Retail Sales Data Analysis

Final Project Report

***Abstract*— Analysis and prediction of sales at Costello Ace hardware stores using the data set from years 2015-2018.**

Retail Sales Analysis is an examination of the wealth of information that the retail store proprietors have readily available. Ability to track and examine business decisions as well as customers' purchases and behaviours makes Retail Sales Analysis a powerful tool for organizations.

I. Overview Of The Report

The report begins with Sales Prediction Analysis where several methods for sales prediction have been employed. We discuss about the baseline model, then the more convoluted models. This is followed by the analysis on Customers. The next section deals with elastic pricing. We have conducted extensive research on the factors which effect the pricing of a product. Research on class code embed-dings have produced promising results as well. Finally we discuss the results, conclusion and end it with acknowledgements.

II. Sales Prediction

Time series forecasting involves building models and fitting them on historical data to predict future observations.

On initial observation of figure 1 we see that there is a reduction in sales from years 2015 to 2018. We see spikes at common time intervals throughout the year leading to the hypothesis that the sales pattern is seasonal.

Different Steps/Models used to perform sales prediction:

A. Data Preparation

We created a new dataframe by melting the data on a weekly basis and based on the department. This was done to get more data points for constructing the model. Thus, the new data frame has three columns:

- Department Code: Code for the department.
- Weeks: extending from 1 to 210 (across 4 years)
- Value: Sales for that particular department for that week.

B. Feature Engineering

Initially we build two features that are intrinsically used in time series analysis:

- n_Lag (shift =n) : n steps back in time that is the nth previous week sales)
- n_Lag_Diff : Difference between sales of the current and previous week.

	DC	Week	Sales	1_Lag	2_Lag	3_Lag	1_Lag_Diff	2_Lag_Diff	3_Lag_Diff
142	2	5	830.0	1111.0	1096.0	913.0	15.0	183.0	341.0
177	2	6	915.0	830.0	1111.0	1096.0	-281.0	15.0	183.0
212	2	7	1271.0	915.0	830.0	1111.0	85.0	-281.0	15.0
247	2	8	1384.0	1271.0	915.0	830.0	356.0	85.0	-281.0
282	2	9	1022.0	1384.0	1271.0	915.0	113.0	356.0	85.0

Fig. 2. Feature Engineering - calculation of lags

C. Cross Validation for Time Series

Forward Chaining validation is a resampling procedure used to evaluate machine learning models on a limited data sample.

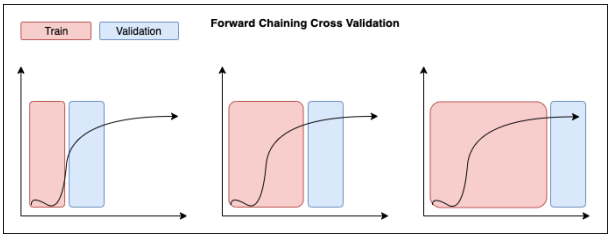

Fig. 3. Figure shows the functioning of Forward chaining cross validation.

The results of a K-fold cross-validation run are often summarized with the model skill scores. We have used Root Mean Squared Log Error (RMSLE) to evaluate the model.

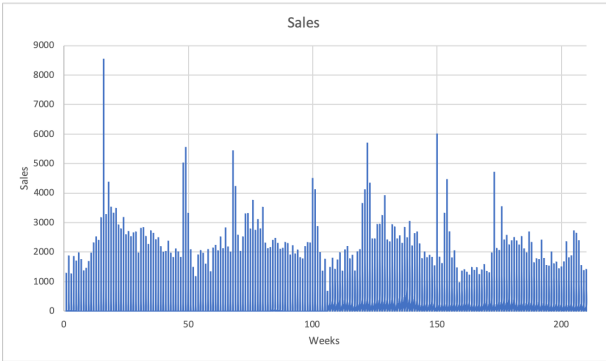

Fig. 4. The sales in store 9 over the years has a seasonal pattern.

D. Baseline Model

As part of the baseline model, we logically assume that the Sales for each department is constant throughout the 210 weeks (which will take the value of Week-1). Applying

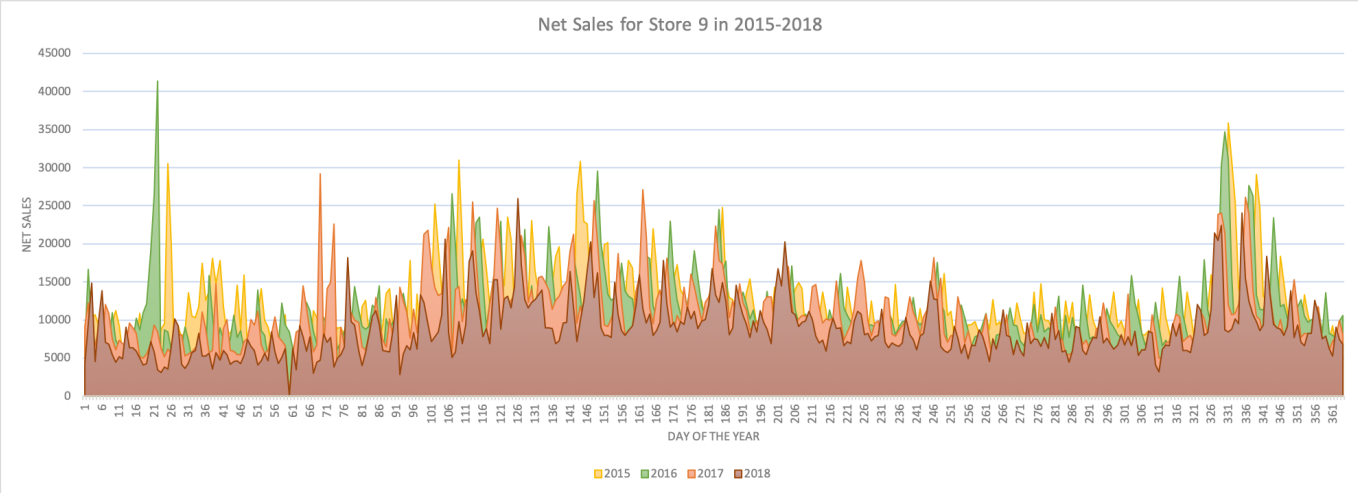

Fig. 1. Net sales for store 9 in 2015-2018

KFold CV to evaluate the model, the total error obtained is 0.599. Following observations are made:

- There is a year based pattern. Thus K-Fold CV works best if applied on part of the data belonging to the same year.
- The sales of each department varies by a margin (Some departments are seasonal). Hence it is a good option to predict sales per department by building separate models.

```
Fold: 194, Error: 0.973
Fold: 195, Error: 0.382
Fold: 196, Error: 0.537
Fold: 197, Error: 0.804
Fold: 198, Error: 0.407
Fold: 199, Error: 0.445
Fold: 200, Error: 0.447
Fold: 201, Error: 0.631
Fold: 202, Error: 0.444
Fold: 203, Error: 1.273
Fold: 204, Error: 0.374
Fold: 205, Error: 0.485
Fold: 206, Error: 0.449
Fold: 207, Error: 1.212
Total Error 0.599
```

Fig. 5. Error rate for baseline model.

E. Model based on Random Forest

As Random Forest is robust to outliers, unbalanced data and has low bias and moderate variance, we have used it as the next model. The RMSLE obtained after performing KFold CV on the model from weeks 160-210 yielded an error rate of 0.743. 3-Lags were used for the modelling. Due to the highly seasonal pattern of sales in a lot of departments, the model failed to predict the sales well.

To overcome the drawbacks from the above system, we have used statistical method SARIMA and LSTM Neural Networks as illustrated.

F. SARIMA

A popular and widely used statistical method for time series forecasting is the ARIMA model. SARIMA is an acronym that stands for Seasonal AutoRegressive Integrated Moving Average.

The parameters applied to the ARIMA model are:

- p: The number of lag observations included in the model, also called the lag order. Shifting the data backward in time(sequence) is called lag times or lags.
- d: The number of times that the raw observations are difference.
- q: The size of the moving average window, also called the order of moving average.

For the moving average, we have used the window size as 12 weeks since each season lasts a similar duration. On using the SARIMA on the processed dataset, we were able to obtain a model which was able to forecast future weekly sales with an RMSE of $10,000 per week which is a good forecast considering that the sales of the store are very unsteady.

Testing it on the last few weeks of the 2016 dataset, we see that the forecast is quite close to the real data and corresponds to all the peaks and valleys.

G. Sequential LSTM (Long short-term memory)

Another popular and widely used statistical method for time series forecasting is the LSTM model. LSTM stands for Long Short Term Memory. It's a neural network specifically designed to remember previous values for the better prediction which is not available in traditional neural networks. It is perfect for predicting time series values.
Parameter used is **Net Sales Float**

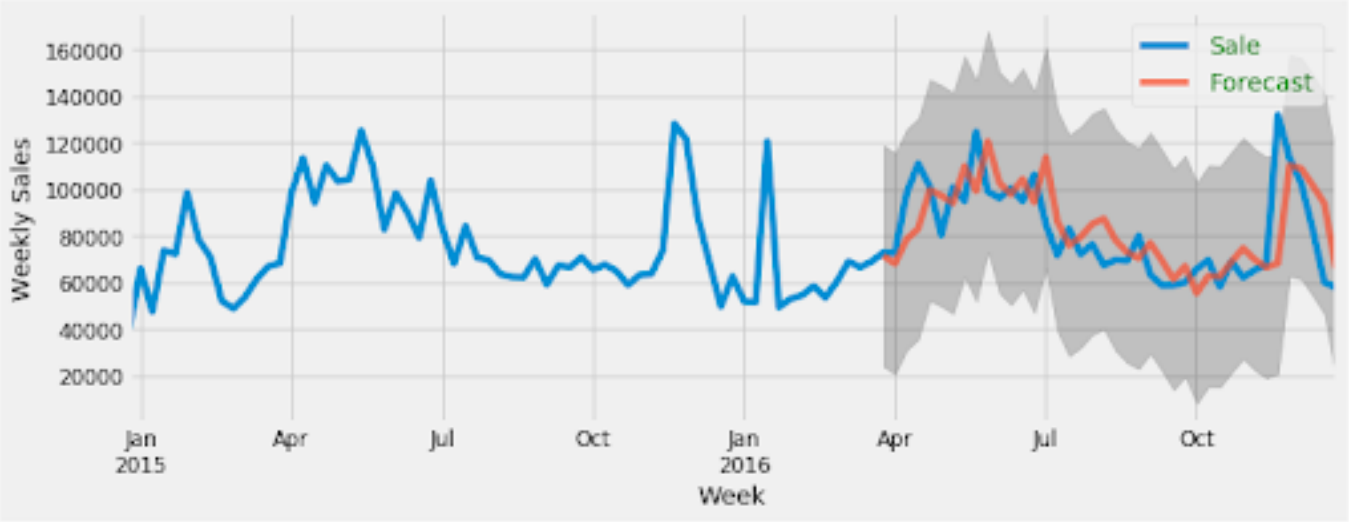

Fig. 6. SARIMA - Plot of Forecasted sales on Real sales

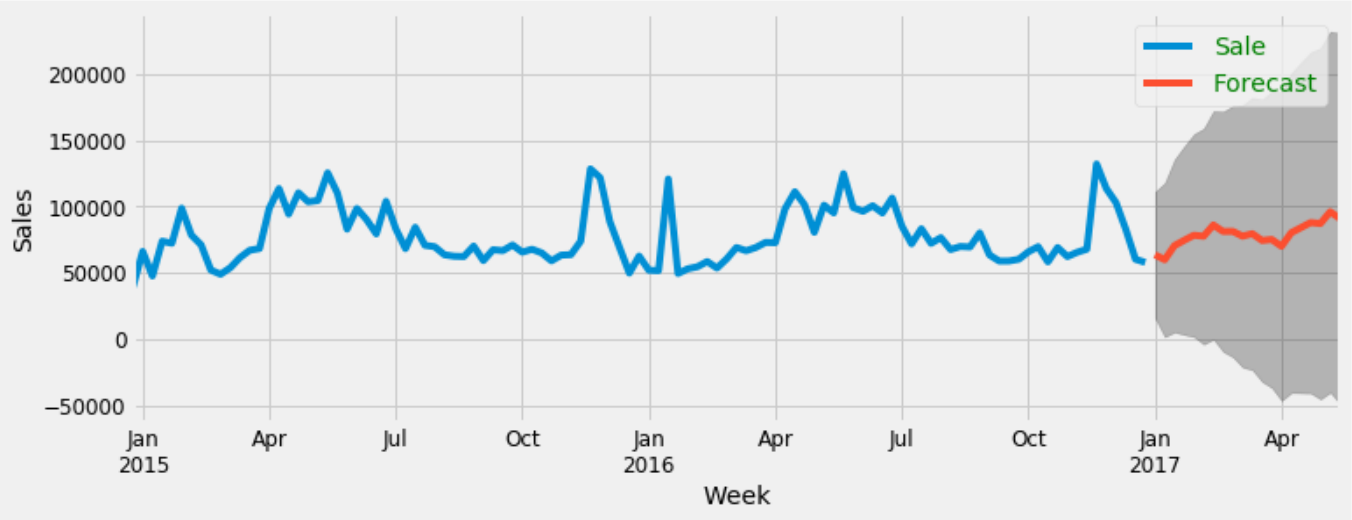

Fig. 7. SARIMA - Predicting the sales for the first 4 months of 2018

The baseline model only used Net Sales Float column for predicting daily sales values. The model parameters were:

- **Optimizer: Adam**
- **Loss Function: Mean Square Error(MSE)**
- **Activation Function: ReLu**
- **Model Type: Sequential**
- **Dropout: 0.15**

The model was run over 90 epochs which resulted in the final loss of 0.047.

The function was trained on the 4 years worth of data and split for training and testing. The training data had 1393 points and was tested on the last 60 of data. The results were convincing.

Then, using the entire data available (1453 points), the model was trained, again, using the same parameters. The loss was reduced even further. The results showed good patterns which is to be expected.

We tried adding other factors that would influence the people from actually going to the store and buying products like weather, snowfall, temperature etc. but most values turned out to be NaN values due to which we had to discard the idea.

III. Customer Analysis

Customer analysis is a critical component of any business plan in all stages of growth. Analysing customers helps the store define who the target market is, and decide how to reach them. From marketing to delivery, customer analysis and customer analytics reveal the most necessary information for any business plan.

```
Epoch 90/90
233/233 [==============================] - 17s 73ms/step - loss: 0.0047
```

Fig. 8. Sequential LSTM - Loss after 90 epochs

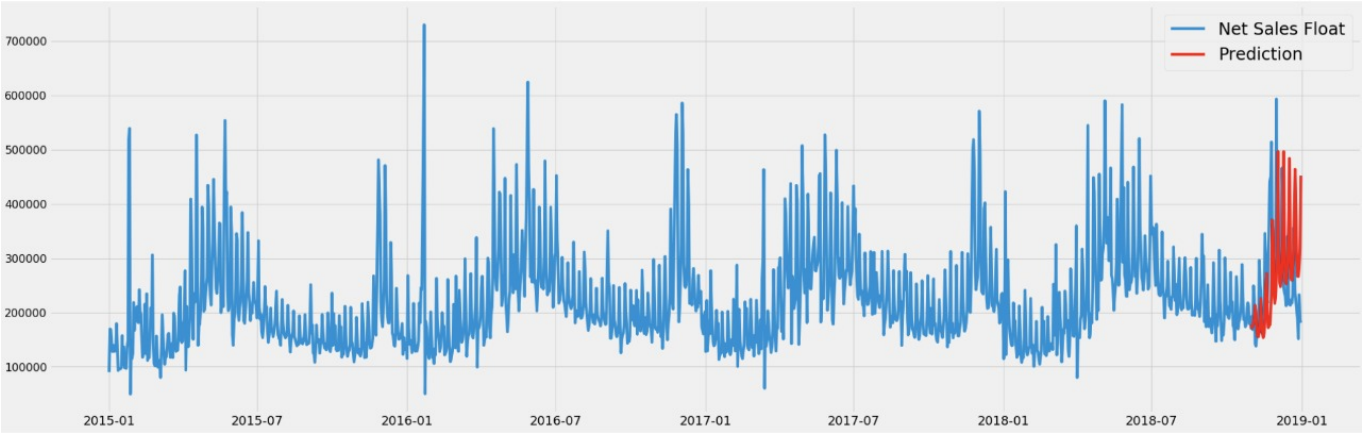

Fig. 9. Sequential LSTM - Prediction for the last 60 days for 2018

```
Epoch 90/90
233/233 [==============================] - 19s 81ms/step - loss: 0.0035
```

Fig. 10. Sequential LSTM - Loss after 90 epochs after training on full data

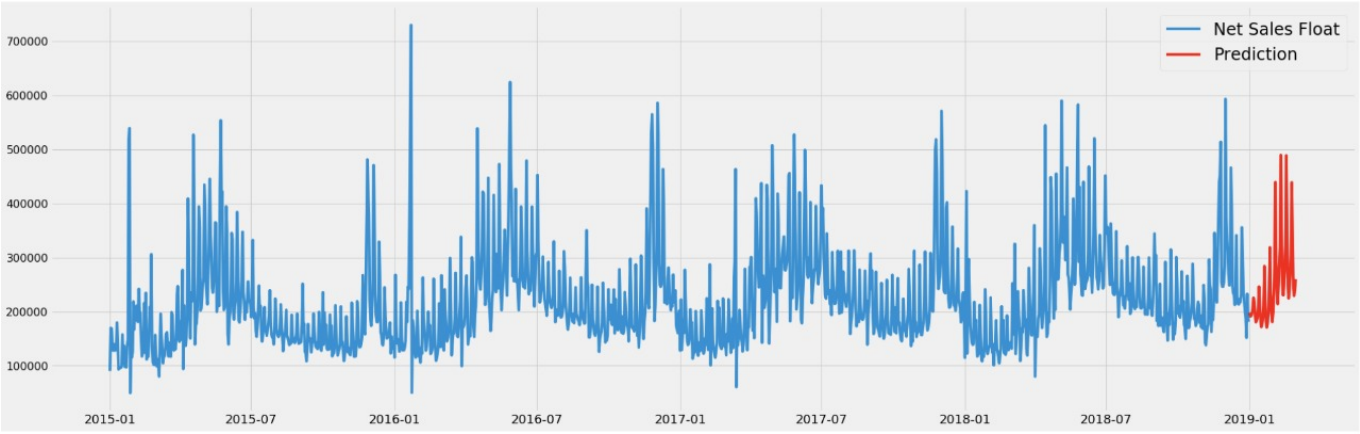

Fig. 11. Sequential LSTM - Predictions for 60 days in 2019

A. *Customer Location Analysis*

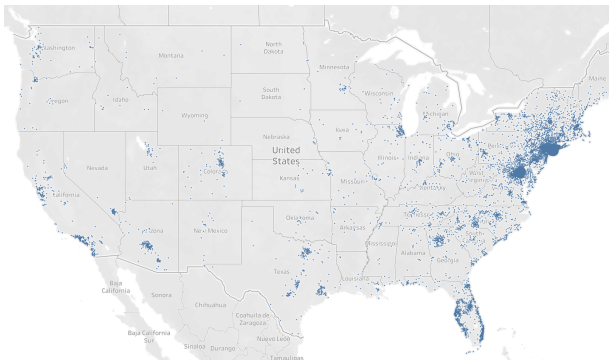

Fig. 12. Map showing the cluster of customers from different parts of America

The Zip Code column is used to plot the density of customers in the USA. Even though the stores are limited to NY and NJ, a lot of registered customers hail from Florida. This presents the prospects of opening stores in other regions as well.

B. Recency, Frequency, Monetary (RFM) Model

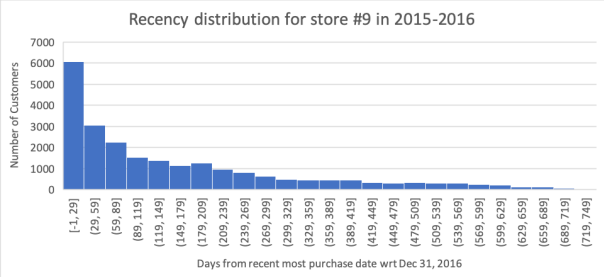

Fig. 13. Recency distribution

Customer segmentation can be done by 3 important features:

- **Recency** - Number of days since the last purchase.
- **Frequency** - Number of transactions made over a given period.
- **Monetary** - Amount spent over a given period of time.

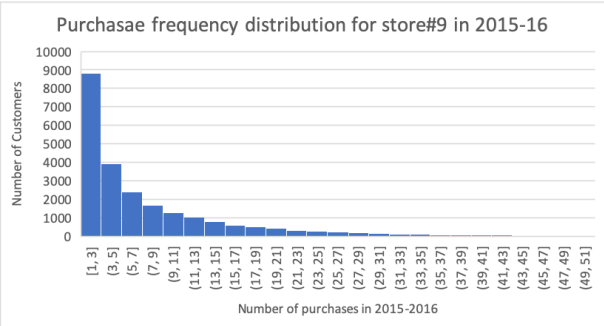

Fig. 14. Frequency distribution

From preliminary analysis, we know that 8000 unique customers (which excludes *5 and other constants) visit a store in a year. Analysis on customer retention plays a major role in preventing customer churn.

Steps Involved in RFM Modelling

- Outliers from the store data were removed (Customers with single transactions and greater than 50 transactions were considered not feasible for the study.)
- Created RFM Features for each customers and planned to utilize 80% quantile for Recency and Monetary.

Customer Number	Recency	Frequency	MonetaryValue	R	F	M
*1000	351.0	11.0	242.83	1	1	2
*10002	27.0	43.0	330.36	2	2	2
*10007	63.0	5.0	23.13	2	1	1
*10024	20.0	19.0	178.81	2	2	2
*10025	526.0	2.0	12.43	1	1	1

Fig. 15. Result after Step 2

- Calculate the RM score and bucket customers into different categories.

	Number of Customers	Recency	MonetaryValue
RMScore			
22	4552	48.0	287.0
21	124	414.0	218.0
12	14084	100.0	54.0
11	4534	446.0	32.0

Fig. 16. Result after Step 3

Results

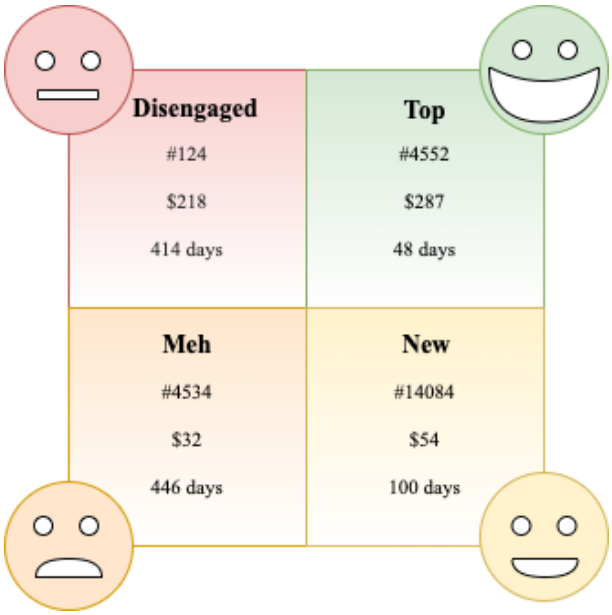

Fig. 17. Customer segregation

- The **Disengaged** customers have a high average monetary value and could be converted into **Top** category

through proper marketing channels.

- There are high number of **Meh** customers who can be converted into **New** customers by providing customized promotions.
- On an average, about 25% of the customers fall into the Top category throughout the different stores.

Detailed monthly RMF analysis could open different avenues which, when coupled with the right promotional and marketing strategy can prove to be a powerful tool.

IV. Price Elasticity

Price Elasticity of Demand(PED) is a measure used in economics to show the responsiveness or the "elasticity" of the quantity demanded of a product or service to a change in its price when nothing but the price of the product changes. In detail, it gives the percentage change in quantity demanded in response to unit percentage change in the price of the product.

This measure is used to understand how sensitive the demand for a product is at a certain price. In marketing, this measure is used to understand how consumers respond to a change in the price of a product.

We have performed this analysis on the top 5 grossing products to understand how the demand varies at different prices. Surprisingly, we did not observe the general market trend where with an increase in the price of a product, the quantity demanded of that product decreases. Instead, we observed that the top grossing products were bought in bulk quantities at a comparatively higher price. We were able to confirm this using Ordinary Least Squares(OLS) Estimation.

OLS Regression Results

Dep. Variable:	Quantity	R-squared:	0.165
Model:	OLS	Adj. R-squared:	0.129
Method:	Least Squares	F-statistic:	4.557
Date:	Mon, 02 Dec 2019	Prob (F-statistic):	0.0437
Time:	06:37:49	Log-Likelihood:	-159.50
No. Observations:	25	AIC:	323.0
Df Residuals:	23	BIC:	325.4
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-180.5193	115.122	-1.568	0.131	-418.667	57.628
Price	6.5589	3.072	2.135	0.044	0.203	12.915

Omnibus:	44.428	Durbin-Watson:	2.287
Prob(Omnibus):	0.000	Jarque-Bera (JB):	174.891
Skew:	3.290	Prob(JB):	1.05e-38
Kurtosis:	14.162	Cond. No.	145.

Fig. 18. OLS Estimation Results

Observations:

- We cannot reject the null hypothesis that Price has no effect on the Quantity demanded because the value of P is sufficiently large.
- The low R-Squared value indicates that our model does not explain a lot of the response variability.
- Since, the regression failed, we will plot graphs to understand the responsiveness of the Quantity demanded at a certain price for a certain product.

From the below graphs, we can see that the Quantity demanded is tending to increase with an increase in price which is not how the general consumer trend works.

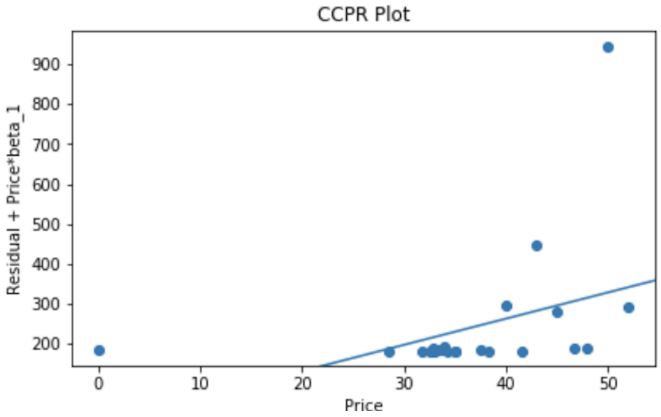

Fig. 19. Component-Component Plus Residual Result

We have observed that the top 5 grossing products are following this trend which means that the store can significantly increase their revenue generated from these products by slightly increasing their price without losing any demand for the product (Inference based on not considering customer retention).

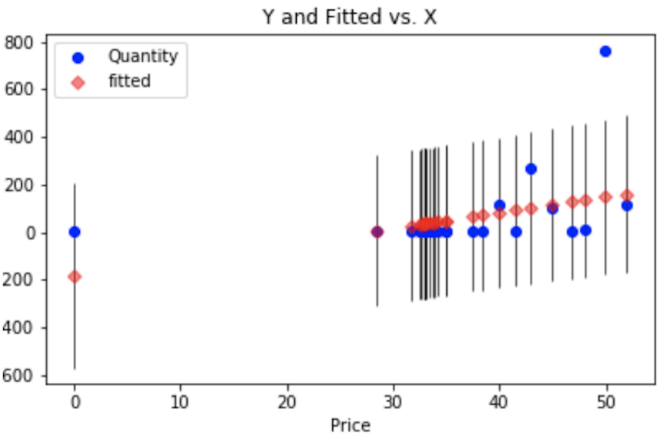

Fig. 20. Plot of fitted and real prices

V. Product Embeddings

Apriori Algorithm is one of the most important algorithms for Market Basket analysis as it helps the customers in purchasing their items with more ease which increases the sales of the market. It has been extensively used in healthcare for the detection of adverse drug combination reactions.

Apriori algorithm only gives us the relevant association rules using which certain inferences can be made on the dataset. We have various metrics to evaluate the value of interest of an association rule like Support, Confidence, Lift and conviction of which we will use the Lift as the evaluation metric as this is a metric which considers both the antecedents and the consequents popularity in the dataset.

Challenges faced:

- The sheer number of products forced us to consider $Class\ Name$ for embeddings rather than $Fineline\ Name$.
- The huge number of products produced sparse matrix which was incomputable using traditional machines. To overcome this, we made use of the Google Cloud platform to deploy an $Ubuntu$ Virtual Machine with 8 cores and 80GBs of RAM initially, which was not fruitful. After a couple of optimizations and tweaks, we managed to get it to work with 120GB of RAM.
- The sparse matrix still proved to be too huge when applying a certain operation to reduce the count of items for a particular row to 1. As a last resort, we increased the RAM to 200GB which helped us overcome this challenge.

Steps Followed:

- Pre-processed the data to include only the $Receipt\ Number$ and $Class\ Name$ and one hot encoded the $Class\ Name$ feature.
- Min support of 0.0005 was chosen which means, atleast 5 out of 1000 transactions must include this.
- Then we pass the dataset to the Apriori model which gives us a set of association rules.
- Once we get the association rules, we consider the top rules by sorting them according the descending order to make sure that we are considering only those rules which occur most frequently in the dataset to avoid those conditions where an item set occurs only once but in combination with another product which could be a false positive.
- The rule with the highest Lift has the following property: The antecedent occurs quite frequently in the dataset and also, the consequent occurs along with the antecedent with a significant frequency.

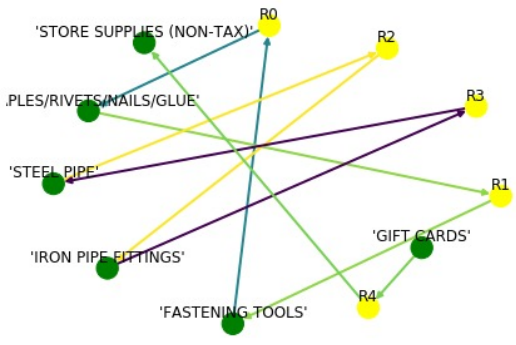

Fig. 21. Embeddings for 2017-2016

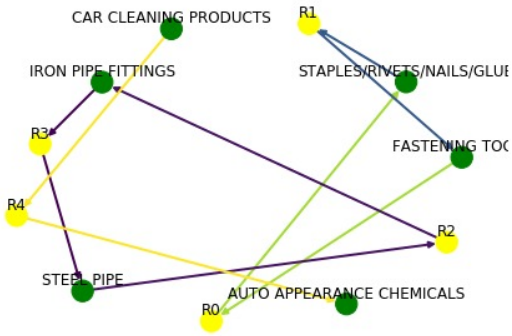

Fig. 22. Embeddings for 2015-2016

In the above figures, each yellow colored node is an association rules prefixed with 'R', the incoming edges originate from antecedents and the outgoing edges go to the consequents. The green nodes are the different Class Names where for an outgoing edge, it is an antecedent for an incoming edge, and a consequent for an outgoing edge. For example, in fig. 21, we have an edge from 'FASTENING TOOLS' to 'R0' and 'R0' to 'STAPLES/RIVETS/NAILS/GLUE'. This corresponds to the first association rule where the antecedent is 'FASTENING TOOLS' and the consequent is 'STAPLES/RIVETS/NAILS/'.

VI. CONCLUSION

With this project report we were able to tackle multiple challenges effectively. The dataset provided several challenges like non uniform values and irregular features but we were able to navigate through them and achieve the results that we have. The analysis in this report is just the tip of this iceberg. There is a lot that can be done on top of this analysis with a few tweaks and changes in the way the data is procured. We have laid the foundation for further analysis of the dataset and any work on top of what we have done will enhance the total sales of the stores.

REFERENCES

[1] https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/
[2] https://github.com/tristanga/Data-Analysis/blob/master/Notebooks/Automatic\%20Customer\%20Segmentation\%20with\%20RFM\%20(Python).ipynb
[3] https://www.seanabu.com/2016/03/22/time-series-seasonal-ARIMA-model-in-python/
[4] https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/
[5] https://colah.github.io/posts/2015-08-Understanding-LSTMs/
[6] https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting/
[7] https://machinelearningmastery.com/dropout-for-regularizing-deep-neural-networks/
[8] https://www.researchgate.net/publication/315918996_Adaptive_Apriori_Algorithm_for_frequent_itemset_mining
[9] https://machinelearningmastery.com/sarima-for-time-series-forecasting-in-python/