Prepared by group 3

Group Project

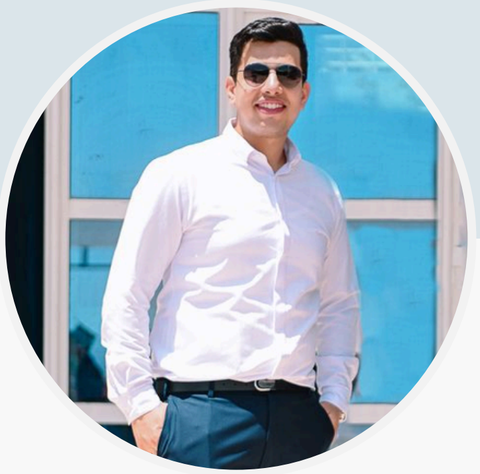
Anova Regression Correlation Analysis

16 December, 2024

Team Members

Sana khiari

Sarra Bouden



Mohamed Amine Brahmi

Team Members

Mehdi Bchir

Ines Neji

Oumaima BelHaj

Analysis Overview and Datasets

Objective:

Distinguish main vulnerabilities of statistical techniques across two datasets

1.Gas Turbine CO and NOx Emission Dataset

Purpose: Study flue gas emissions (CO and NOx)

2.AI4I 2020 Predictive Maintenance Dataset

Purpose: Reflect real predictive maintenance scenarios

1.Gas Turbine CO and NOx Emission Dataset

Variables in the Gas Turbine Dataset

Dataset Context

- Source: UCI Machine Learning Repository
- Collected By: Heysem, Pınar, and Erdinç Uzun (2019)
- Purpose: Predict CO and NOx emissions from gas turbines

Details:

- Timeframe: 2011–2015
- Instances: 36,733 (Five years of data)
- Operating Range: Partial load (75%) to full load (100%)

Variables in the Gas Turbine Dataset

Dependent Variables

- CO Emissions: Carbon monoxide emissions from gas turbines
- NOx Emissions: Nitric oxide emissions from gas turbines

Independent Variables

- Ambient Temperature (AT)
- Ambient Pressure (AP) in mbar
- Humidity (AH) in %
- Air Filter Difference Pressure (AFDP) in mbar
- Gas Turbine Exhaust Pressure (GTEP) in mbar
- Turbine Inlet Temperature (TIT)
- Turbine After Temperature (TAT) in °C
- Compressor Discharge Pressure (CDP) in mbar
- Turbine Energy Yield (TEY) in MWh

Data Importation and Preprocessing Steps

1. Import Data: Used read.table() to load datasets for each year (2011–2015).
2. Combine Data: Concatenated datasets using rbind() into a single data frame, gt_all.

```
gt_2011 <- read.table(file = file.choose(), header = TRUE, sep = ",", dec = ".")
gt_2012<- read.table(file = file.choose(), header = TRUE, sep = ",", dec = ".")
gt_2013<- read.table(file = file.choose(), header = TRUE, sep = ",", dec = ".")
gt_2014<- read.table(file = file.choose(), header = TRUE, sep = ",", dec = ".")
gt_2015<- read.table(file = file.choose(), header = TRUE, sep = ",", dec = ".")


# Concatenate the data frames data from 2011year to  2015
gt_all <- rbind(gt_2011, gt_2012, gt_2013, gt_2014, gt_2015)
View(gt_all)
```

Dimensions of the Dataset

```
# Dimensions (rows, columns)
dim_data <- dim(gt_all)
print(paste("Dimensions (rows, columns):", paste(dim_data, collapse=" x ")))
```

```
> dim_data <- dim(gt_all)
> print(paste("Dimensions (rows, columns):", paste(dim_data, collapse=" x ")))
[1] "Dimensions (rows, columns): 36733 x 11"
```

Total Observations: 36,733 rows across 11 variables.

Data understanding

Display the Data Type of Each Column

```
> sapply(gt_all, class)
       AT         AP         AH       AFDP       GTEP        TIT        TAT        TEY        CDP 
"numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" 
       CO        NOX 
"numeric" "numeric" 
```

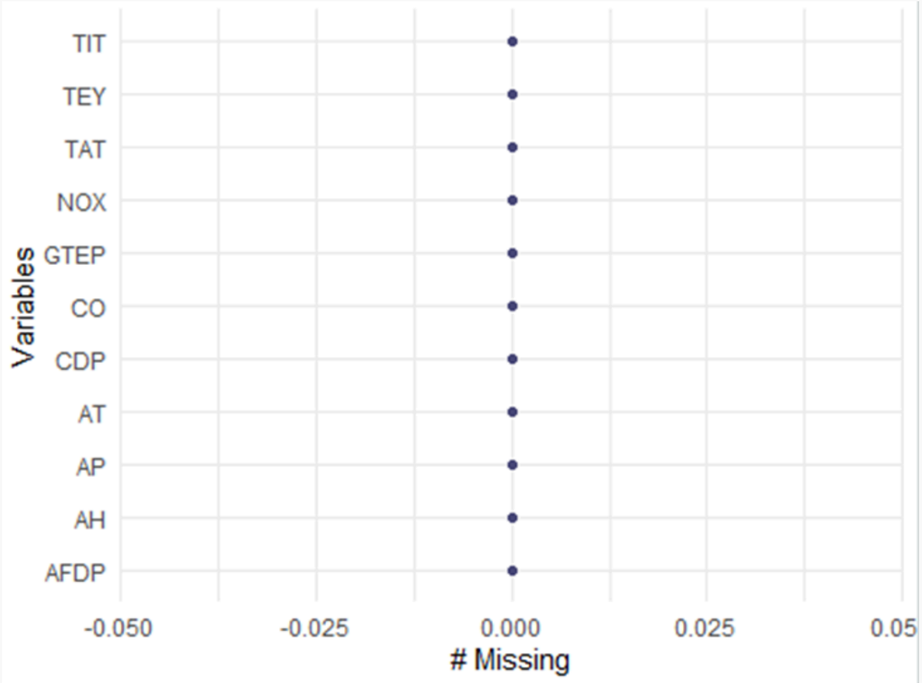
Summary Statistics of the Dataset

```
> summary(gt_all)
       AT                AP              AH              AFDP            GTEP      
 Min.   :-6.235   Min.   : 985.9   Min.   : 24.09   Min.   :2.087   Min.   :17.70  
 1st Qu.:11.781   1st Qu.:1008.8   1st Qu.: 68.19   1st Qu.:3.356   1st Qu.:23.13  
 Median :17.801   Median :1012.6   Median : 80.47   Median :3.938   Median :25.10  
 Mean   :17.713   Mean   :1013.1   Mean   : 77.87   Mean   :3.926   Mean   :25.56  
 3rd Qu.:23.665   3rd Qu.:1017.0   3rd Qu.: 89.38   3rd Qu.:4.377   3rd Qu.:29.06  
 Max.   :37.103   Max.   :1036.6   Max.   :100.20   Max.   :7.611   Max.   :40.72  
      TIT             TAT             TEY             CDP              CO          
 Min.   :1001   Min.   :511.0   Min.   :100.0   Min.   : 9.852   Min.   : 0.00039  
 1st Qu.:1072   1st Qu.:544.7   1st Qu.:124.5   1st Qu.:11.435   1st Qu.: 1.18240  
 Median :1086   Median :549.9   Median :133.7   Median :11.965   Median : 1.71350  
 Mean   :1081   Mean   :546.2   Mean   :133.5   Mean   :12.061   Mean   : 2.37247  
 3rd Qu.:1097   3rd Qu.:550.0   3rd Qu.:144.1   3rd Qu.:12.855   3rd Qu.: 2.84290  
 Max.   :1101   Max.   :550.6   Max.   :179.5   Max.   :15.159   Max.   :44.10300  
      NOX        
 Min.   : 25.91  
 1st Qu.: 57.16  
 Median : 63.85  
 Mean   : 65.29  
 3rd Qu.: 71.55  
 Max.   :119.91  
```

Data Preparation

Check of missing values

```
library(naniar)

# Check the number of missing values for each column
gg_miss_var(gt_all)
```



Check duplicate rows

```
> duplicated_rows <- gt_all[duplicated(gt_all), ]
> print(paste("Number of duplicated rows:", nrow(duplicated_rows)))
[1] "Number of duplicated rows: 7"
```

Removed all duplicates

```
> gt_all <- gt_all[!duplicated(gt_all), ]
> print(paste("le nombre des lignes duplicate  :", sum(duplicated(gt_all))))
[1] "le nombre des lignes duplicate  : 0"
```

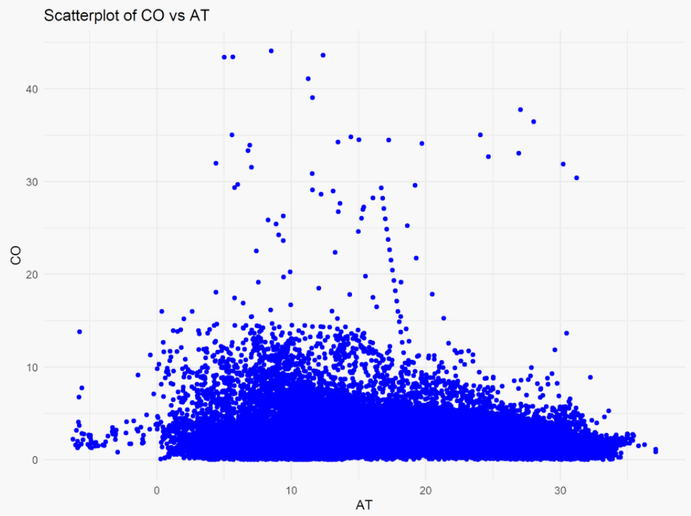
Data Importation and Preprocessing Steps

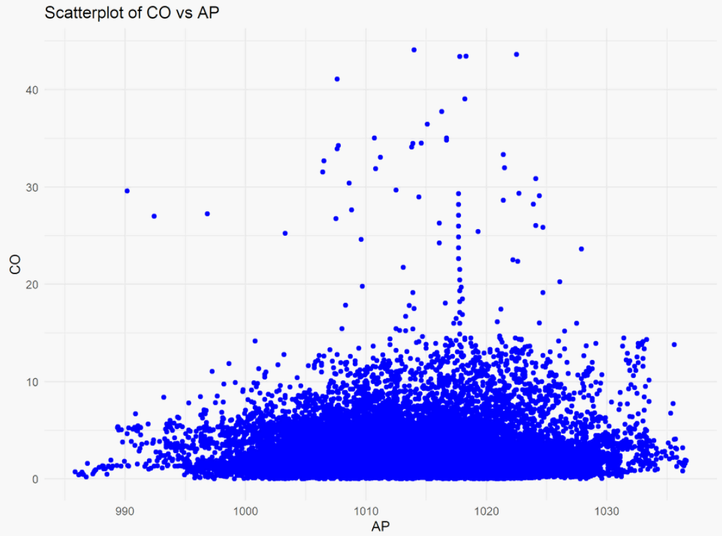
```r
# Z-Score Standardization function
z_score_scale <- function(x) {
  (x - mean(x)) / sd(x)
}

# Apply standardization to each column
standardized_data <- as.data.frame(lapply(gt_all, z_score_scale))

# View standardized data
print(standardized_data)
View(standardized_data)

# Check the column names again to ensure CO is the dependent variable
print(names(standardized_data))  # Should print the column names including "CO"

# Load necessary libraries
library(ggplot2)
library(gridExtra)

# Create a scatterplot for each independent variable, excluding "CO"
independent_vars <- setdiff(names(standardized_data), "CO")  # Exclude "CO"

# Create scatterplots for each independent variable against CO
plot_list <- lapply(independent_vars, function(var) {
  ggplot(standardized_data, aes_string(x = var, y = "CO")) +
    geom_point(color = "blue") +
    labs(
      title = paste("Scatterplot of CO vs", var),
      x = var ,
      y = "CO"
    ) +
    theme_minimal()
})

# Check the length of the plot list to ensure it's populated
print(length(plot_list))  # Should print the number of independent variables
```

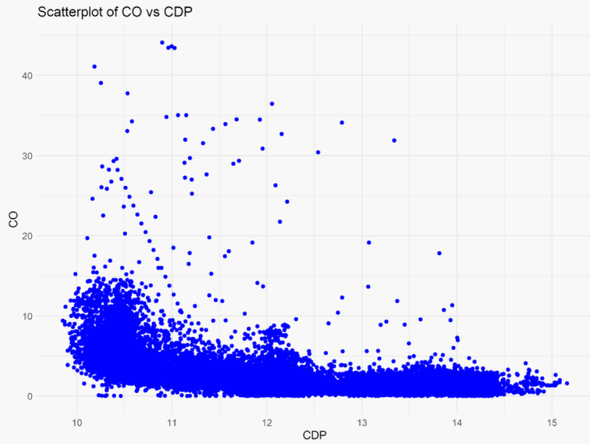
Applying Standardization:
We applied the Z-score standardization to each column of the dataset .

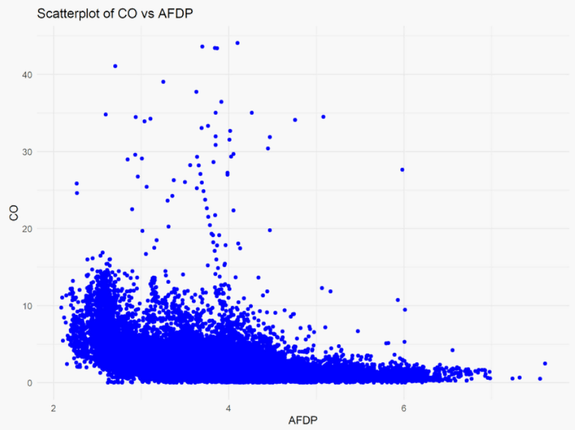
Scatterplots Creation:
Goal: To explore the relationship between the dependent variable (CO) and each independent variable.

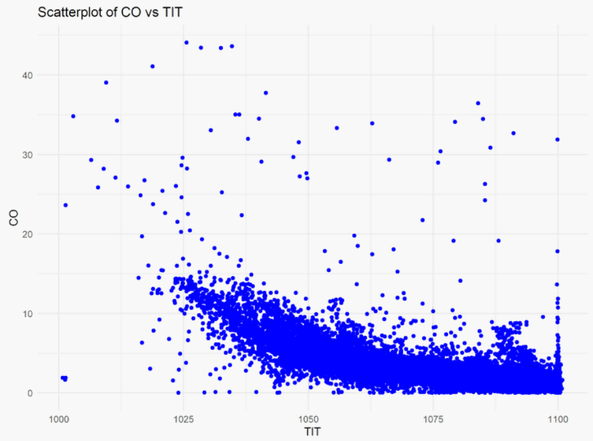
Interpretation of Scatter Plots

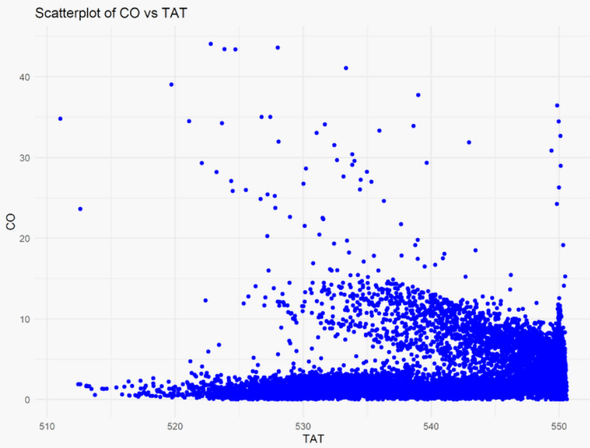




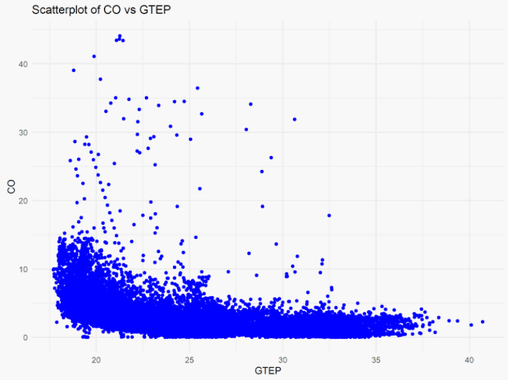


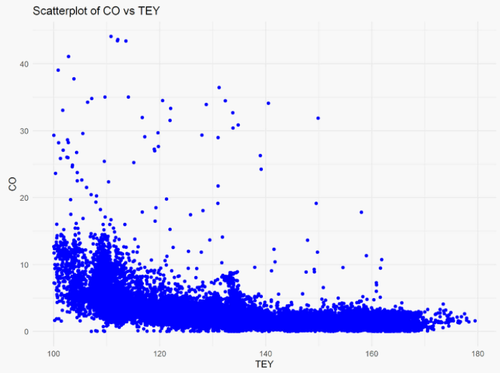


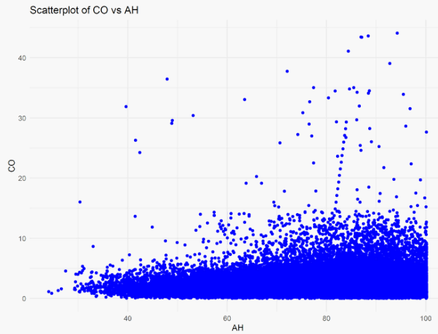




Interpretation of Scatter Plots

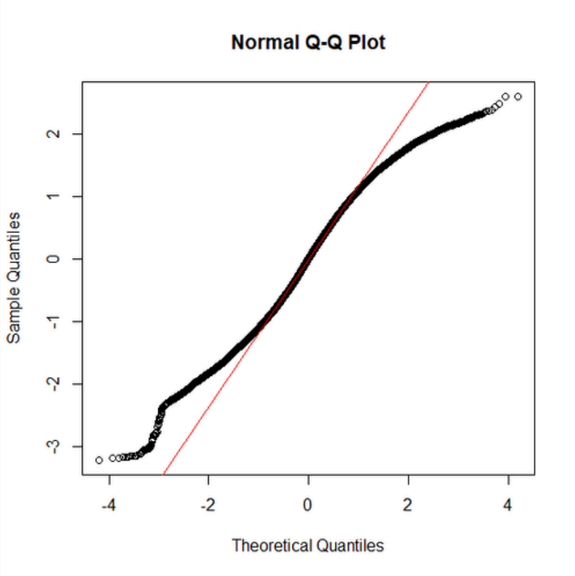


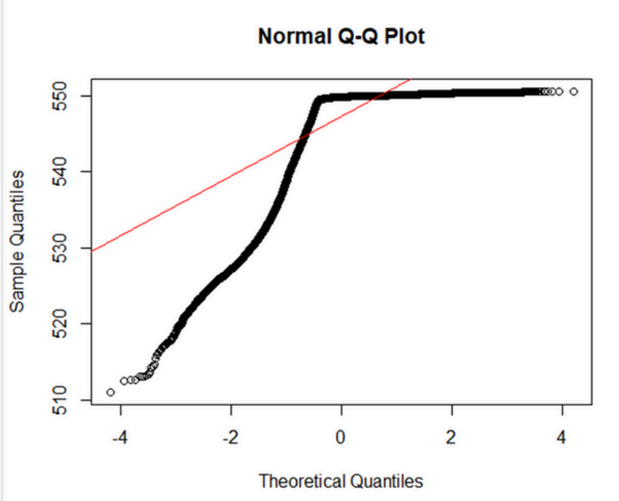


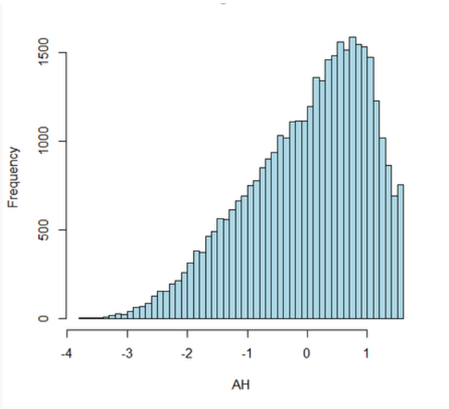


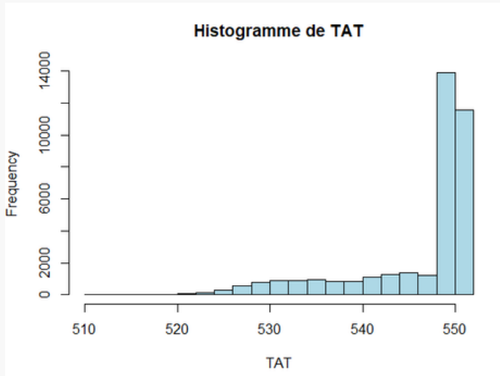
- The scatter plots show how each independent variable is related to the dependent variable (CO).
- By looking at the graphs, we can see there is a relationship between the variables.
- However, it doesn't seem that the relationship is linear.

Normality Check: Histogram and Q-Q Plot









The points on the plot does not follow the straight red line

The histogram does not looks symmetric,

==> the data is not normally distributed.

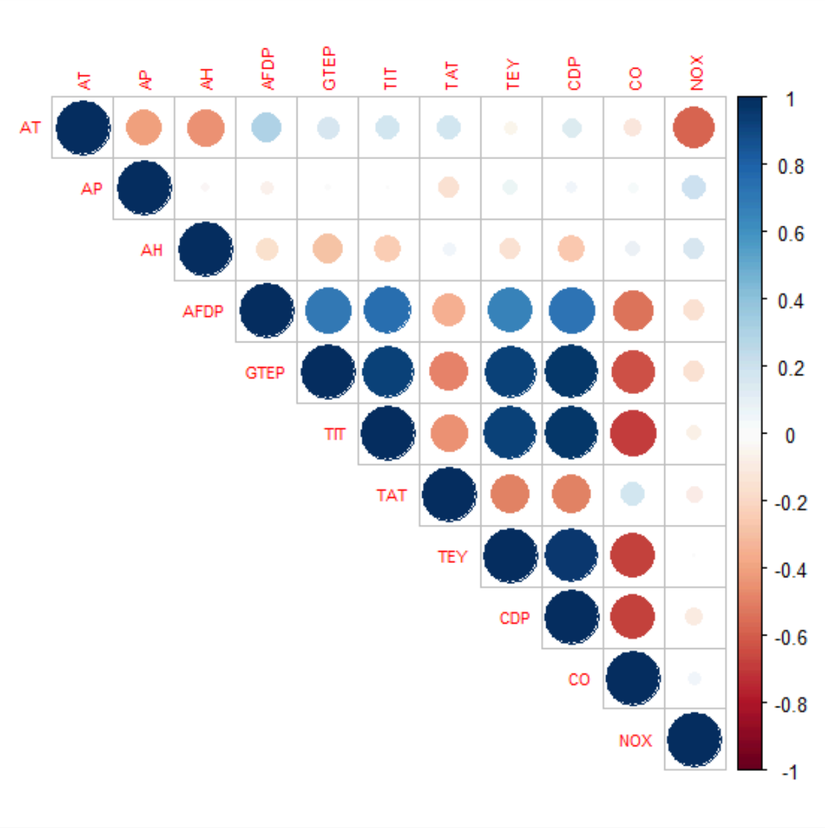
Correlation analysis techniques

- **Pearson correlation** is used when you assume a linear relationship between the variables and that they follow a normal distribution.*

- **Spearman correlation** is more appropriate when the relationship is monotonic but not necessarily linear, or when dealing with non-normally distributed or ordinal data.

====> in our case we will use Spearman correlation

Correlation analysis techniques



Strong Positive Correlations:
Variables like GTEP and TEY show dark blue,
Strong Negative Correlations:
Variables like AT and TEY exhibit a dark red relationship, signifying a strong negative correlation.
Weak or No Correlation:
Lighter or neutral colors indicate weak or no linear relationship.

Correlation analysis techniques

```
> print(co_spearman)

        Spearman's rank correlation rho

data:  standardized_data$CO and standardized_data$TIT
S = 1.4022e+13, p-value < 2.2e-16
alternative hypothesis: true rho is not equal to 0
sample estimates:
       rho
-0.6984499
```

=>Spearman's Rank Correlation: There is a moderately strong **negative monotonic relationship between CO and TIT** with a correlation coefficient of **-0.698**, and this relationship is statistically significant (p-value < 2.2e-16)

```
> high_corr_vars <- findCorrelation(spearman_corr, cutoff = 0.8)
> # Print the variables to exclude due to multicollinearity
> print("Variables to exclude due to high correlation:")
[1] "Variables to exclude due to high correlation:"
> print(names(standardized_data)[high_corr_vars])
[1] "CDP"  "GTEP" "TIT"
>
```

=>Three variables ("CDP", "GTEP", and "TIT") that are highly correlated with others in the dataset.

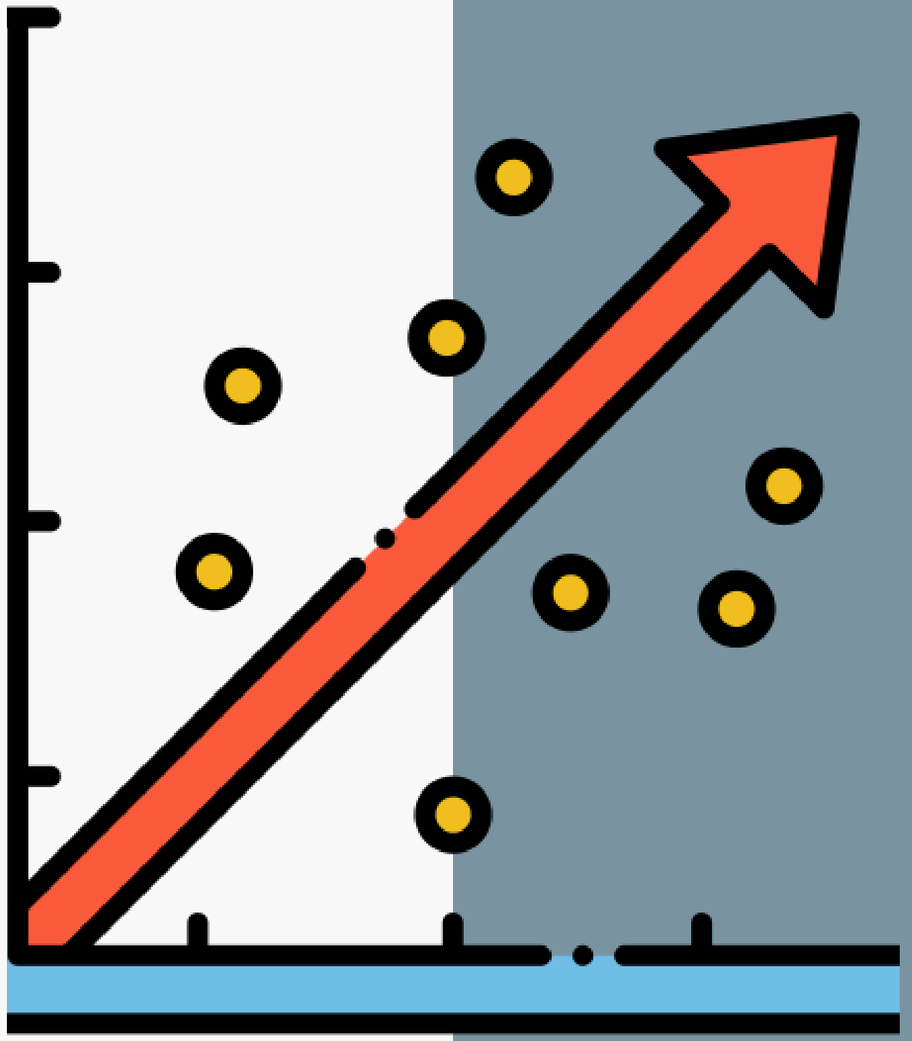
So we removed the highly correlated variables (CDP, GTEP, and TIT) from the dataset to prevent multicollinearity

Objective of regression

The objective of regression is to understand and predict the relationship between two or more variables . It helps us figure out how changes in one variable affect another

types of regression

- Linear Regression (simple or multiple)
- Log-Linear Regression
- Nonlinear Regression
- .....



Regressin model first execution

1. Full Model Creation:

A linear regression model (full_model)

The model summary shows:

Adjusted R-squared: **0.54**, indicating that about **54.%** of the variability in CO is explained by the independent variables.

```
Call:
lm(formula = CO ~ AT + AP + AH + AFDP + TAT + TEY, data = standardized_data)

Residuals:
    Min      1Q  Median      3Q     Max
-5.3008 -0.3174 -0.0545  0.2450 15.3510

Coefficients:
              Estimate Std. Error  t value Pr(>|t|)
(Intercept) -3.678e-15  3.515e-03    0.000    1.000
AT          -1.088e-01  5.261e-03  -20.673   <2e-16 ***
AP          -3.693e-03  4.001e-03   -0.923    0.356
AH          -7.252e-02  4.295e-03  -16.883   <2e-16 ***
AFDP        -8.484e-02  5.363e-03  -15.819   <2e-16 ***
TAT         -5.865e-01  5.156e-03 -113.740   <2e-16 ***
TEY         -9.330e-01  5.859e-03 -159.246   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6735 on 36719 degrees of freedom
Multiple R-squared:  0.5464,    Adjusted R-squared:  0.5464
F-statistic:  7373 on 6 and 36719 DF,  p-value: < 2.2e-16
```

Regressin model first execution

2. Stepwise Model Selection: (using The BIC-based model)

A stepwise regression was performed to improve the model by removing insignificant variables.

Stepwise Model Results:

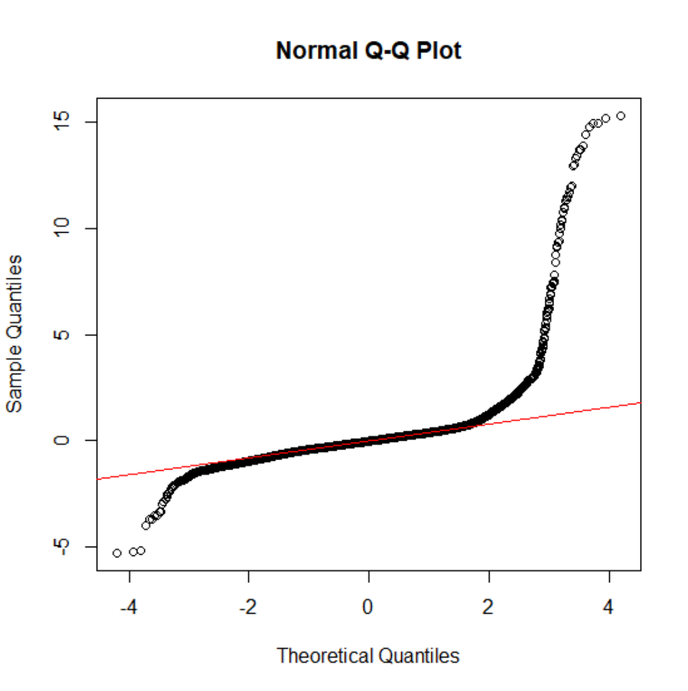
AP was removed because it contributed the least to the model based on AIC and BIC criteria.

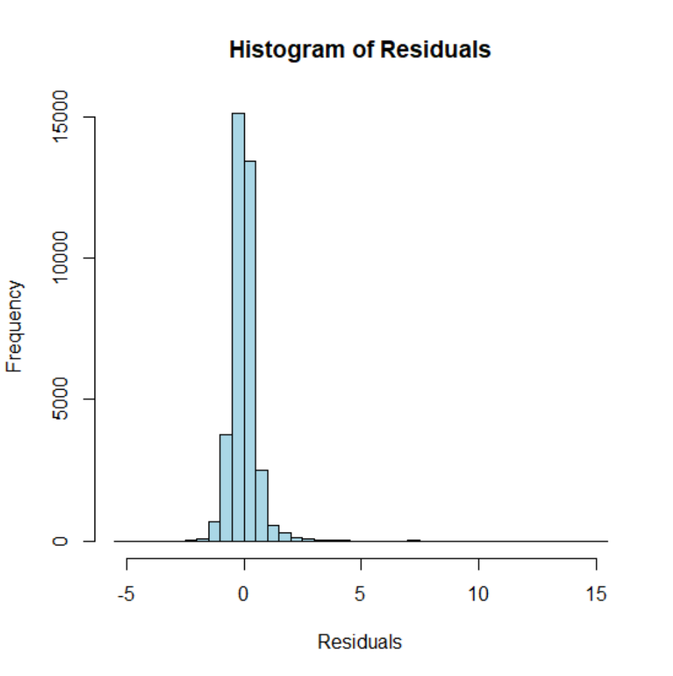
Adjusted R-squared: 0.5463651 slightly reduced but still comparable to the full model.

```
> stepwise_model_bic <- stepAIC(full_model, direction
Start:  AIC=-28963.8
CO ~ AT + AP + AH + AFDP + TAT + TEY

       Df Sum of Sq   RSS      AIC
- AP    1       0.4 16658 -28973.5
<none>              16657 -28963.8
- AFDP  1     113.5 16771 -28724.9
- AH    1     129.3 16786 -28690.3
- AT    1     193.9 16851 -28549.3
- TAT   1    5868.6 22526 -17889.7
- TEY   1   11503.8 28161  -9689.5

Step:  AIC=-28973.46
CO ~ AT + AH + AFDP + TAT + TEY

       Df Sum of Sq   RSS      AIC
<none>              16658 -28973.5
- AFDP  1     114.2 16772 -28733.0
- AH    1     134.0 16792 -28689.8
- AT    1     222.7 16880 -28496.2
- TAT   1    5889.8 22547 -17865.0
```

Regressin model first execution

1. Normality Check of Residuals

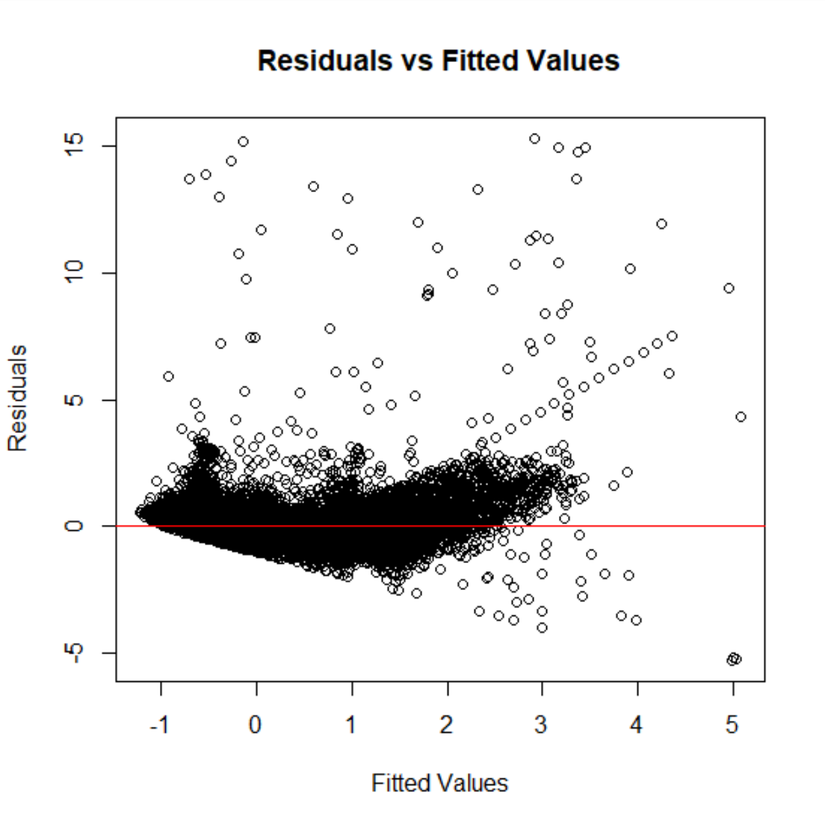




=>it can be seen that the normality assumption is not met

Regressin model first execution

2. Homoscedasticity Check



Plot should show a random scatter around zero if homoscedasticity holds. But here is not the case

The article uses ANOVA for model comparison, however, we cannot apply it because the residuals do not follow a normal distribution, violating a key assumption of the method.

Regressin model first execution

4. Model Diagnostics:

Durbin-Watson Test: Statistic = 0.87, which is relatively low (close to 0). This suggests positive autocorrelation in the residuals, meaning that the residuals are not independent of each other. Ideally, a value close to 2 is preferred, indicating no autocorrelation.

Residual standard error: ~0.66, suggesting the model's predictions are moderately precise.

```
> dw_test <- dwtest(stepwise_model_bic)
> cat("Durbin-Watson statistic: ", dw_test$statistic, "\n")
Durbin-Watson statistic:  0.8763425
>
```

Regressin model first execution

5. Coefficients of the Final Model

```
> # Coefficients of the final model
> cat("Coefficients:\n")
Coefficients:
> print(summary(stepwise_model_bic)$coefficients)
                Estimate  Std. Error       t value      Pr(>|t|)
(Intercept) -4.048462e-16 0.003448342 -1.174031e-13  1.000000e+00
AT          -1.535872e-01 0.008888548 -1.727923e+01  1.238488e-66
AH          -4.816680e-02 0.004146938 -1.161503e+01  3.918712e-31
AFDP        -4.733891e-02 0.005438715 -8.704063e+00  3.332943e-18
GTEP         1.866437e-01 0.018612623  1.002780e+01  1.234000e-23
TIT         -5.541141e-01 0.021342403 -2.596306e+01 2.777085e-147
TAT         -2.460623e-01 0.011001653 -2.236594e+01 4.603160e-110
TEY         -1.285673e+00 0.052518296 -2.448048e+01 2.700693e-131
CDP          8.932929e-01 0.050950527  1.753256e+01  1.539431e-68
> |
```

Regressin model first execution

Interpretation:

Both the full and stepwise models demonstrate that independent variables such as TIT, TEY, GTEP, and CDP have the strongest effects on CO.

The stepwise model is preferred as it simplifies the model without a significant loss of explanatory power.

Autocorrelation in residuals may require additional diagnostics or adjustments .

Regressin model second execution

1. Influential Points

Cook's Distance and Leverage are used to detect influential points in the regression model.

Criteria for Influence: Cook's Distance > 1

Leverage > 2 * (Number of observations / Number of predictors)

Result: No influential points identified based on the given thresholds.

```
> cat("Influential points (Cook's distance > 1 or leverage > threshold):", influential_points, "\n")
Influential points (Cook's distance > 1 or leverage > threshold):
>
```

Regressin model second execution

2. Multivariate Outliers (Mahalanobis Distance)

Mahalanobis Distance was used to detect multivariate outliers.

Threshold: p-value < 0.001.

Identified Multivariate Outliers: A separate set of outliers identified based on multivariate criteria.

3. Outlier Detection

Standard Residuals: Points with absolute residuals > 3.29 are considered outliers.

Identified Outliers: 1651 data points (4.49% of the dataset).

```
> all_outliers <- unique(c(outliers_std_residuals, multivariate_outliers))
> length(all_outliers)
[1] 1651
> # Calculate the percentage of outliers
> percentage_outliers <- (length(all_outliers) / nrow(standardized_data)) * 100
> # Print the result
> cat("Percentage of outliers: ", percentage_outliers, "%\n")
Percentage of outliers:  4.495453 %
>
```

Regressin model second execution

4. Model Refitting

Outliers were removed from the dataset, and the model was refitted.

```
> cat("Model Summary:\n")
Model Summary:
> cat("R-squared:", model_summary$r.squared, "\n")
R-squared: 0.6178168
> cat("Adjusted R-squared:", model_summary$adj.r.squared, "\n")
Adjusted R-squared: 0.6177514
```

===>R-squared: 0.6177

- The increase in R-squared after dropping outliers indicates that the model is now more accurate in explaining the variation in CO.

=>The outliers were negatively influencing the model.

Regressin model second execution

5. Final Model Coefficients

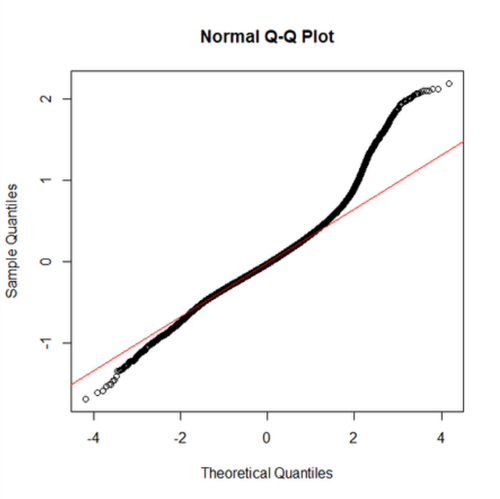
Coefficients of the Final Model: Display key predictors and their significance.

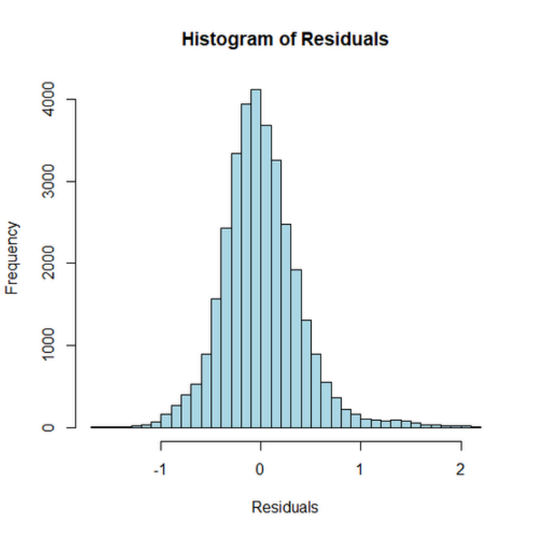
Significant Variables: All coefficients are significant with p-values < 0.05 .

```
Residuals:
     Min       1Q   Median       3Q      Max
-1.56993 -0.27284 -0.04435  0.24059  2.54345

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.055801   0.002297  -24.29  < 2e-16 ***
AT          -0.121178   0.003546  -34.18  < 2e-16 ***
AP          -0.014220   0.002740   -5.19 2.12e-07 ***
AH          -0.069315   0.002856  -24.27  < 2e-16 ***
AFDP        -0.069613   0.003628  -19.19  < 2e-16 ***
TAT         -0.382877   0.003787 -101.10  < 2e-16 ***
TEY         -0.728380   0.004213 -172.89  < 2e-16 ***
---
```

Regressin model second execution

1. Normality Check of Residuals

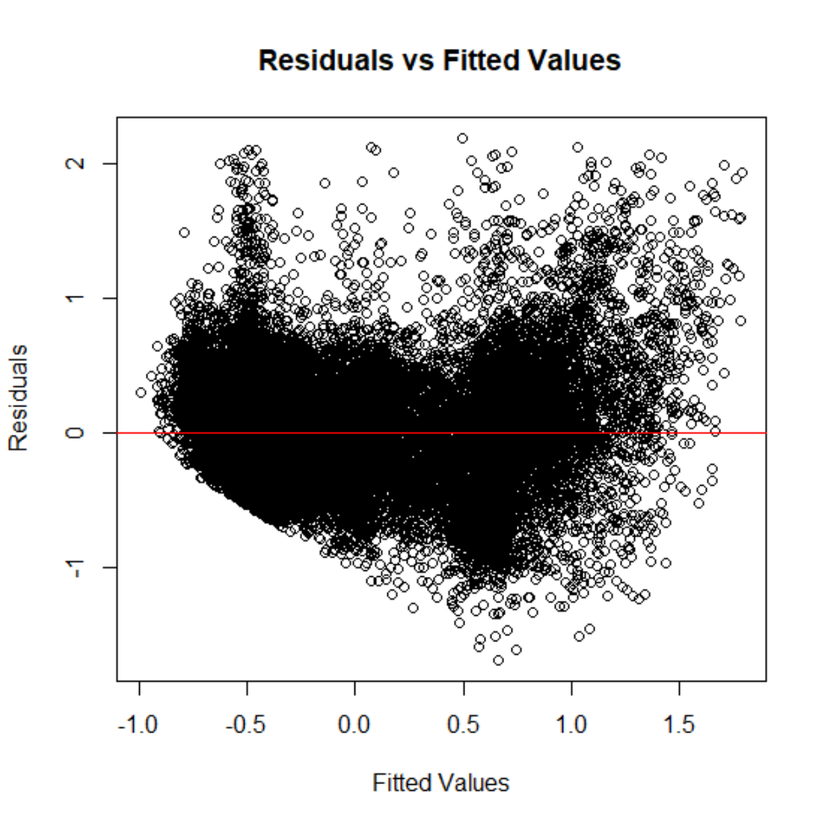




After removing the outliers, the distribution of residuals improves. While it is still not perfectly normal, this shows a noticeable improvement compared to before.

Regressin model second execution

2. Homoscedasticity Check



The residuals show improved homoscedasticity. Although the variance is not perfectly consistent, the pattern is noticeably better than before.

2.AI4I 2020 Predictive Maintenance Dataset

Dataset Overview

Source: UCI Machine Learning Repository

Dataset Name: AI4I 2020 Predictive Maintenance

Size: 10,000 observations, 14 variables

Variables in the Dataset

```
# importation des données
data2<-read.table(file=file.choose(),header=T,sep=",",dec=".")
View(data2)
```

Dimensions of the Dataset

```
> print(paste("Dimensions (lignes, colonnes):", paste(dim_data, collapse=" x ")))
[1] "Dimensions (lignes, colonnes): 10000 x 14"
```

Display the Data Type of Each Column

```
> # Display the Data Type of Each Column
> sapply(data2, class)
                     UDI               Product.ID                     Type
               "integer"              "character"              "character"
   Air.temperature..K. Process.temperature..K. Rotational.speed..rpm.
               "numeric"              "numeric"              "integer"
          Torque..Nm.         Tool.wear..min.         Machine.failure
               "numeric"              "integer"              "integer"
                   TWF                     HDF                     PWF
               "integer"              "integer"              "integer"
                   OSF                     RNF
               "integer"              "integer"
```

Variables in the Dataset

Summary Statistics of the Dataset

```
> summary(data2)
      UDI         Product.ID            Type           Air.temperature..K.
 Min.   :    1   Length:10000       Length:10000       Min.   :295.3
 1st Qu.: 2501   Class :character   Class :character   1st Qu.:298.3
 Median : 5000   Mode  :character   Mode  :character   Median :300.1
 Mean   : 5000                                         Mean   :300.0
 3rd Qu.: 7500                                         3rd Qu.:301.5
 Max.   :10000                                         Max.   :304.5
 Process.temperature..K. Rotational.speed..rpm.  Torque..Nm.    Tool.wear..min.
 Min.   :305.7           Min.   :1168           Min.   : 3.80   Min.   :  0
 1st Qu.:308.8           1st Qu.:1423           1st Qu.:33.20   1st Qu.: 53
 Median :310.1           Median :1503           Median :40.10   Median :108
 Mean   :310.0           Mean   :1539           Mean   :39.99   Mean   :108
 3rd Qu.:311.1           3rd Qu.:1612           3rd Qu.:46.80   3rd Qu.:162
 Max.   :313.8           Max.   :2886           Max.   :76.60   Max.   :253
 Machine.failure       TWF               HDF               PWF               OSF
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000
 Median :0.0000   Median :0.0000   Median :0.0000   Median :0.0000   Median :0.0000
 Mean   :0.0339   Mean   :0.0046   Mean   :0.0115   Mean   :0.0095   Mean   :0.0098
 3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.0000
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
      RNF
 Min.   :0.0000
 1st Qu.:0.0000
 Median :0.0000
 Mean   :0.0019
 3rd Qu.:0.0000
 Max.   :1.0000
>
```

Data Quality Check

Objective: Ensure data integrity before proceeding with the analysis.

Check	Result
Missing Values	None
Duplicated Rows	None

```
> print("Missing Values Per Column:")
[1] "Missing Values Per Column:"
> print(missing_values)
                     UDI               Product.ID                     Type      Air.temperature..K.
                       0                        0                        0                        0
Process.temperature..K.   Rotational.speed..rpm.            Torque..Nm.         Tool.wear..min.
                       0                        0                        0                        0
         Machine.failure                      TWF                      HDF                      PWF
                       0                        0                        0                        0
                     OSF                      RNF
                       0                        0
> # Check for duplicated rows
> duplicated_rows <- ai4i2020[duplicated(ai4i2020), ]
> print(paste("Number of duplicated rows:", nrow(duplicated_rows)))
[1] "Number of duplicated rows: 0"
```

Variables in the Dataset

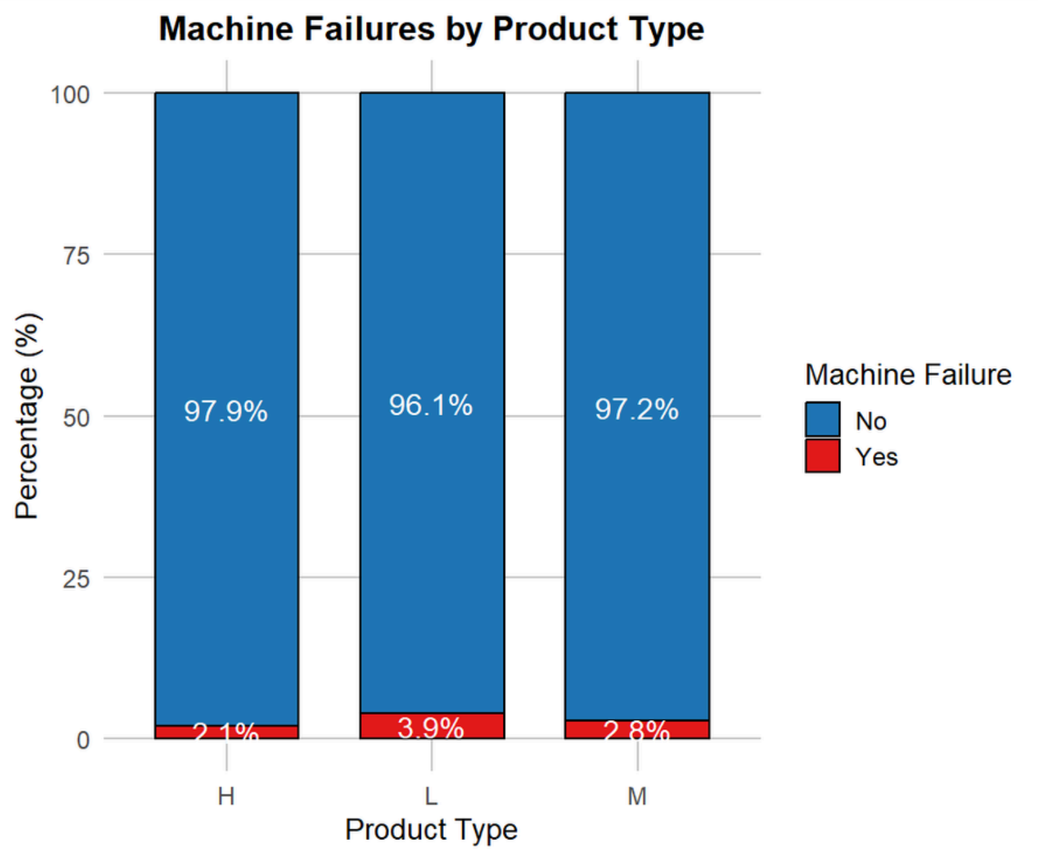
Data transformation

```
data <- data2 %>% select(Type, Machine.failure)
# Convert Machine.failure to a factor for analysis
data$Machine.failure <- factor(data$Machine.failure, labels = c("No", "Yes"))
```

Converting Machine.failure to a factor allows this variable to be treated as a categorical variable for statistical tests or predictive models.

The result of this operation is assigned to a new variable called data. This new dataset will contain only the columns Type and Machine.failure.

Crosstabulation of Product Type and Machine Failure



Crosstabulation is a method used to explore the relationship between two or more categorical variables. It presents the joint distribution of the variables in a table format

==>The majority of machines do not fail, regardless of product type.

Next we will explore 4 non-parametric tests:

Chi-square test: Examining the relationship between product quality and machine failure.

ANOVA test: Analyzing the impact of Rotational speed on product quality.

T-test: Investigating the relationship between machine failure and Rotational speed

T-test: Assessing the effect of rotation speed on TWF.

Chi-Squared Test & Cramér's V Analysis

The Chi-Squared Test and Cramér's V are statistical tools used to analyze categorical data and determine if there is an association between variables.

- **Chi-Squared Test** helps determine if there is an association between categorical variables.
- **Cramér's V** quantifies the strength of the association.

==>Both tests are essential in the analysis of categorical data and are often used together to **not only determine if an association exists but also how strong that association is.**

Chi-Squared Test & Cramér's V Analysis

```
> # Chi-square test
> chi_test <- chisq.test(crosstab)
> # Print results
> print(chi_test)

        Pearson's Chi-squared test

data:  crosstab
X-squared = 13.752, df = 2, p-value = 0.001032
```

The **p-value is less than 0.05**, which indicates a **statistically significant relationship between the variables.**

We **reject the null hypothesis(no association)** and conclude that:

- Product Type (Type) and Machine Failure (Machine.failure) are not independent.
- Machine failure distribution varies depending on the product type.

Chi-Squared Test & Cramér's V Analysis

```
> library(lsr)
> cramersV(crosstab)
[1] 0.03708331
>
```

Range: 0 (no association) to 1 (strong association).

- A value of **0.0371** suggests a **weak association**.

==> **While the relationship is statistically significant, the effect is weak**

ANOVA TEST

1. Normality Test Results

Objective: Assess if rotational speed follows a normal distribution within each product type (H, L, M).

```
> print("Normality Test Results:")
[1] "Normality Test Results:"
> print(normality_results)
# A tibble: 3 × 5
  Type  Kolmogorov_Statistic Kolmogorov_p_value Shapiro_Statistic Shapiro_p_value
  <chr>                <dbl>              <dbl>             <dbl>           <dbl>
1 H                    0.110          6.60e-11             0.863        1.15e-28
2 L                    0.107          6.93e-60            NA           NA
3 M                    0.101          8.97e-27             0.871        2.33e-44
>
```

==> For all three types (H, L, M), rotational speed does not follow a normal distribution, as shown by both Kolmogorov-Smirnov and Shapiro-Wilk tests .

==> we can not use ANOVA .

ANOVA TEST

2. Kruskal-Wallis Test Results

Objective: Compare the median rotational speeds across product types (H, L, M), because that the data is not normally distributed.

```
> M                          0.101                8.97e-27                  0.
> print(kruskal_test)

        Kruskal-Wallis rank sum test

data:  Rotational.speed..rpm. by Type
Kruskal-Wallis chi-squared = 0.2479, df = 2, p-value = 0.8834
```

Since p_value = 0.8834 > 0.05, we fail to reject the null hypothesis (H0).
This means there is no statistically significant difference in rotational speed across the three product types (H, L, M).

T-test

t-test compares the means of two groups (with unequal variances) and tests whether there is a significant difference between them.

In the next slides we will conduct the following analyses:

1. **Normality Tests:** Using the Kolmogorov-Smirnov and Shapiro-Wilk tests to assess the distribution of data.
2. **Levene's Test:** Checking for equality of variances across groups.
3. **T-Test** for Equality of Means: Applying Welch's test to compare group means .

First t-test

1. Normality Test Results

As demonstrated in the previous slides, rotational speed does not follow a normal distribution for any of the three types (H, L, M), as confirmed by both the Kolmogorov-Smirnov and Shapiro-Wilk tests.

2.Equality of Variances)

```
> cat("Levene's Test Results for Equality of Variances:\n")
Levene's Test Results for Equality of Variances:
> print(levene_test)
Levene's Test for Homogeneity of Variance (center = median)
        Df F value    Pr(>F)
group    1  60.901 6.603e-15 ***
      9998
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

The p-value < 0.001, so we reject H0 that the variances are equal across the groups.

Therefore, the assumption of equal variances is not met.

==> we accept H1

First t-test

3. t-Test Results

```
Signif. codes:  0       0.001       0.01       0.05  .  0.1     1
> # Print t-Test Results
> cat("t-Test Results for Equality of Means (Welch's Test):\n")
t-Test Results for Equality of Means (Welch's Test):
> print(t_test_result)

        Welch Two Sample t-test

data:  Rotational.speed..rpm. by Machine.failure
t = 2.0868, df = 342.5, p-value = 0.03765
alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
95 percent confidence interval:
  2.514282 85.032295
sample estimates:
mean in group 0 mean in group 1
       1540.260        1496.487
```

p-value (0.03765) is less than 0.05, which means there is a statistically significant difference in the mean rotational speeds between the two groups.

Group 0 (No machine failure) has a slightly higher mean rotational speed than Group 1 (With machine failure).

Second t-test

1. Normality Test Results

```
> # Print the normality results
> print(normality_results)
# A tibble: 2 × 5
  TWF   Kolmogorov_Statistic Kolmogorov_p_value Shapiro_Statistic Shapiro_p_value
  <fct>                 <dbl>              <dbl>             <dbl>           <dbl>
1 0                     0.104           1.18e-93             0.872        4.73e-54
2 1                     0.156           2.13e- 1             0.852        3.43e- 5
```

1. **test Kolmogorov-Smirnov:**

For Group "No", the data does not follow a normal distribution (p-value < 0.05).

For Group "Yes", the data may follow a normal distribution, but this is not strong evidence (p-value > 0.05).

2. **Shapiro-Wilk:**

Both groups (Group "No" and Group "Yes") do not follow a normal distribution (p-values < 0.05).

Second t-test

2.Test for Equality of Variances

```
Levene's Test for Equality of Variances:
> print(levene_test)
Levene's Test for Homogeneity of Variance (center = median)
        Df F value Pr(>F)
group    1  0.8785 0.3486
      9998
```

p-value> 0.05

we accept Ho

=> There is no significant difference in variance between the two groups

Second t-test

3. t-Test Results

```
> # Print t-test results
> cat("\nt-Test for Equality of Means (Welch's Test):\n")

t-Test for Equality of Means (Welch's Test):
> print(t_test_result)

        Welch Two Sample t-test

data:  Rotational.speed..rpm. by TWF
t = -0.90565, df = 45.316, p-value = 0.3699
alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
95 percent confidence interval:
 -88.72530  33.67644
sample estimates:
mean in group 0 mean in group 1
       1538.649        1566.174
```

The means of the two groups are quite close to each other, and the difference is not statistically significant, as we saw from the p-value> 0.05.

The t-test results indicate that there is no statistically significant difference in the mean rotational speed between the two groups: Group "No" (TWF = 0) and Group "Yes" (TWF = 1).

Conclusion

By using tests like the Chi-square test, ANOVA, and T-tests, we gained valuable insights into the effects of rotational speed, product quality, and machine performance. We also conclude that the removal of outliers significantly improved data quality, thereby enhancing the reliability and robustness of our results.

Thank you