ML

df.head()

df.info()

df.columns

df.shape

df.types

df.describe() To get statistics of each columns

df.isnull() df.isna()

df['dropoff\_latitude'].fillna(value=df['dropoff\_latitude'].mean(),inplace = True)

df['pickup\_datetime']=pd.to\_datetime(df['pickup\_datetime'])

df=df.assign(hour=df['pickup\_datetime'].dt.hour,

day=df['pickup\_datetime'].dt.day,

month=df['pickup\_datetime'].dt.month,

year=df['pickup\_datetime'].dt.year,

dayofweek=df['pickup\_datetime'].dt.dayofweek)

Q1=df1[col].quantile(0.25)

df1[col]=np.clip(df1[col],lower\_bound,upper\_bound)

fig,axis = plt.subplots(figsize = (10,6))

sns.heatmap(df.corr(),annot = True)

x = df[['pickup\_longitude','pickup\_latitude','dropoff\_longitude','dropoff\_latitude','passenger\_count','hour','day','month','year','dayofweek','dist\_travel\_km']]

from sklearn.model\_selection import train\_test\_split

X\_train,X\_test,y\_train,y\_test = train\_test\_split(x,y,test\_size = 0.33)

from sklearn.linear\_model import LinearRegression

regression = LinearRegression()

from sklearn.ensemble import RandomForestRegressor

rf = RandomForestRegressor(n\_estimators=100)

rf.fit(X\_train,y\_train)

y\_pred = rf.predict(X\_test)