kaviyadevi 20106064

In [2]:
```
#to import libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [3]:
```
#to import dataset
data=pd.read_csv(r"C:\Users\user\Downloads\7_uber - 7_uber.csv")
data
```

Out[3]:

key	fare_amount	pickup_datetime	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_lati
2015- 05-07 52:06	7.5	2015-05-07 19:52:06 UTC	-73.999817	40.738354	-73.999512	40.72
2009- 07-17 04:56	7.7	2009-07-17 20:04:56 UTC	-73.994355	40.728225	-73.994710	40.75
2009- 08-24 45:00	12.9	2009-08-24 21:45:00 UTC	-74.005043	40.740770	-73.962565	40.77
2009- 06-26 22:21	5.3	2009-06-26 08:22:21 UTC	-73.976124	40.790844	-73.965316	40.80
2014- 08-28 47:00	16.0	2014-08-28 17:47:00 UTC	-73.925023	40.744085	-73.973082	40.76
...	...	...	...	...	...	
2012- 10-28 49:00	3.0	2012-10-28 10:49:00 UTC	-73.987042	40.739367	-73.986525	40.74
2014- 03-14 09:00	7.5	2014-03-14 01:09:00 UTC	-73.984722	40.736837	-74.006672	40.73
2009- 06-29 42:00	30.9	2009-06-29 00:42:00 UTC	-73.986017	40.756487	-73.858957	40.69
2015- 05-20 56:25	14.5	2015-05-20 14:56:25 UTC	-73.997124	40.725452	-73.983215	40.69
2010- 05-15 08:00	14.1	2010-05-15 04:08:00 UTC	-73.984395	40.720077	-73.985508	40.76

ns

In [4]:
```
df=data.head(100)
df
```

Out[4]:

	Unnamed: 0	key	fare_amount	pickup_datetime	pickup_longitude	pickup_latitude	dropoff_l
0	24238194	2015-05-07 19:52:06	7.5	2015-05-07 19:52:06 UTC	-73.999817	40.738354	-7
1	27835199	2009-07-17 20:04:56	7.7	2009-07-17 20:04:56 UTC	-73.994355	40.728225	-7
2	44984355	2009-08-24 21:45:00	12.9	2009-08-24 21:45:00 UTC	-74.005043	40.740770	-7
3	25894730	2009-06-26 08:22:21	5.3	2009-06-26 08:22:21 UTC	-73.976124	40.790844	-7
4	17610152	2014-08-28 17:47:00	16.0	2014-08-28 17:47:00 UTC	-73.925023	40.744085	-7
...	...	...	...	...	...	...	
95	25431833	2015-04-11 08:47:47	9.5	2015-04-11 08:47:47 UTC	-73.978432	40.752399	-7
96	44792012	2011-10-03 20:29:00	4.5	2011-10-03 20:29:00 UTC	-73.990055	40.756413	-7
97	18571020	2010-04-26 03:12:44	3.3	2010-04-26 03:12:44 UTC	-73.982326	40.731314	-7
98	37942404	2011-11-18 09:51:00	30.9	2011-11-18 09:51:00 UTC	-73.995888	40.759078	-7
99	29024472	2009-08-30 14:03:55	26.9	2009-08-30 14:03:55 UTC	-73.990137	40.756007	-7

100 rows × 9 columns

DATA CLEANING AND PREPROCESSING

In [5]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100 entries, 0 to 99
Data columns (total 9 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Unnamed: 0         100 non-null    int64  
 1   key                100 non-null    object 
 2   fare_amount        100 non-null    float64
 3   pickup_datetime    100 non-null    object 
 4   pickup_longitude   100 non-null    float64
 5   pickup_latitude    100 non-null    float64
 6   dropoff_longitude  100 non-null    float64
 7   dropoff_latitude   100 non-null    float64
 8   passenger_count    100 non-null    int64  
dtypes: float64(5), int64(2), object(2)
memory usage: 7.2+ KB
```

In [6]:

```
#to display summary of statistics
df.describe()
```

Out[6]:

	Unnamed: 0	fare_amount	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_lati
count	1.000000e+02	100.000000	100.000000	100.000000	100.000000	100.00
mean	2.810554e+07	11.065700	-71.019759	39.123621	-71.015479	39.12
std	1.635033e+07	9.029756	14.569902	8.026358	14.569028	8.02
min	2.268700e+05	2.500000	-74.013173	0.000000	-74.016152	0.00
25%	1.422691e+07	5.475000	-73.992601	40.733982	-73.989142	40.73
50%	2.710896e+07	8.100000	-73.982002	40.752764	-73.979396	40.75
75%	4.480811e+07	12.600000	-73.968615	40.765572	-73.960980	40.77
max	5.508597e+07	56.800000	0.000000	40.850558	0.000000	40.87

In [9]:

```
#to display the column heading
df.columns
```

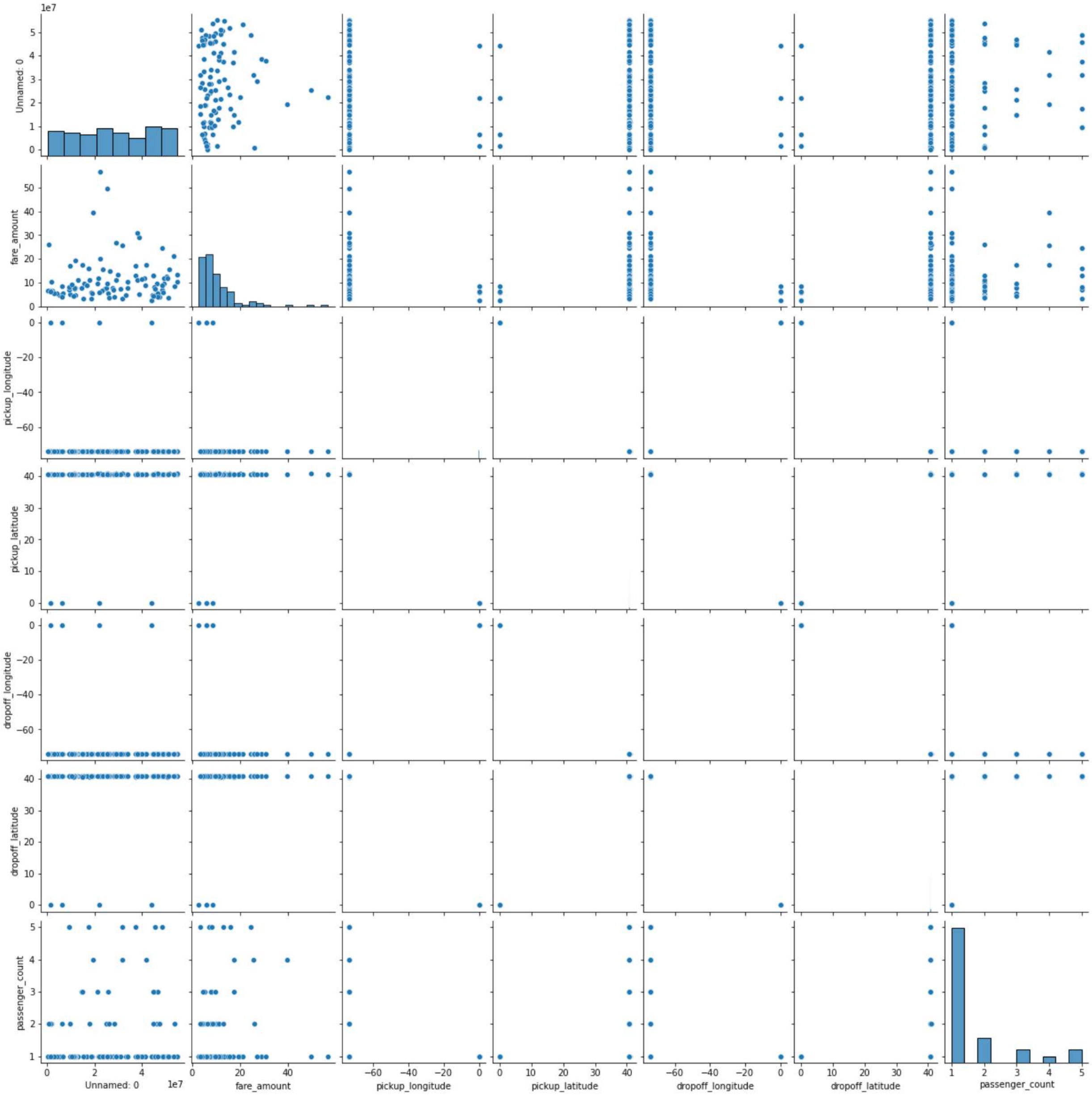
Out[9]:

```
Index(['Unnamed: 0', 'key', 'fare_amount', 'pickup_datetime',
       'pickup_longitude', 'pickup_latitude', 'dropoff_longitude',
       'dropoff_latitude', 'passenger_count'],
      dtype='object')
```

EDA and DATA VISUALIZATION

```
In [8]: sns.pairplot(df)
```

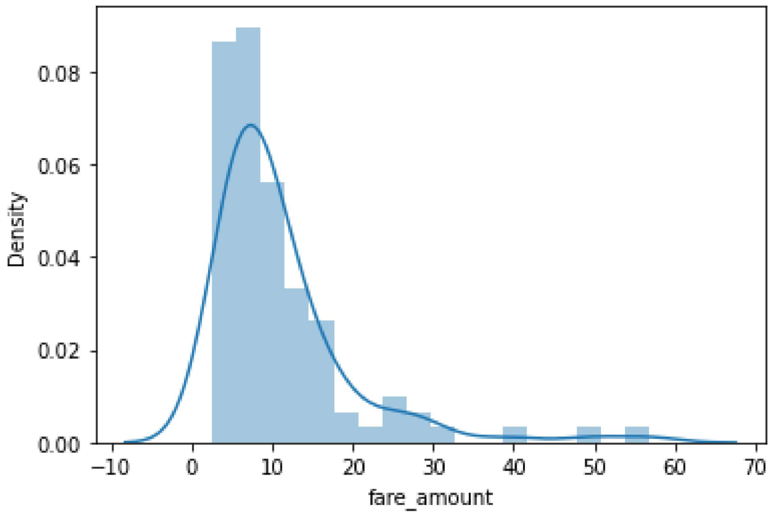
```
Out[8]: <seaborn.axisgrid.PairGrid at 0x18dae831220>
```



In [10]:
```
sns.distplot(df['fare_amount'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2557: Futur
eWarning: `distplot` is a deprecated function and will be removed in a future v
ersion. Please adapt your code to use either `displot` (a figure-level function
with similar flexibility) or `histplot` (an axes-level function for histogram
s).
  warnings.warn(msg, FutureWarning)
```

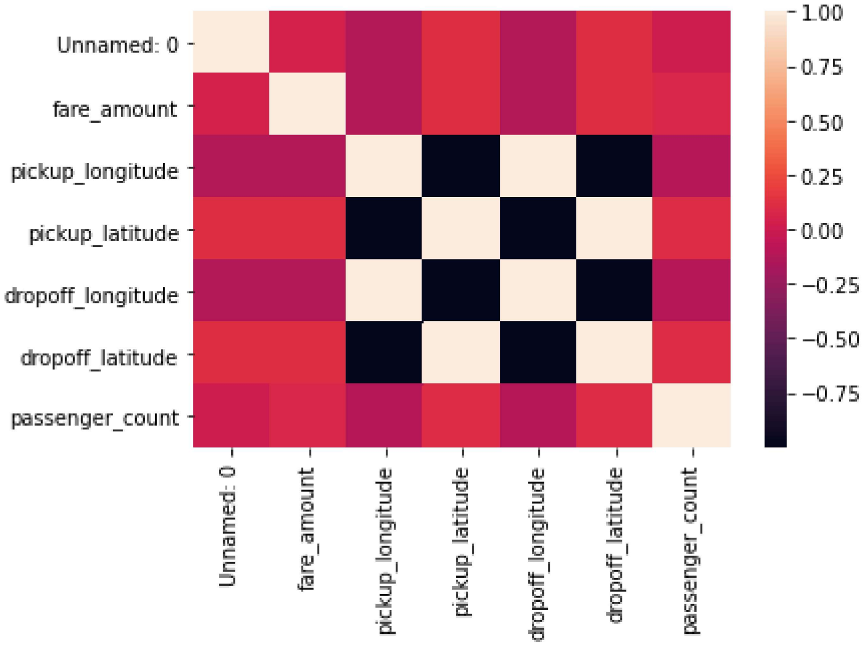
Out[10]:
```
<AxesSubplot:xlabel='fare_amount', ylabel='Density'>
```



In [12]:
```
df1=df[['Unnamed: 0', 'key', 'fare_amount', 'pickup_datetime',
       'pickup_longitude', 'pickup_latitude', 'dropoff_longitude',
       'dropoff_latitude', 'passenger_count']]
```

In [13]:
```
sns.heatmap(df1.corr())
```

Out[13]:
```
<AxesSubplot:>
```



TRAINNING MODEL

In [17]:
```
x=df1[['Unnamed: 0','pickup_longitude', 'pickup_latitude', 'dropoff_longitude',
       'dropoff_latitude', 'passenger_count']]
y=df1[['fare_amount']]
```

In [18]:
```
#to split my dataset into trainning and test

from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
```

In [19]:
```
from sklearn.linear_model import LinearRegression

lr=LinearRegression()
lr.fit(x_train,y_train)
```

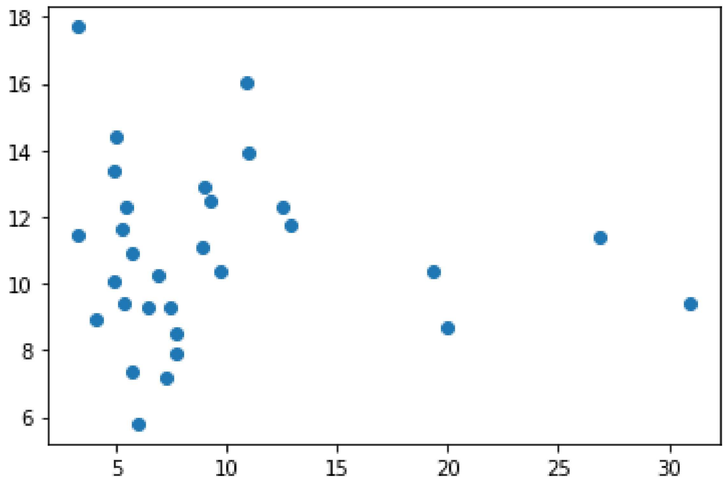
Out[19]:
```
LinearRegression()
```

In [20]:
```
#to find intercept
print(lr.intercept_)
```

```
[4.84853634]
```

In [22]:
```
prediction = lr.predict(x_test)
plt.scatter(y_test,prediction)
```

Out[22]:
```
<matplotlib.collections.PathCollection at 0x18dc1378d00>
```



In [23]:
```
print(lr.score(x_test,y_test))
```

```
-0.25934583718512516
```