TEAM_557

CONTENTS

OVERVIEW:

The Scenario:

XYZ company is the manufacturer of electrical fans (ceiling fans, table fans) and has pan-India
presence. The company sales to the final consumers through thousands of dealers spread across all major towns in India. The company has 4 regional warehouses which cater to 4 different regions (North, East, South, West) in India. The dealers place orders at the region-specific warehouse, which is fulfilled by the inventory available at the point of receiving
the order. The warehouse manager must forecast demand from dealers 1 month in advance.

The Problem:

The company in recent times has been facing serious challenges in terms of high inventory and
low fill rate across all warehouses.
A low fill rate indicates not enough stock or poorly organized inventory.
High stock levels result in a heavy cost burden. Not only does the actual inventory itself tie up capital, but it also requires more maintenance costs. A lot of personnel will be required, making salary expenses and incidental wage costs rapidly accumulate.

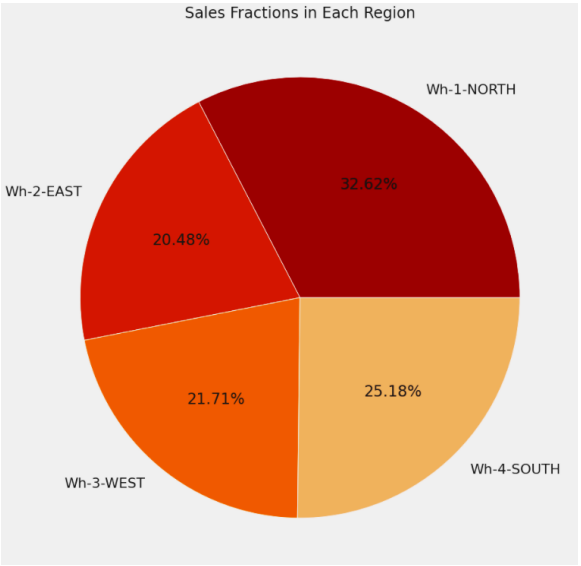
Task :

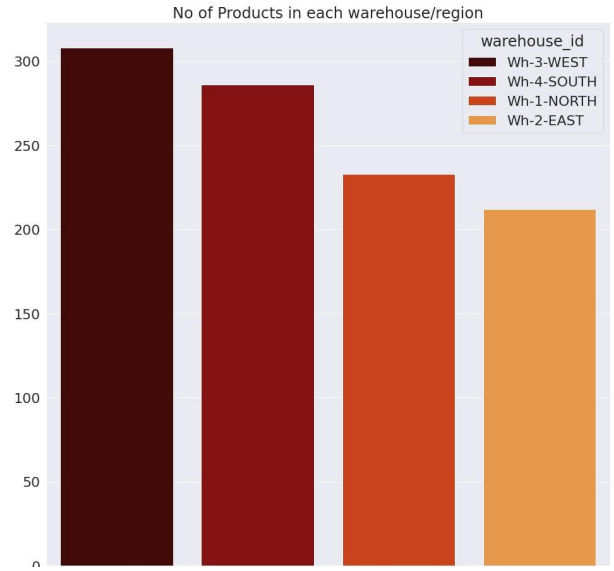
The company has identified poor forecasting accuracy as the primary challenge for poor inventory management. We need to help the company improve forecast accuracy by introducing advanced Time Series forecasting methods as well ML algorithms. The company also wants us to enhance forecast accuracy by incorporating the internal sales data.
Monthly Sales data by SKU by the warehouse is available between April 2018 and May 2021. The task is to predict sales by SKU by Warehouse for the month of June 2021.

Approach :

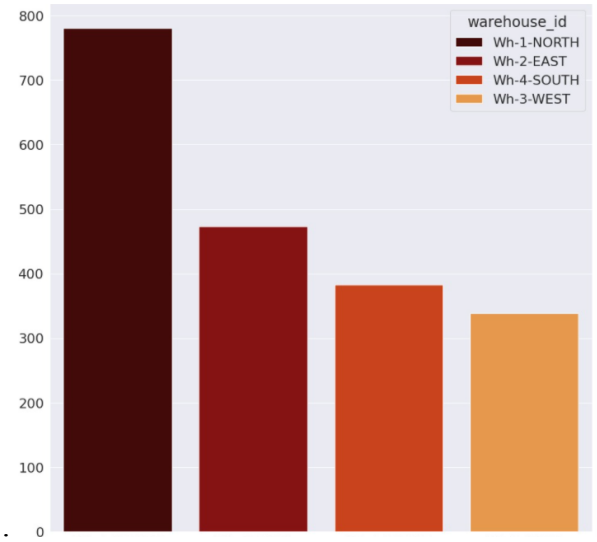
We have used data visualisation tools to generate insights and developed intelligent features to support our forecasting models. Identification of anomalies was done and treated suitably. The sales data for different regions and different products were significantly different from each other and hence made into separate clusters. Forecasting models from various categories like Time Series and Machine Learning were implemented. We evaluated them on the validation set for model selection. Since there was no one superior model, we used an ensemble approach, stacking results from the best performing models to produce robust predictions and prevent overfitting.

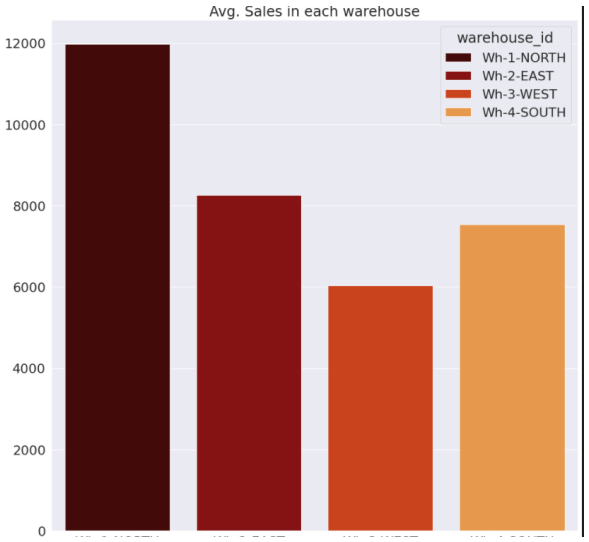
EXPLORATORY DATA ANALYSIS



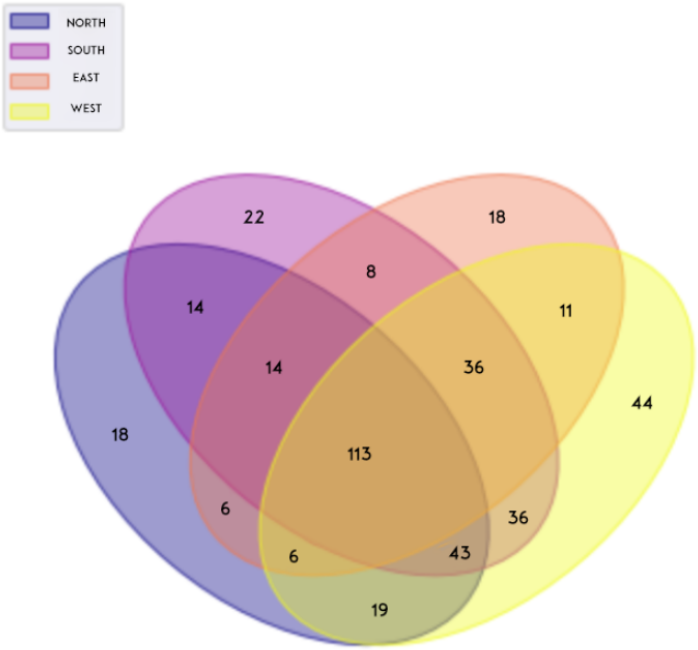


The bar plot on the left shows the number of products in each region/warehouse. We infer that the largest variety of products is present in the West warehouse. This pie chart shows the distribution of sales across the 4 warehouses.





The bar plot on the right shows the average sales data region wise for the time period April 2018 to May 2021. It is worth noting that values are in several thousand. On the left, the bar plot contrastingly shows a similar sales average but only for the covid affected months of April, May and June 2020. These values are much smaller compared to the right bar plot.

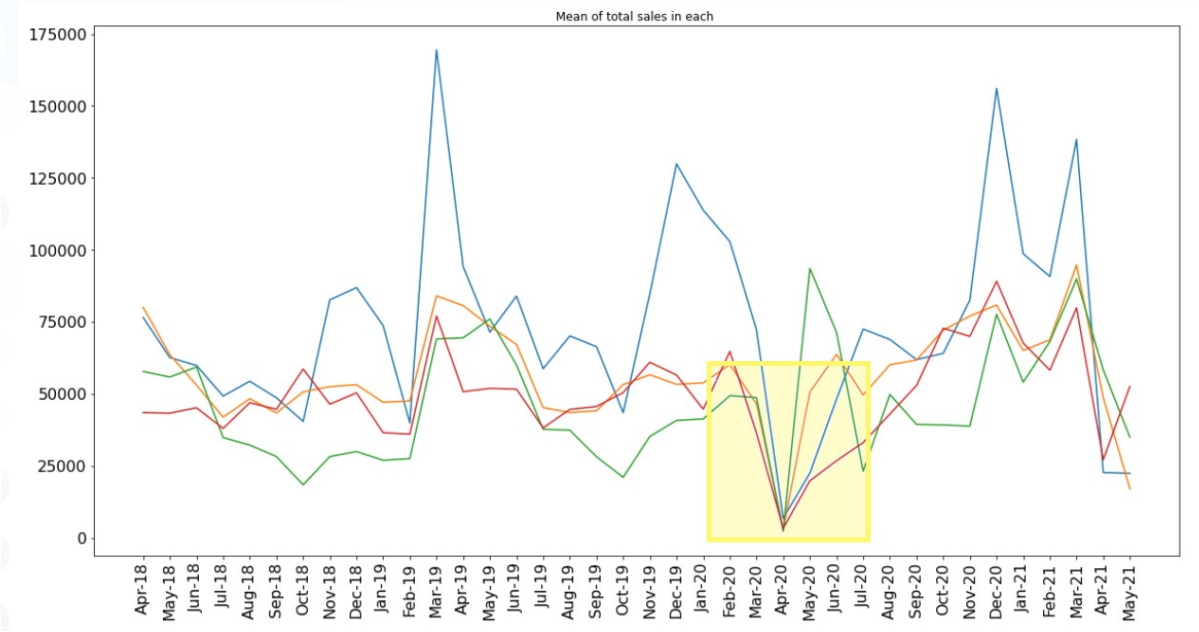


.

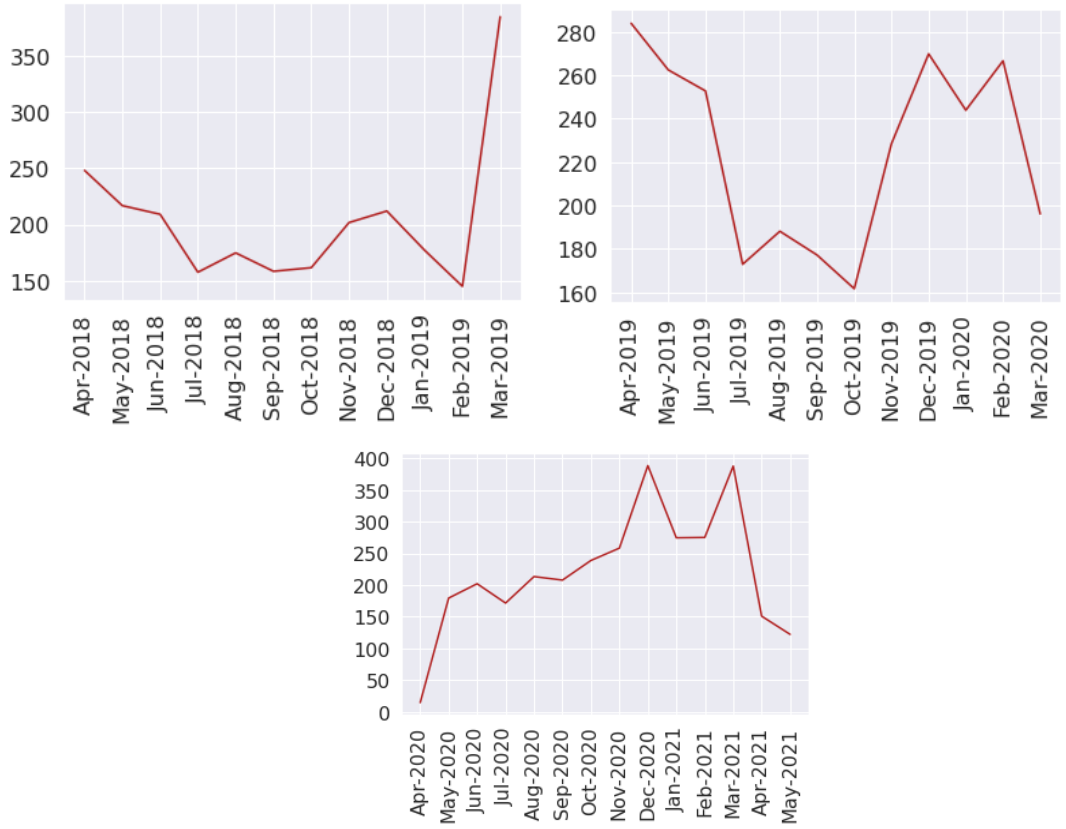
This Venn Diagram showcases the unique number of product Ids across the 4 regions. The intersections make it possible to observe the number of products being shared across regions.

TIME SERIES ANALYSIS

We here see the time series presented by the complete data as a whole i.e we represent 2018 - 2021 years worth of sales data, combining sales of every product :

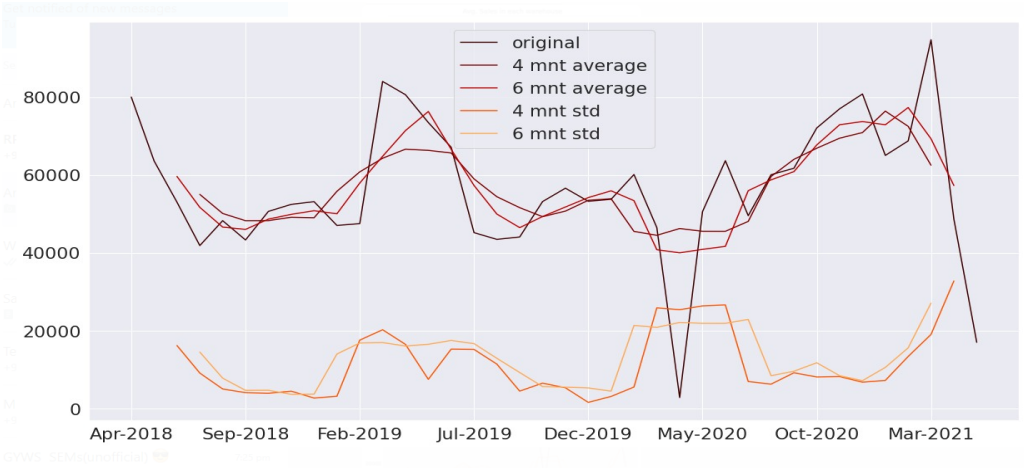


Then we see here similar time series of the given data in a year wise manner :

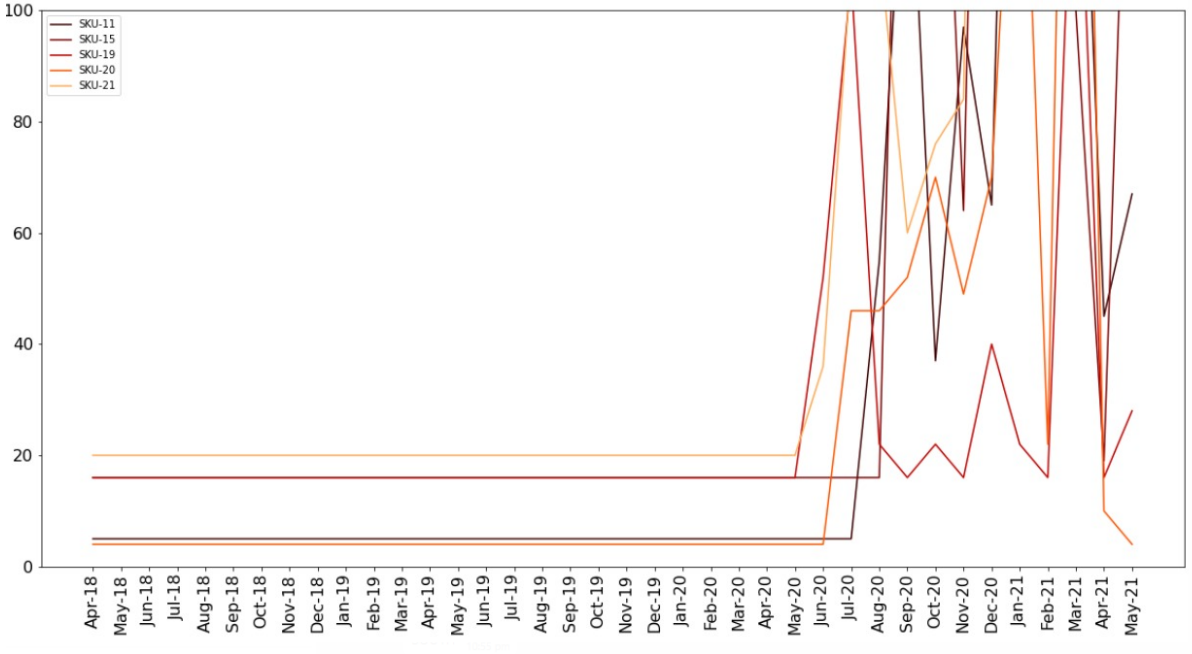


The sharp drop caused by COVID in the sales is easily noticeable in the plots especially the one showing data across all years.Other inferences drawn from the year wise plots are :

- In general - higher sales in December and March.
- There is a noticeable dip across the years around the month of October.



This plot illustrates that the 6 and 4 month moving averages follow similar trend and peak once around February 2019 and then towards the end of the data around March and May 2021.



Feature Scaling

Objective:

- To rescale the features such to reduce the computation time and improving the learning accuracy.
- To prevent the optimization from getting stuck in local optima.

Min-max Scaler

MinMaxScaler scales all the data features in the range *[0, 1]* or else in the range *[-1, 1]* if there are negative values in the dataset. For each value in a feature, MinMaxScaler subtracts the minimum value in the feature and then divides by the range. The range is the difference between the original maximum and original minimum. MinMaxScaler preserves the shape of the original distribution.

$$y = \frac{x_i - min(x)}{max(x) - min(x)}$$

Log Transform

Log transformation is a data transformation method in which it replaces each variable x with a log(x). When our original continuous data do not follow the bell curve, we can log transform this data to make it as "normal" as possible so that the statistical analysis results from this data become more valid. In other words, the log transformation reduces or removes the skewness of our original data.

$$y = log_b x_i$$

Standard Scaler

StandardScaler follows Standard Normal Distribution (SND). Therefore, it makes mean = 0 and scales the data to unit variance. Unit variance means dividing all the values by the standard deviation. Centering and scaling happen independently on each feature by computing the relevant statistics on the samples in the training set.

$$y = \frac{x_i - \mu}{\sigma}$$

$$where\ \mu\ is\ the\ mean\ and\ \sigma\ is\ the\ standard\ deviation$$

Note: After performing all the above feature scaling methods, we observed that the log transform gives the best result. Hence, we have used it.

CLUSTERING AND FORECASTING STRATEGY

Clustering

Objective:

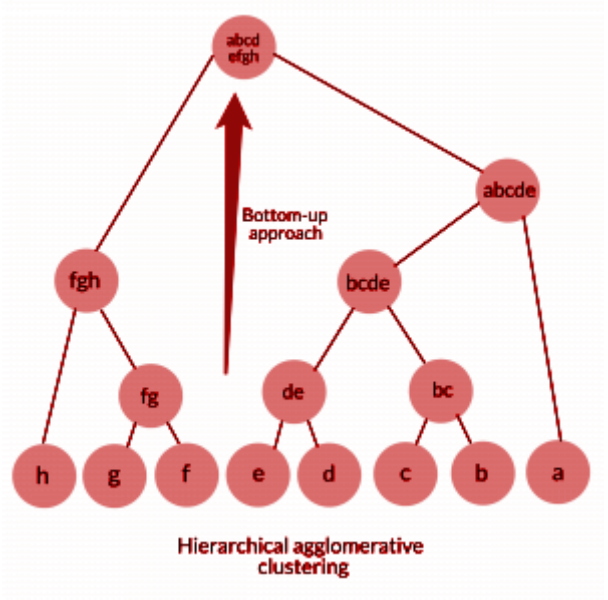
- To group together the products[sku-id] which are showing the same patterns/trends in their inventory sales.
- Finding the outliers [the product id showing extreme patterns]
- Fit the appropriate models on each cluster to predict the inventory sales for June-2021

How we used it for forecasting

- Using the different clustering techniques we grouped together the products showing similar time series trends.
- We got the optimal number of clusters using elbow methods and seeing the quality of each cluster.
- Afterward, we fitted the model on each cluster and got the prediction using those models for that particular cluster.

Agglomerative Hierarchical Clustering

Agglomerative clustering is used to group objects in clusters based on their similarity. The algorithm starts by treating each object as a singleton cluster. Next, pairs of clusters are successively merged until all clusters have been merged into one big cluster containing all objects. The result is a tree-based representation of the objects, named *dendrogram*.

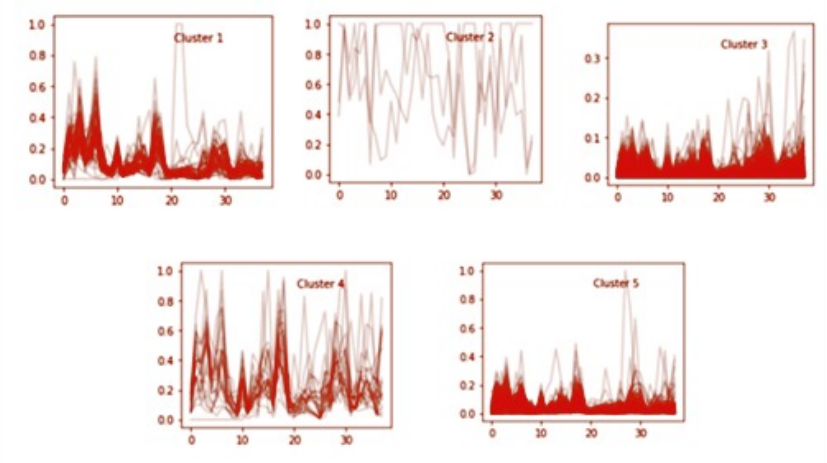


We create dendrograms to find the hierarchical relationship among the products [SKU-id] in the dataset to get an idea about the number of clusters that can be made.

Dynamic Time Warping

Dynamic Time Warping (DTW) is a technique to measure similarity between two temporal sequences that do not align exactly in time, speed, or length.
It minimizes the sum of squared DTW distance between the barycenter (cluster centroid) and the sequences in the cluster.

We have used DTW to cluster the trends of SKU's throughout the given months. The DTW Barycenter Averaging (DBA) algorithm finds this average trend by minimizing the sum of squared DTW distance between the barycenter and the time-series of prices for all the months for particular SKU.

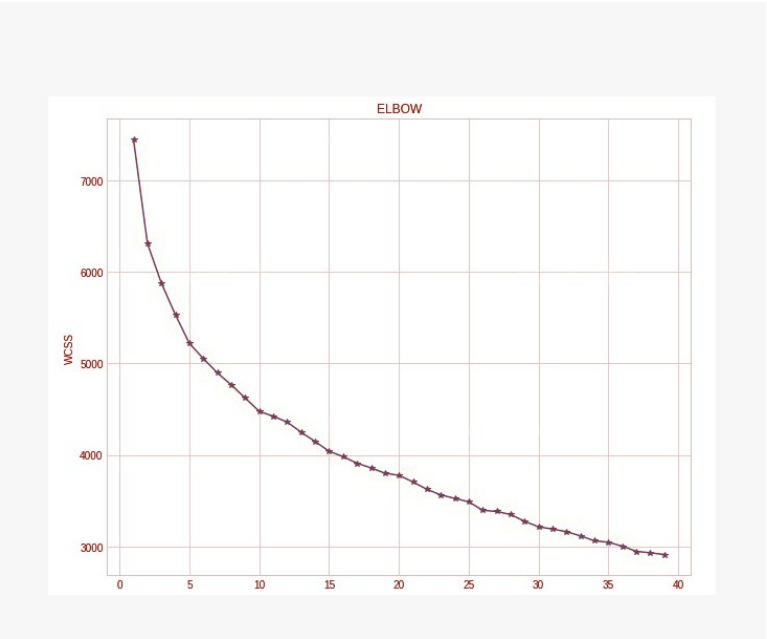


We can see products having similar trends gets clubbed into the same cluster

Note: The performance of the clusters using DTW clustering was the same as other method since the time axis was same for all the products.

Evaluation method for different clustering techniques:

Elbow method:



WCSS: is the sum of squared distance between each point and the centroid in a cluster.

In this method, we vary the number of clusters (K). For each value of K, we are calculating WCSS (Within-Cluster Sum of Square). Then we plotted plotted the WCSS with the K value, As the number of clusters increases, the WCSS value will start to decrease. The K value corresponding to the largest decrease in WCSS point is the optimal number of clusters.
Note: Since we trained one model for each cluster. So there was a tradeoff between the number of cluster and WCSS value.

Graphical View:

For each cluster, we took random k products and then compared the time series trend plot for each product w.r.t to the same cluster's products and with the product from different clusters.

Dendrogram:

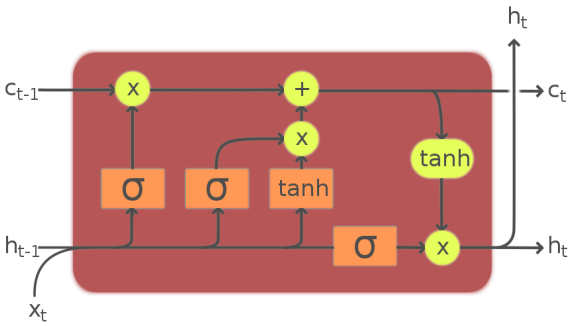
A dendrogram is a diagram that shows the hierarchical relationship between objects. The number of intersections that would be produced when a horizontal line cuts the dendrograms is the number of clusters. We got six clusters of the given data from the dendrograms after using the linkage 'complete'(which uses the maximum distance between all observations of the two sets).

Forecasting Models:

Long Short Term Memory:

Theory:

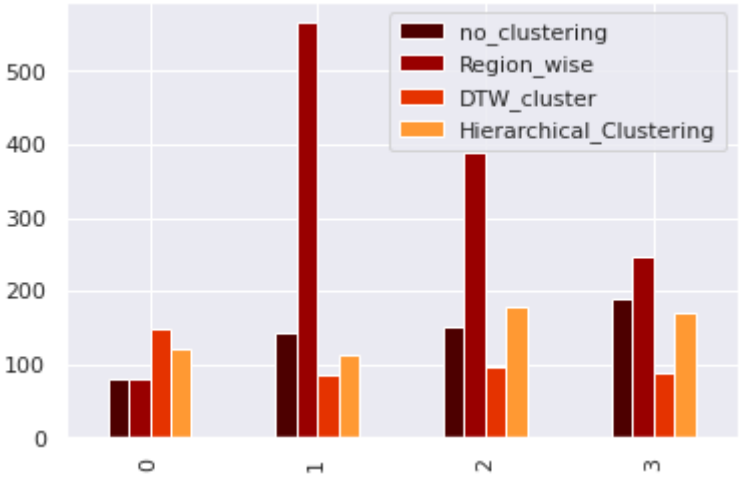
Long short-term memory (LSTM) is an artificial recurrent neural network (RNN) architecture. Unlike standard feedforward neural networks, LSTM has feedback connections.A standard LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell.



Why LSTM?

Having feedback connections in the neural network makes it viable for handling of time series data as each subsequent feature is informed by the previous one.Also, since there can be lags of unknown duration between important events in a time series, LSTM hold an advantage over traditional forecasting methods like Markov models or RNNs due to it relative insensitivity to gap lengths.

Result on time series data:



The best result on applying LSTM model was found to be on the DTW clusters of the dataset. MAPE value on May 2021=147%

Catboost:

Theory:

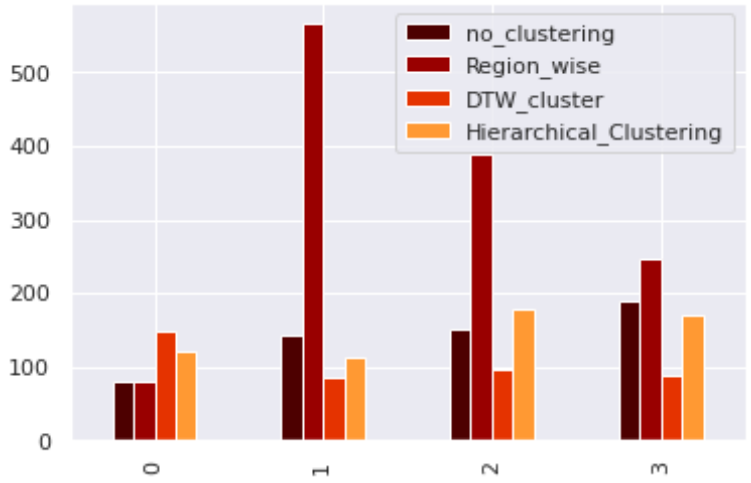
CatBoost is a recently open-sourced machine learning algorithm from Yandex. It can easily integrate with deep learning frameworks like Google's TensorFlow and Apple's Core ML. It can work with diverse data types to help solve a wide range of problems that businesses face today. To top it up, it provides best-in-class accuracy.
As discussed, the library works well with multiple Categories of data, such as audio, text, image including historical data.

"Boost" comes from gradient boosting machine learning algorithm as this library is based on gradient boosting library. Gradient boosting is a powerful machine learning algorithm that is widely applied to multiple types of business challenges like fraud detection, recommendation items, forecasting and it performs well also. It can also return very good result with relatively less data, unlike DL models that need to learn from a massive amount of data.

Why Catboost?

Catboost has a lot of advantages such as handling natively categorical and missing values, can handle a lot of features, scales well and can infer a lot of time series within the same model.As our dataset contains only time series data, Catboost proves to be a very nice model to fit onto the dataset.

Results:



The best result for the Catboost model was found to on region-wise dataset.

MAPE value for May 2021=80.55%

SARIMAX

Theory:

Seasonal Autoregressive Integrated Moving Average, SARIMA is an extension of ARIMA that explicitly supports univariate time series data with a seasonal component.The implementation is called SARIMAX instead of SARIMA because the "X" addition to the method name means that the implementation also supports exogenous variables,which are parallel time series variates that are not modelled directly via AR, I, or MA processes, but are made available as a weighted input to the model. For a SARIMAX model of order (1,0,1) and a seasonal order (2,0,1,5) the equation looks like:

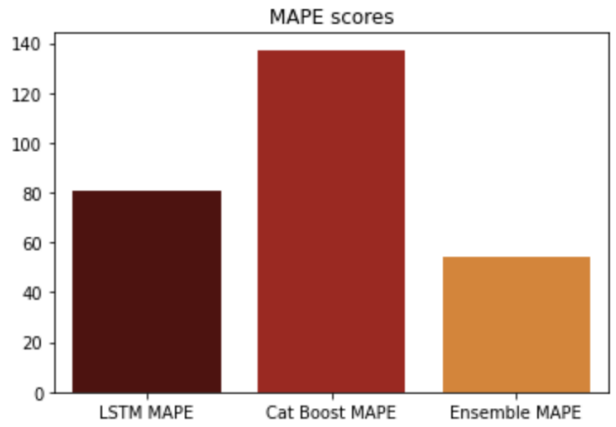
$$y_t = c + \varphi_1 y_{t-1} + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \Phi_1 (y_{t-5} + \varphi_1 y_{t-6}) + \Phi_2 (y_{t-10} + \varphi_1 y_{t-11}) + \Theta_1 (\varepsilon_{t-5} + \theta_1 \varepsilon_{t-6} + \theta_2 \varepsilon_{t-7}) + \varepsilon_t$$

Why Sarimax? :

In our data set, we noticed peaks occurring in the months of December and January every year for some products in the north warehouse; similar observations were made in other warehouses as well, which was indicative of seasonality in the dataset and since SARIMAX is used on data sets that have seasonal cycles as it takes into account the seasonality of the series to give more accurate predictions. It does this by taking in additional arguments p, d, and q for the seasonality aspect as well as an argument called "s" which is the periodicity of the data's seasonal cycle.

Results:

MODEL SELECTION:



Of the models we have used, CATBoost and LSTM turned out to be the ones giving consistently low MAPE values. Also, these 2 are from different classes of forecasting algorithms , capturing different types of relationship and hence, they have been used to form an ensemble.
After taking ensemble of these two models, the predicted values for June 2021 were coming out to be in decimal form.Now, obviously the actual sales must be integers.Instead of just rounding off the predicted values we have taken the floor function as we noticed that the predicted values were consistently a little higher than the true values.
We have observed that the predicted values are almost consistently higher than the actual values. Hence, with an aim of reducing the average MAPE score, for each SKU-ID, we have used that particular model which is giving lower MAPE.
We have intentionally done this because of two reasons:

1. To take care of the higher scale of predicted values compare to the actuals
2. Keeping in mind the fact that the ultimate goal is to make forecasts with lowest possible MAPE

CONCLUSION:

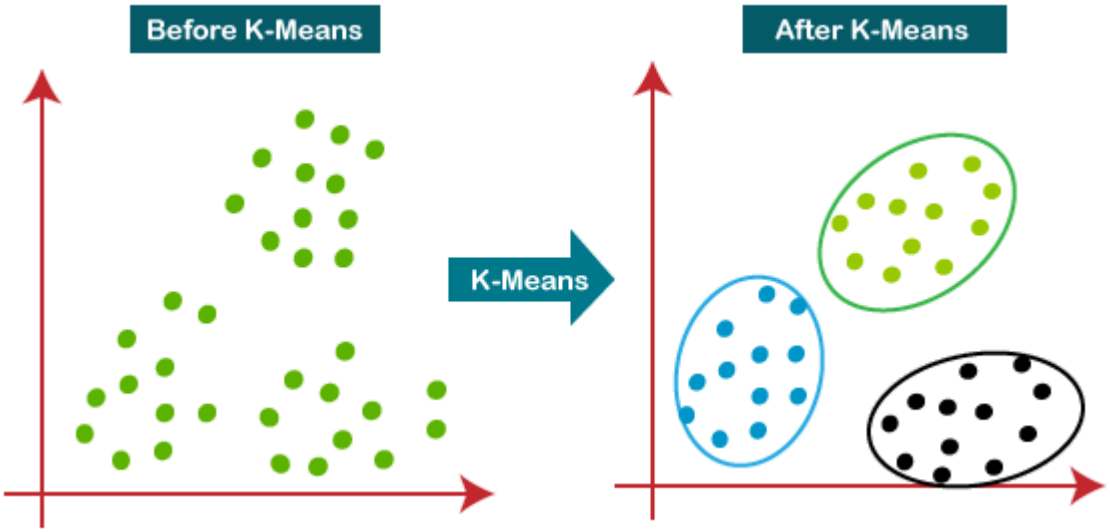
Our ensemble model returned desirable results, with the overall average MAPE reaching a lower figure
than those obtained in any of the individual models, as can be seen from the following graph.
In general, Ensemble Models overperform individual models because of their robustness and the combined power of individual models.
At the same time, the model results aren't extremely accurate which ensures that it isn't prone to overfitting on unseen data. Overall, the model result was satisfactory and we have used the same technique to make predictions on the test set.

ANNEXURE

K-Means
Covid effects on time series
Panel Regression
Stacked Data Approach

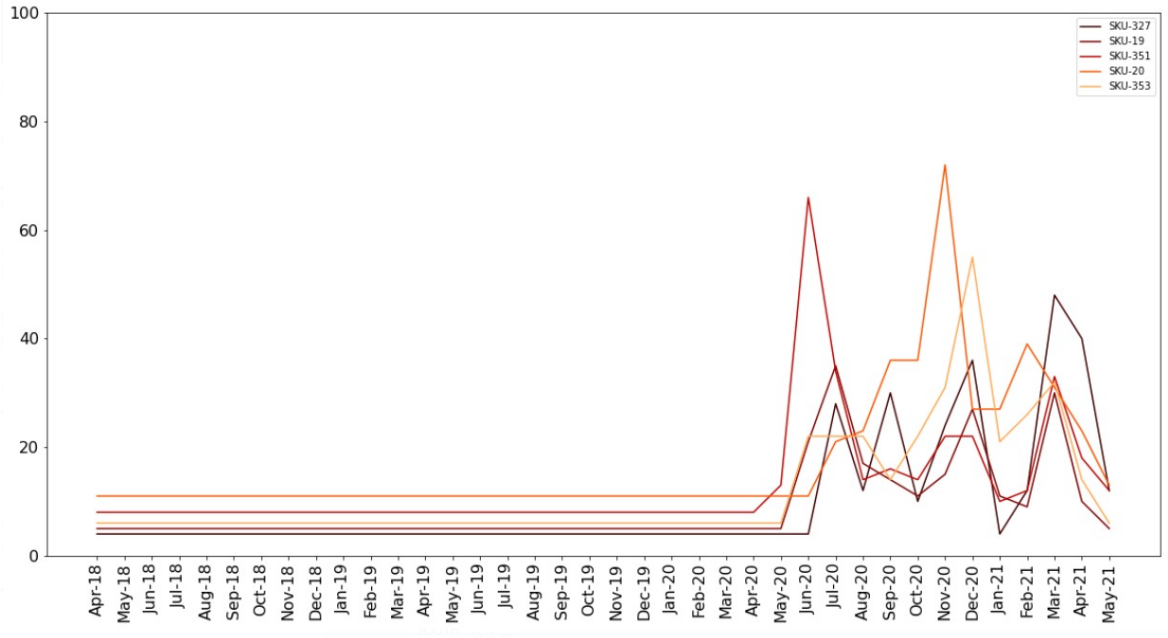
K-means Clustering

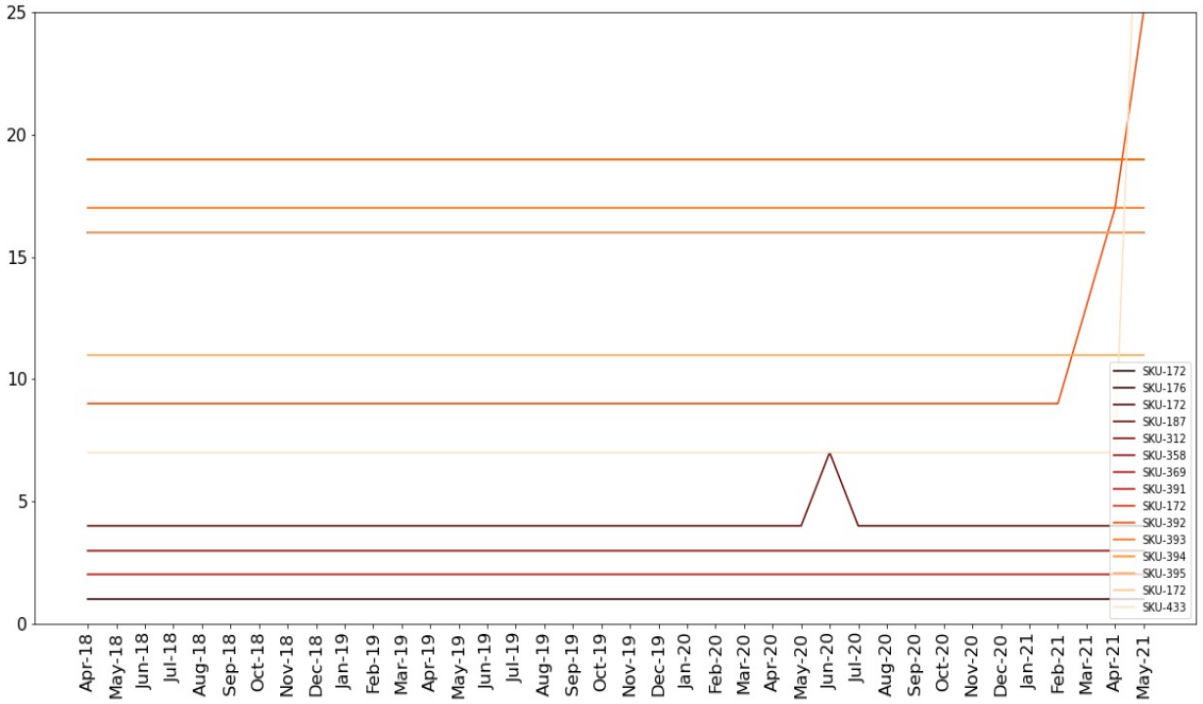
This algorithm computes centroids and repeats until the optimal centroid is found. It is presumptively known how many clusters there are. It is also known as the flat clustering algorithm. The number of clusters found from data by the method is denoted by the letter 'K' in K-means



COVID EFFECTS ON TIME SERIES

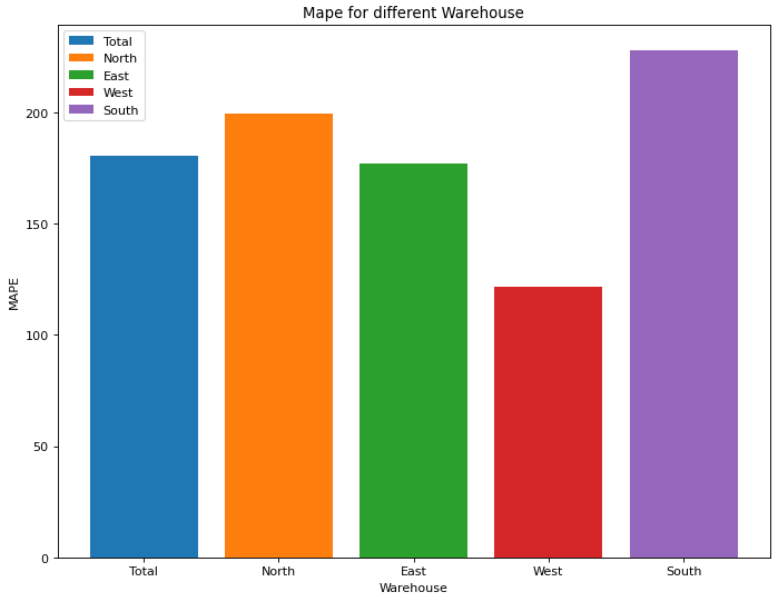
These two plots showcase some products (SKU Ids) which showed extremely **peculiar behavior**. These products have shown a constant sales behaviour in the data since the beginning, but then show a sharp increase post COVID. **We identified these very intelligently by comparing the pre covid mean sales and then computing the mean for same number of months post COVID**, then the products showing **greatest differences** between these values were plotted.**The first plot represents such products found in the North region, the second one corresponds to those in the South region.**





The above plot shows the product Ids which have shown constant sales behaviour throughout the time period we have the data for.

Mape for different Warehouses:



Panel Regression

Theory:

Panel regression is a modelling method adapted to panel data, also called longitudinal data or cross-sectional data. It is widely used in econometrics, where the behaviour of statistical units (i.e. panel units) is followed across time. Panel regression allows controlling both for panel unit effect and for time effect when estimating regression coefficients.
In our case we use the Random Effects Model.

Why Random Effects Model?

There are three types of Panel Regression Models: 1) Pooled OLS, 2) FE-model, 3) RE-model. Pooled OLS model ignores time and individual characteristics and focuses only on dependencies between the individuals. RE-model determines individual effects of unobserved, independent variables as random variables over time. They are able to “switch” between Ordinary Least Squares and Fixed Effects and hence, can focus on both, dependencies between and within individuals. RE-models determine which model to take according to the serial correlation of the error terms.

Result on time series data:

Different Types of Panel Regressions

1) PooledOLS: PooledOLS can be described as simple OLS (Ordinary Least Squared) model that is performed on panel data. It ignores time and individual characteristics and focuses only on dependencies between the individuums. However, simple OLS requires that there is no correlation between unobserved, independent variable(s) and the IVs (i.e. exogeneity).

2) FE-Model: The Fixed Effects regression model is used to estimate the effect of intrinsic characteristics of individuals in a panel data set.The FE-model determines individual effects of unobserved, independent variables as constant ("fix") over time. Within FE-models, the relationship between unobserved, independent variables and the IVs (i.e. endogeneity) can be existent

3) RE-Model: RE-model determines individual effects of unobserved, independent variables as random variables over time. They are able to "switch" between Ordinary Least Squares and Fixed Effects and hence, can focus on both, dependencies between and within individuals. RE-models determine which model to take according to the serial correlation of the error terms.

Choosing between PooledOLS and FE/RE: Basically, there are five assumptions for simple linear regression models that must be fulfilled. Two of them can help us in choosing between PooledOLS and FE/RE.These assumptions are (1) Linearity, (2) Exogeneity, (3a)

Homoskedasticity and (3b) Non-autocorrelation, (4) Independent variables are not Stochastic and (5) No Multicolinearity.If assumption (2) or (3) (or both) are violated, then FE or RE might be more suitable.

On performing the Breusch-Pagan-Test, p-values are less than 0.05 thus, heteroskedasticity is indicated. To confirm the same we perform Durbin Watson Test, which also gives us a value less than 0.05, thus we don't use PooledOLS model for the given dataset.

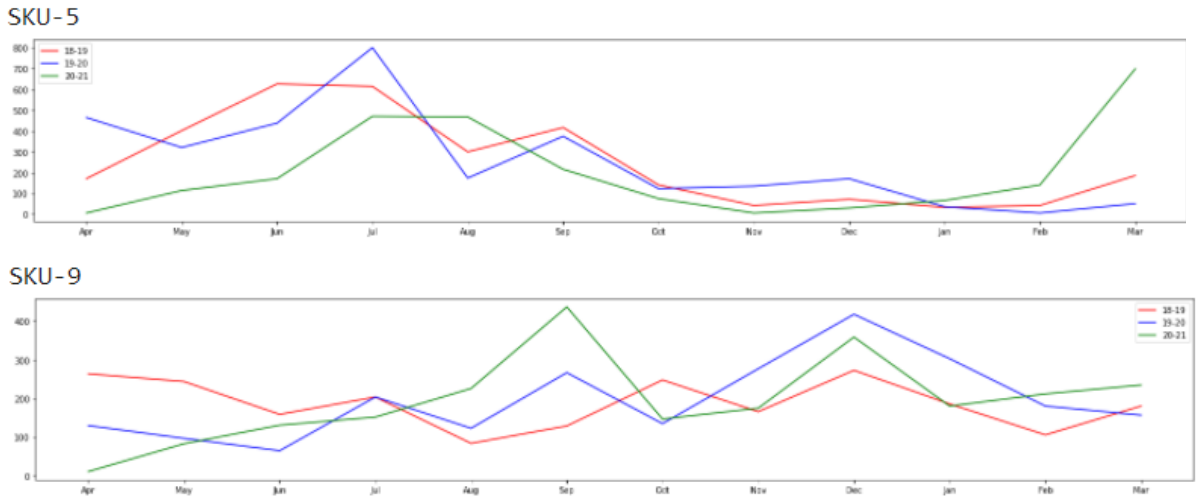
The Hausman-Test is a test of endogeneity. By running the Hausman-Test, the null hypothesis is that the covariance between IV(s) and alpha is zero. If this is the case, then RE is preferred over FE. If the null hypothesis is not true, we must go with the FE-model. In our case on performing, Hausman Test we find that RE model is more preferable over the FE model.

Stacked Data Approach in CAT Boost

1. Data Visualisations and Insights:

A. Yearly data visualizations:

- We tried visualizing the time series data year wise like (2018-2019,19-20,20-21)
- Some plots of specific SKU id's look like this:

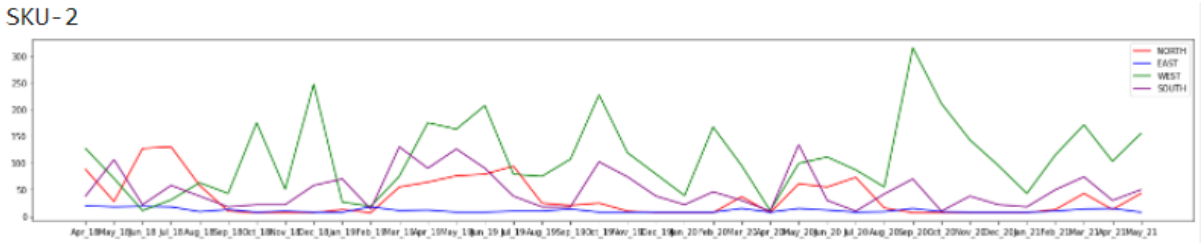


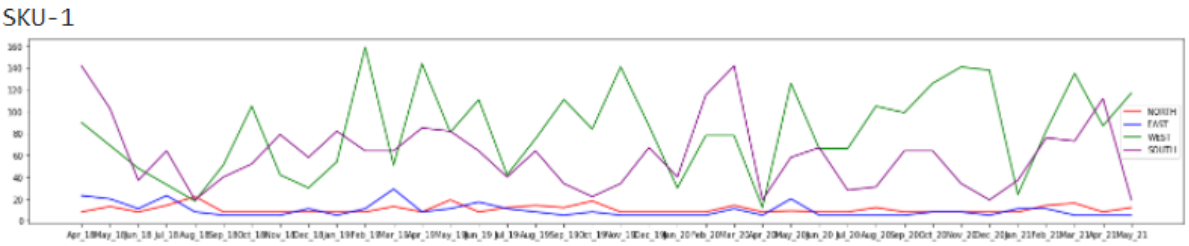
- Now we can clearly see that there is a similar trend in the pricing of SKU in most the SKU
- This can be a major insight we can get as there is a seasonality in the time series pricing

B. SKU in every region:

- We also plotted how a single SKU pricing is working in different regions(North,East,West,South)

- The plots looks like :





- Most SKU in a different region have a similar dip in the region of around Apr-20
- We can infer that the extreme dip might have occurred due to the COVID_19 lockdown

2. Data Preprocessing and Exploiting yearly seasonality

a. Managing the April-20 (COVID) dip

- We managed the covid dip by replacing the covid dip (Apr_20, May_20, Jun_20) with the Average of the (Apr, May, Jun) of 2018 and 2019
- This handles the covid dip in our data, Now if we see the data we have Data from Apr_18 to May_21

b. Stacking data approach

- To leverage the fact that the SKU pricing is performing similarly each year. We in fact divided the whole 3 years of data into parts and stacked the first 2 year's data into a single training file and we considered last year's data as Test data
- So our data is from July_18, August_18……..May_21, our 3 parts of the data will be 1st part = (July_18 to June_19)
  2nd part = (July_19 to June_20)
  3rd part = (July_20 to May_21)
- Now we can clearly see that we have the June values of 2 years i.e. first and the second part, Now stacked up the 1st and 2nd parts vertically (vstack) and that makes our train data double the actual no of instances given in the data
- Now the later part i.e 3rd part becomes our test data from july_20 to may_21
- So as a whole we are actually trying to convert the given time series model into a supervised learning model to leverage the fact that the yearly pricing of each SKU is almost similar
- So when we train our model we essentially take the X_train as the data of (July to May) of the stacked data and our y_train will be the June column of the stacked data

- Now the 3rd part which consists of the data from july_20 to may_21 essentially becomes our X_test and we try to predict the June_21
- Thus converting the whole time series into a supervised learning algorithm technique can give us more freedom and assurance of our model performance and the training phase

3. Trying out different models and hyperparameter tuning

a. Different models:

- We implemented various models like (XGBoost,LightGBM,SVR,CATBoost,DTRegresso)
- We validated our models by first predicting May 2021 and validating it with the may_21 given in the data
- MAPE metric is considered as our performance metric thus giving us total freedom on selecting the models which performed well
- CAT Boost is identified as a better predictor compared to other models so we went on with CAT Boost

b. Outlier classification and Re-predicting

- We used CAT Boost to find predictions and we have used a threshold to find the outliers int our data set when are predicting for may 21 so we considered the same as our outliers in our June predictions and re-predicted those specific SKU_id's for a better prediction of our final june_21
- Thus we achieved good results on our map values in may_21 predictions and achieved an ensemble MAPE of less that 60% when using both Stacked approach + LSTM predictions