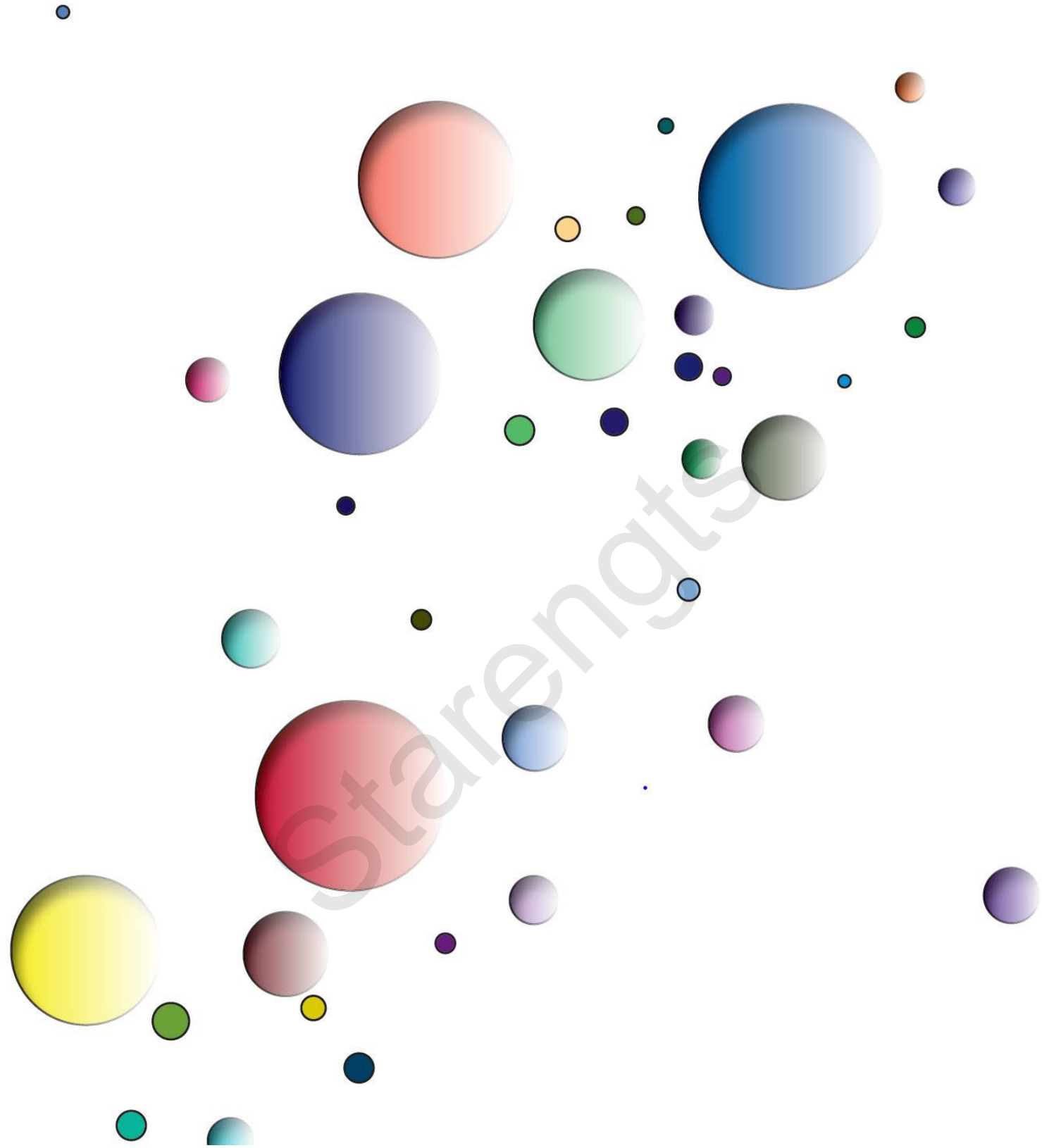
Date: 03 - July -2022



Starengts
Predictive Analytics

Table of Contents

Capstone Project Final Submission: Supply Chain (Product Shipment Weight)

List of Figures:

List of Tables:

1. Introduction

1.1 Introduction about the Problem

The FMGC Company recently get into the business of instant noodles & dealing with the demand-supply issue. The company is not able to fulfil the demand generated by the warehouse or they over dispatched the noodles quantity to full fill the requirements. The company is not able to find out what the optimum quantity should dispatch to their warehouses.

1.2 Why we need to solve the problem

- We need to understand the supply-chain business of the company & try to analyse the demand pattern based on different areas in the country to overcome form demand & supply issues.
- We also need to reduce the inventory cost losses by forecasting the optimums weight of the product to be shipped to each warehouse.
- Because of demand & supply issues, many warehouses will underperform & other warehouses will deal with higher inventory carrying costs.
- We need to understand the business opportunities based on demand in each warehouse & fulfil the requirements of the warehouse.
- The low supply to warehouses leads to direct business loss, we need to understand high performing warehouses and low performing warehouses & plan supply quantity according.
- The low performing warehouse needs to understand & plan an advertising campaign to improve the performance of the warehouse.
- Because higher demand & low supply leads to customer dissatisfaction and the probability the customer move to another brand.
- There will be a probability FMGC company will lose its brand value in high performance and low supply area or zone

2. EDA and Business Implication

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

To understand the data, Let us first read the dataset:

	Ware_house_ID	WH_Manager_ID	Location_type	WH_capacity_size	zone	WH_regional_zone	num_refill_req_l3m	transport_issue_l1y	Competitor_in_mkt	retail_shop_num	...	electric_supply	dist_from_hub
0	WH_100000	EID_50000	Urban	Small	West	Zone 6	3	1	2	4651	...	1	91
1	WH_100001	EID_50001	Rural	Large	North	Zone 5	0	0	4	6217	...	1	210
2	WH_100002	EID_50002	Rural	Mid	South	Zone 2	1	0	4	4306	...	0	161
3	WH_100003	EID_50003	Rural	Mid	North	Zone 3	7	4	2	6000	...	0	103
4	WH_100004	EID_50004	Rural	Large	North	Zone 5	3	1	2	4740	...	1	112

5 rows × 24 columns

workers_num	wh_est_year	storage_issue_reported_l3m	temp_reg_mach	approved_wh_govt_certificate	wh_breakdown_l3m	govt_check_l3m	product_wg_ton
29.0	NaN	13	0	A	5	15	17115
31.0	NaN	4	0	A	3	17	5074
37.0	NaN	17	0	A	6	22	23137
21.0	NaN	17	1	A+	3	27	22115
25.0	2009.0	18	0	C	6	24	24071

Table 1.1 Data set –FMGC Company

The data is given about the warehouse, each warehouse have a unique identification – and each manager assigned for unique ID based on the warehouse, Warehouse also decided into location (Urban & Rural) , Warehouse capacity – (small, medium & large) The data available even its not having establishment year. The data available for target variable is only 3 months.

The data is available from establishment year since FY1996 to FY2023.

Some year of establishment is missing as we can understand the warehouse is new and still under government approval.

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

Let us see the data visualization:

1. Let us see the shape of the data set – the set having **25000 Rows** and **25 columns**
2. The total size of dataset is 600000
3. Please see the below descriptive details

	count	mean	std	min	25%	50%	75%	max
num_refill_req_l3m	25000.00	4.09	2.61	0.00	2.00	4.00	6.00	8.00
transport_issue_l1y	25000.00	0.77	1.20	0.00	0.00	0.00	1.00	5.00
Competitor_in_mkt	25000.00	3.10	1.14	0.00	2.00	3.00	4.00	12.00
retail_shop_num	25000.00	4985.71	1052.83	1821.00	4313.00	4859.00	5500.00	11008.00
distributor_num	25000.00	42.42	16.06	15.00	29.00	42.00	56.00	70.00
flood_impacted	25000.00	0.10	0.30	0.00	0.00	0.00	0.00	1.00
flood_proof	25000.00	0.05	0.23	0.00	0.00	0.00	0.00	1.00
electric_supply	25000.00	0.66	0.47	0.00	0.00	1.00	1.00	1.00
dist_from_hub	25000.00	163.54	62.72	55.00	109.00	164.00	218.00	271.00
workers_num	24010.00	28.94	7.87	10.00	24.00	28.00	33.00	98.00
wh_est_year	13119.00	2009.38	7.53	1996.00	2003.00	2009.00	2016.00	2023.00
storage_issue_reported_l3m	25000.00	17.13	9.16	0.00	10.00	18.00	24.00	39.00
temp_reg_mach	25000.00	0.30	0.46	0.00	0.00	0.00	1.00	1.00
wh_breakdown_l3m	25000.00	3.48	1.69	0.00	2.00	3.00	5.00	6.00
govt_check_l3m	25000.00	18.81	8.63	1.00	11.00	21.00	26.00	32.00
product_wg_ton	25000.00	22102.63	11607.76	2065.00	13059.00	22101.00	30103.00	55151.00

Table 1.2 Data Description

```
<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 1.3 Data information -1

- We can see the each variable have 25000 data points, while **Number of workers** & **Established year** is having missing values.
- We can also see the **flood_impacted, flood_proof, electric_supply & temp_reg_mach** minimum value is 0 and maximum is one & STD below 1.
- The average product weight is 22102 and max is 55151.

2.3 Understanding of attributes

Let us understand the attributes:

The 8 types of attributes is object time, 14 data variables is int64 type & 2 variable is float64 type

The size of the data set is 4.6+MB

The first two columns is unique ID

Worker number and year of establishment having missing data

We can see the name given to variable is not self-exclamatory, we recommend to rename so can management understand easily

We have rename the variables please see the changed variables name:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25000 entries, 0 to 24999
Data columns (total 24 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   WH_house_ID                             25000 non-null  object
 1   WH_Manager_ID                           25000 non-null  object
 2   WH_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   Number_of_refill_req_in_last_3_months   25000 non-null  int64
 7   Transport_issue_in_last_one_year        25000 non-null  int64
 8   Number_of_competitor_in_mkt             25000 non-null  int64
 9   Number_of_ratail_shop                   25000 non-null  int64
 10  WH_owner_type                           25000 non-null  object
 11  Number_of_distributor                   25000 non-null  int64
 12  WH_in_Flood_impacted_area               25000 non-null  int64
 13  WH_in_flood_proof_area                  25000 non-null  int64
 14  generator_backup                        25000 non-null  int64
 15  Distance_bet_warehouse_&_production_hub 25000 non-null  int64
 16  Number_of_workers                       24010 non-null  float64
 17  WH_established_year                     13119 non-null  float64
 18  storage_issue_reported_in_last_3_months 25000 non-null  int64
 19  WH_temp_regulat                         25000 non-null  int64
 20  approved_wh_govt_certificate            24092 non-null  object
 21  WH_breakdown_in_last_3_months           25000 non-null  int64
 22  government_audit_in_last_3_months       25000 non-null  int64
 23  Product_weight_in_ton                   25000 non-null  int64
dtypes: float64(2), int64(14), object(8)
memory usage: 4.6+ MB
```

Table 1.4 Data information -2

The variable name is important to easy understanding & in visualization also.

2.4 Exploratory Data Analysis

2.4.1 Univariate analysis

We have 8 object type variables in which 2 is unique identifier, let us analysis the reaming 6 first:

Warehouse location Type:

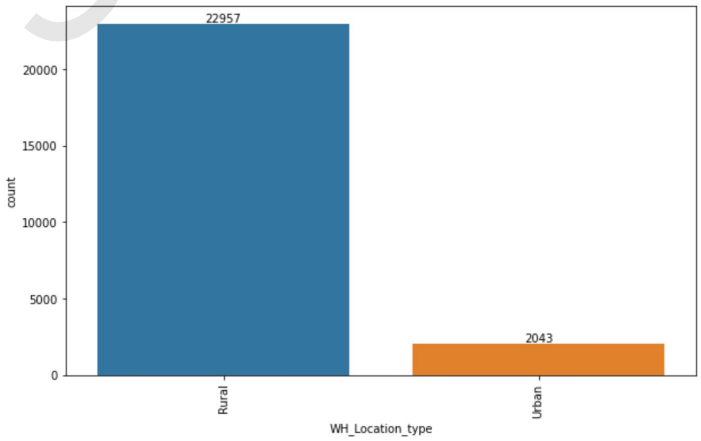


Fig.1.1 Warehouse location Type

We can see that the rural area having more number of warehouse as compare to urban area, the 92 % of warehouse is in the rural area and 8% in the urban are.

Warehouse capacity size:

We can see the warehouse capacity size both large & medium have almost same number of warehouse and small capacity will be 4811 number

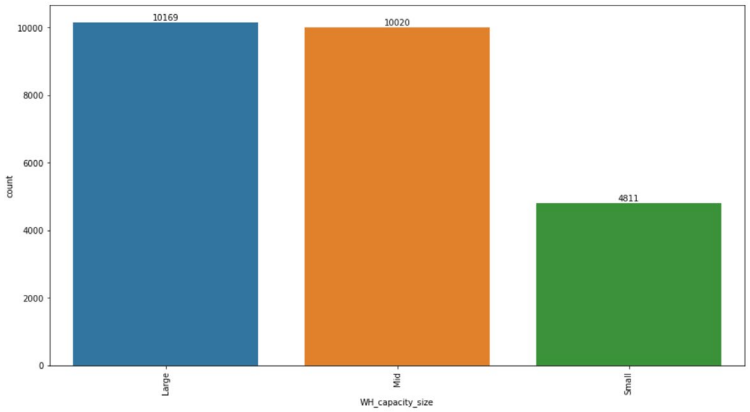


Fig.1.2 Warehouse capacity size

Zone:

We can see the number of warehouse divided in to 4 zone North zone have the highest number of warehouse and east have the lowest number of warehouse

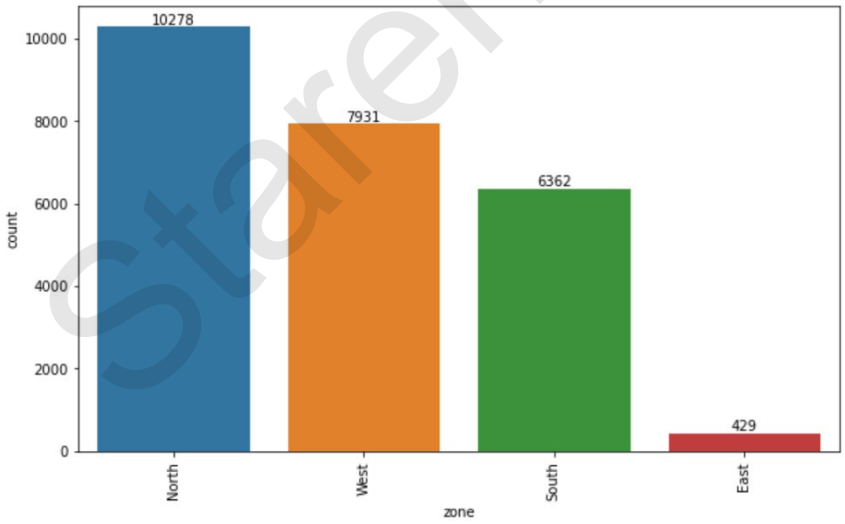


Fig.1.3 Zone

WH Regional Zone:

As per below bar chart we can understand that the zone6 have highest number of warehouse followed by zone 5 and lest is zone 1.

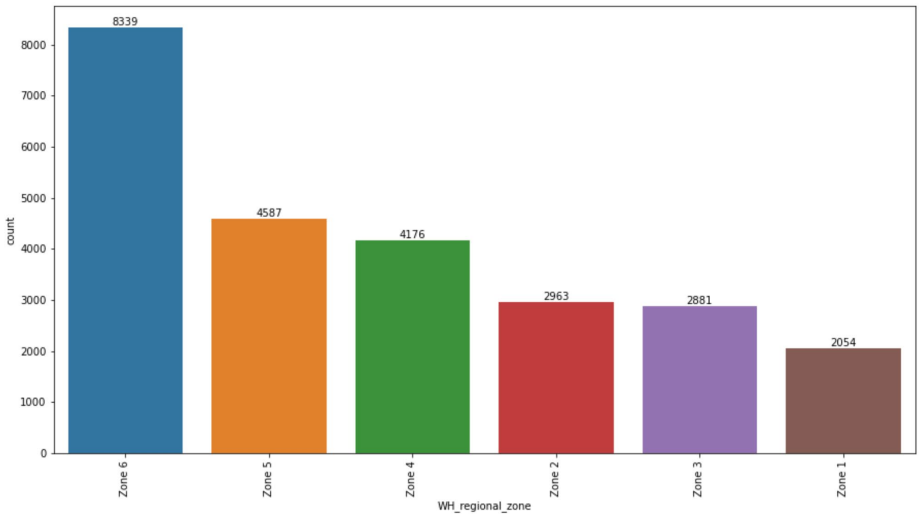


Fig.1.4 WH Regional Zone

WH Owner Type

We can see that the company Owned warehouse 55% and rented is 45%.

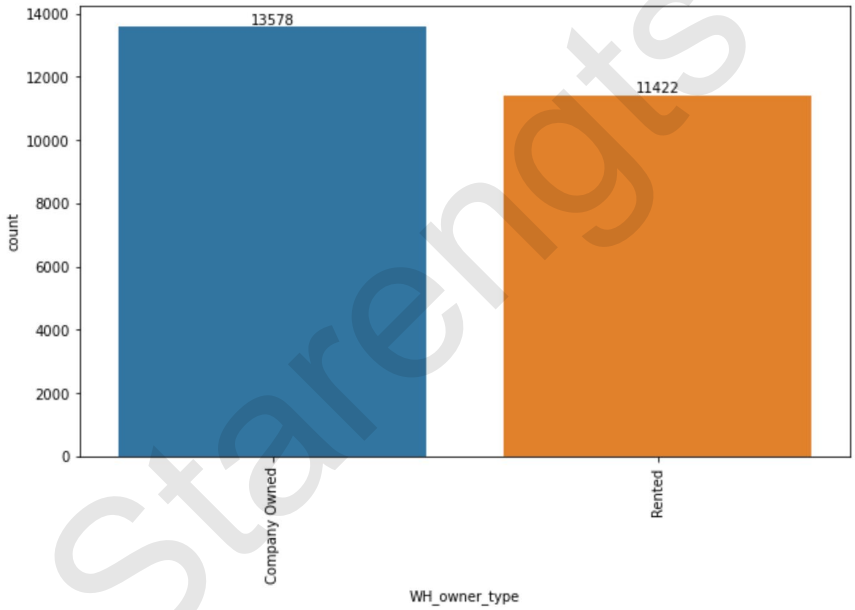


Fig.1.5 WH Owner Type

Now let us see the univariate analysis for continuous variables:

We can see the some variables have only zero and one values, let us understand first.

Warehouse in flood impacted area:

We can see the 9% warehouse is in flood impacted area.

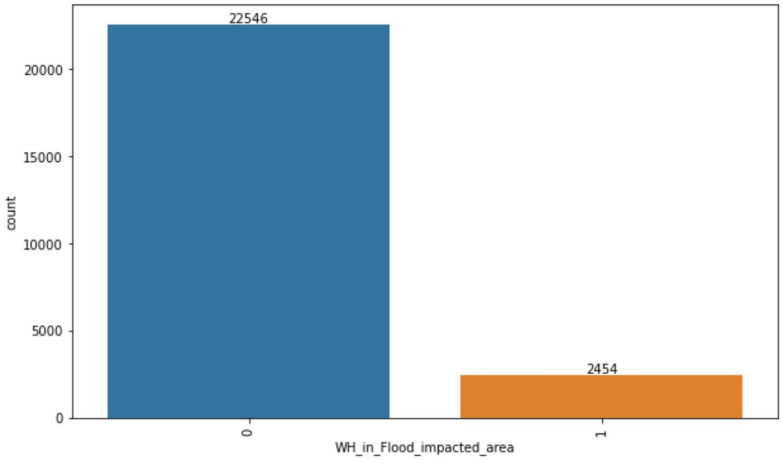


Fig.1.6 Warehouse in flood impacted area

Warehouse in flood proof area:

The 5.4% Warehouse is in flood proof area

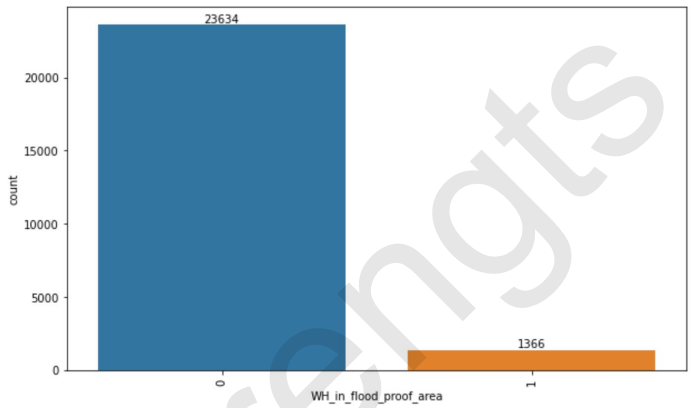


Fig.1.7 Warehouse in flood proof area

Generator Back up support in warehouse:

We can see the 67% warehouse have the backup generator support and 45% will not have.

We will see the correlation with target variable in next topic.

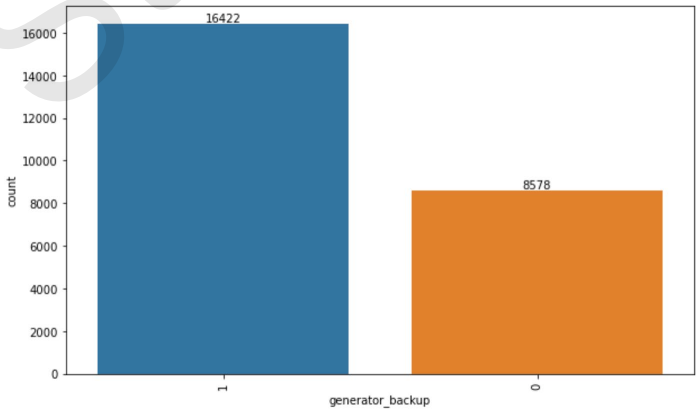


Fig.1.8 Generator back up support in warehouse

Warehouse have temperature regulate machine:

30% warehouse having the temperature regulation option

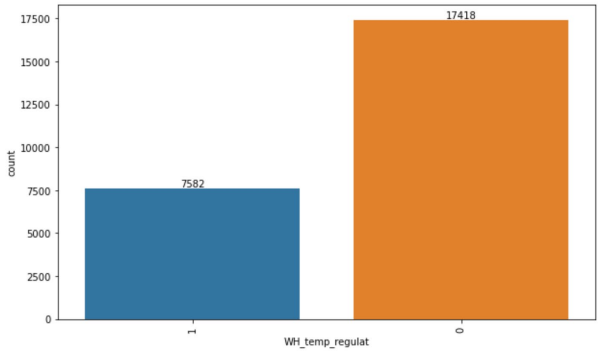


Fig.1.9 Warehouse have temperature regulate machine

Approved warehouse by government with certification:

We can see the some warehouse is not applicable, we have changed the NaN with Not applicable, because some warehouse is still under approval process.

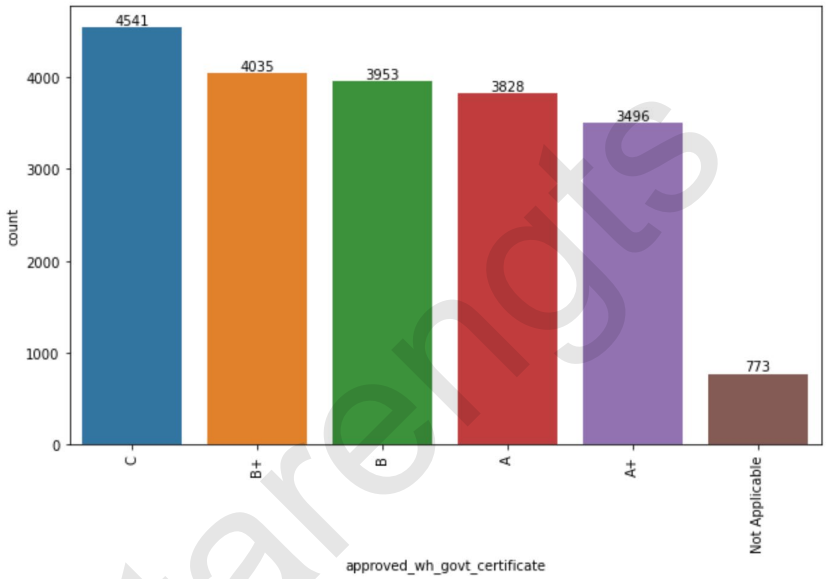


Fig.1.10 Approved warehouse by government with certification

Let us see now continues variables: Box Plot

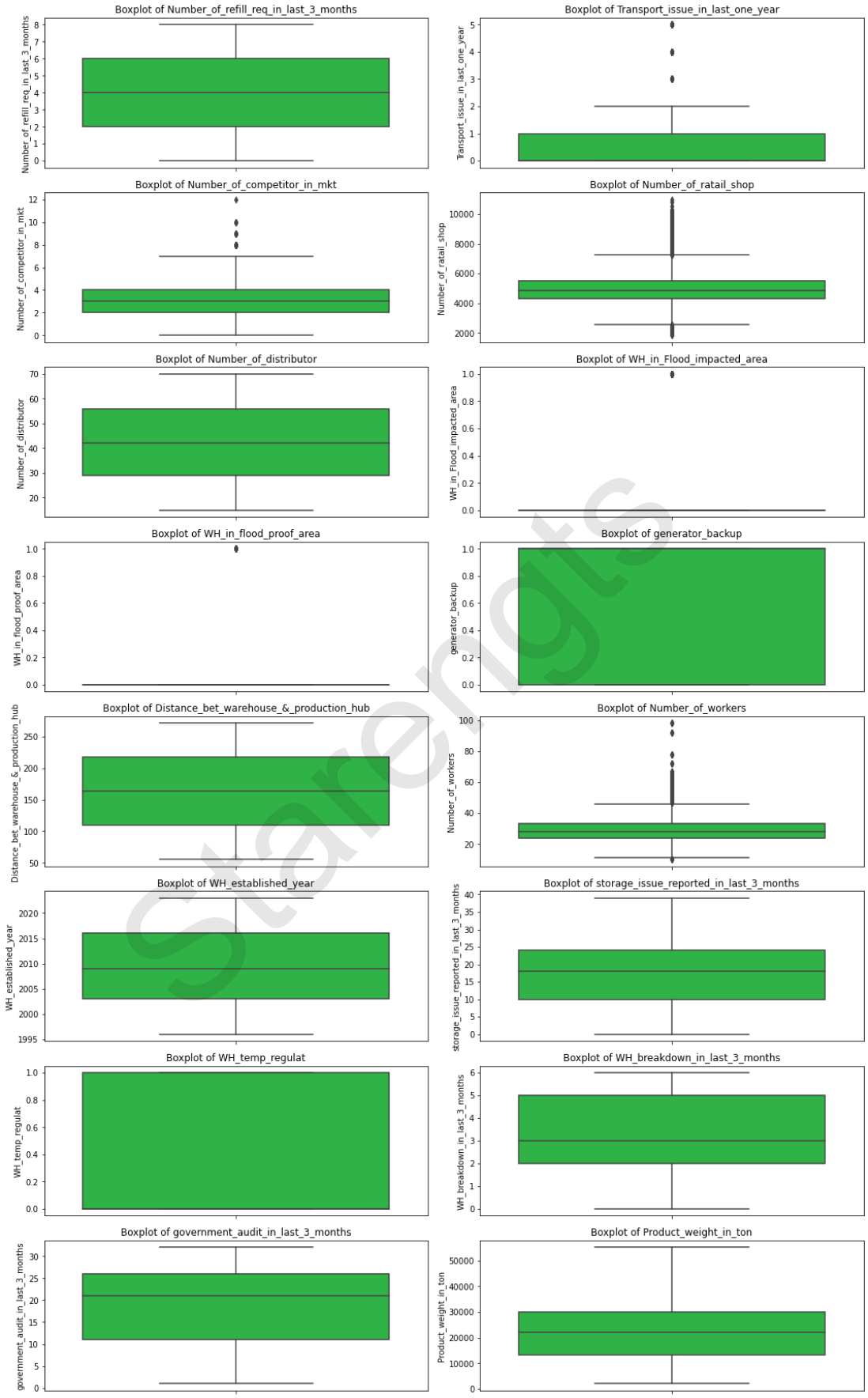


Fig.1.11 Continues variables-Box Plot

Let us see now continues variables: Histogram

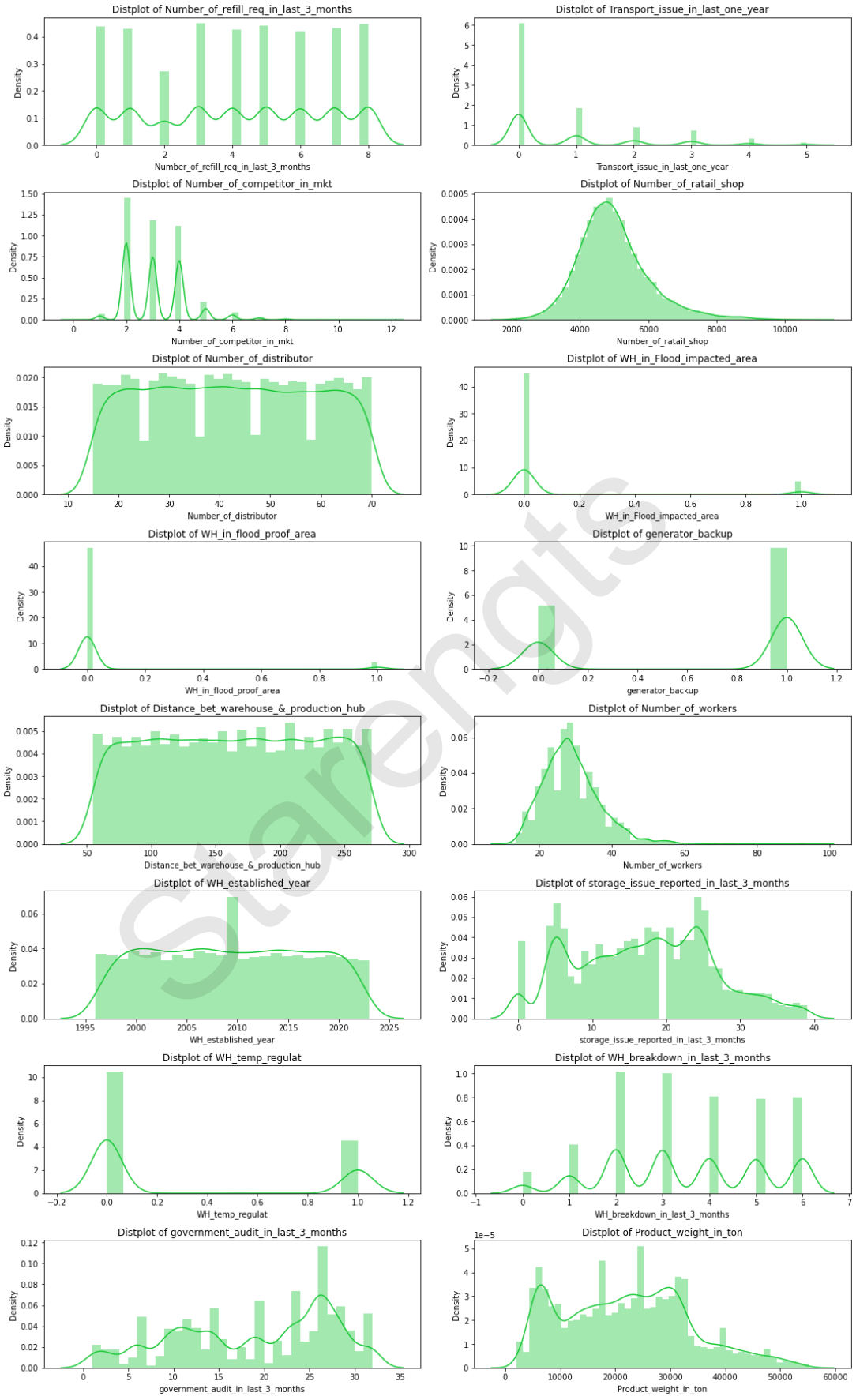


Fig.1.12 Continues variables- Histogram

As per all the continue variables below are the observation

1. The continuous data set have the **outliers** – Transport issue in last one year, Number of competitor in the market, Number of retail shop & number of works.
2. We can also see the skewness like **transport issue in last one year, number of retail shop, Number of worker & number of competitor is highly right skew.** Based on skewness values we will do the outlier treatment
3. Some data have only nominal values 1 & 0, that we classified based on category.

2.4.2 Bivariate analysis (relationship between different variables, correlations)

Let is compare Warehouse location Type this with our target variable:

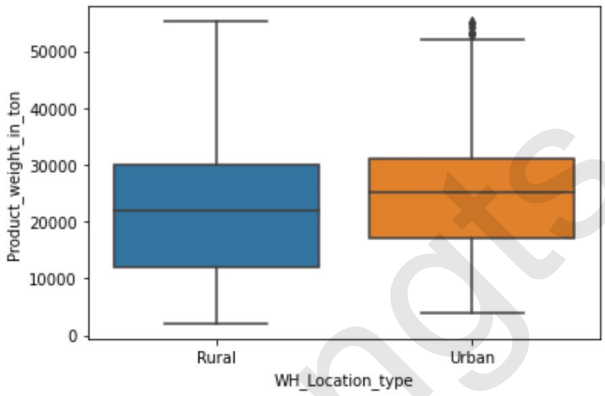


Fig.1.13 Warehouse location Type this with our target variable

Comparison of location: Shows that the median product weight dispatch in the urban are is greater than the rural area

Comparison of dispersion: We can see the difference in interquartile ranges and over all data set is higher for rural area & overall range of product weight is higher in rural area

Comparison of skewness: Both the data look like right skewness, rural area slightly more right skewness

Comparison of potential outliers: Urban area can see the outlier.

Warehouse capacity size v/s Product Capacity in tone:

Let see if any relationship with the target variables:

It seems that the warehouse capacity size will not influence the output

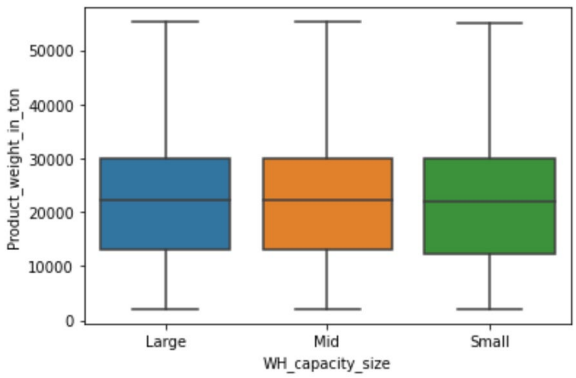


Fig.1.14 Warehouse capacity size v/s Product Capacity in tone

Warehouse regional zone v/s Product Capacity in tone:

We can't find any relation between reginal zone 5, 4, & 3 product capacity directly but zone 2 have the higher median then other, we can say that the zone 2 is good performer, Zone 1 is low performer

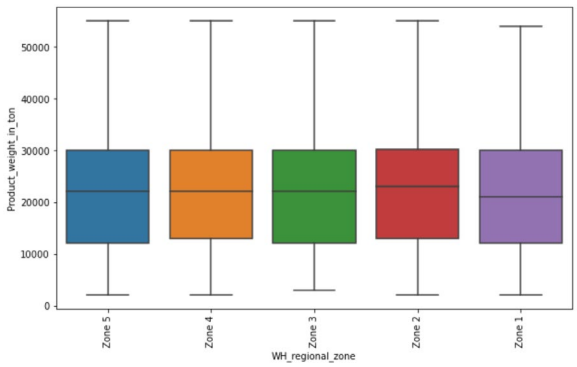


Fig.1.15 Warehouse regional zone v/s Product Capacity in tone

Zone v/s Product Capacity in tone:

We can observe the East zone have slightly higher median value, we can say that the east zone is performing well as compare to other zone

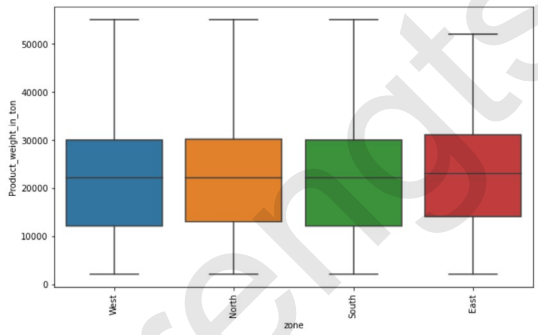


Fig.1.16 Zone v/s Product Capacity in tone

Warehouse owner type v/s Product Capacity in tone:

We can't find the relation ship

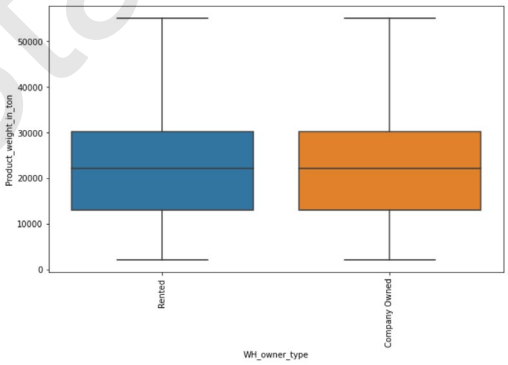


Fig.1.17 Warehouse owner type v/s Product Capacity in tone

Warehouse in flood impacted area v/s Product Capacity in tone:

We can't find the relation ship

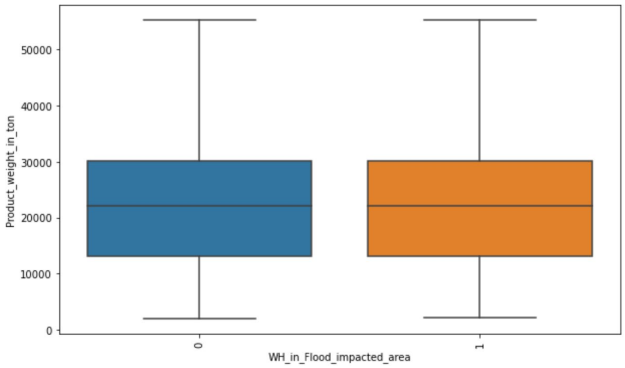


Fig.1.18 Warehouse in flood impacted area v/s Product Capacity in tone

Warehouse in flood proof area v/s Product Capacity in tone:

We can't find the relation ship

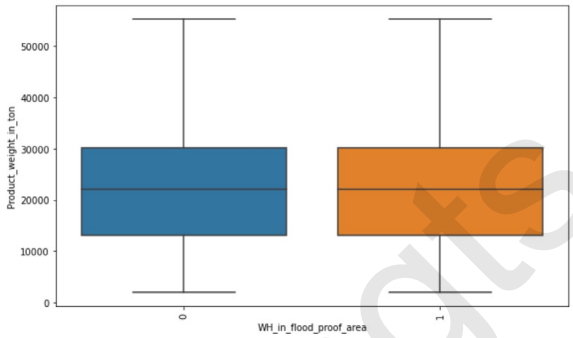


Fig.1.19 Warehouse in flood proof area v/s Product Capacity in tone

Generator Backup v/s Product Capacity in tone:

We can't find the relation ship

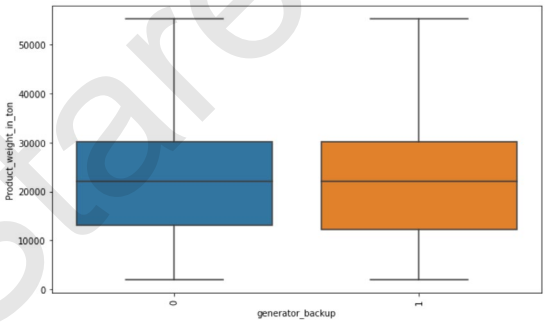


Fig.1.20 Generator Backup v/s Product Capacity in tone

Warehouse temperature regulator machine v/s Product Capacity in tone:

We can see those warehouse have the temperature regulator machine having the higher median value

We can say that the because of temperature control they can store the more product as comparer the non-regulated teamp. Control.

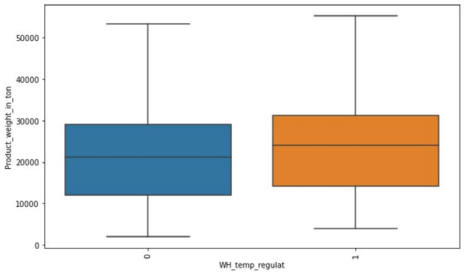


Fig.1.21 Warehouse temperature regulator machine v/s Product Capacity in tone

Warehouse establishment year v/s Product Capacity in tone:

We can see the median value shifted down after 2005, this is the clear indicator that the old warehouse preformation well as compare to new warehouse

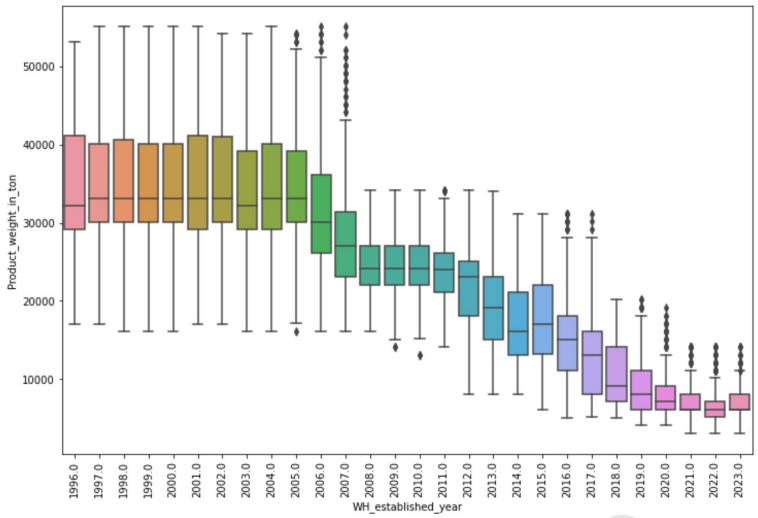


Fig.1.22 Warehouse establishment year v/s Product Capacity in tone

Storage issue reported in last 3 months v/s Product Capacity in tone:

We can see that the direct correlation below issue reported and product weight, however this is misleading for model building. It's just to say that the when number of storage product is increases the storage issue will increases. We should drop this column for model building or we will get the 99 % accuracy of model.

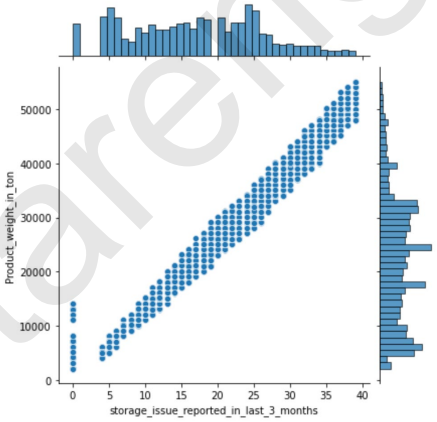


Fig.1.23 Storage issue reported in last 3 months v/s Product Capacity in tone

Zone v/s Regional zone:

We can see the north zone having the highest warehouse and east zone have the lowest warehouse.

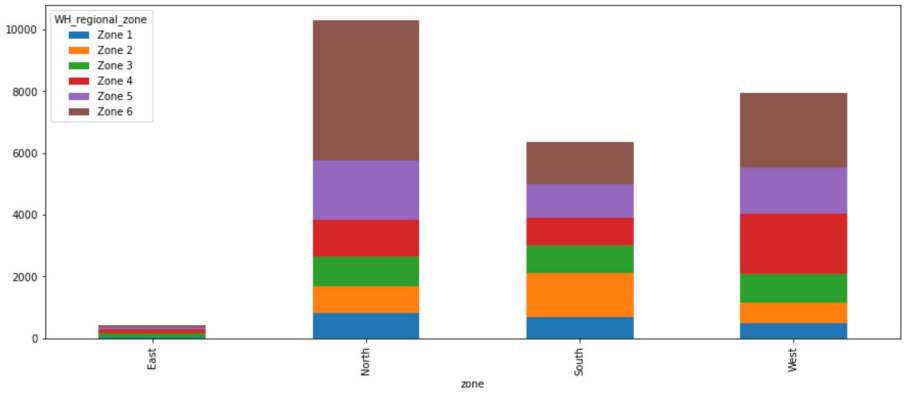


Fig.1.24 Zone v/s Regional zone

Zone v/s Warehouse capacity:

The 50 % largest warehouse capacity cover by north zone.

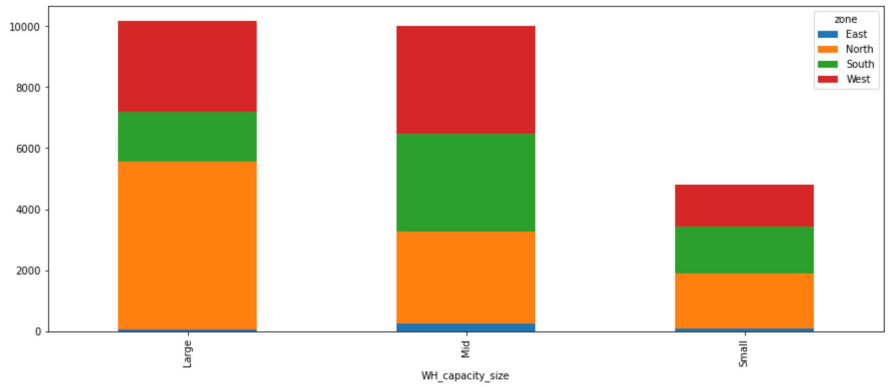


Fig.1.25 Zone v/s Warehouse capacity

Let us see the all continues variable correlation in a glance:

- As we already earlier discussed the highest positive correlation between product weight and the number of issue reported.
- The highest negative correlation between product weight and year of establishment – 40 % data is missing
- Year of establishment storage issue reported have the high negative correlation, we can say that as we discussed the old warehouse performing good and compare to new.

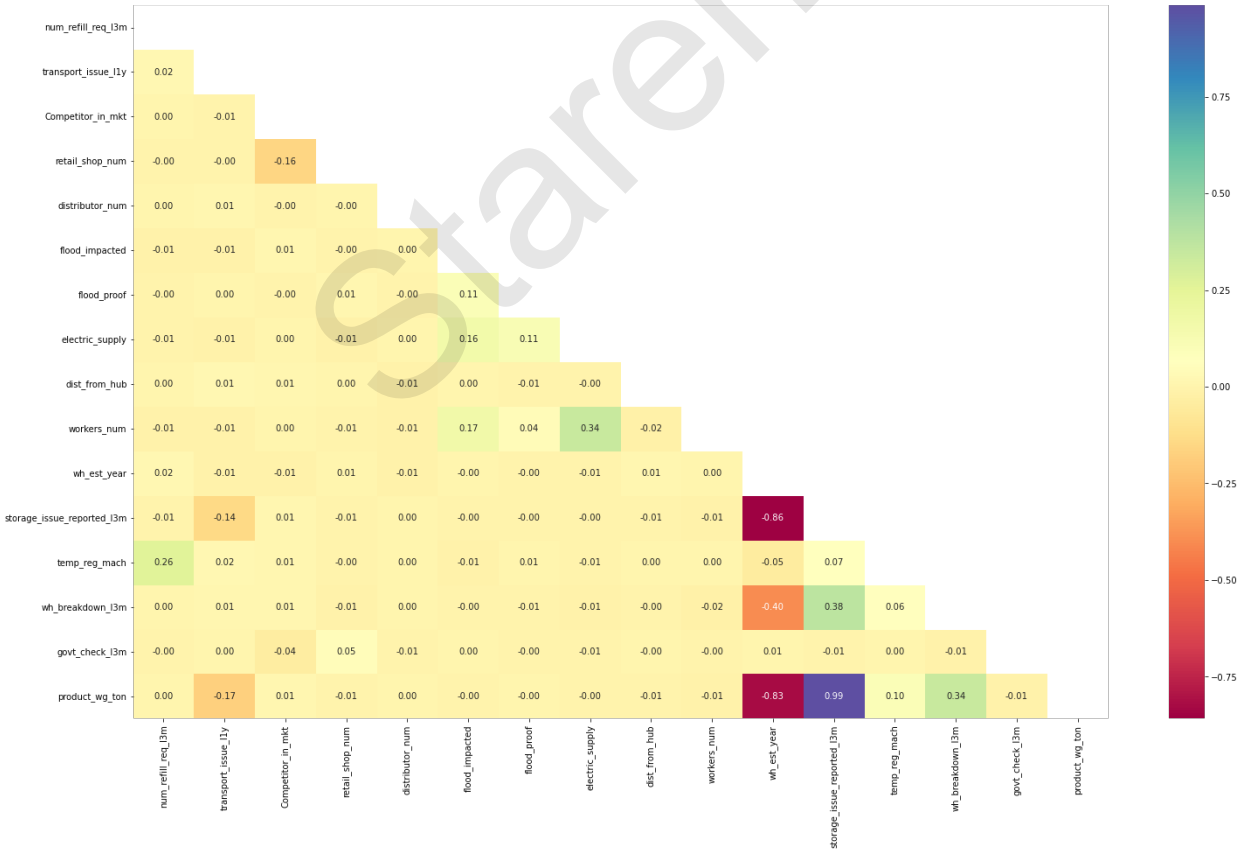


Fig.1.26 Continues variable correlation in a glance

2.4.3 Impact of Analysis on the business

1. The company invested in warehouse 92% in a rural area & compare to urban are only 8%.
2. The warehouse is divided percentage-wise - 40% of large size warehouse, 40% of warehouse and a medium having 20% is small size warehouse
3. The number of warehouses divided into 4 zones and north having the highest warehouse ~40% & the east contribution only 2 %.
4. In Regional Zone – Zone 6 has the highest number of warehouses ~33%
5. Company Owned warehouse 55% and rented is 45%
6. Company have 9% warehouse in the flood-impacted area
7. Company have a 5.4% warehouse in a flood-proof area
8. Company have 67% warehouse with backup generator support and 45% will not have
9. Only 30% of warehouses have the temperature regulation option
10. Company still under government approval for 773 warehouses
11. No good correlation found maximum variables – Only 2 variable has good correlation 1st Year of establishment (Negative) & second is number of storage issue reported in last 3 months (Positive)
12. We can see the mid capacity warehouse in east and larger capacity warehouse in east have higher median value, we can say that mid and large capacity warehouse performance good in east zone
13. The east zone still performing better than the other zone but did not have any warehouses in zone -2 and the lowest number of warehouses in the east zone.
    a. Company needs to look into the east zone and increase the number of warehouses
    b. East zone 2 needs to develop a new warehouse as they don't have any warehouse in that area
14. The warehouse with a temperature controller has higher performance as compared with non-temperature control, Company needs to understand that the noodle can store for a much longer time at a controlled temperature.
15. We can see that the old warehouse is much better than the new warehouse, 50% of the supply will come from the old warehouses. The company needs to focus on why the new warehouse not performing well and provide some marketing campaigns in new warehouses.
16. The mid and large capacity warehouse performance is good in the east zone, However, the number of warehouses is very less, Company can divert the product to the large east & mid-size east zone for some product quantity.
18. East zone 4 and east 5 are performing very well in overall all zone. Let the company can understand and increase the product quantity.
19. Lager & small Company-owned warehouse is less performing as compared to rented warehouse all(Small, large and mid), Company needs to understand the ground reality why this company owned large and medium warehouse not performing well.

3. Data Cleaning and Pre-processing

3.1 Removal of unwanted variables

We can see as per the correlation plot and the data information below are the variables we should remove

a) WH_Manager_ID – We already have one unique id for the warehouse the manger ID will not used
b) WH_established_year- The warehouse establishment year data is missing more than 40%, the variable will not leads to good model building.

c) storage_issue_reported_in_last_3_months – We can see the storage positive correlation with target variable, this will leads to model building inconsistency & we model will take us in different direction with 99% accuracy.

Please see the below data set after dropping 3 columns:

Now we have total 21 columns.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25000 entries, 0 to 24999
Data columns (total 21 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   WH_house_ID                           25000 non-null  object
 1   WH_Location_type                      25000 non-null  object
 2   WH_capacity_size                      25000 non-null  object
 3   zone                                  25000 non-null  object
 4   WH_regional_zone                      25000 non-null  object
 5   Number_of_refill_req_in_last_3_months 25000 non-null  int64
 6   Transport_issue_in_last_one_year      25000 non-null  int64
 7   Number_of_competitor_in_mkt           25000 non-null  int64
 8   Number_of_ratail_shop                 25000 non-null  int64
 9   WH_owner_type                         25000 non-null  object
 10  Number_of_distributor                 25000 non-null  int64
 11  WH_in_Flood_impacted_area             25000 non-null  int64
 12  WH_in_flood_proof_area                25000 non-null  int64
 13  generator_backup                      25000 non-null  int64
 14  Distance_bet_warehouse_&_production_hub 25000 non-null  int64
 15  Number_of_workers                     24010 non-null  float64
 16  WH_temp_regulat                       25000 non-null  int64
 17  approved_wh_govt_certificate          24092 non-null  object
 18  WH_breakdown_in_last_3_months         25000 non-null  int64
 19  government_audit_in_last_3_months     25000 non-null  int64
 20  Product_weight_in_ton                 25000 non-null  int64
dtypes: float64(1), int64(13), object(7)
memory usage: 4.0+ MB
```

Table 1.5 Data information -3

3.2 Missing Value treatment

As we already know as per data info 2 variables having the missing values:

Now let us check before dropping the columns, how many variables having the missing values.

wh_est_year 11881

workers_num 990

approved_wh_govt_certificate 908

As we already dropped the year of establishment column, Let see once again the missing values in the variables

Number_of_workers 990

approved_wh_govt_certificate 908

Now we have two variables with missing values, now let us treat the number of workers first:

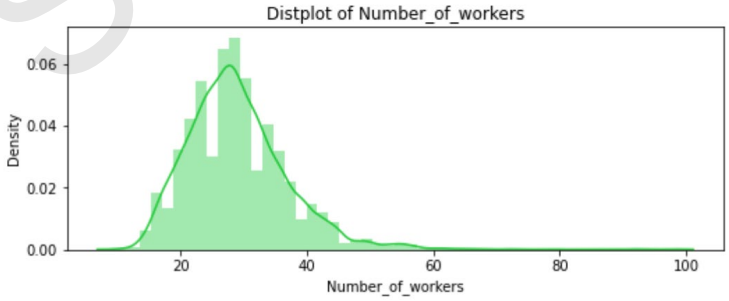


Fig.1.27 Missing Value treatment

We can see the worker number is uniformly distributed, we will use the mean for missing value treatment.

After treating the missing values let us keck again. Count of NULL values after imputation

approved_wh_govt_certificate 908, we cannot find the any missing values in worker number.

Let's we treat the approval warehouse government certificate variables:

We can see the all NA values and year of establishment, where the number year is missing the government approval is not available, it means the warehouse is under establishment and awaiting for government certification.
For considering this we will not remove the NaN but will replace with Not Available.
Let us final check the missing values again- Are there any missing values? – False.
Let us check the info again:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25000 entries, 0 to 24999
Data columns (total 21 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   WH_house_ID                             25000 non-null  object
 1   WH_WH_Location_type                     25000 non-null  object
 2   WH_capacity_size                        25000 non-null  object
 3   zone                                    25000 non-null  object
 4   WH_regional_zone                        25000 non-null  object
 5   Number_of_refill_req_in_last_3_months   25000 non-null  int64
 6   Transport_issue_in_last_one_year        25000 non-null  int64
 7   Number_of_competitor_in_mkt             25000 non-null  int64
 8   Number_of_ratail_shop                   25000 non-null  int64
 9   WH_owner_type                           25000 non-null  object
 10  Number_of_distributor                   25000 non-null  int64
 11  WH_in_Flood_impacted_area               25000 non-null  int64
 12  WH_in_WH_in_flood_proof_area_area       25000 non-null  int64
 13  generator_backup                        25000 non-null  int64
 14  Distance_bet_warehouse_&_production_hub 25000 non-null  int64
 15  Number_of_workers                       25000 non-null  float64
 16  WH_temp_regulat                         25000 non-null  int64
 17  approved_wh_govt_certificate            25000 non-null  object
 18  WH_breakdown_in_last_3_months           25000 non-null  int64
 19  government_audit_in_last_3_months       25000 non-null  int64
 20  Product_weight_in_ton                   25000 non-null  int64
dtypes: float64(1), int64(13), object(7)
memory usage: 4.0+ MB
```

Table 1.6 Data information -4

We can see the all the variables having 25000 data points.

3.3 Outlier treatment

We can see that in the boxplot we have outlier present in data set, Let us see the skewness of each variable and take the decision for outlier treatments-

```
Number_of_refill_req_in_last_3_months     -0.08
Transport_issue_in_last_one_year           1.61
Number_of_competitor_in_mkt                0.98
Number_of_ratail_shop                      0.91
Number_of_distributor                      0.02
WH_in_Flood_impacted_area                  2.70
WH_in_flood_proof_area                     3.92
generator_backup                          -0.66
Distance_bet_warehouse_&_production_hub   -0.01
Number_of_workers                          1.08
WH_temp_regulat                            0.86
WH_breakdown_in_last_3_months             -0.07
government_audit_in_last_3_months         -0.36
Product_weight_in_ton                      0.33
dtype: float64
```

Table 1.7 Outlier treatment

We can see the skewness of some variables in above the -0.5 to+0.5.

1. Transport issue in last one year- 1.61 - We will considered this for outlier treatment
2. Number of competitor in mkt- 0.98 - We will considered this for outlier treatment
3. Number of retail shop- 0.91 - We will considered this for outlier treatment
4. Number of workers- 1.08 - We will considered this for outlier treatment
5. WH in Flood impacted area- 2.70 – We will not considered because considered category data (1, 0)
6. WH in flood proof area 3.92– We will not considered because considered category data (1, 0)

7. Generator backup 0.66– We will not considered because considered category data (1, 0)

Let us see the data before outlier treatment:

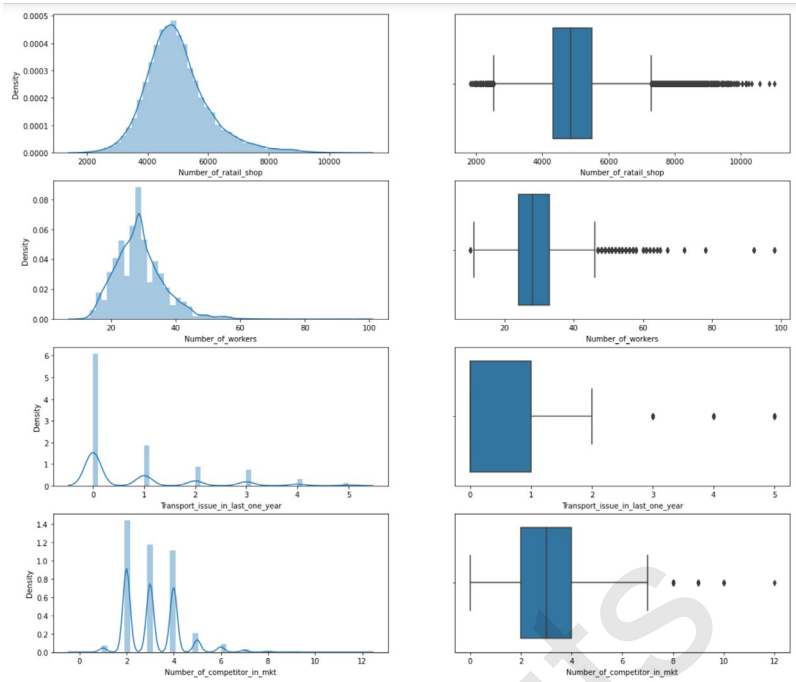


Fig.1.28 Data before outlier treatment

Let us see the data after outlier treatment:

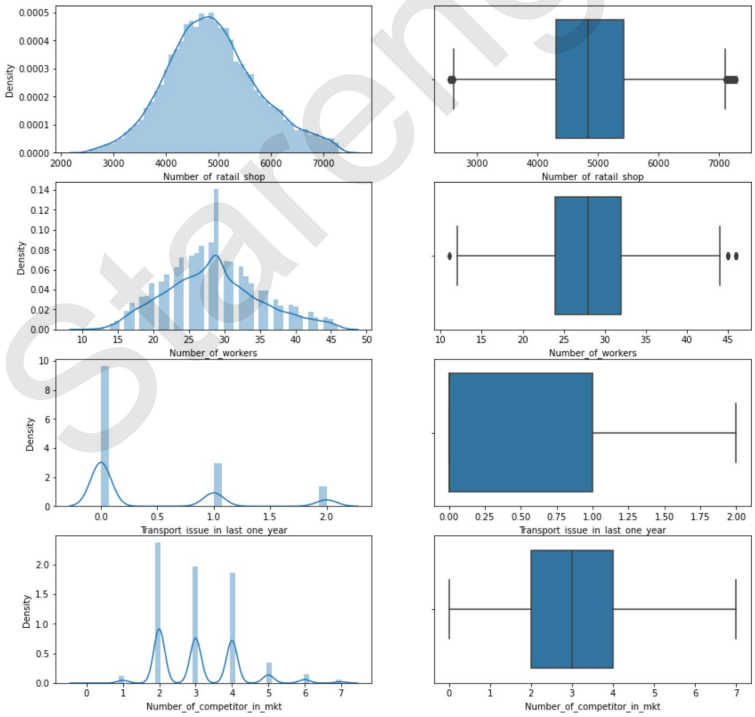


Fig.1.29 Data after outlier treatment

3.4 Variable transformation

We have not seen much variables for transformation, however the warehouse ID should be convert in Index

After the model building we need to inform the managements with warehouse ID.

Let us see the data head.

WH_house_ID	WH_WH_Location_type	WH_capacity_size	zone	WH_regional_zone	Number_of_refill_req_in_last_3_months	T
WH_100000	Urban	Small	West	Zone 6	3	
WH_100001	Rural	Large	North	Zone 5	0	
WH_100002	Rural	Mid	South	Zone 2	1	
WH_100004	Rural	Large	North	Zone 5	3	
WH_100005	Rural	Small	West	Zone 1	8	

Table 1.8 Variable transformation

3.5 Addition of new variables

Yes, there is possibility to create the number of new variables:

1. We have added the new column (Zone & WH Regional Zone)
   The combination of the both Column will give us the four zone with combination of 6 reginal zone
   We can see the mid capacity warehouse in east and larger capacity warehouse in east have higher median value, we can say that mid and large capacity warehouse performance good in east zone.

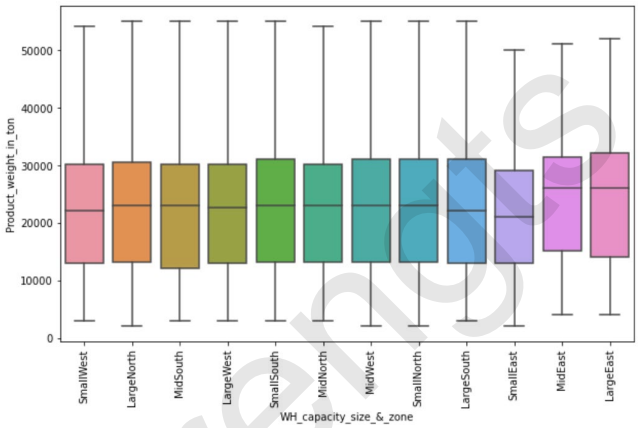


Fig.1.30 Addition of new variables-1

2. The second column with (WH Capacity size & Zone)
   The combination of the both column will give us small, large and mid-capacity with zone wise.

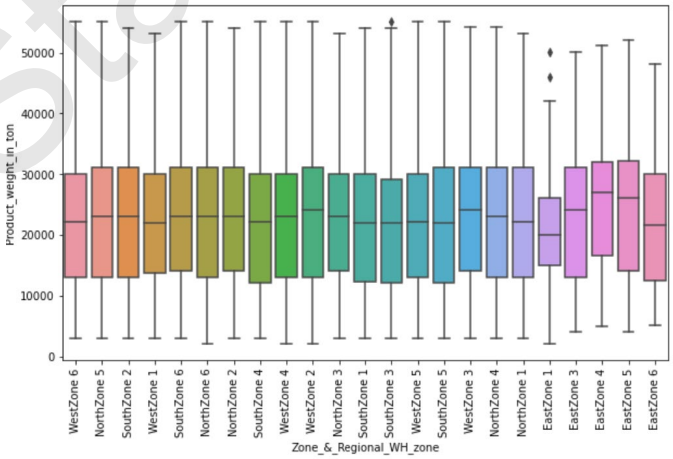


Fig.1.31 Addition of new variables-2

We can see the east zone 4 and east zone 5 have the highest median values.

3. The third variables is WH_capacity_size_&_WH_owner_type, it's the combination of WH_capacity_size and WH_owner_type.

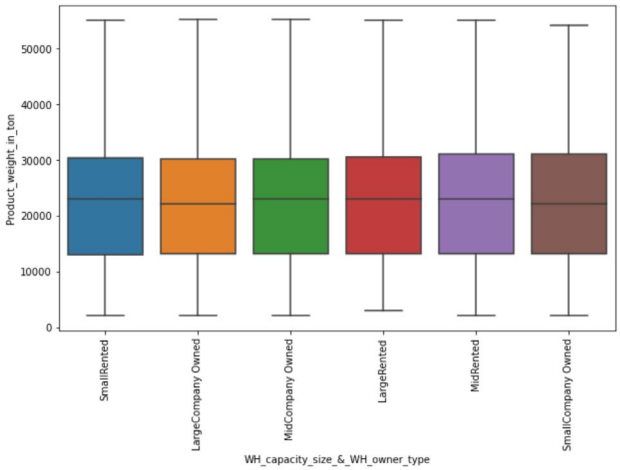


Fig.1.32 Addition of new variables-3

We can see the Lager & small Company owned warehouse is less performing the as compared to rented warehouse (Small, large and mid)

Let us see the final variable information: now we have 23 columns

```
<class 'pandas.core.frame.DataFrame'>
Index: 20626 entries, WH_100000 to WH_124999
Data columns (total 23 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   WH_Location_type                      20626 non-null  object
 1   WH_capacity_size                      20626 non-null  object
 2   zone                                  20626 non-null  object
 3   WH_capacity_size_&_zone               20626 non-null  object
 4   WH_regional_zone                      20626 non-null  object
 5   Zone_&_Regional_WH_zone               20626 non-null  object
 6   Number_of_refill_req_in_last_3_months 20626 non-null  int64
 7   Transport_issue_in_last_one_year      20626 non-null  int64
 8   Number_of_competitor_in_mkt           20626 non-null  int64
 9   Number_of_ratail_shop                 20626 non-null  int64
 10  WH_owner_type                         20626 non-null  object
 11  WH_capacity_size_&_WH_owner_type      20626 non-null  object
 12  Number_of_distributor                 20626 non-null  int64
 13  WH_in_Flood_impacted_area             20626 non-null  int64
 14  WH_in_flood_proof_area                20626 non-null  int64
 15  generator_backup                      20626 non-null  int64
 16  Distance_bet_warehouse_&_production_hub 20626 non-null  int64
 17  Number_of_workers                     20626 non-null  float64
 18  WH_temp_regulat                       20626 non-null  int64
 19  approved_wh_govt_certificate          20626 non-null  object
 20  WH_breakdown_in_last_3_months         20626 non-null  int64
 21  government_audit_in_last_3_months     20626 non-null  int64
 22  Product_weight_in_ton                 20626 non-null  int64
dtypes: float64(1), int64(13), object(9)
memory usage: 4.3+ MB
```

Table 1.9 Data information -5

3.6 Log transformation for target variables

We are dealing with continuous data set and not worry about the data unbalanced.

However if you see the target variable distribution, it is right skewness and that is highest performing warehouse and we have very less number of data point to learn the model, Hence the data is unbalance in nature.

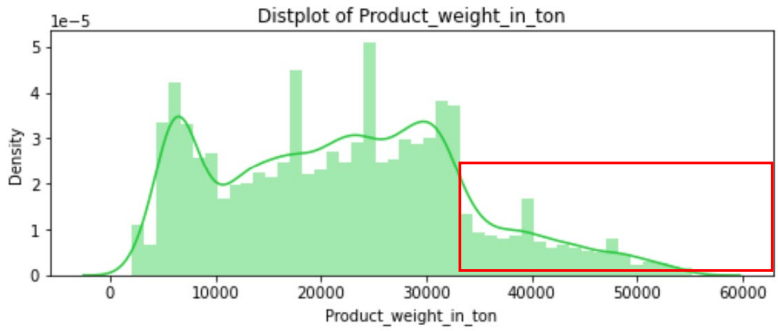


Fig.1.33 Business insights -1

We can do the log transformation to overcome this problems of data unbalanced.

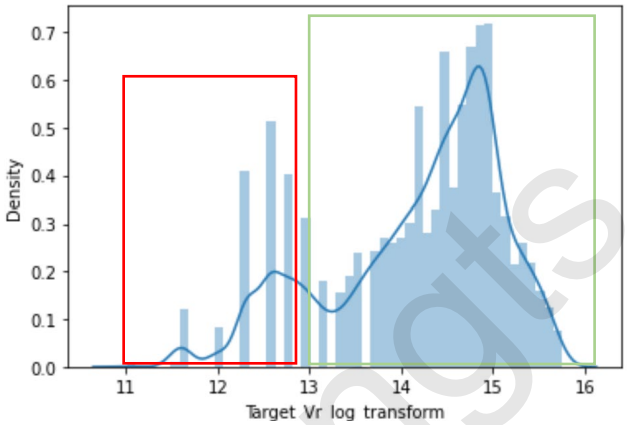


Fig.1.34 Business insights -2

We can see the after log transform we are achieving almost normal distribution.
Only expect some values in left side, because we have the outlier in lower side only. Where the warehouse performance is lesser.
In context of business if we not do the log transform of balance the data, the model will committing error on highly performing warehouse then the loss will be higher.

4. Model building

- As we have seen that in project note-1 the data variable doesn't have good correlation with target variables.
- The only one good negative correlation we found in the establishment year, However we have dropped in the project note -1
- Considering the good correlation we are taking the variables in model building, However we are assuming , if we not consider establishment year the model will perform very poorly
- As we know this is the regression problem and we have to predict the product weights in tonnes
- Below changes we have done in project note -1
    - All columns are rename correctly
    - Missing values treated
    - Manger ID and storage issue reported in last 3 months columns dropped

- For model building we are keeping the year of establishment columns, as we know this variable have 40% missing values and correlation with target variables, hence we are dropping all the missing values and remain dataset will be used for model building.
- Originally the shape of data set is (25000, 24) with 600000 data points after the dropping the missing values (11578, 21) with 439964 data points

Before building model let us check if data set ready for it:

- We can see the same category (Object) variables available in the data set, Let us convert into the integer variables

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25000 entries, 0 to 24999
Data columns (total 24 columns):
 #   Column                                     Non-Null Count  Dtype
---  ------                                     --------------  -----
 0   WH_house_ID                                25000 non-null  object
 1   WH_Manager_ID                              25000 non-null  object
 2   WH_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   Number_of_refill_req_in_last_3_months      25000 non-null  int64
 7   Transport_issue_in_last_one_year           25000 non-null  int64
 8   Number_of_competitor_in_mkt                25000 non-null  int64
 9   Number_of_ratail_shop                      25000 non-null  int64
 10  WH_owner_type                              25000 non-null  object
 11  Number_of_distributor                      25000 non-null  int64
 12  WH_in_Flood_impacted_area                  25000 non-null  int64
 13  WH_in_flood_proof_area                     25000 non-null  int64
 14  generator_backup                           25000 non-null  int64
 15  Distance_bet_warehouse_and_production_hub  25000 non-null  int64
 16  Number_of_workers                          24010 non-null  float64
 17  WH_established_year                        13119 non-null  float64
 18  storage_issue_reported_in_last_3_months    25000 non-null  int64
 19  WH_temp_regulat                            25000 non-null  int64
 20  approved_wh_govt_certificate               24092 non-null  object
 21  WH_breakdown_in_last_3_months              25000 non-null  int64
 22  government_audit_in_last_3_months          25000 non-null  int64
 23  Product_weight_in_ton                      25000 non-null  int64
dtypes: float64(2), int64(14), object(8)
memory usage: 4.6+ MB
```

Table 2.1 – Data set info before label encoding

- We will use the Label Encoding for the this data set as maximum is label variable
- Now we can see that the variables is data type is changed, However few more columns is added because of label encoding

```
<class 'pandas.core.frame.DataFrame'>
Index: 11578 entries, WH_100004 to WH_124999
Data columns (total 38 columns):
 #   Column                                          Non-Null Count  Dtype
---  ------                                          --------------  -----
 0   Number_of_refill_req_in_last_3_months           11578 non-null  int64
 1   Transport_issue_in_last_one_year                11578 non-null  int64
 2   Number_of_competitor_in_mkt                     11578 non-null  int64
 3   Number_of_ratail_shop                           11578 non-null  int64
 4   Number_of_distributor                           11578 non-null  int64
 5   WH_in_Flood_impacted_area                       11578 non-null  int64
 6   WH_in_flood_proof_area                          11578 non-null  int64
 7   generator_backup                                11578 non-null  int64
 8   Distance_bet_warehouse_and_production_hub       11578 non-null  int64
 9   Number_of_workers                               11578 non-null  float64
 10  WH_established_year                             11578 non-null  float64
 11  WH_temp_regulat                                 11578 non-null  int64
 12  WH_breakdown_in_last_3_months                   11578 non-null  int64
 13  government_audit_in_last_3_months               11578 non-null  int64
 14  Product_weight_in_ton                           11578 non-null  int64
 15  WH_Location_type_Rural                          11578 non-null  uint8
 16  WH_Location_type_Urban                          11578 non-null  uint8
 17  WH_capacity_size_Large                          11578 non-null  uint8
 18  WH_capacity_size_Mid                            11578 non-null  uint8
 19  WH_capacity_size_Small                          11578 non-null  uint8
 20  zone_East                                       11578 non-null  uint8
 21  zone_North                                      11578 non-null  uint8
 22  zone_South                                      11578 non-null  uint8
 23  zone_West                                       11578 non-null  uint8
 24  WH_regional_zone_Zone_1                         11578 non-null  uint8
 25  WH_regional_zone_Zone_2                         11578 non-null  uint8
 26  WH_regional_zone_Zone_3                         11578 non-null  uint8
 27  WH_regional_zone_Zone_4                         11578 non-null  uint8
 28  WH_regional_zone_Zone_5                         11578 non-null  uint8
 29  WH_regional_zone_Zone_6                         11578 non-null  uint8
 30  WH_owner_type_Company_Owned                     11578 non-null  uint8
 31  WH_owner_type_Rented                            11578 non-null  uint8
 32  approved_wh_govt_certificate_A                  11578 non-null  uint8
 33  approved_wh_govt_certificate_A+                 11578 non-null  uint8
 34  approved_wh_govt_certificate_B                  11578 non-null  uint8
 35  approved_wh_govt_certificate_B+                 11578 non-null  uint8
 36  approved_wh_govt_certificate_C                  11578 non-null  uint8
 37  approved_wh_govt_certificate_Not_Applicable     11578 non-null  uint8
dtypes: float64(2), int64(13), uint8(23)
memory usage: 1.7+ MB
```

Table 2.2 – Data set info after label encoding

Let us create the target variable:

- We will drop the target variable "Product_weight_in_ton" column from the original dataframe and create the "X"
- Create the "Y" with only target variable "Product_weight_in_ton"
- Let us see the "X" dataset

WH_house_ID	Number_of_refill_req_in_last_3_months	Transport_issue_in_last_one_year	Number_of_competitor_in_mkt	Number_of_ratail_shop
WH_100004	3	1	2	4740
WH_100005	8	0	2	5053
WH_100006	8	0	4	4449
WH_100008	8	1	4	5381
WH_100010	7	1	3	4623

Table 2.3 – Data set after separating the target variable

- Let us see the "Y" target variables

```
WH_house_ID
WH_100004    24071
WH_100005    32134
WH_100006    30142
WH_100008    18082
WH_100010    21125
Name: Product_weight_in_ton, dtype: int64
```

Table 2.4 – Data set with only target variable

Let us split the dataset into train & test:

- We are considering the 70:30 for train:test
- Please refer the Jupyer note book for library used
- After splitting the data set we will fit the train and test data set into the model
- Before fitting the data into the model let us discuss about the model

4.1 Model Selection (Clear on why a particular model was chosen):

- Considering the regression problems we will select the regression model only
- As many model available we used only 4 model for this data set
- The model selected as below :
  - Linear Regression
  - ridge (Regularization – Liner regression)
  - lasso (Regularization – Liner regression)
  - Elastic net (Regularization – Liner regression)
  - Decision Tree Regression
  - Random Forest Regression
  - ANN Regression

Build The Model:

- We are using the Ensemble modelling, However we also done the linear regression separately
- Let us see the train & test data set

WH_house_ID	Number_of_refill_req_in_last_3_months	Transport_issue_in_last_one_year	Number_of_competitor_in_mkt	Number_of_ratail_shop
WH_117895	5	0	2	5343
WH_113019	8	0	3	4260
WH_104759	5	0	3	5040
WH_115213	6	0	3	4116
WH_107816	8	0	4	4227

Table 2.5 – Training data set

- We can see that the test set having the 3474 rows and 37 columns

WH_house_ID	Number_of_refill_req_in_last_3_months	Transport_issue_in_last_one_year	Number_of_competitor_in_mkt	Number_of_ratail_shop
WH_120579	6	1	2	4952
WH_122667	6	0	3	4677
WH_119064	4	0	5	4854
WH_114121	6	0	2	4816
WH_124177	5	0	4	5826

Table 2.6 – Testing data set

- We can see that the train set having the 8104rows and 37 columns
- Let us first invoke the Linear Regression function model.fit (X train and X test)
- As mentioned we cannot show the code in business reports, it's difficult to the show the model running, Please refer the jupyter note book
- Let us run the OSL & see the output results
- We will combine the X train & test to run the Linear Regression (Ordinary Least Squares
- Let us see the OLS output

OLS Regression Results

Dep. Variable:	Product_weight_in_ton	R-squared:	0.708
Model:	OLS	Adj. R-squared:	0.707
Method:	Least Squares	F-statistic:	675.3
Date:	Sat, 11 Jun 2022	Prob (F-statistic):	0.00
Time:	22:22:21	Log-Likelihood:	-82590.
No. Observations:	8104	AIC:	1.652e+05
Df Residuals:	8074	BIC:	1.655e+05
Df Model:	29		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	9.481e+05	7764.585	122.111	0.000	9.33e+05	9.63e+05
Number_of_refill_req_in_last_3_months	-19.6247	41.771	-0.470	0.639	-101.508	62.258
Transport_issue_in_last_one_year	-1304.2529	132.625	-9.834	0.000	-1564.233	-1044.273
Number_of_competitor_in_mkt	16.8541	73.406	0.230	0.818	-127.040	160.748
Number_of_ratail_shop	0.1034	0.085	1.212	0.226	-0.064	0.271
Number_of_distributor	3.7249	4.476	0.832	0.405	-5.049	12.498
WH_in_Flood_impacted_area	-234.7761	255.541	-0.919	0.358	-735.702	266.150
WH_in_flood_proof_area	343.2177	330.566	1.038	0.299	-304.777	991.212
generator_backup	13.4866	167.413	0.081	0.936	-314.687	341.660
Distance_bet_warehouse_and_production_hub	0.3535	1.156	0.306	0.760	-1.912	2.619
Number_of_workers	2.4646	11.689	0.211	0.833	-20.448	25.377
WH_established_year	-1292.3543	10.654	-121.303	0.000	-1313.239	-1271.470
WH_temp_regulat	1598.3330	156.807	10.193	0.000	1290.952	1905.714
WH_breakdown_in_last_3_months	239.3515	48.890	4.896	0.000	143.514	335.189
government_audit_in_last_3_months	1.6020	9.620	0.167	0.868	-17.255	20.459
WH_Location_type_Rural	4.739e+05	3892.241	121.758	0.000	4.66e+05	4.82e+05
WH_Location_type_Urban	4.742e+05	3876.700	122.328	0.000	4.67e+05	4.82e+05
WH_capacity_size_Large	2.915e+05	2394.266	121.749	0.000	2.87e+05	2.96e+05
WH_capacity_size_Mid	3.647e+05	2987.270	122.077	0.000	3.59e+05	3.71e+05
WH_capacity_size_Small	2.92e+05	2393.764	121.970	0.000	2.87e+05	2.97e+05
zone_East	2.37e+05	2002.105	118.369	0.000	2.33e+05	2.41e+05
zone_North	2.372e+05	1948.530	121.724	0.000	2.33e+05	2.41e+05
zone_South	2.369e+05	1951.052	121.436	0.000	2.33e+05	2.41e+05
zone_West	2.37e+05	1951.339	121.478	0.000	2.33e+05	2.41e+05
WH_regional_zone_Zone_1	1.944e+05	1613.914	120.475	0.000	1.91e+05	1.98e+05
WH_regional_zone_Zone_2	1.219e+05	1008.007	120.916	0.000	1.2e+05	1.24e+05
WH_regional_zone_Zone_3	1.214e+05	1011.779	119.961	0.000	1.19e+05	1.23e+05
WH_regional_zone_Zone_4	1.214e+05	1008.287	120.421	0.000	1.19e+05	1.23e+05
WH_regional_zone_Zone_5	1.946e+05	1606.237	121.157	0.000	1.91e+05	1.98e+05
WH_regional_zone_Zone_6	1.944e+05	1596.867	121.752	0.000	1.91e+05	1.98e+05
WH_owner_type_Company_Owned	4.741e+05	3883.562	122.069	0.000	4.66e+05	4.82e+05
WH_owner_type_Rented	4.741e+05	3882.470	122.108	0.000	4.66e+05	4.82e+05
approved_wh_govt_certificate_A	763.1517	212.032	3.599	0.000	347.515	1178.789
approved_wh_govt_certificate_B	-1787.5341	209.076	-8.550	0.000	-2197.377	-1377.691
approved_wh_govt_certificate_C	-187.8857	199.946	-0.940	0.347	-579.832	204.061
approved_wh_govt_certificate_Not_Applicable	358.0626	445.133	0.804	0.421	-514.512	1230.638

Omnibus:	353.513	Durbin-Watson:	1.994
Prob(Omnibus):	0.000	Jarque-Bera (JB):	532.847
Skew:	0.402	Prob(JB):	1.97e-116
Kurtosis:	3.965	Cond. No.	1.01e+16

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 2.26e-21. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

Table 2.7 – OLS output

- Let us check the underroot of mean_sq_error is standard deviation i.e. avg variance between predicted and actual is 6482.08
- Let us check the model score training set – 71 %

- Let us check the model score test set – 70 %
- We will do the both model building and test the model in one go and use Ensemble modelling
- We can show both different out for model building and testing, Hence will show how we build (Note: please don't considered this is a code)
    - We use the regression model that is as LinearRegression
    - We use ridge for regularization of linear regression is as Ridge
    - We use ridge for regularization of linear regression lasso is as Lasso
    - We use ridge for regularization of linear regression elastic net is as ElasticNet
    - We use artificial neural network annr as a MLPRegressor with random state
    - We use rfr as a Random Forest Regressor with random state
    - We use dtr as a tree Decision Tree Regressor with random state
- Please refer the jupyter notebook
- We do the scaling the model for only for ANNR
- After assign all model we call the fit function for all model together

Test your predictive model against the test set using various appropriate performance metrics

- We are going to use the RMSE and test accuracy to test the prediction for all model
- As we building the all model as random will not get accurate results and call for model tuning
- Let us see the results before the model tuning

	Train RMSE	Test RMSE	Training Score	Test Score
Linear Regression	6370.00	6482.08	0.72	0.70
ridge	6370.00	6482.09	0.72	0.70
lasso	6370.03	6481.76	0.72	0.70
elastic_net	6478.43	6588.33	0.71	0.69
Decision Tree Regressor	0.00	8495.30	1.00	0.49
Random Forest Regressor	2293.78	6131.03	0.96	0.73
ANN Regressor	9712.86	9700.82	0.34	0.33

Table 2.8 – Model output with model tuning

Interpretation of the model(s)

- We build the 7 model 3 model for Liner regression regularization like ridge, lasso & Elastic net
- We can see the good r square and adjusted r square value in OLS model-0.707 & 0.708
- All the leaner regression model is constant, However need to model tuning to check the better performance
- The decision tree is over fitted and need to do model tuning, We can see the 0 RMSE in Train data
- The random forest is also over fitted to do model tuning

4.2 Effort to improve model performance

❖ **Liner regression regularization**

- Let us see the cross validation scores using the CV – 5 and scoring –r2
    - Output
        - CV Mean: 0.7129562297548979
        - STD: 0.010339546149871846

❖ **Decision Tree Regression**

a. We use the Grid Search CV for model tuning
    i. Parameter used in grid are as below

1. Max depth
2. Min sample leaf
3. Min sample split

b. After doing number of iteration we found that the model is fit for below values
   i. Final model tuned Parameter used
      1. Max depth - 7
      2. Min sample leaf - 30
      3. Min sample split - 300
      4. CV – 3

❖ **Random Forest Regression**
- We use the Grid Search CV for model tuning
  - Parameter used in grid are as below
    1. Max depth
    2. Max features
    3. Min sample leaf
    4. Min sample split
    5. N Estimators
- After doing number of iteration we found that the model is fit for below values
  - Final model tuned Parameter used
    1. Max depth - 8
    2. Max features - 19
    3. Min sample leaf - 15
    4. Min sample split - 2
    5. N Estimators - 600

❖ **ANN Regression**
- We use the Grid Search CV for model tuning
  - Parameter used in grid are as below
    1. Hidden layer sizes
    2. Activation
    3. Solver
- After doing number of iteration we found that the model is fit for below values
  - Final model tuned Parameter used
    1. Hidden layer sizes - 900
    2. Activation - relu
    3. Solver - adam

b) Model Regularization
- We will do the model tuning for all the model for liner regression we will use the regularization of model
- **Ridge Regression (L2 Regularization)**
  - We using alpha values from 0.001, 0.01, 0.1, 1, 10,50, 100, 500 till 1000
  - We use Grid Search CV with estimator is ridge, parameter grid, scoring is 'r2', verbose is 1, n jobs is 1.
    - Best Score: 0.7130505083583776
    - Best Parameter: alpha: 100

- **Lasso Regression (L1 Regularization)**
  - We using alpha values from 0.001, 0.01, 0.1, 1, 10,50, 100, 500 till 1000
  - We use Grid Search CV with estimator is lasso, parameter grid, scoring is 'r2', verbose is 1, n jobs is 1.
    - Best Score: 0.7133146314107421
    - Best Parameter: alpha: 10

- As we know a Lasso regression uses L1 regularization to force some coefficients to be exactly zero, this means some features are completely ignored by the model. This can be thought of as a type of automatic feature selection.
- Lasso can be a good model choice when we have a large number of features but expect only a few to be important. This can make the model easier to interpret and reveal the most important features.
- Higher values of α force more coefficients to zero and can cause under fitting.
- Lower values of alpha lead to fewer non-zero features and can cause overfitting. Very low values of alpha will cause the model to resemble linear regression.
- We used the default value for alpha above, which might not give the best performance. The optimum value of alpha will vary with each dataset.

We can see the lasso has been set many coefficient variables to zero which not making any sense for model output

Number_of_refill_req_in_last_3_months:	-19.08535656851533
Transport_issue_in_last_one_year:	-1314.3277572931613
Number_of_competitor_in_mkt:	20.9319838518536
Number_of_ratail_shop:	0.0929353389180115
Number_of_distributor:	3.170720883342406
WH_in_Flood_impacted_area:	-220.56095438533052
WH_in_flood_proof_area:	236.73985280862738
generator_backup:	9.217050481625982
Distance_bet_warehouse_and_production_hub:	0.25075352570841725
Number_of_workers:	3.6884501368133074
WH_established_year:	-1282.5201716120705
WH_temp_regulat:	313.62546101564817
WH_breakdown_in_last_3_months:	225.68752094246014
government_audit_in_last_3_months:	2.7404371451698184
WH_Location_type_Rural:	-234.6591735211741
WH_Location_type_Urban:	0.0
WH_capacity_size_Large:	-105.28350360610763
WH_capacity_size_Mid:	-0.0
WH_capacity_size_Small:	359.9211640431977
zone_East:	-0.0
zone_North:	87.3540587847753
zone_South:	-111.71578480248671
zone_West:	1.6415766708063935
WH_regional_zone_Zone_1:	0.0
WH_regional_zone_Zone_2:	417.807168685946
WH_regional_zone_Zone_3:	-100.8326949214752
WH_regional_zone_Zone_4:	-0.0
WH_regional_zone_Zone_5:	119.96484073035593
WH_regional_zone_Zone_6:	-93.20900159844516
WH_owner_type_Company_Owned:	-63.134187686207156
WH_owner_type_Rented:	0.0
approved_wh_govt_certificate_A:	878.8434683809177
approved_wh_govt_certificate_A+:	2665.565457333243
approved_wh_govt_certificate_B:	-1664.0049619703236
approved_wh_govt_certificate_B+:	-1261.3787634365328
approved_wh_govt_certificate_C:	35.30246077457454
approved_wh_govt_certificate_Not_Applicable:	-0.0

Table 2.9 – Lasso model coefficient without log transfer -1

- **Elastic-Net Regression**

  Elastic-net is a linear regression model that combines the penalties of Lasso and Ridge.
  We use the l1_ratio parameter to control the combination of L1 and L2 regularization. When l1_ratio and = 0 we have L2 regularization (Ridge) and when l1_ratio = 1 we have L1 regularization (Lasso). Values between zero and one give us a combination of both L1 and L2 regularization.

We have already fitted elastic-net with default parameters now use grid search to find optimal values for alpha and l1_ratio.

- We using alpha values from 0.001, 0.01, 0.1, 1, 10,50, 100, 500 till 1000
- We use l1_ratio from 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1
- We use Grid Search CV with estimator is elastic net, parameter grid, scoring is 'r2', verbose is 5, n jobs is 1.
    - Best Score: 0.7133146314107421
    - Best Parameter: alpha: 10 & l1_ratio: 1

5. Model validation

Coefficient

We have also done the log transformation of target variables but all the data will get over fit

Let us see the lasso model –coefficient after target variable log transformation

```
Number_of_refill_req_in_last_3_months:  -19.08535656851533
Transport_issue_in_last_one_year:  -1314.3277572931613
Number_of_competitor_in_mkt:  20.9319838518536
Number_of_ratail_shop:  0.0929353389180115
Number_of_distributor:  3.170720883342406
WH_in_Flood_impacted_area:  -220.56095438533052
WH_in_flood_proof_area:  236.73985280862738
generator_backup:  9.217050481625982
Distance_bet_warehouse_and_production_hub:  0.25075352570841725
Number_of_workers:  3.6884501368133074
WH_established_year:  -1282.5201716120705
WH_temp_regulat:  313.62546101564817
WH_breakdown_in_last_3_months:  225.68752094246014
government_audit_in_last_3_months:  2.7404371451698184
WH_Location_type_Rural:  -234.6591735211741
WH_Location_type_Urban:  0.0
WH_capacity_size_Large:  -105.28350360610763
WH_capacity_size_Mid:  -0.0
WH_capacity_size_Small:  359.9211640431977
zone_East:  -0.0
zone_North:  87.3540587847753
zone_South:  -111.71578480248671
zone_West:  1.6415766708063935
WH_regional_zone_Zone_1:  0.0
WH_regional_zone_Zone_2:  417.807168685946
WH_regional_zone_Zone_3:  -100.8326949214752
WH_regional_zone_Zone_4:  -0.0
WH_regional_zone_Zone_5:  119.96484073035593
WH_regional_zone_Zone_6:  -93.20900159844516
WH_owner_type_Company_Owned:  -63.134187686207156
WH_owner_type_Rented:  0.0
approved_wh_govt_certificate_A:  878.8434683809177
approved_wh_govt_certificate_A+:  2665.565457333243
approved_wh_govt_certificate_B:  -1664.0049619703236
approved_wh_govt_certificate_B+:  -1261.3787634365328
approved_wh_govt_certificate_C:  35.30246077457454
approved_wh_govt_certificate_Not_Applicable:  -0.0
```

Table 2.10 – Lasso model coefficient without log transfer -2

```
Number_of_refill_req_in_last_3_months:  0.001961700517871504
Transport_issue_in_last_one_year:  -0.0016076979177894227
Number_of_competitor_in_mkt:  0.0
Number_of_ratail_shop:  2.8499367234810786e-06
Number_of_distributor:  4.933037172686993e-05
WH_in_Flood_impacted_area:  0.0
WH_in_flood_proof_area:  0.0
generator_backup:  -0.0
Distance_bet_warehouse_and_production_hub:  -9.534591919022607e-06
Number_of_workers:  -6.608204788399944e-05
WH_established_year:  -0.010018810418212381
WH_temp_regulat:  -0.0069180966947574
WH_breakdown_in_last_3_months:  0.019492293377144337
government_audit_in_last_3_months:  0.00014927654704123945
Product_weight_in_ton:  4.154583125104259e-05
WH_Location_type_Rural:  -0.009717692922452252
WH_Location_type_Urban:  0.0
WH_capacity_size_Large:  -0.0
WH_capacity_size_Mid:  0.0
WH_capacity_size_Small:  0.0
zone_East:  -0.0
zone_North:  0.0
zone_South:  -0.0
zone_West:  0.0
WH_regional_zone_Zone_1:  0.0
WH_regional_zone_Zone_2:  -0.0
WH_regional_zone_Zone_3:  -0.0
WH_regional_zone_Zone_4:  0.0014935743528176615
WH_regional_zone_Zone_5:  -0.0
WH_regional_zone_Zone_6:  -0.0
WH_owner_type_Company_Owned:  0.0
WH_owner_type_Rented:  -0.0
approved_wh_govt_certificate_A:  0.016513325848637222
approved_wh_govt_certificate_A+:  0.019734317588680984
approved_wh_govt_certificate_B:  -0.0
approved_wh_govt_certificate_B+:  0.0
approved_wh_govt_certificate_C:  -0.021744474455642475
approved_wh_govt_certificate_Not_Applicable:  -0.19695216520132722
```

Table 2.11 – Lasso model coefficient with log transfer of Y

We can see the difference before and after long transform of target variables the almost 28 variable set to zero.

Below are the result after target variable log transformation (RMSE & AccuracyAcore):

We can see all model seems overfired as we know that the all variables not having good correlation with target variables and only years of establishment having the good negative correlation hence after log transform of target variables the all model is over fitted

	Train RMSE	Test RMSE	Training Score	Test Score
Linear Regression	0.17	0.17	0.92	0.92
ridge	0.17	0.17	0.92	0.92
lasso	0.17	0.17	0.92	0.92
elastic_net	0.17	0.17	0.92	0.92
Decision Tree Regressor	0.00	0.00	1.00	1.00
Random Forest Regressor	0.00	0.00	1.00	1.00
ANN Regressor	0.05	0.09	0.99	0.98

Table 2.12 – Model output after log transformation of Y

Now let us see the final model output with test score:

After apply final parameters after model tuning below are the results –

	Train RMSE	Test RMSE	Training Score	Test Score
Linear Regression	6370.00	6482.08	0.72	0.70
ridge	6370.94	6483.66	0.72	0.70
lasso	6371.08	6481.78	0.72	0.70
elastic_net	6371.08	6481.78	0.72	0.70
Decision Tree Regressor	5951.54	6019.73	0.75	0.74
Random Forest Regressor	5569.27	6000.01	0.78	0.74
ANN Regressor	5766.11	6243.30	0.77	0.72

Table 2.13 – Final model output RMSE and test score

Below are the size of error in percentage for test RMSE:

Linear Regression Test –RMSE 28.820099689493965
Lasso Regression Test –RMSE 28.824985985830644
Elastic net Regression Test-RMSE 28.824985985830644
Decision Tree Regression Test-RMSE 26.926840832968747
Random Forest Regression Test-RMSE 25.187409824079758
ANN Regression Test-RMSE 26.012605739678946

We can easily see the most optimal and most consistent model both:

The lowest % error model is Random Forest model, lowest RMSE score and highest test accuracy – Most optimum model is Random Forest model

The Liner regression have slightly lower score however in most consistent in all regularization model

Interpretation of the most optimum model and its implication on the business

We can see the random forest is optimum model for current data set:

	Train RMSE	Test RMSE	Training Score	Test Score
Linear Regression	6370.00	6482.08	0.72	0.70
ridge	6370.94	6483.66	0.72	0.70
lasso	6371.08	6481.78	0.72	0.70
elastic_net	6371.08	6481.78	0.72	0.70
Decision Tree Regressor	5951.54	6019.73	0.75	0.74
Random Forest Regressor	5569.27	6000.01	0.78	0.74
ANN Regressor	5766.11	6243.30	0.77	0.72

Table 2.14 – Final model output RMSE and test score with optimum mode

Let us see the features explained by this model:

1. We can see the year of establishment is the only variable that will have an impact on product weight in tones
2. The other variable will have a very small impact on the product weight
3. The rest of all 28 variables will not have any impact on product weight
4. Reduction breakdown will improve the warehouse holding capacity
5. Converting more warehouse into the A+ certification will help to improve the storage capacity warehouse

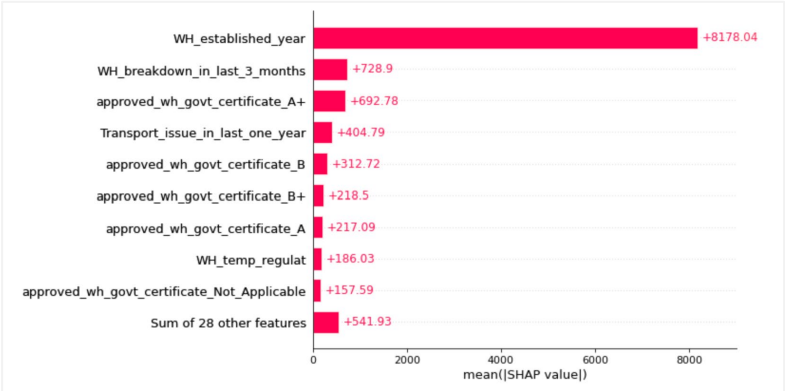


Fig.2.1- Optimum model output feature explained

6. Final interpretation / Recommendation:

1. The only correlation is available in data is year of establishment and model predicted values
2. The older warehouse performance is well company need to focus on new warehouse
3. below output highlighted in red is under performance warehouse – Company is not selling required product quantity to warehouse.
4. The warehouse highlighted in green is performing well but warehouse stored products more than required quantity. If there is higher demand in mid-size or small size Warehouse Company need to think about the increases the warehouse capacity

WH_house_ID	Product_weight_in_ton	Predicted Value	WH Performance
WH_100004	24071	24317	-246
WH_100005	32134	26626	5508
WH_100006	30142	22933	7209
WH_100008	18082	18490	-408
WH_100010	21125	30805	-9680
WH_100011	14115	13210	905
WH_100012	5124	7602	-2478
WH_100013	30063	25488	4575
WH_100018	38082	38630	-548
WH_100020	24062	17115	6947
WH_100021	17117	23495	-6378
WH_100022	40150	36829	3321
WH_100023	32098	33235	-1137

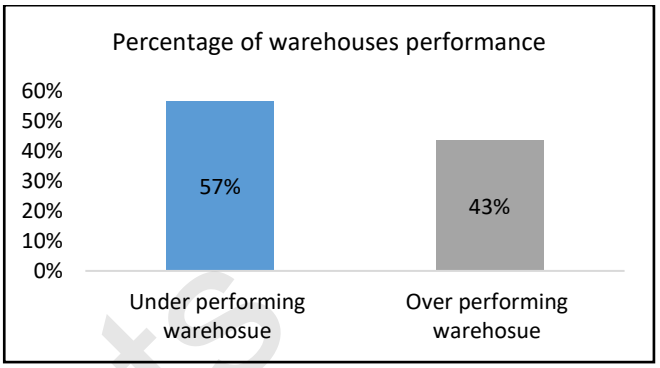


Table 2.15 – Product weight in tonnes model predicted output

5. The old warehouse performing well need to deploy same strategy on new warehouse
6. We can also see the small % of certification will also add the high product in warehouse
   To converting the more warehouse to A+ certification will add value
7. The mid and large capacity warehouse performance good in east zone, However the number of warehouse is very less, Company can divert the product to large east & mid-size east zone some product quantity.
8. East zone 4 and east 5 is performing the very good in overall all zone. Let company can understand and increase the product quantity.
9. Lager & small Company owned warehouse is less performing as compare to rented warehouse all(Small, large and mid),Company needs to understand the ground reality why this company owned large and medium warehouse not performing well.

- ❑ 57% Warehouse having the opportunity to increase the output (Underperforming)
- ❑ 43% Warehouse has the opportunity to reassess the capacity (Over performing)
- ❑ If the company listen to this model output then the output can be increased by 25 Million product weight in tones ~20% in the case of an underperformance warehouse (Diff- between the sum of predicted –the sum of actual, Underperforming)
- ❑ In case of an underperforming warehouse need to send extra item & should also support them by creating more marketing campaign, more marketing strategy and appointing more distributors & reseller.
- ❑ As the instant noodles business is not a specific geographical or any area-specific business, we can do a marketing campaign to improve the business in low performance warehouses.
- ❑ The company need to reassess the warehouse capacity to reduce the storage issues & capacity is increases the number of refills will come down with respect to transportation will also come down (In this case the overall cost of the supply chain will come down)

======= Thank you for support ======

===END===