**Demand Forecasting and Procurement**

**Motivation**

Nowadays accurate sales forecasting is essential for a business house to enable it to produce the required quantity at the right time. Production industry often struggles with the problem of under-utilization of capacity due to shortage of raw materials. Forecasting enables businesses to plan and make informed decisions about future operations, marketing, and resource allocation. Accurate forecasting can help businesses anticipate future demand, identify potential problems or opportunities, and adjust their strategies accordingly. It helps in overall business planning, budgeting, and risk management. Forecasting allows companies to efficiently allocate resources for future growth and manage their cash flow. It can also help businesses optimize their inventory levels, production schedules, and staffing requirements. Understanding future sales helps businesses enhance customer relationship management. By anticipating customer needs and preferences, companies can tailor their products and services to meet expectations, fostering customer loyalty.

**Objectives**

* To forecast the sales quantity for the next period of time so that the organization can have inventory for the upcoming demands in advance.
* The organization may lose customers due to unavailability of products (facing a stock-out situation). The forecasted sales quantity will help to maintain sufficient inventory which will be enough to meet future customers' demands.
* Sudden demand from customers has to be fulfilled by importing goods via airway which incurs huge costs. The forecasted values will give time for goods which can be imported via sea which drops cost to 20 %.
* To establish sales forecasts for setting realistic and achievable goals for the organization. It provides a foundation for strategic planning and guides the development of business objectives.
* To have budgeting and financial planning for revenue projections, enabling organizations to create accurate budgets, allocate funds appropriately, and make sound financial decisions.

**Data Description**

Data consists of order details from April-2017 to September-2023 including sales quantity. Data contains 293 features in total with almost 121000 of rows. The data for each financial year is merged and a master data is created for further processing.

**Modeling Techniques**

For forecasting, different statistics-based models like AUTO-ARIMA, machine learning based models like Random Forest, XG Boost and Deep Learning based models like Long Short Term Memory Networks (LSTM), Gated Recurrent Unit (GRU) Model etc. are used. I addition to this modeling also uses transformers-based models like TFT and NBEATS models. As forecasting is to be done for each “Item Code” separately, single forecasting models is not accurate enough for all Item Code. So, every time a different model is fitting closely to different Item Codes.

**Data Analysis and Data Preparation**

**Data cleaning**

* All the separate data files are merged into one called a master data file.
* Data type for all the features is checked to confirm whether all features are imported with correct data type or not. If not, the data type for respective important feature is converted to required format.
* Features with null values are checked to determine which feature needs missing value imputation in further analysis.
* Features with all null values are detected from dataset and dropped.
* Features having only one unique value are diagnosed and dropped from data.
* Features with one class dominating over 95% of total data available, are detected and eliminated from dataset.
* Few of the features are having most of the values are null. So, dropping out those features which have null value greater than 70%.
* A few features are found to be redundant for further analysis. Also, a few features were duplicated with respect to other columns. So, dropped one of them from all such pairs.
* Detected and Converted date features in datetime.
* For imputation, categorical and numerical data type features are separated from the rest of the features. For categorical features, pairs of feature names and codes are identified. Then unique pair of class and its code is listed out. Feature names are imputed by extracting its name from code feature and vice versa.
* For the rest of the categorical features, missing values are imputed with mode.

**Data Analysis**

* Historical sales pattern is one of the important things for AI model building. So, it is necessary to determine whether the Item Codes are following any pattern in the past or not.
* Depending upon such sales pattern or frequency of demand, all the Items Codes are classified. The classification made is as follows,
* Runner: - Items Codes sold almost every month.
* Repeater: - Items Codes sold at least once in every quarter.
* Stranger: - Items Codes sold at least once every half year.
* Yearly: - Items Codes sold at least once in every year.
* Rare: - Items Codes sold for less than 6 times in whole period.

***NOTE:*** *The focus is on the runner item data as it is sold almost every month*.

The overall data comprises of sales values of 77 months (about 6 years) i.e., from April-2017 to September-2023. In the EDA part it is observed that for some runner items it shows the decreasing trend in year 2020 (It is due to COVID).

So, keeping the above things in mind, it is decided to work on different experiments for forecasting. These experiments are Explained below.

**Experiments for Model Building**

* **Experiment 1:** Modal will be built by considering all the data. Data till December-2022 is used for training purpose and rest of the data as validation for trained model.
* **Experiment 2**: To analyze the COVID effect, instead of using all the available data, data up to December 2020 is considered for modeling out of which data up to December-2019 is used for training and rest of the data is for testing purpose.
* **Experiment 3:** Omit the original data available for covid period and replace it with forecasted data from second experiment. Then proceed as per the Experiment 1.
* **Experiment 4:** Omit the data for COVID period and connect December 2019 directly to January 2021. Build a model on modified data.

**Flow for Experimentation**

Extract the Data

Prepare Data for Model Building

Split the Data Sequentially

Build the Model

Check the result

**Experiment 1: Normal Analysis**

(Model building is performed on the original data)

**Details of Data used**

* Data comprises of sales values of 77 months (about 6.5 years) i.e., from April-2017 to September-2023
* Training data uses first 68 months (about 5.5 years) data i.e., from April-2017 to December-2022.
* Testing / Validation data uses the rest of the month’s sales values i.e., from January-2023 to September-2023.

**Experiment 2: COVID Data Analysis**

(Model building is performed on the data up to COVID)

**Details of Data used**

* The overall data considered for analysis is only from April-2017 to December-2020.
* COVID period is considered from January 2020 to December 2020.
* Training data is considered from April-2017 to December-2019 while testing data is from January 2020 to December 2020.

**Experiment 3: Imputed Data Analysis**

(Model building is performed on the imputed data)

**Details of Data Used**

* Here, the actual data for COVID year (i.e., calendar year 2020) is replaced by forecasted values from the second experiment. The intention behind such imputation is to nullify the effect of COVID and use predicted values from AI Models for further analysis.
* The imputed value varies every time when AI model Technique used for model building changes.
* Overall data comprises of sales values of 77 months (about 6.5 years) i.e., from April-2017 to September-2023
* Training data uses first 68 months (about 5.5 years) data i.e., from April-2017 to December-2022 which includes the imputed data for COVID period.
* Testing / Validation data uses the rest of the month’s sales values i.e., from January-2023 to September-2023.

**Experiment 4: COVID Data Drop Analysis**

(Model building is performed on the data excluding COVID period)

**Details of Data Used**

* As COVID period is considered from January 2020 to December 2020, data for same period is dropped from original data for further analysis.
* Training data uses first 50 months (about 4 years) data i.e., from April-2017 to December-2022.
* Testing / Validation data uses rest of the 12 months sales values i.e., from January-2023 to September-2023.

All four experiments are performed using the above-mentioned forecasting modeling technique and their results are summarized in a table for better understanding. It is observed that different experiments with different modeling techniques are fitting best to actual data for a particular Item Code.

Deciding the best model or top best fitting models is crucial task. So, the forecasted values obtained from all models need to be processed. The post-processing methods used are described as below,

**Post-Processing**

After analyzing results from different models, it is observed that single model is not capable of predicting whole deviation from the validation data. So, it is better to combine all the models and see the obtained results. Post processing involves analyzing the data and combining more than one model to have better results.

**Aim for post-processing**

To select top performing models for item code and forecasting the further values using those selected models.

**Approaches for Post Processing**

Post Processing is performed using five different approaches which are mentioned below.

* **First Approach**

Results from different models for validation data i.e., from Janunary-2023 to September-2023 are stored in separate data-frame with columns as different model name and rows as forecast from model for that month. Mean for all the forecasted values from all the model for that month is calculated. Similarly for all the months mean is calculated, and those means are tested against actual data. Along with mean maximum and minimum values also calculated and validated against actual values.

* **Second Approach**

In this approach, top performing models are chosen on the basis of RMSE and MAE values. Models that have lower RMSE’s are chosen as top performing model and taking the mean of forecasted values of those models and test against the actual values from the data.

* **Third Approach**

Data-Frame containing forecasted values along with actual values for that month is converted into supervised learning problem with target columns as actual values and features as different models. ML based and Ensemble based models are fitted on the data and collected the importance for each feature and weighted mean for each forecast value using above weights are calculated and tested against actual values.

* **Fourth Approach**

Created separate data frame which include features as different models and rows as RMSE values quarter-wise i.e., from January-2023 to March-2023 as first quarter RMSE values, from April-2023 to June-2023 as second quarter RMSE value and from July-2023 to September- 2023 as third quarter RMSE value. Line Graph for all the quarter wise RMSE is plotted, and the top three models are selected from the graph with downward trend of RMSE.

* **Fifth- Approach**

From validated values data-Frame, slope for each model (feature) is calculated using differencing and that is compared with the slope of actual line. Difference Between actual line slope and slope of model line is calculated and plotted line plot for that difference to choose best performing model with lower deviation.

* **Sixth-Approach (Ensemble Model)**

The forecast values of each model aren’t fitting closely with the actual values. it is necessary to process the forecast values and then give new modified values as new sales forecast. “Cosine Similarity” is technique used for determining which model is close to actual data. Forecast values of each model are used for calculating cosine similarity with actual values. Then models are arranged in descending order depending upon their cosine similarity. Top 3 models with highest cosine similarity are selected for further processing. The top models are assigned with weights as per their cosine similarity values. The forecasted value of each model for specific month is multiplied with respective weight and summation all weighted forecast values are taken as final value of forecast. This approach worked properly and results of this model are fitting closely to actual data.

**Results Analysis**

Analysis uses results of nine models and one ensemble model for each item code. It is necessary to shortlist result of a model to propose its values as the final forecast values. RMSE, MAE and R-Square are three metrics calculated for each model. Initially, models are sorted based on least RMSE values for each item code. Sorting with RMSE values didn’t work properly as less than 50% of item codes yielded the appropriate results. Thus, models sorted by based on MAE in ascending order gives most appropriate results for all the runner and repeater item codes. The logic build in Jupyter Notebook is productionized using Rubiscape and dashboard is created in Rubisight.

**Results Summary**

|  |  |
| --- | --- |
| **Class** | **Count of Item Codes** |
| Zero FC | 9 |
| > -100% | 4 |
| `-25% to -100% | 93 |
| ±25% | 44 |
| 25% to 100% | 46 |
| 100% to 300% | 57 |
| 300% to 1000% | 48 |
| 1000%+ | 11 |

It is observed from the dashboard analysis that accuracy of the forecasting model has to be improved based on the closeness between actual sales data. The variance between actual sales and forecasted sales value must be within range of 25%. But with current analysis, only 44 item codes out of 312 are falling in the acceptance range. It is necessary to modified the approach for achieving better results.

**Modified Approach**

**Preprocessing Steps**

* Actual data contained some spikes in sales which is statistically outlier for model. It is necessary to makes changes in such data before forwarding it to the model building.
* Firstly, values which are above upper whisker level are taken out of data and again upper whisker level for new data is calculated. The values which lie above this new upper whisker limit are brought back to this upper whisker limit. To execute such task, a function is written and is called while building the model.
* From the research, it is observed that adding new feature by using existing features can improve the accuracy of results. Techniques like moving average, lead, lag, exponential moving average and recency can be used to generate features.
* Using all these new features, model is built for 20 sample item codes to check whether this technique is working or not. It is observed that model is getting overfitted due to considering all the new features generated.
* Thus, feature selection has to be performed for data of each item code to eliminate highly inter-corelated features. Features are selected by applying machine learning algorithm for finding out feature importance. Selecting only those features which are adding maximum amount of information to the data.
* It is observed that some of the item codes are having zero sales values for past six consecutive months. Model yields zero forecast value for such item codes. Instead of including such item codes for model building, such item codes are identified and eliminated from further analysis.
* Item codes like Scrap, Insurance, P & F and Freight are dropped from the analysis as client didn’t demanded forecast for such item codes.

**Model Building**

* Total 77 months of data is available for model building. Each model is initially trained on data for 68 months and next 9 months data is used for validation purpose. The forecast is generated from such trained model which is termed as old forecast for that model.
* Same model is fitted on all the data, that is, 77 months data and values are forecasted based on it. These forecasted values are termed as new forecast for that model. We have two set of forecast values for each model of each item code.
* The models used are hyper tuned and code is optimized to reduce the computational time.
* All the Runner and Repeater Item codes are now passed through this first set of Model configurations.
* Sales quantity data for month of October is now used to find out the variance between forecasted values and Actual October sales. Variance with both old forecast and new forecast is calculated against actual October month forecast.
* It is found that new forecast values are matching nicely with the actual data compared with old forecast values. So, new forecast values are used as a final forecast for the respective item code.
* It is found that almost 30 to 35% od item codes are having variance less than 25%. Another set of hyper tuned models is prepared for such item codes which are having variance greater than 25%.
* Data for such item codes is again passed through second set of hyper tuned models. The results obtained from second set of configurations is merged with results first set of configurations. Variance for results of second set is calculated and model having least variance is selected as a best fit model for that item code.
* Item codes are then classified according to their variance value. The summary of results obtained from modified analysis is enlisted in the table given below.

**Results Summary**

|  |  |  |
| --- | --- | --- |
| **Variance Class** | **Number of Runner Item Codes** | **Number of Repeater Item Codes** |
| 0 to 25 | 23 | 110 |
| 25 to 50 | 6 | 26 |
| 50 to 75 | 3 | 11 |
| 75 to 100 | 6 | 85 |
| 100 to 300 | 3 | 15 |
| 300 to 500 | 1 | 4 |
| 500 to 1000 | 0 | 3 |
| Above 1000 | 0 | 1 |

From table, it can be seen that almost 133 item codes within acceptable range. Also 50% of total item codes are having variance between 25% to 100% which is quite good as compared to previous methodology. Less than 10 % of item codes are yielded results which cannot be accepted as they are having huge amount of variance. Following figures shows distribution of bar chart for runner and repeater item codes separately. Items codes bounded by yellow lines may be considered as within acceptable range. Out of 42 runner item codes, 23 falling under variance less than 25% and out of 255 item codes, 110 item codes are under acceptable range.

An interesting observation made while analysis is that, for some of the item codes, the absolute difference between actual and forecasted sales quantity is small but when percentage variance for the same is calculated, it comes out to be very large or beyond the acceptable range. The example given below shows the same thing.

**For example:**

forecast sales = 8

actual sales = 5

variance = (8-5)/8

= 37.5 %

Thus, absolute difference for each item code is calculated and item codes are classified accordingly. The item codes which are having absolute difference less than 5 can be considered within acceptable range. The count below doesn’t include item codes which are already with the range of 25% variance. The table below enlist the count of runner and repeater item codes as per lying in the class of difference.

|  |  |  |
| --- | --- | --- |
| **Absolute Difference Class** | **Number of Runner Item Codes** | **Number of Repeater Item Codes** |
| 0 to 3 | 0 | 16 |
| 3 to 5 | 0 | 10 |
| 5 to 10 | 1 | 33 |
| Above 10 | 18 | 86 |

Repeater item codes are having almost 26 item codes which are having deviation of less than or equal to five from actual value even if variance value is greater than 25%.