In [1]:
```
import pandas as pd
import numpy as np
from sklearn.feature_extraction import DictVectorizer
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore")
```

In [2]:
```
df = pd.read_parquet('https://d37ci6vzurychx.cloudfront.net/trip-data/yello
```

In [ ]:
```
```

In [3]:
```
len(df.columns)
```

Out[3]:
```
19
```

In [4]:
```
df['duration'] = df['tpep_dropoff_datetime'] - df['tpep_pickup_datetime']
df['duration'] = df['duration'].dt.total_seconds()/60
```

In [5]:
```
std_dev = np.std(df['duration'], ddof=1)
std_dev
```

Out[5]:
```
42.594351241920904
```

In [6]:
```
len(df['duration'])
```

Out[6]:
```
3066766
```

In [7]:
```
df['duration'].describe()
```

Out[7]:
```
count    3.066766e+06
mean     1.566900e+01
std      4.259435e+01
min     -2.920000e+01
25%      7.116667e+00
50%      1.151667e+01
75%      1.830000e+01
max      1.002918e+04
Name: duration, dtype: float64
```

In [8]:
```
df_cleaned = df[df['duration'] >= 1][df['duration'] <= 60]
```

In [9]:
```
df_cleaned['duration'].describe()
```

Out[9]:
```
count    3.009173e+06
mean     1.420486e+01
std      9.939386e+00
min      1.000000e+00
25%      7.216667e+00
50%      1.155000e+01
75%      1.818333e+01
max      6.000000e+01
Name: duration, dtype: float64
```

In [10]:
```
pct_left = (len(df_cleaned['duration'])*100)/len(df['duration'])
pct_left
```

Out[10]:
```
98.1220282212598
```

In [11]:
```
categorical = ['PULocationID', 'DOLocationID']
```

In [12]:
```
df_categorical = df_cleaned[categorical].astype(str)
```

In [13]:
```
df_categorical.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 3009173 entries, 0 to 3066765
Data columns (total 2 columns):
 #   Column        Dtype 
---  ------        ----- 
 0   PULocationID  object
 1   DOLocationID  object
dtypes: object(2)
memory usage: 68.9+ MB
```

In [14]:
```
dv = DictVectorizer()
train_dict = df_categorical.to_dict(orient='records')
```

In [15]:
```
target = 'duration'
```

In [16]:
```
X_train = dv.fit_transform(train_dict)
y_train = df_cleaned[target].values
```

In [17]:
```
model = LinearRegression()
```

In [18]:
```
model.fit(X_train, y_train)
```

Out[18]:
```
LinearRegression()
```

In [19]:
```
y_pred = model.predict(X_train)
```

In [20]:
```
mse = mean_squared_error(y_train, y_pred, squared=False)
r2 = r2_score(y_train, y_pred)

print(f"Mean Squared Error: {mse:.2f}")
print(f"R² Score: {r2:.2f}")

#sns.distplot(y_pred, label='prediction')
#sns.distplot(y_train, label='actual')
```

```
Mean Squared Error: 7.65
R² Score: 0.41
```

In [21]:
```
df_val = pd.read_parquet('https://d37ci6vzurychx.cloudfront.net/trip-data/y
df_val['duration'] = df_val['tpep_dropoff_datetime'] - df_val['tpep_pickup_
df_val['duration'] = df_val['duration'].dt.total_seconds()/60
```

In [22]:
```
df_val_cleaned = df_val[df_val['duration'] >= 1][df_val['duration'] <= 60]
```

In [23]:
```
df_val_cat = df_val_cleaned[categorical].astype(str)
```

In [24]:
```
df_val_cat.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2855951 entries, 0 to 2913954
Data columns (total 2 columns):
 #   Column        Dtype 
---  ------        ----- 
 0   PULocationID  object
 1   DOLocationID  object
dtypes: object(2)
memory usage: 65.4+ MB
```

In [25]:
```
val_dict = df_val_cat.to_dict(orient='records')
```

In [29]:
```
X_test = dv.transform(val_dict)
y_test = df_val_cleaned[target].values
```

In [30]:
```
y_val_pred = model.predict(X_test)
```

In [31]:
```
mse = mean_squared_error(y_test, y_val_pred, squared=False)
r2 = r2_score(y_test, y_val_pred)

print(f"Mean Squared Error: {mse:.2f}")
print(f"R² Score: {r2:.2f}")
```

```
Mean Squared Error: 7.81
R² Score: 0.40
```