```
In [3]:  # import libaries
         import numpy as np
         import pandas as pd
         import matplotlib.pyplot as plt
```

```
In [27]: x=pd.read_csv(r"C:\Users\user\Downloads\uber - uber.csv")
```

Out[27]:

	Unnamed: 0	key	fare_amount	pickup_datetime	pickup_longitude	pickup_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 08:22:21	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 01:09:00	7.5	2014-03-14 01:09:00 UTC	-73.984722	40.736837
199997	27804658	2009-06-29 00:42:00	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 04:08:00	14.1	2010-05-15 04:08:00 UTC	-73.984395	40.720077

200000 rows × 9 columns

In [28]:

```
x=x.head(10)
```

Out[28]:

	Unnamed: 0	key	fare_amount	pickup_datetime	pickup_longitude	pickup_latitude	dropo
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 08:22:21	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	
5	44470845	2011-02-12 02:27:09	4.9	2011-02-12 02:27:09 UTC	-73.969019	40.755910	
6	48725865	2014-10-12 07:04:00	24.5	2014-10-12 07:04:00 UTC	-73.961447	40.693965	
7	44195482	2012-12-11 13:52:00	2.5	2012-12-11 13:52:00 UTC	0.000000	0.000000	
8	15822268	2012-02-17 09:32:00	9.7	2012-02-17 09:32:00 UTC	-73.975187	40.745767	
9	50611056	2012-03-29 19:06:00	12.5	2012-03-29 19:06:00 UTC	-74.001065	40.741787	

In [29]:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 9 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Unnamed: 0         10 non-null     int64  
 1   key                10 non-null     object 
 2   fare_amount        10 non-null     float64
 3   pickup_datetime    10 non-null     object 
 4   pickup_longitude   10 non-null     float64
 5   pickup_latitude    10 non-null     float64
 6   dropoff_longitude  10 non-null     float64
 7   dropoff_latitude   10 non-null     float64
 8   passenger_count    10 non-null     int64  
dtypes: float64(5), int64(2), object(2)
memory usage: 848.0+ bytes
```

In [30]:

Out[30]:

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

In [31]:

```
d=x[['Unnamed: 0', 'key', 'fare_amount', 'pickup_datetime',
'pickup_longitude', 'pickup_latitude', 'dropoff_longitude']]
```

Out[31]:

	Unnamed: 0	key	fare_amount	pickup_datetime	pickup_longitude	pickup_latitude	drop
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 08:22:21	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	
5	44470845	2011-02-12 02:27:09	4.9	2011-02-12 02:27:09 UTC	-73.969019	40.755910	
6	48725865	2014-10-12 07:04:00	24.5	2014-10-12 07:04:00 UTC	-73.961447	40.693965	
7	44195482	2012-12-11 13:52:00	2.5	2012-12-11 13:52:00 UTC	0.000000	0.000000	
8	15822268	2012-02-17 09:32:00	9.7	2012-02-17 09:32:00 UTC	-73.975187	40.745767	
9	50611056	2012-03-29 19:06:00	12.5	2012-03-29 19:06:00 UTC	-74.001065	40.741787	

In [32]:

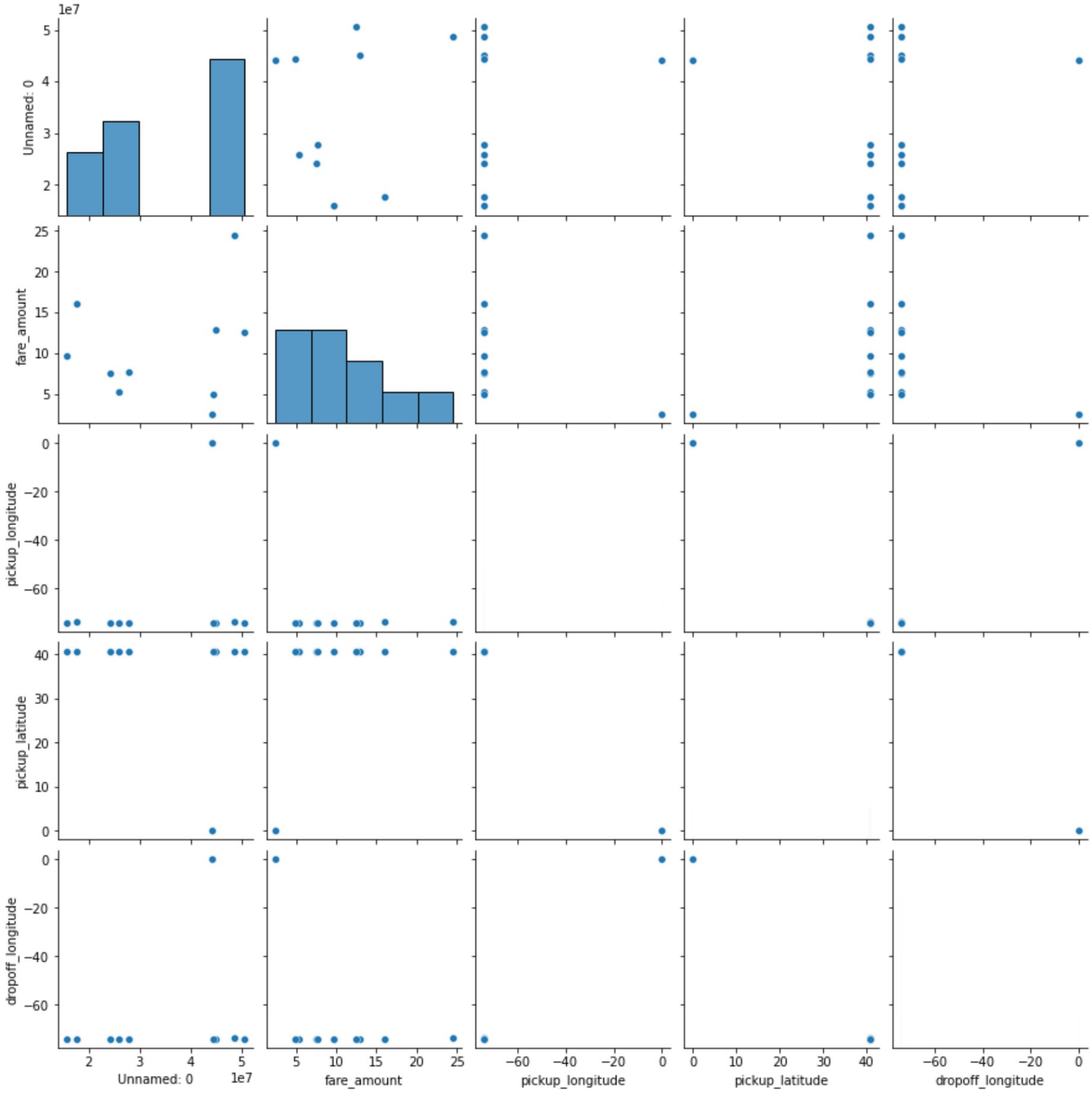
```
d.describe()
```

Out[32]:

	Unnamed: 0	fare_amount	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_l
count	1.000000e+01	10.000000	10.000000	10.000000	10.000000	10.
mean	3.443881e+07	10.350000	-66.580708	36.667971	-66.570116	36.
std	1.342943e+07	6.460693	23.394088	12.883834	23.390384	12.
min	1.582227e+07	2.500000	-74.005043	0.000000	-74.002720	0.
25%	2.465233e+07	5.850000	-73.998451	40.730757	-73.989303	40.
50%	3.601534e+07	8.700000	-73.975656	40.741278	-73.967168	40.
75%	4.485598e+07	12.800000	-73.963340	40.745346	-73.962684	40.
max	5.061106e+07	24.500000	0.000000	40.790844	0.000000	40.

In [33]:
```
sns.pairplot(d)
```

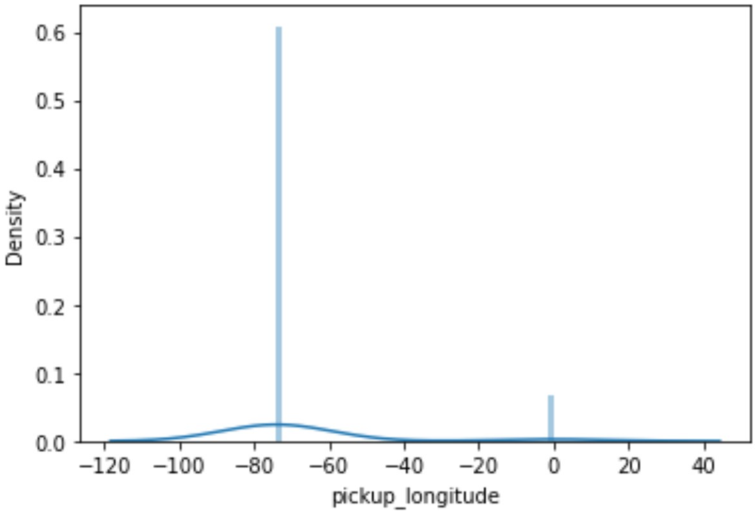
Out[33]:
```
<seaborn.axisgrid.PairGrid at 0x1e645b6ffa0>
```



In [34]:
```
sns.distplot(x['pickup_longitude'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2557: Fut
ureWarning: `distplot` is a deprecated function and will be removed in a futu
re version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for hi
stograms).
  warnings.warn(msg, FutureWarning)
```

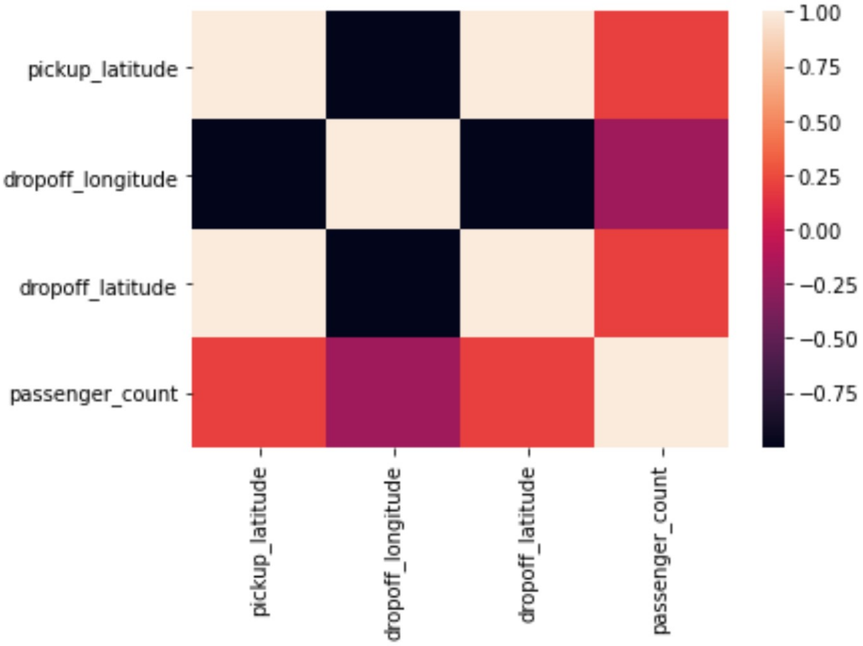
Out[34]:
```
<AxesSubplot:xlabel='pickup_longitude', ylabel='Density'>
```



In [35]:
```
x1=x[['pickup_latitude', 'dropoff_longitude',
'dropoff_latitude', 'passenger_count']]
```

In [36]:
```
sns.heatmap(x1.corr())
```

Out[36]:
```
<AxesSubplot:>
```



In [39]:
```
x=x1[[ 'pickup_latitude','dropoff_longitude',
'dropoff_latitude' ]]
```

In [40]:
```
# to split my dataset into traning and test date

from sklearn.model_selection import train_test_split
```

In [41]:
```
from sklearn.linear_model import LinearRegression

lr=LinearRegression()
```

Out[41]:
```
LinearRegression()
```

In [42]:

```
1.0008520148648716
```

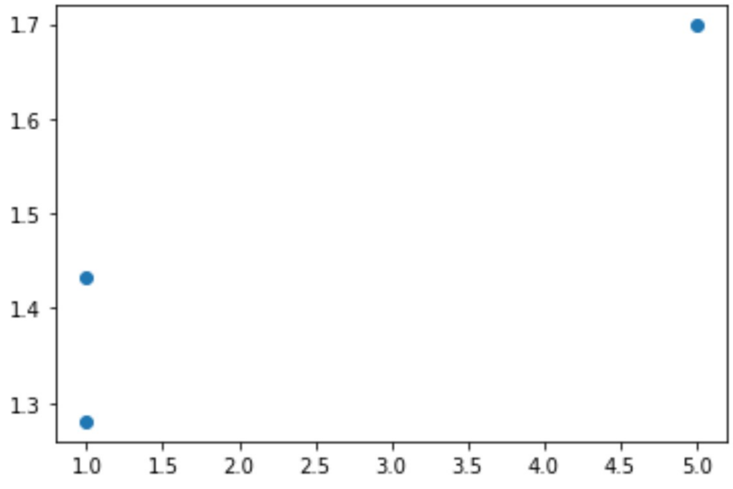
In [43]:
```
coeff=pd.DataFrame(lr.coef_,x.columns,columns=['Co-efficient'])
```

Out[43]:

	Co-efficient
pickup_latitude	-11.774920
dropoff_longitude	11.487929
dropoff_latitude	32.635256

In [44]:
```
prediction=lr.predict(x_test)
```

Out[44]:
```
<matplotlib.collections.PathCollection at 0x1e650c87970>
```



In [45]:

Out[45]:
```
-0.04663327432886266
```

In [46]:

Out[46]:

```
0.6475311824818931
```

In [47]:

In [48]:

```
rr=Ridge(alpha=10)
rr.fit(x_train,y_train)
```

Out[48]:

```
-0.03129906325361609
```

In [49]:

```
la=Lasso(alpha=10)
```

Out[49]:

```
Lasso(alpha=10)
```

In [50]:

Out[50]:

```
-0.06377551020408134
```

In [ ]: