Data Analytics

Project Presentation

Name : Purvik S Nukal
SRN: PES1UG20CS315
Name : Rahul Ranganath
SRN: PES1UG20CS316

Abstract and Scope

- Every day a huge number of transactions take place across multiple e-commerce platforms connecting a number of retail stores and a huge amount of data from the customer transaction is generated accordingly. This data generated is not utilized effectively. The data can be utilized to draw meaningful conclusions and predictions about the viability and profitability of a product.
- Due to rapid technological improvements, increased customer choices, and product customization, there is a constant fluctuation in consumer demand. This leads to an increase in uncertainty among retail store owners resulting in poor decision-making and a huge loss in revenue.

Abstract and Scope

- The main objective of the project is to build a Business Growth recommendation system for retail store owners in order to provide them with insights into what may be the necessary steps that need to be taken to increase profits and help them in identifying all the necessary steps and adopt strategies to minimize expenses by identifying the products that are high in demand in the market and take the longest to sell.
- The model gives recommendations to the retail store owners based on the analysis of customer purchase activity of an item like transaction history, frequency of selling a product, the monetary value of customers, and recency of transaction. Ultimately, the retail store owners can achieve a big boost in sales and avoid poor decision-making with respect to the selling of products.

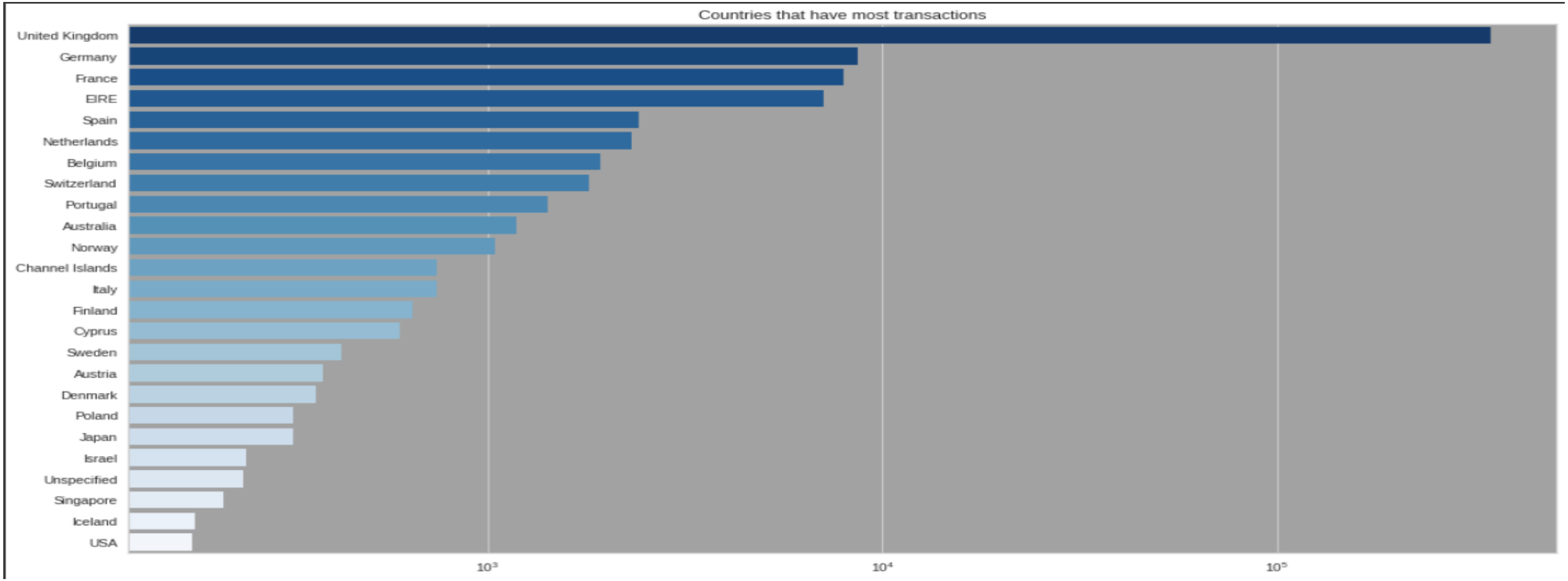
Table of Contents:

Preprocessing and Exploratory Data Analysis

Data was imported and cleaned thus removing all missing values, duplicates, and corrupted values present in the dataset.
Data Cleaning is followed by Exploratory Data Analysis by plotting bar charts, boxplots and outliers.

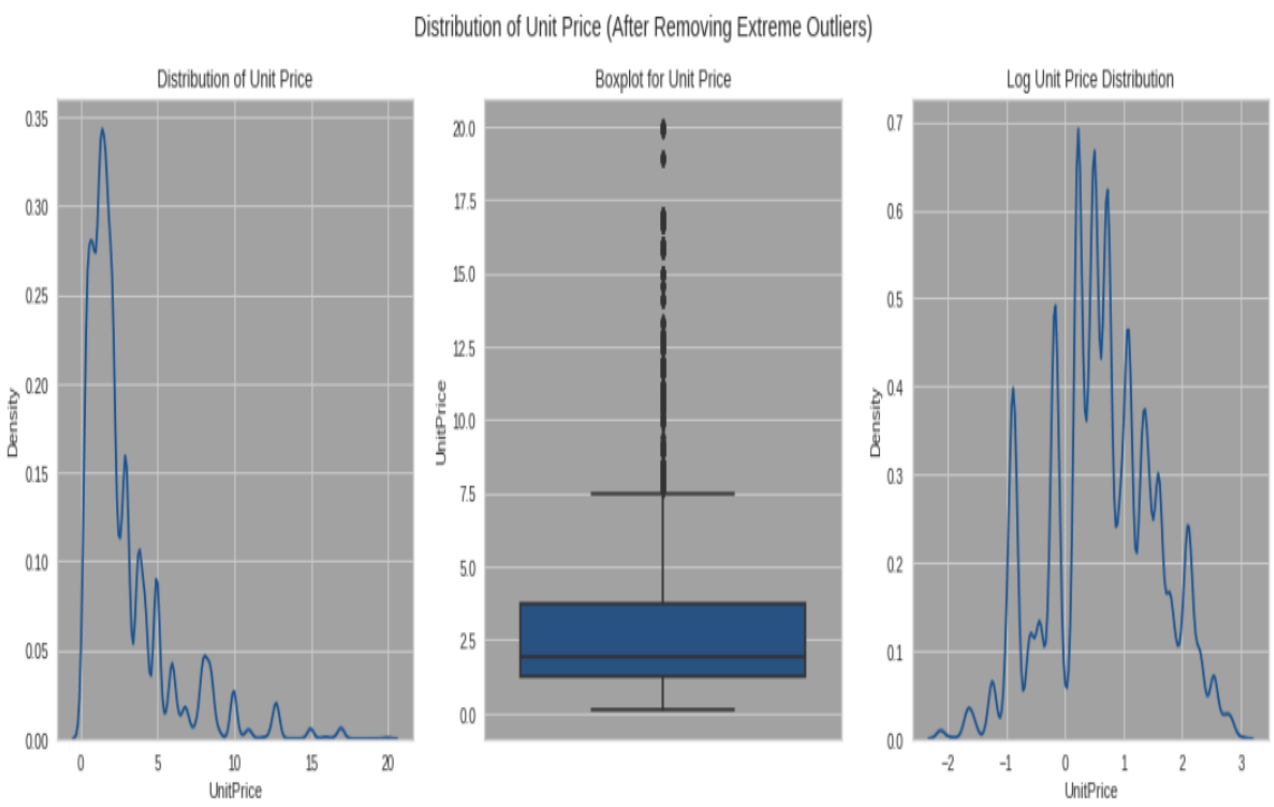
Preprocessing and Exploratory Data Analysis

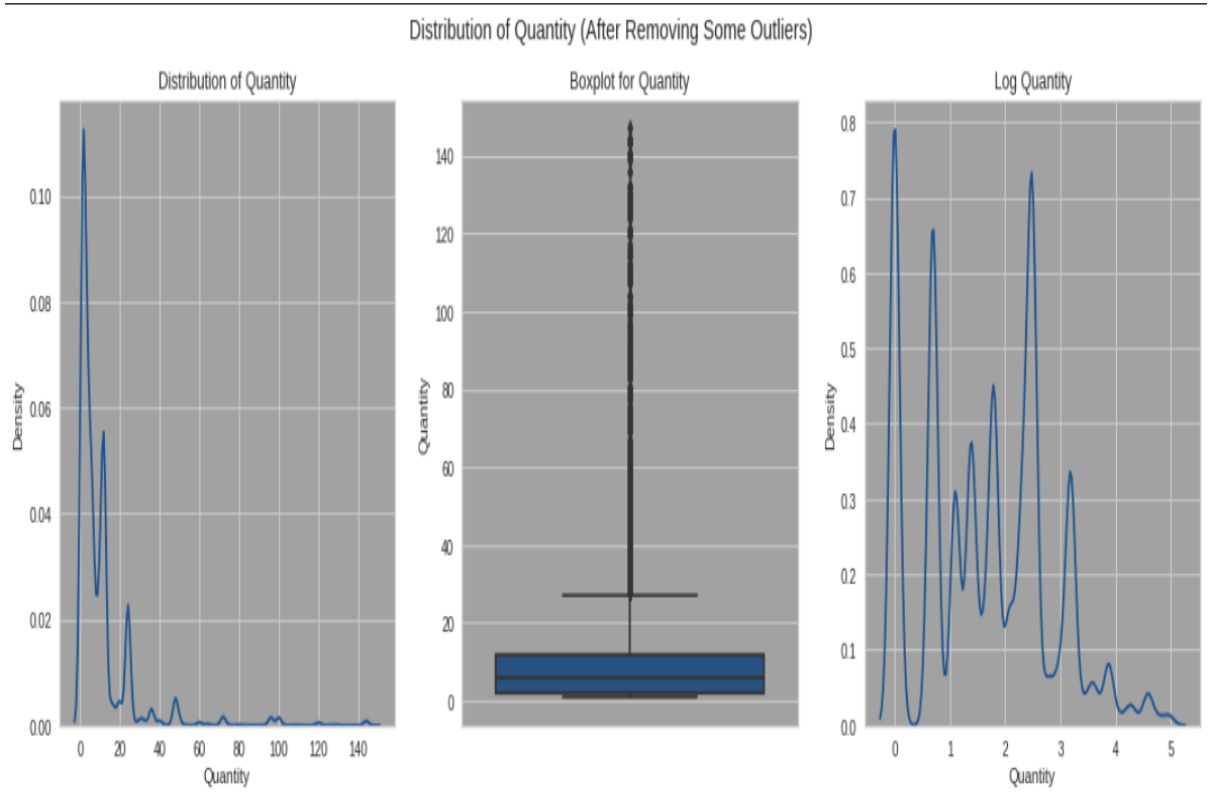
We observe that out of all the countries United Kingdom comes out on top in terms of the number of transactions



Preprocessing and Exploratory Data Analysis

Outlier Analysis was done using boxplots where extreme outliers were removed

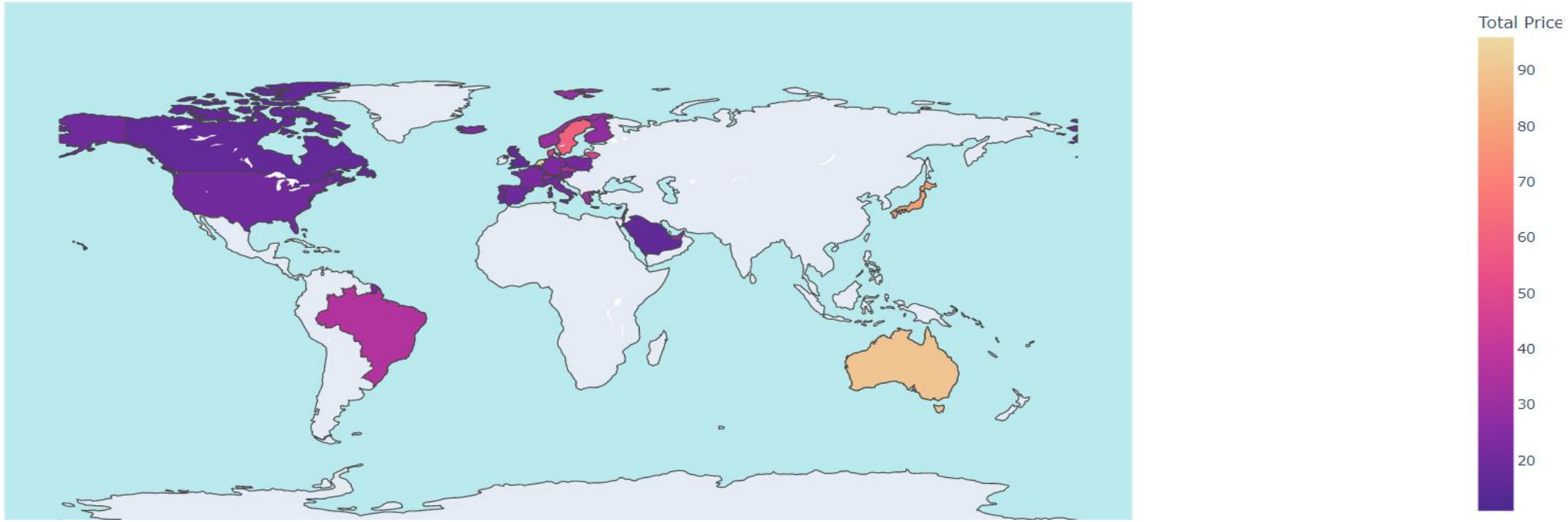




Preprocessing and Exploratory Data Analysis

There are four countries that tend to buy have an average cart bigger than the others: the Netherlands, Australia, Japan, and Sweden. Based on the plots for quantity and total prices, it seems that those countries have bigger carts because of quantities, a customer from those countries will buy more quantities than in others.
Hong Kong and Singapore customers tend to be more attracted by expensive items.

Average Total Price Per Cart By Country



Cohort Analysis

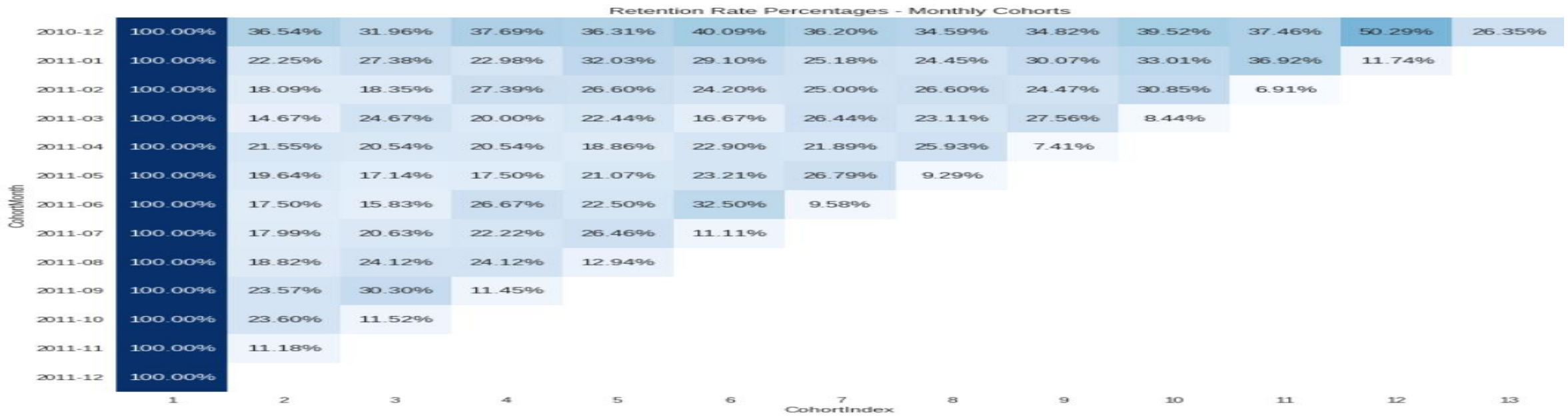
A cohort simply means that a group of people have the same characteristics. Our dataset contains invoice records for more than one year. Let's apply cohort analysis. We can create monthly cohorts. We will group customers for the first invoice record. The cohort index will be the number of months since the first transaction.

CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12-01	873.0	319.0	279.0	329.0	317.0	350.0	316.0	302.0	304.0	345.0	327.0	439.0	230.0
2011-01-01	409.0	91.0	112.0	94.0	131.0	119.0	103.0	100.0	123.0	135.0	151.0	48.0	NaN
2011-02-01	376.0	68.0	69.0	103.0	100.0	91.0	94.0	100.0	92.0	116.0	26.0	NaN	NaN
2011-03-01	450.0	66.0	111.0	90.0	101.0	75.0	119.0	104.0	124.0	38.0	NaN	NaN	NaN
2011-04-01	297.0	64.0	61.0	61.0	56.0	68.0	65.0	77.0	22.0	NaN	NaN	NaN	NaN
2011-05-01	280.0	55.0	48.0	49.0	59.0	65.0	75.0	26.0	NaN	NaN	NaN	NaN	NaN
2011-06-01	240.0	42.0	38.0	64.0	54.0	78.0	23.0	NaN	NaN	NaN	NaN	NaN	NaN
2011-07-01	189.0	34.0	39.0	42.0	50.0	21.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08-01	170.0	32.0	41.0	41.0	22.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09-01	297.0	70.0	90.0	34.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10-01	356.0	84.0	41.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11-01	322.0	36.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12-01	39.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

The above data is our cohort table. Its interpretation is simple. For example, We have 873 unique customers with whose first transaction is in 2010-12. Its cohort month is 2010-12 and its cohort index is 1. Go to the one right cell, it is 319. It means, 319 unique customers retain their customer ship for next month.

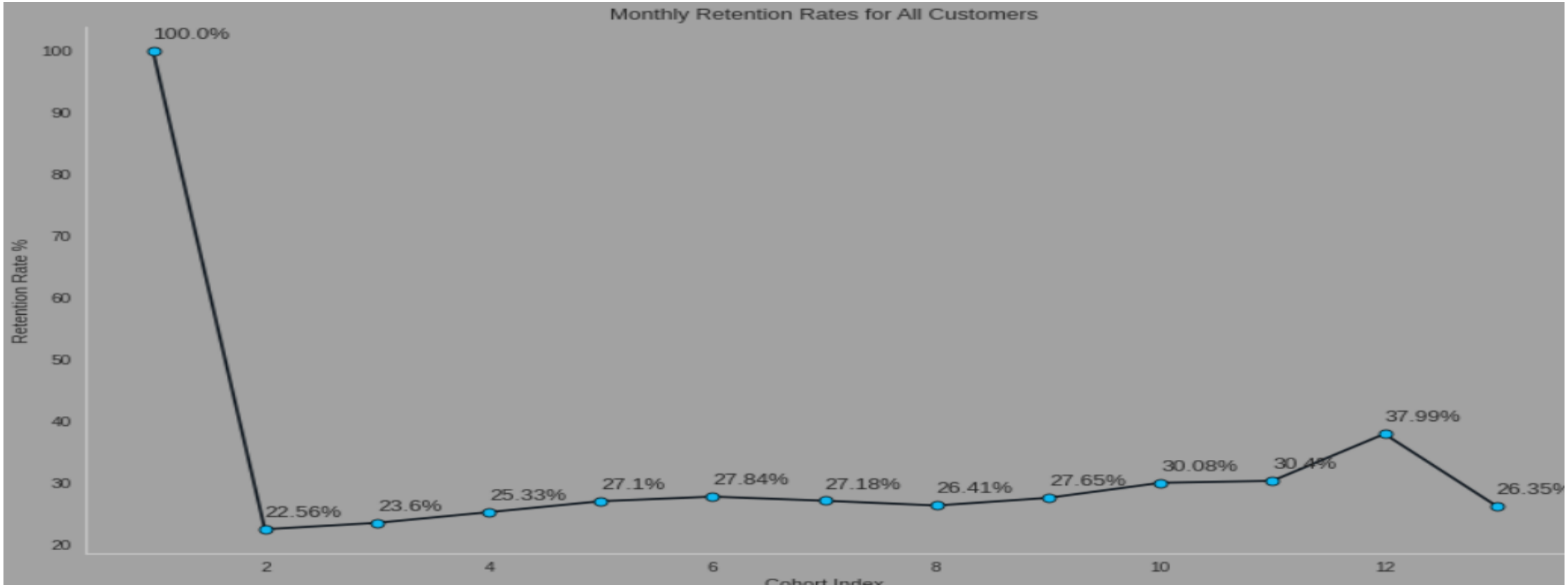
Retention Rate

Retention tables show a group of people that visited your site or used your app for the first time during a certain time frame. They also display the progressive ‘drop-off’ or decline in activity over time for that particular group (a cohort).



Let's look at above retention rate chart and interpret it. 40.05% of customers that made their first shopping in January 2011, use this company after five months.

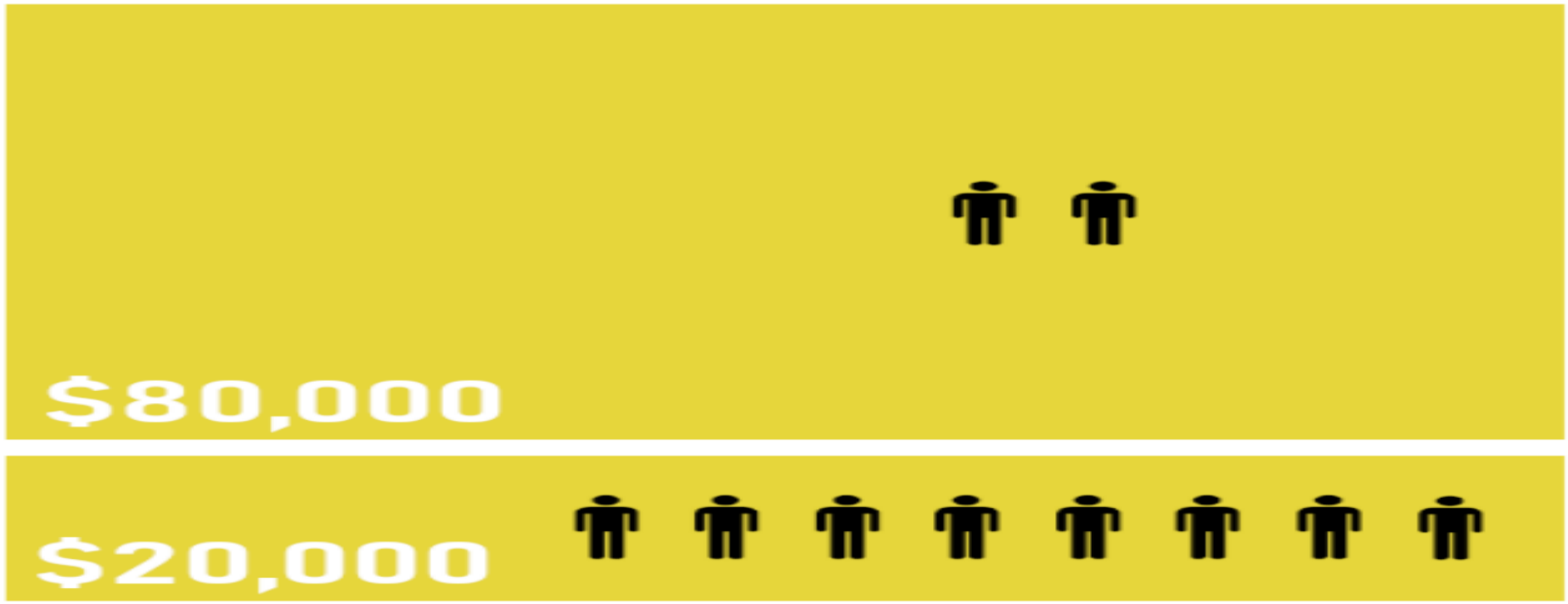
Retention Rate



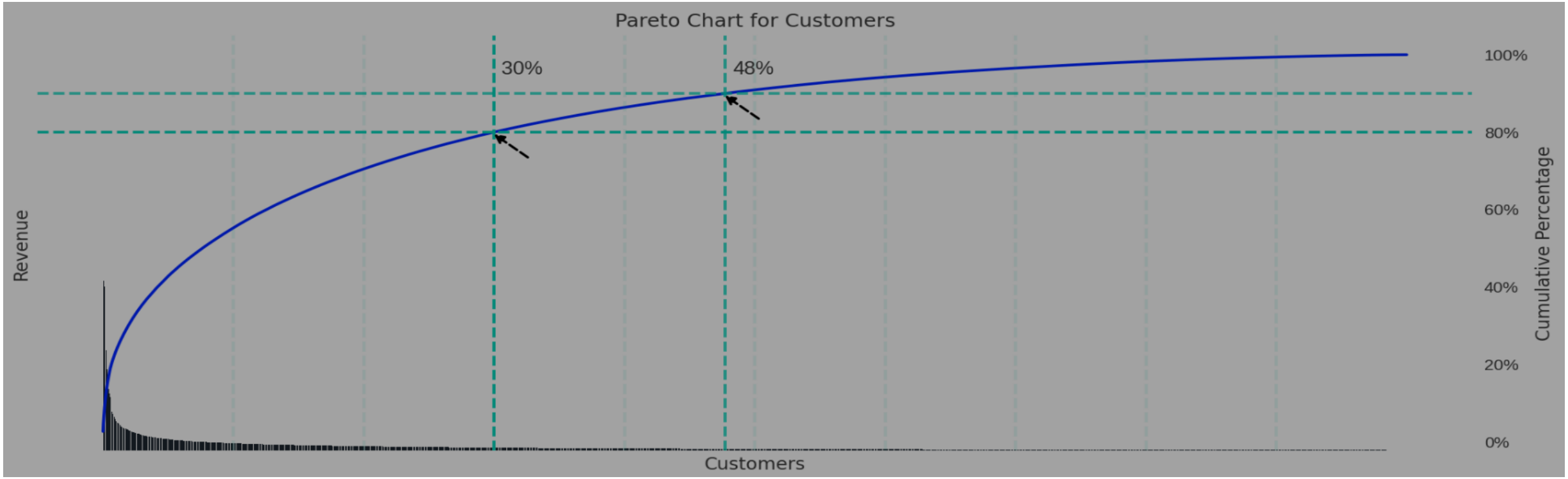
Retention rate increases significantly for last months of the year. Probably, Thanksgiving, Black Friday and Christmas causes it. There are lots of special day at the end of year.

Pareto Principle

The Pareto principle states that for many outcomes, roughly 80% of consequences come from 20% of causes (the “vital few”).Other names for this principle are the 80/20 rule, the law of the vital few, or the principle of factor sparsity.
Lets implement Pareto's 80-20 rule to our dataset. We have two hypothesis:
1) 80% of company's revenue comes from 20% of total customers.
2) 80% of company's revenue comes from 20% of total products.

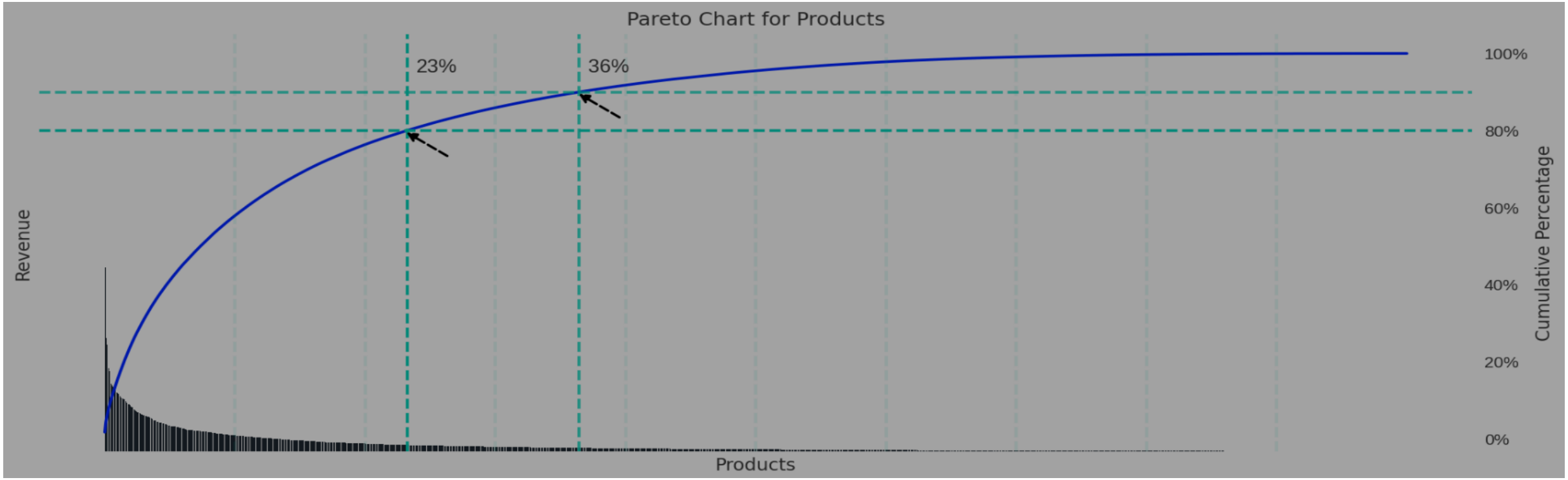


Pareto Principle for Customers



We can see that 80% of company's revenue comes from top 30% of customers. Also, 90% of company's revenue comes from top 48% of customers.

Pareto Principle for Products



We can see that 80% of company's revenue comes from top 23% of products that have most revenue. Also, 90% of company's revenue comes from 36% of products that have most revenue.

Maybe, if the company reduce by half its variety of items, revenue couldn't decrease significantly.

Segmentation of Customers using RFM Analysis

RFM analysis is a marketing technique used to quantitatively rank and group customers based on the recency, frequency and monetary total of their recent transactions to identify the best customers and perform targeted marketing campaigns.
The Segments in RFM Analysis are:

Champions: Bought recently, buy often and spend the most

Loyal customers: Buy on a regular basis. Responsive to promotions.

Potential loyalist: Recent customers with average frequency.

New customers: Bought most recently, but not often.

Promising: Recent shoppers, but haven't spent much.

Needs attention: Above average recency, frequency and monetary values. May not have bought very recently though.

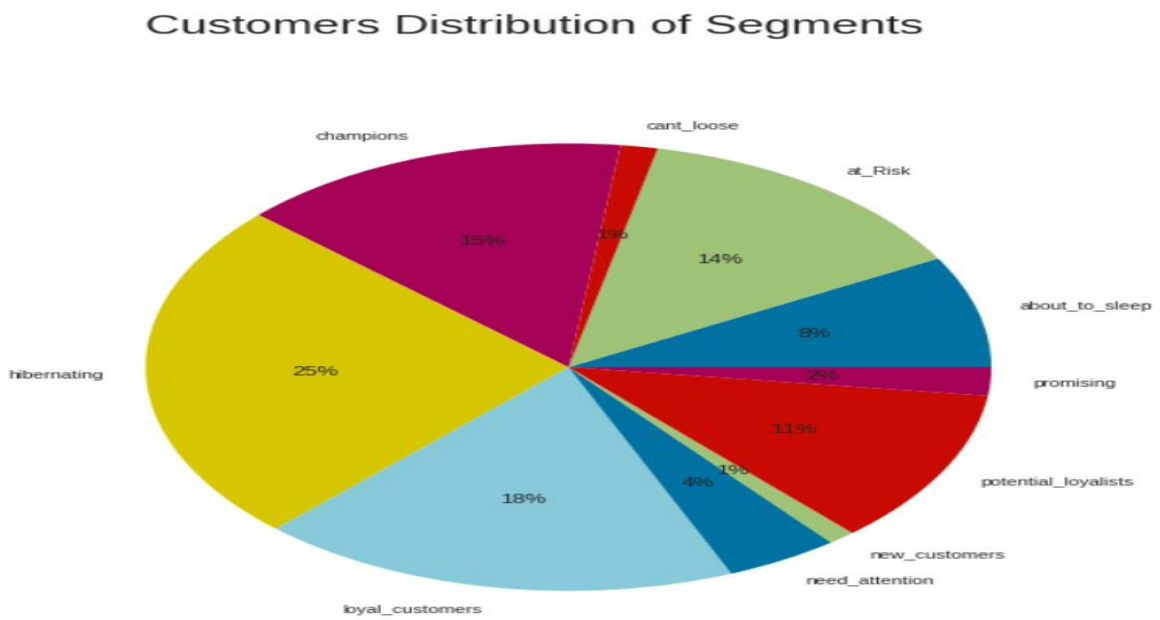
About to sleep: Below average recency and frequency. Will lose them if not reactivated.

At risk: Some time since they've purchased. Need to bring them back!

Can't lose: Used to purchase frequently but haven't returned for a long time.

Hibernating: Last purchase was long back and low number of orders. May be lost.

Segmentation of Customers using RFM Analysis



15% of customers considered as Champions. These customers are responsible for a big share of your revenue so we can put a lot of effort into keeping improving their experience. What we can do: Give them something extra that the regulars do not get, for example, limited series of products or special discounts to make them feel valued. Use communication similar to the "Loyal" segment. For example, making them ambassadors, giving them a margin of your profits for bringing you, new customers. Ask them for feedbacks as they might know the products and services very well.

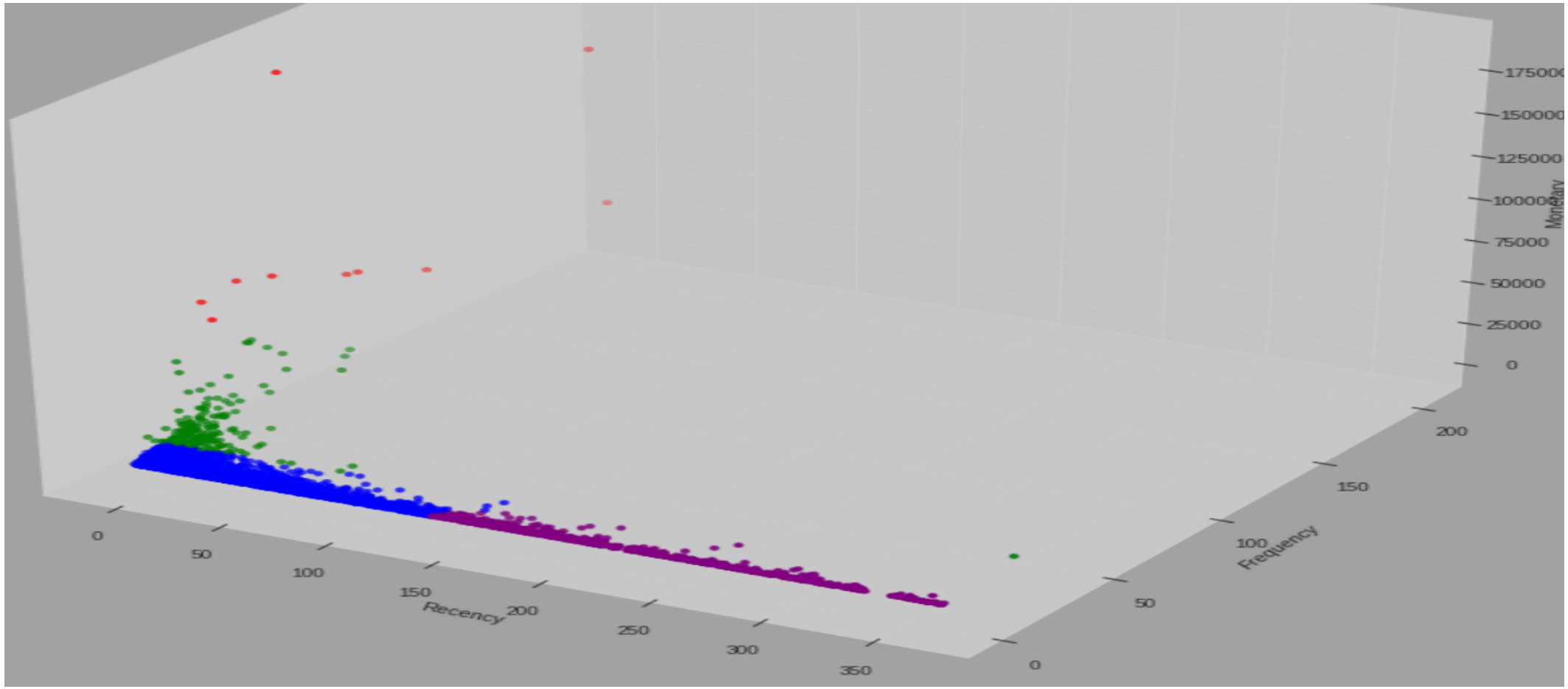
Clustering - Segmentation of Customers using K-Means Clustering

K-Means Clustering is an unsupervised learning algorithm that is used to solve clustering problems in machine learning or data science. It allows us to cluster the data into different groups and is a convenient way to discover the categories of groups in the unlabelled dataset on its own without the need for any training. It is a centroid-based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters. The optimal value of k is determined using the **Elbow method** and **silhouette score**.

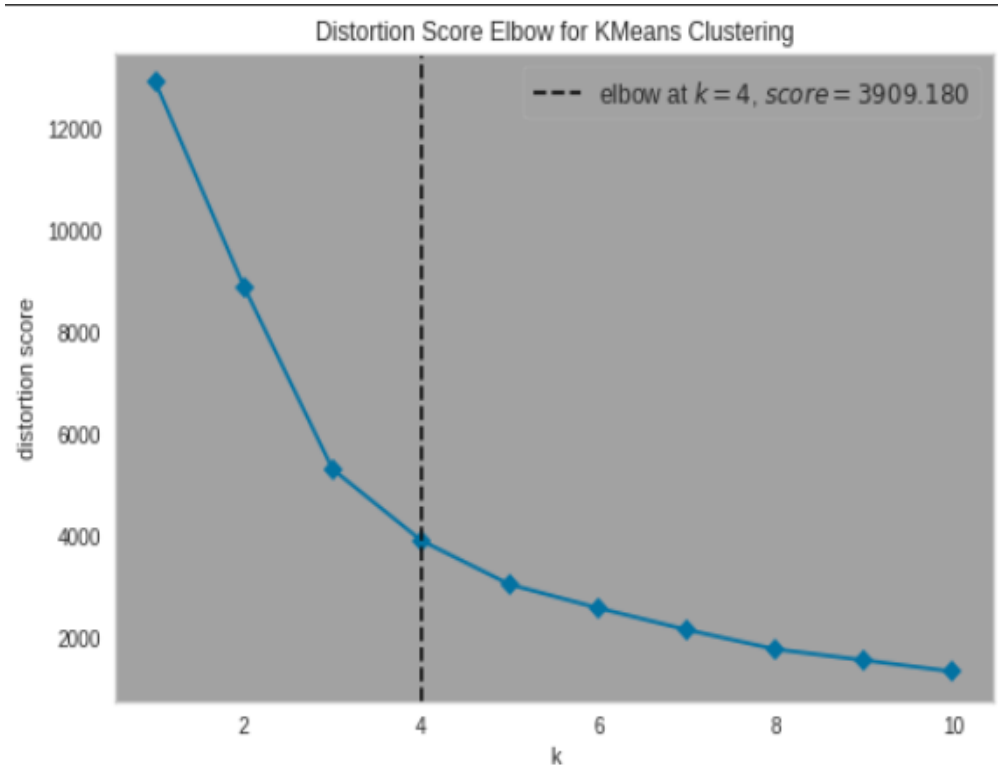
Algorithm 1 k-means algorithm

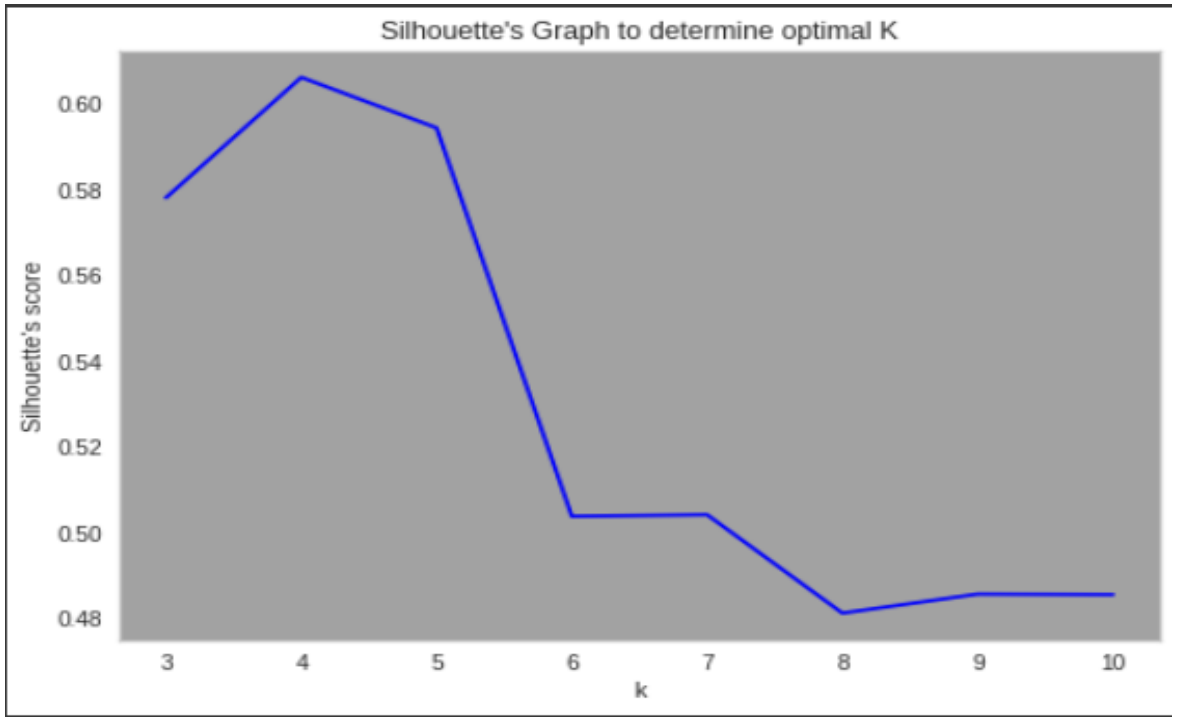
1: Specify the number k of clusters to assign.
2: Randomly initialize k centroids.
3: **repeat**
4: **expectation:** Assign each point to its closest centroid.
5: **maximization:** Compute the new centroid (mean) of each cluster.
6: **until** The centroid positions do not change.

Clustering - Segmentation of Customers using K-Means



Clustering - Segmentation of Customers using K-Means





Clustering - Segmentation of Customers using K-Means

From the KMeans clustering, we can sort every customer into 4 different clusters that seem to have different behaviors.

Cluster 0: "Punctual customers" - Those are the customers that buy more punctual items on the website.

Cluster 1: "Hibernating customers" - Those are the customers that buy at the lowest frequency, the lowest recently and that spend the least money.

Cluster 2: "Exceptional customers" - Those are the customers that we want to keep, that buy at the highest frequency, the most recent and that spend the most money.

Cluster 3: " Recent customers" - Those are customers that have been active quite recently that might be interesting to keep stimulated.

Selection of Frequently Purchased Itemsets using Apriori Algorithm

The Apriori algorithm calculates the frequently bought item sets based on a minimum support threshold value and generates association rules like confidence and lift. It is built on the idea that the subset of a frequently bought items set is also a frequently bought item set.

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(60 teatime fairy cake cases)	(pack of 60 pink paisley cake cases)	0.035644	0.035521	0.015206	0.426598	12.009794	0.013940	1.682028
1	(pack of 60 pink paisley cake cases)	(60 teatime fairy cake cases)	0.035521	0.035644	0.015206	0.428076	12.009794	0.013940	1.686162
2	(60 teatime fairy cake cases)	(pack of 72 retrospot cake cases)	0.035644	0.051404	0.017422	0.488774	9.508552	0.015590	1.855531
3	(pack of 72 retrospot cake cases)	(60 teatime fairy cake cases)	0.051404	0.035644	0.017422	0.338922	9.508552	0.015590	1.458763
4	(alarm clock bakelike green)	(alarm clock bakelike ivory)	0.042231	0.026040	0.015267	0.361516	13.882899	0.014167	1.525425
...	...	...	...	...	...	...	...	...	...
211	(lunch bag cars blue, lunch bag pink polkadot)	(lunch bag red retrospot)	0.024378	0.068887	0.015637	0.641414	9.311109	0.013957	2.596625
212	(lunch bag red retrospot, lunch bag pink polka...	(lunch bag cars blue)	0.028995	0.054235	0.015637	0.539278	9.943285	0.014064	2.052789
213	(lunch bag cars blue)	(lunch bag red retrospot, lunch bag pink polka...	0.054235	0.028995	0.015637	0.288309	9.943285	0.014064	1.364362
214	(lunch bag red retrospot)	(lunch bag cars blue, lunch bag pink polkadot)	0.068887	0.024378	0.015637	0.226988	9.311109	0.013957	1.262105
215	(lunch bag pink polkadot)	(lunch bag cars blue, lunch bag red retrospot)	0.052204	0.025425	0.015637	0.299528	11.780963	0.014309	1.391313

Selection of Frequently Purchased Item sets using Apriori Algorithm

Association Rules
Support

It tells us about the combination of items bought together frequently. It gives the part of transactions that contain both items (say A and B)

support = occurrence of item / total no of transaction.

Confidence

It tells us how likely B will be purchased provided A is purchased, for the no. of times A is bought.

confidence = support (X Union Y) / support(X).

Lift

It indicates the strength of a rule over the randomness of A and B being bought together. It basically measures the strength of any association rule

lift = support (X Union Y)/ support(X) * support(Y) .

Selection of Frequently Purchased Item sets using Apriori Algorithm

For a certain threshold of confidence and lift, the below item sets are filtered out.
In this way, all the required item sets can be found. Further fine-tuning of the item sets to find the more probable ones can be done by increasing the threshold values of support and lift.

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
36	(pink regency teacup and saucer)	(green regency teacup and saucer)	0.030288	0.037614	0.024809	0.819106	21.776682	0.023670	5.320157
181	(roses regency teacup and saucer , pink regenc...	(green regency teacup and saucer)	0.023455	0.037614	0.020869	0.889764	23.655193	0.019987	8.730217

Revenue Forecasting using Prophet Algorithm

Time series forecasting can be challenging as there are many different methods you could use and many different hyperparameters for each method.
The Prophet library is an open-source library designed for making forecasts for univariate time series datasets.
It is easy to use and designed to automatically find a good set of hyperparameters for the model in an effort to make skillful forecasts for data with trends and seasonal structure by default.
Prophet gives lots of options in the model-building step.

holidays: You can use this for adding special days to the model.

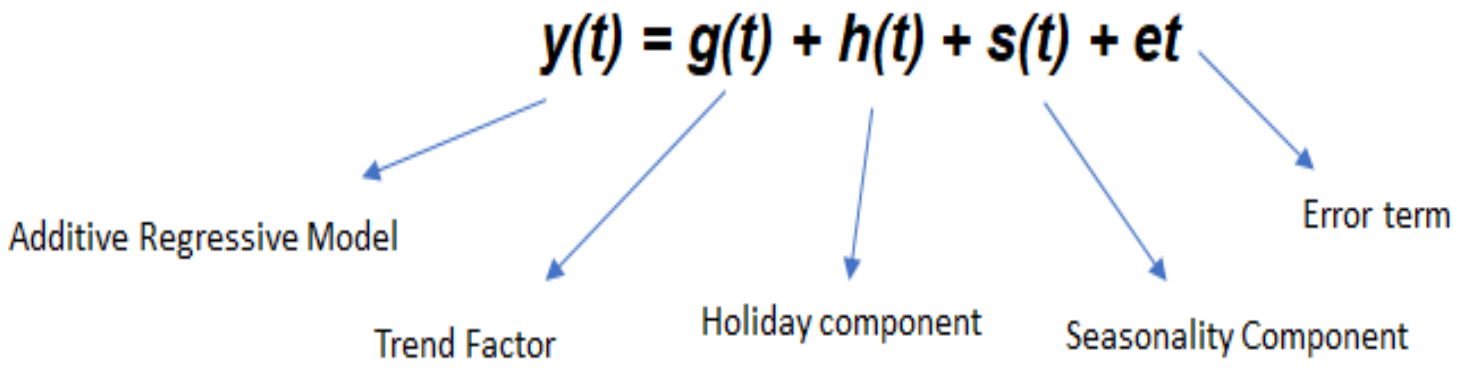
seasonality: If the data has seasonality, you can set daily seasonality, weekly seasonality, and yearly seasonality parameters to True.

_**prior_scale**: This parameter controls the flexibility of components' effects.

Revenue Forecasting using Prophet Algorithm

The Prophet model has the form $y(t)=g(t)+s(t)+h(t)+\varepsilon$, where:

- $g(t)$ is the trend function.
- $s(t)$ is the periodic component (seasonalities)
- $h(t)$ represents holidays/events which occur on a potentially irregular basis.
- ε is the error term (which is often assumed to be normally distributed)



Revenue Forecasting using Prophet Algorithm

Hyperparameters

- **holidays**
- **n_changepoints** is the number of changes that happen in the data. The prophet model detects them on its own. By default, its value is 25, which is uniformly placed in the first 80% of the time series. Changing *n_changepoints* can add value to the model.
- **changepoint_prior_scale** to indicate how flexible the changepoints are allowed to be. In other words, how much can the changepoints fit into the data. If you make it high it will be more flexible, but you can end up overfitting. By default, this parameter is set to 0.05
- **Seasonality_mode** There are 2 types of model seasonality mode. **Additive & multiplicative**. By default Prophet fits additive seasonalities, meaning the effect of the seasonality is added to the trend to get the forecast. Prophet can model multiplicative seasonality by setting seasonality_mode='multiplicative' in the model.
- **holiday_prior_scale** just like changepoint_prior_scale, holiday_prior_scale uses smoothening the effect of holidays. By default, its value is 10, which provides very little regularization. Reducing this parameter dampens holiday effects
- **Seasonalities with the fourier_order** Prophet model, by default finds the seasonalities and add the default parameters of the seasonality. We can modify the seasonalities effect by adding custom seasonalities as add_seasonality in the model with different Fourier orders. By default, Prophet uses a Fourier order of 3 for weekly seasonality and 10 for yearly seasonality.

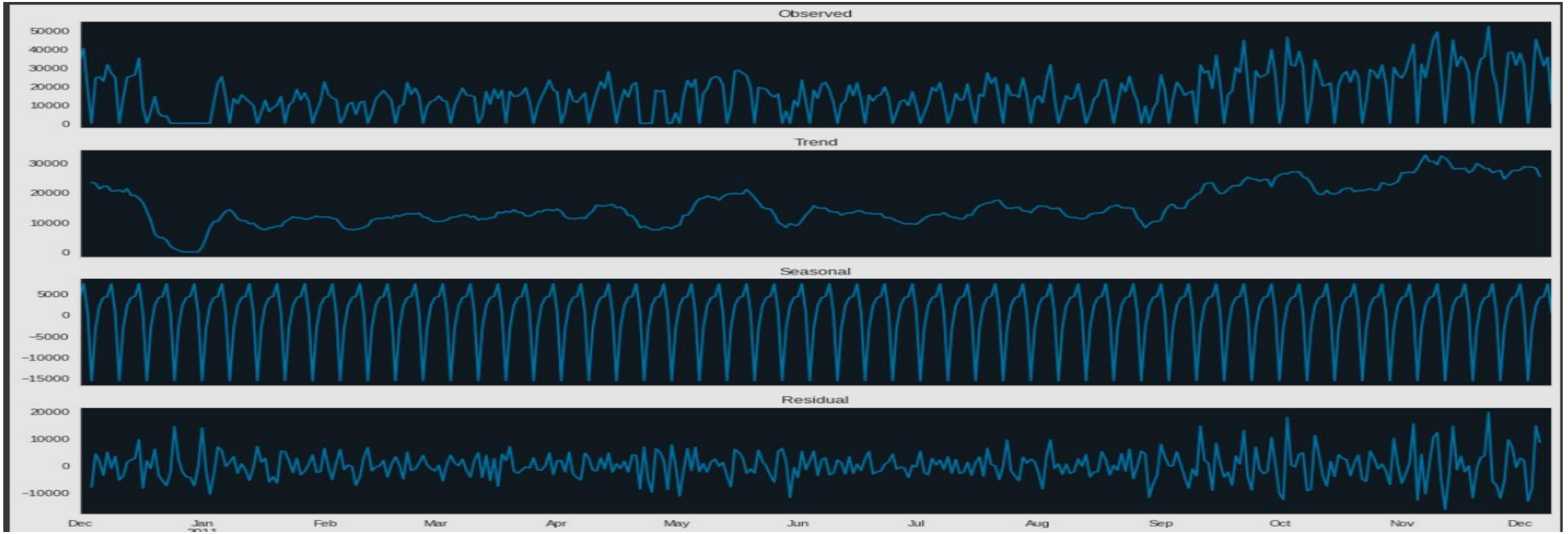
Revenue Forecasting using Prophet Algorithm

We have used the following values for the hyperparameters in our model.

```
model = Prophet(
    holidays = black_friday,
    daily_seasonality = True,
    weekly_seasonality = True,
    holidays_prior_scale = 1,
    seasonality_prior_scale = 5,
    changepoint_prior_scale = 1,
)
```

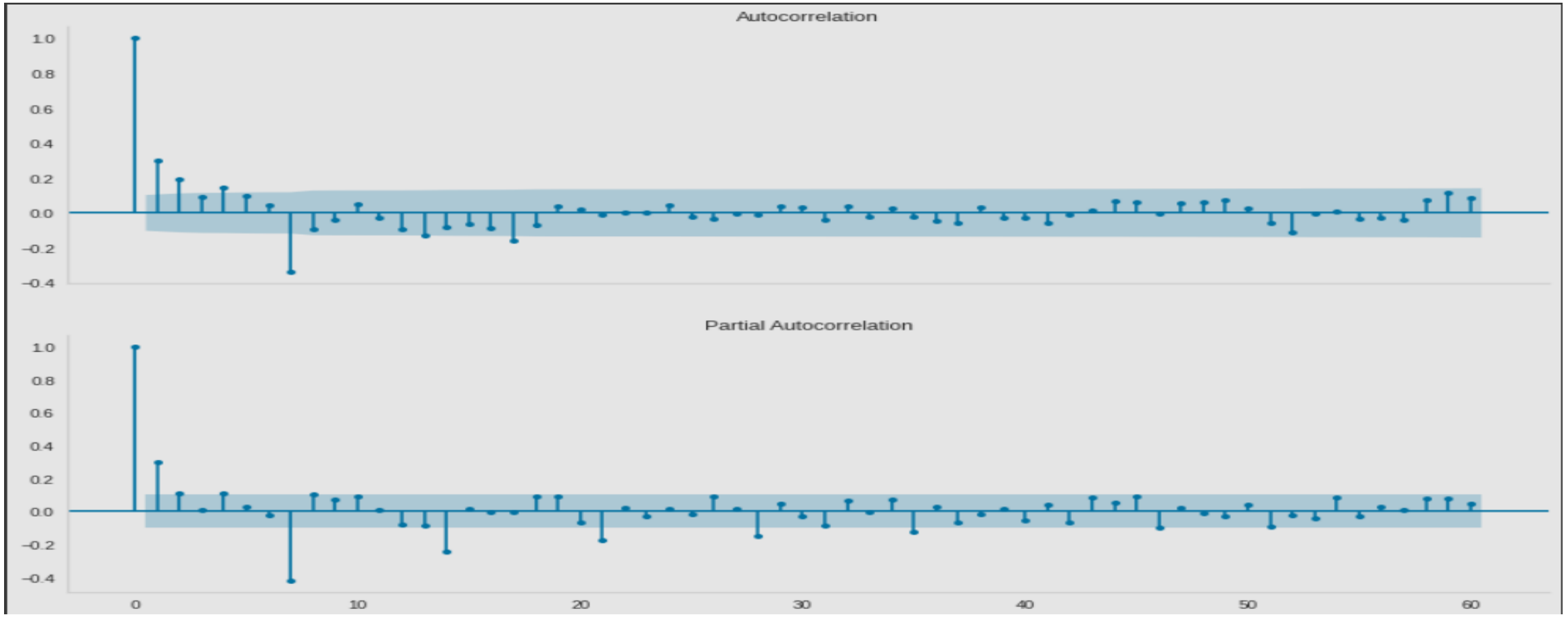
Revenue Forecasting using Prophet Algorithm

The series is checked for stationarity using the Augmented Dickey-Fuller Test and when not found stationary the time series is decomposed into the trend, seasonality, and residual plots.
We have used differencing with a period of 7 days to convert it into stationary series.



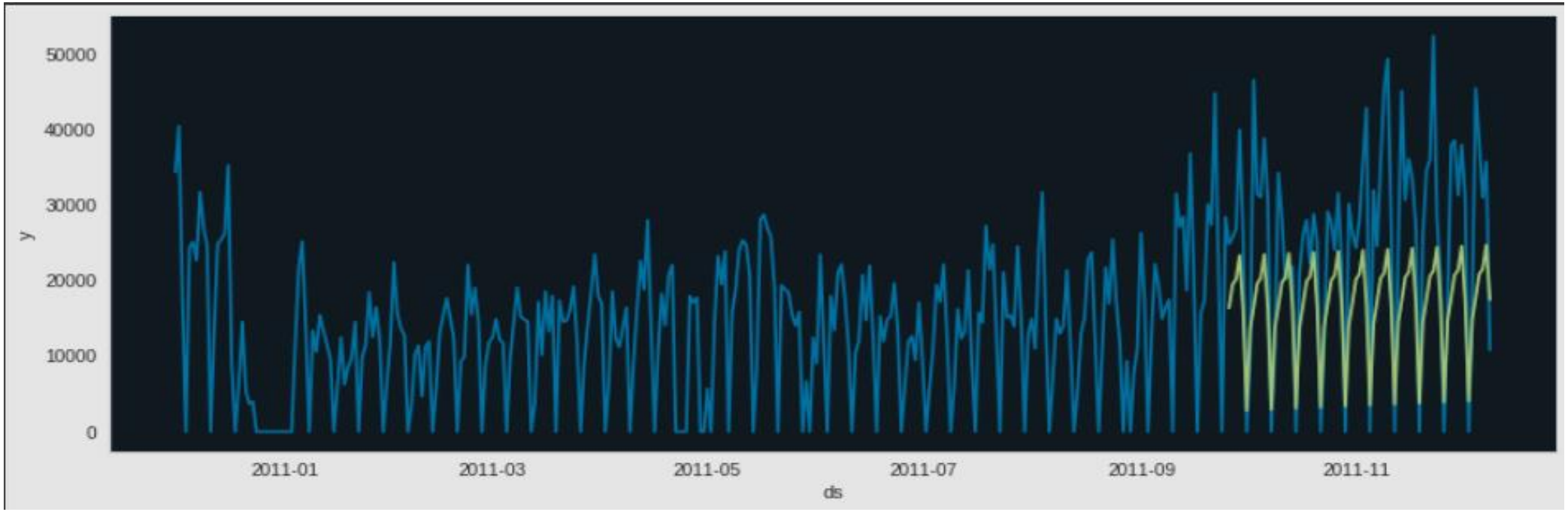
Revenue Forecasting using Prophet Algorithm

ACF and PACF plots are plotted and the series is clearly stationary.



Revenue Forecasting using Prophet Algorithm

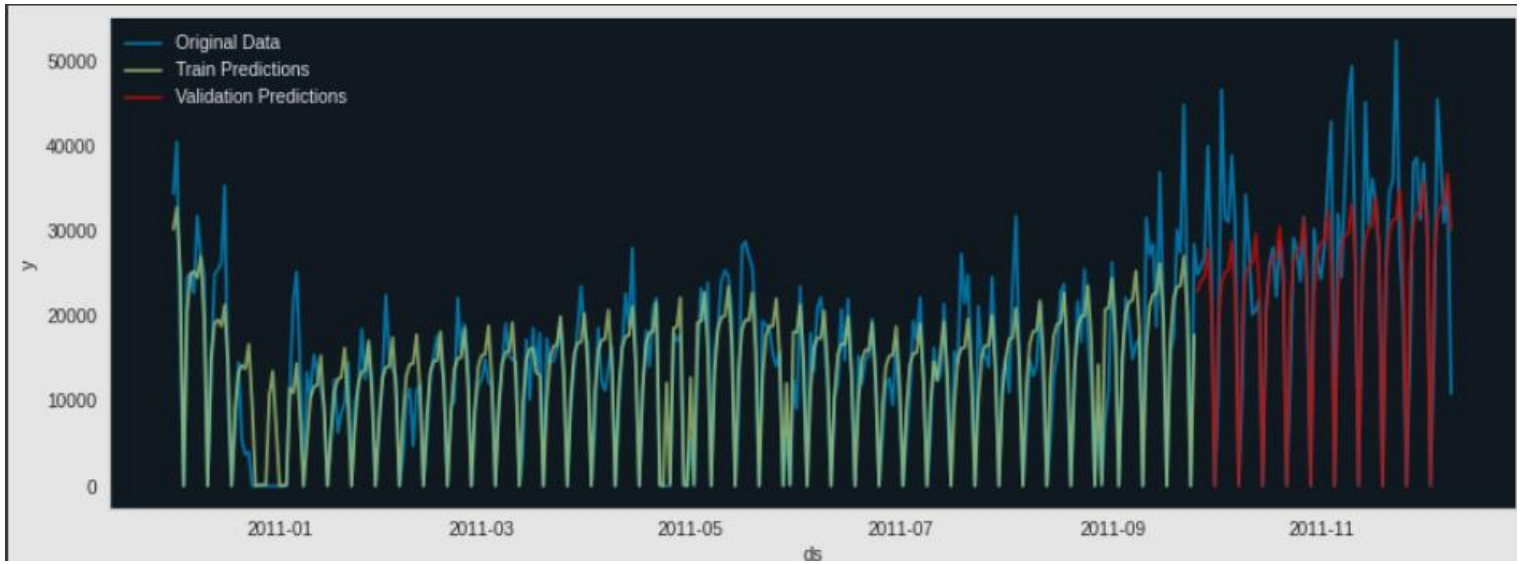
We have used daily sales data for model training. We observe 0 revenue values on Saturdays, and we consider it as an extra regressor.
The simple model is void of parameters like holidays and hence does not capture the peaks. For the training set, we have considered the first 80% of records of all days, and the remaining part is the validation set.
For this model, the forecasting accuracy metrics used are R2 score, mean squared error, root mean squared error and Pearson's correlation coefficient.



Revenue Forecasting using Prophet Algorithm

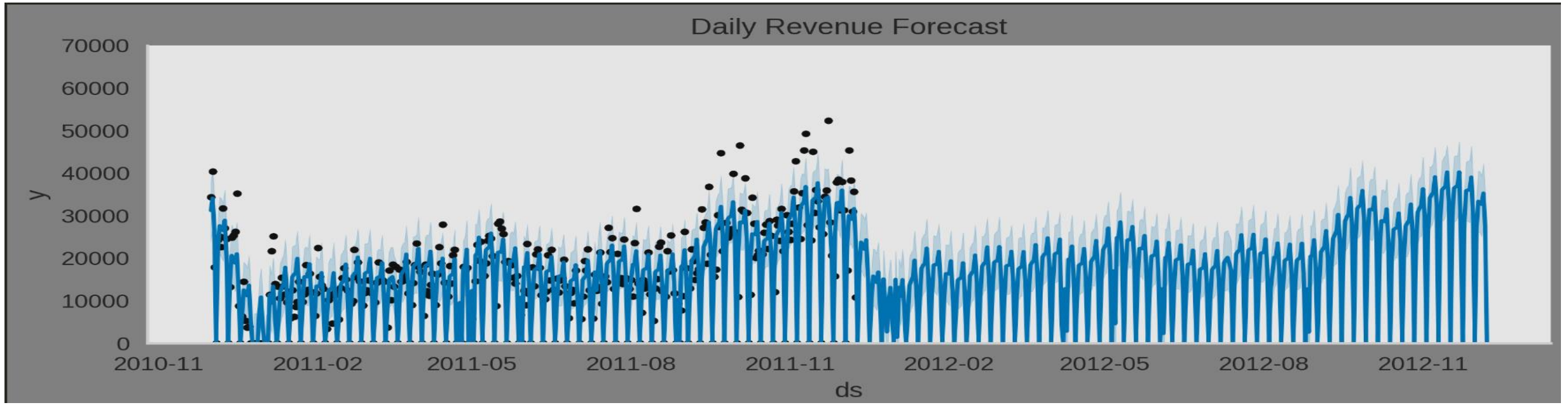
Holiday parameter is added and weekly seasonality is taken into consideration (set to True) while training and validating of the model, and daily and yearly seasonality is disabled or set to False.

Negative revenue predictions when encountered are set to 0.



Revenue Forecasting using Prophet Algorithm

Weekly seasonality is taken into consideration while training and validating of the model, and yearly seasonality is enabled used for forecasting the next year’s revenue.
Here is the results of this year's predictions and next year's forecast. Black dots represent actual revenue, and blue lines represent forecasts.
In the last months of the year, our predictions on the border of upper confidence interval, or sometimes outside of it.
If we look at next year's forecast, we can see minor peaks in the middle of the year, and also we have positive trend late of the year.



Revenue Forecasting using Prophet Algorithm

Performance Metrics we got for the model

```
For Training set:

R2:  0.7346422620130042
MSE:  21307932.71577869
RMSE:  4616.051637035562
Correlation:  0.8578247982880061

For Validation set:

R2:  0.6751995610737641
MSE:  56186566.63706101
RMSE:  7495.769916230154
Correlation:  0.8366898947852 6
```

Conclusion and Future work

1. In conclusion, after performing data cleaning and thorough Exploratory Data Analysis we observe that the customers from United Kingdom have the highest number of transactions and the highest revenue is generated from the same country due to the purchasing of expensive items.

2. We also observe the sales trend to peak especially at November and December as people buy gifts for their loved ones at that time of the year.

3. After plotting the following graphs:
   Country vs total price
   Country vs unit price
   Country vs Quantity
We can conclude that despite having the maximum number of customers using the retail store, the people of the United Kingdom tend to spend lesser money on gifts compared to people from other countries and tend to spend money on a lesser number of gifts.

The low unit cost and total spending of the people of the United Kingdom can be attributed to the fact that the United Kingdom has the most number of customers and sales(wrt the number of items purchased).

It will be easier for the company to maintain a large stock of products, it will also be easier for the company to maintain supply chains in and around the United Kingdom which will help the company reduce labor and other expenses which causes a reduction in the price of the items sold.

4. Some countries tend to have bigger average prices in carts like Australia, Japan, Sweden, and the Netherlands (which can either be due to people's choices or due to products themselves becoming more expensive).

There are differences in the best-selling items based on the country. The best-selling item overall is the white hanging heart t-light holder which is not the case when checked separately for each country.

Customers tend to buy items well in advance before Christmas probably fearing an absence of stock during the days Christmas comes near.

Conclusion and Future work

5. We also observe Retention rate increases significantly for the last months of the year. Probably, Thanksgiving, Black Friday, and Christmas cause it. There are lots of festive days at the end of the year.

6. From **Pareto Analysis**, We can see that 80% of the company's revenue comes from the top 30% of customers. Also, 90% of the company's revenue comes from the top 48% of customers. We also see that 80% of the company's revenue comes from the top 23% of products that have the most revenue. Also, 90% of the company's revenue comes from 36% of the products that have the most revenue.

Maybe, if the company reduces half its variety of items, revenue couldn't decrease significantly.

7. After **Customer Segmentation** we see that there are 15% of customers considered as Champions. These customers are responsible for a big share of your revenue so we can put a lot of effort into keeping improving their experience.

8. RFM Analysis separated all the customers into 10 informative categories based on their recency, frequency, and monetary.

9. K-means Clustering separated the customers into 4 clusters that have similarities based on their recency, frequency, and monetary. This time the categories are more flexible and not based on a grade.

From RFM and K-means, we learned that there are a few exceptional customers whom we would want to keep offering discounts, or an ambassador title. There is also a quarter of the customers that are not really active and don't spend a lot.

10. Using **Apriori Algorithm** we obtain the frequently purchased items and association rules like confidence and lift based on the minimum threshold support value.

11. Ultimately we perform forecasting via Prophet Model and we observe that:
In the last months of the year, our predictions are on the border of the upper confidence interval, or sometimes outside of it and if we look at next year's forecast, we observe minor peaks in the middle of the year, and the overall trend of the revenue vs days graph tends to be positive which is good for the retail store owner.

Thank You