

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

```
In [3]: df=pd.read_csv(r"C:\Users\Admin\Downloads\uber - uber.csv")
df
```

Out[3]:

	Unnamed: 0	key	fare_amount	pickup_datetime	pickup_longitude	pickup_latitude	dropoff_latitude
0	24238194	2015-05-07 19:52:06	7.5	2015-05-07 19:52:06 UTC	-73.999817	40.738354	
1	27835199	2009-07-17 20:04:56	7.7	2009-07-17 20:04:56 UTC	-73.994355	40.728225	
2	44984355	2009-08-24 21:45:00	12.9	2009-08-24 21:45:00 UTC	-74.005043	40.740770	
3	25894730	2009-06-26 8:22:21	5.3	2009-06-26 08:22:21 UTC	-73.976124	40.790844	
4	17610152	2014-08-28 17:47:00	16.0	2014-08-28 17:47:00 UTC	-73.925023	40.744085	
...	...	...	...	...	...	...	...
199995	42598914	2012-10-28 10:49:00	3.0	2012-10-28 10:49:00 UTC	-73.987042	40.739367	
199996	16382965	2014-03-14 1:09:00	7.5	2014-03-14 01:09:00 UTC	-73.984722	40.736837	
199997	27804658	2009-06-29 0:42:00	30.9	2009-06-29 00:42:00 UTC	-73.986017	40.756487	
199998	20259894	2015-05-20 14:56:25	14.5	2015-05-20 14:56:25 UTC	-73.997124	40.725452	
199999	11951496	2010-05-15 4:08:00	14.1	2010-05-15 04:08:00 UTC	-73.984395	40.720077	

200000 rows × 9 columns

```
In [4]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200000 entries, 0 to 199999
Data columns (total 9 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Unnamed: 0            200000 non-null int64
1   key                   200000 non-null object
2   fare_amount           200000 non-null float64
3   pickup_datetime       200000 non-null object
4   pickup_longitude      200000 non-null float64
5   pickup_latitude       200000 non-null float64
6   dropoff_longitude     199999 non-null float64
7   dropoff_latitude      199999 non-null float64
8   passenger_count       200000 non-null int64
dtypes: float64(5), int64(2), object(2)
memory usage: 13.7+ MB
```

```
In [5]: df.columns
```

```
Out[5]: Index(['Unnamed: 0', 'key', 'fare_amount', 'pickup_datetime',
              'pickup_longitude', 'pickup_latitude', 'dropoff_longitude',
              'dropoff_latitude', 'passenger_count'],
              dtype='object')
```

```
In [6]: df1=df.head(100)
df1
```

Out[6]:

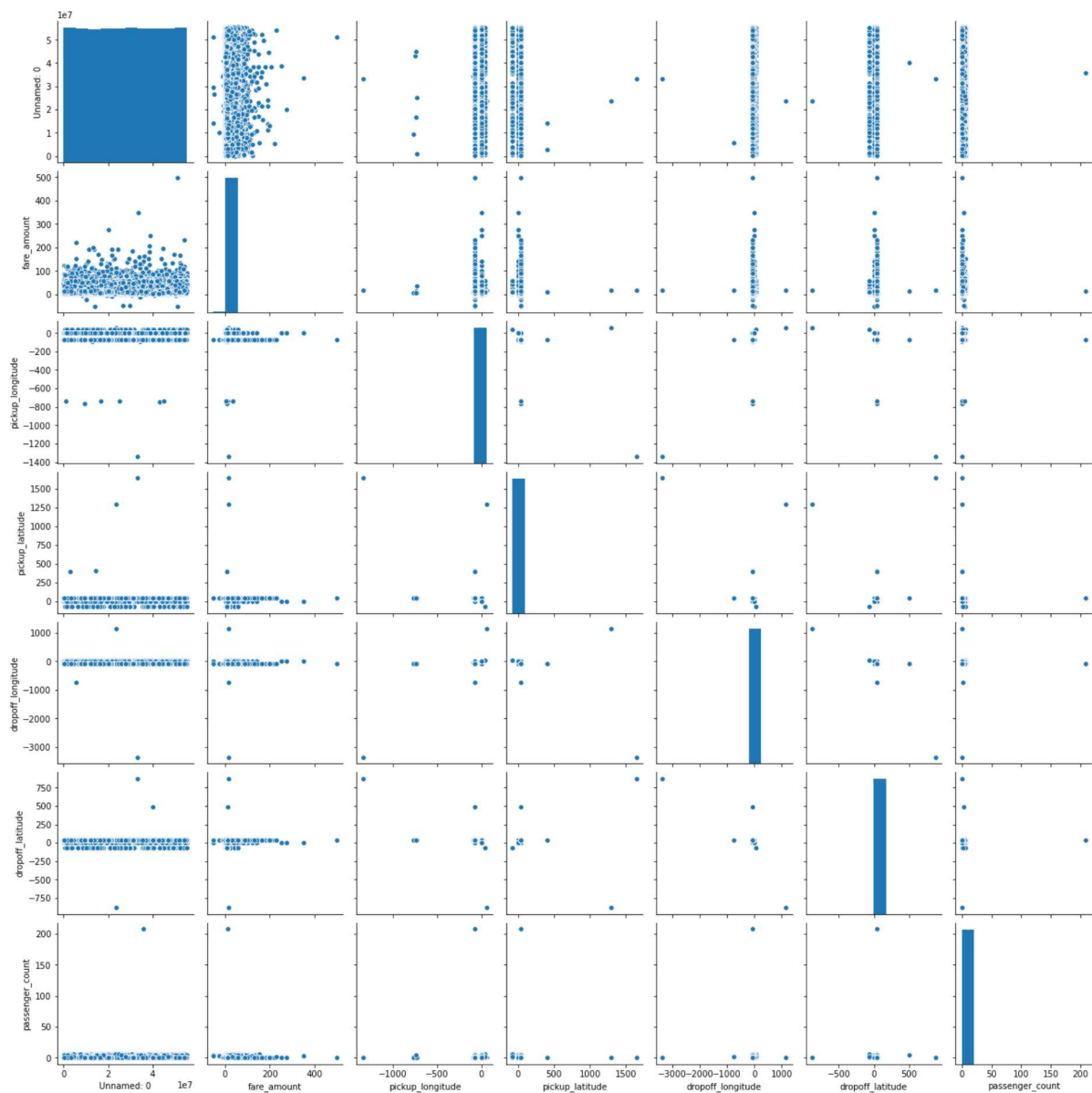
	Unnamed: 0	key	fare_amount	pickup_datetime	pickup_longitude	pickup_latitude	dropoff
0	24238194	2015-05-07 19:52:06	7.5	2015-05-07 19:52:06 UTC	-73.999817	40.738354	
1	27835199	2009-07-17 20:04:56	7.7	2009-07-17 20:04:56 UTC	-73.994355	40.728225	
2	44984355	2009-08-24 21:45:00	12.9	2009-08-24 21:45:00 UTC	-74.005043	40.740770	
3	25894730	2009-06-26 8:22:21	5.3	2009-06-26 08:22:21 UTC	-73.976124	40.790844	
4	17610152	2014-08-28 17:47:00	16.0	2014-08-28 17:47:00 UTC	-73.925023	40.744085	
...	...	...	...	...	...	...	...
95	25431833	2015-04-11 8:47:47	9.5	2015-04-11 08:47:47 UTC	-73.978432	40.752399	
96	44792012	2011-10-03 20:29:00	4.5	2011-10-03 20:29:00 UTC	-73.990055	40.756413	
97	18571020	2010-04-26 3:12:44	3.3	2010-04-26 03:12:44 UTC	-73.982326	40.731314	
98	37942404	2011-11-18 9:51:00	30.9	2011-11-18 09:51:00 UTC	-73.995888	40.759078	
99	29024472	2009-08-30 14:03:55	26.9	2009-08-30 14:03:55 UTC	-73.990137	40.756007	

100 rows × 9 columns



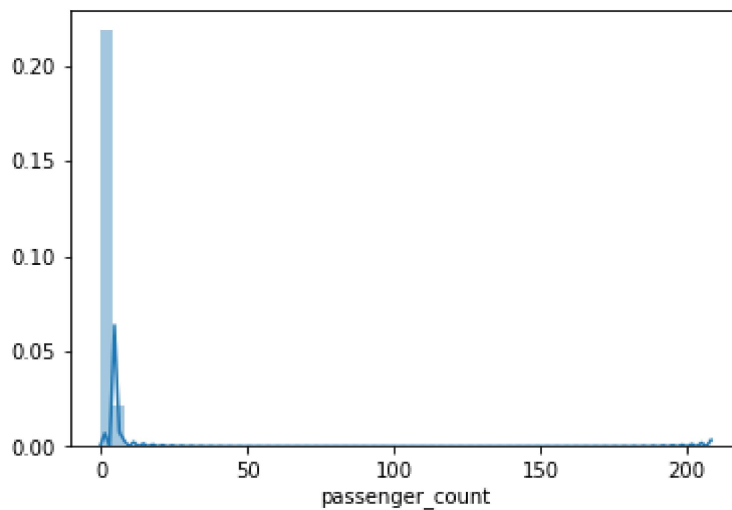
```
In [7]: sns.pairplot(df)
```

```
Out[7]: <seaborn.axisgrid.PairGrid at 0x1a8d1abe2b0>
```



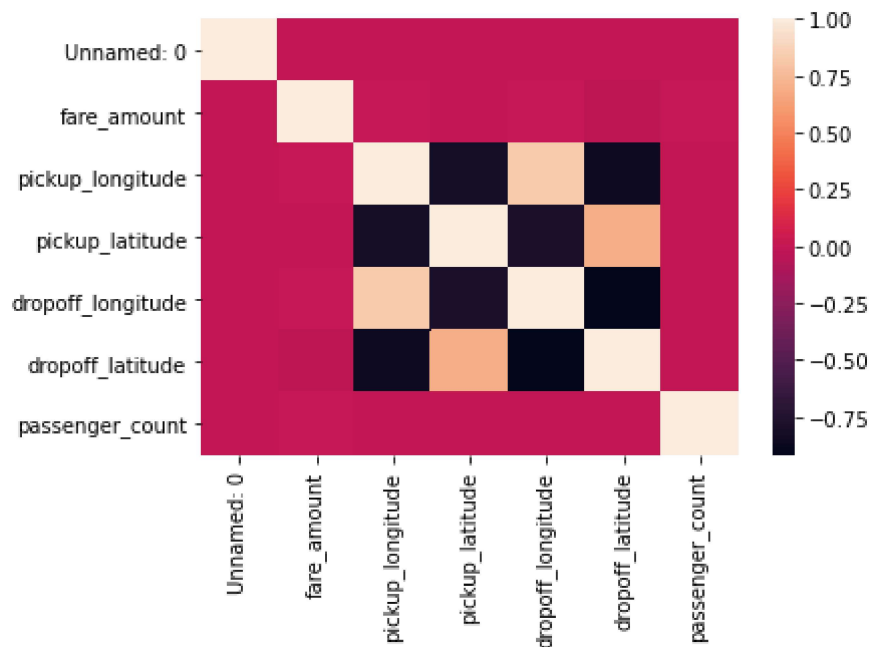
```
In [8]: sns.distplot(df['passenger_count'])
```

```
Out[8]: <matplotlib.axes._subplots.AxesSubplot at 0x1a8d4d6b730>
```



```
In [9]: sns.heatmap(df.corr())
```

```
Out[9]: <matplotlib.axes._subplots.AxesSubplot at 0x1a8d4e37b50>
```



```
In [13]: x=df[['Unnamed: 0', 'fare_amount', 'pickup_longitude', 'pickup_latitude']]
         y=df['passenger_count']
```

```
In [14]: 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 [15]: from sklearn.linear_model import LinearRegression  
lr= LinearRegression()  
lr.fit(x_train,y_train)
```

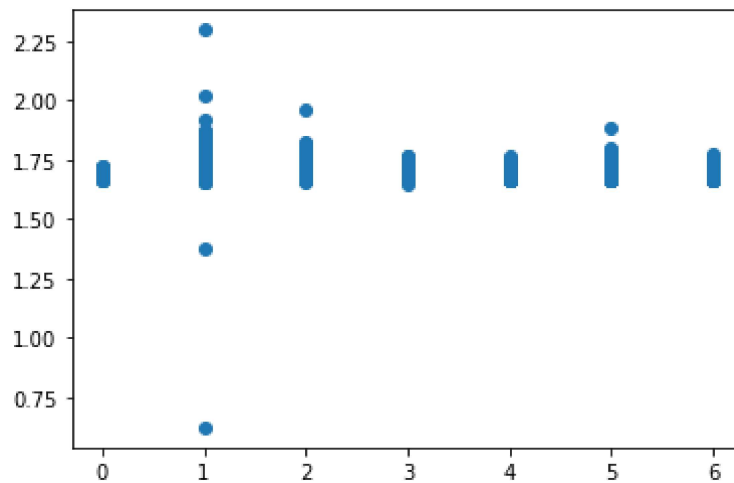
Out[15]: LinearRegression()

```
In [16]: print(lr.intercept_)
```

1.67211818954291

```
In [17]: prediction= lr.predict(x_test)  
plt.scatter(y_test,prediction)
```

Out[17]: <matplotlib.collections.PathCollection at 0x1a8e106e070>



```
In [18]: print(lr.score(x_test,y_test))
```

0.00016652562945440597

```
In [ ]:
```