Capstone Project Report

Sales forecast for an E-commerce startup to help the business make better decisions.

olist
empowering commerce

Index:

1. Problem statement

Sales forecasting is challenging for any business because of the limited historical data, unprecedented external factors like weather, natural events, government policies and mostly because of the volatile nature of market.

1.1 Background on the subject matter area

While doing research I found these statistics data from Intangent. I am quoting some of the shocking figures from their blog:

'93 percent of sales leaders are unable to forecast revenue within 5 percent, even with two weeks left in the quarter.'

'80% of sales orgs DO NOT have a forecast accuracy of greater than 75%'.

These figures were collected from respected research institutions like CSO Insights and Gartner. Contrary to that, Aberdeen research has highlighted the importance of sales forecasting, I am quoting again:

'97% of companies that implemented best-in-class forecasting processes achieved quotas, compared to 55% that did not.'

1.2 Goal

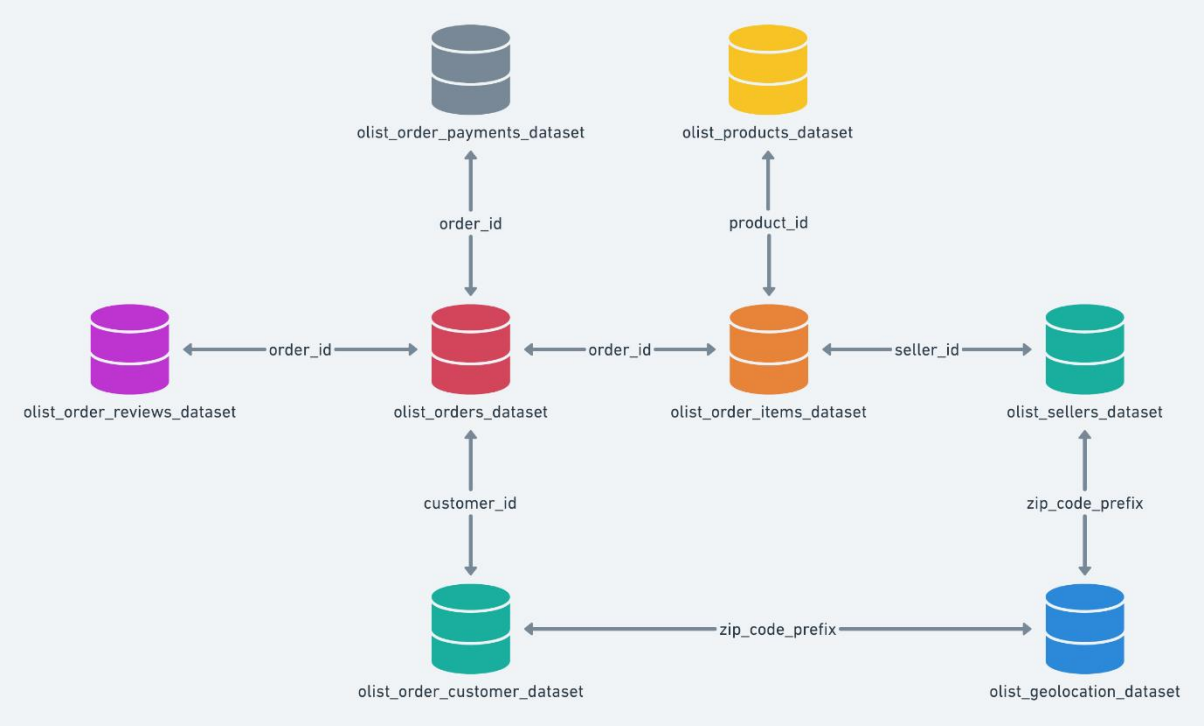
In my capstone project, I have tried to find out how the business is performing and have explored different machine learning techniques to do sales forecasting. My aim is to identify the best approach for forecasting for a ecommerce startup on a limited historical data.

2. Data collection and Data schema

The data is a public dataset available on Kaggle and was provided by Olist, which is an E-commerce startup business that was founded in 2015. The dataset has observations recorder from September 2016 to August 2018 with around 100k orders made at multiple marketplaces in Brazil.

Data Schema

The data is divided in multiple datasets for better understanding and organization here is the high-level schema provided by Kaggle.



There are eight individual CSV file. Each file was loaded and cleaned separately. Tables were merged one by one to create a master dataset with 110013 orders and 29 features. The dataset is huge to be captured in a screenshot, therefore I am just specifying high level features in our final merged data:

- Order details: Order id, customer id, seller id, product details, item bought, total order value, freight charge, review, purchase timestamp and estimated delivery date.
- Customer details: Customer location and customer id.
- Product attributes: Description word count, dimension, and weight information.
- Seller details: Seller location and seller id.

There was a Payment table with information about payment method used, but it was not included in final data wireframe.

I also scrapped the holiday information from the National Holiday of Brazil website.

2.1 Summary of cleaning and pre-processing

The data cleaning for this project primarily involved removing duplicates, imputing null values and merging all the individual tables to make a master table. Although I didn't use a lot of the features, but my aim was to preserve as much data point as possible

so that I can do detailed analysis on this data later from business and seller's perspective.

With regards to data imputation:

Imputing latitude and longitude: To impute the missing latitude and longitude coordinates for both customer and seller location, I tried to find the mean coordinate values for city. Since I had city and state information in our table, I tried finding the mean coordinates for city (assuming that cities of different states have different names) and used that to fill missing coordinates.

For **Orders table** I filtered only the orders that had status 'delivered' and removed all the other rows with different status. Since business is concerned only about the fully closed delivered order.

The **Products tables** had nulls in Product category column along with product description length, product name length and product photo quantity columns for same observations. I tried matching the weight, height, length and width of the rows which had NANs for product category name and product description length features **to the rows where I had all feature values.** I used the matching category values to fill the missing values. Where there was no match, I created another category 'Other' and used mean of the known values to fill missing. I then added column form another table which had **English translated names for all the product categories**.

Target: Sales amount

In order to do Sales forecast there was no sales amount column so using the available column **qty (number of items) and price (which is unit price), I created total_amount column.**

Note: I have not considered freight charges in the calculation of 'total_amount' because I found that when Olist started its business it was outsourcing the logistics to third party and therefore I want to give business insight of only the sales from the products sold at the Olist platform.

I also found that Olist had acquired PAX, its logistic partner later in the year 2020, check here.

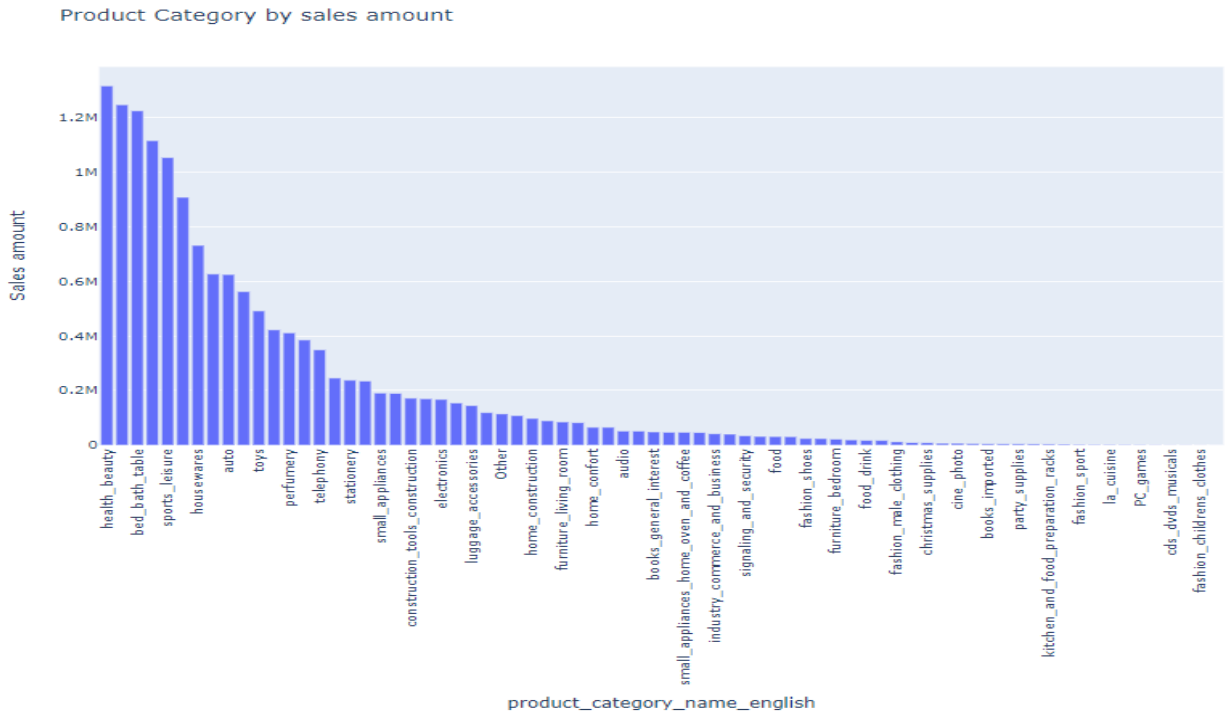
Later I joined all the tables and aggregated the total_amount by date. It gave the daily sales data from Olist from 2016-2017. I found that the data only had very few observations before 2017, so I deleted rows before Jan 01 2017.

The final dataset had 606 datapoints with last five observations look like this:

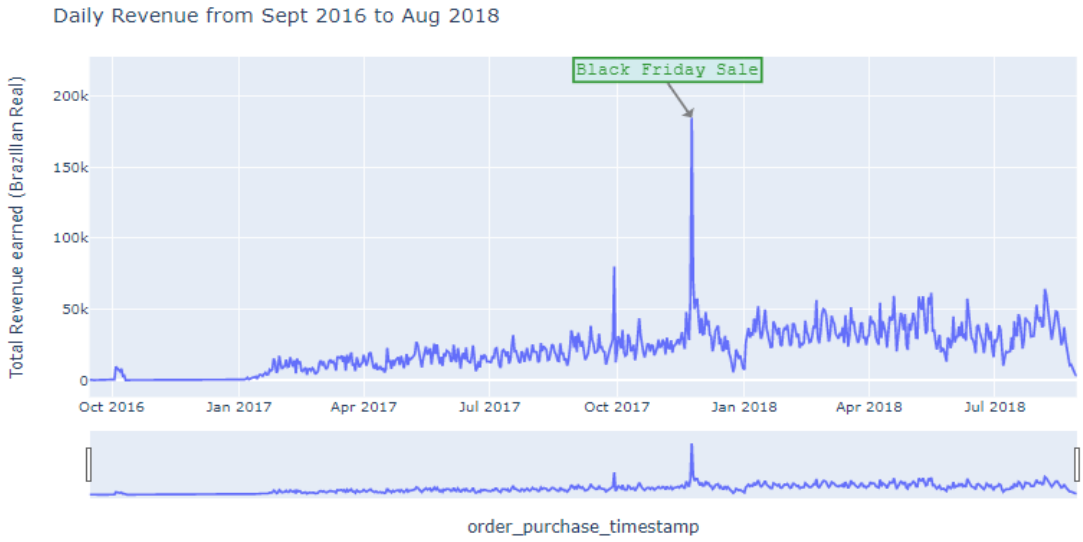
order_purchase_timestamp	total_amount
2018-08-25	10891.40
2018-08-26	8526.19
2018-08-27	5542.90
2018-08-28	4088.37
2018-08-29	2670.54

3. High level Insights

- There are around 96K unique orders with 93K unique customers which make up 96.79 % of the total customers in database. Only 3.21% of the customers are repeat customers. It may be because the data is the initial data when Olist had just started its business and therefore there are all the new customers in the database.
- There are around 32K unique products that belong to 74 different product categories ranging from health, luxury fashion accessories, furniture, electronics, sports and more.
- The highest earning product category is heath-beauty, watches-gifts and bed and bath.

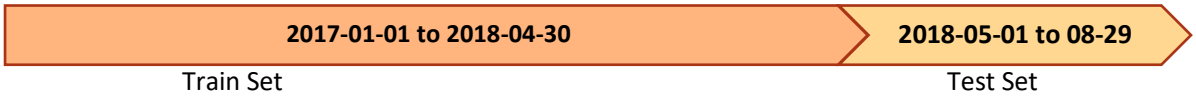


- The overall revenue earned as of Aug 2018 is 14.9 million Brazilian Real (R$).
- There was a highest sale of 184K R$ that was recorded on Black Friday event.



4. Modelling and results

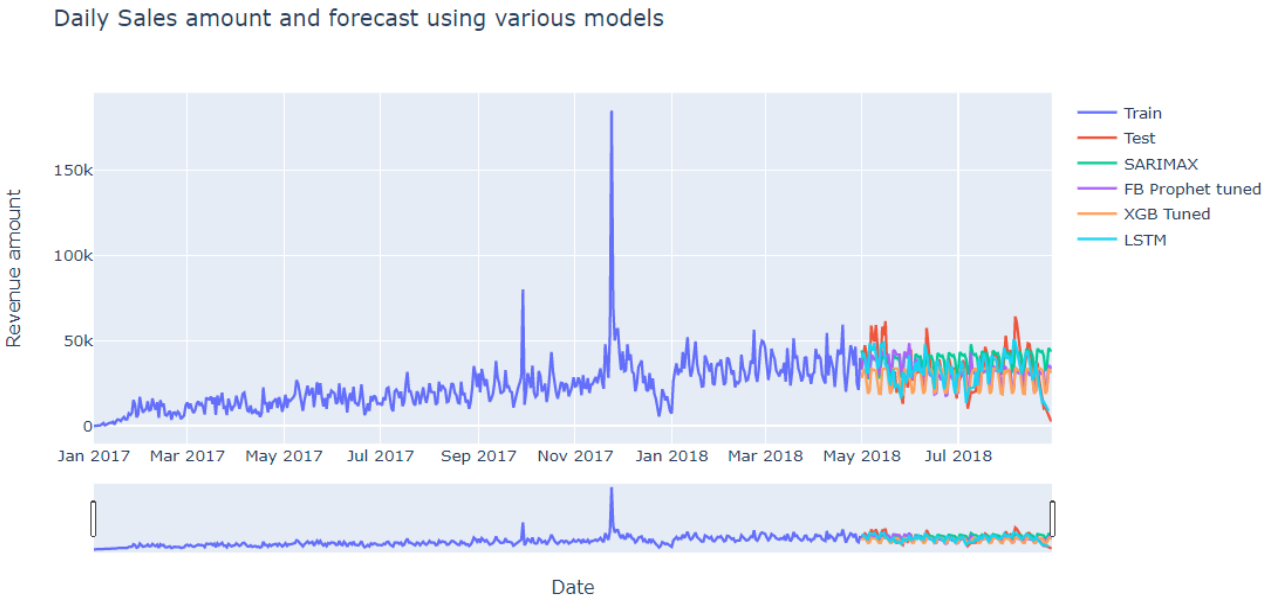
Before applying modelling I split the data into test and train set. The split was 80% train and 20 % test. Since there is a limited data, I could not carve out a validation set.



I applied a total of five models to predict the sales amount. I will briefly try to explain what I have done in each of these models:

1. SARIMA: In order to apply SARIMA, I first checked for stationarity using ADF and KPSS test, decomposed the timeseries to check seasonality and found that it has weekly seasonality, plot ACF and PACF to get the order of the SARIMA model. Applied the model and calculated MAPE and RMSE.
2. SARIMAX: I added an exogenous variable i.e., holiday so that model could pick up the holiday components. I did a grid search on the SRIMAX model to get fine tuned order of the SARIMAX that gave me slightly less RMSE and MAPE.
3. FB Prophet: Using an automatic prophet model without any parameter tuning yield worse result than the two previous model.
4. Tuned Fb Prophet: I decided to tune the parameters of FB prophet using grid search and it gave me better results than SARIMA and SARIMAX. I received a MAPE of 51.45%.

5. XG Boost Regression: I had to create features X by extracting information like dayofmonth, dayofweek, year, month, week etc. With tuning n_estimators, max_depth and learning rate I was able to get MAPE of 47.58% but my model predicted output with no trend.
6. LSTM: I later applied one step ahead univariate neural network model Long short-term memory and it gave exemplary results.



Results:

I am evaluating my models on the basis of MAPE (Mean Absolute Percentage Error) and RMSE (Root Mean Squared Error).

Model	RMSE	MAPE
SARIMA(1,1,1)(0,1,1)(7)	13810.59	68.99
SARIMAX(1,1,2)(0,1,1)(7) Including impact of holidays	13312.72	65.66
Baseline Prophet	27437.77	71.78
Baseline Prophet with holiday	15393.64	77.88
Tuned Prophet with holiday	12142.69	51.45
XGBoost Regression including Holiday	11493.59	52.18
Tuned XGBoost Regression including Holiday	12349.61	47.58
LSTM (one step Prediction)	2803.14	9.37

My LSTM model has MAPE of 9.37%. Despite getting this good result I am reluctant to proceed with this model as there is limited historical data and there are chances that it has remembered the data points. If we could get more data point to retest this model then I could be more confident on its performance.

Since XGBoost is not capturing the trend, I am proceeding ahead with Tuned FB Prophet.

5. Conclusion

I am going ahead with the tuned FB prophet model that is forecasting with 51.45 % Mean absolute percentage error since it is able to pick both seasonality and trend and has also been able to capture the variation between weeks. **It is a good baseline to further improve the model by incorporating more external features like inflation rate, weather data and customer satisfaction.**

Generally, a business aims to manage inventory based on forecast of a week or month beforehand. This model can be further improved to give a weekly and monthly forecast to help business better prepare. We have done a comprehensive forecast for all the product categories. For future work, we can do separate forecast for different product categories.

I found that it was very challenging to do time series with limited data and many external factors can be incorporated to improve the performance.