Predicting Frequency and Claims of Health Insurance with Machine Learning Techniques

Pedro Gonçalves
pedro.o.c.goncalves@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

May 2021

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, insurance costs

1. Introduction

In the health insurance industry and, particularly, in Multicare, since 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 indica-

tors 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.

$$Loss\ Ratio = \frac{Claims\ Costs}{Total\ Earned\ Premiums} \quad (1)$$

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:

$$Claims\ Costs = Reported\ Claims + IBNR \quad (2)$$

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.

2. Baseline

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.

A time series is a set of observations x_t where each of them is recorded at a given time t. [9]

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.[5]

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.[5]

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

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

$$X_t - \phi_1 X_{t-1} - ... - \phi_p X_{t-p} = Z_t + \theta_1 Z_{t-1} + ... + \theta_q Z_{t-q} \quad (3)$$

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.[5]

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.
- 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.

  Grubbs's test is defined by the following hypothesis:

  H_0 : The data set has no outliers.

  H_1: The data set has 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.[4]

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.

  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 $\{X_t\}$, $\gamma_X(t+h,t) = \mathrm{Cov}(X_{t+h}, X_t) = E[(X_{t+h} - \mu_X(t+h))(X_t - \mu_X(t))]$ is independent of t for each h. [5]

  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. [2]

  On the other hand, in the ADF test, the null hypothesis is the data having a unit root. [11] 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.[5]

  In a time series, differencing is a method of transforming a non-stationary time series to make it stationary. [1] It consists of subtracting consecutive observations.

$$X_t^{'} = X_t - X_{t-1} \tag{4}$$

  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.

$$X_t^{(n)} = X_t^{(n-1)} - X_{t-1}^{(n-1)} \tag{5}$$

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

$$X_t^{'} = X_t - X_{t-m} \tag{6}$$

  where m is the duration of the season.[7]

  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.[5]

- 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 $d = 0$.

3. Methodology

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.

$$Loss\ Ratio = \frac{Claims\ Costs}{Total\ Earned\ Premiums} \tag{7}$$

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:

$$Claims\ Costs = Reported\ Claims + IBNR \tag{8}$$

As we showed in the previous section, the forecast for IBNR is made separately, 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*).

$$Reported\ Claims = Medium\ Cost \times Total\ Number\ Claims \tag{9}$$

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.

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

3.1. Decision Trees

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. [8]

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.

3.2. Random Forests

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:

$$\hat{P}(y = k|x) = \frac{1}{N} \sum_{i=1}^{N} P_i(y = k|x) \qquad (10)$$

3.3. 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.

3.4. 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. [10]

3.5. Claim Catalogs

In this section, we introduce 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 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

3.6. Feature Importances

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. [3]

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

3.7. 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, 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:

$$Reported\_Claims\_Pred\_Client_i\_Annuity_j = \sum_{k=1}^{N} N\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j \times C\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j \tag{11}$$

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

$$Loss\_Ratio\_Pred\_Client_i\_Annuity_j = \frac{Reported\_Claims\_Pred\_Client_i\_Annuity_j}{Total\_Earned\_Premiums\_Client_i\_Annuity_j} \tag{12}$$

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.

4. 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:

$$Error(\%) = \frac{Claims\ Forecasted - Claims\ Real}{Claims\ Real} \times 100\% \quad (13)$$

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 $Reported\ Claims$.

$$Reported\ Claims = Loss\ Ratio \times Total\ Earned\ Premiums \quad (14)$$

4.1. Forecasting Pipeline

The forecasting architecture is presented in section 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.

$$TRC = \sum_{i=1}^{10} NC_i \times MC_i \quad (15)$$

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.

After performing a performance comparison analysis of the three different classifiers for the number of claims and the three regressors for the cost, the best ones for the forecasting pipeline were then chosen accordingly to this results and are the following:

Table 2: 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).

4.2. 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:

$$Reported\_Claims\_Pred\_Client_i\_Annuity_j = \sum_{k=1}^{N} N\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j \times C\_Claims\_Pred\_Catalog_k\_Client_i\_Annuity_j \quad (16)$$

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:

$$Error(\%) = \frac{Claims\ Forecasted - Claims\ Real}{Claims\ Real} \times 100\% \quad (17)$$

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.

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.

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

Table 3: 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.

4.3. 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 [6].

$$Price\_Client_i\_Annuity_{j+1}\_ARIMA = Total\_Claims\_Client_i\_Annuity_j \times 1.0101 = (Total\_Claims\_9Months\_Real + Total\_Claims\_3Months\_ARIMA) \times 1.0101 \quad (18)$$

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

$$Price\_Client_i\_Annuity_{j+1}\_NewModel = Total\_Claims\_Client_i\_Annuity_j \times 1.0101 = (Total\_Claims\_9Months\_Real + Total\_Claims\_3Months\_NewModel) \times 1.0101 \quad (19)$$

Since we have the real total amount of claims verified in annuity $j+1$, 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.

$$Difference = Price\ Estimation - Real\ Price \quad (20)$$

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

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

5. Conclusions

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

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 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.

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).

References

[1] António Pacheco Pires. Notas de Séries Temporais, 2001.

[2] 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.

[3] Max Kuhn, Kjell Johnson. Applied Predictive Modeling, 2013.

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

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

[6] 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.

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

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

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

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

[11] William H. Greene . Econometric Analysis, 1997.