Data Visualization

Data visualization is the practice of translating information into a visual context, such as a map or graph, to make data easier for the human brain to understand and pull insights from. The main goal of data visualization is to make it easier to identify patterns, trends and outliers in large datasets. The term is often used interchangeably with others, including information graphics, information visualization and statistical graphics.

Univariate analysis: provides summary statistics for each field in the raw data set or summary only on one variable. Some examples are histograms, Box plot, Violin plot.

Histograms: Histograms group the data in bins and is the fastest way to get idea about the distribution of each attribute in dataset.

Density plots: A density plot is a representation of the distribution of a numeric variable. It uses a kernel density estimate to show the probability density function of the variable (see more). It is a smoothed version of the histogram and is used in the same concept.

Box plots: A box and whisker plot—also called a box plot—displays the five-number summary of a set of data. The five-number summary is the minimum, first quartile, median, third quartile, and maximum. In a box plot, we draw a box from the first quartile to the third quartile. A vertical line goes through the box at the median.

Python provides certain libraries for performing visualization on the given dataset. Some of the important data visualization libraries are as follows:

·Matplotlib

·Seaborn

·Bokeh

·Altair

·Plotly

·ggplot

These python libraries will be used in performing the univariate and bivariate visualizations.

Steps to visualization includes the following:

Let’s understand this with the help of with a fake advertising data set, indicating whether or not a particular internet user clicked on an Advertisement on a company website. We will try to create a model that will predict whether or not they will click on an ad based off the features of that user.

This data set contains the following features:

- 'Daily Time Spent on Site': consumer time on site in minutes
- 'Age': cutomer age in years
- 'Area Income': Avg. Income of geographical area of consumer
- 'Daily Internet Usage': Avg. minutes a day consumer is on the internet
- 'Ad Topic Line': Headline of the advertisement

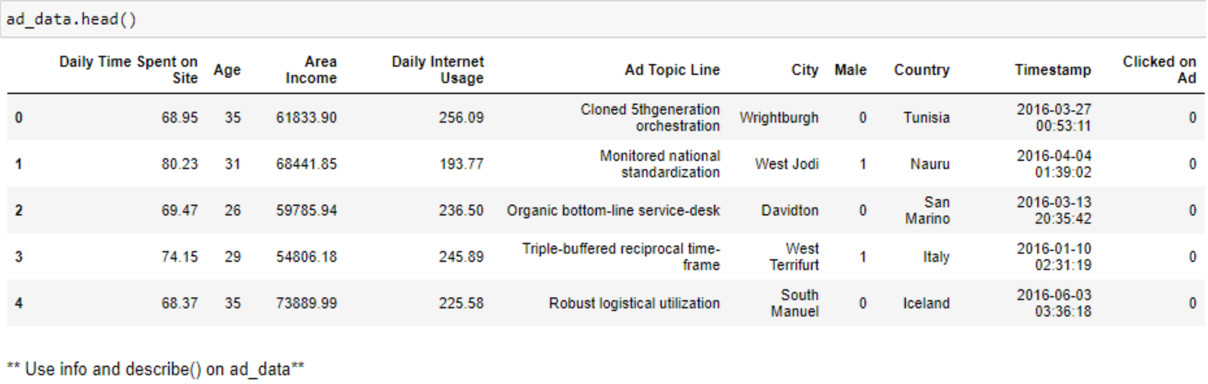
- 'City': City of consumer
- 'Male': Whether or not consumer was male
- 'Country': Country of consumer
- 'Timestamp': Time at which consumer clicked on Ad or closed window
- 'Clicked on Ad': 0 or 1 indicated clicking on Ad

Import Libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

Get the Data

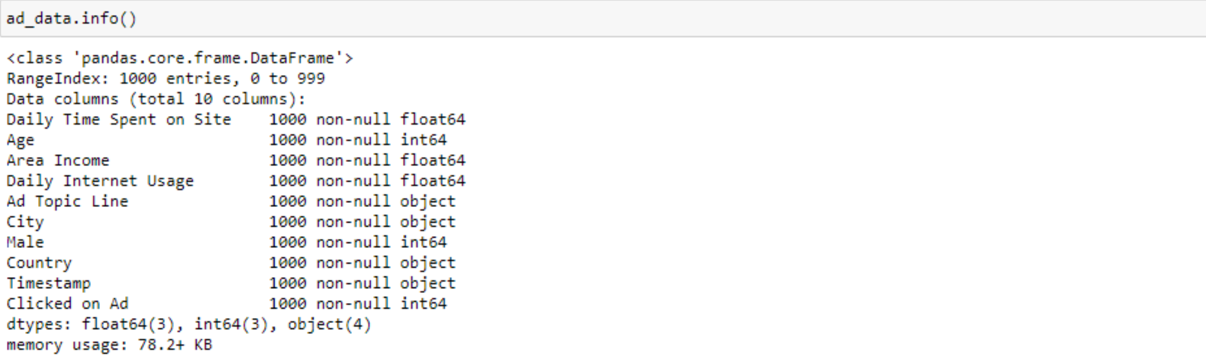
```
ad_data.head()
```



```
ad_data.head()
```

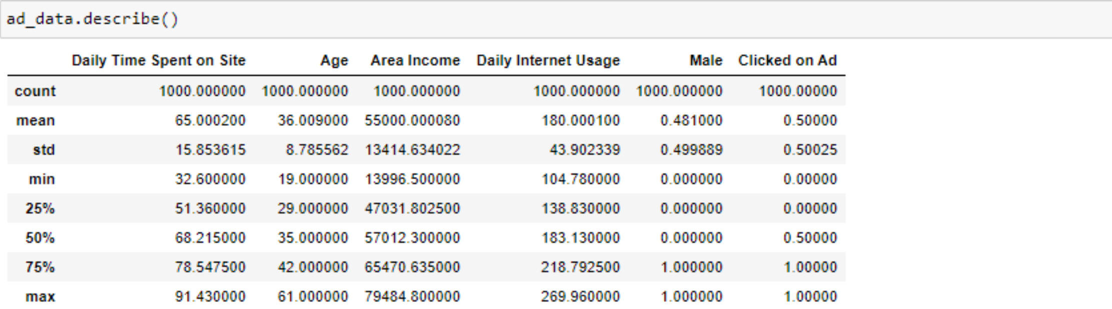
	Daily Time Spent on Site	Age	Area Income	Daily Internet Usage	Ad Topic Line	City	Male	Country	Timestamp	Clicked on Ad
0	68.95	35	61833.90	256.09	Cloned 5thgeneration orchestration	Wrightburgh	0	Tunisia	2016-03-27 00:53:11	0
1	80.23	31	68441.85	193.77	Monitored national standardization	West Jodi	1	Nauru	2016-04-04 01:39:02	0
2	69.47	26	59785.94	236.50	Organic bottom-line service-desk	Davidton	0	San Marino	2016-03-13 20:35:42	0
3	74.15	29	54806.18	245.89	Triple-buffered reciprocal time-frame	West Terrifurt	1	Italy	2016-01-10 02:31:19	0
4	68.37	35	73889.99	225.58	Robust logistical utilization	South Manuel	0	Iceland	2016-06-03 03:36:18	0

** Use info and describe() on ad_data**



```
ad_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 10 columns):
Daily Time Spent on Site    1000 non-null float64
Age                         1000 non-null int64
Area Income                 1000 non-null float64
Daily Internet Usage        1000 non-null float64
Ad Topic Line               1000 non-null object
City                        1000 non-null object
Male                        1000 non-null int64
Country                     1000 non-null object
Timestamp                   1000 non-null object
Clicked on Ad               1000 non-null int64
dtypes: float64(3), int64(3), object(4)
memory usage: 78.2+ KB
```



```
ad_data.describe()
```

	Daily Time Spent on Site	Age	Area Income	Daily Internet Usage	Male	Clicked on Ad
count	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000	1000.00000
mean	65.000200	36.009000	55000.000080	180.000100	0.481000	0.50000
std	15.853615	8.785562	13414.634022	43.902339	0.499889	0.50025
min	32.600000	19.000000	13996.500000	104.780000	0.000000	0.00000
25%	51.360000	29.000000	47031.802500	138.830000	0.000000	0.00000
50%	68.215000	35.000000	57012.300000	183.130000	0.000000	0.50000
75%	78.547500	42.000000	65470.635000	218.792500	1.000000	1.00000
max	91.430000	61.000000	79484.800000	269.960000	1.000000	1.00000

Exploratory Data Analysis

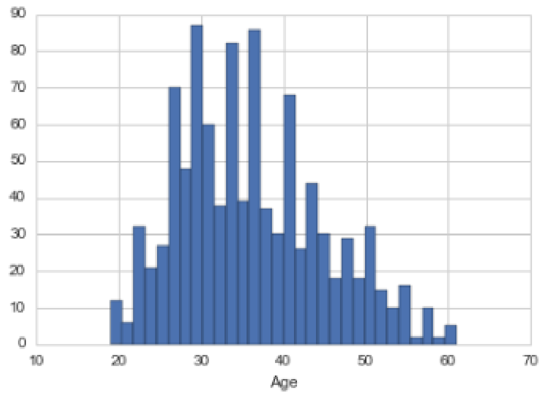
Let's use seaborn to explore the data!

Try recreating the plots shown below!

Create a histogram of the Age

```
sns.set_style('whitegrid')
ad_data['Age'].hist(bins=30)
plt.xlabel('Age')
```

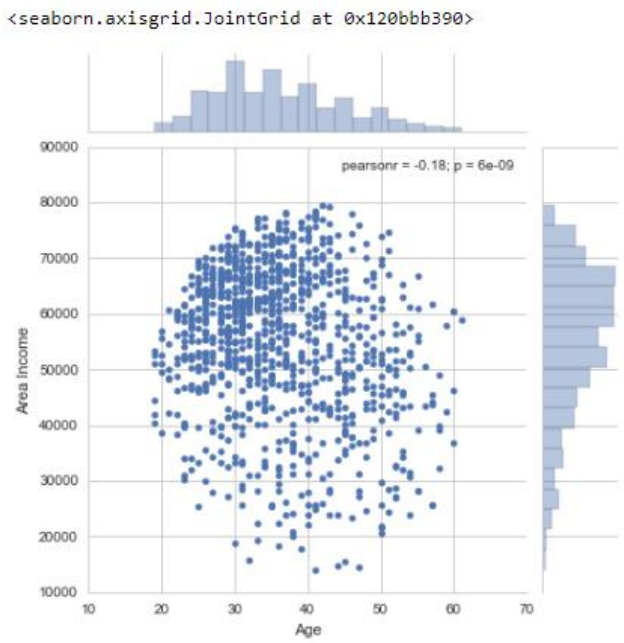
```
<matplotlib.text.Text at 0x11a05b908>
```



Create a jointplot showing Area Income versus Age.

```
sns.jointplot(x='Age',y='Area Income',data=ad_data)
```

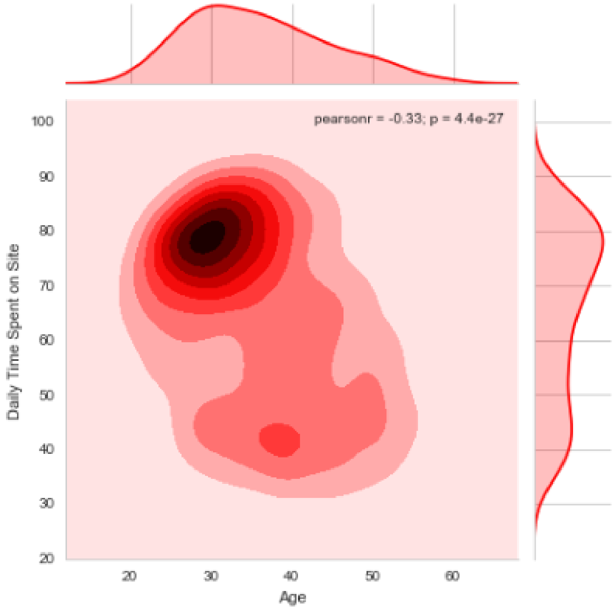
```
<seaborn.axisgrid.JointGrid at 0x120bbb390>
```



```
<seaborn.axisgrid.JointGrid at 0x120bbb390>
```

Create a jointplot showing the kde distributions of Daily Time spent on site vs. Age.

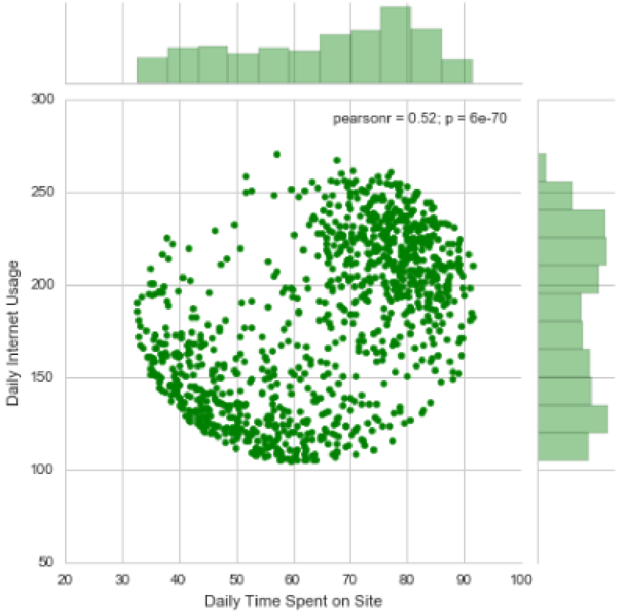
```
sns.jointplot(x='Age',y='Daily Time Spent on Site',data=ad_data,color='red',kind='kde');
```



Create a jointplot of 'Daily Time Spent on Site' vs. 'Daily Internet Usage

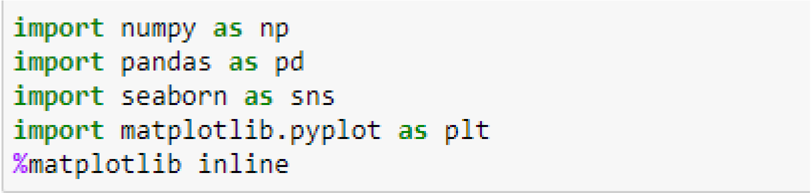
```
sns.jointplot(x='Daily Time Spent on Site',y='Daily Internet Usage',data=ad_data,color='green')
```

```
<seaborn.axisgrid.JointGrid at 0x121e8cb00>
```



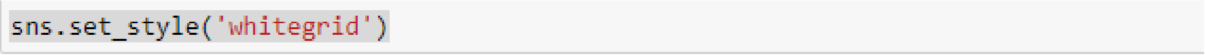
Now let’s implement data visualization on the ‘Titanic’ dataset taken from Kaggle.

1.**Importing the Libraries**

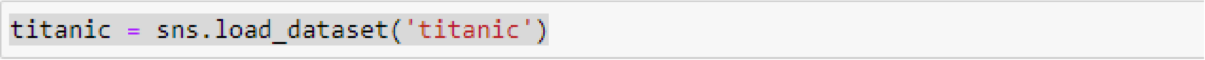


```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

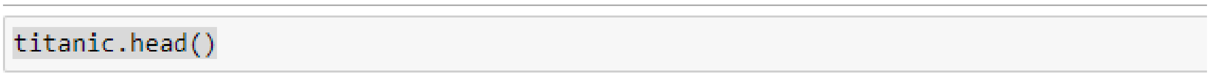
Further performing the univariate visualization we will be considering a sample dataset named ‘titanic’.



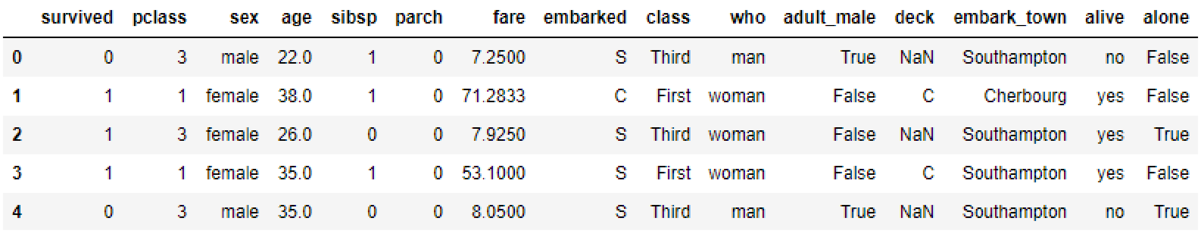
```
sns.set_style('whitegrid')
```



```
titanic = sns.load_dataset('titanic')
```

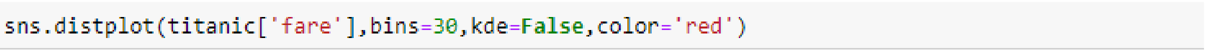


```
titanic.head()
```

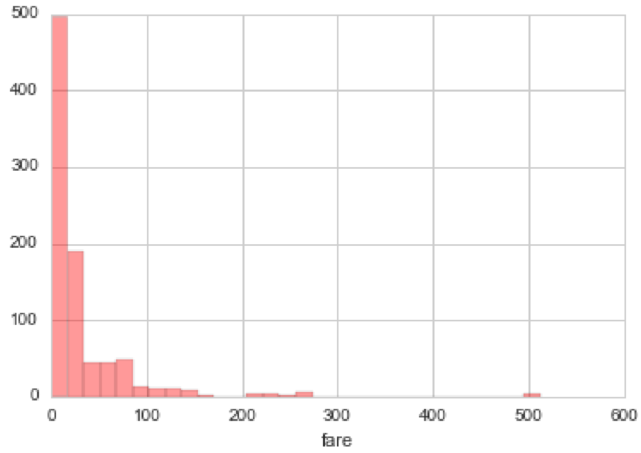


	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	C	Southampton	yes	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southampton	no	True

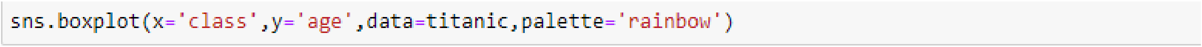
Now further creating a **Distribution plot** for the fare column of the titanic dataset using seaborn.



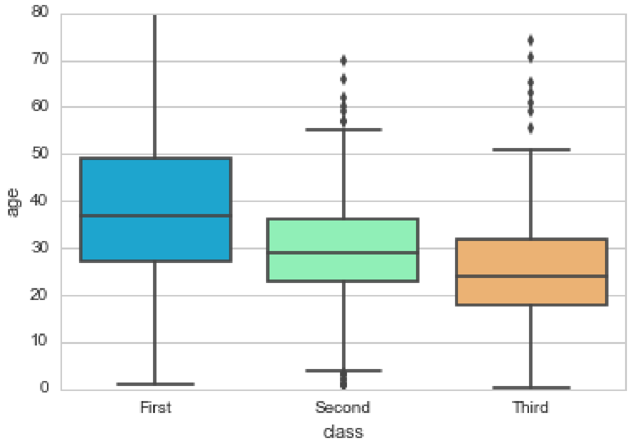
```
sns.distplot(titanic['fare'],bins=30,kde=False,color='red')
```



Now further creating a **Boxplot plot** for the fare column of the titanic dataset using seaborn.

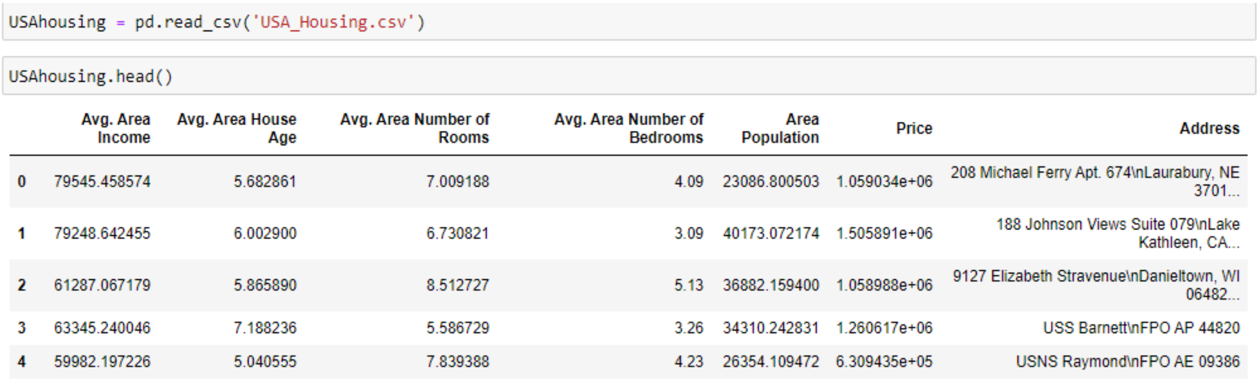


```
sns.boxplot(x='class',y='age',data=titanic,palette='rainbow')
```



Multivariate Visualization: is performed to understand interactions between different fields in the dataset (or) finding interactions between variables more than 2. Example Pair plot and 3D scatter plot.

Now we will be performing our multivariate visualizations on a new dataset named ‘USA_Housing.csv’



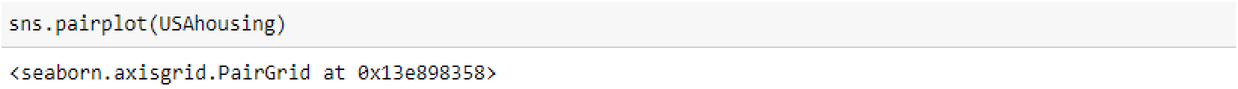
```
USAhousing = pd.read_csv('USA_Housing.csv')
```

```
USAhousing.head()
```

	Avg. Area Income	Avg. Area House Age	Avg. Area Number of Rooms	Avg. Area Number of Bedrooms	Area Population	Price	Address
0	79545.458574	5.682861	7.009188	4.09	23086.800503	1.059034e+06	208 Michael Ferry Apt. 674\nLaurabury, NE 3701...
1	79248.642455	6.002900	6.730821	3.09	40173.072174	1.505891e+06	188 Johnson Views Suite 079\nLake Kathleen, CA...
2	61287.067179	5.865890	8.512727	5.13	36882.159400	1.058988e+06	9127 Elizabeth Stravenue\nDanieltown, WI 06482...
3	63345.240046	7.188236	5.586729	3.26	34310.242831	1.260617e+06	USS Barnett\nFPO AP 44820
4	59982.197226	5.040555	7.839388	4.23	26354.109472	6.309435e+05	USNS Raymond\nFPO AE 09386

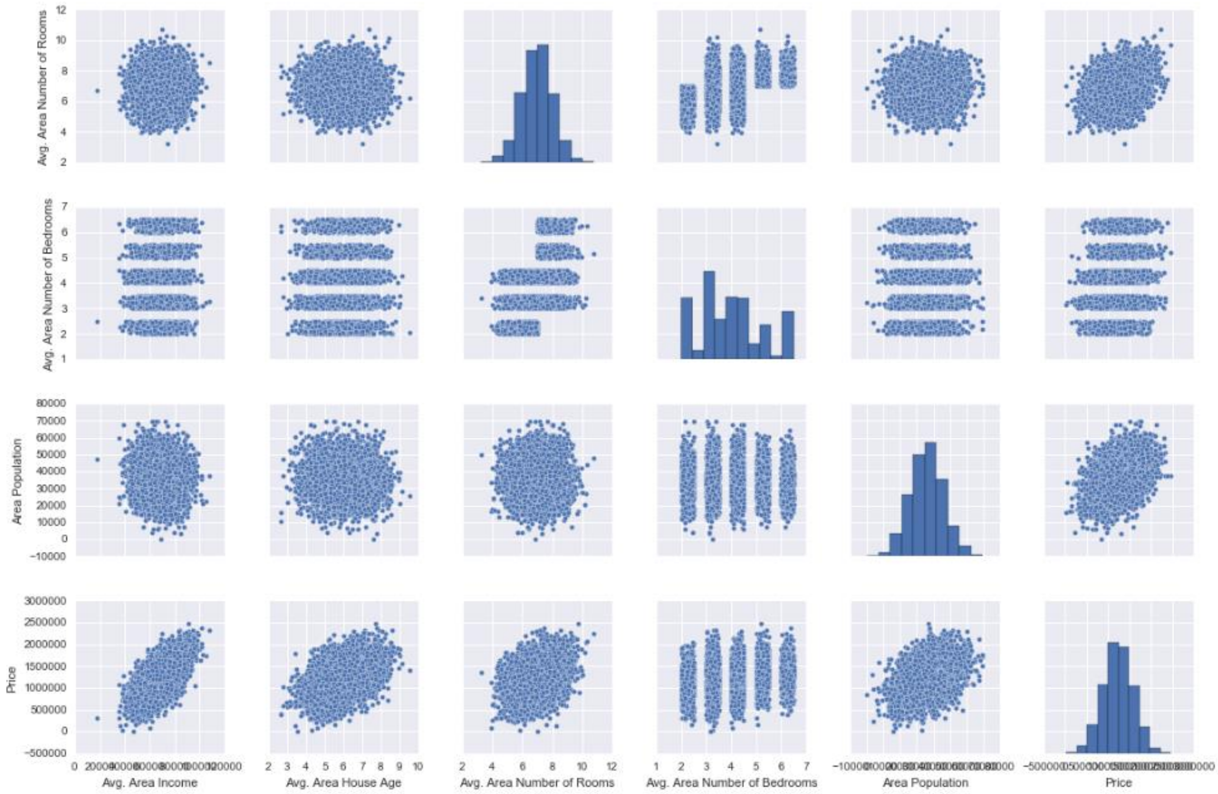
Now we will be performing some of the multivariate visualizations on the dataset “USA_Housing.csv” such as Pairplot , Heatmap and distplot, etc. using seaborn library of Python.

The below graph gives the **pair plot** of the mentioned dataset.

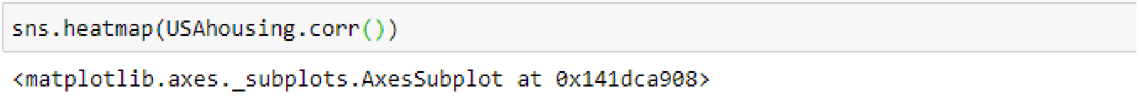


```
sns.pairplot(USAhousing)
```

```
<seaborn.axisgrid.PairGrid at 0x13e898358>
```

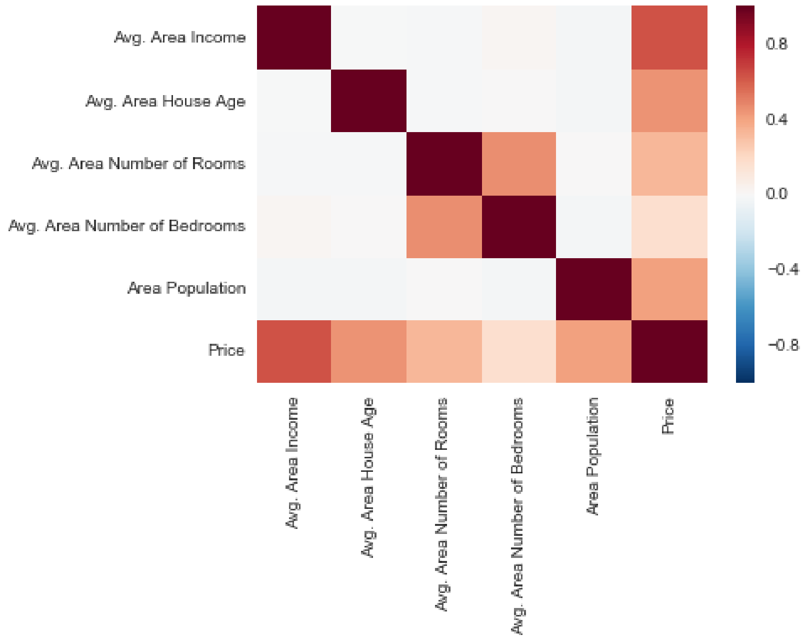


Now, we will be forming a **heatmap** signifying the correlation for the given dataset using seaborn library.

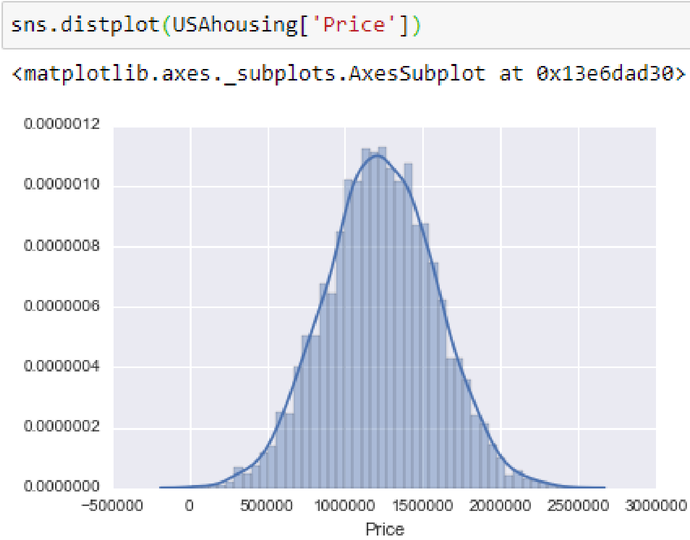


```
sns.heatmap(USAhousing.corr())
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x141dca908>
```



Next ,the **distplot** for the USA_Housing.csv dataset is as follows:



```
sns.distplot(USAhousing['Price'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x13e6dad30>
```

Hence, with the help of multivariate visualization, we can understand interaction between multiple attributes of our dataset.