Conducting Hypothesis Testing on supply chain management dataset

In [1]:

```
import numpy as np
import pandas as pd
```

In [2]:

```
import warnings
warnings.filterwarnings('ignore')
```

In [3]:

```
df=pd.read_csv("C:\\Users\\vamsi\\OneDrive\\Desktop\\Vamshi Data\\Supplychain train
```

In [4]:

```
df.head(10)
```

Out[4]:

	Ware_house_ID	WH_Manager_ID	Location_type	WH_capacity_size	zone	WH_regional_zone	nur
0	WH_100000	EID_50000	Urban	Small	West	Zone 6	
1	WH_100001	EID_50001	Rural	Large	North	Zone 5	
2	WH_100002	EID_50002	Rural	Mid	South	Zone 2	
3	WH_100003	EID_50003	Rural	Mid	North	Zone 3	
4	WH_100004	EID_50004	Rural	Large	North	Zone 5	
5	WH_100005	EID_50005	Rural	Small	West	Zone 1	
6	WH_100006	EID_50006	Rural	Large	West	Zone 6	
7	WH_100007	EID_50007	Rural	Large	North	Zone 5	
8	WH_100008	EID_50008	Rural	Small	South	Zone 6	
9	WH_100009	EID_50009	Rural	Small	South	Zone 6	

10 rows × 24 columns

In [5]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 22150 entries, 0 to 22149
Data columns (total 24 columns):
 #   Column                        Non-Null Count  Dtype 
---  ------                        --------------  ----- 
 0   Ware_house_ID                 22150 non-null  object
 1   WH_Manager_ID                 22150 non-null  object
 2   Location_type                 22150 non-null  object
 3   WH_capacity_size              22150 non-null  object
 4   zone                          22150 non-null  object
 5   WH_regional_zone              22150 non-null  object
 6   num_refill_req_l3m            22150 non-null  int64 
 7   transport_issue_l1y           22150 non-null  int64 
 8   Competitor_in_mkt             22150 non-null  int64 
 9   retail_shop_num               22150 non-null  int64 
 10  wh_owner_type                 22150 non-null  object
 11  distributor_num               22150 non-null  int64
 12  flood_impacted                22150 non-null  int64
 13  flood_proof                   22150 non-null  int64
 14  electric_supply               22150 non-null  int64
 15  dist_from_hub                 22150 non-null  int64
 16  workers_num                   21273 non-null  float64
 17  wh_est_year                   11605 non-null  float64
 18  storage_issue_reported_l3m    22150 non-null  int64
 19  temp_reg_mach                 22150 non-null  int64
 20  approved_wh_govt_certificate  21345 non-null  object
 21  wh_breakdown_l3m              22150 non-null  int64
 22  govt_check_l3m                22150 non-null  int64
 23  product_wg_ton                22150 non-null  int64
dtypes: float64(2), int64(14), object(8)
memory usage: 4.1+ MB
```

In [6]:

```
df.dtypes
```

Out[6]:

```
Ware_house_ID                    object
WH_Manager_ID                    object
Location_type                    object
WH_capacity_size                 object
zone                             object
WH_regional_zone                 object
num_refill_req_l3m                int64
transport_issue_l1y               int64
Competitor_in_mkt                 int64
retail_shop_num                   int64
wh_owner_type                    object
distributor_num                   int64
flood_impacted                    int64
flood_proof                       int64
electric_supply                   int64
dist_from_hub                     int64
workers_num                     float64
wh_est_year                     float64
storage_issue_reported_l3m        int64
temp_reg_mach                     int64
approved_wh_govt_certificate     object
wh_breakdown_l3m                  int64
govt_check_l3m                    int64
product_wg_ton                    int64
dtype: object
```

Descrpitive Stats

In [7]:

```
df.describe().T
```

Out[7]:

	count	mean	std	min	25%	50%	75%
num_refill_req_l3m	22150.0	4.097020	2.606289	0.0	2.00	4.0	6.0
transport_issue_l1y	22150.0	0.777201	1.201747	0.0	0.00	0.0	1.0
Competitor_in_mkt	22150.0	3.103928	1.142886	0.0	2.00	3.0	4.0
retail_shop_num	22150.0	4983.115711	1050.634225	1821.0	4309.25	4859.0	5499.0
distributor_num	22150.0	42.386998	16.057730	15.0	29.00	42.0	56.0
flood_impacted	22150.0	0.098691	0.298253	0.0	0.00	0.0	0.0

	count	mean	std	min	25%	50%	75%
flood_proof	22150.0	0.054492	0.226991	0.0	0.00	0.0	0.0
electric_supply	22150.0	0.656072	0.475028	0.0	0.00	1.0	1.0
dist_from_hub	22150.0	163.613725	62.660709	55.0	109.00	164.0	218.0
workers_num	21273.0	28.936398	7.843431	10.0	24.00	28.0	33.0
wh_est_year	11605.0	2009.401206	7.527223	1996.0	2003.00	2009.0	2016.0
storage_issue_reported_l3m	22150.0	17.116659	9.174193	0.0	10.00	18.0	24.0
temp_reg_mach	22150.0	0.304199	0.460078	0.0	0.00	0.0	1.0
wh_breakdown_l3m	22150.0	3.487765	1.691661	0.0	2.00	3.0	5.0
govt_check_l3m	22150.0	18.767765	8.644778	1.0	11.00	20.0	26.0
product_wg_ton	22150.0	22086.780813	11626.192340	2065.0	12151.00	22099.0	30102.0

In [8]:

```
df.shape
```

Out[8]:

```
(22150, 24)
```

In [9]:

```
#checking missing values:

df.isnull().sum()
```

Out[9]:

```
Ware_house_ID                       0
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                       877
wh_est_year                     10545
storage_issue_reported_l3m          0
temp_reg_mach                       0
approved_wh_govt_certificate      805
wh_breakdown_l3m                    0
govt_check_l3m                      0
product_wg_ton                      0
dtype: int64
```

In [10]:

```
#Boolean Output:
df.isnull().any()
```

Out[10]:

```
Ware_house_ID                   False
WH_Manager_ID                   False
```

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

In [11]:

```
df.isna().apply(pd.value_counts).T
```

Out[11]:

	False	True
Ware_house_ID	22150.0	NaN
WH_Manager_ID	22150.0	NaN
Location_type	22150.0	NaN
WH_capacity_size	22150.0	NaN
zone	22150.0	NaN
WH_regional_zone	22150.0	NaN
num_refill_req_l3m	22150.0	NaN
transport_issue_l1y	22150.0	NaN
Competitor_in_mkt	22150.0	NaN
retail_shop_num	22150.0	NaN
wh_owner_type	22150.0	NaN
distributor_num	22150.0	NaN
flood_impacted	22150.0	NaN
flood_proof	22150.0	NaN
electric_supply	22150.0	NaN
dist_from_hub	22150.0	NaN
workers_num	21273.0	877.0
wh_est_year	11605.0	10545.0
storage_issue_reported_l3m	22150.0	NaN
temp_reg_mach	22150.0	NaN

	False	True
approved_wh_govt_certificate	21345.0	805.0
wh_breakdown_l3m	22150.0	NaN
govt_check_l3m	22150.0	NaN
product_wg_ton	22150.0	NaN

In [12]:

```
#percentage of missing values
a=df.isna().sum()
perc=(a/len(df))*100
```

In [13]:

```
perc
```

Out[13]:

```
Ware_house_ID                    0.000000
WH_Manager_ID                    0.000000
Location_type                    0.000000
WH_capacity_size                 0.000000
zone                             0.000000
WH_regional_zone                 0.000000
num_refill_req_l3m               0.000000
transport_issue_l1y              0.000000
Competitor_in_mkt                0.000000
retail_shop_num                  0.000000
wh_owner_type                    0.000000
distributor_num                  0.000000
flood_impacted                   0.000000
flood_proof                      0.000000
electric_supply                  0.000000
dist_from_hub                    0.000000
workers_num                      3.959368
wh_est_year                     47.607223
storage_issue_reported_l3m       0.000000
temp_reg_mach                    0.000000
approved_wh_govt_certificate     3.634312
wh_breakdown_l3m                 0.000000
govt_check_l3m                   0.000000
product_wg_ton                   0.000000
dtype: float64
```

In [14]:

```
import matplotlib.pyplot as plt
import seaborn as sns
```

In [15]:

```
corr=df.corr()
corr
```

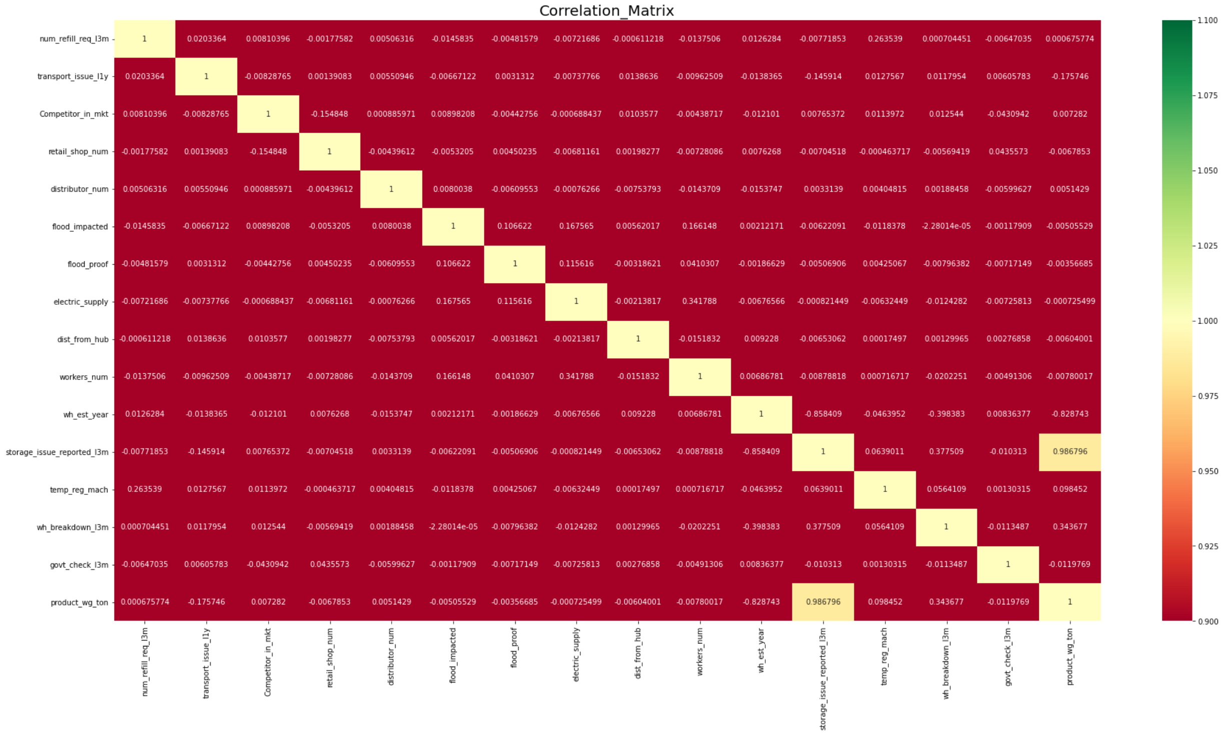
Out[15]:

	num_refill_req_l3m	transport_issue_l1y	Competitor_in_mkt	retail_shop_nu
num_refill_req_l3m	1.000000	0.020336	0.008104	-0.0017
transport_issue_l1y	0.020336	1.000000	-0.008288	0.0013
Competitor_in_mkt	0.008104	-0.008288	1.000000	-0.1548
retail_shop_num	-0.001776	0.001391	-0.154848	1.0000
distributor_num	0.005063	0.005509	0.000886	-0.0043

	num_refill_req_l3m	transport_issue_l1y	Competitor_in_mkt	retail_shop_nu
flood_impacted	-0.014583	-0.006671	0.008982	-0.0053
flood_proof	-0.004816	0.003131	-0.004428	0.0045
electric_supply	-0.007217	-0.007378	-0.000688	-0.0068
dist_from_hub	-0.000611	0.013864	0.010358	0.0019
workers_num	-0.013751	-0.009625	-0.004387	-0.0072
wh_est_year	0.012628	-0.013837	-0.012101	0.0076
storage_issue_reported_l3m	-0.007719	-0.145914	0.007654	-0.0070
temp_reg_mach	0.263539	0.012757	0.011397	-0.0004
wh_breakdown_l3m	0.000704	0.011795	0.012544	-0.0056
govt_check_l3m	-0.006470	0.006058	-0.043094	0.0435
product_wg_ton	0.000676	-0.175746	0.007282	-0.0067

In [16]:

```
#correlation Matrix plotting
plt.figure(figsize=(30,15))
plt.title('Correlation_Matrix',fontsize=20)
sns.heatmap(corr,cmap='RdYlGn',annot=True,vmax=1.0,vmin=1.0,fmt='g')
plt.show()
```



OneHotEncoding

In [41]:

```
from sklearn.preprocessing import LabelEncoder
```

In [18]:

```
le=LabelEncoder()
```

In [19]:

```
df.head()
```

Out[19]:

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

5 rows × 24 columns

In [20]:

```
df['WH_regional_zone']=le.fit_transform(df['WH_regional_zone'])
```

In [21]:

```
df['WH_regional_zone'].value_counts()
```

Out[21]:

```
5    7376
4    4045
3    3708
1    2642
2    2552
0    1827
Name: WH_regional_zone, dtype: int64
```

In [22]:

```
df.head()
```

Out[22]:

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

5 rows × 24 columns

In [23]:

```
df.skew()
```

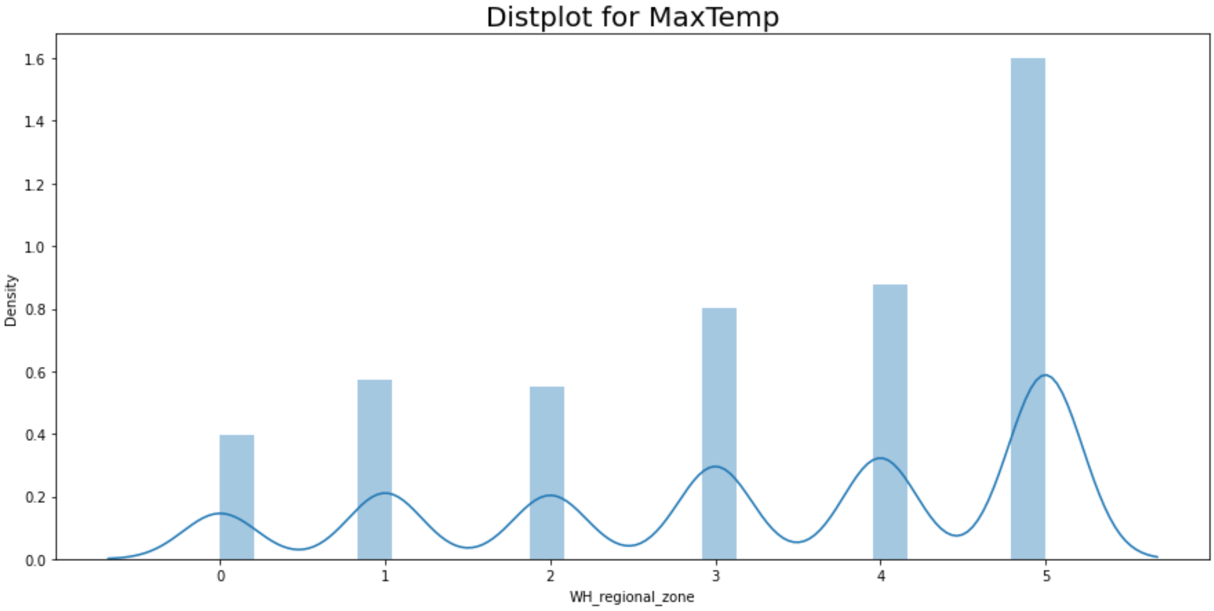
Out[23]:

```
WH_regional_zone              -0.544420
num_refill_req_l3m            -0.081390
transport_issue_l1y            1.605424
Competitor_in_mkt              0.985102
retail_shop_num                0.905324
distributor_num                0.017210
flood_impacted                 2.691308
flood_proof                    3.925685
electric_supply               -0.657167
dist_from_hub                 -0.009042
```

```
workers_num                  1.042478
wh_est_year                  0.007485
storage_issue_reported_l3m   0.117473
temp_reg_mach                0.851244
wh_breakdown_l3m            -0.072809
govt_check_l3m              -0.357737
product_wg_ton               0.336012
dtype: float64
```

In [24]:

```
plt.figure(figsize=(15,7))
plt.title('Distplot for MaxTemp',fontsize=20)
sns.distplot(df['WH_regional_zone'])
plt.show()
```



We consider Highest curve means -ve skewed -0.54Data is not normal distribution

In [31]:

```
df['WH_regional_zone'].mean()
```

Out[31]:

```
3.2474040632054177
```

In [32]:

```
df['product_wg_ton'].mean()
```

Out[32]:

```
22086.780812641082
```

Conducting Hypothesis Testing

Null Hypothesis = Average product weight shipment equal for all zones

Alternative Hypothesis = Average product weight shipment Not equal for all zones

In [35]:

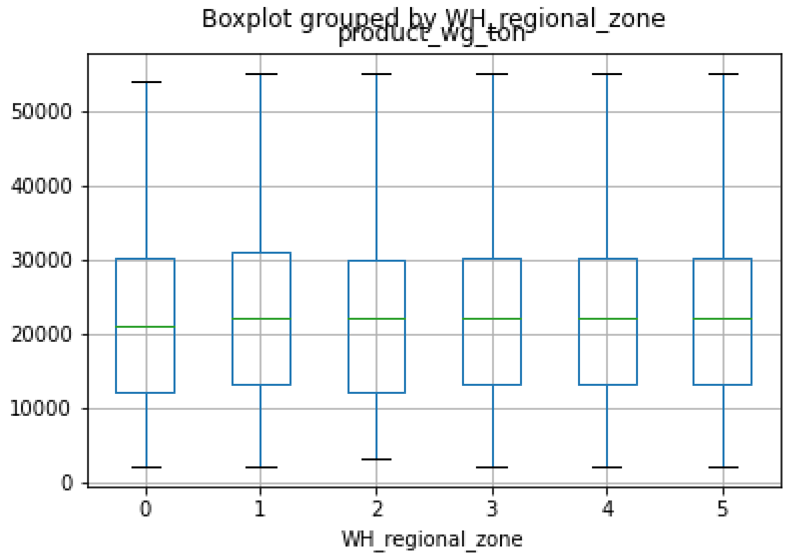
```
import statsmodels.api as sm
from statsmodels.formula.api import ols
```

In [37]:

```
df.boxplot('product_wg_ton',by='WH_regional_zone')
```

Out[37]:

```
<AxesSubplot:title={'center':'product_wg_ton'}, xlabel='WH_regional_zone'>
```



Conducting Anova Test:

In [39]:

```
new=ols('product_wg_ton ~ WH_regional_zone',data=df).fit()
Anova=sm.stats.anova_lm(new,typ=2)
```

In [40]:

```
Anova
```

Out[40]:

	sum_sq	df	F	PR(>F)
WH_regional_zone	2.213923e+04	1.0	0.000164	0.989789
Residual	2.993844e+12	22148.0	NaN	NaN

Here P>0.05 [0.98>0.05] so

Null Hypothesis Accepted and Alternative Hypothesis Rejected due to P value not less than 0.005

Average product weight of shipment is equal for all Zones