Hands-on Activity 4.1: Advanced Data Analytics and Machine Learning

Name: Muyo, Mark Danielle L.
Course and Section: CPE 019 - CPE32S9
Date Performed: February 21, 2024
Date Submitted: February 21, 2024
Instructor: Engr. Roman Richard

PART 1: Do the following objectives:
Part 1: Import the Libraries and Data
Part 2: Plot the Data
Part 3: Perform Simple Linear Regression on the SURVIVAL feature column
(you can check the internet on how you can perform simple linear regression)

Part 1: Import the Libraries and Data

In [1]:
```
import pandas as pd

testFile = "/content/titanic_test.csv"
testFrame = pd.read_csv(testFile)

trainFile = "/content/titanic_train.csv"
trainFrame = pd.read_csv(trainFile)
```

In [2]:
```
testFrame.head()
```

Out[2]:

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	Q
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	S
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	Q
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	S
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	S

In [3]:
```
trainFrame.head()
```

Out[3]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Er
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	

In [4]:
```
testFrame.describe()
```

Out[4]:

	PassengerId	Pclass	Age	SibSp	Parch	Fare
count	418.000000	418.000000	332.000000	418.000000	418.000000	417.000000
mean	1100.500000	2.265550	30.272590	0.447368	0.392344	35.627188
std	120.810458	0.841838	14.181209	0.896760	0.981429	55.907576
min	892.000000	1.000000	0.170000	0.000000	0.000000	0.000000
25%	996.250000	1.000000	21.000000	0.000000	0.000000	7.895800
50%	1100.500000	3.000000	27.000000	0.000000	0.000000	14.454200
75%	1204.750000	3.000000	39.000000	1.000000	0.000000	31.500000
max	1309.000000	3.000000	76.000000	8.000000	9.000000	512.329200

In [5]:
```
trainFrame.describe()
```

Out[5]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

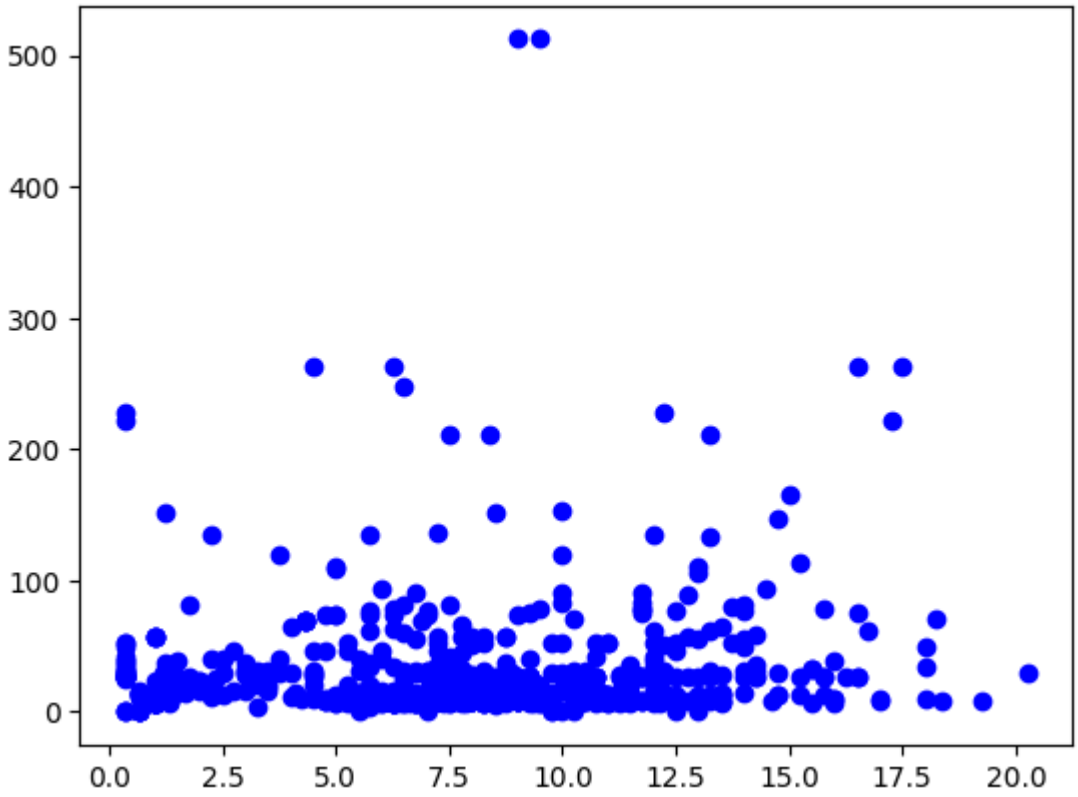
Part 2: Plot the Data

In [6]:
```
import numpy as np
import matplotlib.pyplot as plt
```

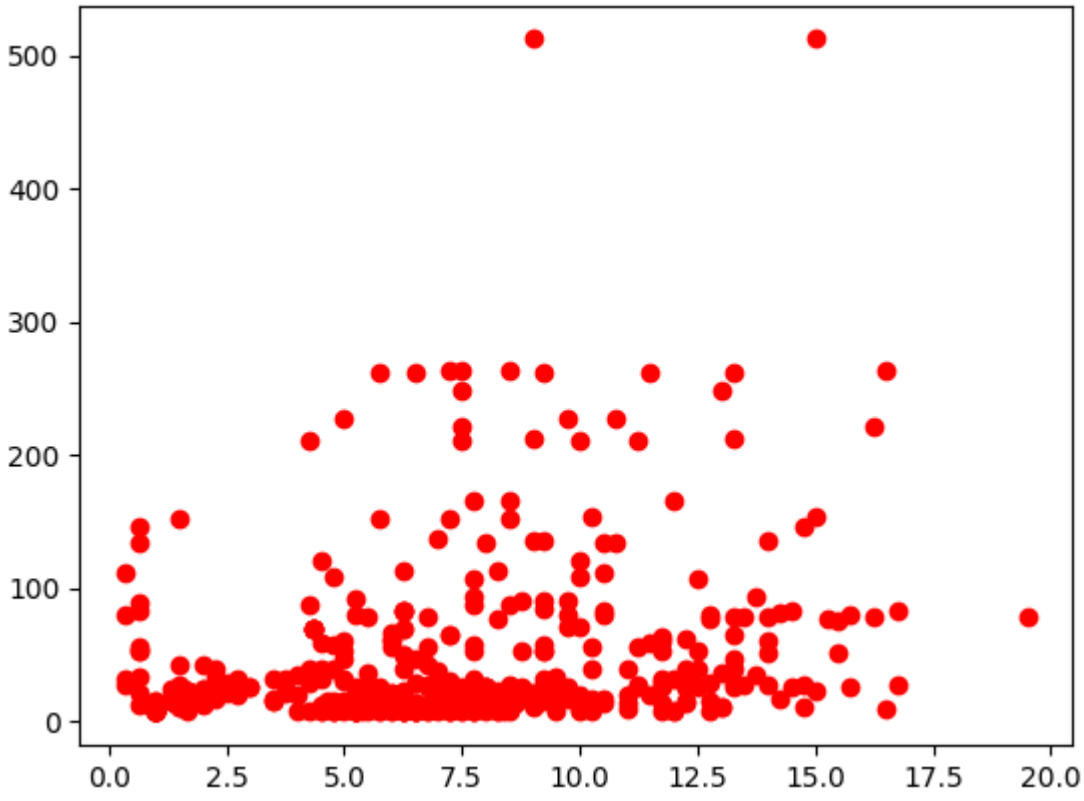
In [7]:
```
male_1 = trainFrame[(trainFrame.Sex == 'male')]
male_2 = testFrame[(testFrame.Sex == 'male')]

female_1 = trainFrame[(trainFrame.Sex == 'female')]
female_2 = testFrame[(testFrame.Sex == 'female')]
```

In [43]:
```
male = pd.concat([male_1, male_2])
male_mean = male [["Pclass", "Age", "SibSp", "Parch"]].mean(axis=1)
plt.scatter(male_mean, male["Fare"], color='blue')
plt.show()
%matplotlib inline
```



In [45]:
```
female = pd.concat([female_1, female_2])
female_mean = female [["Pclass", "Age", "SibSp", "Parch"]].mean(axis=1)
plt.scatter(female_mean, female["Fare"], color='red')
plt.show()
%matplotlib inline
```



In [14]:
```
print(male_mean.isnull().sum())
print(female_mean.isnull().sum())
```

```
0
0
```

In [15]:
```
male_mean.fillna(male_mean.mean(), inplace=True)
female_mean.fillna(female_mean.mean(), inplace=True)
```

In [16]:
```
male.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 843 entries, 0 to 417
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  843 non-null    int64  
 1   Survived     577 non-null    float64
 2   Pclass       843 non-null    int64  
 3   Name         843 non-null    object 
 4   Sex          843 non-null    object 
 5   Age          658 non-null    float64
 6   SibSp        843 non-null    int64  
 7   Parch        843 non-null    int64  
 8   Ticket       843 non-null    object 
 9   Fare         842 non-null    float64
 10  Cabin        154 non-null    object 
 11  Embarked     843 non-null    object 
dtypes: float64(3), int64(4), object(5)
memory usage: 85.6+ KB
```

In [17]:
```
female.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 466 entries, 1 to 414
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  466 non-null    int64  
 1   Survived     314 non-null    float64
 2   Pclass       466 non-null    int64  
 3   Name         466 non-null    object 
 4   Sex          466 non-null    object 
 5   Age          388 non-null    float64
 6   SibSp        466 non-null    int64  
 7   Parch        466 non-null    int64  
 8   Ticket       466 non-null    object 
 9   Fare         466 non-null    float64
 10  Cabin        141 non-null    object 
 11  Embarked     464 non-null    object 
dtypes: float64(3), int64(4), object(5)
memory usage: 47.3+ KB
```

Part 3: Perform Simple Linear Regression on the SURVIVAL feature column.

In [21]:
```
from sklearn import linear_model

male_LRM = linear_model.LinearRegression()
male_LRM.fit(male_mean.values.reshape(-1,1), male["Fare"])
```

```
female_LRM = linear_model.LinearRegression()
female_LRM.fit(female_mean.values.reshape(-1,1), female["Fare"])
```

Out[21]:

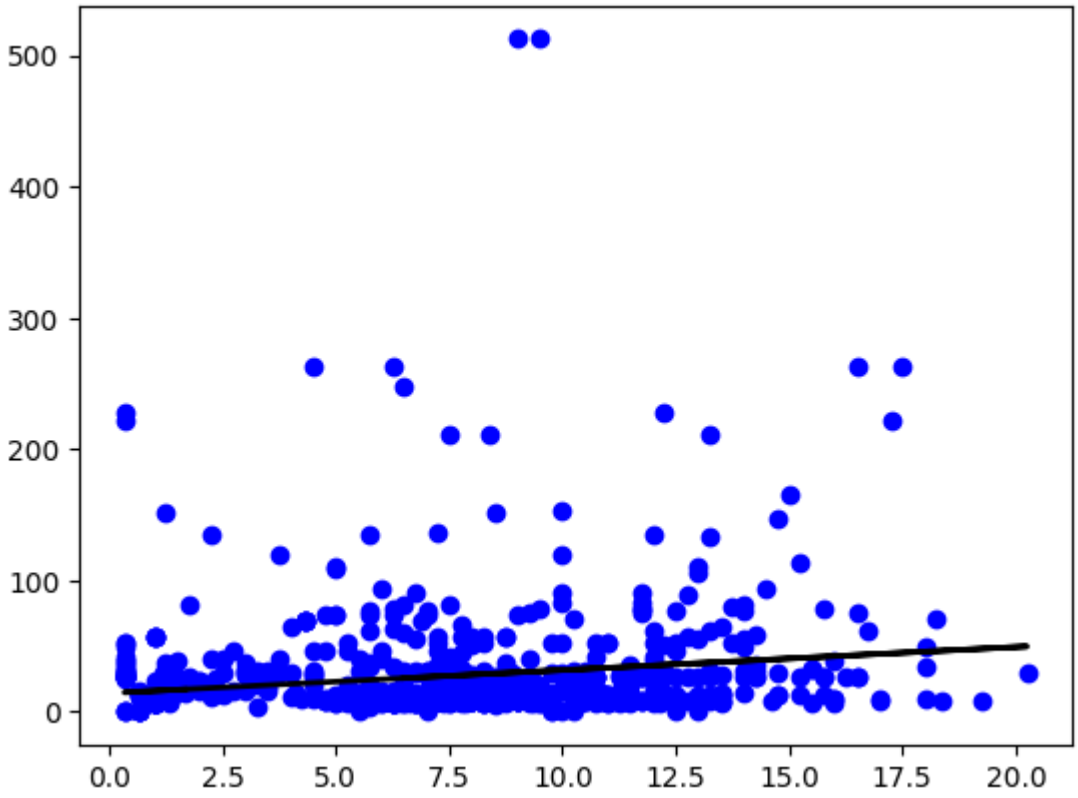
```
LinearRegression()
```

In [22]:

```
male_predictions = male_LRM.predict(male_mean.values.reshape(-1, 1))
female_predictions = female_LRM.predict(female_mean.values.reshape(-1, 1))
```

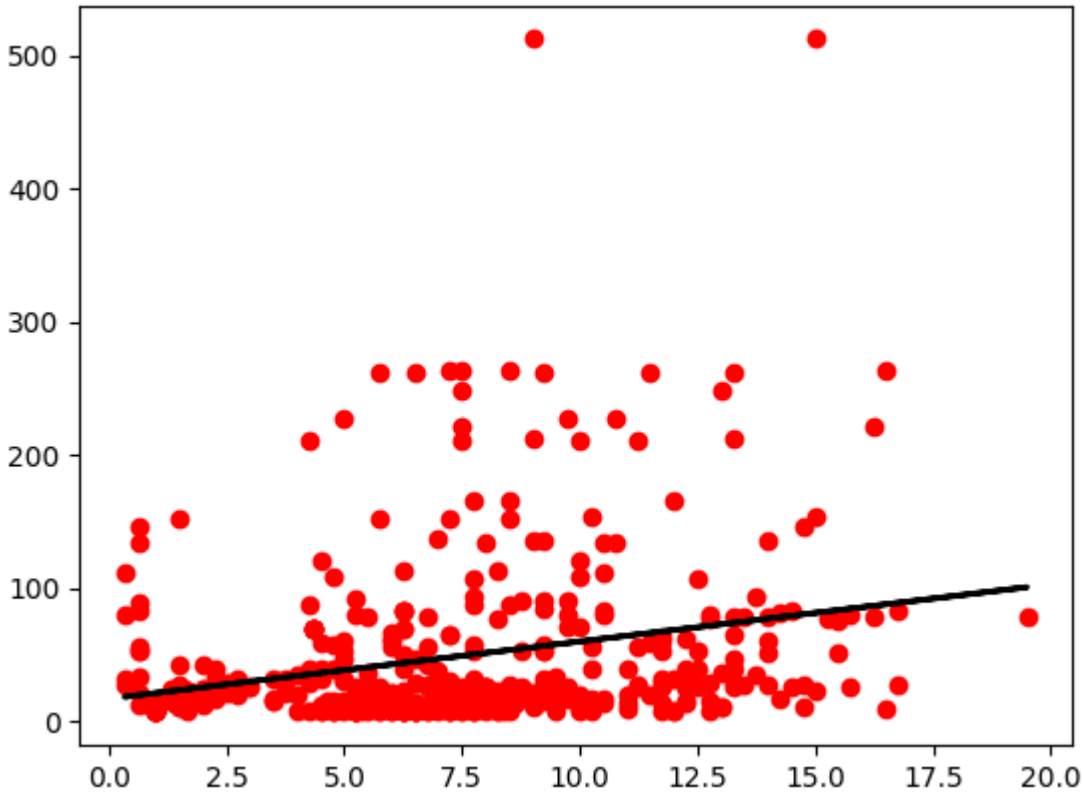
In [46]:

```
plt.scatter(male_mean, male["Fare"], color='blue')
plt.plot(male_mean, male_predictions, color='black', linewidth=2)
plt.show()
%matplotlib inline
```



In [47]:

```
plt.scatter(female_mean, female["Fare"], color='red')
plt.plot(female_mean, female_predictions, color='black', linewidth=2)
plt.show()
%matplotlib inline
```



Conclusion

- In this activity, we imported several Python libraries including Pandas, Matplotlib, and Numpy to prepare the data and build models. The Titanic dataset was imported from a CSV file containing information on passengers aboard the Titanic. Key variables included passenger class, age, gender, fare paid, and whether the passenger survived. Initially, the data was explored through visualizations. A scatterplot was created with the mean of Passenger Class, Age, Siblings/Spouses Aboard, and Parent/Children aboard on the x-axis and fare paid on the y-axis. For the linear regression, I also used the mean of several features and gathered the relationship on how many people had survived, and with that inputs, I used it for survival prediction in the Titanic.