Laptop Price Predictions



Oleh:

Dhoni Hanif Supriyadi

1. Business Understanding

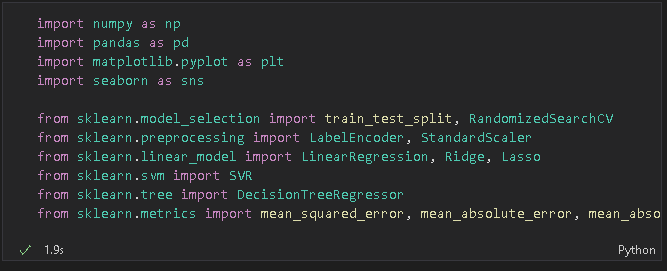
At this stage, a Data Scientist will try to understand the business first. Here, I want to make a price prediction for laptops. As we know, laptop prices will change every year and laptops also have different prices between one type and another. By knowing that this data is continuous data, what I will use is the regression method. Here, I will make a laptop price prediction with an emphasis on Exploratory Data Analysis (EDA) to get insight from the existing data so I can conclude what this data looks like.

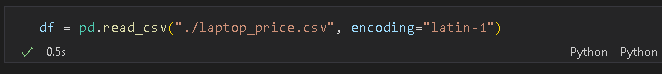
1. Data Collection

At this stage, a Data Scientist will collect the data first. Here, I have collected laptop data with id, name, type, product, ram, weight, screen size, and others sourced from kaggle.com with a total of 1303 data and 13 variables.

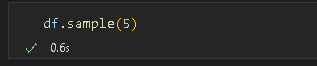
1. Data Preparation

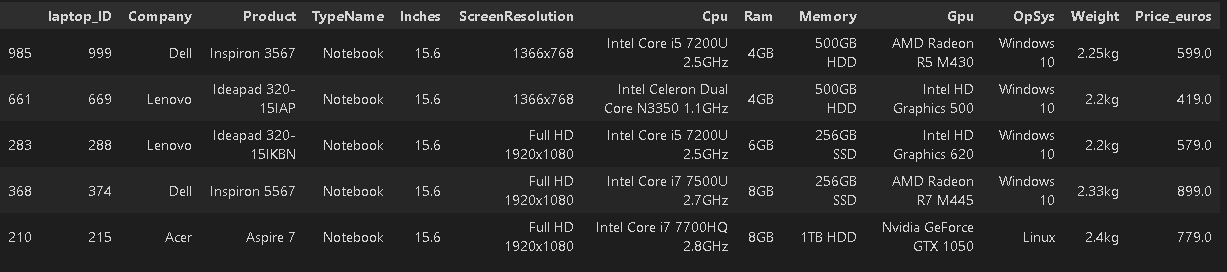
At this stage, a Data Scientist will read, check, and clean the data first. If there is a null value, a Data Scientist will fill it in with the mean, median, and mode values ​​and can even delete data that contains the null value. Here, I use python and jupyter notebook.

The first thing I did was import the libraries needed as shown below.

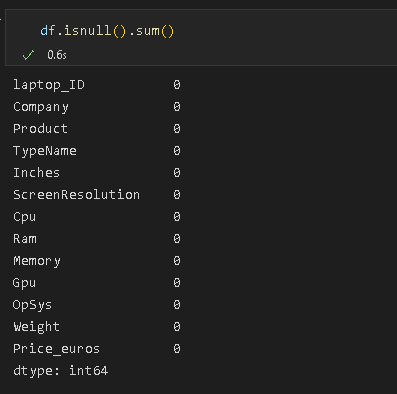
Next, I did a read on the csv data. Seeing that this data is in csv form, I use the pandas.read\_csv method as shown below.

Then, I checked 5 sample data randomly with pandas

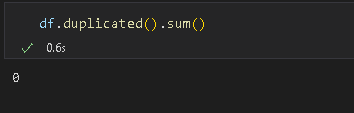




Next, I checked whether there was a missing value or not with the isnull() method from the pandas library as follows.

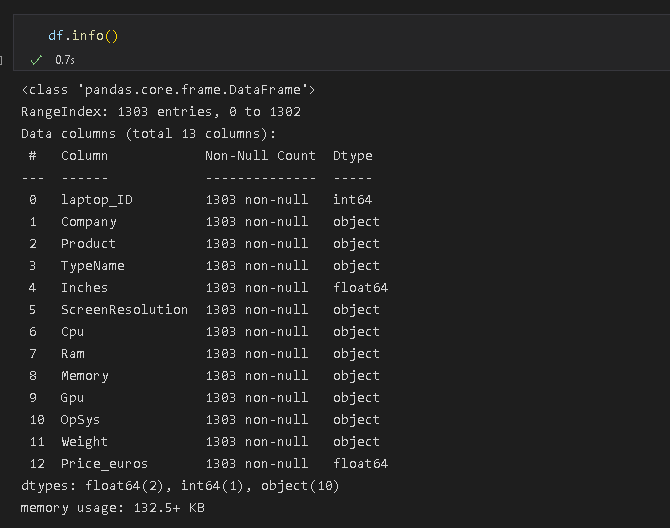


As we can see, this data holds no null or missing values. Next we check whether there is duplicate data or not with pandas.



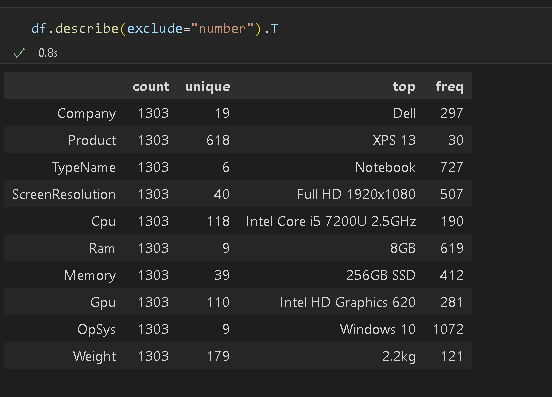
As we can see, data doesn't store duplicate values ​​either. In this way, we can continue the process by checking the data type, mean, median, standard deviation and others.

We check the information from the data first as follows.

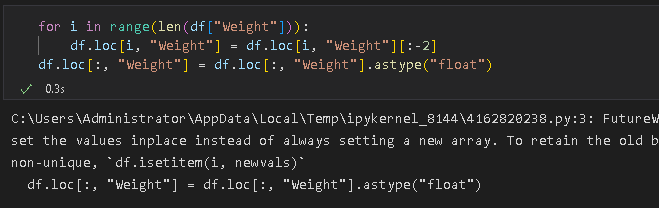


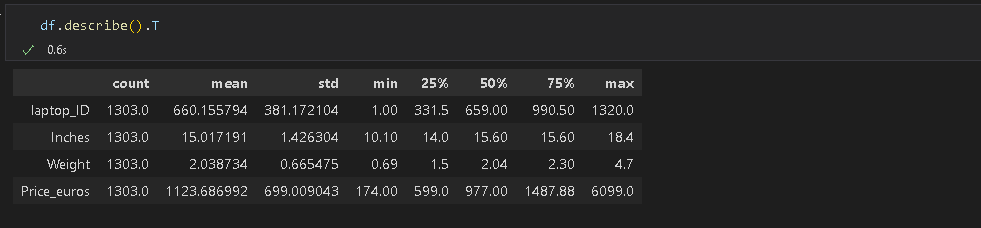
Here, the data does not contain null values ​​with a total of 1303 data and data types, there are several data types such as object, integer, and float with memory usage of 132.5+ KB.

Next, we check the amount of data, unique value, mode and frequency of each variable of type object to get that information. We do as follows.



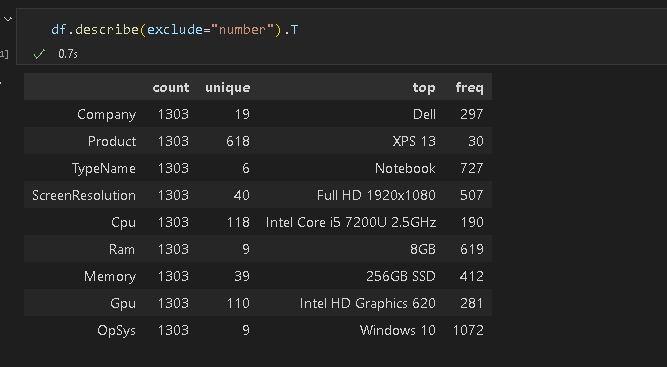
As we can see, some data has a unique value that is not that big so we can say that the variable does not store data with high variation and some data also says the opposite, that data stores data with high variation such as product, cpu, gpu data and weights.Melihat data Weight yang bervariasi dan menyimpan data angka, maka kita dapat mengubahnya menjadi data numerik. Sebelum itu, kita harus menghilangkan kata “kg” dalam data tersebut seperti berikut.



After we remove the word "kg" in the data, then we change the data type to float. Next, we check the mean, median, standard deviation, quartiles, min, and max values ​​of some numeric type data as follows.

As we can see, the 2 data sets have very high standard deviations compared to the other 2 data sets. These 2 data also store a max value that is so large compared to the other 2 data. Seeing that the laptop\_ID data has such a large standard deviation, it can be concluded that this data varies greatly. Maybe we'll see in the analysis stage.

Then, we try to print describe of object variable like this.



We can say that:

- Product have high unique value and TypeName have low unique value

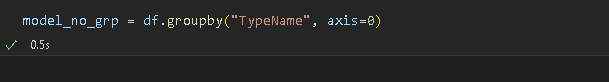
- So much users use windows 10 with frequency 1072

- The most product that people use is XPS 13

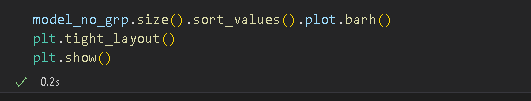
- We can use TypeName as the type of product

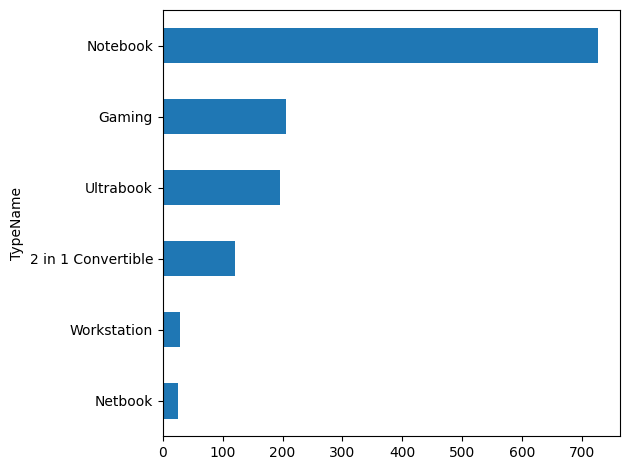
1. Exploratory Data Analysis (EDA)

First, lets make group data based on TypeName like this.



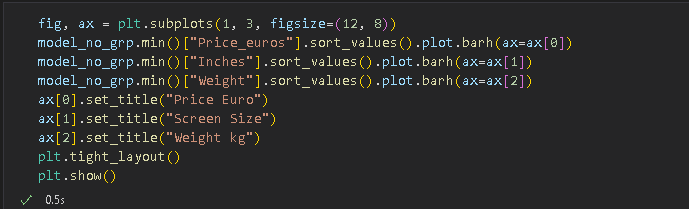
Then, lets plotting first the size or how much data of data TypeName.

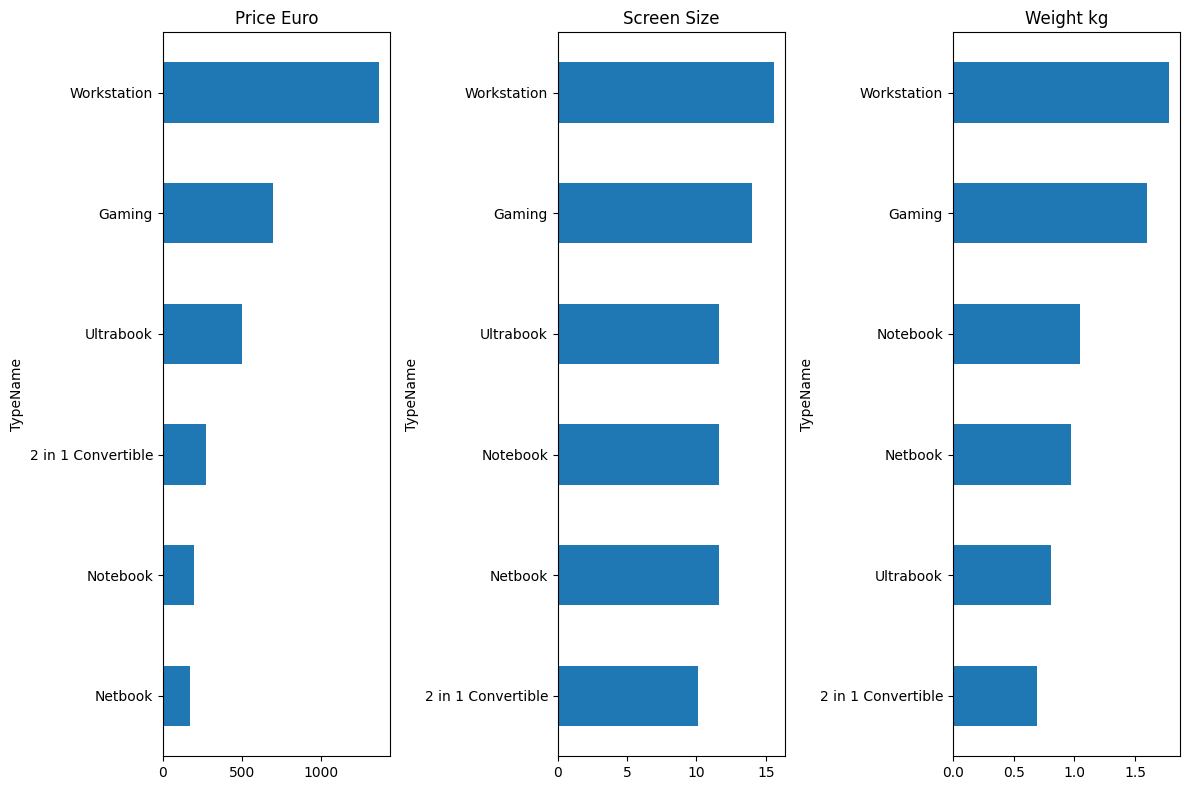




We can say that so much users use Notebook in this data and just little users who use Netbook. People who use Gaming type is the second highest data of data TypeName.

Then, lets visualize the minimum price euros, screen size, and weight of data based on TypeName.





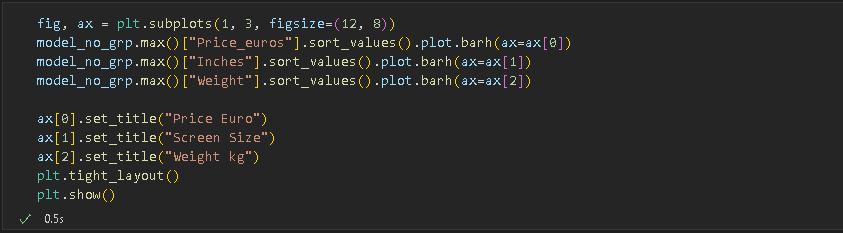
We can say that:

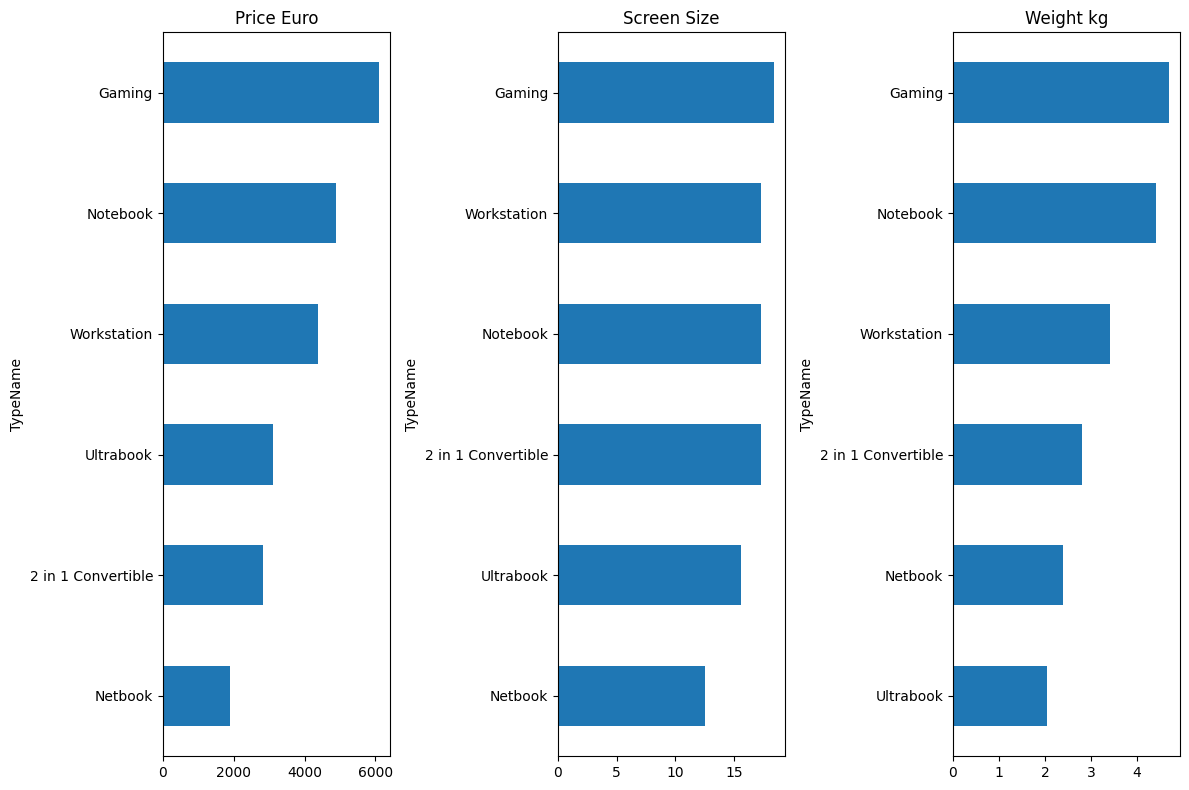
- The minimum price, screen size, and weight of Workstation is higher than the other

- The minimum price of Netbook is lower than the other but the minimum screen size of Netbook is higher than 2 in 1 Convertible and the minimum weight of Netbook is higher than Ultrabook and 2 in 1 Convertible

- 2 in Convertible is the type that has higher minimum price than Notebook and Netbook but this type is the lowest minimum screen size and weight

After that, lets visualize the maximum of price euros, screen size, and weight of data based on TypeName.





From this, we can say that :

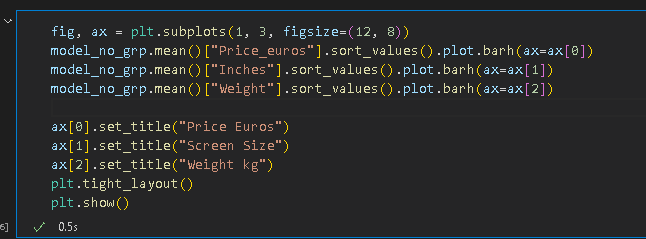
- The maximum of price, screen size, and weight is Gaming type

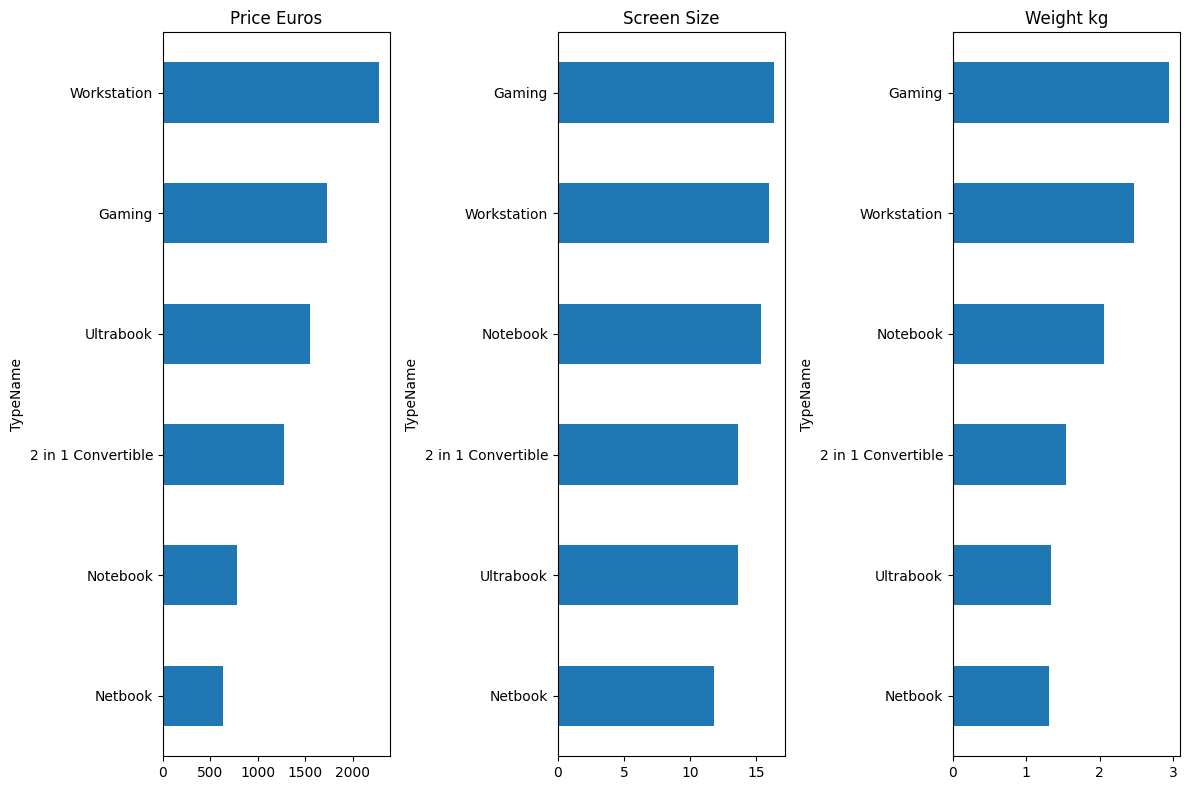
- Workstation is the highest minimum price, screen size, and weight than the other but has lower maximum price, screen size, and weight than Gaming

- Netbook is the lowest maximum price and screen size but this type has higher maximum weight than Ultrabook

- Ultrabook has higher maximum price than 2 in 1 Convertible and Netbook, but this type has lower screen size than 2 in 1 Convertible and higher than Netbook and this type is the lowest weight

Then, lets visualize the mean of price euros, screen size, and weight of data based on TypeName.



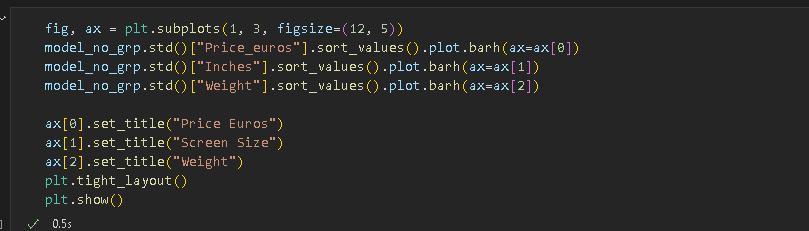


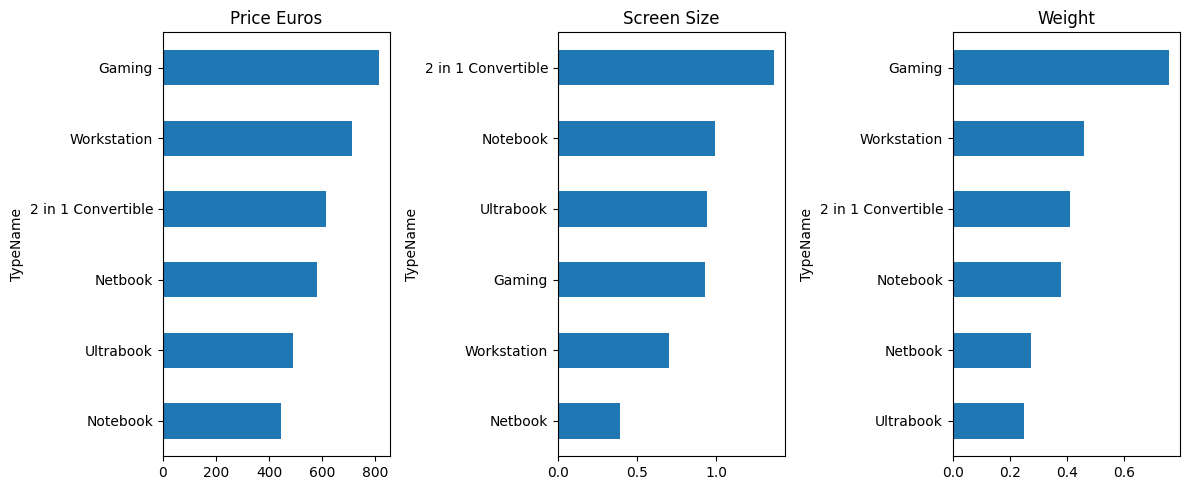
From this, we can say that:

- Workstation is the highest average price but has lower average screen size and weight than Gaming type

- Netbook is the lowest average price, screen size, and weight

After we know about min, max, and mean like above, lets visualize about standard deviation for data dissemination.





From this, we can say that:

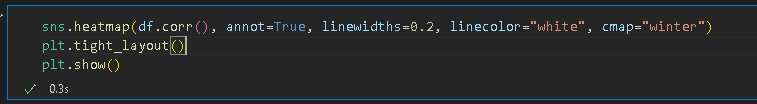
- Gaming is the highest standard deviation in price euros and weight

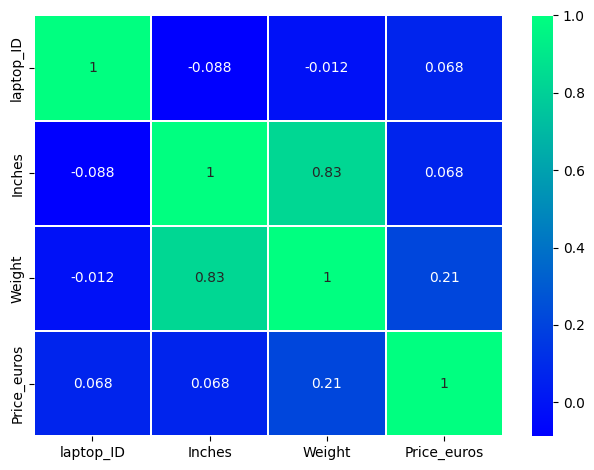
- Netbook is the lowest standard deviation in price euros and screen size but has higher standard deviation in weight than Ultrabook

- 2 in 1 Convertible is the highest standard deviation in screen size

- Workstation is the second highest standard deviation in price euros and weight but has lower standard deviation in screen size than other except Netbook

Now, we know about min, max, mean and standard deviation from this data based on TypeName. After that, lets check about correlation from this data.





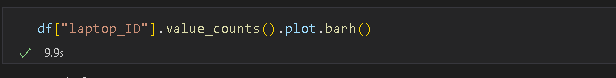
From this, we can say that:

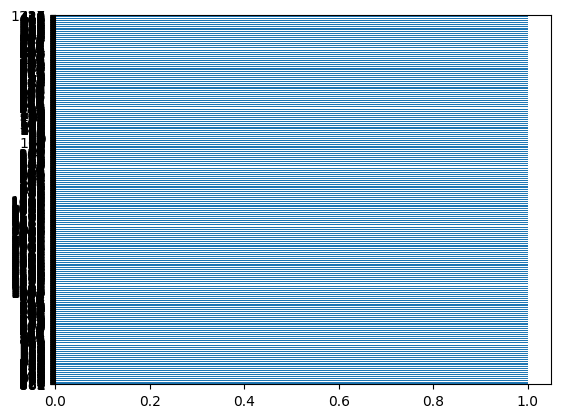
- Laptop\_ID is has weak correlation with other data

- Inches has weak correlation with price but has high correlation with weight

- Weight has weak correlation with price but still higher than inches and has high correlation with inches

We know that Laptop\_ID has high standard deviation but has weak correlation with other data. So, lets check about values in laptop\_ID.



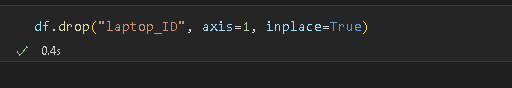


From this, we can say that:

- This data have no unique values. Thats why this data has weak correlation with other data

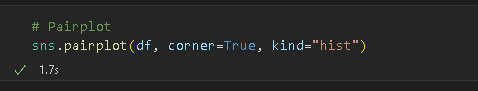
- We dont need this data. So, we can drop this data

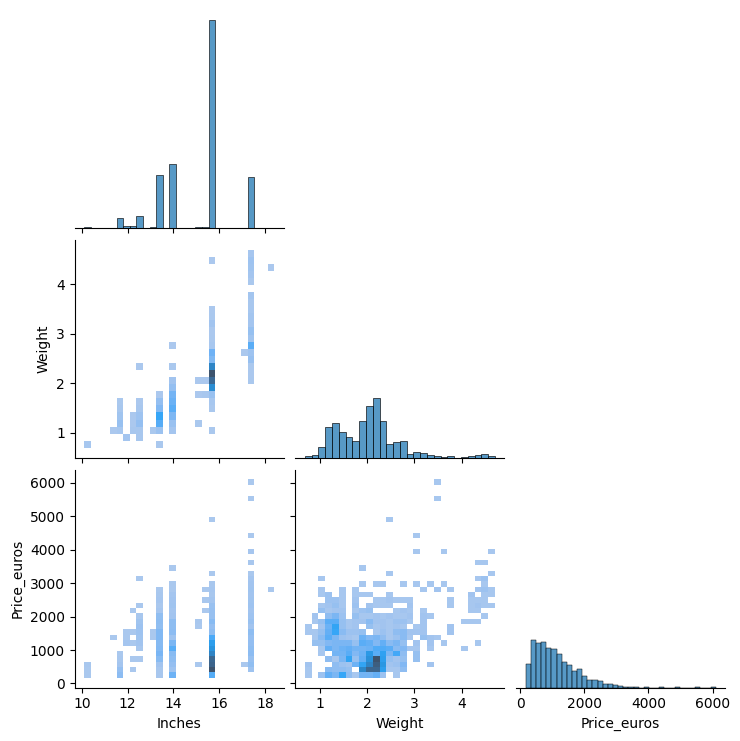
Lets, drop the data.



Now, we finally drop the data of laptop\_ID. We don’t need this data because this data has so much unique values and low correlation with other data.

Then, lets check the pairplot of the data for sure.

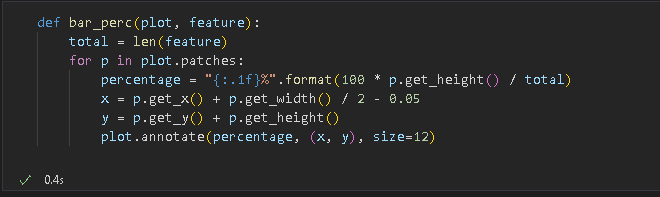


From This, we can say that:

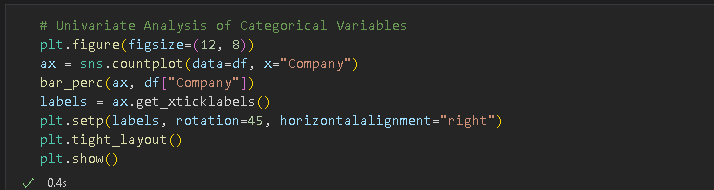
- The data has positive correlation with other data

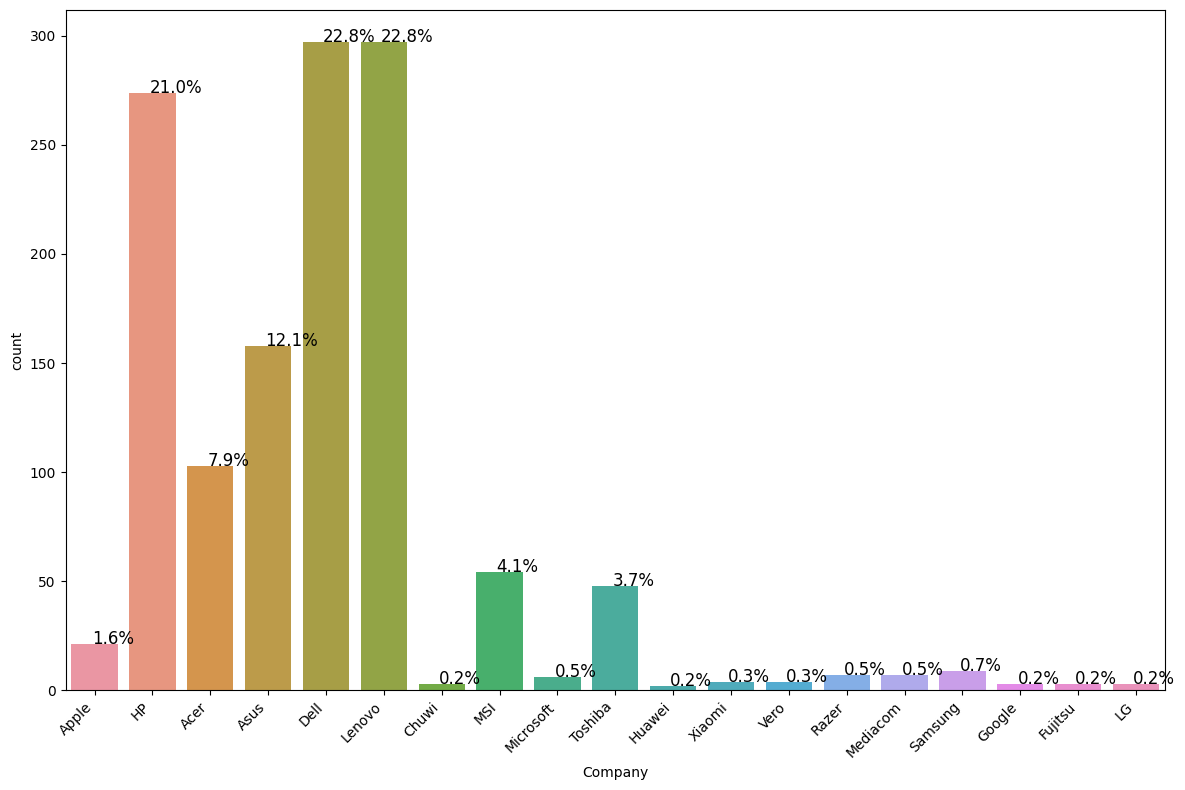
- We get the same insights as from the above heatmap. Need to confirm the same with further analysis.

Lets, make function that can make annotate for the visualization.



Now, lets do the Univariate Analysis for categorical variables like this.



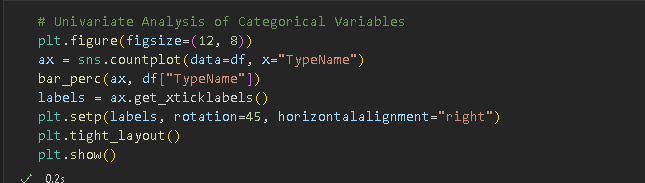


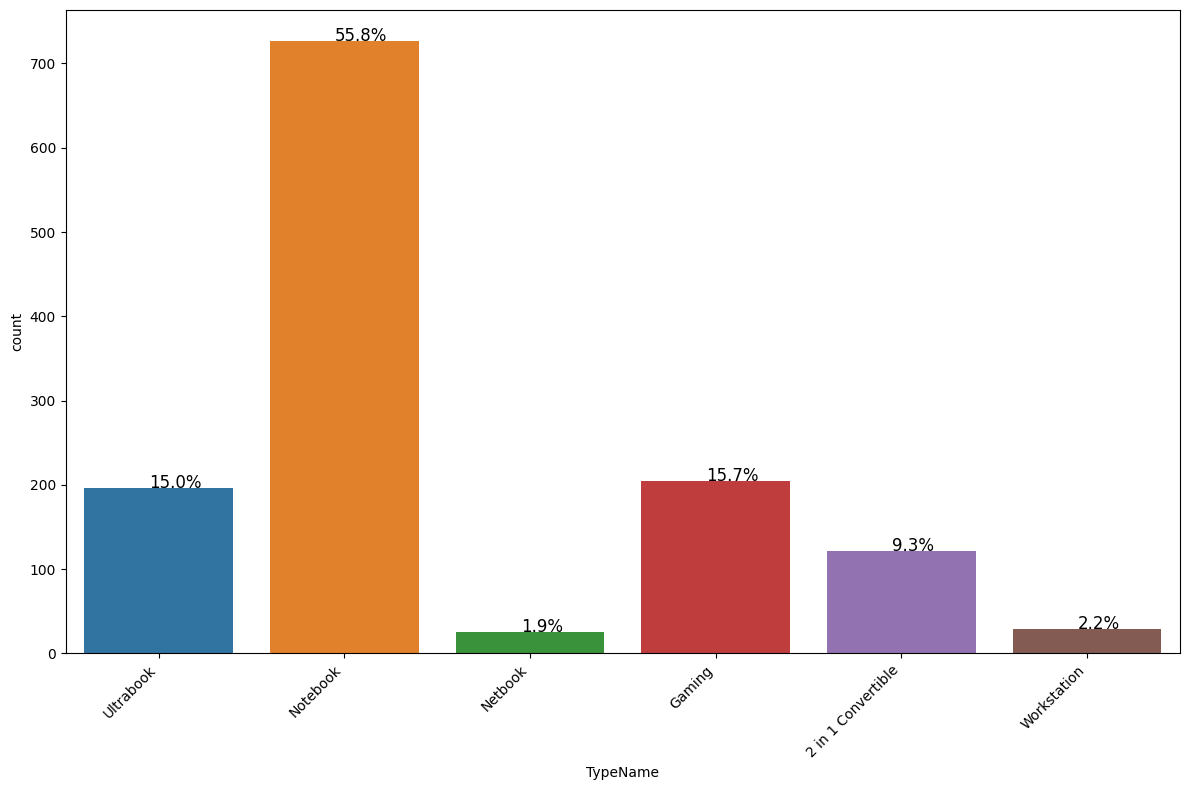
From this, we can say that:

- The company that has highest values is Dell and Lenovo

- The company that has lowest values is LG, Fujitsu, Google, Huawei, and Chuwi

Then, lets visualize the TypeName variable.





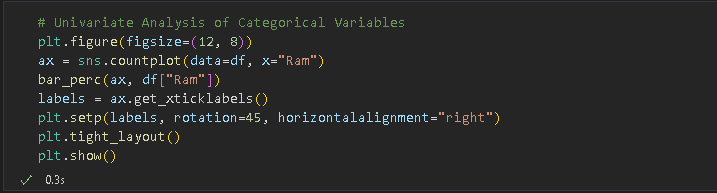
From this, we can say:

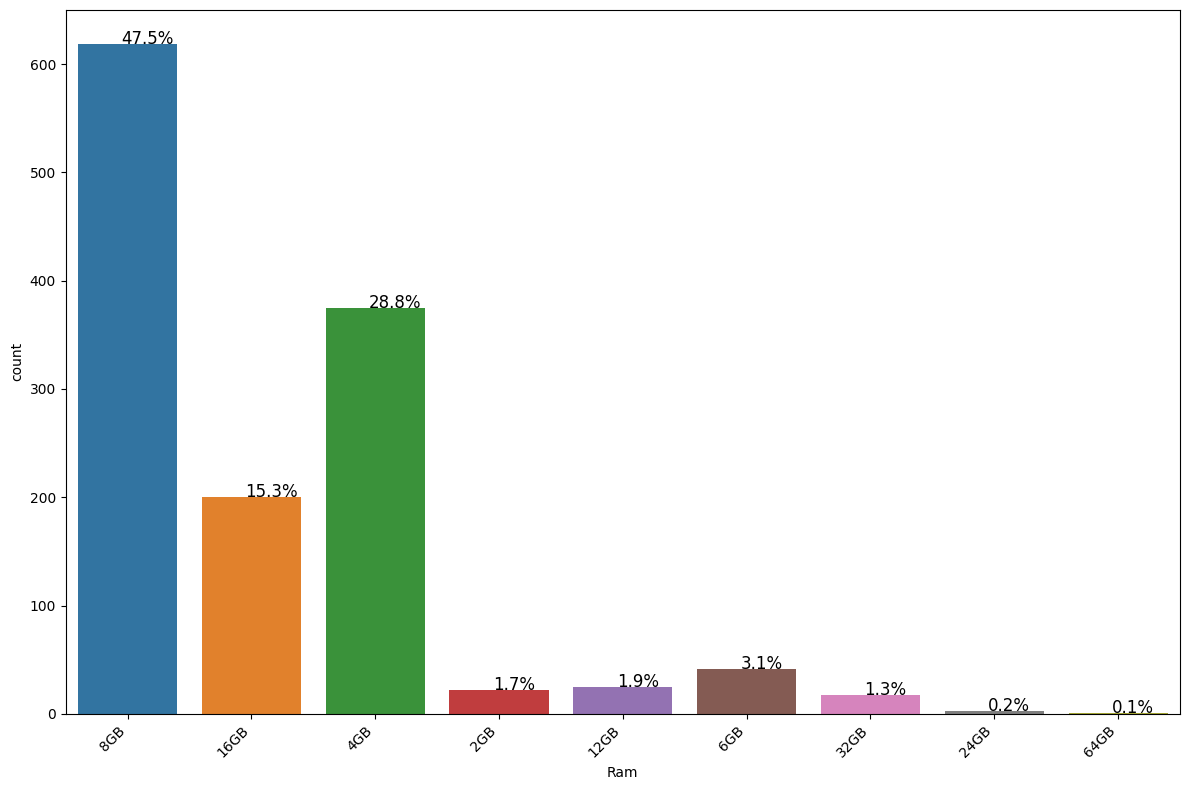
- The data Type which is highest value is Notebook. It means, so much people buy Notebook than other

- The data Type which is lowest value is Netbook. It just 1.9 % from all data TypeName

- Gaming is the second highest value after Notebook

Then, lets visualize the Ram variable.



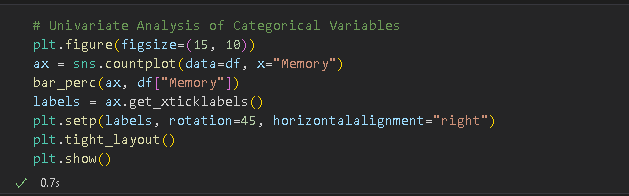


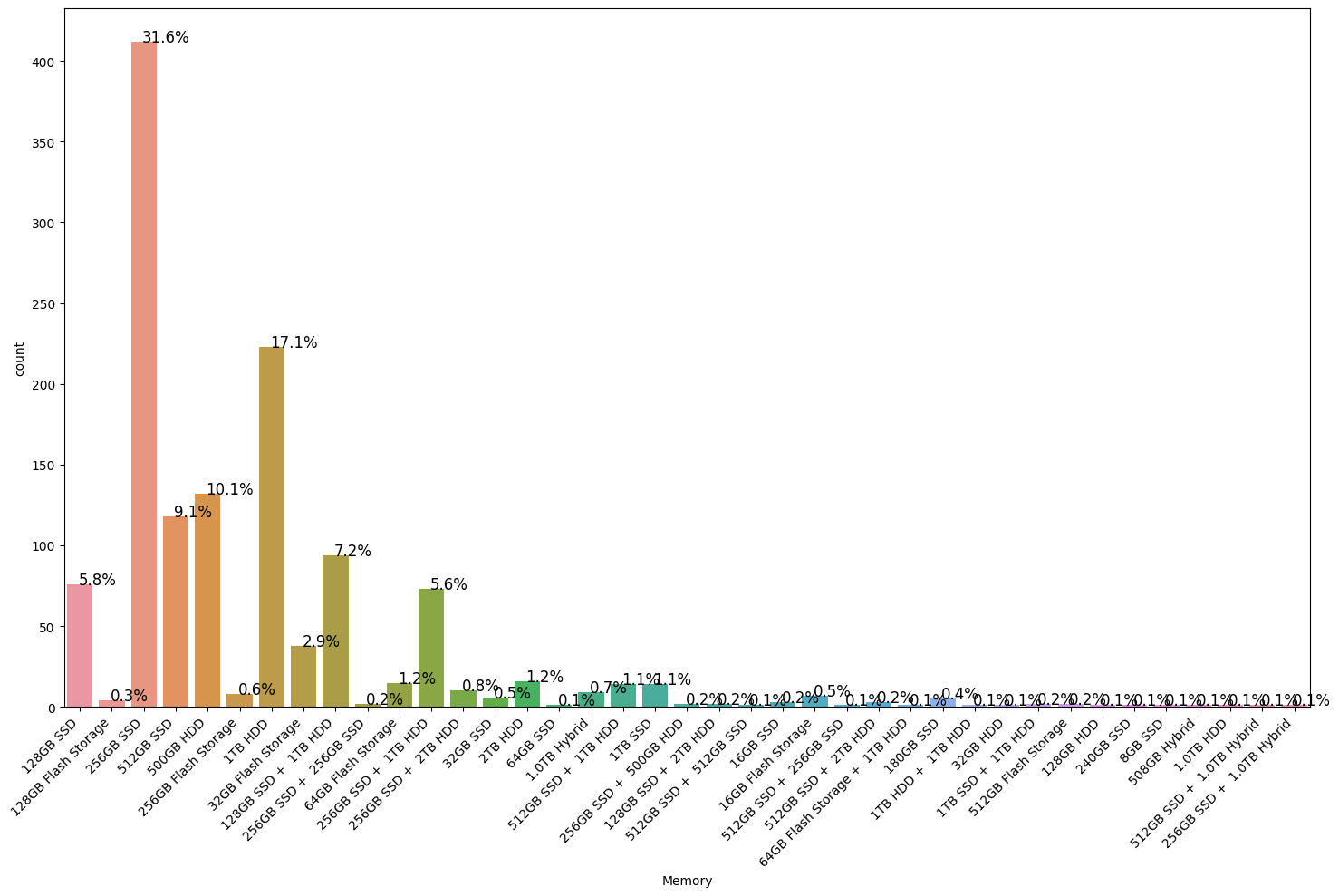
From this, we can say that:

- The data Ram which is highest value is 8GB. It's mean so much people use ram 8GB than other ram

- The data Ram which is lowest value is 64GB. It's 0.1 % people from all data that use 64GB

Then, lets visualize the Memory variable.



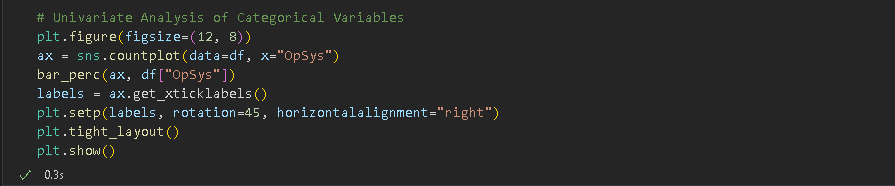


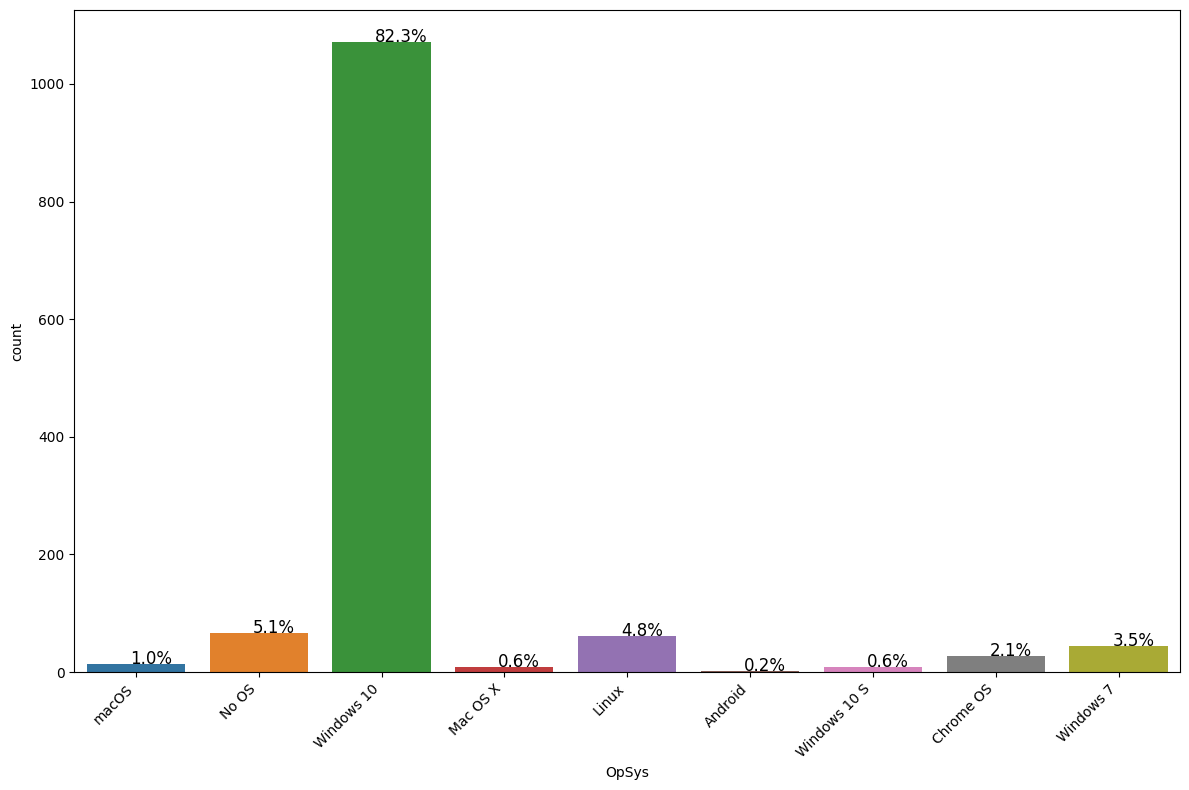
From this, we can say that:

- The data Memory which is highest value is 256GB SSD. It's mean 31.6% people from all data use 256GB SSD

- The data Memory which is lowest value is very much that has just 0.1 % data

Then, lets visualize the Operating System variable.



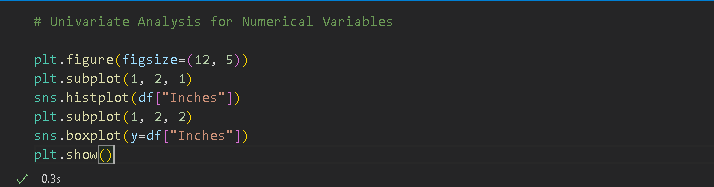


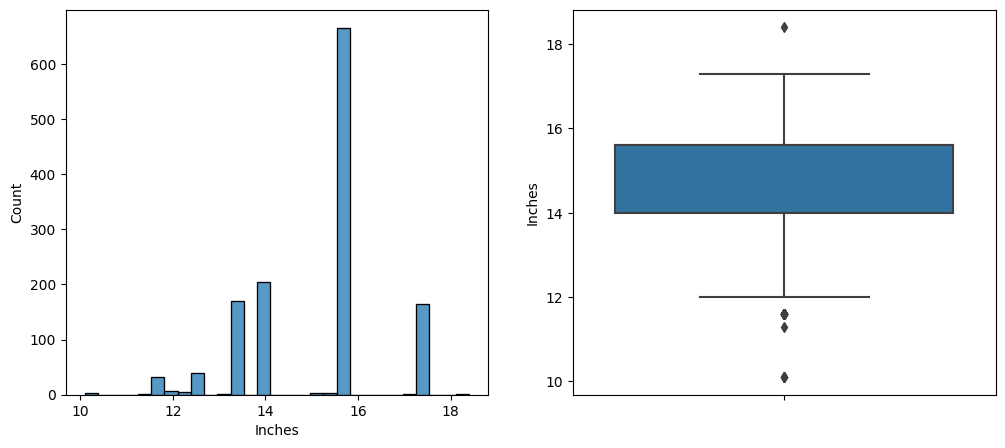
From this, we can say that:

- The data Operation System which is highest value is Windows 10. It's mean 82.3% people from all data use Windows 10

- The data Operation System which is lowest value is Android. It just 0.2 % people from all data use Android

After we know about categorical variables, now lets do the Univariate Analysis for numerical variables. First, lets visualize the Inches or screen size variable.





From this, we can say that:

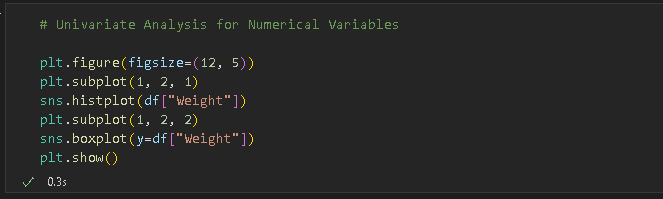
- The highest data from inches data is around 15 - 16

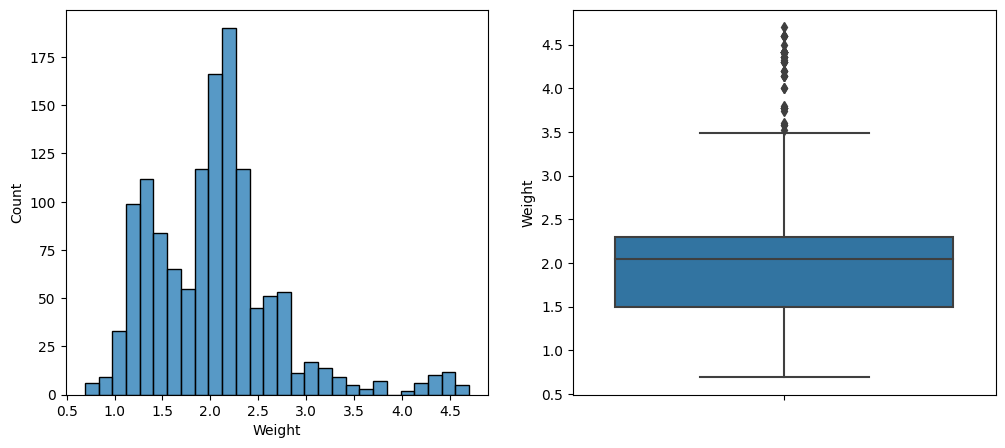
- Data inches has 3 low outlier and 1 high outlier

- The minimum value is 10.1 and the maximum is 18.4

- The interquartile range (IQR) is between 14 and around 15

Then, lets visualize the weight variable.



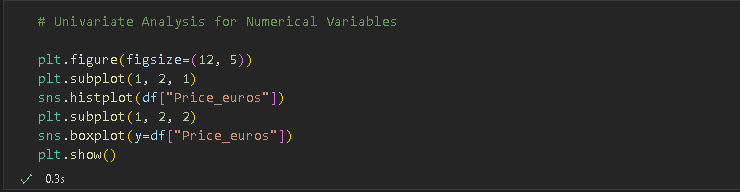


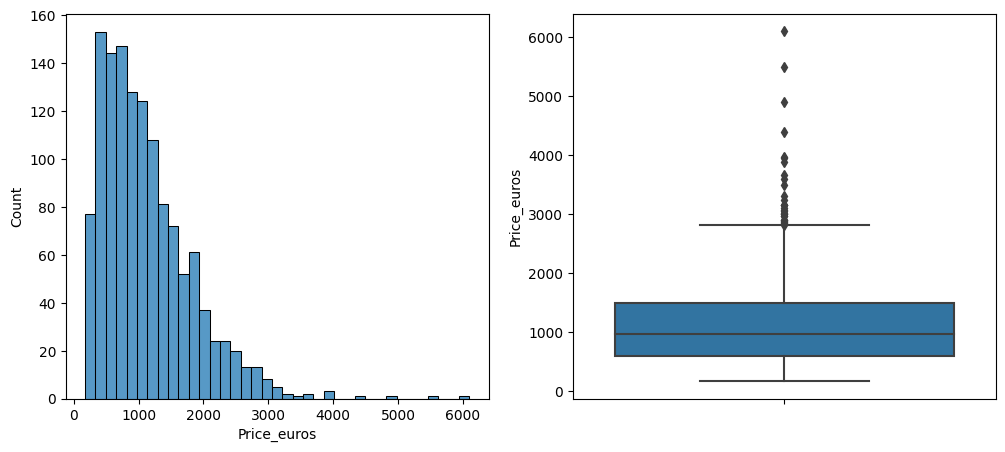
From this, we can say that:

- The minimum value is 0.69 and the maximum value is 4.7

- Data has many high outlier

- The interquartile range (IQR) is between 1.5 and around 2





From this, we can say that:

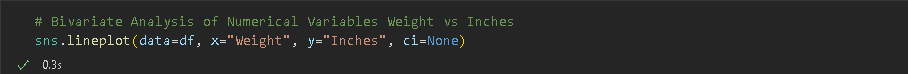
- The minimum value of data is 174.0 and the maximum value of data is 6099.0

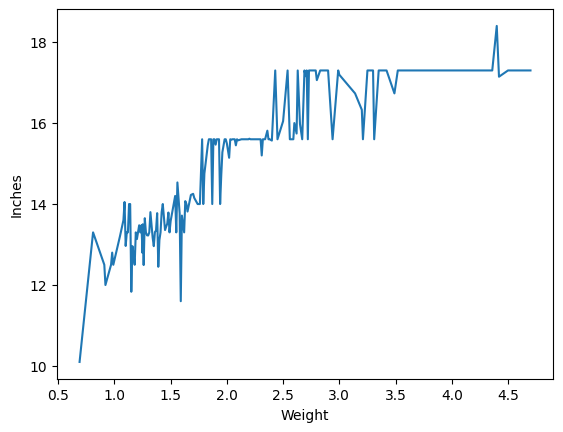
- The highest value of data is 474 which has 174 data

- Many product that has price under 4000 euro

- Data has high outlier

Now, we finally did the Univariate Analysis. Now, lets do the Bivariate Analysis. First, lets do the Bivariate Analysis of weight and inches.





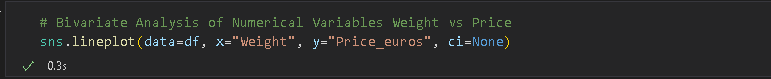
From this, we can say that:

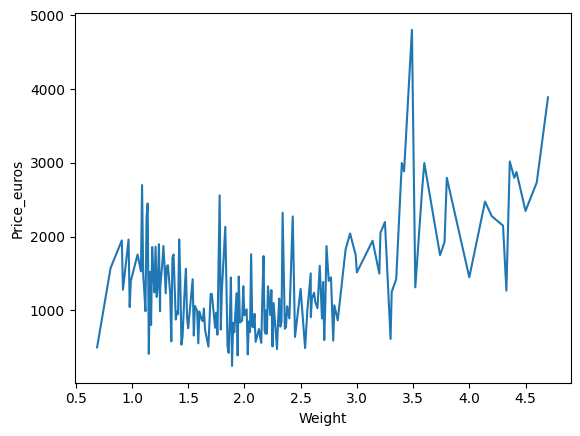
- These 2 data has positive correlation

- The bigger weight has higher inches

- Some high weight has decreased

Then, lets do the Bivariate Analysis of weight and price euros.





From this, we can say that:

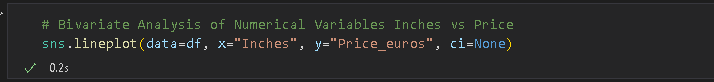
- Lots of data weight around 1.0 kg - 3.0 kg has low price

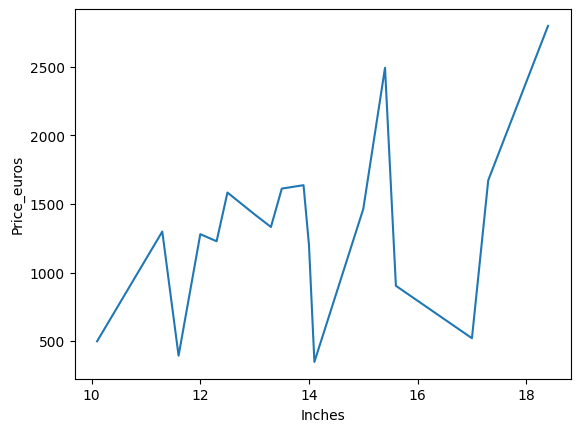
- some data has higher price with weight around 3.4kg - 3.5kg

- Data around 3kg has decreased greatly

- The maximum weight has high price which is above 3000 euro

Then, lets do the Bivariate Analysis of Inches and prices euros.





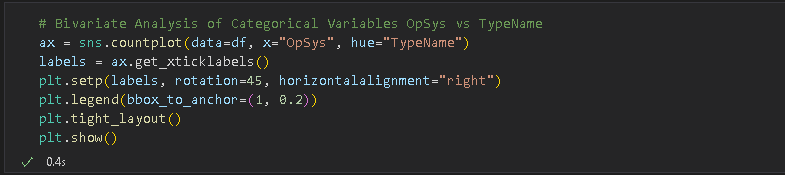
From this, we can say that:

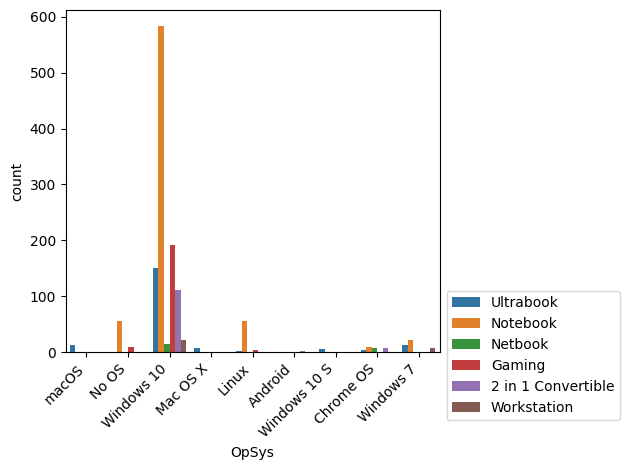
- The minimum of screen size has low price and the maximum of screen size has highest price

- Some data of screen size has low price and get decreased

- lots of data of screen size has high screen size and high price

We did the Bivariate Analysis for numerical variables. Now, lets do the Bivariate Analysis for categorical variables. First, lets do the Bivariate Analysis for Operating System and TypeName Variables.





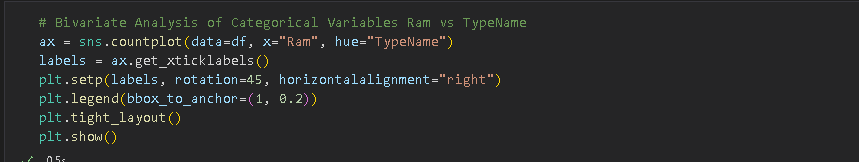
From this, we can say that:

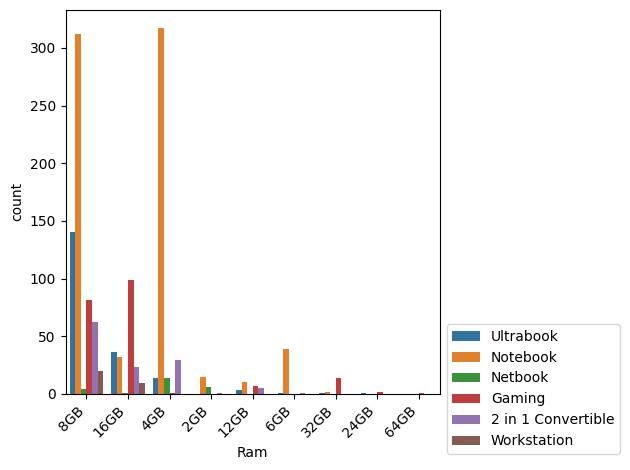
- Lots of people use Windows 10

- The highest type that people use in Windows 10 is Notebook and the lowest type that people use in Windows 10 is Netbook

- Android has just 1 type that people use and it is Workstation

Then, lets do the Bivariate Analysis for Ram and TypeName variables.





From This, we can say that:

- Lots of people use Ram 8GB and the highest is Notebook

- The highest value is Notebook

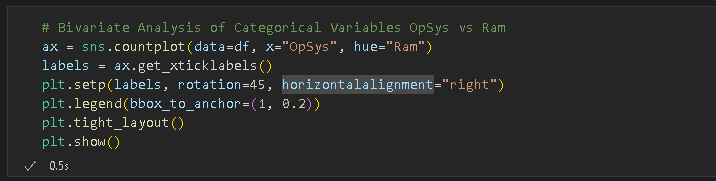
- Lots of people that use Ram 4GB use Notebook

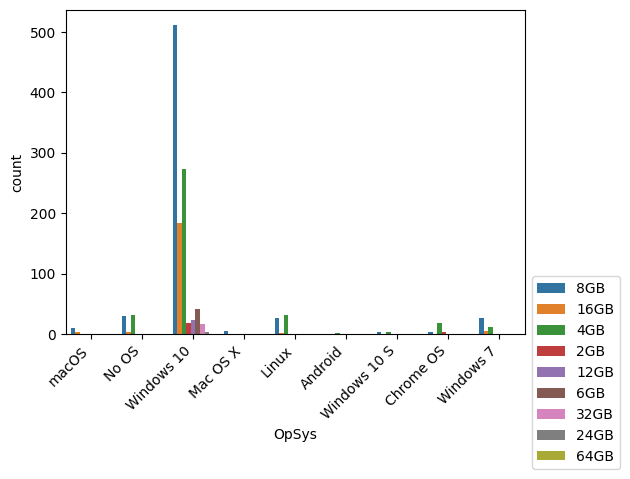
- People who use 64GB use Gaming type

- Lots of people who use above 16GB use Gaming type

- Lots of people who use under 16GB use Notebook

Then, lets do the Bivariate Analysis for Operating System and Ram variables.





From this, we can say that:

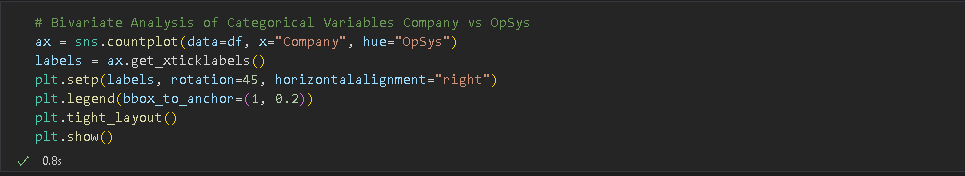
- Lots of people who use macOS, No OS, Windows 10, and Windows 7 prefer use 8GB than other

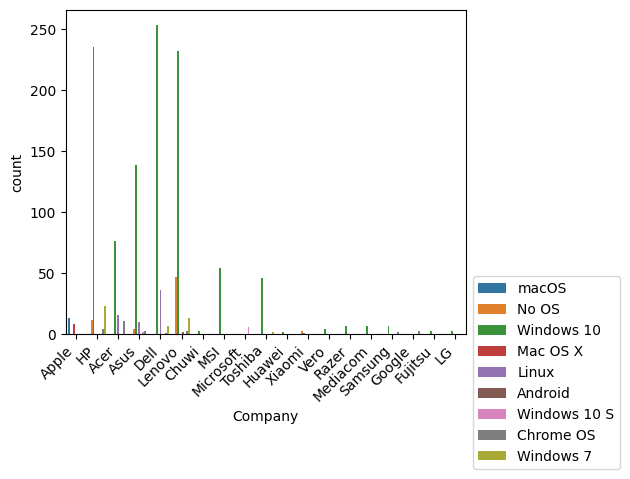
- Most people using Windows 10 with Ram 8GB

- The highest people who use Linux is preper use Ram 4GB than 8GB

- 1 people use Android with Ram 4GB

Then, lets do the Bivariate Analysis for Company and Operating System variables.





From this, we can say that:

- Many companies sell Operation System Windows 10 than other Operation System

- Dell has the highest sell for Windows 10

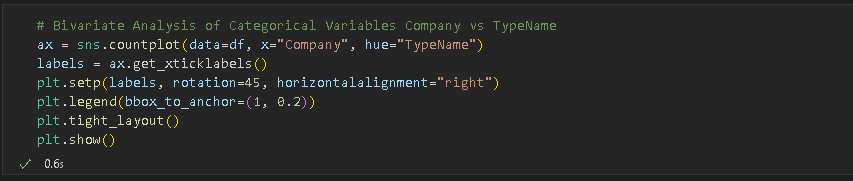
- Lenovo has the highest sell for No OS

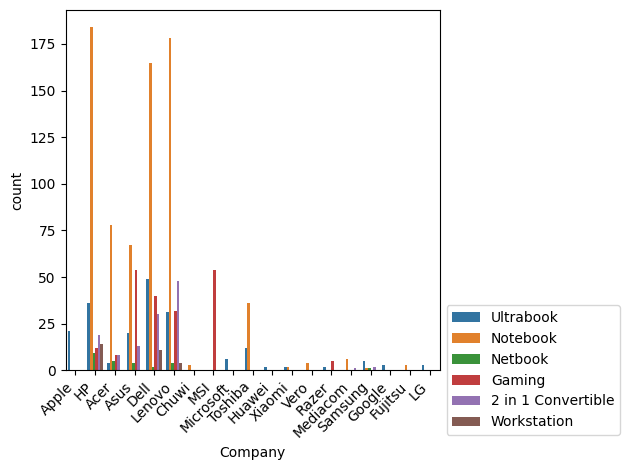
- Apple has the highest sell for macOS and mac OS X

- Dell has the highest sell for Linux

- Microsoft has the highest sell for Windows 10 S

Then, Lets do the Bivariate Analysis for Company and TypeName.





From this, we can say that:

- Many companies sell type of product is Notebook

- HP company is the highest sell for type Notebook and Lenovo company is the second highest sell for type Notebook after HP company

- Apple, Microsoft, Huawei, Google, and LG company just sell for type Ultrabook

- MSL company is the highest sell for type Gaming

- Dell company is the highest sell for type Ultrabook

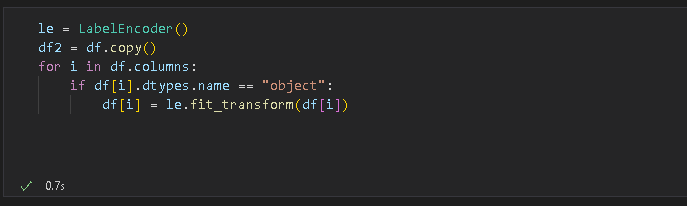
- Hp, Acer, Asus, Dell, Lenovo, and Samsung company sell much type of product

Now, we finally clear the Exploratory Data Analysis (EDA). Lets continue to the next step.

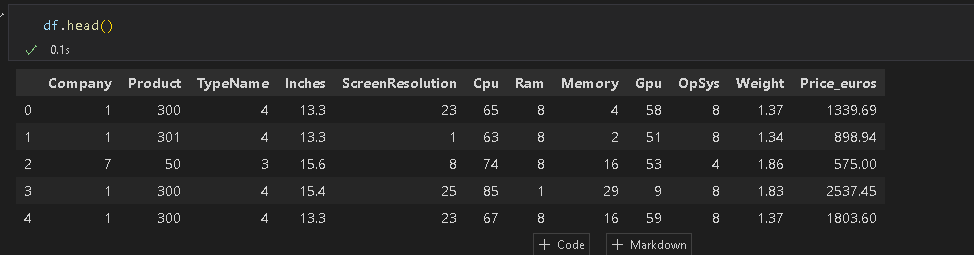
1. Model Building

At this stage, a Data Scientist will create a machine learning model. According to the requirements of this task, the models that I will create are Linear Regression, Ridge, Lasso, Decision Tree, and Support Vector Machine models.

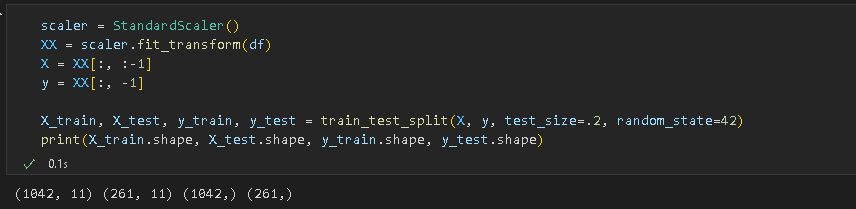
First, lets encode the data object with Label Encoder.



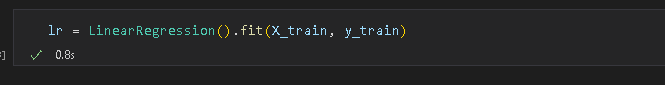
And then, lets check 5 data from top.



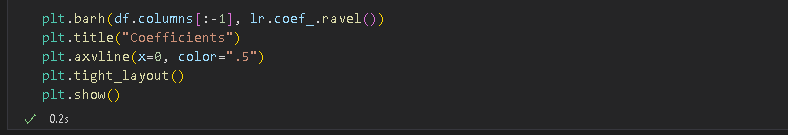
After that, lets transform the data cause the data has high scala with other data and divided to training data and testing data.

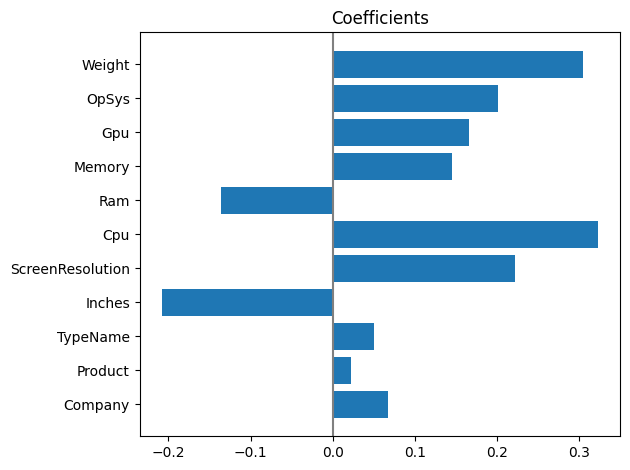


Now, lets make the first model. We make Linear Regression model first like this.



Then, lets check the coefficient of each variables.



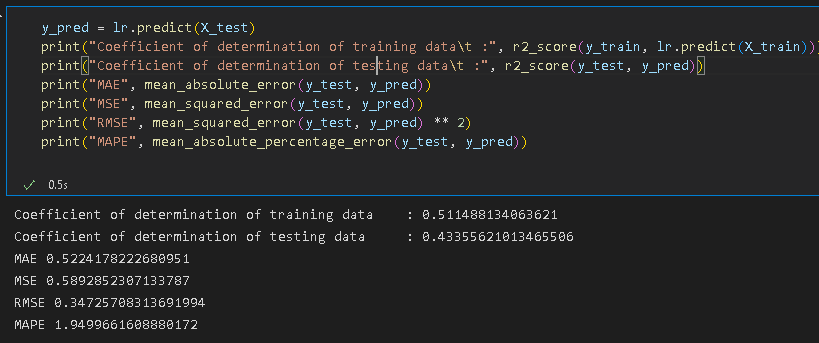


From this we can say that:

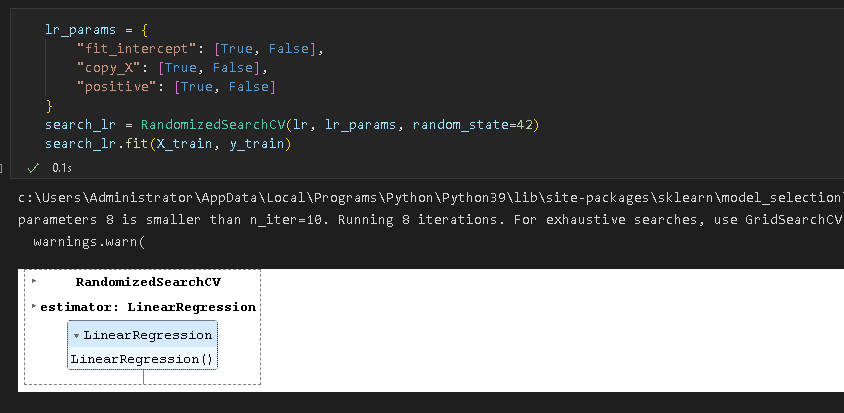
- Weights is the highest coefficient

- Product has no coeffient and it means product dont have any correlation with price

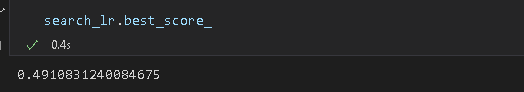
Then, lets check the performance of model.



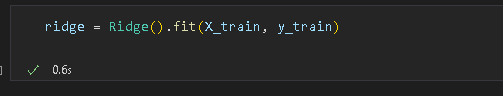
The model has bad performance. Maybe we must set the parameter of model for get performance.



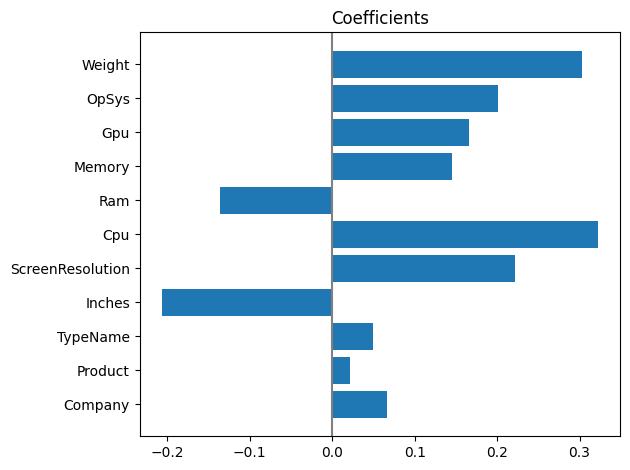
The parameter has built same like before, lets check the best score of model.



The performance of model still bad. Lets make other model like Ridge.



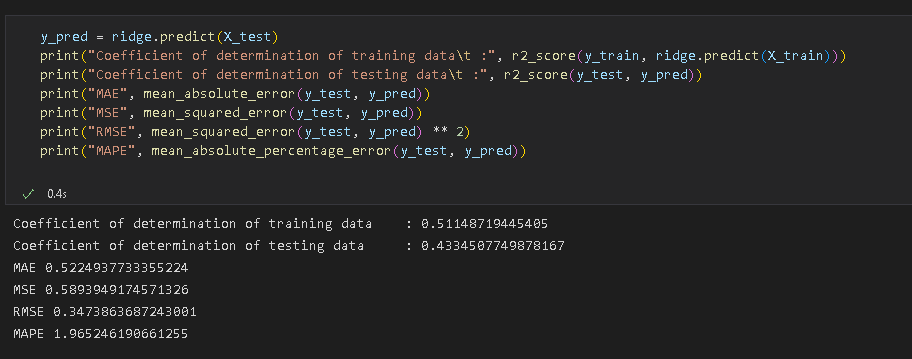
And then, lets check coefficient of each variables.



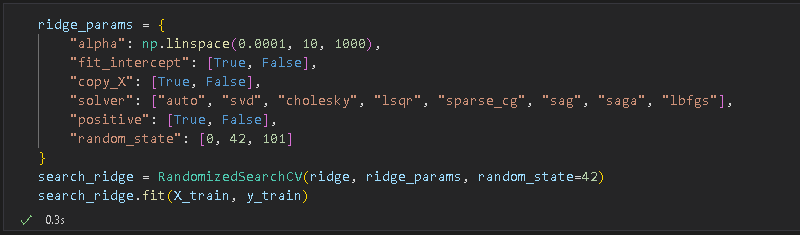
From this, we can say that:

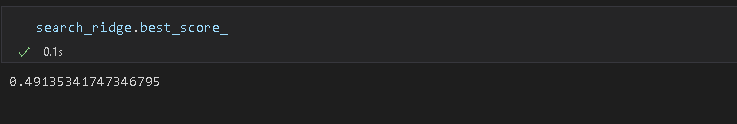
* All variables has coeffient and correlation with the dependen variable
* CPU is the highest variable and Product is the lowest variable

Then, lets check the performance of model.

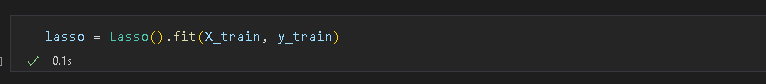


The performance of model is worse than before. Lets set the best parameter of the model.

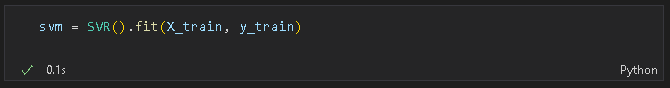




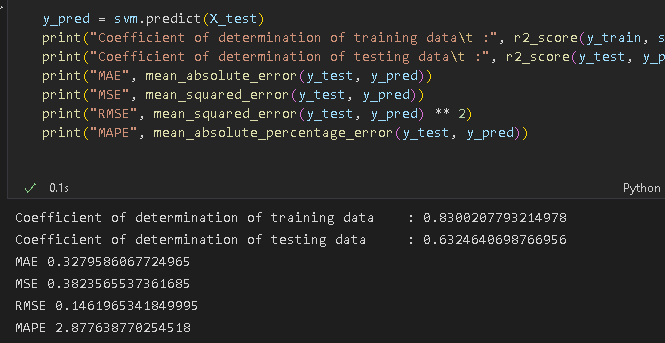
The performance of model still bad and worse then before. Lets make Lasso model.



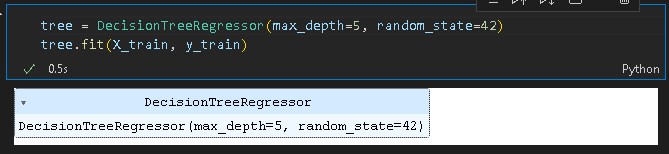
And then, lets make Support Vector Machine model.



Lets, check the evaluation from this model

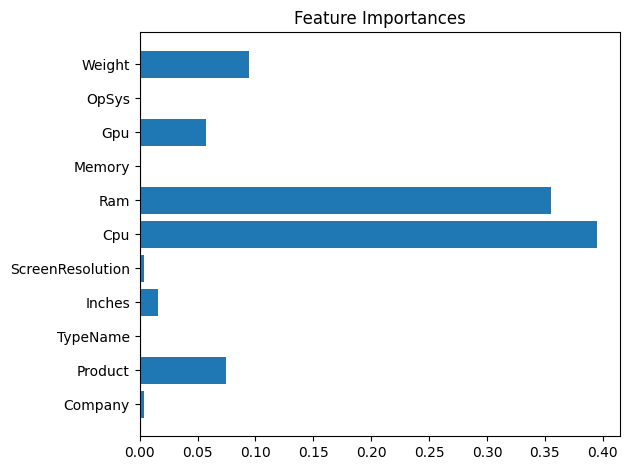


Now, get better model than before. Before we use this model, lets check other model like Decision Tree model.



We made Decision Tree model. Lets check the feature importances.

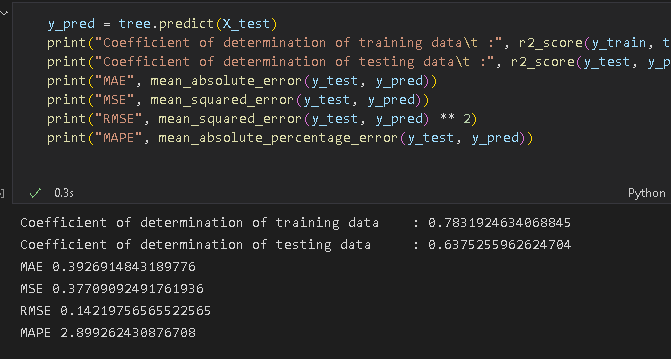




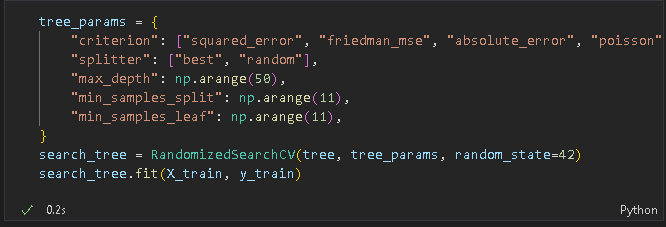
From this, we can say that:

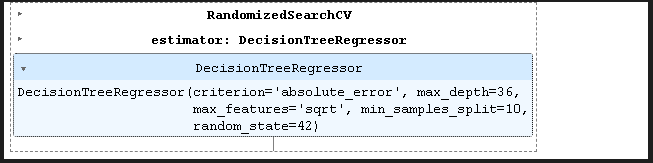
* Some feature don’t have data or 0 feature importances like OpSys, Memory, and TypeName
* Cpu is the highest feature importances
* We know that Weight must be get more feature importances cause that feature have strong correlation with target

Then, lets check the performance of this model.

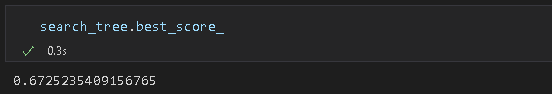


Now, we get best performance than before. We can use this model, too. But, before that, we can test of any parameter with this model with RandomizedSearchCV.

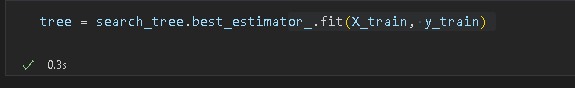




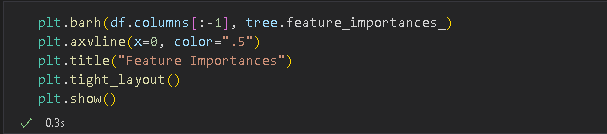
Lets check the best score from this model.

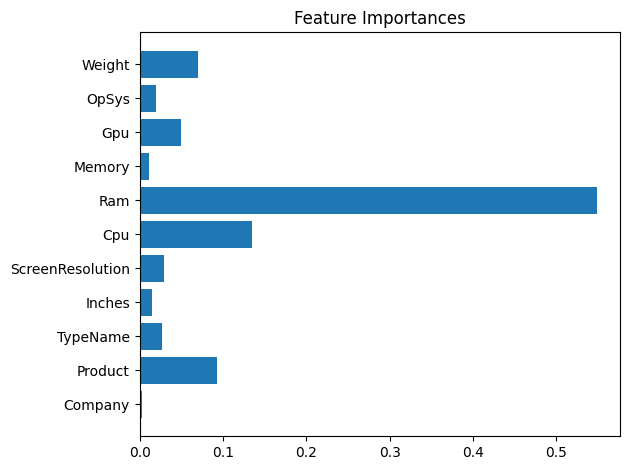


We get nice best score. Now, lets make Decision Tree model again.



After that, lets check the feature importances again.

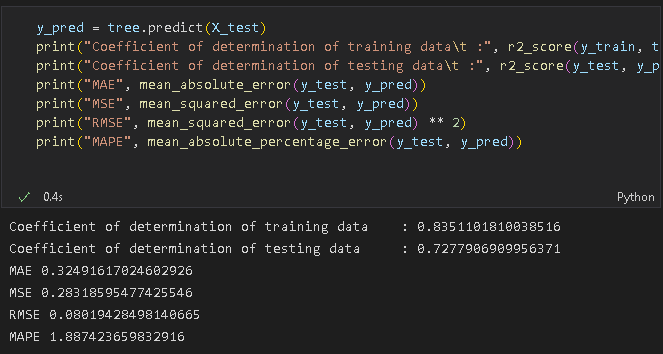




From this, we can say that:

* Now, all feature is importance for this data
* Ram is the highest feature importance for this data
* OpSys, Memory, and TypeName have feature importance now. Its mean these feature is importance too for this data
* Company is the lowest feature importances

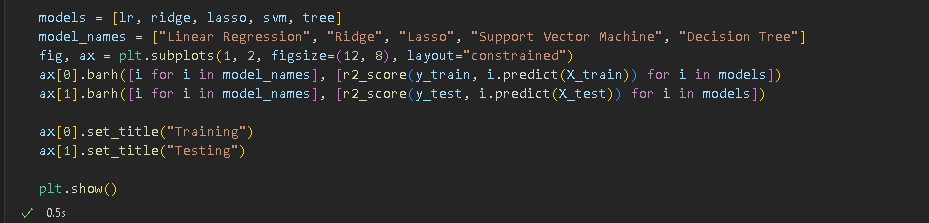
Then, lets check the performance of this model.

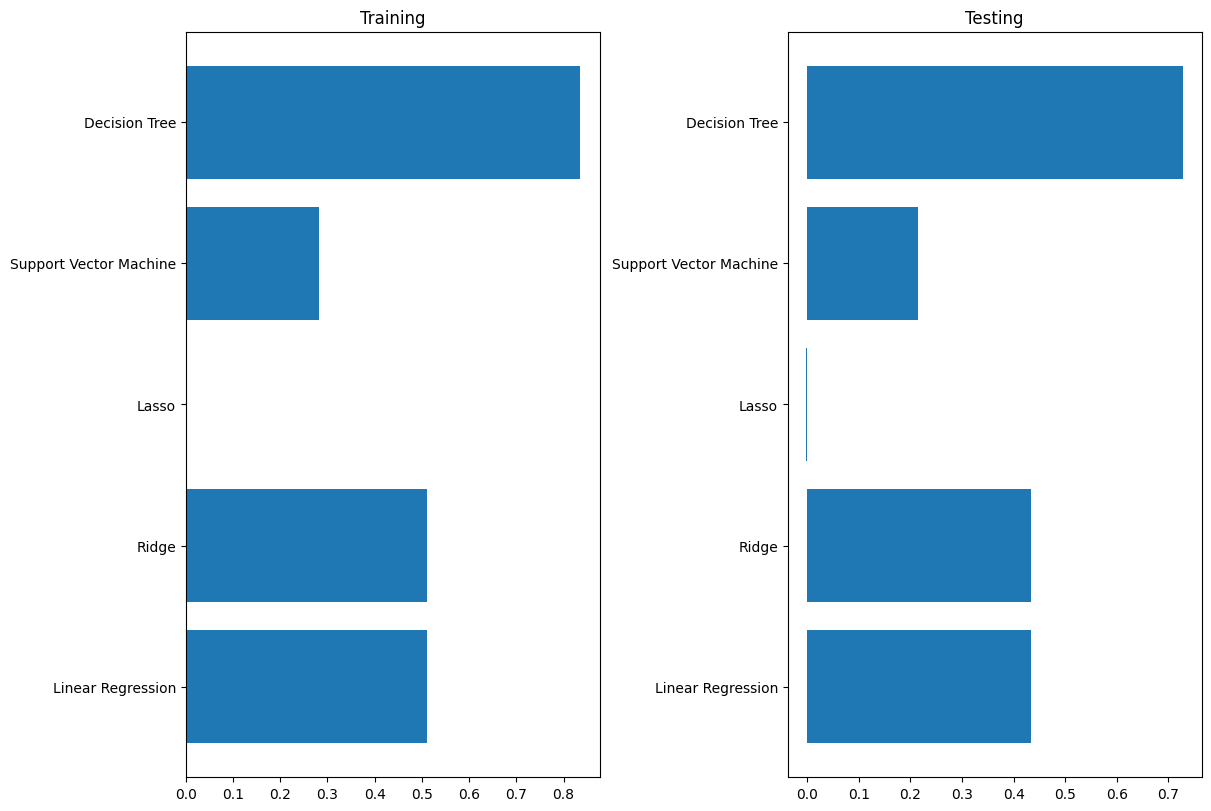


The performance of this model is better than the other model that we built.

1. Evaluation Model

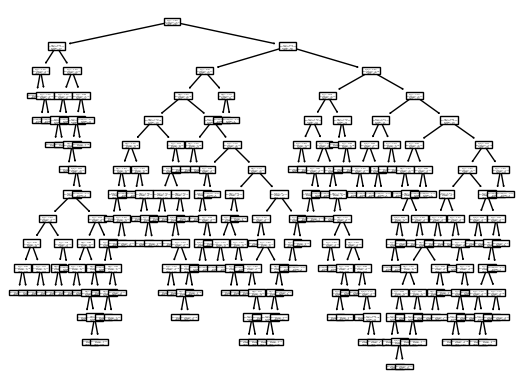
We have built many model like Linear Regression, Ridge, Lasso, Support Vector Machine, and Decision Tree. Now, we have to choose which model is better than the other model that we built. We can do the comparison of model first like this.





As we can see graphic above, we can conclude that, we can use this Decision Tree model for sure. We can also check why this Decision Tree model has higher performance than the other model.





We can understand why this model has higher parameter than the other model from this plot tree