Store Sales - Time Series Forecasting

1st Ismet Okan Celik
Stevens Institute of Technology
icelik@stevens.edu

2nd Swapnali Patki
Stevens Institute of Technology
spatki1@stevens.edu

3rd Rishab Katteri
Stevens Institute of Technology
rkatteri@stevens.edu

***Abstract*—We will be using a time-series related data set to forecast store sales on data from Corporación Favorita, a large Ecuadorian-based grocery retailer. More specifically, we will try to build an adequate ARIMA model, use multiple regression, and optimal feature engineering that more accurately predicts the unit sales for thousands of items sold at different Favorita stores. Using test and training data, and with a couple of predictions, we will see what features are important or what features we should engineer in order to maximize profit and to predict when it will happen when Corporación Favorita reaches a certain sales point. We will proceed with whichever model would prove to be the best in forecasting, but we assume that a robust time series forecast might prove to be the best way to model this data.**

I. Introduction

After pre-processing the data and handling the training/testing process through different learning system, we can use this to best predict certain goals for this company. Having a solid understanding of time series forecasting through ARIMA models and its various time series models would aid us in giving a detailed look into this analysis. We also have great motivators like multiple regression and logistic regression as some ancillary processes in order to best describe relationships between predictor variables. Adding a residual analysis for both cases may also motivate other processes, such as influence diagnostics. Finally, we can try and test how it will work against out-of-sample data, so using an optimal k for k-fold cross-validation might prove useful in understanding the nature of the market.

The provided data from this Kaggle competition will be used for this project. The data can be found through the following link: https://www.kaggle.com/c/store-sales-time-series-forecasting. It mainly contains the transaction records, any holiday events, store locations with the training and testing dataset.The data might need a bit of cleaning and preprocessing. We look forward to gathering insight on foresight and see what techniques we can use at our disposal from class in a constructive and innovative way.

II. Related Work

According to our research, it appears as if this Kaggle dataset competition is more novel than we initially thought. There may not be many existing solutions due to its overall rigor and lack of proper guidance. The accuracy of the prediction is an essential aspect of sales forecasting. As a result, several attempts have been undertaken to improve the accuracy of this procedure. After comparing the actual sales with the expected results, the accuracy is calculated using several error measuring methods such as RMSE and MAPE in publications in this subject. Some relevant works are reviewed in this section.

2.1. Classical time-series forecasting methods

Each situation may require a distinct technique in order to cope with time-series forecasting. Moving Average (MA) is one of the most straightforward prediction approaches for time-series with no discernible seasonal trend. A more advanced variant of MA known as Auto-regressive Integrated Moving Average (ARIMA) has been employed in various articles Another traditional forecasting approach is seasonal ARIMA (SARIMA). This method has proven to be effective in a variety of applications, including forecasting tourism demand and estimating vehicular traffic flow.However, due of its linear structure and inability to detect nonlinear and highly volatile patterns, SARIMA has been shown to have limits in terms of prediction.

2.2. Artificial Neural Networks time-series forecasting methods

Other approaches for predicting include Artificial Neural Networks (ANN). Aggregate time-series are vast collections of time-series aggregated using various approaches, and they are typically more accurate than disaggregate predictions. Winters' exponential smoothing and Box Jenkins approaches fared worse than ANN prediction. In addition, multivariate regression exhibited the lowest MAPE average of all the methods. Verstraete, Aghezzaf, and Desmet (2019) created an ANN-based framework for forecasting the influence of short- and long-term weather uncertainty on retail items. ANN models have been popular techniques for sales forecasting especially because of their flexibility for detecting patterns in data.

2.3. Deep learning

Deep neural networks are based on the use of Multilayer Perceptrons (MLPs) with additional hidden layers to mimic more complex tasks. Solar power forecasting, stock price prediction, and short-term traffic forecasts have all benefitted from this effective ANN-based technique. A Convolutional Neural Network (CNN) is a type of deep network that has lately been employed in time-series forecasting. CNN learns useful features from data automatically and then uses this approach to anticipate sales. In comparison to the used deep learning models, machine learning models such as XGBoost produced superior outcomes.

–

III. Our Solution

A. Description of Dataset

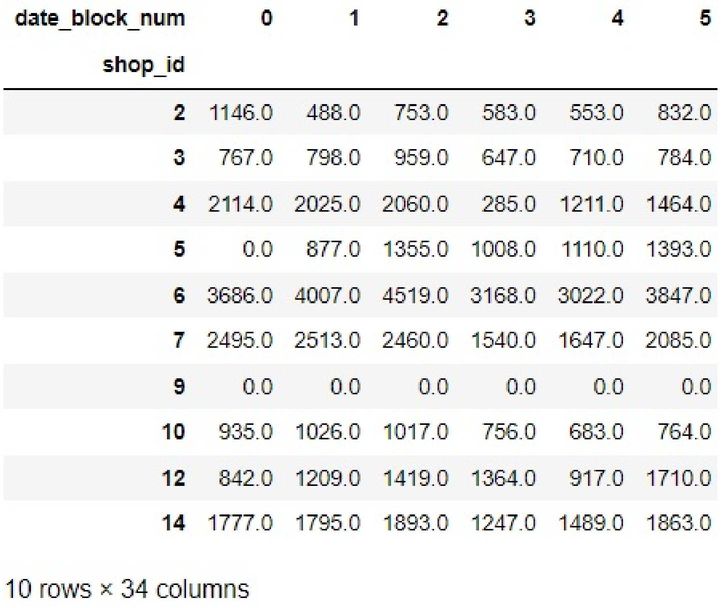
We plan to forecast sales for thousands of product families sold in Ecuador's most popular retailers. Dates, shop and product information, whether the item was being marketed, and sales statistics are all part of the training data.

The training data comprises of time series of features store number, family, and onpromotion as well as the target sales. store number file identifies the store at which the products are sold. family file identifies the type of product sold. sales file gives the total sales for a product family at a particular store at a given date. Fractional values are possible since products can be sold in fractional units (1.5 kg of cheese, for instance, as opposed to 1 bag of chips). on promotion file gives the total number of items in a product family that were being promoted at a store at a given date.

B. Machine Learning Algorithms

We approach the problem with a proper mindset with time series as such, so we tried and built our initial solution around a Box-Jenkins model. First, we look at the Autocorrelation Plot and the Partial Autocorrelation Plot, or named ACF and PACF plots, to check stationarity within our original dataset. If it is not stationary, we would have to utilize 3 methods to help make the dataset stationary. Through differencing the dataset, we would take each of the differences of the consecutive values and model those differences. Through detrending, we would try and model the noise/trend created by the original dataset and subtract that from our original dataset, thereby taking the significant trend out of the model and hopefully stabilizing the nonstationarity. Through transforming the data, we would see if any of the models follow a certain parent function and stabilize the variance in order to make it stationary. Regardless of whichever technique we use, we would then administer a Dicky-Fuller test to confirm whether the time series is stationary. We then move onto finding the proper ARIMA model, and by using the ACF and PACF plots, we would be able to detect how many MA and AR terms we need through observing the amount of bars that are over the confidence interval. If not, we would also administer an ARMA best subsets model or an Extended Autocorrelation Plot, or an EACF, to completely make sure that we are not forgetting any relevant information. This would be our next step, as we reduce parameter redundancy and get a preliminary model to test. We proceed with parameter estimation using Maximum Likelihood, Least Squares, or Method of Moments, and we check the significance of the estimated parameters. We can also compare the models using the log likelihood, Akaike Information Criteria, or the Bayesian Information Criteria. Finally we run a residual analysis and using analyses like the Q-Q plot and Shapiro-Wilk test, we can see if any last minute changes to our model is necessary. With this out of the way, we can finally get to modelling the actual original dataset. We think most of this is necessary to solve this problem as it is the standard in the real life world in forecasting, so it would only serve as a good starting point to analyze trends.

If we use multiple linear regression and logistic regression, we could add a bit of clarity to the forecasting we mentioned above. For example, if we used multiple linear regression, we could tell what the best independent features are in predicting our response variable of sales. Using stepwise regression, forward selection, backwards selection, best subsets selection, and other techniques to choose the best predictor variables, we would best optimize sales and predict future sales. However, to avoid overfitting, we would also run a quick cross-validation according to the proper k. Along with some additional feature engineering to avoid multicollinearity and to reduce the hypothesis space, we would be able to make more sense of the response variable. Logistic regression might also be helpful as we can see if we can model how a person would be likely in purchasing a certain product. Since we have around 44 stores to work with, we can see if a person is likely to buy the best 3 products through a binary response, Yes or No. If we follow the same procedure as we did with multiple linear regression, we would obtain a model which would best identify future responses to different products and whether sales are expected to go up or down depending on customer influence.



date_block_num	0	1	2	3	4	5
shop_id						
2	1146.0	488.0	753.0	583.0	553.0	832.0
3	767.0	798.0	959.0	647.0	710.0	784.0
4	2114.0	2025.0	2060.0	285.0	1211.0	1464.0
5	0.0	877.0	1355.0	1008.0	1110.0	1393.0
6	3686.0	4007.0	4519.0	3168.0	3022.0	3847.0
7	2495.0	2513.0	2460.0	1540.0	1647.0	2085.0
9	0.0	0.0	0.0	0.0	0.0	0.0
10	935.0	1026.0	1017.0	756.0	683.0	764.0
12	842.0	1209.0	1419.0	1364.0	917.0	1710.0
14	1777.0	1795.0	1893.0	1247.0	1489.0	1863.0

10 rows × 34 columns

Fig. 1. Orginized Data (Number of Shops:44, Number of Months:33)

C. Implementation Details

Looking at our initial dataset, we realized it was unorganized insofar we couldn't start any significant analysis. We had to use a few data cleaning techniques to get what we wanted, which was a standard time series dataset in which we could also perform regression methods on.

So far, we found Non-Active stores on the dataset and deleted these stores. After that, we turned the unorganized data into proper time-series dataset, replace the NaN values with zero, and change the negative to positive values. Visualized the dataset for each store and their sales to understand the time-series. Each line represents one store. We can see from visualized data that time-series are not stationary; for that reason, we applied some techniques we mentioned earlier to make the time-series stationery. One of them was to take differance between values.

After we compute first order difference transformation, data

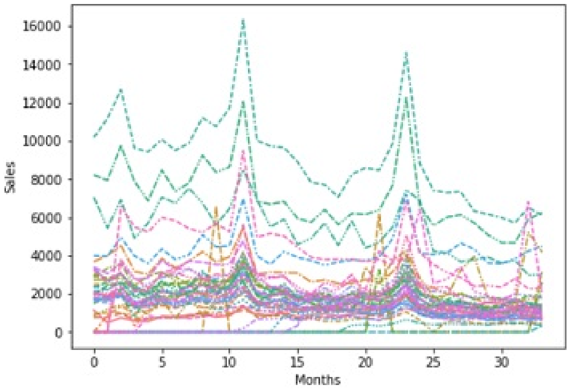


Fig. 2. Monthly Sales for Every Individual Stores

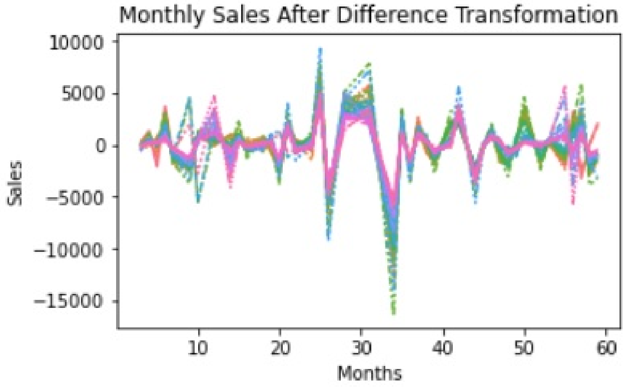


Fig. 3. Monthly Sales Difference Transformation

seem stationary, but as we can see on the figure there are some big spikes on the data. That's why we are not really sure whether it is stationary or not. For this reason we computed Auto Correlation Function (ACF) and Partial Auto Correlation Function (PACF) we plotted Auto Correlation Function and Partial Auto Correlation Function to analyze the data better.

After we plotted ACF and PACF for each store, we saw that

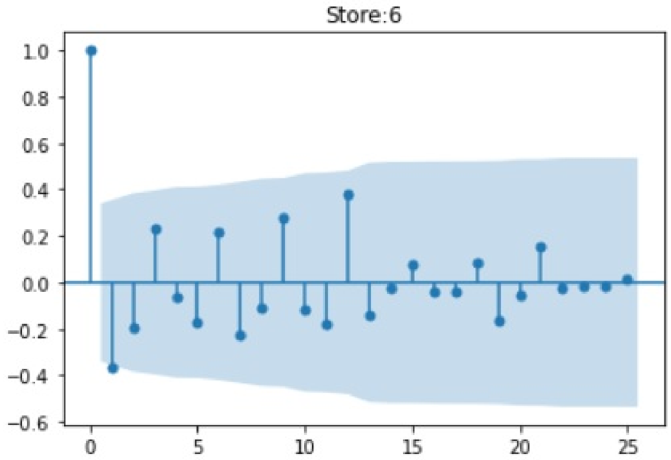


Fig. 4. Auto Correlation-Store:6

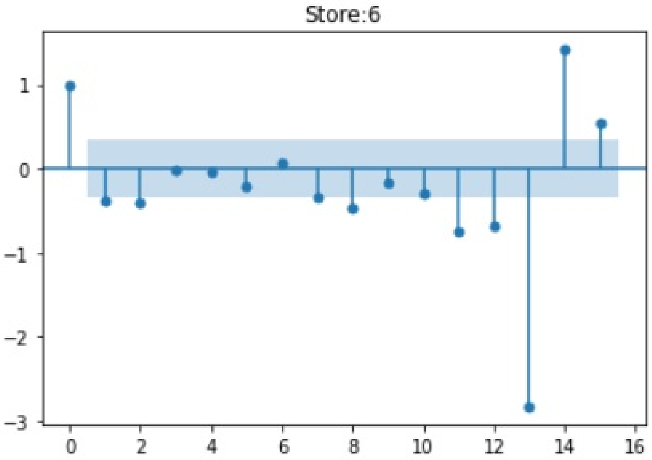


Fig. 5. Partial Auto Correlation-Store:6

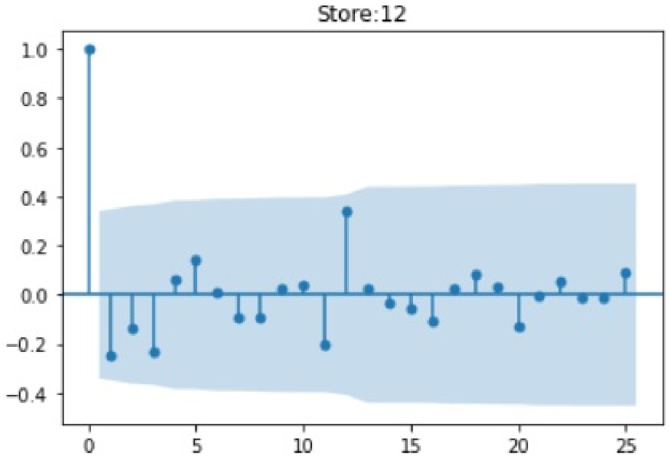


Fig. 6. Auto Correlation-Store:12

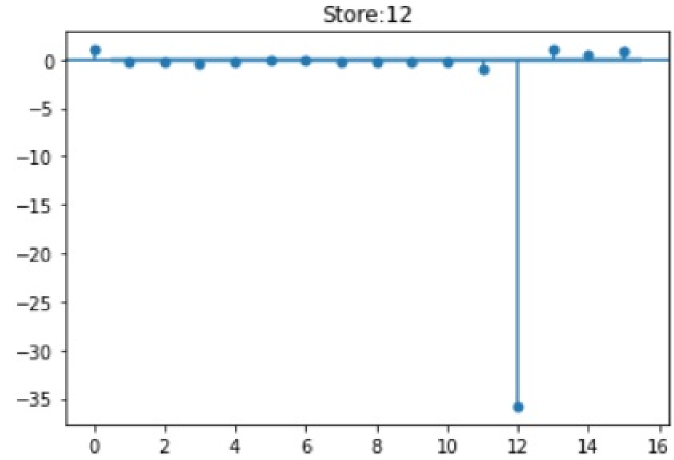


Fig. 7. Partial Auto Correlation-Store:12

some stores had big spikes on PACF plots. We plan to apply residual analysis to pick the best ARIMA model, which may work best for all the stores, or try to pick different ARIMA models for each shop.