Leeds University Business School

UNIVERSITY OF LEEDS

Dissertation/Project Coversheet

Student ID Number:	2	0	1	5	7	1	8	2	7
Student Name	Suresh Balaji Srinivas								
Module Code:	LUBS5579M								
Programme of Study:	Business Analytics and Decision Sciences								
Supervisor:	Prof. Wessam Abouarghoub								
Title:	Predicting smartphone price using DNN and predicting the SoC price using Multivariate Linear Regression								
Declared Word Count:	9027								

Please Note:

Your declared word count must be accurate, and should not mislead. Making a fraudulent statement concerning the work submitted for assessment could be considered academic malpractice and investigated as such. If the amount of work submitted is higher than that specified by the word limit or that declared on your word count, this may be reflected in the mark awarded and noted through individual feedback given to you.

It is not acceptable to present matters of substance, which should be included in the main body of the text, in the appendices ("appendix abuse"). It is not acceptable to attempt to hide words in graphs and diagrams; only text which is strictly necessary should be included in graphs and diagrams.

By submitting an assignment you confirm you have read and understood the University of Leeds **Declaration of Academic Integrity** (http://www.leeds.ac.uk/secretariat/documents/academic_integrity.pdf).

Table of Contents

List of Figures

Abstract

In this project, we have first scrapped the required data from websites like Intel, Geekbench 5 and GSMArena using a python-based web scraping tool called scrapy. Once we have the data, we clean the data to perform the analysis on the data. This data is used to predict the smartphone price and the SoC price. These prices are predicted using Deep Neural Network and Multivariate Linear Regression. With the help of this, we want to help the consumer in making a better choice in purchasing their smartphone. With the SoC price prediction the consumer can actually know the price of the chipset which the manufacturers do not reveal to anyone, this data can be used to better understand the cost-to-performance graph.

Chapter 1: Introduction

1.1 Background

The first smartphone was discovered in the year 1984, by IBM, it did include a touch interface but was a keypad that could be used to access the basic applications that were used at that time like the calendar, email, message, and phone app to make calls. Similarly, there were quite a few phones that were launched these were known as 1G phones or first-generation phones. The actual smartphone revolution did not start until the launch of the first iPhone by Steve Jobs. This revolutionized the entire smartphone industry because of the larger touchscreen, and in the past few years people could only use the watered-down version of the internet with their smartphones, but after the release of iPhone 1 people could use the internet on their smartphones as they use on the desktop. The Apple iPhone 1 was launched in the June of 2007. This was a year ahead of the first android smartphone. The processor that was used by Apple was manufactured by Samsung. The first android smartphone was launched in the year 2008 in the month of September. It was launched by HTC. It was a collaboration between T-Mobile and HTC. Apple relied on Samsung for their chips until they started manufacturing their own SoCs. It was the iPhone 4 that got the chips from Apple. The performance of Apple silicon was way better than the ones that android phones use. Most android phones use chips that are manufactured by Snapdragon.

Nowadays there are only two different types of OS, iOS for Apple iPhones and android for all other smartphones. The android phone companies do release their own flavour of the OS, it is basically just a skin on top of the Android versions that Google releases. For instance, Samsung uses One OS, OnePlus uses Oxygen OS, Xiaomi uses MIUI, etc. The screens that are used on smartphones have drastically improved in the last decade from TFT panels to AMOLED panels and from 60Hz refresh rate screens to 90Hz and then to 120Hz. In the year 2021, we saw the introduction of screens with a variable refresh rate. The storage size of phones has also increased drastically in the past decade. Increasing from 8GB to 2TB. The cameras in smartphones have also improved drastically. The price of smartphones has also varied drastically with the betterment of the specifications of the smartphones. The flagship smartphones these days cost around 1000$ to 1200$ which is a lot. These prices are skyrocketing these days and not everyone is able to afford these phones.

1.2 Research Objectives

The main objective is to collect the data of all the mobile phones and to use multivariate linear regression to find the cost of the SoC using independent variables like the type of GPU used, number of cores in the CPU, number of high-performance cores, number of efficiency cores, etc. to determine the price of the chip. Then we use Deep Neural Network to find the price of the smartphone using independent variables like the RAM size, battery capacity, storage size, number of cameras, the megapixels of the camera, display size, display resolution, display refresh rate, etc.

Chapter 2: Literature Review

The following literature review is divided into three parts, the background of the smartphone and its components, the machine learning algorithms that are used for predicting the price of the smartphones and the processors, and finally the price prediction.

2.1 Smartphones

Smartphones can be considered a boon or a bane to society, it depends on how we utilize the phone. As said by Arora, Srivastava and Garg, 2020, we can say that the price of a smartphone is something that the consumer will look into before deciding on whether they should get the phone or not. Price is the first thing that we look at before looking at the specifications of the phone like the storage size, the number of cameras, the RAM, the processor that is used, the battery size, etc. But as said by Chandrashekhara, Thungamani, Babu and Manjunath, 2019 "In current market situations, customers do not purchase products based on price alone." The top five smartphone companies based on their market share are Samsung, Apple, Oppo, Vivo and Xiaomi. Many smartphone companies have to come up with new features and technology to have a good market share and have an edge over their competitors as said by Chandrashekhara, Thungamani, Babu and Manjunath, 2019. This is where all the data can be used by the company to figure out what kind of innovations can be done to gain global market share.

As said by Arora, Srivastava and Garg, 2020, there are a lot of smartphones that are being bought and sold every single day. This shows how important smartphones have become in the consumer's everyday life. These smartphones are like a key that opens the door to the world wide web where people can get all sorts of information.

2.1.1 Smartphone Components

Smartphones constitute various important components that when brought together can be made into a smartphone. Various components make up the smartphone like the display, the processor, the cameras, the battery, the sensors and the operating system.

The display is one of the important components of the smartphone since that is the one that is used by the customers, it is the main thing that people use to interact with their smartphone. The two most common types of displays that are used in smartphones are:

i. Liquid Crystal Display (LCD)
ii. Organic Light Emitting Diode (OLED)

The processors used in smartphones have gained a lot of importance in the past few years and have significantly improved their computational calibre as said by Iyer and Pawar, 2019. In this paper, they have used data from the past from companies like AMD, ARM and Qualcomm. Most smartphones use chips manufactured by Media Tek, Qualcomm, Apple, Samsung, Unisoc and HiSilicon.

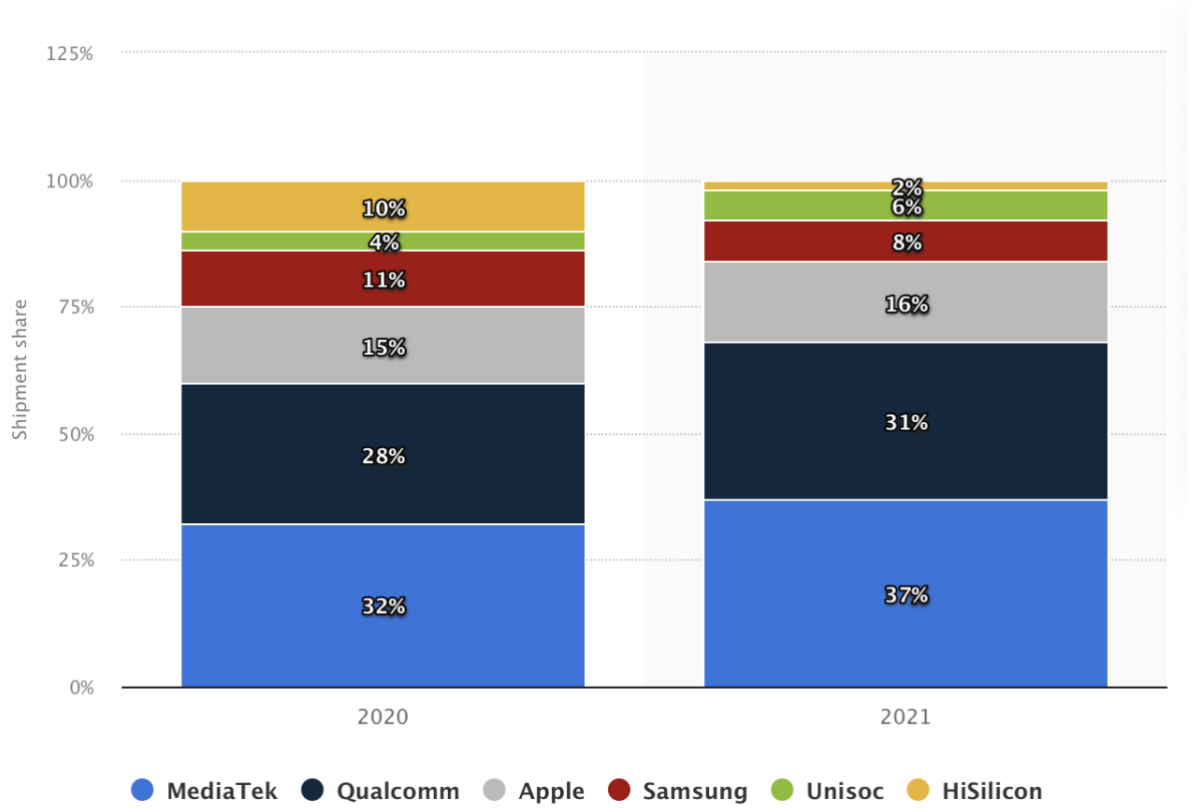

Figure 1 Global Market Share of SoCs Manufactured in the Year 2020 and 2021

The above graph by Thomas Alsop (2022) shows the market share of the chips that are manufactured by different companies that are used in smartphones for the years 2020 and 2021.

The camera quality in smartphones has seen exponential growth in the past few years. Some companies like Samsung, Vivo, Oppo, etc. have increased the megapixels of their cameras from 8MP to 24MP to 48MP to 108MP in the past few years while there are a few companies like Google and Apple that have not improved the megapixels of their camera array but have heavily invested in the image processing and have provided some of the best quality images. The Pixel lineup of Google is known for their amazing photo quality, while Apple's iPhones use 12MP camera arrays they are known to have the best video quality as compared to any smartphones that are present.

The battery is also important in the smartphone as it is the lifeline of the phone. The most commonly used types are Lithium-ion and Lithium-polymer. The capacity of these batteries is usually measured in milli amp hours (mAh). The higher the mAh the larger the capacity of the battery and it lasts longer.

Almost all the smartphones that are getting released these days have multiple sensors that are included in them like gyroscope, GPS, accelerometer, compass, proximity sensor, etc. Lane, et.al.,2010.

The operating system is the software that helps us get all the hardware together for the consumer to use the phone. The two main operating systems are Android and iOS. All Apple-manufactured smartphones use iOS while the other manufacturers use Android as their base operating system but since android is open source, unlike iOS the manufacturers of android phones generally put a skin on top of the base android software some of the examples are One UI by Samsung, Oxygen OS by Oneplus, etc.

2.2 Machine Learning Algorithms

A fresh wave of publicity has recently been focused on machine learning (ML) and artificial intelligence, which has been sparked by the enormous and continuously growing amounts of data and computing power as well as the development of better learning algorithms (Badillo et al., 2020). Arthur Samuel described artificial intelligence as a term in 1959, "Field of study that gives computers the ability to learn without being explicitly programmed". The reason why we need to use machine learning algorithms is that sometimes we will not be able to understand the patterns and the trends of the data because the human brain needs a lot of time to process and understand this data but a computer can do it a lot quicker (Mahesh, 2018).
There are two types of machine learning algorithms they are Unsupervised and Supervised learning.

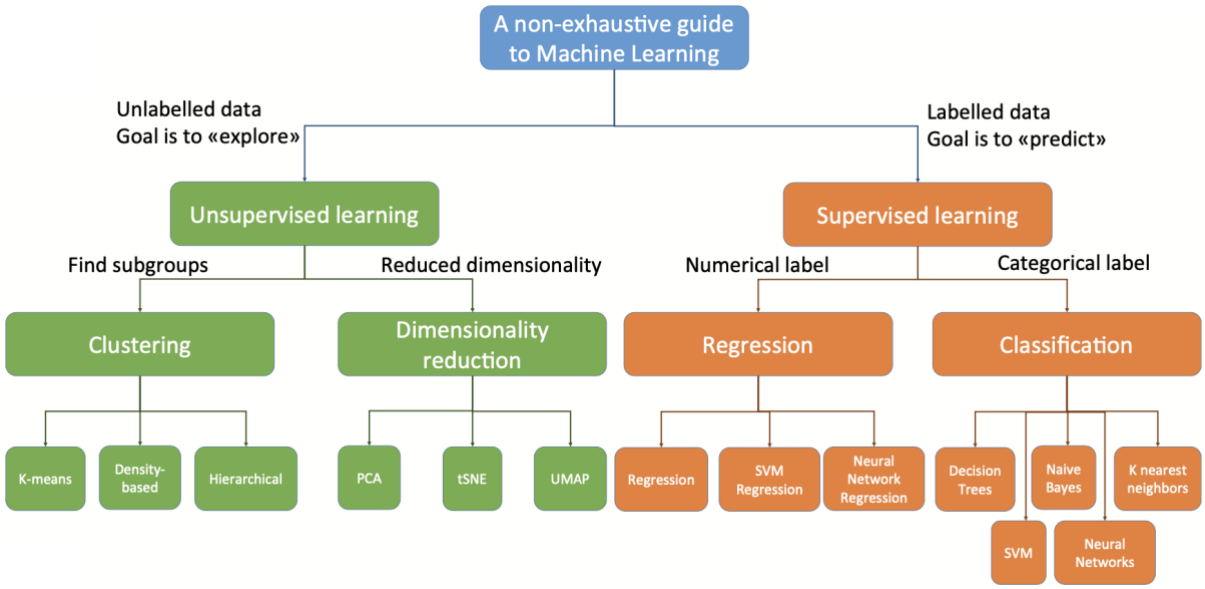

Figure 2 Machine Learning Techniques

The above flowchart by (Badillo et al., 2020) shows the various machine learning techniques that are used in general.

The use of labelled datasets distinguishes the machine learning strategy known as supervised learning. These datasets are intended to "supervise" or "train" algorithms to correctly classify data or forecast outcomes. Labelled inputs and outputs allow the model to monitor its precision and improve over time. (Sathya and Abraham, 2013)

The two types of Supervised Learning are:

i. Regression, and
ii. Classification

A statistical method for examining relationships between variables is regression analysis. Typically, the goal of an investigation is to determine the relationship between two variables, such as the impact of a price rise on demand or changes in the specifications of the smartphone. To investigate these questions, the researcher gathers data on the relevant underlying factors and uses regression to calculate the quantitative impact of the causative variables on the variable they affect.

In our study, we will be using Multivariate Linear Regression and Deep Neural Networks.
Multivariate linear regressions are commonly used to model the predictive relationships of interrelated responses on a set of predictors in chemometrics, econometrics, financial engineering, psychometrics, and many other areas of application (Yuan, Ekici, Lu and Monteiro, 2007).
A DNN is a collection of neurons organised in multiple layers, with neurons receiving neuron activations from the previous layer as input and performing a simple computation. The network's neurons develop and implement a complex non-linear mapping from input to output. This mapping is learned from data by updating the weights of each neuron using an error backpropagation technique (Montavon, Samek and Müller, 2018).

2.3 Multivariate Linear Regression

In many statistical analyses, the linear regression model is an important and effective tool for evaluating the connection between variables. Regression analysis is generally used for forecasting response variable values at interesting predictor variable values, identifying predictors related to the response variable, and evaluating how changes in the predictor variables impact the response variable (Weisberg, 2005).
The typical linear regression methodology assumes a scalar response variable. However, it is possible to be interested in exploring numerous response variables at the same time. In this case, each response might be subjected to a regression analysis independently. Such a study would miss out on detecting correlations between responses. A multivariate linear regression model is required for analysis in regression scenarios where associations of several answers are of interest. (Santiago, Guo and Sigman, 2018)

Figure below depicts the general approach for creating multidimensional descriptive models. The major components involved in this process are as follows:

1. identification and acquisition of relevant parameters;
2. design of an initial set of data for model construction (i.e., the training set);
3. intercorrelation assessment;
4. preliminary model development involving the identification of univariate trends and execution of multivariate linear regression; and
5. validation of multivariate models using cross- and external validation methods. The successful construction of accurate, informative models should enable virtual screening to speed up reaction optimization and predictor variable analysis to gain mechanistic insights. This section contains detailed guidance for each step of model development and evaluation.

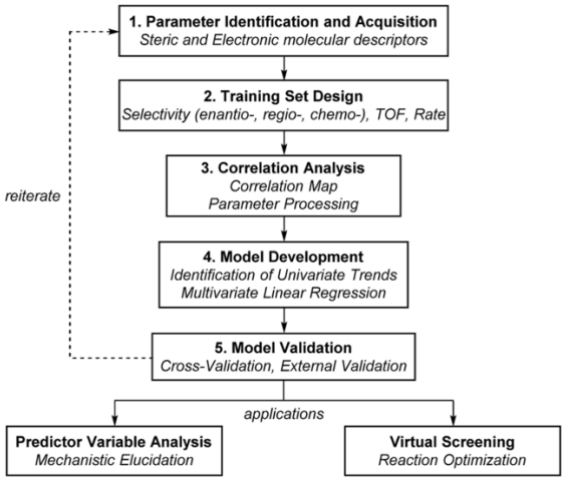

Figure 3 General Approach for Creating Multidimensional Descriptive Models

(Santiago, Guo and Sigman, 2018)

2.4 Deep Neural Network

A DNN is a collection of neurons structured in many layers, where neurons receive neuron activations from the previous layer as input and perform a simple computation (e.g., a weighted sum of the input followed by a nonlinear activation). The network's neurons work together to create a sophisticated nonlinear mapping from input to output. This mapping is learned from data by adjusting the weights of each neuron using an error backpropagation algorithm. A neuron in the top layer usually represents the concept that must be interpreted. Top-layer neurons are abstract (we cannot see them), whereas the DNN's input domain (e.g., image or text) is frequently interpretable. We will show you how to create a model in the input vector that is interpretable and represents the abstract learning concept. The framework for activation maximisation can be used to develop the prototype. (Montavon, Samek and Müller, 2018)

Machine learning algorithms are used in unsupervised learning to examine and group unlabelled data sets. These algorithms find patterns in the data without the help of a person. (Sathya and Abraham, 2013)

There are two types of Unsupervised Learning, they are:

i. Clustering, and
ii. Dimensionality Reduction

As stated by T. Soni Madhulatha, 2012 the most significant unsupervised learning problem is clustering, which, like all other problems of this type, involves identifying a structure in a set of unlabelled data. Therefore, a cluster is a group of objects that are "similar" to one another and "dissimilar" to those found in other clusters. The terms cluster analysis, automatic categorization, numerical taxonomy, and typological analysis are also used as synonyms for the word "data clustering."

As stated in the paper written by P. Juneau, 2014, the technique of obtaining a set of degrees of freedom that can be utilised to replicate the majority of a data collection's variability is also known as dimensionality reduction.

2.5 Price Prediction

As stated by Liu, Huang, Han and Yang, 2020 the method that could be used for selecting the appropriate variables is the principal component analysis method, these variables influence the pricing of recycled mobile phones, and then the pricing of recycled mobile phones is established using a fuzzy neural network to achieve an accurate analysis of the mobile phone price and the key characteristic variables. Finally, the concept of momentum is introduced into the fuzzy neural network parameter optimization method to realise the parameter update of the pricing model. They also have compared a few methods but have found in their analysis that Fuzzy Neural Network with Momentum in Updating the Parameters has the least errors and the highest accuracy.

From the study that was conducted by Asim and Khan, 2018 we can state that the except for the combination of WrapperattributEval and Decision Tree J48 classifier, the results of both Feature selection algorithms and classifiers were comparable. This combination achieved maximum accuracy while selecting only the most necessary features. It is important to note that adding irrelevant features to the data set reduces the efficiency of both classifiers in Forward selection. Backward selection loses efficiency if any important feature is removed from the data set. The main cause of the low accuracy rate is the small number of instances in the data set. Another thing to keep in mind while working is that converting a regression problem to a classification problem introduces more errors.

In a study done by Güvenç, Çeti and Koçak, 2021 the authors compared KNN and DNN classifiers to predict mobile phone price ranges.

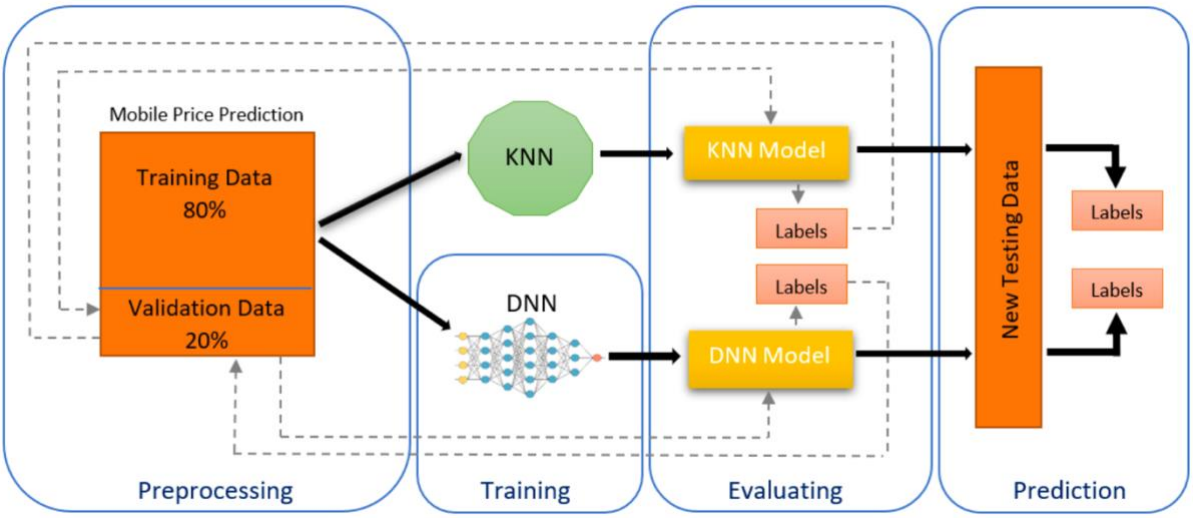

Figure 4 Flow Stages of the Study

The above figure shows the flow of the stages of the study that is conducted by Güvenç, Çeti and Koçak, 2021 and they found from the study that DNN had better accuracy than KNN, and also the performance of both the models was high but DNN had outperformed KNN in terms of the performance as well. The researchers used 80% of the data as the training set and the remaining 20% as the testing data set.

Andrey Viktorovich, Viktor Aleksandrovich, Igor Leopoldovich and Irina Vasilevna, 2018, used their original ideas like residual regressor, logit transform and neural network machine to predict the sales prices of the houses.

2.6 Web Scrapping

Data analysis is the process of deriving solutions to problems through data questioning and interpretation. The analysis process comprises discovering difficulties, resolving the accessibility of appropriate data, determining which method can aid in the solution of the fascinating challenge, and communicating the conclusion. The data must be separated into multiple processes for analysis, such as starting with its specification, assembling, organising, cleaning, re-analyzing, using models and algorithms, and the final output. Web information scraping and public support are excellent methods for naturally developing content on the web. A large number of people used these tactics in research and business for making content or delivering criticism to improve the exactness of company advertising that enables individuals to give resources in advancing and developing the firm. Web scraping is renowned for "Screen Scraping" and "Web Data Extraction" in general. (Revati, Jacob and Ashok, 2022)The web scrubber programming is intended to be comprehensive for all significant data from various online stores and mining and collecting it into the new website. The web scraper tool is used for deriving information from the web host, as well as web orders, web mining and data mining, online esteem change observing and value correlation, element survey scratching (to watch the challenge), gathering land postings, atmosphere data checking, webpage change area, inspect, following on the web closeness and reputation, web mash-up, and web data joining. Pages are created with content-based advancement dialects (HTML and XHTML) and frequently contain a plethora of cooperative data in the content structure. However, because

most internet pages are designed for human end users rather than robotized use, this may be the case. As a result, the web scraping toolbox was created. (Revati, Jacob and Ashok, 2022)

Web scraping is a technique for extracting data from the World Wide Web (WWW) and saving it to a file system or database for subsequent retrieval or analysis. It is also known as web extraction or harvesting. Web data is commonly scraped using Hypertext Transfer Protocol (HTTP) or a web browser. This can be done manually by a user or automatically by a bot or web crawler. Because a massive amount of data sets are continuously created on the World wide web, web scraping is considered an effective and powerful way for collecting big data. (Zhao, 2017). Data is a crucial component of any research, whether academic, marketing, or scientific. People may choose to collect and analyse from several websites. The various websites that belong to the specific category display information in various formats. Even if you only have one website, you may not be able to see all of the data at once. The information may be spread across numerous pages and sections. The majority of websites do not permit you to store a copy of the data presented on their websites in your local storage. The only alternative is to manually duplicate and paste the data displayed by the website into a local file on your computer. This is a time consuming and laborious task. (De S Sirisuriya, 2022)

As stated by (De S Sirisuriya, 2022) there are nine types of web scraping techniques. “They are

i. Traditional copy and paste
ii. Text grapping and regular expression
iii. Hypertext Transfer Protocol Programming
iv. Hypertext Markup Language Parsing
v. Document Object Model Parsing
vi. Web Scrapping Software
vii. Vertical aggregation software
viii. Semantic annotation recognizing
ix. Computer vision webpage analysers”

In our project, we use web scrapping software to collect the data.

Web scraping is the process of extracting data from HTML files found on the web using a computer. This data is frequently in the form of patterned data, such as lists or tables. Application programming interfaces are sets of commands used by programmes that interact with internet pages and extract data (APIs). These APIs can be 'trained' to retrieve patterned data from individual websites or all comparable pages on a website. Alternatively, automated interactions with webpages can be integrated into APIs, allowing links within a page to be 'clicked,' and data retrieved from the following pages. This is very handy for obtaining information from numerous search result pages. (R Haddaway, 2022)

There is a variety of Web Scraping Software available on the market that may assist you in scraping data from any website you choose. The following is a list of scraping tools. (De S Sirisuriya, 2022)

i. Visual web ripper
ii. Web content extractor
iii. Mozanda web scrapper
iv. UIPath
v. Out Wit Hub
vi. Screen Scrapper
vii. Web Harvy
viii. Easy web exctractor
ix. Web sun dew
x. Web data extractor
xi. Helium scrapper
xii. Webextractor360
xiii. Fminer
xiv. Scrappy
xv. Import.io
xvi. WebScrapper

In our project we use scrapy. A collaborative and open-source platform for extracting data from websites. Scrapy is written in Python and is available for Linux, Windows, and Mac. Scrapy is an application framework for crawling localities and extracting constructed data that may be used for a wide range of supported applications, including data mining, information administration, and real-time reporting. Even though Scrapy was initially intended for web scraping, it may also be used to evacuate data utilising APIs (for example, Amazon AWS) or as an all-around useful web crawler. Python is used to write Sketchy. Consider this Wiki example: "A simple online photo gallery may offer three options to users, as specified by HTTP GET parameters in the URL." (Thomas and Mathur, 2019)

2.7 Scrapy

Scrapy is a Python web scraping framework. It provides you with all of the tools you need to easily extract data from websites, analyse it as you see fit, and store it in the structure and format you desire. Because the internet is so diverse, there is no "one-size-fits-all" technique for pulling data from websites. Ad hoc ways are frequently used, and if you begin writing code for every minor operation you undertake, you will soon end up constructing your scraping framework. Scrapy is the framework in question.

Chapter 3: Methodology

We use Deep Neural Network and Multivariate Linear Regression to calculate the price of the smartphones and the price of the SoC that is used. To do this we need to collect the required data. The data regarding all the smartphones launched are on a website named GSMArena. Also for predicting the price of the SoC we will be needing the price of the SoC and this is found on Intel's website.

3.1 Data Selection and Data Collection

To collect the data from the websites mentioned above, we will have to use web scraping techniques. Some of the methods are

x. Traditional copy and paste
xi. Text grapping and regular expression
xii. Hypertext Transfer Protocol Programming
xiii. Hypertext Markup Language Parsing
xiv. Document Object Model Parsing
xv. Web Scrapping Software
xvi. Vertical aggregation software
xvii. Semantic annotation recognizing
xviii. Computer vision webpage analysers"

We use Scrappy for web scraping and collecting data from GSMArena and Intel's websites. The first step is to create a virtual environment, the reason why we create a virtual environment is that it does not affect the other applications and make any changes to the system. This is useful for protecting the system that is being used for scraping the data from the websites. We then create a folder inside the virtual environment and then activate it. The next step was to install the scrapy module. After installing the module we need to create a new scrapy project. In scrapy we always need to create a spider file, hence after the above steps, we create a spider file in the folder. This spider file constantly gets the data from the websites. We always build one class with a unique name and establish requirements when constructing a spider. The first step is to name the spider by using the name variable and then enter the starting URL through which the spider will begin crawling. Define some strategies for crawling considerably deeper into that website. For the time being, let's scrape and save all of the URLs. The main goal is to obtain each URL and then request it. Retrieve all of its URLs or anchor tags. To accomplish this, we must add another function, parse, to retrieve data from the specified URL.

We can then fetch the data using selectors and select the required data. We then have to create a parse method to fetch all the URLs and this will loop the whole process and will continuously happen. Scrapy by default will skip the URLs that have been used. This is the main reason we have chosen the Scrapy module so that the data cleaning process might get a bit easier. The final step for data collection is to run the spider and export the data that is retrieved from the website into a JSON file. We can then import this data into an excel workbook. Excel can be used for cleaning the data.

3.2 Data Cleaning

The problem of data quality and cleaning is a big hurdle for any research that relies on quantitative data. We have identified five critical aspects of data quality:

1. Precision
2. Completion
3. Individuality
4. Promptness
5. Reliability.

The initial data had a lot of issues and the quality of the data was very bad with lots of issues where the data of one column was mixed with the data of other columns. It had a lot of unnecessary columns that are there. For our research we needed two different datasets both of them being extracted from websites using web scrapping tools. The first dataset that is used to find the smartphone price had the columns, link – which had the smartphone link to the GSMArena website, image – which had the image link of the smartphone, name – which had the name of the smartphone, release date – which contains the date of the release of the smartphone in the global market, weight – this has the weight of the smartphone along with the thickness, OS – this has the data regarding what operating system the smartphone is using, storage – this has the data regarding the storage capacities of the smartphones, fans – this has the data regarding the number of fans that the phone has, popularity – this has the data in percentage regarding the popularity of the smartphone on the day of the launch, hits – this has the data of how many times the website GSMArena has seen people view the smartphone in their website, next few columns like the screen size, screen resolution, RAM, SoC, the data that is present in these columns are self-explanatory. The next columns are net2g, net3g, net4g and net5g these columns have the data of the bands and the frequency of these bands in their respective columns. The speed column has the storage speed and the type of storage that is used in the smartphone for instance USB 3.0 or USB 3.1. The year column has the data regarding the date in which the phone was launched. The body other column gives us data regarding the IP rating of the smartphone, display type column gives us data regarding the kind of display that is used in the smartphone like TFT, LCD or AMOLED panels etc. The sim column has the type of sim that can be used in the smartphone. The display resolution column has the display resolution given in pixels. The chipset column has the data regarding the chipset that is used in the smartphone. The GPU column has the names of the gpu that is used in the phone like the Arden0 640,etc. We have the wlan and the Bluetooth columns that have the data regarding the bands and the frequency that the smartphone supports. The radio column has the data in a character format but is a factor that is it has two variables either yes or no. The sensors column has the data regarding the different types of sensors like the FaceID, fingerprint sensor, etc. Colours column has the different colours the smartphone was released. The tbench column has the data of the AnTuTu and Geekbench 5 test results. The battery column has the exact battery size of the battery in the smartphone. There are a few other columns which are nit that important in the dataset. These columns have been removed so that the accuracy of the model will increase in the end.

In the below figure, we can see the snippet of the original data.

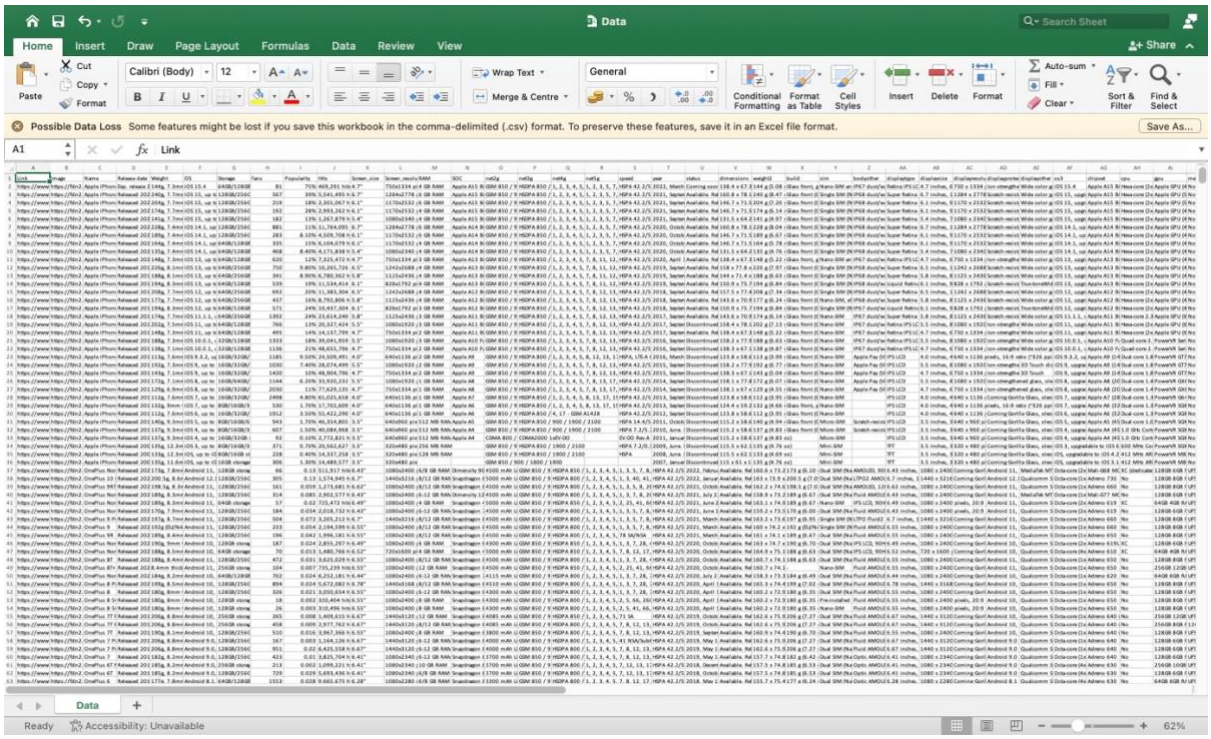

Figure 5 Snippet of the Original Data Extracted From GSMArena

The first step that we have taken to clean the data is to remove all the unnecessary columns in the dataset. The columns that were removed are link, image, release date, fans, popularity, hits, year, status, os3, cam1features, cam1video, cam2features, cam2video, wlan, radio, usb, batdescription, colours, models, featuresothers, sar-us, sar-eu, tbench, batlife, battalktime, gprstext, edge, batstandby1, optionalother. These columns are removed as they have a lot of missing values greater than sixty per cent. This is the reason we cannot even impute the values in these columns, and most of the data in these columns have characters as the data type which makes it even harder to impute values or input the values for every missing column.
We then use the weight column that has both the weight and the thickness of the smartphone. We split these two data into two separate columns using the LEFT and RIGHT functions in excel. After splitting these columns, we can notice that some anomalies are present in the data, they are the weight of some phones are too less, like 100.5g or 100g. To cross-verify this anomaly we use the screen size column, this column has the screen size of the smartphones we can see that there are screen sizes that are less than two inches these are smartwatches that have been included in the data set. We can also see that in the screen size column we can see that there are screen sizes like 18 inches, and 17 inches, these cannot be the screen sizes of smartphone displays. So first we filter the screen sizes, we filter the screen sizes that are too small and too large. We can see that there are many entries like smartwatches and tablets. So, we can now filter out using the operating system. The entries that have Tizen OS, Watch OS and Wear OS can be removed as these are the operating systems that are used by the wearables. We can then remove the tablets that are included in the data set by filtering out the data using the screen size we can see that there is only one smartphone that has a screen size of thirteen inches, and the rest of them that are larger are all tablets we can verify this by looking at the name of the product and also the operating system for Apple devices since they

have iPad OS as their operating system. The next step is to convert the character data type to numeric data type for instance in the RAM column we have the values as 4GB RAM and this is considered as a character type variable, we can use the LEFT and RIGHT functions again in excel to just get the number, but this is still a character data type to change it to a numeric variable we need to use the text to columns feature in excel. Similarly, we have to do the same for the weight, thickness, battery size, screen size, internal memory, cam1, cam2 and price columns. We can see that there are multiple missing values in the data set. We can also remove all the other unimportant columns.

For the price prediction of the SoCs, we will scrape the data from geekbench 5 a website that is used for benchmarking the CPUs. We can see the different CPUs that have been benchmarked and their scores have been published. So first we scrape the data from the website using scrappy. We get the following columns after scraping the data, they are,
name – which has the name of the chip that has been benchmarked, platform – this column has the operating system of the device that houses the chip, there are five different operating systems in this column they are, iOS, Android, Linux, Windows and macOS, date – this has the date the result was published, single core score – this column has the score of the SoC when the single core of the SoC is tested, multi-core score- this column has the value of the scores that are benchmarked by geek bench software for all the cores that are present in the SoC. It also has two more columns that have the name of the smartphone and the laptop that is benchmarked and the name of the CPU. This data set does not have any missing values. The first step is to delete the columns that are of no use to us. We can delete the name of the device and the URL for now. The next step is to extract the data from Intel's website which is the price of the processors that are made by intel. Intel is the only company that has the price of the processors that are manufactured by them listed on their website. Once we collect the price data from the website we then move on to the Geekbench 5 data and we use the functions LEFT and RIGHT in excel to split the model name into two parts, this is done since the model name contains the name of the processor and the number of cores that are included in the processor. The figure below shows the data that has been scrapped from Geekbench 5 website.

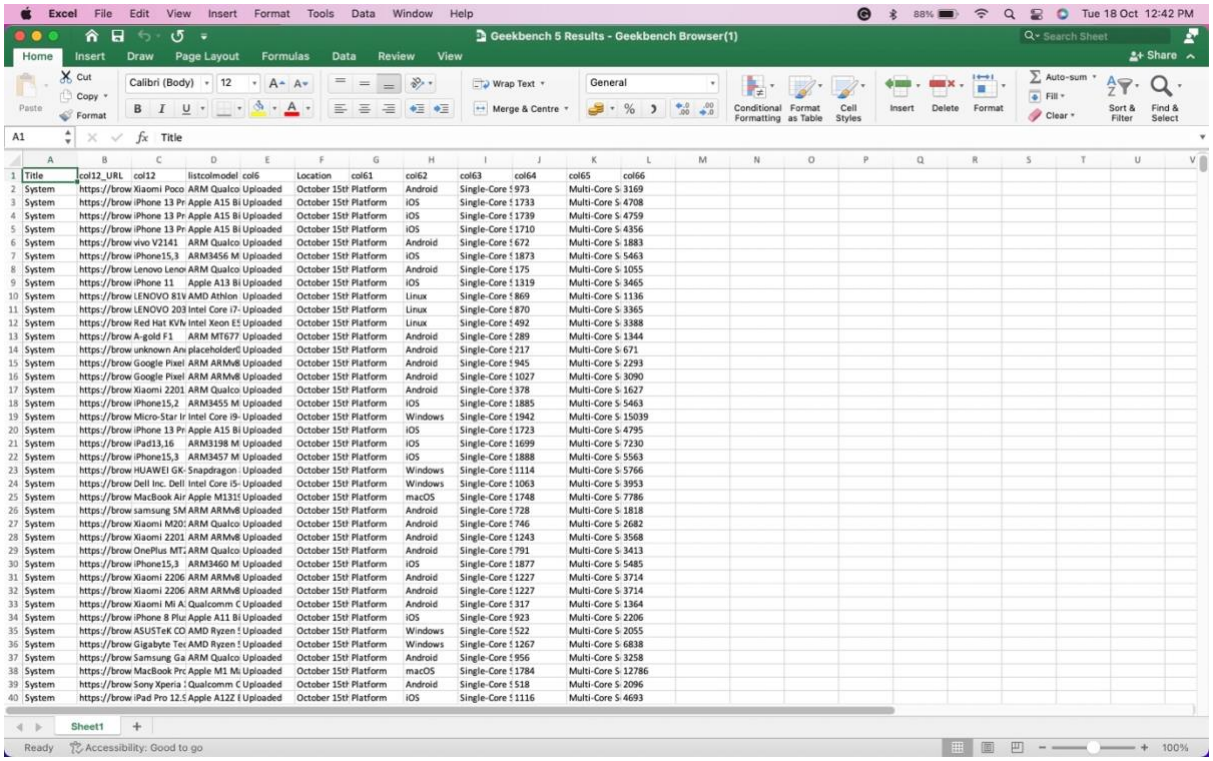

Figure 6 Data Extracted from Geekbench 5

Now we have to use the VLOOKUP function in excel to match the model's name and input the price data from the intel price data set to the geekbench 5 dataset. Once we get the price in the geekbench 5 dataset we remove the observations which has the platform of Linux and macOS to get an accurate imputation of the prices for the smartphones.

3.3 Handling Missing Values

In the smartphone data set the final columns that we use for the analysis have the following columns, Screen_size – where the screen size is in inches, RAM – where the values are in GB, Battery – here the battery capacity is given in mAH, weight – here the weight is given in grams, Internal_Memory – here the capacity of the storage is given in GB, Cam1MP – the camera module's megapixels is specified here, Cam2MP – the second camera module's megapixel is specified here, Price – the price of the smartphone is given here.
There are three columns where there are character variables that are as factors, they are memoryslot, gps and nfc, these three columns have the observations as either yes or no, these are converted as numeric values using R during the process of imputing the missing values. To impute the missing values, we have used three different methods and have selected the method which had the best accuracy. The three methods for imputing the missing values in GSMArena dataset are done using R; the three methods are, missforest, multiple imputation and simple imputation. The multiple imputation method has the best accuracy of the three methods that have been used to impute the missing values.

As for the Geekbench 5 data set we have used multiple imputations to impute the price of all the chips for both android and iOS platform devices with the help of the Intel price for their

processors, we have taken into account the number of cores, the single core score and the multi-core score. Once the values have been imputed, we now remove the windows platform and just have the platforms iOS and Android.

3.4 Analysis

After we have clean data, we move on to the analysis part, we use the datasets to perform analysis. We find the relation of the price with the other variables, etc. We visualize the data that is there, both the data for the prediction of smartphone prices and the data for the chip prediction. We use ggplot 2 for all the visualizations since it is the most elegant graphics package. We use geometric point graphs to look at the comparisons of the price with respect to the dependent variables. We can use bar plots to see the comparison of the single core and multi core performance for the two different platforms. We can also plot the correlation of the independent variables with the dependent variables.
We can plot the independent variable vs the dependent variables for a better understanding of the relationship between the variables. We then find the correlation matrix of all the data in both the datasets that we have, we then plot this correlation matrix into a correlation plot.

3.5 Deep Neural Net

After the analysis part, we create a deep neural network to predict the price of the smartphone. For this, we first create a neural net where the output layer is the price and the input layer has the screen size, battery, RAM, weight, memory slot, internal memory, cam 1 mp, cam 2 mp, GPS and NFC. We also have one hidden layer with ten neurons. The next step is to convert the data into matrix form to feed the values to the neural net, and then we also split the data into two parts one for testing and the other for training the deep neural net. We then normalize both the training and testing data, once we normalize the values, we create the model. For this, we use the Keras model sequential. In this model that we create we use the default values that are used in the R. After that we compile the model, then fit the model and then evaluate the model. Once all this is done, we can see that the accuracy is not that great so we have to fine-tune the model. To fine-tune the model, we change the number of layers that are there in the model we now include two hidden layers. The input layer is the same then we have a hidden layer with ten neurons and then we have another hidden layer with five neurons and then we have the output layer of price. While creating the model we still use the Keras model sequential package but now we have three layers, the first layer has a hundred neurons and a dropout rate of forty per cent. The second layer has fifty neurons and a dropout rate of thirty per cent. The third layer has twenty neurons with a dropout rate of twenty per cent. The activation function that we use is “relu”. We use dropout rates to avoid overfitting the model, the second time when we did not have any dropout rate we had an issue with overfitting leading to a loss in the accuracy of the model. After this, we compile the model, and we use the mean square error type of compilation, with a learning rate of 0.02% and the metrics of mean absolute error. We had used the learning rate of 0.04% as well but that also led to a drop in the accuracy of the model, this was also caused due to the overfitting of the model. After that, we fit the model and use a hundred epochs with a validation split of twenty per cent. Then we finally evaluate the model and measure the accuracy of the model with the help of the testing data set.

3.6 Multivariate Linear Regression

For this, we use the Geekbench 5 dataset that has the price values imputed using multiple imputations. The data is loaded into R for the analysis process. The first step is to create a multivariate linear regression model, once this is done we take the coefficients of the input variables (single-core score, multi-core score and the number of cores). Once we calculate and get the coefficients of these input variables, we can now ask the user to input the variables and once they input the variables into the system the program runs and gives the output as the price of the chip that is used. The figure below shows the user input UI.

```
Enter the single core score : 748
> x2 <- readline(prompt = "Enter the multi core score : ");
Enter the multi core score : 3784
> x3 <- readline(prompt = "Enter the number of cores : ");
Enter the number of cores : 8
```

Figure 7 User Input UI for Chip Price Prediction

Once the user inputs the values for the single-core score, the multi-core score and the number of cores, we get the output price of the chip.

Chapter 4: Results

4.1 Smartphone Price Prediction

The below figure shows the summary of the dataset that is used for predicting the price of smartphones.

```
> summary(df)
  Screen_size          RAM             Battery          weight       memoryslot
 Min.   :1.770   Min.   : 0.000   Min.   :   34   Min.   : 52.7   Length:4947
 1st Qu.:4.700   1st Qu.: 1.000   1st Qu.: 2000   1st Qu.:140.0   Class :character
 Median :5.500   Median : 2.000   Median : 3000   Median :163.0   Mode  :character
 Mean   :5.369   Mean   : 3.296   Mean   : 3068   Mean   :167.4
 3rd Qu.:6.400   3rd Qu.: 4.000   3rd Qu.: 4000   3rd Qu.:186.0
 Max.   :8.010   Max.   :18.000   Max.   :13200   Max.   :500.0

 Internal_Memory     Cam1MP           Cam2MP           gps                nfc
 Min.   :  0.1   Min.   :  0.00   Min.   : 0.000   Length:4947        Length:4947
 1st Qu.:  4.0   1st Qu.:  8.00   1st Qu.: 1.900   Class :character   Class :character
 Median : 16.0   Median : 13.00   Median : 5.000   Mode  :character   Mode  :character
 Mean   : 68.0   Mean   : 18.06   Mean   : 7.799
 3rd Qu.: 64.0   3rd Qu.: 16.00   3rd Qu.: 8.000
 Max.   :256.0   Max.   :108.00   Max.   :60.000

     Price
 Min.   :  10.0
 1st Qu.: 130.0
 Median : 190.0
 Mean   : 225.1
 3rd Qu.: 280.0
 Max.   :1300.0
 NA's   :1422
```

Figure 8 Summary of the Data Used for Smartphone Price Prediction After the Data is Cleaned

The figure below shows the missing values that are present in the data that is being used for predicting the smartphone price.

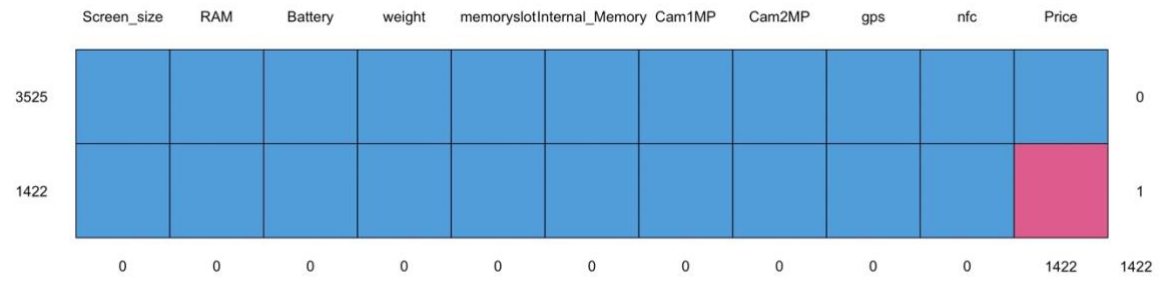

Figure 9 Missing Values in the Dataset

The figure below shows the summary of the final data that is used for the analysis after the missing values are imputed.

```
> summary(final_df)
  Screen_size         RAM            Battery          weight        memoryslot    Internal_Memory
 Min.   :1.770   Min.   : 0.000   Min.   :   34   Min.   : 52.7   Min.   :1.000   Min.   :  0.1
 1st Qu.:4.700   1st Qu.: 1.000   1st Qu.: 2000   1st Qu.:140.0   1st Qu.:2.000   1st Qu.:  4.0
 Median :5.500   Median : 2.000   Median : 3000   Median :163.0   Median :2.000   Median : 16.0
 Mean   :5.369   Mean   : 3.296   Mean   : 3068   Mean   :167.4   Mean   :1.845   Mean   : 68.0
 3rd Qu.:6.400   3rd Qu.: 4.000   3rd Qu.: 4000   3rd Qu.:186.0   3rd Qu.:2.000   3rd Qu.: 64.0
 Max.   :8.010   Max.   :18.000   Max.   :13200   Max.   :500.0   Max.   :2.000   Max.   :256.0
     Cam1MP          Cam2MP           gps             nfc            Price
 Min.   :  0.00   Min.   : 0.000   Min.   :1.000   Min.   :1.000   Min.   :  10.0
 1st Qu.:  8.00   1st Qu.: 1.900   1st Qu.:2.000   1st Qu.:1.000   1st Qu.: 120.0
 Median : 13.00   Median : 5.000   Median :2.000   Median :1.000   Median : 190.0
 Mean   : 18.06   Mean   : 7.799   Mean   :1.974   Mean   :1.314   Mean   : 226.6
 3rd Qu.: 16.00   3rd Qu.: 8.000   3rd Qu.:2.000   3rd Qu.:2.000   3rd Qu.: 280.0
 Max.   :108.00   Max.   :60.000   Max.   :2.000   Max.   :2.000   Max.   :1300.0
```

Figure 10 Summary of Final Dataset Used

The figure below shows if there are any missing values that are present in the final dataset that is going to be used for the smartphone price prediction.

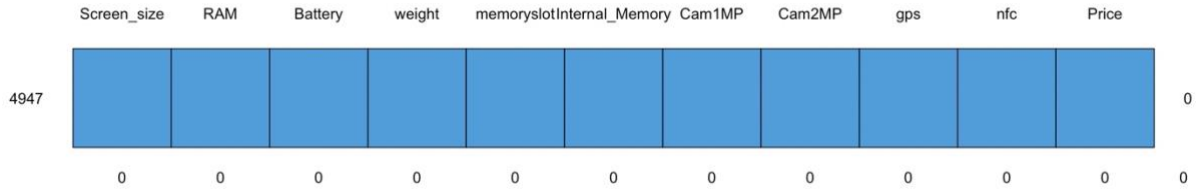

Figure 11 Missing Values in Final Dataset

The figure below shows the Price vs Screen Size of the smartphones.

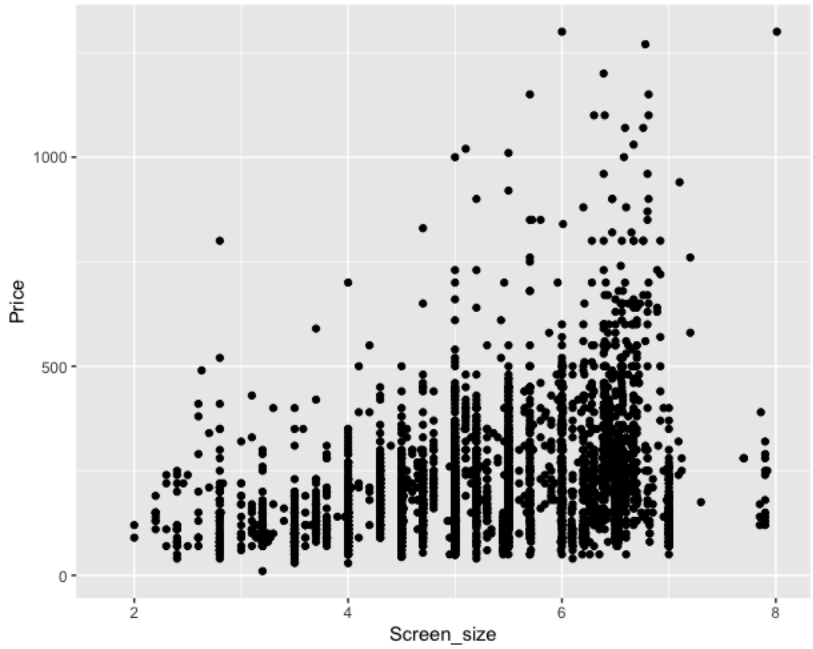

Figure 12 Price vs Screen Size

The next figure shows the Ram vs Price graph of the smartphones.

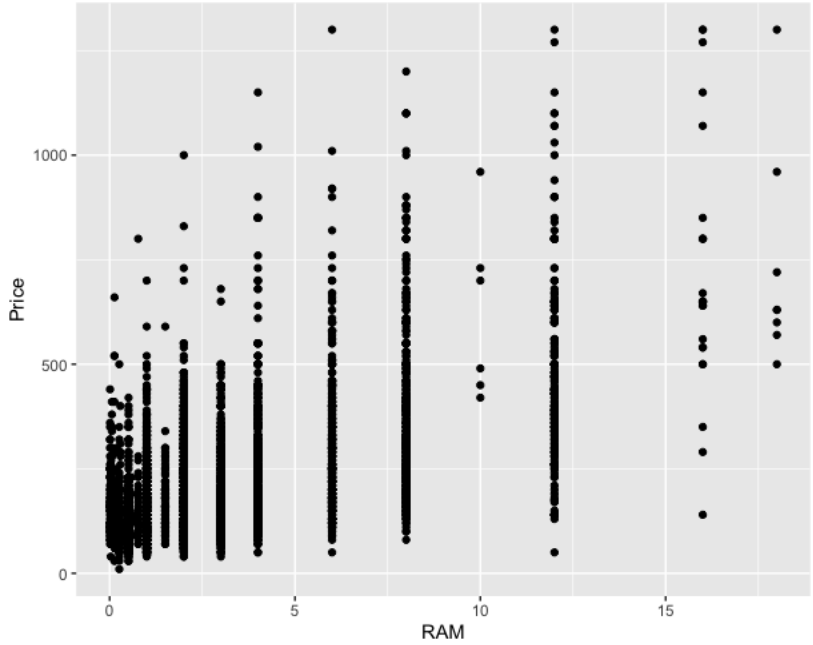

Figure 13 RAM vs Price

The next figure shows the graph between weight and price.

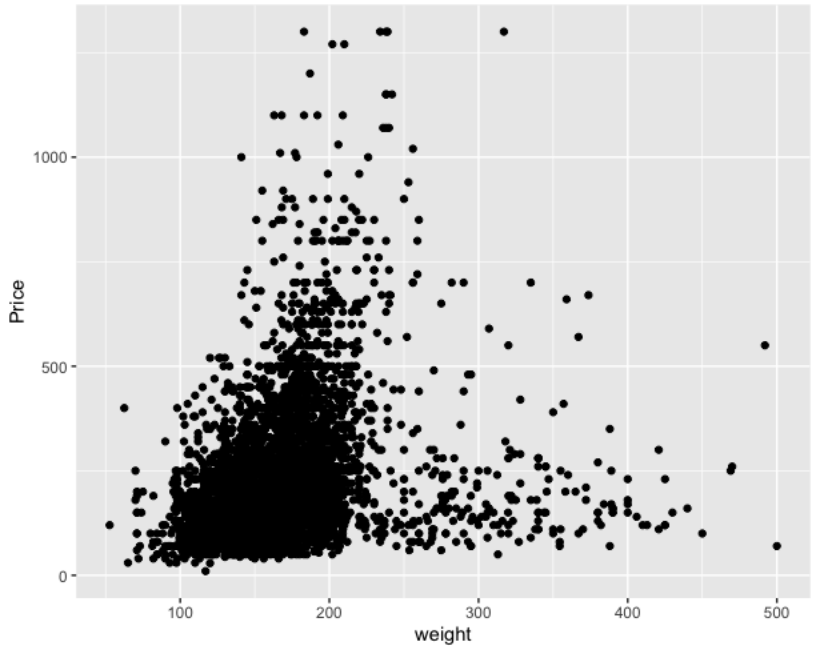

Figure 14 Weight vs Price

The next figure shows the graph of the count of smartphones that have a memory slot and does not have a memory slot. Here the value of one means there is no memory slot available and the value of two means there is a memory slot that is present in the smartphone.

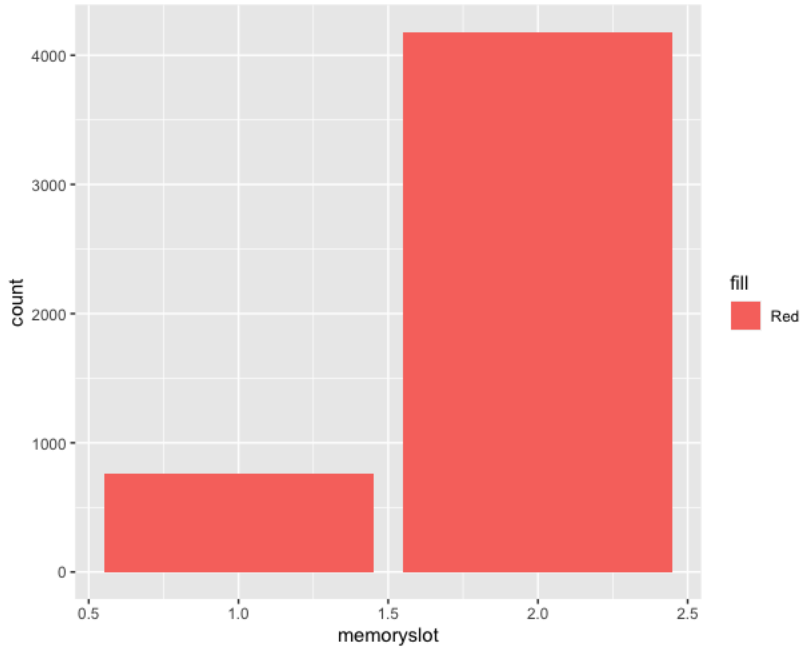

Figure 15 No. of phones with and without a memory slot

The below figure shows the price of the phone vs the memory slot.

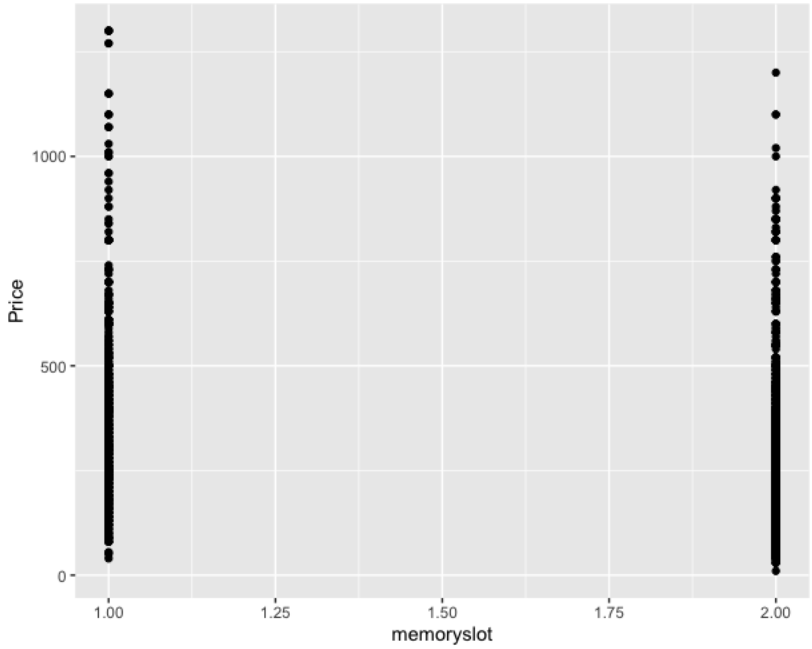

Figure 16 Memory Slot vs Price

The next plot shows the relation between internal memory (GB) and price.

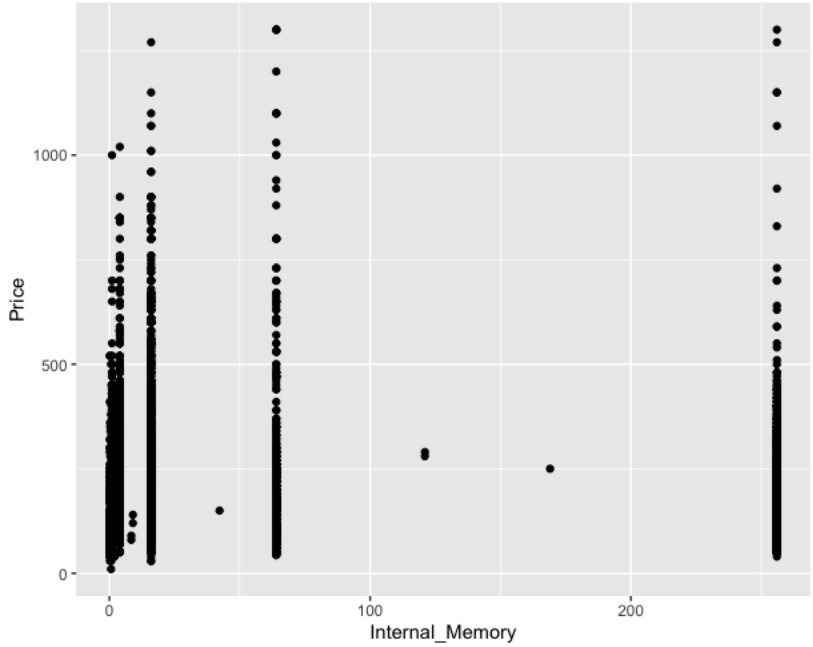

Figure 17 Internal Memory vs Price

The below graph shows the megapixels of the primary camera versus the price of the smartphone.

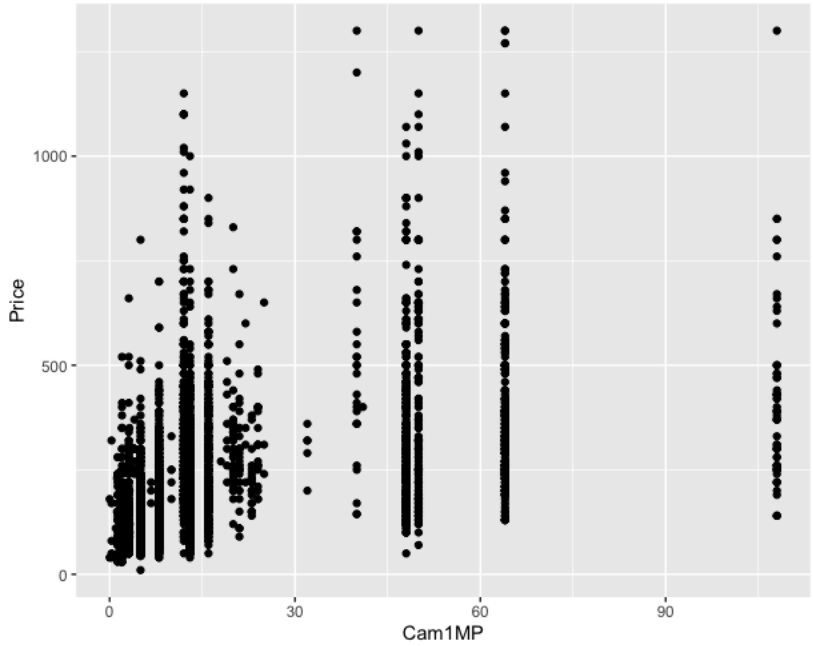

Figure 18 Primary Camera (in megapixels) vs Price

The below graph shows the megapixels of the secondary camera versus the price of the smartphone.

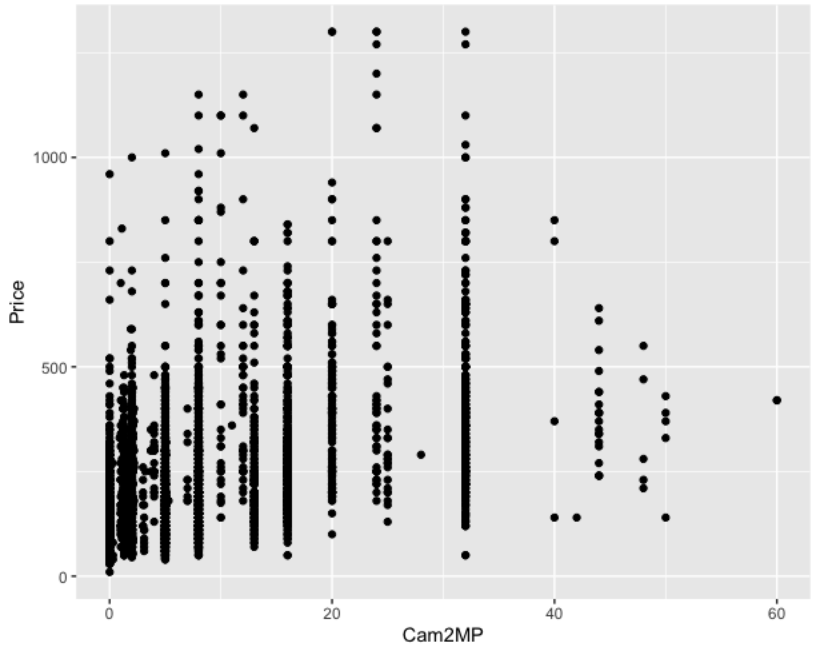

Figure 19 Secondary Camera (in megapixels) vs Price

The next graph shows the number of phones that have GPS and the number of phones that do not have GPS. Here one means there is no GPS and two means there is a GPS that is present in the smartphone.

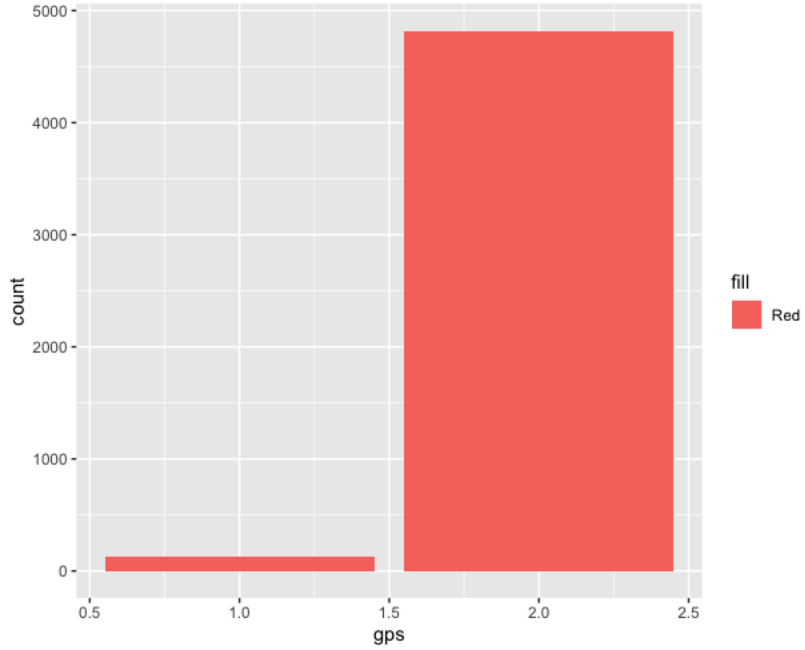

Figure 20 Number of Phones with GPS and No GPS

The below graph shows the GPS versus the price of the smartphone.

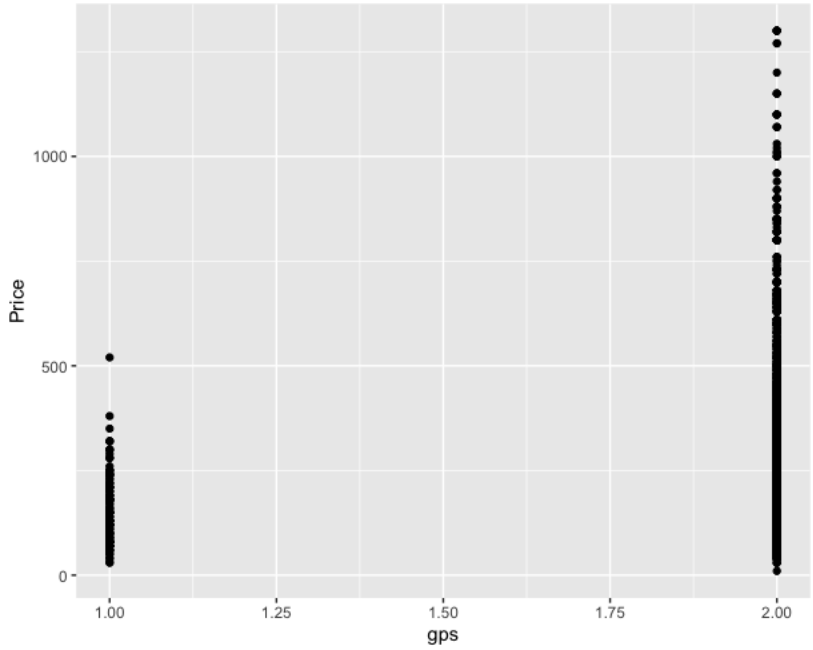

Figure 21 GPS vs Price

The next graph shows the number of phones that have NFC and the number of phones that do not have NFC. Here one means there is no NFC and two means there is an NFC that is present in the smartphone.

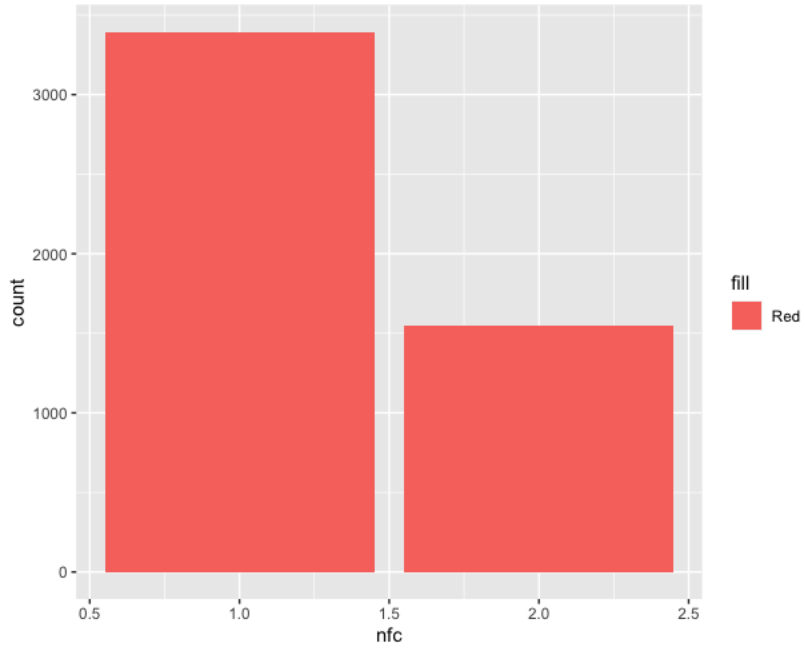

Figure 22 Number of Phones with NFC and No NFC

The below graph shows the NFC versus the price of the smartphone.

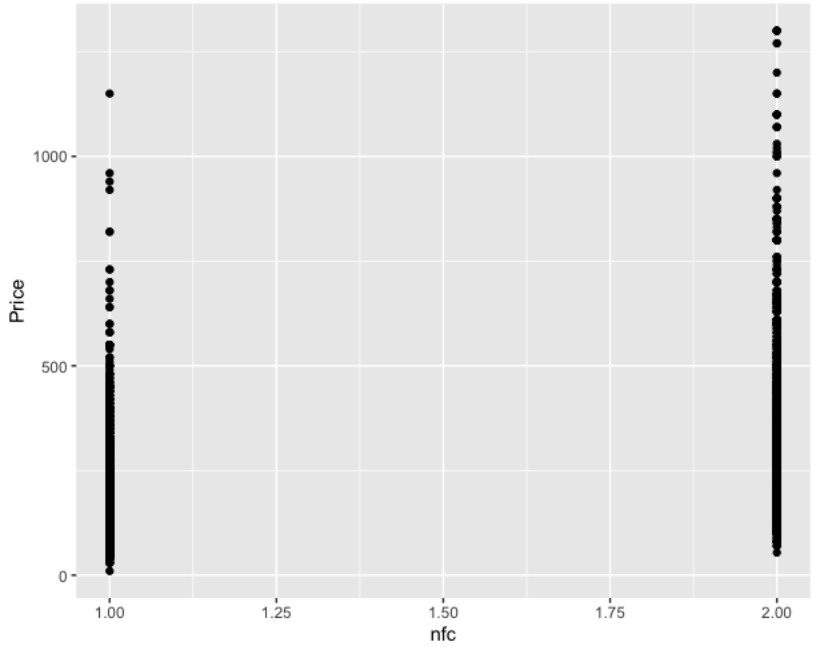

Figure 23 NFC vs Price

The next figure shows the correlation matrix of all the variables that are present in the dataset.

	Screen_size	RAM	Battery	weight	memoryslot	Internal_Memory	Cam1MP	Cam2MP	gps	nfc	Price
Screen_size	1.00000000	0.6623557	0.8275183	0.67818568	-0.20609869	-0.03577867	0.5642097	0.5949856	0.25044023	0.27330588	0.34828573
RAM	0.66235575	1.0000000	0.6881655	0.35055494	-0.46139695	-0.21755418	0.7810247	0.7748456	0.14970551	0.43427638	0.59284442
Battery	0.82751826	0.6881655	1.0000000	0.63086768	-0.18678668	-0.12432843	0.6127937	0.5862637	0.17309154	0.27653373	0.35377721
weight	0.67818568	0.3505549	0.6308677	1.00000000	-0.12455051	-0.03672823	0.3063204	0.2642812	0.01633750	0.14431173	0.25497828
memoryslot	-0.20609869	-0.4613969	-0.1867867	-0.12455051	1.00000000	0.02314892	-0.3489119	-0.3193636	-0.03407483	-0.28435799	-0.42419342
Internal_Memory	-0.03577867	-0.2175542	-0.1243284	-0.03672823	0.02314892	1.00000000	-0.1967330	-0.1994154	0.06828135	-0.04373082	-0.09109737
Cam1MP	0.56420970	0.7810247	0.6127937	0.30632040	-0.34891190	-0.19673296	1.0000000	0.7218928	0.10502526	0.38136948	0.44979972
Cam2MP	0.59498557	0.7748456	0.5862637	0.26428118	-0.31936360	-0.19941543	0.7218928	1.0000000	0.13594811	0.34526817	0.48288740
gps	0.25044023	0.1497055	0.1730915	0.01633750	-0.03407483	0.06828135	0.1050253	0.1359481	1.00000000	0.10974994	0.06981410
nfc	0.27330588	0.4342764	0.2765337	0.14431173	-0.28435799	-0.04373082	0.3813695	0.3452682	0.10974994	1.00000000	0.49827402
Price	0.34828573	0.5928444	0.3537772	0.25497828	-0.42419342	-0.09109737	0.4497997	0.4828874	0.06981410	0.49827402	1.00000000

Figure 24 Correlation Matrix of Smartphone Dataset

The next figure shows the correlation plot of the above correlation matrix.

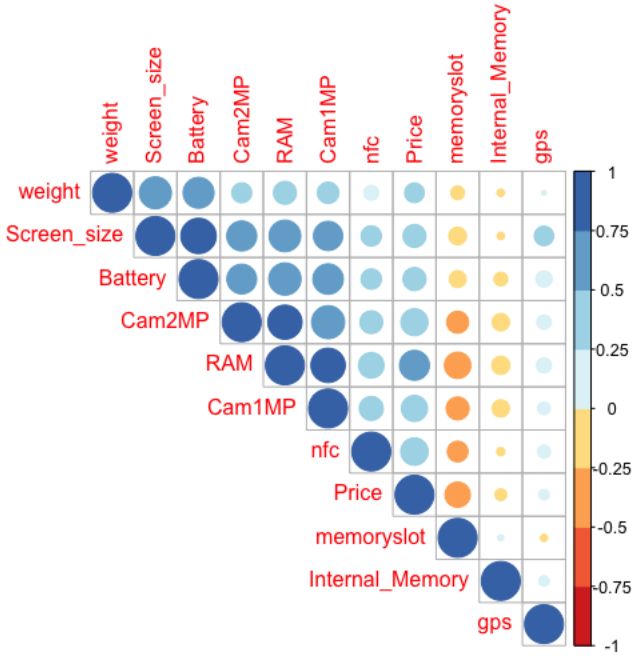

Figure 25 Correlation Plot of Smartphone Dataset

The next figure shows the neural network that is used for the price prediction of the smartphone. In the figure, we can see the two hidden layers with ten and five neurons respectively and also the input layer in the deep neural network and the output of the deep neural network.

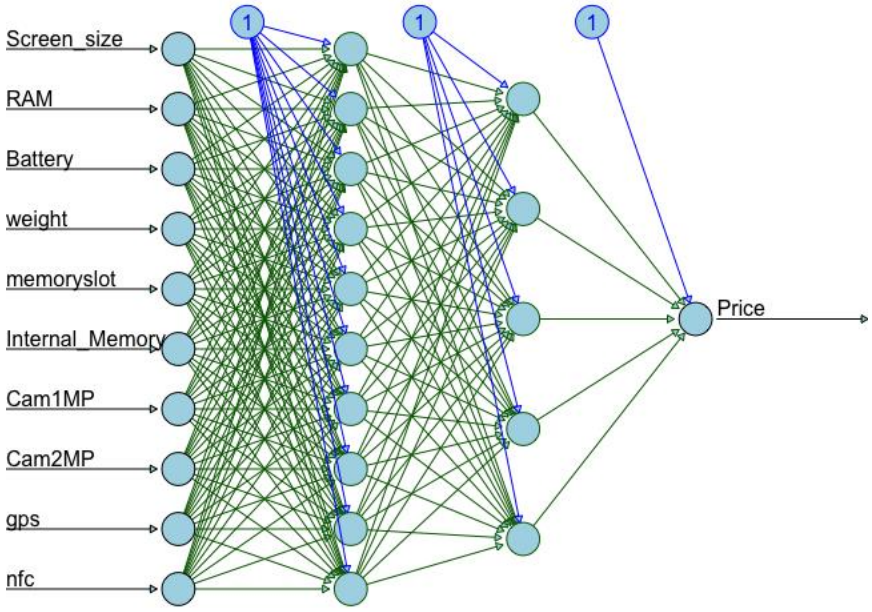

Figure 26 Neural Network

The below figure shows the loss value of the testing and the training data and also the mean absolute error for all of the hundred epochs.

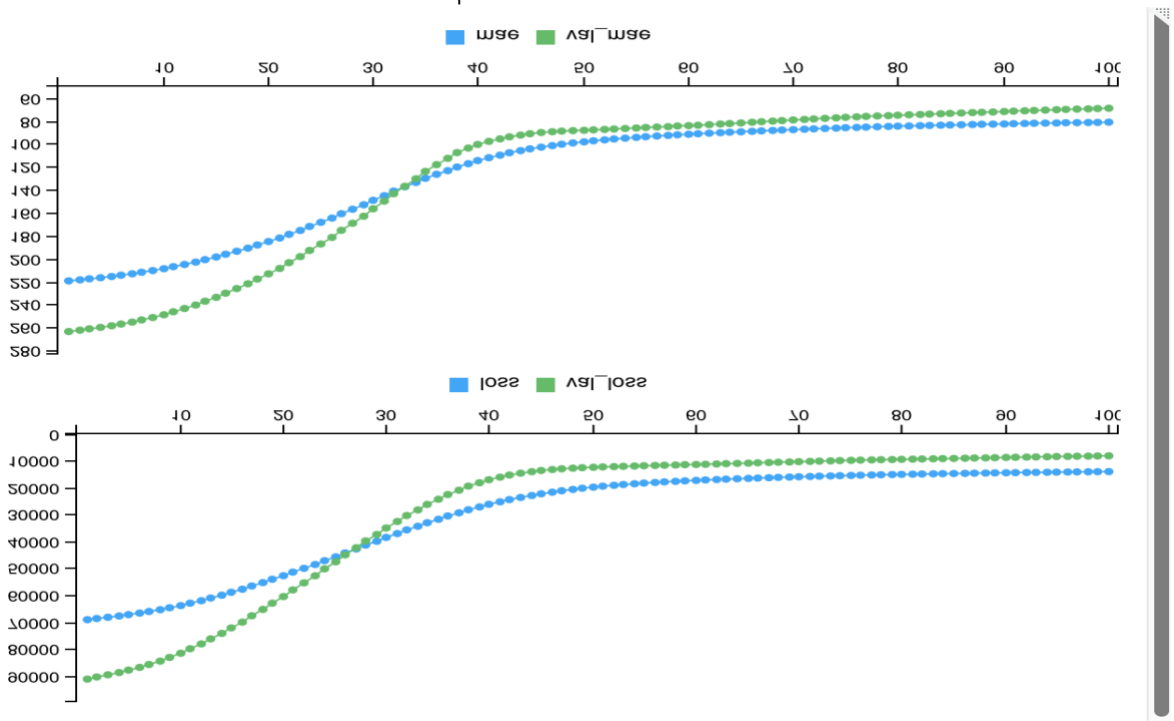

Figure 27 Loss Value and the Mean Absolute Error

The figure below shows the predicted values and the tested values and we can see that there is some linearity in the values.

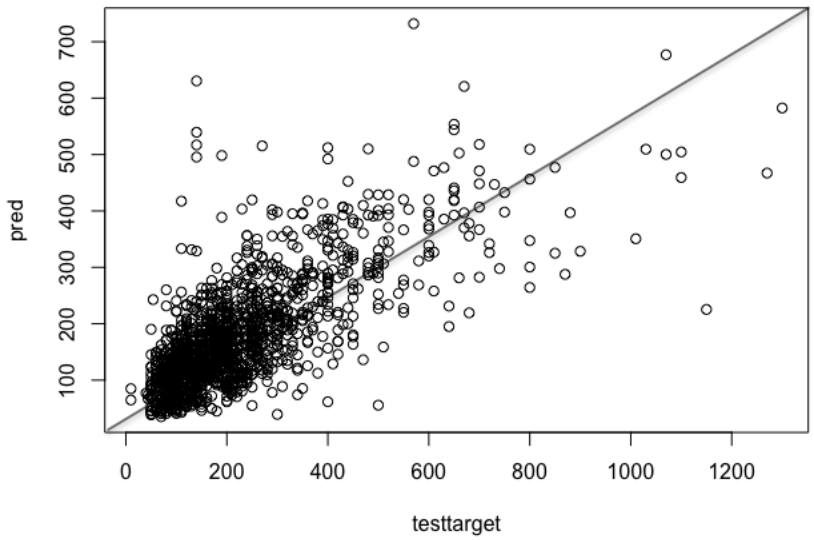

Figure 28 Predicted Values and Tested Values

The figure below shows the value of the loss and the mean absolute error of the deep neural network after all the fine tuning as 2.92 per cent. This is the final output for the analysis of smartphone price prediction.

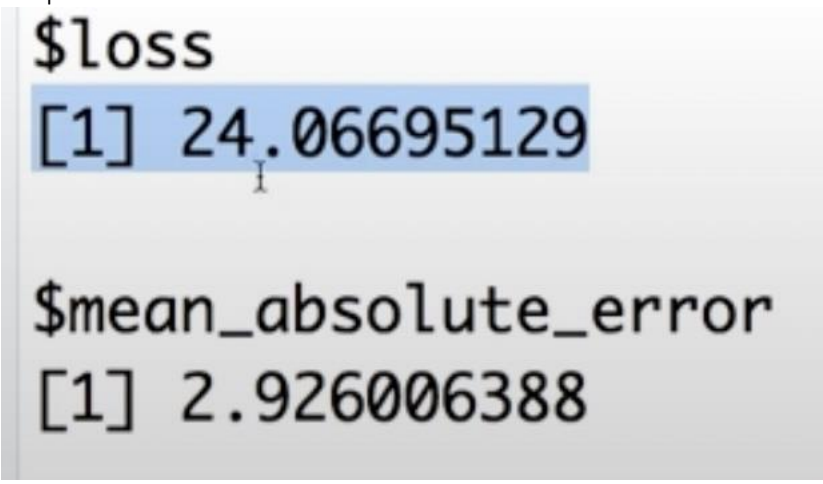

Figure 29 Final Output

4.2 Chip Price Prediction

The below figure shows the summary of the dataset that is used for predicting the price of the chipset.

```
> summary(df)
 listcolmodel         Platform         Single_core_score Multi_core_score
 Length:14646       Length:14646       Min.   :  59      Min.   :  130
 Class :character   Class :character   1st Qu.: 570      1st Qu.: 1821
 Mode  :character   Mode  :character   Median : 899      Median : 2889
                                       Mean   : 970      Mean   : 2928
                                       3rd Qu.:1312      3rd Qu.: 3612
                                       Max.   :1906      Max.   :13544

 Number_of_cores      Price
 Min.   : 2.000   Min.   :   7.96
 1st Qu.: 6.000   1st Qu.: 186.32
 Median : 8.000   Median : 288.48
 Mean   : 7.222   Mean   : 276.06
 3rd Qu.: 8.000   3rd Qu.: 349.04
 Max.   :20.000   Max.   :3050.40
```

Figure 30 Summary of the Dataset that is Used for the Chip Price Prediction After the Dataset is Cleaned

The below figure shows the count of the smartphones' platforms (Android or iOS).

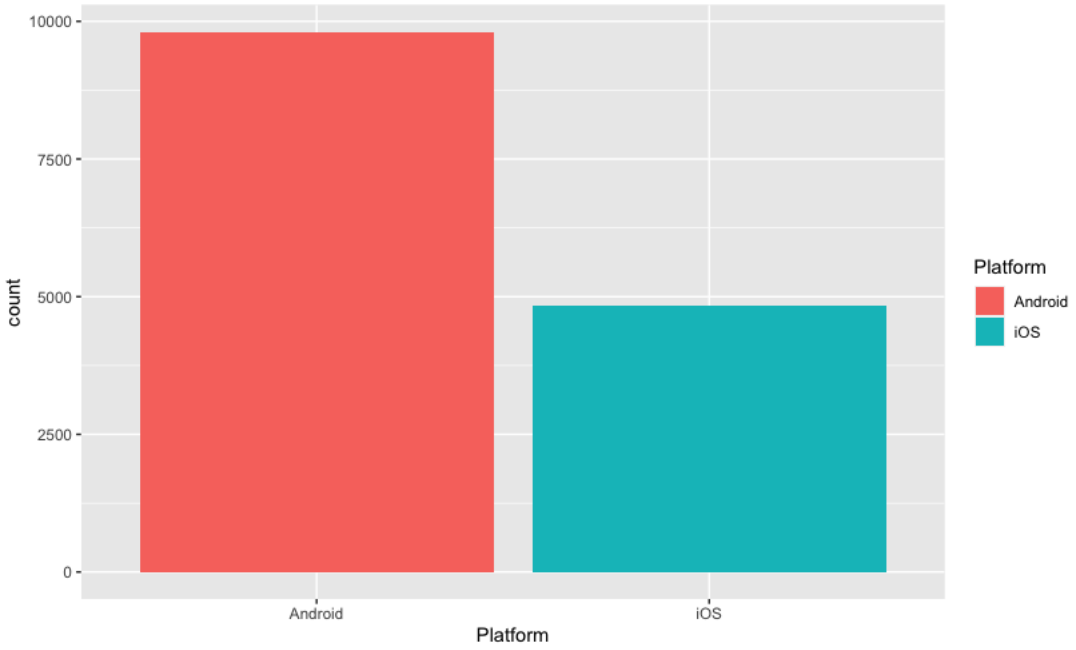

Figure 31 Count of the Smartphones' Platform

The next figure shows the correlation matrix of the Chipset Price dataset.

```
> c
                  Single_core_score Multi_core_score Number_of_cores      Price
Single_core_score         1.0000000        0.9192429     -0.36882619 0.81796617
Multi_core_score          0.9192429        1.0000000     -0.17694531 0.94643322
Number_of_cores          -0.3688262       -0.1769453      1.00000000 0.08214001
Price                     0.8179662        0.9464332      0.08214001 1.00000000
>
```

Figure 32 Correlation Matrix of the Chipset Price Dataset

The below figure shows the correlation plot of the correlation matrix of the chipset price dataset.

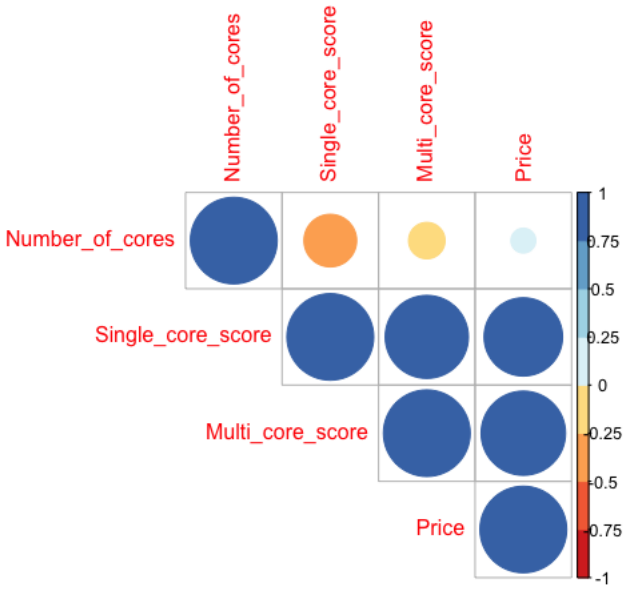

Figure 33 Correlation Plot of Chipset Price Dataset

The next figure shows the final output of the chip price prediction when the user has inputted the values according to them. The final output is the price of the chipset with the given single-core score, multi-core score and the number of cores.

```
Enter the single core score : 120
Enter the multi core score : 2000
Enter the number of cores : 4
> Price_Predicted
(Intercept)
   112.4472
```

Figure 34 Final Output (Price of the Chipset)

We can see that there is a 2.9% error in the DNN method that has been used to predict the smartphone price. This error rate has been achieved after a lot of trial and error methods by changing the values of the learning rate, the number of neurons that are present in the hidden layer, the number of hidden layers, the dropout rate of each and every hidden layer were also changed to get the final output of the smartphone price prediction.

Similarly, we get the output for the second part of the analysis where we find the price of the chipset that is used in the smartphone when the user inputs the single-core score, the multi-core score and the number of cores.

Chapter 5: Conclusion and Shortcomings

Smartphones can be considered a boon or a bane to society, it depends on how we utilize the phone. As said by Arora, Srivastava and Garg, 2020, we can say that the price of a smartphone is something that the consumer will look into before deciding on whether they should get the phone or not. Price is the first thing that we look at before looking at the specifications of the phone like the storage size, the number of cameras, the RAM, the processor that is used, the battery size, etc. We have predicted the price of the smartphones using Deep Neural Network by taking the dependent variables as the Price, RAM, internal storage, NFC, GPS, Battery size, etc. There are a lot of smartphones that are being bought and sold every single day. This shows how important smartphones have become in the consumer's everyday life. These smartphones are like a key that opens the door to the world wide web where people can get all sorts of information. This is one of the main reasons for me to take up this project to help people know the price of the smartphone before it gets launched in the global smartphone market and we can predict the price using the above mentioned technical specifications.

The companies that manufacture the SoCs have not made the price of the SoCs available for the general public to combat this issue the second half of the project was dedicated to predict the price of the SoCs when the scores are published on geekbench 5. We can just input the single core score, the multi-core score and the number of cores to get the cost of the chips. The processors used in smartphones have gained a lot of importance in the past few years and have significantly improved their computational calibre. We do need to know the price of the chips to make a better analysis of the cost-to-performance curve to make informed choices on the purchase of the smartphone.

We use web scrapping techniques to gather the data that is required for our project, we use scrapy to get the data from the websites and we use deep neural network and multivariate linear regression to find the price of the smartphone and the price of the chipset.

5.1 Shortcomings and Limitations

There are quite a few limitations for this project, the data that is collected by us can have a few issues in them like how accurate the data is since we have scraped the data from the internet which means that the data is prone to human error when they have published the data in the internet. There are few ethical issues where in we are not sure if we can scrap the data that is given in the websites. The other limitation for this study is the accuracy and the selection of the dependent variables. There is a lot of data that is present but the issue is that we are not very sure as to which dependent variable will be more accurate to get the final result. Because of the limited data and time available, the research was based on illustrative representations. The rest of the analysis was calculated and concluded based on the researcher's efforts to investigate the phenomenon and link it to the topic's field. The research objectives and research questions were addressed from the outset. met. However, the research's quality can be increased in the future with time.

6.List of References

1. Alsop, T., 2022. *Global smartphone AP/SoC chipset market share 2021 | Statista* . [online] Statista. Available at: https://www.statista.com/statistics/796887/smartphone-system-on-chip-market-share-by-vendor-worldwide/
2. Santiago, C.B., Guo, J.-Y. and Sigman, M.S. (2018). Predictive and mechanistic multivariate linear regression models for reaction development. *Chemical Science*, [online] 9(9), pp.2398–2412. doi:10.1039/c7sc04679k.
3. Lane, N., Miluzzo, E., Lu, H., Peebles, D., Choudhury, T., & Campbell, A. (2010,September). A survey of Mobile Phone Sensing. IEEE Communications Magazine,pp. 140-150.
4. Badillo, S., Banfai, B., Birzele, F., I. Davydov, I., Hutchinson, L., Kam-Thong, T., Siebourg-Polster, J., Steiert, B. and David Zhang, J., 2020. An Introduction to Machine Learning. *CLINICAL PHARMACOLOGY & THERAPEUTICS* , VOLUME 107(NUMBER 4). Also the reference for fig 2.
5. Alsop, T., 2022. *Global smartphone AP/SoC chipset market share 2021 | Statista* . [online] Statista. Available at: https://www.statista.com/statistics/796887/smartphone-system-on-chip-market-share-by-vendor-worldwide/ [Accessed 12 August 2022].
6. Arora, P., Srivastava, S. and Garg, B., 2020. MOBILE PRICE PREDICTION USING WEKA. *International Journal of Scientific Development and Research (IJSDR)* , Volume 5(Issue 4).
7. Chandrashekhara, K., Thungamani, M., Babu, C. and Manjunath, T., 2019. Smartphone Price Prediction in Retail Industry Using Machine Learning Techniques. *Emerging Research in Electronics, Computer Science and Technology* , Lecture Notes in Electrical Engineering(545).
8. Iyer, S. and Pawar, A., 2019. Machine Learning Model for Predicting Price of Processors using Multivariate Linear Regression. *Second International Conference on Smart Systems and Inventive Technology* ,.
9. Mahesh, B., 2018. Machine Learning Algorithms - A Review. *International Journal of Science and Research (IJSR)* , 9(1).
10. Sathya, R. and Abraham, A., 2013. Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification. *International Journal of Advanced Research in Artificial Intelligence* , 2(2).
11. Madhulatha, T., 2012. AN OVERVIEW ON CLUSTERING METHODS. *IOSR Journal of Engineering* , 02(04), pp.719-725.
12. Juneau, P., 2014. A Tutorial on Dimensionality Reduction in Large Claims Data Sets. *Value in Health* , 17(7), p.A555.
13. Yuan, M., Ekici, A., Lu, Z. and Monteiro, R., 2007. Dimension reduction and coefficient estimation in multivariate linear regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* , 69(3), pp.329-346
14. Montavon, G., Samek, W. and Müller, K., 2018. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing* , 73, pp.1-15.
15. Liu, H., Huang, J., Han, H. and Yang, H., 2020. An Improved Intelligent Pricing Model for Recycled Mobile Phones. *2020 Chinese Automation Congress (CAC)* ,.

16. Asim, M. and Khan, Z., 2018. Mobile Price Class prediction using Machine Learning Techniques. *International Journal of Computer Applications* , 179(29), pp.6-11.
17. Güvenç, E., Çeti, G. and Koçak, H., 2021. Comparison of KNN and DNN Classifiers Performance in Predicting Mobile Phone Price Ranges. *Advances in Artificial Intelligence Research (AAIR)* , Vol. 1(No. 1), pp.pp. 19-28.
18. Ganesh Iyer, S. and Dipakkumar Pawar, A., 2019. Machine Learning Model for Predicting Price of Processors using Multivariate Linear Regression. *Second International Conference on Smart Systems and Inventive Technology* ,.
19. Andrey Viktorovich, P., Viktor Aleksandrovich, P., Igor Leopoldovich, K. and Irina Vasilevna, P., 2018. Predicting Sales Prices of the Houses Using Regression Methods of Machine Learning. *IEEE* ,.
20. Zhao, B., 2017. Web Scraping. *Encyclopedia of Big Data* , pp.1-3.
21. De S Sirisuriya, S., 2022. *A Comparative Study on Web Scraping* . [online] Ir.kdu.ac.lk. Available at: http://ir.kdu.ac.lk/handle/345/1051 [Accessed 11 October 2022].
22. R Haddaway, N., 2022. The Use of Web-scraping Software in Searching for Grey Literature. *Grey Journal* , Volume 11(Number 3).
23. Thomas, D. and Mathur, S., 2019. Data Analysis by Web Scraping using Python. *IEEE Xplore* ,.
24. Revati, M., Jacob, C.R. and Ashok, K. (2022). Data Analysis by Web Scrapping: An Application Python. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, [online] 13(03), pp.439–447. Available at: https://www.turcomat.org/index.php/turkbilmat/article/view/13011/9334.
25. Weisberg, S., 2005. *Applied linear regression* (Vol. 528). John Wiley & Sons Montavon, G., Samek, W. and Müller, K.-R. (2018). Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73, pp.1–15. doi:10.1016/j.dsp.2017.10.011.