Laptop Specs in Relation to Price

A. Otis & H. Yu Chen

COMP 4442-2

June 9th, 2022

Data Source

The **data was discovered on Kaggle**, **the primary resource** being scrapped **data** from flipkart.com; **collected from a chrome extension app** called **“Instant Data Scrapper”**.

Included are relevant to factors that are suggested to affect laptop prices, such as company name and owned laptop brands, the price of the laptop when first released and later in product’s life, among other tech related specifications.

Responses in columns **were transformed into multivariate** (i.e. 1, 2, 3, ..., etc.) **form**. Additionally, **another column is added for** a **logistic transformation of** the **response variable**.

Raw Set:
Observations = 896
Attributes = 34

→

Analysis ready Set:
Observations = 896
Attributes = 35

Bayesian Data Analysis, in Principle

There exists **two established frameworks** regarding the field of statistics, **Frequentist and Bayesian**. Frequentist being the framework we are now all well familiar with (e.g p-values and point estimate for variable of interest as a result), **our group is utilizing the Bayesian framework, which results in a distribution**.

$$\boldsymbol{Bayes\ Theorem}: \ \boldsymbol{P(A|B)\ \alpha\ P(B|A)P(A)}$$

$$Where,$$

$$P(A|B) \rightarrow Posterior\ Distribution$$

$$P(B|A) \rightarrow Likelihood\ Distribution, \quad assuming\ event\ A\ has\ occured, what\ is\ probability\ of\ event\ B?$$

$$P(A) \rightarrow Prior\ Distribution, \quad Inital\ guess\ of\ the\ variable\ of\ interest$$

A resulting posterior distribution can be interpreted as a report on both the level certainty and uncertainty (i.e. $\pm 0, \pm 1, \ldots, \pm n\ standard\ deviations$)regarding the probability of an event and model parameters

Research Question

What is the probability of a laptop’s market price
Given various of commonly used specifications in the industry?

Specification examples:

- Ssd
- Hdd
- Ram

Bayesian Regression

To get our **Prior Distribution**, we will **simply run a regular multiple regression**. Without industry expert input, this type of prior is commonly referred to as a “non-informative prior”

$$y_i = (\beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n) + (\varepsilon_i) \quad , for\ data\ (x_n, y_n)$$

$$Where, \quad \varepsilon_i \leftarrow Noise$$

To get our **Likelihood Distribution** the model runs thousands of samples, each with their own likelihood parameter; which together create a distribution of the likelihood parameters.

Specifically, we will be **utilizing the JAGS package.**

Data Satisfaction

Predictor Variables:

ram_gb
ssd
hdd

Response Variable:

log.latest_price

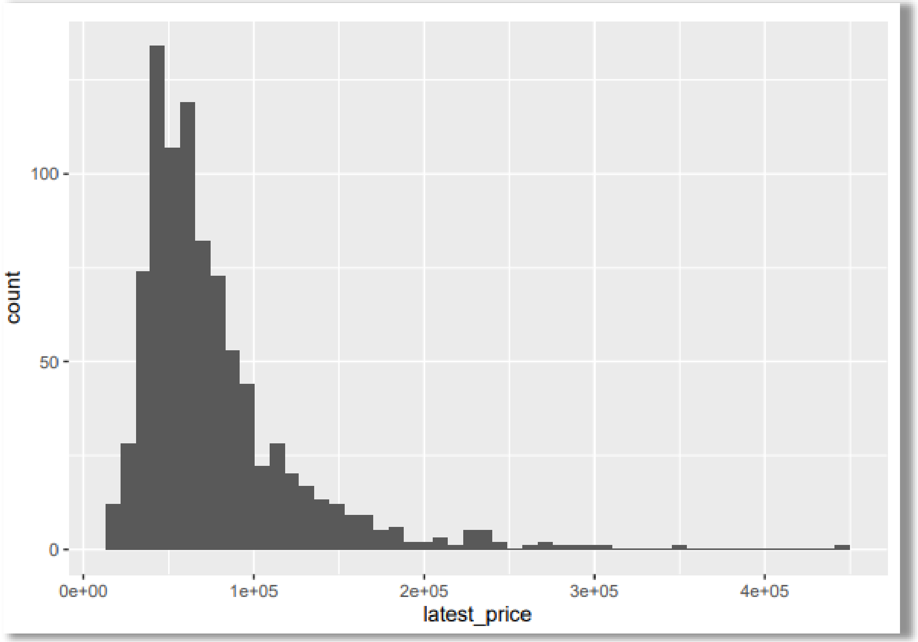
1. <u>Linearity</u>
   (i.e.) Linear relationship between predictors and response

2. <u>Homoscedasticity</u>
   (i.e.) Residuals vs Fitted show that variance remains about the same for any value of predictor variables

3. <u>Independence</u>
   (i.e.) Residual vs Factor Plot shows random dispersal about a horizontal trend line

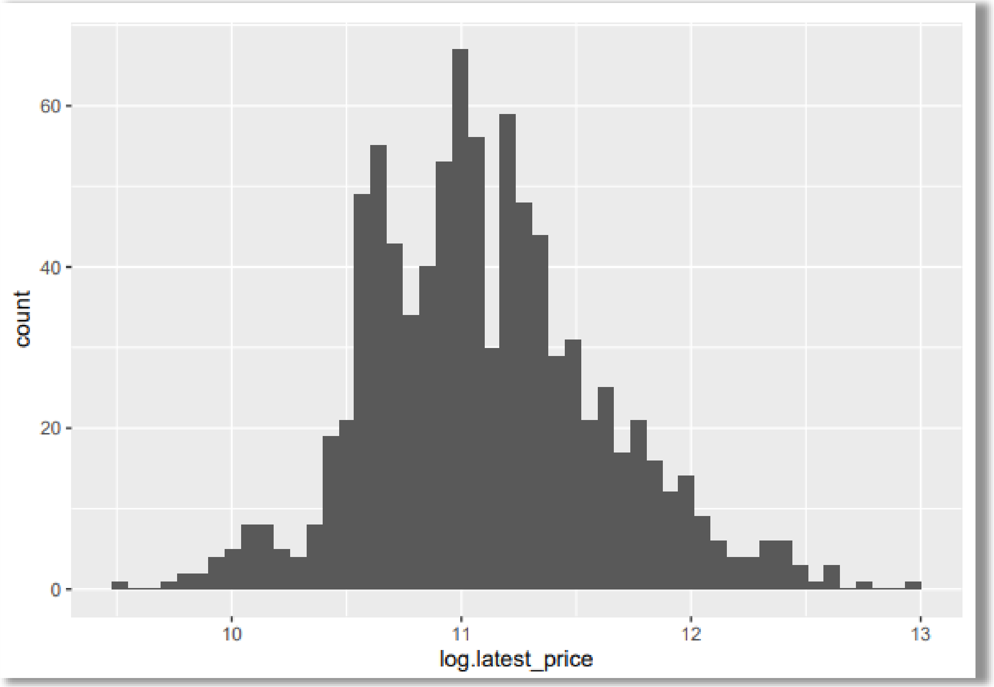
4. <u>Normality</u>
   (i.e.) Residuals provide a well fit QQ-plot

Exploratory Analysis



Non-Transformed Response

Log transformation



Transformed Response

Algorithm: Markov chain Monte Carlo (MCMC)

Imagine a target distribution you want to analyze, have data on it, but can no longer collect the data? The solution would be to use the Markov Chain Monte Carlo (MCMC). Thus, **MCMC is simply a algorithm for sampling from a distribution.**

One of the most common ways MCMC is used is to draw samples from the **posterior probability distribution** of some model in Bayesian inference

Regarding Analysis,
Four Markov Chains are set up for the predictors, “brand”, “ram_gb”, “hdd”, “ssd”

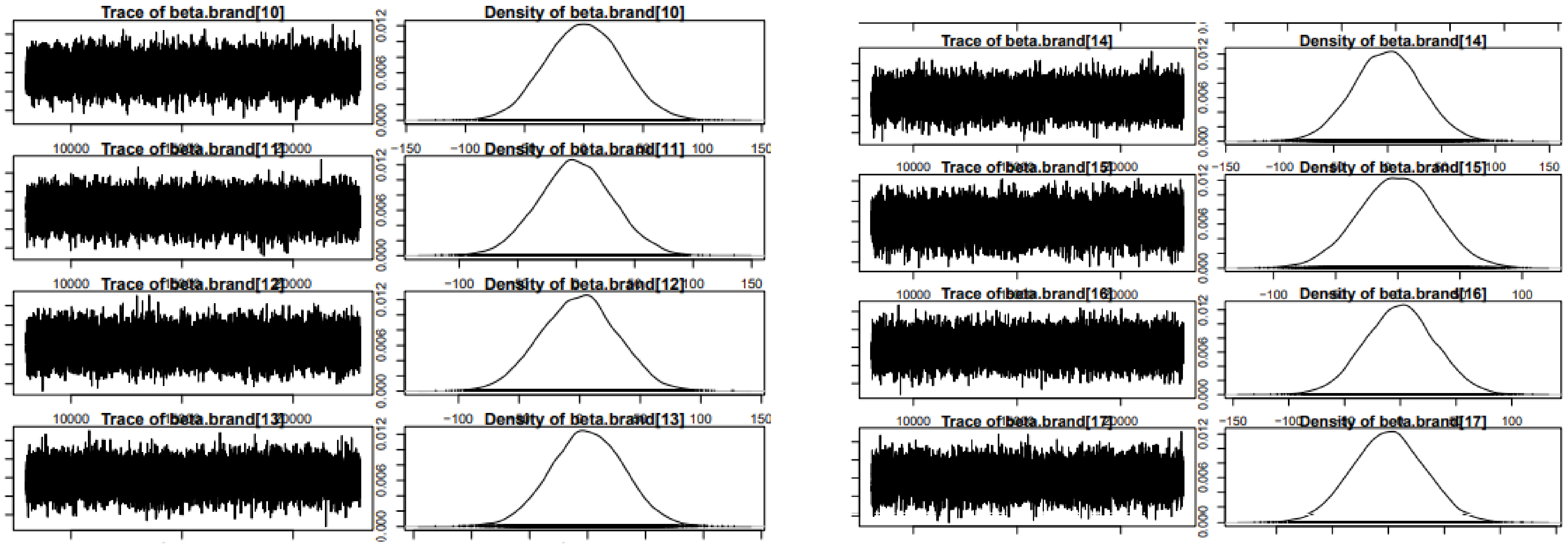
Run MCMC Sampler

```r
update(mod1, 8000)

mod1_sim <- coda.samples(model = mod1,
                         variable.names = c("beta0",
                                            "beta.brand[1]",
                                            "beta.brand[n2]",
                                            "beta.brand[3]",
                                            "beta.brand[4]",
                                            "beta.brand[5]",
                                            "beta.brand[6]",
                                            "beta.brand[7]",
                                            "beta.brand[8]",
                                            "beta.brand[9]",
                                            "beta.brand[10]",
                                            "beta.brand[11]",
                                            "beta.brand[12]",
                                            "beta.brand[13]",
                                            "beta.brand[14]",
                                            "beta.brand[15]",
                                            "beta.brand[16]",
                                            "beta.brand[17]",
                                            "beta.brand[18]",
                                            "beta.brand[19]",

                                            "beta.ram_gb[1]",
                                            "beta.ram_gb[2]",
                                            "beta.ram_gb[3]",
                                            "beta.ram_gb[4]",
                                            "beta.ssd[1]",
                                            "beta.ssd[2]",
                                            "beta.ssd[3]",
                                            "beta.ssd[4]",
                                            "beta.ssd[5]",
                                            "beta.ssd[6]",
                                            "beta.ssd[7]",
                                            "beta.ssd[8]",
                                            "beta.hdd[1]",
                                            "beta.hdd[2]",
                                            "beta.hdd[3]",
                                            "beta.hdd[4]"),
                         n.iter = 15000)
```

Burn-in (i.e. iterations) is a technique that helps improve the outcome quality.

the goal is to keep the model outcomes that closely match the real word data to build our **posterior distribution** (i.e. confirmation if the MCM chains have converged)

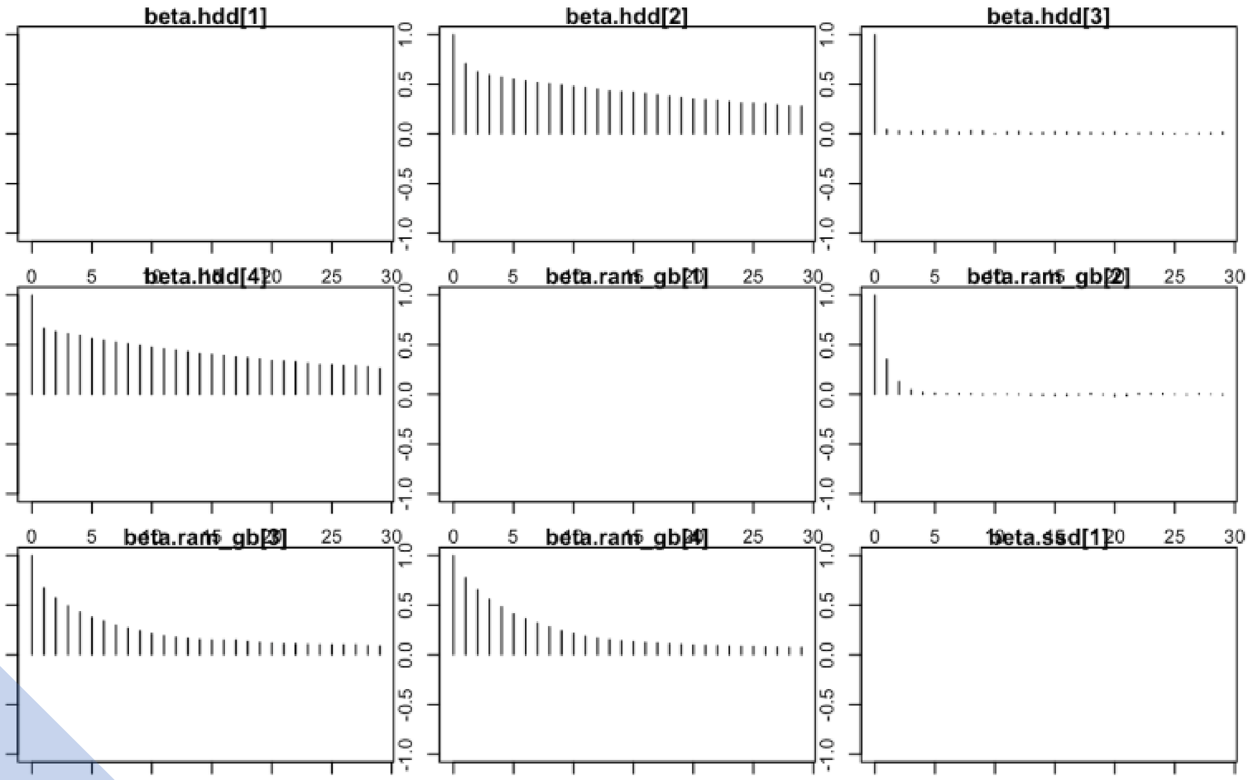
MCMC Diagnostics (Posterior Distributions)



In trace plots, we want to try to **avoid any flat areas or too many consecutive steps in one direction**, indicative of chains not converging.

Posterior distributions look relatively smooth and the mixing among chains, all good signs for convergence!

MCMC Diagnostics

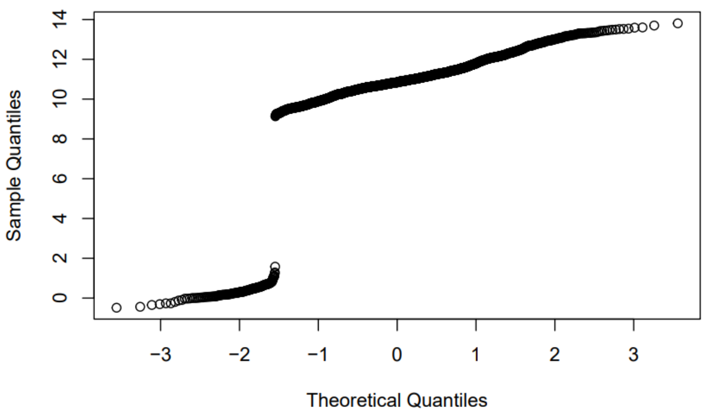


Another way to check for convergence is to look at the autocorrelations between the samples returned by our MCMC.

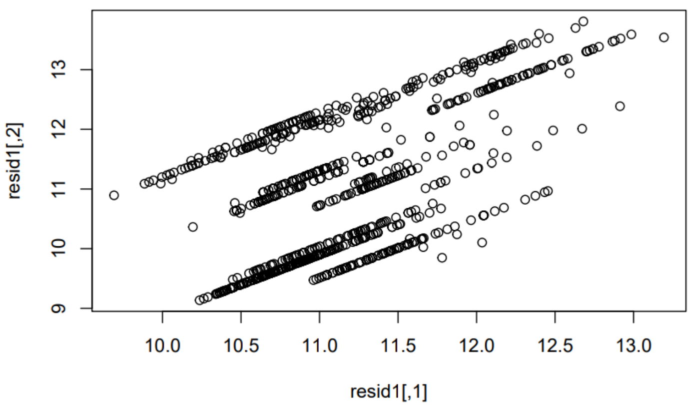
The lag-k autocorrelation is the correlation between every sample and the sample k steps before. This autocorrelation should become smaller as k increases, (**i.e. samples can be considered as independent**.)

Model Validity w/ Brand Included

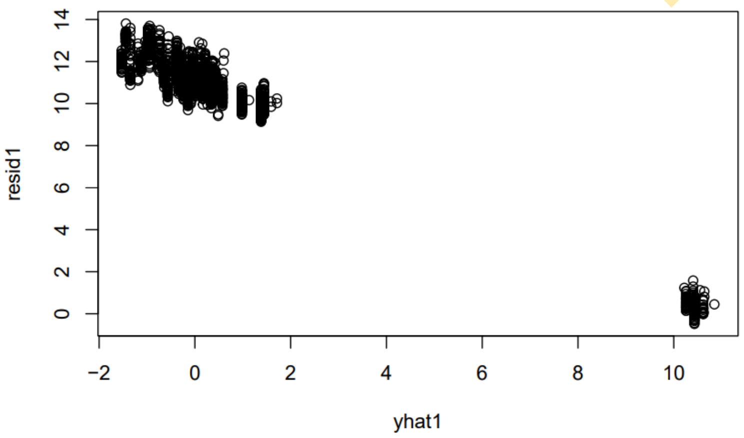
- Below are Diagnostic plots of the Bayesian Regression model fit on the data set, with <u>inclusion of the variable “Brand”</u> from a vector of predicted model values



Check Normality
- QQ-Plot



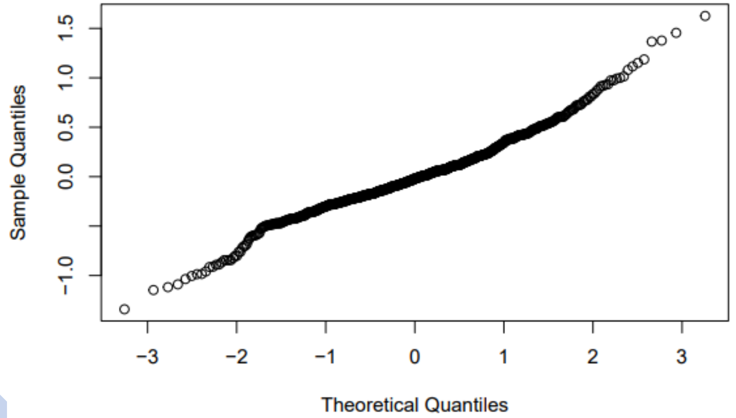
Check Independence
- Residuals vs Factor



Check Linearity
- Residuals vs Fitted plot

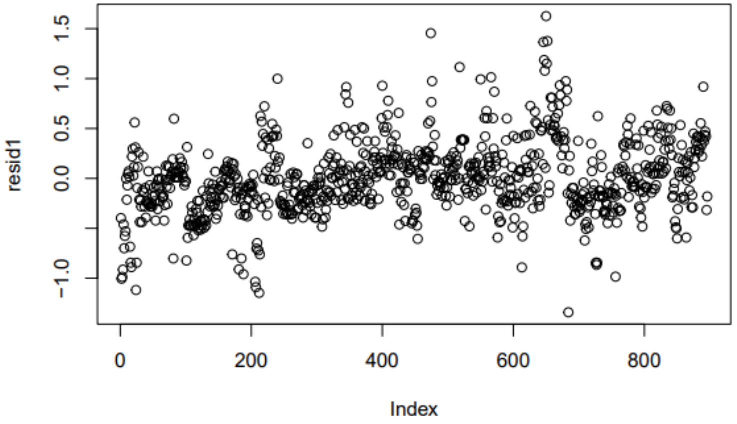
Model Validity w/o Brand Included

Below are Diagnostic plots of the Bayesian Regression model fit on the data set, excluding the variable "Brand" from a vector of predicted model values



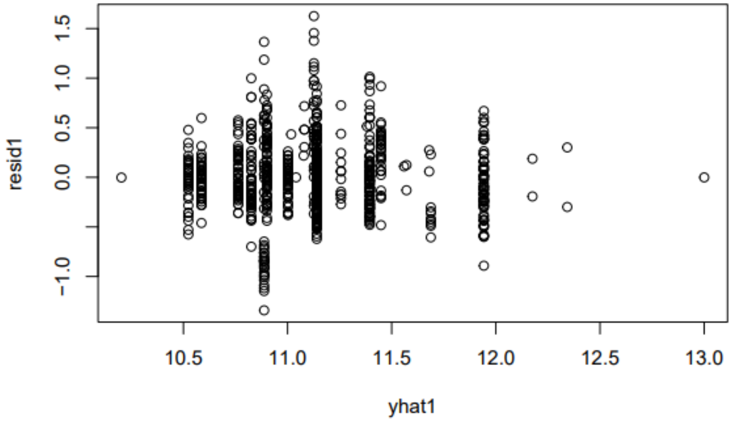
Check Normality

- QQ-Plot



Check Independence

- Residuals vs Factor



Check Linearity

- Residuals vs Fitted plot

Conclusions

The Bayesian regression **model with brands included do not fit our data** well. However, the **Bayesian model with brands excluded did fit the data well**, and therefore would result in more valid results/interpretations.

It is possible that the brand portion of the data is oversaturated with one type of response (e.g. majority of responses consisting of the most popular products at the time).

Thus, the **model with brands included could be improved by the following**.

- The removal of brands that have no significant effect (i.e. MCM chains related to that variable not converging).
- Re-specifying the parameters for the posterior distribution
- More data can be collected regarding brands and added into the model to see if more valid results come out.

Recall, Research Question: What is the probability of a laptop's market price. Given various of commonly used specifications in the industry?

While our presentation does not answer this question, it does provide a great starting point in the form of a model that fits the data. We recommend any further study on the topic follow up with predictions made with the Bayesian Regression model

References

Videos:

1. *Bayesian Modeling with R and Stan (Reupload)*. (2018, November 15). [Video]. YouTube.
2. *Bayesian Statistics - Regression with JAGS Part 3*. (2021, May 24). [Video]. YouTube.

Literature/Academic:

1. A. (2022, May 16). *Bayesian Statistics Overview and your first Bayesian Linear Regression Model*. Towardsdatascience. Retrieved June 6, 2022.
2. Durso, C. (2022), COMP4442-2: Advanced Probability and Statistics 2.
3. Falster, D. F. R. (2013, June 10). *Markov Chain Monte Carlo - Nice R Code*. Github. Retrieved June 1, 2022.
4. Karajannis, N. (2017, November 20). *RPubs - Bayes Regression using JAGS*. Rpubs.Com. Retrieved May 28, 2022.
5. Htoon, K. S. (2021, December 13). *Log Transformation: Purpose and Interpretation - Kyaw Saw Htoon*. Medium. Retrieved June 6, 2022.
6. Sbnfk. (2019). *mcmc_diagnostics.utf8.md*. Github. Retrieved June 2, 2022.

Data:

1. *Laptop Specs and latest price*. (2022, April 3). Kaggle.