Psychic Predictions – A Retail Sales Forecasting System

BrainStation Capstone Project

Rishab Gupta

Abstract

Sales forecasting is an essential component of consumer-oriented markets such as retail, fashion and electronics. In such industries, where demand is volatile, recognizing and adapting to changing consumer needs serves to provide a competitive edge over other players in the market. While many factors may be at work when adjusting business strategies to adhere to consumer demand, one major aspect is historical sales. In this paper, we implemented the traditional ARIMA model followed by Facebook's Prophet model to analyze historical retail sales from a Brazilian E-commerce public dataset, where we were able to predict up to 60 days of sales into the future.

Introduction

Consumer-oriented markets such as the retail industry is driven by consumer demand and product sales. For any product, the longer it remains on the shelf, the more loss it causes the business owner as it takes up shelf space and reduces inventory capacity which may otherwise be used for income generating merchandise. As such, sales forecasting can have a significant impact on the success and performance of a company, giving businesses invaluable insights within this everchanging space and aid business owners and management in maximizing and extending profits over the long term *(1).* Furthermore, where consumer demand can change year over year, month over month and even day over day, with additional factors such as holidays, seasonality, and economic conditions adding on complexity, understanding the retail space has become more crucial than ever before *(2, 3, 4).* Although, a vast number of factors can be involved in understanding the future of a product in the market, one key factor is historical sales. Given how a product has sold in the past, its demand in the present, what is the likelihood the product will continue to sell in the future. This understanding forms the backbone of timeseries analysis, which uses existing sales data to predict future events, and more importantly, future consumer behaviour *(5).*

In this report, we will highlight the implementation of ARIMA and Facebook Prophet on a Public Brazilian E-Commerce Dataset obtained from Kaggle. Given information on over 100,000 retail transactions made between the years of 2016 and 2018, we have worked to employ timeseries analysis to predict future sales of products within the "Bed Bath Table" category. While we were limited by the lack of two full cycles of data and extensive domain knowledge within the Brazilian E-Commerce space, our analysis here has shown to predict sales up to 60 days into the future, demonstrating the rich insights that can be gained from such simple yet robust machine learning techniques.

Project Goal

Can we use machine learning to predict future sales of products given historical transactions?

Data Processing, Cleaning and Exploration

A Brazilian E-Commerce Public Dataset was obtained for this project from Kaggle which highlighted over 100,000 retail transactions made on the Olist website, distributed over 9 csv files. Olist is the largest department store in the

Brazilian Marketplace and functions much like amazon, giving consumers the opportunity to buy products from various companies across Brazil and have the companies (sellers) ship the items to the customers after purchase.

During the initial preprocessing, cleaning and EDA steps, we worked to compile all the information into one master dataframe. With information on order status, customer reviews, seller/customer location, product names and categories, payment methods, purchase timestamps, and payment value, this was a rich dataset offering much more information than necessary for the project goal on hand. Because we initially intended to use customer reviews as a feature in the dataset to predict future sales, the data compilation steps worked to gather this information in addition to the sales and timestamp information, yielding review scores, payment value and order purchase timestamp for the final dataset. Order status was used to filter out only the order that were delivered, and payment method was used to exclude orders made using a voucher to ensure that we truly focused on sales for which consumers used their own money to make the purchase.

With how the information was organized in the files, while we did not have null values to impute, we were given duplicate rows for each order in which the customer bought more than 1 quantity of the item in question. Thus, during processing, all duplicates were also removed to yield a dataset with ~95,000 orders.

This information was then grouped by product category, and the category with the greatest number of datapoints (purchases) was chosen to build a timeseries model for future sales predictions. Due to limited datapoints for individual items within a category, we were forced to group information by product category, at which point, we lost all variance in product review scores. Thus, review scores were dropped from the final dataset.

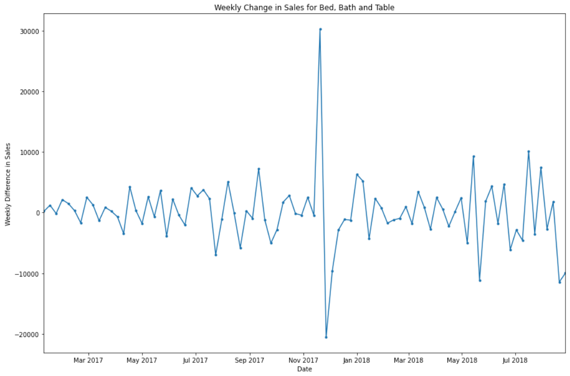
Furthermore, due to a lack of evenly spaced time data, we also worked to group sales by week, to generate an evenly spaced dataframe for ARIMA modelling. Because this is not a requirement for Facebook Prophet, given its ability to handle missing values, we also built a separate dataframe, with the same information, but with time grouped by 'Date', yielding 2 final sales vs. time dataframes ready for modelling.

For a detailed transcript of the data exploration steps taken, please refer to the documentation within the "Preprocessing, Cleaning and EDA" Jupyter Notebook.

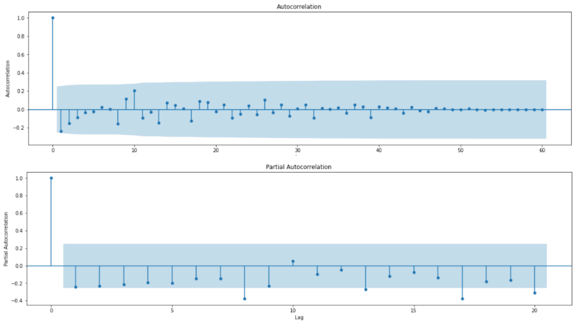
ARIMA Model Sales Forecasting

The first model that we implemented was the traditional Autoregressive Integrated Moving Average (ARIMA) model, where we attempted to predict weekly sales over a 6 month period.

In addition to the requirements of having evenly spaced datapoints in time, the ARIMA model also requires our data to be stationary prior to timeseries analysis. We implemented 1st order differencing to yield weekly change in sales as a method of creating data stationarity. Stationarity was then confirmed using the Augmented Dickey Fuller test, before moving fourth with modelling.



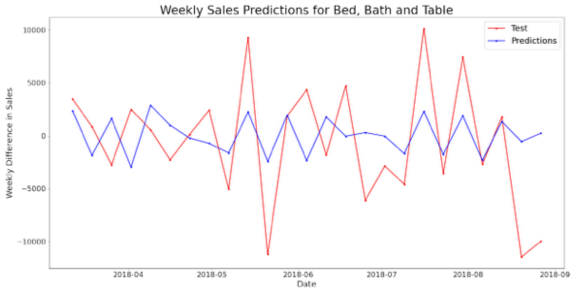
The data was next split into train and test sets (70% train, 30% test), after which autocorrelation and partial autocorrelation was determined to select the Autoregressive parameters.



While, we did not have any lag terms for which autocorrelation is statistically significant, we plotted the partial autocorrelation with a max lag of 20 as this is the point where much of the autocorrelation variance has diminished. From this, we did indeed find lag terms (8, 13, 17, 20) for which the partial autocorrelation was beyond the confidence intervals, suggesting statistical significance.

Following this, we were next able to build our ARIMA model. Provided we did not have tests such as the autocorrelation and partial autocorrelation above to determine our integrative and moving average terms for the ARIMA model, we iterated over various values for the integrative and moving average terms, as well as the different trend values, to determine the ideal combination of AR, I, MA and trend. These iterations were trained on the train dataset, followed by an evaluation of the root mean square error (RMSE), on the test dataset, where we selected for the combination of values which yielded the smallest RMSE.

```
MAPE: 121.13%
```



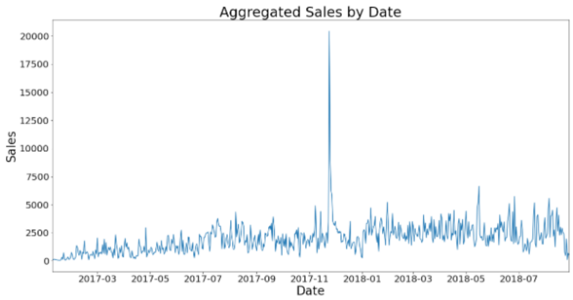
The RMSE metric was used over the mean absolute percentage error (MAPE) due to RMSE's ability to apply weights to error values, where the larger error value has a higher weight, which in essence amplifies the effect of a large error. Thus, this would mean that we find a model where the predicted values are as close to the actual sales as possible, which may not be the case for MAPE as it does not apply weights to each error term. However, we did report the MAPE value for the best model, as it is the most intuitive to understand.

From the best model (above), we can see that it is does not perform very well. While it does tend to follow the generic trend of the actual sales, the predictions are far from perfect. Even over the 1st couple weeks, the model struggles to predict sales accurately, which is concerning, as this would be where the model performs best with diminishing returns over time. Thus, while this was a great 1st step, we also implemented Facebook's Prophet prediction model to see if we can do better.

For a detailed transcript of the ARIMA Model implementation, please refer to the documentation within the "ARIMA Model" Jupyter Notebook.

Facebook Prophet

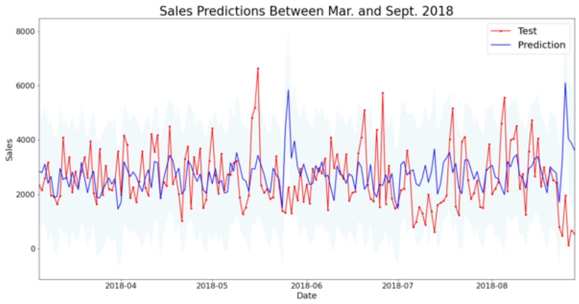
Prophet is not bound by the boundaries that we faced in ARIMA, namely the need to have evenly spaced datapoints and the lack of flexibility in seasonality optimization. This is especially important for our data as we were only left with 87 datapoints to implement the ARIMA model due to the need of evenly spaced data. By having the ability to look at sales by date, we were granted with much more detail in our dataset on which the model can be trained.



Although previously, in the ARIMA model, the large spike in the data was attributed to a spike in sales around Christmas time, as we had a weekly sum of the sales, it looks more like an outlier as we still see the same spike in this dataset, which was aggregated by date. During Prophet modelling, both the data with this outlier and without this outlier were modeled on to determine which dataset provides the best results. Interestingly, the data with the outlier included yielded the smallest MAPE value, suggesting that its retention, was somehow aiding in the model's ability to predict future sales.

The data was split into train and test sets (70% train and 30% test), followed by iterative modelling with the addition of parameters at each step to determine the most predictive model. With this data, we found that when accounting for weekly seasonality, quarterly seasonality and Brazilian holidays, we yielded the best model, with a MAPE value of 69.19%.

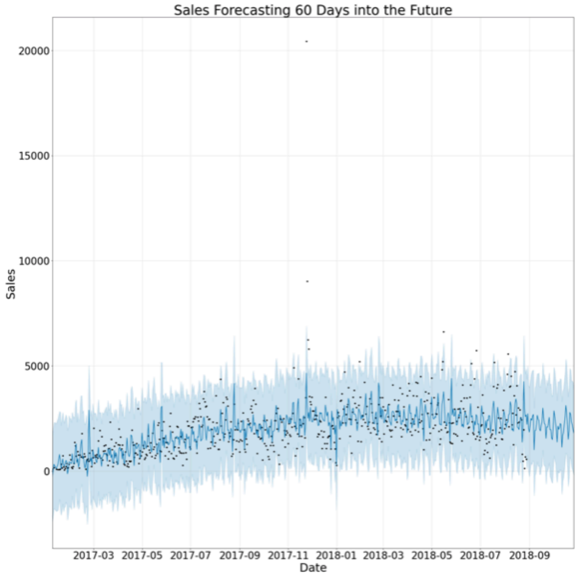
```
MAPE: 69.19%
```



Given the model's ability to handle missing values, we see here that Prophet gives us the opportunity to determine more daily sales forecasting which was not possible with the ARIMA model. While, we did find a model that had a smaller MAPE value over the entire 6 month long prediction period, it is important to note that the quality of predictions of these models do diminish over time, which is an unfortunate drawback of timeseries in general, that is unavoidable.

As such, to identify the best predictive model, a MAPE value was calculated for the 1st month of predictions, as this is where every model performed its best. Here, we were able to find that the model shown here, actually had the smallest MAPE value (27.17%) vs the other model which had a smaller MAPE for the entire 6 month period.

Finally, given the insights obtained here, the model was trained on the entire dataset, and a grid search was performed over various parameters (changepoint prior scale, seasonality prior scale, and holiday prior scale) to find the most optimal model which would be used to predict sales truly into the future. From our results, we found the optimal model with a MAPE value of 75.



Using the most optimal parameters for our model, we found that we were able to predict up to 60 of sales into the future, truly showcases the model's value in the industry.

For a detailed transcript of the Prophet Model implementation, please refer to the documentation within the "FB Prophet Model" Jupyter Notebook.

Findings

Going back to our original project goal, we have determined that we can indeed use machine learning techniques and historical sales data to predict future sales of products. While we are limited to near future predictions, due to diminishing predictive quality over time, our analysis thus far has shown that it is very possible to predict future sales. In our case, Facebook's Prophet model showcased the best potential in sales predictions, in part due to its ability to handle missing vales and be flexible with seasonality allowing us more control over how the model is fit to the data. This said, one drawback of such flexibility is the need to have significant domain knowledge in order to attribute the correct parameters on the model to yield results.

Conclusion

In conclusion, we have shown the implications of timeseries forecasting using a Brazilian E-Commerce dataset obtained from Kaggle. Despite not having two cycles worth of data, we were able to optimize Facebook's Prophet model to yield 60 days of future sales. For any business such insights can be crucial in maintaining a competitive edge in the marketplace where consumer demand is so volatile. Having a good understanding of a product's future in the market can truly aid a company in allocating budgets and managing inventory.

For next steps, we would love to gather more domain knowledge within the Brazilian E-Commerce space to further optimize the Prophet model. We would also like to attempt to build a recurrent neural network as another method for timeseries forecasting and see if its performance can yield better results than Prophet, especially for forecasting over a longer time period.

References

1. https://yourbusiness.azcentral.com/top-10-reasons-sales-forecasting-important-24818.html

2. Fissahn, J. (2001). *Marktorientierte Beschaffung in der Bekleidungsindustrie*. Dissertation, Münster University.

3. Thomassey, S. (2010). Sales forecasts in clothing industry: The key success factor of the supply chain management. *International Journal of Production Economics*, 128(2), 470–483.

4. Beheshti-Kashi, S., Karimi, H. R., Thoben, K., Lütjen, M., & Teucke, M. (2014;2015;). A survey on retail sales forecasting and prediction in fashion markets. *Systems Science & Control Engineering, 3*(1), 154-161. doi:10.1080/21642583.2014.999389

5. https://smallbusiness.chron.com/explain-forecasting-retail-37966.html