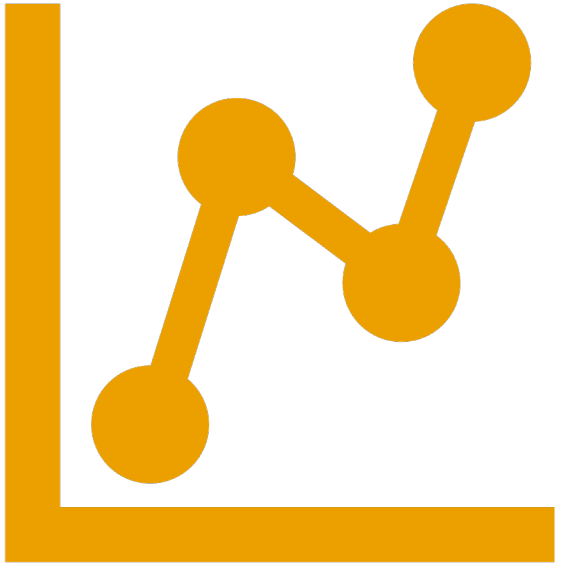
Forecasting Supply Needs: A Regression Based Approach



Problem Statement:

An FMCG company in the instant noodles business has observed a mismatch between demand and supply across its warehouses which lead to inventory cost losses. The management aims to optimize supply distribution by reducing excess supply in low-demand areas and shortages in high demand areas.

Objectives:

- Build an analytical model to determine the optimal product weight for each warehouse shipment based on historical data and relevant logistical factors.
- Provide evidence-based suggestions to help supply chain teams to maximize operational effectiveness and minimize stockouts.

Exploratory Data Analysis:

- The first version of the dataset contained **24 variables**.
- The column names were not meaningful.
- There were null values in multiple columns like **workers_num** and **wh_est_year**.

```
> str(df)
'data.frame':   25000 obs. of  24 variables:
 $ Ware_house_ID               : chr  "WH_100000" "WH_100001" "WH_100002" "WH_100003" ...
 $ WH_Manager_ID               : chr  "EID_50000" "EID_50001" "EID_50002" "EID_50003" ...
 $ Location_type               : chr  "Urban" "Rural" "Rural" "Rural" ...
 $ WH_capacity_size            : chr  "Small" "Large" "Mid" "Mid" ...
 $ zone                        : chr  "West" "North" "South" "North" ...
 $ WH_regional_zone            : chr  "Zone 6" "Zone 5" "Zone 2" "Zone 3" ...
 $ num_refill_req_l3m          : int  3 0 1 7 3 8 8 1 8 4 ...
 $ transport_issue_l1y         : int  1 0 0 4 1 0 0 0 1 3 ...
 $ Competitor_in_mkt           : int  2 4 4 2 2 2 4 4 4 3 ...
 $ retail_shop_num             : int  4651 6217 4306 6000 4740 5053 4449 7183 5381 3869 ...
 $ wh_owner_type               : chr  "Rented" "Company Owned" "Company Owned" "Rented" ...
 $ distributor_num             : int  24 47 64 50 42 37 38 45 42 35 ...
 $ flood_impacted              : int  0 0 0 0 1 0 0 0 0 0 ...
 $ flood_proof                 : int  1 0 0 0 0 0 0 0 0 0 ...
 $ electric_supply             : int  1 1 0 0 1 1 1 0 1 0 ...
 $ dist_from_hub               : int  91 210 161 103 112 152 77 241 124 78 ...
 $ workers_num                 : int  29 31 37 21 25 35 27 23 22 43 ...
 $ wh_est_year                 : int  NA NA NA NA 2009 2009 2010 NA 2013 NA ...
 $ storage_issue_reported_l3m  : int  13 4 17 17 18 23 24 18 13 6 ...
 $ temp_reg_mach               : int  0 0 0 1 0 1 0 0 1 0 ...
 $ approved_wh_govt_certificate: chr  "A" "A" "A" "A+" ...
 $ wh_breakdown_l3m            : int  5 3 6 3 6 3 3 6 5 6 ...
 $ govt_check_l3m              : int  15 17 22 27 24 3 6 24 2 2 ...
 $ product_wg_ton              : int  17115 5074 23137 22115 24071 32134 30142 24093 18082 7130 ...
```

Ware_house_ID	WH_Manager_ID	Location_type
0	0	0
WH_capacity_size	zone	WH_regional_zone
0	0	0
num_refill_req_l3m	transport_issue_l1y	Competitor_in_mkt
0	0	0
retail_shop_num	wh_owner_type	distributor_num
0	0	0
flood_impacted	flood_proof	electric_supply
0	0	0
dist_from_hub	workers_num	wh_est_year
0	990	11881
storage_issue_reported_l3m	temp_reg_mach	approved_wh_govt_certificate
0	0	0
wh_breakdown_l3m	govt_check_l3m	product_wg_ton
0	0	0

Exploratory Data Analysis: contd…

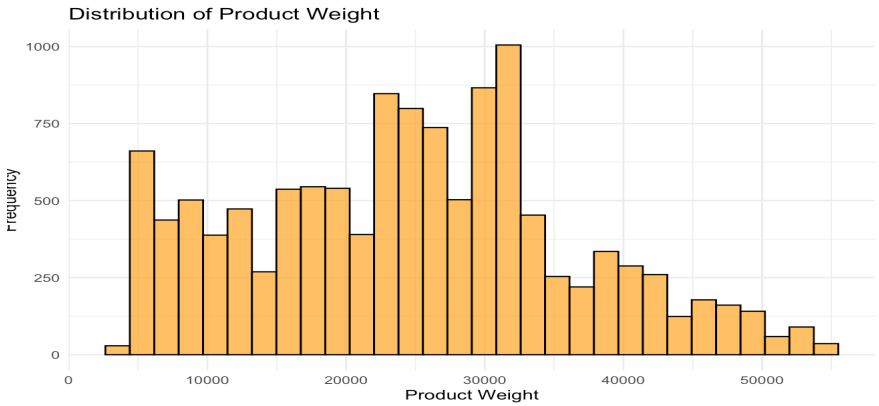
- Data cleaning was done by removing irrelevant columns and null values and changing the column names.
- The modified dataset has **22 variables**.
- There are 6 categorical variables: location, capacity, zone, reg_zone, warehouse_owner and govt_cert and remaining variables are numerical.

```
> # Removing rows with null values
> df <- na.omit(df)
> colSums(is.na(df))
            Ware_house_ID              WH_Manager_ID              Location_type         WH_capacity_size                       zone 
                        0                          0                          0                          0                          0 
       WH_regional_zone          num_refill_req_l3m          transport_issue_l1y          Competitor_in_mkt          retail_shop_num 
                        0                          0                          0                          0                          0 
          wh_owner_type              distributor_num                flood_impacted                 flood_proof           electric_supply 
                        0                          0                          0                          0                          0 
          dist_from_hub               workers_num                 wh_est_year           storage_issue_reported_l3m temp_reg_mach 
                        0                          0                          0                          0                          0 
approved_wh_govt_certificate       wh_breakdown_l3m             govt_check_l3m             product_wg_ton 
                        0                          0                          0                          0 
```

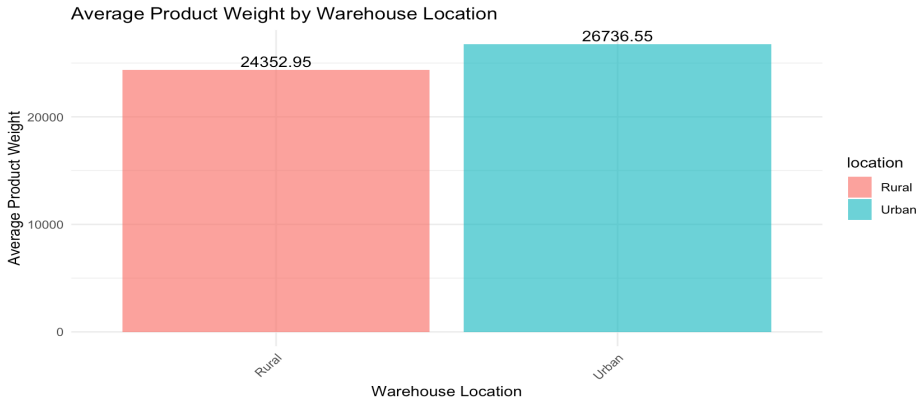
```
> cat("Number of numerical variables:", num_numerical, "\n")
Number of numerical variables: 16
> cat("Number of categorical variables:", num_categorical, "\n")
Number of categorical variables: 6
```

```
> str(df)
'data.frame':   12127 obs. of  22 variables:
 $ location              : chr  "Rural" "Rural" "Rural" "Rural" ...
 $ capacity              : chr  "Large" "Small" "Large" "Small" ...
 $ zone                  : chr  "North" "West" "West" "South" ...
 $ reg_zone              : chr  "Zone 5" "Zone 1" "Zone 6" "Zone 6" ...
 $ refill                : int  3 8 8 8 7 7 4 6 4 8 ...
 $ transport_issue       : int  1 0 0 1 1 0 0 1 1 0 ...
 $ competitor            : int  2 2 4 4 3 5 3 2 4 2 ...
 $ retail_shop           : int  4740 5053 4449 5381 4623 4627 5012 6858 4598 5678 ...
 $ warehouse_Owner       : chr  "Company Owned" "Rented" "Company Owned" "Rented" ...
 $ distributors          : int  42 37 38 42 31 40 48 26 58 31 ...
 $ flood_impacted        : int  1 0 0 0 0 0 0 0 0 0 ...
 $ flood_proof           : int  0 0 0 0 0 0 0 0 0 0 ...
 $ electric_supply       : int  1 1 1 1 1 0 0 1 1 1 ...
 $ distance_hub          : int  112 152 77 124 150 225 95 242 159 65 ...
 $ workers_num           : int  25 35 27 22 37 16 28 36 22 41 ...
 $ w_est_year            : int  2009 2009 2010 2013 1999 2017 2022 2008 2001 2016 ...
 $ storage_issues        : int  18 23 24 13 17 11 4 22 29 19 ...
 $ temperature_regulation: int  0 1 0 1 0 0 0 1 1 0 ...
 $ govt_cert             : chr  "C" "A+" "B" "A+" ...
 $ warehouse_breakdown   : int  6 3 3 5 4 2 1 5 5 4 ...
 $ govt_check            : int  24 3 6 2 6 28 1 11 27 1 ...
 $ product_weight        : int  24071 32134 30142 18082 21125 14115 5124 30063 38082 24062 ...
```

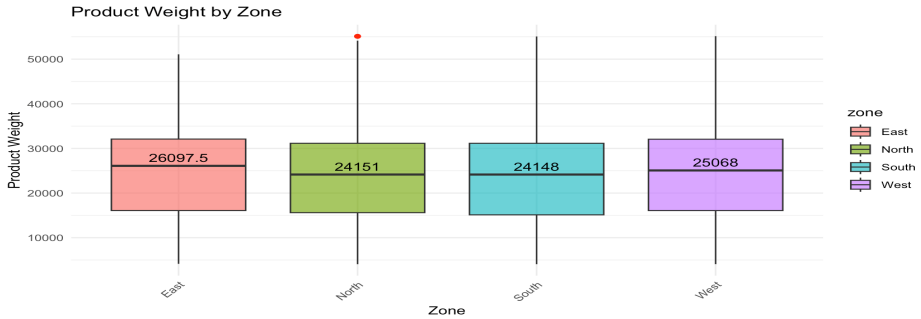
Analysis of Target variable:



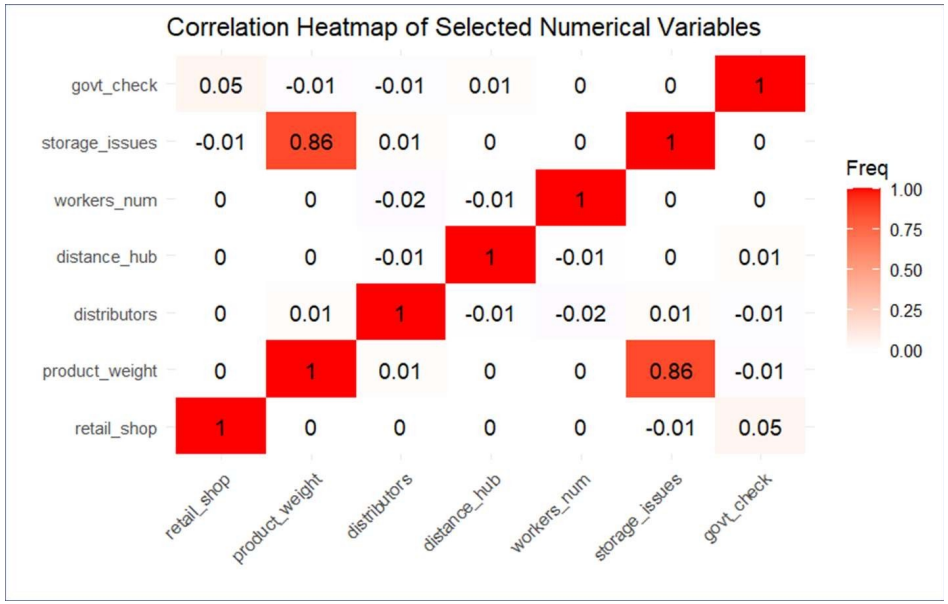
A histogram of product weight shows that most of the product weights fall between 20,000–30,000 tons.



Overall product weight is higher in urban areas than rural suggesting that demand is likely greater in urban regions.



The East zone has the highest median product weight, followed by the West zone. The North and South zones have almost identical median weights, indicating similar demand levels.



- There is a strong positive correlation (0.86) between product weight and storage issues, suggesting that warehouses with higher product weight are more likely to report storage problems.
- There is a slight positive correlation (0.05) between retail shops and government checks, indicating that locations with more retail shops are subjected to more government checks which could impact logistics and operations.

Data Pre-Processing

Standardisation

- Process of re-scaling features so they have
- Mean = 0 and Standard Deviation = 1
- Ensures all the features contribute equally to the model

Encoding

- Converted categorical data into numerical data i.e. 1,0
- Created binary columns for each category by taking a reference column

Log Transformation

Before Log Transformation

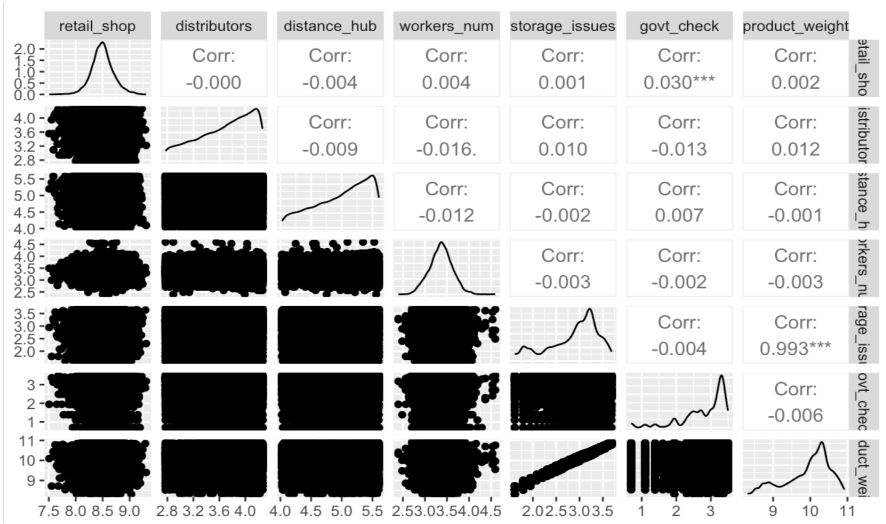
```
'data.frame':   12127 obs. of  6 variables:
 $ retail_shop   : int  4740 5053 4449 5381 4623 4627 5012 6858 4598 5678 ...
 $ distributors  : int  42 37 38 42 31 40 48 26 58 31 ...
 $ distance_hub  : int  112 152 77 124 150 225 95 242 159 65 ...
 $ workers_num   : num  25 35 27 22 37 16 28 36 22 41 ...
 $ storage_issues: int  18 17 32 15 17 11 4 22 36 11 ...
 $ govt_check    : int  24 3 6 2 6 28 1 11 27 1 ...
```

After Log Transformation

```
'data.frame':   12127 obs. of  13 variables:
 $ retail_shop                  : num  8.46 8.53 8.4 8.59 8.44 ...
 $ distributors                 : num  3.76 3.64 3.66 3.76 3.47 ...
 $ distance_hub                 : num  4.73 5.03 4.36 4.83 5.02 ...
 $ workers_num                  : num  3.26 3.58 3.33 3.14 3.64 ...
 $ storage_issues               : num  2.94 2.89 3.5 2.77 2.89 ...
 $ govt_check                   : num  3.22 1.39 1.95 1.1 1.95 ...
 $ product_weight               : num  10.09 10.38 10.31 9.8 9.96 ...
```

Before Pre-Processing

After Pre-Processing:



Model Validation Approach

Train-Test Split:

- Train Data (9701 rows)
- Test Data (2426 rows)

K-Cross Validation:

- It ensures that the model generalizes well to unseen data by repeatedly training and testing it on different subsets of the dataset.
- Split dataset into K equal-sized subsets
- Perform K iterations of training and testing

Assumptions

In the regression models, We considered some assumptions for dataset such as

- The relation is linear
- The random errors εi are normally distributed ~$N(0,\sigma 2)$
- The variance $\sigma 2$ is constant
- The expectation value is zero
- The random errors εi are independent
- No Multicollinearity between independent variables

Model Selection

Simple Linear Regression (SLR)

E(Yi) = β0+β1Xi+εi

Multiple Linear Regression w/o interactions and Polynomial Terms (MLR w/o I&Q)

E(Yi) = β0+β1Xi1+β2Xi2 +β3Xi3 +β4Xi4 + εi

Multiple Linear Regression with interactions and Polynomial Terms (MLR with I&Q)

E(Yi) = β0+β1Xi1+β2Xi2 +β3Xi3 +β4Xi4 + β5Xi1^2 + β6Xi2^2 + β7(Xi3 * Xi4) + εi

Model Selection - Simple Linear Regression

Formula:

```
#SLR Model
slr_model <- lm(product_weight ~  storage_issues , data =train_df)
summary(slr_model)
anova(slr_model)
```

Summary:

```
Call:
lm(formula = product_weight ~ storage_issues, data = train_df)

Residuals:
     Min       1Q   Median       3Q      Max
-1.06926 -0.15811 -0.01239  0.13320  1.13113

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    9.966757   0.002559    3895   <2e-16 ***
storage_issues 0.522261   0.002560     204   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.252 on 9699 degrees of freedom
Multiple R-squared:  0.811,     Adjusted R-squared:  0.811
F-statistic: 4.163e+04 on 1 and 9699 DF,  p-value: < 2.2e-16
```

Anova:

```
Analysis of Variance Table

Response: product_weight
                  Df  Sum Sq Mean Sq F value    Pr(>F)
storage_issues     1 2643.62 2643.62   41625 < 2.2e-16 ***
Residuals       9699  615.98    0.06
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- SLR Model explains 81.1% of the variance in product weight
- Residual standard error (0.252) is very low which represents average deviation of residuals
- Model fits the data well with a extra sum of squares of 2643.62 units of variance contributed by storage issues feature

Simple Linear Regression - Hypothesis Testing

t-test for slope ($\beta 1$) of SLR:

Hypothesis:

- Null Hypothesis(Ho): $\beta 1 = 0$
- No linear relationship between independent and dependent variable

Alternate Hypothesis (Ha): $\beta 1 \neq 0$. linear relationship between independent and dependant variable

T-statistic (t*) : $\mathbf{b1} / \mathbf{s}(\mathbf{b1})$ Significance Level (alpha): 5% Decision Rule:

If |t* >tcritical or p-value < α: Reject Null Hypothesis (Ho)

If |t* <= tcritical or p-value >= α: Fail to reject Null Hypothesis (Ho)

```
Call:
lm(formula = product_weight ~ storage_issues, data = train_df)

Residuals:
     Min       1Q   Median       3Q      Max
-1.06926 -0.15811 -0.01239  0.13320  1.13113

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    9.966757   0.002559    3895   <2e-16 ***
storage_issues 0.522261   0.002560     204   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.252 on 9699 degrees of freedom
Multiple R-squared:  0.811,     Adjusted R-squared:  0.811
F-statistic: 4.163e+04 on 1 and 9699 DF,  p-value: < 2.2e-16
```

Results:

Since p-values < significance level, we reject the null hypothesis and conclude there's a linear association between Product weight and storage issues features

Simple Linear Regression - Hypothesis Testing

Anova test for SLR:

Hypothesis:

- Null Hypothesis(Ho): $\beta 1 = 0$
- Storage issue feature has no effect on product weight feature
- Alternate Hypothesis (Ha): $\beta 1 \neq 0$

Storage issue feature does have effect on product weight feature

F-statistic (F*) : MSR/MSE Significance

Level (alpha): 5% Decision Rule:

- If F*> F critical or p-value < α: Reject Null Hypothesis (Ho)
- If F*<=F critical or p-value >= α: Fail to reject Null Hypothesis (Ho)

Results:

Since p-values < significance level, we reject the null hypothesis and conclude storage issues significantly affects product weight

```
Call:
lm(formula = product_weight ~ storage_issues, data = train_df)

Residuals:
     Min       1Q   Median       3Q      Max
-1.06926 -0.15811 -0.01239  0.13320  1.13113

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    9.966757   0.002559    3895   <2e-16 ***
storage_issues 0.522261   0.002560     204   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.252 on 9699 degrees of freedom
Multiple R-squared:  0.811,     Adjusted R-squared:  0.811
F-statistic: 4.163e+04 on 1 and 9699 DF,  p-value: < 2.2e-16
```

```
Analysis of Variance Table

Response: product_weight
                Df  Sum Sq Mean Sq F value    Pr(>F)
storage_issues   1 2643.62 2643.62   41625 < 2.2e-16 ***
Residuals     9699  615.98    0.06
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Simple Linear Regression - Hypothesis Testing

Correlation Coefficient (ρ) Test:

Hypothesis:

- Null Hypothesis(Ho): $\rho = 0$
- Storage issue and product weight feature are independant
- Alternate Hypothesis (Ha): $\rho \neq 0$

Storage issue and product weight feature are dependant

Estimation of Pearson product moment correlation coefficient (ρ):

```
r_xy : 0.9005695
t_statistic: 204.023
t_critical: 1.9602
```

Confidence Interval for rho(ρ)

```
Fisher Z transformation (Z') 1.475225

The 95% confidence interval for the true correlation coefficient is:
 0.6216 <= rho <= 0.6337
```

t-statistic (t*) : sqrt(n-2) (r_xy / sqrt(1-(r_xy) ^2)) Significance Level (alpha) : 5%

Decision Rule:

- If t* > t critical or **p-value < α**: Reject Null Hypothesis (Ho)
- If t* <= t critical or **p-value >= α**: Fail to reject Null Hypothesis (Ho)

Results:

Since t* > t critical , we reject the null hypothesis and conclude there is a significant relationship between storage issues and product weight

MLR w/o Quadratic & Interaction terms

Anova:

Formula with all predictors:

```
#First Order MLR Model
fo_model <- lm(product_weight ~ location + capacity + zone + reg_zone + refill + transport_issue + competitor +
                                retail_shop + warehouse_Owner + distributors +
                                flood_impacted + flood_proof + electric_supply + distance_hub +
                                workers_num + w_est_year + storage_issues +
                                temperature_regulation + govt_cert +
                                warehouse_breakdown + govt_check , data = train_df)
```

```
Analysis of Variance Table

Response: product_weight
                        Df  Sum Sq Mean Sq    F value    Pr(>F)
storage_issues           1 2643.62 2643.62 70300.4734 < 2.2e-16 ***
w_est_year              26  211.37    8.13   216.1888 < 2.2e-16 ***
govt_cert                4   31.00    7.75   206.0952 < 2.2e-16 ***
transport_issue          4    9.24    2.31    61.4514 < 2.2e-16 ***
temperature_regulation   1    0.84    0.84    22.2567 2.419e-06 ***
govt_check               1    0.11    0.11     3.0299   0.08177 .
location                 1    0.08    0.08     2.1171   0.14570
Residuals             9662  363.33    0.04
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Stepwise Model Selection (Both Direction):

Combines forward selection and backward elimination search procedures to simply it to reduce overfitting

```
reg.null <- lm(product_weight ~ 1,data=train_df)
best_fo_model <- step(reg.null,direction="both",scope=list(upper= fo_model,lower=reg.null))
```

Summary:

```
Residual standard error: 0.1939 on 9662 degrees of freedom
Multiple R-squared:  0.8885,    Adjusted R-squared:  0.8881
F-statistic:  2027 on 38 and 9662 DF,  p-value: < 2.2e-16
```

- Above Model explains 88.5% of the variance in product weight
- Adjusted R-square is approx. equal to Multiple R-square i.e. no irrelevant predictors in the model
- Residual standard error (0.1939) is very low and high degree of freedom (n-p) indicates lesser number of predictors fitted for the model

Extra Sum of Squares: Test for Regression Some Coefficients βk:

Hypothesis

- Null Hypothesis(Ho): βgovt_check = βlocation = 0

Govt check and Location do not significantly improve the model

- Alternate Hypothesis (Ha): βgovt_check ≠ βlocation ≠ 0 Govt check and Location significantly improve the model

F-statistic (F*) : $MSR(Xq,\ldots,Xp-1|X1,X2,\ldots,Xq-1)$/ MSE

Significance Level (alpha) : 5% Decision Rule:

- **If |F*| > Fcritical** or **p-value < α**: Reject Null Hypothesis (Ho)
- **If |F*| <= Fcritical** or **p-value >= α**: Fail to reject Null Hypothesis (Ho)

```
> anova(best_fo_model, best_fo_model_1)
Analysis of Variance Table

Model 1: product_weight ~ storage_issues + w_est_year + govt_cert + transport_issue +
    temperature_regulation + govt_check + location
Model 2: product_weight ~ storage_issues + w_est_year + govt_cert + transport_issue +
    temperature_regulation
  Res.Df    RSS Df Sum of Sq      F  Pr(>F)
1   9662 363.33
2   9664 363.53 -2  -0.19355 2.5735 0.07632 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Results:

Since p-value >= α, we **failed to reject the null hypothesis** and conclude govt check and location feature do not significantly improve the model

Best Subset MLR w/o Quadratic & Interaction terms

Formula:

```
best_fo_model_1 <- lm(product_weight ~ storage_issues + w_est_year + govt_cert + transport_issue +
  temperature_regulation , data = train_df)
```

Summary:

```
Residual standard error: 0.194 on 9664 degrees of freedom
Multiple R-squared:  0.8885,    Adjusted R-squared:  0.8881
```

Anova:

```
Analysis of Variance Table

Response: product_weight
                       Df  Sum Sq Mean Sq   F value    Pr(>F)
storage_issues          1 2643.62 2643.62 70277.588 < 2.2e-16 ***
w_est_year             26  211.37    8.13   216.119 < 2.2e-16 ***
govt_cert               4   31.00    7.75   206.028 < 2.2e-16 ***
transport_issue         4    9.24    2.31    61.431 < 2.2e-16 ***
temperature_regulation  1    0.84    0.84    22.250 2.428e-06 ***
Residuals            9664  363.53    0.04
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
                       GVIF Df GVIF^(1/(2*Df))
storage_issues         3.102658  1        1.761436
w_est_year             3.270324 26        1.023048
govt_cert              1.555713  4        1.056796
transport_issue        1.026658  4        1.003294
temperature_regulation 1.456053  1        1.206670
```

- All predictors significantly contribute to explaining the variation
- All predictors have low Variance Inflation Factor (VIF < 2), indicating no significant multicollinearity concerns

MLR with Quadratic & Interaction terms

Full Model:

```
full_model <- lm(product_weight ~ (location + capacity + zone + reg_zone + refill + transport_issue + competitor +
                                   retail_shop + warehouse_Owner + distributors +
                                   flood_impacted + flood_proof + electric_supply + distance_hub +
                                   workers_num + w_est_year + storage_issues +
                                   temperature_regulation + govt_cert +
                                   warehouse_breakdown + govt_check + retail_shop_retail_shop + distributors_distributors +
                                   distance_hub_distance_hub + workers_num_workers_num + storage_issues_storage_issues + govt_check_govt_check )^2 , data = train_df)
```

Stepwise Model Selection (Both Direction):

```
reg.null <- lm(product_weight ~ 1,data=train_df)
both_model <- step(reg.null,direction="both",scope=list(upper= full_model,lower=reg.null))
```

Annova:

```
Response: product_weight
                                                    Df  Sum Sq Mean Sq    F value    Pr(>F)
storage_issues_storage_issues                        1 2650.91 2650.91 79574.1698 < 2.2e-16 ***
w_est_year                                          26  206.18    7.93   238.0435 < 2.2e-16 ***
govt_cert                                            4   31.09    7.77   233.2781 < 2.2e-16 ***
storage_issues                                       1    2.39    2.39    71.6751 < 2.2e-16 ***
transport_issue                                      4    9.11    2.28    68.3745 < 2.2e-16 ***
temperature_regulation                               1    0.83    0.83    24.7662 6.585e-07 ***
govt_check_govt_check                                1    0.10    0.10     3.0268  0.081934 .
capacity                                             2    0.14    0.07     2.1353  0.118271
location                                             1    0.08    0.08     2.5095  0.113194
electric_supply                                      1    0.05    0.05     1.4428  0.229720
storage_issues_storage_issues:w_est_year            26   33.68    1.30    38.8843 < 2.2e-16 ***
w_est_year:storage_issues                           26    4.04    0.16     4.6666 3.704e-14 ***
govt_cert:storage_issues                             4    0.49    0.12     3.6475  0.005655 **
storage_issues_storage_issues:storage_issues         1    0.18    0.18     5.5135  0.018890 *
temperature_regulation:govt_check_govt_check         1    0.10    0.10     3.0359  0.081474 .
storage_issues:temperature_regulation                1    0.10    0.10     2.9897  0.083828 .
storage_issues_storage_issues:temperature_regulation 1    0.10    0.10     2.9501  0.085903 .
storage_issues_storage_issues:capacity               2    0.18    0.09     2.7013  0.067170 .
govt_check_govt_check:electric_supply                1    0.09    0.09     2.8292  0.092598 .
capacity:electric_supply                             2    0.18    0.09     2.6592  0.070056 .
Residuals                                         9593  319.58    0.03
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

F-test for Interaction Terms

Interaction model:

```
interaction_model <- lm(product_weight ~ storage_issues_storage_issues + w_est_year +
                        govt_cert + storage_issues + transport_issue + temperature_regulation +
                        govt_check_govt_check + capacity + location + electric_supply +
                        storage_issues_storage_issues:w_est_year + w_est_year:storage_issues +
                        govt_cert:storage_issues + storage_issues_storage_issues:storage_issues +
                        temperature_regulation:govt_check_govt_check + storage_issues:temperature_regulation +
                        storage_issues_storage_issues:temperature_regulation + storage_issues_storage_issues:capacity +
                        govt_check_govt_check:electric_supply + capacity:electric_supply, train_df)
```

Additive Model:

```
additive_model <- lm(product_weight ~ storage_issues_storage_issues + w_est_year +
  govt_cert + storage_issues + transport_issue + temperature_regulation +
  govt_check_govt_check + capacity + location + electric_supply, data =train_df)
```

Hypothesis

- Null Hypothesis(Ho): interaction terms coefficient = 0

Interaction terms does not add variation i.e. not improve the model

- Alternate Hypothesis (Ha): interaction terms coefficient = 0≠ 0 Interaction terms does add variation i.e. improve the model

Significance Level (alpha) : 5% Decision.

Rule:

- **If F* > Fcritical** or **p-value < α**: Reject Null Hypothesis (Ho)
- **If F* <= Fcritical** or **p-value >= α**: Fail to reject Null Hypothesis (Ho)

```
> anova(additive_model, both_model)
Analysis of Variance Table

Model 1: product_weight ~ storage_issues_storage_issues + w_est_year +
    govt_cert + storage_issues + transport_issue + temperature_regulation +
    govt_check_govt_check + capacity + location + electric_supply
Model 2: product_weight ~ storage_issues_storage_issues + w_est_year +
    govt_cert + storage_issues + transport_issue + temperature_regulation +
    govt_check_govt_check + capacity + location + electric_supply +
    storage_issues_storage_issues:w_est_year + w_est_year:storage_issues +
    govt_cert:storage_issues + storage_issues_storage_issues:storage_issues +
    temperature_regulation:govt_check_govt_check + storage_issues:temperature_regulation +
    storage_issues_storage_issues:temperature_regulation + storage_issues_storage_issues:capacity +
    govt_check_govt_check:electric_supply + capacity:electric_supply
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1   9658 358.72
2   9593 319.58 65    39.142 18.076 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Results:

Since p-value < α, we **reject the null hypothesis** and conclude Interaction terms does add variation i.e. significantly improve the

Test for Regression individual Coefficients βk:

We'll take Electric Supply feature and Test if it significant or not

Hypothesis:

- Null Hypothesis(Ho): βelectric_supply= 0

Electric supply do not significantly improve the model

- Alternate Hypothesis (Ha): βelectric_supply ≠ 0 Electric supply significantly improve the model

F-statistic (F*) : $MSR(Xq|X1,X2,\ldots,Xq-1)$/ MSE

Significance Level (alpha) : 5% Decision Rule:

- **If |F*| > Fcritical** or **p-value < α**: Reject Null Hypothesis (Ho)
- **If |F*| <= Fcritical** or **p-value >= α**: Fail to reject Null Hypothesis (Ho)

```
Analysis of Variance Table

Model 1: product_weight ~ storage_issues_storage_issues + w_est_year +
    govt_cert + storage_issues + transport_issue + temperature_regulation +
    govt_check_govt_check + capacity + location + electric_supply +
    storage_issues_storage_issues:w_est_year + w_est_year:storage_issues +
    govt_cert:storage_issues + storage_issues_storage_issues:storage_issues +
    temperature_regulation:govt_check_govt_check + storage_issues:temperature_regulation +
    storage_issues_storage_issues:temperature_regulation + storage_issues_storage_issues:capacity +
    govt_check_govt_check:electric_supply + capacity:electric_supply
Model 2: product_weight ~ storage_issues_storage_issues + w_est_year +
    govt_cert + storage_issues + transport_issue + temperature_regulation +
    govt_check_govt_check + capacity + location + storage_issues_storage_issues:w_est_year +
    w_est_year:storage_issues + govt_cert:storage_issues + storage_issues_storage_issues:storage_issues +
    temperature_regulation:govt_check_govt_check + storage_issues:temperature_regulation +
    storage_issues_storage_issues:temperature_regulation + storage_issues_storage_issues:capacity +
    govt_check_govt_check:electric_supply + capacity:electric_supply
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1   9593 319.58
2   9594 319.63 -1 -0.052248 1.5684 0.2105
```

Results:

Since p-value >= α, we **failed to reject the null hypothesis** and conclude electric supply feature do not significantly improve the model

Best Subset MLR with Quadratic & Interaction terms

Formula:

```
updated_both_model <- lm(product_weight ~ storage_issues_storage_issues + w_est_year +
                         govt_cert + storage_issues + transport_issue + temperature_regulation +
                         storage_issues_storage_issues:w_est_year + w_est_year:storage_issues +
                         govt_cert:storage_issues + storage_issues_storage_issues:storage_issues , data =train_df)
```

Summary:

```
Residual standard error: 0.1827 on 9606 degrees of freedom
Multiple R-squared:  0.9016,    Adjusted R-squared:  0.9006
F-statistic: 936.2 on 94 and 9606 DF,  p-value: < 2.2e-16
```

Anova:

```
> anova(updated_both_model)
Analysis of Variance Table

Response: product_weight
                                          Df  Sum Sq Mean Sq    F value    Pr(>F)
storage_issues_storage_issues              1 2650.91 2650.91 79380.9178 < 2.2e-16 ***
w_est_year                                26  206.18    7.93   237.4654 < 2.2e-16 ***
govt_cert                                  4   31.09    7.77   232.7116 < 2.2e-16 ***
storage_issues                             1    2.39    2.39    71.5010 < 2.2e-16 ***
transport_issue                            4    9.11    2.28    68.2085 < 2.2e-16 ***
temperature_regulation                     1    0.83    0.83    24.7061 6.793e-07 ***
storage_issues_storage_issues:w_est_year  26   33.65    1.29    38.7517 < 2.2e-16 ***
w_est_year:storage_issues                 26    4.00    0.15     4.6065 6.890e-14 ***
govt_cert:storage_issues                   4    0.48    0.12     3.5622  0.006567 **
storage_issues_storage_issues:storage_issues  1  0.18    0.18     5.4624  0.019450 *
Residuals                               9606  320.79    0.03
```

Multicollinearity (VIF Values):

	GVIF	Df	GVIF^(1/(2*Df))
storage_issues_storage_issues	5.317517e+06	1	2305.974292
w_est_year	7.211596e+27	26	3.433455
govt_cert	1.725969e+00	4	1.070605
storage_issues	5.030673e+06	1	2242.916126
transport_issue	1.054377e+00	4	1.006641
temperature_regulation	1.480145e+00	1	1.216612
storage_issues_storage_issues:w_est_year	1.188174e+111	26	136.790326
w_est_year:storage_issues	1.216337e+111	26	136.851965
govt_cert:storage_issues	7.471416e+00	4	1.285806
storage_issues_storage_issues:storage_issues	4.319322e+02	1	20.782979

- All predictors significantly contribute to explaining the variation
- Some predictors have very high Variance Inflation Factor (VIF > 2), indicating significant multicollinearity concerns

Multicollinearity

Iteration 0:

	GVIF	Df	GVIF^(1/(2*Df))
storage_issues_storage_issues	5.317517e+06	1	2305.974292
w_est_year	7.211596e+27	26	3.433455
govt_cert	1.725969e+00	4	1.070605
storage_issues	5.030673e+06	1	2242.916126
transport_issue	1.054377e+00	4	1.006641
temperature_regulation	1.480145e+00	1	1.216612
storage_issues_storage_issues:w_est_year	1.188174e+111	26	136.790326
w_est_year:storage_issues	1.216337e+111	26	136.851965
govt_cert:storage_issues	7.471416e+00	4	1.285806
storage_issues_storage_issues:storage_issues	4.319322e+02	1	20.782979

Iteration 1:

Residual standard error: 0.1827 on 9606 degrees of freedom
Multiple R-squared: 0.9016, Adjusted R-squared: 0.9006
F-statistic: 936.2 on 94 and 9606 DF, p-value: < 2.2e-16

consider setting type = 'predictor'; see ?vif

	GVIF	Df	GVIF^(1/(2*Df))
w_est_year	7.211596e+27	26	3.433455
govt_cert	1.725969e+00	4	1.070605
storage_issues	5.030673e+06	1	2242.916124
transport_issue	1.054377e+00	4	1.006641
temperature_regulation	1.480145e+00	1	1.216612
w_est_year:storage_issues_storage_issues	2.382228e+114	27	131.247876
w_est_year:storage_issues	1.216337e+111	26	136.851965
govt_cert:storage_issues	7.471416e+00	4	1.285806
storage_issues:storage_issues_storage_issues	4.319322e+02	1	20.782979

Best Subset MLR with Quadratic & Interaction terms after removing high VIF Values

Formula:

```
updated_both_model_mc2 <- lm(product_weight ~ w_est_year + govt_cert + storage_issues + transport_issue +
                             temperature_regulation, data =train_df)
```

Summary:

```
Residual standard error: 0.194 on 9664 degrees of freedom
Multiple R-squared:  0.8885,    Adjusted R-squared:  0.8881
F-statistic:  2139 on 36 and 9664 DF,  p-value: < 2.2e-16
```

Anova:

```
> anova(updated_both_model_mc2)
Analysis of Variance Table

Response: product_weight
                        Df  Sum Sq Mean Sq  F value    Pr(>F)
w_est_year              26 2546.34  97.936 2603.524 < 2.2e-16 ***
govt_cert                4   34.39   8.598  228.571 < 2.2e-16 ***
storage_issues           1  305.26 305.255 8114.867 < 2.2e-16 ***
transport_issue          4    9.24   2.311   61.431 < 2.2e-16 ***
temperature_regulation   1    0.84   0.837   22.250 2.428e-06 ***
Residuals             9664  363.53   0.038
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

VIF:

```
> vif(updated_both_model_mc2)
                             GVIF Df GVIF^(1/(2*Df))
w_est_year               3.270324 26        1.023048
govt_cert                1.555713  4        1.056796
storage_issues           3.102658  1        1.761436
transport_issue          1.026658  4        1.003294
temperature_regulation   1.456053  1        1.206670
```

Model Comparison

Simple Linear Regression (SLR)

- BIC : 814.3759
- Adjusted R-square : 81.1%

MLR w/o I&Q

- BIC : -3980.228
- Adjusted R-square : 88.81%

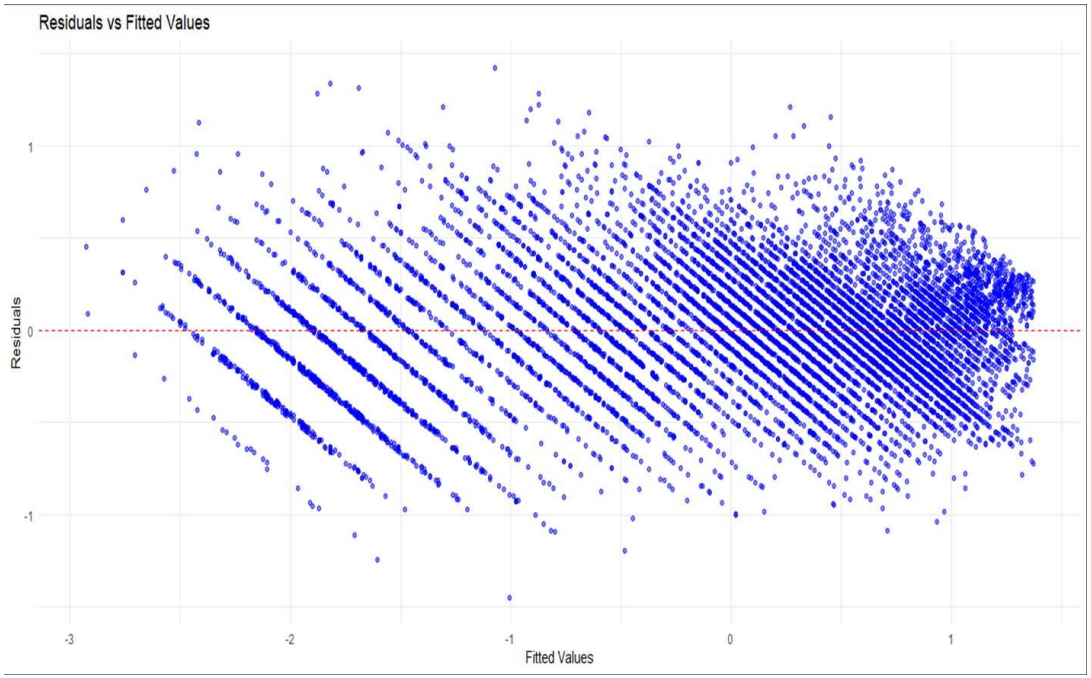
MLR with I&Q

- BIC : -3980.228
- Adjusted R-square : 89.76%

Considering above values of each subset models, the best subset model is

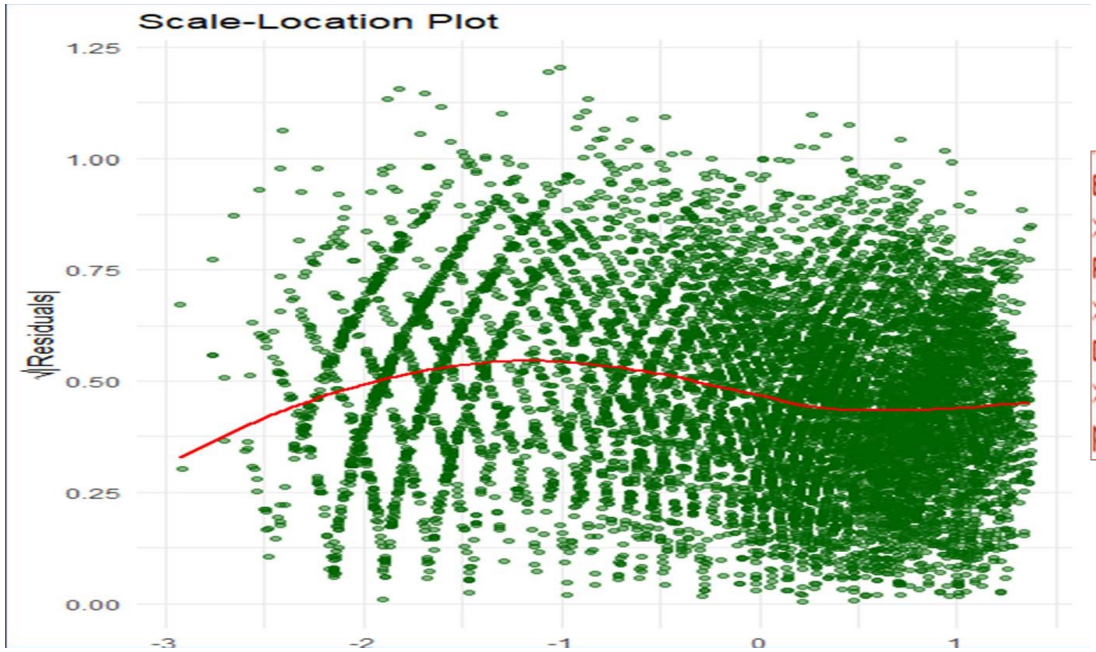
MLR with I&Q

Fitted vs residuals and test for lack of fitness



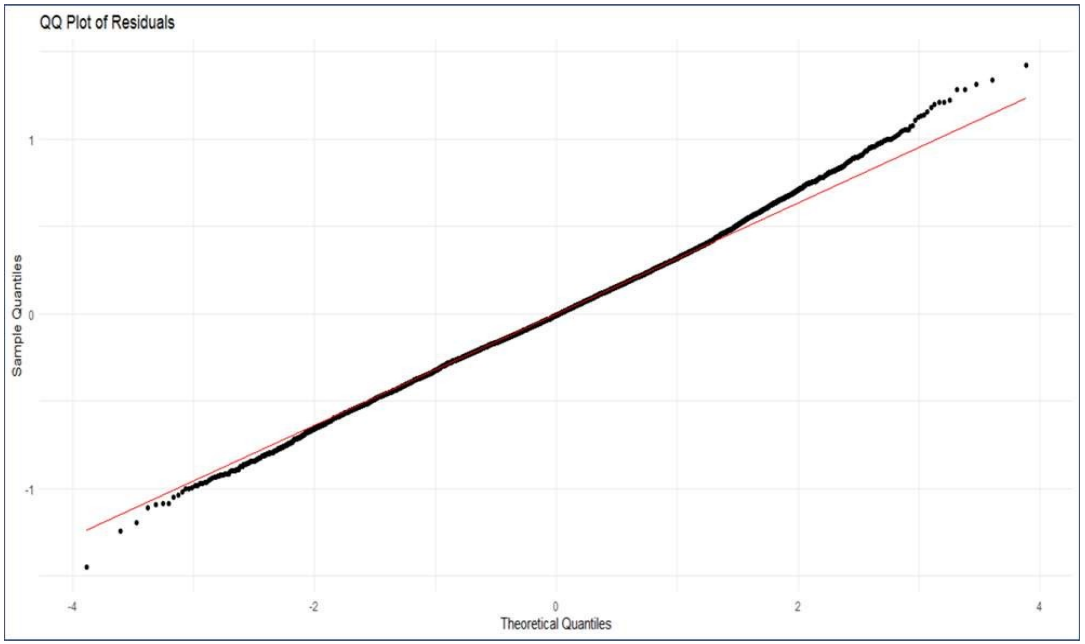

```
> cat("F-statistic for Lack-of-Fit:", F_stat, "\n")
F-statistic for Lack-of-Fit: 0.3460874
> cat("p-value for Lack-of-Fit:", p_value, "\n")
p-value for Lack-of-Fit: 1
> cat("Degrees of Freedom (LOF):", df_LOF, "\n")
Degrees of Freedom (LOF): 5574
> cat("Degrees of Freedom (Pure Error):", df_PE, "\n")
Degrees of Freedom (Pure Error): 4090
```

Scale-location and Brown test



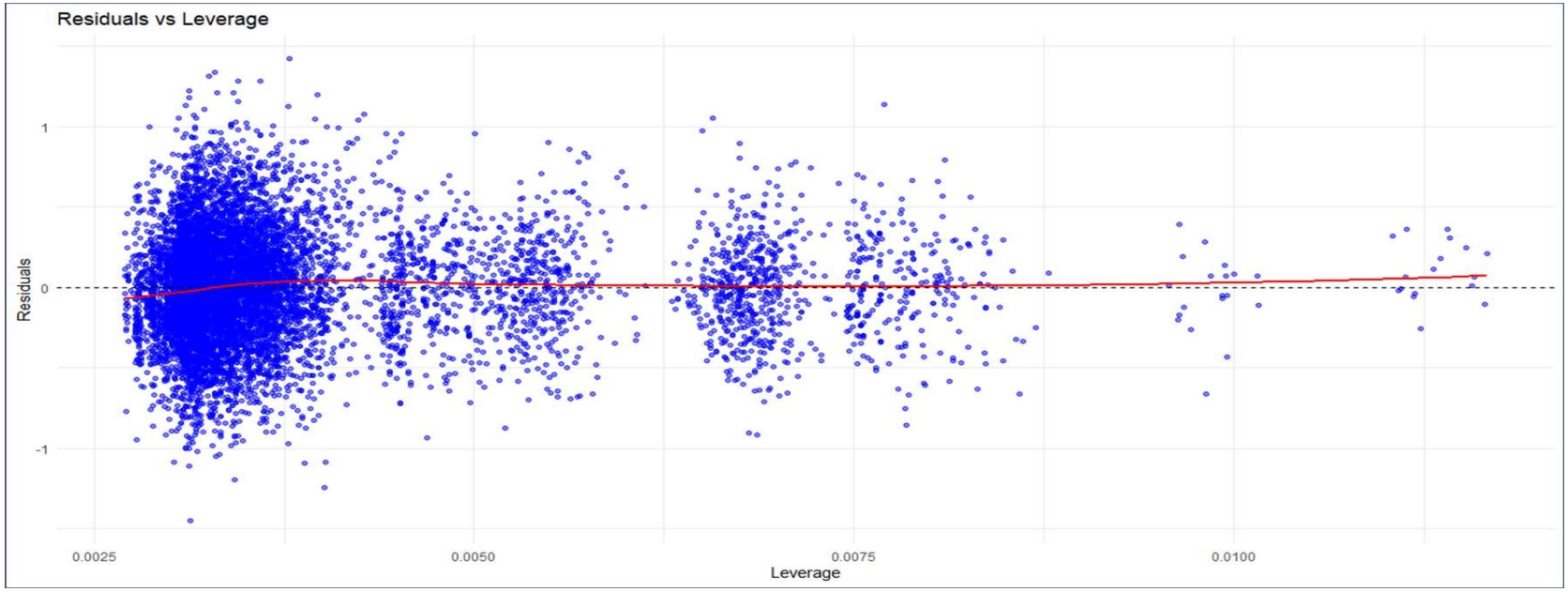

```
Brown-Forsythe Test Results:
> cat("F-statistic:", bf_test$statistic, "\n")
F-statistic: 44.79737
> cat("Degrees of Freedom:", bf_test$parameter, "
Degrees of Freedom: 3 5347.542
> cat("p-value:", bf_test$p.value, "\n")
p-value: 1.386091e-28
```

QQ plot




```
Shapiro-Wilk Test Results:
> cat("W-statistic:", shapiro_test$statistic, "\n")
W-statistic: 0.9972575
> cat("p-value:", shapiro_test$p.value, "\n")
p-value: 7.104407e-08
```

Residuals vs leverage



Outlier Analysis

Outlying on X-direction

Leverage

```
> print(leverage_values[1:100])
        2463         2511        10419         8718         2986         1842         9334         3371        11638         4761
0.004952616 0.005246644 0.003241986 0.003068780 0.003714976 0.003566736 0.002930429 0.003741430 0.003220417 0.003193404
        6746         9819         2757         5107         9145         9209        10205         2888         6170         2567
0.003928040 0.003075145 0.002871127 0.003207894 0.003233212 0.003616122 0.003092926 0.003260871 0.003291136 0.003931702
        9642         9982         2980         1614          555         4469         9359        10784        10730         7789
0.003678132 0.003138685 0.003111012 0.003097379 0.003077370 0.005050736 0.006867594 0.003379870 0.003095018 0.002965777
        9991         9097         1047         7067         3004         3207         7989         3995         8358          217
0.003320415 0.003261736 0.003642176 0.003221177 0.004013600 0.005358601 0.003067703 0.003117942 0.003187620 0.003019622
        9506         8157        10821         6216         8780         1599         4237         3937         4089         2907
0.002869192 0.003379476 0.002986763 0.003287833 0.003338529 0.003121573 0.003638213 0.003223227 0.004717726 0.004492632
         294         8469           41         8508         7391         6672         7284        11014        10987         2504
0.003352756 0.003240535 0.003131586 0.004569953 0.003153921 0.003592371 0.003452146 0.002723611 0.006877961 0.006980755
        6742         9375         8944        11473         8566        10034         6129        10274         4612         2117
0.003335888 0.003204894 0.006520346 0.003141812 0.003108644 0.005336902 0.003167770 0.002906897 0.003203759 0.003180871
        6134          755         6553         5428         9198        10777         7127        10531         9640         3358
0.004587716 0.005387556 0.004028119 0.003180440 0.003478784 0.003505376 0.004031912 0.003481743 0.003279145 0.003342730
        3980         9326         3230         5603        10126         9693         4576         3783         7831        10106
0.003018757 0.003525098 0.003402344 0.005573054 0.003542722 0.003056364 0.003455089 0.003087931 0.003217648 0.003164016
        5967         9301         7816         9267        11338         1386        10476         4706         2378         4044
0.003792884 0.002966514 0.003515674 0.003544707 0.003339343 0.003176841 0.003119560 0.003288832 0.002980411 0.003343788
>
```

Threshold:

```
> print(threshold)
[1] 0.00762808
```

Outlying Points:

```
> print(high_leverage[1:100])
  712  5509  2432  3082  9587 10054   193 10307  7274 11211  8006  9174  4494 10719   703  1722  3101  5868  1406  2558
  165   240   353   406   436   448   473   569   638   644   705   711   759   802   919   948  1021  1025  1179  1200
12117  1762  5988  8412  9713  8284  1427 10247  4307  4023  5466 10482  7956  8903  3909  4455  9947  9233  4621  1158
 1328  1471  1527  1529  1554  1591  1611  1617  1642  1741  1753  1759  1849  1978  2024  2077  2201  2297  2327  2352
 5455 11332  7965  8137  2416  5019  4931  6076 12052  8096  3058 11408 11201  5012  8495  3040  3486 10072  9036   326
 2403  2417  2449  2479  2493  2555  2664  2709  2740  2792  2795  2863  2874  2881  3087  3103  3112  3162  3222  3278
 5131 11815  1000  9284  9064  1334   559  8668  6227   345   229  3819 11279 11669  9738  6073  1009  7553 11776 11931
 3330  3331  3352  3381  3416  3423  3433  3558  3627  3660  3685  3724  3744  3756  3776  3837  3839  3860  3884  3918
 4796  9538  9992  4208  6101   708 10779  9290  2282  3131  1669   918 11275  4123  7793  3266  2301 11690  1914  6060
 3919  3948  4101  4186  4207  4225  4226  4267  4274  4376  4531  4568  4578  4587  4598  4738  4754  4803  4891  4968
```

Outlying on Y-direction

Studentize Residuals

```
> print(studentized_residuals[1:100])
        2463         2511        10419         8718         2986         1842         9334         3371        11638         4761         6746
-1.676087271 -1.524235671  0.946116247 -0.167677802 -1.269229641  0.250716303 -0.879865955 -0.516316434 -0.373706093  0.673350679  0.160879634
        9819         2757         5107         9145         9209        10205         2888         6170         2567         9642         9982
 0.167043679  0.447527014  0.404086477 -1.223876162  0.903348797 -0.869750121  0.475831940 -0.138332674 -0.462980217 -0.082984903 -0.628690313
        2980         1614          555         4469         9359        10784        10730         7789         9991         9097         1047
-0.303018330 -2.121539368 -0.002638947 -0.516816292  0.345955026 -0.472975176  0.843885491 -0.295190283  0.861218200  0.072765308  0.416667438
        7067         3004         3207         7989         3995         8358          217         9506         8157        10821         6216
-0.832682498 -0.644734685 -0.031088464  1.527663368  0.436429567  0.547139637  0.252869886 -1.182527577 -0.172013699 -0.208829747  0.377006677
        8780         1599         4237         3937         4089         2907          294         8469           41         8508         7391
-0.882252247  1.412392051  0.989440977 -0.235034926 -0.467284332  0.898863696 -1.522145195  1.238847604  1.048478812  0.709782483 -0.980076152
        6672         7284        11014        10987         2504         6742         9375         8944        11473         8566        10034
 0.822471024 -0.539328881  0.442404758 -0.017811035  0.292443960  1.687049036 -1.271984905 -1.188551649 -1.707052140 -1.477375064 -0.556079550
        6129        10274         4612         2117         6134          755         6553         5428         9198        10777         7127
 0.629952202 -1.114275895  1.066463539  0.476892749 -1.792727386  0.638909346 -0.481918957  0.308015732  0.286230278  0.321711887  0.812040725
       10531         9640         3358         3980         9326         3230         5603        10126         9693         4576         3783
-0.418969485  0.400326872 -1.007173208  1.156726426 -0.870278751 -0.458301164 -0.819212375  0.542262520  1.121062506 -4.539903289  0.141804536
        7831        10106         5967         9301         7816         9267        11338         1386        10476         4706         2378
-0.271651259 -0.958723120  1.029856315 -0.052870920  1.103851412 -0.747511629  0.924604101  0.615406780  0.204156392 -1.262162327 -1.080560863
        4044
-0.003360325
```

Interpretation: y_outliers <- which(abs(studentized_residuals) > 3)

Outlying Points:

```
> cat("Outliers on Y-direction", y_outliers, "\n")
Outliers on Y-direction 87 179 196 219 259 414 466 585 711 927 976 1089 1101 1128 1146 1183 1361 1451 1617 1641 1663 1963 2189 2251 2254 2425 2444 2769 3098 315
7 3307 3381 3548 3553 3590 3779 3891 3918 3923 3948 3951 4051 4101 4102 4183 4310 4358 4448 4589 4891 5032 5103 5209 5273 5557 5788 5869 5885 5931 5946 5998 610
5 6186 6198 6204 6222 6447 6476 6628 6664 6712 6855 6901 6946 7370 7533 7535 7546 7559 7690 7922 8164 8214 8386 8393 8642 8764 8796 8865 8879 8962 9017 9070 909
8 9162 9238 9242 9260 9333 9345 9433 9486
```

Influential Cases Detection

DEFITTS

Cook's Distance

DFBETAS

DFFITS

It measures the influence of the observations in the predicted variable

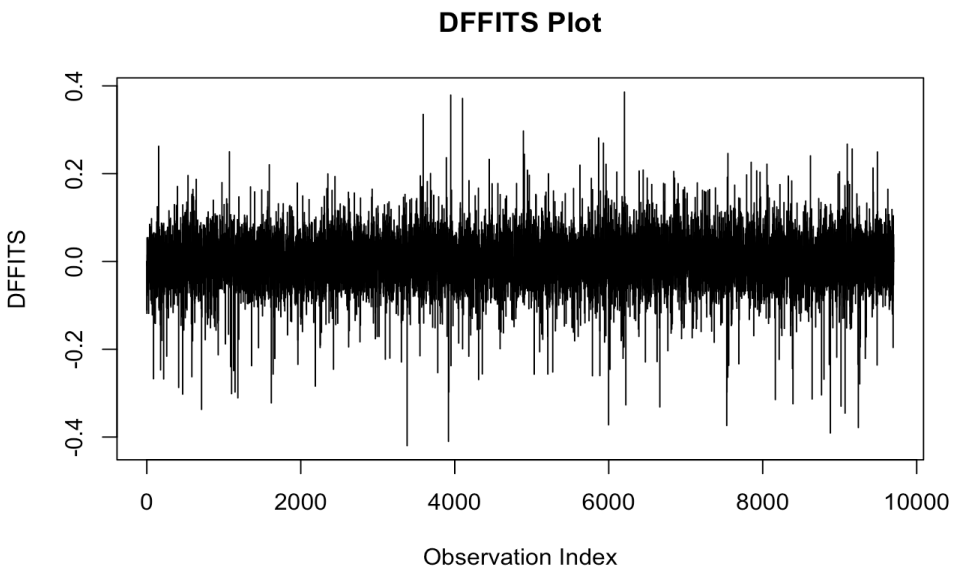
```
> print(dfbetas_values[1:100])
  [1] -1.030405e-02  1.147424e-03 -8.627184e-04  1.219008e-03  5.703592e-04  8.747898e-04 -7.955768e-05  4.584525e-04 -6.052667e-04  8.058309e-04  1.778491e-03 -6.860254e-05
 [13]  7.684681e-04 -1.260949e-04  4.674512e-03 -5.209136e-04  1.539730e-03 -1.302435e-03  1.066934e-02 -5.089646e-03  3.947268e-03  8.353963e-04 -3.751487e-04 -1.651833e-03
 [25]  1.157419e-03  1.645887e-03  5.573523e-03  2.769231e-03 -3.676081e-04  4.212894e-05 -7.019775e-05 -2.518104e-03  1.549864e-04  5.074593e-04  3.925665e-04  4.982940e-03
 [37] -3.055144e-03  1.918902e-03 -6.740058e-04 -2.968705e-04  1.166025e-04 -3.579917e-04 -4.643323e-04 -9.989668e-06  5.491319e-03 -4.153989e-04 -7.088631e-04  7.023354e-04
 [49]  2.149346e-03 -1.365677e-05 -1.833996e-03 -1.974604e-04 -7.239931e-05 -1.150435e-03  4.049627e-03 -6.643864e-04 -1.172176e-03 -3.341852e-04  2.005666e-03  8.976523e-04
 [61] -2.961203e-03  1.692448e-03  6.230190e-04  3.968820e-03  9.868678e-04 -3.289777e-04  8.436896e-04  7.381254e-05 -3.513929e-03  1.170829e-04  3.717003e-04 -1.081901e-03
 [73] -5.911531e-04 -5.927768e-04 -3.496305e-04 -9.992058e-04  4.113214e-04 -3.525497e-03  2.436848e-03  1.242111e-03 -1.187887e-03  6.850201e-04 -2.236647e-03 -3.411274e-03
 [85]  4.875427e-04  3.062334e-04  6.988477e-04 -8.825309e-04  1.495734e-03 -7.120717e-04  7.193679e-04 -4.447155e-06  6.196517e-04 -2.466676e-03 -5.890988e-03  6.329622e-04
 [97]  4.463558e-04 -7.512411e-04  3.045642e-04 -5.563699e-04
```

Cutoff value:

```
> print(cutoff)
[1] 0.1235158
```

Influential points:

```
> print(influential[1:100])
 4576  6387  6810  7864  4650   473  2037 10762  1905  9617  2339   259  1760   646  4374  2086  2894  9071  8860  6263 11187 10461  9657 10832  7306
   87   105   111   136   154   176   179   196   219   259   280   358   385   398   414   444   447   466   467   481   499   520   535   579   585
  606  7593  1658  5892  1436  4925   786  8641  9174   789  4494  4061  6483 10719  4163  9736   163  8601  7320  9423  2082  4297   825  7443  3101
  586   587   595   599   607   623   636   642   711   716   759   788   790   802   827   864   870   881   899   922   927   929   976  1004  1021
 8974 11459  6288 12011  9774  5799  8558  8390  2244 10257  1469  3323  4081  9496  5664 11759  4982  8260  9557  8284 10247  2908  4079  6927  4760
 1037  1046  1062  1073  1089  1101  1128  1146  1183  1195  1196  1347  1361  1401  1451  1493  1503  1570  1572  1591  1617  1641  1656  1663  1709
 4023  3014  1984  3901   970  2523  4288  3919  1518 12076  3820  5087  1936  3934  4405  5266 11094  9822   688  2618 10322 11380  9560  4621  4857
 1741  1772  1828  1853  1858  1869  1879  1881  1955  1961  1963  1997  2001  2004  2022  2025  2108  2168  2189  2251  2254  2269  2305  2327  2338
```



Cook's Distance

Measures the influence of the observations in overall model.

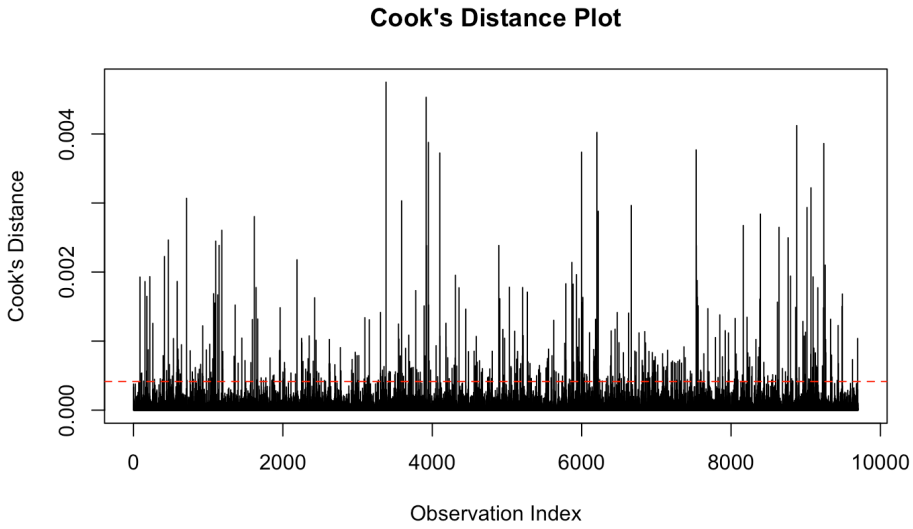
```
> print(cooks_d[1:100])
        2463         2511        10419         8718         2986         1842         9334         3371        11638         4761         6746
3.778341e-04 3.311382e-04 7.868889e-05 2.339342e-06 1.623393e-04 6.081747e-06 6.149604e-05 2.706005e-05 1.219580e-05 3.925983e-05 2.758858e-06
        9819         2757         5107         9145         9209        10205         2888         6170         2567         9642         9982
2.326511e-06 1.558741e-05 1.420365e-05 1.313081e-04 8.004494e-05 6.343268e-05 2.002130e-05 1.707927e-06 2.286915e-05 6.871771e-07 3.363659e-05
        2980         1614          555         4469         9359        10784        10730         7789         9991         9097         1047
7.745185e-06 3.778198e-04 5.810623e-10 3.664860e-05 2.237044e-05 2.050596e-05 5.975686e-05 7.006028e-06 6.678411e-05 4.683380e-07 1.715379e-05
        7067         3004         3207         7989         3995         8358          217         9506         8157        10821         6216
6.056012e-05 4.527592e-05 1.407429e-07 1.940624e-04 1.610227e-05 2.587493e-05 5.234805e-06 1.087453e-04 2.711986e-06 3.531229e-06 1.267287e-05
        8780         1599         4237         3937         4089         2907          294         8469           41         8508         7391
7.046952e-05 1.688089e-04 9.661622e-05 4.828347e-06 2.797579e-05 9.854871e-05 2.106259e-04 1.348455e-04 9.333395e-05 6.251326e-05 8.213767e-05
        6672         7284        11014        10987         2504         6742         9375         8944        11473         8566        10034
6.591711e-05 2.723507e-05 1.444786e-05 5.938526e-08 1.625060e-05 2.574147e-04 1.405860e-04 2.505691e-04 2.481719e-04 1.839289e-04 4.484524e-05
        6129        10274         4612         2117         6134          755         6553         5428         9198        10777         7127
3.408569e-05 9.782884e-05 9.879559e-05 1.961571e-05 4.002403e-04 5.976422e-05 2.538844e-05 8.181924e-06 7.730572e-06 9.840845e-06 7.214974e-05
       10531         9640         3358         3980         9326         3230         5603        10126         9693         4576         3783
1.657722e-05 1.425124e-05 9.195206e-05 1.094928e-04 7.241537e-05 1.938180e-05 1.016545e-04 2.825710e-05 1.041312e-04 1.927407e-03 1.683578e-06
        7831        10106         5967         9301         7816         9267        11338         1386        10476         4706         2378
6.438750e-06 7.885028e-05 1.091362e-04 2.248089e-07 1.161844e-04 5.372501e-05 7.741590e-05 3.262331e-05 3.525474e-06 1.420609e-04 9.433257e-05
        4044
1.023998e-09
```

Threshold:

```
> print(threshold)
[1] 0.0004139073
```

Influential Points:

```
> print(influential[1:100])
 4576  6387  6810  7864  4650   473  2037 10762  1905  9617  2339   259  1760   646  4374  2086  2894  9071  8860  6263 11187 10461  9657
   87   105   111   136   154   176   179   196   219   259   280   358   385   398   414   444   447   466   467   481   499   520   535
10832  7306   606  7593  1658  5892  1436  4925   786  8641  9174   789  4494  4061  6483 10719  4163  9736   163  8601  7320  9423  2082
  579   585   586   587   595   599   607   623   636   642   711   716   759   788   790   802   827   864   870   881   899   922   927
 4297   825  7443  3101  8974 11459  6288 12011  9774  5799  8558  8390  2244 10257  1469  3323  4081  9496  5664 11759  4982  8260  9557
  929   976  1004  1021  1037  1046  1062  1073  1089  1101  1128  1146  1183  1195  1196  1347  1361  1401  1451  1493  1503  1570  1572
 8284 10247  2908  4079  6927  4760  4023  3014  1984  3901   970  2523  4288  3919  1518 12076  3820  5087  1936  3934  4405  5266 11094
 1591  1617  1641  1656  1663  1709  1741  1772  1828  1853  1858  1869  1879  1881  1955  1961  1963  1997  2001  2004  2022  2025  2108
 9822   688  2618 10322 11380  9560  4621  4857
 2168  2189  2251  2254  2269  2305  2327  2338
```

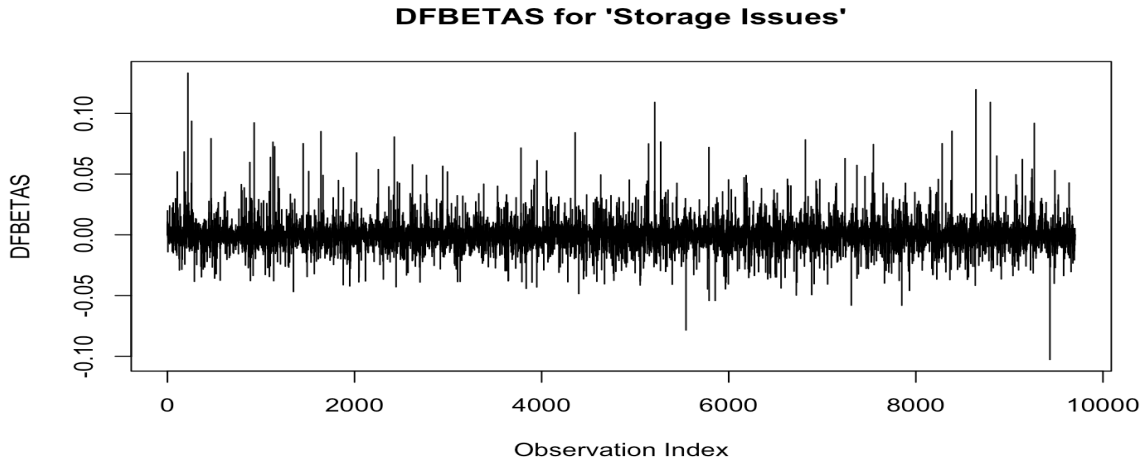


DFBETAS

Analyzes how individual observations affect specific regression coefficients.

```
$storage_issues
10987 11473  9198  6387  8011  4685  9985  8536 11067  2037  2313  6379   185   413
   59    64    75   105   122   135   139   144   155   179   180   190   192   194
 3581  3201  1905  6815  9637  5793  4213  4807  9426  9617  2470 10748  5214 11360
  206   216   219   228   232   236   241   256   258   259   275   276   307   319
 4357 10078  7727  8124  5423  7880  7042  3318  1517  8812 11054  7392  1498 11287
  376   403   418   419   423   426   476   497   512   521   525   530   532   537
 1831  4546 11528  2004   606   602  7032 11032  7639  8778  1352 11132  2801 11525
  546   557   564   584   586   618   643   663   671   676   687   726   755   771
 6483  7575  5003  7904  8601  7601  2082  4338  6736  4311  4812 11205  7053  3149
  790   805   821   853   881   923   927   933   939   947   959   962   964   967
 5278 11657  7319  9020  2970  5317  5799  8558 11310  8390   956  9027  2306  2437
 1030  1036  1040  1059  1085  1096  1101  1128  1130  1146  1155  1163  1171  1185
 7324  2287  3822  4312  1899  1003  9336  5216  5883 10645  1571  8870  9496 11543
 1192  1199  1222  1301  1304  1307  1308  1314  1317  1320  1343  1379  1401  1429
 5664  4815  9315  6195  5736 10623  1611  6880 11364  2745   572  6123  8684  2586
 1451  1453  1465  1470  1478  1484  1487  1496  1510  1519  1521  1541  1546  1563
10030  7242 11485  1761 10496  2908   793  5222 10147  2229 10718  7531 12102 11228
 1577  1585  1593  1613  1618  1641  1653  1660  1687  1692  1697  1760  1778  1782
 5386  4879  4376  2871  1887  2523  7419  3919  1701  8371  6819  7139  2873  9170
 1806  1823  1839  1845  1848  1869  1877  1881  1910  1917  1918  1948  1950  1956
 1781   774  6370  4405  9940  2021  6001  2173  5301  4001  1007  3295  7285  9124
 1967  1993  2021  2022  2023  2036  2046  2048  2065  2067  2080  2082  2105  2117
 3215   474  6143  6606  5961  1964  4725 10322  7754  2195 11995  4010  7868  9356
```

```
> #threshold
> n <- nrow(train_df)  # Number of observations
> threshold <- 2 / sqrt(n)
> print(threshold)
[1] 0.02030588
>
```



Model Evaluation

Train-Test Split:

```
> R_squared <- 1 - (SS_Residual / SS_Total)
> R_squared
[1] 0.8868954
```

```
The MSPR value and R-square value of test data is 0.03829644 and 0.8868954
```

```
The ratio of MSPR and MSE is: 1.021966
```

K-cross Validation:

```
Linear Regression

12127 samples
    5 predictor

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 9701, 9700, 9702, 9702, 9703
Resampling results:

  RMSE       Rsquared   MAE
  0.1947053  0.8874002  0.1530196
```

Inference in Regression Analysis

$$b_k - s(b_k)t\left(1-\frac{\alpha}{2};n-p\right) \le \beta_k \le b_k + s(b_k)t\left(1-\frac{\alpha}{2};n-p\right)$$

95 %Confidence Interval for βk of MLR Model

```
> print(conf_intervals)
                                 Coefficient     Estimate  Lower_95_CI   Upper_95_CI
(Intercept)                      (Intercept) 10.257407831 10.224011222 10.2908044400
w_est_year1997                 w_est_year1997 -0.004355300 -0.044139953  0.0354293534
w_est_year1998                 w_est_year1998  0.008010816 -0.029095884  0.0451175159
w_est_year1999                 w_est_year1999  0.012617894 -0.024596558  0.0498323460
w_est_year2000                 w_est_year2000  0.016378655 -0.020238453  0.0529957629
w_est_year2001                 w_est_year2001  0.001709980 -0.035414332  0.0388342922
w_est_year2002                 w_est_year2002 -0.006838073 -0.043855797  0.0301796507
w_est_year2003                 w_est_year2003 -0.003757591 -0.041205335  0.0336901537
w_est_year2004                 w_est_year2004  0.020890102 -0.016255381  0.0580355851
w_est_year2005                 w_est_year2005  0.003440587 -0.034220861  0.0411020346
w_est_year2006                 w_est_year2006 -0.035805483 -0.072602676  0.0009917103
w_est_year2007                 w_est_year2007 -0.111135743 -0.148102476 -0.0741690097
w_est_year2008                 w_est_year2008 -0.162147227 -0.199687966 -0.1246064892
w_est_year2009                 w_est_year2009 -0.182804207 -0.220530478 -0.1450779363
w_est_year2010                 w_est_year2010 -0.177422926 -0.214791284 -0.1400545687
w_est_year2011                 w_est_year2011 -0.175330558 -0.212822288 -0.1378388274
w_est_year2012                 w_est_year2012 -0.241695438 -0.279178942 -0.2042119343
w_est_year2013                 w_est_year2013 -0.297396082 -0.335353959 -0.2594382051
w_est_year2014                 w_est_year2014 -0.342639799 -0.380660858 -0.3046187399
w_est_year2015                 w_est_year2015 -0.340294635 -0.378514590 -0.3020746807
w_est_year2016                 w_est_year2016 -0.414797171 -0.453286804 -0.3763075377
w_est_year2017                 w_est_year2017 -0.502793043 -0.541835121 -0.4637509648
w_est_year2018                 w_est_year2018 -0.608203669 -0.647995597 -0.5684117402
w_est_year2019                 w_est_year2019 -0.701686722 -0.741718000 -0.6616554440
w_est_year2020                 w_est_year2020 -0.752543095 -0.793526154 -0.7115600359
w_est_year2021                 w_est_year2021 -0.817342081 -0.861098411 -0.7735857518
w_est_year2022                 w_est_year2022 -0.849939096 -0.898736501 -0.8011416917
govt_certA+                       govt_certA+  0.011089449 -0.003573829  0.0257527279
govt_certB                         govt_certB -0.117898904 -0.130128683 -0.1056691257
govt_certB+                       govt_certB+ -0.105981699 -0.118264515 -0.0936988836
govt_certC                         govt_certC -0.036786122 -0.048966769 -0.0246054744
storage_issues                 storage_issues  0.305306369  0.298504182  0.3121085572
transport_issue1             transport_issue1 -0.044119312 -0.054515531 -0.0337230921
transport_issue2             transport_issue2 -0.074750500 -0.093833292 -0.0556677078
transport_issue3             transport_issue3 -0.124880136 -0.148633758 -0.1011265132
transport_issue4             transport_issue4 -0.097039647 -0.120795679 -0.0732836148
temperature_regulation1 temperature_regulation1  0.022787837  0.013317928  0.0322577454
```

95 %Confidence Interval for E(Xh)

$$\hat{Y}_h - s(\hat{Y}_h)t\left(1-\frac{\alpha}{2};n-p\right) \leq E(X_h) \leq \hat{Y}_h + s(\hat{Y}_h)t\left(1-\frac{\alpha}{2};n-p\right)$$

```
The 95% confidence interval for E(X_h) is:
 8.7814 <= E(X_h) <= 8.7917
```

Prediction Interval for New Observation *Yh(new)*

$$\hat{Y}_h - \text{s(pred)}\, t\left(1-\frac{\alpha}{2}; n-p\right) \le \boldsymbol{Y_{h(new)}} \le \hat{Y}_h + \text{s(pred)}\, t\left(1-\frac{\alpha}{2}; n-p\right)$$

```
The 95% prediction interval for New Observation is:
 8.4054 <= Y_h_new <= 9.1677
```

$$\hat{Y}_h - \text{s(meanpred)}\, t\left(1-\frac{\alpha}{2}; n-p\right) \le \textbf{mean of m obsr.} \le \hat{Y}_h + \text{s(meanpred)}\, t\left(1-\frac{\alpha}{2}; n-p\right)$$

```
The 95% prediction interval for New Observation is:
 8.7598 <= mean of m obsr <= 8.8133
```

Confidence of Region (CR) for Regression Surface

$$\hat{Y}_h - Ws(\hat{Y}_h) \leq \text{CR at } X_h \leq \hat{Y}_h + Ws(\hat{Y}_h)$$

```
The 95% confidence interval for E(X_h) is:
 8.7814 <= E(X_h) <= 8.7917
```

Conclusion

- The regression model shows strong performance with an R-squared value of 0.8885, explaining 88.85% of the variation in product weight.
- The model performs consistently, as indicated by the MSPR/MSE of 1.021966.
- All the confidence intervals are narrow that shows key coefficients reflects reliable and stable population estimates.
- The prediction interval for new observations is precise, indicating robustness in model predictions.

Recommendations

- Dimensionality reduction: Use Principal Component Analysis to transform highly correlated predictors into orthogonal components to reduce multicollinearity
- Using optimization algorithms like weighted linear regression or Elastic Net Regression, combining Lasso L1 and Ridge L2 penalties to balance feature selection and coefficient shrinkage
- Future Validation: Regularly validate the model with updated data to ensure its robustness and adaptability to changing patterns in production and operations

References

Kutner, Michael H., Nachtsheim, Christopher J., Neter, John(2004). Applied Linear Regression Models (4th ed.). McGraw-Hill Education.

Montgomery, D. C., Peck, E. A., & Vining, G. G. (2021). Introduction to linear regression analysis (6th ed.)Wiley.

Thank you!