A Report

On

DEALER INVENTORY OPTIMIZATION

BY

Anirudh 2017A4PS0919G

Akanksha Kumar 2017B4A30803P

Saksham Gupta 2017A7PS0218P

At

MARUTI SUZUKI INDIA LIMITED

Gurgaon, India

A Practice School I station of

JULY, 2019

A Report

On

DEALER INVENTORY OPTIMIZATION

BY

Anirudh 2017A4PS0919G

Akanksha Kumar 2017B4A30803P

Saksham Gupta 2017A7PS0218P

Submitted in partial fulfilment of the Practice School – I course

BITS F221

At

MARUTI SUZUKI INDIA LIMITED

Gurgaon, India

A Practice School I station of

JULY 2019

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE

PILANI (RAJASTHAN)

Practice School Division

Station:	MARUTI SUZUKI INDIA LIMITED
Centre:	GURGAON
Duration:	21 MAY, 2019 TO 13 JULY, 2019
Date of Start:	21 MAY 2019
Date of Submission:	09 JULY 2019
Title of the Project:	DEALER INVENTORY OPTIMIZATION
ID No./Name(s)/Discipline of the students	(1) 2017B4A30803P / AKANKSHA KUMAR MSc. Mathematics with B.E. EEE (2) 2017A4PS0919G / ANIRUDH Mechanical Engineering (3) 2017A7PS0218P / SAKSHAM GUPTA Computer Science Engineering
Name of the Guide:	Abhiraj Patkar
Designation of the Guide:	Assistant Manager, ITNI
Name of the PS Faculty:	Pradipta Chattopadhyay
Key Words:	EOQ(Economic Order Quantity), Demand Forecasting, EDA (Exploratory Data Analysis
Project Area:	Supply chain Management , Machine Learning

Abstract

Maruti Suzuki India Limited currently holds a market share of more than 50% of the Indian passenger car market with 3552 sales outlets all over the country. Appropriate inventory management is crucial for the smooth functioning of the organization and to minimize the losses due to stock unavailability and frequent orders .Presently, all dealers of Maruti Suzuki are prescribed to keep 21 days of inventory keeping the order cycle of 15 days. A model to predict expected customer demand using Demand Forecasting and to find out the exact relation between the stock days (days of inventory), delay days (delay in delivery) and service ratio of the dealer for every SKU (Stock Keeping Unit) is described in this report. Demand taken as input from the dealer or from the demand forecasting model has been used in the EOQ model to find out the order cycle for each SKU and the required relation between Stock Days, Delay Days and Service Ratio.

Signature(s) of Student(s) Signature of PS Faculty

Date: Date:

Acknowledgements

I thank the management of BITS-Pilani, Mr. Kenichi Ayukawa, Managing Director and Chief Executive Officer, Maruti Suzuki India Limited for giving me an opportunity to undergo my Practice School-1 program at an automobile manufacturing firm. I would like to express my sincere gratitude towards Mr. Brijesh Sharma, PS –I program coordinator at Maruti Suzuki India Limited, Gurgaon.

I am thankful to Mr. Abhiraj Patkar, Assistant Manager, ITNI my PS guide for giving me an opportunity to pursue this project under his guidance.

I am grateful to Dr. Pradipta Chattopadhyay, Assistant Professor, Department of Chemical Engineering, BITS-Pilani, Pilani Campus for the guidance and support he has provided as our PS-I instructor. I would like to thank Arisha Mahmood for her support and coordination of all the PS related activities.

I would like to thank my friends and family for the motivation and help they have provided during the course of my practice school.

TABLE OF CONTENTS

1. Introduction

1.1 Organization Review

Maruti Suzuki India Limited (MSIL) is a leading automobile manufacturer in India which started as a Joint Venture between Government of India (GoI) and Suzuki Motors Limited (SMC) with a share of 74% with GoI and 26% with SMC. Today, SMC has acquired 56% of the shares and the remaining 44% lie with the Public. The prime objective of MSIL is to maximize production while prioritizing safety. The company has set its motto as "Zero Accidents. Zero Defects"

The company's objective is to manufacture a low cost "People's Car" for middle-class people. By far, it has launched 16 Base Models with a wide range of 1000+ variants.

MSIL manufactures its cars in 3 plants at Gurgaon and Manesar in Haryana and Mehsana in Gujarat. Gurgaon plant is the oldest plant and Gujarat Plant is the newest plant. Apart from these, the Engineering Division for Research and Development is located in Rohtak.

1.2 Objective and Scope of the Project

Inventory control and management is one of the major issues faced by industries these days. This problem is complicated because of the various factors involved. In Maruti Suzuki, where on an average every 9 s, one vehicle comes out of the assembly line, it becomes even more important to handle the large volume of products. Demand of the various SKUs (Model, Variant, Colour) is variable and needs to be considered while managing the inventory. This project is about choosing between infrequent small orders vs frequent small orders balancing the ordering cost and the inventory cost.

The main objective of the project is to find out the relationship between stock days(days of inventory), delay days (delay in delivery) and service ratio for every dealer network for each SKU.

FIG 1

Demand forecasting will be performed on the sales data to obtain the demand which will be further used in the EOQ model to get the desired relationship between stock days, delay days and service ratio.

2. Project Theory

2.1 Inventory Management

Balancing inventory costs and service levels is a huge challenge in supply chain management. Supply chain and financial teams struggle with losses from escalating service inventory and write-offs. Without optimization, obsolete inventory accumulates and causes delays in delivery. The problem complexity increases when the hidden costs of unavailable inventory such as service penalties, expedited shipping costs and lesser customer satisfaction are considered.

Maruti Suzuki has prescribed all its dealers to keep 21 days of inventory. Their ordering period cycle is 15 days. In order to optimise the stock days with permissible delay days, we need to find the service ratio for each dealer per SKU.

Stock Days- This is referred to as the days of inventory that should be kept by the dealer group to avoid a stock out.

Service Ratio-This is the fraction of customer demand that is met through immediate stock availability, without backorders or lost sales.

Permissible Delay Days- This is the difference between the date of delivery of the product to the customer and the date of booking.

2.2 Exploratory Data Analysis(EDA)

Exploratory Data Analysis(EDA) refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations.

It is basically a process of looking into the data understanding it and getting comfortable with it. It is required to generate powerful features and to build accurate models.

Exploratory Data Analysis allows to:

- Better understand the data
- Build intuition about the data
- Generate hypothesis
- Find insights
- Find magic features

2.2.1 Feature Selection

Feature Selection is one of the core concepts in machine learning which hugely impacts the performance of your model. The data features that you use to train your machine learning models have a huge influence on the performance you can achieve.
Irrelevant or partially relevant features can negatively impact model performance.
Importance of good feature selection:

1. Reduces Overfitting: Less redundant data means less opportunity to make decisions based on noise.
2. Improves Accuracy: Less misleading data means modeling accuracy improves.
3. Reduces Training : fewer data points reduce algorithm complexity and Time algorithms train faster.

Feature Selection techniques:

1. Univariate Selection
2. Feature Importance
3. Correlation Matrix with Heatmap

To test the hypothesis we come up with after Exploratory Data Analysis, we do Visualization.

2.2.2 Data Visualization

Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.
Data visualization is one of the core skills in data science. In order to start building useful models, we need to understand the underlying dataset.

There are three different types of analysis:-

1. Univariate – In univariate analysis we use a single feature to analyze its properties. It is done to find out about the distribution of the variable(Standard Normal, Exponential, Student-t), missing values, mean, variance and outliers.

2. Bivariate - When we compare the data between exactly 2 features then its called bivariate analysis. It is done to find relation/equation between 2 variables.

3. Multivariate - Comparing more than 2 variables is called as Multivariate analysis. It is done to find out whether there are any dependent or redundant variables.

FIG 2

These are some of the tools used in data visualization.

2.3 Demand Forecasting

Demand Forecasting is the process in which historical sales data is used to develop an estimate of an expected forecast of customer demand. To businesses, Demand Forecasting provides an estimate of the amount of goods and services that its customers will purchase in the foreseeable future. Critical business assumptions like turnover, profit margins, cash flow, capital expenditure, risk assessment and mitigation plans, capacity planning, etc. are dependent on Demand Forecasting.

Short to medium term tactical plans like pre-building, make-to-stock, make-to-order, contract manufacturing, supply planning, network balancing, etc. are execution based. Demand Forecasting also facilitates important management activities like decision making, performance evaluation, judicious allocation of resources in a constrained environment and business expansion planning.

2.3.1 Time Series Models

The problem at our hand is a time-series forecasting problem. A time series is a sequence of observation of data points measured over a time interval. The difference in this case and normal regression case is that the data is time-stamped and a time dimension is also added.

Understanding a time series problem

Since we are dealing with a time series problem, we have to take into account certain factors like -

- The model should take into account the **seasonal** nature of data.
- The model should take care of the **timestamping** of data.
- The model should be able to identify **trends**.
- The model should not consider a jump in sales as an **outlier.**

Models studies by us during the course of this project are as follows:

1. **Autoregression (AR)**

Autoregression (AR) method models the next step in the sequence as a linear function of the observations at prior time steps. Good for univariate time series without trend and seasonal components.

2. **Autoregressive Integrated Moving Average (ARIMA)**

The Autoregressive Integrated Moving Average (ARIMA) method models the next step in the sequence as a linear function of the differenced observations and residual errors at prior time steps.

It combines both Autoregression (AR) and Moving Average (MA) models as well as a differencing pre-processing step of the sequence to make the sequence stationary, called integration (I).The method is suitable for univariate time series with trend and without seasonal components.

3. Seasonal Autoregressive Integrated Moving-Average (SARIMA)

It combines the ARIMA model with the ability to perform the same autoregression, differencing, and moving average modeling at the seasonal level.

The method is suitable for univariate time series with trend and/or seasonal components.

4. Holt Winter's Exponential Smoothing (HWES)

The Holt Winter's Exponential Smoothing (HWES) also called the Triple Exponential Smoothing method models the next time step as an exponentially weighted linear function of observations at prior time steps, taking trends and seasonality into account.

The method is suitable for univariate time series with trend and/or seasonal components.

5. XGBoost

XGBoost stands for eXtreme Gradient Boosting.

Gradient boosting is an approach where new models are created that predict the residuals or errors of prior models and then added together to make the final prediction. It is called gradient boosting because it uses a gradient descent algorithm to minimize the loss when adding new models.

6. Random Forest

The random forest is a classification algorithm consisting of many decisions trees. It uses bagging and feature randomness when building each individual tree to try to create an

uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree.

7. Recurrent Neural Network

A recurrent neural network is a class of artificial neural network where connections between nodes form a directed graph along a sequence. This allows it to exhibit dynamic temporal behavior for a time sequence.

They are networks with loops in them, allowing information to persist.

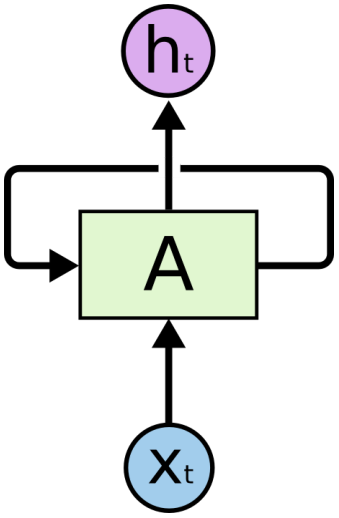


FIG 3:*Recurrent Neural Networks have loops.*

In the above diagram, a chunk of neural network, A, looks at some input xt and outputs a value ht . A loop allows information to be passed from one step of the network to the next.

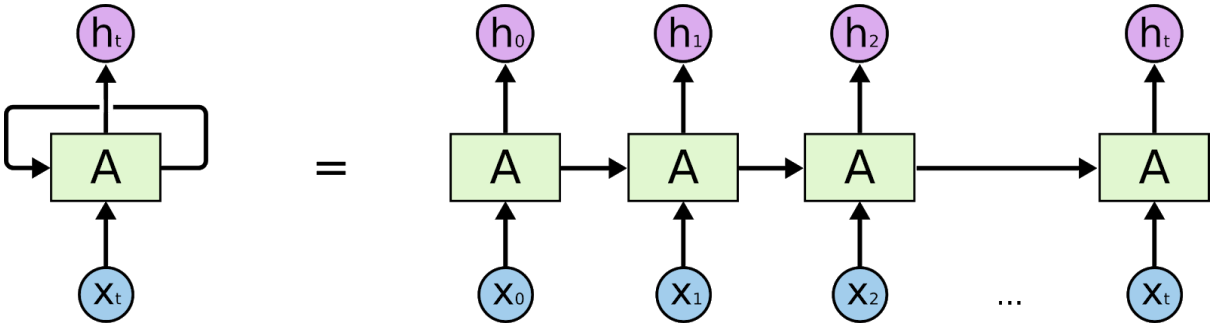


FIG 4:*An unrolled recurrent neural network*

This chain-like nature reveals that recurrent neural networks are intimately related to sequences and lists.

6. Long Short-Term Memory (LSTM)

Long Short Term Memory networks are a special kind of RNN, capable of learning long-term dependencies. LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn!

All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.

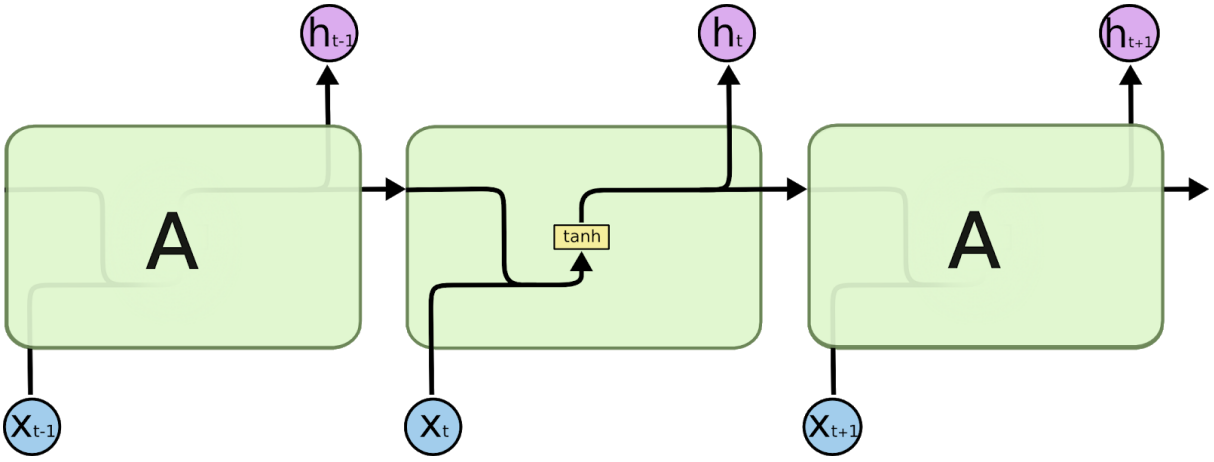


FIG 5:*The repeating module in a standard RNN contains a single layer.*

LSTMs also have this chain like structure, but the repeating module has a different structure. Instead of having a single neural network layer, there are four, interacting in a very special way.

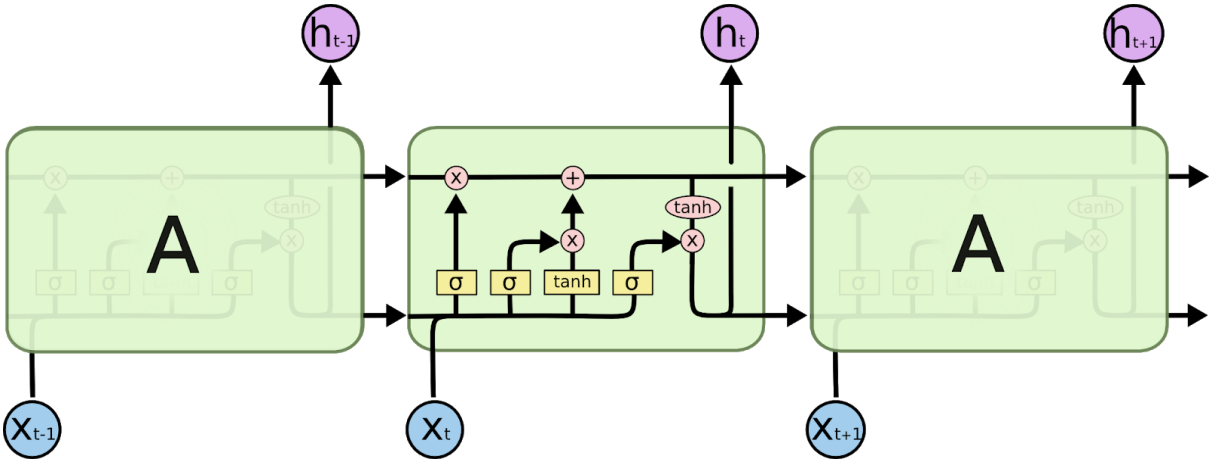
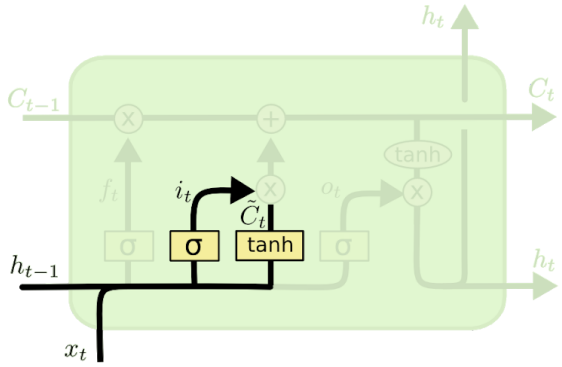


FIG 6:*The repeating module in an LSTM contains four interacting layers.*

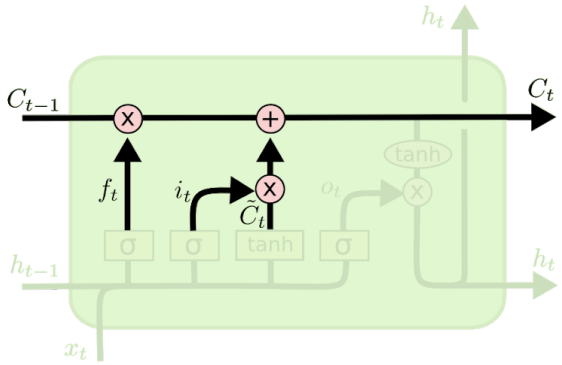
7. Step-by-Step LSTM Walk Through



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$

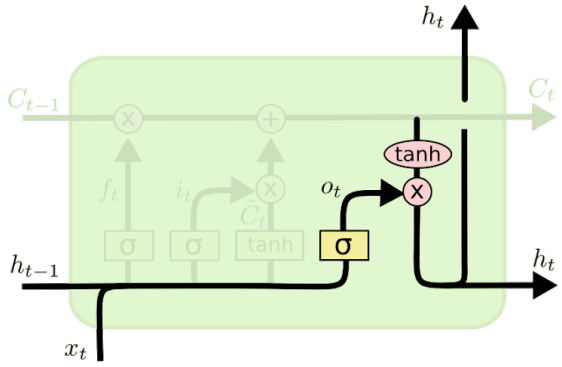
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

FIG 7



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

FIG 8



$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

FIG 9

2.4 Economic Order Quantity

EOQ is a tool used to determine the volume and frequency of orders required to satisfy a given level of demand while minimizing the cost per order.

2.4.1 Importance of EOQ

The Economic Order Quantity is a set point designed to help companies minimize the cost of ordering and holding inventory. The cost of ordering an inventory falls with the increase in ordering volume due to purchasing on economies of scale. However, as the size of inventory grows, the cost of holding the inventory rises. EOQ is the exact point that minimizes both these inversely related costs.

2.4.2 EOQ Formula

The Economic Order Quantity formula is calculated by minimizing the total cost per order by setting the first order derivative to zero. The components of the formula that make up the total cost per order are the cost of holding inventory and the cost of ordering that inventory. The key notations in understanding the EOQ formula are as follows:

2.4.3 Components of the EOQ Formula:

D: Annual Quantity Demanded

Q: Volume per Order

S: Ordering Cost (Fixed Cost)

C: Unit Cost (Variable Cost)

H: Holding Cost (Variable Cost)

i: Carrying Cost (Interest Rate)

1) **Ordering Cost**

The number of orders that occur annually can be found by dividing the annual demand by the volume per order. The formula can be expressed as:

$$\text{Number of Orders} = \frac{D}{Q}$$

FIG 10

2) Holding Cost

Holding inventory often comes with its own costs. This cost can be in the form of direct costs incurred by financing the storage of said inventory or the opportunity cost of holding inventory instead of investing the money tied up in inventory elsewhere. As such, the holding cost per unit is often expressed as the cost per unit multiplied by the interest rate, expressed as follows:

$$H = iC$$

With the assumption that demand is constant, the quantity of stock can be seen to be depleting at a constant rate over time. When inventory reaches zero, an order is placed and replenishes inventory as shown:

12

FIG 11

As such, the holding cost of the inventory is calculated by finding the sum product of the inventory at any instant and the holding cost per unit. It is expressed as follows:

$$\text{Annual Holding Cost} = \frac{Q}{2} \times H$$

FIG 12

3)Total Cost and the Economic Order Quantity

Summing the two costs together gives the annual total cost of orders. To find the optimal quantity that minimizes this cost, the annual total cost is differentiated with respect to Q. It is shown as follows:

13

$$\text{Annual Total Cost (TC)} = \frac{D}{Q} \times S + \frac{Q}{2} \times H$$

FIG 13

$$EOQ = \frac{dTC}{dQ} = \sqrt{\frac{2SD}{H}}$$

FIG 14

3. Data Pre-processing

3.1 Dataset Required

Historical sales data of the year 2017 and 2018 for two models - WagonR and Swift for the states Uttar Pradesh and Karnataka is used for demand forecasting. The model will further be applied to other models and other states.

3.2 Dataset Constraints

Being sales data, the data was nowhere close to being linear and also there were a lot of discrepancies and missing values. So the following were also added to our list of concerns regarding model selection.

- The model should be able to **smoothen** data, or else we will have to smoothen the data ourselves.(We actually had to smoothen it ourselves in the end because the smoothening models were not performing well.)
- The model should have a good algorithm to deal with missing values.

14

3.3 Feature Selection

The major features used for demand forecasting will be the following:

Date of Sale	SKU ID	DEALER ID	UNITS SOLD
01-07-2018	1	1	3

4. EDA of the dataset

On performing EDA on the given sales data, we analysed how the sales of each SKU varied over the months for every dealer group.The following graph describes the sales for the company using historical data. This graph also took care of the seasonality factor which is observed in the demand of any product.

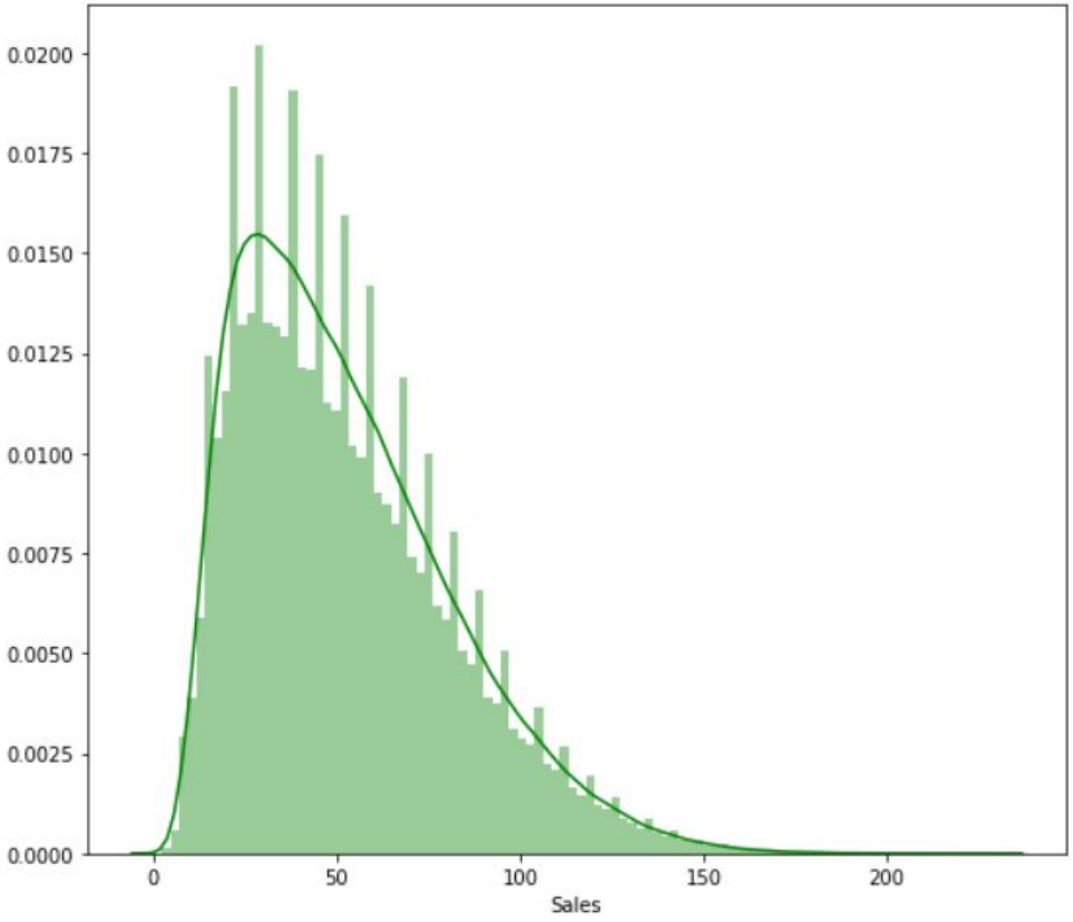


FIG 15

Various graphs for visualising the datasets grouping them dealer groups wise, SKU wise , month wise etc. Following is a histogram displaying trends with the various groupings.

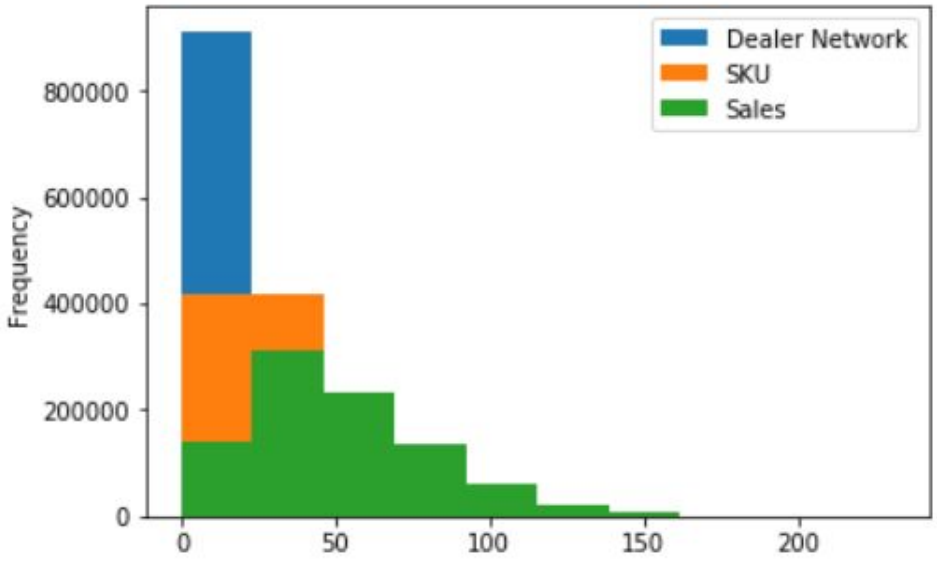


FIG 16

We also visualised how delay days for every SKU varied with service ratio. Similarly, graphs were made between all the input variables in the dataset.

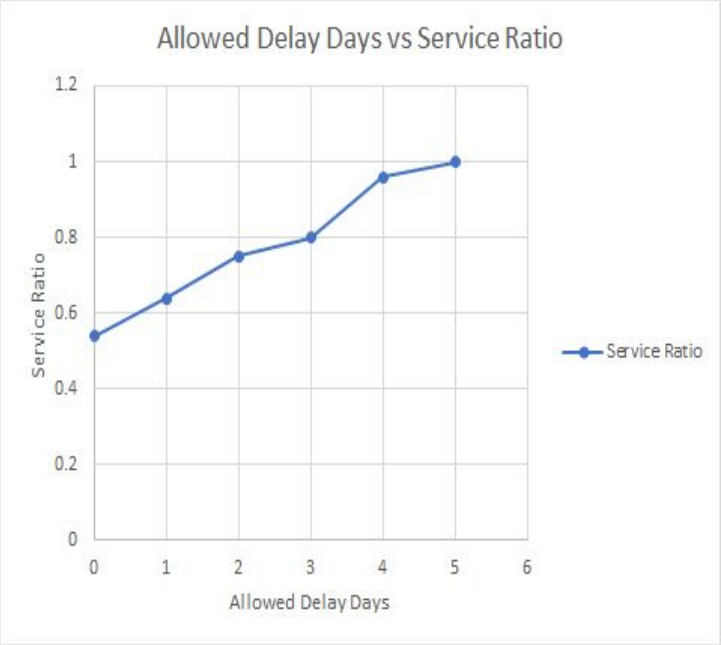


FIG 17

Amongst which, delay days and service ratio showed an obvious trend that can be seen in the following graph.

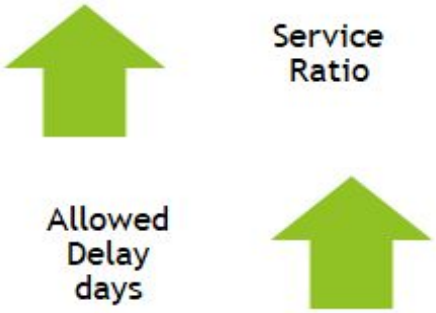


FIG 18

5. Models employed for Demand Forecasting

For our dataset, we employed ARIMA model , LSTM and XGBoost. Our results came out after pooling up the data SKU wise. LSTM showed better results in comparison to the other models.

1) **Econometric forecasting technique:**

Econometric forecasting utilizes autoregressive integrated moving-average and complex mathematical equations, to establish relationships between demand and factors that influence the demand. An equation is derived and fine-tuned to ensure a reliable historical representation. Finally, the projected values of the influencing variables are inserted into the equation to generate a forecast.

2) Regression analysis:

This helps understand relationships and help predict continuous variables based on other variables in the dataset. This technique is designed to identify meaningful relationships among data variables, specifically looking at the connections between a dependent variable and other independent factors that may or may not affect it.

6. Validation of the demand forecasting model

In machine learning, model validation is the process where a trained model is evaluated with a testing data set. The testing data set is a separate portion of the same data set from which the training set is derived. The main purpose of using the testing data set is to test the generalization ability of a trained model. Model validation is carried out after model training. There are many validation strategies(80-20 rule, Leave the Last, k-fold). In our project we used k-Fold Cross-Validation.

6.1 K-Fold Cross-Validation

Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample. The procedure has a single parameter called k that refers to the number of groups that a given data sample is to be split into. As such, the procedure is often called k-fold cross-validation.

7. Cost Optimization

The demand obtained from the demand forecasting model is used in the EOQ optimization model. Our model also allows the user to enter the demand and get the results accordingly. The user also inputs the inventory and ordering cost for the particular SKU. Using the code given in Appendix 2, the order period cycle is calculated.

FIG 19

The order period cycle is used to give the relation between stock days, delay days and service ratio. (Refer Appendix 2 for the code.) The figure below shows the result for T= 19 days.

STOCK DAYS	ALLOWED DELAY DAYS	SERVICE RATIO
19	0	1.0
18	1	1.0
18	0	0.94
17	2	1.0

17	1	0.94
17	0	0.89

8. Conclusion

After looking at the performance of the aforementioned algorithms with the given set of training and testing data, it can be concluded that LSTM gives better results when trained for sufficient number of epochs. It can also be concluded that LSTM along with EDA and EOQ achieves better accuracy of prediction of stock days and order quantity compared to any other neural network algorithm. Training the model with a large dataset is helpful to train it for random sales patterns and varying demand since the LSTM doesn't tend to forget the weights learned in the previous epochs. In this paper it can be concluded that the EOQ model achieves good accuracy if applied on the clusters of data. Future directions of the research include predicting the more accurate order quantity and stock days considering many other factors influencing the variables like transit time, demand and inventory cost, and devising more efficient algorithms.

9. Appendices

Appendix A: Input Variables

- Dealer code
- Model of the car
- Variant
- Month
- Demand per month
- Transit time
- Inventory holding cost
- Reordering cost

Appendix B: Applying EOQ to relate the business terms - Service Ratio, Delay Days, Stock Days and Demand(EOQ)

```
#Getting input
DEMAND= "ENTER DEMAND PER MONTH"
HC = "ENTER HOLDING COST PER UNIT PER MONTH"
OC = "ENTER ORDERING COST PER UNIT PER MONTH"

T =   (30 * 2 *d) / (HC * OC)                    # ORDER PERIOD FROM EOQ
for STOCK_DAYS in range(T ,-1 ,-1):
   for ALLOWED_DD in rangeI( T-STOCK_DAYS, -1, -1):
      SERVICE_RATIO = (STOCK_DAYS+ DELAY DAYS)/T
      print(SERVICE_RATIO)
```

Appendix C: Implementation in python

```
import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

df=pd.read_csv()

grouped=df.groupby(['PARENT_GROUP','VARIANT_CD','COLOR_CD'])    #grouping by SKU

num_group=grouped.ngroups           #getting the number of SKU

ser_ratio=[]

del_days=[]

for name, group in grouped:                    #selecting a SKU
```

```
min_d_day=group['DELAY_DAYS'].min()
max_d_day=group['DELAY_DAYS'].max()                                    for
spec_d_day in range(min_d_day,max_d_day):                              #selecting
a number in that range
        num_d_day=0                  #total number of entries of that SKU
        req_d_day=0        #allowed delay days (less than or equal to spec_d_day)
        for d_day in group['DELAY_DAYS']:
                num_d_day=num_d_day+1
                if d_day<=spec_d_day:
                        req_d_day=req_d_day+1
        del_days.append(spec_d_day)
        ser_ratio.append(req_d_day/num_d_day)
plt.plot(x='del_days', y='ser_ratio', label=name)
```

Appendix D: Feature Selection Code

1. Univariate Selection

```
import pandas as pd
import numpy as np
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import chi2
data = pd.read_csv("D://DIO//train.csv")
X = data.iloc[:,0:20]  #independent columns
```

```
y = data.iloc[:,-1]    #target column i.e price range

#apply SelectKBest class to extract top 10 best features

bestfeatures = SelectKBest(score_func=chi2, k=10)

fit = bestfeatures.fit(X,y)

dfscores = pd.DataFrame(fit.scores_)

dfcolumns = pd.DataFrame(X.columns)

#concat two dataframes for better visualization

featureScores = pd.concat([dfcolumns,dfscores],axis=1)

featureScores.columns = ['Specs','Score']  #naming the dataframe columns

print(featureScores.nlargest(10,'Score'))  #print 10 best features
```

2. Feature Importance

```
import pandas as pd

import numpy as np

data = pd.read_csv("D://DIO//train.csv")

X = data.iloc[:,0:20]  #independent columns

y = data.iloc[:,-1]    #target column i.e price range

from sklearn.ensemble import ExtraTreesClassifier

import matplotlib.pyplot as plt

model = ExtraTreesClassifier()

model.fit(X,y)
```

```
print(model.feature_importances_) #use inbuilt class feature_importances of tree based classifiers

#plot graph of feature importances for better visualization

feat_importances = pd.Series(model.feature_importances_, index=X.columns)

feat_importances.nlargest(10).plot(kind='barh')

plt.show()
```

3.Correlation Matrix with Heatmap

```
import pandas as pd

import numpy as np

import seaborn as sns

data = pd.read_csv("D://DIO//train.csv")

X = data.iloc[:,0:20]  #independent columns

y = data.iloc[:,-1]    #target column i.e price range

#get correlations of each features in dataset

corrmat = data.corr()

top_corr_features = corrmat.index

plt.figure(figsize=(20,20))

#plot heat map

g=sns.heatmap(data[top_corr_features].corr(),annot=True,cmap="RdYlGn")
```

Appendix E - Validation pseudo-code

Divide the training data set into K bins of about equal size.

repeat K times:

leave out one bin

train both algorithms on all other bins.

check the performance of both trained models on the left-out bin.

Repeat for each bin.

Take the averages of the performance for all bins, for each algorithm. The one with better performance can be expected to be better for this data set. Now train it on the full data set

10.References

https://blog.arkieva.com/demand-forecasting/

https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15

https://www.tableau.com/learn/articles/data-visualization

Wang H., Zheng H. (2013) Model Validation, Machine Learning. In: Dubitzky W., Wolkenhauer O., Cho KH., Yokota H. (eds) Encyclopedia of Systems Biology. Springer, New York, NY

https://machinelearningmastery.com/k-fold-cross-validation/

http://www.lokad.com/resources