Business Report

SMDM Project Business Report DSBA

Capstone project – Supply Chain Management

Sanjay Srinivasan

PGP-DSBA Online

JULY' 21 Batch

Date: 03-07-2022

INDEX

List Of Figures

1. Introduction

Business Problem:

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.

Brief introduction about the problem statement and the need of solving it:

Brief introduction about the problem statement

The objective of this report is to find that, how the machine learning model supports the supply chain to overcome the demand and supply mismatch in every zone and warehouse. A FMCG company has entered into the instant noodles business two years back. The data is gathered based on the FMCG Company's demand and supply mismatch of the product instant noodles. The higher management has noticed that there is a mismatch in the demand and supply of instant noodles.

The demand and supply mismatch can be overcome by following these:

- First of all, finding the demand and supply mismatch.
- Secondly, find the optimum weight of the product been shipped to each warehouse at different zone and regions of the country.

Need of solving it:

1. Company will lose heavily on logistic movement of goods / products
2. In order to sale the product, goods has to be moved where there is high supply or high demand zone.
3. Can minimize the inventory based on ROP (Reorder product) and ROQ (Reorder Quantity)
4. Profit of the company can be increased.
5. Stock maintenance in the inventory can be done.
6. Product quality can be improved.

2. EDA and Business Implication

Non visual Understanding of data:

The Dataset consist of 25000 rows of data with 22 independent variable and 1 target variable.
7 – Object type variable, 2 – float variable and 14 – integer type variable.

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

Fig 1.1 Sample Dataset.

Description of each and every variable in the dataset.

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

```
(25000, 23)
```

Fig 1.2 Shape of the Dataset.

	num_refill_req_l3m	transport_issue_l1y	Competitor_in_mkt	retail_shop_num	distributor_num	flood_impacted	flood_proof	electric_supply	dist_from_hub
count	25000.000000	25000.000000	25000.000000	25000.000000	25000.000000	25000.000000	25000.000000	25000.000000	25000.000000
mean	4.089040	0.773680	3.104200	4985.711560	42.418120	0.098160	0.054640	0.656880	163.537320
std	2.606612	1.199449	1.141663	1052.825252	16.064329	0.297537	0.227281	0.474761	62.718609
min	0.000000	0.000000	0.000000	1821.000000	15.000000	0.000000	0.000000	0.000000	55.000000
25%	2.000000	0.000000	2.000000	4313.000000	29.000000	0.000000	0.000000	0.000000	109.000000
50%	4.000000	0.000000	3.000000	4859.000000	42.000000	0.000000	0.000000	1.000000	164.000000
75%	6.000000	1.000000	4.000000	5500.000000	56.000000	0.000000	0.000000	1.000000	218.000000
max	8.000000	5.000000	12.000000	11008.000000	70.000000	1.000000	1.000000	1.000000	271.000000

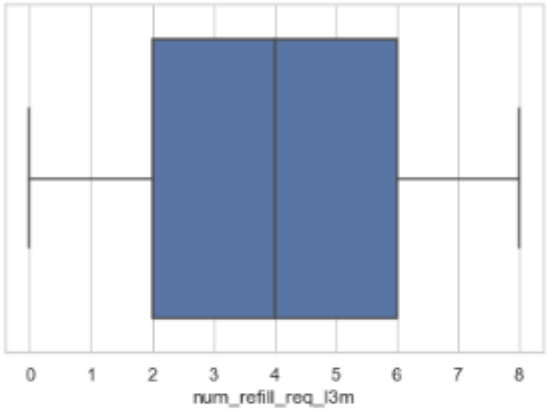
Fig 1.3 Description of the data.

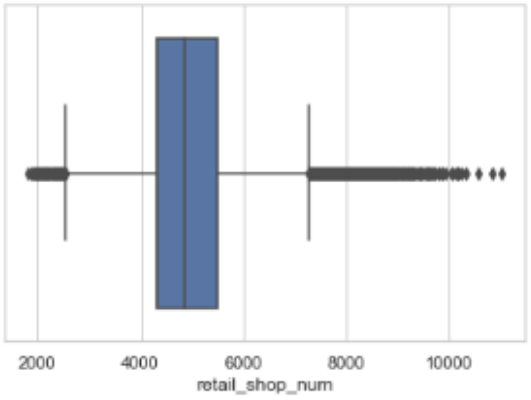
```
<class 'pandas.core.frame.DataFrame'>
Index: 25000 entries, WH_100000 to WH_124999
Data columns (total 23 columns):
WH_Manager_ID                  25000 non-null object
Location_type                  25000 non-null object
WH_capacity_size               25000 non-null object
zone                           25000 non-null object
WH_regional_zone               25000 non-null object
num_refill_req_l3m             25000 non-null int64
transport_issue_l1y            25000 non-null int64
Competitor_in_mkt              25000 non-null int64
retail_shop_num                25000 non-null int64
wh_owner_type                  25000 non-null object
distributor_num                25000 non-null int64
flood_impacted                 25000 non-null int64
flood_proof                    25000 non-null int64
electric_supply                25000 non-null int64
dist_from_hub                  25000 non-null int64
workers_num                    24010 non-null float64
wh_est_year                    13119 non-null float64
storage_issue_reported_l3m     25000 non-null int64
temp_reg_mach                  25000 non-null int64
approved_wh_govt_certificate   24092 non-null object
wh_breakdown_l3m               25000 non-null int64
govt_check_l3m                 25000 non-null int64
product_wg_ton                 25000 non-null int64
dtypes: float64(2), int64(14), object(7)
memory usage: 4.6+ MB
```

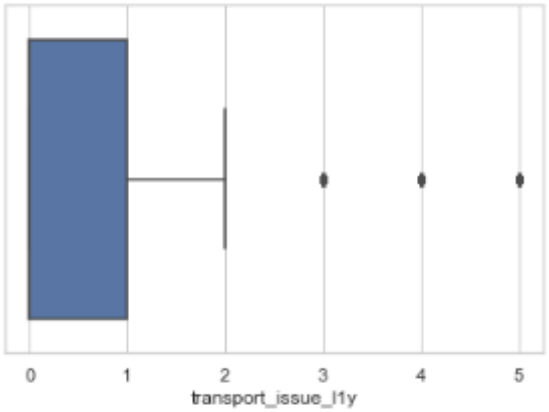
Fig 1.4 Variable info of the dataset.

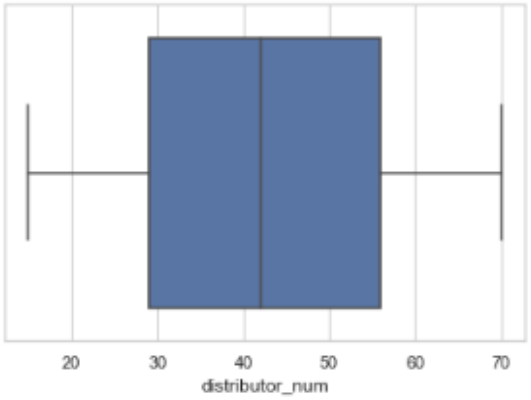
Visual Understanding of data:

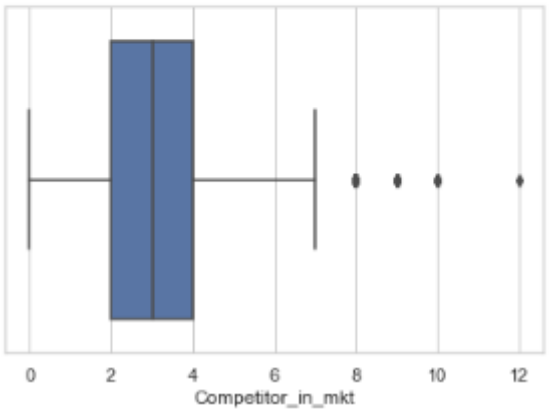
Univariate Analysis:

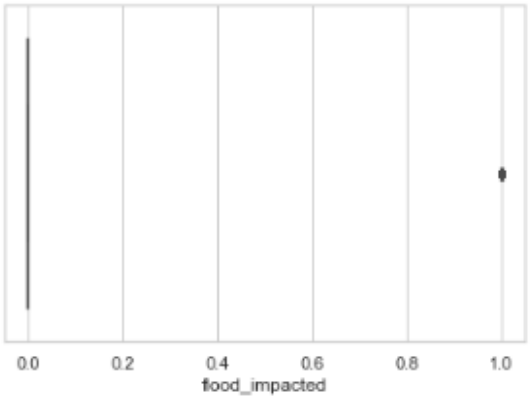

0
1
2
3
4
5
6
7
8
num_refill_req_l3m

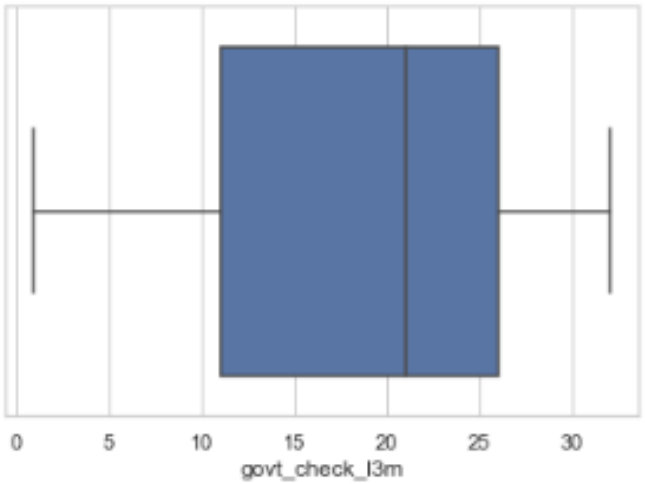

2000
4000
6000
8000
10000
retail_shop_num

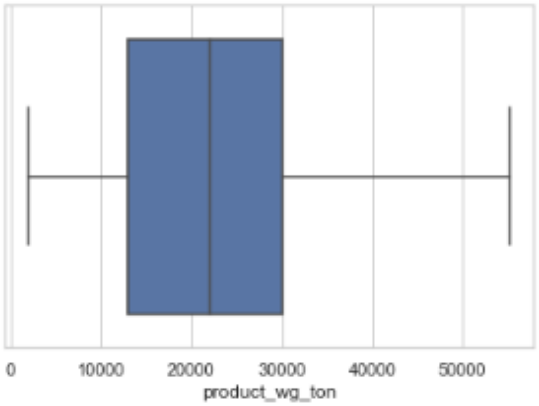

0
1
2
3
4
5
transport_issue_l1y


20
30
40
50
60
70
distributor_num


0
2
4
6
8
10
12
Competitor_in_mkt


0.0
0.2
0.4
0.6
0.8
1.0
flood_impacted


0
5
10
15
20
25
30
govt_check_l3m


0
10000
20000
30000
40000
50000
product_wg_ton

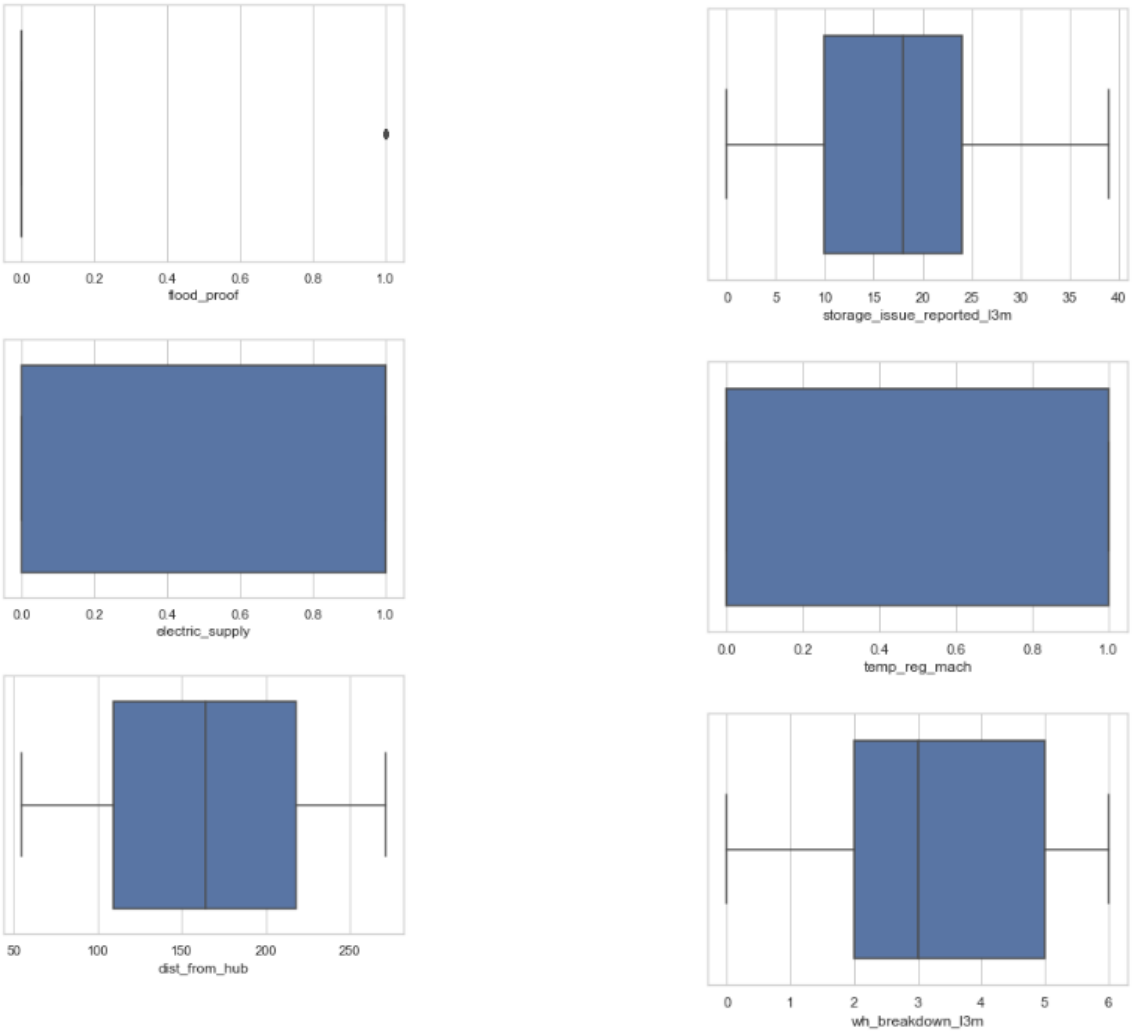


Fig 1.5 Univariate plot

Product_wg_ton – Target column is right skewed and there is no outliers present in the target column.
Competitor_in_mkt - This independent variable has outlier in the dataset
transport_issue_l1y - This independent variable has outlier in the dataset and right skewed values are present in the dataset.
retail_shop_num - This independent variable has more outliers and values are slightly right skewed.

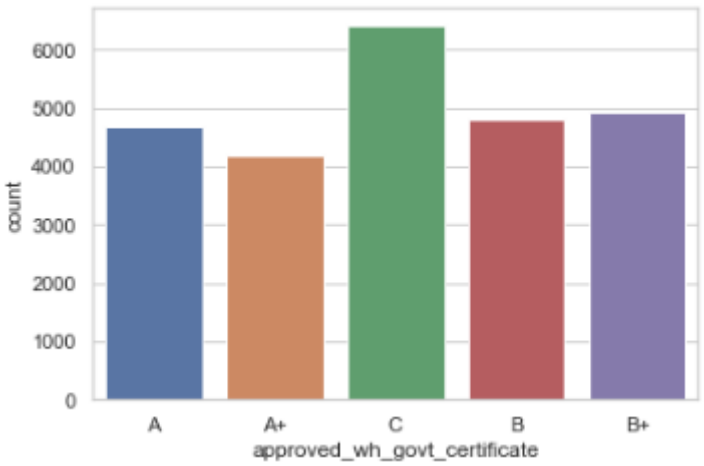


Fig 1.6 Count plot for approved_wh_govt_certificate

From this count plot C certificate has the highest number of warehouse with government certificate A+ has the least number of warehouse government certificate.

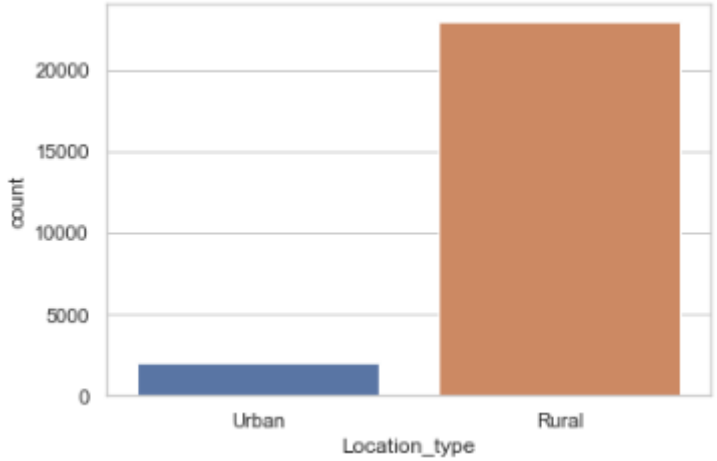


Fig 1.7 Count plot for Location_type

Most number of ware houses is located in the rural area.

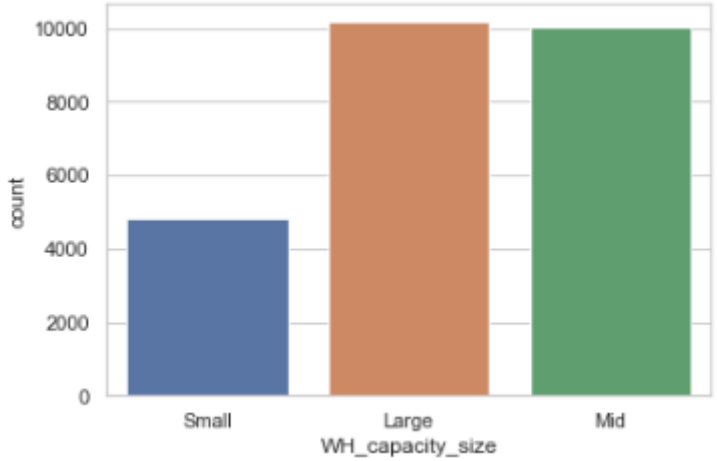


Fig 1.8 Count plot for WH_capacity_size

The Large capacity warehouse are having the ware house capacity higher than the other capacity ware houses.

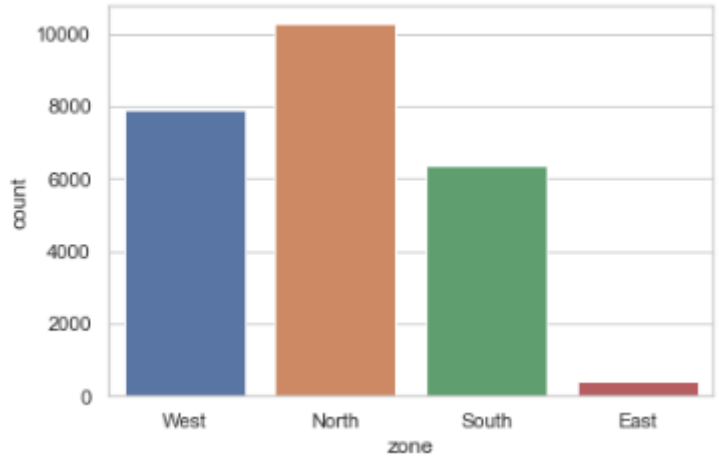


Fig 1.9 Count plot for Zone

North zone has the highest number of warehouse are built and East zone has the least number of ware houses.

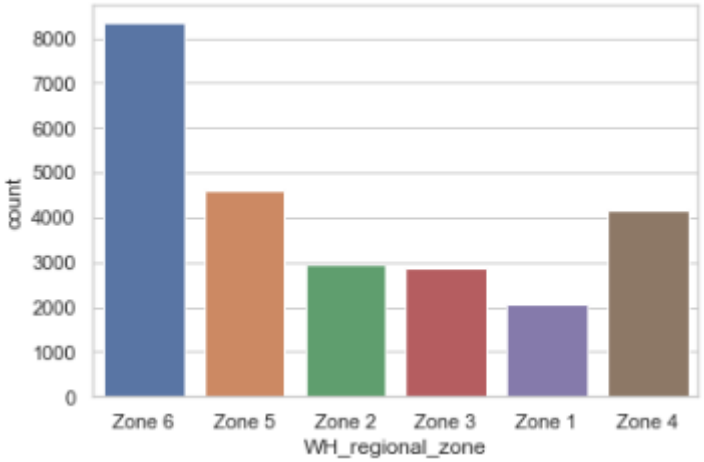


Fig 1.10 Count plot for Warehouse in regional zone

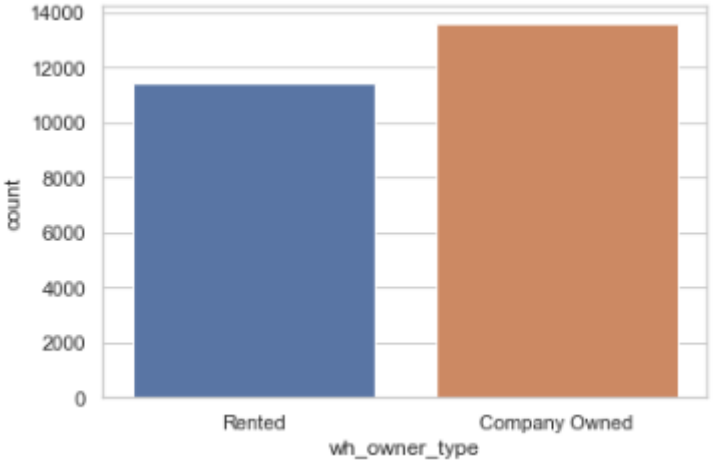


Fig 1.11 Count plot for Warehouse owner type

Most number of warehouse are owned by companies

Bivariate analysis

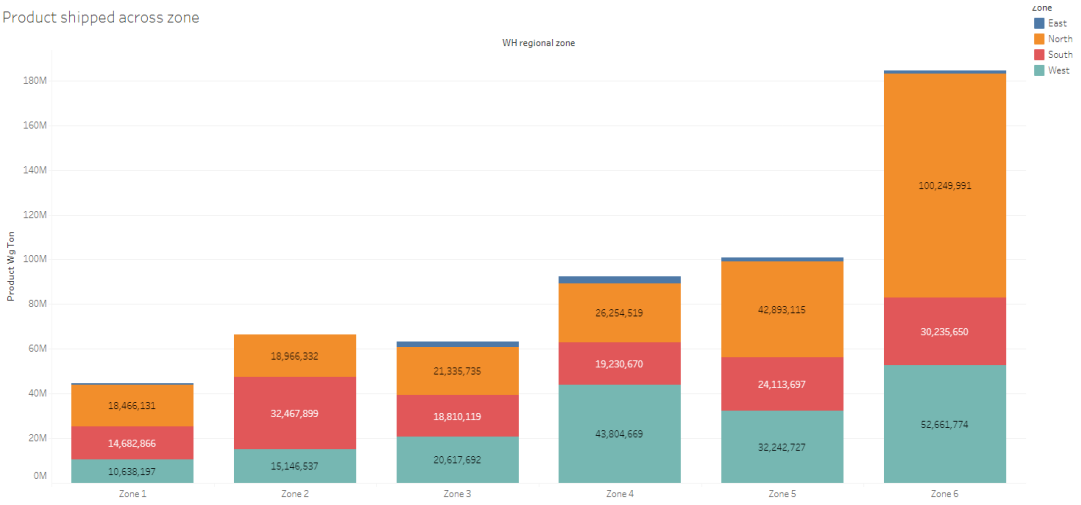


Fig 1.12 Product Shipped across zone

From this we can infer that product shipped in the east zone is very less. In North zone , Product shipping is higher every zone. We can infer that the supply is higher in the north zone and supply is lower in east zone.

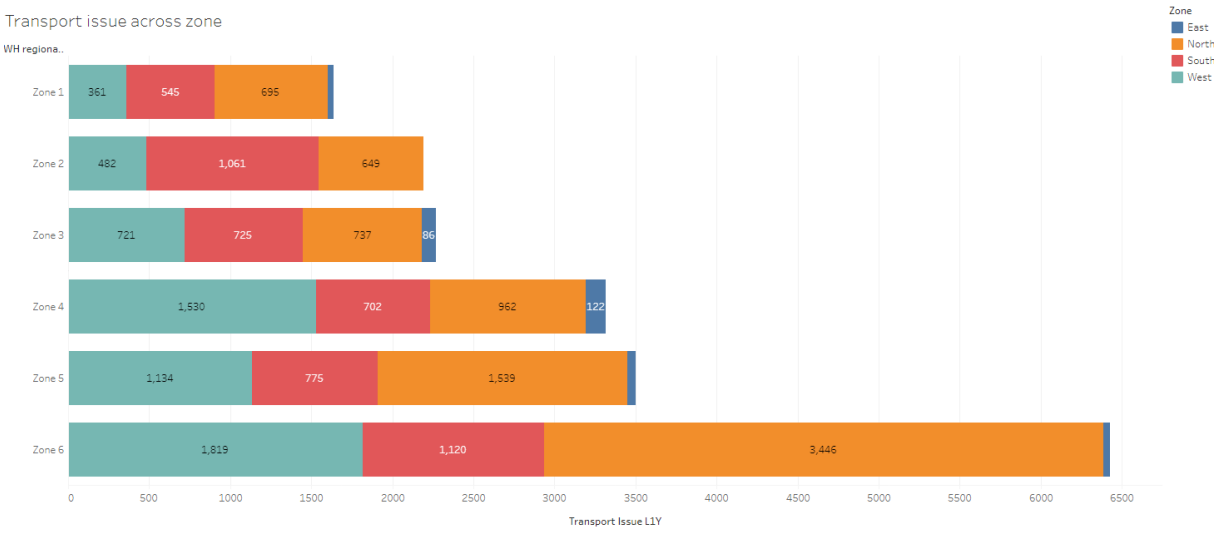


Fig 1.13 Transport issue of shipment across zone

Zone 6 has more number of transport issues when compared with other zone. In zone 6, maximum number of transport issue occurred in north zone.

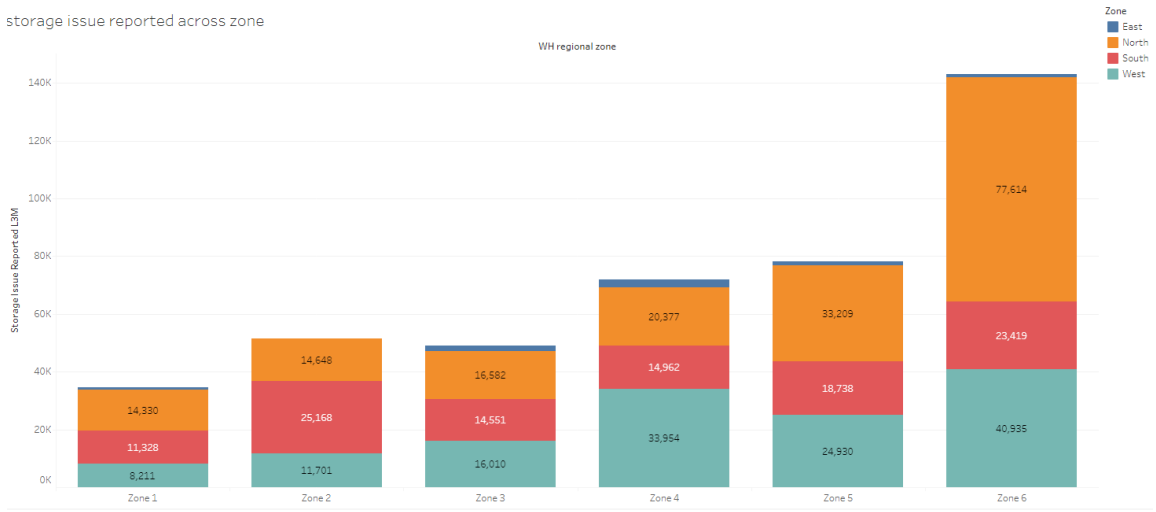


Fig 1.14 Storage issue of shipment across zone

The Zone -6 has the high supply products and high storage issues . The Storage area (Warehouse) has the limitation of stocking up the entire shipped product. Each Warehouse in every zone has the issue of storing issue in the warehouse.

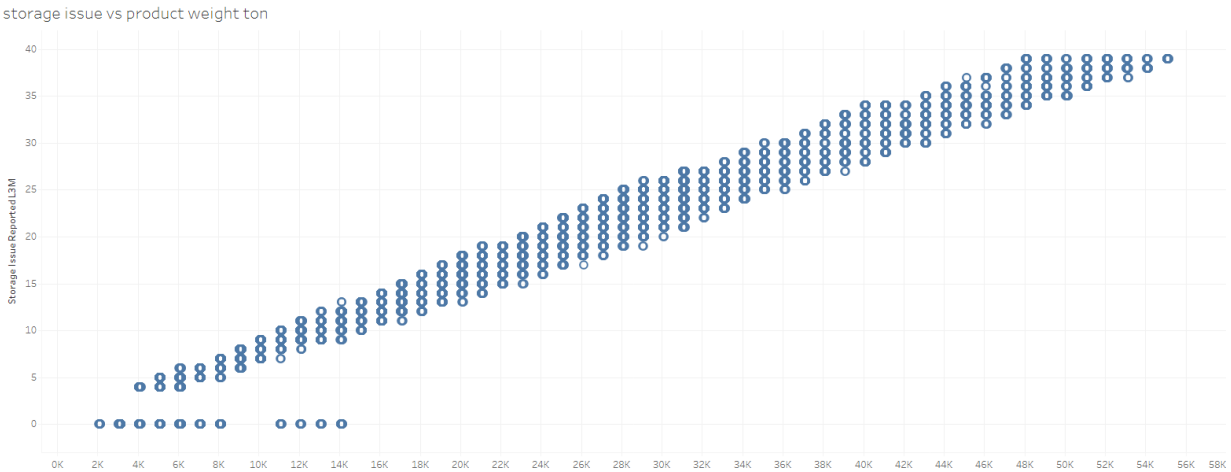


Fig 1.15 Storage issue across product weight ton

Multivariate Analysis:

	num_refill_req_l3m	transport_issue_l1y	Competitor_in_mkt	retail_shop_num	distributor_num	flood_impacted	flood_proof	electric_
num_refill_req_l3m	1.000000	0.018549	0.002985	-0.001186	0.003995	-0.010548	-0.001123	-0.0
transport_issue_l1y	0.018549	1.000000	-0.005826	-0.001826	0.008993	-0.009596	0.000022	-0.0
Competitor_in_mkt	0.002985	-0.005826	1.000000	-0.156943	-0.001492	0.009338	-0.003444	0.0
retail_shop_num	-0.001186	-0.001826	-0.156943	1.000000	-0.000395	-0.003774	0.007223	-0.0
distributor_num	0.003995	0.008993	-0.001492	-0.000395	1.000000	0.004611	-0.003409	0.0
flood_impacted	-0.010548	-0.009596	0.009338	-0.003774	0.004611	1.000000	0.107015	0.
flood_proof	-0.001123	0.000022	-0.003444	0.007223	-0.003409	0.107015	1.000000	0.
electric_supply	-0.007959	-0.009299	0.001759	-0.009207	0.000454	0.164815	0.114811	1.0
dist_from_hub	0.000048	0.014336	0.008407	0.000429	-0.011838	0.000749	-0.005315	-0.0
workers_num	-0.013764	-0.009004	0.000050	-0.005406	-0.014682	0.168425	0.041228	0.:
wh_est_year	0.015363	-0.012910	-0.011202	0.005721	-0.012295	-0.000668	-0.003329	-0.0
storage_issue_reported_l3m	-0.006602	-0.144327	0.009543	-0.006632	0.003396	-0.003157	-0.002712	-0.0
temp_reg_mach	0.260928	0.018207	0.009524	-0.001273	0.002827	-0.008554	0.005636	-0.0
wh_breakdown_l3m	0.000608	0.012990	0.012733	-0.008420	0.004286	-0.001744	-0.005151	-0.0
govt_check_l3m	-0.003302	0.002190	-0.043455	0.045749	-0.007934	0.000587	-0.003600	-0.0
product_wg_ton	0.001415	-0.173992	0.008884	-0.006615	0.004999	-0.002299	-0.000441	-0.0

Fig 1.16 Correlation data

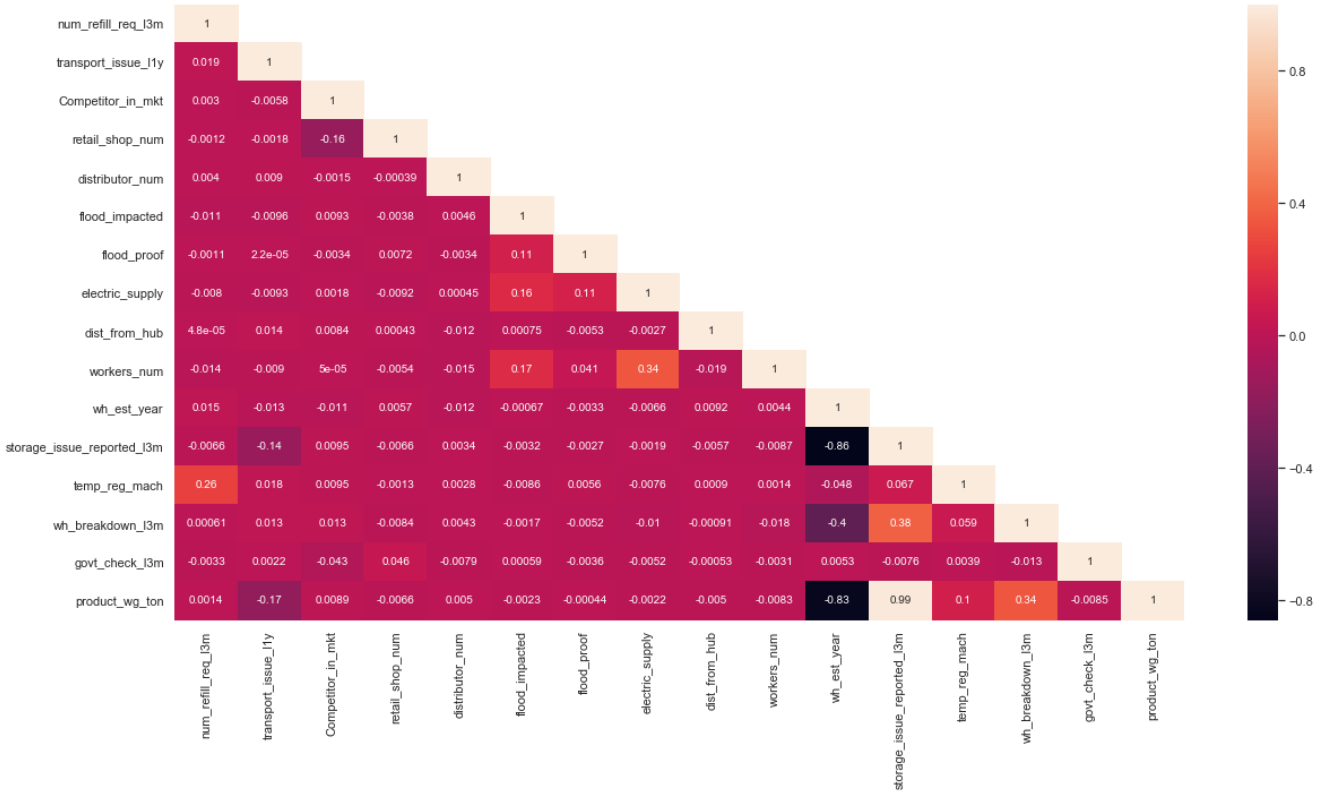


Fig 1.17 Heatmap of Correlation plot

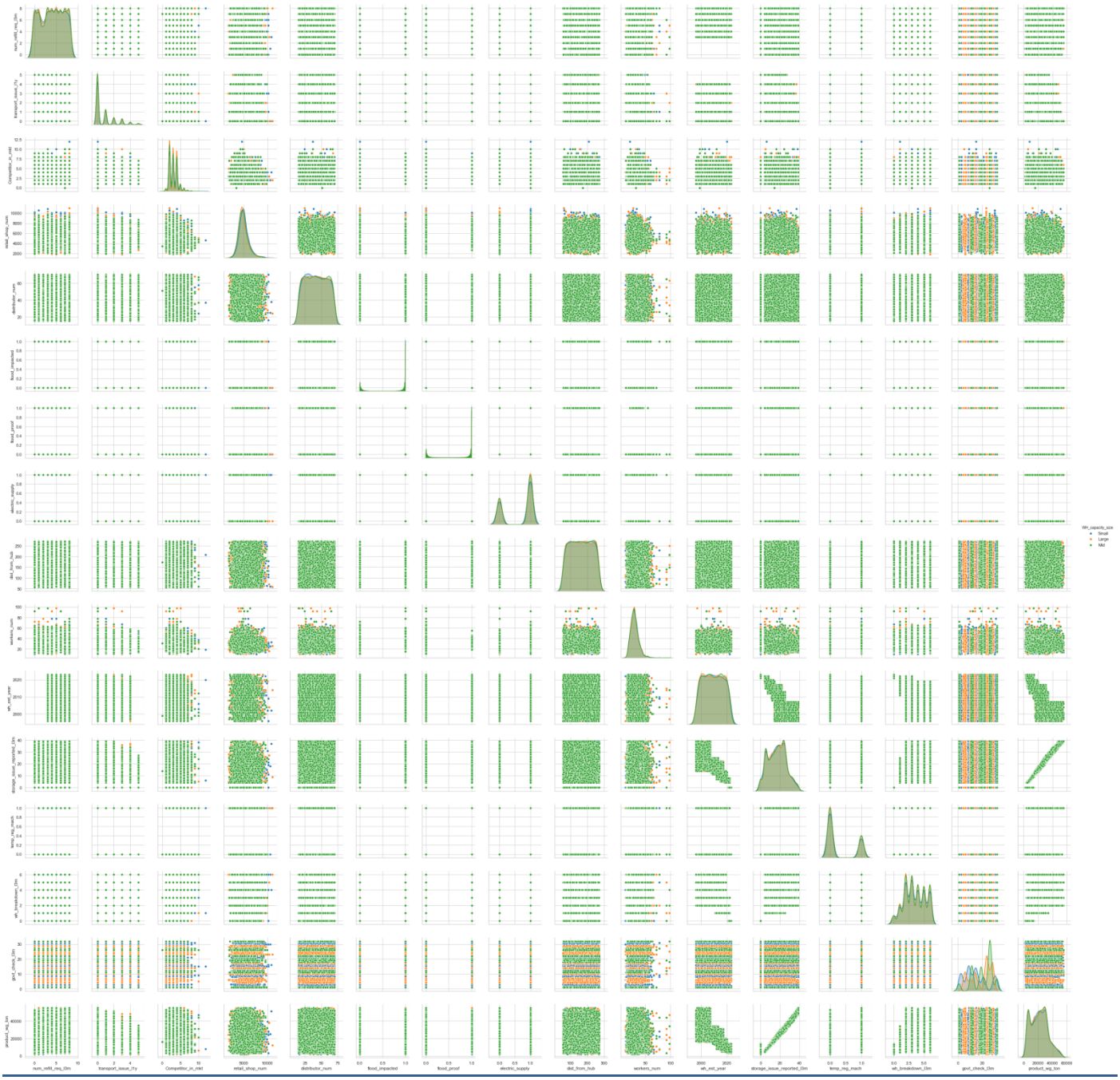


Fig 1.18 Pairplot

How your analysis is impacting the business?

From the above analysis, we can infer that, there is high supply in zone 6 when compared to other zone and Transport issues and storage issues are there in the every zone.

There is no correlation between the data.

3. Data Cleaning and Pre-processing

a.)Approach used for identifying and treating missing values and outlier treatment (and why). Need for variable transformation (if any). Variables removed or added and why (if any)

WH_Manager_ID	0
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	

WH_Manager_ID	0.000
Location_type	0.000
WH_capacity_size	0.000
zone	0.000
WH_regional_zone	0.000
num_refill_req_l3m	0.000
transport_issue_l1y	0.000
Competitor_in_mkt	0.000
retail_shop_num	0.000
wh_owner_type	0.000
distributor_num	0.000
flood_impacted	0.000
flood_proof	0.000
electric_supply	0.000
dist_from_hub	0.000
workers_num	3.960
wh_est_year	47.524
storage_issue_reported_l3m	0.000
temp_reg_mach	0.000
approved_wh_govt_certificate	3.632
wh_breakdown_l3m	0.000
govt_check_l3m	0.000
product_wg_ton	0.000
dtype: float64	

Fig 1.19 Missing values present in the data

Outlier Treatment:

From the boxplot we can infer that, outlier treatment is not required for this dataset and discrete values are present in the data.

Need for variable transformation (if any):

Yes, Variable transformation is needed, because object type(text type) data are present in the data

```
(25000, 30)
```

Fig 1.20 Shape of the dataset after variable transformation.

Ware_house_ID	Location_type_Urban	WH_capacity_size_Mid	WH_capacity_size_Small	zone_North	zone_South	zone_West	WH_regional_zone_Zone 2	WH_regional
WH_100000	1	0	1	0	0	1	0	
WH_100001	0	0	0	1	0	0	0	
WH_100002	0	1	0	0	1	0	1	
WH_100003	0	1	0	1	0	0	0	
WH_100004	0	0	0	1	0	0	0	

Fig 1.21 sample dataset after variable transformation.

```
<class 'pandas.core.frame.DataFrame'>
Index: 12127 entries, WH_100004 to WH_124999
Data columns (total 31 columns):
Location_type_Urban                12127 non-null uint8
WH_capacity_size_Mid               12127 non-null uint8
WH_capacity_size_Small             12127 non-null uint8
zone_North                         12127 non-null uint8
zone_South                         12127 non-null uint8
zone_West                          12127 non-null uint8
WH_regional_zone_Zone 2            12127 non-null uint8
WH_regional_zone_Zone 3            12127 non-null uint8
WH_regional_zone_Zone 4            12127 non-null uint8
WH_regional_zone_Zone 5            12127 non-null uint8
WH_regional_zone_Zone 6            12127 non-null uint8
wh_owner_type_Rented               12127 non-null uint8
approved_wh_govt_certificate_A+    12127 non-null uint8
approved_wh_govt_certificate_B     12127 non-null uint8
approved_wh_govt_certificate_B+    12127 non-null uint8
approved_wh_govt_certificate_C     12127 non-null uint8
num_refill_req_l3m                 12127 non-null int64
transport_issue_l1y                12127 non-null int64
Competitor_in_mkt                  12127 non-null int64
retail_shop_num                    12127 non-null int64
distributor_num                    12127 non-null int64
flood_impacted                     12127 non-null int64
flood_proof                        12127 non-null int64
electric_supply                    12127 non-null int64
dist_from_hub                      12127 non-null int64
workers_num                        12127 non-null float64
wh_est_year                        12127 non-null float64
temp_reg_mach                      12127 non-null int64
wh_breakdown_l3m                   12127 non-null int64
govt_check_l3m                     12127 non-null int64
product_wg_ton                     12127 non-null int64
dtypes: float64(2), int64(13), uint8(16)
memory usage: 1.7+ MB
```

Fig 1.22 info of the dataset after variable transformation and missing value treatment

Variables removed or added and why (if any):

- As 'WH_Manager_ID' and 'Ware_house_ID' are unique values, we are dropping 'WH_Manager_ID' and setting 'Ware_house_ID' as an index value.
- As 'wh_est_year' having 48% of null values, so we are dropping 'wh_est_year' independent variable from the dataframe.
- 'storage_issue_reported_l3m' independent variable is highly correlated with the target column. So 'storage_issue_reported_l3m' can also be dropped from the dataframe.

4. Model building

Clear on why was a particular model(s) chosen. Effort to improve model performance.

Clear on why was a particular model(s) chosen:

Random Forest Regressor model:

- It reduces over fitting in decision trees and helps to improve the accuracy
- It is flexible to both classification and regression problems
- It works well with both categorical and continuous value.

```
Fitting 3 folds for each of 324 candidates, totalling 972 fits

RandomizedSearchCV(cv=3, estimator=RandomForestRegressor(), n_iter=1000,
                   n_jobs=-1,
                   param_distributions={'max_depth': [20, 40, 60, 80, 100, 120],
                                        'max_features': ['auto', 'sqrt'],
                                        'min_samples_leaf': [1, 2, 4],
                                        'min_samples_split': [2, 5, 10],
                                        'n_estimators': [150, 200, 250]},
                   random_state=42, verbose=2)
```

Fig 1.23 Random Forest Regressor model initializing

```
{'n_estimators': 200,
 'min_samples_split': 10,
 'min_samples_leaf': 4,
 'max_features': 'auto',
 'max_depth': 20}
```

Fig 1.24 Random Forest Regressor model best parameters

```
RandomForestRegressor(max_depth=20, min_samples_leaf=4, min_samples_split=10,
                      n_estimators=200)
```

Fig 1.25 Random Forest Regressor model best Estimators

```
R square value for training data  0.8995615739370619
R square value for testing data  0.7528227319905008
```

Fig 1.26 Random Forest Regressor model R-Square value

```
Mean squared error for the training data is  0.1000949310077825
Root Mean squared error for the training data is  0.3163778295136726
Mean squared error for the testing data is  0.24992776370322636
Root Mean squared error for the testing data is  0.4999277584843898
```

Fig 1.27 Random Forest Regressor model MSE and RMSE value

```
Mean Absolute Error: 0.24338012476772533
Mean Absolute Error: 0.39539155924888203
```

Fig 1.28 Random Forest Regressor model MAE value

```
Mean Absolute Percentage Error: 1.2531876379631535
```

Fig 1.29 Random Forest Regressor model MAPE value

Ada Boosting Regressor model:

- Adaboost is less prone to overfitting as the input parameters are not jointly optimized.
- The accuracy of weak classifiers can be improved by using Adaboost.

```
Fitting 3 folds for each of 48 candidates, totalling 144 fits

RandomizedSearchCV(cv=3, estimator=AdaBoostRegressor(random_state=42),
                   n_iter=100, n_jobs=-1,
                   param_distributions={'learning_rate': [10, 20, 30, 50],
                                        'loss': ['linear', 'square',
                                                 'exponential'],
                                        'n_estimators': [100, 200, 500, 1000]},
                   verbose=2)
```

Fig 1.30 ADA Boosting Regressor model Intializing

```
{'n_estimators': 100, 'loss': 'exponential', 'learning_rate': 10}
```

Fig 1.31 ADA Boosting Regressor model best parameters

```
AdaBoostRegressor(learning_rate=10, loss='exponential', n_estimators=100,
                  random_state=42)
```

Fig 1.32 ADA Boosting Regressor model best Estimators

```
R square value for training data  0.4396371508425988
R square value for testing data  0.42895869298009626
```

Fig 1.33 ADA Boosting Regressor model R-Square value

```
Mean squared error for the training data is  0.5585990280305757
Root Mean squared error for the training data is  0.7473948274042146
Mean squared error for the testing data is  0.5781465228555599
Root Mean squared error for the testing data is  0.7603594694981841
```

Fig 1.34 ADA Regressor model MSE and RMSE value

```
Mean Absolute Error: 0.5969521872099579
Mean Absolute Error: 0.6034548305222793
```

Fig 1.35 ADA Boosting Regressor model MAE value

```
Mean Absolute Percentage Error: 1.2758834043466822
```

Fig 1.36 ADA Boosting Regressor model MAPE value

Bagging Regressor model:

- **Ease of implementation**
- **Reduction of variance**
- significantly raises the stability of models in improving accuracy

```
Fitting 3 folds for each of 30 candidates, totalling 90 fits

RandomizedSearchCV(cv=3, estimator=BaggingRegressor(), n_iter=1000, n_jobs=-1,
                   param_distributions={'max_features': [1, 2, 4, 6, 8],
                                        'max_samples': [0.5, 1],
                                        'n_estimators': [50, 100, 150]},
                   random_state=42, verbose=2)
```

Fig 1.37 Bagging Regressor model Intializing

```
{'n_estimators': 50, 'max_samples': 0.5, 'max_features': 8}
```

Fig 1.38 Bagging Regressor model best parameters

```
BaggingRegressor(max_features=8, max_samples=0.5, n_estimators=50)
```

Fig 1.39 Bagging Regressor model best Estimators

```
R square value for training data  0.6027028141291111
R square value for testing data  0.2558257082067964
```

Fig 1.40 Bagging Regressor model R-Square value

```
Mean squared error for the training data is  0.35058320815167454
Root Mean squared error for the training data is  0.5921006740003549
Mean squared error for the testing data is  0.660298745281547
Root Mean squared error for the testing data is  0.8125876846725817
```

Fig 1.41 Bagging Regressor model MSE and RMSE value

```
Mean Absolute Error: 0.4784914880915685
Mean Absolute Error: 0.652715004989947
```

Fig 1.42 Bagging Regressor model MAE value

```
Mean Absolute Percentage Error: 1.0169903284964843
```

Fig 1. 43 Bagging Regressor model MAPE value

Ridge Regression model:

- Avoids overfitting model
- Performs well in large multivariate data.

```
GridSearchCV(cv=5, estimator=Ridge(),
             param_grid={'alpha': [1e-15, 1e-10, 1e-08, 0.001, 0.01, 1, 5, 10,
                                   20, 30, 35, 40, 45, 50, 55, 100],
                         'fit_intercept': [True],
                         'random_state': [42, 273, 450, 236, 970],
                         'solver': ['auto', 'svd'], 'tol': [0.001, 0.0001]},
             scoring='neg_mean_squared_error')
```

Fig 1.44 Ridge Regressor model Intializing

```
{'alpha': 20,
 'fit_intercept': True,
 'random_state': 42,
 'solver': 'auto',
 'tol': 0.001}
```

Fig 1.45 Ridge Regressor model best parameters

```
Ridge(alpha=20, random_state=42)
```

Fig 1.46 Ridge Regressor model best Estimators

```
R square value for training data  0.7118444078318602
R square value for testing data  0.7148166970351346
```

Fig 1.47 Ridge Regressor model R-Square value

```
Mean squared error for the training data is  0.28724858178719953
Root Mean squared error for the training data is  0.535955764767205
Mean squared error for the testing data is  0.2887317133782299
Root Mean squared error for the testing data is  0.537337615822892
```

Fig 1.48 Ridge Regressor model MSE and RMSE value

```
Mean Absolute Error: 0.4249243262763847
Mean Absolute Error: 0.4224104606872155
```

Fig 1.49 Ridge Regressor model MAE value

```
Mean Absolute Percentage Error: 1.2785270014265517
```

Fig 1.50 Ridge Regressor model MAPE value

ANN Regression model:

- Storing information on the entire network
- Having a distributed memory
- Parallel processing capability

```
Fitting 3 folds for each of 192 candidates, totalling 576 fits

RandomizedSearchCV(cv=3, estimator=MLPRegressor(), n_iter=1000, n_jobs=-1,
                   param_distributions={'activation': ['identity', 'logistic',
                                                      'tanh', 'relu'],
                                        'alpha': [5e-05, 0.0005],
                                        'hidden_layer_sizes': [1, 50],
                                        'max_iter': [100, 300],
                                        'solver': ['lbfgs', 'sgd', 'adam'],
                                        'tol': [0.0001, 1e-06]},
                   random_state=42, verbose=2)
```

Fig 1.51 ANN Regressor model Intializing

```
{'tol': 1e-06,
 'solver': 'lbfgs',
 'max_iter': 100,
 'hidden_layer_sizes': 1,
 'alpha': 5e-05,
 'activation': 'relu'}
```

Fig 1.52 ANN Regressor model best parameters

```
MLPRegressor(alpha=5e-05, hidden_layer_sizes=1, max_iter=100, solver='lbfgs',
             tol=1e-06)
```

Fig 1.53 ANN Regressor model best Estimators

```
R square value for training data  0.7438463677219264
R square value for testing data  0.7385995100135008
```

Fig 1.54 ANN Regressor model R-Square value

```
Mean squared error for the training data is  0.25534709874153905
Root Mean squared error for the training data is  0.5053188090122305
Mean squared error for the testing data is  0.26463514591835285
Root Mean squared error for the testing data is  0.5144270073765109
```

Fig 1.55 ANN Regressor model MSE and RMSE value

```
Mean Absolute Error: 0.4004900386156925
Mean Absolute Error: 0.41033367908514595
```

Fig 1.56 ANN Regressor model MAE value

```
Mean Absolute Percentage Error: 1.371283628665219
```

Fig 1.57 ANN Regressor model MAPE value

Effort to improve model performance.

Hyper Parameter tuning has been performed for increasing the performance of the model.

5. Model validation

How was the model validated? Just accuracy, or anything else too?

The model was validated through calculating the multiple parameters for each model.

Parameters are

- R Squared value (R^2 value)
- Mean Squared Error value (MSE)
- Root Mean Squared Error value (MSE)
- Mean Absolute Error value (MAE)
- Mean Absolute Percentage Error value (MAPE)

A good or best model needs to have the value in the range.

- RMSE - 0.2-0.5
- Mean absolute error (MAE) -10
- R^2 value - near to 1
- Mean squared error (MSE) - near to 0

Best model is identified from the calculated value of each parameter for each model are:

	MAE	MAPE	MSE	R Square value	R Square value train	RMSE	Rsquare value train	intercept	RMSE / MAE
Random Forest Regression model tuning	0.395392	1.253188	0.249928	0.752823	0.899562	0.499928	NaN	Nan	1.264387
RF Model	0.396622	1.248043	0.253430	0.749685	0.964797	0.503418	NaN	Nan	1.269263
ann Regression model tuning	0.410334	1.371284	0.264635	0.738600	0.743846	0.514427	NaN	Nan	1.253680
ANN model MLP Regressor	0.412270	1.329751	0.270512	0.732813	0.777603	0.520107	NaN	Nan	1.261569
Support Vector Model Regression	0.420699	1.336387	0.286953	0.716574	0.824993	0.535680	NaN	-0.55373	1.273309
Huber Regression	0.422009	1.283370	0.288428	0.715121	0.711770	0.537055	NaN	0	1.272617
Linear Regression	0.422296	1.280847	0.288671	0.714877	NaN	0.537281	0.711839	-0.00397057	1.272284
Ridge Regression	0.422299	1.280297	0.288678	0.714870	0.711848	0.537287	NaN	-0.00399231	1.272293
Ridge Regression with model Tuning	0.422410	1.278527	0.288732	0.714817	0.711844	0.537338	NaN	Nan	1.272075
AdaBoost Regression model	0.461022	1.484113	0.307914	0.695871	0.702167	0.554900	NaN	Nan	1.203630
ADA boosting Regression model tuning	0.603455	1.275883	0.578147	0.428959	0.439637	0.760359	NaN	Nan	1.260011
Bagging Regression model tuning	0.652715	1.016990	0.660299	0.255826	0.602703	0.812588	NaN	Nan	1.244935
Lasso Regression	0.817240	1.000787	1.012623	-0.000178	0.000000	1.006292	NaN	-0.00268507	1.231329

Fig 1.58 Sorting best model based on Least MSE

6. Final interpretation / recommendation

Detailed recommendations for the management/client based on the analysis done.

	Features	Importance
0	wh_est_year	0.816591
1	retail_shop_num	0.024038
2	dist_from_hub	0.023038
3	distributor_num	0.020069
4	workers_num	0.016743
5	govt_check_l3m	0.014216
6	transport_issue_l1y	0.013239

Fig 1.59 Feature Selection based on the best model

These are top 7 features that are important for this data.

Wh_est_year – During early periods the supply was higher and the year progress the shipping of product weight is reduced. May be population has been increased over the years and production/manufacturing of the product has not been increased.

retail_shop_num – The Availability of the products in the shops, needs to be increased. Because demand is higher and supply is less.

dist_from_hub – The warehouse may be located far from the manufacturing plant. This may take lot off time for transportation of goods and cost of transportation may increase.

Insights and Recommendations:

- Inventory can be managed based on the optimization of warehouse details.
- Transportation cost can be minimized.
- Supply of goods can be time delay.
- Better collaboration with suppliers.
- Better quality control.
- Shipping optimisation.
- Reduced inventory and overhead costs.
- Improved risk mitigation.
- Stronger cash flow.

Recommendations:

- Need to increase warehouse in the urban areas.
- East zone has the least warehouse and the demand and supply is low. Need to promote more about the product and offers in the East zone.
- Need to give more offers to the Dealers for buying products as bulk orders.
- Need to employee more workers in the warehouse for monitoring the inventory. Increasing of workers in warehouse can help the workers to monitor the product.
- Need to give more offers to the customer based on the zone to sell more products as the competitor are there in the market.
- Minimum order and Maximum orders needs to be implemented for the Dealers or wholeseller for the Quality of the product.