Sales forecast

Matheus Gonçalves Ferreira.

June 06, 2021

Machine Learning Engineering NanodegreeCapstone Project Report

overview

In the sales industry, especially in the area where the project is being focused which is the commercialization of health plans, one of the most important biases for a good commercialization of plans is statistics based on the forecast and sales of commercialized products so that, based on this, decisions and strategies are made for greater sales volatility. Based on this, the architecture and parameterization of a neural network based on Machine Learning was decided, using linear regression comparison, Random Forest Regression, XGBOOST, LSTMs and time series prediction using ARIMA.

I work in a company responsible for sustaining and developing a commercialization application of health plans, having as customers and utilities of the system the companies responsible for the distribution of health plans and / or companies that own and pioneer health plans in Brazil. Based on my professional knowledge and in order to preserve the sales data, fictitious data were created in order to feed this project.

Problem statement

Currently the companies and operators responsible for the distribution of health plans compose their sales strategies based on past sales control and containing future manual forecasts and with high and uncertain error scales. The need for the composition of this strategy is the fact that the composition and training of teams of brokers, having goals and targets to be met.

In the COVID-19 pandemic, sales of health plans overloaded and manual sales forecasting became unfeasible to be calculated and predicted manually, in cases where, at the end of a statistical calculation to predict sales for the next month, the current month of the calculation had already entered into force.

The company in which I provide services is always focused on automating the sales and administrative processes of our customers, aiming at this I had the idea of creating an artificial intelligence based on Machine Learning to automate the process of sales forecasts of health products.

strategy

Because it is a future forecast and based on inaccurate manual calculations, I relied on the comparison by processes of 5 prediction models, they are:

- LSTM (Long Short-Term Memory) - Recurrent Neural Network Architecture (RNN) used in deep learning that "remembers" values at binary intervals. It was adopted because it is adequate to predict time series with unknown duration time intervals.
- XGBOOST - Decision tree-based machine learning algorithm that uses a Gradient Boosting structure. It was chosen because it is an appropriate algorithm for dealings involving tabular data (DataFrames) and because it has an adequacy for processing large data sets.
- Linear Regression – It is a basic type of predictive analysis ideal for examining a set of predictor variables, thus treating an initially unestimated value.
- Random Forest Regression – It is a supervised learning algorithm that is based on the learning method together and on many decision trees. Random Forest is a Bagging technique, so all calculations are performed in parallel and there is no interaction between decision trees when constructing them. RF can be used to solve classification and regression tasks.
- ARIMA - Enables the prediction of a time series using past values of the series.

Metric

To analyze this model will be used the 5 models of predictions mentioned above, so that, based on them, can be extracted the models with better and most adequate prediction. For the calculation basis of these models, the following metrics are being used:

- RMSE (Root Mean Squared Error) – calculates the average quadratic root of errors between actual values and predictions.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2}$$

- Mean Absolut Error (MAE) - Calculates the absolute mean error between observed values and predictions.

$$\mathrm{MAE} = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

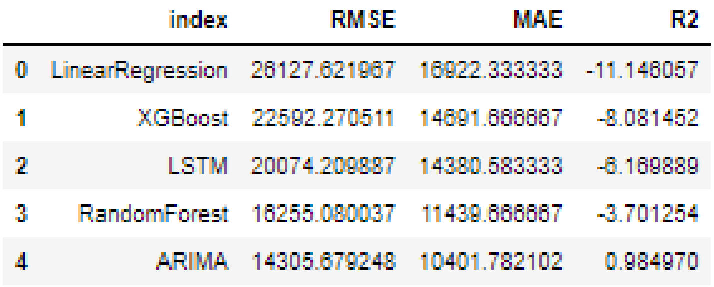
- R2 (Coefficient of determination) - Measure of adjustment of a generalized linear static model, to the values observed in a random variable.

$$R^2 = \frac{SQR}{SQT} = 1 - \frac{SQE}{SQT} = \frac{\widehat{\beta}_1 \sum_{i=1}^{n}(x_i - \bar{x})Y_i}{\sum_{i=1}^{n}(Y_i - \bar{Y})^2},$$

$$R^2 = \frac{\widehat{\beta}_1 \sum_{i=1}^{n} (x_i - \bar{x}) Y_i}{\sum_{i=1}^{n} (Y_i - \bar{Y})^2} = \frac{\sum_{i=1}^{n} (x_i - \bar{x}) Y_i \sum_{i=1}^{n} (x_i - \bar{x}) Y_i}{\sum_{i=1}^{n} (x_i - \bar{x})^2 \sum_{i=1}^{n} (Y_i - \bar{Y})^2} = \frac{\left(\sum_{i=1}^{n} (x_i - \bar{x}) Y_i \right)^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2 \sum_{i=1}^{n} (Y_i - \bar{Y})^2}.$$

$$0 \leq R^2 \leq 1.$$

For this we will have:



	index	RMSE	MAE	R2
0	LinearRegression	26127.621967	16922.333333	-11.146057
1	XGBoost	22592.270511	14691.666667	-8.081452
2	LSTM	20074.209887	14380.583333	-6.169889
3	RandomForest	16255.080037	11439.666667	-3.701254
4	ARIMA	14305.679248	10401.782102	0.984970

Data Exploration

For analysis of the training and test data, we will have 3 distinct CSV files, the train.csv file, containing the date, store, item and sales data, the test file.csv containing the date, store, and item data, the real_base.csv containing the sales data.

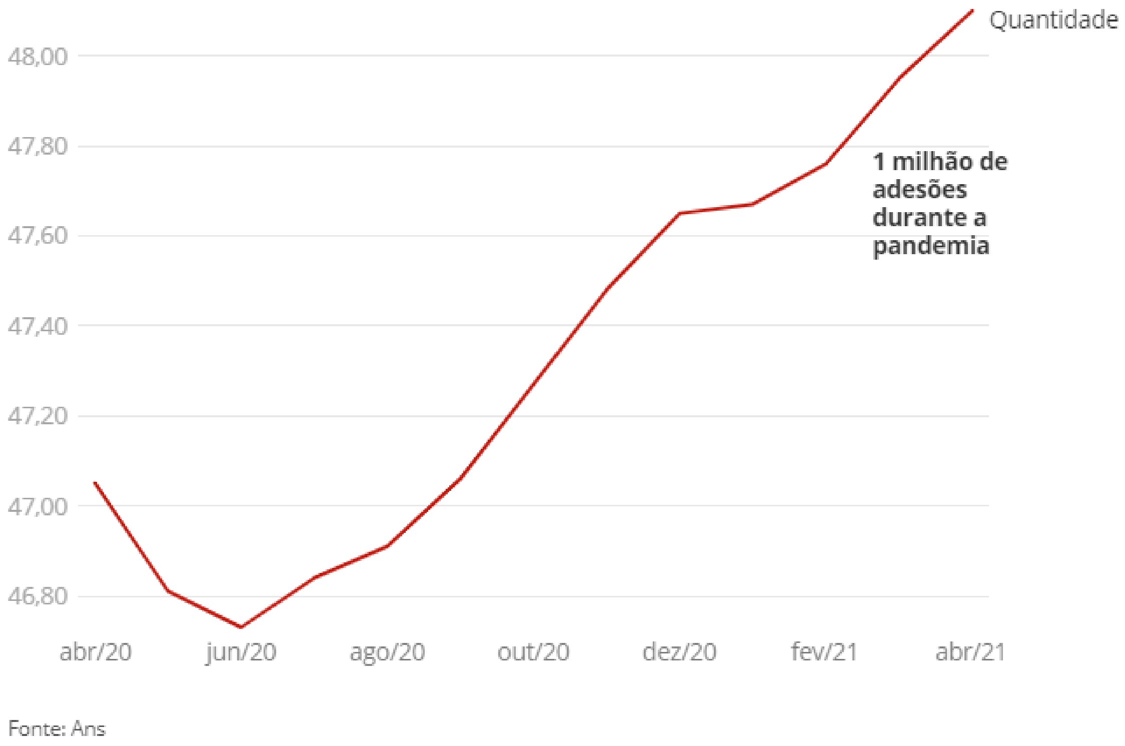
These files were architected with fictitious data based on actual sales data for exploration and analysis of the models.

- Train.csv - has the training data of the model.
  - Date – Date on which sales were made, these dates are repeated according to the item and store wedding

    - Store – Each health insurance sales company has a number of carriers to be sold their products, this set of data encompasses the operators of each plan and sold for each day.
    - Item – Each product has its own identifier called "ANS Registration", which is the record contained in the National Health Agency of Brazil, this registration is unique and unique to each health plan. In order to preserve the integrity of the original data, this record has been replaced and fed by a generic number.
    - Sales – Amount of sales of the distinguished products, this sales amount is filtered from the wedding of the above items.
- Test.csv - You have the model test data.
    - Id - Own and unique identifier of the dataset (not to be confused with item)
    - Date – Date on which sales were made, these dates are repeated according to the item and store wedding
    - Store – Each health insurance sales company has a number of carriers to be sold their products, this set of data encompasses the operators of each plan and sold for each day.
    - Item – Each product has its own identifier called "ANS Registration", which is the record contained in the National Health Agency of Brazil, this registration is unique and unique to each health plan. In order to preserve the integrity of the original data, this record has been replaced and fed by a generic number.

- Real_base.csv - You have the actual sales data.
  - Id - Own and unique identifier of the dataset (not to be confused with item)
  - Sales - Amount of sales of the distinguished products.

The training sales data are increasing and ascending in order to simulate the estimate of real growth of the commercialization of health plans at the time of COVID-19. The growth curve of a real customer's health plan sales can be seen in the simulation below (restricted actual data is not in effect).



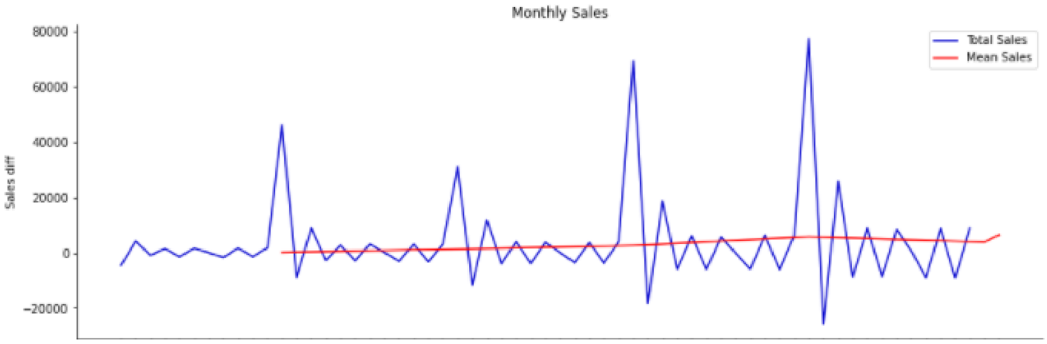
Source: ANS and Globo

Translation: One million seizures during the pandemic.

methodology

Initially, the data will be loaded and transformed into a structure for using the models, each row of data having the sales estimates of a single day. To combine all data by monthly sales, initially I process and transform all data from the 'store' column to numerical data, in order to improve functionality and data management, later a methodology of joining data by date and sum of your sales will be adopted.

The difference between sales in each month and added to the data frame will then be calculated as a new column, so that the data can become stationary.



Later the models will be configured, for the ARIMA model the 'date' columns will be required and the sales diff, for the other models, a table will be created determining,based on the 12 months, their salesresources, thus generated a DataFrame with 13 columns, one for each of the 12 months and the column of the independent variable, the sales difference. A supervised methodology will be adopted for the composition of the data.

After that there will be an ARIMA DataFrame and a Supervised One.

modelling

For modeling, auxiliary functions will be used, these functions are contained in the file predict.py, tain.py, test.py and model.py. For data modeling, three distinct approaches are used:

- Regressive Modeling (Linear Regression, Random Forest Regression, and XGBOOST) – A scikit-learn library adjustment prediction structure approach (library references are distinct in code for better understanding) will be used, thus setting up a base modeling structure for requesting each model.
- LSTM – The LSTM used will be simple and used for additional accuracy, seasonal features and additional model complexity.
- ARIMA - The SARIMAX statistics model pack is used to train the model and generate dynamic forecasts.

Finally, the models developed to be determined will be compared to the ones with the best performance, for this it is examined the quadratic root of the mean error and the mean absolute error, although they have differences in the mathematical calculation base, it is ideal for comparing the performance of the model.

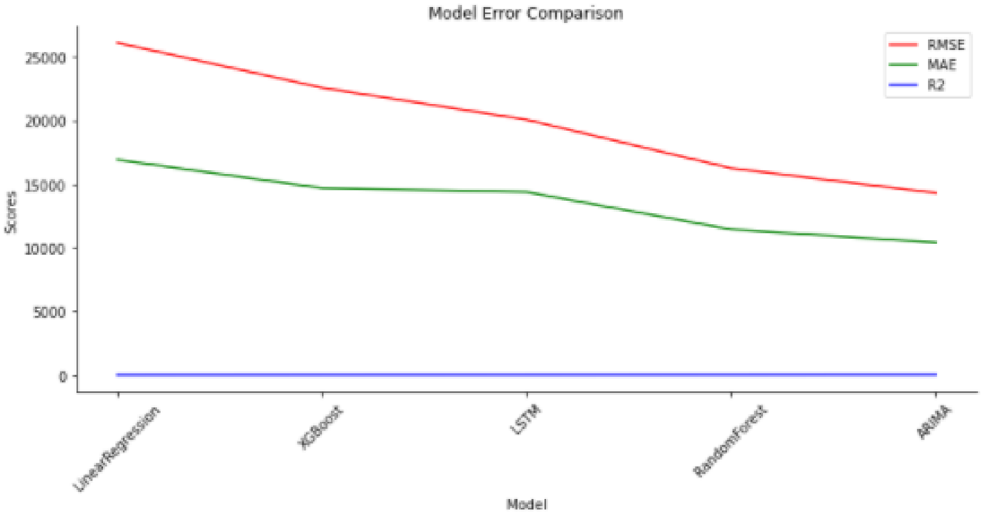
- RMSE (Root Mean Squared Error) – calculates the average quadratic root of errors between actual values and predictions.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2}$$

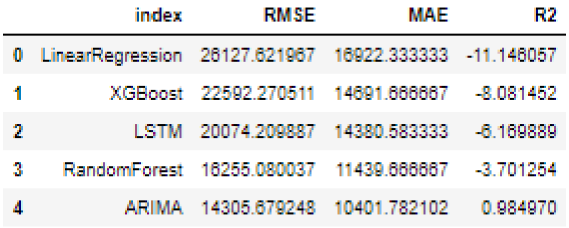
- Mean Absolut Error (MAE) - Calculates the absolute mean error between observed values and predictions.

$$\mathrm{MAE} = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

From these auxiliary functions, the scores of each model are calculated, being saved in a dictionary and preserved, in which it will be transformed and used for comparison of Pandas data. After these visualizations, it can be seen the graph below in which, as can be visualized, by degree of proximity, the ARIMA model is the one that was most effective in the training performed.



Thus, the forecasts described below can be seen.



	index	RMSE	MAE	R2
0	LinearRegression	26127.621967	16922.333333	-11.146057
1	XGBoost	22592.270511	14691.666667	-8.081452
2	LSTM	20074.209887	14380.583333	-6.169889
3	RandomForest	16255.080037	11439.666667	-3.701254
4	ARIMA	14305.679248	10401.782102	0.984970

Workflow

The project consists of two front lines, the Jupyter Notebook files, where we have:

- 01-clean_data_ead.ipynb – File containing all initial data preparation of modeling, data cleansing, data processing and initial parameterizations.
- 02-models.ipynb - File responsible for data modeling,preparation of models and Dataframes.
- 03-results.ipynb - Archive responsible for displaying and printing the data processed for comparison.

We also have the composition of the auxiliary files in Python to feed the Jupyter Notebook files described above, these Python files have functions that will be used in the Jupyter Notebook files and a Python file containing all external imports for use in the code.

- Train.py - File containing the functions responsible for the training bases of the model.
- Test.py - File containing the functions responsible for the test bases of the model.
- Predict.py - File containing all the functions responsible for predicting the models.
- Model.py - File containing all the functions responsible for modeling.
- Import_libs.py - File containing all external imports.

In the "date" directory you can view the files containing the data passed to the model in order to feed it and the files generated from the treaties of the models.

In the"model_output"directory, you can view all data charts generated by the modeling treaties.

conclusion

The stipulated training presented satisfactory data so that predictions with low amount of data are stipulated, so that it is necessary that the model is implemented to support more robust data and with greater volatility.

References

https://towardsdatascience.com/5-machine-learning-techniques-for-sales-forecasting-598e4984b109

https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.Grouper.html

https://www.ti-enxame.com/pt/python/como-agrupar-por-mes-partir-de-um-campo-data-usando-python-pandas/833749839/

https://strftime.org/

https://pandas.pydata.org/pandas-docs/stable/user_guide/timeseries.html#offset-aliases

https://www.kaggle.com/c/demand-forecasting-kernels-only https://www.linkedin.com/pulse/rmse-ou-mae-como-avaliar-meu-modelo-de-machine-learning-rezende/?originalSubdomain=pt

http://www.portalaction.com.br/analise-de-regressao/16-coeficiente-de-determinacao

https://g1.globo.com/economia/noticia/2021/05/21/planos-de-saude-ganham-mais-de-1-milhao-de-clientes-na-pandemia-e-atingem-maior-nivel-em-quase-5-anos.ghtml