Capstone Project

Supply Chain

Final Report

Piyush Kumar Singh
PGP – DSBA Online
May-21 Batch

Date: 04/06/2022

Table of Contents

List of Figures

List of Tables

1) Introduction to Problem Statement

a) Defining problem statement

A FMCG company has entered into the instant noodles business two years back. Their higher management has notices that there is a miss match in the demand and supply. Where the demand is high, supply is pretty low and where the demand is low, supply is pretty high. In both the ways it is an inventory cost loss to the company; hence, the higher management wants to optimize the supply quantity in each and every warehouse in entire country.

File: Data.csv

Target variable: product_wg_ton

Data dictionary

Variable	Business Definition
Ware_house_ID	Product warehouse ID
WH_Manager_ID	Employee ID of warehouse manager
Location_type	Location of warehouse like in city or village
WH_capacity_size	Storage capacity size of the warehouse
zone	Zone of the warehouse
WH_regional_zone	Regional zone of the warehouse under each zone
num_refill_req_l3m	Number of times refilling has been done in last 3 months
transport_issue_l1y	Any transport issue like accident or goods stolen reported in last oneyear
Competitor_in_mkt	Number of instant noodles competitor in the market
retail_shop_num	Number of retails shop who sell the product under the warehouse area
wh_owner_type	Company is owning the warehouse or they have get the warehouse onrent
distributor_num	Number of distributer works in between warehouse and retail shops
flood_impacted	Warehouse is in the Flood impacted area indicator
flood_proof	Warehouse is flood proof indicators. Like storage is at some height not directly on the ground
electric_supply	Warehouse have electric back up like generator, so they can run the warehouse in load shedding
dist_from_hub	Distance between warehouse to the production hub in Kms
workers_num	Number of workers working in the warehouse
wh_est_year	Warehouse established year
storage_issue_reported_l3m	Warehouse reported storage issue to corporate office in last 3 months.Like rat, fungus because of moisture etc.
temp_reg_mach	Warehouse have temperature regulating machine indicator
approved_wh_govt_certificate	What kind of standard certificate has been issued to the warehouse from government regulatory body
wh_breakdown_l3m	Number of time warehouse face a breakdown in last 3 months. Like strike from worker, flood, or electrical failure
govt_check_l3m	Number of time government Officers have been visited the warehouse to check the quality and expire of stored food in last 3 months
product_wg_ton	Product has been shipped in last 3 months. Weight is in tons

Table.1

b) Need of the study/project

The objective of this exercise is to build a model, using historical data that will determine an optimum weight of the product to be shipped each time to the warehouse. This will help in meeting the demand of the product and increase in revenue. It will also help in reducing the supply of product to warehouse where the demand in not high thus reducing loss of inventory.

We also to analysis the demand pattern in different pockets of the country so management can drive the advertisement campaign particular in those pockets.

c) Understanding business/social opportunity

- If a warehouse has more products and the demand of the product is less at that location it means loss of inventory as product will be expired soon.
- If a warehouse has less product but the demand in more at the location it result in loss of opportunity to make revenue.
- Both these condition are not favourable if we are trying to maximise our profits. Thus we are using predictive modelling techniques to find the optimal weight of the product that can be send to each of the warehouse so that the demand is met at that location and also the wastage of inventory is also reduced.
- If the demand is increasing for a particular area finding optimal weight also helps us make decision on advance stock that needs to be in warehouse so that business don't lose customers to competitors in case of very high demand.
- Once the optimal weight is assigned to each of the warehouse, we can then focus on the advertising part to target the market where the demand is less of the product. We can also properly allocated the funds for advertising depending on the demand of the location where the warehouse is located. Where the demand is more advertisement for that area can be reduced and where the demand is less, more funds can allocated to advertise in those areas.
- Finding optimal weight of the product will also help us identify which of the warehouse are not performing, are not able to push the product into the market. Business can choose to close such warehouse where is product is sitting ideal for long periods of time and whose warehouse which are not fully 100% utilised. For such locations where warehouses need to be closed the demand for that location can be met by nearby warehouse from different region till the demand is high enough to make the opening of new warehouse feasible. This can be opportunity of cost reduction activity for the business.
- Finding the optimal weight of the product for each warehouse will also help us to reduce logistic cost of transporting the product. It will make the delivery of the product timely and as per demand, reduce unnecessary movement of goods.

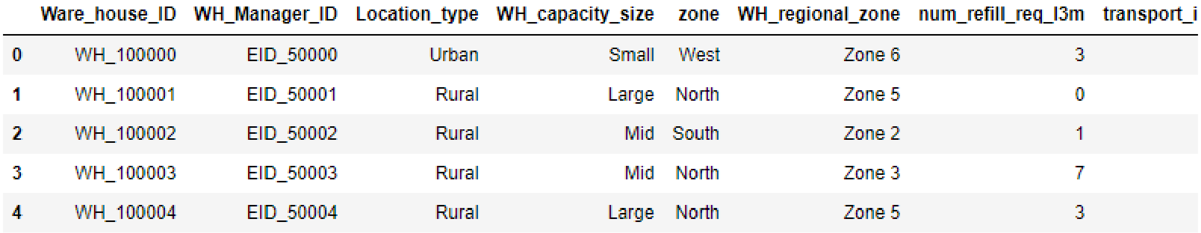
Data Exploration

a) Understanding how data was collected in terms of time, frequency and methodology

For the problem statement we are given only 3 months historical data, so time and frequency are not applicable for our dataset. Based on the 3 months data given to us we need to predict for the next 3 months in future. So if a new warehouse is to open in a new area based on our prediction we can find out the optimal weight required for the warehouse.

b) Visual inspection of data (rows, columns, descriptive details)

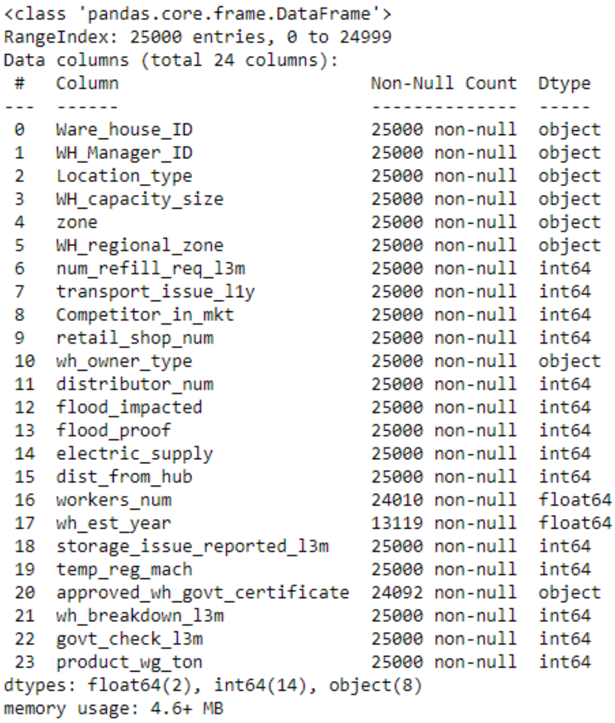
- The following dataset is provided to us in csv format and has 25000 rows and 24 columns.
- The following is the head of the dataset.



	Ware_house_ID	WH_Manager_ID	Location_type	WH_capacity_size	zone	WH_regional_zone	num_refill_req_l3m	transport_i
0	WH_100000	EID_50000	Urban	Small	West	Zone 6	3	
1	WH_100001	EID_50001	Rural	Large	North	Zone 5	0	
2	WH_100002	EID_50002	Rural	Mid	South	Zone 2	1	
3	WH_100003	EID_50003	Rural	Mid	North	Zone 3	7	
4	WH_100004	EID_50004	Rural	Large	North	Zone 5	3	

Table.2

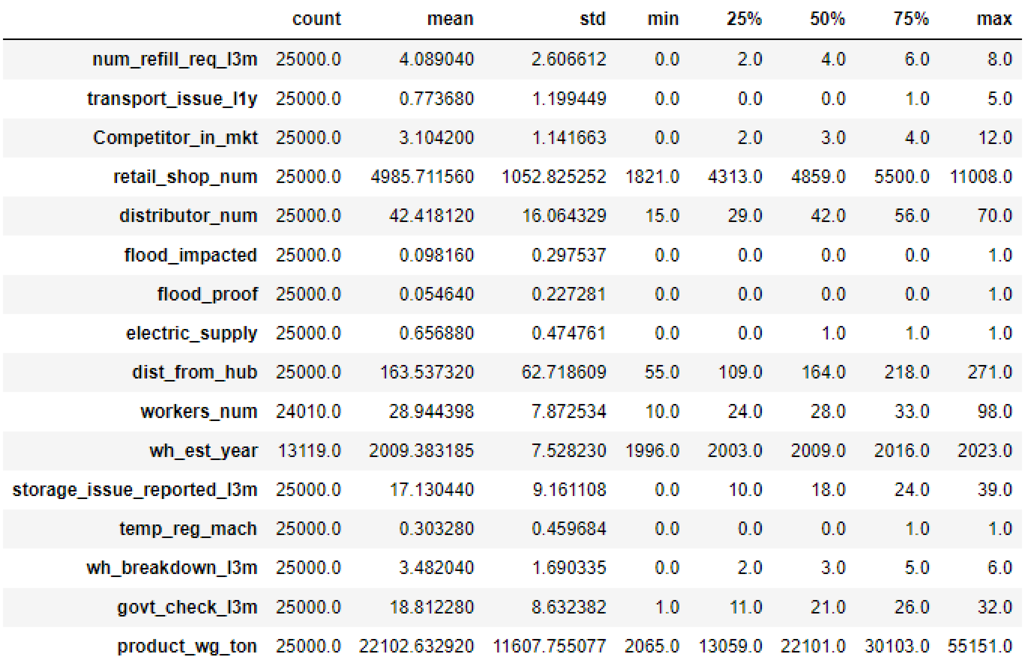
- Out of the 24 columns in the dataset, there are
  - Float64 type – 2 columns
  - Int64 type – 14 columns
  - Object type – 8 columns



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25000 entries, 0 to 24999
Data columns (total 24 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   Ware_house_ID                 25000 non-null  object
 1   WH_Manager_ID                 25000 non-null  object
 2   Location_type                 25000 non-null  object
 3   WH_capacity_size              25000 non-null  object
 4   zone                          25000 non-null  object
 5   WH_regional_zone              25000 non-null  object
 6   num_refill_req_l3m            25000 non-null  int64
 7   transport_issue_l1y           25000 non-null  int64
 8   Competitor_in_mkt             25000 non-null  int64
 9   retail_shop_num               25000 non-null  int64
 10  wh_owner_type                 25000 non-null  object
 11  distributor_num               25000 non-null  int64
 12  flood_impacted                25000 non-null  int64
 13  flood_proof                   25000 non-null  int64
 14  electric_supply               25000 non-null  int64
 15  dist_from_hub                 25000 non-null  int64
 16  workers_num                   24010 non-null  float64
 17  wh_est_year                   13119 non-null  float64
 18  storage_issue_reported_l3m    25000 non-null  int64
 19  temp_reg_mach                 25000 non-null  int64
 20  approved_wh_govt_certificate  24092 non-null  object
 21  wh_breakdown_l3m              25000 non-null  int64
 22  govt_check_l3m                25000 non-null  int64
 23  product_wg_ton                25000 non-null  int64
dtypes: float64(2), int64(14), object(8)
memory usage: 4.6+ MB
```

Table.3

- Ware_house_ID and WH_Manager_OD columns are non-contributing columns so I am dropping them from our analysis.
- Below table gives data description for all the continuous variable in the dataset.



	count	mean	std	min	25%	50%	75%	max
num_refill_req_l3m	25000.0	4.089040	2.606612	0.0	2.0	4.0	6.0	8.0
transport_issue_l1y	25000.0	0.773680	1.199449	0.0	0.0	0.0	1.0	5.0
Competitor_in_mkt	25000.0	3.104200	1.141663	0.0	2.0	3.0	4.0	12.0
retail_shop_num	25000.0	4985.711560	1052.825252	1821.0	4313.0	4859.0	5500.0	11008.0
distributor_num	25000.0	42.418120	16.064329	15.0	29.0	42.0	56.0	70.0
flood_impacted	25000.0	0.098160	0.297537	0.0	0.0	0.0	0.0	1.0
flood_proof	25000.0	0.054640	0.227281	0.0	0.0	0.0	0.0	1.0
electric_supply	25000.0	0.656880	0.474761	0.0	0.0	1.0	1.0	1.0
dist_from_hub	25000.0	163.537320	62.718609	55.0	109.0	164.0	218.0	271.0
workers_num	24010.0	28.944398	7.872534	10.0	24.0	28.0	33.0	98.0
wh_est_year	13119.0	2009.383185	7.528230	1996.0	2003.0	2009.0	2016.0	2023.0
storage_issue_reported_l3m	25000.0	17.130440	9.161108	0.0	10.0	18.0	24.0	39.0
temp_reg_mach	25000.0	0.303280	0.459684	0.0	0.0	0.0	1.0	1.0
wh_breakdown_l3m	25000.0	3.482040	1.690335	0.0	2.0	3.0	5.0	6.0
govt_check_l3m	25000.0	18.812280	8.632382	1.0	11.0	21.0	26.0	32.0
product_wg_ton	25000.0	22102.632920	11607.755077	2065.0	13059.0	22101.0	30103.0	55151.0

Table.4

Inferences

- We can see that data scaling will be required before modelling can be done.
- Maximum product weight in a warehouse is 55151 tons and lowest is 2065 tons.
- Maximum breakdown incidences in a warehouse is 6 and minimum is 0.
- Maximum storage issues reported in a warehouse is 39 and minimum is 0.
- Maximum competitors in market at certain location near warehouse is 12 and minimum is 0.
- Maximum refill done in a warehouse is 8 and minimum is 0. Mean refill done for a warehouse is 4.

c) Understanding of attributes (variable info, renaming if required)

- All the variables in the dataset are correctly named and there is no need for renaming them as none of them contain spaces between them or any special character.
- As we dropped 2 non-contributing variable earlier, we are now left with only 22 columns for our analysis.
- All the rest variables are useful predictors for our target variable product_wg_ton.

Exploratory Data Analysis

a) Univariate analysis

(Distribution and spread for every continuous attribute, distribution of data in categories for categorical ones)

Univariate Analysis – Continuous Variables

We perform the univariate analysis on the data set and display distplot to check distribution of each column and use boxplot to check for outliers if any.

num_refill_req_l3m -

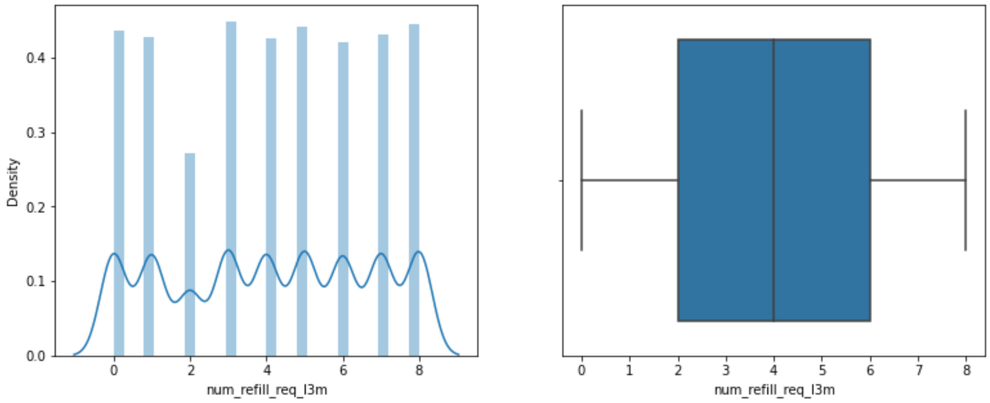


Fig.1

- **num_refill_req_l3m** is fairly symmetrical, std = 2.60, mean = 4.08.
- SKEWNESS = -0.07 and Kurtosis = -1.22
- There are no outliers present.
- Data is spread from 0 to 8 value.

transport_issue_l1y -

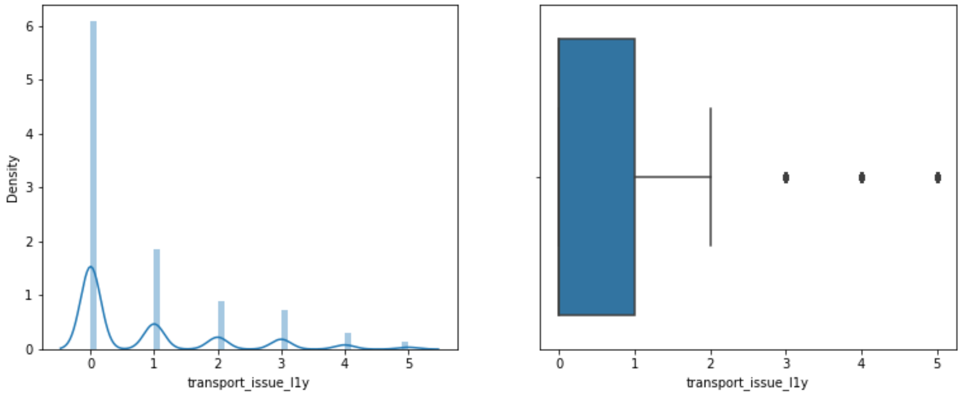


Fig.2

- **transport_issue_l1y** is highly positive skewed, std = 1.19, mean = 0.77.
- SKEWNESS = 1.61 and Kurtosis = 1.83
- There are outliers present but we can ignore them as these values are relevant to our analysis.
- Data is spread from 0 to 5 value.

Competitor_in_mkt -

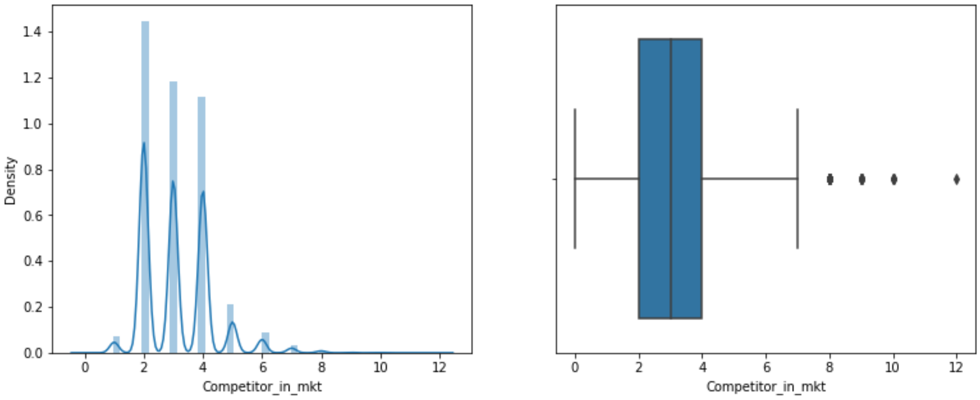


Fig.3

- **Competitor_in_mkt** is moderately positive skewed, std = 1.14, mean = 3.10.
- SKEWNESS = 0.97 and Kurtosis = 1.78
- There are outliers present but we can ignore them as these values are relevant to our analysis.
- Data is spread from 0 to 12 value.

retail_shop_num -

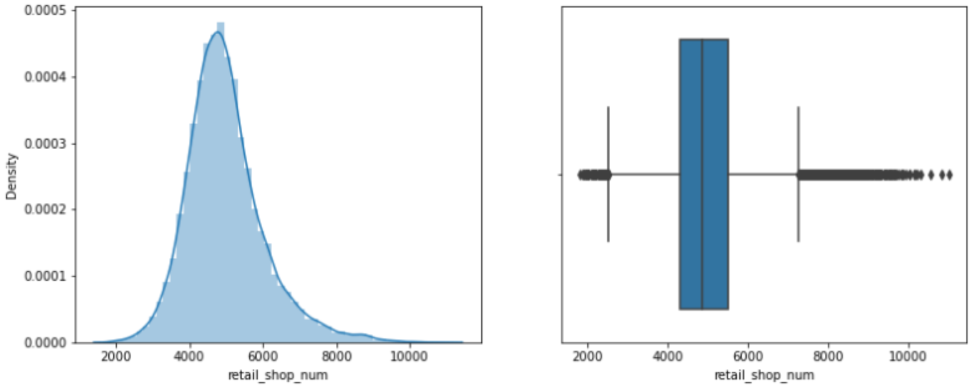


Fig.4

- **retail_shop_num** is moderately positive skewed, std = 1052.82, mean = 4985.71.
- SKEWNESS = 0.90 and Kurtosis = 1.85
- There are outliers present.
- Data is spread from 1821 to 11008 value.

distributor_num –

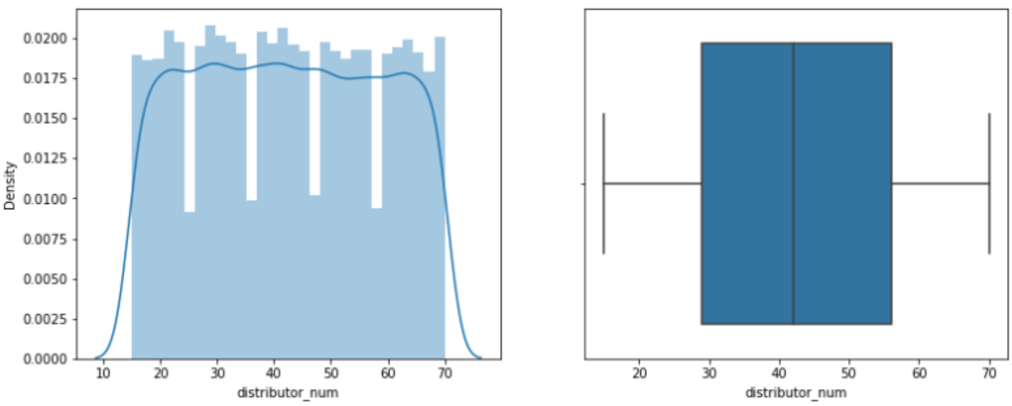


Fig.5

- **distributor_num** is fairly symmetric, std = 16.06, mean = 42.41.
- SKEWNESS = 0.01 and Kurtosis = -1.18
- There are no outliers present.
- Data is spread from 15 to 70 value.

flood_impacted -

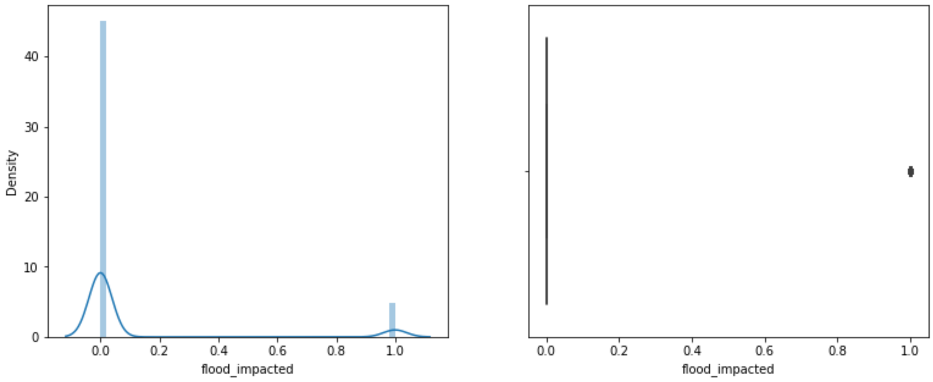


Fig.6

- **flood_impacted** is highly positive skewed, std =0.29 , mean = 0.09.
- SKEWNESS = 2.70 and Kurtosis = 5.29
- No need to look for outliers for this variable as it has only 2 values.
- 0 and 1 value only present.

flood_proof -

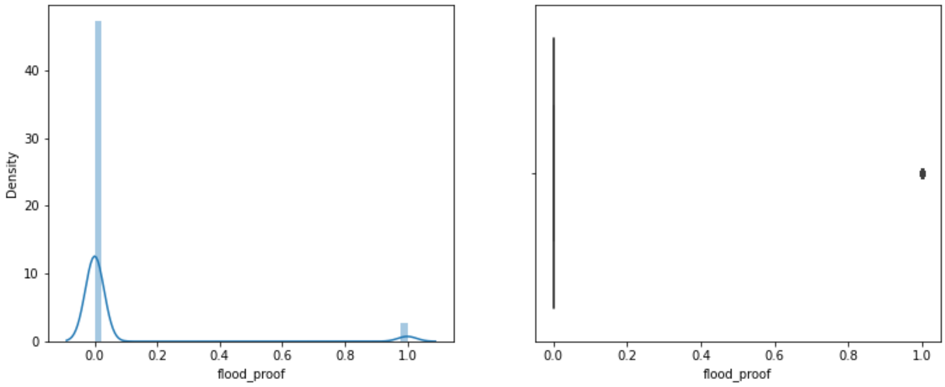


Fig.7

- **flood_proof** is highly positive skewed, std = 0.22, mean = 0.05.
- SKEWNESS = 3.91 and Kurtosis = 13.36
- No need to look for outliers for this variable as it has only 2 values.
- 0 and 1 value only present.

electric_supply -

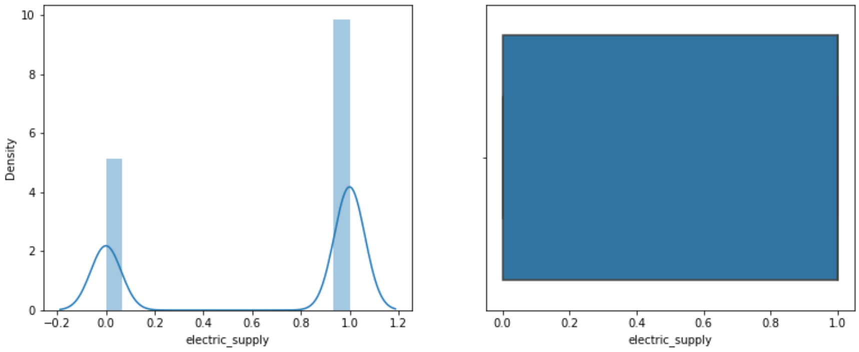


Fig.8

- **electric_supply** is moderately negative skewed, std = 0.47, mean = 0.65.
- SKEWNESS = -0.66 and Kurtosis = -1.56
- No need to look for outliers for this variable as it has only 2 values.
- 0 and 1 value only present.

dist_from_hub -

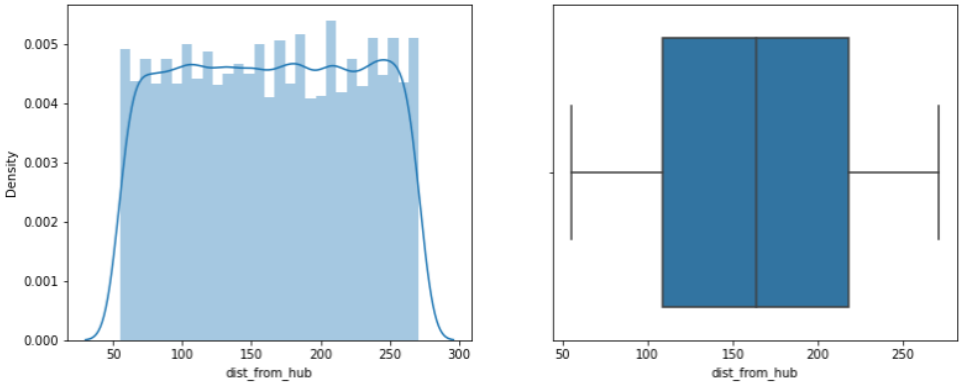


Fig.9

- **dist_from_hub** is fairly symmetric, std = 62.71, mean = 163.53.
- SKEWNESS = -0.005 and Kurtosis = -1.20
- There are no outliers present.
- Data is spread from 55 to 271 value.

workers_num -

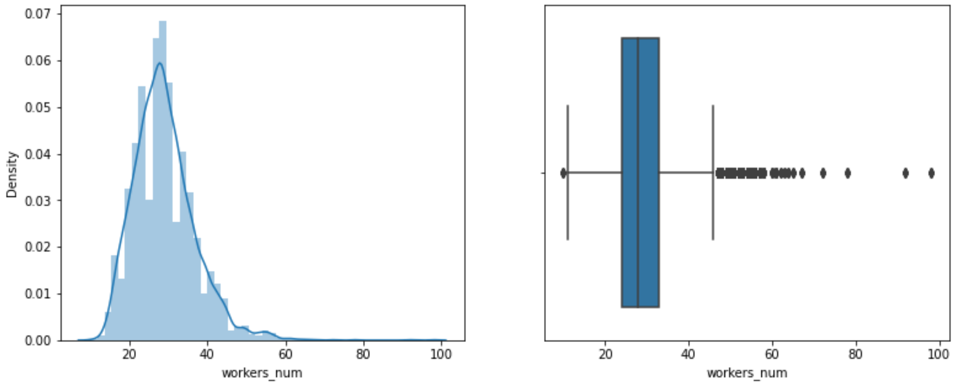


Fig.10

- **workers_num** is highly positive skewed, std = 7.87, mean = 28.94.
- SKEWNESS = 1.05 and Kurtosis = 3.40
- There are outliers present.
- Data is spread from 10 to 98 value.

wh_est_year -

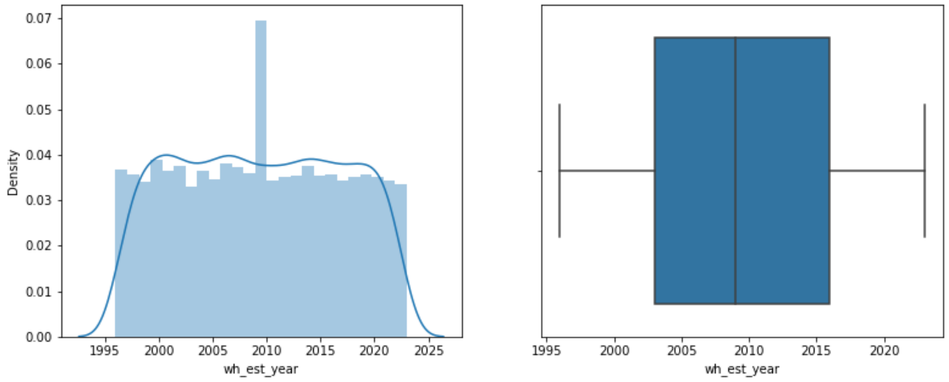


Fig.11

- **wh_est_year** is fairly symmetric, std = 7.52, mean = 2009.38.
- SKEWNESS = 0.012 and Kurtosis = -1.17
- There are no outliers present.
- Data is spread from 1996 to 2023 value.

storage_issue_reported_l3m -

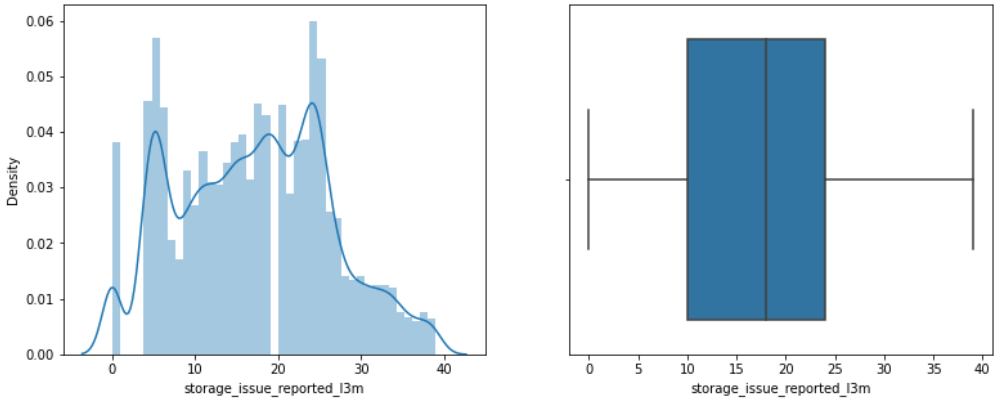


Fig.12

- **storage_issue_reported_l3m** is fairly symmetric, std =9.16 , mean = 17.13.
- SKEWNESS = 0.11 and Kurtosis = -0.68
- There are no outliers present.
- Data is spread from 0 to 39 value.

temp_reg_mach -

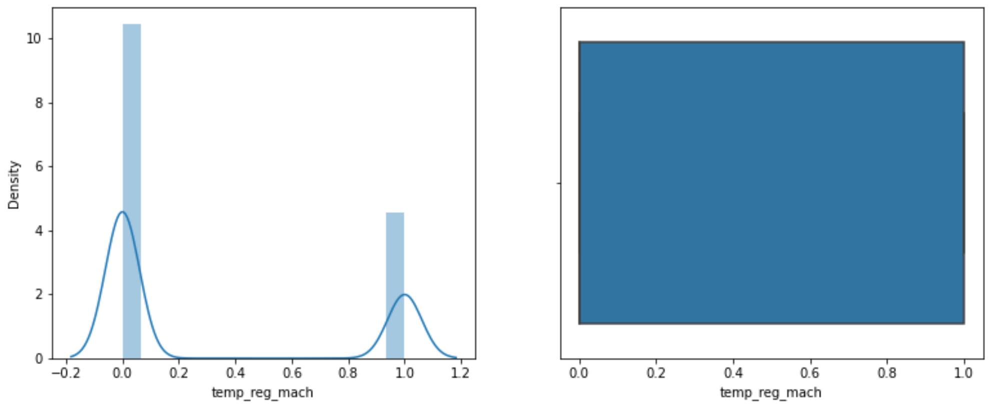


Fig.13

- **temp_reg_mach** is moderately positive skewed, std = 0.45, mean = 0.30.
- SKEWNESS = 0.85 and Kurtosis = -1.26
- No need to look for outliers for this variable as it has only 2 values.
- 0 and 1 value only present.

wh_breakdown_l3m -

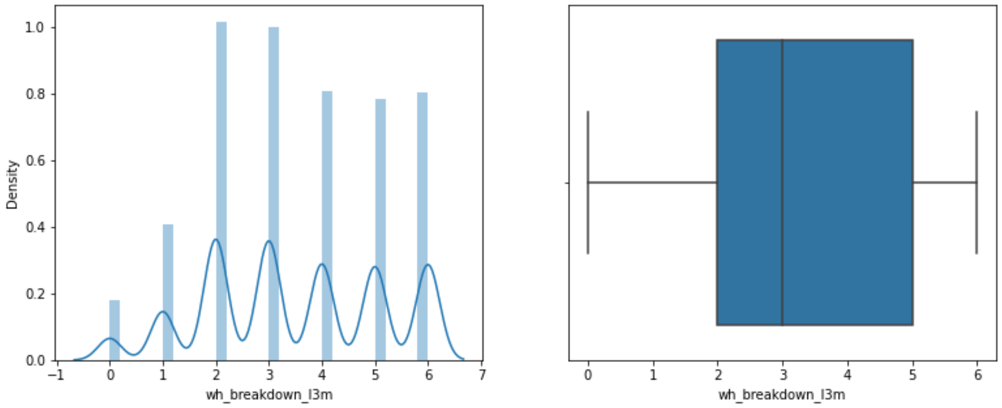


Fig.14

- **wh_breakdown_l3m** is fairly symmetric, std = 1.69, mean = 3.48.
- SKEWNESS = -0.06 and Kurtosis = -0.95
- There are no outliers present.
- Data is spread from 0 to 6 value.

govt_check_l3m -

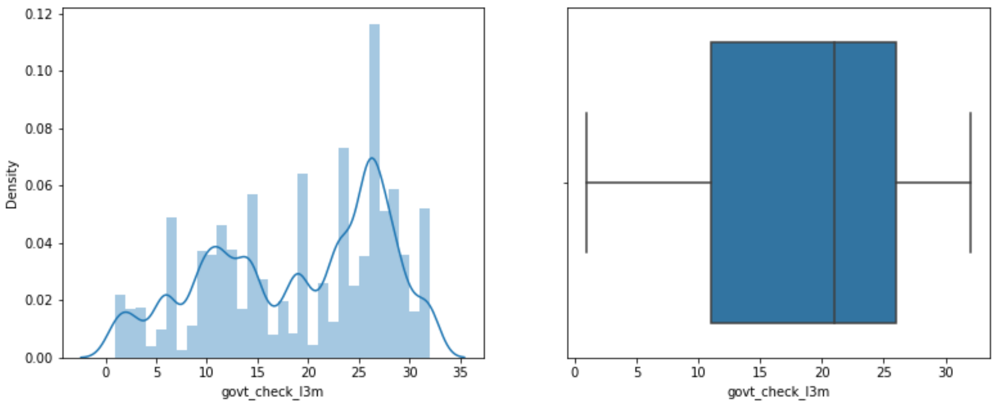


Fig.15

- **govt_check_l3m** is fairly symmetric, std = 8.63, mean = 18.81.
- SKEWNESS = -0.36 and Kurtosis = -1.05
- There are no outliers present.
- Data is spread from 1 to 32 value.

product_wg_ton – Target Variable

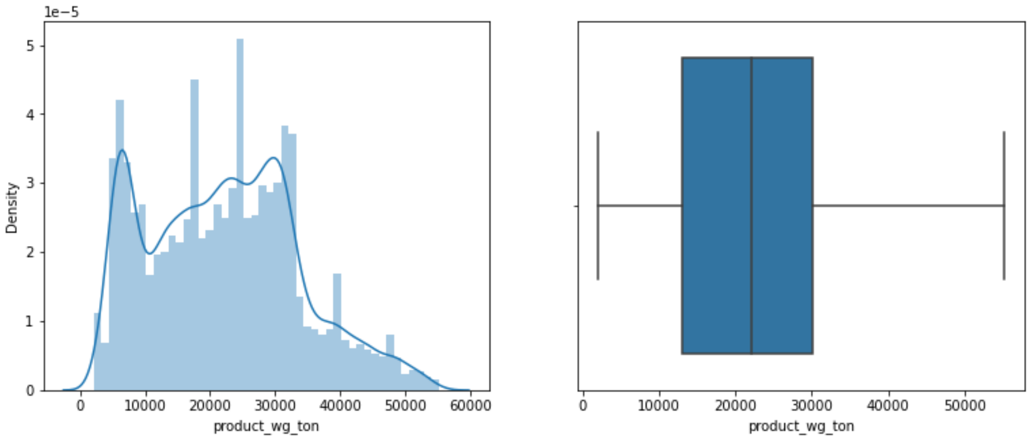
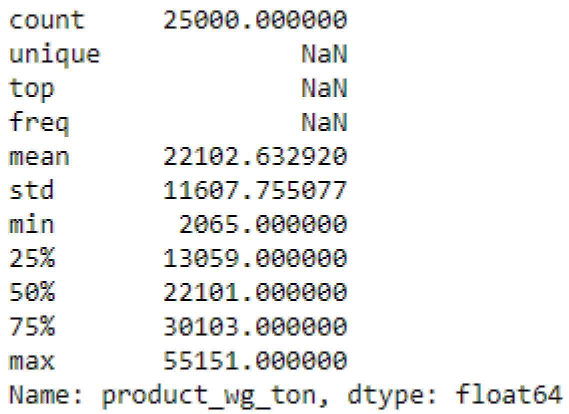


Fig.16

- Our target variable is continuous in nature so regression models needs to be performed.
- **product_wg_ton** is fairly symmetric, std = 11607.75, mean = 22102.63.
- SKEWNESS = 0.33 and Kurtosis = -0.50
- There are no outliers present.
- Data is spread from 2065 to 55151 value.

- Descriptive stats of target variable

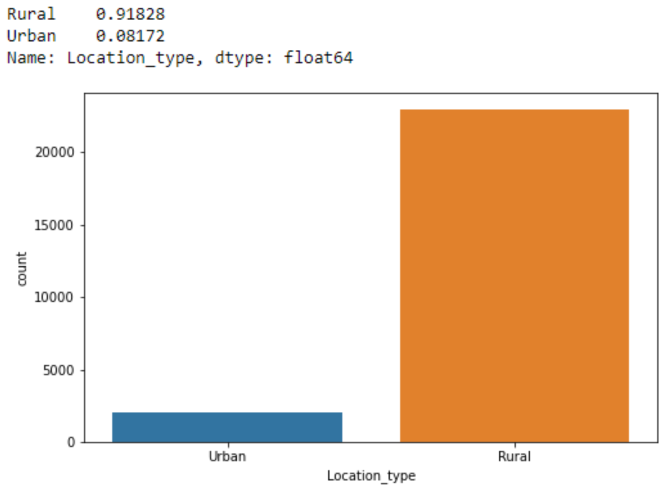


```
count     25000.000000
unique             NaN
top                NaN
freq               NaN
mean      22102.632920
std       11607.755077
min        2065.000000
25%       13059.000000
50%       22101.000000
75%       30103.000000
max       55151.000000
Name: product_wg_ton, dtype: float64
```

Table.5

Univariate Analysis – Categorical Variables

Location_type -

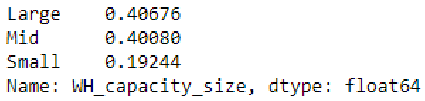


```
Rural     0.91828
Urban     0.08172
Name: Location_type, dtype: float64
```

Fig.17

- Most warehouses are located in rural region accounting to 91% of the total number of warehouses.

WH_capacity_size -



```
Large    0.40676
Mid      0.40080
Small    0.19244
Name: WH_capacity_size, dtype: float64
```

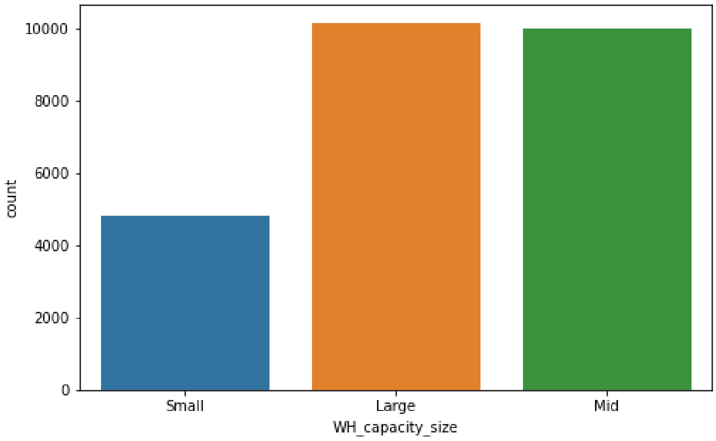
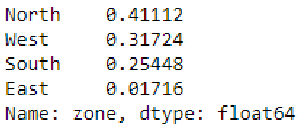


Fig.18

- There are 3 categories of warehouse namely small, medium and large given in the dataset.
- Most warehouses are of large and medium type.
- Size of warehouses in terms of weight is not provided.

Zone -



```
North    0.41112
West     0.31724
South    0.25448
East     0.01716
Name: zone, dtype: float64
```

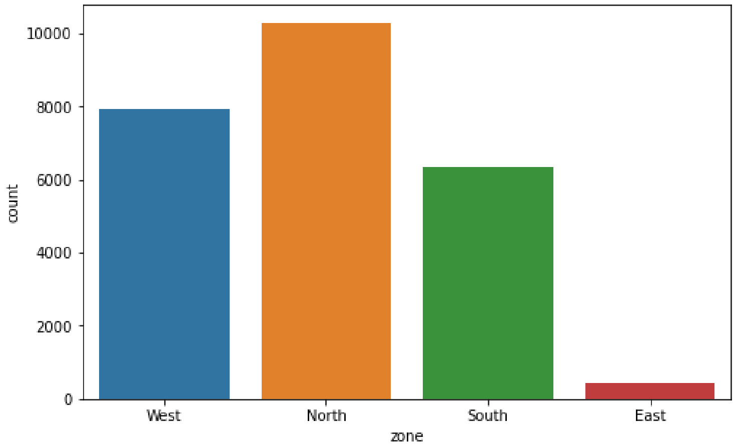
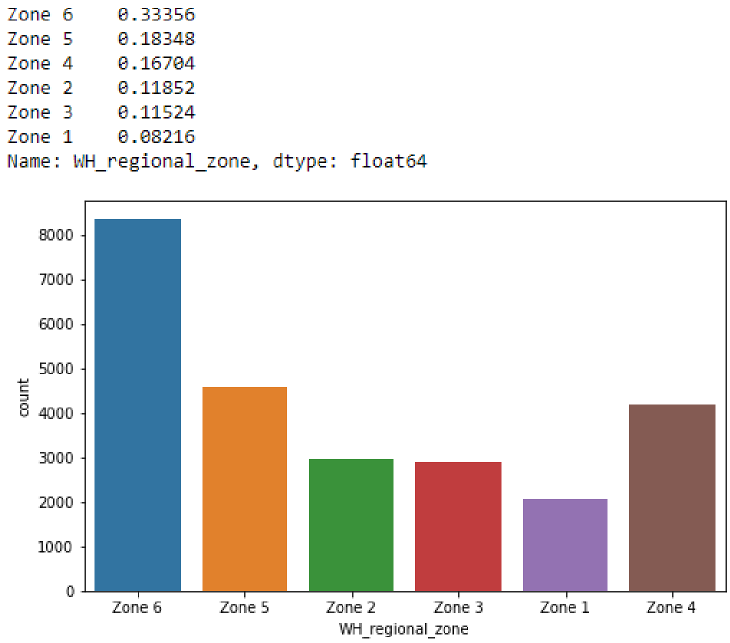


Fig.19

- There are 4 zones where the warehouses are located namely west, north, south and east.
- East zone has lowest warehouses and north zone has most number of warehouses.

WH_regional_zone -

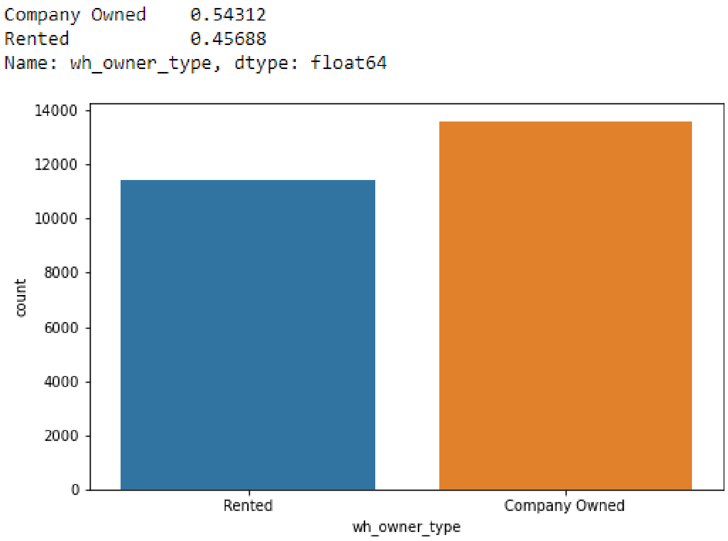


```
Zone 6    0.33356
Zone 5    0.18348
Zone 4    0.16704
Zone 2    0.11852
Zone 3    0.11524
Zone 1    0.08216
Name: WH_regional_zone, dtype: float64
```

Fig.20

- There are 6 regional zone in our dataset.
- Zone 6 has highest number of warehouses and Zone 1 has lowest number of warehouse present in them.

Wh_owner_type -

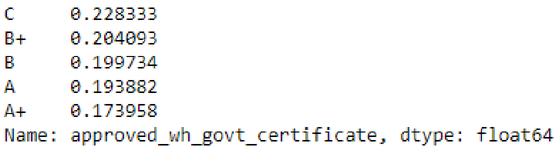


```
Company Owned    0.54312
Rented           0.45688
Name: wh_owner_type, dtype: float64
```

Fig.21

- There are more number of company owned warehouse.
- There is only 9% difference in terms of rented and company owned warehouses.

approved_wh_govt_certificate -



```
C      0.228333
B+     0.204093
B      0.199734
A      0.193882
A+     0.173958
Name: approved_wh_govt_certificate, dtype: float64
```

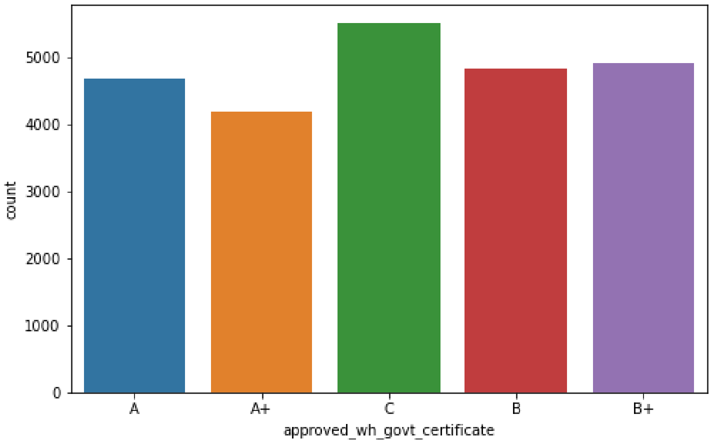


Fig.22

- There are 5 category of ranking given to the warehouses.
- Distribution of ranking of the warehouses are shown as per above graph.
- Most warehouses are given ‘C ‘certificate.

b) Bivariate analysis (relationship between different variables, correlations)

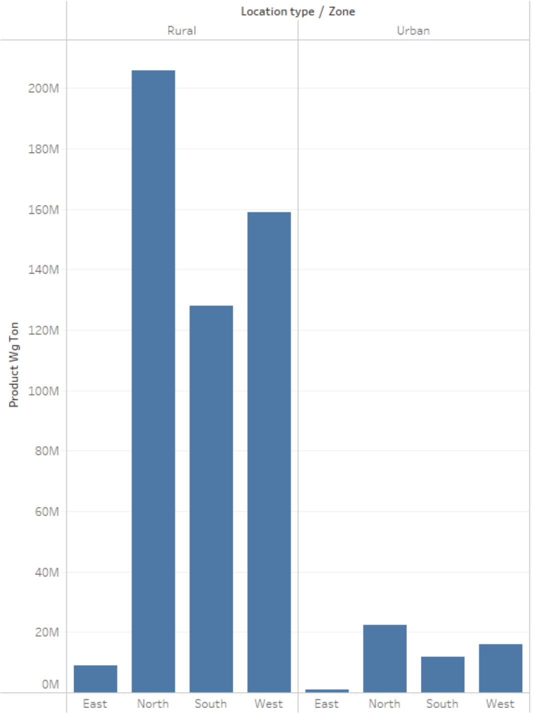


Fig.23

- We can infer from the graph that the product by weight in north zone of rural region is highest.
- Overall also more product is available in rural region.

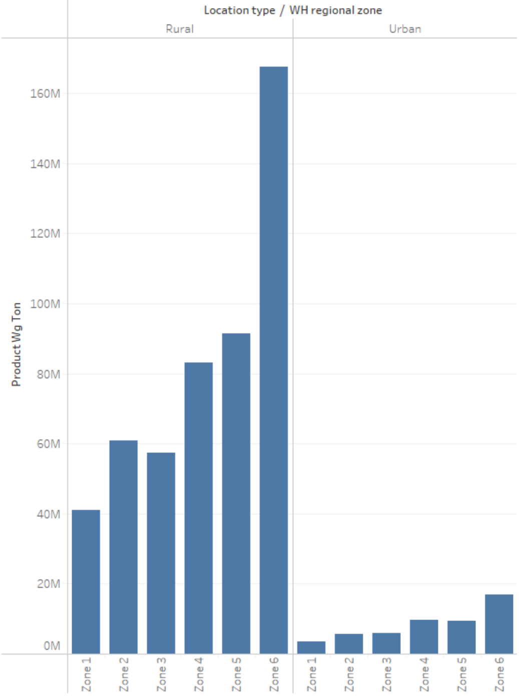


Fig.24

- If look by regional zone then zone 6 has highest product by weight and zone 1 in urban region has lowest.

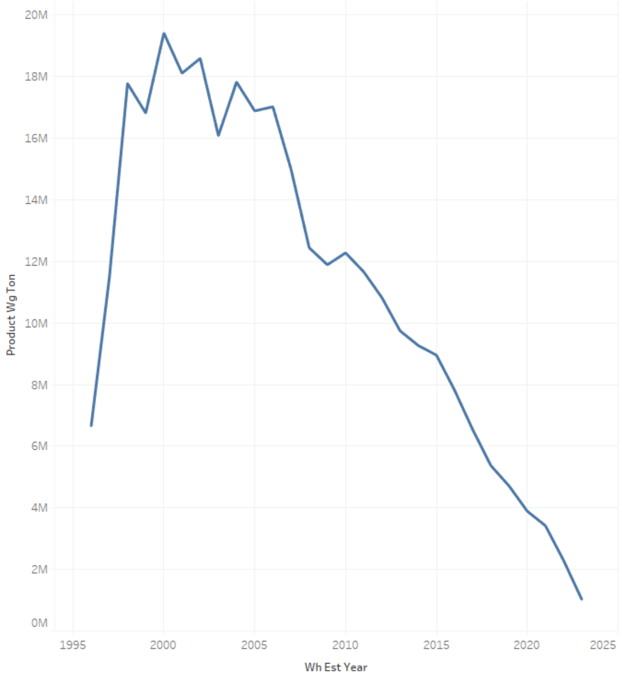


Fig.25

- We can infer that warehouse that are old have more capacity to handle product in higher quantity than the new ones.

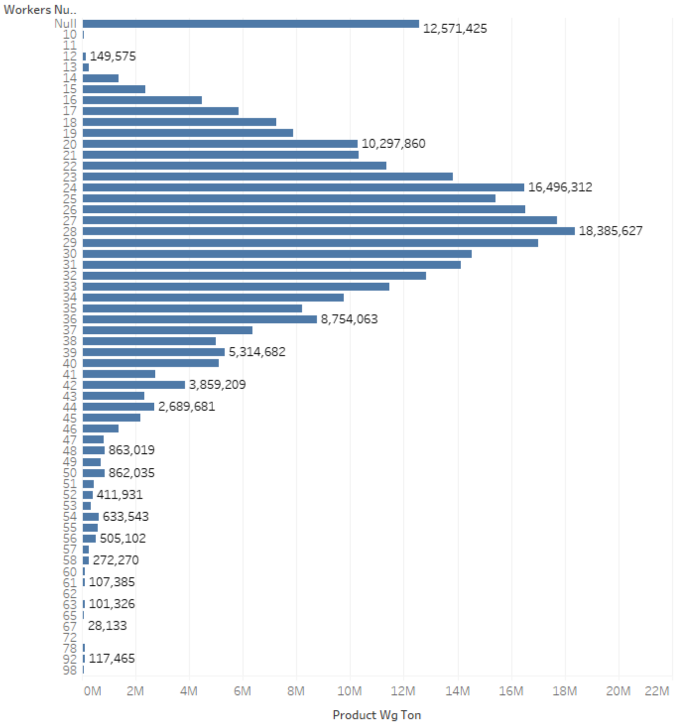


Fig.26

- Since there are missing value sin this column we are seeing null values also.
- From the graph we can infer that lot of warehouses have high number of workers and are handling very low to quantity of product. Warehouses with workers above 40 are handling very low weights of product.

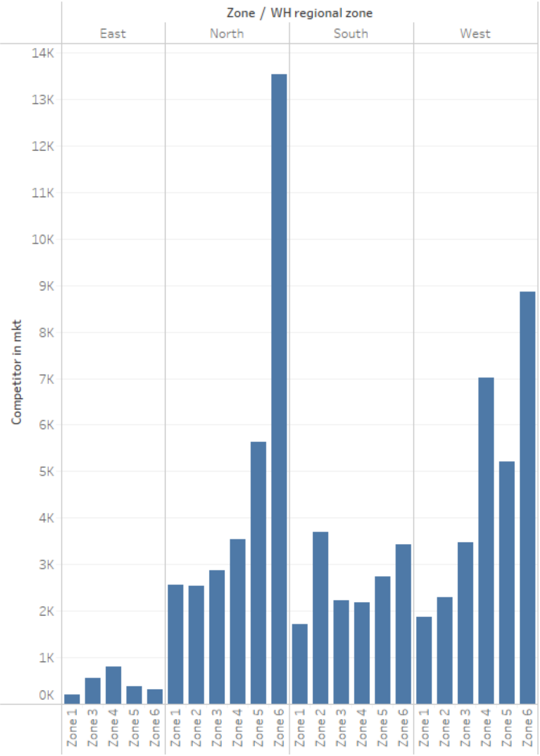


Fig.27

- We can infer that zone 6 of north region has highest number of competitors.

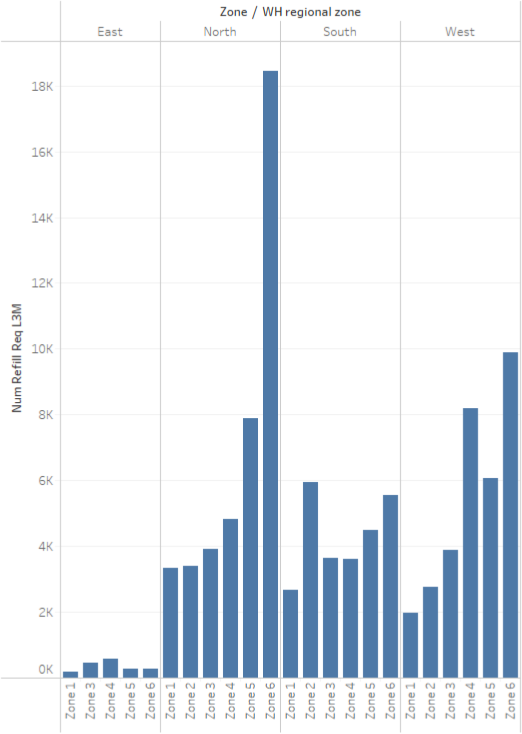


Fig.28

- We can infer that zone 6 of north region has highest number of refills also.

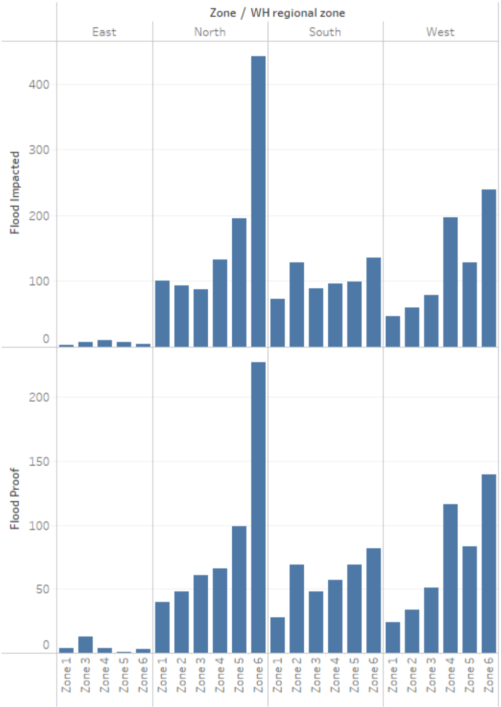


Fig.29

- We can infer that north region is highly prone to flood.

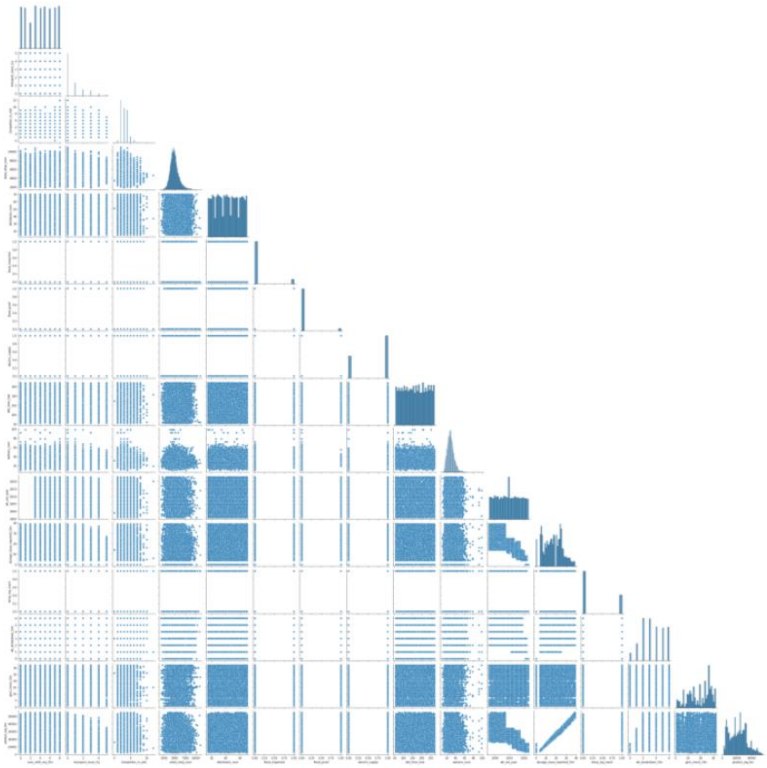


Fig.30

- We can infer that only a few columns are showing some type of correlation between them.

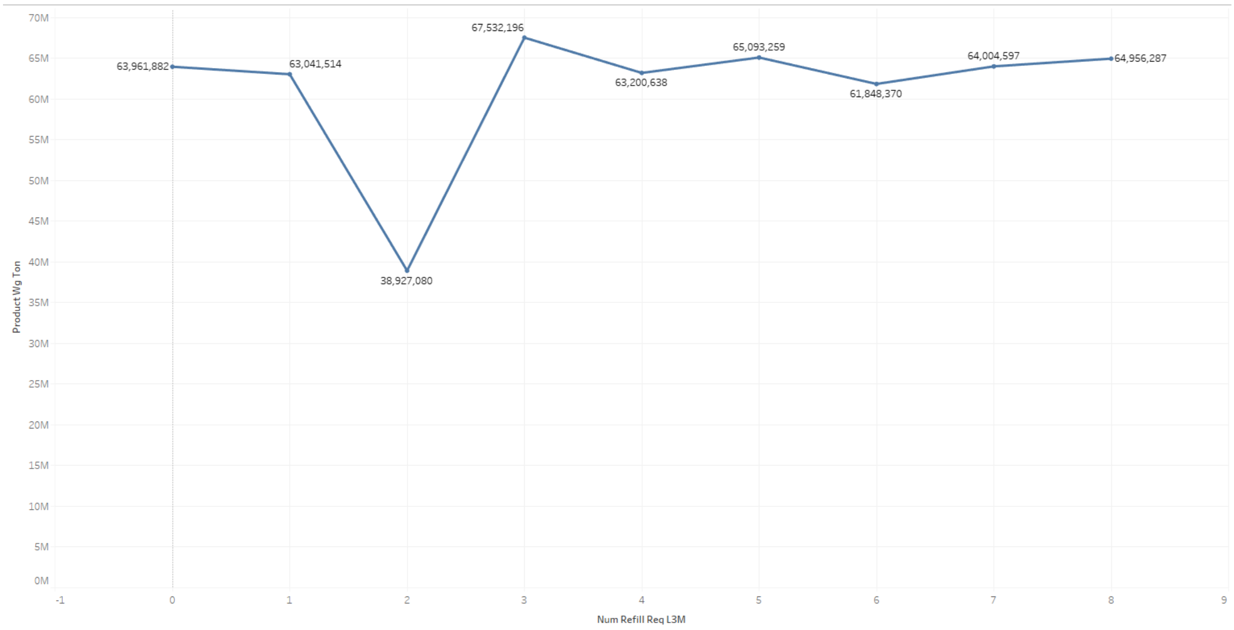


Fig.31

- W can infer that warehouses having only 2 refills are handling least product.

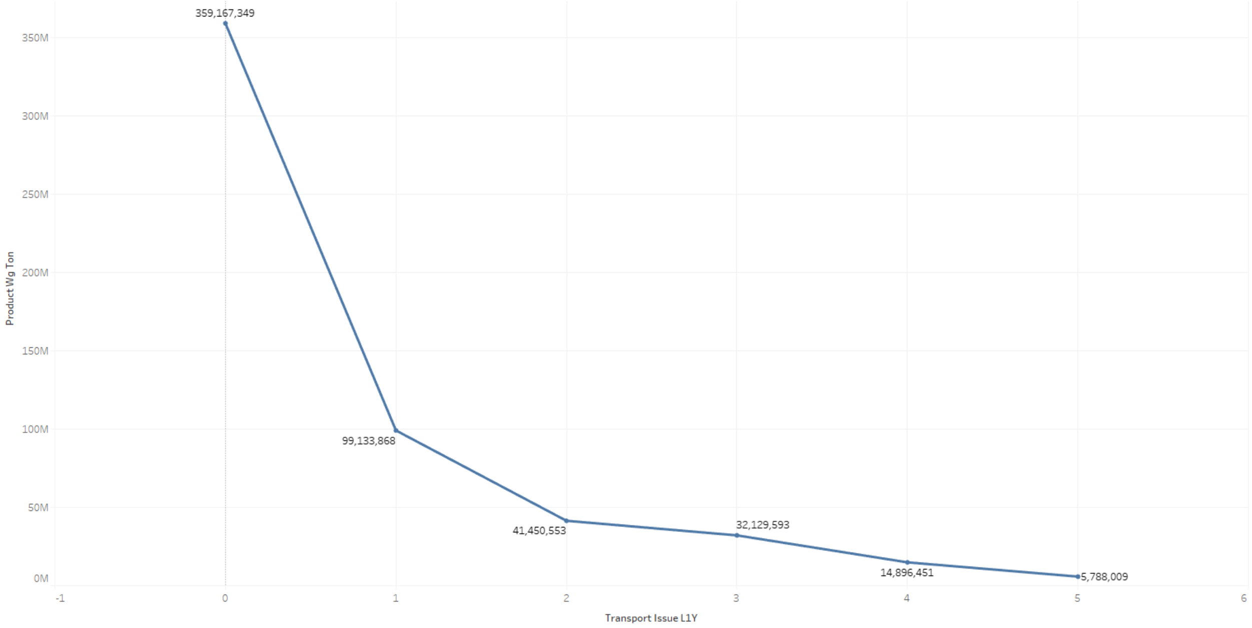


Fig.32

- We can infer that if transport issue are reported even once by a warehouse the product available with them decreases significantly.

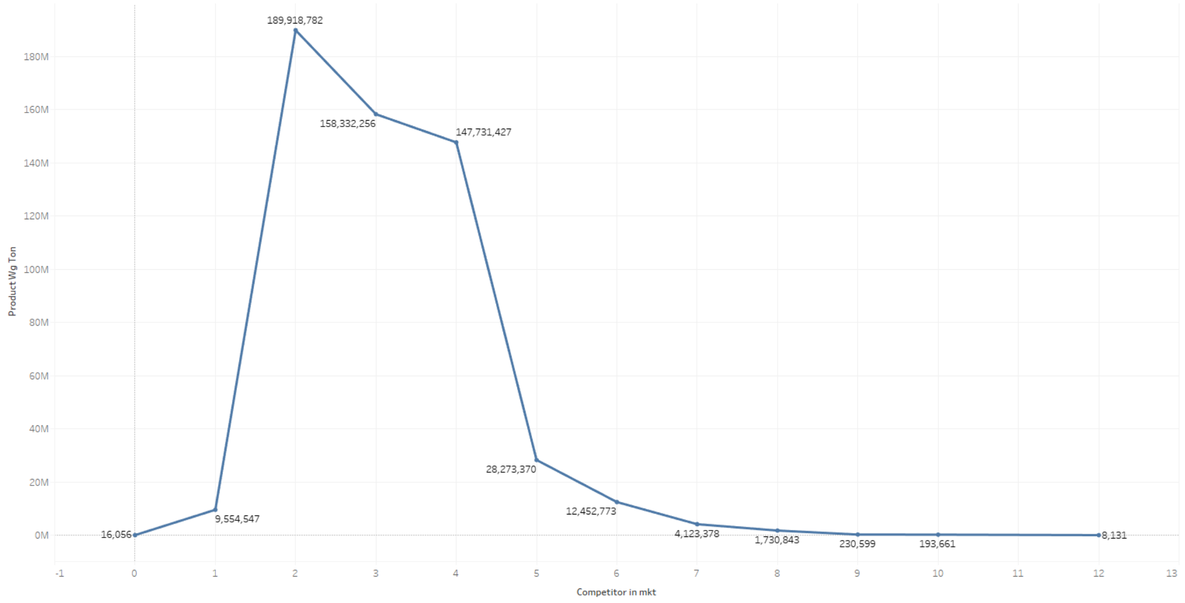


Fig.33

- We can infer that product weight is decreasing in warehouses where there are more than 2 competitors in market.

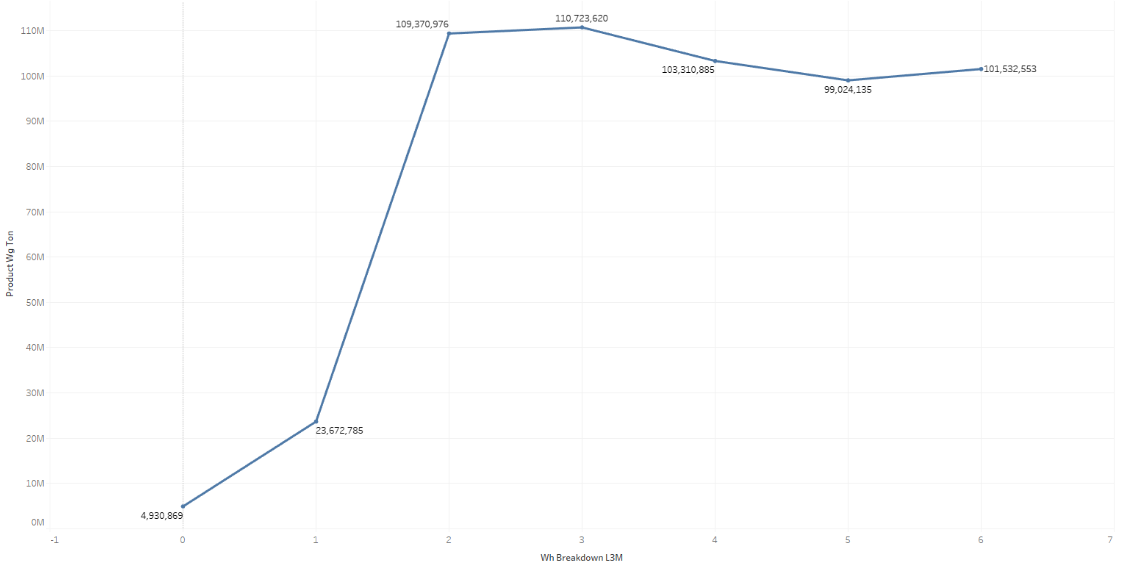


Fig.34

- We can infer that having breakdowns in warehouse have no significant impact on the product quantity it can handle.

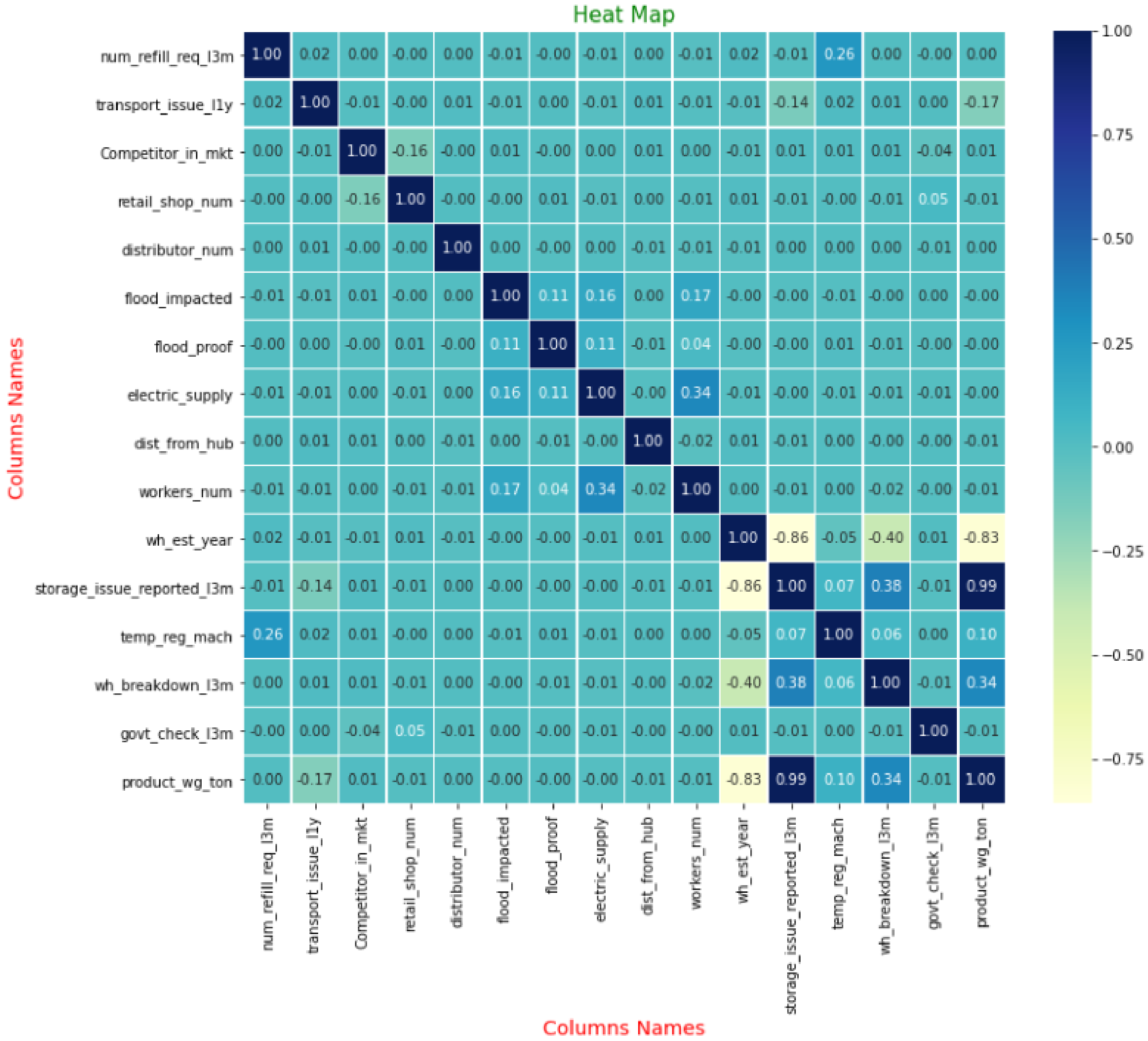


Fig.35

- Columns that are showing significant correlation are as flows:
  - Product_wg_ton and storage_issue_reported_l3m showing positive correlation
  - Product_wg_ton and wh_est_year showing negative correlation
  - wh_est_year and storage_issue_reported_l3m showing negative correlation

Business Insight from data Clusters

Is the data unbalanced? If so, what can be done? Please explain in the context of the business

- From target level perspective our variable is in continuous form so this is a regression type problem. So we do not need to check of data imbalance for a continuous target variable here.

Any business insights using clustering (if applicable)

- We have performed K-means clustering here on scaled data to identify well separated clusters within our dataset.
- By checking the elbow curve we can identify the optimal number of clusters, but since the elbow curve is smooth we are using silhouette sample and score values to identify the optimal number of clusters. The moment any of the values turns negative for a specific number of clusters we stop at the cluster count.

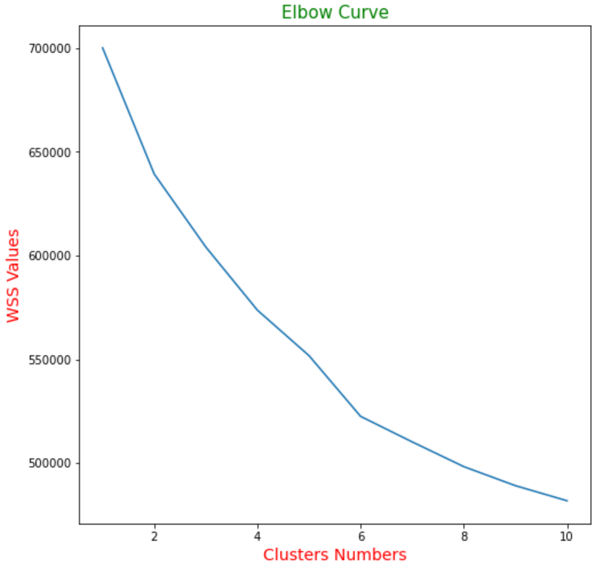
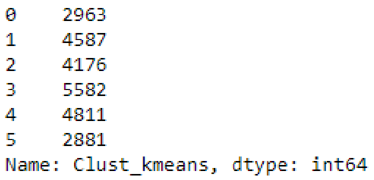


Fig.36

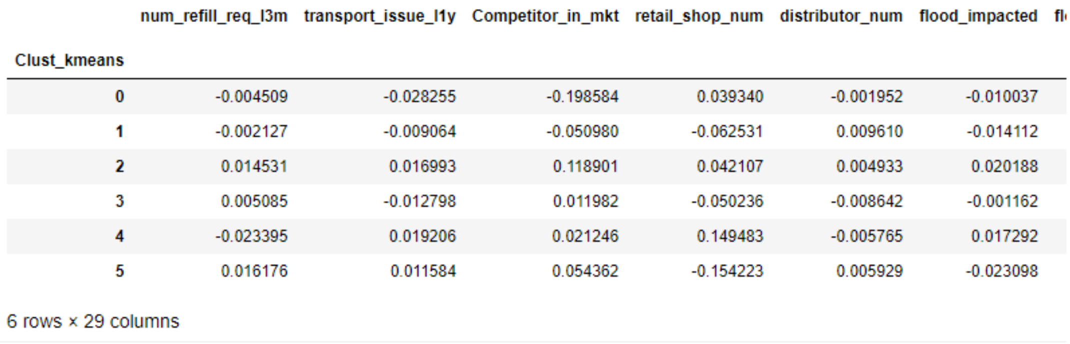
- We see that starting from cluster value of 7 the silhouette sample values turns to negative so we take 6 as our optimal cluster count.
- The counts of records in each cluster is as below:



```
0    2963
1    4587
2    4176
3    5582
4    4811
5    2881
Name: Clust_kmeans, dtype: int64
```

Table.6

Any other business insights



	num_refill_req_l3m	transport_issue_l1y	Competitor_in_mkt	retail_shop_num	distributor_num	flood_impacted	fl
Clust_kmeans							
0	-0.004509	-0.028255	-0.198584	0.039340	-0.001952	-0.010037	
1	-0.002127	-0.009064	-0.050980	-0.062531	0.009610	-0.014112	
2	0.014531	0.016993	0.118901	0.042107	0.004933	0.020188	
3	0.005085	-0.012798	0.011982	-0.050236	-0.008642	-0.001162	
4	-0.023395	0.019206	0.021246	0.149483	-0.005765	0.017292	
5	0.016176	0.011584	0.054362	-0.154223	0.005929	-0.023098	

6 rows × 29 columns

Table.7

By analysing the cluster profile we can make a few inferences as below:

- Warehouses in clusters 0,1,4 have low number of refill counts than cluster 2,3,5.
- Warehouses in clusters 0,1 have few competitors in market compared to other clusters.
- Warehouses in clusters 0,2 have low worker count compared to rest clusters.
- Warehouses in clusters 1,2 are further away than rest clusters.
- Warehouses in clusters 1,2,5 have more number of distributers than rest clusters.

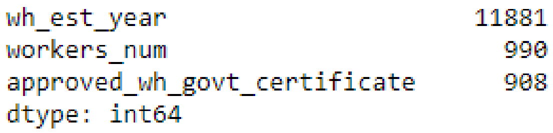
Data Cleaning and Pre-processing

Removal of unwanted variables

- Since wh_est_year has more than 47.5% of the data missing in it, I have decided to drop this column.
- 2 columns Ware_house_Id and WH_Manager_ID were already removed from the start as they are just codes to ware houses and have no predicting power for our target variable.

Missing Value treatment

- 3 columns contain missing value we have already dropped wh_est_year column altogether.
- For the rest 2 columns 'workers_num' and 'approved_wh_gov_certificate' we are using KNN imputer to fill in the missing values.
- There are no duplicates present in the dataset.



```
wh_est_year                    11881
workers_num                      990
approved_wh_govt_certificate     908
dtype: int64
```

Table.8

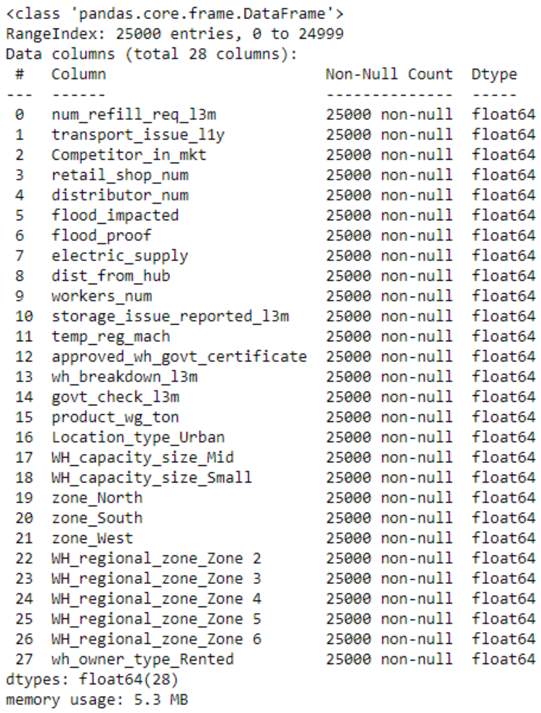
Outlier treatment

- So most of the columns do not have outliers and some do have but they represent categorical values so we can ignore them.
- Out of all the continuous columns only "retail_shop_num" and "worker_num" require outlier treatment.
- I have treated them separately leaving the rest columns as it is.

Variable transformation

- Variable transformation for all the categorical columns needs to be performed, so that we can fit the data to our model. The transformation required for the various columns is mentioned below :
  - ➢ Location_type – One hot encoding performed
  - ➢ WH_regional_zone - One hot encoding performed
  - ➢ Zone - One hot encoding performed
  - ➢ WH_regional_zone - One hot encoding performed
  - ➢ Wh_owner_type - One hot encoding performed
  - ➢ Approved_wh_govt_certificate – Label encoding performed (as per column value ranking)

- Dataset after performing variable transformation, outlier treatment and missing value treatment using KNN imputer is now ready for model building. Below is the info of our pre-processed data:



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25000 entries, 0 to 24999
Data columns (total 28 columns):
 #   Column                        Non-Null Count  Dtype  
---  ------                        --------------  -----  
 0   num_refill_req_l3m            25000 non-null  float64
 1   transport_issue_l1y           25000 non-null  float64
 2   Competitor_in_mkt             25000 non-null  float64
 3   retail_shop_num               25000 non-null  float64
 4   distributor_num               25000 non-null  float64
 5   flood_impacted                25000 non-null  float64
 6   flood_proof                   25000 non-null  float64
 7   electric_supply               25000 non-null  float64
 8   dist_from_hub                 25000 non-null  float64
 9   workers_num                   25000 non-null  float64
 10  storage_issue_reported_l3m    25000 non-null  float64
 11  temp_reg_mach                 25000 non-null  float64
 12  approved_wh_govt_certificate  25000 non-null  float64
 13  wh_breakdown_l3m              25000 non-null  float64
 14  govt_check_l3m                25000 non-null  float64
 15  product_wg_ton                25000 non-null  float64
 16  Location_type_Urban           25000 non-null  float64
 17  WH_capacity_size_Mid          25000 non-null  float64
 18  WH_capacity_size_Small        25000 non-null  float64
 19  zone_North                    25000 non-null  float64
 20  zone_South                    25000 non-null  float64
 21  zone_West                     25000 non-null  float64
 22  WH_regional_zone_Zone 2       25000 non-null  float64
 23  WH_regional_zone_Zone 3       25000 non-null  float64
 24  WH_regional_zone_Zone 4       25000 non-null  float64
 25  WH_regional_zone_Zone 5       25000 non-null  float64
 26  WH_regional_zone_Zone 6       25000 non-null  float64
 27  wh_owner_type_Rented          25000 non-null  float64
dtypes: float64(28)
memory usage: 5.3 MB
```

Table.9

Addition of new variables (Feature Engineering)

- One feature engineering only seems feasible that can be done separately.
- From **wh_est_year** we can find out the year of each warehouse. This variable can help find age of each warehouse.
- But for building our model we have already dropped this variable as it has too many data points missing.

Checking Log transformation of some Variable that are not normally distributed

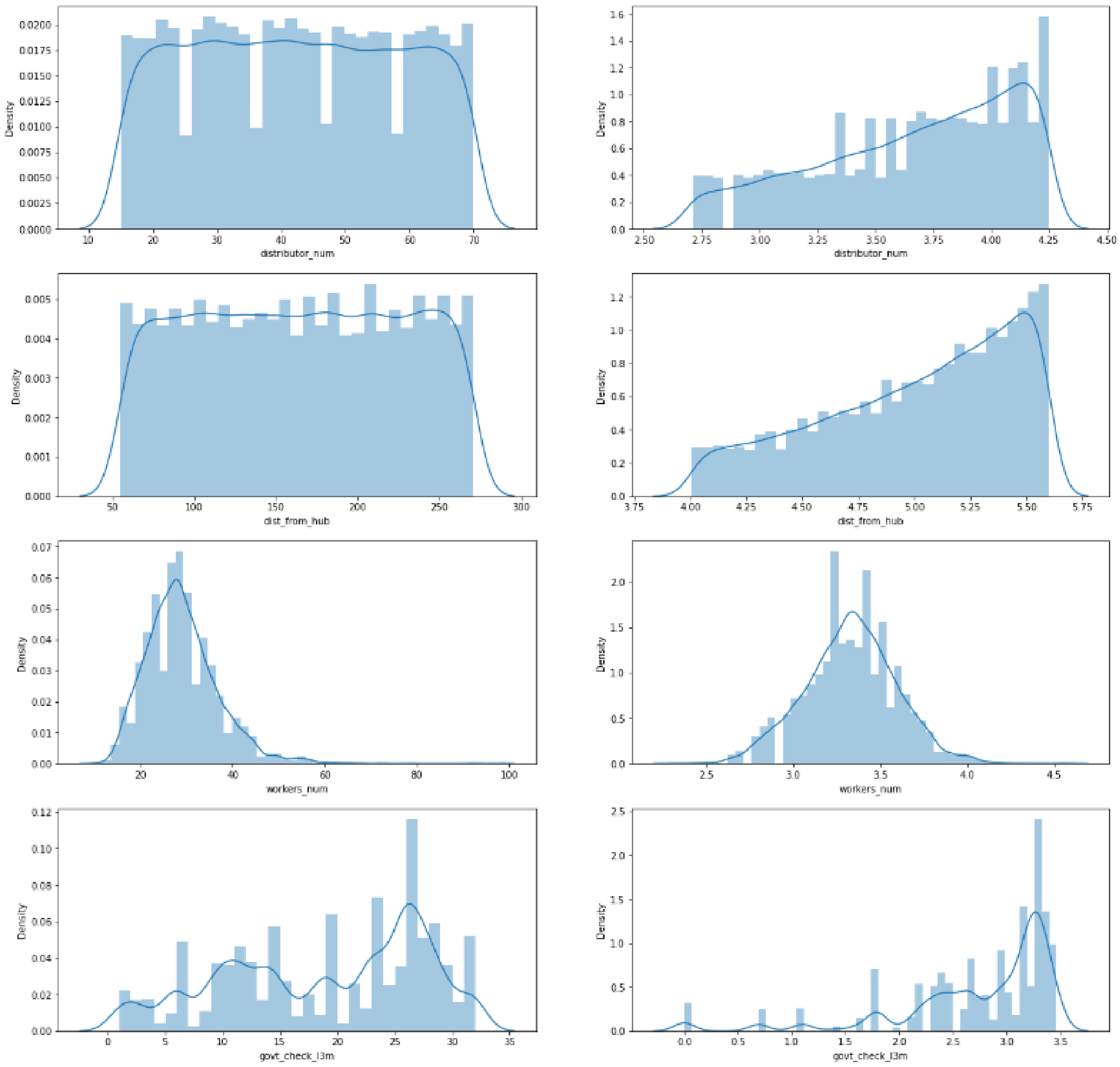


Fig.37

- We can infer that log transformation results in making the variable 'worker_num' normally distributed and for the rest variables the data is not that normally distributed.
- We here just checking the effect of log transformation on the variables but for model building we have not used any log transformation of variables.

Model building and interpretation.

Evaluation Metrics that we have used:

Machine learning model cannot have 100 per cent efficiency otherwise the model is known as a biased model, which further includes the concept of overfitting and under fitting.

It is necessary to obtain the accuracy on training data, but it is also important to get a genuine and approximate result on unseen data otherwise Model is of no use.

So to build and deploy a generalized model we require to evaluate the model on different metrics which helps us to better optimize the performance, fine-tune it, and obtain a better result.

If one metric is perfect, there is no need for multiple metrics. To understand the benefits and disadvantages of Evaluation metrics because different evaluation metric fits on a different set of a dataset.

1. Mean Squared Error (MSE)

MSE is a most used and very simple metric with a little bit of change in mean absolute error. Mean squared error states that finding the squared difference between actual and predicted value.
So, above we are finding the absolute difference and here we are finding the squared difference.

What actually the MSE represents? It represents the squared distance between actual and predicted values. we perform squared to avoid the cancellation of negative terms and it is the benefit of MSE.

$$MSE = \frac{1}{n} \Sigma \underbrace{\left(y - \widehat{y} \right)^2}_{\text{The square of the difference between actual and predicted}}$$

Fig.38

<u>Advantages of MSE:</u>

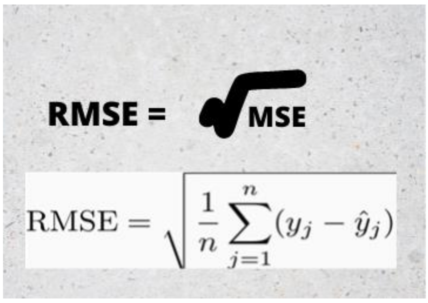
- The graph of MSE is differentiable, so you can easily use it as a loss function.
- Value lies between 0 to infinity.
- Small value indicates better model

<u>Disadvantages of MSE:</u>

- The value you get after calculating MSE is a squared unit of output. For example, the output variable is in meter (m) then after calculating MSE the output we get is in meter squared.
- If you have outliers in the dataset then it penalizes the outliers most and the calculated MSE is bigger.

2. Root Mean Squared Error (RMSE)

As RMSE is clear by the name itself, that it is a simple square root of mean squared error.



$$\text{RMSE} = \sqrt{\text{MSE}}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)}$$

Fig.39

Advantages of RMSE:

- The output value you get is in the same unit as the required output variable which makes interpretation of loss easy.
- Value lies between 0 to infinity.
- Small value indicates better model

Disadvantages of RMSE:

- It is not that robust to outliers as compared to MAE (Mean Absolute Error).

3. R Squared (R2)

R2 score is a metric that tells the performance of your model, not the loss in an absolute sense that how many wells did your model perform.

In contrast, MAE and MSE depend on the context as we have seen whereas the R2 score is independent of context.

So, with help of R squared we have a baseline model to compare a model which none of the other metrics provides. The same we have in classification problems which we call a threshold which is fixed at 0.5. So basically R2 squared calculates how must regression line is better than a mean line.

Hence, R2 squared is also known as Coefficient of Determination or sometimes also known as Goodness of fit.

$$\textbf{R2 Squared} = 1 - \frac{\text{SSr}}{\text{SSm}}$$

SSr = Squared sum error of regression line

SSm = Squared sum error of mean line

Fig.40

Now, how will you interpret the R2 score? Suppose If the R2 score is zero then the above regression line by mean line is equal means 1 so 1-1 is zero. So, in this case, both lines are overlapping means model performance is worst, It is not capable to take advantage of the output column.

Now the second case is when the R2 score is 1, it means when the division term is zero and it will happen when the regression line does not make any mistake, it is perfect. In the real world, it is not possible.

So we can conclude that as our regression line moves towards perfection, R2 score move towards one. And the model performance improves.

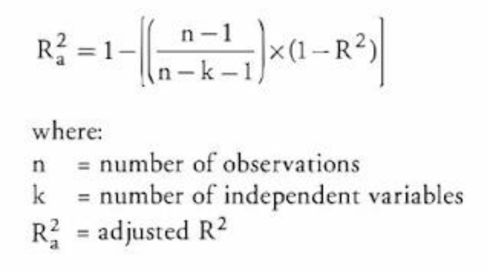
The normal case is when the R2 score is between zero and one like 0.8 which means your model is capable to explain 80 per cent of the variance of data.

4. Adjusted R Squared

The disadvantage of the R2 score is while adding new features in data the R2 score starts increasing or remains constant but it never decreases because it assumes that while adding more data variance of data increases.

But the problem is when we add an irrelevant feature in the dataset then at that time R2 sometimes starts increasing which is incorrect.

Hence, to control this situation Adjusted R Squared came into existence.



$$R_a^2 = 1 - \left[\left(\frac{n-1}{n-k-1}\right) \times (1-R^2)\right]$$

where:
n = number of observations
k = number of independent variables
R_a^2 = adjusted R^2

Fig.41

Now as K increases by adding some features so the denominator will decrease, n-1 will remain constant. R2 score will remain constant or will increase slightly so the complete answer will increase and when we subtract this from one then the resultant score will decrease. So this is the case when we add an irrelevant feature in the dataset.

And if we add a relevant feature then the R2 score will increase and (1-R2) will decrease heavily and the denominator will also decrease so the complete term decreases, and on subtracting from one the score increases.

5. Mean Absolute Percentage Error

The **mean absolute percentage error** (MAPE) is the percentage equivalent of MAE. The equation looks just like that of MAE, but with adjustments to convert everything into percentages.

$$MAPE = \frac{\overset{\text{Multiplying by 100\% converts to percentage}}{\boxed{100\%}}}{n} \Sigma \left| \frac{\overbrace{y - \widehat{y}}^{\text{The residual}}}{\underset{\text{Each residual is scaled against the actual value}}{\boxed{y}}} \right|$$

Fig.42

Just as MAE is the average magnitude of error produced by your model, the MAPE is how far the model's predictions are off from their corresponding outputs on average. Like MAE, MAPE also has a clear interpretation since percentages are easier for people to conceptualize. Both MAPE and MAE are robust to the effects of outliers thanks to the use of absolute value.

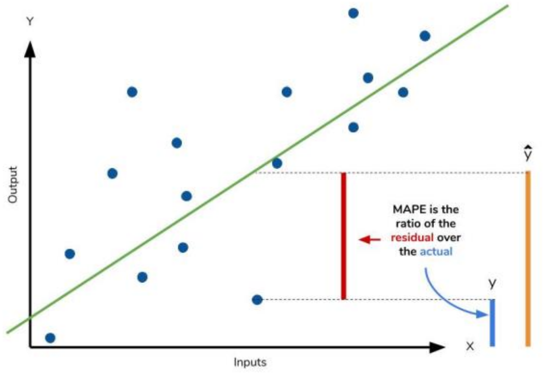


Fig.43

However for all of its advantages, we are more limited in using MAPE than we are MAE. Many of MAPE's weaknesses actually stem from use division operation. Now that we have to scale everything by the actual value, MAPE is undefined for data points where the value is 0. Similarly, the MAPE can grow unexpectedly large if the actual values are exceptionally small themselves. Finally, the MAPE is biased towards predictions that are systematically less than the actual values themselves. That is to say, MAPE will be lower when the prediction is lower than the actual compared to a prediction that is higher by the same amount. The quick calculation below demonstrates this point.

$$MAPE = \frac{100\%}{n} \Sigma \left| \frac{y - \widehat{y}}{y} \right|$$

$\hat{y}$ is smaller than the actual value	$\hat{y}$ is greater than the actual value
n = 1 $\hat{y}$ = 10 y = 20	n = 1 $\hat{y}$ = 20 y = 10
MAPE = 50%	MAPE = 100%

Fig.44

The lower the value for MAPE, the better the machine learning model is at predicting values. Inversely, the higher the value for MAPE, the worse the model is at predicting values.

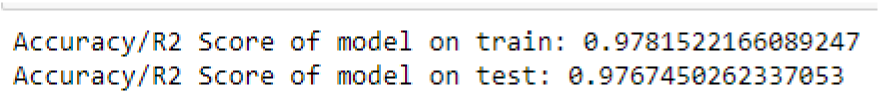
Linear Regression Model

- Of all the handy ML models out there, linear regression is a common, simple predictive analysis algorithm. It is applicable when dependent variable is continuous in nature. This is a supervised machine learning algorithm.
- This algorithm tries to estimate the values of the coefficients of the variables present in the dataset. The coefficients determine the strength of the correlation.
- It attempts to make a model that gives the relationship between two variables by applying a linear equation to observed data.

Assumptions/Condition for Linear Regression:

1. Linearity: The relationship between the independent variable and the mean of the dependent variable is linear.
2. Homoscedasticity: The variance of residual is the same for any value of the independent variable.
3. Independence: Observations are independent of each other.
4. Normality: For any fixed value of the dependent variable, the dependent variable is normally distributed.

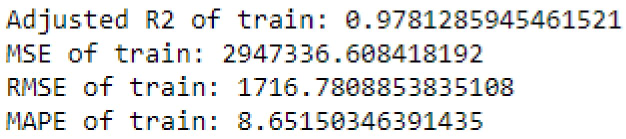
Performance Metrics – Linear Regression Model



```
Accuracy/R2 Score of model on train: 0.9781522166089247
Accuracy/R2 Score of model on test: 0.9767450262337053
```

Table.10

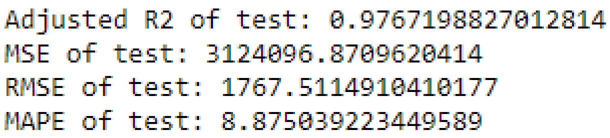
- **Train set**



```
Adjusted R2 of train: 0.9781285945461521
MSE of train: 2947336.608418192
RMSE of train: 1716.7808853835108
MAPE of train: 8.65150346391435
```

Table.11

- **Test set**



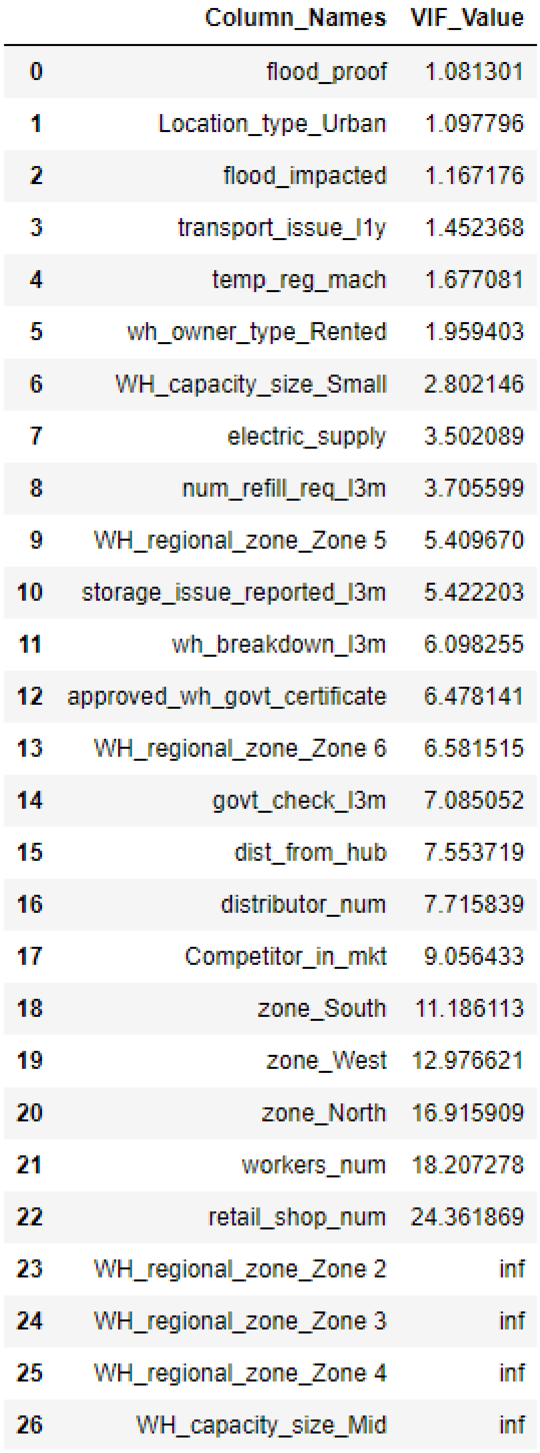
```
Adjusted R2 of test: 0.9767198827012814
MSE of test: 3124096.8709620414
RMSE of test: 1767.5114910410177
MAPE of test: 8.875039223449589
```

Table.12

Inferences:

- Linear Regression model is performing well in both train and test set hence it is a valid model.

Feature Importance of Linear Regression Model:



	Column_Names	VIF_Value
0	flood_proof	1.081301
1	Location_type_Urban	1.097796
2	flood_impacted	1.167176
3	transport_issue_l1y	1.452368
4	temp_reg_mach	1.677081
5	wh_owner_type_Rented	1.959403
6	WH_capacity_size_Small	2.802146
7	electric_supply	3.502089
8	num_refill_req_l3m	3.705599
9	WH_regional_zone_Zone 5	5.409670
10	storage_issue_reported_l3m	5.422203
11	wh_breakdown_l3m	6.098255
12	approved_wh_govt_certificate	6.478141
13	WH_regional_zone_Zone 6	6.581515
14	govt_check_l3m	7.085052
15	dist_from_hub	7.553719
16	distributor_num	7.715839
17	Competitor_in_mkt	9.056433
18	zone_South	11.186113
19	zone_West	12.976621
20	zone_North	16.915909
21	workers_num	18.207278
22	retail_shop_num	24.361869
23	WH_regional_zone_Zone 2	inf
24	WH_regional_zone_Zone 3	inf
25	WH_regional_zone_Zone 4	inf
26	WH_capacity_size_Mid	inf

Table.13

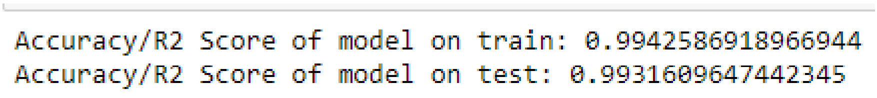
Decision Tree Regression Model

- Decision trees can also be applied to regression problems, using the DecisionTreeRegressor module. As in the classification setting, the fit method will take as argument arrays X and y, only that in this case y is expected to have floating point values instead of integer values.
- We are using grid search for decision tree for hyper parameter tuning the parameter used are listed below. We have also used cross validation to make the predication more accurate.

Hyper parameters used for Grid Search:

1. **criterion** - The function to measure the quality of a split.
2. **max_depth** - The maximum depth of the tree. If None, then nodes are expanded until all leaves are pure or until all leaves contain less than min_samples_split samples.
3. **min_samples_leaf** - The minimum number of samples required to be at a leaf node.
4. **min_samples_split** - The minimum number of samples required to split an internal node.
5. **max_features** - The number of features to consider when looking for the best split.

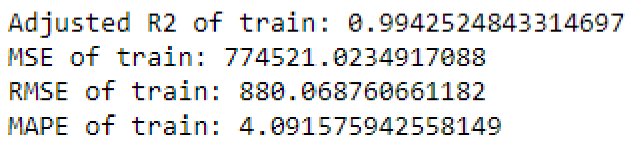
Performance Metrics – Decision Tree Regression Model



```
Accuracy/R2 Score of model on train: 0.9942586918966944
Accuracy/R2 Score of model on test: 0.9931609647442345
```

Table.14

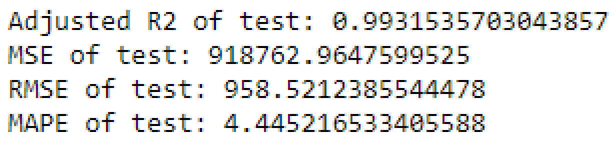
- **Train set**



```
Adjusted R2 of train: 0.9942524843314697
MSE of train: 774521.0234917088
RMSE of train: 880.068760661182
MAPE of train: 4.091575942558149
```

Table.15

- **Test set**



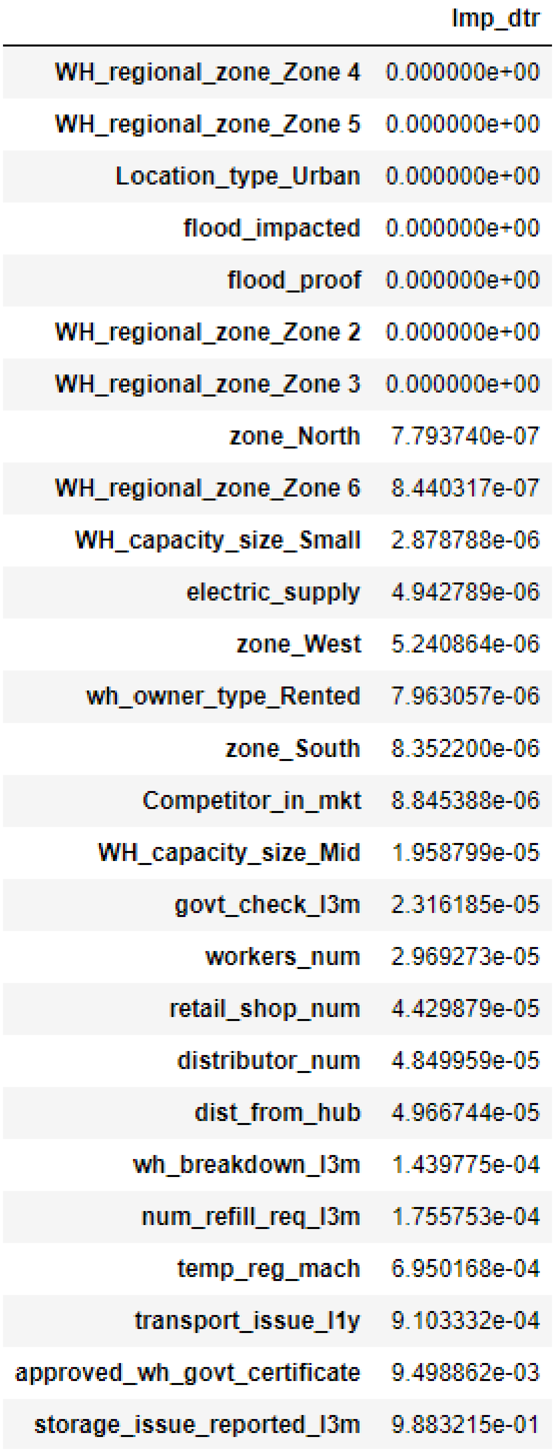
```
Adjusted R2 of test: 0.9931535703043857
MSE of test: 918762.9647599525
RMSE of test: 958.5212385544478
MAPE of test: 4.445216533405588
```

Table.16

Inferences:

- Decision Tree Regression model is performing well in both train and test set hence it is a valid model.

Feature Importance of Decision Tree Regression Model:



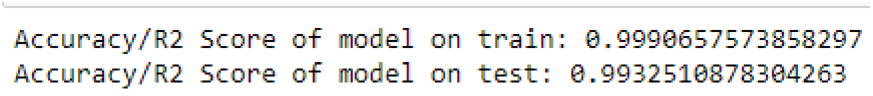
	Imp_dtr
WH_regional_zone_Zone 4	0.000000e+00
WH_regional_zone_Zone 5	0.000000e+00
Location_type_Urban	0.000000e+00
flood_impacted	0.000000e+00
flood_proof	0.000000e+00
WH_regional_zone_Zone 2	0.000000e+00
WH_regional_zone_Zone 3	0.000000e+00
zone_North	7.793740e-07
WH_regional_zone_Zone 6	8.440317e-07
WH_capacity_size_Small	2.878788e-06
electric_supply	4.942789e-06
zone_West	5.240864e-06
wh_owner_type_Rented	7.963057e-06
zone_South	8.352200e-06
Competitor_in_mkt	8.845388e-06
WH_capacity_size_Mid	1.958799e-05
govt_check_l3m	2.316185e-05
workers_num	2.969273e-05
retail_shop_num	4.429879e-05
distributor_num	4.849959e-05
dist_from_hub	4.966744e-05
wh_breakdown_l3m	1.439775e-04
num_refill_req_l3m	1.755753e-04
temp_reg_mach	6.950168e-04
transport_issue_l1y	9.103332e-04
approved_wh_govt_certificate	9.498862e-03
storage_issue_reported_l3m	9.883215e-01

Table.17

Random Forest Regression Model

- A Random Forest is an ensemble technique capable of performing both regression and classification tasks with the use of multiple decision trees and a technique called Bootstrap and Aggregation, commonly known as **bagging**. The basic idea behind this is to combine multiple decision trees in determining the final output rather than relying on individual decision trees.
  Random Forest has multiple decision trees as base learning models. We randomly perform row sampling and feature sampling from the dataset forming sample datasets for every model. This part is called Bootstrap.

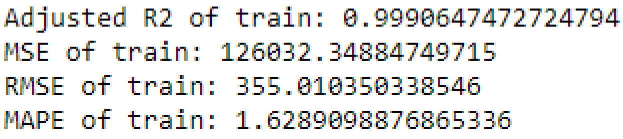
Performance Metrics – Random Forest Regression Model



```
Accuracy/R2 Score of model on train: 0.9990657573858297
Accuracy/R2 Score of model on test: 0.9932510878304263
```

Table.18

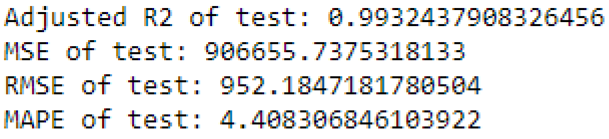
- **Train set**



```
Adjusted R2 of train: 0.9990647472724794
MSE of train: 126032.34884749715
RMSE of train: 355.010350338546
MAPE of train: 1.6289098876865336
```

Table.19

- **Test set**



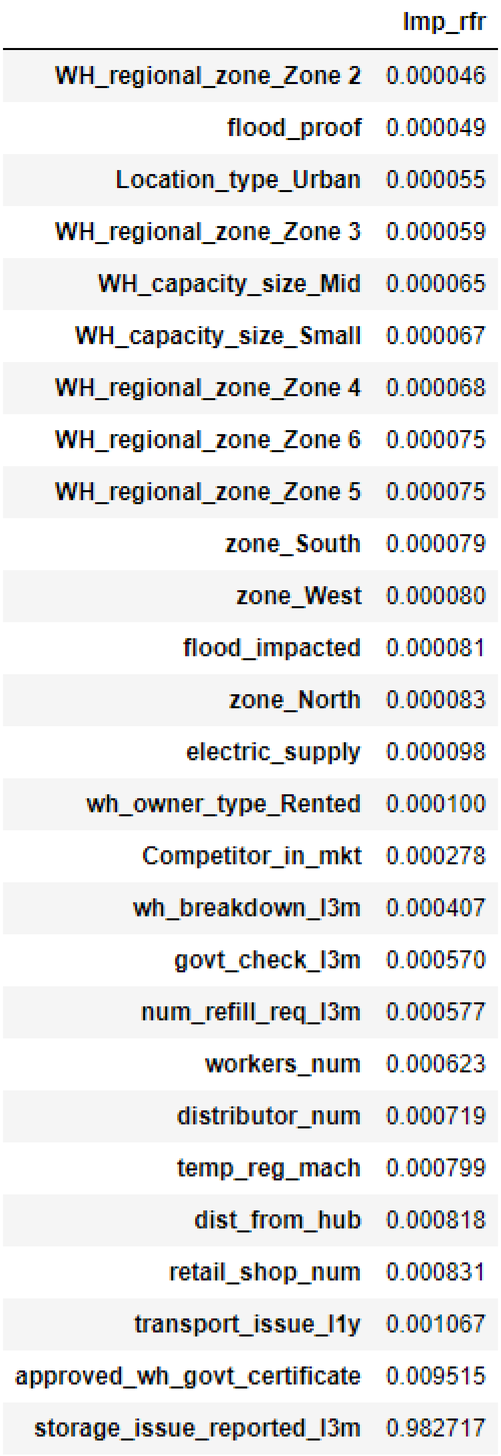
```
Adjusted R2 of test: 0.9932437908326456
MSE of test: 906655.7375318133
RMSE of test: 952.1847181780504
MAPE of test: 4.408306846103922
```

Table.20

Inferences:

- Random Forest Regression model is performing well in both train and test set hence it is a valid model.

Feature Importance of Random Forest Model:



	Imp_rfr
WH_regional_zone_Zone 2	0.000046
flood_proof	0.000049
Location_type_Urban	0.000055
WH_regional_zone_Zone 3	0.000059
WH_capacity_size_Mid	0.000065
WH_capacity_size_Small	0.000067
WH_regional_zone_Zone 4	0.000068
WH_regional_zone_Zone 6	0.000075
WH_regional_zone_Zone 5	0.000075
zone_South	0.000079
zone_West	0.000080
flood_impacted	0.000081
zone_North	0.000083
electric_supply	0.000098
wh_owner_type_Rented	0.000100
Competitor_in_mkt	0.000278
wh_breakdown_l3m	0.000407
govt_check_l3m	0.000570
num_refill_req_l3m	0.000577
workers_num	0.000623
distributor_num	0.000719
temp_reg_mach	0.000799
dist_from_hub	0.000818
retail_shop_num	0.000831
transport_issue_l1y	0.001067
approved_wh_govt_certificate	0.009515
storage_issue_reported_l3m	0.982717

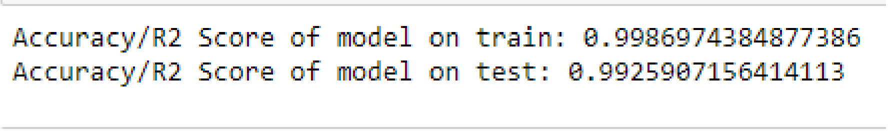
Table.21

Ensemble Modelling

Bagging Regression Model

- A Bagging regression is an ensemble meta-estimator that fits base regressors each on random subsets of the original dataset and then aggregate their individual predictions (either by voting or by averaging) to form a final prediction. Such a meta-estimator can typically be used as a way to reduce the variance of a black-box estimator (e.g., a decision tree), by introducing randomization into its construction procedure and then making an ensemble out of it.

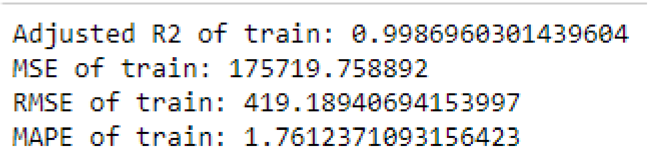
Performance Metrics – Bagging Regression Model



```
Accuracy/R2 Score of model on train: 0.9986974384877386
Accuracy/R2 Score of model on test: 0.9925907156414113
```

Table.22

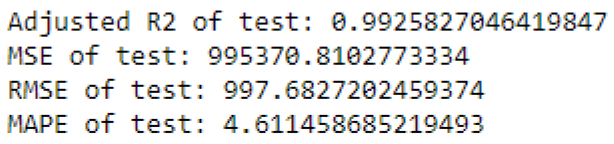
- **Train set**



```
Adjusted R2 of train: 0.9986960301439604
MSE of train: 175719.758892
RMSE of train: 419.18940694153997
MAPE of train: 1.7612371093156423
```

Table.23

- **Test set**



```
Adjusted R2 of test: 0.9925827046419847
MSE of test: 995370.8102773334
RMSE of test: 997.6827202459374
MAPE of test: 4.611458685219493
```

Table.24

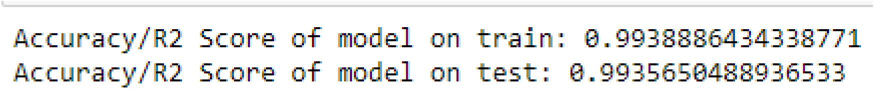
Inferences:

- Bagging Regression model is performing well in both train and test set hence it is a valid model.

Gradient Boosting Regression Model

- GB builds an additive model in a forward stage-wise fashion; it allows for the optimization of arbitrary differentiable loss functions. In each stage a regression tree is fit on the negative gradient of the given loss function. By fitting the new model to the residuals, the overall learner gradually improves in areas where residual are initially high.

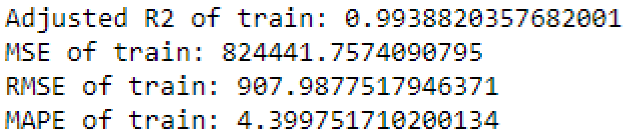
Performance Metrics – Boosting Regression Model



```
Accuracy/R2 Score of model on train: 0.9938886434338771
Accuracy/R2 Score of model on test: 0.9935650488936533
```

Table.25

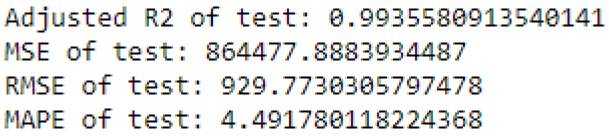
- **Train set**



```
Adjusted R2 of train: 0.9938820357682001
MSE of train: 824441.7574090795
RMSE of train: 907.9877517946371
MAPE of train: 4.399751710200134
```

Table.26

- **Test set**



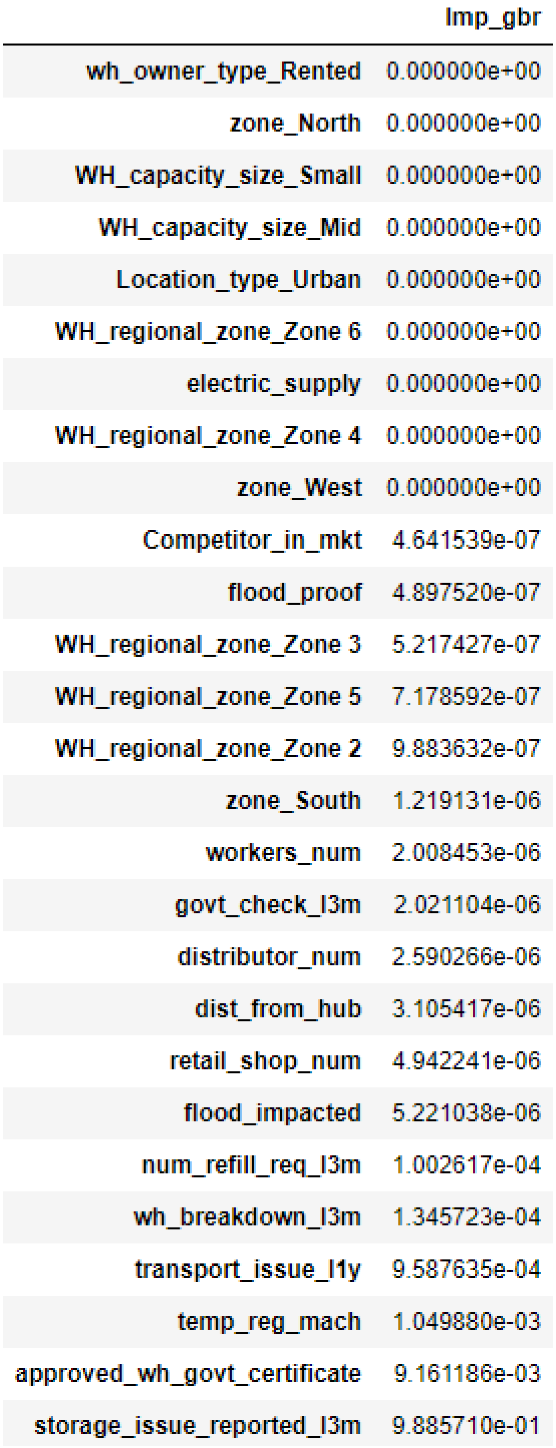
```
Adjusted R2 of test: 0.9935580913540141
MSE of test: 864477.8883934487
RMSE of test: 929.7730305797478
MAPE of test: 4.491780118224368
```

Table.27

Inferences:

- Gradient Boost Regression model is performing well in both train and test set hence it is a valid model.

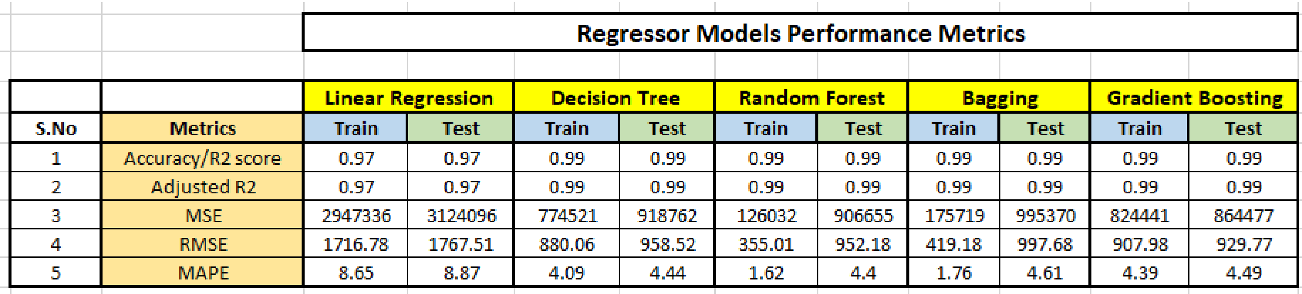
Feature Importance of Gradient Boosting Regression Model:



	Imp_gbr
wh_owner_type_Rented	0.000000e+00
zone_North	0.000000e+00
WH_capacity_size_Small	0.000000e+00
WH_capacity_size_Mid	0.000000e+00
Location_type_Urban	0.000000e+00
WH_regional_zone_Zone 6	0.000000e+00
electric_supply	0.000000e+00
WH_regional_zone_Zone 4	0.000000e+00
zone_West	0.000000e+00
Competitor_in_mkt	4.641539e-07
flood_proof	4.897520e-07
WH_regional_zone_Zone 3	5.217427e-07
WH_regional_zone_Zone 5	7.178592e-07
WH_regional_zone_Zone 2	9.883632e-07
zone_South	1.219131e-06
workers_num	2.008453e-06
govt_check_l3m	2.021104e-06
distributor_num	2.590266e-06
dist_from_hub	3.105417e-06
retail_shop_num	4.942241e-06
flood_impacted	5.221038e-06
num_refill_req_l3m	1.002617e-04
wh_breakdown_l3m	1.345723e-04
transport_issue_l1y	9.587635e-04
temp_reg_mach	1.049880e-03
approved_wh_govt_certificate	9.161186e-03
storage_issue_reported_l3m	9.885710e-01

Table.28

Model Validation and Choosing Best Model:



		Regressor Models Performance Metrics									
		Linear Regression		Decision Tree		Random Forest		Bagging		Gradient Boosting	
S.No	Metrics	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
1	Accuracy/R2 score	0.97	0.97	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
2	Adjusted R2	0.97	0.97	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
3	MSE	2947336	3124096	774521	918762	126032	906655	175719	995370	824441	864477
4	RMSE	1716.78	1767.51	880.06	958.52	355.01	952.18	419.18	997.68	907.98	929.77
5	MAPE	8.65	8.87	4.09	4.44	1.62	4.4	1.76	4.61	4.39	4.49

Table.29

Best Model Inferences:

- If we look at only accuracy all the models are performing very well.
- But if we have to select one out of all the models we can infer that Random Forest Model is performing best on basis of MAPE and RMSE metrics.
- It has least MAPE for train set at 1.62 % and at test set 4.4%
- Also RMSE values for train set is 355 and test set is 952 also lowest among all the models.
- Accuracy and Adjusted R2 value are coming out to be same as adjusted R2 value is impacted by addition of irrelevant variable in dataset. Since we are not adding or removing any variables from the dataset both values can coming out to be same.
- Accuracy on train and test set for Random Forest Model is 99%.

Random Forest Model Prediction Visualisation:

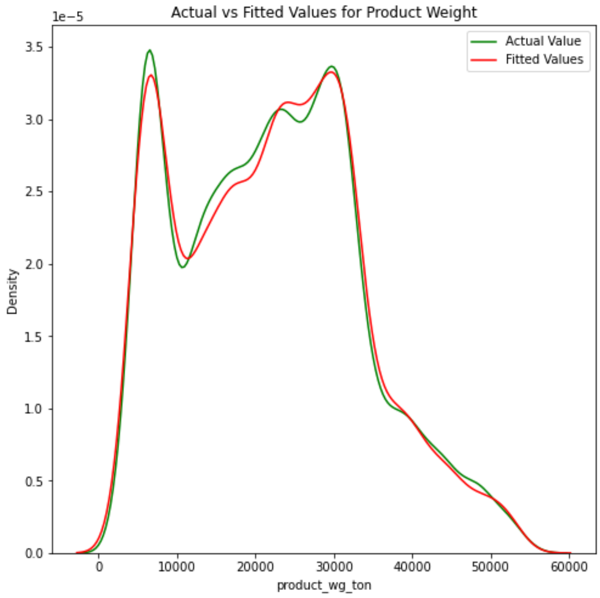
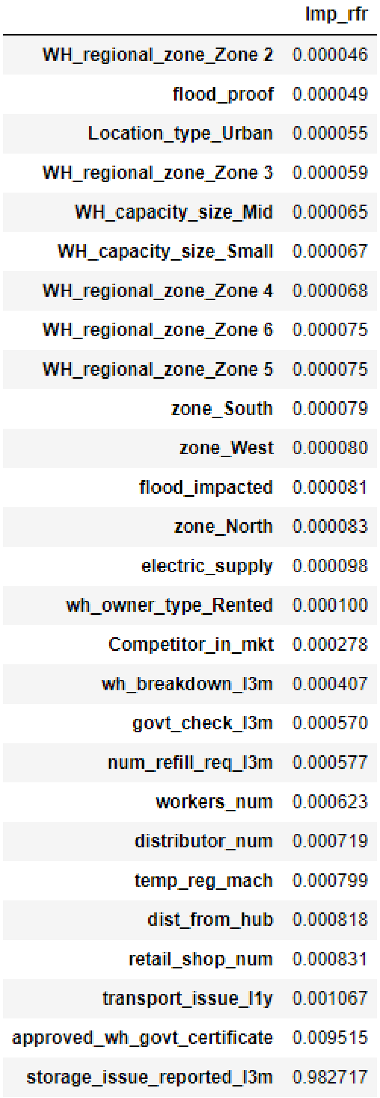


Fig.45

- As we can also visually see from the plot that for our test set our model is able to predict the values very accurately.

Feature Importance of Random Forest Model (our best model):



	Imp_rfr
WH_regional_zone_Zone 2	0.000046
flood_proof	0.000049
Location_type_Urban	0.000055
WH_regional_zone_Zone 3	0.000059
WH_capacity_size_Mid	0.000065
WH_capacity_size_Small	0.000067
WH_regional_zone_Zone 4	0.000068
WH_regional_zone_Zone 6	0.000075
WH_regional_zone_Zone 5	0.000075
zone_South	0.000079
zone_West	0.000080
flood_impacted	0.000081
zone_North	0.000083
electric_supply	0.000098
wh_owner_type_Rented	0.000100
Competitor_in_mkt	0.000278
wh_breakdown_l3m	0.000407
govt_check_l3m	0.000570
num_refill_req_l3m	0.000577
workers_num	0.000623
distributor_num	0.000719
temp_reg_mach	0.000799
dist_from_hub	0.000818
retail_shop_num	0.000831
transport_issue_l1y	0.001067
approved_wh_govt_certificate	0.009515
storage_issue_reported_l3m	0.982717

Table.21

Business Insights:

- From the above feature importance table features whose value are close to 1 are considered important features for our target variable predication and features whose value are close to 0 are considered not relevant for our target variable predictions.
- The storage_issue_reported_l3m is the most important attribute and it has high impact on our target variable, so business should make decision keeping this on mind.
- Other important features are approved_wh_govt_certificate and tansport_issue_l1y.

EDA using our best predictor (storage_issue_reported_l3m_) as per Random Forest Regression model:

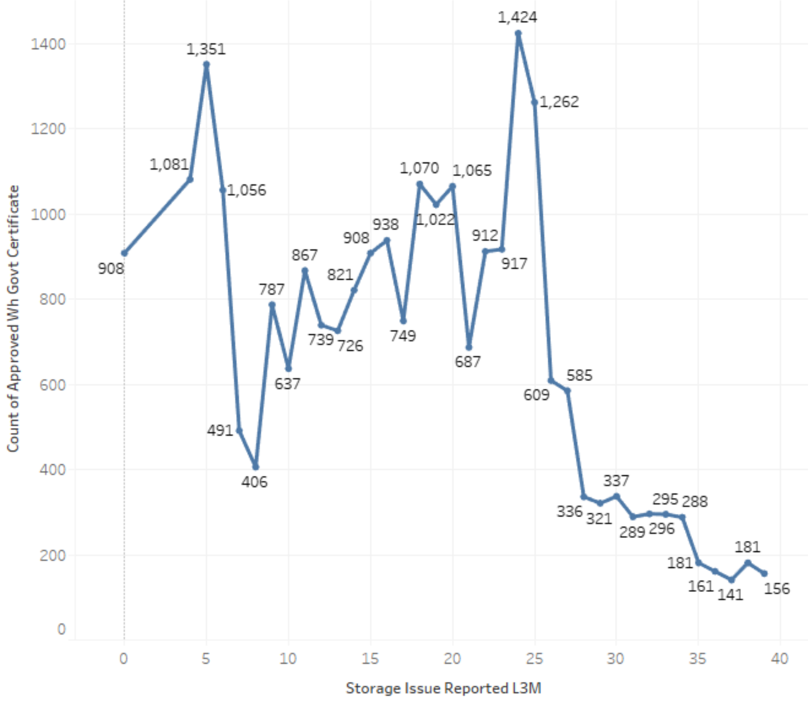


Fig.46

- From the above graph we can infer that approved government certificates reduces as number of storage issues increase.

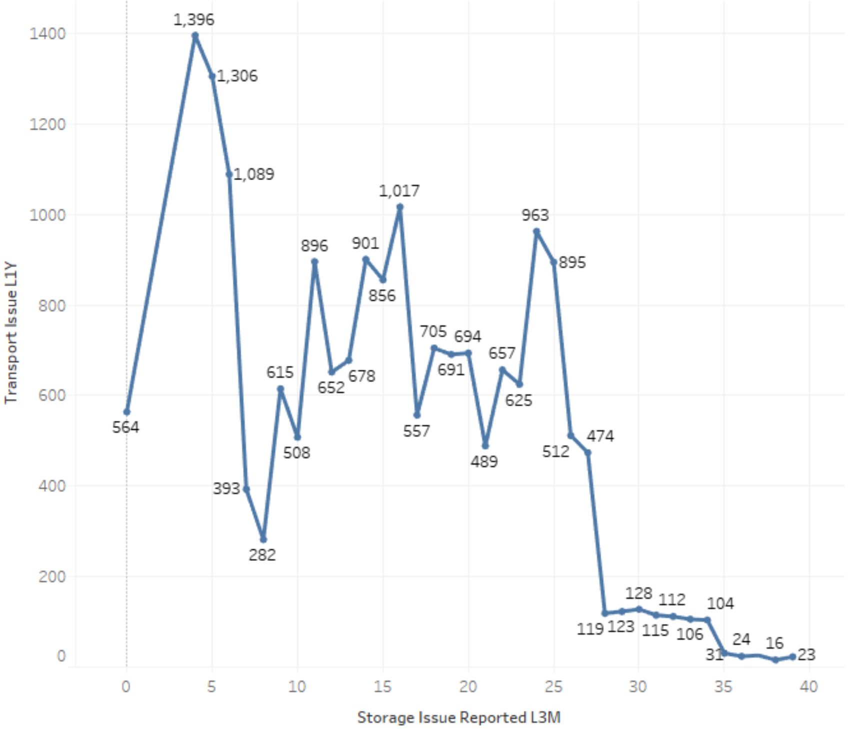


Fig.47

- From the above graph we can infer that there are lot of transport related issue when storage issues are very low.

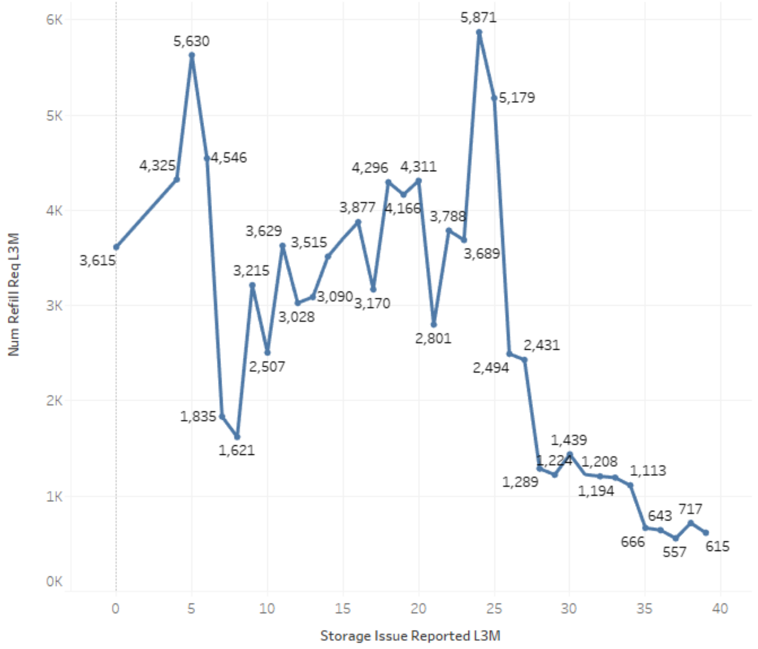


Fig.48

- From the above graph we can infer number of refill in warehouses reduced drastically as number issue with warehouses increases above 25.

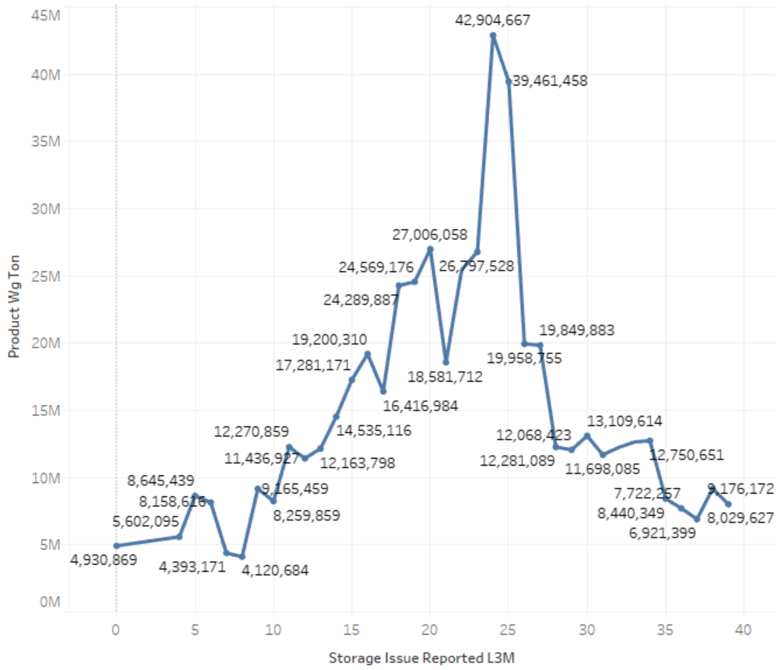


Fig.49

- In the above graph we compare our target variable 'product_wg_ton' with our best predictor, we can infer that even with storage issues product quantity keeps on increasing till number of storage issues are at 25 but after that the product demand decreases drastically for all the warehouse having issue above 25.

Final Interpretation/ Recommendation:

- Warehouses having storage related issues more than 25 are handling very less weights. Business can inspect for causes here and can decide whether they are operationally profitable or not.
- Transport related issues results in significant reduction in product weights available at the warehouses. Business need to check on this.
- Reason for why newer warehouses are not having as much product as compared to warehouse between periods 1998 to 2006, business need to inspect.
- Warehouses that have been made flood proof do not have lot of product available with them. Investment in making them flood proof is effective at all.
- Having electric supply in warehouses positive impact on the product availability.
- Viability of urban areas warehouses need to accessed as product in urban is very less compared to rural areas. If viable advertisement campaign in urban areas can used to push more product in the market.
- Promotion and adverting strategies need to implement for region where there are more than 2 competitors in market.

END