VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF APPLIED SCIENCE

PROBABILITY & STATISTICS (MT2013)

Assignment (Semester: 231)

Predicting Intel's CPU Clock Speed Using Statistical Methods

Advisor: Phan Thị Khánh Vân, FAS - HCMUT

Students: Lê Nguyễn Gia Bảo - 2210216.
Trần Đình Đăng Khoa - 2211649.
Bùi Vũ Thiên Đăng - 2252151.
Trần Tuấn Minh Khoa - 2252365.
Nguyễn Hữu Trí - 2252842.

HO CHI MINH CITY, APRIL 2024

Contents

1 Abstract

In the rapidly advancing field of computer hardware technology, understanding and predicting the clock speed (frequency) of central processing units (CPUs) is crucial for both manufacturers and consumers. This project, **"Predicting Intel CPU Clock Speed Using Statistical Methods"**, aims to develop robust predictive models for CPU clock speed based on detailed specifications of Intel CPUs. By employing statistical techniques such as **Linear Regression** and **Analysis of Variance (ANOVA)**, this study seeks to identify the key features that significantly influence CPU clock speed.

The dataset utilized in this study comprises a comprehensive collection of Intel CPU specifications, including attributes such as the number of cores, number of threads, cache size, power consumption, and various architectural details. Data preprocessing steps involve handling missing values, normalizing data, and encoding categorical variables to ensure the dataset is suitable for rigorous statistical analysis.

To identify significant predictors of CPU clock speed, ANOVA is used to assess the impact of categorical variables, providing insights into how different CPU series and generations affect clock speeds. Linear regression is then employed to model and predict CPU clock speed based on these significant features. This method directly establishes the relationship between the dependent variable (clock speed) and the independent variables (CPU specifications).

The predictive modeling component of this project primarily relies on linear regression techniques. Linear regression provides a foundational understanding of the linear relationships between the predictors and CPU clock speed. The performance of the regression models is evaluated using key metrics such as R-squared, Mean Squared Error (MSE), and visual inspection of residual plots.

Our analysis reveals that features such as the number of cores, number of threads, cache size, and power consumption are significant determinants of CPU clock speed. The linear regression model offers valuable insights into the impact of these features on clock speed, allowing for accurate predictions based on the given specifications. While ANOVA provides supplementary information on the influence of categorical variables, it is the linear regression model that forms the core of our predictive analysis.

This project contributes to the broader understanding of CPU performance dynamics, providing a methodological framework that can be applied to other hardware components or similar predictive tasks. The findings have practical implications for manufacturers in optimizing CPU design and for consumers in making informed purchasing decisions. By leveraging statistical methods and regression analysis, this study offers a data-driven approach to predicting CPU clock speed, enhancing transparency and efficiency in the marketplace.

2 Introduction

The Central Processing Unit (CPU) is often referred to as the "brain" of the computer due to its fundamental role in executing instructions and managing the operations of other components. It processes data, performs calculations, and manages tasks, making it a critical component that directly impacts a computer's performance and efficiency. As technology continues to advance rapidly, the variety and complexity of CPUs available in the market have also increased, necessitating more sophisticated methods to evaluate and predict their clock speed (frequency).

Predicting CPU clock speed accurately is crucial for several reasons. For manufacturers, understanding the factors that influence CPU clock speed can aid in optimizing CPU design, enhancing product development, and targeting the right market segments. Accurate clock speed predictions can help manufacturers maintain a balance between performance and cost-effectiveness. For consumers, knowledge of CPU performance dynamics enables informed purchasing decisions, ensuring that they obtain the best value for their money. This is particularly important given the diverse range of CPUs available, each with different specifications and performance levels.

This report focuses on the analysis of CPU specifications to predict clock speed using a variety of statistical methods. The dataset used in this study consists of detailed specifications of Intel CPUs, one of the leading CPU manufacturers in the world. Intel CPUs are widely used in various computing devices, from desktops and laptops to servers and workstations, making them an ideal subject for this study.

The primary goal of this report is to identify the key features that significantly influence CPU clock speed and to develop predictive models that can accurately estimate clock speeds based on the identified features. To achieve this, we employ several statistical techniques, including Analysis of Variance (ANOVA) and regression analysis. Each of these methods plays a vital role in understanding the relationships between different CPU specifications and their corresponding clock speeds.

Analysis of Variance (ANOVA) is utilized to determine the impact of categorical variables on CPU clock speed. By comparing means across different groups, ANOVA helps identify whether certain categorical features, such as CPU series or generation, significantly affect clock speed. This analysis provides supplementary insights that enhance our understanding of how different CPU models and architectures impact performance.

The regression analysis forms the core of our predictive modeling approach. Linear regression is applied to provide a foundational understanding of the linear relationships between the selected features and CPU clock speed. Given the complexity of CPU performance, we also consider polynomial regression to capture potential non-linear interactions among the variables. By comparing the performance metrics of these models, such as R-squared and Mean Squared Error (MSE), we can determine the most effective model for predicting CPU clock speed.

Data preprocessing is a crucial step in this analysis, ensuring the reliability and accuracy of

our models. This involves handling missing values, normalizing data, and encoding categorical variables. Proper preprocessing enhances the quality of the dataset, making it suitable for rigorous statistical analysis and modeling.

In summary, this report aims to provide a comprehensive analysis of Intel CPU specifications to predict their clock speeds. By leveraging statistical methods and regression models, we seek to develop robust predictive models that can assist manufacturers in optimizing CPU design and help consumers make informed purchasing decisions. The findings of this study will contribute to the broader understanding of CPU performance dynamics and demonstrate the value of combining various statistical techniques to create accurate and reliable clock speed predictions.

3 Background Knowledge on Statistical Methods

3.1 Hypothesis Testing

Hypothesis testing is a fundamental statistical procedure used to make inferences about population parameters based on sample data. It is essential in various fields, including scientific research, quality control, and decision-making processes. The primary objective of hypothesis testing is to evaluate the plausibility of a specific claim or hypothesis concerning a population parameter, such as the mean or variance, which is crucial for understanding CPU clock speed determinants.

In hypothesis testing, two mutually exclusive hypotheses are formulated: the null hypothesis (H_0) and the alternative hypothesis (H_a). The null hypothesis typically represents the status quo, the baseline assumption, or the claim that the researcher wishes to test against. The alternative hypothesis represents the opposite or the alternative claim that the researcher aims to support or conclude if the null hypothesis is rejected.

The process of hypothesis testing involves the following steps:

1. Formulate the null hypothesis (H_0) and the alternative hypothesis (H_a).
2. Specify the significance level (α), which is the probability of rejecting the null hypothesis when it is true (Type I error).
3. Calculate the test statistic from the sample data.
4. Determine the critical region or the critical value(s) based on the significance level and the chosen test.
5. Compare the test statistic with the critical region or critical value(s).
6. Make a decision: Reject or fail to reject the null hypothesis.

The decision to reject or fail to reject the null hypothesis is based on the comparison between the test statistic and the critical region or critical value(s). If the test statistic falls within the critical region, the null hypothesis is rejected in favor of the alternative hypothesis. If the test statistic does not fall within the critical region, the null hypothesis is not rejected.

Types of hypothesis tests relevant to this project include:

- Tests for Means: Comparing the mean clock speeds of different CPU series or generations.
- Tests for Correlation and Regression Coefficients: Assessing the relationship between CPU specifications (e.g., number of cores) and clock speeds.

Hypothesis testing is subject to two types of errors: Type I error (rejecting the null hypothesis when it is true) and Type II error (failing to reject the null hypothesis when it is false). The significance level (α) controls the probability of committing a Type I error, while the power of the test ($1 - \beta$) represents the probability of correctly rejecting the null hypothesis when it is false, where β is the probability of committing a Type II error.

3.2 Analysis of Variance (ANOVA)

Analysis of Variance (ANOVA) is a statistical method used to determine if there are statistically significant differences between the means of three or more independent groups. For this project, ANOVA helps us understand the impact of categorical variables, such as CPU series and generation, on CPU clock speeds by comparing the average clock speeds across different groups.

Key Concepts:

- Hypotheses:
  - Null Hypothesis (H_0): Assumes that there are no differences in the mean CPU clock speeds among the different groups.
  - Alternative Hypothesis (H_a): Assumes that at least one group mean is different from the others.
- Between-Group Variability: Measures the variation in CPU clock speeds between different groups, reflecting the effect of the categorical variable on clock speeds.
- Within-Group Variability: Measures the variation in CPU clock speeds within each group, reflecting natural clock speed variations among CPUs of the same group.
- F-Statistic: The ratio of between-group variability to within-group variability. A higher F-statistic indicates a greater likelihood that the observed differences between group means are statistically significant.
- P-Value: The probability of observing the data assuming the null hypothesis is true. A low p-value (typically < 0.05) suggests that the differences between group means are statistically significant.

Procedure for Conducting ANOVA:

- Categorize Data: Group the dataset based on categorical variables such as CPU Series (e.g., Intel Core i3, i5, i7) and CPU Generation (e.g., 9th Gen, 10th Gen).
- Calculate Group Means: Determine the mean CPU clock speed for each group.
- Compute Sum of Squares:
  - Total Sum of Squares (SST): Measures the total variation in CPU clock speeds.
  - Between-Group Sum of Squares (SSB): Measures the variation in CPU clock speeds between different groups.
  - Within-Group Sum of Squares (SSW): Measures the variation in CPU clock speeds within each group.
- Calculate Mean Squares:
  - Mean Square Between (MSB): SSB divided by the degrees of freedom between groups.

  - Mean Square Within (MSW): SSW divided by the degrees of freedom within groups.

- Compute F-Statistic.
- Determine P-Value: Using statistical software or F-distribution tables, find the p-value corresponding to the calculated F-statistic.
- Post-Hoc Analysis (if necessary): If ANOVA indicates significant differences, conduct post-hoc tests to identify which specific groups differ from each other.

Application in This Project:

- Group Data by CPU Series: Calculate the mean clock speed for each series.
- Perform ANOVA Test for CPU Series: Formulate hypotheses, conduct ANOVA, calculate F-statistic and p-value, and interpret results.
- Group Data by CPU Generation: Calculate the mean clock speed for each generation.
- Perform ANOVA Test for CPU Generation: Formulate hypotheses, conduct ANOVA, calculate F-statistic and p-value, and interpret results.

By using ANOVA, we can systematically evaluate the impact of these categorical variables on CPU clock speeds, providing valuable insights into performance trends and refining our predictive models.

3.2.1 Formulas (for reference)

- The total sum of squares: $\mathbf{SST} = \sum_{i=1}^{k}\sum_{j=1}^{n}(X_{ij} - \bar{X})^2 = \sum_{i,j} X_{ij}^2 - \frac{X^2}{N}$.
- The treatment sum of squares: $\mathbf{SSTr} = \sum_{i=1}^{k} n(\bar{X}_i - \bar{X})^2 = \sum_{i=1}^{k} \frac{X_i^2}{n} - \frac{X^2}{N}$.
- The error sum of squares: **SSE = SST - SSTr**.
- Treatment degree of freedom: $df(\mathbf{SSTr}) = k - 1$. Error degree of freedom $df(\mathbf{SSE}) = N - k = nk - k$.
- The mean square for treatment: $\mathbf{MSTr} = \frac{\mathbf{SSTr}}{k-1}$.
- The mean square for error: $\mathbf{MSE} = \frac{\mathbf{SSE}}{nk-k}$.
- If H_0 is true, then the statistic $F = \frac{\mathbf{MSTr}}{\mathbf{MSE}} \sim F_{k-1,nk-k}$: Fisher random variable. If $F > F_{\alpha,k-1,nk-k}$ we reject H_0.

3.3 Linear Regression

Linear regression is a fundamental statistical technique used to model the relationship between a dependent variable (CPU clock speed) and one or more independent variables (CPU specifications). It is widely employed to analyze and make predictions based on observed data.

Objective: To find the best-fitting straight line that describes the relationship between CPU clock speed and its specifications. This line is represented by a linear equation, which takes the following form:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n + \varepsilon$$

Where:

- Y is the dependent variable (CPU clock speed)
- β_0 is the intercept (the value of Y when all independent variables are zero)
- $\beta_1, \beta_2, ..., \beta_n$ are the coefficients (slopes) associated with the respective independent variables
- $x_1, x_2, ..., x_n$ are the independent variables (CPU specifications)
- ε is the error term, representing the difference between the observed values and the predicted values

The process of linear regression involves estimating the values of the coefficients (β_0, β_1, β_2, ..., β_n) using a set of observed data points. This estimation is typically performed using the method of least squares, which aims to minimize the sum of squared differences between the observed values and the predicted values obtained from the linear equation. Linear regression models can be classified into two main types:

- Simple Linear Regression: This model involves only one independent variable and is represented by the equation $Y = \beta_0 + \beta_1 x + \varepsilon$.
- Multiple Linear Regression: This model involves two or more independent variables and is represented by the equation $Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n + \varepsilon$

Procedure for Linear Regression:

1. Formulate the Model: Define the relationship between CPU clock speed and its specifications.
2. Estimate Coefficients: Use the method of least squares to estimate the values of the coefficients ($\beta_1, \beta_2, ..., \beta_n$).
3. Evaluate the Model:
    - R-Squared: Measures the proportion of variance in the dependent variable explained by the independent variables.

- P-Values: Assess the significance of each coefficient.
- Residual Analysis: Check for patterns in the residuals to validate assumptions.

4. Interpret Coefficients: Understand the magnitude and direction of the impact of each independent variable on CPU clock speed. For instance, a positive coefficient for the number of cores indicates that as the number of cores increases, the clock speed tends to increase.

5. Predict: Use the model to predict CPU clock speeds based on new values of the independent variables. The predicted values can guide decisions in CPU design and marketing strategies.

Assumptions of Linear Regression:

- Linearity: The relationship between the dependent and independent variables is linear.
- Normality of Residuals: The residuals (errors) are normally distributed.
- Homoscedasticity: Constant variance of residuals.
- Independence: Observations are independent of each other.

By understanding and applying linear regression, we can develop robust models to predict CPU clock speeds based on their specifications, providing valuable insights for manufacturers and consumers.

3.3.1 Formulas (for reference)

Covariance and Correlation

- Covariance between the random variables X and Y is:

$$\text{cov}(X,Y) = \sigma_{XY} = E[(X - E(X))(Y - E(Y))] = E(XY) - E(X)E(Y)$$

- The correlation between the random variables X and Y is:

$$\rho_{XY} = \frac{\text{cov}(X,Y)}{\sqrt{V(X)V(Y)}} = \frac{\sigma_{XY}}{\sigma_X \sigma_Y} = \frac{S_{xy}}{\sqrt{S_{xx} \cdot S_{yy}}}, \ -1 \leq \rho_{XY} \leq 1$$

Least Square Method

A statistical procedure to find the best fit for a set of data points.
Suppose that we have n pairs of observations $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$. We have to find the linear regression model for the data as $y_i = \beta_0 + \beta_1 x_i + \varepsilon$
The sum of the squares of the deviations of the observations from the true regression line is

$$L = \sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_1) \rightarrow min$$

The least squares estimators $\hat{\beta}_0, \hat{\beta}_1$ must satisfy

$$\begin{cases} n\hat{\beta}_0 + \hat{\beta}_1 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i \\ \hat{\beta}_0 \sum_{i=1}^{n} x_i + \hat{\beta}_1 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i \end{cases}$$

The least squares estimates of the intercept and the slope in the simple linear regression model are

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}, \quad \hat{\beta}_1 = \frac{S_{xy}}{S_{xx}}$$

where

$$S_{xy} = \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) = \left(\sum_{i=1}^{n} x_i y_i \right) - \frac{1}{n} \left(\sum_{i=1}^{n} x_i \right) \left(\sum_{i=1}^{n} y_i \right)$$

$$S_{xx} = \sum_{i=1}^{n} (x_i - \bar{x})^2 = \left(\sum_{i=1}^{n} x_i^2 \right) - \frac{1}{n} \left(\sum_{i=1}^{n} x_i \right)^2$$

The **fitted** or **estimated regression line** is therefore:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$

- $e_i = y_i - \hat{y}_i$: The error in the fit of the model to the i^{th} observation y_i and is called **residual**.
- $\text{SSE} = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$: The sum of squares of the residuals.
- $\text{SST} = \sum_{i=1}^{n} (y_i - \bar{y})^2 = \sum_{i=1}^{n} y_i^2 - n\bar{y}^2 = S_{yy}$: The total sum of square of the response variable.
- $\text{SSR} = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 = \beta_1 S_{xy}$: The sum of squares for regression.
- $r^2 = 1 - \frac{SSE}{SST} = \rho_{XY}^2$: The coefficient of determination.

Estimator of Variance: $\hat{\sigma}^2 = \frac{SSE}{n-2} = \frac{SST - \beta_1 S_{xy}}{n-2}$

3.4 Remark

To predict the clock speed of CPUs, we primarily rely on **Linear Regression** as it directly models the relationship between the CPU specifications (independent variables) and the clock speed (dependent variable). **ANOVA** can be used as a supplementary method to understand the influence of categorical variables on CPU clock speeds, but it is not essential for the prediction model itself.

4 Data Pre-processing

The dataset utilized in this study is derived from a CSV file containing comprehensive specifications of Intel CPUs. This dataset serves as the foundation for our analysis, providing detailed information on various CPU features that are essential for understanding clock speed trends. In this section, we describe the dataset's structure, contents, and the relevance of its attributes to our study.

4.1 Dataset Overview

The CSV file comprises numerous rows, each representing an individual Intel CPU model. Each row contains multiple attributes describing the specifications and characteristics of the CPU. The dataset encompasses a diverse range of CPU models across different series, generations, and intended applications (e.g., consumer desktops, laptops, server-grade CPUs), ensuring a comprehensive coverage of CPU types and their respective attributes.

4.2 Data Relevance and Usefulness

The following attributes from the dataset are particularly relevant for analyzing CPU clock speeds:

- **Processor_Base_Frequency**: This attribute represents the base clock speed of the CPU, measured in gigahertz (GHz). Higher base frequencies generally indicate faster processing capabilities and overall performance.
- **nb_of_Cores**: The number of cores in a CPU impacts its ability to handle multiple tasks simultaneously. CPUs with more cores can potentially support higher clock speeds under optimal conditions, affecting overall performance trends.
- **nb_of_Threads**: This attribute represents the number of threads supported by the CPU, which is often influenced by technologies like Intel's Hyper-Threading. A higher number of threads can contribute to better resource utilization and potentially higher effective clock speeds for certain workloads.
- **TDP (Thermal Design Power)**: The TDP indicates the maximum amount of heat the CPU cooling system needs to dissipate. This attribute is closely related to the CPU's power consumption and thermal characteristics, which can impact clock speed potential and performance.
- **Lithography**: This attribute refers to the manufacturing process node (e.g., 14nm, 10nm) used in producing the CPU. Advancements in manufacturing processes can lead to higher clock speeds and improved efficiency in newer CPU generations.

Our objective is to analyze historical trends in CPU clock speeds using these attributes. Understanding how clock speeds have evolved across different CPU generations, architectural improvements, and technological shifts is crucial for identifying patterns and making informed predictions. By focusing on these critical attributes, we aim to uncover insights into the factors

driving CPU clock speed improvements and contributing to the overall technological progress in CPU development.

4.3 Load Data

In this section, we load and clean the data for our analysis.

```
# Load necessary library
library (dplyr)

# Importing data
intel_cpu <- read.csv ("~/Downloads/Intel_CPUs.csv")

# Specify the desired columns
desired_cols <- c ("Processor_Base_Frequency", "nb_of_Cores", "nb_of_
Threads", "TDP", "Lithography")

# Subset the data to only include the desired columns
intel_cpu_subset <- intel_cpu %>% select (one_of(desired_cols))

# Remove rows with base frequency measured in MHz
intel_cpu_subset <- intel_cpu_subset %>%
    filter (!grepl ("MHz", Processor_Base_Frequency))

# Convert necessary columns to appropriate types if they are not
# Removing 'GHz' and converting Processor_Base_Frequency to numeric
intel_cpu_subset$Processor_Base_Frequency <-
    as.numeric (gsub (" GHz", "", intel_cpu_subset$Processor_Base_
Frequency))

# Additional conversion to ensure nb_of_Cores, nb_of_Threads, TDP, and
Lithography are numeric where applicable
intel_cpu_subset$nb_of_Cores <-
    as.numeric (intel_cpu_subset$nb_of_Cores)
intel_cpu_subset$nb_of_Threads <-
    as.numeric (intel_cpu_subset$nb_of_Threads)
intel_cpu_subset$TDP <-
    as.numeric (gsub (" W", "", intel_cpu_subset$TDP))
intel_cpu_subset$Lithography <-
    as.numeric (gsub (" nm", "", intel_cpu_subset$Lithography))
```

4.4 Handling missing values

This dataset, like others, is not perfect, as some rows of values are empty of values. We must decide on how to treat empty/missing values, the process of which is called imputation.
Some common imputation methods are:

- **Listwise deletion**: Remove whole rows/list. Also known as complete case deletion.
- **Median imputation**: Fill with median of available values.

- **Mean imputation**: Fill with mean of available values.
- **Regression imputation**: Use linear regression to extrapolate missing values.

Some less common imputation methods are:

- **Pairwise deletion**: Similar to listwise deletion but only remove when missing required variables.
- **Hot-deck imputation**: Pick randomly from available values.
- **Cold-deck imputation**: Pick from a donor dataset.

We utilized various imputation methods to preprocess data.

First, we reselect among the originally selected data columns so as to minimize missing data. So far, we have selected 6 columns as explained in the above section. After this consideration, we have removed the column Max_Turbo_Frequency, due to it having a very limited amount of data points especially in correspondence to older CPU processors.

Then, we employed listwise deletion to remove cases of missing or bad data. We could have chosen other similar methods like mean imputation, median imputation or regression imputation, and in truth these methods have their own pros and cons. What led us to choose listwise were based on context:

- We wanted to quickly prototype our programs. Implementation of listwise deletion is quick and simple.
- We wanted a neutral algorithm. Unlike methods like regression imputation, this does not introduce sample bias.

In contrast and for example, here's why using median imputation is not suitable for our attributes and why removing rows with missing values is a better approach:

- Processor_Base_Frequency:
  - Contextual Integrity: The base frequency is a critical performance metric measured in GHz. Imputing a median value may introduce inaccuracies because the actual base frequencies are often specific to CPU models and designs. These values are precise and need to reflect the true performance characteristics of the CPU.
  - Range and Units: Base frequencies vary within specific ranges, and a median value might not accurately represent the operational characteristics of the CPUs, especially when considering variations in architecture and generation.
- nb_of_Cores and nb_of_Threads:
  - Specific Configuration: The number of cores and threads is an intrinsic property of a CPU model, determined by its design and intended use. These are exact numbers that are integral to the CPU's capability. Imputing a median would mean assigning an arbitrary number of cores or threads that do not exist in the actual hardware configurations, leading to misleading results.

  - Consistency: Each CPU model has a specific number of cores and threads. Removing rows with missing values ensures we only analyze data with complete and accurate configurations.

- TDP (Thermal Design Power):
  - Thermal Characteristics: TDP is a precise measure of the maximum heat a CPU can generate under typical load, directly influencing cooling solutions and performance. Using a median TDP might not accurately reflect the thermal design considerations specific to different CPU models.
  - Power Management: Different CPUs have different power and thermal characteristics. It's crucial to work with exact TDP values to maintain the accuracy of the analysis.

- Lithography:
  - Manufacturing Process: Lithography measures the process technology node (e.g., 14nm, 10nm). These are fixed and precise values associated with the manufacturing process of the CPU. A median value would not accurately represent any actual process node and could distort the analysis of technological trends.

```
# Remove rows with missing values in the specific columns
cols_to_check <- c ("Processor_Base_Frequency", "nb_of_Cores", "nb_of_
Threads", "TDP", "Lithography")
intel_cpu_subset <- intel_cpu_subset[complete.cases (intel_cpu_subset[,
cols_to_check]), ]

# Display the resulting subset
print (intel_cpu_subset)
```

4.5 Handling outliers

Outliers in our dataset can have both positive and negative implications. Outliers can bring out the true nature of our sample population. Outliers can also mean experimental errors, affecting the integrity and accuracy of our data analysis. In this assignment we chose to preprocess outlier values. Specifically, our initial observations have shown that our dataset is skewed. Thus, we will specifically employ log transformation to adjust the datapoints at certain columns.

Log transformation simply means applying a base-10 logarithm function to a column, specifically TDP (Thermal Design Power):

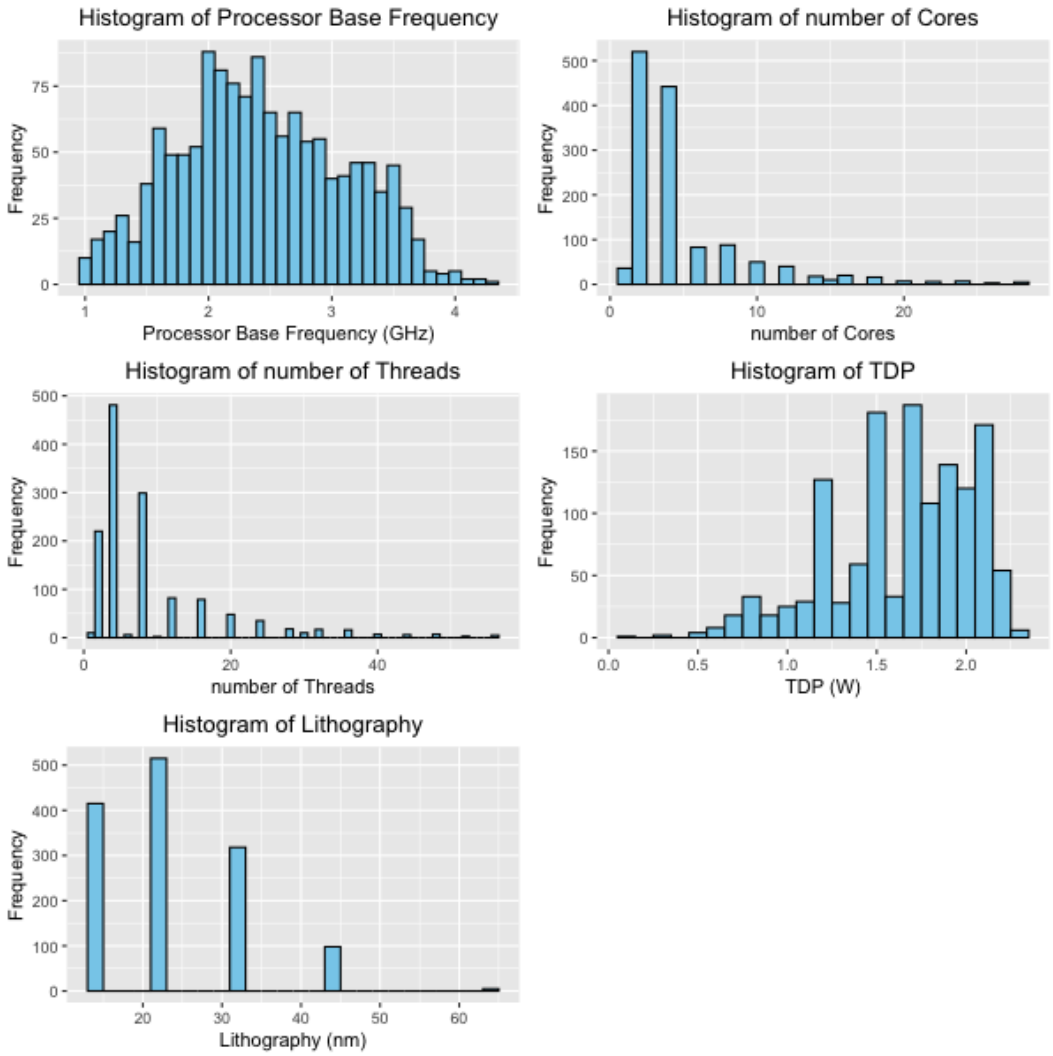
```
intel_cpu_temp <- intel_cpu_subset
intel_cpu_temp$TDP <- log10(intel_cpu_temp$TDP)
```

5 Descriptive Statistic

5.1 Summary Statistics

```
 Product_Collection Vertical_Segment   Processor_Number      Status          Launch_Date        Lithography
 Length:2283        Length:2283        Length:2283        Length:2283        Length:2283        Length:2283
 Class :character   Class :character   Class :character   Class :character   Class :character   Class :character
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character



 Recommended_Customer_Price  nb_of_Cores     nb_of_Threads    Processor_Base_Frequency Max_Turbo_Frequency
 Length:2283                Min.   : 1.000   Min.   : 1.000   Length:2283              Length:2283
 Class :character           1st Qu.: 1.000   1st Qu.: 4.000   Class :character         Class :character
 Mode  :character           Median : 2.000   Median : 4.000   Mode  :character         Mode  :character
                            Mean   : 4.067   Mean   : 8.728
                            3rd Qu.: 4.000   3rd Qu.: 8.000
                            Max.   :72.000   Max.   :56.000
                                             NA's   :856
    Cache            Bus_Speed             TDP          Embedded_Options_Available Conflict_Free
 Length:2283        Length:2283        Length:2283        Length:2283                Length:2283
 Class :character   Class :character   Class :character   Class :character           Class :character
 Mode  :character   Mode  :character   Mode  :character   Mode  :character           Mode  :character



 Max_Memory_Size    Memory_Types       Max_nb_of_Memory_Channels Max_Memory_Bandwidth ECC_Memory_Supported
 Length:2283        Length:2283        Min.   : 1.000            Length:2283          Length:2283
 Class :character   Class :character   1st Qu.: 2.000            Class :character     Class :character
 Mode  :character   Mode  :character   Median : 2.000            Mode  :character     Mode  :character
                                       Mean   : 2.615
                                       3rd Qu.: 3.000
                                       Max.   :16.000
                                       NA's   :869
 Processor_Graphics_ Graphics_Base_Frequency Graphics_Max_Dynamic_Frequency Graphics_Video_Max_Memory Graphics_Output
 Mode:logical        Length:2283             Length:2283                    Length:2283               Length:2283
 NA's:2283           Class :character        Class :character               Class :character          Class :character
                     Mode  :character        Mode  :character               Mode  :character          Mode  :character



 Support_4k     Max_Resolution_HDMI Max_Resolution_DP  Max_Resolution_eDP_Integrated_Flat_Panel DirectX_Support
 Mode:logical   Length:2283         Length:2283        Length:2283                               Length:2283
 NA's:2283      Class :character    Class :character   Class :character                          Class :character
                Mode  :character    Mode  :character   Mode  :character                          Mode  :character



 OpenGL_Support PCI_Express_Revision PCI_Express_Configurations_ Max_nb_of_PCI_Express_Lanes      T
 Mode:logical   Length:2283          Length:2283                  Min.   : 0.0                Length:2283
 NA's:2283      Class :character     Class :character             1st Qu.:16.0                Class :character
                Mode  :character     Mode  :character             Median :16.0                Mode  :character
                                                                  Mean   :20.4
                                                                  3rd Qu.:32.0
                                                                  Max.   :48.0
                                                                  NA's   :1104
 Intel_Hyper_Threading_Technology_ Intel_Virtualization_Technology_VTx_  Intel_64_         Instruction_Set
 Length:2283                       Length:2283                          Length:2283        Length:2283
 Class :character                  Class :character                     Class :character   Class :character
 Mode  :character                  Mode  :character                     Mode  :character   Mode  :character



 Instruction_Set_Extensions Idle_States        Thermal_Monitoring_Technologies  Secure_Key         Execute_Disable_Bit
 Length:2283                Length:2283        Length:2283                      Length:2283        Length:2283
 Class :character           Class :character   Class :character                 Class :character   Class :character
 Mode  :character           Mode  :character   Mode  :character                 Mode  :character   Mode  :character
```

5.2 Distribution and Histograms



- Processor Base Frequency:
  - The distribution of processor base frequencies is relatively normal, with most CPUs having base frequencies between 2 and 3 GHz.
  - A few CPUs have base frequencies below 2 GHz and above 3 GHz, indicating that while the majority of CPUs fall within a mid-range frequency, there are some outliers at both ends.
- Number of Cores:
  - The distribution of the number of cores is right-skewed, with a large number of CPUs having between 2 and 8 cores.
  - There are fewer CPUs with higher core counts, and very few CPUs exceed 16 cores.

  - This suggests that while multi-core processors are common, high-core-count CPUs are less frequent.

- Number of Threads:
  - Similar to the number of cores, the distribution of the number of threads is right-skewed. The majority of CPUs have between 2 and 16 threads.
  - There are some CPUs with a significantly higher number of threads, indicating the presence of high-performance CPUs with technologies like Hyper-Threading.

- TDP (Thermal Design Power):
  - The distribution of Thermal Design Power (TDP) shows that most CPUs have a TDP between 50 and 125 watts.
  - There are a few CPUs with higher TDP values, which typically correspond to more powerful processors that require more cooling.
  - This indicates that most CPUs are designed to operate within a standard thermal envelope, but high-performance CPUs demand more power.

- Lithography:
  - The lithography histogram shows several peaks, indicating different manufacturing process nodes.
  - The most common process nodes are 14 nm and 22 nm.
  - There are also CPUs manufactured at smaller nodes like 10 nm, and larger nodes like 32 nm and 45 nm.
  - This reflects the advancements in semiconductor manufacturing technology, with newer CPUs being produced at smaller process nodes for improved performance and efficiency.

Implications:

- Processor Base Frequency: The relatively normal distribution suggests a common range of base frequencies, with few extreme values. This will influence our regression model, as we need to account for these outliers.

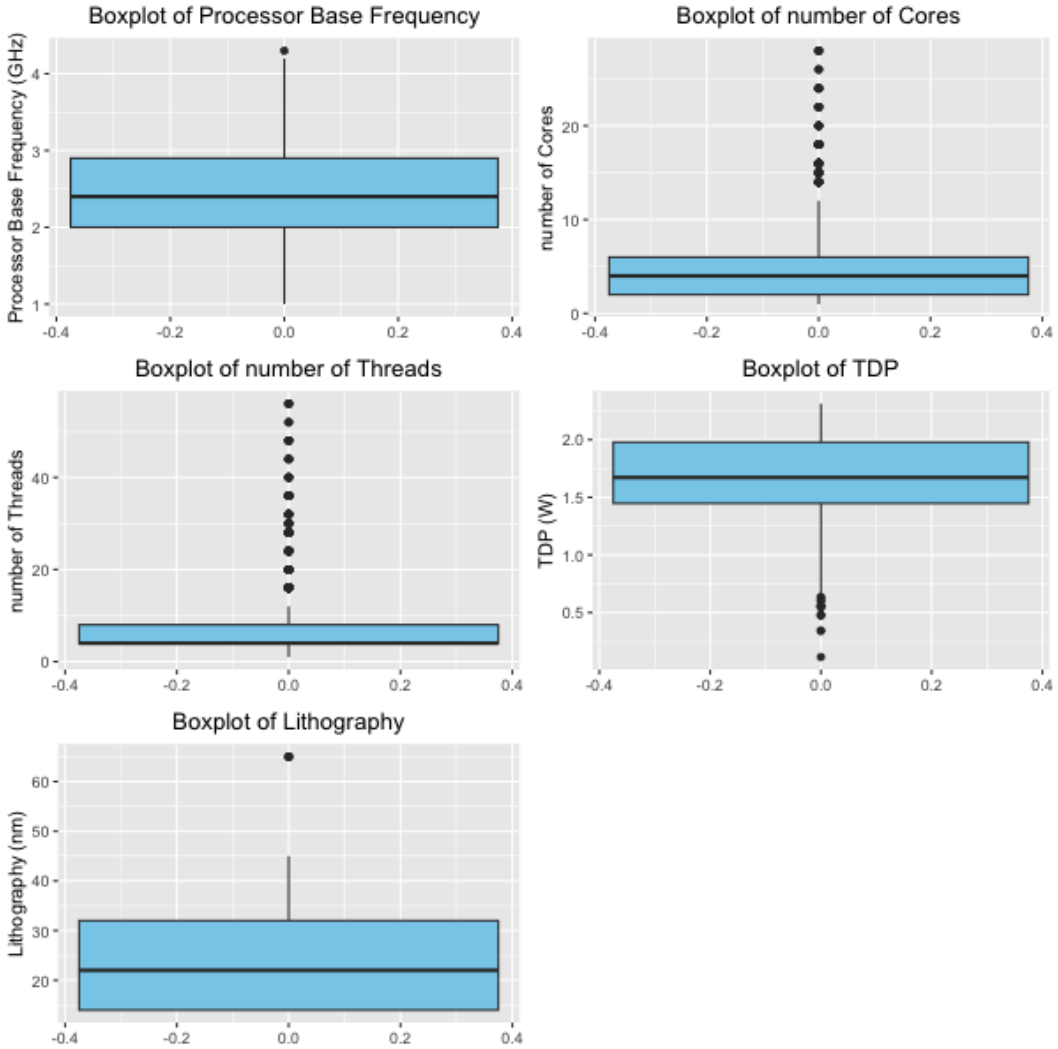
- Number of Cores and Threads: The right-skewed distributions indicate that while multi-core and multi-threading capabilities are common, the extent varies widely. High-core and high-thread CPUs, although fewer, represent the high-performance segment and should be carefully considered in our model.

- TDP: The distribution suggests that most CPUs are designed within certain power and thermal limits, but there are high-performance CPUs with higher TDP. This attribute is crucial as it impacts the CPU's ability to sustain higher clock speeds under load.

- Lithography: The presence of multiple peaks reflects the evolution of manufacturing processes over time. This attribute is essential for our model as newer process nodes typically allow for higher clock speeds and better efficiency.

By understanding these distributions, we can better prepare our data for inferential statistics and modeling, ensuring that our regression model accurately captures the relationship between these attributes and CPU clock speeds. This will involve handling outliers appropriately, scaling the features, and ensuring that the model is trained on a representative sample of the data.

5.3 Boxplots for Outliers



The boxplots provide a visual representation of the distribution, central tendency, and variability of key attributes in the dataset, as well as the presence of outliers. Here's a detailed explanation of each boxplot:

- Boxplot of Processor Base Frequency (GHz):
  - The boxplot shows the median processor base frequency around 2.5 GHz, with the interquartile range (IQR) spanning from approximately 2 GHz to 3 GHz. There are a few outliers above 3.5 GHz, indicating that some CPUs have significantly higher base frequencies than the majority.
- Boxplot of Number of Cores:
  - The median number of cores is around 4, with the IQR ranging from 2 to 6 cores.
  - Numerous outliers are present above 8 cores, reflecting that high-core-count CPUs, while less common, are still significant.
- Boxplot of Number of Threads:
  - The median number of threads is around 4, with an IQR from 4 to 8 threads.
  - There are many outliers above 16 threads, indicating that some CPUs, likely with Hyper-Threading, support a significantly higher number of threads.
- Boxplot of TDP (W):
  - The median TDP is around 65 W, with the IQR ranging from approximately 55 W to 95 W.
  - There are several outliers below 50 W and above 125 W, showing that while most CPUs fall within a certain thermal range, some require much less or much more cooling capacity.
- Boxplot of Lithography (nm):
  - The median lithography value is 22 nm, with the IQR ranging from 14 nm to 32 nm.
  - There is one outlier at 65 nm, indicating an older manufacturing process compared to the more advanced and common nodes.

Implications: The boxplots provide several insights that are crucial for our project on predicting CPU clock speeds:

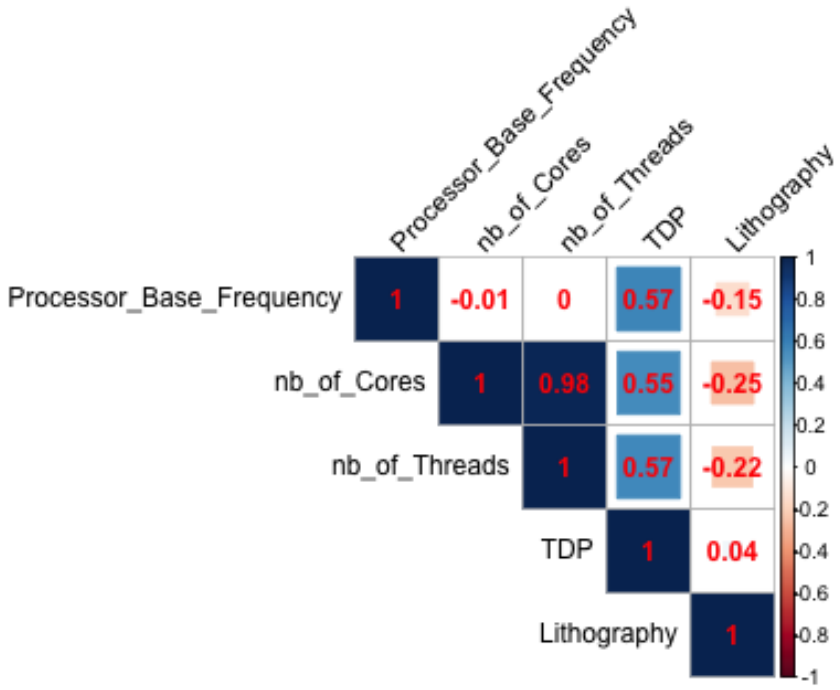
- Processor Base Frequency:
  - The central tendency and spread of base frequencies are important for understanding the typical performance range of CPUs in the dataset.
  - The presence of outliers indicates that while most CPUs have base frequencies in a common range, some high-performance CPUs exist with higher base frequencies, which need to be considered in our model.
- Number of Cores and Threads:

  - The distributions show that multi-core and multi-threaded CPUs are common, but high-core and high-thread counts are less frequent, representing high-performance segments.
  - Outliers in these attributes reflect high-performance CPUs, which can affect the overall performance trends and must be accurately modeled.

- TDP:
  - The central tendency and variability in TDP indicate the power and thermal characteristics of most CPUs.
  - The outliers suggest that while most CPUs operate within a standard thermal range, some require much higher or lower power, influencing their ability to sustain higher clock speeds.

- Lithography:
  - The distribution of lithography values highlights the evolution of manufacturing technology, with most CPUs produced using advanced nodes like 14 nm and 22 nm.
  - The outlier at 65 nm represents older technology, which typically correlates with lower efficiency and performance.

5.4 Correlation Matrix



The correlation matrix provides a comprehensive view of the linear relationships between pairs of variables in our dataset. Each cell in the matrix represents the correlation coefficient between two variables, ranging from -1 (perfect negative correlation) to 1 (perfect positive correlation). Here's a detailed explanation of the matrix for our project:

Key Observations:

- Processor Base Frequency:
  - Correlation with TDP: The strongest positive correlation is with TDP (0.57). This indicates that CPUs with higher base frequencies generally have higher thermal design power requirements. This is expected as higher clock speeds typically result in greater power consumption and heat generation.

  - Negative Correlation with Lithography: There is a moderate negative correlation with Lithography (-0.15). This suggests that CPUs manufactured with smaller process nodes tend to have higher base frequencies. Advanced manufacturing technologies often result in better performance characteristics.

- Number of Cores:
  - Strong Positive Correlation with Number of Threads (0.98): This high correlation is expected as CPUs with more cores generally support more threads, especially with technologies like Hyper-Threading.
  - Negative Correlation with Lithography (-0.25): Indicates that CPUs with a higher number of cores tend to be manufactured using smaller lithography nodes, aligning with advancements in CPU design that pack more cores into smaller spaces.
- Number of Threads:
  - Positive Correlation with TDP (0.57): Similar to the base frequency, a higher number of threads is associated with higher TDP, reflecting increased power consumption and heat dissipation needs.
- TDP:
  - Slight Positive Correlation with Lithography (0.04): This weak correlation suggests that the thermal design power does not have a strong relationship with the manufacturing process node.
- Lithography:
  - Negative Correlations with Other Variables: Lithography shows a general negative correlation with performance-related attributes, reinforcing the trend that smaller manufacturing nodes (indicative of newer technologies) are associated with higher performance capabilities.

x Implications:

- The correlation matrix is crucial for understanding the relationships between different CPU attributes and their collective impact on clock speed. Here are some key takeaways for our project on predicting CPU clock speeds:
- TDP and Processor Base Frequency: The strong positive correlation suggests that models predicting clock speed should account for TDP as a significant predictor.
- Manufacturing Process (Lithography): The negative correlations with performance attributes highlight the importance of considering lithography advancements in performance modeling.
- Number of Cores and Threads: The strong inter-correlation indicates that either variable could be used interchangeably in some modeling contexts, but including both provides a more nuanced understanding of CPU capabilities.

6 Inferential Statistics

6.1 Two-way analysis of variance(ANOVA)

Our model is about using two-way ANOVA to determine the relationship between the dependent Processor Base Frequency with two independent which are number of Cores and Thermal Design Power(TDP). Because we want to check validity of this ANOVA model, we will have two assumption for this:

Normality: Processor Base Frequency should be normal distribution for each combination of number of Cores and TDP.

Homogeneity: Processor Base Frequency should be roughly across all combinations of number of Cores and TDP.

6.1.1 Verify the assumption

Normality: verify if the data is normal distribution
We use Shapiro-Wilk normality test to check if this model is normal distribution:

Null hypothesis H_0: Processor Base Frequency should be approximately normally distributed for each combinations of number of Cores and TDP.

Alternative hypothesis H_1: Processor Base Frequency should not be approximately normally distributed for each combinations of number of Cores and TDP.

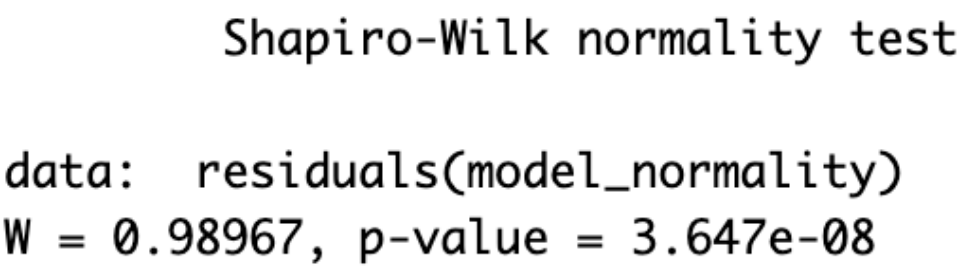


Figure 1: Shapiro-Wilk's test

From the result, p-value is smaller than 0.05 so we reject H_0 and accept H_1. Therefore, Processor Base Frequency should not be approximately normally distributed for each combinations of number of Cores and TDP. So we need to plot out the model diagnose plot to check if it is normal distribution or not:

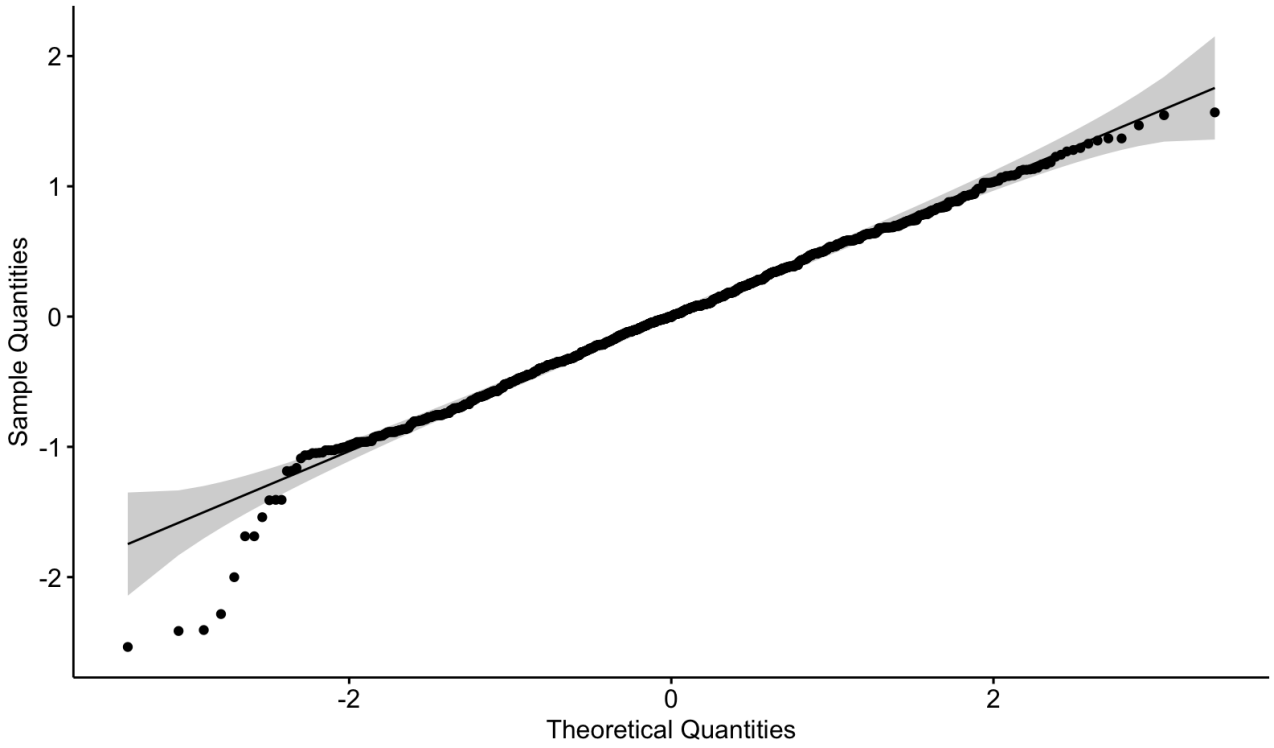


Figure 2: ggqqplot from ANOVA test

We see that in ggqqplot, most of the points lie closely on the line. So we can assume that Processor Base Frequency is approximately normally distributed for each combinations of number of Cores and TDP.

Homogeneity: verify the uniformity of the variance.

We use Levene's test to check if this model is homogeneity:

Null hypothesis H_0: Processor Base Frequency is homogeneity across groups of number of Cores and TDP.

Alternative hypothesis H_1: Processor Base Frequency is not homogeneity across groups of number of Cores and TDP.

```
Levene's Test for Homogeneity of Variance (center = median)
       Df F value    Pr(>F)
group  175  2.2534 2.281e-15 ***
      1175
```

Figure 3: Leneve's Test for Homogeneity of Variance

From the result, we see that p-value is smaller than 0.05 so we reject H_0 and accept H_1. Therefore, Processor Base Frequency is not homogeneity across groups of number of Cores and TDP.

6.1.2 Calculate ANOVA

Null hypothesis H_0: Processor Base Frequency follows the same distribution across group defined by the independent variables number of Cores and TDP.

Alternative hypothesis H_1: Processor Base Frequency follows different distribution across group defined by the independent variables number of Cores and TDP.

```
                                Df Sum Sq Mean Sq F value   Pr(>F)
factor(TDP)                     94  398.6   4.241  30.246  < 2e-16 ***
factor(Lithography)              4   16.3   4.086  29.139  < 2e-16 ***
factor(TDP):factor(Lithography)  77   32.0   0.416   2.964 5.42e-15 ***
Residuals                      1175  164.7   0.140
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 4: Calculate ANOVA

From the result, we see that Pr(>F) value is smaller than 0.05 so we reject H_0 and accept H_1. Therefore, Processor Base Frequency follows different distribution across group defined by the independent variables number of Cores and TDP.

6.2 Multiple Linear Regression to Predict CPU Clock Speed

6.2.1 Data Splitting

After finalizing the data we need for statistical analysis, we have to split the data further into two subsets: **training set and test set**. Training set helps us build and train the model, allowing it to learn the patterns and connections within our data. Then, the test set will be tested on the validated model to give an objective assessment of the model's efficacy and performance on fresh, unseen data. In data science and machine learning, this train-test split strategy is a standard procedure since it ensures that the model generalises effectively to new data and helps prevent over-fitting. In this project, the training set consists of 70% of the original data, and the remaining 30% makes up the test set.

```
smp_size <- floor(0.70 * nrow(intel_cpu_subset))
set.seed(123)
train_ind <- sample(seq_len(nrow(intel_cpu_subset)), size = smp_size)

train_set <- intel_cpu_subset[train_ind, ]
test_set <- intel_cpu_subset[-train_ind, ]
```

6.2.2 Regression Model

The main objective of this section is constructing a model that portrays the effect of other factors on the CPU clock speed. To achieve this we applied a Multi Regression model, in which the dependent variable is Processor Base Frequency - the variable representing CPU clock speed, and the rest are independent. Our model appears as the formula below:

$$ProcessorBaseFrequency = \beta_0 + \beta_1 nb\_of\_Cores + \beta_2 nb\_of\_Threads + \beta_3 TDP + \beta_4 Lithography$$

We start with our **first model**, also known as the **base model**, built with all the independent variables available.

In this model, our respone variables consist of: nb_of_Cores, nb_of_Threads, TDP and Lithography. Now, we should remove the variables that are proven insignificant to our analysis, which can be determined based on their Pr values (last column). If $Pr < 0.05$, the variable is significant. With this insight, the variable that will be deducted is nb_of_Threads. Subsequently, we now construct the **new and theoretically improved model**.

However, our group came to the decision of using the base model, since the difference in Adjusted R^2 value between the two is not significant enough (0.0001) to implement the second model as an improvement over the first. Hence, our final model now should look like this:

$$ProcessorBaseFrequency = 2.8134 - 0.1279 \cdot nb\_of\_Cores + 0.0002 \cdot nb\_of\_Threads + 0.0162 \cdot TDP - 0.0324 \cdot Lithography$$

Observing the results R gave on model 2, the p - value correlating with F statistics is less than $2.2 \cdot 10^{-16}$. This suggests that our data is robust and valuable for statistical analysis. Additionally, it guarantees that future results from this model provide good evaluations about the relationship between the Processor Base Frequency and the remaining variables. Generally, the regression coefficients (β_i) and the p - values hold the most influences on the independent variables.

6.2.3 Assumptions of Linear Regression

6.2.4 Testing

We first run the model on our **training set** (70% of the original data) to validate the model before actual testing. To gauge the correspondence between our estimates and the actual values, we look at their distribution and compare them.

The values plotted on this graph is quite widely distributed from the regression line, proving that the predicted and actual values form an average linear relationship. This confirms that the model is effective.

We now run the model on the preceding **test set** (30% of the original data) to access the performance of our model. Then, we put the predicted values into a new column in the dataset

for easy plotting.

```
    predicted_values <- predict(regression_model, test_set)
    test_set["Predicted"] <- predicted_values
```

Then we plot the graph following the exact same way as we did with the training set.

The graph portrays a linear relationship between the predicted and actual values, indicating that the model works on unseen data. This concludes that this model is good at predicting the clock speed based on the key variables mentioned.

6.3 Conclusion

After implementing a Multi Regression Model to predict the CPU Clock Speed, we were able to identified 4 variables that are significant to the Processor Base Frequency. The model aid manufacturers in pinpointing the factors that affect the CPU performance through its clock speed, providing appropriate strategies in product development, while also help customer choosing the right CPU specifications for their needs. Overall, the results that the model predict is justifiably similar to the actual data.

7 Conclusion

8 References

[1] Abdullah Al Sefat, G.M. Rasiqul Islam Rasiq, Nafiul Nawjis, and Sk Mehedi. *CPU Performance Prediction Using Various Regression Algorithms*, pages 163–171. 01 2021.

[2] Sheldon M Ross. *Introduction to Probability and Statistics for Engineers and Scientists*. Academic Press, 2020.

[3] Peter Bruce and Andrew Bruce. *Practical Statistics for Data Scientists*. O'Reilly, 2017.

[4] Douglas C Montgomery and George C Runger. *Applied Statistics and Probability for Engineers*. Wiley, 6th edition, 2006.

[5] William Stallings. *Computer Organization and Architecture*. Prentice Hall Professional Technical Reference, 6th edition, 2002.

[6] Nguyễn Tiến Dũng (chủ biên) và Nguyễn Đình Huy. *Xác suất - Thống kê & Phân tích số liệu*. Đại học Quốc gia TP. Hồ Chí Minh, 2019.

[7] Phan Thị Khánh Vân. Lecture notes on Probability and Statistics, LMS - HCMUT. https://lms.hcmut.edu.vn, 2024. Accessed: May 2024.

[8] Github. Hands-On Programming with R. https://rstudio-education.github.io/hopr/basics.html. Accessed: May 2024.

[9] [Học một chút]. Thống kê trong R-Studio. https://youtube.com/playlist?list=PL9XZJEFXvG9E9HJn2n1Gu_TLAsVPc1MkJ&feature=shared, 2021. Accessed: May 2024.

[10] Intel. What is Clock Speed? https://www.intel.com/content/www/us/en/gaming/resources/cpu-clock-speed.html. Accessed: May 2024.

[11] ILISSEK. Computer Parts (CPUs and GPUs). https://www.kaggle.com/datasets/iliassekkaf/computerparts, 2017. Accessed: April 2024.