Supply Chain Project
Final Report

Name: Varsha Srinivasan

Table of Contents

Problem 1: Capstone Project – Supply Chain

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

Link for Tableau -
https://public.tableau.com/app/profile/varsha.srinivasan/viz/SupplyChain_16632473116430/Zone-TotalNumberofWorkers?publish=yes

https://public.tableau.com/views/SupplyChain2_16655245777580/zone-floodimpact?:language=en-US&publish=yes&:display_count=n&:origin=viz_share_link

FINAL REPORT

1. Introduction

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.

Need for 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. Also to analysis the demand pattern in different pockets of the country so management can drive the advertisement campaign particular in those pockets.

Understanding business/social opportunity

- The information from the analysis can be used to predict number of workers required to manage the warehouse according to the value of warehouse size, weight of product shipped, etc.
- Set up of warehouse on the basis of zone, regions and distance from hub can be studied and done .
- We can identify the zones in which less number of retail shops sell the product under the warehouse area. Promotions can be done for the same.

2. EDA and Business Implication

	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
0	WH_100000	EID_50000	Urban	Small	West	Zone 6	3	1	2
1	WH_100001	EID_50001	Rural	Large	North	Zone 5	0	0	4
2	WH_100002	EID_50002	Rural	Mid	South	Zone 2	1	0	4
3	WH_100003	EID_50003	Rural	Mid	North	Zone 3	7	4	2
4	WH_100004	EID_50004	Rural	Large	North	Zone 5	3	1	2

retail_shop_num	wh_owner_type	distributor_num	flood_impacted	flood_proof	electric_supply	dist_from_hub	workers_num	wh_est_year
4651	Rented	24	0	1	1	91	29.0	NaN
6217	Company Owned	47	0	0	1	210	31.0	NaN
4306	Company Owned	64	0	0	0	161	37.0	NaN
6000	Rented	50	0	0	0	103	21.0	NaN
4740	Company Owned	42	1	0	1	112	25.0	2009.0

storage_issue_reported_l3m	temp_reg_mach	approved_wh_govt_certificate	wh_breakdown_l3m	govt_check_l3m	product_wg_ton
13	0	A	5	15	17115
4	0	A	3	17	5074
17	0	A	6	22	23137
17	1	A+	3	27	22115
18	0	C	6	24	24071

Table 1: Sample of the Dataset

The data is read from the csv file and the above tables shows the first 5 rows of the dataset.The data has been collected from the year 1996 until 2023 over 28 years span. From the year 1998-2021 data of above 460 warehouses was collected each year. After the year 2021 the number of data collected on Warehouses has decreased. The number of rows (observations) is 25000. The number of columns (variables) is 24. product_wg_ton is the target variable. Since it is a numeric value regression is performed.

	Ware_house_ID	WH_Manager_ID	Location_type	WH_capacity_size	zone	WH_regional_zone	num_refill_req_l3m	transport_issue_l1y
count	25000	25000	25000	25000	25000	25000	25000.000000	25000.000000
unique	25000	25000	2	3	4	6	NaN	NaN
top	WH_100000	EID_50000	Rural	Large	North	Zone 6	NaN	NaN
freq	1	1	22957	10169	10278	8339	NaN	NaN
mean	NaN	NaN	NaN	NaN	NaN	NaN	4.089040	0.773680
std	NaN	NaN	NaN	NaN	NaN	NaN	2.606612	1.199449
min	NaN	NaN	NaN	NaN	NaN	NaN	0.000000	0.000000
25%	NaN	NaN	NaN	NaN	NaN	NaN	2.000000	0.000000
50%	NaN	NaN	NaN	NaN	NaN	NaN	4.000000	0.000000
75%	NaN	NaN	NaN	NaN	NaN	NaN	6.000000	1.000000
max	NaN	NaN	NaN	NaN	NaN	NaN	8.000000	5.000000

Competitor_in_mkt	retail_shop_num	wh_owner_type	distributor_num	flood_impacted	flood_proof	electric_supply	dist_from_hub	workers_num
25000.000000	25000.000000	25000	25000.000000	25000.000000	25000.000000	25000.000000	25000.000000	24010.000000
NaN	NaN	2	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	Company Owned	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	13578	NaN	NaN	NaN	NaN	NaN	NaN
3.104200	4985.711560	NaN	42.418120	0.098160	0.054640	0.656880	163.537320	28.944398
1.141663	1052.825252	NaN	16.064329	0.297537	0.227281	0.474761	62.718609	7.872534
0.000000	1821.000000	NaN	15.000000	0.000000	0.000000	0.000000	55.000000	10.000000
2.000000	4313.000000	NaN	29.000000	0.000000	0.000000	0.000000	109.000000	24.000000
3.000000	4859.000000	NaN	42.000000	0.000000	0.000000	1.000000	164.000000	28.000000
4.000000	5500.000000	NaN	56.000000	0.000000	0.000000	1.000000	218.000000	33.000000
12.000000	11008.000000	NaN	70.000000	1.000000	1.000000	1.000000	271.000000	98.000000

wh_est_year	storage_issue_reported_l3m	temp_reg_mach	approved_wh_govt_certificate	wh_breakdown_l3m	govt_check_l3m	product_wg_ton
13119.000000	25000.000000	25000.000000	24092	25000.000000	25000.000000	25000.000000
NaN	NaN	NaN	5	NaN	NaN	NaN
NaN	NaN	NaN	C	NaN	NaN	NaN
NaN	NaN	NaN	5501	NaN	NaN	NaN
2009.383185	17.130440	0.303280	NaN	3.482040	18.812280	22102.632920
7.528230	9.161108	0.459684	NaN	1.690335	8.632382	11607.755077
1996.000000	0.000000	0.000000	NaN	0.000000	1.000000	2065.000000
2003.000000	10.000000	0.000000	NaN	2.000000	11.000000	13059.000000
2009.000000	18.000000	0.000000	NaN	3.000000	21.000000	22101.000000
2016.000000	24.000000	1.000000	NaN	5.000000	26.000000	30103.000000
2023.000000	39.000000	1.000000	NaN	6.000000	32.000000	55151.000000

Table 2:Dataset Description

Ware_house_ID and WH_Manager_ID can be dropped because they have 25000 unique values and it isn't of any use in the analysis. The most type of standard certificate has been issued to the warehouse from government regulatory body is C and most of the data is about warehouses from rural areas. The average number of time warehouse face a breakdown in last 3 months is around 3. 163 Kms is the mean distance of between warehouse to the production hub. On an average 22102 tons of products has been shipped in the last 3 months.

Understanding of attributes

```
<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:Dataset Info

There are 16 numeric and 8 object data types according to the info. Very few of the variables have missing values since they have less than 25000 non null values. electric_supply, flood_proof, flood_impacted, temp_reg_mach can be considered of categorical variables since they consist of only 0s and 1s.

```
Number of duplicate rows = 0
```

There aren't any duplicate rows in the dataset

Univariate Analysis

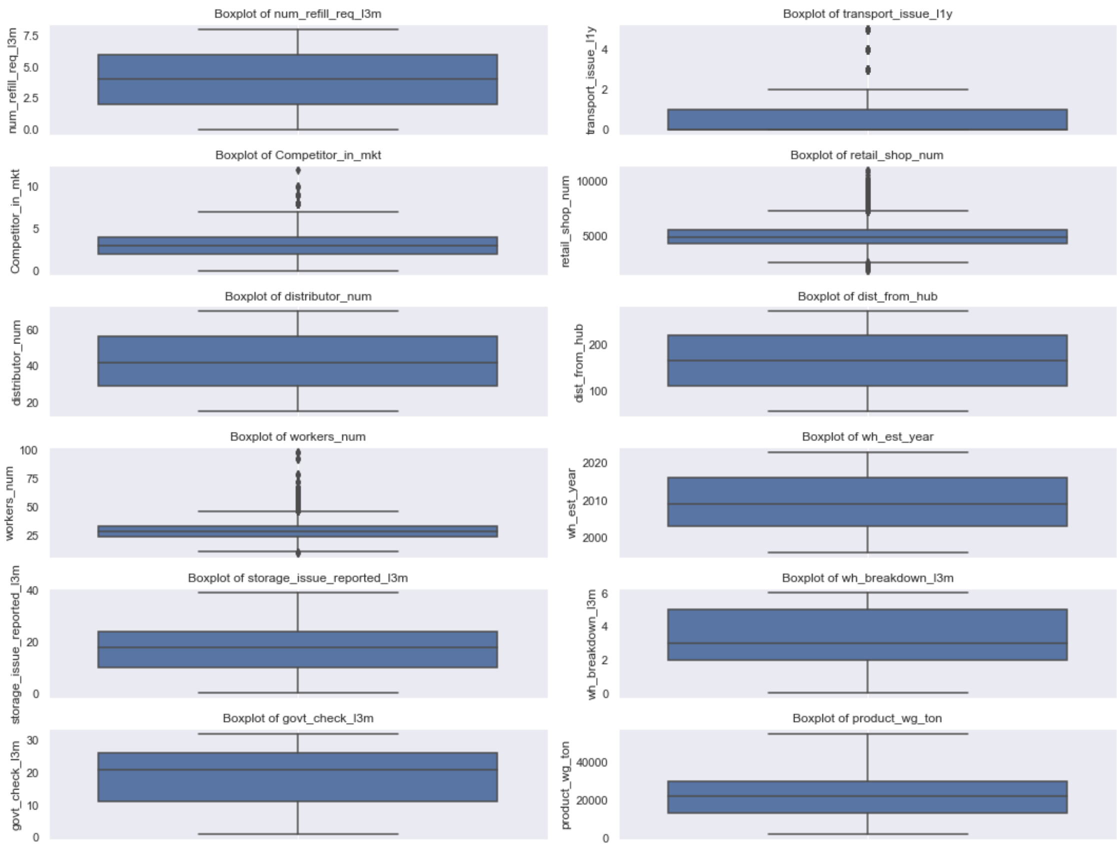

Figure 1:Boxplot with Outliers

transport_issue_l1y, workers_num, Competitor_in_mkt, retail_shop_num are the variables with outliers.

Description of num_refill_req_l3m

count 25000.000000
mean 4.089040
std 2.606612
min 0.000000
25% 2.000000
50% 4.000000
75% 6.000000
max 8.000000
Name: num_refill_req_l3m, dtype: float64

Interquartile range (IQR) of is 4.0
Range of values: 8

Distribution of num_refill_req_l3m

From Figure 12 we can infer that there are around 2900 warehouses with no refills in the past 3 months.

Description of transport_issue_l1y

```
count    25000.000000
mean         0.773680
std          1.199449
min          0.000000
25%          0.000000
50%          0.000000
75%          1.000000
max          5.000000
Name: transport_issue_l1y, dtype: float64
```

Interquartile range (IQR) of is 1.0
Range of values: 5

Distribution of transport_issue_l1y

From Figure 13 we can infer that most of the warehouses doesn’t have any transport issue like accident or goods stolen reported in last one year.

Description of Competitor_in_mkt

```
count    25000.000000
mean         3.104200
std          1.141663
min          0.000000
25%          2.000000
50%          3.000000
75%          4.000000
max         12.000000
Name: Competitor_in_mkt, dtype: float64
```

Interquartile range (IQR) of is 2.0
Range of values: 12

Distribution of Competitor_in_mkt

From Figure 14 we can infer that the mean number of competitors in the market is 3.

Description of retail_shop_num

count   25000.000000
mean   4985.711560
std   1052.825252
min   1821.000000
25%   4313.000000
50%   4859.000000
75%   5500.000000
max   11008.000000
Name: retail_shop_num, dtype: float64

Interquartile range (IQR) of is 1187.0
Range of values: 9187

Distribution of retail_shop_num

From Figure 15 we can infer that the average number of retails shop who sell the product under the warehouse area is 4985. The distribution is slightly right skewed.

Description of distributor_num

count   25000.000000
mean   42.418120
std   16.064329
min   15.000000
25%   29.000000
50%   42.000000
75%   56.000000
max   70.000000
Name: distributor_num, dtype: float64

Interquartile range (IQR) of is 27.0
Range of values: 55

Distribution of distributor_num

From Figure 16 we can infer that the average number of distributors works in between warehouse and retail shops is 42.

Description of dist_from_hub

```
count    25000.000000
mean       163.537320
std         62.718609
min         55.000000
25%        109.000000
50%        164.000000
75%        218.000000
max        271.000000
Name: dist_from_hub, dtype: float64
```

Interquartile range (IQR) of is 109.0
Range of values: 216

Distribution of dist_from_hub

From Figure 17 we can infer that the average distance is 163 Kms from warehouse to the production hub.

Description of workers_num

```
count    24010.000000
mean        28.944398
std          7.872534
min         10.000000
25%         24.000000
50%         28.000000
75%         33.000000
max         98.000000
Name: workers_num, dtype: float64
```

Interquartile range (IQR) of is nan
Range of values: 88.0

Distribution of workers_num

From Figure 18 we can infer that the average number of workers working in the warehouse is 28. The distribution is slightly positive skewed.

Description of wh_est_year

count   13119.000000
mean   2009.383185
std   7.528230
min   1996.000000
25%   2003.000000
50%   2009.000000
75%   2016.000000
max   2023.000000
Name: wh_est_year, dtype: float64

Interquartile range (IQR) of is nan
Range of values: 27.0

Distribution of wh_est_year

From Figure 19 we can infer that the most of the data is collected from the year 2010.

Description of storage_issue_reported_l3m

count   25000.000000
mean   17.130440
std   9.161108
min   0.000000
25%   10.000000
50%   18.000000
75%   24.000000
max   39.000000
Name: storage_issue_reported_l3m, dtype: float64

Interquartile range (IQR) of is 14.0
Range of values: 39

Distribution of storage_issue_reported_l3m

From Figure 20 we can infer that the average number of times storage issues are reported to corporate office in last 3 months like rat, fungus because of moisture etc is 17 .

Description of wh_breakdown_l3m

count   25000.000000
mean   3.482040
std   1.690335
min   0.000000
25%   2.000000

```
50%        3.000000
75%        5.000000
max        6.000000
Name: wh_breakdown_l3m, dtype: float64

Interquartile range (IQR) of is  3.0
Range of values:  6
```

Distribution of wh_breakdown_l3m

From Figure 21 we can infer that There are very less number of no breakdown events in the warehouses. Most of the warehouses have faced breakdowns 2 or 3 times in the last 3 months.

Description of govt_check_l3m

```
count    25000.000000
mean        18.812280
std          8.632382
min          1.000000
25%         11.000000
50%         21.000000
75%         26.000000
max         32.000000
Name: govt_check_l3m, dtype: float64

Interquartile range (IQR) of is  15.0
Range of values:  31
```

Distribution of govt_check_l3m

From Figure 22 we can infer that the average number of times government Officers have been visited the warehouse is 18 to check the quality and expire of stored food in last 3 months.

Description of product_wg_ton

```
count    25000.000000
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
```

Interquartile range (IQR) of is 17044.0
Range of values: 53086

Distribution of product_wg_ton

From Figure 23 we can infer that the product_wg_ton is nearly symmetrically distributed.

Value Count of Location_type

Rural 22957
Urban 2043
Name: Location_type, dtype: int64

Description of Location_type

count 25000
unique 2
top Rural
freq 22957
Name: Location_type, dtype: object

Countplot of Location_type

From Figure 24 we can infer that the least number of warehouses are there in Urban location.

Value Count of WH_capacity_size

Large 10169
Mid 10020
Small 4811
Name: WH_capacity_size, dtype: int64

Description of WH_capacity_size

count 25000
unique 3
top Large
freq 10169
Name: WH_capacity_size, dtype: object

Countplot of WH_capacity_size

From Figure 25 we can infer that the most of the warehouses are large-sized.

Value Count of zone

North 10278
West 7931
South 6362
East 429
Name: zone, dtype: int64

Description of zone

count 25000
unique 4
top North
freq 10278
Name: zone, dtype: object

Countplot of zone

From Figure 26 we can infer that the North zone has most number of warehouses while Ease zones has least number of warehouses.

Value Count of WH_regional_zone

Zone 6 8339
Zone 5 4587
Zone 4 4176
Zone 2 2963
Zone 3 2881
Zone 1 2054
Name: WH_regional_zone, dtype: int64

Description of WH_regional_zone

count 25000
unique 6
top Zone 6
freq 8339
Name: WH_regional_zone, dtype: object

Countplot of WH_regional_zone

From Figure 27 we can infer that the less number of warehouses are in Zone 1.

Value Count of wh_owner_type

Company Owned 13578
Rented 11422

Name: wh_owner_type, dtype: int64

Description of wh_owner_type

count 25000
unique 2
top Company Owned
freq 13578
Name: wh_owner_type, dtype: object

Countplot of wh_owner_type

From Figure 28 we can infer that the most of the warehouses (13578) are Company owned.

Value Count of approved_wh_govt_certificate

C 5501
B+ 4917
B 4812
A 4671
A+ 4191
Name: approved_wh_govt_certificate, dtype: int64

Description of approved_wh_govt_certificate

count 24092
unique 5
top C
freq 5501
Name: approved_wh_govt_certificate, dtype: object

Countplot of approved_wh_govt_certificate

From Figure 29 we can infer that the most of the warehouses are C certified by the government

Value Count of electric_supply

1 16422
0 8578
Name: electric_supply, dtype: int64

Description of electric_supply

count 25000.000000

mean 0.656880
std 0.474761
min 0.000000
25% 0.000000
50% 1.000000
75% 1.000000
max 1.000000
Name: electric_supply, dtype: float64

Countplot of electric_supply

From Figure 30 we can infer that the 16422 warehouses have back up for electricity like generators.

Value Count of flood_proof

0 23634
1 1366
Name: flood_proof, dtype: int64

Description of flood_proof

count 25000.000000
mean 0.054640
std 0.227281
min 0.000000
25% 0.000000
50% 0.000000
75% 0.000000
max 1.000000
Name: flood_proof, dtype: float64

Countplot of flood_proof

From Figure 31 we can infer that the around 23600 warehouses aren't flood proof.

Value Count of flood_impacted

0 22546
1 2454
Name: flood_impacted, dtype: int64

Description of flood_impacted

count 25000.000000
mean 0.098160
std 0.297537
min 0.000000
25% 0.000000
50% 0.000000
75% 0.000000
max 1.000000
Name: flood_impacted, dtype: float64

Countplot of flood_impacted

From Figure 32 we can infer that the most of the warehouse areas (around 22500) aren’t impacted by floods.

Value Count of temp_reg_mach

0 17418

1 7582

Name: temp_reg_mach, dtype: int64

Description of temp_reg_mach

count 25000.000000
mean 0.303280
std 0.459684
min 0.000000
25% 0.000000
50% 0.000000
75% 1.000000
max 1.000000
Name: temp_reg_mach, dtype: float64

Countplot of temp_reg_mach

From Figure 33 we can infer that the most of the warehouses (17418) doesn’t have a temperature regulator.

Skewness and Kurtosis

```
Skewness of num_refill_req_l3m is -0.08
Kurtosis of num_refill_req_l3m is -1.22
Skewness of transport_issue_l1y is 1.61
Kurtosis of transport_issue_l1y is 1.84
Skewness of Competitor_in_mkt is 0.98
Kurtosis of Competitor_in_mkt is 1.79
Skewness of retail_shop_num is 0.91
Kurtosis of retail_shop_num is 1.85
Skewness of distributor_num is 0.02
Kurtosis of distributor_num is -1.19
Skewness of dist_from_hub is -0.01
Kurtosis of dist_from_hub is -1.2
Skewness of workers_num is 1.06
Kurtosis of workers_num is 3.41
Skewness of wh_est_year is 0.01
Kurtosis of wh_est_year is -1.18
Skewness of storage_issue_reported_l3m is 0.11
Kurtosis of storage_issue_reported_l3m is -0.68
Skewness of wh_breakdown_l3m is -0.07
Kurtosis of wh_breakdown_l3m is -0.95
Skewness of govt_check_l3m is -0.36
Kurtosis of govt_check_l3m is -1.06
Skewness of product_wg_ton is 0.33
Kurtosis of product_wg_ton is -0.5
```

Skewness essentially measures the symmetry of the distribution. In positively skewed, the mean of the data is greater than the median as a large number of data-pushed on the right-hand side. In negatively skewed, the mean of the data is less than the median as a large number of data-pushed on the left-hand.

If the skewness is between -0.5 & 0.5, the data are nearly symmetrical. If the skewness is between -1 & -0.5 (negative/left skewed) or between 0.5 & 1(positive/right skewed), the data are slightly skewed. If the skewness is lower than -1 (negative/left skewed) or greater than 1 (positive/right skewed), the data are extremely skewed.

In this dataset product_wg_ton , govt_check_l3m, storage_issue_reported_l3m are symmetrically distributed.

Kurtosis refers to the degree of presence of outliers in the distribution. If kurtosis > 3, then it is called as Leptokurtic or heavy-tailed distribution as the kurtosis is more than normal distribution. If kurtosis = 3, then it is called as Mesokurtic as the kurtosis is same as the normal distribution. If kurtosis < 3, then it is called as Platykurtic or short-tailed distribution as the kurtosis is less than normal distribution.

Workers_num is Leptokurtic or heavy-tailed distribution. All other variables are Platykurtic or have short-tailed distribution.

Bivariate analysis

From Figure 34 we can infer that the on an average flood impacted areas has slightly less amount of product shipped in the last 3 months than not impacted areas.

From Figure 35 we can infer that the on an average, Mid size warehouses has slightly higher amount of product shipped in the last 3 months than other sizes.

From Figure 36 we can infer that the on an average, warehouses in the East Zone has slightly higher amount of product shipped in the last 3 months than warehouses in the other zones.

From Figure 37 we can infer that the on an average, warehouses in the Urban location has higher amount of product shipped in the last 3 months than warehouses in the Rural locations.

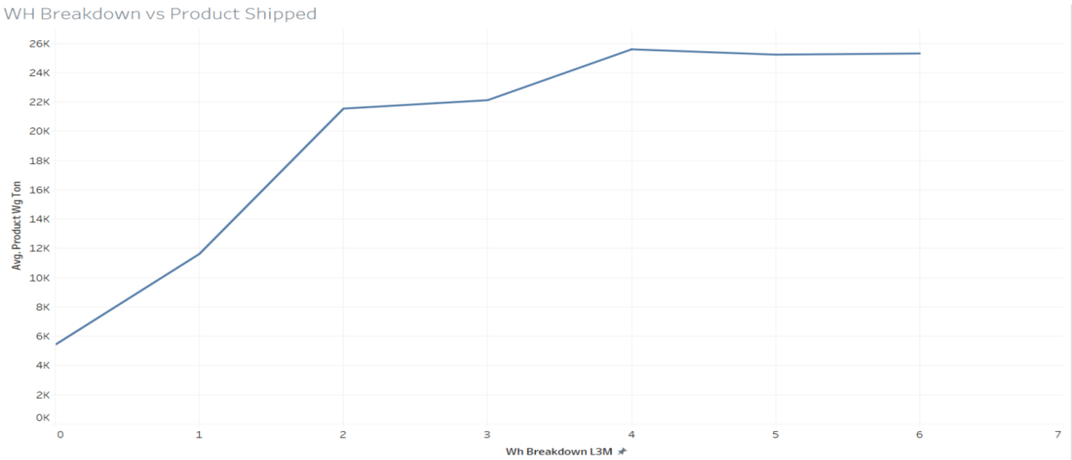

Figure 2:Lineplot of WH Breakdown and Product wg ton

From the above lineplot we can see that the average number of product shipped started to increase when the number of breakdown increases.

From Figure 38 we can infer that the lineplot we can say that the recently established warehouses have less total amount of products shipped compared to the time period 1998-2005.

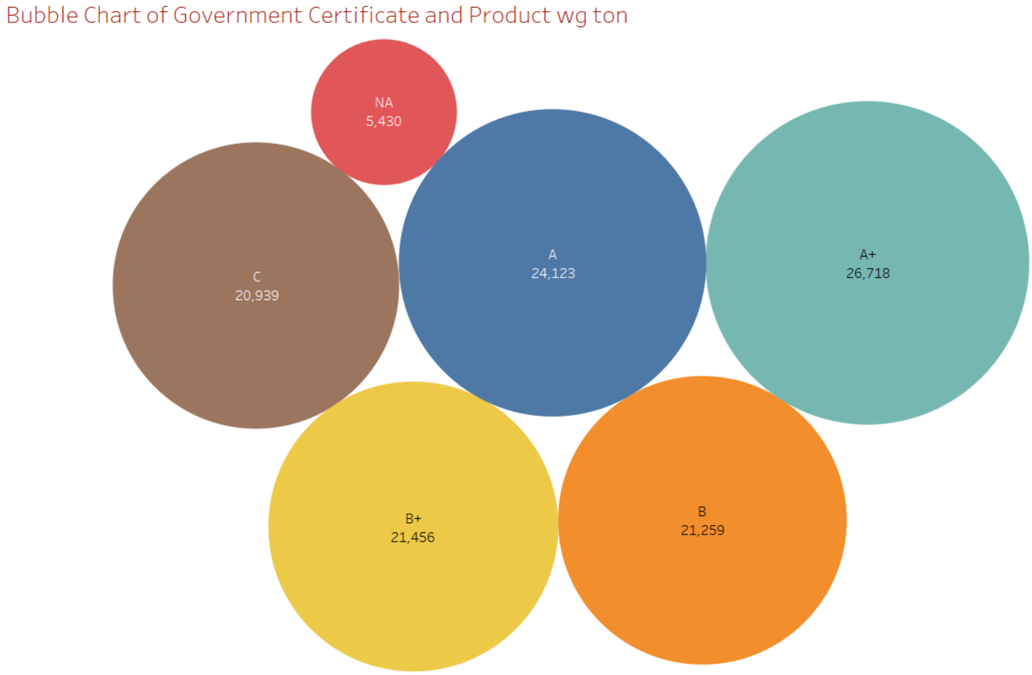

Figure 3:Bubble Chart of Government Certificate and Product wg ton

The more the average weight of product shipped in the last 3 months to the warehouse the higher it is certified. A+ certified warehouses has the highest average weight of product shipped. C certified warehouses has the least average weight of product shipped. 'NA' values are null values to be treated.

From Figure 39 we can infer that the Zone 2 has the most average weight of product shipped. Zone 1 has the least average weight of product shipped.

From Figure 40 we can infer that the rented warehouse has the slightly higher average weight of product shipped than the Company owned ones.

From Figure 41 we can infer that the North Zone has more number of workers in the warehouse.

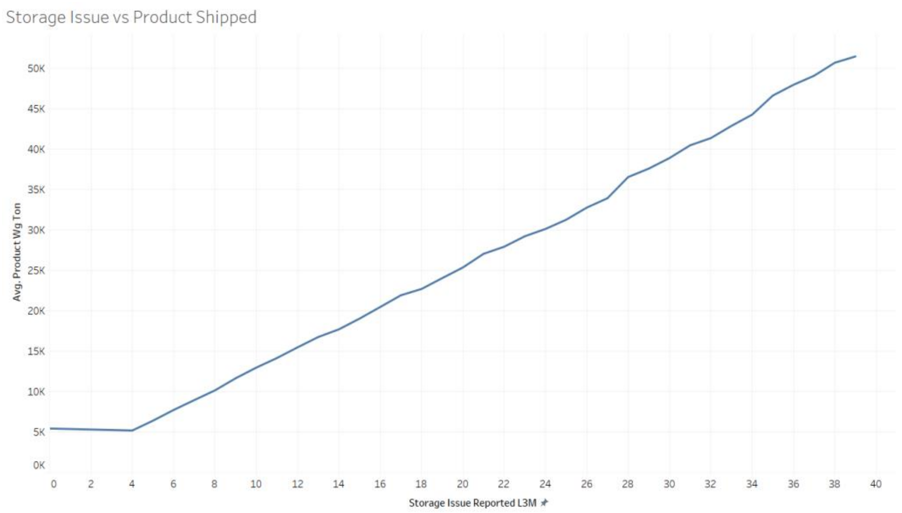

Figure 4: Lineplot of Storage Issue and Product Shipped

As the average amount of product shipped increases the storage issues increases too.

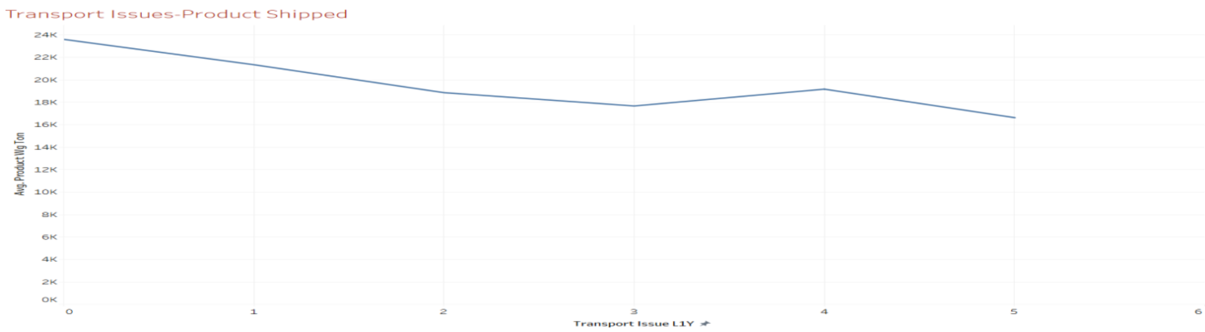

Figure 5:Lineplot of Transport Issue and Product Shipped

As the transport issue increases the average amount of product shipped decreases.

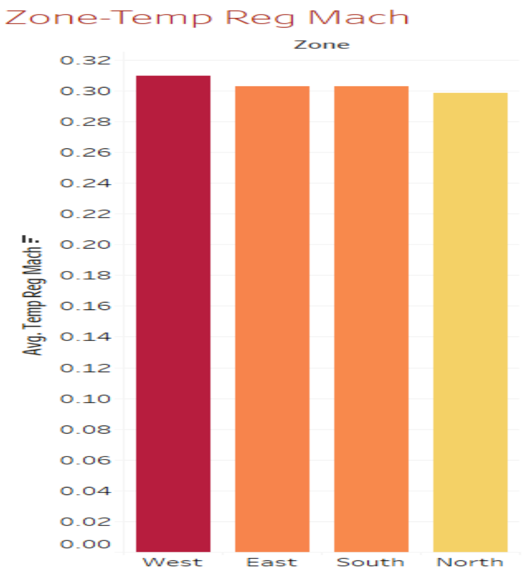

Figure 6:Barplot of Zone and Temp Reg Mach

West Zone has more warehouses with temperature regulating machine

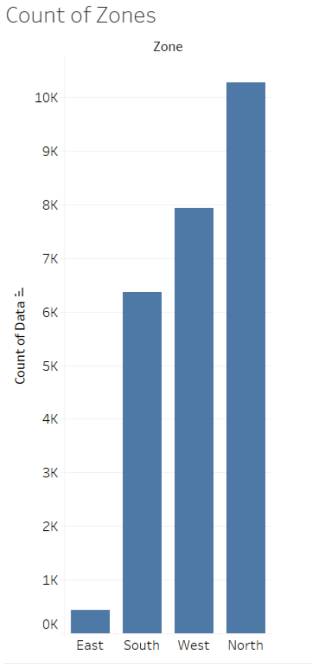

Figure 7:Count of Zones

Less number of warehouses in East Zone.

Multivariate Analysis

From Figure 42 we can infer that across all regions and zones most of the warehouses are C certified.

From Figure 43 we can infer that less number of small-sized warehouses are flood proof than mid and large sized ones.

From Figure 44 we can infer that warehouses in Urban areas with temperature regulators has higher average weight of product shipped than ones in Rural areas.

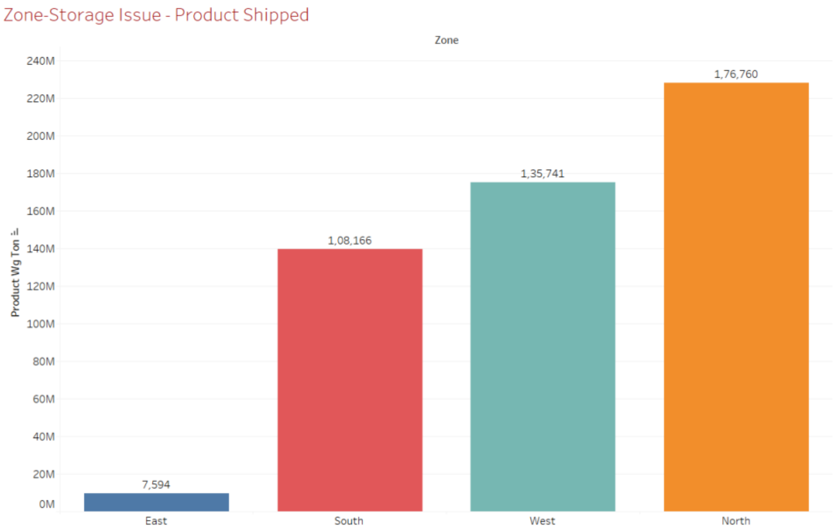

Figure 8:Barplot of Zone-Storage Issue -Product Shipped

East zone has less product shipped and storage issues.

Figure 9:Barplot of Zone-Retail Shop -Competitor

East Zone has least average number of retail shop selling the product and more average number of competitors.

Pairplot

From Figure 45 Pairplot we can infer that many of the features doesn't have a significant correlation.

HeatMap

From Figure 46 Heatmap we can infer that storage issue reported and product_wg_ton are highly poisitvely correlated.wh_est_year is highly negatively correlated with product_wg_ton and Storage issues reported which means that with successive years the values of the 2 variables are decreasing. The remaining variables are have a low correlation.

How your analysis is impacting the business?

- Storage issues reported are decreasing in successive years.
- Number of Warehouses established each year is decreasing.
- East Zone has less number of retail shops selling the product than other zones. More advertising can be done for the products in that zone.

3. Data Cleaning and Pre-processing

Removal of unwanted variables

Ware_house_ID and WH_Manager_ID can be dropped because they have 25000 unique values and it isn't of any use in the analysis.

Missing Value treatment

The inference from the data with missing values could adversely impact business decisions. Hence they are treated.

```
Location_type                      0
WH_capacity_size                   0
zone                               0
WH_regional_zone                   0
num_refill_req_l3m                 0
transport_issue_l1y                0
Competitor_in_mkt                  0
retail_shop_num                    0
wh_owner_type                      0
distributor_num                    0
flood_impacted                     0
flood_proof                        0
electric_supply                    0
dist_from_hub                      0
workers_num                      990
wh_est_year                    11881
storage_issue_reported_l3m         0
temp_reg_mach                      0
approved_wh_govt_certificate     908
wh_breakdown_l3m                   0
govt_check_l3m                     0
product_wg_ton                     0
dtype: int64
```

Workers_num, wh_est_year, approved_wh_govt_certificate are the columns with null values. The percentage of values missing in these columns are as below:

```
workers_num                      0.03960
wh_est_year                      0.47524
approved_wh_govt_certificate     0.03632
dtype: float64
```

Nearly 48% of the values of Wh_est_year is null and so it can be dropped as imputing it can change the dataset significantly. The other 2 variables have around 4% of their values missing and hence it is imputed. The remaining variables with missing values along with the number of missing values are:

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

Workers_num is a numerical column and it has outliers. Hence the missing values are imputed with median of Workers_num which is 28.

approved_wh_govt_certificate is a categorical column. Hence the missing values are imputed with mode (most frequent value) of approved_wh_govt_certificate which is 'C' certification.

After the treatments there aren't any null/missing values in the dataset.

```
Location_type                   0
WH_capacity_size                0
zone                            0
WH_regional_zone                0
num_refill_req_l3m              0
transport_issue_l1y             0
Competitor_in_mkt               0
retail_shop_num                 0
wh_owner_type                   0
distributor_num                 0
flood_impacted                  0
flood_proof                     0
electric_supply                 0
dist_from_hub                   0
workers_num                     0
storage_issue_reported_l3m      0
temp_reg_mach                   0
approved_wh_govt_certificate    0
wh_breakdown_l3m                0
govt_check_l3m                  0
product_wg_ton                  0
dtype: int64
```

Outlier treatment

They are often abnormal observations that skew the data distribution, and arise due to inconsistent data entry, or erroneous observations. To ensure that the trained model generalizes well to the valid range of test inputs, it's important to detect and remove outliers.

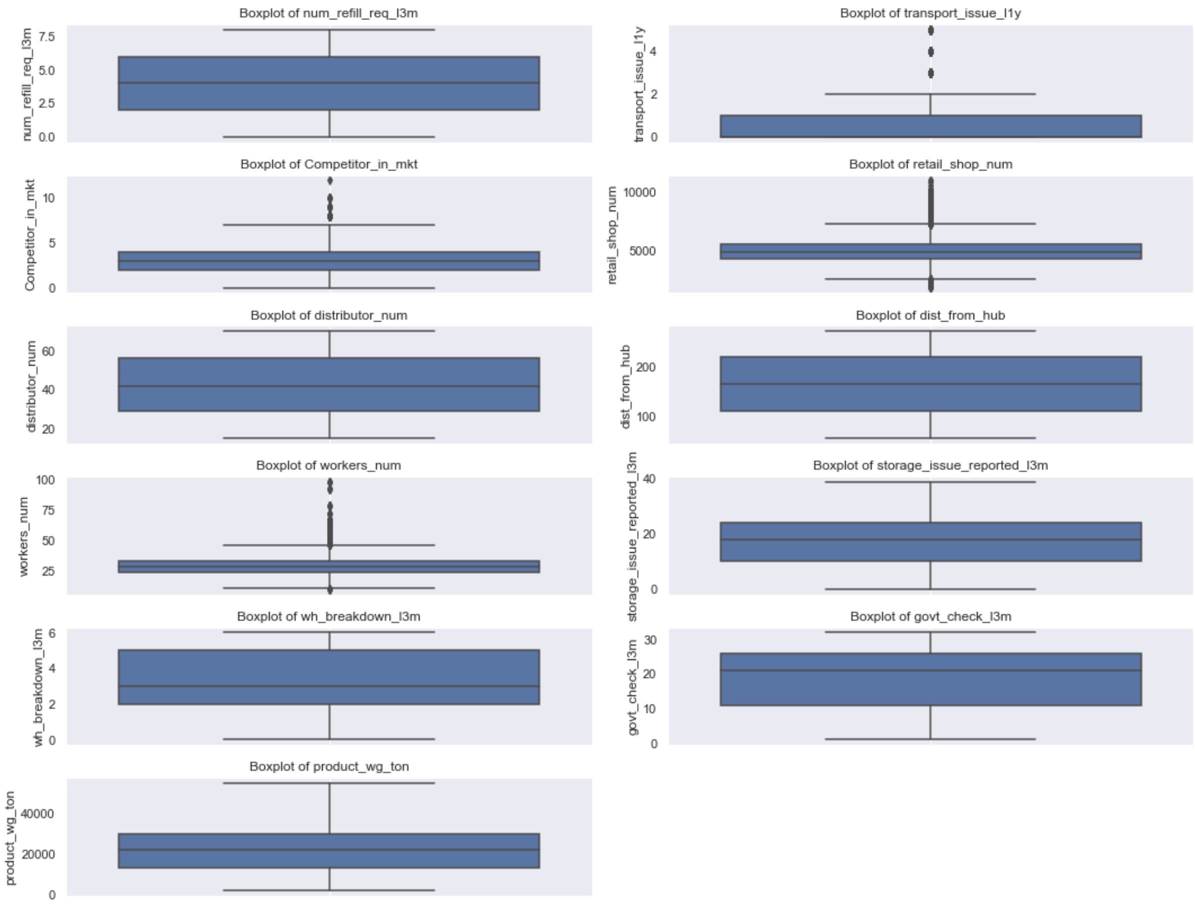

Figure 10:Boxplot with Outliers

transport_issue_l1y, workers_num, Competitor_in_mkt, retail_shop_num are the variables with outliers.

The values of upper bound, lower bound ,number of outliers and their proportion above the upper bound and below the lower bound for each numeric variable is given below:

```
Lower Bound in num_refill_req_l3m is :  -4.0
Upper Bound in num_refill_req_l3m is :  12.0
Number of outliers above num_refill_req_l3m upper bound :  0
Number of outliers below num_refill_req_l3m lower bound :  0
% of Outlier in num_refill_req_l3m upper:  0 %
% of Outlier in num_refill_req_l3m lower:  0 %
-----------------------------------------------------
Lower Bound in transport_issue_l1y is :  -1.5
Upper Bound in transport_issue_l1y is :  2.5
Number of outliers above transport_issue_l1y upper bound :  2943
Number of outliers below transport_issue_l1y lower bound :  0
% of Outlier in transport_issue_l1y upper:  12 %
% of Outlier in transport_issue_l1y lower:  0 %
-----------------------------------------------------
Lower Bound in Competitor_in_mkt is :  -1.0
Upper Bound in Competitor_in_mkt is :  7.0
Number of outliers above Competitor_in_mkt upper bound :  96
Number of outliers below Competitor_in_mkt lower bound :  0
% of Outlier in Competitor_in_mkt upper:  0 %
% of Outlier in Competitor_in_mkt lower:  0 %
-----------------------------------------------------
Lower Bound in retail_shop_num is :  2532.5
Upper Bound in retail_shop_num is :  7280.5
Number of outliers above retail_shop_num upper bound :  867
Number of outliers below retail_shop_num lower bound :  81
% of Outlier in retail_shop_num upper:  3 %
% of Outlier in retail_shop_num lower:  0 %
-----------------------------------------------------
Lower Bound in distributor_num is :  -11.5
Upper Bound in distributor_num is :  96.5
Number of outliers above distributor_num upper bound :  0
Number of outliers below distributor_num lower bound :  0
% of Outlier in distributor_num upper:  0 %
% of Outlier in distributor_num lower:  0 %
-----------------------------------------------------
Lower Bound in dist_from_hub is :  -54.5
Upper Bound in dist_from_hub is :  381.5
Number of outliers above dist_from_hub upper bound :  0
Number of outliers below dist_from_hub lower bound :  0
% of Outlier in dist_from_hub upper:  0 %
% of Outlier in dist_from_hub lower:  0 %
-----------------------------------------------------
Lower Bound in workers_num is :  10.5
Upper Bound in workers_num is :  46.5
Number of outliers above workers_num upper bound :  602
Number of outliers below workers_num lower bound :  5
% of Outlier in workers_num upper:  2 %
% of Outlier in workers_num lower:  0 %
-----------------------------------------------------
Lower Bound in storage_issue_reported_l3m is :  -11.0
Upper Bound in storage_issue_reported_l3m is :  45.0
Number of outliers above storage_issue_reported_l3m upper bound :  0
Number of outliers below storage_issue_reported_l3m lower bound :  0
```

```
% of Outlier in storage_issue_reported_l3m upper:  0 %
% of Outlier in storage_issue_reported_l3m lower:  0 %
----------------------------------------------------
Lower Bound in wh_breakdown_l3m is :  -2.5
Upper Bound in wh_breakdown_l3m is :  9.5
Number of outliers above wh_breakdown_l3m upper bound :  0
Number of outliers below wh_breakdown_l3m lower bound :  0
% of Outlier in wh_breakdown_l3m upper:  0 %
% of Outlier in wh_breakdown_l3m lower:  0 %
----------------------------------------------------
Lower Bound in govt_check_l3m is :  -11.5
Upper Bound in govt_check_l3m is :  48.5
Number of outliers above govt_check_l3m upper bound :  0
Number of outliers below govt_check_l3m lower bound :  0
% of Outlier in govt_check_l3m upper:  0 %
% of Outlier in govt_check_l3m lower:  0 %
----------------------------------------------------
Lower Bound in product_wg_ton is :  -12507.0
Upper Bound in product_wg_ton is :  55669.0
Number of outliers above product_wg_ton upper bound :  0
Number of outliers below product_wg_ton lower bound :  0
% of Outlier in product_wg_ton upper:  0 %
% of Outlier in product_wg_ton lower:  0 %
---------------------------------------------------
```

transport_issue_l1y need not be treated as they can denote rare incidents like accidents or goods stolen.

The outliers of workers_num, Competitor_in_mkt, retail_shop_num are treated by capping the outliers above the upper bound with the value of upper bound and outliers below the lower bound with the value of lower bound.

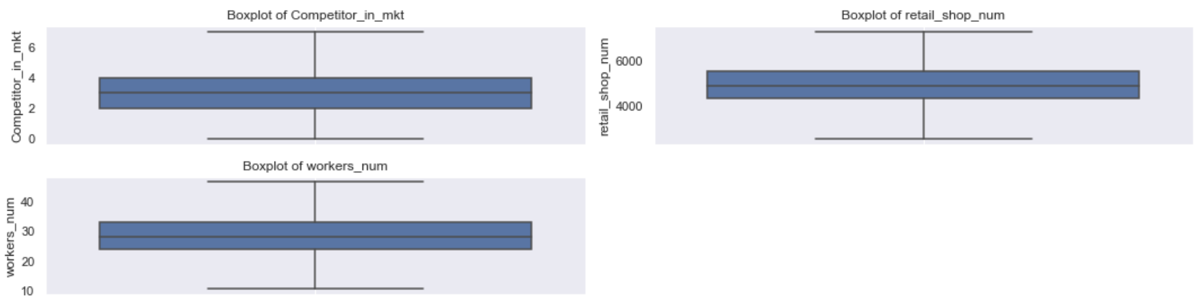

Figure 11: Boxplot - Outliers treated Variables

There aren't any outliers in those variables after treatment.

Variable Transformation

LabelEncoder is used to convert the labels into a numeric form so as to make them into machine-readable form. All categorical variables are encoded by Label Encoding.

The storage capacity size of the warehouse is encoded as follows:

```
{'Small': 0, 'Mid': 1, 'Large': 2}
```

The approved government certificate is encoded as follows:

```
{'C': 0, 'B': 1, 'B+': 2, 'A': 3, 'A+': 4}
```

The numerical variables can be transformed by Z-score if needed which is done after splitting the data into train and test.

4. Model building

Variables with VIF<5 and p-value<=0.05 are significant variables with less multicollinearity which are important to predicting the target variables. Such significant variables are used to build the model.

	variables	VIF
11	flood_proof	1.079536
0	Location_type	1.098069
10	flood_impacted	1.162842
5	transport_issue_l1y	1.441780
16	temp_reg_mach	1.702731
8	wh_owner_type	1.939171
17	approved_wh_govt_certificate	3.036388
12	electric_supply	3.519292
4	num_refill_req_l3m	3.659013
1	WH_capacity_size	5.119643
15	storage_issue_reported_l3m	5.477392
2	zone	5.492260
18	wh_breakdown_l3m	6.080037
19	govt_check_l3m	6.106202
3	WH_regional_zone	6.545343
13	dist_from_hub	7.236418
9	distributor_num	7.431246
6	Competitor_in_mkt	8.125715
14	workers_num	16.848394
7	retail_shop_num	18.969499

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	1286.9619	134.875	9.542	0.000	1022.593	1551.331
Location_type	-108.6778	49.271	-2.206	0.027	-205.254	-12.101
WH_capacity_size	-9.3670	21.336	-0.439	0.661	-51.187	32.453
zone	-2.9978	15.707	-0.191	0.849	-33.785	27.789
WH_regional_zone	-7.5969	9.512	-0.799	0.424	-26.242	11.048
num_refill_req_l3m	-0.3669	5.343	-0.069	0.945	-10.840	10.107
transport_issue_l1y	-310.9296	11.320	-27.466	0.000	-333.119	-288.740
Competitor_in_mkt	-7.7865	12.335	-0.631	0.528	-31.964	16.391
retail_shop_num	-0.0135	0.014	-0.957	0.338	-0.041	0.014
wh_owner_type	12.1975	27.965	0.436	0.663	-42.616	67.011
distributor_num	1.2187	0.835	1.460	0.144	-0.417	2.855
flood_impacted	17.6720	46.696	0.378	0.705	-73.858	109.202
flood_proof	56.2171	60.247	0.933	0.351	-61.873	174.307
electric_supply	9.3058	31.023	0.300	0.764	-51.502	70.114
dist_from_hub	0.2670	0.214	1.249	0.212	-0.152	0.686
workers_num	-0.2238	2.040	-0.110	0.913	-4.223	3.776
storage_issue_reported_l3m	1254.9724	1.624	772.843	0.000	1251.790	1258.155
temp_reg_mach	848.5634	31.632	26.826	0.000	786.562	910.565
approved_wh_govt_certificate	109.8782	10.130	10.846	0.000	90.021	129.735
wh_breakdown_l3m	-244.4290	8.624	-28.344	0.000	-261.332	-227.526
govt_check_l3m	-0.1288	1.655	-0.078	0.938	-3.374	3.116

Table 4: VIF, p-value and other measures

The significant variables are:

```
['flood_proof',
 'flood_impacted',
 'num_refill_req_l3m',
 'electric_supply',
 'wh_owner_type',
 'Location_type',
 'transport_issue_l1y',
 'storage_issue_reported_l3m',
 'approved_wh_govt_certificate',
 'temp_reg_mach',
```

```
'wh_breakdown_l3m']
```

Data is split into train and test set such that test data has 30% of data

The top 5 rows of train data is:

flood_proof	flood_impacted	num_refill_req_l3m	electric_supply	wh_owner_type	Location_type	transport_issue_l1y	storage_issue_reported_l3m
0	0	0	1	1	0	2	20
0	0	6	1	1	0	2	15
0	0	6	1	1	0	0	31
0	0	6	1	0	0	1	28
0	0	4	1	0	1	4	23

approved_wh_govt_certificate	temp_reg_mach	wh_breakdown_l3m
1	0	5
4	1	3
1	0	2
1	0	2
2	0	6

Table 5: Train Data - Predictors

The top 5 rows of test data is:

flood_proof	flood_impacted	num_refill_req_l3m	electric_supply	wh_owner_type	Location_type	transport_issue_l1y	storage_issue_reported_l3m
0	0	5	1	1	0	0	23
0	0	1	0	1	0	1	5
0	0	5	1	0	0	0	6
0	0	6	1	0	0	3	18
0	0	7	1	0	0	0	24

approved_wh_govt_certificate	temp_reg_mach	wh_breakdown_l3m
0	1	5
1	0	1
4	1	3
0	1	4
2	1	6

Table 6: Test Data - Predictors

Standardization or Z-Score Normalization is applied on models using Linear, Ridge and Lasso Regression.

Performance Metrics

Score: R-squared (R2) is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model. When score is called on regressors, the coefficient of determination - R2 is calculated by default. As in classifiers, the score method is simply a shorthand to calculate R2 since it is commonly used to assess the performance of a regressor.

$$R^2 = 1 - (RSS/TSS)$$

R^2	=	coefficient of determination
RSS	=	sum of squares of residuals
TSS	=	total sum of squares

Equation 1:R2 Score

RMSE: Root Mean Square Error is the measure of how well a regression line fits the data points. It is the square root of value obtained from Mean Square Error. It can also be construed as Standard Deviation of the residuals.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (Predicted_i - Actual_i)^2}{N}}$$

Equation 2:RMSE

MAPE: MAPE can be considered as a loss function to define the error termed by the model evaluation. Using MAPE, we can estimate the accuracy in terms of the differences in the actual v/s estimated values. Lower the MAPE, better fit is the model.

$$MAPE = \frac{1}{n} \sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right|$$

Equation 3:MAPE

Model 1 : Linear Regression - with scaled data

Linear Regression is a supervised machine learning algorithm where the predicted output is continuous and has a constant slope. It’s used to predict values within a continuous range. It is used to determine the character and strength of the association between a dependent variable and a series of other independent variables.

The coefficient of the variables are:

```
The coefficient for flood_proof is 53.916
The coefficient for flood_impacted is 18.602
The coefficient for num_refill_req_l3m is -1.117
The coefficient for electric_supply is 8.179
The coefficient for wh_owner_type is 11.93
The coefficient for Location_type is -110.898
The coefficient for transport_issue_l1y is -372.613
The coefficient for storage_issue_reported_l3m is 11534.658
The coefficient for approved_wh_govt_certificate is 109.975
The coefficient for temp_reg_mach is 848.5
The coefficient for wh_breakdown_l3m is -413.874
```

The **intercept** for our model is 21662.542.The **score** for train data is 0.977.The **RMSE** for train data is 1772.312.The **MAPE** for train data is 0.09.

Model 2: Linear Regression - with Unscaled data

The coefficient of the variables are:

```
The coefficient for flood_proof is 53.916
The coefficient for flood_impacted is 18.602
The coefficient for num_refill_req_l3m is -0.428
The coefficient for electric_supply is 8.179
The coefficient for wh_owner_type is 11.93
The coefficient for Location_type is -110.898
The coefficient for transport_issue_l1y is -310.481
The coefficient for storage_issue_reported_l3m is 1254.944
The coefficient for approved_wh_govt_certificate is 109.975
The coefficient for temp_reg_mach is 848.5
The coefficient for wh_breakdown_l3m is -244.307
```

The intercept for our model is 1241.417.The **score** for train data is 0.977.The **RMSE** for train data is 1772.312.The **MAPE** for train data is 0.09.

Model 3: Ridge Regression - with Unscaled data

Regularization refers to techniques that are used to calibrate machine learning models in order to minimize the adjusted loss function and prevent overfitting or underfitting.

Ridge regression is a model tuning method that is used to analyse any data that suffers from multicollinearity. This method performs L2 regularization, i.e. it adds a factor of sum of squares of coefficients in the optimization objective. When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large, this results in predicted values being far away from the actual values. By adding a degree of bias to the regression estimates, ridge regression reduces the standard errors.

The coefficient of the variables are:

```
Ridge model: [[ 5.38587101e+01  1.85904777e+01 -4.16367193e-01  8.18021
158e+00
   1.19253815e+01 -1.10805342e+02 -3.10468872e+02  1.25494244e+03
   1.09997220e+02  8.48231907e+02 -2.44299995e+02]]
```

The ridge coefficients are a reduced factor of the simple linear regression coefficients and thus never attain zero values but very small values.

The intercept for our model is 1241.393.The **score** for train data is 0.977.The **RMSE** for train data is 1772.312.The **MAPE** for train data is 0.09.

Model 4: Ridge Regression - with Scaled data

Both the independent and dependent variables require standardization through subtraction of their averages and a division of the result with the standard deviations.

The coefficient of the variables are:

```
Ridge model: [[ 5.38628505e+01  1.85830636e+01 -1.09575922e+00  8.18505
446e+00
   1.19255051e+01 -1.10630770e+02 -3.72709790e+02  1.15338350e+04
   1.10083809e+02  8.48245817e+02 -4.13559992e+02]]
```

The intercept for our model is 21662.403.The **score** for train data is 0.977.The **RMSE** for train data is 1772.312.The **MAPE** for train data is 0.09.

Model 5:Lasso Regression - Unscaled data

Lasso Regression is an extension of linear regression that adds a regularization penalty to the loss function during training. It is a type of regularized linear regression that includes an L1 penalty. This has the effect of shrinking the coefficients for those input variables that do not contribute much to the prediction task. This penalty allows some coefficient values to go to the value of zero, allowing input variables to be effectively removed from the model, providing a type of automatic feature selection. It favours subsets of features that have less collinearity.

The coefficient of the variables are:

```
Lasso model: [ 3.60456897e+01  8.41882916e+00 -5.58769220e-02  4.394905
60e+00  6.14445208e+00 -9.72763407e+01 -3.09877585e+02  1.25490513e+03
  1.09947684e+02  8.43010481e+02 -2.43933655e+02]
```

The intercept for our model is 1246.465.The **score** for train data is 0.977.The **RMSE** for train data is 1772.329.The **MAPE** for train data is 0.09.

Model 6:Lasso Regression - scaled data

The coefficient of the variables are:

```
Lasso model: [ 3.60745601e+01  8.42904851e+00 -0.00000000e+00
4.40933220e+00   6.13321523e+00 -9.70192898e+01 -3.71938103e+02
1.15330270e+04   1.10033656e+02  8.42921043e+02 -4.12366221e+02]
```

The intercept for our model is 21670.066.The **score** for train data is 0.977.The **RMSE** for train data is 1772.33.The **MAPE** for train data is 0.09.

Model 7: Ensemble method – Random Forest

Ensemble methods are techniques that aim at improving the accuracy of results in models by combining multiple models instead of using a single model.

The bootstrapping Random Forest algorithm combines ensemble learning methods with the decision tree framework to create multiple randomly drawn decision trees from the data, averaging the results to output a new result that often leads to strong predictions.It reduces overfitting in decision trees and helps to improve the accuracy

The feature importance of the variables are:

```
flood_proof                     0.000175
flood_impacted                  0.000258
num_refill_req_l3m              0.001704
electric_supply                 0.000440
wh_owner_type                   0.000514
Location_type                   0.000195
transport_issue_l1y             0.001360
storage_issue_reported_l3m      0.983909
approved_wh_govt_certificate    0.009318
temp_reg_mach                   0.000906
wh_breakdown_l3m                0.001221
```

storage_issue_reported_l3m , approved_wh_govt_certificate are the most important variables according to this model .

The **score** for train data is 0.998.The **RMSE** for train data is 551.175.The **MAPE** for train data is 0.024.

Model 8:Ensemble method – XGBoost

Gradient boosting is one of the variants of ensemble methods where you create multiple weak models and combine them to get better performance as a whole.**XGBoost** (eXtreme Gradient Boosting) is an advanced implementation of gradient boosting algorithm. XGBoost has an in-built capability to handle missing values. It provides various intuitive features, such as parallelisation, distributed computing, cache optimisation.

The **score** for train data is 0.995. The **RMSE** for train data is 813.148.The **MAPE** for train data is 0.039.

Model 9:Ensemble method – BaggingRegressor

A Bagging regressor is an ensemble meta-estimator that fits base regressors each on random subsets of the original dataset and then aggregate their individual predictions (by averaging) to form a final prediction. It significantly raises the stability of models in improving accuracy and reducing variance, which eliminates the challenge of overfitting.

The **score** for train data is 0.997.The **RMSE** for train data is 672.069.
The **MAPE** for train data is 0.031.

Model 10:GridSearchCV - Lasso Regression Scaled

Scaling of features is essential in LASSO. This is because LASSO's penalty function includes the sum of the absolute value of the feature coefficients.

```
{'alpha': [0, 5, 7],
 'random_state': [1],
 'selection': ['cyclic', 'random'],
 'tol': [0.0001, 1e-05]}
```

Alpha -Constant that multiplies the L1 term, controlling regularization strength. 0 means OLS.

Selection - If set to ‘random’, a random coefficient is updated every iteration rather than looping over features sequentially by default.

tol : The tolerance for the optimization: if the updates are smaller than tol, the optimization code checks the dual gap for optimality and continues until it is s maller than tol.
The coefficients according to this variable are:

```
Lasso model: [     0.              0.              0.              0.
     0.          -14.26266313  -367.39824829 11523.36258238
```

```
   109.80136286   813.76606913  -403.08921648]
```

The best estimators are:

```
Lasso(alpha=7, random_state=1, selection='random', tol=1e-05)
```

The variables with coefficients 0 are dropped.

The **score** for train data is 0.977.The **RMSE** for train data is 1772.692.
The **MAPE** for train data is 0.09.

Model 11:GridSearchCV - Lasso Regression Unscaled

Fit on the same parameters as the scaled model.

```
{'alpha': [0, 5, 7],
 'random_state': [1],
 'selection': ['cyclic', 'random'],
 'tol': [0.0001, 1e-05]}
```

The coefficients are:

```
Lasso model: [   0.            0.            0.            0.
0.   -15.65073856 -306.03817262 1254.66876369  109.2043835 813.76200099
 -241.50562001]
```

The best estimators are:

```
Lasso(alpha=7, random_state=1, selection='random')
```

The variables with coefficients 0 are dropped.

The **score** for train data is 0.977. The **RMSE** for train data is 1772.651. The **MAPE** for train data is 0.09.

Model 12:GridSearchCV - Ridge Regression Scaled

The GridSearchCV model is fit on these parameters.

```
{'alpha': [0, 5, 7],
 'solver': ['svd', 'cholesky', 'lsqr'],
 'tol': [0.0001, 1e-05]}
```

The coefficients are:

```
Ridge model: [[ 5.39161752e+01  1.86016596e+01 -1.11665719e+00  8.17920
053e+00
   1.19296003e+01 -1.10897605e+02 -3.72613196e+02  1.15346583e+04
   1.09975459e+02  8.48499621e+02 -4.13873568e+02]]
```

The best estimators are:

```
Ridge(alpha=0, random_state=1, solver='cholesky', tol=0.0001)
```

The **score** for train data is 0.977.The **RMSE** for train data is 1772.312.The **MAPE** for train data is 0.09.

Model 13:GridSearchCV – XGBRegressor

The objective function contains loss function and a regularization term. It tells about the difference between actual values and predicted values, i.e how far the model results are from the real values.

The parameters and values chosen for GridSearch are:

```
{'n_estimators': [80, 100],
 'learning_rate': [0.01, 0.2],
 'subsample': [0.5, 0.7, 0.8],
 'gamma': [0, 1, 3],
 'colsample_bytree': [0.5, 0.7],
 'colsample_bylevel': [0.5, 0.7],
 'max_depth': [3, 5]}
```

n_estimators - The number of trees in the forest.
learning_rate -The learning rate is the shrinkage you do at every step you are making. If you make 1 step at eta = 1.00, the step weight is 1.00. If you make 1 step at eta = 0.25, the step weight is 0.25. Decreasing this hyperparameter reduces the likelihood of overfitting.
Subsample -Subsample ratio of the training instances. Setting it to 0.5 means that XGBoost would randomly sample half of the training data prior to growing trees. and this will prevent overfitting. Subsampling will occur once in every boosting iteration.Decreasing this hyperparameter reduces the likelihood of overfitting.
colsample_bylevel - Subsample ratio for the columns used, for each level inside a tree.Decreasing this hyperparameter reduces the likelihood of overfitting.
colsample_bytree - Subsample ratio for the columns used, for each tree. Decreasing this hyperparameter reduces the likelihood of overfitting.
Gamma – Minimum loss reduction required for any update to the tree.
Max_depth - Maximum allowed depth of the trees.Decreasing this hyperparameter reduces the likelihood of overfitting.

The **score** for train data is 0.994.The **RMSE** for train data is 921.463. The **MAPE** for train data is 0.045.

Model 14:GridSearchCV - Bagging Regressor Random Forest

Bagging model is built on Random Forest

```
{'base_estimator__max_depth': [8, 10, 12],
 'max_features': [3, 4],
 'base_estimator__min_samples_split': [350, 525]}
```

min_samples_split: The minimum number of samples required to split an internal node.Train data has 17500 rows and optimum value for min sample split is 2%-3% of training set.
max-depth : The number of splits that each decision tree is allowed to make. Values of max-depth is suggested to be taken from 8-15 to avoid overfitting and underfitting.
max_features: The number of features to consider when looking for the best split.Value of Max feature is taken as square root of number of independent variables to half of the number of independent variables.
The best estimators are:

```
{'base_estimator__max_depth': 8,
 'base_estimator__min_samples_split': 350,
 'max_features': 4}
```

The **score** for train data is 0.689.The **RMSE** for train data is 6492.373.

The **MAPE** for train data is 0.413.

5. Model Validation

R2 score, RMSE and MAPE is used to validate the model.

Performance metrics of the Test data for all the models are as follows:

Model 1 : Linear Regression - with scaled data

The **score** for test data is 0. 978.The **RMSE** for test data is 1719.118.
The **MAPE** for test data is 0.089.
The MAPE must be lesser than 15 and the difference between MAPE of train and test is 10. The model is good and has low MAPE.

Model 2: Linear Regression - with Unscaled data

The **score** for test data is 0. 978.The **RMSE** for test data is 1713.173.
The **MAPE** for test data is 0.089.
The model is good and has low MAPE.

Model 3: Ridge Regression - with Unscaled data

The **score** for test data is 0. 978.The **RMSE** for test data is 1713.174.
The **MAPE** for test data is 0.089.
The model is good and has low MAPE.

Model 4: Ridge Regression - with Scaled data

The **score** for test data is 0. 978.The **RMSE** for test data is 1719. 068.The **MAPE** for test data is 0.089.
The model is good and has low MAPE.

Model 5:Lasso Regression - Unscaled data

The **score** for test data is 0. 978.The **RMSE** for test data is 1713. 189.The **MAPE** for test data is 0.089.
The model is good and has low MAPE.

Model 6:Lasso Regression - scaled data

The **score** for test data is 0. 978.The **RMSE** for test data is 1719. 041.The **MAPE** for test data is 0.089.
The model is good and has low MAPE.

Model 7: Ensemble method – Random Forest

The **score** for test data is 0. 993.The **RMSE** for test data is 972.685.The **MAPE** for test data is 0.045.
The model is good and has low MAPE.

Model 8:Ensemble method – XGBoost

The **score** for test data is 0. 994.The **RMSE** for test data is 897.604.The **MAPE** for test data is 0.043.
The model is good and has low MAPE. The standards for a good R-Squared reading can be much higher, such as 0.9 or above.

Model 9:Ensemble method – BaggingRegressor

The **score** for test data is 0. 993. The **RMSE** for test data is 932.699.The **MAPE** for test data is 0.044.

The model is good and has low MAPE.

Model 10:GridSearchCV - Lasso Regression Scaled

The **score** for test data is 0. 978.The **RMSE** for test data is 1718.72.The **MAPE** for test data is 0.089.

The model is good and has low MAPE.

Model 11:GridSearchCV - Lasso Regression Unscaled

The **score** for test data is 0.978. The **RMSE** for test data is 1713.384. The **MAPE** for test data is 0.088.
The model is good and has low MAPE.

Model 12:GridSearchCV - Ridge Regression Scaled

The **score** for test data is 0. 978. The **RMSE** for test data is 1719.118.
The **MAPE** for test data is 0.089.
The model is good and has low MAPE.

Model 13:GridSearchCV – XGBRegressor

The **score** for test data is 0. 994.The **RMSE** for test data is 912.132.

The **MAPE** for test data is 0.046.

The model is good and has low MAPE.

Model 14:GridSearchCV - Bagging Regressor Random Forest

The **score** for test data is 0. 686.The **RMSE** for test data is 6450.974.

The **MAPE** for test data is 0.413.

The model is average and has high MAPE.

	R2 Score	RMSE	MAPE
Basic LR Scaled Train	0.977	1772.312	0.090
Basic LR Scaled Test	0.978	1719.118	0.089
Basic LR Unscaled Train	0.977	1772.312	0.090
Basic LR Unscaled Test	0.978	1713.173	0.089
Basic Lasso Scaled Train	0.977	1772.330	0.090
Basic Lasso Scaled Test	0.978	1719.041	0.089
Basic Lasso Unscaled Train	0.977	1772.329	0.090
Basic Lasso Unscaled Test	0.978	1713.189	0.089
Basic Ridge Scaled Train	0.977	1772.312	0.090
Basic Ridge Scaled Test	0.978	1719.068	0.089
Basic Ridge Unscaled Train	0.977	1772.312	0.090
Basic Ridge Unscaled Test	0.978	1713.174	0.089
Basic RF Train	0.998	551.175	0.024
Basic RF Test	0.993	972.685	0.045
Bagging RF Train	0.997	672.069	0.031
Bagging RF Test	0.993	932.699	0.044
Basic XGBoost Train	0.995	813.148	0.039
Basic XGBoost Test	0.994	897.604	0.043
Grid Ridge Scaled Train	0.977	1772.312	0.090
Grid Ridge Scaled Test	0.978	1719.118	0.089
Grid Lasso Scaled Train	0.977	1772.692	0.090
Grid Lasso Scaled Test	0.978	1718.720	0.089
Grid Lasso Unscaled Train	0.977	1772.651	0.090
Grid Lasso Unscaled Test	0.978	1713.384	0.088
Grid XGBoost Train	0.994	921.463	0.045
Grid XGBoost Test	0.994	912.132	0.046
Grid Bag RF Train	0.689	6492.373	0.413
Grid Bag RF Test	0.686	6450.974	0.409

Table 7:Models-Performance Metrics

Of all the models GridSearchCV Bagging Random Forest didn’t perform well. Basic Random Forest is the optimum model as it has the highest R2 score and lowest MAPE,RMSE.

6. Final interpretation / recommendation

Business Insights

	Imp
storage_issue_reported_l3m	0.983909
approved_wh_govt_certificate	0.009318
num_refill_req_l3m	0.001704
transport_issue_l1y	0.001360
wh_breakdown_l3m	0.001221
temp_reg_mach	0.000906
wh_owner_type	0.000514
electric_supply	0.000440
flood_impacted	0.000258
Location_type	0.000195
flood_proof	0.000175

Table 8:Feature Importance

- According to Random Forest model the most important features are number of times storage issue reported in the last 3 months, standard certificate issued to the warehouse by the government regulatory body, number of times transport issue in the past 1 year, number of times warehouse breakdown in last 3 months, number of times refilling done in the last 3 months.

- The more the average weight of product shipped in the last 3 months to the warehouse the higher it is certified. A+ certified warehouses has the highest 39 average weight of product shipped. C certified warehouses has the least average weight of product shipped.

- From the above lineplot we can see that the number of warehouse breakdown increases as the average number of product shipped started to increase. This can be due to difficulty in maintenance.

- Storage issue increases as the average amount of product shipped increases. This might denote the difficulty faced by the warehouse workers to maintain the warehouse. North Zone has the highest total number of storage issues and total weight of products shipped.

- As the transport issue increases the average amount of product shipped decreases.

Recommendations

- Storage issues like insects, rat, fungus due to moisture can be reduced using rodenticides, temperature regulating machines.

- Transport issues like accidents and stealing of goods can be avoided by monitoring the drivers.

- The warehouse can be made flood proof for the storage to be flood proof and electricity backup like generators can be setup to prevent warehouse breakdown.

- North Zone has less warehouses with temperature regulating machine which is a cause of storage issues. Hence such warehouses can be equipped with temperature regulating machines.

- East Zone has least average number of retail shop selling the product and more average number of competitors. This can also be due to less number of warehouse in East Zone. So ad campaigns can be done and more warehouses can be set up in East zone to boost the product sales here.

- The most of the dataset is from warehouses in rural areas. This can be because of more land area available in rural areas to build warehouses and blue collar jobs offered in the warehouse. More number of Warehouses can be set up in Urban for easier access to airport and ports.

- The dataset contains less number of records about small-sized warehouses. More data about small warehouses can be collected.

- Highly certified warehouses are less in number. The reason can be because of non-availability of facilities like temperature regulators, back up for electrical supply ,transport issues ,etc . These facilities can be set up and upgraded.

APPENDIX

1. Histplot of num_refill_req_l3m

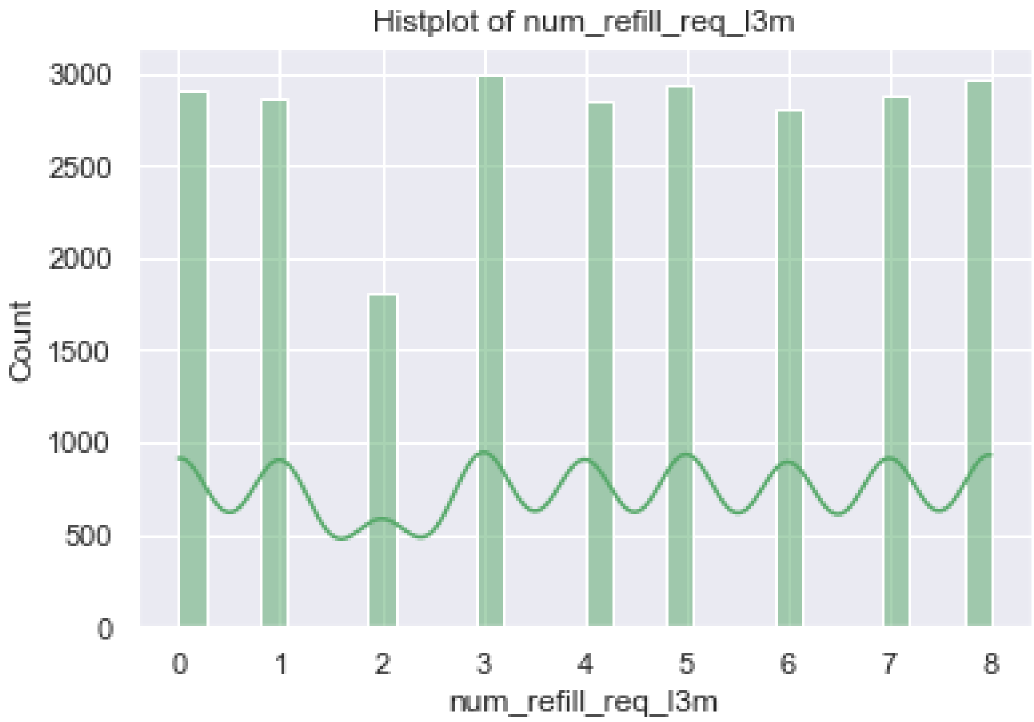

Figure 12: Histplot of num_refill_req_l3m

2. Histplot of transport_issue_l1y

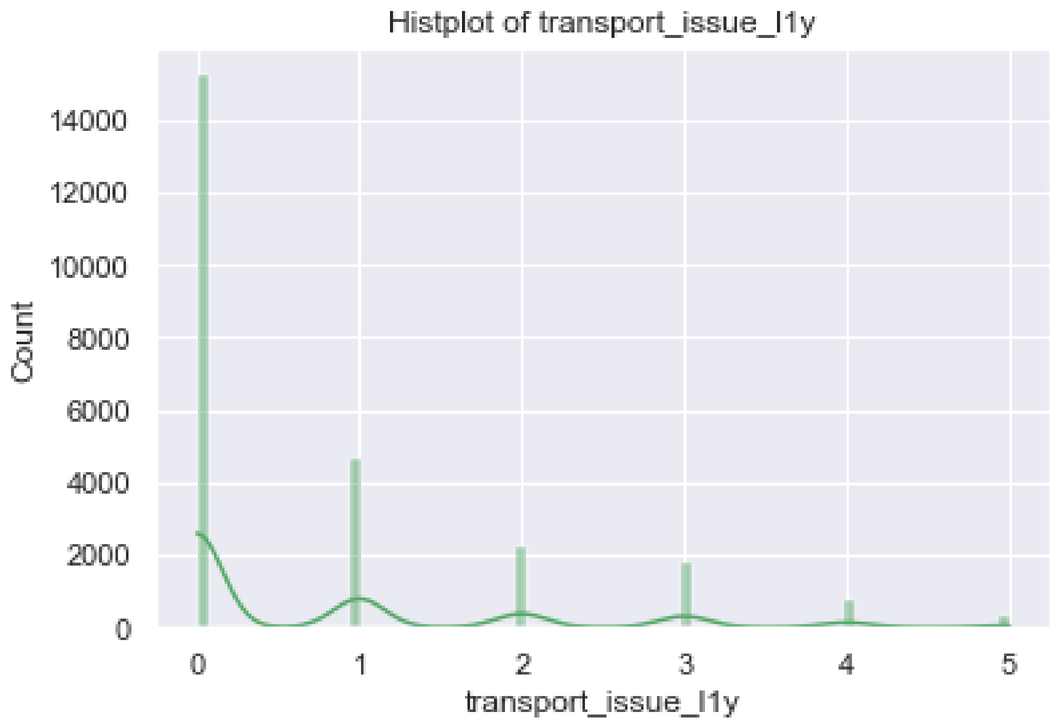

Figure 13:Histplot of transport_issue_l1y

3. Histplot of Competitor_in_mkt

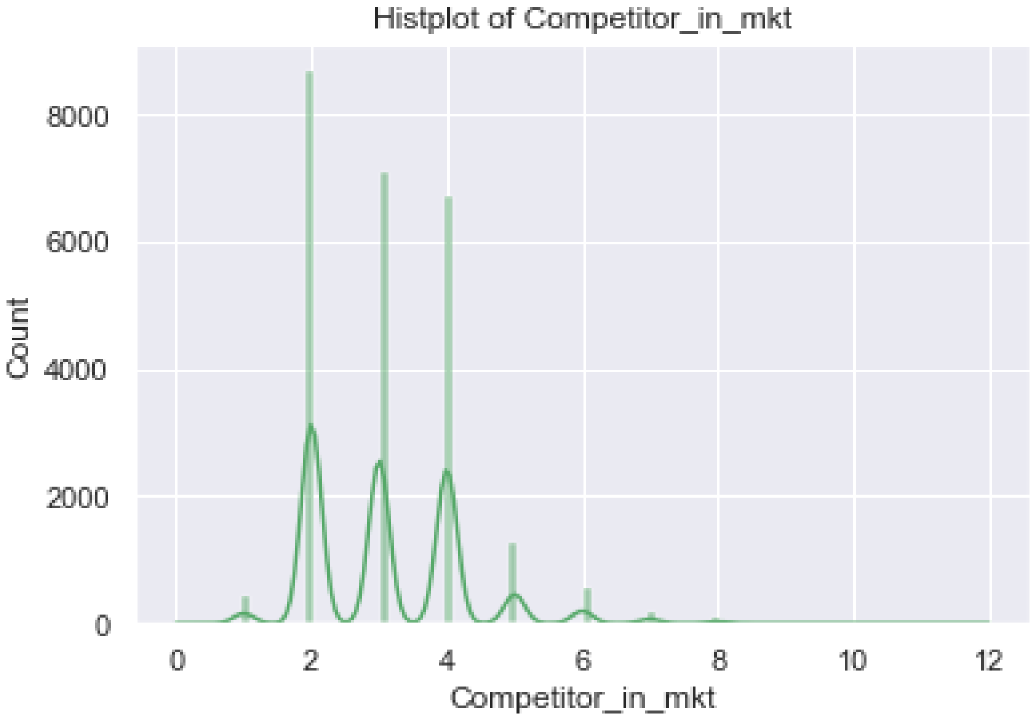

Figure 14:Histplot of Competitor_in_mkt

4. Histplot of retail_shop_num

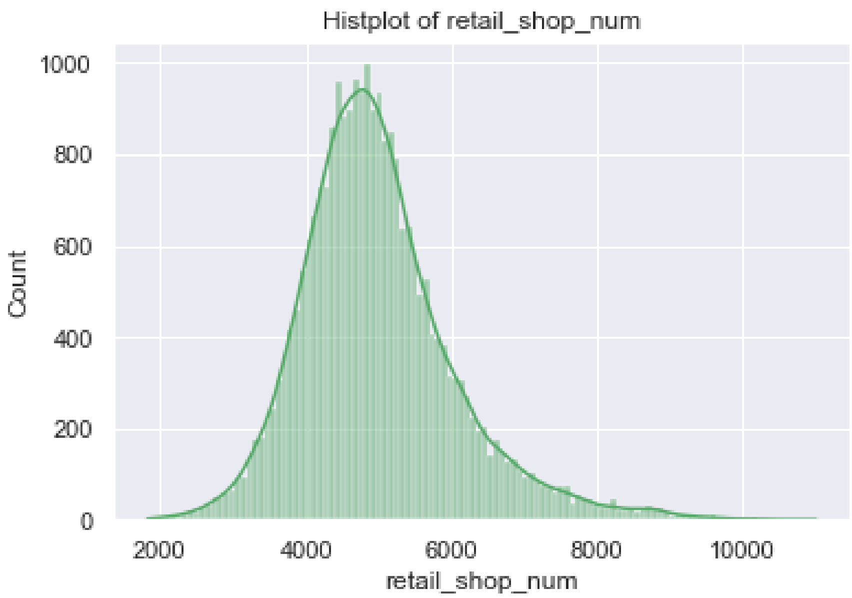

Figure 15:Histplot of retail_shop_num

5. Histplot of distributor_num

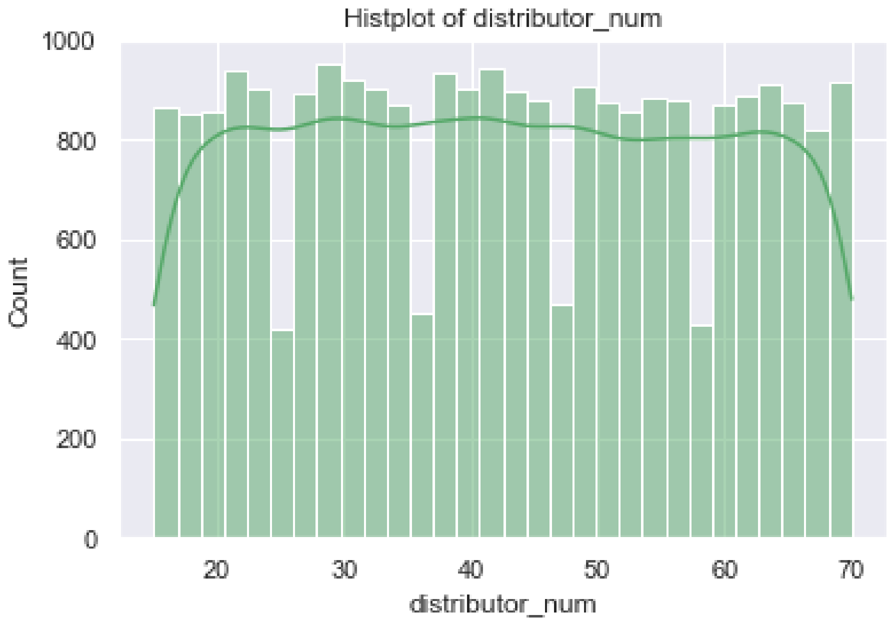

Figure 16:Histplot of distributor_num

6. Histplot of dist_from_hub

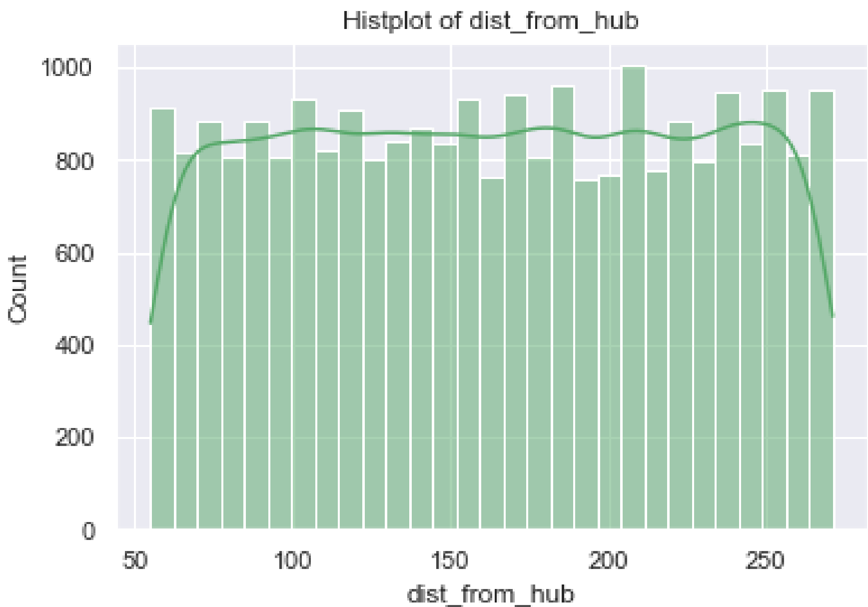

Figure 17:Histplot of dist_from_hub

7. Histplot of workers_num

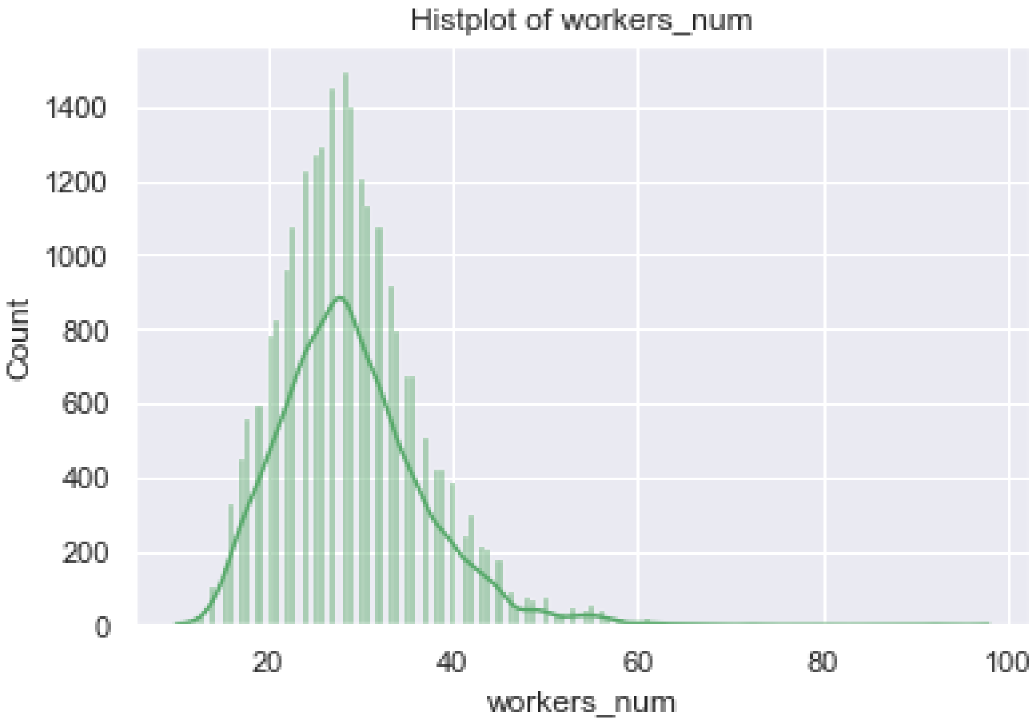

Figure 18:Histplot of workers_num

8. Histplot of wh_est_year

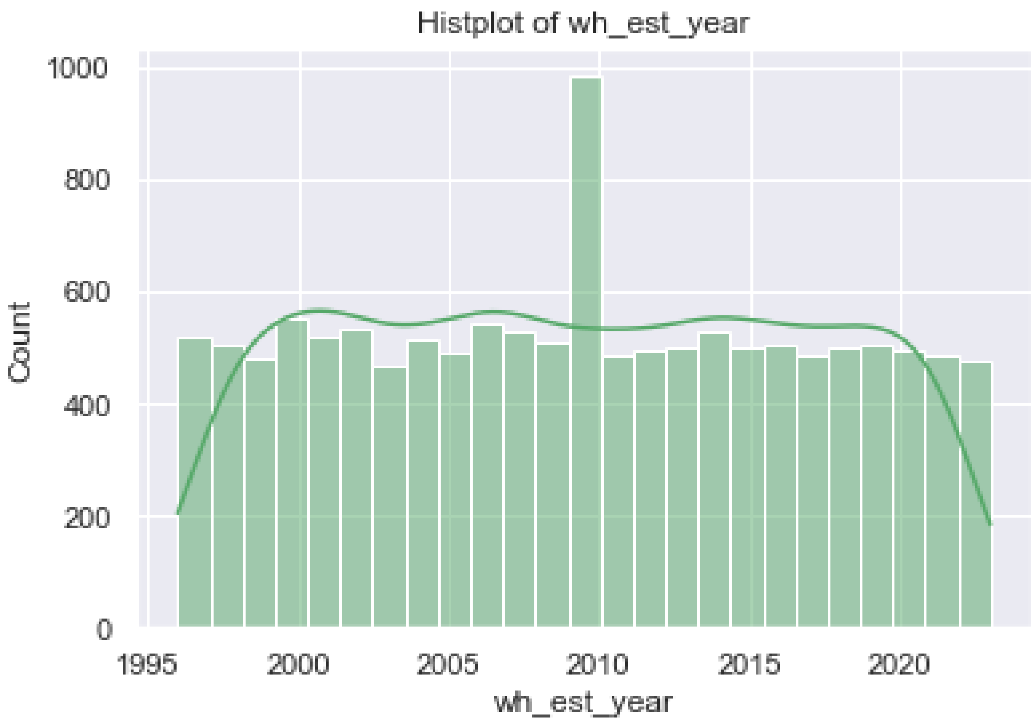

Figure 19: Histplot of wh_est_year

9. Histplot of storage_issue_reported_l3m

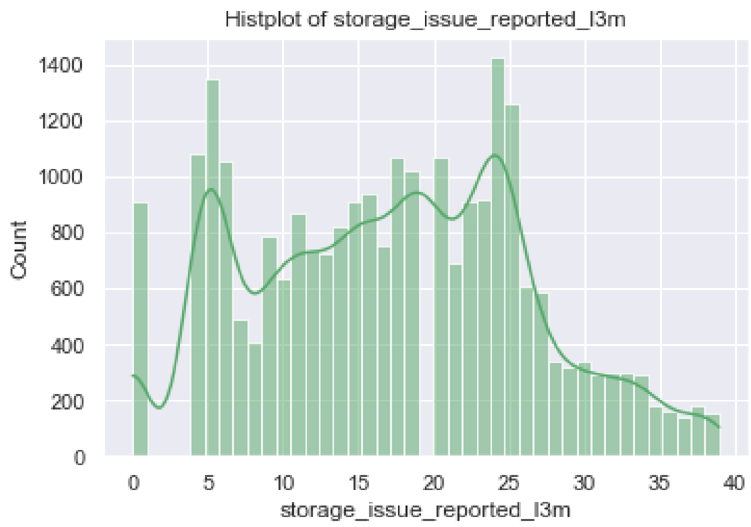

Figure 20:Histplot of storage_issue_reported_l3m

10. Histplot of wh_breakdown_l3m

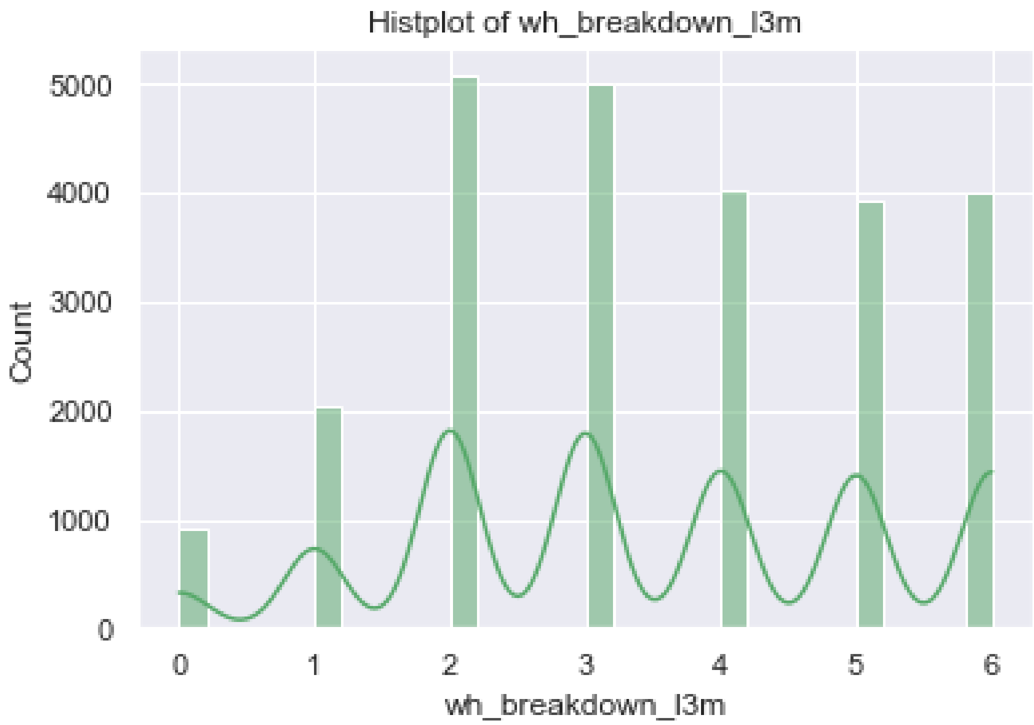

Figure 21:Histplot of wh_breakdown_l3m

11. Histplot of govt_check_l3m

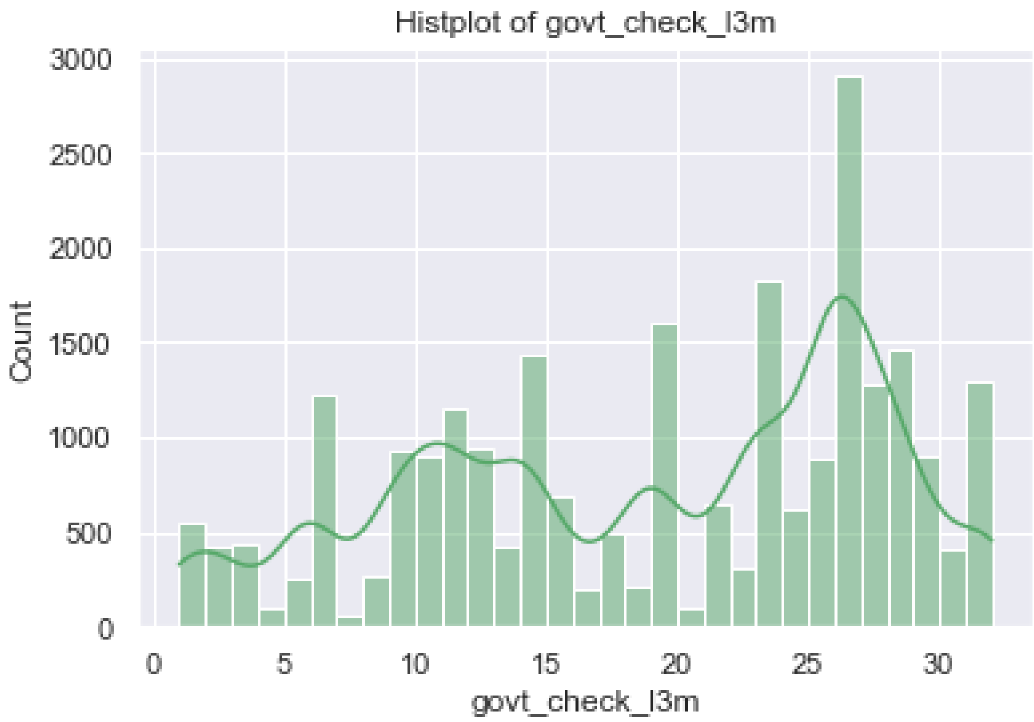

Figure 22: Histplot of govt_check_l3m

12. Histplot of product_wg_ton

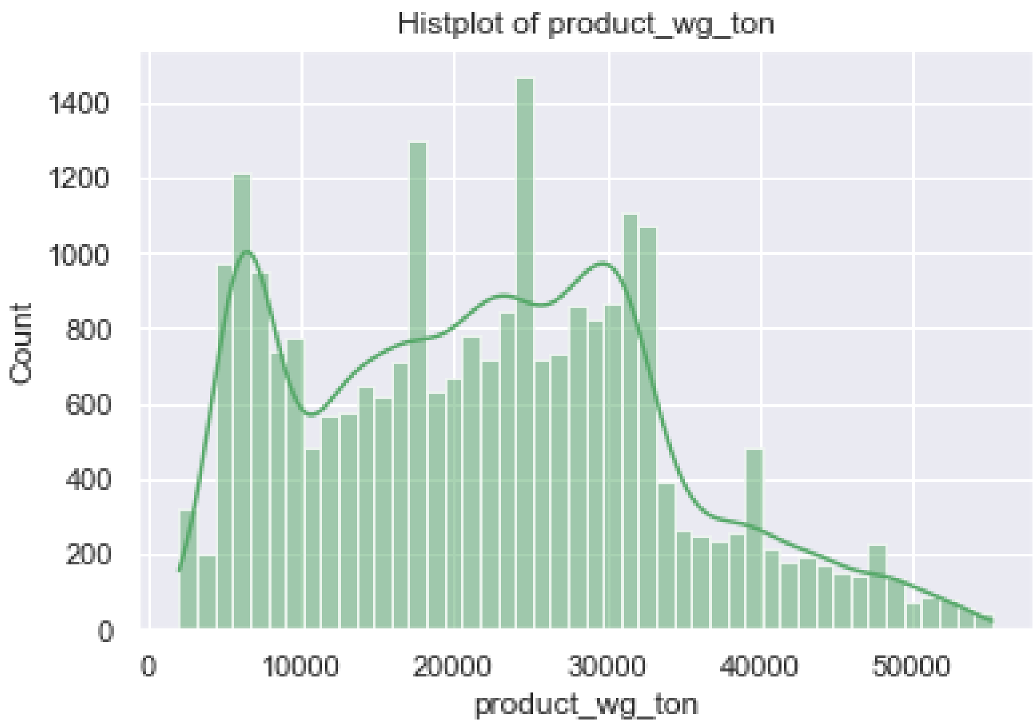

Figure 23:Histplot of product_wg_ton

13. Countplot of Location_type

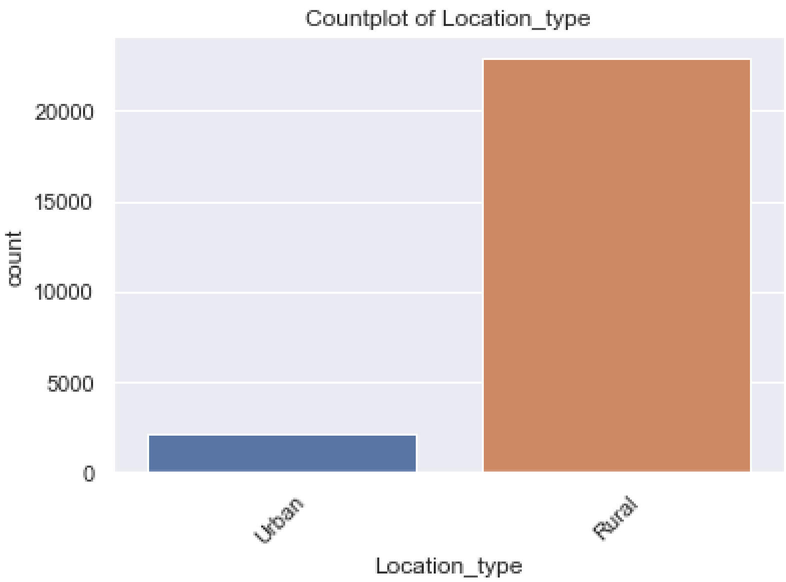

Figure 24:Countplot of Location_type

14. Countplot of WH_capacity_size

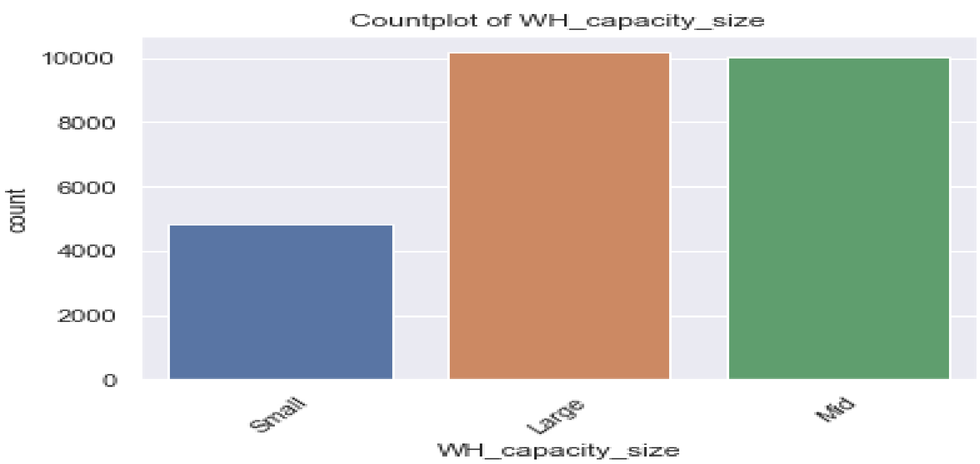

Figure 25:Countplot of WH_capacity_size

15. Countplot of zone

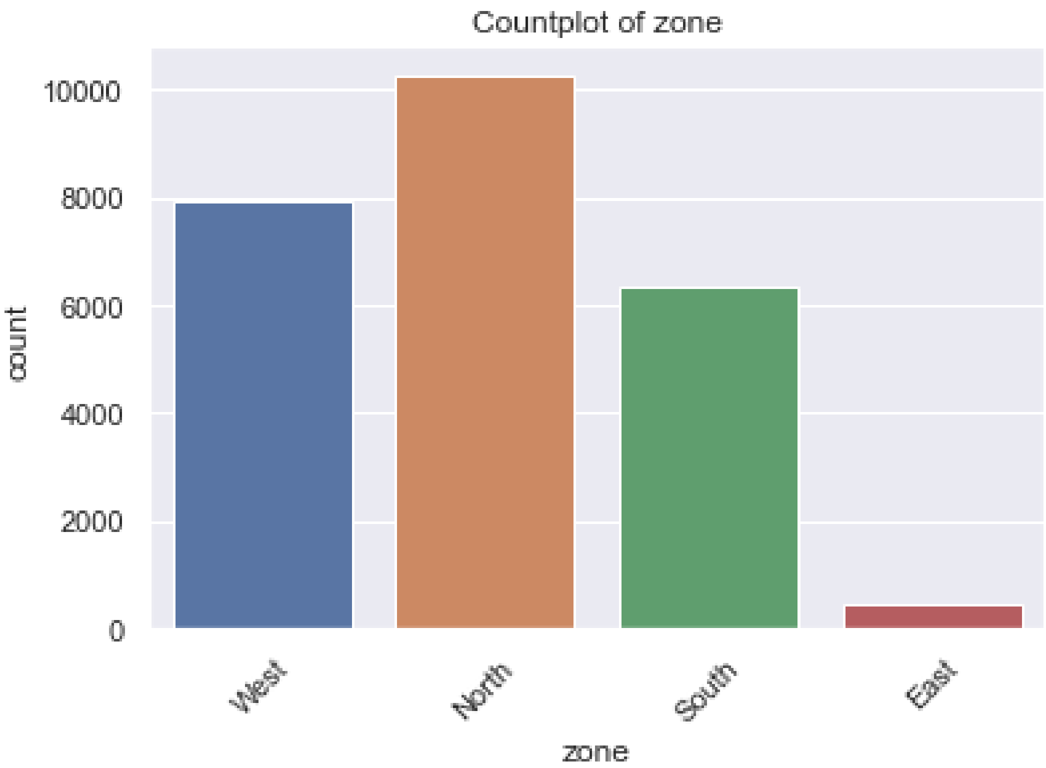

Figure 26:Countplot of zone

16. Countplot of WH_regional_zone

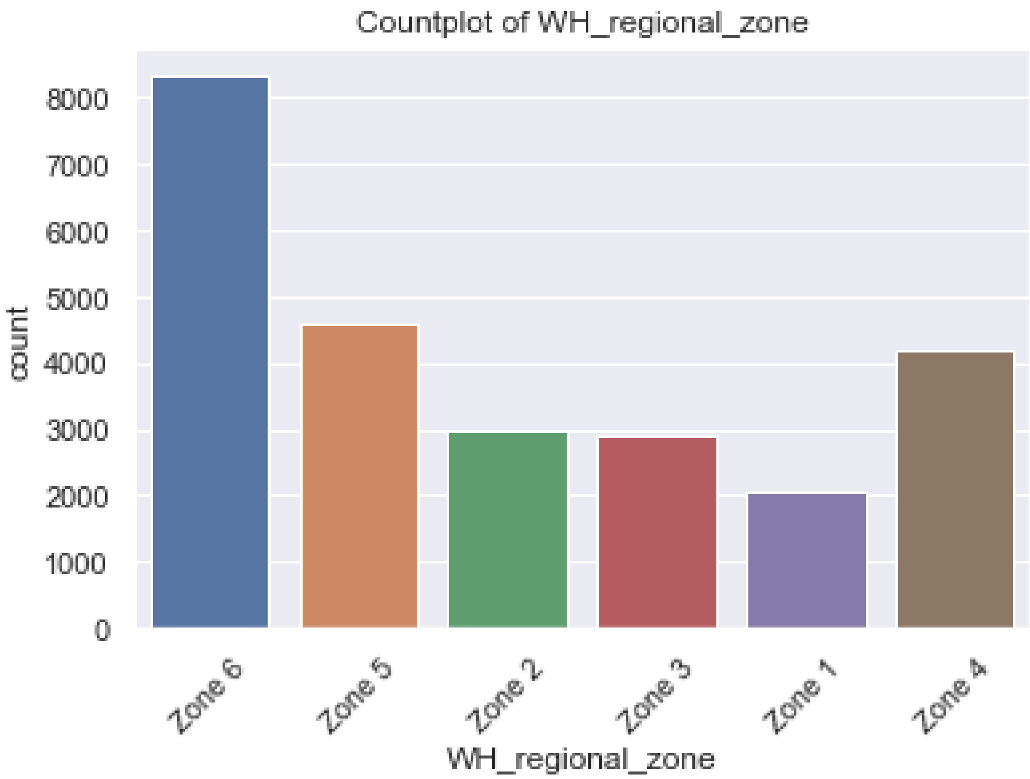

Figure 27:Countplot of WH_regional_zone

17. Countplot of wh_owner_type

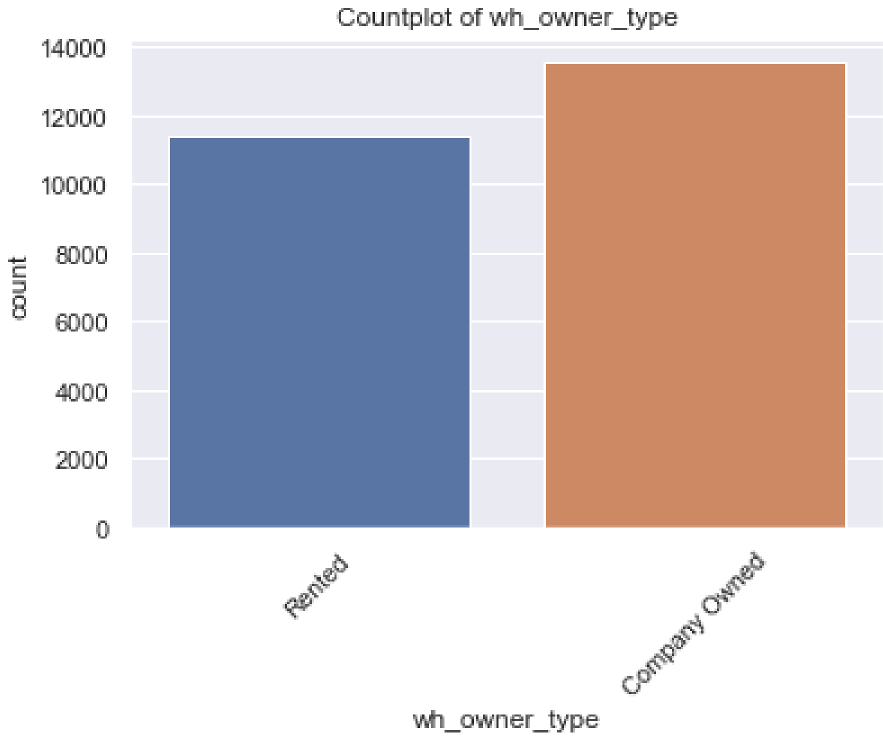

Figure 28:Countplot of wh_owner_type

18. Countplot of approved_wh_govt_certificate

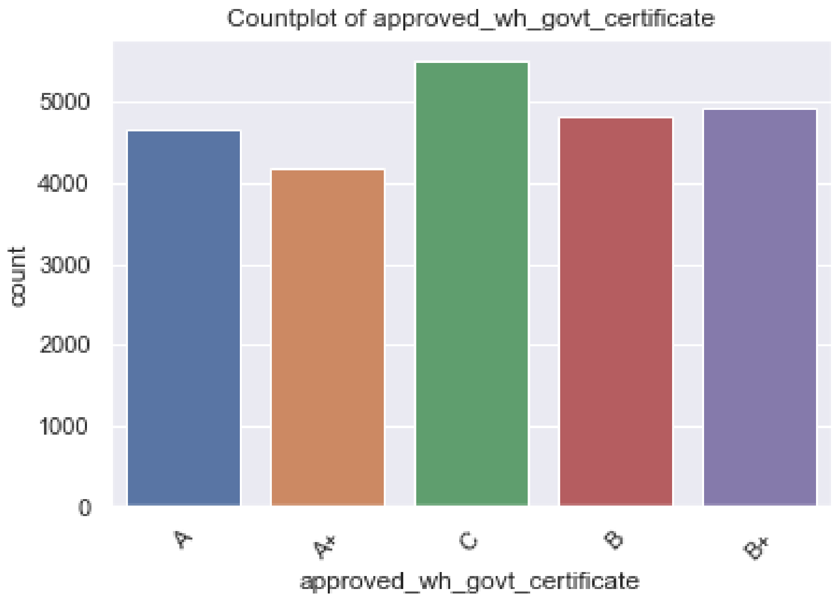

Figure 29:Countplot of approved_wh_govt_certificate

19. Countplot of electric_supply

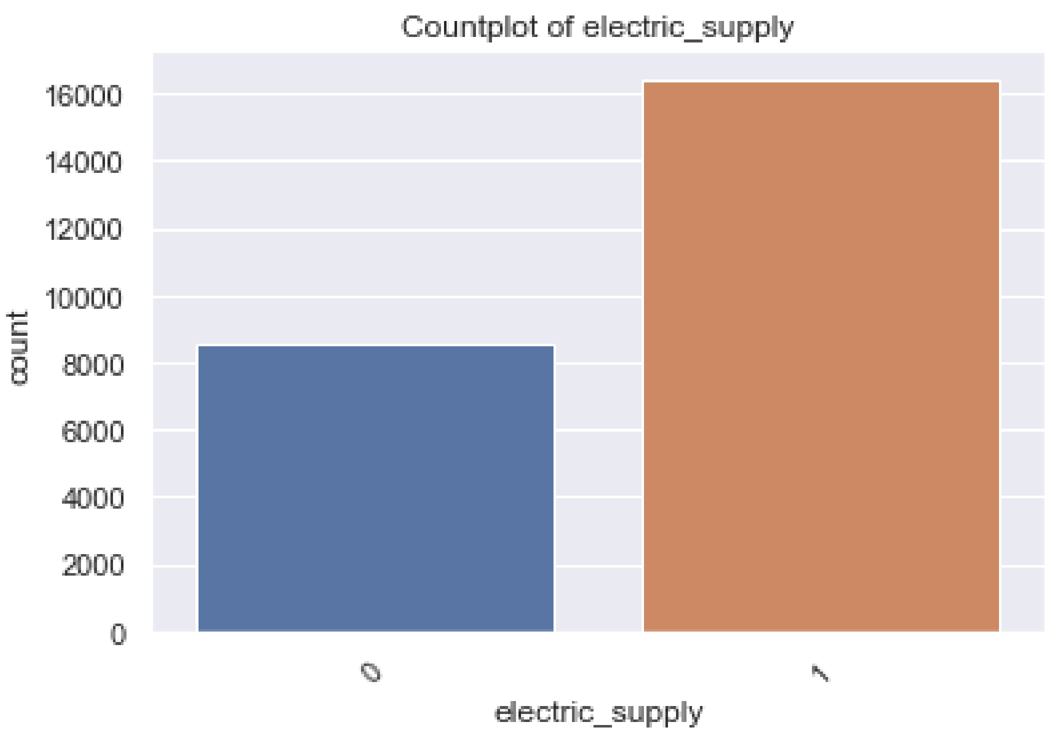

Figure 30:Countplot of electric_supply

20. Countplot of flood_proof

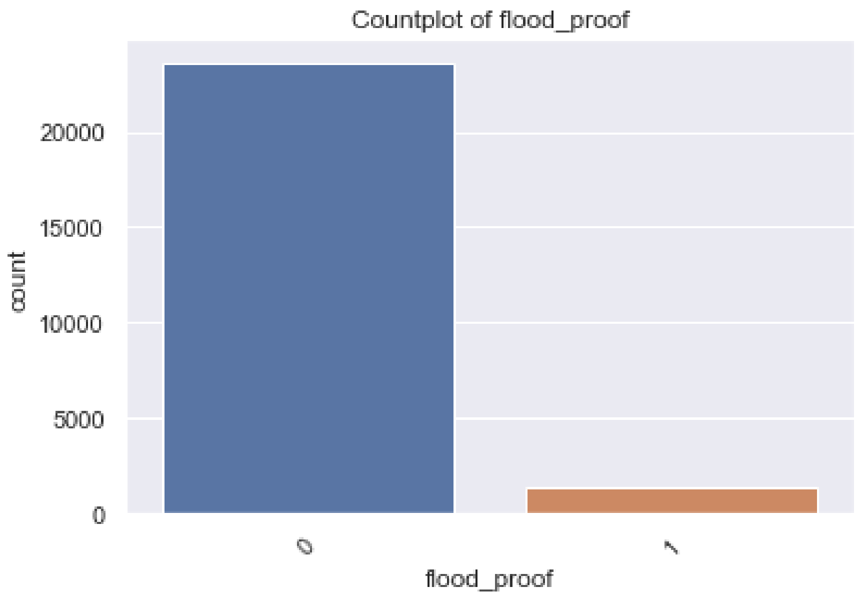

Figure 31:Countplot of flood_proof

21. Countplot of flood_ impacted

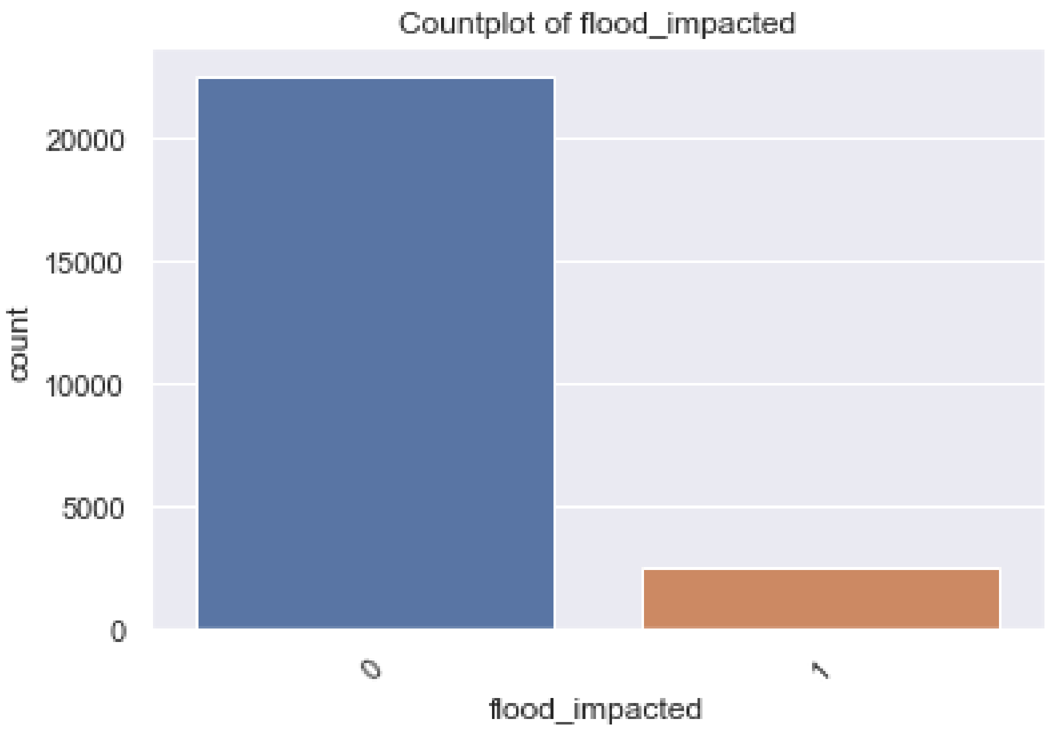

Figure 32:Countplot of flood_impacted

22. Countplot of temp_reg_mach

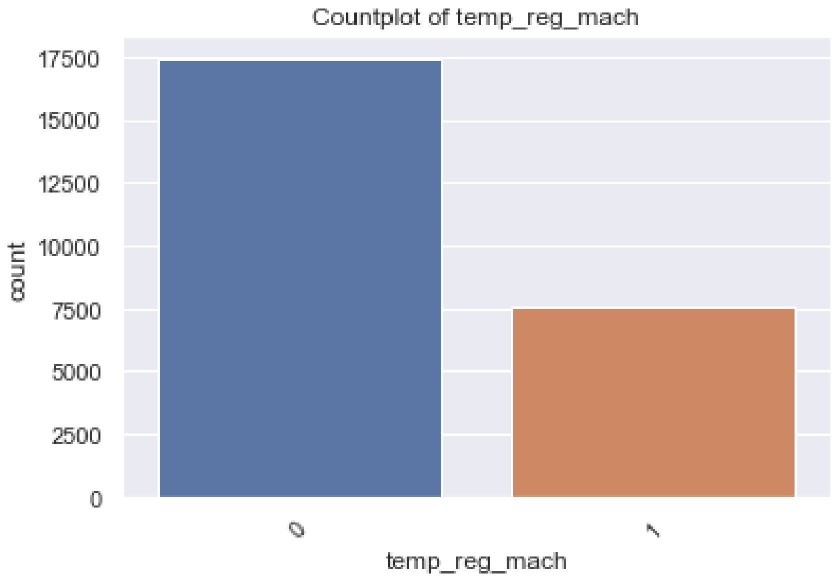

Figure 33:Countplot of temp_reg_mach

23. Barplot of Flood Impacted and Product wg ton

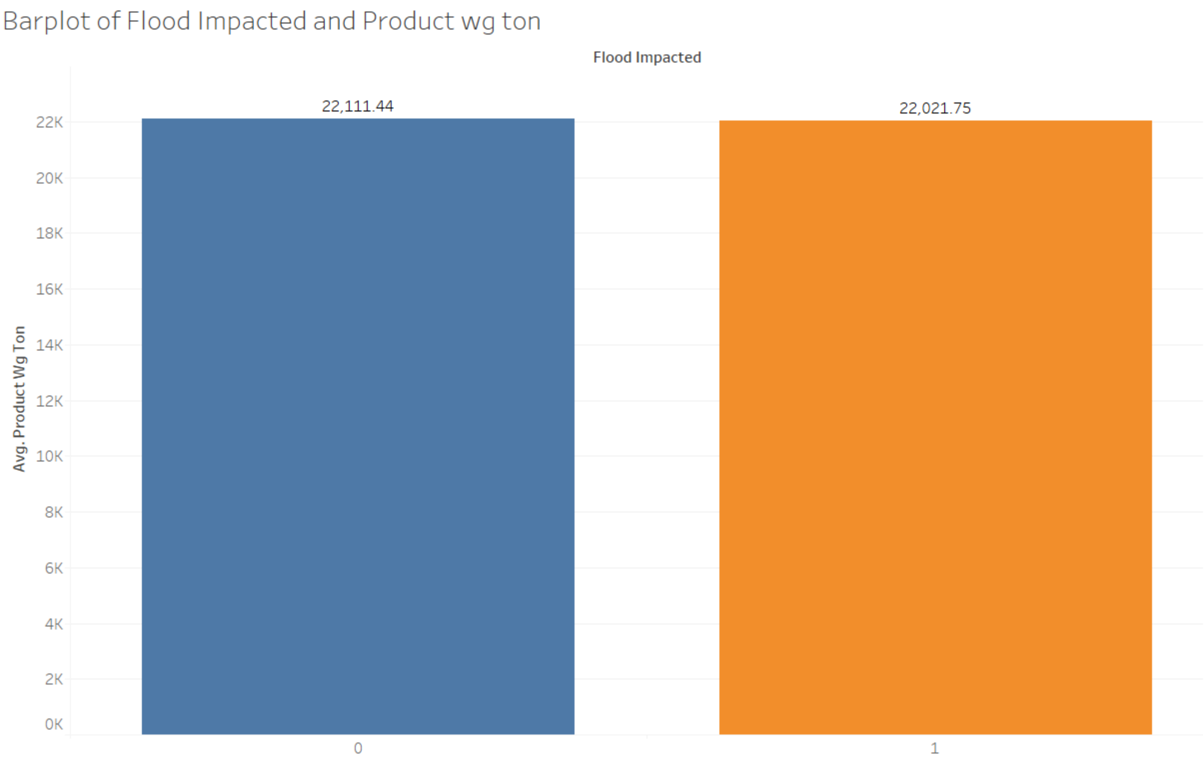

Figure 34:Barplot of Flood Impacted and Product wg ton

24. Barplot of Capacity size and Product wg ton

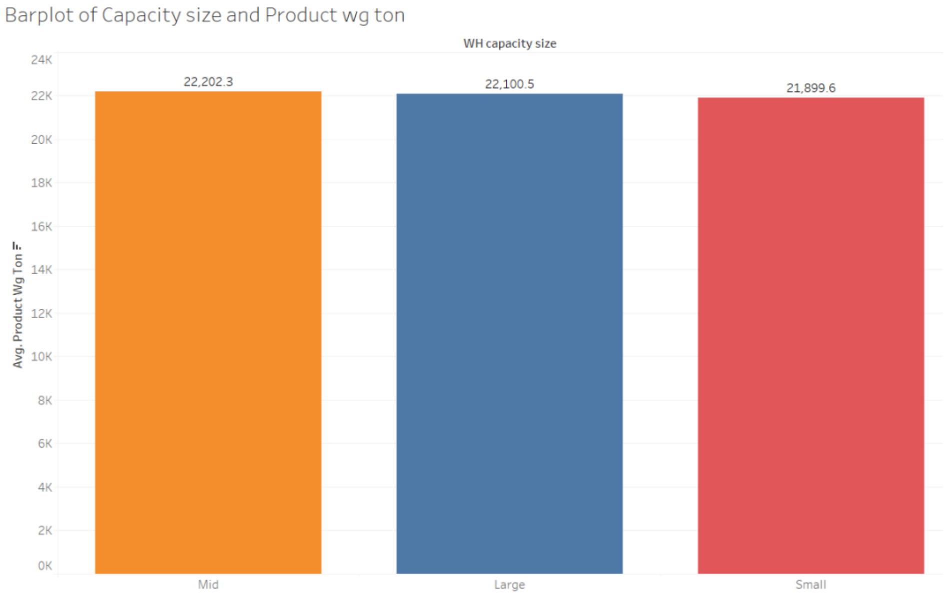

Figure 35: Barplot of Capacity size and Product wg ton

25. Barplot of Zone and Product wg ton

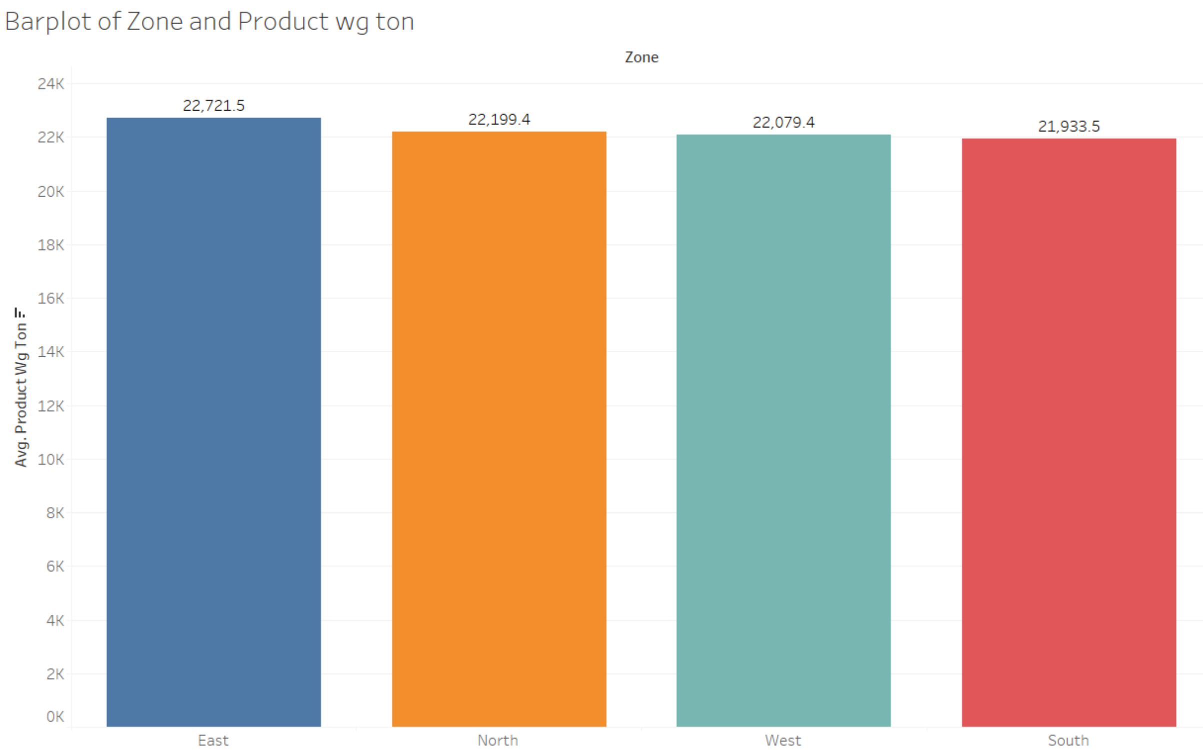

Figure 36: Barplot of Zone and Product wg ton

26. Barplot of Location Type and Product wg ton

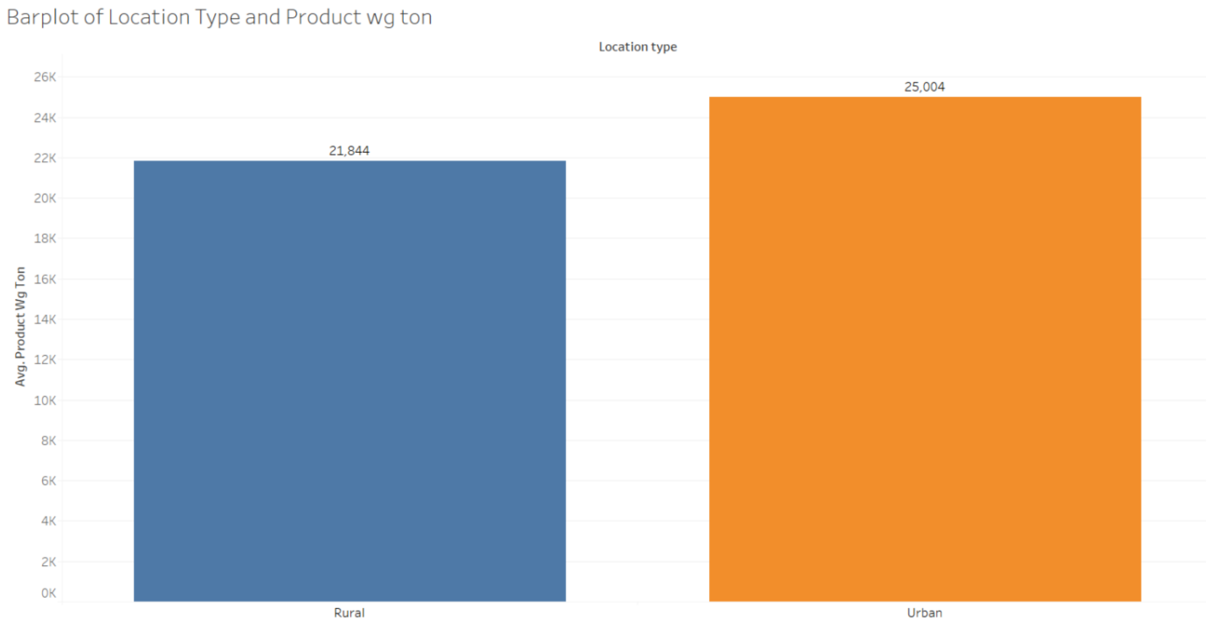

Figure 37: Barplot of Location Type and Product wg ton

27. Barplot of WH Established Year and Product wg ton

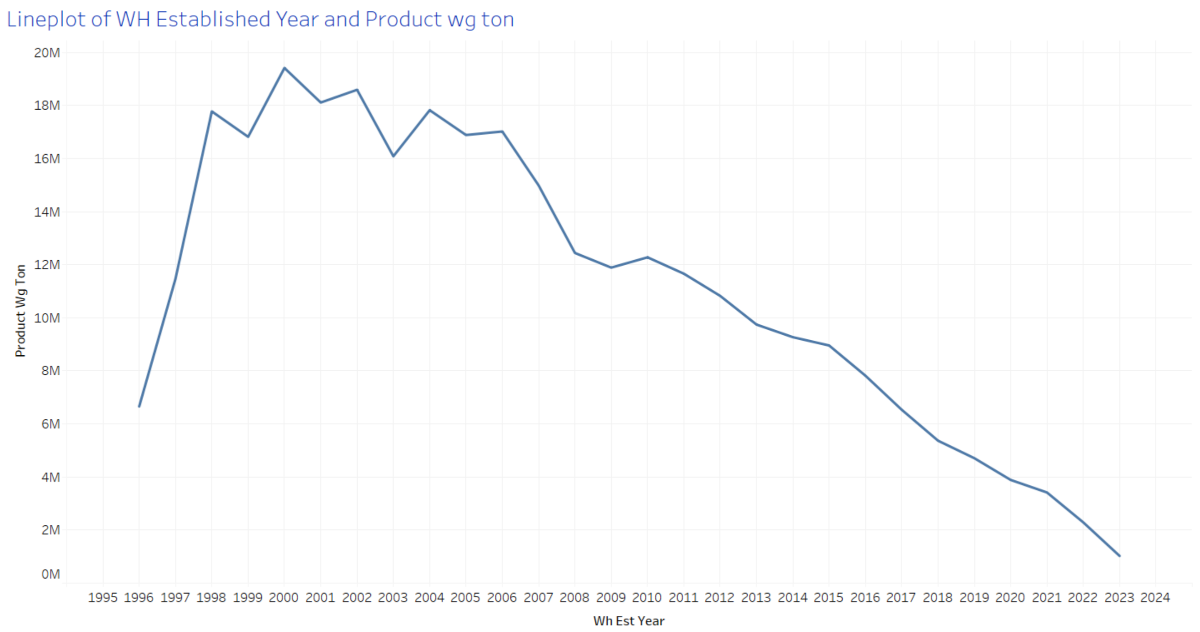

Figure 38:Lineplot of WH Established Year and Product wg ton

28. Barplot of Regional zone and Product wg ton

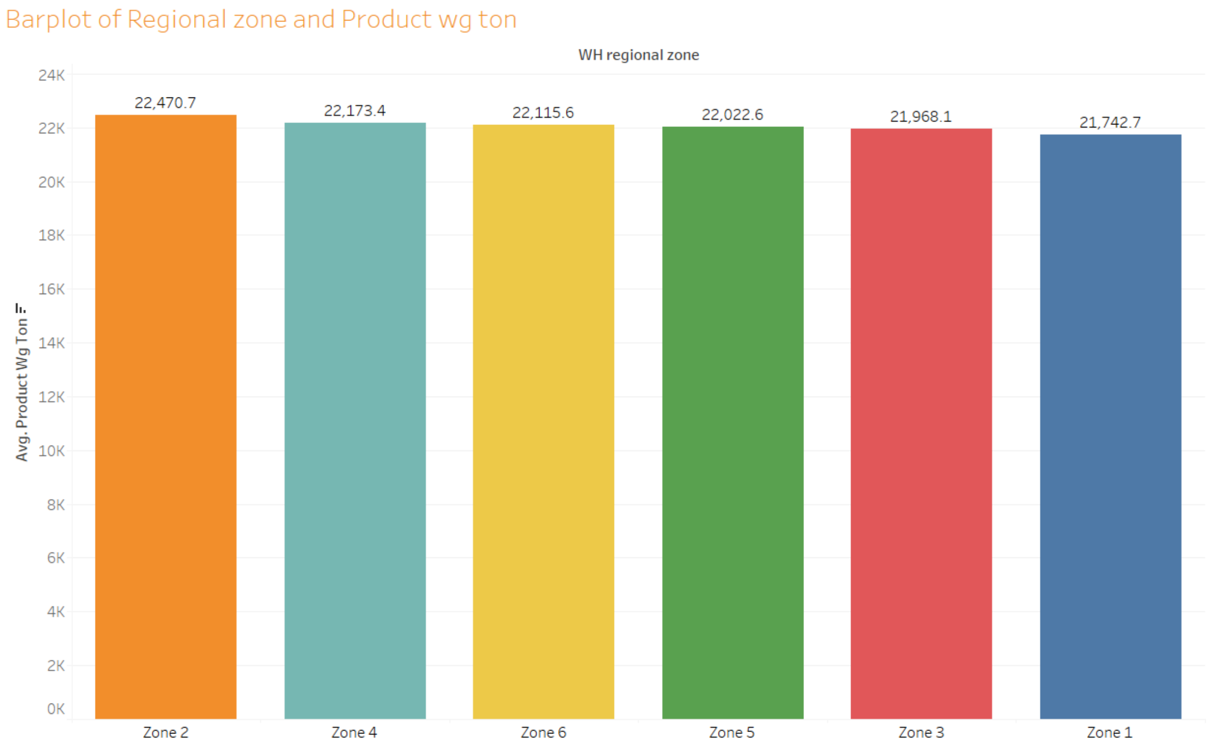

Figure 39:Barplot of Regional zone and Product wg ton

29. Barplot of Owner type and Product wg ton

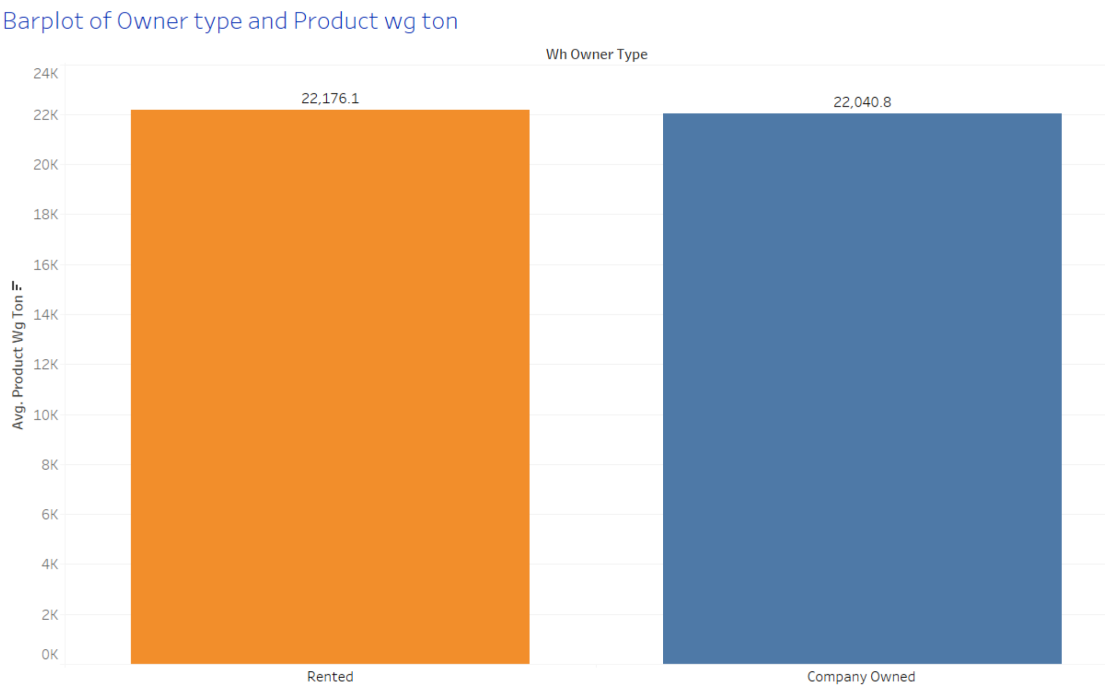

Figure 40:Barplot of Owner type and Product wg ton

30. Barplot of Zone and Total Number of Workers

Piechart of Zone-Total Number of Workers

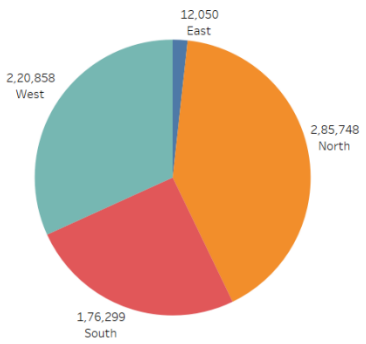

Figure 41:Piechart of Zone-Total Number of Workers

31. Barplot of Zone, Region and Certificate

Barlpot of Zone-Region-Certificate

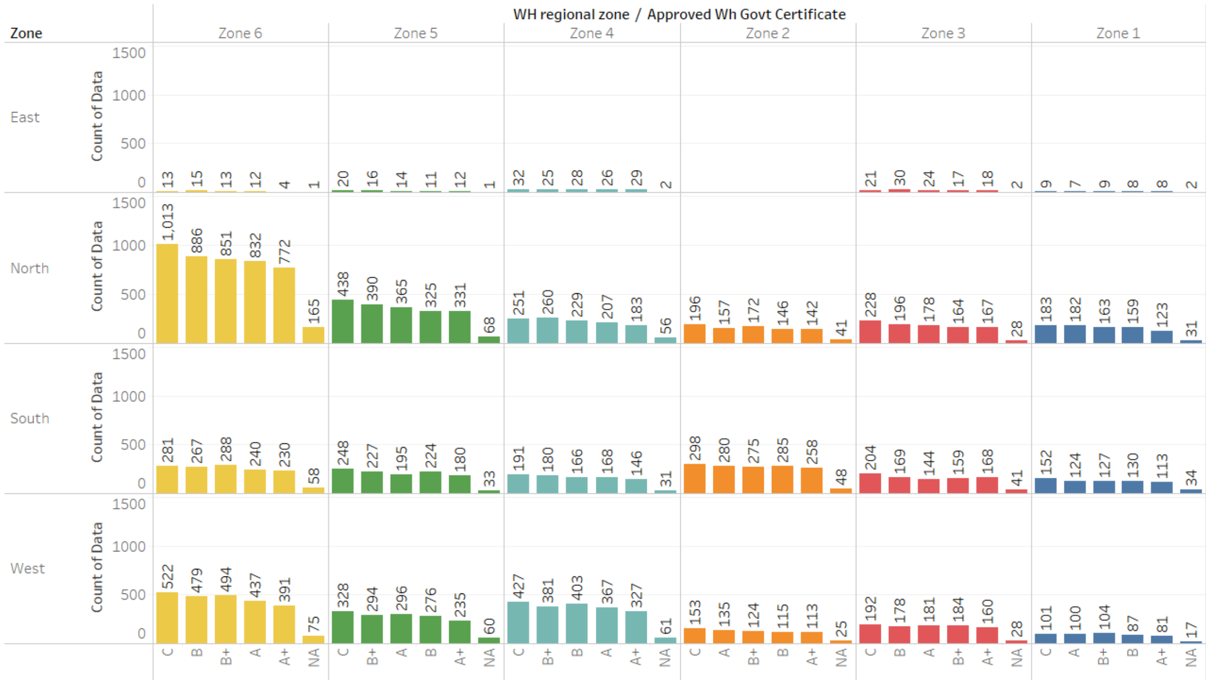

Figure 42:Barlpot of Zone-Region-Certificate

32. Barplot of Owner Type, Capacity Size and Flood Proof

Circle plot of Owner Type-Capacity Size-Flood Proof

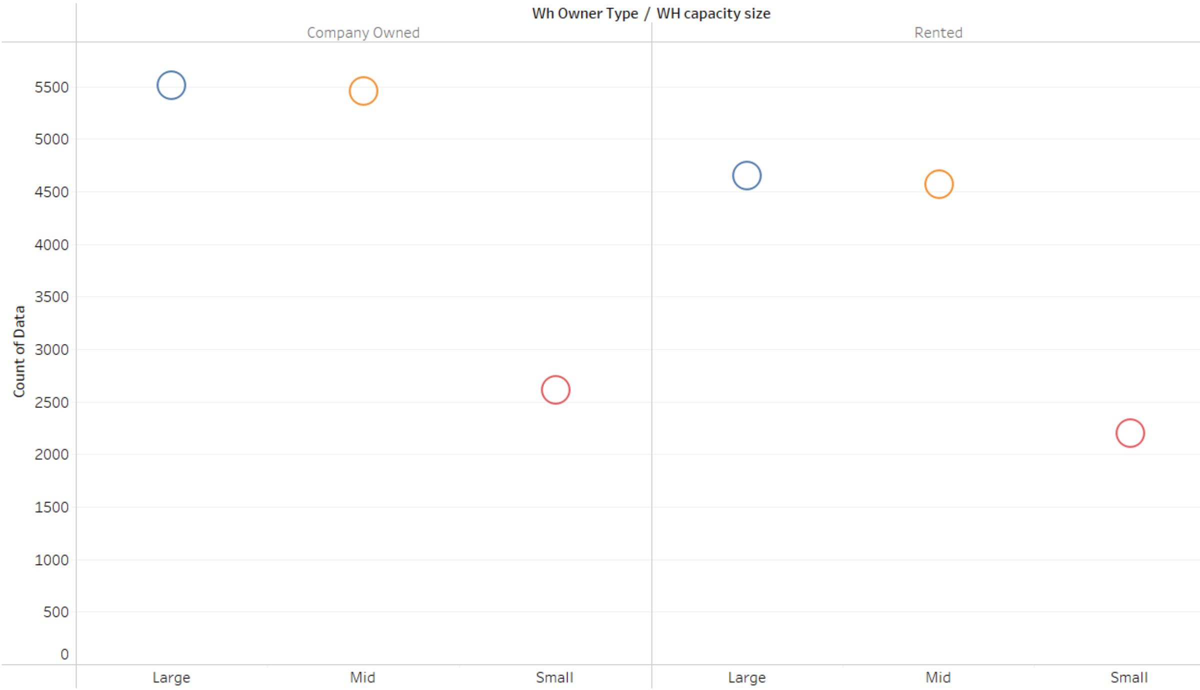

Figure 43:Circle plot of Owner Type-Capacity Size-Flood Proof

33. Barplot of Location,Temperature Regulator and Product Wg Ton

Stacked bar chart of Location-Temperature Regulator-Product Wg Ton

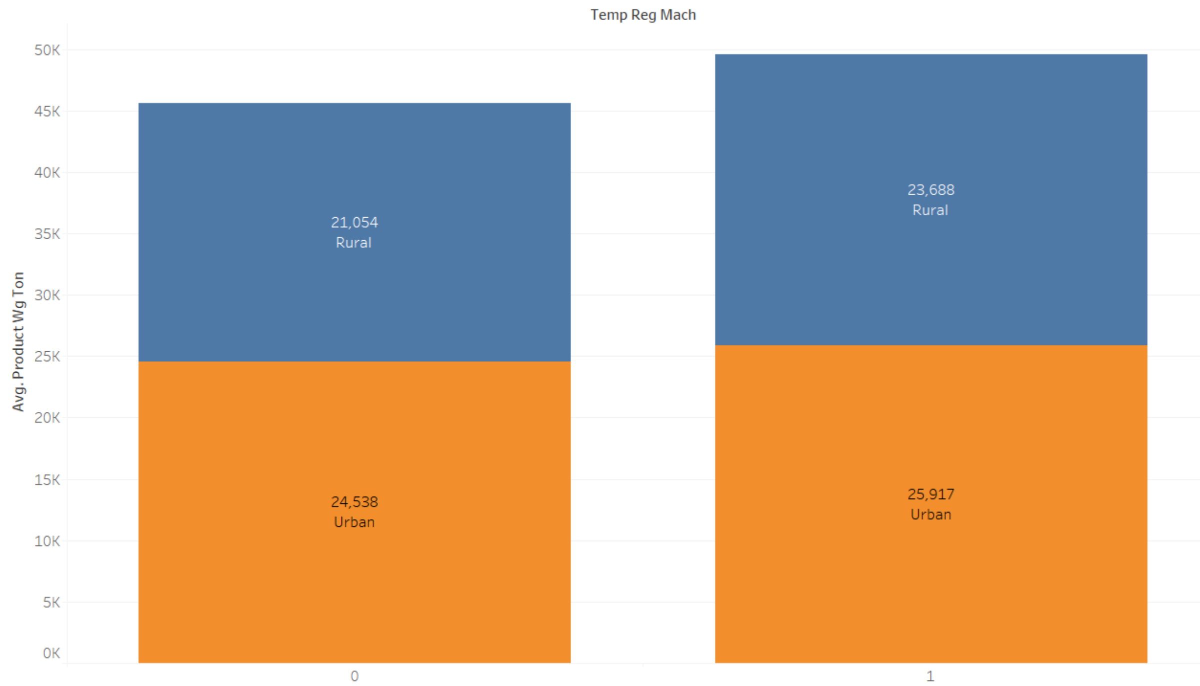

Figure 44:Stacked bar chart of Location-Temperature Regulator-Product Wg Ton

34. Pairplot

Pairplot

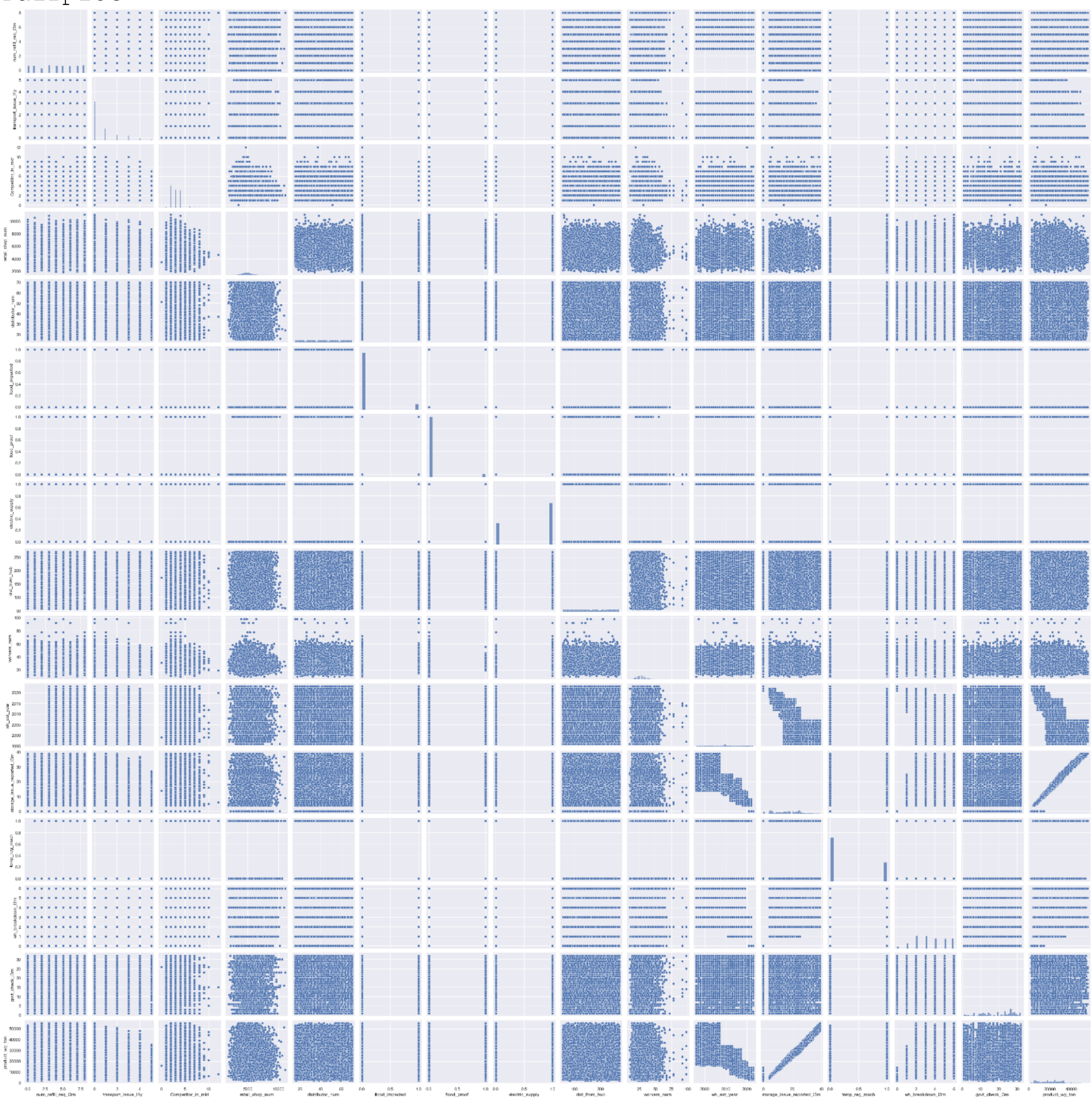

Figure 45:Pairplot of 7 Features

35. Heatmap

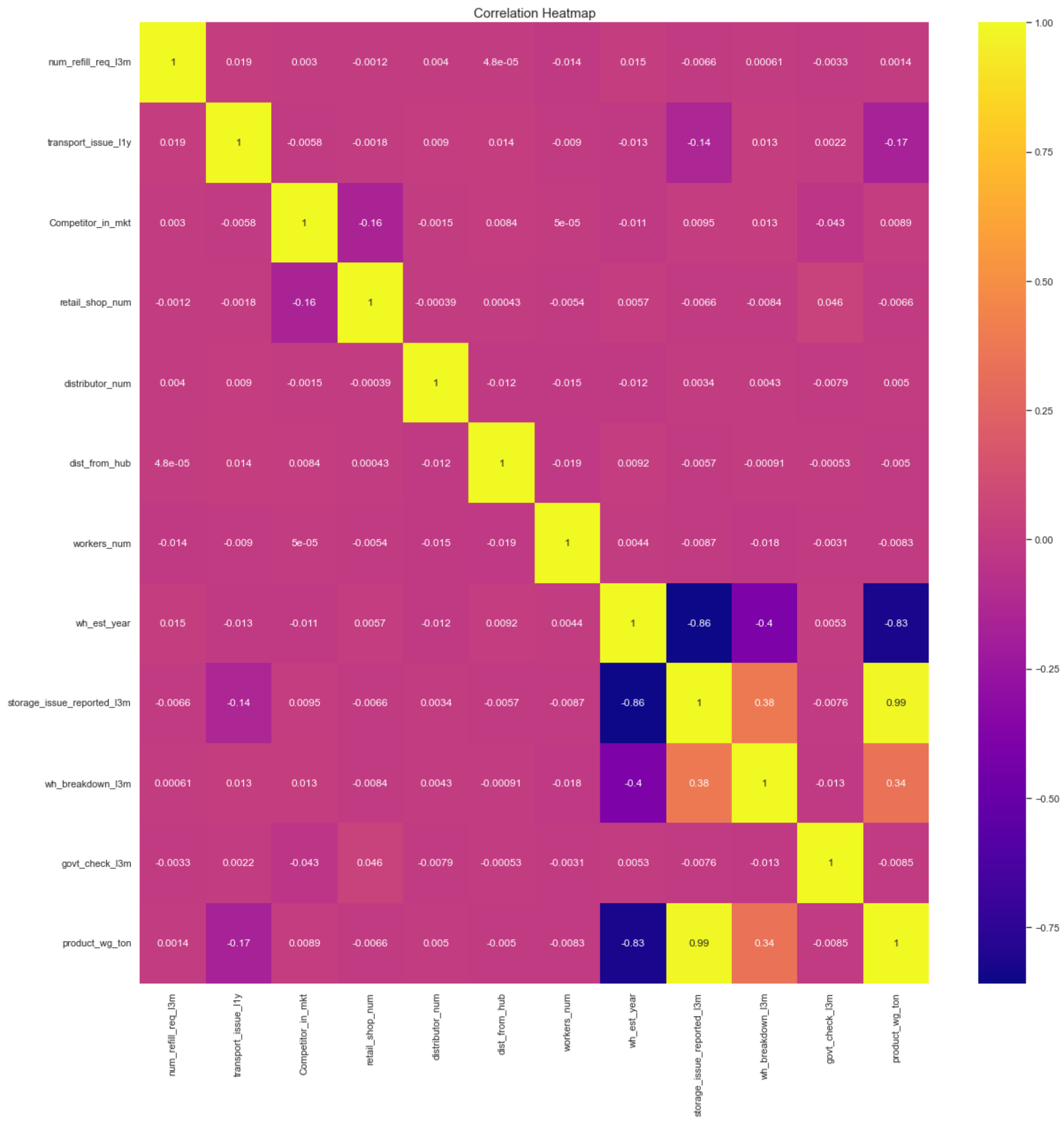

Figure 46:Correlation Heatmap

---X-------X-------X---