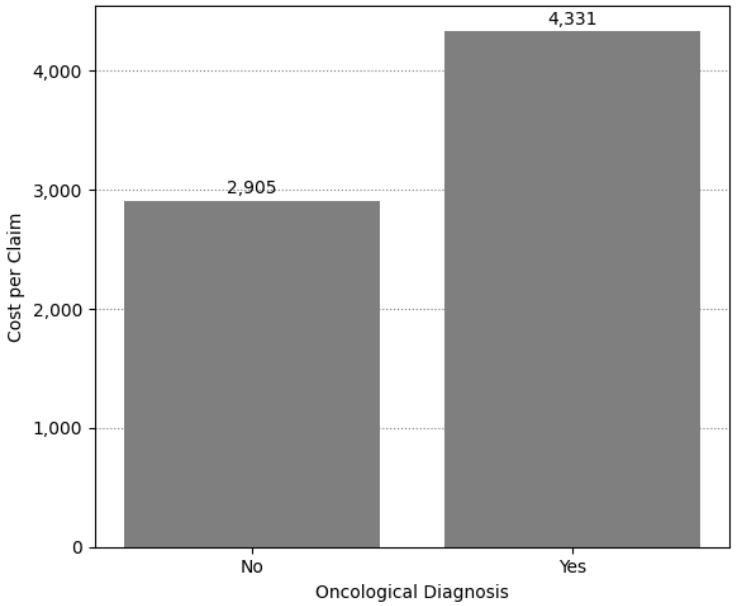


Figure 3.24 - Cost per Claim by Oncological Diagnosis

## 3.3. PRE-PROCESSING

Data is rarely prepared to be immediately used in modelling processes. There is a need to treat it, to be able to draw acceptable and valid conclusions. Pre-processing is a set of processes that are used to clean data and make it model-ready.

### 3.3.1. Missing Values

When an observation does not have a record for a given variable, it is called a missing value. Some models do not accept missing values as input, but even if they did, it is good practice to try and fill the missing values with their real value or a proxy. There are several methods to achieve this goal and some of them will be discussed, and used, further on.

After analysing the missing values of both the Cost and Frequency datasets, it is possible to see that there are some variables with an absurd amount of missing values. When a variable has this amount of missing information, it is not helpful and it is more useful if it is just dropped. For this reason, all variables with over 30% of missing values, except the deductibles on both dataframes and the variables related to outpatient claims on the frequency one, were removed from the datasets.

After removing these variables, the remaining needed to be analysed to determine how to correct the missing values problem: whether by a simple imputation or if there is the possibility of getting the correct value.

#### 3.3.1.1. Deductibles

First of all, the different types of deductibles need to be explained. There is the Minimum, the Maximum and the Normal Deductible. Not all health insurance plans have these types of deductibles but, in the ones that do, the customer pays an amount between the Minimum and the Maximum Deductible.

The Minimum and Maximum Deductibles are monetary variables and represent the boundaries for what the client has to pay, unlike the Normal Deductible, which is a percentage and, as such, ranges between zero and one hundred. It represents the share of the claim cost that the customer has to bear if that value is between the Minimum and Maximum Deductibles.

The first step to calculate the amount of money the client will have to pay is to multiply the cost of the claim by the Normal Deductible. If this value is between the Minimum and Maximum Deductible, that is the amount the customer will pay. In case this value is under the Minimum or over the Maximum Deductible, the client will pay the Minimum Deductible or Maximum Deductible, respectively.

Therefore, if a customer has a normal deductible of 10%, and a minimum and maximum deductible of 200€ and 500€, respectively, it means that the client has to pay 10% of the claim cost if that value is between 200 and 500 euros. Imagining that this client has a claim of 3,000€, they will have to pay 10% of the claim, as 300€ is comprised between 200 and 500 euros. If the amount the client has to pay is inferior to 200€, for example in a claim worth 1,500€, they pay the minimum deductible. Similarly, if the amount to pay exceeds 500€, for instance in a 6,000€ claim, the customer will pay the maximum deductible, considering that this client does not exceed their spending limit.

Some observations did not have information regarding the maximum deductible but, even if it is not specified in the plan, there is always a maximum amount of money that the insurance will afford. This value can be calculated by multiplying the spending limit by the co-pay. This procedure was performed in both cost and frequency datasets.

Additionally, some records have all three deductibles as missing values, because this information is not specified in the insurance plan. To correct this situation, those variables were imputed with the value zero, as there is no deductible.

#### 3.3.1.2. Health Region

In both datasets, there is a variable regarding the health region in which the customer is located. This region is associated, by the SNS, to a person depending on the district they live in, thus the District variable was used to correct these missing values, in the following way:

If a customer lives in:

- Braga, Bragança, Porto, Viana do Castelo or Vila Real, they are assigned to the *Região de Saúde do Norte* (North Health Region);
- Aveiro, Castelo Branco, Coimbra, Guarda, Leiria or Viseu, they are assigned to the *Região de Saúde do Centro* (Centre Health Region);
- Lisbon, Santarém or Setúbal, they are assigned to the *Região de Saúde de Lisboa e Vale do Tejo* (Lisbon and Tagus Valley Health Region);

- Beja, Évora or Portalegre, they are assigned to the *Região de Saúde do Alentejo* (Alentejo Health Region);
- Faro, they are assigned to the *Região de Saúde do Algarve* (Algarve Health Region).

There is not an official health region for the Madeira and Azores archipelagos, nonetheless, the customers who live there were assigned to a fictional Islands Health Region.

#### 3.3.1.3. Outpatient Claims Variables

Except in cases of emergency, the Inpatient claims are usually preceded by some form of medical follow-up, usually in the form of consultations, blood tests, etc. For this reason, it was decided to include this information in the models, in the form of Outpatient claims, divided by its subcategories, in the previous three months. Three variables were created for each subcategory: the total cost of the claims, the total amount of claims and the number of days since the last claim.

The customers that did not have any claims in the previous three months had missing values in all three variables, so an imputation of zero was applied to these features.

#### 3.3.1.4. Other Variables

When analysing the rest of the variables, it was discovered that two of them had incoherent scales and were, therefore, removed. Both variables had been extracted from the SNS Transparency Portal.

The remaining features that still had some missing values, but less than thirty per cent, were corrected by imputing the median of each variable, respectively. The median was chosen over the mean because it is less susceptible to outliers.

### 3.3.2. District

The original District variable holds the client's district of residence, if said client lives in mainland Portugal. If a customer lives in either of the archipelagos, the island of residence is shown in the District variable. Ultimately, this means the District variable has twenty-nine different categories.

To reduce this variable's cardinality, it was decided to group the islands in their respective archipelago, so instead of having eleven islands on the district variable, two from Madeira and nine from the Azores, there will only be two categories, named after each archipelago.

Instead of the twenty-nine original categories, there are now twenty. Nevertheless, there was still the need to reduce this variable's cardinality, as twenty categories are too many. For that reason, it was decided to group the least frequent districts in a single category called Other.

After investigating which districts appear more frequently in the dataset, it was possible to conclude that seventy-five per cent of the clients live in three districts, as Table 3.6 shows. Hence, it was decided to keep those three and merge the remaining. In the end, the District variable was reduced to four categories: Lisbon, Porto, Setúbal and Other.

| District | Clients (Total) | Clients (%) |
|---|---|---|
| Lisbon | 43,734 | 41.0 |
| Porto | 25,059 | 23.5 |
| Setúbal | 11,628 | 10.9 |
| Braga | 4,695 | 4.4 |
| Santarém | 3,660 | 3.4 |
| Aveiro | 3,510 | 3.3 |
| Faro | 2,539 | 2.4 |
| Leiria | 1,945 | 1.8 |
| Coimbra | 1,764 | 1.7 |
| Madeira | 1,490 | 1.4 |
| Viseu | 1,481 | 1.4 |
| Viana do Castelo | 1,126 | 1.1 |
| Évora | 867 | 0.8 |
| Vila Real | 669 | 0.6 |
| Castelo Branco | 634 | 0.6 |
| Portalegre | 505 | 0.5 |
| Beja | 458 | 0.4 |
| Guarda | 319 | 0.3 |
| Azores | 283 | 0.3 |
| Bragança | 242 | 0.2 |

Table 3.6 – Client Distribution by District

### 3.3.3. Unlimited Spending Limits

Some insurance plans have unlimited spending limits, but they still need to be inserted into the company's system. To illustrate this situation, these spending limits are usually represented by a series of nines. In these datasets, the unlimited spending limits are shown as 9,999,999€, 99,999,999€ and 9,999,999,999€.

The highest real spending limit in the dataset is 200,000€, so it is possible to see that the unlimited limits are clear outliers. To correct this situation, but still acknowledging that the customers who have these spending limits can spend much more than the remaining ones, it was decided to change the unlimited spending limits to a dummy value of 500,000€.

### 3.3.4. Inflation

As it is known, money does not have the same value over time and this study spans over the course of three years, from 2017 to 2019. So, to be able to compare the monetary variables between the different years, it was necessary to adjust them to the inflation of the respective year.

Given that the topic of this study is health insurance, the inflation on health services was used, instead of the general inflation.

### 3.3.5. Correlation

Analysing the correlation between variables is a crucial part of the pre-processing stage, since highly correlated features provide us the same information. Having correlated variables in the dataset can produce biased results as well as increasing computational time, making the modelling process significantly more inefficient.

In both cost and frequency datasets, the large amount of variables complicates the manual analysis of which of the correlated variables should be kept and which should be discarded. Thus, to simplify this process, the python package collinearity, and its function SelectNonCollinear, were used.

The collinearity package calculates the Pearson's correlation coefficient between the variables and identifies the pairs above a user-defined threshold, which in this case was 0.7. Simultaneously, the algorithm computes the importance of each feature concerning the target variable, using a univariate approach. After performing both processes, the algorithm selects the most important variable from each set of correlated variables.

After performing this procedure in both datasets, cleansing them of correlated features, the cost dataset ended with thirty-nine variables, out of the original seventy-four, and the frequency dataset went from one hundred and seven features to fifty-nine.

### 3.3.6. Outliers

Outliers or extreme values are observations that, on one or multiple variables, are very distant from the remaining ones. Having these observations in the datasets can skew the results of some models, as they can drastically change the variables' mean and, as such, is best to remove them from the datasets.

To perform this task, the boxplot of each variable was visually analysed and, based on these graphs, the outlier observations were identified.

After pinpointing the outliers on the cost and frequency datasets, they were removed from both data frames. Additionally, it was decided to remove from the frequency dataset all clients whose claims were classified as cost outliers, given that said claims were not considered for the cost model, and could bias the frequency estimations.

## 3.4. MODELLING

### 3.4.1. Train Test Split

When developing a model, the dataset used for the study needs to be split into the train and test datasets. The first one is used to build and train the model that is going to be used to answer the problem. The second one is utilized to test the quality of the model's predictions. The partition in train and test datasets can be made randomly or can be manually defined.

As mentioned previously, this project is focused on the period between 2017 and 2019 and its goal is to use client history to predict the last trimester of their annuity, so as to complete the year and assist in the policy renovation process.

With this goal in mind, the train and test datasets were defined in the following way:

- Train Dataset: every month that the customer stayed in the company, apart from the last three.
- Test Dataset: the last trimester of the client in the company, within the study period.

After defining the train and test datasets for both studies, Cost and Frequency, it became possible to proceed to the modelling stage of the project.

### 3.4.2. Models

The goal of the cost model is to calculate the cost of a claim based on previous claims and the variables mentioned before. The object of this study is money, which is a continuous variable, thus we are before a regression problem.

Several algorithms were tested, and the results they produced will be discussed further on, namely: Linear Regression, Decision Tree Regressor, Random Forest Regressor, Gradient Boosting Regressor and Extreme Gradient Boosting (XGBoost) Regressor.

On the other hand, the main objective of the frequency model is to estimate whether a client will or will not file Inpatient claims and it uses a set of social, economic, and demographic variables related to each customer as input. The output is binary, one if the customer is estimated to file a claim in that month, otherwise zero. Hence, the frequency problem can be categorised as a classification.

The algorithms that were evaluated to estimate the frequency were almost the same as in the study of the cost, but applied to classification problems, instead of regression: Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Gradient Boosting Classifier and Extreme Gradient Boosting (XGBoost) Classifier.

#### 3.4.2.1. Linear and Logistic Regression

Linear Regression is a simple and traditional form of analysing problems where it is assumed that there is a linear relation between the dependent variable, which must be represented on a continuous scale, and the independent variables. The Linear Regression produces a mathematical equation that best

describes the relationship between the variables and also provides estimates of regression variables and a test of significance for each variable (Worster et al., 2007).

When the target variable is on a categorical scale, the Logistic Regression is best suited. The rationale behind the logistic regression is similar to the linear regression in the sense that it creates a model to describe the impact of multiple predictors on a single response variable. But unlike the linear regression, the independent variables in a logistic regression do not need to be linearly related or normally distributed. Since the relation between the predictor and outcome variables is not assumed to be a linear function, the measure of association between the dependent and independent variables is represented by an odds ratio instead of a multiplicative factor. By comparing the odds ratios between the predictor variables, it is possible to determine which factors are of greatest importance, while their statistical significance is indicated by their confidence intervals.

#### 3.4.2.2. Decision Tree

Rokach and Maimon (2005) define a decision tree as an algorithm built through a recursive partition of the instance space, which is an iterative process where the data is split successively into the several branches that the tree creates and consists of three types of nodes: the first one is called root and does not have incoming edges; the rest of the nodes have exactly one incoming edge and are divided in: internal nodes, if they have outgoing edges; and finally there are terminal nodes, more commonly known as leaves.

In a decision tree, the root node is calculated by evaluating which variable provides the best partition of the data, based on a measure that quantifies the improvement of the estimation based on that partition, usually the Gini index or information gain. This process is repeated and at each stage, the splits are evaluated using the previously mentioned metric and the tree chooses the partition that maximizes it. The iterations stop when a node cannot be split any further: when a partition that can change the value of that node's output does not exist, or when splitting the data does not add value to the predictions.

Ultimately, the output for each observation can be calculated by navigating from the root of the tree down to the leaves, according to the outcome of each individual split.

#### 3.4.2.3. Random Forest

A Random Forest, as the name suggests, is a tree-based algorithm that consists of an ensemble of several individual learners, specifically decision trees. The decision tree algorithm can be seen as a weak learner, but by combining several weak learners it is possible to build a strong learner, which has its outputs calculated through a majority vote.

The individual learners are trained using a bootstrapping technique that forms a dataset, by sampling with replacement instances of the original training set, which is called in-bag. The number of examples in this new dataset is the same as the ones on the training dataset, thus it may contain duplicate examples. When using the bootstrapping technique, a part of the training data is kept out of the in-bag sample, usually one-third. This left-over data is known as the out-of-bag data and it is used to obtain an unbiased estimate of the error as trees are added to the forest. It is also used to get estimates of each variable importance (Livingston, 2005).

Afterwards, a random subset of features is chosen for each tree, the random subspace, and they form the nodes and leaves of the individual learners using standard tree-building algorithms. Each tree is grown to the fullest extent possible without pruning.

#### 3.4.2.4. Gradient Boosting and Extreme Gradient Boosting

The Gradient Boosting algorithm can also be classified as an ensemble technique, but instead of relying on a simple averaging of models in the ensemble, it is based on a sequential strategy of ensemble formation, as Natekin and Knoll (2013) state. Unlike the Random Forest, where each tree is built independently, Gradient Boosting is an additive model – it adds new trees to the ensemble sequentially. At each iteration, the new weak tree base learners are trained concerning the error of the whole ensemble learnt thus far.

In Gradient Boosting, the learning procedure consecutively fits new models to provide a more accurate estimate of the response variable, to compensate for the shortcomings of the existing learners. The main idea behind this algorithm is to build the new base learners to be as highly correlated as possible with the negative gradient of the loss function associated with the whole ensemble.

Extreme Gradient Boosting (XGBoost) is also an additive ensemble of decision trees. XGBoost is a machine learning system that can scale tree-boosting algorithms reliably and is optimized for fast parallel tree construction. The biggest advantage of this method is its performance – when compared to other gradient-boosting algorithms, XGBoost is much faster, allowing it to handle larger amounts of data while maintaining good results.

### 3.4.3. Scaling

The variables present in both studies are in very different scales, therefore, to be able to compare them, there is a need to scale them. To achieve this goal, four distinct methods were used: the Standard Scaler; MinMax between zero and one; MinMax between minus one and one; and the Robust Scaler.

The Standard Scaler, or standardization, is calculated by subtracting the mean of the variable from each observation and then dividing it by the standard deviation.

The MinMax method is used to normalize the data in the desired scale. In this case, two scales were chosen: between zero and one and between minus one and one. In the first scenario, the new values are calculated by subtracting the minimum value of that variable from the observation and then dividing by the difference between the maximum and minimum values. In the second instance, the procedure is very similar. The division used to put the observation between zero and one is multiplied by two and then one is subtracted from this product, resulting in all observations being transformed to range between minus one and one.

The Robust Scaler uses the interquartile range to scale the data. In other words, it removes the median and scales the data in the range between the first quartile and the third quartile. Like the previous methods, it also consists of a division, specifically, amongst the difference between the observed value and the first quartile and the difference between the third and first quartiles.

The results of the scaling methods were compared and the best one for each of the different tested algorithms was used.

### 3.4.4. Recursive Feature Elimination

The first algorithms to be tested for both cost and frequency estimations were, respectively, the linear and logistic regressions. Unlike tree-based models, where the algorithm itself chooses the relevant features to make partitions, these models use all given variables to produce results, therefore there can be a lot of unhelpful and useless information that can lead to not having the best results possible, making the models less effective. Hence, there is a need to select the input variables to be considered by these algorithms.

To achieve this goal, the Recursive Feature Elimination (RFE) method was used. RFE selects the features that are more relevant to predict the target variable by fitting the desired machine learning algorithm, in this case, linear and logistic regressions, and ranking the features by importance. After having the importance of each variable, the model is fitted to the N most important features. The number of variables to include is decided by evaluating the r-squared coefficient, in the case of the linear regression, or the f1-score, in the case of the logistic regression, of the model with just the most important feature, then with the two most important features and so on, until having the r-squared or f1-score of the model that includes all variables. When the inclusion of a feature worsens the r-squared or f1-score, the cut-off point is found and only the variables ranked above this point are kept.

### 3.4.5. Hyperparameter Tuning

After deciding on the scaling method, the results of the algorithms can be improved by finetuning its hyperparameters. To perform this process, the function GridSearchCV, from the python package scikit-learn, was initially used, but was later abandoned due to its high running time and optuna was used instead.

Similarly to the grid search, this package trains the desired model with the different hyperparameter combinations specified by the user and then compares the results of all combinations using a user-specified metric, in this case, r-squared for the cost models and f1-score for the frequency ones. This procedure, combined with the scaling, generates the best possible results for these problems, as every step is optimized.

In the case of the decision trees, this optuna hyperparameter tuning is known as pre-pruning and its results were also compared to the post-pruning results, obtained by adjusting the ccp_alpha parameter. Only one of the processes, pre-pruning or post-pruning, is chosen.

In the tables below, it is possible to see the results of the hyperparameter tuning procedure for both models: cost, a regression problem; and frequency, a classification problem.

Table 3.7 shows the parameters that were chosen to tune the two decision trees. Tuning the ccp_alpha parameter proved to be more powerful than the optuna approach, showing that, in this case, post-pruning beats pre-pruning.

| Model | Parameter | Value |
|---|---|---|
| Cost Model | ccp_alpha | 33823.6031 |
| Frequency Model | ccp_alpha | 0.000028019 |

Table 3.7 – Decision Tree Parameters

Regarding the Random Forests, only the Cost Model was tuned, as is shown in Table 3.8. The Frequency Model provided the best results when using the default parameters.

| Model | Parameter | Value |
|---|---|---|
| Cost Model | criterion | absolute_error |
| | n_estimators | 143 |
| | max_depth | 9 |
| | min_samples_split | 5 |
| | min_samples_leaf | 7 |

Table 3.8 – Random Forest Parameters

Table 3.9 and Table 3.10 show the parameter tuning results for both Gradient Boosting and XGBoost algorithms. Unlike the Random Forests, both Cost and Frequency models were more accurate when tuned.

| Model | Parameter | Value |
|---|---|---|
| Cost Model | n_estimators | 80 |
| | loss | lad |
| | criterion | mse |
| | max_depth | 9 |
| | max_features | sqrt |
| Frequency Model | n_estimators | 140 |
| | loss | exponential |
| | criterion | friedman_mse |
| | max_depth | 12 |
| | min_samples_leaf | 1 |
| | max_features | log2 |

Table 3.9 – Gradient Boosting Parameters

| Model | Parameter | Value |
|---|---|---|
| Cost Model | booster | gblinear |
| | max_depth | 6 |
| | n_estimators | 136 |
| | tree_method | exact |
| Frequency Model | booster | gbtree |
| | max_depth | 10 |
| | tree_method | auto |
| | sampling_method | uniform |

Table 3.10 – XGBoost Parameters

### 3.4.6. Oversampling and Undersampling

As mentioned previously, the frequency dataset is very imbalanced, as there are much more people who do not file Inpatient claims. This bias can impact several machine-learning algorithms, even leading some to ignore the minority class entirely. To correct this problem, both oversampling and undersampling techniques were used to resample the training dataset.

The random oversampling procedure consists of duplicating instances of the minority class and can be helpful when using machine-learning algorithms that are affected by an askew distribution or models that rely on good splits of data. On the other hand, random undersampling consists of deleting observations of the majority dataset and is used when the training dataset is very large or when there is a sufficient amount of minority class instances. This process also decreases computational time, as there are fewer observations.

However, these techniques do not come without drawbacks. Both procedures help correct the imbalance problem, but the oversampling can increase computational time and resource use, as well as being possible to provide overfitted estimations. In the case of the undersampling technique, valuable information can be lost, as it deletes random instances of the majority class. For this reason, the results obtained with the resampled dataset should always be compared to those obtained using the original dataset.

Since the frequency dataset is very large, it has over one million records, the undersampling technique is more indicated for this case. However, an oversampling of the minority class was performed before the undersampling, to have a more robust sample. Trying to match the number of instances of the minority class to those of the majority would be inefficient, computationally wise. Thus, the minority class was oversampled until the number of its observations equalled ten per cent of the instances of the majority class. After securing more samples of people with Inpatient claims, the undersampling procedure was then performed by deleting records of people with no claims until it matched the number of observations of the, already oversampled, minority class.

# 4. RESULTS AND DISCUSSION

After dealing with all the problems that presented themselves during the pre-processing stage of the project, it was possible to build the proposed models: Linear and Logistic Regression, Decision Trees, Random Forest, Gradient Boosting and XGBoost.

As mentioned previously, the goal of this project is to estimate the costs associated to client claims in the last three months of their annuity. With this goal in mind, two models were built: the cost model and the frequency model.

## 4.1. COST MODELS

To forecast the costs of the last trimester of the annuity, the models were built on an insured person level. However, since the goal of the project is to forecast at a business level, the costs were grouped by the company that each individual belonged to and only then were the results evaluated. In Table 4.1, it is possible to see how the different algorithms performed in terms of the coefficient of determination ($R^2$), Mean Absolute Error (MAE) and Root Mean Squared Error, as well as the difference between the Real Average Cost and Estimated Average Cost, in the train dataset. In Table 4.2, the same evaluation was made, but this time regarding the test dataset.

| Model | Real Average Cost (€) | Estimated Average Cost (€) | $R^2$ | MAE | RMSE |
|---|---|---|---|---|---|
| Linear Regression | 471,297.44 | 471,297.44 | 0.99961 | 11,328.11 | 14,652.10 |
| Decision Tree | | 471,297.47 | 0.99969 | 9,592.89 | 13,020.11 |
| Random Forest | | 382,292.53 | 0.94823 | 89,004.92 | 169,454.74 |
| Gradient Boosting | | 391,050.76 | 0.95390 | 80,582.79 | 159,911.47 |
| XGBoost | | 471,068.56 | 0.99983 | 6,555.92 | 9,789.46 |

Table 4.1 – Performance of the different Cost models on the Train Dataset

| Model | Real Average Cost (€) | Estimated Avg. Cost (€) | $R^2$ | MAE | RMSE |
|---|---|---|---|---|---|
| Linear Regression | 125,365.15 | 124,951.83 | 0.99656 | 9,094.82 | 13,360.19 |
| Decision Tree | | 124,175.26 | 0.99509 | 9,704.06 | 15,963.99 |
| Random Forest | | 99,595.94 | 0.93751 | 26,665.50 | 56,932.14 |
| Gradient Boosting | | 102,162.35 | 0.94882 | 23,992.51 | 51,524.56 |
| XGBoost | | 125,777.10 | 0.99704 | 9,180.43 | 12,399.24 |

Table 4.2 – Performance of the different Cost models on the Test Dataset

The XGBoost algorithm provides the best results for the cost problem and it will be used in the next stages of forecasting.

## 4.2. FREQUENCY MODELS

Identically to the cost models, the frequency was also estimated on an insured-person level. However, since the goal of the project is to forecast at a business level, the costs were grouped by the company that each individual belongs to and only then were the results evaluated.

In classification problems, the response variable is a number, in this case zero or one, but it is calculated as a probability. This probability is transformed into a number based on a threshold, which is usually 0.5. This means that the model will consider that every observation to which is assigned a probability higher than the threshold will file a claim. Given the complexity of the frequency problem, a new threshold, calculated resorting to the Precision-Recall Curves, was considered for each of the tested models. The new threshold is the value that maximizes the F1-Score and, in Table 4.3, it is possible to see the thresholds considered for each algorithm.

| Algorithm | New Threshold |
|---|---|
| Logistic Regression | 0.959468 |
| Decision Tree | 0.959468 |
| Random Forest | 0.270000 |
| Gradient Boosting | 0.936423 |
| XGBoost | 0.909646 |

Table 4.3 – New Thresholds considered for each Algorithm

In Table 4.4, it is possible to see how the different algorithms performed in terms of Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), as well as the difference between the Real Average Number of Claims and Estimated Average Number of Claims, in the train dataset. Just like in the case of the cost models, the same evaluation was made for the test dataset, shown in Table 4.5.

| Model | Real Average Number of Claims | Estimated Average Number of Claims | MAE | RMSE |
|---|---|---|---|---|
| Logistic Regression | 5,494 | 582 | 4,913 | 9,118 |
| Decision Tree | | 2,589 | 2,905 | 5,429 |
| Random Forest | | 5,991 | 499 | 1,067 |
| Gradient Boosting | | 2,879 | 2,615 | 4,874 |
| XGBoost | | 5,302 | 192 | 417 |

Table 4.4 – Performance of the different Frequency Models on the Train Dataset

| Model | Real Average Number of Claims | Estimated Average Number of Claims | MAE | RMSE |
|---|---|---|---|---|
| Logistic Regression | 38 | 120 | 82 | 217 |
| Decision Tree | | 747 | 708 | 1,670 |
| Random Forest | | 206 | 168 | 449 |
| Gradient Boosting | | 22 | 16 | 30 |
| XGBoost | | 36 | 7 | 11 |

Table 4.5 – Performance of the different Frequency Models on the Test Dataset

The XGBoost algorithm provides the best results for the frequency problem and it will be used in the next stages of forecasting.

### 4.3. FORECAST

The goal of this project is to propose a new model to project claim costs of the last trimester of each client's annuity. These projections are made by using the best-performing algorithms and multiplying the estimate of the cost of each client's claims by the estimate of the number of claims.

To evaluate the performance of this model, it is necessary to compare it to the real claim cost and calculate the error associated with it. This error is calculated as follows:

$$Error\ (\%) = \frac{Predicted\ Reported\ Claims\ -\ Real\ Reported\ Claims}{Real\ Reported\ Claims} \times 100\%$$

It is through the error measure that the company can verify if it is projecting above or below the real cost that it is going to bear. Even if this metric is not as scientific as other statistical error measures, in a real business context, it is very important, since projecting below the real costs will result in the company losing money.

In Table 4.6 it is possible to see how the new proposed model performs, regarding the different clients present in the study. The reported claims values refer to the last trimester of their annuity.

While the new proposed model is not perfect, it shows promising results, with a global error of 9%. Yet, more important than the performance of this model individually, is its performance when compared to the currently in-use baseline model. This comparison is shown in the following section.

## 4.4. Comparison with Baseline Method

At Multicare, this last trimester projection is currently being made resorting to an ARIMA model. However, the ARIMA only looks at past claim history and fails to comprehend how each variable can be important to the forecasting process. It is for this reason that this project is focused on the use of machine learning techniques to answer this problem.

In Table 4.6, it is possible to see how the proposed model compares to the baseline method, the ARIMA, in terms of percentual error.

| Client | Annuity | Real Reported Claims (€) | Predicted Reported Claims (New Model) (€) | Proposed Model Error (%) | Predicted Reported Claims (ARIMA) (€) | ARIMA Error (%) |
|---|---|---|---|---|---|---|
| A | 1 | 11,482.72 | 8,224.41 | -28 | 14,471.59 | 26 |
| B | 1 | 30,392.21 | 24,644.65 | -19 | 43,777.21 | 44 |
| C | 1 | 20,387.14 | 14,289.55 | -30 | 9,141.66 | -55 |
| D | 1 | 36,398.78 | 9,236.20 | -75 | 9,806.83 | -73 |
| E | 1 | 44,907.33 | 19,832.75 | -56 | 38,701.01 | -14 |
| F | 1 | 169,078.32 | 307,453.04 | 82 | 194,732.53 | 15 |
| F | 2 | 145,290.68 | 278,919.83 | 92 | 199,077.90 | 37 |
| G | 1 | 1,401.03 | 4,238.11 | 202 | 4,631.36 | 231 |
| H | 1 | 4,750.02 | 6,583.59 | 39 | 7,643.24 | 61 |
| I | 1 | 101,974.61 | 100,970.84 | -1 | 51,133.27 | -50 |
| J | 1 | 11,050.13 | 60,487.23 | 447 | 3,136.87 | -72 |
| K | 1 | 0.00 | 3,535.54 | - | 1,530.82 | - |
| L | 1 | 21,034.48 | 24,889.70 | 18 | 22,667.38 | 8 |
| M | 1 | 502,742.56 | 497,244.53 | -1 | 323,120.57 | -36 |
| N | 1 | 3,867.15 | 4,337.94 | 12 | 5,277.39 | 36 |
| N | 2 | 9,062.88 | 7,026.39 | -22 | 5,104.48 | -44 |
| O | 1 | 513,712.56 | 394,547.61 | -23 | 287,482.75 | -44 |
| P | 1 | 0.00 | 5,931.15 | - | 4,188.14 | - |
| P | 2 | 6,822.18 | 7,730.09 | 13 | 4,071.90 | -40 |

Table 4.6 – Comparison by Client Between the Proposed Model and the ARIMA baseline

Doing a global analysis, it is possible to conclude that the new model, proposed in this study, is able to recover more than half a million euros, when compared to the ARIMA projections, as it is possible to see in Table 4.7, by subtracting the total claim costs of the ARIMA model to the Proposed Model total claim costs.

| | **Total Claim Costs (€)** | **MAE** | **Difference to Real (€)** | **Difference to Real (%)** |
|---|---|---|---|---|
| Real Reported Claims | 1,634,354.78 | - | - | - |
| Proposed Model Projections | 1,780,123.15 | 28,202.95 | 145,768.37 | 9 |
| ARIMA Model Projections | 1,229,696.90 | 32,950.51 | -404,657.88 | -25 |

Table 4.7 – Global Comparison Between the Proposed Model and the ARIMA

As was already mentioned, the new model projects 9% above the real costs, while the ARIMA model has a global error of -25%. This value shows the true power and potential of the new model, as it forecasts above the real costs, instead of under-pricing clients as the ARIMA does. This means that by using the new model, the company will no longer lose money.

# 5. CONCLUSIONS

As technology and knowledge progress, companies need to stay updated on current and new methods and use them to try and get competitive advantages over their competitors. The renewal of insurance contracts is one of the most important parts of managing Tailor Made clients, therefore the proposals need to be as accurate as possible.

This project was born out of the need to provide more accurate estimations in the first part of the renewal: the forecast of the costs in the last three months of the clients' annuity. To achieve this goal there were two main steps. First, retrieving and preparing the data; second, building models and assessing their performance.

For the first step, two datasets were built: Cost, to estimate the cost of each inpatient claim, and Frequency, to predict how many claims an insured person is going to file in the period of the study. These are the two components of the estimation process and are estimated separately, as they usually evidence different behaviours and can, consequentially, be explained by different variables.

As for the second step, Linear and Logistic Regressions, Decision Trees, Random Forest, Gradient Boosting and XGBoost algorithms were all tested and evaluated to decide which was the best approach to solve the problem in hand.

After having the data ready and the models built, all the procedures were assessed and XGBoost revealed itself to be the best-performing algorithm in both cost and frequency estimation tasks and was thus chosen to be the procedure to apply in this project. This evaluation was done using the traditional metrics of R-Squared ($R^2$), Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).

After comparing the results of the model proposed in this study with the current baseline model, an ARIMA-based model, it was possible to conclude that the new estimation method is much more powerful and accurate. To make this assessment, an error metric, that is widely used in the company, was recurred to and it showed that the ARIMA projections were 25% below the real costs, contrasting with the estimates of the new proposed model, which are 9% above the real costs. This not only makes the predictions more accurate, as it also saves the company money along the process. When considering only the clients in this pilot study, the new model represents a recovery of more than half a million euros.

# 6. LIMITATIONS AND RECOMMENDATIONS FOR FUTURE WORKS

This project represents a new approach to an existing problem and that is one of the reasons why it was defined as a pilot study. Even though the results are promising, only having a set of clients constitutes a limitation. Therefore, the results need to be interpreted with caution.

The second reason why this study was defined as a pilot is the amount of data, particularly the frequency dataset. As mentioned previously, this dataset contains over one million observations, which made the modelling stage of this project very time-consuming. If this project moves forward and is implemented for the entirety of Multicare's clients, there is going to emerge a clear necessity for more computational power.

In terms of future work, the first step is to add the Reimbursement, given that this project, as stated in the introduction, is focused on the claims that happened under the Healthcare Provider Network. In order to contemplate every Inpatient claim and be able to estimate the full costs, this step is of great necessity. Additionally, and since not every reimbursement claim is known immediately, the IBNR (Incurred But Not Reported) costs also need to be taken into consideration. Currently, these costs are estimated using a technique known as Chain Ladder, which is a traditional actuarial technique. As of now, a machine learning-based approach to this problem is under development.

After addressing these problems and assuring they can be correctly implemented, the remaining clients should be added. Additionally, even though that in the case of the Inpatient coverage the variables external to the company, the social-economic and demographic indicators, were not deemed important for the models, this might not be true for the other coverages. Currently, only the insured people present in this pilot study have their address georeferenced, but this is a mandatory step to be able to use these external variables.

The renewal process consists of two parts: predicting the last three months of the annuity and predicting the next annuity, but this project only answers the first step of the renewals. Even though this is already an improvement and can potentially save the company money, it is only half the process and predicting the next annuity is as important to the renewal process as the last three months' projections. Combining both components will allow Multicare to give clients more adjusted insurance rates.

Lastly, and to finalize the redesign of the renewal process, the methodology applied in this project, as well as the future work identified in this section, should be implemented for the remaining coverages. The new renewal system will only be usable by the company once all the identified steps are carried out.

# 7. REFERENCES


Abdelhadi, S., Elbahnasy, K., & Abdelsalam, M. (2020). A proposed model to predict auto insurance claims using machine learning techniques. *Journal of Theoretical and Applied Information Technology*, *98*(22).

Albrecher, H., Bommier, A., Filipović, D., Koch-Medina, P., Loisel, S., & Schmeiser, H. (2019). Insurance: models, digitalization, and data science. *European Actuarial Journal*, *9*(2), 349-360.

Chen, T., & Guestrin, C. (2015). Xgboost: Reliable large-scale tree boosting system. In *Proceedings of the 22nd SIGKDD Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA* (pp. 13-17).

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). New York, NY, USA: ACM.

Fauzan, M. A., & Murfi, H. (2018). The accuracy of XGBoost for insurance claim prediction. *Int. J. Adv. Soft Comput. Appl*, *10*(2).

Guelman, L. (2012). Gradient boosting trees for auto insurance loss cost modeling and prediction. *Expert Systems with Applications*, *39*(3), 3659-3667.

Hanafy, M., & Ming, R. (2021). Machine learning approaches for auto insurance big data. *Risks*, *9*(2), 42.

Harris, C. R., Millman, K. J., van der Walt, S. J. et al. (2020). Array programming with NumPy. *Nature*, *585*, 357–362.

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95.

Inc., P. T. (2015). Collaborative data science. Montreal, QC: Plotly Technologies Inc. Retrieved from https://plot.ly

Livingston, F. (2005). Implementation of Breiman's random forest machine learning algorithm. ECE591Q Machine Learning Journal Paper, 1-13.

McKinney, W. (2010). Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference* (Vol. 445, pp. 51–56).

Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, *7*, 21.

Pedregosa, F., Varoquaux, G., Gramfort, A. et al. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(Oct), 2825–2830.

Pesantez-Narvaez, J., Guillen, M., & Alcañiz, M. (2019). Predicting motor insurance claims using telematics data – XGBoost versus logistic regression. *Risks*, *7*(2), 70.

Rokach, L., & Maimon, O. (2005). Decision trees. In *Data mining and knowledge discovery handbook* (pp. 165-192). Springer, Boston, MA.

Van Rossum, G. (2020). *The Python Library Reference, release 3.8.2*. Python Software Foundation.

Worster, A., Fan, J., & Ismaila, A. (2007). Understanding linear and logistic regression analyses. *Canadian Journal of Emergency Medicine*, *9*(2), 111-113.