# DS Track Challenge 1

## (Retail Dataset)

**Indrani Singha Roy**

Loom Recording Link

# Introduction / Agenda

- Exploratory Data Analysis (EDA)
- Clustering
- Time Series
- Evaluation of Results & Insights
- Future Work / Improvements

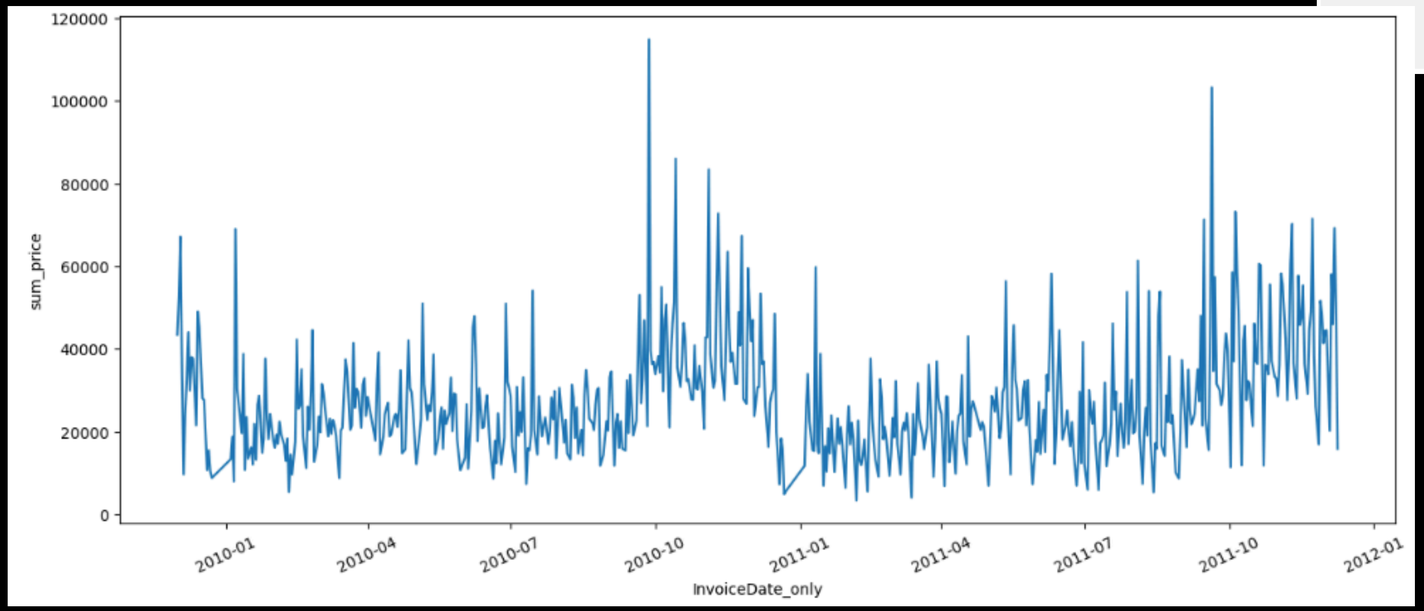
# EDA

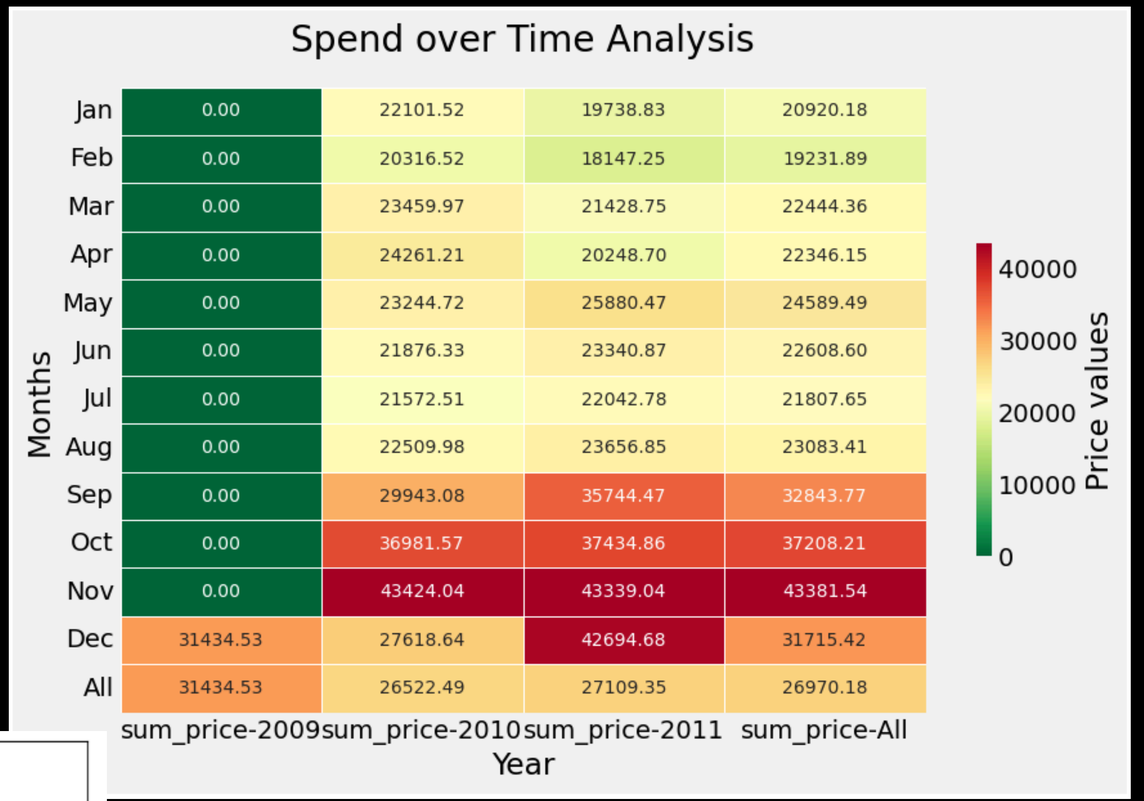
- Combining both years of data & investigating indicated:
  - Columns ‘Customer ID’ and ‘Description’ had missing values. The rows were dropped.
  - Datatypes of columns had to be changed since Invoice no. stock codes don’t have any ordinality & should be object datatype or categorical
  - 34,335 duplicate rows & it was verified that they don’t have a counter-balancing negative transaction so these could be safely dropped.
  - 18,390 records with Invoice starting from C (or cancellations). All of them had -ve quantities. Further in some cases, there were corresponding +ve entries (with different invoice number & timestamp). In order to show the true spend at customer level, we decide to retain them

# EDA (cont.)

- UK is the dominant country for ‘Total Price’ , count of Customer IDs , Invoice Counts etc., however, the other countries do make up for about $500k revenue so we have retained all.

- Aggregating at date & month level gives us the time series across all customers & heat map on right



# Clustering

Prepping Data for Clustering (RFM)

KMeans Clustering Results

# Prepping Data for Clustering (cont.)

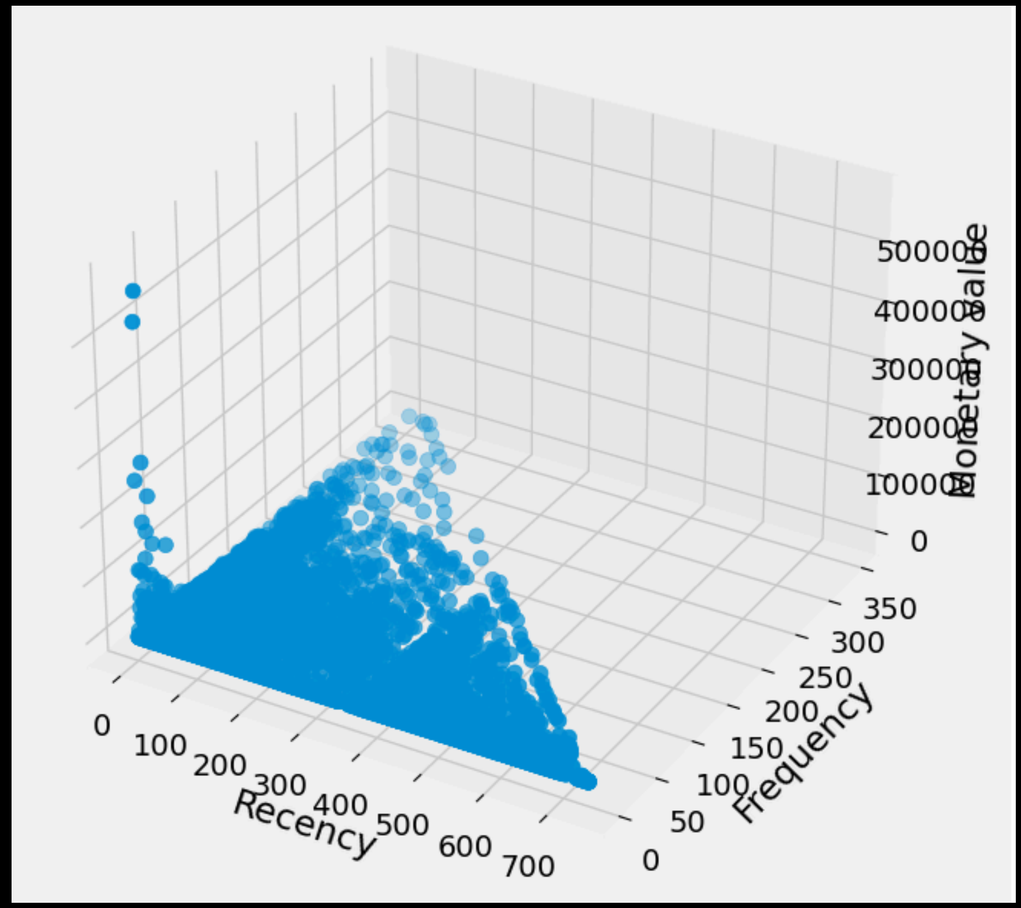
- Convert transaction level data —> Customer Level Data with groupby functions

| Customer ID | unique_invoices | unique_stockcode | unique_transactiondates | min_transactiondate | max_transactiondate | total_spend | days_between |
|---|---|---|---|---|---|---|---|
| **14911.0** | 510 | 2557 | 285 | 2009-12-01 | 2011-12-08 | 265757.91 | 737 |

- RFM Model
  - Recency ——> ‘days_since_lastpurchase’
  - Frequency ——> 'days_between'/ 'unique_transactiondates
  - Monetary ——> ‘total spend’
- Corollary : Recent single purchase customers would record high on recency but their frequency rank would be low. There are about 1461 customers with single transactions (not all recent).
- Quartile based Score (1-4)

# Prepping Data for Clustering (RFM)

- Once scores are assigned, RFM Score is calculated
- RFM Score = recency_score + frequency_score + monetary_score
- Final Snapshot of dataset (for 5858 Customers) & 3D Map of distribution along these 3 features on right




| Customer ID | unique_invoices | unique_stockcode | unique_transactiondates | monetary | days_between | recency | frequency | repeat_customer | RFM_score |
|---|---|---|---|---|---|---|---|---|---|
| **12347.0** | 8 | 126 | 8 | 4921.53 | 402 | 3 | 50.25 | 1 | 3 |
| **12348.0** | 5 | 25 | 5 | 2019.40 | 363 | 76 | 72.60 | 1 | 5 |
| **12349.0** | 5 | 139 | 5 | 4404.54 | 717 | 19 | 143.40 | 1 | 4 |

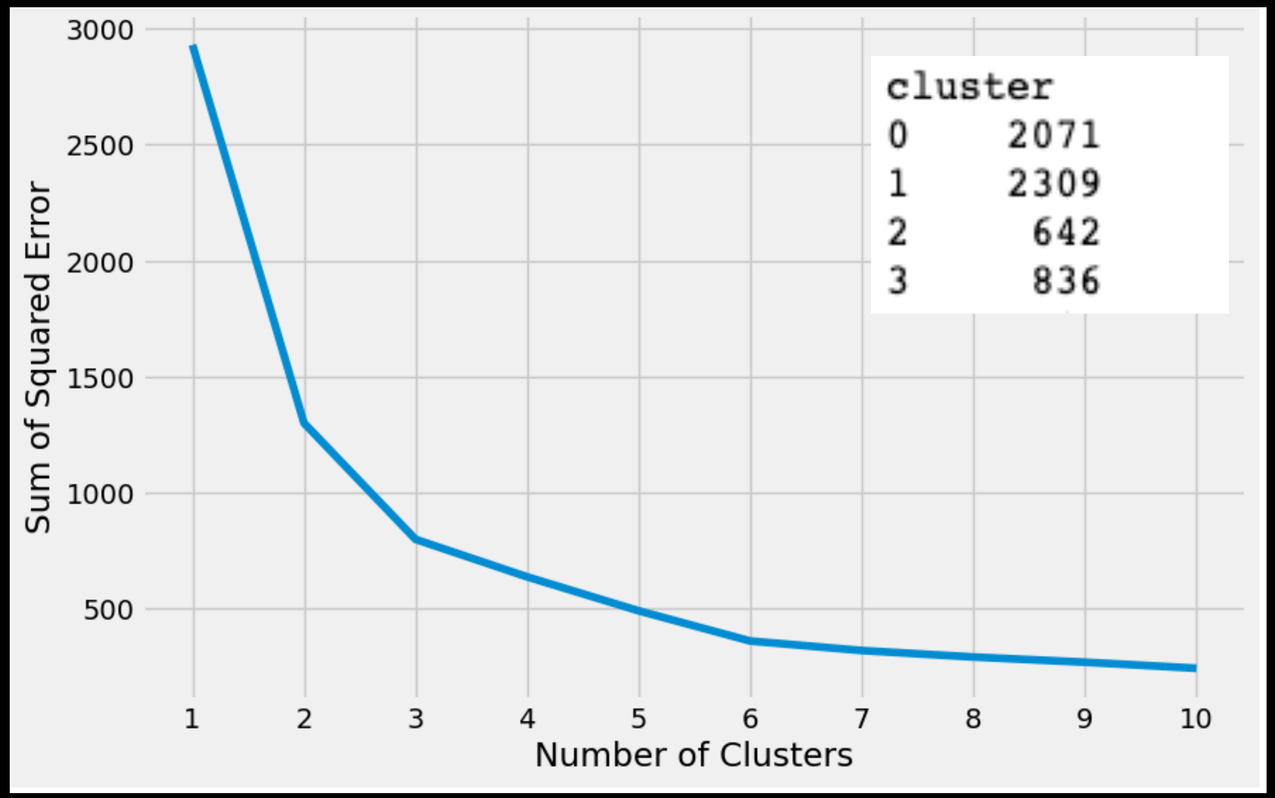
# Clustering

Prepping Data for Clustering (RFM)

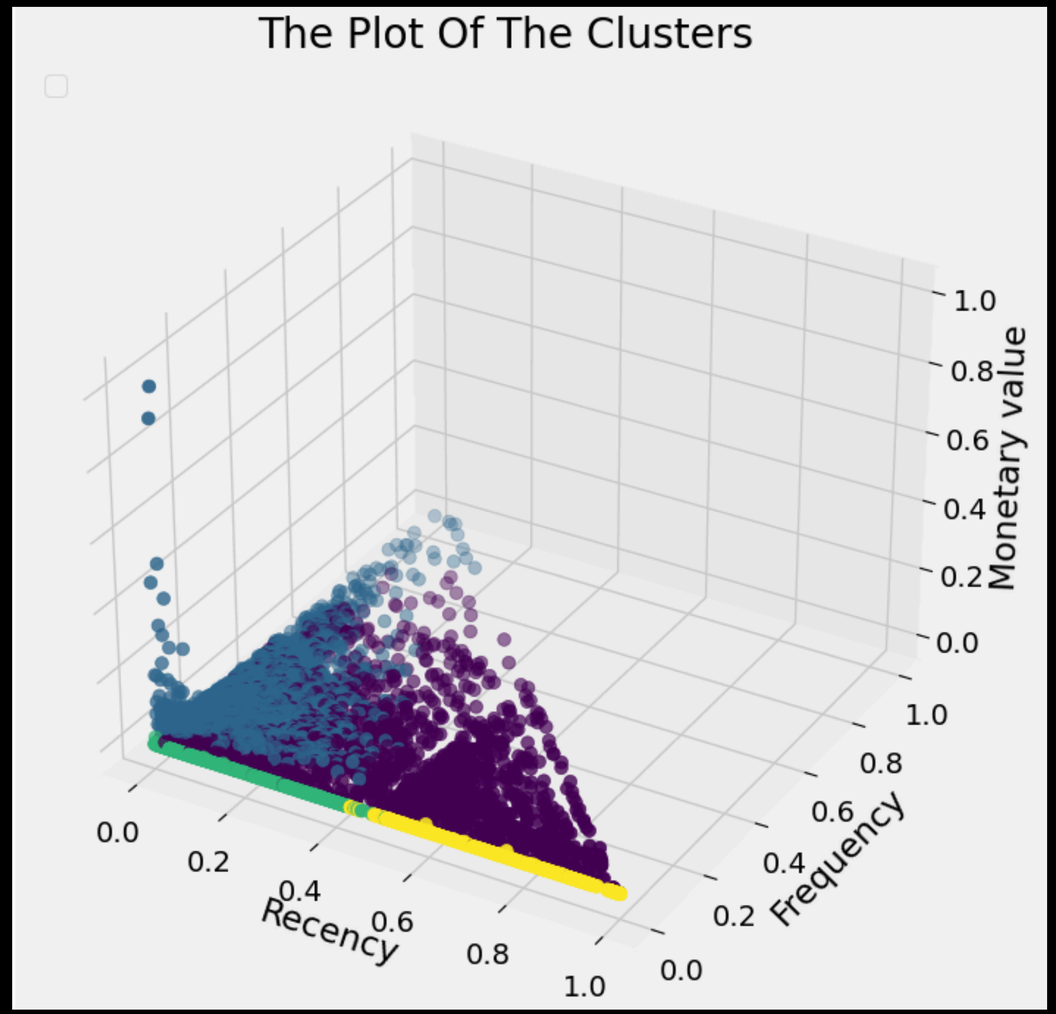
KMeans Clustering Results

# KMeans Clustering

- Dataset modified was scaled / normalized as it is important for KMeans which is Euclidean distance based method
- Elbow Method yielded 3-4 clusters as optimal




| cluster | |
|---|---|
| 0 | 2071 |
| 1 | 2309 |
| 2 | 642 |
| 3 | 836 |



# Takeaways

- Cluster 0 (2071 customers) :  haven't purchased for a long time (poor recency score), but for the time they were customers, had high / low frequency which implies they have been repeat customers at some point (recall that frequency will be 0 if its only single transcation ), however they did not add much to the monetary value

- Cluster 1 (2309 customers) : Preferred customers who are (repeat customers +  recent purchasers) or (add monetary value)

- Cluster 2 (642 customers) : Good Prospects since they have recently purchased (even though once) so we can strategically convert them into returning customers

- Cluster 3 (836 customers) : Lost customers since they only had single transaction long ago and did not return to purchase recently (implies that monetary value is low)

# Time Series Modeling / Forecasting

XGB Method

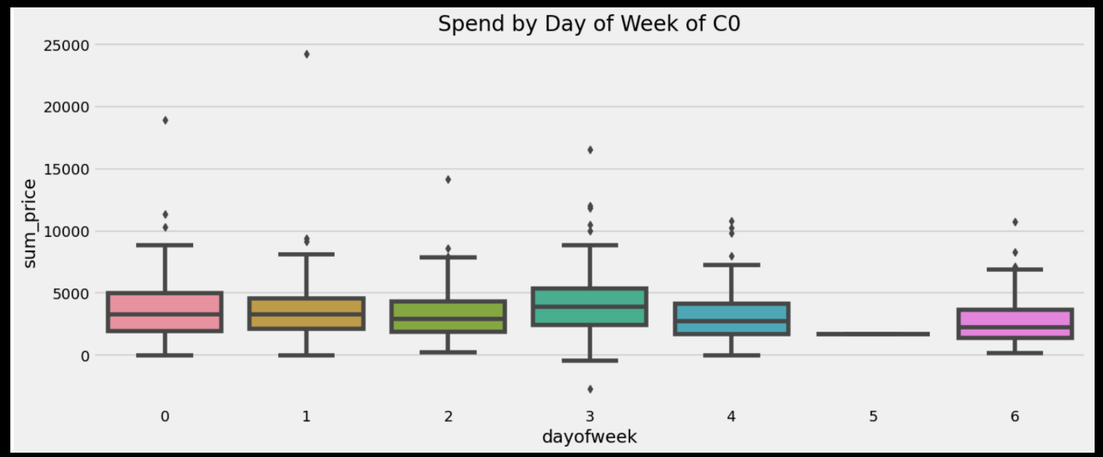
FB Prophet

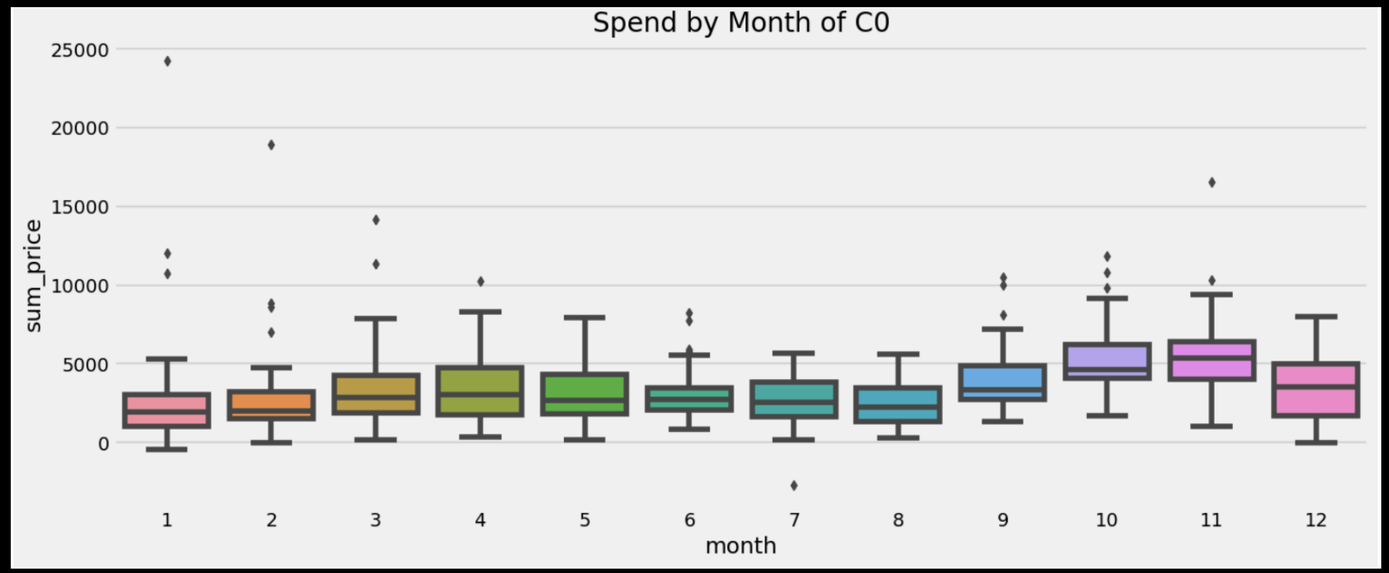
# XGB Regressor for Time Series Modeling

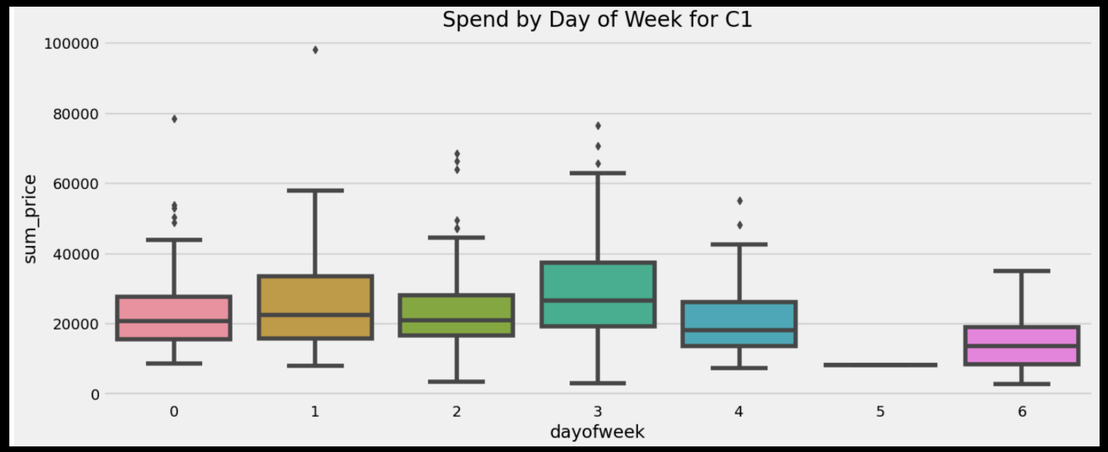
- Used last 27 days as the testing dataset
- Created function to leverage date specific features + lagging + rolling features
- We can test the model for Cluster 0 & 1 first since they contain the bulk of the data

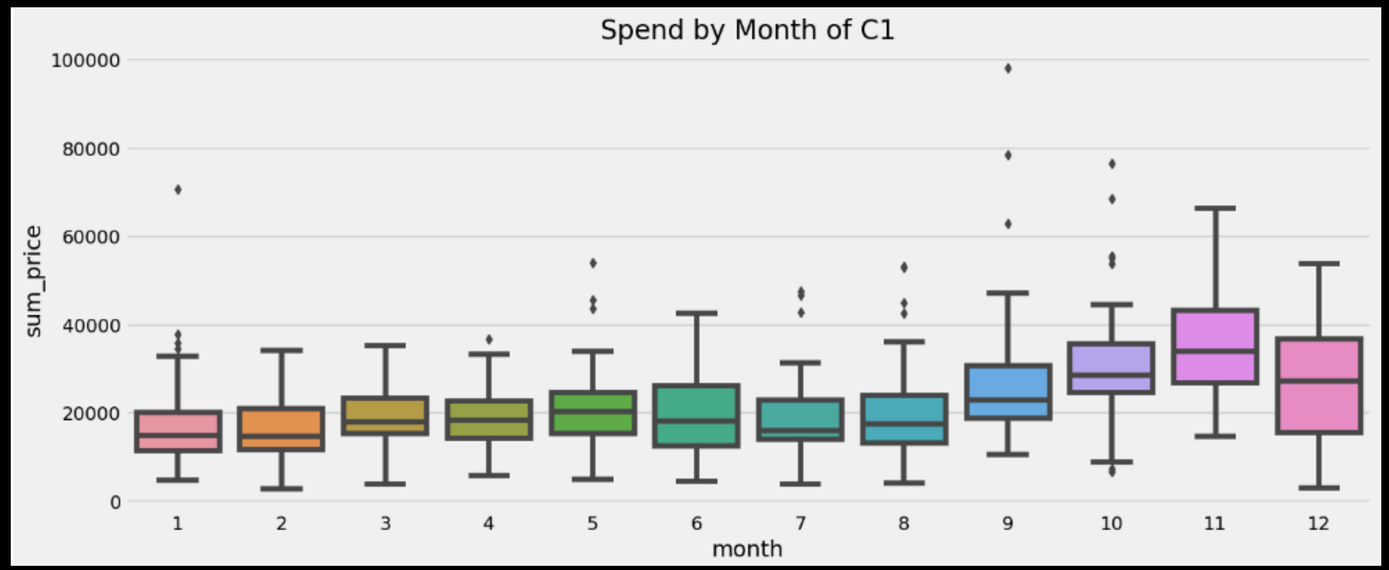
```
def create_features_lags_rolling(df):
    """
    Create time series dataset
    """
    df=df.copy()
    df['month']=df.index.month
    df['year']=df.index.year
    df['quarter']=df.index.quarter
    df['dayofyear']=df.index.dayofyear
    df['dayofweek']=df.index.dayofweek
    df['target_lag_7'] = df['sum_price'].shift(7)
    df['target_lag_1'] = df['sum_price'].shift(1)
    df['target_rollingmean_7'] = df['sum_price'].rolling(window = 7).mean()
    df['target_rollingmean_2'] = df['sum_price'].rolling(window = 2).mean()
    df['target_rollingmean_14'] = df['sum_price'].rolling(window = 14).mean()
    return df
```

# XGB Regressor for Time Series Modeling



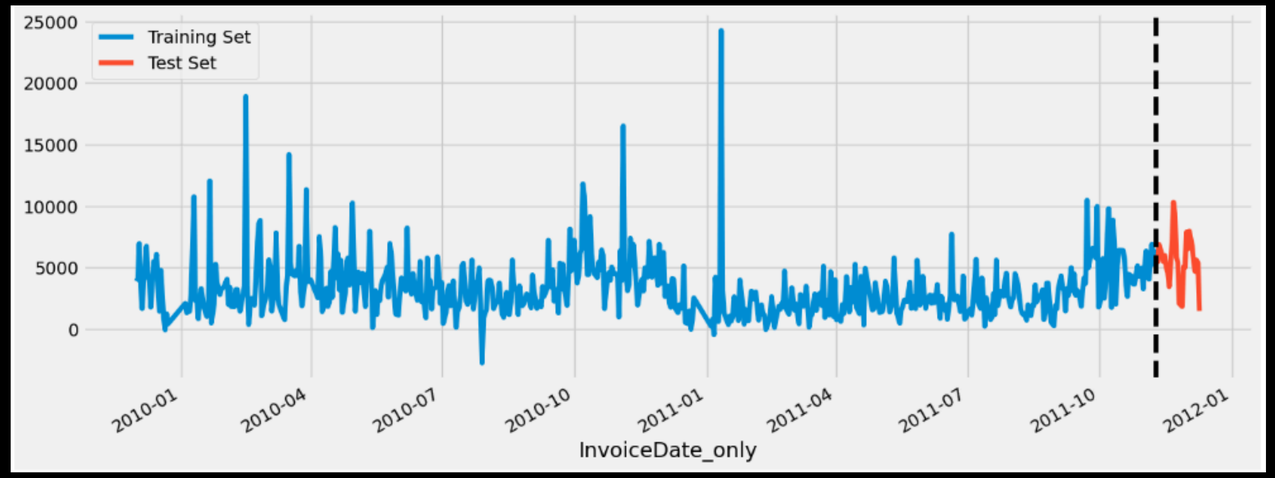



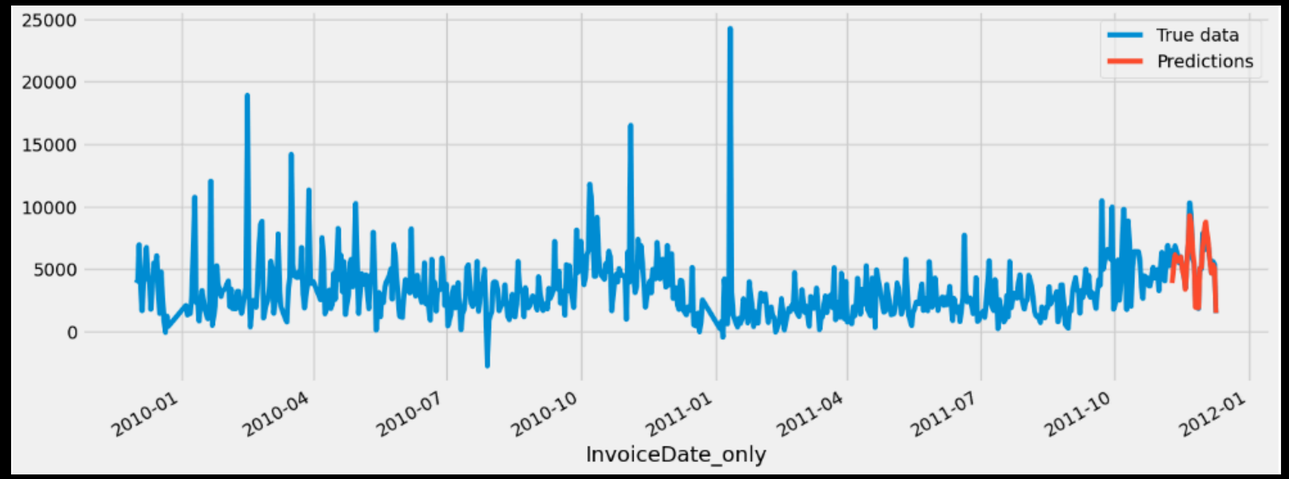



# Results Comparison

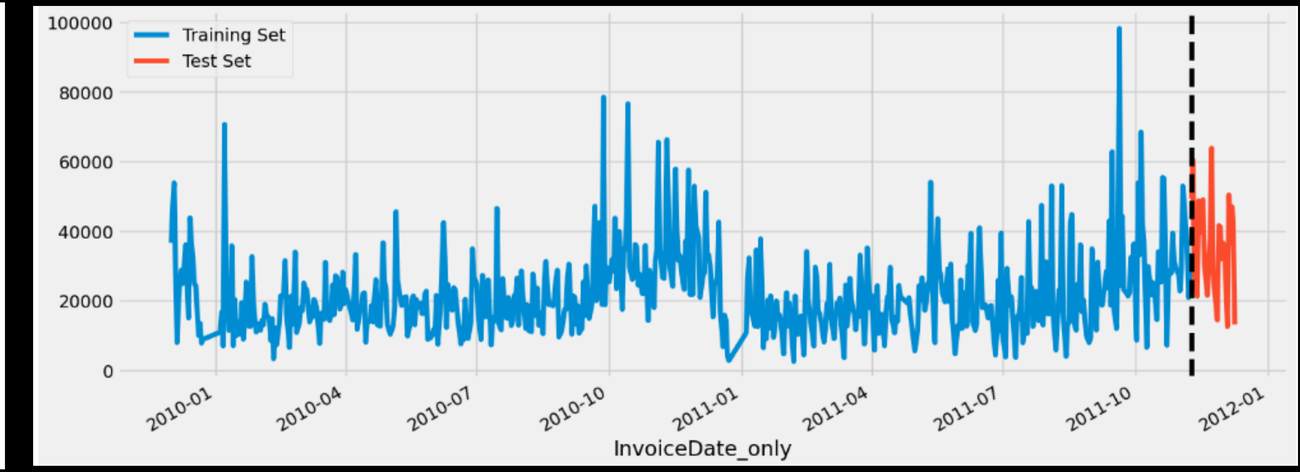
## Cluster 0

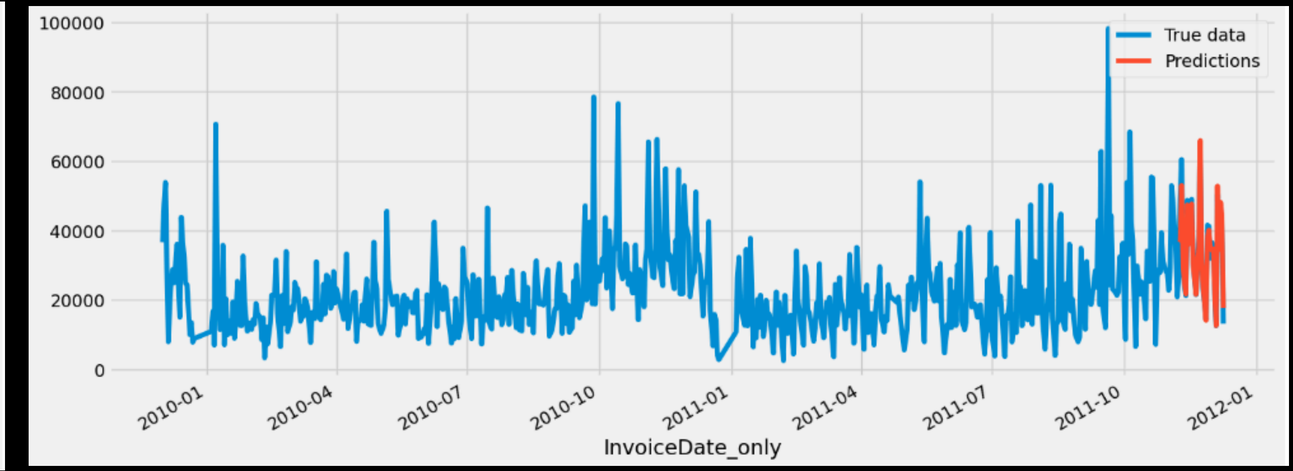


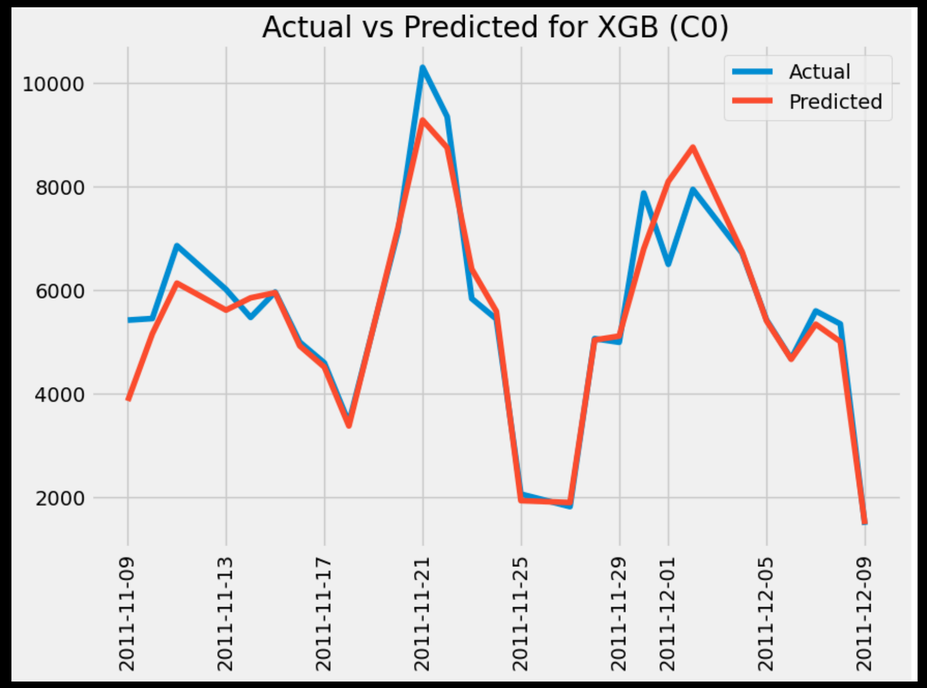

Decent RMSE for  598 and MAPE=6.21

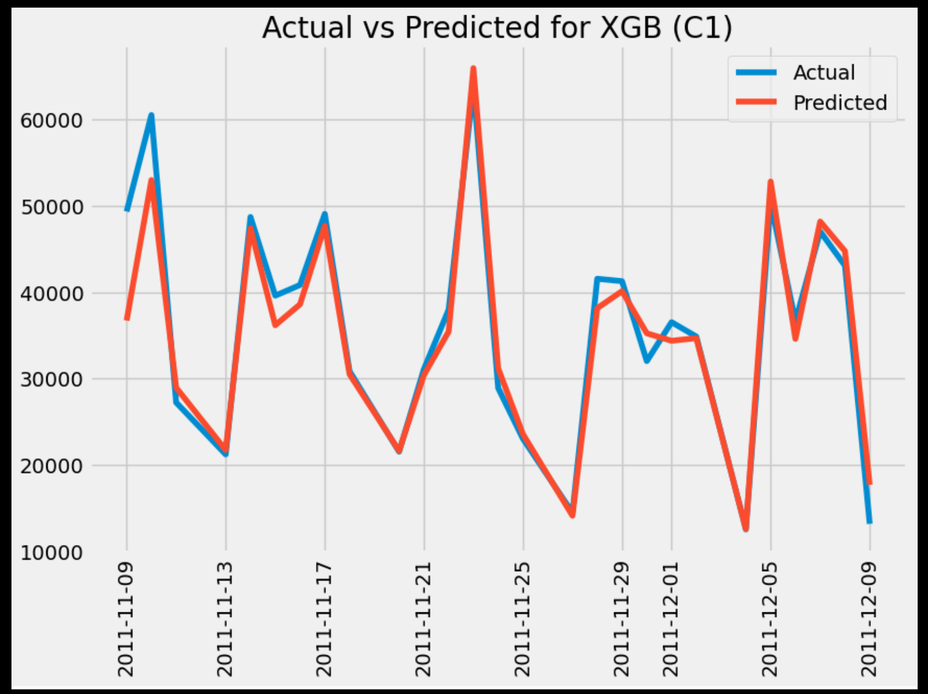
## Cluster 1



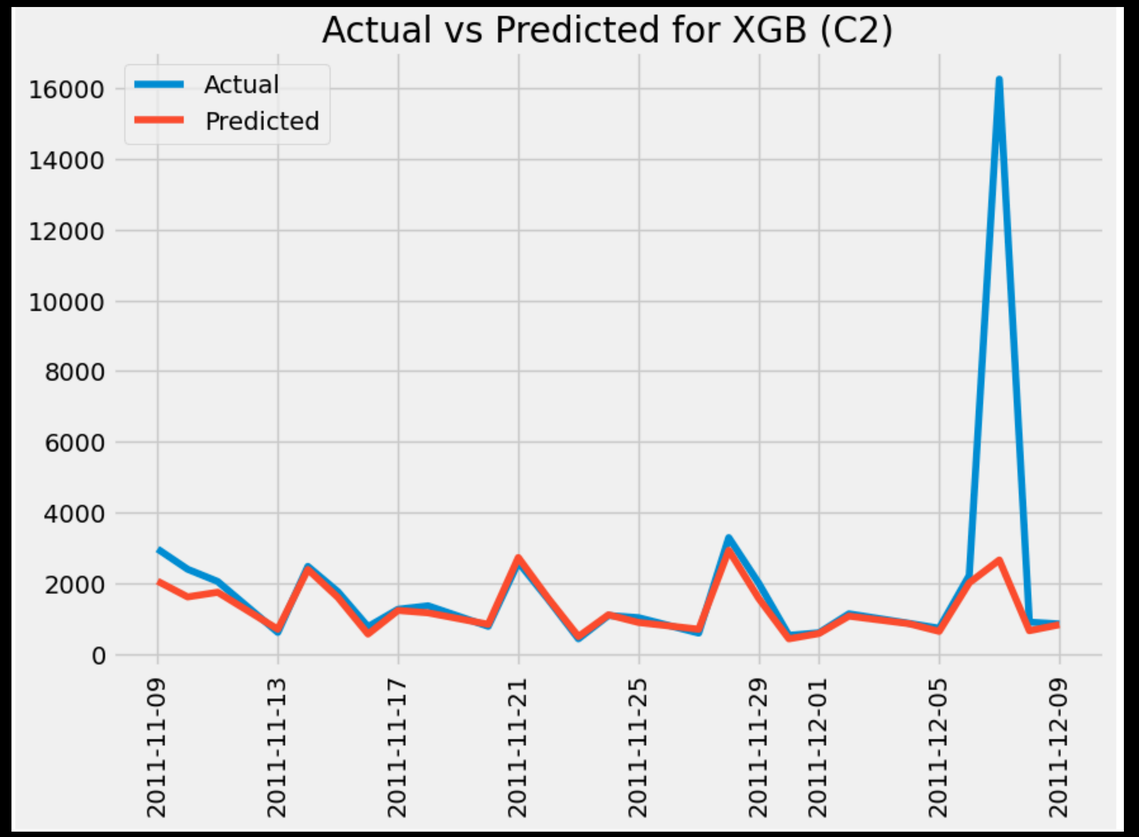




Decent RMSE for  3430.71 and MAPE=6.32




## Cluster 2




RMSE for 2631.98 and MAPE=15.12

Note : This model couldn't predict for last cluster since this the group of lost customers who didn't purchase for a long time; last purchase being in 2011 Q1

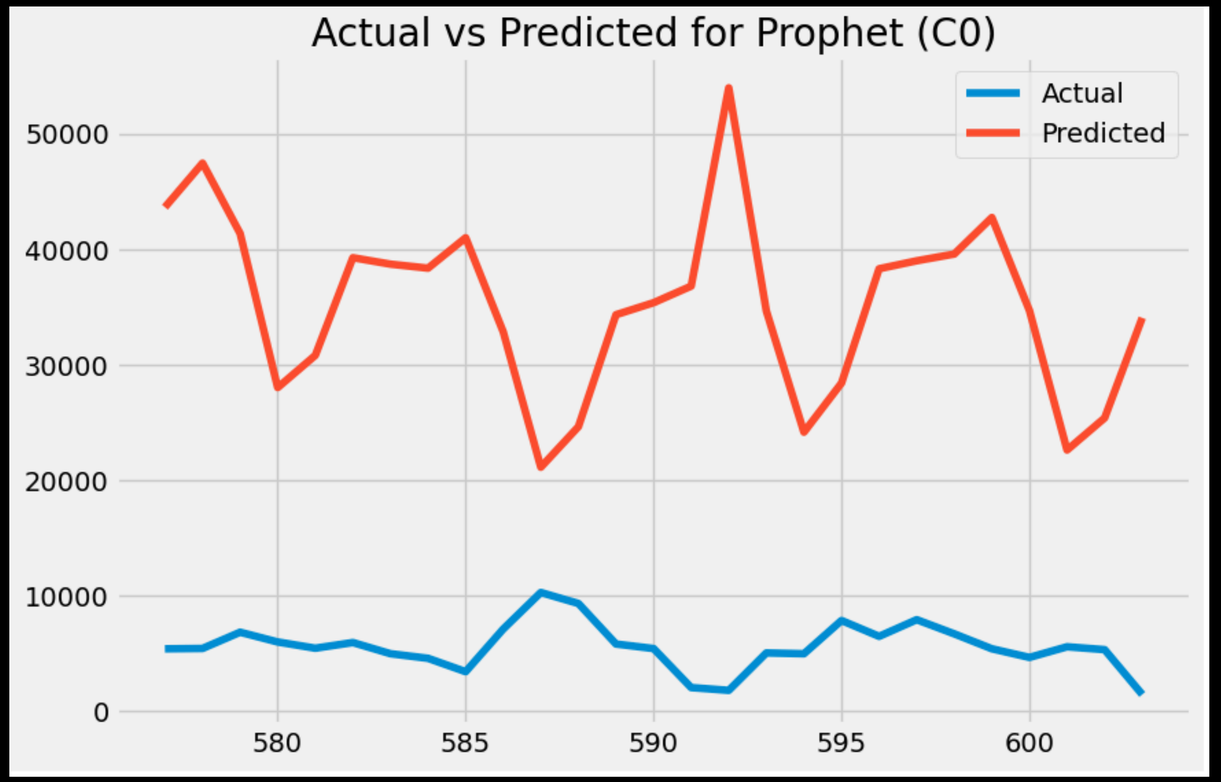
# Time Series Modeling / Forecasting

XGB Method

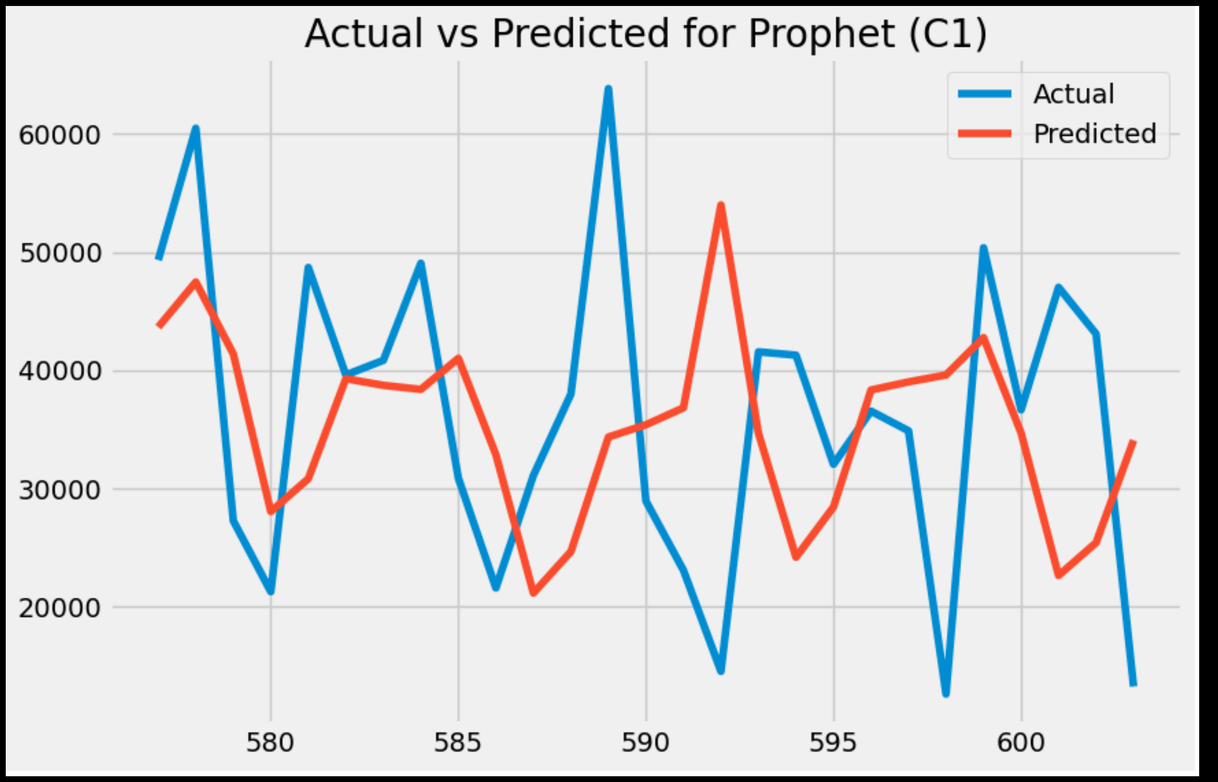
FB Prophet

# Using Prophet

- Used last 27 days as the testing dataset
- Prophet handles date features under the hood with addition of custom regressors & country holidays ; also created a feature rich data frame with upper + lower confidence levels




Cluster 0 : RMSE 2770




Cluster 1 : RMSE 15580

# Future Work + Improvements *(Clustering)*

- Augmenting the data used for clustering to include Product purchase details at customer level rather than just RFM derived metrics. For example, breaking down the description like so to group the nouns & find the occurrence frequencies to see popular products

| | Description | Description_m1 | POS Tagged Text | nouns_text |
|---|---|---|---|---|
| 0 | 15CM CHRISTMAS GLASS BALL 20 LIGHTS | 15cm christmas glass ball 20 light | [(15cm, CD), (christmas, JJ), (glass, NN), (ba... | [glass, ball, light] |
| 1 | PINK CHERRY LIGHTS | pink cherry light | [(pink, NN), (cherry, NN), (light, NN)] | [pink, cherry, light] |
| 2 | WHITE CHERRY LIGHTS | white cherry light | [(white, JJ), (cherry, NN), (light, NN)] | [cherry, light] |
| 3 | RECORD FRAME 7" SINGLE SIZE | record frame 7 single size | [(record, NN), (frame, NN), (7, CD), (single, ... | [record, frame, size] |
| 4 | STRAWBERRY CERAMIC TRINKET BOX | strawberry ceramic trinket box | [(strawberry, JJ), (ceramic, JJ), (trinket, NN... | [trinket, box] |
| 5 | PINK DOUGHNUT TRINKET POT | pink doughnut trinket pot | [(pink, NN), (doughnut, NN), (trinket, NN), (p... | [pink, doughnut, trinket, pot] |
| 6 | SAVE THE PLANET MUG | save planet mug | [(save, JJ), (planet, NN), (mug, NN)] | [planet, mug] |
| 7 | FANCY FONT HOME SWEET HOME DOORMAT | fancy font home sweet home doormat | [(fancy, JJ), (font, NN), (home, NN), (sweet, ... | [font, home, sweet, home, doormat] |
| 8 | CAT BOWL | cat bowl | [(cat, NN), (bowl, NN)] | [cat, bowl] |
| 9 | DOG BOWL , CHASING BALL DESIGN | dog bowl chasing ball design | [(dog, NN), (bowl, NN), (chasing, VBG), (ball,... | [dog, bowl, ball, design] |

| | |
|---|---|
| bag | 70136 |
| heart | 67830 |
| box | 44066 |
| design | 43520 |
| vintage | 42029 |
| retrospot | 41792 |
| cake | 32372 |
| metal | 30971 |
| christmas | 29739 |
| pink | 28528 |
| holder | 28252 |
| sign | 26935 |
| lunch | 24654 |
| pack | 23548 |
| paper | 23273 |
| card | 21242 |
| tin | 20711 |
| glass | 19898 |
| case | 18954 |
| decoration | 18703 |

- This would also need categorizing them in semantic groups to find a pattern that could help with recommendation/ strategy to improve sales

# Future Work + Improvements *(Time Series Modeling)*

- Combine the best of both worlds : Use the Prophet generate data frame with the XGB data frame (with lagged variables and rolling means information) & then subject to any Gradient Boosting method to see if the results have improved.

- As an additional regressor, we could also add a ‘Cyber Monday’ regressor function that is often important from retail standpoint & could lead to high fluctuations. This probably warrants a bit more of domain knowledge

- It was a personal preference to stick with explainability where we can see the weightage of different factors on the results but a worthwhile & effective alternative might be LSTM (NN) or NBEATs model (deep learning methods)