Joint-Human Machine Learning

Assignment 1

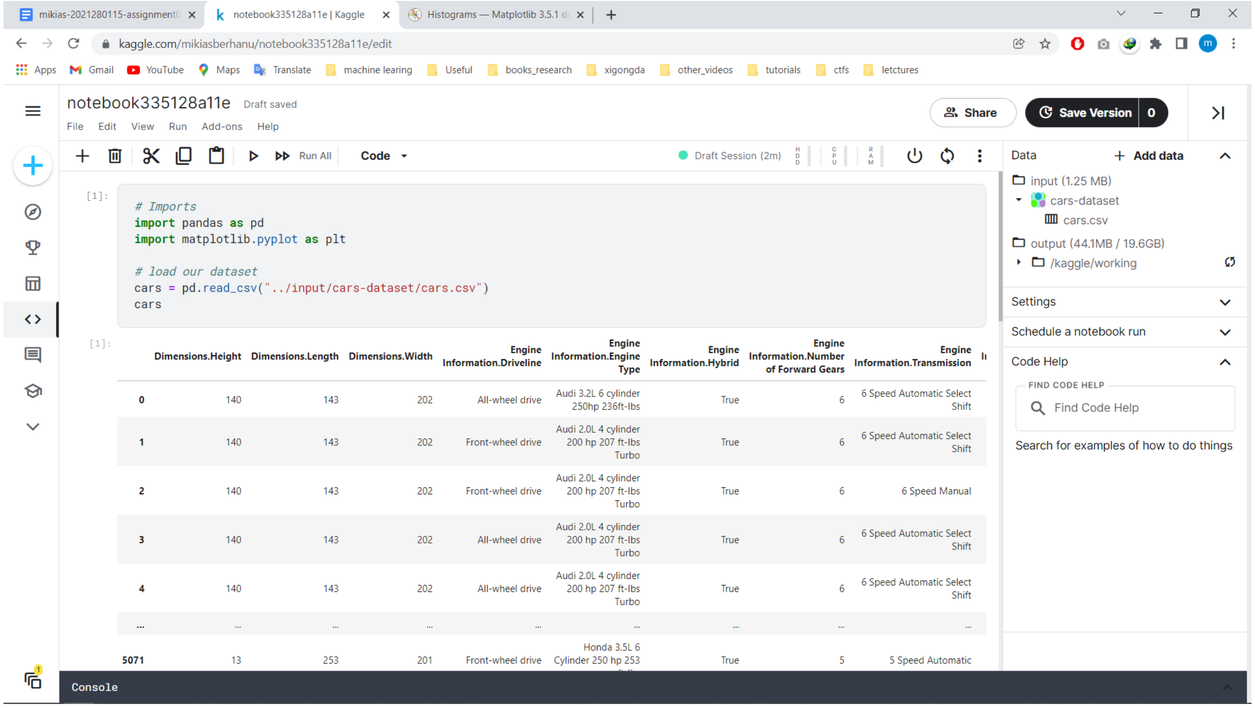
Mikias Berhanu, 2021280115

Data Visualization using Charts and Different Plots

Data Visualization is a way of representing data in a graphic form which is an efficient way to describe data especially when it's in the form of time series. Basically it's a process by which a large amount of data and metrics is translated into charts and graphs. Data visualization is important when it comes to understanding data and getting insights from it. Data visualization makes it easier for the human brain to understand patterns, trends, outliers etc… For this assignment I will use python programming language and a popular python library called Matplotlib which is used for plotting different graphs, charts and more and also pandas library which is used for reading and manipulating datasets with different formats. I'm going to use the kaggle notebook which is an online platform used for running data science and machine learning projects.

Lab Assignment 1

The first thing to do is to load our dataset to our workspace using the pandas library since the dataset is in the form of csv we can use the read csv method.

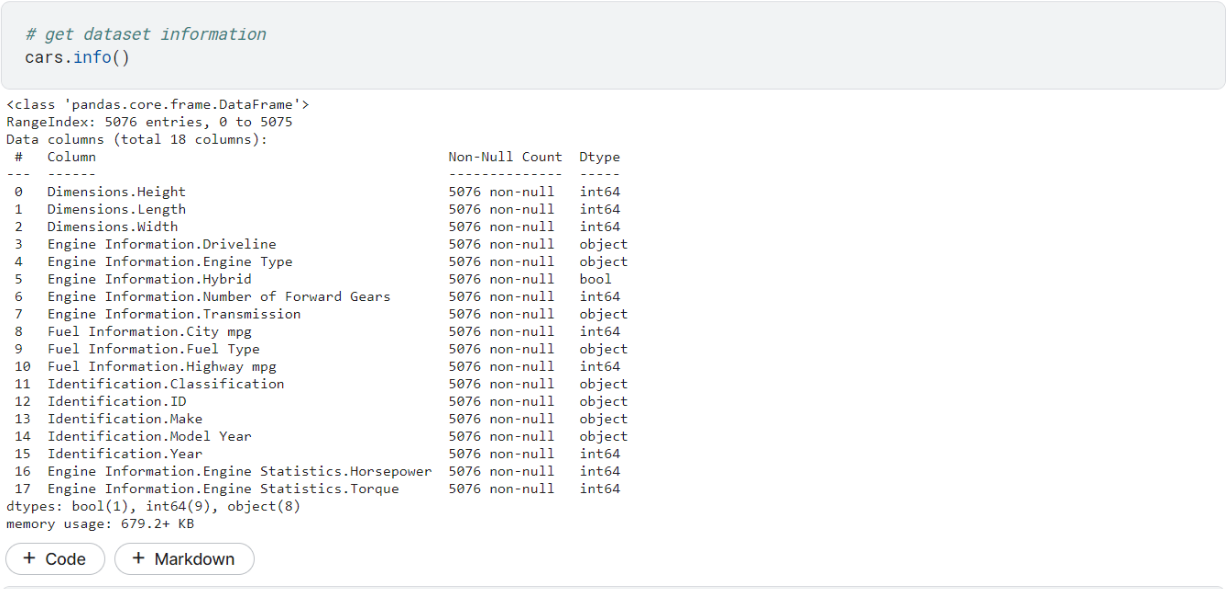


```
[1]:
# Imports
import pandas as pd
import matplotlib.pyplot as plt

# load our dataset
cars = pd.read_csv("../input/cars-dataset/cars.csv")
cars
```

[1]:

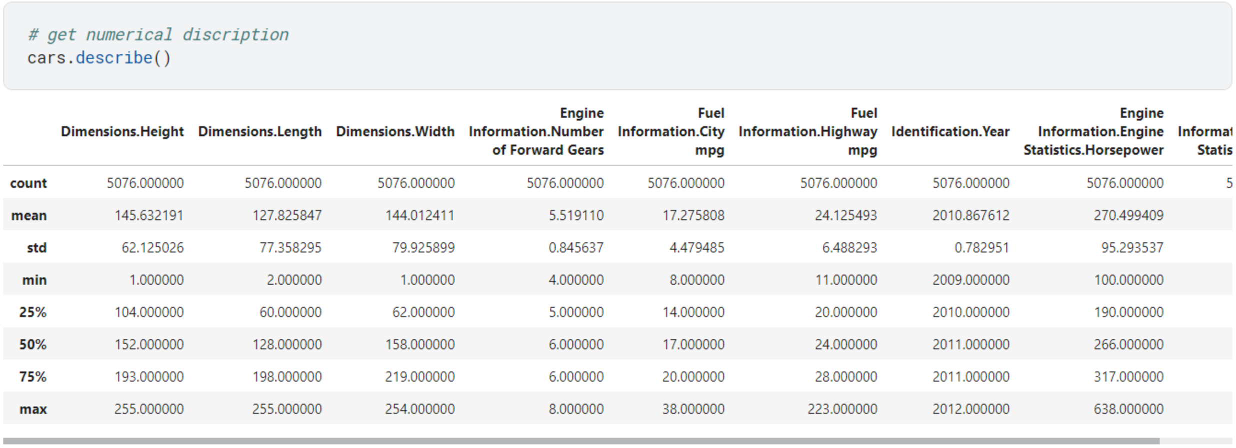
	Dimensions.Height	Dimensions.Length	Dimensions.Width	Engine Information.Driveline	Engine Information.Engine Type	Engine Information.Hybrid	Engine Information.Number of Forward Gears	Engine Information.Transmission
0	140	143	202	All-wheel drive	Audi 3.2L 6 cylinder 250hp 236ft-lbs	True	6	6 Speed Automatic Select Shift
1	140	143	202	Front-wheel drive	Audi 2.0L 4 cylinder 200 hp 207 ft-lbs Turbo	True	6	6 Speed Automatic Select Shift
2	140	143	202	Front-wheel drive	Audi 2.0L 4 cylinder 200 hp 207 ft-lbs Turbo	True	6	6 Speed Manual
3	140	143	202	All-wheel drive	Audi 2.0L 4 cylinder 200 hp 207 ft-lbs Turbo	True	6	6 Speed Automatic Select Shift
4	140	143	202	All-wheel drive	Audi 2.0L 4 cylinder 200 hp 207 ft-lbs Turbo	True	6	6 Speed Automatic Select Shift
...	...	...	...	...	...	...	...	...
5071	13	253	201	Front-wheel drive	Honda 3.5L 6 Cylinder 250 hp 253	True	5	5 Speed Automatic

Next we can get information and numerical description of our dataset using panda’s functions which makes everything easy to use. The info method tells us basic information about the dataset whereas the describe method tells the numeric structure of the dataset.



```
# get dataset information
cars.info()
```

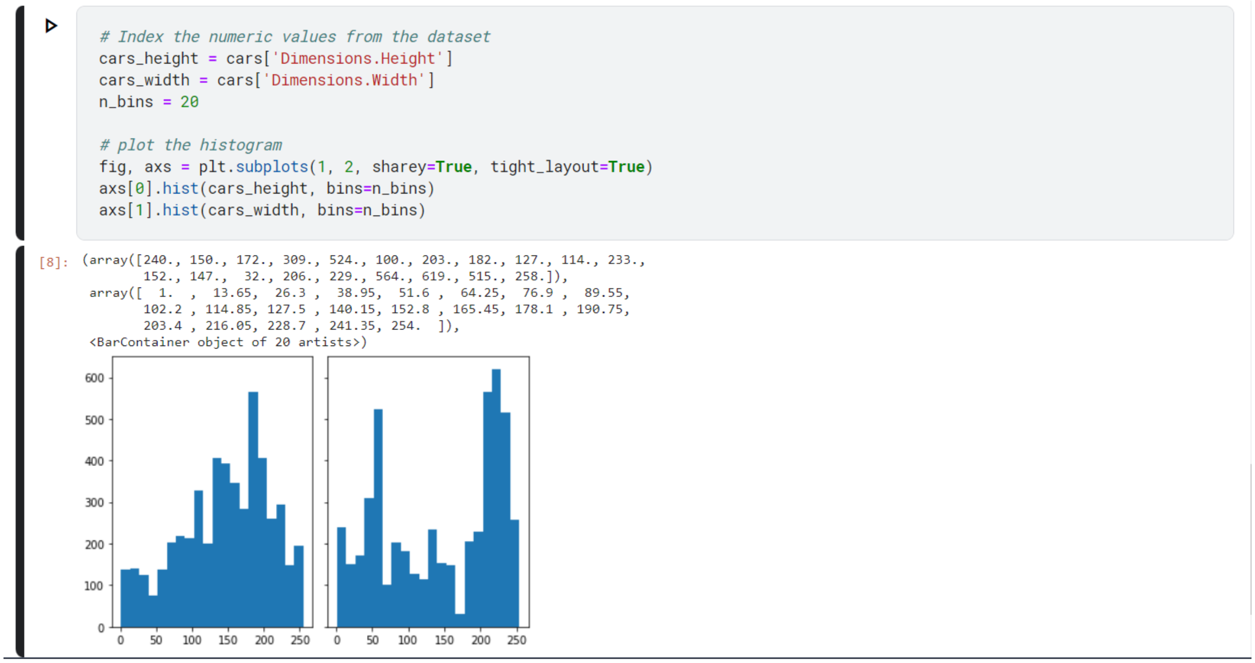
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5076 entries, 0 to 5075
Data columns (total 18 columns):
 #   Column                                      Non-Null Count  Dtype 
---  ------                                      --------------  ----- 
 0   Dimensions.Height                           5076 non-null   int64 
 1   Dimensions.Length                           5076 non-null   int64 
 2   Dimensions.Width                            5076 non-null   int64 
 3   Engine Information.Driveline                5076 non-null   object
 4   Engine Information.Engine Type              5076 non-null   object
 5   Engine Information.Hybrid                   5076 non-null   bool  
 6   Engine Information.Number of Forward Gears  5076 non-null   int64 
 7   Engine Information.Transmission             5076 non-null   object
 8   Fuel Information.City mpg                   5076 non-null   int64 
 9   Fuel Information.Fuel Type                  5076 non-null   object
 10  Fuel Information.Highway mpg                5076 non-null   int64 
 11  Identification.Classification               5076 non-null   object
 12  Identification.ID                           5076 non-null   object
 13  Identification.Make                         5076 non-null   object
 14  Identification.Model Year                   5076 non-null   object
 15  Identification.Year                         5076 non-null   int64 
 16  Engine Information.Engine Statistics.Horsepower  5076 non-null   int64 
 17  Engine Information.Engine Statistics.Torque     5076 non-null   int64 
dtypes: bool(1), int64(9), object(8)
memory usage: 679.2+ KB
```



```
# get numerical discription
cars.describe()
```

	Dimensions.Height	Dimensions.Length	Dimensions.Width	Engine Information.Number of Forward Gears	Fuel Information.City mpg	Fuel Information.Highway mpg	Identification.Year	Engine Information.Engine Statistics.Horsepower	Informat Statis
count	5076.000000	5076.000000	5076.000000	5076.000000	5076.000000	5076.000000	5076.000000	5076.000000	5
mean	145.632191	127.825847	144.012411	5.519110	17.275808	24.125493	2010.867612	270.499409	
std	62.125026	77.358295	79.925899	0.845637	4.479485	6.488293	0.782951	95.293537	
min	1.000000	2.000000	1.000000	4.000000	8.000000	11.000000	2009.000000	100.000000	
25%	104.000000	60.000000	62.000000	5.000000	14.000000	20.000000	2010.000000	190.000000	
50%	152.000000	128.000000	158.000000	6.000000	17.000000	24.000000	2011.000000	266.000000	
75%	193.000000	198.000000	219.000000	6.000000	20.000000	28.000000	2011.000000	317.000000	
max	255.000000	255.000000	254.000000	8.000000	38.000000	223.000000	2012.000000	638.000000	

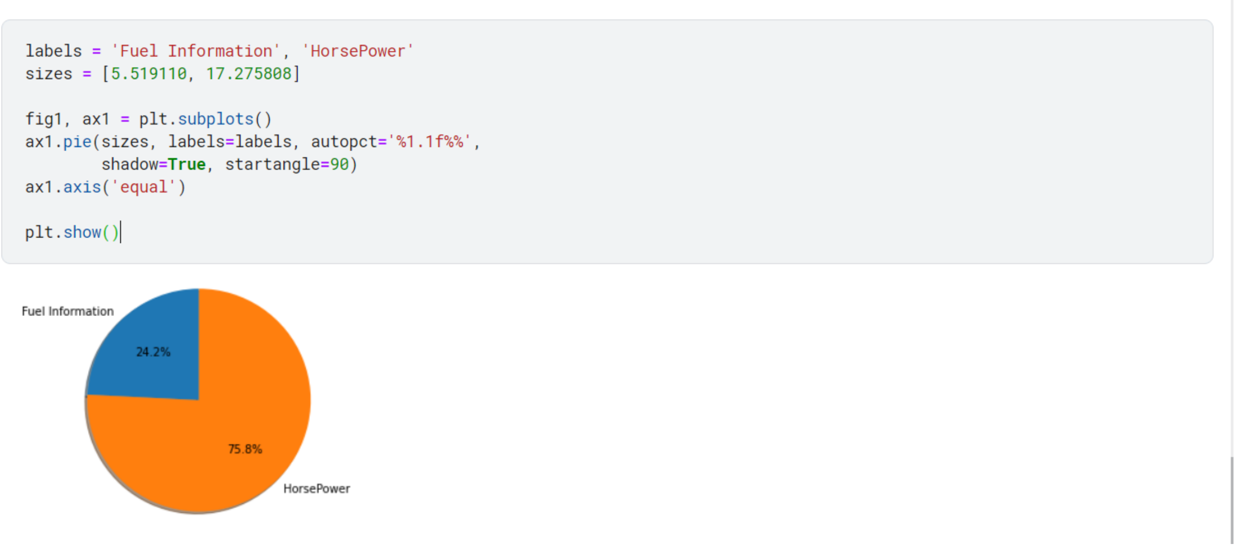
Now we can use the Matplotlib library to plot the histogram for our dataset. As an example if we want to get the histogram for the car’s width and height we can use the hist function from matplotlib. From the figure below we can see that the car's magnitude seems to be higher in some instances compared to the car’s height. The bin values are used to group the numerical data with equal width.



```
# Index the numeric values from the dataset
cars_height = cars['Dimensions.Height']
cars_width = cars['Dimensions.Width']
n_bins = 20

# plot the histogram
fig, axs = plt.subplots(1, 2, sharey=True, tight_layout=True)
axs[0].hist(cars_height, bins=n_bins)
axs[1].hist(cars_width, bins=n_bins)
```

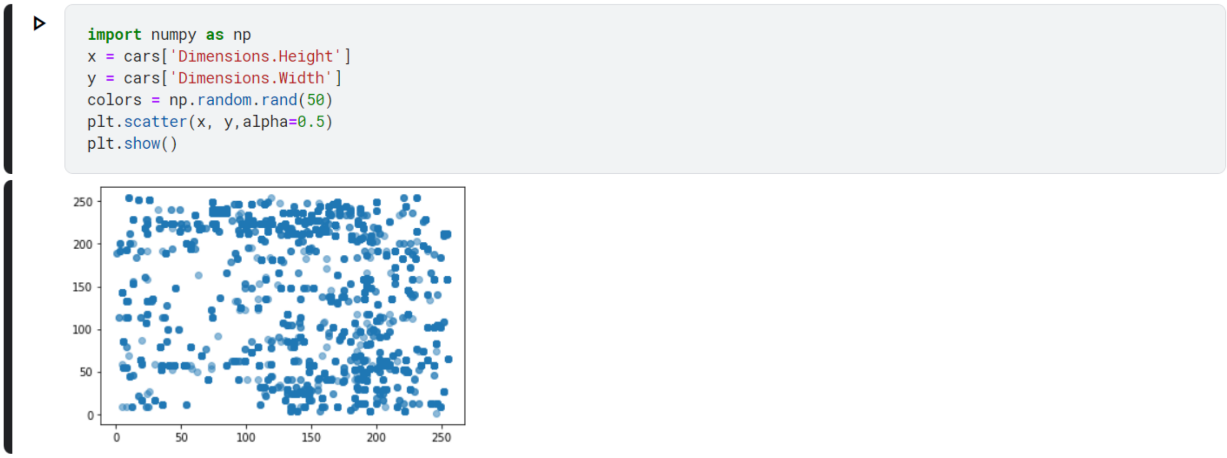
```
[8]: (array([240., 150., 172., 309., 524., 100., 203., 182., 127., 114., 233.,
             152., 147.,  32., 206., 229., 564., 619., 515., 258.]),
      array([  1.  ,  13.65,  26.3 ,  38.95,  51.6 ,  64.25,  76.9 ,  89.55,
             102.2 , 114.85, 127.5 , 140.15, 152.8 , 165.45, 178.1 , 190.75,
             203.4 , 216.05, 228.7 , 241.35, 254.  ]),
      <BarContainer object of 20 artists>)
```

We can also plot a pie chart using the same technique for this case we can plot the car’s Fuel Information and HorsePower. For this I used the mean value of each column as a value for the pie chart, of course the values might change depending on the situation and the purpose of the pie chart.



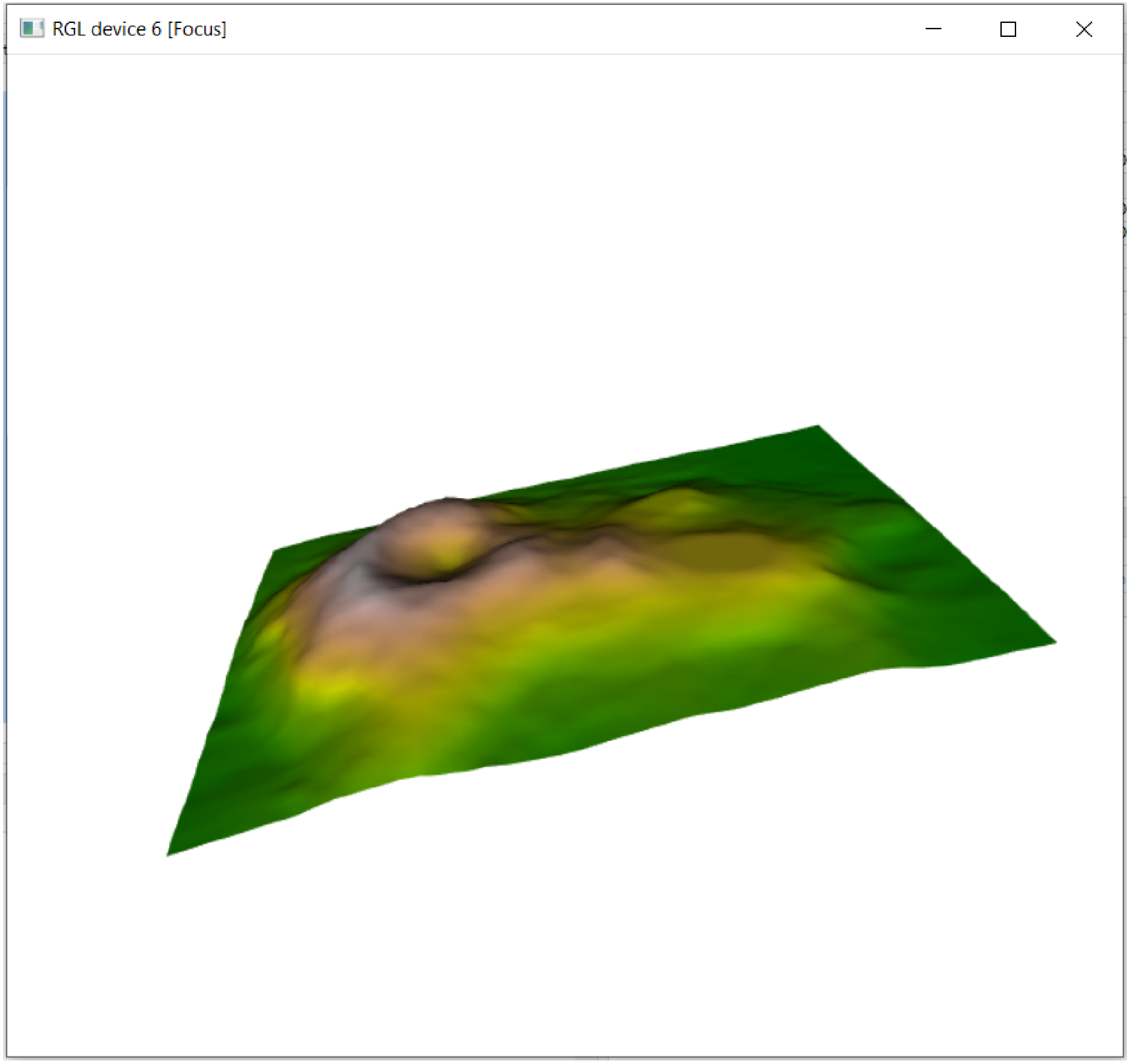
```
labels = 'Fuel Information', 'HorsePower'
sizes = [5.519110, 17.275808]

fig1, ax1 = plt.subplots()
ax1.pie(sizes, labels=labels, autopct='%1.1f%%',
        shadow=True, startangle=90)
ax1.axis('equal')

plt.show()
```

Another graph we can plot is scatter plot which is a mathematical graph where data points are plotted on the cartesian coordinate plane. Usually this is done between two data points; this helps us to observe the relationship between the two variables. Let's plot the scatter plot to see the relationship between a car's width and car’s height.



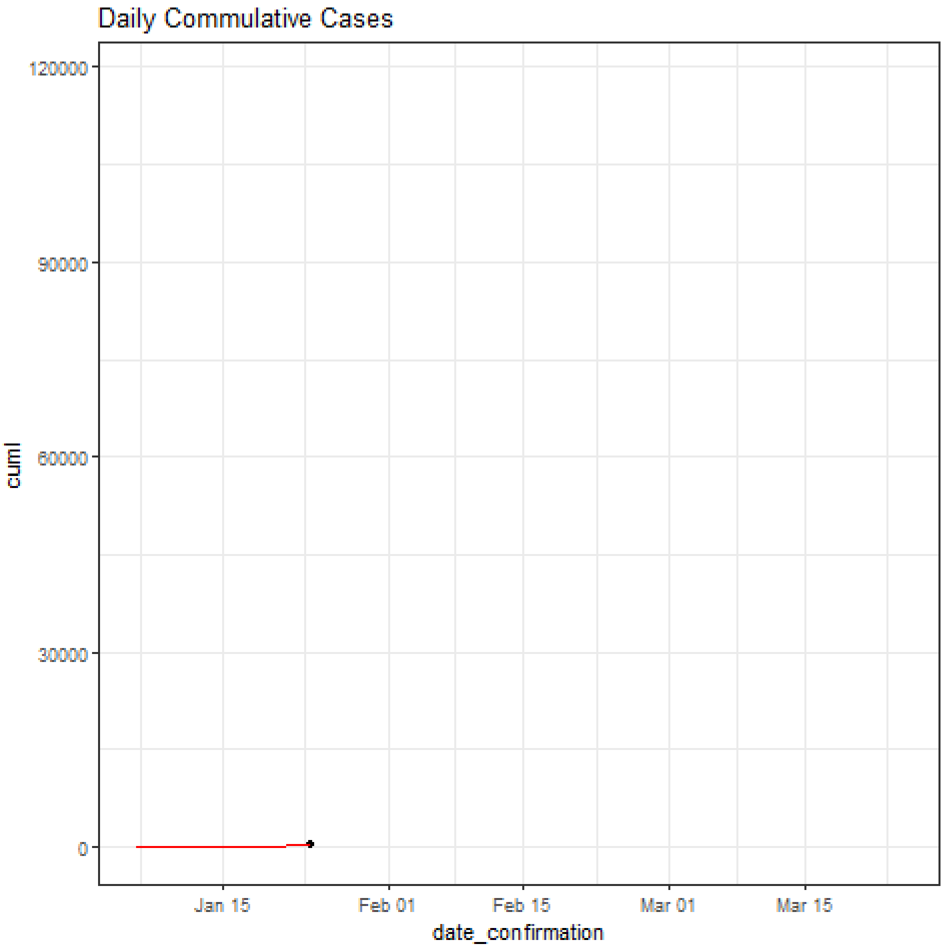
```
import numpy as np
x = cars['Dimensions.Height']
y = cars['Dimensions.Width']
colors = np.random.rand(50)
plt.scatter(x, y,alpha=0.5)
plt.show()
```

R programming language and R studio comes with a number of datasets and one of them is the volcano dataset which contains topographic data of the 50 volcano fields in Auckland volcanic field. We can plot 3D plots to understand the geography of the volcan and the dataset we have. The 3D plot clearly shows the geography of the volcano, the surrounding environment and also the volcano it self.



Lab Assignment 2

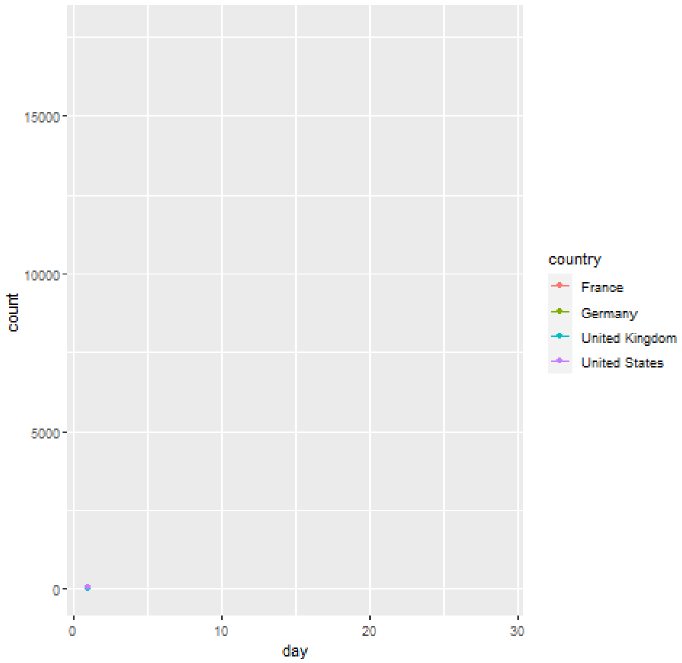
The second assignment is plotting animated time series using the covid dataset. This dataset has a number of information like travel history, source, symptoms, admission to hospital etc... We will plot a graph to see how to covid case spiked using the date of confirmation confirmation data, we will format the confirmation date so that it can be easily interpreted on the graph. This is done using the R programming language inside R studio. The graph shows that from january 15 the cases slowly start to increase and after march 15 the cases are at their highest possible number.



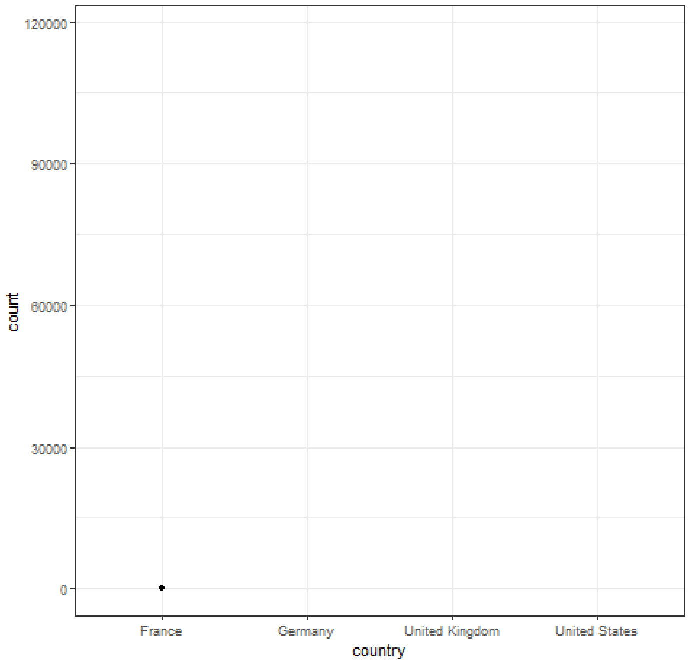
Lab Assignment 3

The third assignment is plotting animated line plots and bar charts on the latest COVID dataset. The line plot is plotted for the United States, France, United Kingdom and Germany on a daily basis. This shows that the covid cases increase day by day almost exponentially from February to April. The bar plots are plotted for the same countries on a monthly basis and clearly shows that the United States reported the highest number of COVID cases compared to the other three countries and Germany reported the lowest COVID cases compared to the other three nations.

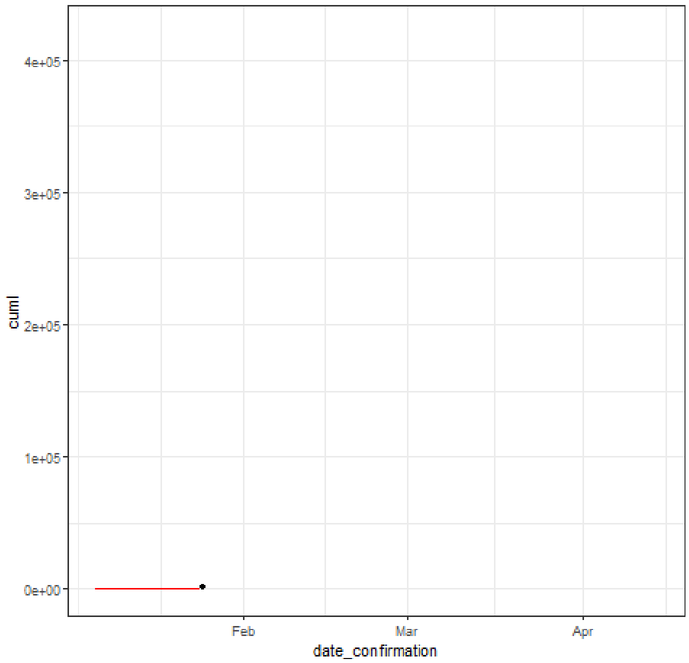
Animated daily plot



Animated bar plot by countries

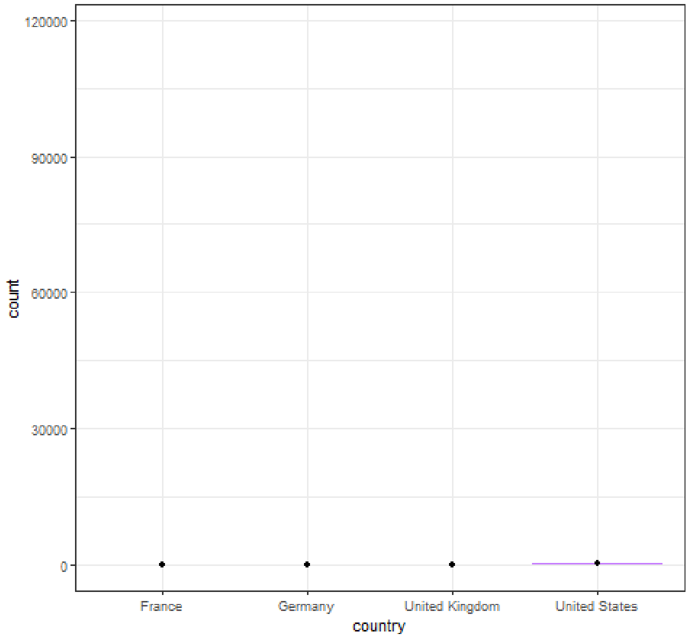


Daily Commulative Cases



Animated Bar Plot by Month

Month: 2



The source code and images of this assignment are linked here