VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

PROBABILITY AND STATISTICS (MT2013)

Assignment

The impact of CPU's characteristics on its Thermal Design Power

---

Advisor: Phan Thi Huong
Students: Thai Quang Phat - 2252606 (CC03, ***Leader***)
Bui Quoc Bao - 2152411 (CC03)
Nguyen Truong Duc Tai - 2252723 (CC03)
Dao Nam Anh - 2252016 (CC03)
Tran Nguyen Nam Hai - 2252190 (CC03)

HO CHI MINH CITY, MAY 2024

Contents

1 Member list & Workload

Mission

No.	Student ID	Full name	Workload	Percentage
1	2252606	Thai Quang Phat	Write report, Code R for all sections, Research for extension	100%
2	2252621	Bui Quoc Bao	Check report	60%
3	2252378	Nguyen Truong Duc Tai	Code R for descriptive statistics	80%
4	2252280	Dao Nam Anh	Check report	50%
5	2252377	Tran Nguyen Nam Hai	Check report	60%

2 Data Introduction

2.1 Overview of the dataset

The dataset is retrieved from Computer Parts (CPUs and GPUs) by author Ilissek. The dataset contains two CSV files: `all_gpus.csv` for Graphics Processing Units (GPUs), and `intel_cpus.csv` for Central Processing Units (CPUs).

In this project, we mainly focus on data from the `intel_cpus.csv` file. This dataset describes some important attributes of the CPU in the market, including the performance (through *Processor Base Frequency*), the power consumption and heat (through *Thermal Design Power* and *Temperature*), and the technology trend (through *Number of Cores* and *Lithography*).

Some notable attributes of this dataset are:

- **Vertical Segment.** Describes some segments of the market where the CPU is mainly used in, some of it includes Mobile, Desktop, Embedded, and Server.
- **Status.** Describes the status of the CPU in the market. Some of it includes: 'Launch' if the product has been launched and still in the market, 'End of Life' if the product is no longer being produced to the market, but can still be supported if have problems, 'End of Interactive Support' if the product is no longer in the market and have to support service.
- **Launch Date.** Describes the launch date of the product, formatted by `<Quarter>'<Year>`, for example, `Q1'15` means that the product was launch on the First Quarter in year 2015.
- **Lithography.** The semiconductor technology used to manufacture an integrated circuit, and is reported in nanometer
- **Recommended Customer Price.** The price recommended by Intel for retailers selling the CPU.
- **Number of Cores.** A hardware term that describes the number of independent central processing units.
- Processor Base Frequency. Describes the rate at which the processor's transistors open and close.
- **Thermal Design Power (TDP).** Represents the average power, in watts, the processor dissipates when operating at Base Frequency with all cores
- **Temperature (T).** The maximum temperature allowed on the chip.

3 Background

3.1 Regression Model

Regression is a statistical method for determining the relationship between features and an outcome variable or result. Machine learning, it's utilized as a method for predictive modeling, in which an algorithm is employed to forecast continuous outcomes.

3.1.1 Linear Regression Model

Multiple linear regression, often known as multiple regression, is a statistical method that predicts the result of a response variable by combining numerous explanatory variables. Multiple regression is a variant of linear regression (ordinary least squares) in which just one explanatory variable is used. The formula for multiple linear regression is

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \varepsilon$$

where:

- y_i is the dependent variable.
- $x_{i1}, x_{i2}, \ldots$ are the explanatory variables.
- β_0 is the intercept of the line.
- $\beta_1, \beta_2, \ldots$ are the slope coefficients for each explanatory variable.
- ε is the model's error term (also known as the residuals).

When dealing with linear regression, the following assumptions for data are required.

1. **Linearity.** the relationship between the dependent variable and independent variable are linear.
2. **Independence.** the observations are independent of each other, which means

   $$Cov(\varepsilon_i, \varepsilon_j) = 0, \ \forall i \neq j$$

   where $Cov(\varepsilon_i, \varepsilon_j)$ is the covariance between the errors for observations i and j.
3. **Homoscedasticity.** the variance of the errors is constant across all levels of the independent variable(s)

   $$Var(\varepsilon_i) = \sigma^2, \ \forall i$$

   where $Var(\varepsilon_i)$ is the variance of the error for observations i and σ^2 is a constant.
4. **Normality.**, the errors are normally distributed

   $$\varepsilon \sim N(0, \sigma^2)$$

   where ε is the error term and $N(0, \sigma^2)$ denotes a normal distribution with mean 0 and variance σ^2.

3.1.2 Multivaritate Adaptive Regression Spline (MARS)

Multivariate Adaptive Regression Splines (MARS) is an advanced statistical modeling technique that extends traditional regression models to capture more complex relationships between variables.

MARS models work by constructing piecewise linear regressions, which are more flexible than traditional polynomial models. The basic formula for a MARS model involves creating a series of "basis functions" that are combined to form the final model. These basis functions are typically products or interactions of truncated power functions, known as "hinge functions." The general formula of a hinge function is:

$$B(x) = \max(0, x - c)^d$$

where c is a constant defining the location of the hinge (knot), and d is typically set to 1, making the function linear past the knot.

The overall MARS model can be expressed as:

$$Y = \beta_0 + \sum_{i=1}^{k} \beta_i B_i(x)$$

Here β_0 is the intercept, β_i are coefficients, and $B_i(x)$ are the basis functions constructed from the input variables x. The model adapts to the data by automatically selecting appropriate basis functions and their interactions, optimizing the placement of knots, and pruning less significant terms to prevent overfitting. This process allows MARS to model nonlinearities and interactions effectively, making it a powerful tool for predicting complex patterns.

3.1.3 Polynomial Regression

In statistics, polynomial regression is a form of regression analysis in which the relationship between the independent variable x and the dependent variable y is modeled as an nth degree polynomial in x. Polynomial regression fits a nonlinear relationship between the value of x and the corresponding conditional mean of y, denoted $E(y|x)$. Polynomial regression is considered to be a special case of multiple linear regression.

The formula for polynomial regression is

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \varepsilon$$

where

- y is the dependent variable.
- β_i are coefficients.
- ε is an unobserved random error with mean zero conditioned on scalar variable x.

When using polynomial regression, the following assumptions are made

1. **The dependency is polynomial.** the behavior of a dependent variable y can be explained by a linear, or curvilinear, additive relationship between the dependent variable and a set of k independent variables ($x_i, i = 1$ to k)),
2. **Linearity or Curvilinearity.** the relationship between the dependent variable y and any independent variable x_i is linear or curvilinear (specifically polynomial),
3. **Independence.** the independent variables x_i are independent of each other, and the errors are independent, normally distributed with mean zero and a constant variance.

3.1.4 Smoothing Regression

Smoothing splines, like kernel regression and k-nearest-neighbors regression, provide a flexible way of estimating the underlying regression function $r(x) = E(Y|X = x)$. Though they can be defined for higher dimensions, we'll assume for simplicity throughout that $x \in \mathbb{R}$, i.e., there is only one predictor variable.

Before introducing smoothing splines, however, we first have to understand what a spline is. In words, a kth order spline is a piecewise polynomial function of degree k, that is continuous and has continuous derivatives of orders $1, ...k-1$, at its knot points.

Formally, a function $f : R \to R$ is a kth order spline with knot points at $t_1 < ... < t_m$, if

- f is a polynomial of degree k on each of the intervals $(-\infty, t_1], [t_1, t_2], ..., [t_m, +\infty)$.
- $f^{(j)}$, the jth derivative of f, is continuous at $t_1, \ldots, t_m$ for each $j = 0, 1, \ldots, k-1$.

Splines have some very special properties are have been a topic of interest among statisticians and mathematicians for a long time.A regularized regression over the natural spline basis, placing knots at all points $x_1, ...x_n$. Smoothing splines circumvent the problem of knot selection (as they just use the inputs as knots), and simultaneously, they control for overfitting by shrinking the coefficients of the estimated function (in its basis expansion).

3.2 Statistics measurements

3.2.1 Box Plot

A Box Plot is a graphical method used to visualize data distribution, providing insights and supporting informed decision-making. It displays key summary statistics such as the median, quartiles, and potential outliers in a concise and visual manner, making it easy to summarize the distribution, identify potential outliers, and compare different datasets. Which includes minimum value, first quartile (Q1), median (Q2), third quartile (Q3) and maximum value.

3.2.2 Histogram

A histogram is a graphical representation used in statistics to visually display the distribution of numerical data through bins (or classes). It resembles a bar chart but differs in that each bar represents a range of data values (the bin width) rather than a single value. The height of each bar shows the frequency of data points that fall within each bin, providing insights into the central tendency, variability, and skewness of the data.

3.2.3 The Q-Q Plot

A **Q-Q plot**, A quantile-quantile (Q-Q) plot compares distributions, showing how properties like location, scale, and skewness differ between two datasets or theoretical distributions. Deviations from a straight diagonal line in the plot indicate that the data may not follow a normal distribution.

3.2.4 R-Squared (R^2)

The **R-squared** coefficient represents the proportion of variation in the dependent variable (y) that is accounted for by the regression line, compared to the variation explained by the mean of y. Essentially, it

measures how much more accurately the regression line predicts each point's value compared to simply using the average value of y.

The coefficient of determination which is represented by R^2 is determined using the following formula:

$$R^2 = 1 - \frac{\text{residual sum of squares}}{\text{total sum of squares}}$$

The adjusted coefficient of determination is the multiple coefficient of determination R^2 modified to account for the number of variables and the sample size. It is calculated by

$$R^2_{\text{adj}} = 1 - \frac{(1 - R^2)(N - 1)}{(N - p - 1)}$$

where R^2 is the normal R-Squared value, N is the sample size of sample, and p is the number of predictors.

3.2.5 P-Value

The p-value is a statistical measure used in hypothesis testing to assess evidence against a null hypothesis. It helps decide whether to reject or support the null hypothesis, providing insights into the statistical significance of an independent variable in predicting the dependent variable.

A small p-value suggests that the observed data is unlikely to have occurred by random chance alone, leading to the rejection of the null hypothesis. However, it's crucial to choose the appropriate test based on the data and research question and interpret the p-value in the context of the specific test being used.

3.3 Analysis of Variance ANOVA

Analysis of variance (ANOVA) is a statistical method used to test for differences among two or more population means by analyzing the variances of samples taken from the populations.

One-way ANOVA is a statistical method to compare the variances of multiple levels of a single factor.

For each observation under the treatment i under the j observation called y_{ij} we have the linear combination

$$Y_{ij} = \mu + \tau_i + \varepsilon_{ij} \quad \begin{cases} i = 1, 2, \ldots, a \\ j = 1, 2, \ldots, n \end{cases}$$

where

- Y_{ij} is a random variable denoting the (ij)th observation.
- μ is a parameter common to all treatments called the **overall mean**.
- τ_i is a parameter associated with the ith treatment called the ith **treatment effect**.
- ε_{ij} is a random error component.

Notice that the model can be written as

$$Y_{ij} = \tau_i + \varepsilon_{ij} \quad \begin{cases} i = 1, 2, \ldots, a \\ j = 1, 2, \ldots, n \end{cases}$$

where

- $\mu_i = \mu + \tau_i$ is the mean of the ith treatment.
- ε_{ij} are the errors which follows the normal distribution, which means $\varepsilon_{ij} \sim N(0, \sigma^2)$.

The linear combination is the underlying model for a single-factor experiment. Furthermore, because we require that the observations are taken in **random order** and that the environment (often called the experimental units) in which the treatments are used is as uniform as possible, this experimental design is called a **completely randomized design (CRD)**.

The a factor levels in the experiment could have been chosen in two different ways.

1. The experimenter could have specifically chosen the a treatments. This is called the **fixed-effects model**.
2. Alternatively, the a treatments could be a random sample from a larger population of treatments. This is called the **random-effects**, or **components of variance model**.

Sum of Squares: Single Factor Experiment.

$$SS_T = SS_{\text{treatment}} + SS_E$$

This means that the **sum of squares** can be partitioned into the **treatment sum of squares** and **error sum of squares**.

Expected Value for Sums of Squares. The expected value of the treatment sum of squares is

$$E\left(SS_{\text{Treatments}}\right) = (a-1)\sigma^2 + n\sum_{i=1}^{a}\tau_i^2$$

and the expected value of the error sum of squares is

$$E\left(SS_E\right) = a(n-1)\sigma^2$$

There are $an = N$ observations; thus, SS_T has $an - 1$ degrees of freedom. There are a levels of the factor, so $SS_{\text{Treaments}}$ has $a - 1$ degrees of freedom. The ratio

ANOVA F-test.

$$F_0 = \frac{SS_{\text{Treatments}}/(a-1)}{SS_E/[a(n-1)]} = \frac{MS_{\text{Treatments}}}{MS_E}$$

has an F-distribution with $a - 1$ and $a(n - 1)$ degrees of freedom.

Consequently, we should reject H_0 if the statistic is large. This implies an upper-tailed, one-tailed critical region. Therefore, we would reject H_0 if

$$f_0 > f_{\alpha,a,a(n-1)}$$

where f_0 is the computed value of F_0.

From here we have the following computing results

Computing Formulas for ANOVA. The error sum of squares is obtained as

$$SS_E = SS_T - SS_{\text{Treatments}}$$

3.4 Kruskal - Wallis H-Test

The Kruskal–Wallis test by ranks, **Kruskal–Wallis H test**, or one-way ANOVA on ranks is a non-parametric method for testing whether samples originate from the same distribution.It is used for comparing two or more independent samples of equal or different sample sizes. The parametric equivalent of the Kruskal–Wallis test is the one-way analysis of variance (ANOVA).

We have the statistic test formula

$$H = \frac{12}{N(N+1)} \sum_{i=1}^{k} \frac{R_i^2}{n_i} - 3(N+1)$$

where

- N is the total sample size.
- k is the number of groups we are comparing.
- R_i is the sum of ranks for group i.
- n_i is the sample size of group i.

When using the H-test, the following assumptions are made, **ordinal variables**, **independence** ,**sample size.** of 5 or more. We then compare H to a critical cutoff point determined by the chi-square (χ^2) distribution. In this test, we specify H_0 being that the medians of each group are the same, meaning that all groups come from the same distribution.

3.5 Bartlett's test

In statistics, Bartlett's test is used to test homoscedasticity, that is, if multiple samples are from populations with equal variances. Some statistical tests, such as the analysis of variance, assume that variances are equal across groups or samples, which can be verified with Bartlett's test.

If there are k sample test with sizes n_i, then the Bartlett's test statistic is

$$\chi^2 = \frac{(N-k)\ln\left(S_p^2\right) - \sum_{i=1}^{k}(n_i-1)\ln\left(S_i^2\right)}{1 + \frac{1}{3(k-1)}\left(\sum_{i=1}^{k}\left(\frac{1}{n_i-1}\right) - \frac{1}{N-k}\right)}$$

where

- $N = \sum_{i=1}^{k} n_i$ is the total number of observations across groups.
- S_i^2 is the variance of group i.
- $S_p^2 = \frac{1}{N-k}\sum_{i=1}^{k}(n_i-1)S_i^2$ is the pooled variance.

With the null hypothesis being H_0 that all k population variances are equal against the alternative that at least two are different.

The test statistic has aproximately a χ^2_{k-1} distribution. Thus the null hypothesis rejects if $\chi^2 > \chi^2_{k-1,\alpha}$.

3.6 Shapiro-Wilk test[1]

The Shapiro-Wilk test tests the null hypothesis that a sample $x_1, ..., x_n$ came from a normally distributed population. The test statistic is

$$W = \frac{\left(\sum_{i=1}^{n} a_i x_{(i)}\right)^2}{\sum_{i=1}^{n}(x_i - \overline{x})^2}$$

where

- $x_{(i)}$ with parentheses enclosing the subscript index i is the ith order statistic, i.e., the ith-smallest number in the sample (not to be confused with x_i).
- $\overline{x} = \frac{(x_1 + \cdots + x_n)}{n}$ is the sample mean.
- a_i are coefficients given by $(a_1, \ldots, a_n) = \frac{m^\mathsf{T} V^{-1}}{C}$ where C is a vector norm

  $$C = \|V^{-1}m\| = (m^\mathsf{T} V^{-1} V^{-1} m)^{1/2}$$

  and the vector m is

  $$m = (m_1, \ldots, m_n)^\mathsf{T}$$

  is made of the expected values of the order statistics of independent and identically distributed random variables sampled from the standard normal distribution; finally, V is the covariance matrix of those normal order statistics.

The null-hypothesis of this test is that the population is normally distributed. Thus, if the p value is less than the chosen alpha level, then the null hypothesis is rejected and there is evidence that the data tested are not normally distributed.

3.7 Post-hoc Comparision tests

In research, post hoc analysis entails running extra statistical tests after examining data to find differences between group means. However, this leads to a multiple testing issue, as each analysis is essentially a separate test. Critics call post hoc analysis conducted without considering this problem adequately ”data dredging”, as it often produces misleading results.

3.7.1 Tukey’s range test

Tukey’s range test, also known as Tukey’s HSD (honestly significant difference) test, is a single-step multiple comparison procedure and statistical test. It can be used to correctly interpret the statistical significance of the difference between means that have been selected for comparison because of their extreme values.

[1] A 2011 study concludes that Shapiro-Wilk has the best power for a given significance, followed closely by Anderson-Darling when comparing the Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors, and Anderson-Darling tests.

Using Tukey's range test will need to have the following assumptions: **normal distribution**, **homogeneity of variance**, and **independence**.

Tukey's test compares the means of every treatment to the means of every other treatment; that is, it applies simultaneously to the set of all pairwise comparisons

$$\mu_i - \mu_j \ ,$$

and identifies any difference between two means that is greater than the expected standard error. The Tukey method is conservative when there are unequal sample sizes.

The Tukey method uses the studentized range distribution. Suppose that we take a sample of size n from each of k populations with the same normal distribution $N(\mu, \sigma^2)$, the following random variable has studentized distribution

$$q \equiv \frac{\overline{y}_{\max} - \overline{y}_{\min}}{S\sqrt{2/n}}$$

where $\overline{y}_{\min}$ is the smallest of all sample means, $\overline{y}_{\max}$ is the largest of all sample means, and S^2 is the pooled sample variance.

This test indicates two means are different if $q > g(\alpha, f) \times S$, where, S is the standard error of this statistic, and $g(\alpha, f)$ is studentized range distribution of significant level α and even degree of freedom f.

3.7.2 Dunn's test

Dunn's z-test statistic approximates exact rank-sum test statistics by using the mean rankings of the outcome in each group from the preceding Kruskal–Wallis test ($W_i = W_i/n_i$, where W_i is the sum of ranks, and n_i is the sample size for the ith group) and basing inference on the differences in mean ranks in each group. To compare group A with group B, we calculate $z_i = \dfrac{y_i}{\sigma_i}$ where $i \in \{1, \ldots, m\}$ be the multiple comparisions, $y_i = \overline{W}_A - \overline{W}_B$.

σ_i is the standard deviation given by

$$\sigma_i = \sqrt{\left\{ \frac{N(N+1)}{12} - \frac{\sum_{s=1}^{r} \tau_s^2 - \tau_s}{12(N-1)} \right\} \left(\frac{1}{n_A} + \frac{1}{n_B} \right)}$$

where

- N is the total number of observations across all groups.
- r is the number of tied ranks.
- τ_s is the number of observations tied at the sth specific tied value.

When there are no ties, the term with the summation in the denominator equals zero.

There are many ways for adjustment, we decide to use **Bonferroni adjustment.** which multiplies each p-value by m as $p^* = pm$

4 Data Preproceeding

4.1 Importing Data

We first load some libraries for later use

```
pacman::p_load(
  rio,           # for dealing with basic import export
  ggplot2,       # for dealing with plot formats
  zoo            # for dealing with year quarter formats
)
```

After having the libraries imported, we import the data. Since the names of the attributes are long and may cause error in calling (through wrong syntax), renaming attributes are made, and then exported to `cpu-short.csv`.

```
# Rename labels - easier to use
names(data) <- c("market", "status", "ldate", "litho", "rprice", "ncore", "bfreq", "tdp", "temp")
names(data)

head(data)    #showing 6 first rows of the data set

export(data, "cpu-short.csv")
```

Running the above code should create a new file name `cpu-short.csv` with renamed attributes.

4.2 Data Cleaning

4.2.1 Standardization Data

The original dataset has many missing data, as can be seen in Figure 1 below.

	A	B	C	D	E	F	G	H	I
1	market	status	ldate	litho	rprice	ncore	bfreq	tdp	temp
2	Mobile	Launched	Q3'16	14 nm	$393.00	2	1.30 GHz	4.5 W	100Â°C
3	Mobile	Launched	Q3'17	14 nm	$297.00	4	1.60 GHz	15 W	100Â°C
4	Mobile	Launched	Q3'17	14 nm	$409.00	4	1.80 GHz	15 W	100Â°C
5	Desktop	End of Life	Q1'12	32 nm	$305.00	4	3.60 GHz	130 W	66.8Â°C
6	Mobile	Launched	Q1'17	14 nm	$281.00	2	1.20 GHz	4.5 W	100Â°C
7	Mobile	Launched	Q1'15	14 nm	$107.00	2	1.50 GHz	15 W	105Â°C
8	Mobile	Launched	Q3'13	22 nm	N/A	2	1.46 GHz	4.3 W	80Â°C
9	Desktop	Launched	Q3'13	22 nm	N/A	2	2.41 GHz	10 W	100Â°C
10	Desktop	Launched	Q1'13	22 nm	$42.00	2	2.60 GHz	55 W	
11	Mobile	End of Interactive Sup		90 nm	N/A	1	2.80 GHz	88 W	75Â°C
12	Mobile	Launched	Q3'12	22 nm	$134.00	2	2.40 GHz	35 W	90 C
13	Mobile	End of Interactive Sup		90 nm	N/A	1	1.30 GHz	5.5 W	100Â°C
14	Mobile	Launched	Q1'15	14 nm	$161.00	2	1.90 GHz	15 W	105Â°C
15	Mobile	Launched	Q3'15	14 nm	$161.00	2	2.10 GHz	15 W	100Â°C

Figure 1: Data in cpu-short.csv

Since the values vary in types (such as string, non-standard year-quarter format and numeric-string), we might want transform them into reproducible types, so that the analysis later on is easier, homogeneous and accurate.

Note that the cleaning process **do not** eliminate rows with **N/A** values, unless necessary, since some of it might contain important values. In later sections, when we focus on a specific pattern of the data, only by

then that the data will have a tailored **N/A** cleaning, and we will not, by chance, loose any important instance.

Note that we columns 'market' and 'status' are left unchanged, we proceed to normalize other columns.

4.2.2 Cleaning data

We have opted to retain the original dataset, converting it into a format more suitable for analysis. For data presented as ranges, we have chosen to consistently select the highest value within each range.

4.2.3 Export Clean Data

After all the above process is done, we can export data to `clean-cpu.csv` and have the following result.

	A	B	C	D	E	F	G	H	I
1	market	status	ldate	litho	rprice	ncore	bfreq	tdp	temp
2	Mobile	Launched	2016.5	14	393	2	1.3	4.5	100
3	Mobile	Launched	2017.5	14	297	4	1.6	15	100
4	Mobile	Launched	2017.5	14	409	4	1.8	15	100
5	Desktop	End of Life	2012	32	305	4	3.6	130	66.8
6	Mobile	Launched	2017	14	281	2	1.2	4.5	100
7	Mobile	Launched	2015	14	107	2	1.5	15	105
8	Mobile	Launched	2013.5	22		2	1.46	4.3	80
9	Desktop	Launched	2013.5	22		2	2.41	10	100
10	Desktop	Launched	2013	22	42	2	2.6	55	
11	Mobile	End of Interactive Supp		90		1	2.8	88	75
12	Mobile	Launched	2012.5	22	134	2	2.4	35	90
13	Mobile	End of Interactive Supp		90		1	1.3	5.5	100
14	Mobile	Launched	2015	14	161	2	1.9	15	105
15	Mobile	Launched	2015.5	14	161	2	2.1	15	100

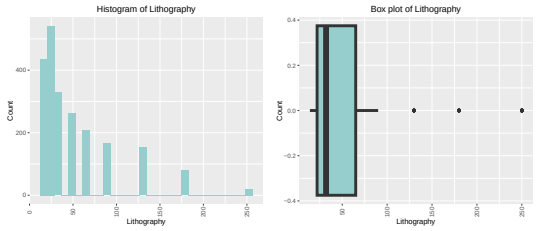
Figure 2: Data in cpu-clean.csv

5 Descriptive Statistics

5.1 Observation of some Attributes

Of each attributes plotted below, `NA` rows had been removed for cleaner data.

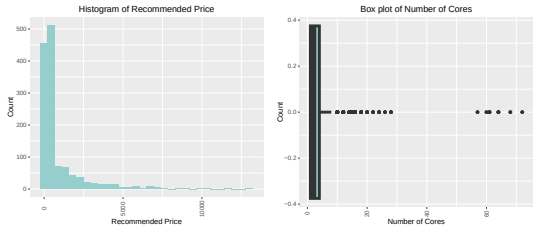
Lithography



(a) Lithography histogram (b) Lithography box plot

Figure 3: Lithography plots

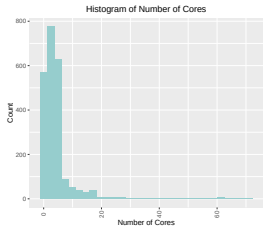
Recommended Price



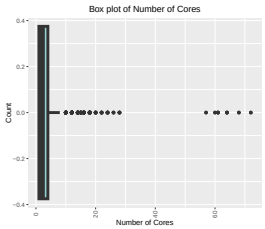
(a) R-Price histogram (b) R-Price box plot

Figure 4: Recommended Price plots

Number of Cores



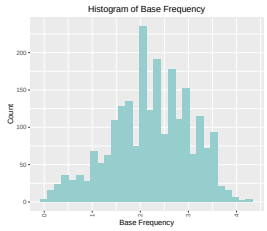
(a) N-Cores histogram



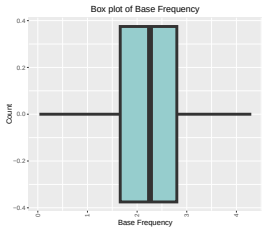
(b) N-Cores box plot

Figure 5: Number of Cores plots

Base Frequency



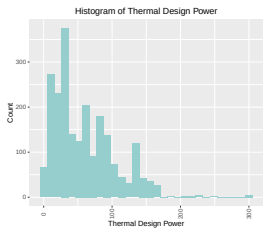
(a) B-Frequency histogram



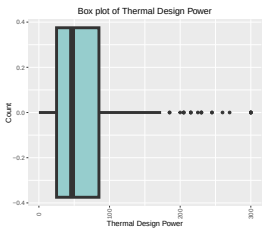
(b) B-Frequency box plot

Figure 6: Number of Cores plots

Thermal Design Power



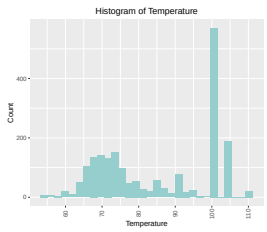
(a) TDP histogram



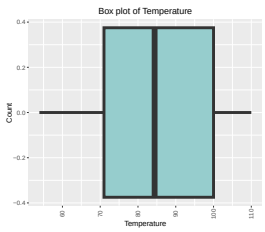
(b) TDP box plot

Figure 7: Thermal Design Power plots

Temperature



(a) Temperature histogram



(b) Temperature box plot

Figure 8: Temperature plots

Our analysis yields the following insights:

- **Lithography.** The histogram suggests non-normality and the box plot shows a negative skew possibly due to outliers. Addressing these outliers is crucial for further analysis.
- **Recommended Price.** The left-skewed distribution indicates a strategy focused on lower prices to enhance market penetration and affordability.
- **Number of Cores.** There is no significant performance gain with more cores, suggesting that the typical 2-3 cores in CPUs balance power and efficiency effectively.
- **Base Frequency.** The distribution is normal, with a median that aligns well with the center of the plot.
- **Thermal Design Power.** The distribution closely resembles a normal distribution, making it suitable for statistical methods that assume normality.
- **Temperature.** A slight gap at the median in the histogram suggests data irregularities that may need adjustments for precise analysis.

5.2 Attributes in Relationship

We now take a look at some plots generated by data from lithography. Using box-plot to show the distribution of lithography across launch dates with median, as well as histogram to show value distribution of lithography across launch dates.

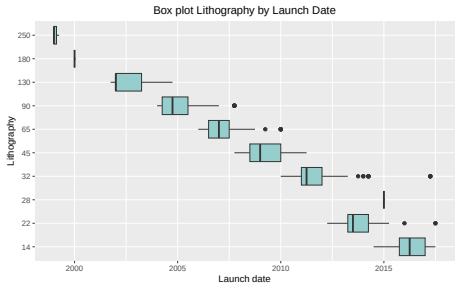


Figure 9: Lithography boxplot

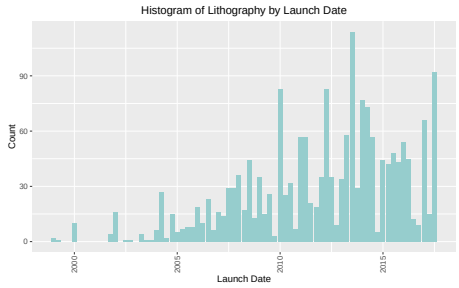


Figure 10: Lithography histogram

Examining the box plot in Figure 9, we observe that the median is closer to the lower end of the box, indicating a positive skew (right-skewed). Post-2015, the distribution becomes symmetrical around the median. Lithography, which spans over time, proves more informative than Launch date in our models, leading us to prefer Lithography for analysis Figure 11.

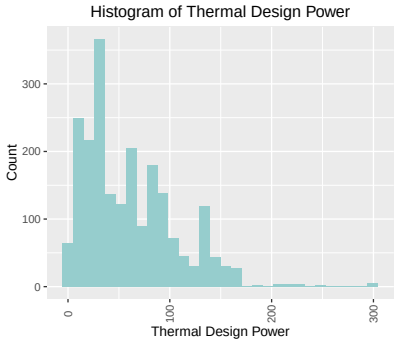


Figure 11: Thermal Design Power Histogram

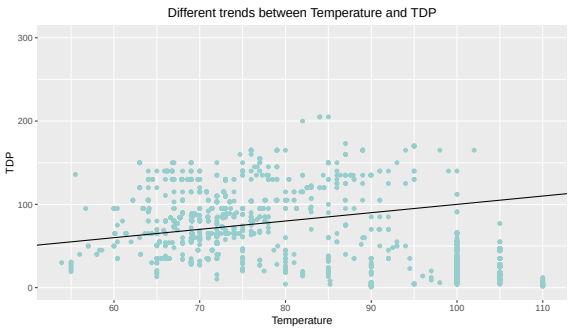
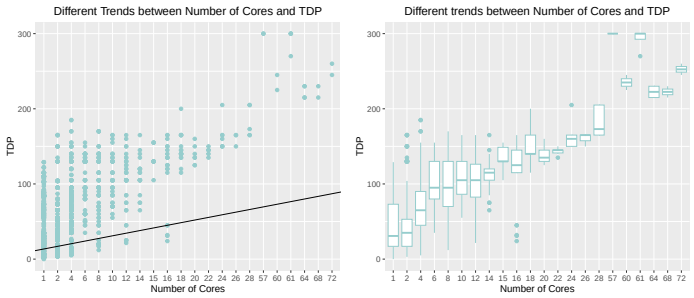


Figure 12: Thermal Design Power in relation with Temperature

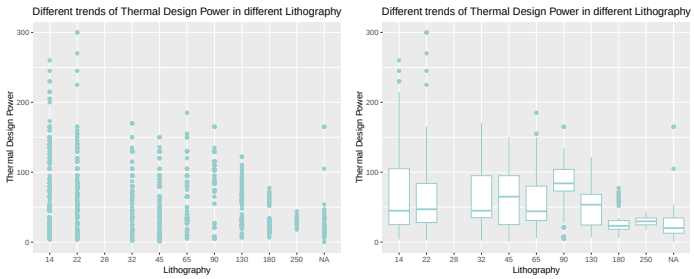
Next we take a look at the dependence of Thermal Design Power on different ranges of Temperature.
As we can see, TDP with value greater or equals to 125 W tends to scatter less in the plot. Keeping these data might cause error in analyse later on, which can be quickly seen by the **regression line** (black). From this observation, we can remove these spurious data, or treat it as outliers.



(a) Histogram plot (b) Box plot

Figure 13: Thermal Design Power in relation with Number of Cores

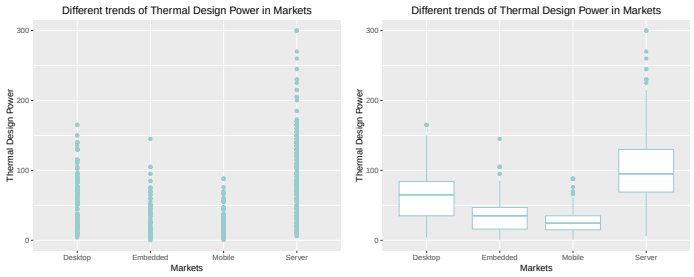
Upon observing the relationship between Thermal Design Power (TDP) and Number of Cores, we notice that TDP tends to increase with the Number of Cores. However, there are relatively few data points, and some outliers may have minimal influence on the overall trend.



(a) Histogram plot (b) Box plot

Figure 14: Thermal Design Power in relation with Lithography

The distribution of Thermal Design Power (TDP) across Lithography appears to exhibit a uniform pattern, resulting in a regression line that closely parallels the x-axis. This observation holds promise for subsequent analyses.



(a) Histogram plot (b) Box plot

Figure 15: Thermal Design Power in relation with Market

The data reveals an evident increasing trend in the Server category, as the majority of the data points are concentrated within this column. However, this observation underscores the presence of distinct trends across different markets, offering valuable insights for future logistic regression analysis.

5.3 Data summary

We want summarize information of all attributes in the dataset. Using the function `summary()`, we get the following result

```
   market             status              ldate          litho            rprice
Length:2265        Length:2265        Min.   :1999   Min.   : 14.00   Min.   :    2.54
Class :character   Class :character   1st Qu.:2010   1st Qu.: 22.00   1st Qu.:  161.00
Mode  :character   Mode  :character   Median :2013   Median : 32.00   Median :  304.00
                                      Mean   :2012   Mean   : 49.17   Mean   :  852.28
                                      3rd Qu.:2015   3rd Qu.: 65.00   3rd Qu.:  774.00
                                      Max.   :2018   Max.   :250.00   Max.   :13011.00
                                      NA's   :416    NA's   :71       NA's   :965
     ncore            bfreq             tdp              temp
Min.   : 1.000   Min.   :0.032   Min.   :  0.025   Min.   : 53.90
1st Qu.: 1.000   1st Qu.:1.660   1st Qu.: 25.000   1st Qu.: 71.00
Median : 2.000   Median :2.260   Median : 47.000   Median : 84.40
Mean   : 4.075   Mean   :2.222   Mean   : 60.242   Mean   : 84.87
3rd Qu.: 4.000   3rd Qu.:2.800   3rd Qu.: 85.000   3rd Qu.:100.00
Max.   :72.000   Max.   :4.300   Max.   :300.000   Max.   :110.00
                                 NA's   :49        NA's   :254
```

The dataset summary provides an overview of the data. Upon comparing this summary to the plot presented in the previous section, we observe consistency and accuracy in the visualization. It's worth noting that some discrepancies may arise from the removal of `NA` values, which was done to ensure a cleaner plotting process rather than affecting the analysis.

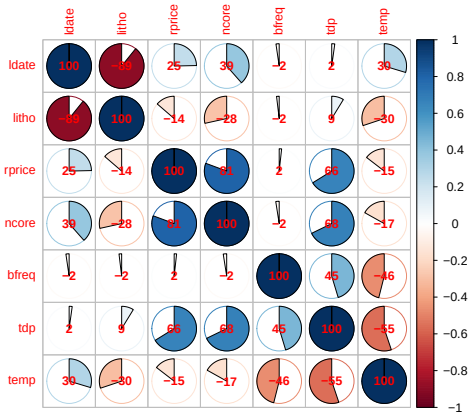


Figure 16: Correlation between attributes

The presented correlation plot illustrates the relationship between attributes after excluding rows containing missing values. A notable observation is the positive correlation of the attribute `tdp` with `rprice`, `ncore`, and `bfreq`. This suggests that these attribute pairs hold potential for prediction using regression models.

Please consider that the correlation plot has been generated after removing missing values. It's essential to acknowledge that real-world predictions may vary depending on actual conditions and data completeness.

6 Inferential Statistics

6.1 Choosing Lithography as CPU era

First, we observe why Lithography has greater impact than Launch date. Before doing so, we remove any missing value rows from the dataset.

We start by looking at the confidence interval and the visualization of Lithography by Launch dates.

	litho	5% quantile	95% quantile	STD Mean	Confidence Interval
1	14	2015.00	2017.500	0.8640376	2.5000
2	22	2012.25	2015.000	0.8159340	2.7500
3	32	2010.00	2012.500	0.9585590	2.5000
4	45	2007.75	2010.500	0.8603672	2.7500
5	65	2006.00	2008.312	0.7363130	2.3125
6	90	2004.00	2007.750	1.0586129	3.7500
7	130	2001.75	2004.250	0.9025055	2.5000
8	180	2000.00	2000.000	0.0000000	0.0000
9	250	1999.00	1999.225	0.1443376	0.2250

Figure 17: Correlation between attributes

The Mean of Standard Deviation and Confidence Interval suggest CPU designs last about two and a half years, defined distinctly by lithography rather than launch dates. This allows us to group CPUs by shared characteristics, using lithography as the primary reference for discussing CPU periods.

6.2 ANOVA Testing

As mentioned earlier, we've observed a strong correlation between Thermal Design Power (TDP) and other attributes. This motivates us to utilize TDP for further analysis.

Since lithography has been chose as a CPU era, we are interested in the difference in the Thermal Design Power between the Lithography era.

Performing ANOVA test requires us to set the null hypothesis, which is

- $\mathbf{H_0}$. The mean of the `tdp` in each type of lithography is the same.
- $\mathbf{H_1}$. The mean of the `tdp` in each type of lithography is **not** the same.

We decide to remove some rows which doesn't make a big contribution to the dataset of `litho`.

```
             Df  Sum Sq Mean Sq F value  Pr(>F)
litho         6   97371   16229   11.18 2.8e-12 ***
Residuals  1783 2588198    1452
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We get the above result after removing rows where `litho` value equals to 28, 180 and 250. The reason to this is these values has small impact on the entire attribute, which may cause error in calculation.

The result is good enough since the p-value is less than 0.05, here we can reject the null hypothesis and accepts H_1 that there exists a pair of `litho` type so that their `tdp` mean is difference.

Normality
We perform to plot the **Q-Q Plot** to check its residuals.

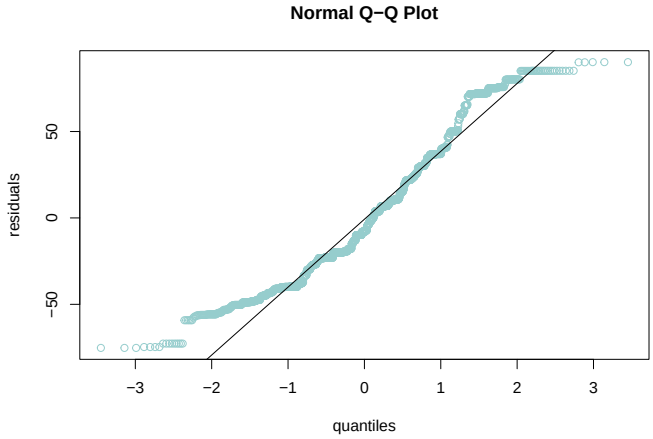


Figure 18: Normal Q-Q Plot

The QQ plot shows that most points align closely with a straight line, indicating that the residuals from the ANOVA model are normally distributed. This alignment suggests that the normality assumption of ANOVA is met, reinforcing the reliability of the statistical conclusions drawn from the model.

Performing **Shapiro-Wilk** test gives us the following result

```
Shapiro-Wilk normality test
data:  residuals(litho_anova_model)
W = 0.95555, p-value < 2.2e-16
```

The p-value once again is less than 0.05, it indicates that there is significant evidence to reject the null hypothesis that the data are normally distributed.

Homoscedasticity.
We use the **Bartlett's test** to test for equal variances and get the following result.

```
Bartlett test of homogeneity of variances
data:  tdp by litho
Bartlett's K-squared = 46.251, df = 6, p-value = 2.639e-08
```

Since the p-value of the Bartlett test is less than 0.05, it suggests that there is significant evidence to reject the null hypothesis that the variances of the groups are equal. In other words, it indicates that the variances

of the groups are significantly different from each other.

Kruskal-Wallis test

```
Kruskal-Wallis rank sum test
data:  tdp by litho
Kruskal-Wallis chi-squared = 66.49, df = 6, p-value = 2.14e-12
```

The p-value for the Kruskal-Wallis test is less than 0.05, which yeilds the result that different Thermal Design Power comes from different distributions.

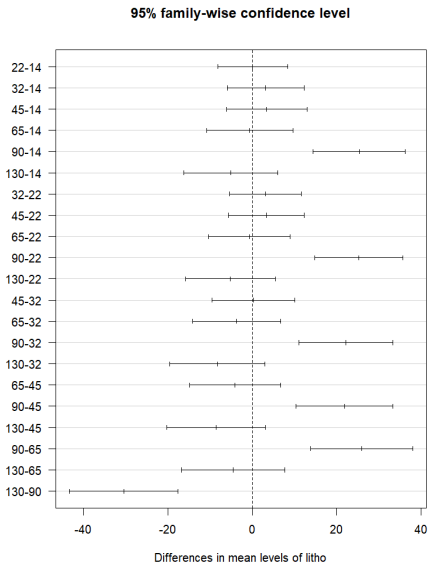
Tukey's range test

Finally we will analyse the result with a post hoc test to see which mean are different from each other. We first do a t-test using Hochberg's method to check if it effectively distinguishes values into distinct groups

```
Pairwise comparisons using t tests with pooled SD
data:  data$tdp and data$litho

    14       22       32       45       65       90
22  0.97     -        -        -        -        -
32  0.97     0.97     -        -        -        -
45  0.97     0.97     0.97     -        -        -
65  0.97     0.97     0.97     0.97     -        -
90  2.4e-10 2.9e-11 8.5e-08 3.7e-07 5.3e-09 -
130 0.97     0.97     0.43     0.43     0.97     7.3e-11
P value adjustment method: hochberg
```

The obtained result is promising, as it effectively distinguishes values into distinct groups. Comparable outcomes were achieved when employing the Tukey's HSD test below.



(a) Tukey HSD plot

```
Tukey multiple comparisons of means
   95% family-wise confidence level

Fit: aov(formula = tdp ~ litho, data = data_litho)

$litho
             diff        lwr        upr     p adj
22-14   0.08990679  -8.187139   8.366952 1.0000000
32-14   3.16448757  -5.981971  12.310946 0.9492441
45-14   3.46548078  -6.123147  13.054109 0.9376923
65-14  -0.59582223 -10.863495   9.671850 0.9999979
90-14  25.35848621  14.383476  36.333496 0.0000000
130-14 -5.13448691 -16.304802   6.035828 0.8246176
32-22   3.07458078  -5.432760  11.581922 0.9377023
45-22   3.37557399  -5.605458  12.356606 0.9254024
65-22  -0.68572902 -10.388422   9.016964 0.9999931
90-22  25.26857942  14.820231  35.716928 0.0000000
130-22 -5.22439370 -15.877707   5.428919 0.7757025
45-32   0.30099321  -9.487120  10.089106 1.0000000
65-32  -3.76030981 -14.214518   6.693899 0.9390613
90-32  22.19399863  11.044280  33.343717 0.0000001
130-32 -8.29897448 -19.640989   3.043040 0.3182195
65-45  -4.06130301 -14.904483   6.781877 0.9265750
90-45  21.89300543  10.377785  33.408226 0.0000005
130-45 -8.59996769 -20.301480   3.101544 0.3128248
90-65  25.95430844  13.867803  38.040814 0.0000000
130-65 -4.53866467 -16.802787   7.725458 0.9304967
130-90 -30.49297311 -43.355105 -17.630841 0.0000000
```

(b) Tukey HSD values

Figure 19: Tukey HSD

Dunn test
We utilize Dunn's test to pinpoint specific groups that exhibit significant differences from each other subsequent to conducting a non-parametric Kruskal-Wallis test.

	Comparison	Z	P.unadj	P.adj
1	130 - 14	-0.3786483	7.049491e-01	1.000000e+00
2	130 - 22	-1.2096563	2.264108e-01	1.000000e+00
3	14 - 22	-1.0459320	2.955924e-01	1.000000e+00
4	130 - 32	-1.9619511	4.976817e-02	1.000000e+00
5	14 - 32	-1.9704740	4.878408e-02	1.000000e+00
6	22 - 32	-1.1008883	2.709453e-01	1.000000e+00
7	130 - 45	-1.7667917	7.726312e-02	1.000000e+00
8	14 - 45	-1.7150018	8.634490e-02	1.000000e+00
9	22 - 45	-0.8670816	3.858973e-01	1.000000e+00
10	32 - 45	0.1612512	8.718956e-01	1.000000e+00
11	130 - 65	-0.9332387	3.506968e-01	1.000000e+00
12	14 - 65	-0.7027623	4.822039e-01	1.000000e+00
13	22 - 65	0.1485663	8.818958e-01	1.000000e+00
14	32 - 65	1.0337584	3.012491e-01	1.000000e+00
15	45 - 65	0.8511139	3.947061e-01	1.000000e+00
16	130 - 90	-6.6190072	3.616193e-11	7.594005e-10
17	14 - 90	-7.3717404	1.684148e-13	3.536710e-12
18	22 - 90	-6.9147479	4.686961e-12	9.842619e-11
19	32 - 90	-5.6397897	1.702580e-08	3.575418e-07
20	45 - 90	-5.5978444	2.170334e-08	4.557701e-07
21	65 - 90	-6.0968153	1.082024e-09	2.272251e-08

Figure 20: Dunn's test result

We can interpret the results of our data as follows:

1. Significant differences exist in the mean and median of Thermal Design Power between CPUs with larger lithography values (130, 90) and CPUs with smaller lithography values (14, 22, 32, 45, 65).
2. The mean TDP values of CPUs with lithography values of 14, 22, 32, 45, and 65 are relatively similar.
3. Lithography not only reflects the release date but also indicates changes in the TDP of CPU designs over time.

6.3 Multilinear Regression Model

6.3.1 Objective

We have chosen to predict Thermal Design Power using other attributes in our dataset. This approach allows us to understand the relationships between Thermal Design Power and these attributes, developing a reliable predictive model to capture underlying data patterns.

6.3.2 Model Definition

In our model, we have

$$\hat{Y} = \beta_0 + \beta_1\hat{X}_1 + \beta_2\hat{X}_2 + \beta_3\hat{X}_3 + \beta_4\hat{X}_4 + \beta_5\hat{X}_5 + \beta_6\hat{X}_6 + \varepsilon$$

where

- $\hat{Y}$ is the predicted Thermal Design Power (tdp) value.
- $\hat{X}_i$ where $i = \{1, 2, \ldots, 6\}$ are respectively the values for Launch Date (ldate), Recommended Price (rprice), Lithography (litho), Number of Cores (ncore), Base Frequency (bfreq) and Temperature (temp).

6.3.3 Model Fitting

We start by choosing the multilinear regression model to predict the value of Thermal Design Power base on some of other attributes in the dataset.

We've settled on an allocation of 80% of our dataset for training purposes, reserving the remaining 20% for testing. This split ratio has demonstrated its efficacy in facilitating accurate predictions for our specific tasks. We build the model using `lm()`.

```
split = sample.split(data, SplitRatio = 0.8)
training_set = subset(data, split == TRUE)
test_set = subset(data, split == FALSE)

regressor = lm(formula = tdp ~ ldate + rprice +
                               litho + ncore +
                               bfreq + temp,
               data = training_set)
summary(regressor)
```

6.3.4 Model Inferential

The `summary()` yields the result:

```
Call:
lm(formula = tdp ~ ldate + rprice + litho + ncore + bfreq + temp,
    data = training_set)

Residuals:
    Min      1Q  Median      3Q     Max
-61.174  -8.282   0.624   8.381  46.046

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  52.968145 968.359731   0.055    0.956
ldate        -0.022750   0.480148  -0.047    0.962
rprice        0.006886   0.001121   6.144 1.36e-09 ***
litho         0.710036   0.100612   7.057 4.15e-12 ***
ncore         5.285320   0.286114  18.473  < 2e-16 ***
bfreq        19.744026   0.975942  20.231  < 2e-16 ***
temp         -0.528536   0.048021 -11.006  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.96 on 690 degrees of freedom
Multiple R-squared:  0.7882,  Adjusted R-squared:  0.7863
F-statistic: 427.9 on 6 and 690 DF,  p-value: < 2.2e-16
```

We now take a look at some calculations made by the model:

- **Estimate.** This is the estimated coefficient for the multilinear regression model.
- **Std.Error.** The standard error for the training, this result seems to be fine since most errors are less than 1.
- **Adjusted R-squared.** The R^2 value of the training set. In this case it's `0.7863`, which is quite good as it can indicates a better fit of the regression model to the observed data.

To assess the predictive accuracy of this model in estimating Thermal Design Power (TDP), we intend to visualize the relationship between predicted and actual TDP values. This involves plotting both sets of

values on the same graph to visually compare their alignment. By doing so, we aim to evaluate the model's performance in forecasting TDP against the ground truth data.

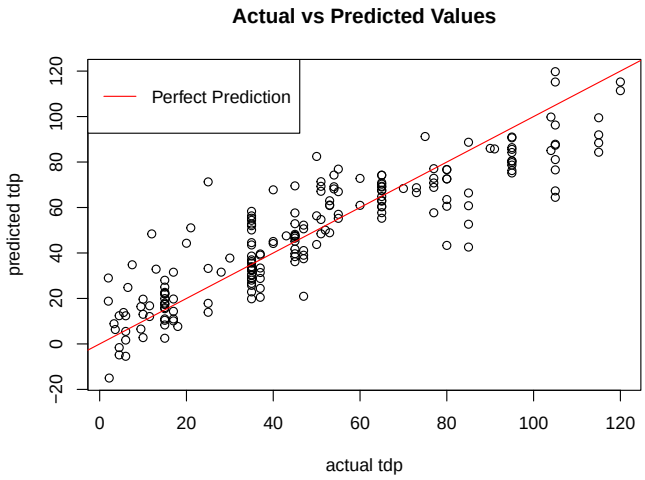


Figure 21: Result of multilinear regression

The plotting yields a quite satisfying result, as the majority points lies on the 'Perfect Prediction' line, which predicts exacts the value for Thermal Design Power. But in-fact, the accuracy for the model is not so high, just enough to fit the regression model.

```
> print(accuracy)
[1] 0.5057831
```

The reason for this can come from lack of information, as because we have removed many records with missing values.

6.3.5 Model Assumption

We now ensure conditions for applying the multilinear regression model, which are:

- Linearlity.
- Residual Errors have a Mean Value of Zero.
- Residual Errors have Constant Variance.
- The errors are normally distributed.

Linearity A linearity test confirms if the relationship between variables is linear, crucial for accurate linear regression models. It helps determine if linear modeling is appropriate or if alternative methods are needed for a better fit, ensuring the reliability of statistical analyses.

We can use the `linearHypothesis()` function to test for linearity. And get the following result

```
Linear hypothesis test

Hypothesis:
ldate = 0
rprice = 0
litho = 0
ncore = 0
bfreq = 0
temp = 0

Model 1: restricted model
Model 2: tdp ~ ldate + rprice + litho + ncore +
     bfreq + temp

  Res.Df    RSS Df Sum of Sq     F    Pr(>F)
1    894 920598
2    888 192604  6    727994 559.4 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
     '.' 0.1 ' ' 1
```

The hypothesis test returns the following values:

- F-test statistic: 559.4
- p-value: < 2.2e-16.

This particular hypothesis test uses the following null and alternative hypotheses:

- $\mathbf{H_0}$. All regression coefficients are equal to zero.
- $\mathbf{H_1}$. At least one regression coefficient is not equal to zero.

Since the p-value of the test is less than .05, we reject the null hypothesis.

It is encouraging to observe that the dataset supports the construction of a multilinear regression model. This outcome is promising for further detailed analyses and predictive modeling.

Residual Errors have a Mean Value of Zero

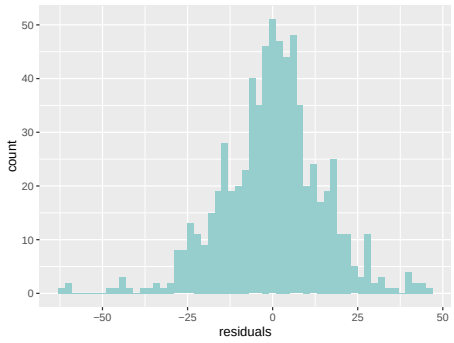


Figure 22: Residual error

The centered distribution of residual values around 0 indicates that the model's predictions are generally accurate, with an equal number of overestimations and underestimations. This suggests an unbiased model, which is desirable.

Residual Errors have Constant Variance

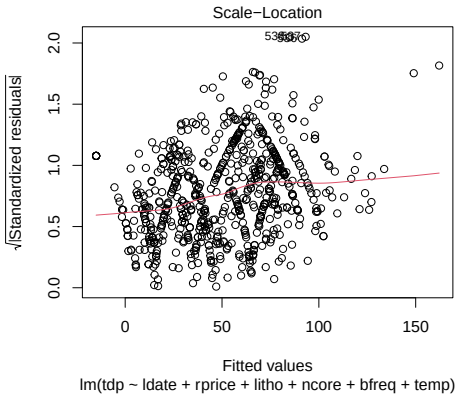


Figure 23: Variance error

We use a scale-location plot to assess the homoscedasticity (constant variance) assumption of the residuals. Also known as a spread-location or scale-residual plot, this diagnostic plot is commonly employed in regression analysis.
Observing the plot, we see that most points scatter evenly across the range of fitted values, which suggest that the assumption of homoscedasticity is met.

The Errors are Normally Distributed

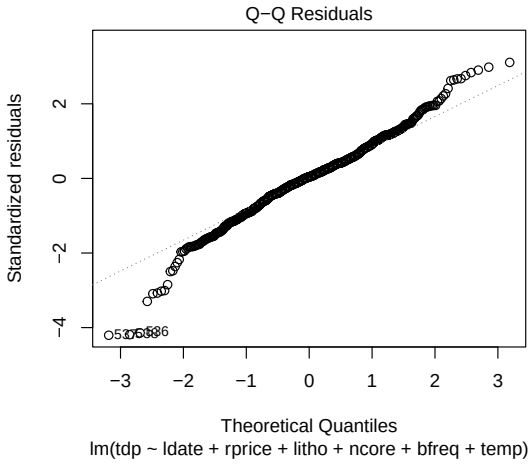


Figure 24: Normality check

As suggested in the previous section, we use the QQ-Plot to check for normality condition. Since most points fall approximately along the diagonal line, it suggests that the residuals are normally distributed, which is a desirable property for our statistical analyses.

Residuals vs Leverage plot

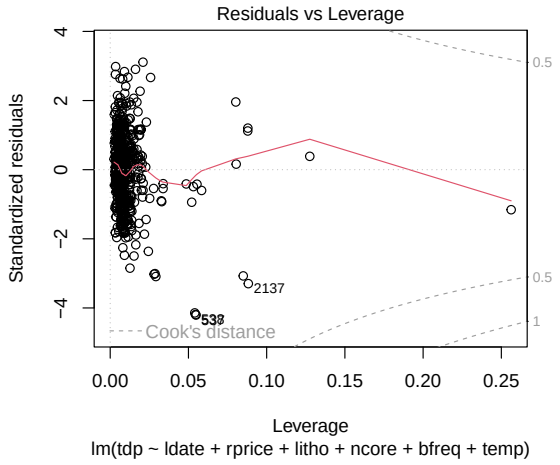


Figure 25: Residuals vs Leverage plot

A residuals vs. leverage plot is a type of diagnostic plot that allows us to identify influential observations in a regression model. In our particular model, we get the following plot.

The residuals vs. leverage plot suggests that if any point in this plot falls outside of Cook's distance (the black dashed lines) then it is considered to be an influential observation. In our case, we see that there are no such points which meets that condition, which shows that our dataset is good enough for multilinear regression model.

Given all above conditions have met, we can see that our multilinear regression model is a quite success.

7 Discussion and Extension

7.1 Discussion

Multiple linear regression is a powerful statistical technique for analyzing the relationship between multiple independent variables and a single dependent variable. Understanding its assumptions, limitations, and applications enables researchers and practitioners to apply it effectively to various real-world problems.

7.1.1 Advantage

Multiple linear regression offers interpretability, flexibility, and transparency, making it easy to understand the relationships between independent variables and the dependent variable. It supports statistical inference, hypothesis testing, uncertainty quantification, variable selection, and prediction tasks, making it a versatile and widely used statistical tool.

7.1.2 Disadvantage

Multiple linear regression, while powerful, has limitations. It assumes linearity, independence, and homoscedasticity, which may not always hold true. Sensitivity to outliers, multicollinearity, and overfitting further challenge its performance. Despite its utility, its inability to capture complex relationships and potential overfitting warrant caution, necessitating consideration of alternative approaches for robust analysis.

7.2 Extension

In our exploration of multilinear regression, we recognized the importance of meeting key assumptions for robust model performance. To overcome limitations associated with linear models and enhance our analysis, we are now employing polynomial regression.

We begin by utilizing **Multivariate Adaptive Regression Splines** (**MARS**) to determine the optimal degree of the polynomial and identify the most predictive attributes. This sophisticated approach allows us to refine our model-building process.

Subsequently, we implement **Polynomial Regression** to evaluate our model's efficacy in fitting the data. This approach enables us to capture more intricate relationships than traditional linear regression, providing a deeper understanding of the underlying patterns in our dataset.

7.2.1 Multivariate Adaptive Regression Splines

Multivariate adaptive regression splines (MARS) provide a convenient approach to capture the nonlinear relationships in the data by assessing cutpoints (knots) similar to step functions. The procedure assesses each data point for each predictor as a knot and creates a linear regression model with the candidate feature(s).

This procedure continues until many knots are found, producing a (potentially) highly non-linear prediction equation. Although including many knots may allow us to fit a really good relationship with our training data, it may not generalize very well to new, unseen data. Consequently, once the full set of knots has been identified, we can sequentially remove knots that do not contribute significantly to predictive accuracy. This process is known as "pruning" and we can use cross-validation, as we have with the previous models, to find

the optimal number of knots.

We first fit a direct engine MARS model with the `earth` package. And get the following output

```
Selected 15 of 16 terms, and 5 of 5 predictors
Termination condition: Reached nk 21
Importance: rprice, bfreq, temp, litho, ncore
Number of terms at each degree of interaction: 1 14 (additive model)
GCV 258.5607    RSS 213817.1    GRSq 0.8761123    RSq 0.8838447
```

All attributes demonstrate significant impact on predicting `tdp`, as evidenced by the inclusion of all five predictors in the model. With an R-squared value of 0.8838, the model explains 88.38% of the variance in tdp, indicating strong explanatory power without overfitting. These results affirm the effectiveness of our polynomial regression approach for capturing data patterns and its potential for predictive tasks.

We take a look at the **model selection** plot

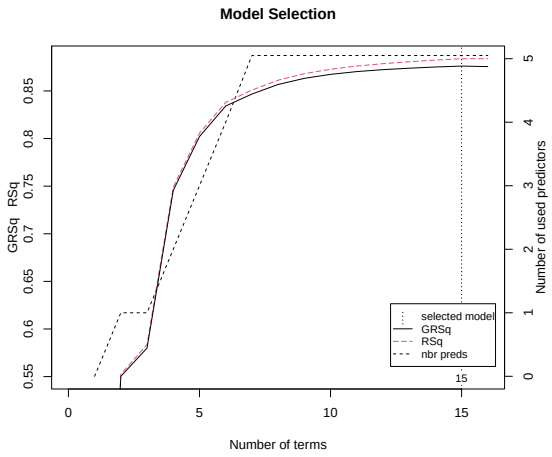


Figure 26: Model Selection

The model selection plot graphs the GCV R^2 (left-hand y-axis and solid black line) based on the number of terms retained in the model (x-axis) which are constructed from a certain number of original predictors (right-hand y-axis). The vertical dashed lined at 15 tells us the optimal number of terms retained where marginal increases in GCV R^2 are less than 0.001.

There are two important tuning parameters associated with our MARS model: the maximum degree of interactions and the number of terms retained in the final model. We need to perform a grid search to identify the optimal combination of these hyperparameters that minimize prediction error (the above pruning process was based only on an approximation of CV model performance on the training data rather than an exact k-fold CV process). As in previous chapters, we'll perform a CV grid search to identify the optimal hyperparameter mix. Below, we set up a grid that assesses 70 different combinations of interaction complexity (degree) and the number of terms to retain in the final model (nprune).

Our model provides the optimal combination includes second degree interaction effects and retains 23 terms. The cross-validated RMSE for these models is displayed as the output below; the optimal model's cross-validated RMSE was 688,112.

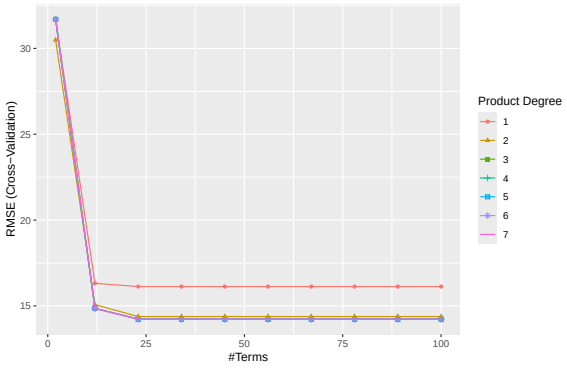


Figure 27: Root Mean Square Error

The output is as follow

```
  degree nprune  RMSE   Rsquared      MAE    RMSESD  RsquaredSD   MAESD
    3     23    14.229  0.9039819  10.63482 1.136289 0.01255743 0.7382791
```

The smallest root mean square is with degree 3, thus inspires us for using a polynomial of degree 3 for regression.

7.2.2 Polynomial Regression

Polynomial regression is a type of regression analysis used in statistics and machine learning to model the relationship between a dependent variable and one or more independent variables. Unlike simple linear regression, which assumes a linear relationship between the variables, polynomial regression allows for a more complex, nonlinear relationship by fitting a polynomial equation to the data.

After having the result of a polynomial of degree 3, we can start build our model. But first we would like to observe other results with different degree orders.

We start by preparing some materials for the model building section

```
df.shuffled <- data_poly[sample(nrow(data_poly)),]
K <- 15
degree <- 4
folds <- cut(seq(1,nrow(df.shuffled)),breaks=K,labels=FALSE)
mse = matrix(data=NA,nrow=K,ncol=degree)
```

We introduce a new term used in the building of the model.

K-Fold Validation
K-fold cross-validation is a widely used technique in machine learning for assessing the performance of a predictive model. The basic idea behind k-fold cross-validation is to divide the dataset into k subsets of

approximately equal size. Then, the model is trained k times, each time using $k-1$ subsets as the training data and the remaining subset as the validation data. Here are the steps used for the technique

1. **Data Splitting.** The dataset is randomly partitioned into k equal-sized folds.
2. **Model Training and Validation.** The model is trained k times, with each iteration using a different fold as the validation set and the remaining k-1 folds as the training set.
3. **Performance Evaluation.** After each training iteration, the model's performance is evaluated on the validation set using a chosen evaluation metric (such as accuracy, precision, recall, or F1-score).
4. **Average Performance.** The performance metrics obtained from each iteration are averaged to obtain a single performance estimate for the model.

To ensure that we fit a model that is flexible but not too flexible, we use k-fold cross-validation to find the model that produces the lowest test MSE.

In this case we try `15` folds, which let the model trains 15 times. After finish running the model, we print out the mean square error for each polynomial degree, and get the following result

```
[1] 489.2938 337.6094 285.7321 280.2628
```

The lowest value of `MSE` is 280.2628, which belongs to a polynomial with degree 4. We have determined the degree of the regression model be 4.

Before plotting the final result, we should analyze some values of the model, which yields the result

```
Call:
lm(formula = tdp ~ poly(rprice, 4, raw = TRUE) + poly(bfreq,
    4, raw = TRUE) + poly(litho, 4, raw = TRUE) + poly(ncore,
    4, raw = TRUE) + poly(temp, 4, raw = TRUE), data = data_poly)

Residuals:
    Min      1Q  Median      3Q     Max
-43.778 -10.139  -0.737   8.877  64.771

Coefficients:
                             Estimate Std. Error t value Pr(>|t|)
(Intercept)                -1.243e+03  7.847e+02  -1.584  0.11337
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 15.67 on 1082 degrees of freedom
Multiple R-squared:  0.8833,  Adjusted R-squared:  0.8812
F-statistic: 409.6 on 20 and 1082 DF,  p-value: < 2.2e-16
```

The R-squared value we obtained is 0.867, indicating that the model explains 86.7% of the variance in the data. Additionally, the p-value, which is much less than 0.05, suggests a highly significant relationship between the predictor variables and the response variable. These results indicate a nearly perfect fit of the model to the data.

However, many coefficients in this result do not achieve high significance levels, which could indicate a potential issue with overfitting. Overfitting occurs when the model fits the training data too closely, capturing noise instead of the underlying pattern. As evidenced by the plot below, the data points closely follow the regression line, suggesting that the model may be overfitted. To further explore this, we can examine the relationship between the variables `tdp` (Thermal Design Power) and `rprice` (Recommended Price).

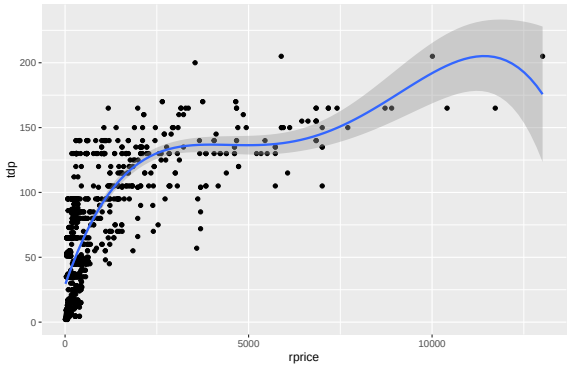


Figure 28: Polynomial Regression Model

In real statistics, this is not considered a good result, since data may vary. Which is why we return to using a polynomial of degree 4. The code for building the model is similar to the process so far. We only need to replace the degree 4 with 3. The summary of the model yields

```
Call:
lm(formula = tdp ~ poly(rprice, 3, raw = TRUE) + poly(bfreq,
    3, raw = TRUE) + poly(litho, 3, raw = TRUE) + poly(ncore,
    3, raw = TRUE) + poly(temp, 3, raw = TRUE), data = data_poly)

Residuals:
    Min      1Q  Median      3Q     Max
-44.775 -10.562  -0.671   9.077  64.716

Coefficients:
---
Residual standard error: 15.8 on 1087 degrees of freedom
Multiple R-squared:  0.8809,  Adjusted R-squared:  0.8792
F-statistic: 535.8 on 15 and 1087 DF,  p-value: < 2.2e-16
```

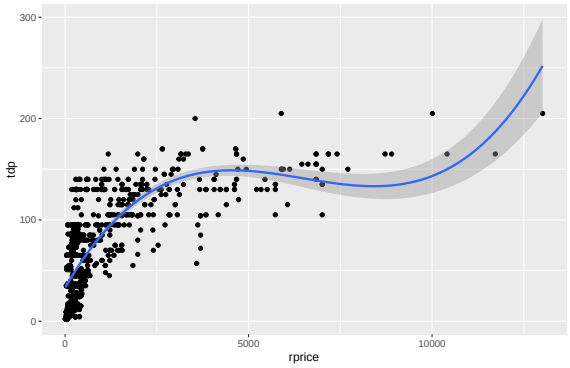


Figure 29: Polynomial Regression Model

The new model shows a more realistic prediction of the model. Thus this can be stated as a good model for the data.

7.2.3 Smoothing Regression

Since smoothing regression primarily focuses on the relationship between two variables, we have chosen to use `tdp` and `rprice` as our attributes. Our objective is to predict Thermal Design Power based on its Price. After fitting the model, we get the following result. This shows that the model has an EDF of $\nu_\lambda = 102.5782$.

```
Call:
ss(x = data.s$rprice, y = data.s$tdp, all.knots =
    TRUE)

Smoothing Parameter  spar = -1.129277   lambda =
    4.135922e-16
Equivalent Degrees of Freedom (Df) 306.7266
Penalized Criterion (RSS) 346712.2
Generalized Cross-Validation (GCV) 603.143
```

To get more information of the model, we run the `summary(mod.ss)` and get the output as

```
Call:
ss(x = data.s$rprice, y = data.s$tdp, all.knots = TRUE)

Residuals:
      Min        1Q    Median        3Q       Max
-64.56961  -7.47698  -0.00171   4.26232  75.30545

Approx. Signif. of Parametric Effects:
            Estimate Std. Error t value  Pr(>|t|)
(Intercept)    147.2      4.207  35.000 0.000e+00 ***
x              202.8     26.171   7.748 2.842e-14 ***
---
Approx. Signif. of Nonparametric Effects:
             Df Sum Sq Mean Sq F value Pr(>F)
s(x)      304.7 525287  1723.8   3.959      0 ***
Residuals 796.3 346712   435.4
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 20.87 on 796.3 degrees of freedom
Multiple R-squared:  0.8479,    Adjusted R-squared:  0.7893
F-statistic: 3.946 on 305.7 and 796.3 DF,  p-value: <2e-16
```

The R-squared value is approximately 0.8, which is fine since this value can control of overfitting estimations. Finally we get plot the model to see the fitness of our model.

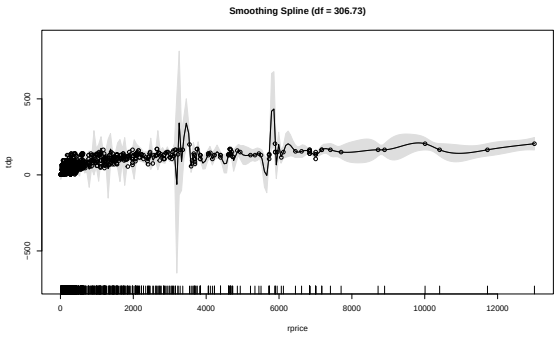


Figure 30: Smoothing Spline

The gray shaded area denotes a 95% Bayesian "confidence interval" for the unknown function.

8 Source

8.1 Code

All the code of this report can be found at *https://github.com/thaiquangphat/probability-and-statistics*.

8.2 Dataset

The dataset can be retrieved from the following locations:

1. Kaggle. *https://www.kaggle.com/datasets/iliassekkaf/computerparts/data*
2. GitHub. *https://github.com/thaiquangphat/probability-and-statistics/tree/main/Dataset*

References

[1] Douglass C.Montgomery, George C. Runger, *Applied Statistics and Probability for Engineers*, WILEY, 2013

[2] Alexis Dinno, *Nonparametric pairwise multiple comparisons in independent groups using Dunn's test*, The Stata Journal (2015) 15, Number 1, pp. 292-300

[3] Bradley Boehmke & Brandon Greenwell, *Multivariate Adaptive Regression Splines*, 2020-02-01
Retrieved from: https://bradleyboehmke.github.io/HOML/mars.html

[4] BIOST, *Polynomial regression*, February 5, 2004.
Retrieved from: https://courses.washington.edu/b515/l10.pdf

[5] Nathaniel E. Helwig, *Smoothing Spline Regression in R*, Department of Psychology & School of Statistics University of Minnesota, January 04, 2021
Retrieved from: http://users.stat.umn.edu/helwig/notes/smooth-spline-notes.htmt#model-form

[6] *The Comprehensive R Archive Network*
Retrieved from: https://cran.r-project.org/