TÉCNICO LISBOA

Predicting Frequency and Claims of Health Insurance with Machine Learning Techniques

Pedro Octávio Couto Gonçalves

Thesis to obtain the Master of Science Degree in

Data Science and Engineering

Supervisor(s): Prof. Arlindo Manuel Limede de Oliveira
Prof. Luís Miguel Veiga Vaz Caldas de Oliveira

Examination Committee

Chairperson: Prof. Mário Alexandre Teles de Figueiredo
Advisor: Prof. Arlindo Manuel Limede de Oliveira
Members of the Committee: Prof. Manuel Fernando Cabido Peres Lopes

May 2021

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

"If you want to go fast, go alone. If you want to go far, go together."

Being a believer that none of the great achievements in life are possible to accomplish alone, the least I can do is use this little piece of text to show my appreciation to everyone that, in any way, has contributed for this work to be possible.

I believe that the initial quote of this text is true in every aspect of life, but in the context of developing a master thesis in a company like Multicare, it becomes even more true. That said, my first acknowledgment must be to my company supervisor Filipa Marques for all the valuable insights, guidance, and omnipresence in every stage of the daily work life. Then I want to thank all of my colleagues at Multicare who helped me during this stage: to all my team colleagues, who have also been there to provide help and guidance when needed; to the Actuarial and Control department of Multicare and especially to its director Maria do Carmo Ornelas, for being always open to support academic work within our projects; to the Advanced Analytics department of Fidelidade, especially to José Vieira and Ricardo Caeiro, for the precious help with the data related issues.

I must address a special thanks to my supervisors Professor Arlindo Oliveira and Professor Luis Caldas de Oliveira for the insightful and detailed comments, as well as their honest advice, that were essential to improve this work. The readiness and availability showed by Professor Arlindo Oliveira cannot be understated.

Finally, I want to give recognition to all my family and friends, especially to my parents and sister for all the support and exceptional guidance in all the important decisions throughout my life.

Abstract

In the health insurance industry, policies are typically one year contracts that are renewed after these twelve months. In Multicare, this renewal starts to be negotiated at the end of the first nine months of the current annuity. At this point it is necessary to set a prediction of how the present annuity will end, i.e, there is the need to forecast the loss ratio of the last three months of the annuity considering the loss ratios of the first nine months.

This problem is currently handled using a time series algorithm, ARIMA, that forecasts future loss ratios considering only the past ones and ignoring all other external information that can also prove useful in predicting the behaviors of the insured population, both in terms of frequency of usage of the insurance and in terms of the cost of medical acts.

This study incorporates a wide variety of external variables coming from different sources in the traditional datasets of Multicare and performs a comparison between several types of tree-based machine learning models, aiming to find the ones that lead to better performances in predicting claims and costs of the insured population.

The main contribution of this work is the proposal of a new prediction model for the claims and costs of the insured population of health insurance and its inevitable comparison with the model that is currently in production in Multicare, based on ARIMA time series.

Keywords: machine learning, forecasting, time series, health insurance, loss ratio, tree algorithms

Resumo

No setor segurador de saúde, as apólices são normalmente contratos de um ano que sofrem uma renovação após esse período. Na Multicare, essa renovação começa a ser negociada ao final dos primeiros nove meses da anuidade atual. Neste ponto, é necessário fazer uma previsão de como a anuidade atual irá terminar, ou seja, há a necessidade de se projetar a sinistralidade dos últimos três meses da anuidade, considerando a sinistralidade dos primeiros nove.

Este problema é atualmente tratado, usando algoritmos de séries temporais, ARIMA, que prevê a sinistrlaidade futura, considerando apenas a sinsitralidade passada e, ignorando todas as outras informações externas que também podem ser úteis na previsão do comportamento da população segurada, tanto em termos de frequência de uso do seguro, como em termos de custo dos atos médicos.

Este estudo incorpora uma grande variedade de variáveis externas provenientes de diferentes fontes nos datasets tradicionais da Multicare e realiza uma comparação entre vários tipos de modelos de aprendizagem automática baseados em árvores, com o objetivo de encontrar aqueles que levam a melhores desempenhos na previsão de sinistros e custos da população segurada.

A principal contribuição deste trabalho é a proposta de um novo modelo de previsão dos sinistros e custos da população segurada e sua inevitável comparação com o modelo atualmente em produção na Multicare, baseado em séries temporais ARIMA.

Keywords: aprendizagem automática, previsão, séries temporais, seguro de saúde, sinsitralidade, algoritmos de árvores

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In this chapter, it is presented an introduction to the problem under study. First, a detailed description of the initial motivation is given in section 1.1, followed by an explanation of how the problem is currently being handled in section 1.2 in the literature and finally a thesis outline in section 1.4.

1.1 Motivation

In the health insurance industry and, particularly, in Multicare, by definition, the insurer receives from its clients in advance an amount of premium that can generate future liabilities. Regarding the subscription of a corporate health insurance policy, two key moments have to be taken into account by the pricing actuaries.

The first one concerns the establishment of a fair price at the moment the policy is subscribed. At this moment the insurer has access to a very limited range of information about the client. The available information includes only the age, gender, and EAC (Economic Activity Code) for each insured person. Pricing a client at this moment, having only this type of information, is delicate and forces actuaries to implement creative and precise models to make sure they predict the loss ratio accurately to propose a fair price to the client.

The second key moment happens with an annual periodicity. After each annuity (the twelve month periods in which an insurance policy is active), the contract needs to be renewed. Health insurance contracts in Multicare are mostly one year contracts with optional renewal at the end. At this time the insurer makes a new proposal to the insured client. In this proposal, both the price of the policy and the conditions of the insurance plan can be subjected to changes.

Contrary to what happens in the subscription moment, in the renewal moment the insurer has access to a larger set of information regarding the client. The most obvious one and probably one of the most important is the information about the claims that occurred in the ending annuity. However, looking back at the past behaviors of a corporate client can only help to predict the future ones up to a certain point, since it does not capture any external events that might influence health expenditures if taken into account. Besides past behavior information, Multicare has also at its disposal other sets of geographical and socioeconomic variables, such as client addresses and respective road distances to the health providers, performance indicators of the nearest public providers, among others that may prove useful and relevant in predicting the behaviors of each insured person.

For the present work, the clients that will be priced are all corporate clients and, as a consequence, the mutualization is done within each company.

In Multicare, the process of renewing a contract and predicting the price of the next annuity of a

corporate client is a long taking process with a lot of legal deadlines to follow. The negotiation begins three months from the end of the annuity, where the pricing actuaries have to predict the loss of those last three months and, based on the total loss of that annuity, i.e., the nine real months plus the three predicted ones, set up a price for the next one. This predicts the next annuity depend largely on the behavior each client has in the present one and gives great importance to accurately predicting the last three months' loss since a bad prediction here can compromise the entire next annuity.

The concept of loss ratio is one of the most important indicators in monitoring a corporate client, but, despite the importance of a good prediction of this indicator, it presents a lot of variation and therefore can prove difficult to predict.

$$Loss\ Ratio = \frac{Claims\ Costs}{Total\ Earned\ Premiums} \tag{1.1}$$

It is defined by the ratio between the total costs of the claims and the total earned premiums received by the company. Assuming that the corporate client remains stable, the value of the denominator (Total Earned Premiums) is a known factor. Given that, the variation in the loss ratio comes from the claims costs. The total claims costs, in turn, are defined by:

$$Claims\ Costs = Reported\ Claims + IBNR \tag{1.2}$$

IBNR stands for *Incurred but not reported* and refers to a claim that has already occurred but has not yet been reported (they are always reported after the accounting date). This means that, since the insurer does not know how many of these losses have occurred, this value is always an estimate.

This thesis urges from the difficulty that arises from this nine month loss ratio prediction, which is a preliminary step before predicting the renewals. Nowadays, as it is shown in the following sections of this introduction, the claims predicting is made using a time series algorithm.

1.2 Literature Review

This section describes how the problem presented in the previous section is addressed in the literature.

One of the classical ways for dealing with the present problem is the one suggested by the United Kingdom's *Institute and Faculty of Actuaries* (IFA). In their Claims Reserving Manual, the problem is treated by calculating the loss ratio for each month of the first nine ones of the annuity.[9] So, in this way, the loss ratio for month i (LR_i) would be:

$$LR_i = \frac{CC_i}{TP} \tag{1.3}$$

where CC_i refers to the total of Claims Costs in month i and TP corresponds to the total amount of premiums received by the company for that whole year.

It is clear that, since the numerator of the previous equation is cumulative, the loss ratio increases when i increases.

After having calculated all the nine values of the Loss Ratio ($LR_{1,\dots,9}$), the IFA suggests then fitting a least squares approximation to those values. With this fit, it is possible then to find the values for the loss ratios of the last three months of the annuity ($LR_{10,\dots,12}$)

Another model pointed out by England et al. (2002) to deal with this issue, is the classic Mack's Chain Ladder (CL) method, which is described in greater detail in Section 1.3.1, because it is the one currently used in Multicare to estimate the IBNR claims.[10]

The work of Wuthrich (2018) proposes a modification of the traditional Mack's Chain Ladder.[1] In the

Chain Ladder method, the aggregated claims payments are assumed to fulfill a regression assumption, however, in Wuthrich's paper the goal was to extend this simplified regression assumption by allowing for the inclusion of individual claims information. The individual claims information that was included in the study were:

- the line of business the individual claim is belonging to (LoB);
- the claims code denoting the labor sector the injured is working in (CC);
- the accident quarter of the occurrence of the individual claim (AQ);
- the age of the injured age in years at claims occurrence (age);
- the injured body part (inj. part).

To model this new information neural networks are used. The loss function to be minimized is the weighted square loss function

$$\mathcal{L}_j = \frac{1}{\sigma_{j-1}^2} \sum_{i=1}^{I-j} \sum_{\boldsymbol{x}: C_{i,j-1}(\boldsymbol{x})>0} C_{i,j-1}(\boldsymbol{x}) \left(\frac{C_{i,j}(\boldsymbol{x})}{C_{i,j-1}(\boldsymbol{x})} - f_{j-1}(\boldsymbol{x}) \right)^2 \tag{1.4}$$

where $C_{i,j}$ is the cumulative claims payments for claims with accident year i done within the first j development periods and having feature value x and f_{j-1} are the Chain Ladder factors.

The architecture chosen for the neural network was a feed forward one with only one hidden layer having twenty neurons. This network has the hyperbolic tangent as its activation function. A schematic representation of this architecture is shown below.

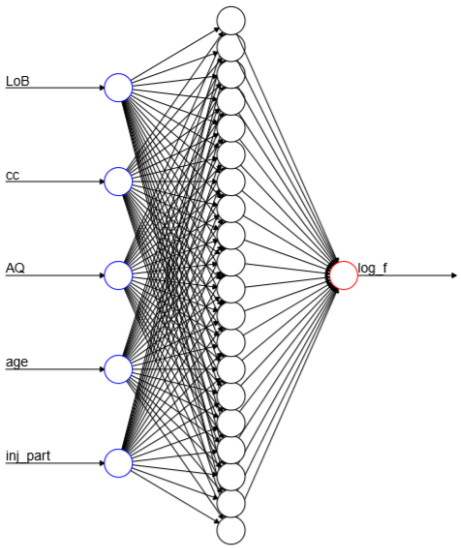

Figure 1.1: Schematic representations of the neural network architecture from Wuthrich (2018).[1]

In this paper, the forecasting process takes into account the difference of the non-zero claims and the zero claims (claims in which $C_{i,j-1}(x) = 0$). Since the data in analysis has a lot of feature combinations for which the total claims are 0 for a given i and j, for the latter a new model that does not take into

account the claims features is proposed. In this model the predicated claims are just given by the formula:

$$C^*_{i,j} \approx C_{i,I-i} \prod_{l=I-i}^{j-1} g_l^{(i)} \tag{1.5}$$

for given $g^{(i)}_{I-i}, \ldots, g^{(i)}_{J-1}$ CL parameters.

At the end of this paper, it is admitted that it is only the start of a broader spectrum of work that can be done incorporating machine learning for the problem of loss ratio forecasting. One next step that the authors suggest that should be considered is the incorporation of dynamic variables (variables that change in time) in the models, instead of only static ones. An attempt to include this type of variables into machine learning models was done by Wuthrich. [2]

In this work, it is proposed a regression tree model. The variables taken into account include static ones, such as claim code (cc) stating the type of claim, diagnosis code (diag) stating the type of injury, the lawyer involved (law), static categorical feature and reporting delay (j), and also dynamic variables like closed at time $i + j + k$ (cl) which indicates if the claim is closed or open at time $i + j + k$ where i is the accident year and $j + k$ are the amount of time it takes until the claim is reported, known as the reporting delay. A closed claim is one in which all compensations are paid. The other dynamic variables are the amount of money paid by the insurer at time $i + j + k$.

The regression functions are estimated using classification and regression tree (CART) methods. It is done using the **rpart** function in R, which successively partitions the feature space into rectangles by solving standardized binary split questions.

This model used different methods to estimate both the Reported Claims and the IBNR.

For the Reported Claims, cross validation is done to find the hyperparameter of the number of leaves, and the cost-complexity plot is done, given an optimal number of leaves of 11.

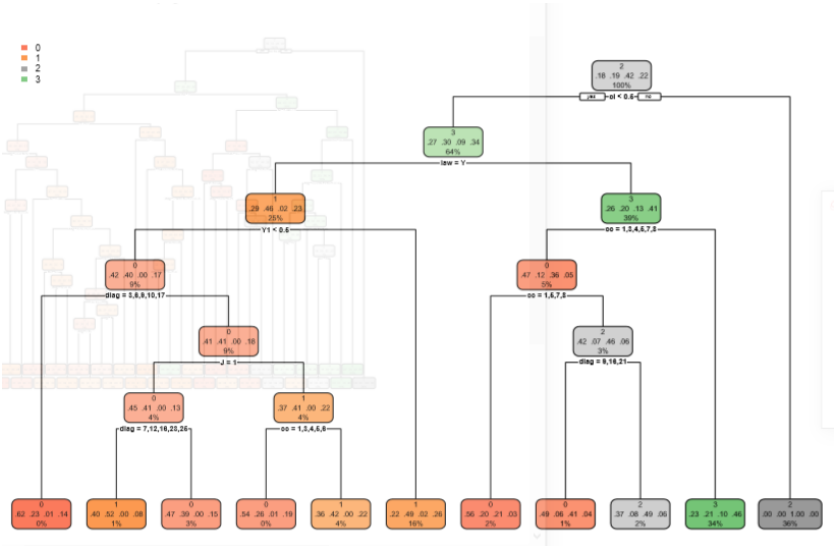

Figure 1.2: Best tree found for forescasting the Reported Claims according to Wuthrich (2018). [2]

For the IBNR claims, the information about the number of reported claims $N_{i,j}$ is not yet observed, thus they had to be predicted. It is assumed that the claims occurrence and reporting process can be described by a homogeneous marked Poisson point process. By doing so, the IBNR claims were calculated resorting once again to the Chain Ladder model.

In the work of Kuo (2019) it is presented a more sophisticated approach to the loss ratio forecasting problem.[3] The authors resort to the use of deep learning, more specifically, neural networks.

The database used in the study refers to various claims from accident years 1988-1997 for a total of fifty companies. The inputs of the neural network developed are the past loss ratios, calculated in two different ways, one with the incremental paid losses and the other one with the total claims outstanding, and also a company code so that it is possible to identify the company.

The architecture of the proposed neural network is the one shown in the figure below.

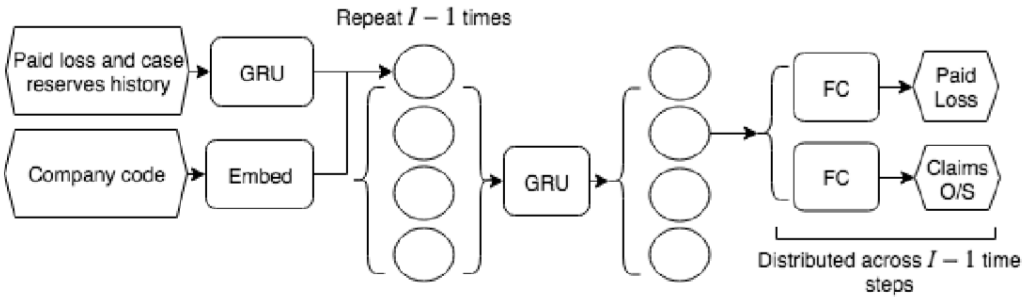

Figure 1.3: Neural Network architecture used by Kuo (2019).[3] GNU refers to Gated Recurrent Unit (a type of recurrent neural network) and FC refers to Fully Connected layer.

For both the encoder and decoder GNU modules 128 hidden units are used and a dropout rate of 0.2. Regarding the company codes input variable, each company is mapped to a fixed length vector in $\mathbb{R}^k$, where k is a hyperparameter. After this, each decoded GNU timestep is concatenated with the company embedded output and enters the fully connected layers. The two subnetworks correspond to the paid loss and case outstanding predictions, respectively, and each consists of a hidden layer of 64 units with a dropout rate of 0.2, followed by an output layer of 1 unit to represent the paid loss or claims outstanding at a time step.

The performance measures used to test this method were the Mean Absolute Percentage Error (MAPE) and the Root Mean Squared Percentage Error (RMSPE). This method performed better than both the traditional Chain Ladder and the Chain Ladder with neural networks.

1.3 Baseline Method

In the following section, the procedures to address the problem shown in the previous sections will be described. These procedures will be the baseline method in our study and at the end, the results of this baseline method will be compared with the ones from the proposed alternative solutions to this problem.

Currently, in Multicare, the problem of predicting the loss ratio in the last three months of each annuity is handled by resorting to time series. But before this, it is mandatory to find a solution to address the estimation of the IBNR claims. In Multicare the method used is the Chain Ladder.

1.3.1 IBNR estimation

The Chain Ladder method is widely used in actuarial science and relies on the assumption that the past loss patterns are indicative of the future ones. To understand how this method works it is important to be familiar with the concept of *run-off* matrices, which are $n \times n$ matrices with the time of occurrence (time in which the claim happened (YYYYMM)) of claims as lines and accounting time (time in which the claim was processed) as columns. Below is an example of a *run-off* matrix like the ones used in Multicare.

accident year	payment delay (in years)									
	0	1	2	3	4	5	6	7	8	9
2004	5,947	9,668	10,564	10,772	10,978	11,041	11,106	11,121	11,132	11,148
2005	6,347	9,593	10,316	10,468	10,536	10,573	10,625	10,637	10,648	
2006	6,269	9,245	10,092	10,355	10,508	10,573	10,627	10,636		
2007	5,863	8,546	9,269	9,459	9,592	9,681	9,724			
2008	5,779	8,524	9,178	9,451	9,682	9,787				
2009	6,185	9,013	9,586	9,831	9,936					
2010	5,600	8,493	9,057	9,282						
2011	5,288	7,728	8,256							
2012	5,291	7,649								
2013	5,676									

Figure 1.4: An example of a *run-off* matrix like the ones used at Multicare. [4]

In the above figure, the elements above the main diagonal correspond to the reported claims. Let $c_{i,j}$ denote the entry of the matrix in column j and line i. Given that, the elements $c_{i,1}$ refer to the values of claims that happened in occurrence time i and were processed in the same month of the occurrence. The elements $c_{i,j}$ where $j > 1$ correspond to claims that happened in occurrence time i and were only processed in the months after.[11]

The elements below the diagonal matrix are all represented as **NA** and they correspond to the values of claims that already occurred but have not been yet reported (IBNR). These are the values that need to be estimated and will be with the Chain Ladder method.

The first step in the Chain Ladder method is calculating the development factors f_k according to the following formula:

$$f_k = \frac{\sum_{i=0}^{n-k} c_{i,k+1}}{\sum_{i=0}^{n-k} c_{i,k}} \ , \ 0 \leq k \leq n-1 \tag{1.6}$$

This way we obtain a development f_k for each column k an we are able to estimate the values below the main diagonal. For each column k we have that the element to be estimated $c_{i,k}$ is given by:

$$c_{i,k} = c_{i,i} \times f_k \ , \ k > i \tag{1.7}$$

where $c_{i,i}$ is the element in the main diagonal of line i. The *run-off* matrix shown below serves for illustrating the method described above.

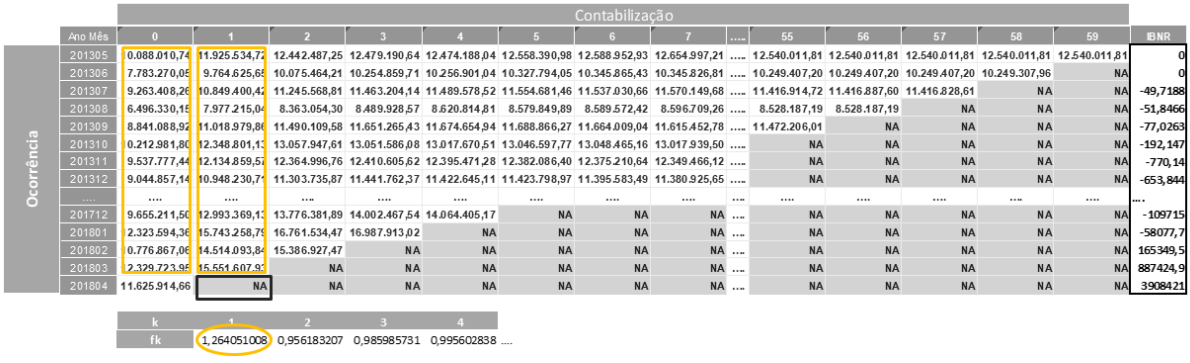

Figure 1.5: An example of a factor development calculation in a *run-off* matrix.

1.3.2 Loss Ratio Estimation

The step that follows the estimation of the IBNR claims is the estimation of the loss ratio for the corporate clients of each renewal date.

Nine months after each contract renewal date, the insurer is in charge of forecasting the loss ratio for the last three months of the annuity based on the past loss ratio (last nine months for a new client and also past annuities for older clients). This forecasting is done using the ARIMA time series model. Below is a definition of time series.

Definition 1.3.2.1. *A time series is a set of observations x_t where each of them is recorded at a given time t. [12]*

To perform time series analysis, the time series data is usually considered as a realization of a stochastic process.

The ARIMA model (autoregressive integrated moving average) is, in fact, a generalization of the ARMA model (autoregressive moving average) that, contrary to ARMA which only models stationary series, can incorporate also a wide variety of non-stationary ones.[13]

Definition 1.3.2.2. *If d is a nonnegative integer, then $\{X_t\}$ is an **ARIMA(p,d,q) process** if $Y_t := (1 - B)^d X_t$ (where B is the backward shift operator) is a casual ARMA(p,q) process.[13]*

To understand the definition of an ARIMA process one must first understand the definition of an ARMA process.

Definition 1.3.2.3. *$\{X_t\}$ is an **ARMA(p,q) process** if $\{X_t\}$ is stationary and if for every t,*

$$X_t - \phi_1 X_{t-1} - ... - \phi_p X_{t-p} = Z_t + \theta_1 Z_{t-1} + ... + \theta_q Z_{t-q} \tag{1.8}$$

where $\{Z_t\} \sim WN(0, \sigma^2)$ and the polynomials $(1 - \phi_1 z - ... - \phi_p z^p)$ and $(1 + \theta_1 z + ... + \theta_q z^q)$ have no common factors.[13]

A loss ratio forecast in Multicare is performed following a set of steps like the ones described below.

- The first step is to calculate the past loss ratio per month from the data displayed in the *run-off* matrices and according to the formula presented in section 1.1
- After having calculated all the loss ratio values per month the next step is identifying any outliers between those values. This is done by resorting to the Grubbs test.

  The Grubbs test is commonly used to find outliers in a univariate data set under the assumption that data are normally distributed. Grubbs test, as shown by its definition, tests outliers one by one.

  Definition 1.3.2.4. *Grubbs's test is defined by the following hypothesis:*

  H_0 *: The data set has no outliers.*

  H_1*: The data set has at least one outlier*

  The Grubbs's test statistic is defined by $G = \frac{max|Y_i - \overline{Y}|}{s}$ where $\overline{Y}$ and s are the sample mean and its standard deviation respectively.[14]

  The maximum and minimum limits above and below which the value is considered an outlier are respectively $\overline{Y} + 3 \times s$ and $\overline{Y} - 3 \times s$.

  The outliers found by the Grubbs Test are then set equal to the smallest or highest (depending on whether the outlier is above or below the interval for which the values are not considered outliers) non-outlier value from the whole loss ratio sample.

- Next in the forecasting process is an important step that is used to verify that the data from the past loss ratios shows evidence of stationarity.

Definition 1.3.2.5. *A time series $\{X_t\}$ is a stationary time series if:*

a) the mean function of $\{X_t\}$, $\mu_X(t) = E[X_t]$ is independent of t,

and,

b) the covariance function of $\{X_t\}$, $\gamma_X(t+h,t) = \mathrm{Cov}\left(X_{t+h}, X_t\right) = E\left[\left(X_{t+h} - \mu_X(t+h)\right)\left(X_t - \mu_X(t)\right)\right]$ is independent of t for each h. [13]

To verify this, two tests are performed, the Augmented Dickey-Fuller test (ADF) and the Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test.

In the KPSS test, the null hypothesis is that the data is stationary around a deterministic trend. [15]

On the other hand, in the ADF test, the null hypothesis is the data having a unit root. [16] A unit root arises when the autoregressive or moving average polynomial of an ARMA model has a root on or near the unit circle. A unit root near 1 of the autoregressive polynomial suggests that the data should be differenced before fitting an ARMA model, whilst a unit root near 1 of the moving average polynomial suggests that the data were overdifferenced.[13]

In a time series, differencing is a method of transforming a non-stationary time series to make it stationary. [17] It consists of subtracting consecutive observations.

$$X_t^{'} = X_t - X_{t-1} \tag{1.9}$$

The computations of this differencing are intended to stabilize the mean of the time series, by eliminating trends. Sometimes computing only the first order difference might not be enough to achieve this, so differencing of higher orders can also be computed.

$$X_t^{(n)} = X_t^{(n-1)} - X_{t-1}^{(n-1)} \tag{1.10}$$

Differencing can also be computed to eliminate the seasonality, which means differencing between an observation and the corresponding observation in the previous season.

$$X_t^{'} = X_t - X_{t-m} \tag{1.11}$$

where m is the duration of the season.[18]

In the ADF test one value that can also alert to the presence of stationarity or not in the data is the ADF statistic, which is a negative number, and the more negative it is, the stronger the rejection of the null hypothesis.[13]

- After being more confident about the stationarity of the data the ARIMA is computed resorting to the R function *auto.arima*, that returns the best ARIMA model according to AIC values. The function searches for all possible models within the order constraints provided. The order (d parameter of the ARIMA model) provided to the function is $d = 0$.

1.4 Objective

Despite the developments of the prediction algorithms in recent years, the problems of predicting accurately both IBNR claims and the loss ratio of the ending months of an insurance annuity are still popular subjects inside the non-life insurance community.

This subject becomes even of greater importance when in the context of health insurance and when dealing with corporate clients, since the loss of a corporate insurance policy means, in most cases, the loss of a significant amount of individual clients and consequently, depending on the client size, the loss of market share.

The main objective of this work is to find a procedure that allows the insurer more accurate predictions to propose a fairer price to each corporate client. Since predicting loss ratios means predicting the amount of claims, it is accepted that the amount of claims might not be very accurately forecasted taking only into account the past values of those claims. Keeping that in mind, in this work some external socio-economic and geographical variables will be introduced in the prediction models. Also, it will only focus on the outpatient coverage considering only claims that happened in our network of providers (i.e. excluding the reimbursement ones) because, from the ones that have a behavioral component, the outpatient coverage is the one that has the greatest financial impact on a renewal.

Chapter 2

Method

In this chapter, we explain more deeply the proposed theoretical methods used for dealing with the forecasting of the loss ratio for corporate clients and detail in a more theoretical way the algorithms used in the model and how they interconnect between themselves.

The baseline method currently in use in Multicare, as explained in the previous chapter, uses time-series algorithms (namely ARIMA) to forecast future loss ratios based on the previous ones. This means that the past loss ratio is the only information that the current model uses to forecast future ones.

The question we ask ourselves at the beginning of this work is: *Is the past loss ratio the only variable relevant for predicting the future loss ratio or can other information that we, as a company, have regarding our clients be also relevant?*

Given this, we propose in this work an approach using machine learning techniques to predict the future loss ratio for corporate clients.

The method proposed will consist of two different forecasting problems. The first one is predicting the number of claims per insured person in the last three months of one annuity inside each corporate client company, whilst the second one consists in predicting the cost of each medical act performed by each insured person in the last three months of each annuity.

2.1 Cost and Number of Claims split

In section 1 it was stated that the loss ratio was calculated as a ratio between the total amount of money paid in claims for one corporate client in one annuity and the total amount of premiums paid by that client in the same annuity.

$$Loss\ Ratio = \frac{Claims\ Costs}{Total\ Earned\ Premiums} \tag{2.1}$$

Since the total amount of premiums (amount of money paid by a client to the insurer in exchange for an insurance policy) is well known, we are interested in forecasting the total costs with claims which are given the following formula:

$$Claims\ Costs = Reported\ Claims + IBNR \tag{2.2}$$

As we showed in the previous section, the forecast for IBNR is made separately from the Reported Claims one, and optimizing them will not be a subject of this work for the simple reason that the problem of IBNR only appears when dealing with reimbursement claims and in this work, we will only deal with claims that occurred within the net of providers of Multicare. Given this, our focus will turn only into

the total amount of reported claims. This one is calculated by multiplying the medium cost of a claim (*Medium Cost*) by the total amount of claims performed by one corporate client in each annuity (*Total Number Claims*).

$$Reported\ Claims = Medium\ Cost \times Total\ Number\ Claims \tag{2.3}$$

Following the previous formula, it becomes obvious that to have the *Reported Claims* value, we must first predict the *Cost* of each claim and compute the mean value over all claims in the dataset and also the *Total Number of Claims*, which can be obtained by predicting the number of claims each insured person will perform in the last three months of each annuity and summing over all insured persons.

2.2 Forecasting Costs and Number of Claims

The number of claims is a variable that takes positive integer values, meaning that computing its prediction is a classification problem. Therefore we will test three different tree-based classifiers for predicting it and compare its performances.

- Decision Tree Classifier
- Random Forest Classifier
- Gradient Boosting Classifier

The approach that will be taken to forecast the **cost** is predicting the cost of each medical act performed by each insured person in the database and compute its mean value.

The **cost of a claim** is simply the amount of money (in euros) requested by the health care provider for each medical act. This variable is typically continuous, therefore its prediction is typically a regression problem.

Three regression algorithms will be tested in this work:

- Decision Tree Regressor
- XGBoost Regressor
- Random Forest Regressor

2.2.1 Models

Decision Tree

A classification tree is built through an iterative process of splitting the data into partitions again and again recursively on each of the branches created, known as **recursive partitioning**.

This **recursive partitioning** works as follows. It starts with a tree with only one leaf, called the **root**. Then, to this leaf, it is assigned a label according to a majority vote among all labels over the training set. After this, it is performed a series of iterations. On each iteration, we examine the effect of splitting a single leaf. We define some “gain” measure that quantifies the improvement due to this split. Then, among all possible splits, we either choose the one that maximizes the gain and perform it or choose not to split the leaf at all. [19]

The recursion is completed when the subset at a node has all the same values of the target variable, or when splitting no longer adds value to the predictions.

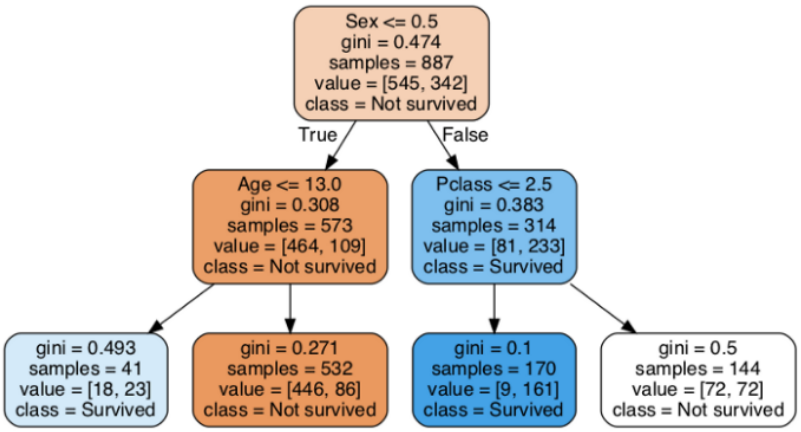

Figure 2.1: An example of a decision tree for classification. [5]

Decision tree classifiers work top-down to find the most adequate variables to split in each node. How they choose the most adequate one depends on what type of metric is used. Different types of metrics typically include the **gini impurity** or the **information gain.**

The **gini impurity** measures how often a randomly chosen element from the set would be incorrectly labeled if it was randomly labeled according to the distribution of labels in the subset. [20]

Let p_i be the fraction of items labeled with class i and consider a set of items with N classes, then the **gini impurity** can be computed as:

$$I_G = 1 - \sum_{i=1}^{N} p_i \tag{2.4}$$

The **information gain** measures the difference between the entropy of the label before and after the split and is used to decide which feature to split on each step. It is based on the concept of entropy, since it is calculated as follows:

$$IG(T, a) = H(T) - H(T|a) = \sum_{i=1}^{N} p_i \cdot log_2(p_i) - \sum_{i=1}^{N} Pr(i|a) \cdot log_2(Pr(i|a)) \tag{2.5}$$

where $H(T)$ is the entropy of the parent node and $H(T|a)$ is the sum of the entropy of the children nodes and p_i represent the percentages of each class $1, ..., N$ present in the child nodes.

To avoid the trees growing in such a way that each observation occupies its node, **stopping criteria** are needed when training a decision tree. The resulting tree would be computationally expensive, difficult to interpret, and would probably not work very well with new data. In the diagram below, the dotted curve represents a decision boundary that accurately separates two classes in an example of training data. For this case, a diagonal red line is probably a better decision boundary for new cases.

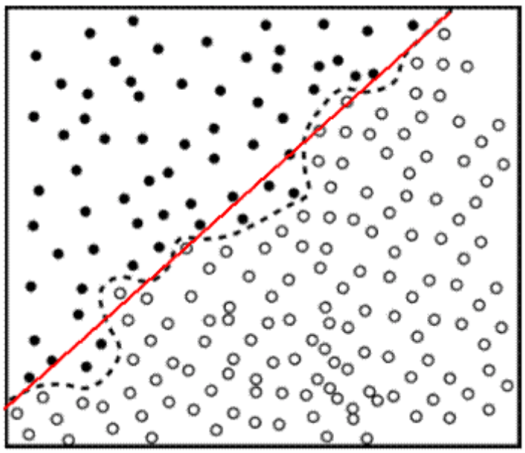

Figure 2.2: An example of a decision boundary between two classes. [6]

The stopping criteria used by decision trees are typically:

- Number of cases in the node is less than some pre-specified limit.
- Purity of the node is more than some pre-specified limit.
- Depth of the node is more than some pre-specified limit.
- Predictor values for all records are identical in which no rule could be generated to split them.[6]

The trees that result from this algorithm are usually very large. For this there are normally two different solutions, one is to limit the number of iterations, leading to a tree with a bounded number of nodes. Another is to **prune** the tree after it is built, hoping to reduce it to a much smaller tree, but still with a similar empirical error. [19]

The decision tree regression algorithm is similar to the decision tree classifier one, but the *True/False* question in the nodes is done resorting to thresholds.

Threshold values have to be estimated during the tree-growing process, usually by exhaustive search. All threshold values are considered for each feature and the one that leads to the best impurity is selected. [7]

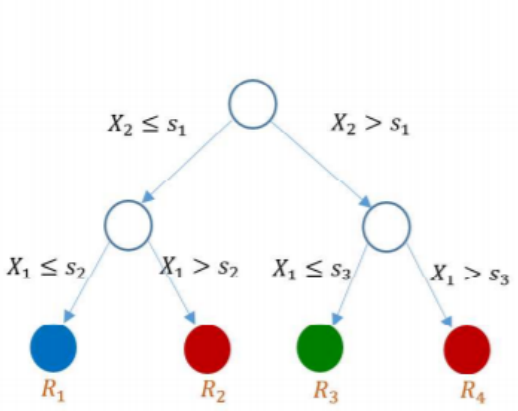

Figure 2.3: An example of a decision tree for a continuous response variable. [7]

Random Forest

The Random Forest is an algorithm based on an ensemble of decision trees trained resorting to a technique called **bagging**. The main premise for this algorithm is that training a small decision tree with few features is computationally cheap, therefore, if we can build several weak decision tree learners in parallel and then combine them by averaging or majority vote we can build a single and strong learner.

The **bagging** method works by taking a training set T and generate N training sets T_i by bootstrap, i.e., by sampling T with replacement, then training a classifier from each set T_i, computing the *a posteriori* distributions $[P_i(y=0|x), ..., P_i(y=K-1|x)]$ and then aggregating all the estimates:

$$\hat{P}(y=k|x) = \frac{1}{N}\sum_{i=1}^{N} P_i(y=k|x) \tag{2.6}$$

In the random forest algorithm, a subset of features is randomly selected at each node and only those features are candidates for splitting features. This method is known as **random subspace**. [7]

Below is illustrated the pseudocode for the random forest training algorithm.

Algorithm 1: Random Forest [21]

Precondition: A training set $S := (x_1, y_1), \ldots, (x_n, y_n)$, features F, and number of trees in forest B.

function RANDOMFOREST (S, F)
  $H \leftarrow \emptyset$
  for $i \in 1, \ldots, B$ do
    $S^{(i)} \leftarrow$ A bootstrap sample from S
    $h_i \leftarrow$ RANDOMIZEDTREELEARN $\left(S^{(i)}, F\right)$
  $H \leftarrow H \cup \{h_i\}$
  end for
  return H
end function
function RANDOMIZEDTREELEARN (S, F)
  At each node:
  $f \leftarrow\leftarrow$ very small subset of F
  Split on best feature in f
return The learned tree
end function

During the **bagging** method about one-third of the cases are left out of the sample. This **out-of-bag** (OOB) data is used to get a running unbiased estimate of the classification error as trees are added to the forest. It is also used to get estimates of **variable importances**.

Along with this work, random forests will also be used several times to measure **variable importances**. To measure the importance for variable X_i the idea is to permute all values of this variable, and measure variable importance by computing the difference in prediction accuracy caused by the permutation.

The variable importance is computed in the following way. Let $\mathcal{B}^t$ denote the out-of-bag samples for a tree t and let $L\left(T_t\left(\mathbf{x}_i\right), y_i\right)$ denote the prediction accuracy at the ith training example. The importance for variable X_j in tree t is defined as

$$VI^{(t)}\left(X_j\right) = \sum_{i \in \mathcal{B}^t} L\left(T_t\left(\mathbf{x}_i\right), y_i\right) - L\left(T_t\left(\mathbf{x}_{i,\pi_j}\right), y_i\right) \tag{2.7}$$

where $\mathrm{x}_{i,\pi_j} = \left(\mathbf{x}_{i,1}, .., \mathbf{x}_{\pi_j(i),j}, \mathbf{x}_{i,j+1}.., \mathbf{x}_{i,p}\right)$, and where π_j is a random permutation of n integers. In classification settings the prediction accuracy $L\left(T_t\left(\mathbf{x}_i\right), y_i\right)$ is defined as $\mathrm{L}(T_t\left(\mathbf{x}_i\right), y_i) = \frac{\sum_{i \in \mathcal{B}^t} I\left(\hat{y}_i^t = y_i\right)}{|\mathcal{B}^t|}$ where $\hat{y}_i^t = T_t\left(\mathbf{x}_i\right)$ denotes the prediction at point $\mathbf{x}_i$ by tree t, and $I(\cdot)$ denotes the indicator function. The variable importance measure for variable X_j is computed as the sum of the importances over all trees in the forest,

$$VI\left(X_j\right) = \frac{\sum_{t \in B} VI^{(t)}\left(X_j\right)}{n} \tag{2.8}$$

where n denotes the total number of trees. [22]

Gradient Boosting

Gradient Boosting, like the Random Forest, is also an ensemble of decision trees, but with two main differences. Gradient Boosting is an additive model, meaning that the trees are built differently, instead of building each tree independently, it builds one tree at a time. Now we take a more theoretical look at how a Gradient Boosting classifier works, as described by Friedman. [23]

Considering a problem of function estimation in the classical supervised learning approach and a dataset $(x, y)_{i=1}^N$, where $x = (x_1, \ldots, x_d)$ are the explanatory variables and y the response variable. The goal is to estimate the function f that transforms x into y, $x \to y$. Let $\widehat{f}(x)$ represent our estimate of f, we want to minimize a specified loss function $\Psi(y, f)$.

$$\widehat{f}(x) = y \tag{2.9}$$

$$\widehat{f}(x) = \arg\min_{f(x)} \Psi(y, f(x)) \tag{2.10}$$

Writing the previous equation in terms of expected values, we want to minimize the expected value of the loss function conditioned on the explanatory variables x.

$$\widehat{f}(x) = \arg\min_{f(x)} \underbrace{E_x[\overbrace{E_y(\Psi[y, f(x)])}^{\text{expected y loss}} \mid x]}_{\text{expectation over the whole dataset}} \tag{2.11}$$

For the problem of function estimating to be tractable, we can restrict the function search space to a parametric family of functions $f(x, \theta)$. This transforms the problem of estimating f into the following.

$$\widehat{f}(x) = f(x, \widehat{\theta}) \tag{2.12}$$

$$\widehat{\theta} = \arg\min_{\theta} E_x\left[E_y(\Psi[y, f(x, \theta)]) \mid x\right] \tag{2.13}$$

To perform parameter estimation normally iterative processes are used, with the simplest and most frequent one being the **steepest gradient descent**.

Given N data points $(x, y)_{i=1}^N$ the goal is to decrease the loss function $J(\theta)$ over the observed data:

$$J(\theta) = \sum_{i=1}^{N} \Psi\left(y_i, f\left(x_i, \widehat{\theta}\right)\right) \tag{2.14}$$

The **steepest gradient descent** is based on consecutive improvements along the direction of the gradient of the loss function $\nabla J(\theta)$ and it is organized as follows:

- Initialize the parameter estimates $\widehat{\theta}_0$. For each iteration t, repeat:

- Obtain a compiled parameter estimate $\widehat{\theta}^t$ from all of the previous iterations:

$$\widehat{\theta}^t = \sum_{i=0}^{t-1} \widehat{\theta}_i \tag{2.15}$$

- Evaluate the gradient of the loss function $\nabla J(\theta)$, given the obtained parameter estimates of the ensemble:

$$\nabla J(\theta) = \{\nabla J\left(\theta_i\right)\} = \left[\frac{\partial J(\theta)}{\partial J\left(\theta_i\right)}\right]_{\theta=\widehat{\theta}^t} \tag{2.16}$$

- Calculate the new incremental parameter estimate $\widehat{\theta}_t$:

$$\widehat{\theta}_t \leftarrow -\nabla J(\theta) \tag{2.17}$$

- Add the new estimate $\widehat{\theta}_t$ to the ensemble. [24]

This additive model works in a forward stage-wise manner, introducing a base learner to improve the shortcomings of existing weak learners. That is, we parameterize the function estimate $\widehat{f}$ in the additive functional form:

$$\widehat{f}(x) = \widehat{f}^M(x) = \sum_{i=0}^{M} \widehat{f}_i(x) \tag{2.18}$$

with M being the number of iterations, $\widehat{f}_0$ the initial guess and $\left\{\widehat{f}_i\right\}_{i=1}^{M}$ the function increments, also called as *boosts*.

In order to make the estimate of the functional more feasible **base learner** functions $h(x, \theta)$ are introduced. Using this we get the optimization rule defined as:

$$\widehat{f}_t \leftarrow \widehat{f}_{t-1} + \rho_t h\left(x, \theta_t\right) \tag{2.19}$$

$$\left(\rho_t, \theta_t\right) = \arg\min_{\rho,\theta} \sum_{i=1}^{N} \Psi\left(y_i, \widehat{f}_{t-1}\right) + \rho h\left(x_i, \theta\right) \tag{2.20}$$

Another difference stands in the way the results are combined, while in the random forest they are combined at the end of the process, usually by majority rules or averaging, in gradient boosting it combines the results along the way. [25]

To summarize this approach we show below the pseudocode algorithm for the Gradient Boosting Machines.

Algorithm 2: Gradient Boosting Machine [24]

Inputs:

– input data $(x, y)_{i=1}^{N}$

– number of iterations M

– choice of the loss function $\Psi(y, f)$

– choice of the base learner model $h(x, \theta)$

Algorithm :

1 : initialize $\widehat{f_0}$ with a constant

2 : **for** $t = 1$ to M do:

3 : compute the negative gradient $g_t(x)$

4 : fit a new base-learner function $h\left(x, \theta_t\right)$

5 : find the best gradient descent step-size ρ_t $\rho_t = \arg\min_{\rho} \sum_{i=1}^{N} \Psi\left[y_i, \widehat{f}_{t-1}\left(x_i\right) + \rho h\left(x_i, \theta_t\right)\right]$

6 : update the function estimate: $\widehat{f_t} \leftarrow \widehat{f}_{t-1} + \rho_t h\left(x, \theta_t\right)$

7 : **end for**

XGBoost

Extreme Gradient Boost (XGBoost) is an additive ensemble of decision trees that is composed of several base learners (decision trees).

XGBoost is a reliable and distributed machine learning system to scale up **tree boosting** algorithms. The system is optimized for **fast parallel** tree construction. [26]

Given a dataset with n observations and m features $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}$ $(|\mathcal{D}| = n, \mathbf{x}_i \in R^m)$, a tree ensemble resorts to K additive functions to forecast the output.

$$\hat{y}_i = \phi\left(\mathbf{x}_i\right) = \sum_{k=1}^{K} f_k\left(\mathbf{x}_i\right), \quad f_k \in \mathcal{F} \tag{2.21}$$

where $\mathcal{F} = \left\{f(x) = w_{q(x)}\right\}$ $(q : R^m \rightarrow T, w \in R^m)$ represents the space of regression trees. Here q represents the structure of each tree that maps an example to the leaf index and w is the weight vector of each leaf.

To learn a tree ensemble, the model tries to optimize the regularized objective function

$$\mathcal{L}(\phi) = \sum_{i} l\left(\hat{y}_i, y_i\right) + \sum_{k} \Omega\left(f_k\right), \quad \Omega(f) = \gamma T + \lambda \|w\|^2 \tag{2.22}$$

where l represents a differentiable convex loss function and Ω a function to measure the complexity of the model and avoid overfitting. The model is then trained in an additive manner, with a new tree being added at each iteration. We can derive a score to measure the quality of a given tree structure q

$$\tilde{\mathcal{L}}^{(t)}(q) = -\frac{1}{2} \sum_{j=1}^{T} \frac{\left(\sum_{i \in I_j} g_i\right)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T \tag{2.23}$$

where $g_i = \partial_{\hat{y}^{(t-1)}} l\left(y_i, \hat{y}^{(t-1)}\right)$ and $h_i = \partial^2_{\hat{y}^{(t-1)}} l\left(y_i, \hat{y}^{(t-1)}\right)$ are the gradient and second order gradient statistics respectively. This score is similar to the impurity score for evaluating decision trees, except that it is derived for wider range of objective functions. [26]

The biggest advantage of the XGBoost is the fact that it is very fast to converge when compared to other gradient boosting algorithms.

Chapter 3

Data Analysis

In this chapter, we dive into the more practical part of this work. It addresses the setup that made our empirical work possible. We discuss the datasets that supported it together with all the cleaning work that was done on them.

We will also explain in more detail the experiments that led to the final results.

3.1 Datasets

In the previous section, the method we will implement in this work is explained in greater detail and consists of forecasting two separate quantities, the number of claims and the cost of claims. For this reason, the data set we will be using in this work is also divided into two.

The first one consists of data containing information about the number of claims per insured person per month of each annuity. Here, the idea of organizing the data by month emerged because in insurance analysis it is essential to have a time unit to measure the **risk exposure** (the measure of potential future loss resulting from a specific activity or event, that, in the insurance industry, are claims) and also to capture seasonality.

The second one has data related to the total cost of the claims per insured person.

3.1.1 Claim Catalogs

In this section, before diving into a more detailed analysis of the datasets, we start by introducing a particular variable that is present in our datasets and that can reveal itself as a very important one further in our analysis.

This variable is called CATALOG and characterizes a claim, indicating its respective **claim catalog**.

The concept of **claim catalog** was created to group the claims by their medical similarity. The goal was to have a variable that could provide a high-level description of each claim. To illustrate this, we show a table below containing four records of claims taken from our dataset. Here we only display three columns, the one indicating that these are outpatient claims, and then the description of the claim and the respective **catalog**, showing how much more high level is the CATALOG variable.

Table 3.1: Excerpt taken from our dataset with an example of four records of claims showing the comparison between the claim description registered in the systems by the provider and the variable indicating the respective **catalog**.

Type of cover	**Claim Description**	**CATALOG**
Outpatient	Medical Assistance	Other Claims
Outpatient	Aspartate transaminase (AST) = GOT	Clinical Analysis
Outpatient	Permanent Medical Care	Emergency Appointments
Outpatient	Abdominal - 2 views+	X-Rays

To assign each of the thousands of descriptions present in our datasets to a suitable **catalog** we had the help of a team of medical doctors of the company. In our datasets we have ten different **catalogs**:

- Medical Appointments
- Emergency Appointments
- Clinical Analysis
- Pathological Anatomy
- Ultrasounds
- Physical and Rehabilitation Medicine
- X-Rays
- MRI
- Computed Tomography
- Other Catalogs

3.1.2 Number of Claims dataset characterization

This database has one line per insured person per month of each annuity, meaning that each insured person appears in the database 12 times in each annuity.

The number of claims database has 989625 lines and 107 variables. The variables are divided into two main groups. A group of variables at the insured person level includes age, gender, the company they work for, and residence addresses, and another group that includes information about the health, socio-economic and demographic conditions in the parish of residence of each insured person. These variables include indicators of the quality of the public health institutions that are closer to the insured person's residence and the real estate prices in the parish of residence. It also includes variables with the distances (in kilometers) and journey times from the residence to the closest public and private health providers by road (considering the shortest road path between the two points).

Table 3.2: Summary table of the variables that were introduced in our datasets resorting to external entities.

Socio-Economical / Demographic		**Health**	
INE	Information about: Residents by Education Residents by employability Retired Residents (data by statistical subsection)	**Transparency Portal (Portuguese National Health Service)**	Information about: Responsiveness of the public health Oncological screenings Nº of appointments Nº of surgeries Nº of users and user rates (data by ACES or Hospital)
Real Estate	Information about: Medium price offer by m2 Medium price transaction by m2 Medium rent contracted by m2 (data by parish)	**Private Entities**	Information about: Responsiveness of private health
Road Network	Distances and times by road between two points		

In the table above we have a summary of the variables that are external to the company data and were introduced in our database.

In the variables that come from INE (Portuguese National Institute of Statistics), the data are shown by **statistical subsection**. The **statistical subsection** corresponds to the 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. In the image below we can see an example of a portion of the map of Portugal divided by a statistical subsection.

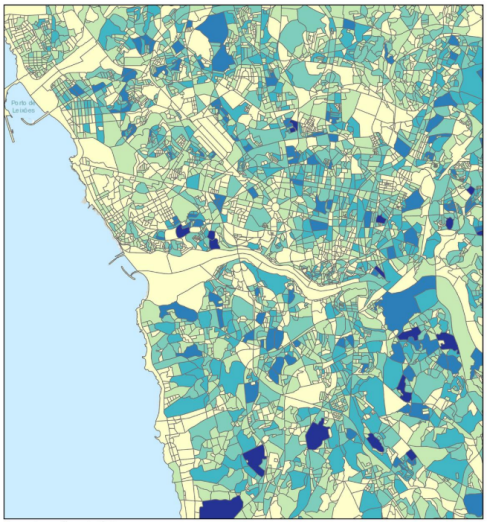

Figure 3.1: Example image of a portion of the portuguese territory divided into statistical subsections.

The variables that came from the Transparency Portal of the Portuguese National Health system are all displayed either at the hospital level or at **ACES influence area** level.

The **ACES influence area** is how the National Health System divides the country in terms of influence areas of provision of primary health care.

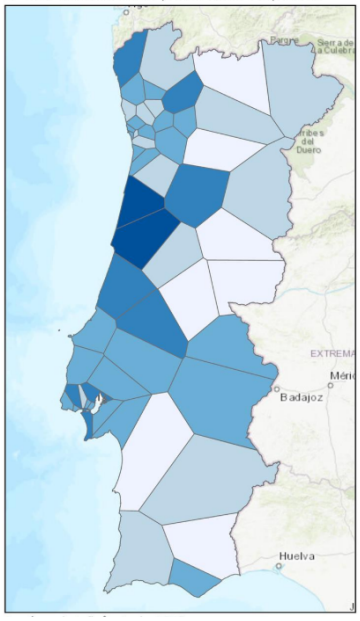

Figure 3.2: Example image of how the portuguese territory is divided under ACES influence areas.

Below we show also an image to illustrate the variables of the distances by road from two points. In the case of our dataset, we have the distances from the residence of each insured person to the closest private and public health providers.

Figure 3.3: Example image of the distance by road from the residence of insured persons to the closest public provider.

Data Types

Regarding data types, the data frame has 87 numerical variables and 20 categorical variables. The response variable, the Number of Claims is a categorical variable with values in the interval $[0, 19]$.

Missing Values

When performing a missing value analysis some variables had more than 90% of missing values, so the decision was to eliminate them. These were some variables related to the real estate information and also some regarding the vaccination programs in the public hospital of the parish of residence. All of the other variables present less than 13% of missing values, so they are all worth keeping.

In those variables that had missing values, such values were replaced by the median value of the respective column. We choose to replace them with the median because it is a more robust method since it is not so affected by the presence of outliers as the mean is.

3.1.3 Cost dataset characterization

This database has one line per claim for each annuity.

The cost database has 227174 observations and 111 variables. Similar to what happens with the number of claims database, here there are also two main groups of variables, the first one with variables indicating the cost of each claim, the age, gender, and other personal information concerning the insured person that had the claim. The second group includes, once again, information about the health and socio-economic conditions in the parish of residence of each insured person that had claims and also the distances in kilometers and travel times by road from the residence to the health care provider where the claim was registered.

Data Types

Regarding data types, the data frame has 88 numerical variables and 23 categorical variables, and the response variable, the Claims Cost is a numerical variable.

Missing Values

When performing a missing value analysis, all of the variables in the dataset have less than 10% of missing values, so they are not worth being eliminated at this point.

Given that in the cost dataset the response variable for the prediction problems that will follow is the Cost of Claims, and that, in some **claim catalogs** this variable can assume a very high range of values, some of which will be treated as outliers further in our analysis, we used again the method of replacing the missing values with the median value of the column.

3.1.4 Number of Claims Analysis and Visualization

Visualization

First, we will look at the number of claims dataset and draw attention to the response variable, the number of claims.

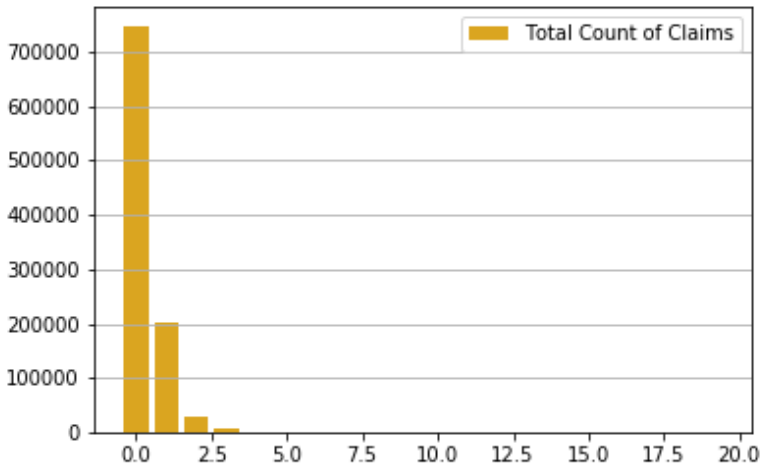

Figure 3.4: Bar plot for the number of claims distribution

From the bar plot, the number of claims that are equal to 0 (meaning the insured person had no claim reported in that month of the respective annuity) is notoriously high when compared to the other values, indicating that this dataset has a clear problem of class imbalance. The entries of the dataset classified with 0 claims represent 75% of all entries.

Next, we take a look at how the insured persons and the number of claims are distributed by age, gender, and district of residence.

The insured persons in this dataset have an average age of 38.6 years (median of age equal to 38 years) and the following age distribution.

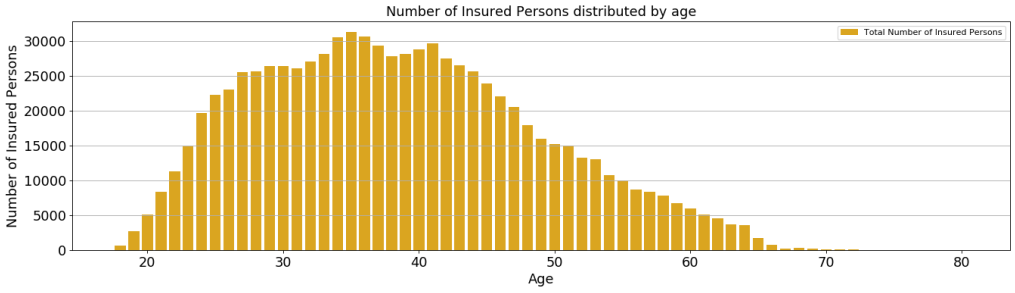

Figure 3.5: Bar plot for the number of insured persons distributed by age.

The gross of insured persons is distributed between 25 and 50 years old, with all the age values between this two having more than 15000 insured persons. This is an expected distribution since, for this work, we are only considering the insurance **policyholders**.

A **policyholder** is an insured person that has agreed with an insurance company that it will provide insurance against particular risks, meaning that, in this definition, policies of the rest of the household are not included. This is also the reason for not having anyone aged below 18 years old in our dataset.

In terms of the number of claims distribution by age, we can see by the plot below that for younger ages, the response variable has low values that tend to increase more in the interval of $[30, 40]$ years old. It is curious to note that after the age of 50, the response variable values start to decrease again. This happens because the number of insured persons in the age interval of $50+$ years old is much lower than for lower age intervals since we are not considering retired policies.

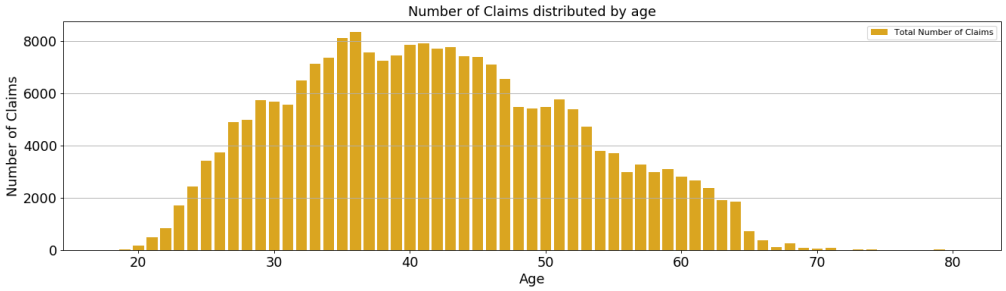

Figure 3.6: Bar plot for the number of claims distribution by age

In terms of gender distribution, in a total of 70418 insured persons, 55.5% are women and 44.5% are men as we can see from the bar plot below.

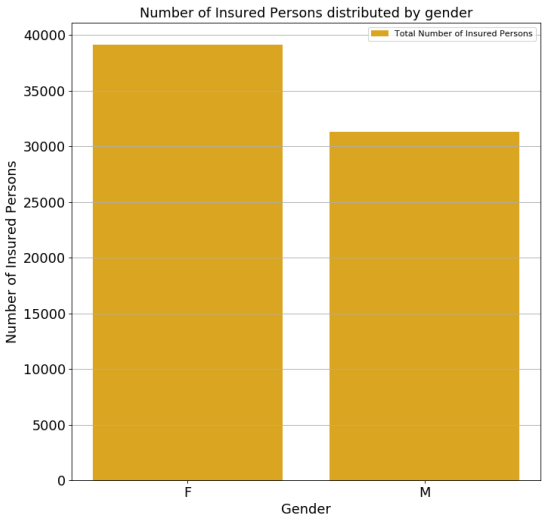

Figure 3.7: Bar plot for the gender distribution

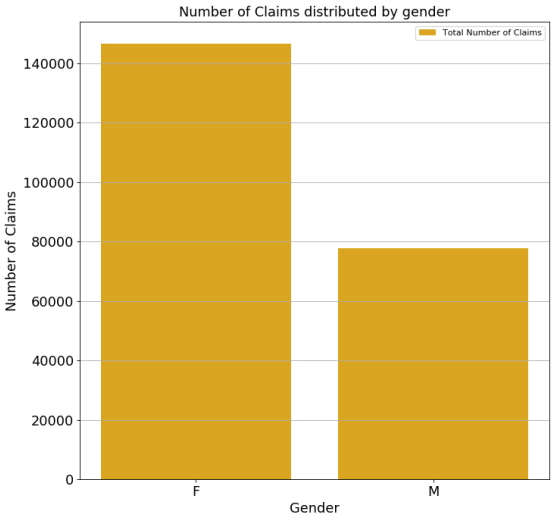

Figure 3.8: Bar plot for the number of claims distribution by gender

It is very interesting to note that, despite having a small percentage difference between the number of females and males in our data sample, the number of claims belonging to clients of the female gender is almost double of those belonging to men, as the plot on the right shows.

When looking at the plot of the distribution of the insured persons by district it is clear and expected that the largest part lives in either Lisbon or Porto districts.

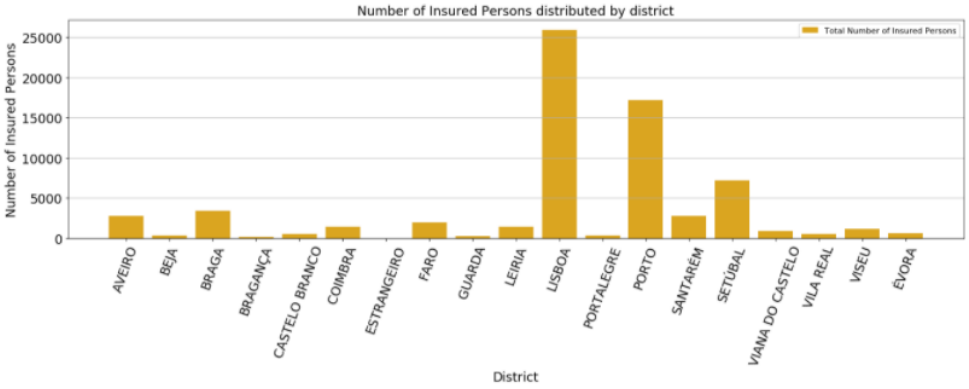

Figure 3.9: Bar plot for the district of residence distribution.

It might be interesting to relate the response variable (number of claims) to the district. Here it is expected that the number of insured persons living in one district positively influences the number of claims, meaning that the districts that have a higher number of insured persons will automatically have a higher number of claims. For this reason, we chose to plot the ratio between the number of claims and the number of insured persons by the district.

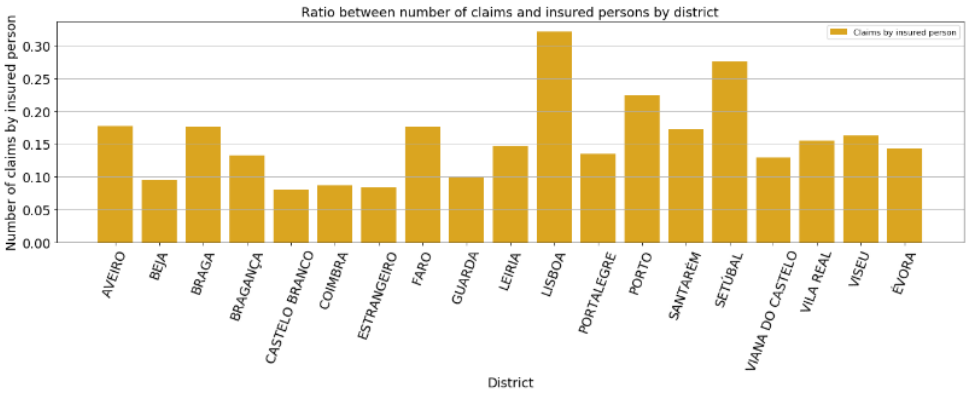

Figure 3.10: Bar plot for the number of claims by insured person distribution by district of residence.

The districts that have a higher number of claims in proportion to their respective number of insured persons are Lisbon, Setubal, and Porto. These same differences between districts are more easily seen in the Portuguese map below.

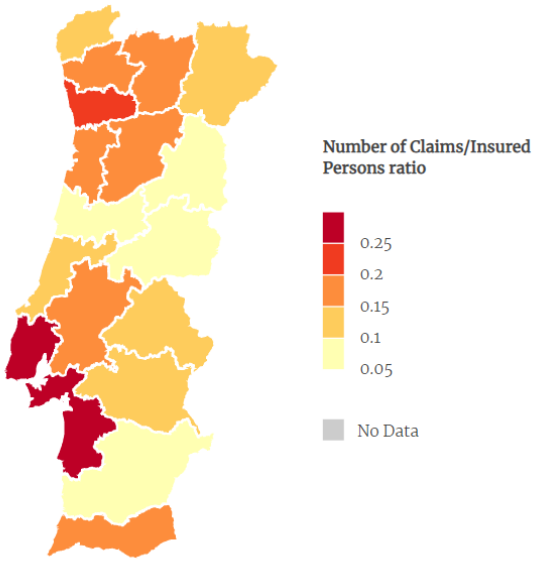

Figure 3.11: Portuguese map for the number of claims by insured person distribution by district of residence.

We see that despite for example the district of Porto has more insured persons than Setubal, the second one has a higher ratio of claims.

Correlations and Variable Importance

In terms of correlations, the first step in our approach was to calculate the pairwise correlations between all the columns in the data frame and also analyze the correlation between all the variables and the response variable (Number of Claims).

First of all, in our analysis, we found out that there are 88 pairs of independent variables that have a pairwise correlation higher than 0.9, which means that they have a high degree of correlation. This type of correlation can later influence the performance of models that assume independence between variables.

An example of two variables with a high correlation is the Travel Time Home-Closest Private Hospital and the Travel Distance Home-Closest Private Hospital. From the scatter plot of these two variables shown below, the linear correlation is visible.

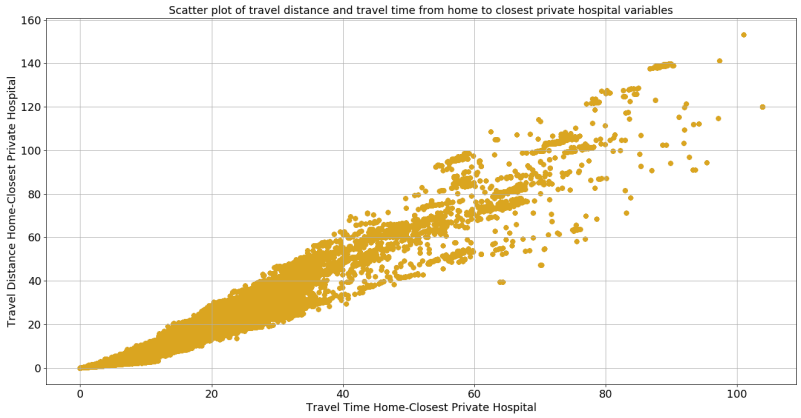

Figure 3.12: Scatter plot of travel distance and travel time from home to closest private hospital variables.

For this reason, one of the approaches we will take further in this work is performing **Principal Componente Analysis** in this dataset.

In terms of correlations between independent variables and the response variable, the values we found were all a bit low.

Table 3.3: Table of the independent variables that achieved a higher correlation coefficient with the response variable in the number of claims dataset.

Variable	Correlation Coefficient
DEDUCTIBLE_NET_NORMAL	0.246
DEDUCTIBLE_YEARLY	0.227
DEDUCTIBLE_NET_MAX	0.222
PERC_COPART_INSU_NET	0.216
PLAFOND	0.180
PROP_ELDERLY_CHRONIC_DISEASE_VACCIN	0.164
TIME_AT_RISK	0.152
USERS_SUBSCRI_FLU_VACCINE	0.141
AGE	0.134

The top three variables that achieve the higher value for the Pearson correlation coefficient are the ones related to **deductibles**.

A **deductible** is the amount paid out of pocket by the policyholder before an insurance provider will pay any expenses. [27] The value of each **deductible** varies from contract to contract and is negotiated between the insurance company and the policyholder at the moment of subscription.

Despite the low values for these correlations, we plotted below the scatter plots between the Number of Claims and the top four correlated variables, i.e., the three different types of deductibles and the percentage of co-payment that the insurance ensures in the health providers included in the **providers net**.

The **providers net** is the list of health providers that have special agreements with the insurance company and normally contracted prices for several medical acts.

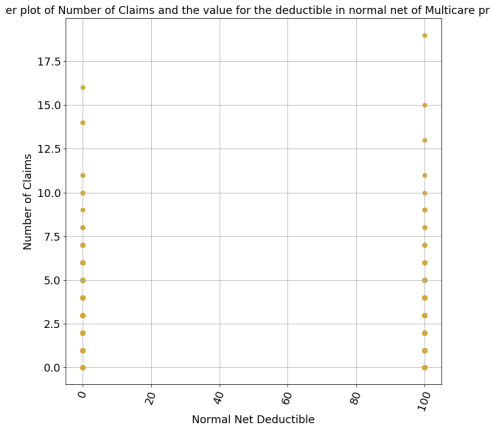

Figure 3.13: Scatter plot of Number of Claims and the value for the deductible in normal net of Multicare providers.

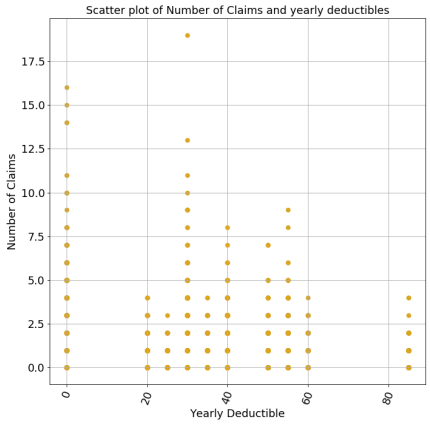

Figure 3.14: Scatter plot of Number of Claims and yearly deductibles.

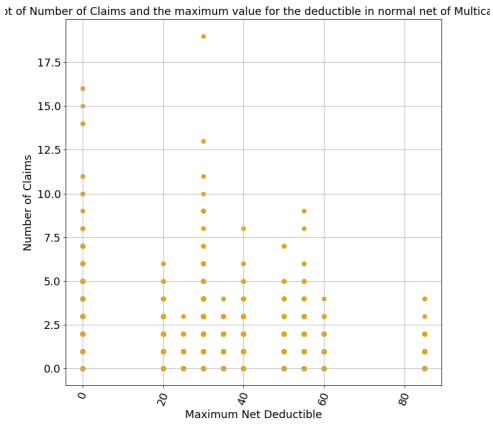

Figure 3.15: Scatter plot of Number of Claims and the maximum value for the deductible in normal net of Multicare providers.

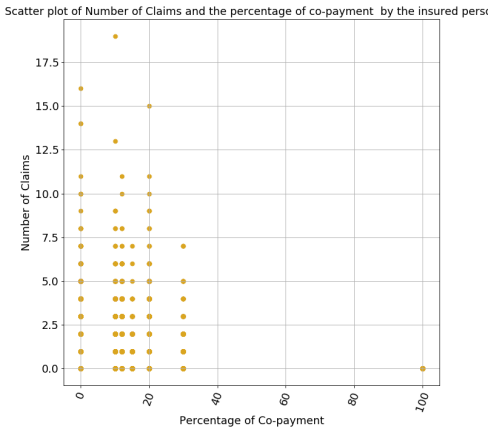

Figure 3.16: Scatter plot of Number of Claims and the percentage of co-payment by the insured person.

By analyzing these plots it is clear why the correlation values are so low.

The fact that the Pearson correlation coefficients between independent variables and the dependable one are not very high does not mean that they are not correlated, it just means that they are not linearly correlated, and we believe that the problem of forecasting the number of claims based on the variables we have available is too complex to be captured by any linear phenomena.

With this in mind, we decided to run another method for checking the importance of the co-variables to explain the response variable. We trained a Random Forest Regressor with 50 estimators and extracted the **features importances**.

Feature Importances are useful to quantify the strength of the relationship between the predictors and the outcome and rank the predictor variables. As the number of attributes becomes large, exploratory analysis of all the predictors may be infeasible, and concentrating on those with strong relationships with the outcome may be an effective training strategy. [28]

According to this method, the features that are more important for the prediction of the Number of Claims are the claim catalogs, the age of the insured persons, the time by road that it takes from the

house of each insured person to the closest public hospital, the month in which the insured person is exposed to risk and their professional occupation.

- CATALOG
- AGE
- CLOSEST_PUBLIC_HOSPITAL_TIME_TRAVEL
- MONTH
- PROFESSIONAL_OCCUPATION

Below we show the scatter plots of some of the above variables with the number of claims.

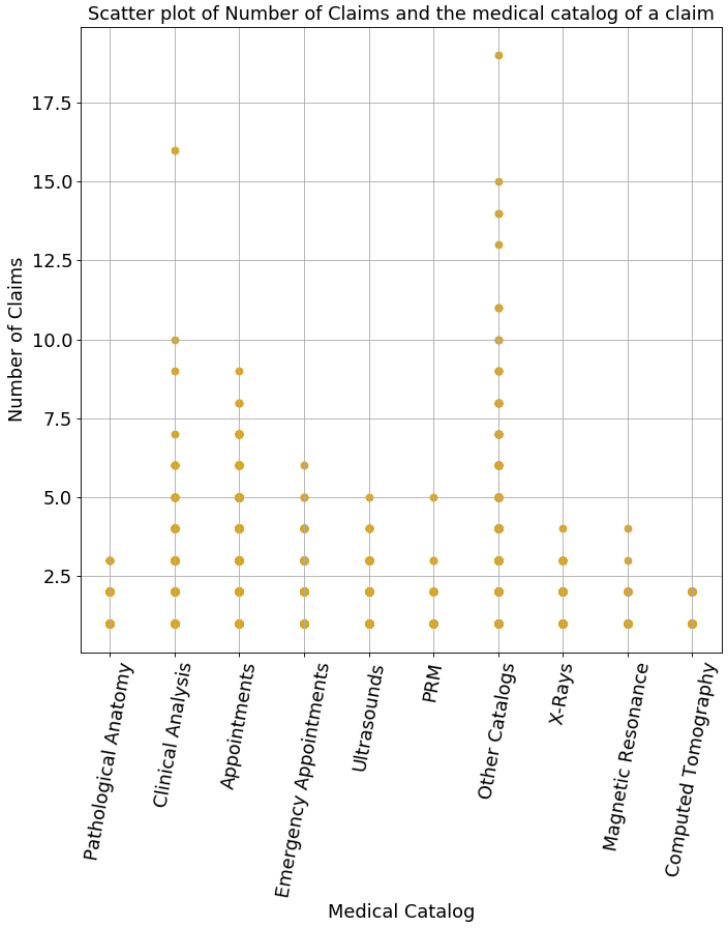

Figure 3.17: Scatter plot of Number of Claims by insured person in one annuity and the medical catalog of a claim.

We can see that there are some medical catalogs in which insured persons registered higher numbers of claims than others. Clinical Analysis is one good example of a catalog that tends to have a high number of claims since most of the individual clinical analysis are usually cheap and when a doctor prescribes what we empirically call clinical analysis, that prescription is generally made of a lot of individual analysis. On the other hand, there are no records of any insured persons having more than two claims of computed tomographies (CT scan) in the same annuity, which is normally an expensive claim that can easily make insurance plafonds run out.

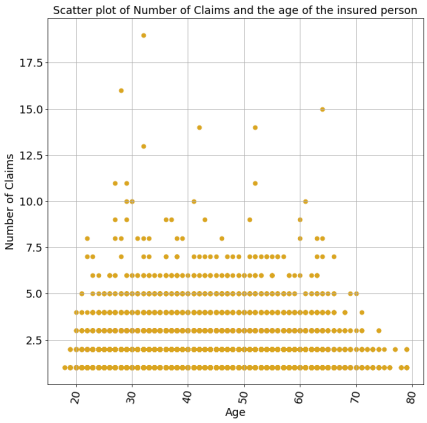

Figure 3.18: Scatter plot of Number of Claims and the age of each insured person.

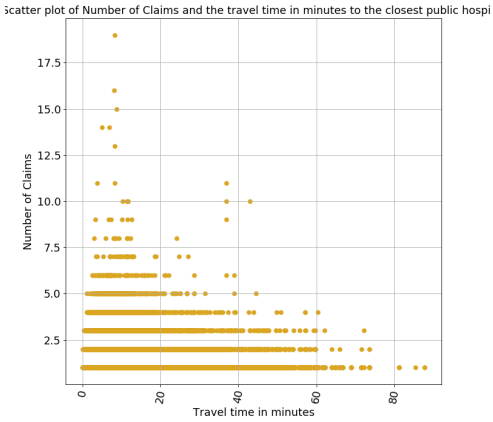

Figure 3.19: Scatter plot of Number of Claims and the travel time in minutes from the each insured person's residence and the closest public hospital.

In the scatter, plot on the left the younger ages and the older ones tend to have a lower number of claims than the other ones. On the right, we see that the insured persons who live closer to a public hospital have more claims using health insurance than the ones that live farther away.

To strengthen our belief that the variable CATALOG is the most important one when it comes to predicting the number of claims, we trained a decision tree classifier with the number of claims dataset and plotted the tree that was trained. In Figure 3.20 we show the first two levels of the decision tree trained. In the first two levels, the tree only splits by CATALOG.

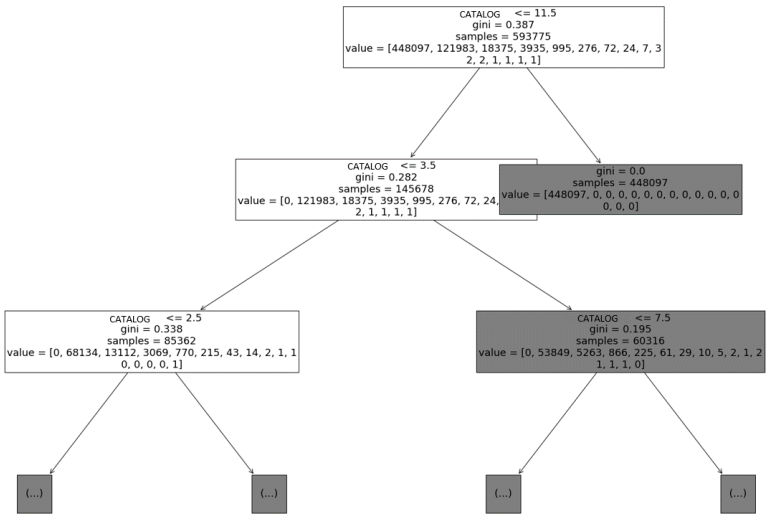

Figure 3.20: Plot of the first two levels of the decision tree trained with the original dataset.

Variable Importance Analysis by claim catalog

It may be interesting to perform the previous analysis but using only the claims of each catalog, instead of the whole data frame, and compare the conclusions.

Since we have ten different catalogs we will create ten datasets and in each of them there will be all of the insured persons, but in the variable number of claims, we will only count claims of the respective catalog.

In the table below we show the different possible catalogs for outpatient claims and summarize the top three most important variables to predict the response variable (Number of Claims) using a Random Forest Regressor and also the number of pairs of variables that have a Pearson correlation higher than 0.9 between them.

Table 3.4: Table with the top three most important variables to predict the number of claims by each catalog and the number of pairwise highly correlated variables.

Catalog	**Top 3 Important Variables (RF)**	**Number of parwise highly correlated variables**
Appointments	AGE MONTH CLOSEST_PUBLIC_HOSPITAL_TIME_TRAVEL	88
Clinical Analysis	AGE CLOSEST_PRIVATE_ANALYSIS_LAB_KM_DISTANCE CLOSEST_PRIVATE_ANALYSIS_LAB_TIME_TRAVEL	114
Pathological Anatomy	AGE PROFESSIONAL_OCCUPATION CLOSEST_PRIVATE_ANALYSIS_LAB_KM_DISTANCE	142
Ultrasounds	AGE PROFESSIONAL_OCCUPATION CLOSEST_PUBLIC_HOSPITAL_KM_DISTANCE	200
Physical and Rehabilitation Medice	CLOSEST_PUBLIC_HOSPITAL_TIME_TRAVEL N_HOME_APPOINTMENTS_PUBLIC_SECTOR CLOSEST_PUBLIC_HOSPITAL_KM_DISTANCE	144
X-Rays	AGE CLOSEST_PUBLIC_HOSPITAL_TIME_TRAVEL PROFESSIONAL_OCCUPATION	196
Magnetic Resonance	AGE CLOSEST_PUBLIC_HOSPITAL_TIME_TRAVEL PROFESSIONAL_OCCUPATION	248
Computed Tomography	AGE WOMEN_ONC_RECORD_PUBLIC_SECTOR N_RETIRED_PARISH_RESIDENCE	76
Emergency Appointments	AGE MONTH PROFESSIONAL_OCCUPATION	88
Other Catalogs	AGE CLOSEST_PUBLIC_HOSPITAL_TIME_TRAVEL CLOSEST_PRIVATE_RADIOLOGY_LAB_KM_DISTANCE	162

Performing this analysis by catalog, therefore, eliminating the variable catalog from the dataset, gave big importance to the age variable in almost all of the datasets, which is in line with our common sense

that age is generally the most important factor to determine the number of claims.

These results, at least from our empirical point of view, seem to make more sense than the ones achieved with the whole dataset. As we saw in the scatter plot between the number of claims and the claim catalog, different catalogs display a very different number of claims, because their medical nature is in some cases completely different and the context in which they are performed is very different.

When trying to build a model for predicting the number of claims with all the catalogs together we are asking the model to capture phenomena that are generally very different.

3.1.5 Cost Analysis and Visualization

Visualization

Contrary to what happens with the number of claims dataset, in the cost one there is only information concerning people that had any claim during the analysis period. This makes the average age rise to almost 42 years old, four years more than that of the number of claims dataset. In this dataset, the response variable is the Cost of Claim, which, as already pointed, indicates how much did a claim cost in euros.

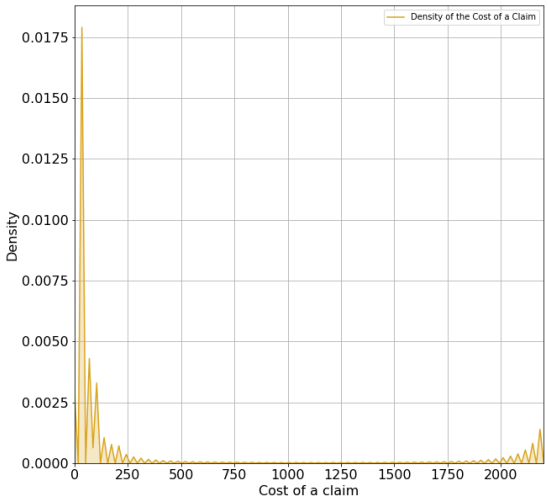

Figure 3.21: Density plot of the cost.

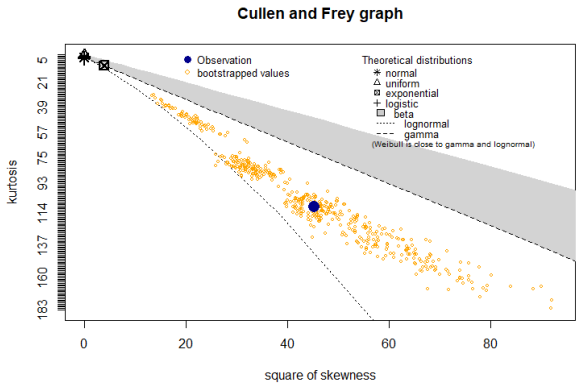

Figure 3.22: Cullen and Frey graph to approximate a possible distribution for the response variable.

From the plot on the left, it is noticeable that the Cost variable has a lot of observations for lower values and very few observations for higher values. The range of this variable is rather big, as the values for claims costs in our database vary in the interval $[0.78, 2200]$.

Using the same Cullen and Frey plot used in the number of claims dataset, the right side plot tells us that the Cost variable can be better approximated by a lognormal distribution with parameters:

$$\mu = 6.98$$

$$\sigma = 1.005$$

In terms of age, the average one is 41.8 years and it has the following age distribution.

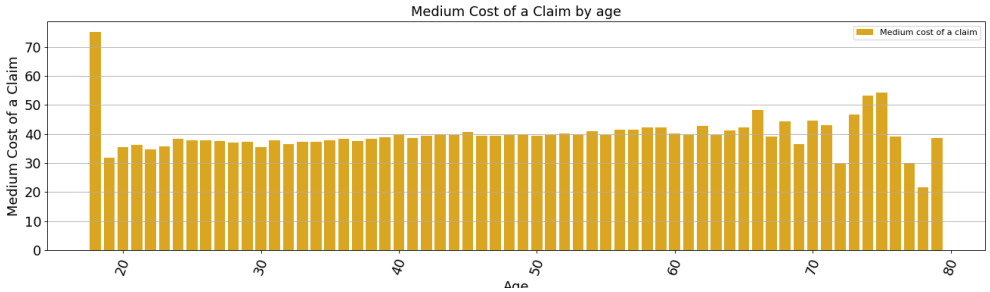

Figure 3.23: Plot of the medium cost (in euros) of a claim per age.

From the plot above we can see that the medium cost of claim values are a bit high at the age of 18 years old then they drop in the age of 20 and remain more or less stable between 30 and 40 euros and then they tend to increase again for insured persons with more than 60 years old.

The high values of the medium cost of a claim in the insured persons with 18 years old are because in our dataset we only have two records of claims, both emergency appointments with a cost of 75 euros each.

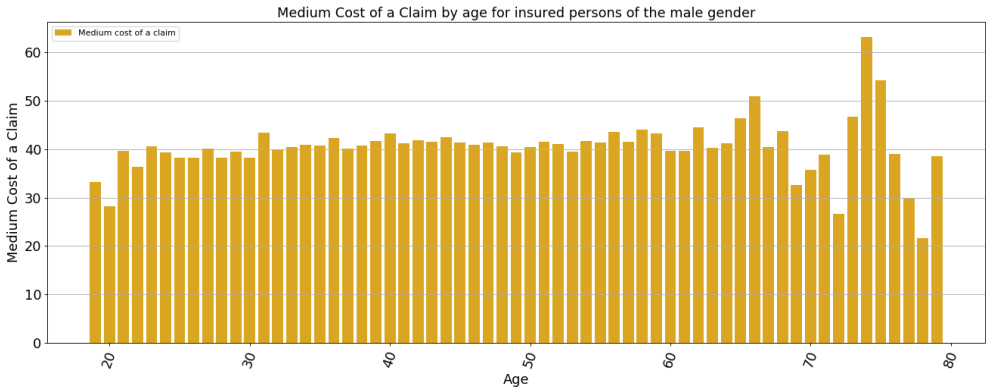

Figure 3.24: Bar plot for the medium cost distribution by age for males.

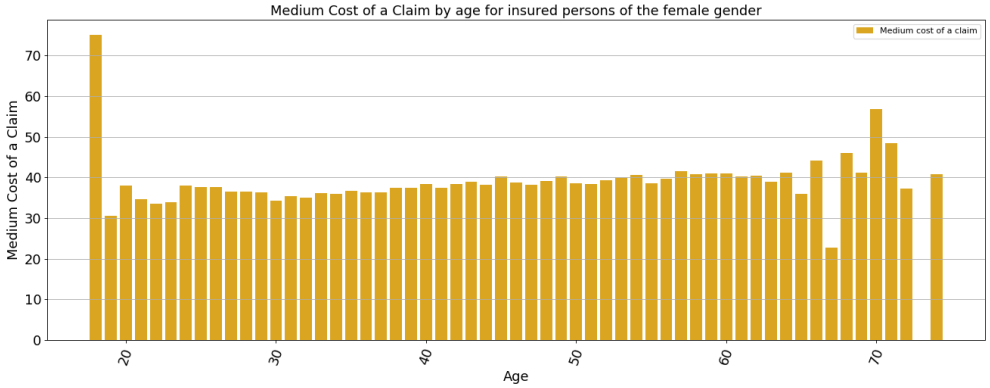

Figure 3.25: Bar plot for the medium cost distribution by age for females.

Generally, the costs of female clients are lower than for male clients, except for the 18 years old range, because, as we saw before, the two claims of emergency appointments both belong to women.

When analyzing the gender distribution and the mean claim costs per gender, the findings are very

interesting. We found that despite that in this data frame the number of female insured persons is almost double than the number of male insured persons like we saw in the number of claims distribution from the last section when we look at the medium cost of a claim, the value for male clients is higher. Women go more often to the doctor since their total number of claims is almost double that of men, but still, on average, their claims are cheaper.

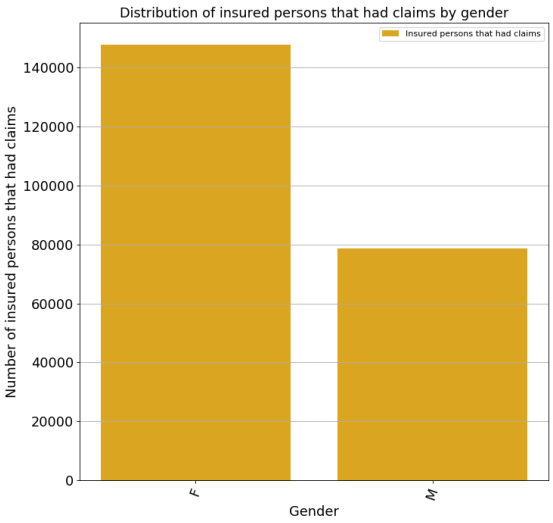

Figure 3.26: Bar plot of the gender distribution.

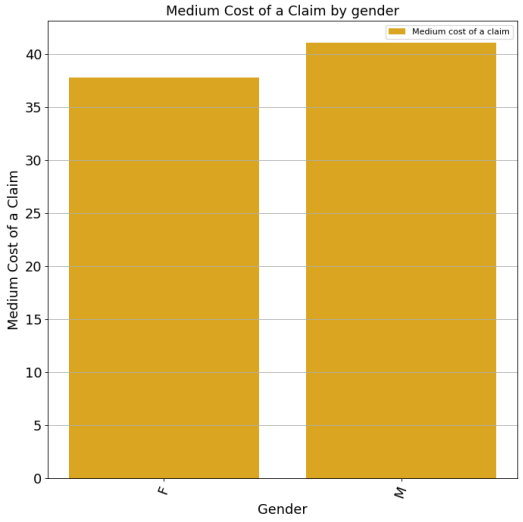

Figure 3.27: Bar plot for the medium claim cost distribution by gender.

This seems to suggest that women go more often to the doctor, therefore might prevent complicated and costly problems in the future, while men might not be so worried about prevention and only resort to the doctor when they already have a problem in a more advanced stage.

In terms of geographical distribution, we show again the plot of the number of claims per district and also one with medium cost per claim in each district.

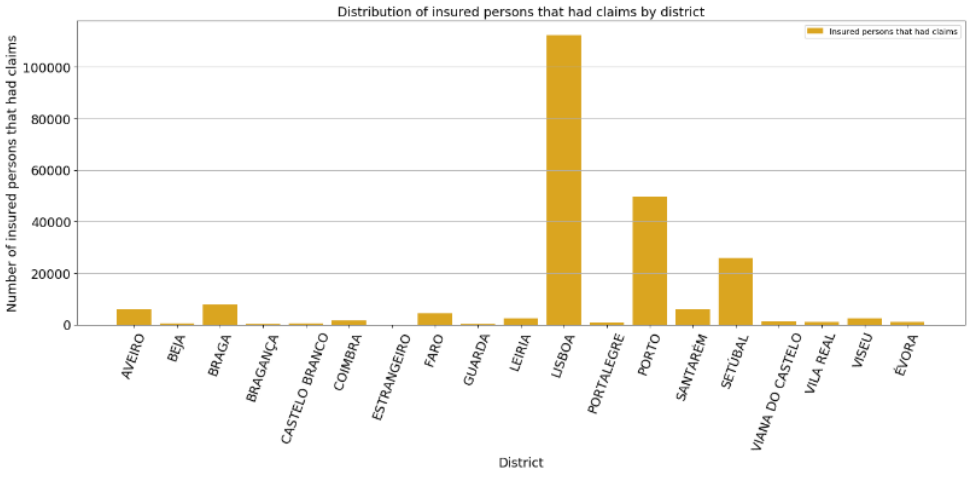

Figure 3.28: Bar plot for the district distribution.

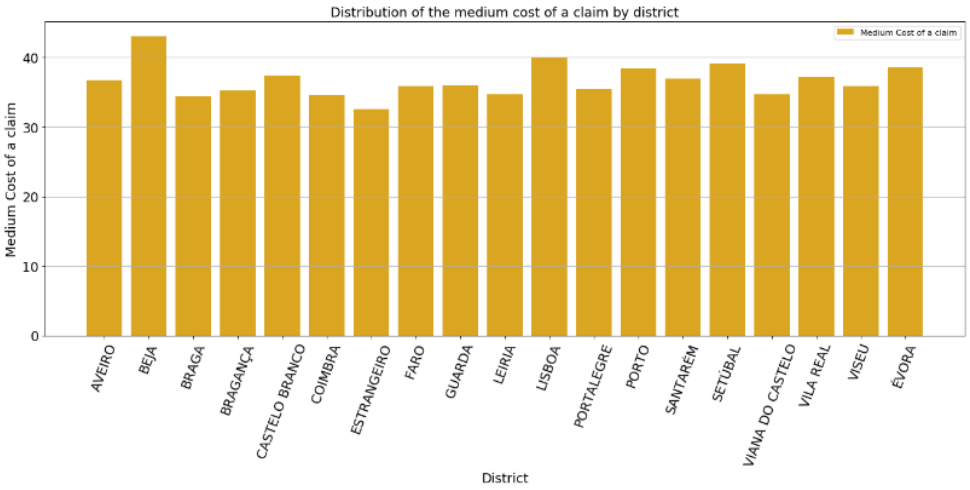

Figure 3.29: Bar plot for the average claim cost distribution by district.

In the above plots we can see that, first, the districts with more claims are, by far, Lisbon, Porto, and Setúbal, which is because they have more insured persons. The second graph represents the medium cost of a claim in each district and is more interesting to look at since we see, for example, that Beja is the district where the medium cost per claim is higher, despite being one of the districts with fewer claims.

Lisbon has the higher medium cost per claim after Beja, being closely followed by Setúbal.

An interesting fact is also the case of the district of Coimbra, despite having a reasonable population, it has a relatively low number of insured persons and also a relatively low number of claims and we know that it is because the public hospital of the city of Coimbra has a very good level of responsiveness.

This can be proved by looking at the plot of the percentage of appointments done inside the limits of what is considered by the Portuguese National Health Service as a reasonable waiting time for an appointment by the district. This percentage was extracted from the Transparency Portal of the Portuguese National Health System (SNS) and is done by the district, therefore including data from all the hospitals inside each district.

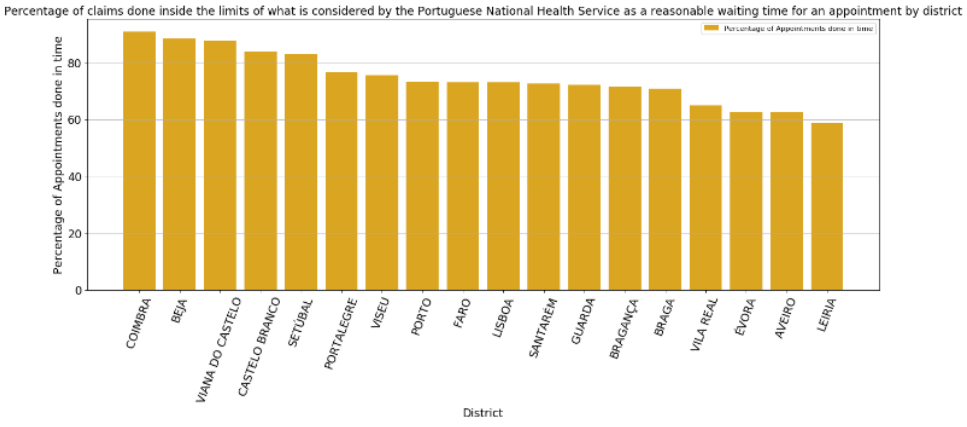

Figure 3.30: Bar plot of the percentage of appointments done inside the limits of what is considered by the Portuguese National Health Service as a reasonable waiting time for an appointment by the district.

Correlations and Variable Importance

In terms of correlations, again the first step on our approach was to calculate the pairwise correlations between all the columns in the dataset and also analyze the correlation between all the variables and

the response variable (Cost of a Claim). First of all, in our analysis, we found out that there are 128 pairs of independent variables that have a pairwise correlation higher than 0.9, which means that they have a high degree of correlation. An example of two variables with a high correlation is the Average contracted rent prices and the Average asked rent prices. From the scatter plot of these two variables shown below, the linear correlation is visible.

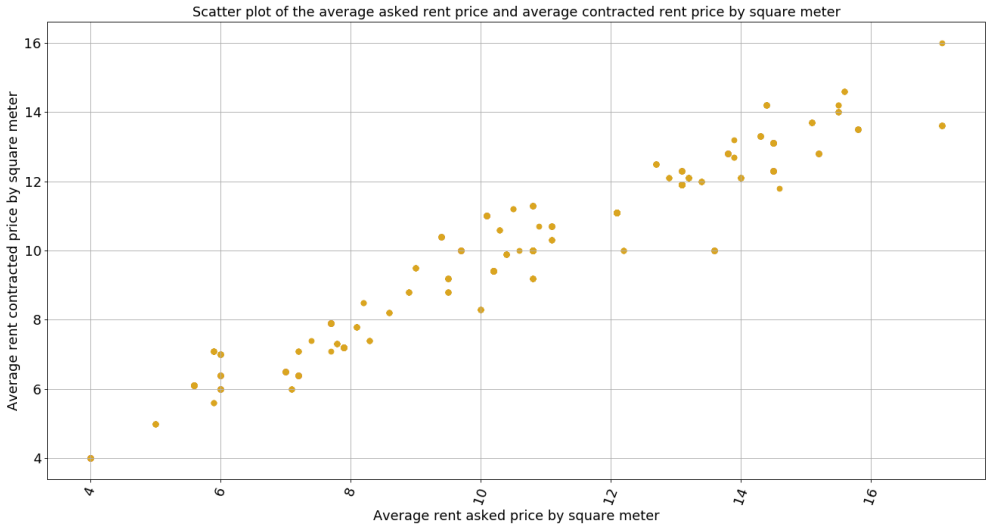

Figure 3.31: Scatter plot of the average asked rent prices in the parish of residence and average contracted rent prices in the parish of residence by square meter variables.

In terms of correlations between independent variables and the response variable, the values we found were all very low.

Table 3.5: Table of the independent variables that achieved a higher correlation coefficient with the response variable in the cost of claims dataset.

Variable	Correlation Coefficient
PERC_CHILDREN_7YO_VACCIN_PROG	0.057820
PERC_TEEN_14YO_VACCIN_PROG	0.055570
AGE	0.053374
VAL_TRANSACT_AVG	0.043400
NUM_AB_AVG_TIME_S_NEW_APART	0.040334
VAL_RENT_REQUEST_AVG	0.039369
USERS_SUBSCRI_FLU_VACCINE	0.037548
PLAFOND	0.036567
PERC_CANCER_TRACKING	0.031094

Similar to what we said regarding the problem of having low correlations between the independent and response variables in the number of claims dataset, in the cost dataset that line of thought is repeated. We believe that the problem of forecasting the cost of claims based on the variables we have available is too complex to be captured by any linear phenomena.

Following the same line of thought was for the number of claims dataset, we trained again a Random Forest Regressor with 50 estimators and extracted the feature importance of each of them. According to this method, the features that are more important for the prediction of the Cost of Claims are the claim catalogs, the age of the insured persons, the month in which the insured person is exposed to risk, the

distance by road in kilometers from each insured person residence and the closest clinical analysis lab and the travel time by road from each insured person residence and the closest private hospital.

- CATALOG
- AGE
- MONTH
- CLOSEST_PRIVATE_ANALYSIS_LAB_KM_DISTANCE
- CLOSEST_PRIV_HOSPITAL_TIME_TRAVEL

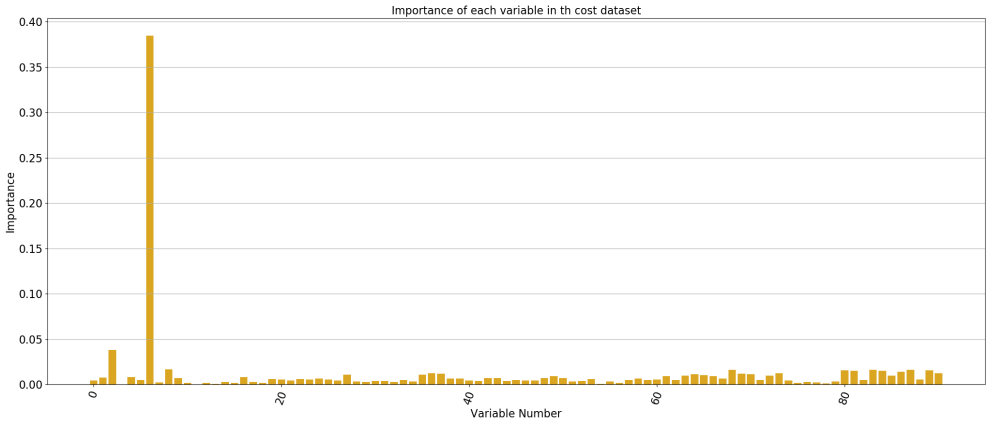

Figure 3.32: Bar plot of the importance of each of the variables to explain the response variable.

The plot above shows all of the variables of the cost dataset in the x-axis and the importance given by the random forest to each of them. The biggest bar by far that can be spotted in the plot is the variable CATALOG. This means that it is the most important variable to explain the cost. This makes perfect sense since the different types of medical catalogs have different tabulated prices for each of the different providers.

Having in mind that the Pearson correlation only captures linear relationships and therefore might not be the most robust way to measure relationships between variables, and as a way to confirm the Random Forest results, we also tried **Power Predictive Score**, a measure that tries to identify if two variables have some kind of relation (not necessarily linear). [29]

Using this method the only variable that achieved a PPS greater than 0 was CATALOG.

The fact that the variable CATALOG is constantly identified as the one which has the best relationship with the response variable indicates that it is crucial and the main differentiator in predicting the cost of a claim and that it might be interesting to analyze the dataset by each CATALOG value and compare results, again, similarly to what was done in the number of claims dataset.

Variable Importance Analysis by claim catalog

Following the same thought as we did for the number of claims dataset, in the table below we show the different possible catalogs for outpatient claims and summarize their total number of observations in our data frame, the average age of those observations, the medium cost inside each of the catalogs, the top three important variables to predict the response variable using a Random Forest Regressor and also the number of pairs of variables that have a Pearson correlation higher than 0.9 between them.

The original dataset was split into 10 different datasets, in each one of them including only the information regarding the claims of the respective type of catalog.

Table 3.6: Table with the average age, medium cost of claims, top three most important variables to predict the cost of claims and number of pairwise highly correlated variables by catalog.

Catalog	Average Age	Medium Cost (€)	Top 3 Important Variables (RF)	Number of parwise highly correlated variables
Appointments	41.5	32.5	AGE PERC_APPOINT_PUBLIC_SECTOR DISTRICT	82
Clinical Analysis	41.6	12	AGE GENDER MONTH	80
Pathological Anatomy	41.2	30	GENDER AGE MONTH	84
Ultrasounds	41.8	34	AGE MONTH DISTRICT	92
Physical and Rehabilitation Medice	41.2	12.1	CLOSEST_PRIV_HOSPITAL_TIME_TRAVEL CLOSEST_RAD_LAB_KM_DISTANCE CLOSEST_RAD_LAB_TRAVEL_TIME	114
X-Rays	44.3	25.1	AGE GENDER CLOSEST_PRIVATE_ANALYSIS_LAB_KM_DISTANCE	92
Magnetic Resonance	44	186	PERC_APPOINT_PUBLIC_SECTOR AGE DISTRICT	76
Computed Tomography	44.1	98.7	AGE N_RETIRED_PARISH_RESIDENCE PERC_APPOINT_PUBLIC_SECTOR	72
Emergency Appointments	42.3	83.3	N_EMER_APPOINT_PUBLIC_SECTOR DISTRICT PERC_APPOINT_PUBLIC_SECTOR	88
Other Catalogs	39.4	47.3	AGE MONTH N_RESID_COLLEGE_DEGREE_PAR_RESID	84

When we split the dataset of the several types of claims, the variable CATALOG is no longer present and the random forest gives more importance to the variables AGE and MONTH in most of the catalog datasets. Also, the GENDER is rated as an important factor to explain the cost of some medical catalogs.

We also note the relations identified between the cost of claims, which are always done in private providers, with some variables regarding the public health services. For example, the variable identified as the most important one to explain the cost of an emergency appointment is the number of emergency appointments performed in the public sector.

In general, the results of the above table are all in line with our empirical knowledge.

3.2 Association Rules

Before introducing algorithms for predicting both the number of claims and cost we decided to perform a market basket analysis on the **claim catalogs** using association rules, in particular, the **apriori algorithm**.

The **apriori algorithm** proceeds by identifying the frequent individual items in the database and extending them to larger and larger item sets as long as those item sets appear sufficiently often in the

database.

Apriori uses **breadth-first search** and a **hash tree** structure to count candidate item sets efficiently. It generates candidate item sets of length k from item sets of length $k-1$. Then it prunes the candidates which have an infrequent subpattern. The candidate set contains all frequent k-length item sets. After that, it scans the transaction database to determine frequent item sets among the candidates. [30]

	antecedents	consequents	antecedent support	consequent support	support	confidence
3	(AnatomiaPatologicaOHE)	(ConsultasOHE)	0.251847	0.853187	0.236282	0.938199
9	(OutrosOHE)	(ConsultasOHE)	0.567634	0.853187	0.530081	0.933842
1	(AnalisesClinicasOHE)	(ConsultasOHE)	0.435149	0.853187	0.397114	0.912594
7	(EcografiaOHE)	(ConsultasOHE)	0.345370	0.853187	0.309878	0.897234
4	(ConsultasUrgenciaOHE)	(ConsultasOHE)	0.272153	0.853187	0.221165	0.812650

Figure 3.33: Association Rules with higher confidence.

We found that with a **confidence** higher than 80%, the clients who have Pathological Anatomy, Clinical Analysis, Ultrasounds, Emergency Appointments, and Other Claims are very likely to do an Appointment next.

Confidence is a measure of how frequently a rule was actually found to be true. It can be present as the following formula depending on the **support** of that rule:

$$conf(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)} \tag{3.1}$$

The **support** of a rule is an indication of how frequently the itemset appears in the dataset. Considering T as the dataset of transactions, t a single transaction and X an itemset, the **support** is defined as:

$$\text{supp}(X) = \frac{|\{t \in T; X \subseteq t\}|}{|T|} \tag{3.2}$$

This is a curious result that is in line with our common sense, given that, every time someone does a medical exam, in most cases, they need to show it to a doctor through an appointment.

After the above result, it might also be interesting to forget these rules that seem obvious and in line with our common sense and try to find some that might not be so obvious at first sight. Below we show a table with the other association rules that were captured by the **apriori algorithm** and that have a confidence higher than 0.5.

	antecedents	consequents	antecedent support	consequent support	support	confidence
18	(AnalisesClinicasOHE)	(OutrosOHE)	0.435149	0.567634	0.318914	0.732886
13	(AnatomiaPatologicaOHE)	(AnalisesClinicasOHE)	0.251847	0.435149	0.168562	0.669304
17	(EcografiaOHE)	(AnalisesClinicasOHE)	0.345370	0.435149	0.227968	0.660068
8	(ConsultasOHE)	(OutrosOHE)	0.853187	0.567634	0.530081	0.621295
19	(OutrosOHE)	(AnalisesClinicasOHE)	0.567634	0.435149	0.318914	0.561830
16	(AnalisesClinicasOHE)	(EcografiaOHE)	0.435149	0.345370	0.227968	0.523885

Figure 3.34: Association rules with higher confidence excluding the most obvious ones.

The results obtained with this association rule mining analysis raise the question of how important the last medical act of an insured person is to predict the number of claims, given that it is important to predict the next medical act.

As a result, two new variables were introduced in our number of claims dataset, one indicating the last medical act performed by an insured person in the current annuity and the other the amount of time from the date of that last act and the current year/month/annuity the insured person is in.

Chapter 4

Experimental Setup

4.1 Number of Claims Setup

Before testing the method described in Section 2, we will test the algorithms presented there in the whole number of claims dataset and measure their performance. We will compare the three algorithms (Decision Trees, Random Forest and Gradient Boosting) to the original dataset, a version where we applied **Principal Component Analysis** and another version of the dataset where we applied a **Random Over Sampler** method.

We will split the whole dataset, into train and test sets, with 80% for training and 20% for testing. The approach here was to test which of these three dataset versions will perform better before moving on to making the predictions by client/annuity. The best-performing one will be used in that next step. The performance measures that will be used in this step are Accuracy and F1-Score. This last one was chosen because the number of claims dataset is highly unbalanced as can be seen from the plot below.

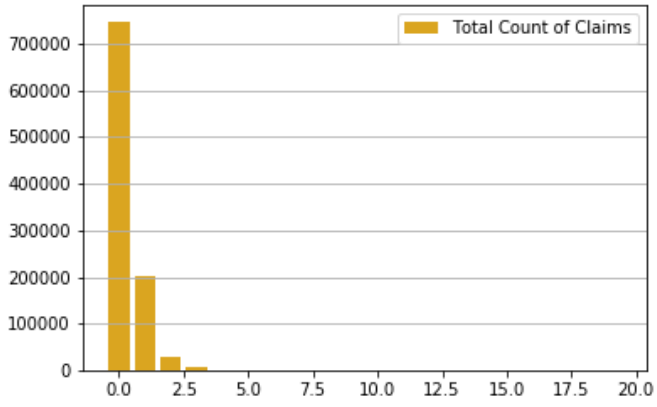

Figure 4.1: Bar plot for the number of claims distribution.

We show again the count of all the possible values of the response variables Number of Claims. We can see clearly that the response variable is imbalanced since the number of values equal to 0 is higher than the sum of the counts of all the other possible values.

4.1.1 Metrics

Accuracy

Accuracy is a classification metric that takes the fraction between the total number of correct predictions over the total number of predictions.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{4.1}$$

This metric might not be very reliable when we have a highly imbalanced dataset, since it can achieve high values and still get the predictions for entire classes completely wrong.

F1-Score

F1-score is calculated from the **precision** and **recall** of the test, where the **precision** is the number of correctly identified positive results divided by the number of all positive results, including those not identified correctly, and the **recall** is the number of correctly identified positive results divided by the number of all samples that should have been identified as positive.

The F1 score is the harmonic mean of the **precision** and **recall**. The highest possible value of F1 is 1, indicating perfect **precision** and **recall**, and the lowest possible value is 0, if either the **precision** or the **recall** is zero. [31]

$$F1 = \frac{TP}{TP + \frac{1}{2}\dot{(}FP + FN)} \tag{4.2}$$

This balance between precision and recall makes this metric a better one for measuring performance in imbalanced datasets.

4.1.2 Principal Component Analysis

Principal Component Analysis is a method to explain the associations among a set of variables through linear combinations of these variables. It is mainly used to perform data reduction.

Let $X = (X_1, ..., X_p)^t$ be a random vector describing a given population, with mean μ and covariace matrix $\sum$. The principal components can then be defined, **algebrically**, as non-correlated linear combinations of the original variables and, **geometrically**, as corresponding to a new coordination system of axes (change of base) to represent the data.

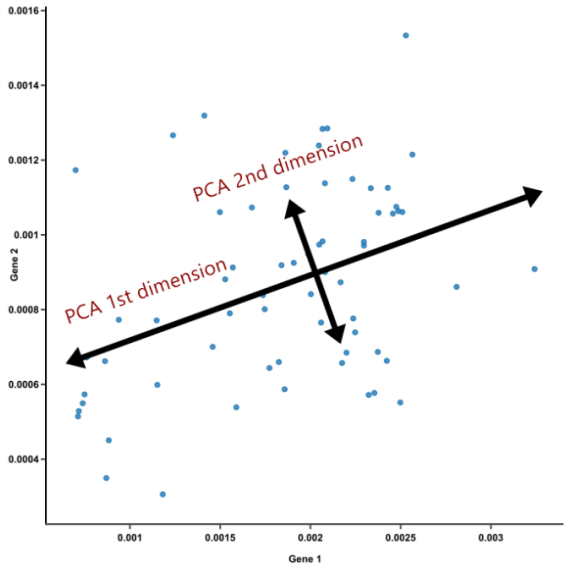

Figure 4.2: Example of a new coordination system of axes to represent the data. [8]

The first principal component, Y_1 is the linear combination of $\boldsymbol{X}, \boldsymbol{a}^t\boldsymbol{X}$, with maximum variance, such that $\|\boldsymbol{a}\| = 1$

The i-th principal component, Y_i is the linear combination of $\boldsymbol{X}, \boldsymbol{a}^t\boldsymbol{X}$, with maximum variance, such that:

(i) $\|\boldsymbol{a}\| = 1$

(ii) $\text{Cov}\left(\boldsymbol{a}^t\boldsymbol{X}, \gamma_k^t\boldsymbol{X}\right) = 0, \Leftrightarrow \boldsymbol{a}^t\boldsymbol{\Sigma}\gamma_k = 0 \left(\Leftrightarrow \boldsymbol{a}^t\gamma_k = 0\right) \; k = 1, \ldots, i-1$

In this work we thought it might be adequate do experiment if applying PCA to our number of claims dataset might help improve the performance results, given that we have a considerable number of variables and this method contributes to making a variable selection and also that there are a lot of variables that have a high correlation between them. [32]

When we applied PCA we chose to keep the first 25 principal components because together they explained more than 80% of the variance of the data.

4.1.3 Random Over Sampling

Random oversampling involves randomly duplicating examples from the minority class and adding them to the training dataset.

Examples from the training dataset are selected randomly with replacement. This means that examples from the minority class can be chosen and added to the new more balanced training dataset multiple times; they are selected from the original training dataset, added to the new training dataset, and then returned or replaced in the original dataset, allowing them to be selected again. [33] [34]

It will be used to try to overcome the severe imbalance problem we have in the number of claims dataset.

4.1.4 Dataset Transformation Results

First, we experimented with the 3 datasets (the original one and the other two variants), first using a decision tree classifier using as a function to measure the split quality the **gini impurity**, a minimum number of samples required to split an internal node of and a minimum number of samples required

to be at a leaf node of 15, then using a random forest classifier with 50 estimators and finally using a gradient boosting classifier with 50 estimators.

The performance results are displayed in the table below.

Table 4.1: Performance results of the three dataset versions using the three different classifiers.

Classifier		Original Dataset	Random Over Sampler Dataset	PCA Dataset
Decision Tree	Accuracy	0.9615	0.9367	0.8683
	F1-Score	0.9551	0.9427	0.8571
Random Forest	Accuracy	0.8516	0.8389	0.8481
	F1-Score	0.8432	0.8371	0.8341
Gradient Boosting	Accuracy	0.7946	0.6124	0.7617
	F1-Score	0.7701	0.6943	0.6988

Despite knowing that the accuracy measure can be misleading when used in the presence of imbalanced datasets we still registered its results. The original dataset was the best in terms of accuracy and F1 score.

For the three classifiers, the original dataset was the one that achieved better results in all metrics, therefore we decided to proceed with our analysis not performing PCA neither Random Over Sampling to the dataset.

4.2 Cost Setup

We start the cost forecasting setup by looking at the density plot for the Cost of Claim variable.

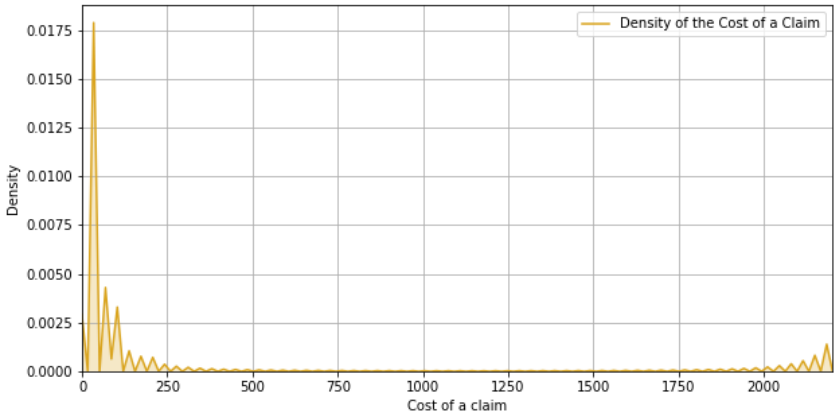

Figure 4.3: Plot of the density of the Cost of a claim variable.

We can see that claims with a lower cost are much more frequent than the more expensive ones. The range of values in the cost variable is very large since the minimum cost of a claim is 0.78 cents and the maximum is 2200 euros. Below there is a table displaying a summary of information about the response variable, the Cost of a Claim.

Table 4.2: Summary statistics of the cost dataset.

Number of obs.	226546
Mean Cost	38.92
Standard Deviation	41.97
Minimum Cost	0.78
Quantile 25	29.00
Quantile 50	32.50
Quantile 75	35.00
Maximum Cost	2200.00

As we saw in the Dataset Section, and similarly to what happened in the number of claims dataset, every time we performed variable importance analysis, the variable that always appeared first was CATALOG.

Since we have a high range of values for the cost variable and the one that most contributes to explain it was CATALOG, we decided to proceed with our analysis considering each individual catalog, i.e, we will form more homogeneous cost groups and forecast the cost of an appointment claim, the cost of a clinical analysis claim, ...

Even though inside each of these groups, the cost variable is much more homogeneous, we still performed an outlier analysis, resorting to a technique called **DBSCAN**, to detect and eliminate any extreme values.

4.2.1 Metrics

Root Mean Squared Error

The **root mean squared error** (RMSE) of an estimator measures the average of the squares of the errors. It takes the differences between the real values and the predicted values, i.e. the residuals, squares them, and then computes the square root of the mean of these squared values.

Given this informal definition, it becomes intuitive to understand that the RMSE values are always positive and that the estimator's performance is better the closer the RMSE value is to zero.

More formally, it can be defined resorting to the following mathematical formula:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2} \tag{4.3}$$

for Y_i the i-th real value, $\hat{Y}_i$ the i-th predicted value and N the size of the sample.

4.2.2 Outlier Analysis

To help understand what was described before, we chose the catalog Clinical Analysis just to illustrate that there are still some extremely high values of the cost that might be selected by our outlier detection method as being outliers.

We can spot a big tail in the density plot of the cost of a Clinical Analysis claim.

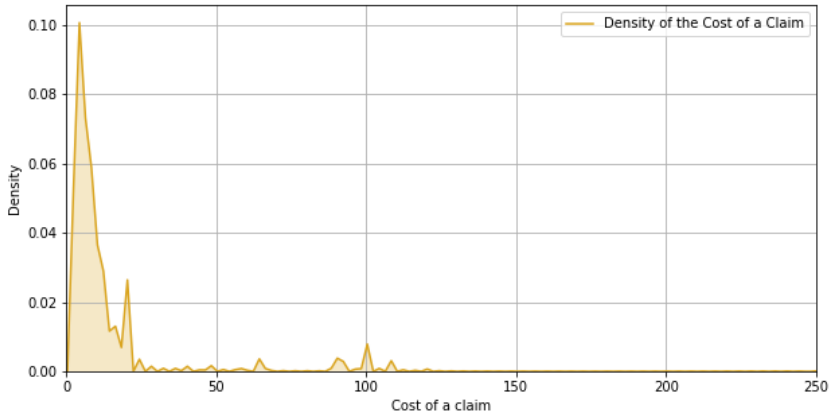

Figure 4.4: Plot of the density of the Cost of a Clinical Analysis claim variable.

Table 4.3: Summary statistics of the cost of clinical analysis claims.

Number of obs.	25853
Mean Cost	12
Standard Deviation	19.38
Minimum Cost	0.78
Quantile 25	3.67
Quantile 50	6.26
Quantile 75	11.08
Maximum Cost	250

The range of values that this variable can take is a bit high, which can be problematic in the training phase. Let's first apply the **DBSCAN** method and then check if it detects the presence of any outlier values.

DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is, in reality, a density-based clustering approach and not an outlier detection method. It grows clusters based on a distance measure. Core points, i.e. points that have a minimum of points in their surrounding, and points that are close enough to those core points together form a cluster.

Nevertheless, it can be used for outlier detection because points that do not belong to any cluster get their class: -1. The algorithm has two parameters (*epsilon*: length scale, and *min_samples*: the minimum number of samples required for a point to be a core point). Finding a good epsilon is critical.

DBSCAN thus makes binary predictions: a point is either an outlier or not. To refine the predictions, we consider the other clusters apart from the main cluster also as outlier clusters, the smaller the cluster, the higher the outlier score.

The used distance function will be the default Euclidean distance.

Taking the above example of the Clinical Analysis claims, we applied the **DBSCAN** algorithm with an *epsilon* of 300 and a *min_samples* of 2 and it identified the values of cost in the range $[120, 250]$ as being outliers and eliminated them.

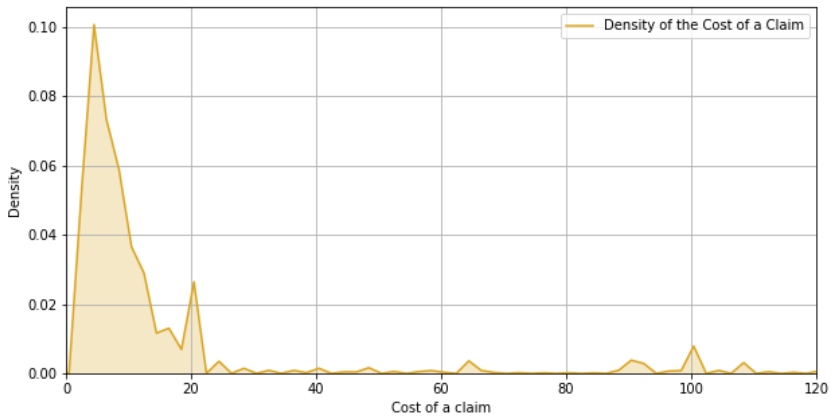

Figure 4.5: Plot of the density of the Cost of a Clinical Analysis claim variable after performing **DBSCAN**.

Table 4.4: Summary statistics of the cost of clinical analysis claims after performing outlier analysis.

Number of obs.	21008
Mean Cost	11.84
Standard Deviation	18.83
Minimum Cost	0.78
Quantile 25	3.7
Quantile 50	6.33
Quantile 75	11.09
Maximum Cost	120

After this outlier analysis, 484 observations were eliminated and the mean cost dropped from 12 euros to 11.84.

The Clinical Analysis catalog was used here as an example of what we did for all of the other catalogs.

4.3 Forecasting Setup

In terms of the number of claims, given that the variable CATALOG was selected as the most important one both in the Random Forest importance method and in the Power Predictive Score and was also the first one to be chosen by the decision tree we trained above, we decided to proceed to forecast the number of claims for each individual catalog.

What does this mean? We have 10 different claim catalogs, meaning we will train a Decision Tree, a Random Forest, and a Gradient Boosting machine to each of the 10 catalogs and compare the performance results using F1-Score. Using this method we will choose for each catalog the best performing classifier and use it to predict the number of claims of the respective catalog.

We will then start the prediction of the number of claims by each client/annuity.

In the end, we will have the total number of claims predicted for the last three months of the annuity for each catalog, $N\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j$.

We will do this for all the client/annuity pairs.

The same will happen with the cost, we will train a Decision Tree, an XGBoost machine, and a Random Forest to each of the 10 catalogs and compare the performance results using RMSE. Using this method we will choose for each catalog the best performing regressor and use it to predict the cost of claims of the respective catalog.

We will then start the prediction of the cost of claims by each client/annuity.

In the end, we will have the mean cost of a claim predicted for the last three months of the annuity for each catalog, $C\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j$.

This means that the total amount of claims for client i in the last three months of annuity j is calculated as follows, given N to be the total number of catalogs:

$$Reported\_Claims\_Pred\_Client_i\_Annuity_j = \sum_{k=1}^{N} N\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j \times C\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j \quad (4.4)$$

With the value calculated above we can easily compute the Loss Ratio for client i in annuity j:

$$Loss\_Ratio\_Pred\_Client_i\_Annuity_j = \frac{Reported\_Claims\_Pred\_Client_i\_Annuity_j}{Total\_Earned\_Premiums\_Client_i\_Annuity_j} \quad (4.5)$$

since the value of $Total\_Earned\_Premiums\_Client_i\_Annuity_j$ is previously known.

In sum, the goals of this work will be to:

- Compare the performances of the three classifiers in the number of claims prediction for each catalog;
- Compare the performances of the three regressors in the cost prediction for each catalog;
- Compare the final predicted loss ratio (using the classifier and regressor that achieved the best performance for each catalog) for each client/annuity with the value of the baseline model (ARIMA, currently in production in Multicare);
- Compare the mean squared error of all predictions for every client/annuity of our model with the baseline model;
- Compare the amount of money saved or spent by the insurance company if either the renewal proposal was made following our new model and the baseline model.

All results of the above experiments will be shown in the Results section below.

Chapter 5

Results

In this chapter, we will show the results of the performance comparisons proposed at the end of the previous chapter.

To generate these predictions we will take each corporate client and their respective annuities and use the values for the first nine months of those annuities to be our training set and the last three months to be our testing set.

One of the error metrics used to measure the performance (that we normally use in Multicare) for both the baseline and our model was the following:

$$Error(\%) = \frac{Reported\ Claims\ Forecasted - Reported\ Claims\ Real}{Reported\ Claims\ Real} \times 100\% \tag{5.1}$$

This means that when the error is negative it means that the model forecasts a value below the real one and when it is positive it forecasts a value above the real one, i.e., an error of -10%, for example, means that the value of claims forecasted by the model is 10% lower than the real value of claims.

This is a piece of information that we want to know, since the model should be above the real value than below, because, in real contract negotiation, it gives the insurer a much more comfortable position when the forecasted value is slightly above the real one than the other way around.

As stated in the introduction section of this work a contract renewal negotiation starts with the forecasting of the last three months of the present annuities. At this point, if the value forecasted by our models is lower than the real value of claims, the client will automatically ask for a discount in the next annuity premium, making it hard for the insurer to assume any negotiation position other than accepting lowering the price or risking losing the client. The lower the forecasted value in comparison to the real one, the higher the discount demanded by the clients. On the other hand, if the forecasted value is above the real one, the insurance company is not forced to lower the premium of the next annuity and as much more margin to negotiate it.

The goal of this work is to forecast the loss ratio of each client/annuity, however, to provide a better understanding of the amount of money involved we will show the results in terms of $Reported\ Claims$.

$$Reported\ Claims = Loss\ Ratio \times Total\ Earned\ Premiums \tag{5.2}$$

5.1 Number of Claims Forecast by catalog

In this section, we will show the result of the three classifiers tested for predicting the number of claims of each catalog and compare them to see which one performs better. In this section, we will not split the insured persons by company. We are assuming that every insured person belongs to the same

company in the same annuity since the point of this experiment is just to test which classifier is better for predicting the number of claims of each catalog and the idea is to have a method that produces solid results independently of the company or annuity we want to test.

We will show a table for each catalog with information about the real number of claims the dataset had and the total number of claims each algorithm predicted, as well as the accuracy and F1 score of those predictions.

Table 5.1: Performance results of the three classifiers tested for all the ten different catalogs.

Catalog	Classifier	Real Number of Claims	Forecasted Number of Claims	Accuracy	F1-Score
Medical Appointments	Decision Tree	26877	22984	0.9294	0.9195
	Random Forest		21941	**0.9469**	**0.9339**
	Gradient Boosting		20822	0.9384	0.9132
Clinical Analysis	Decision Tree	6116	5563	0.9708	0.9623
	Random Forest		5529	**0.9769**	**0.9689**
	Gradient Boosting		5510	0.9733	0.9618
Pathological Anatomy	Decision Tree	2977	2751	0.9780	0.9704
	Random Forest		2757	**0.9815**	**0.9753**
	Gradient Boosting		2750	0.9784	0.9696
Emergency Appointments	Decision Tree	2977	2751	0.9780	0.9704
	Random Forest		2753	**0.9815**	**0.9753**
	Gradient Boosting		2754	0.9784	0.9696
Ultrasounds	Decision Tree	3801	3607	0.9849	0.9793
	Random Forest		3609	**0.9878**	**0.9834**
	Gradient Boosting		3609	0.9875	0.9829
Other Outpatient Claims	Decision Tree	10941	9509	0.9534	0.9438
	Random Forest		9378	0.9666	0.9575
	Gradient Boosting		9385	**0.9672**	**0.9584**
X-Rays	Decision Tree	2890	2745	0.9864	0.9806
	Random Forest		2769	0.9859	**0.9859**
	Gradient Boosting		2767	**0.9893**	0.9858
Magnetic Resonance	Decision Tree	1107	1086	0.9953	0.9929
	Random Forest		1089	**0.9959**	**0.9944**
	Gradient Boosting		1089	**0.9959**	**0.9944**
Computerized Tomography	Decision Tree	918	904	0.9962	0.9944
	Random Forest		907	**0.9970**	**0.9961**
	Gradient Boosting		905	0.9965	0.9950
Physical and Rehabilitation Medicine	Decision Tree	155	148	0.9706	0.9684
	Random Forest		153	**0.9983**	**0.9979**
	Gradient Boosting		149	0.9965	0.9954

5.2 Cost Forecast by catalog

In this section we will show the results of the three regressors tested for predicting the cost of a claim of each catalog and compare them, to see which one performs better. Similarly to what was done with the number of claims dataset, in this section, we will not split the insured persons by company, we are assuming that every insured person belongs to the same company and that n the same annuity since the point of this experiment is just to test which regressor is better for predicting the cost of a claim of each catalog and the idea is to have a method that produces solid results independently of the company or annuity we want to test.

We will show a table for each catalog. In each one, we will compute the mean cost of all claims in the dataset (Real Medium Cost) and compute the mean cost of all claims predicted by each model (Forecasted Medium Cost) as well as the RMSE for each model.

Table 5.2: Performance results of the three regressors tested for all the ten different catalogs.

Catalog	Regressor	Real Medium Cost	Forecasted Medium Cost	RMSE
Medical Appointments	Decision Tree	32.45	32.44	2.17
	XGBoost		32.45	2.07
	Random Forest		32.44	**1.97**
Clinical Analysis	Decision Tree	12.39	12.08	21.37
	XGBoost		12.31	**20.92**
	Random Forest		13.12	21.16
Pathological Anatomy	Decision Tree	37.77	37.52	16.66
	XGBoost		38.22	16.10
	Random Forest		39.19	**15.72**
Emergency Appointments	Decision Tree	83.39	83.19	12.45
	XGBoost		83.01	11.03
	Random Forest		82.96	**10.72**
Ultrasounds	Decision Tree	33.62	33.47	10.62
	XGBoost		33.51	10.08
	Random Forest		33.94	**9.96**
Other Outpatient Claims	Decision Tree	46.66	47.3	**80.73**
	XGBoost		48.53	82.68
	Random Forest		51.26	82.72
X-Rays	Decision Tree	24.08	23.98	17.13
	XGBoost		24.31	17.69
	Random Forest		24.45	**16.75**
Magnetic Resonance	Decision Tree	184.12	184.18	24.98
	XGBoost		183.48	23.14
	Random Forest		183.35	**22.22**
Computerized Tomography	Decision Tree	95.23	95.1	9.72
	XGBoost		95.12	9.39
	Random Forest		95.19	**9.08**
Physical and Rehabilitation Medicine	Decision Tree	5.08	4.19	6.22
	XGBoost		4.7	6.77
	Random Forest		4.25	**5.76**

5.3 Forecasting pipeline

In the previous section the performances of both the regressors and the classifiers, used for predicting the cost of a claim and the number of claims respectively, are presented.

The forecasting architecture is presented in section 4.3 and consists of forecasting the number of claims and the cost of a claim for and computing the total amount of claims for each catalog separately (in euros) and then summing over all the 10 different catalogs like displayed in the formula below.

$$TRC = \sum_{i=1}^{10} NC_i \times MC_i \tag{5.3}$$

where TRC represents the total reported claims, NC_i the total number of claims of catalog i and MC_i the medium cost forecasted for catalog i.

The classifiers and regressors for the forecasting pipeline were then chosen accordingly to the results of sections 5.1 and 5.2 and are the following:

Table 5.3: Chosen classfiers and regressors to forecast number of claims and cost of claims respectively for each catalog.

Catalog	Classifier	Regressor
Medical Appointments	Gradient Boosting	XGBoost
Clinical Analysis	Random Forest	XGBoost
Pathological Anatomy	Random Forest	Random Forest
Emergency Appointments	Random Forest	Random Forest
Ultrasounds	Random Forest	Random Forest
Other Outpatient Claims	Gradient Boosting	Decision Tree
Physical and Rehabilitation Medicine	Random Forest	Random Forest
MRI	Random Forest	Random Forest
Computerized Tomography	Random Forest	Random Forest
X-Rays	Gradient Boosting	Random Forest

As stated before we chose these classifiers and regressors based on their performance in each individual catalogs, assuming that every insured person belonged to the same company in the same annuity. This assumption was made because the goal was to set up an algorithm that can achieve good results regardless of company or annuity, instead of having a different model adapted to each company.

Given this forecasting pipeline, the next step in our approach was to measure the error of our model and compare it with the error of the baseline model (ARIMA).

5.4 Reported Claims forecast comparison

In terms of interest to the insurance company, the most important measure is to know how much our predictions are above or below the real amount spent on claims by a client in one annuity.

Therefore in this section, using the best performing classifier for forecasting the number of claims and the best performing regressor for forecasting the cost in each catalog from the above section, we built a pipeline to predict the total amount of claims spent by client i in annuity j:

$$Reported\_Claims\_Pred\_Client_i\_Annuity_j = \sum_{k=1}^{N} N\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j \times C\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j \quad (5.4)$$

using the predictions of the number of claims of each individual catalog $N\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j$ and the predicitons of the medium cost of each individual catalog $C\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j$.

The error formula was the following:

$$Error(\%) = \frac{Reported\ Claims\ Forecasted - Reported\ Claims\ Real}{Reported\ Claims\ Real} \times 100\% \quad (5.5)$$

This formula gives us an understanding of how far the value of our predicted claims is from the value of the real claims and if we are either above or below the real value.

The problem of having a prediction above or below the real value of claims of a client, might not be of much interest inside the academic context, but it is of great importance in the practical daily decisions of an insurance company since forecasting a lower value means lowering the price in the next annuity and probably ending up losing money, as we will see in greater detail in the next section.

Table 5.4: Comparison table of the errors made by both our new model and the baseline ARIMA model for each client/annuity in our study.

Client	Annuity	Real Reported Claims (€)	Forecasted Reported Claims (New Model) (€)	Forecasted Reported Claims (ARIMA) (€)	**New Model Error (%)**	**ARIMA Error (%)**
A	1	33374	33986	23149	**1.83**	**-30.63**
A	2	742538	837725	552305	**12.81**	**-25.61**
B	1	251159	240843	197633	**-4.10**	**-21.31**
B	2	281214	262885	198545	**-6.51**	**-29.39**
C	1	99307	90897	67836	**-8.46**	**-31.69**
D	1	132250	119454	77055	**-9.67**	**-41.73**
E	1	807220	874895	614837	**8.38**	**-23.83**
F	1	12845	11523	10012	**-10.29**	**-22.05**
G	1	6949	8406	7541	**20.96**	**8.51**
G	2	7024	6977	8247	**-0.66**	**17.41**
H	1	16338	16154	14697	**-1.12**	**-10**
I	1	7055	8551	6204	**21.20**	**-12.06**
I	2	4207	4393	6157	**4.42**	**46.35**
J	1	17649	17377	16679	**-1.54**	**-5.49**
L	1	18434	20021	20216	**8.60**	**9.66**
M	1	5837	6066	5368	**3.92**	**-8.03**
N	1	3313	3167	2789	**-4.40**	**-15.81**
O	1	7964	6541	5314	**-17.86**	**-33.27**
P	1	22228	25500	22838	**14.72**	**2.74**

From the table above we can see that our model achieve a smaller error in sixteen out of nineteen client/annuity pairs. From those three clients where our model had a greater error than the ARIMA baseline model in all of them the value our model predicted was greater than the real one, which, from the company perspective is not a very severe error.

Next, we calculated the total root mean squared error (RMSE) of all the above predictions of both models.

Table 5.5: Comparison between the root squared error of both models for all the clients.

	New Model	**ARIMA**
RMSE	612	67695

The RMSE of our model is more than 100 times smaller than the error of the ARIMA model.

5.5 Money Gained/Lost model comparison

After displaying the errors of our new method compared with the ARIMA baseline model the results of our new model look promising. In terms of percentage error, our model performed better in sixteen out of nineteen client/annuity pairs. When we look at the MSE over all of the client/annuity pairs the value of our new model is much lower than that of the ARIMA.

However, since this is a work that is intended to have a direct impact on the business of an insurance company, one interesting exercise that can be done is to translate all of the error results above into money, and see how much money the company would lose or win if either the renewal proposal was based on the prediction of the ARIMA model against the prediction of our new model.

As explained in the introductory section of this work, in corporate insurance contracts, the process of renewal typically starts when nine months of the current annuity have elapsed. At this time the insurer

makes a prediction of the last three months and based on that prediction it proposes the price for the next annuity following the process described below.

Since the pricing of annuity $j+1$ takes place after the first nine months of annuity j, to price the annuity $j+1$ using ARIMA, we will assume, as exercise, the method of taking the total amount of claims predicted for annuity j (the nine months of real claims that we know of plus the last three months of claims that we estimate using ARIMA) and increasing this value by the average inflation rate in the health sector of the last ten years, which is 1.01%, according to [35].

$$Price\_Client_i\_Annuity_{j+1}\_ARIMA = Total\_Claims\_Client_i\_Annuity_j \times 1.0101 = \\ (Total\_Claims\_9Months\_Real + Total\_Claims\_3Months\_ARIMA) \times 1.0101 \quad (5.6)$$

To make a fair comparison we will use the same method to forecast annuity $j+1$ using our new proposed model.

$$Price\_Client_i\_Annuity_{j+1}\_NewModel = Total\_Claims\_Client_i\_Annuity_j \times 1.0101 = \\ (Total\_Claims\_9Months\_Real + Total\_Claims\_3Months\_NewModel) \times 1.0101 \quad (5.7)$$

The column Real Price in the below tables is the real total amount of claims verified in annuity $j+1$. So, if we compute the difference between the Price Estimation using either ARIMA or the New Model (the proposed price for the next annuity) and the Real Price (total amount of claims in the next annuity) we can check if the company lost or gained money in each company.

$$Difference = Price\ Estimation - Real\ Price \quad (5.8)$$

Table 5.6: Table that shows the difference between the real amount spent in claims in each client/annuity and the price that would be proposed to the client for the next annuity based on the predictions of the baseline ARIMA model.

Company	Annuity	Total Claims 9 Months (Real) (€)	Total Claims 3 Months (ARIMA Estimation) (€)	Price Estimation (ARIMA) (€)	Real Price (€)	**Difference (€)**
A	1	2359898	23149	2407182	3257195	**-850012**
A	2	2343668	552305	2925303	3298922	**-373618**
B	1	948794	197633	1158037	1113615	**44422**
B	2	879201	198545	1088661	1252798	**-164136**
D	1	435797	77055	518045	599836	**-81790**
E	1	2595575	614837	3242926	4051080	**-808153**
F	1	37879	10012	48375	51553	**-3177**
G	1	39916	7541	47936	46503	**1433**
G	2	38868	8247	47591	50195	**-2603**
H	1	63462	14697	78950	77112	**1838**
I	1	34688	6204	41305	33414	**7891**
I	2	27756	6157	34255	47043	**-12787**
J	1	67373	16679	84902	94351	**-9448**
L	1	87157	20216	108459	171377	**-62917**
M	1	21391	5368	27029	22830	**4199**
N	1	11530	2789	14463	19857	**-5393**
O	1	22345	5314	27938	36621	**-8682**
P	1	101799	22838	125898	137143	**-11244**

Table 5.7: Table that shows the difference between the real amount spent in claims in each client/annuity and the price that would be proposed to the client for the next annuity based on the predictions of our new model.

Company	Annuity	Total Claims 9 Months (Real) (€)	Total Claims 3 Months (New Model Estimation) (€)	Price Estimation (New Model) (€)	Real Price (€)	**Difference (€)**
A	1	2359898	33986	2418128	3257195	**-839066**
A	2	2343668	837725	3213614	3298922	**-85307**
B	1	948794	240843	1207482	1207482	**-5795**
B	2	87901	262885	1153652	1159217	**-5564**
D	1	435797	119454	560873	599836	**-38962**
E	1	2595575	874895	3505618	4051080	**-545461**
F	1	37879	11523	49902	51553	**-1650**
G	1	39916	8406	48810	46503	**2307**
G	2	38868	6977	46308	50195	**-3886**
H	1	63462	16154	80422	77112	**3310**
I	1	34688	8551	43676	33414	**10262**
I	2	27756	4393	32474	47043	**-14568**
J	1	67373	17377	85608	94351	**-8742**
L	1	87157	20021	108262	171377	**-63114**
M	1	21391	6066	27734	22830	**4904**
N	1	11530	3167	14845	19857	**-5011**
O	1	22345	6541	29178	36621	**-7442**
P	1	101799	25500	128587	137143	**-8555**

Summing the values of the Difference columns in both tables we get the amount of money gained or lost by the company in the this universe of clients under study.

Table 5.8: Comparison between the money difference of both models for all the clients.

	ARIMA	**New Model**
Balance (€)	-2 334 179	-1 612 345

Chapter 6

Conclusion

This work arose from the need to develop a more accurate method for predicting the loss ratio of the outpatient coverage in corporate clients.

It started with a brief literature review on how to forecast future loss ratios in the health insurance industry. Then a more detailed description of the baseline model was presented, addressing the solutions used currently in Multicare to deal with this problem, both in the perspective of *reported claims*, as well as IBNR claims, without prejudice to the fact that in this work we only addressed the problem of forecasting *reported claims*.

Having presented the problem and the current way to handle it, the next step was to start thinking about an alternative method for the problem. An alternative forecasting method that we could, in the end, compare with the current one and realize if using the alternative method instead of the classical one translated into any economic impact for the insurance company and, if so, how much of an impact did it translate into?

With that previous goal in mind, we started by presenting a theoretical introduction to the algorithms we thought of using in this work, followed by a detailed analysis of the datasets that allowed for the construction of the prediction models.

The two datasets (one having the number of claims as the response variable and another having the cost of a claim as response variable) that embody this work was the result of a collaboration with the Advanced Analytics department of Fidelidade, that made possible the extraction and incorporation of several variables that came from external sources in our datasets and that proved to be useful during this work.

After a process of cleaning the datasets, we started a more deep analysis of them and began to suspect that to have more accurate predictions in the future the best idea was probably to divide the claims into medical catalogs to increase the homogeneity in the datasets.

This decision to divide the claims into groups according to their medical similarity led to a forecasting pipeline that needed to have many forecasting steps, i.e., there was the need to forecast both the number of claims and the cost of claims for each of the medical catalogs. The solution was to divide the original datasets into ten new ones, each regarding the number of claims and cost of claims of each of the ten different medical catalogs and training either a classifier or a regressor in each of them.

In terms of classifiers for forecasting the number of claims from the three tree-based ones tested we ended up using only two of them, random forests and gradient boosting machines because they were the ones that achieve the best results in the tests we performed. For this same reason, from the three regressors we tested initially for forecasting the cost of claims, we only used two, random forests and xgboost machines.

The next step, after defining the algorithms to forecast both the number of claims and cost of claims

for each medical catalog, was to do the forecasting of the reported claims for the last three months of the annuities of each of the fifteen companies in our study and compare the results with the ones from the baseline model.

The first results were very promising for our new model since its predictions were closer to the real values than the baseline model in sixteen out of nineteen client/annuity pairs. In terms of root mean squared error, it achieved a value much smaller than the one achieved by the baseline method.

Since this is a more practical work and one of the main goals is to develop a practical and ready to use solution for the insurance company, a metric that we thought would be important is the amount of money lost or gained by the company if the renewal proposal for the next year was done using the three-month forecast of our new model in opposition to the value forecasted using the baseline model (ARIMA). Overall, if we sum the amounts of money gained/lost by each of the companies in our study, we see that despite the company losing money with both methods, with the new model developed that loss was almost less than one million when compared to the loss generated by the ARIMA predictions.

In the last section of this work, we presented a comparison of the amount of money that would be gained or lost by the company when using both the ARIMA and our new model forecasts of the last three months to construct the next annuity prediction. The problem of calculating the next annuity prediction was handled, as we saw before, by taking the cost of the present annuities and summing 1.01% (the average inflation rate in the health sector of the last ten years) of this value. This process is the main responsible for the losses obtained both with our model and with the ARIMA ($-1.6M$ and $-2.3M$ respectively). Given this, it would be of great importance the development of a forecasting solution to deal with the next annuity predictions that could be based on this one with the respective adjustments, i. e., instead of making a forecast for three months, what is needed in this problem is the forecasting of the next fifteen months (the last three months of the present annuity as well as the twelve months of the next one).

In terms of future work, there is still a lot of ground to cover on this subject. The first step to being done in the future is because this work only is focused exclusively on the outpatient coverage of insured persons and in the claims that occurred within the net of providers of Multicare. So, given this, the first step is clearly to extend this work to the reimbursement claims inside the outpatient coverage. This way we have the loss ratio predictions for the entire outpatient coverage.

The final prediction of the last three months of the annuity of each client is intended to encompass not only the outpatient coverage but all covers, such as hospital stays, stomatology, medicines, and prosthesis, and orthotics. It is, therefore, of great importance to keep this work, extending it for these other covers. The coverage of hospital stays has the particularity of not being a consumption coverage, meaning that it is a coverage that is mostly activated when the insured person needs it and not by option. This particularity means that the consumption behaviors may differ a bit from the other consumption covers, like the outpatient one presented in this work, and therefore it might require a different kind of approach.

At last, one important note for the future is using the same approach that was done in this work but, instead of considering only the first nine months to train the models it would be important to extend the range of the training data with claims before that, i. e., using all the past claims of the previous annuities.

Bibliography

[1] Mario V. Wuthrich, “Neural Networks Applied to Chain-Ladder Reserving,” 2018. European Actuarial Journal 8(2).

[2] Mario V. Wuthrich, “Machine Learning in Individual Claims Reserving,” 2018. Scandinavian Actuarial Journal 2018/6, 465-480.

[3] Kevin Kuo, “DeepTriangle: A Deep Learning Approach to Loss Reserving,” 2019.

[4] Zeinab Amin, Arthur Charpentier, et al, “Loss Data Analytics,” 2020.

[5] Mikkel Duif, “An Introduction to Decision Trees with Python and scikit-learn,” 2020. https://towardsdatascience.com/an-introduction-to-decision-trees-with-python-and-scikit-learn-1a5ba6fc204f.

[6] Alan Fielding, “Clustering and Classification methods for Biologists: Decision Trees,” 2006. http://www.alanfielding.co.uk/multivar/cart.htm.

[7] Jorge S. Marques, “Machine Learning Slides ,” 2017.

[8] Linh Ngo, “Principal component analysis explained simply,” 2018. https://blog.bioturing.com/2018/06/14/principal-component-analysis-explained-simply.

[9] Faculty and Institute of Actuaries Claims Reserving Manual v.1, “Section G: METHODS USING LOSS RATIO & LOSS RATIO PROJECTIONS,” 1997.

[10] P. D. England and R. J. Verrall, “Stochastic claims reserving in general insurance,” 2002. British Actuarial Journal 8(3), 443–518.

[11] Daniela Sofia Marques da Costa, “Metodologias de Estimação de Provisões para Sinistros do Ramo Não Vida ,” 2016. FCUL.

[12] Shay Palachy, “Stationarity in time series analysis,” 2019. *https://towardsdatascience.com/stationarity-in-time-series-analysis-90c94f27322*.

[13] Peter J. Brockwell and Richard A. Davis, “Introduction to Time Series and Forecasting, Second Edition,” 2002. Springer.

[14] NIST/SEMATECH e-Handbook of Statistical Methods, 2003. *https://www.itl.nist.gov/div898/handbook/eda/section3/eda35h.htm*.

[15] Denis Kwiatkowski, Peter C.B. Phillips, Peter Schmidt and Yongcheol Shin, “Testing the null hypothesis of stationarity against the alternative of a unit root,” 1991.

[16] William H. Greene , “Econometric Analysis,” 1997.

[17] António Pacheco Pires, “Notas de Séries Temporais,” 2001.

[18] Rob J. Hyndman and George Athanasopoulos, “Forecasting: Principles and Practice.” Monash University, Australia.

[19] Shai Shalev-Shwartz, Shai Ben-David, “Understanding Machine Learning: From Theory to Algorithms,” 2014.

[20] Rokach, L.; Maimon, O., “Top-down induction of decision trees classifiers-a survey. ,” 2005.

[21] Matthew N. Bernstein, “Random Forests,” 2017.

[22] Adam Hjerpe, “Computing Random Forests Variable Importance Measures (VIM) on Mixed Continuous and Categorical Data,” 2016.

[23] Jerome Friedman, “Greedy Function Approximation: A Gradient Boosting Machine,” 1999.

[24] Alexey Natekin, Alois Knoll, “Gradient Boosting Machines, a tutorial,” 2013.

[25] Stephanie Glen, “Decision Tree vs Random Forest vs Gradient Boosting Machines: Explained Simply,” 2019. https://www.datasciencecentral.com/profiles/blogs/decision-tree-vs-random-forest-vs-boosted-trees-explained.

[26] Tianqi Chen, Carlos Guestrin, “XGBoost: Reliable Large-scale Tree Boosting System,” 2013.

[27] O’Sullivan, Arthur; Sheffrin, Steven M., “Economics: Principles in Action,” 2003.

[28] Max Kuhn, Kjell Johnson, “Applied Predictive Modeling,” 2013.

[29] Florian Wetschoreck, “RIP correlation. Introducing the Predictive PowerScore,” 2020. https://towardsdatascience.com/rip-correlation-introducing-the-predictive-power-score-3d90808b9598.

[30] Rakesh Agrawal, Ramakrishnan Srikant, “Fast Algorithms for Mining Association Rules ,” 1994.

[31] Yutaka Sasaki, “The truth of the F-measure,” 2007.

[32] M. Rosário Oliveira, “Principal Component Analysis and Outlier Detection,” 2019.

[33] Jason Brownlee, “Random Oversampling and Undersampling for Imbalanced Classification,” 2020. https://machinelearningmastery.com/random-oversampling-and-undersampling-for-imbalanced-classification/.

[34] Gustavo E. A. P. A. Batista, Ronaldo C. Prati, Maria Carolina Monard, “A study of the behavior of several methods for balancing machine learning training data,” 2004.

[35] PORDATA (Base de Dados Portugal Contemporâneo), “Taxa de Inflação (Taxa de Variação do Índice de Preços no Consumidor): total e por consumo individual por objectivo,” 2020. https://bit.ly/3hSOsMv.