

bicycle

June 6, 2024

```
[2]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
from pandas.tseries.holiday import USFederalHolidayCalendar # Add this import
```

```
[3]: data = pd.
    ↪read_csv('fremont-bridge-hourly-bicycle-counts-by-month-october-2012-to-present_
    ↪2.csv', index_col='Date', parse_dates=True)
data.columns = ["Total", "East", "West"]
data["Total"] = data["West"] + data["East"]
data.head(5)
```

```
[3]:
```

	Total	East	West
Date			
2012-10-03 00:00:00	13.0	4.0	9.0
2012-10-03 01:00:00	10.0	4.0	6.0
2012-10-03 02:00:00	2.0	1.0	1.0
2012-10-03 03:00:00	5.0	2.0	3.0
2012-10-03 04:00:00	7.0	6.0	1.0

```
[4]: daily = data.resample('d').sum()
daily = daily[['Total']]
```

```
[5]: daily['Mon'] = daily.index.dayofweek == 0
daily['Tue'] = daily.index.dayofweek == 1
daily['Wed'] = daily.index.dayofweek == 2
daily['Thu'] = daily.index.dayofweek == 3
daily['Fri'] = daily.index.dayofweek == 4
daily['Sat'] = daily.index.dayofweek == 5
daily['Sun'] = daily.index.dayofweek == 6
cal = USFederalHolidayCalendar()
holidays = cal.holidays('2012', '2019')
daily['holiday'] = daily.index.isin(holidays).astype(int)
```

```
[6]: weather = pd.read_csv('BicycleWeather.csv', index_col='DATE', parse_dates=True)
weather.head(3)
```

```
[6]:
```

	STATION	STATION_NAME	\
DATE			
2012-01-01	GHCND:USW00024233	SEATTLE TACOMA INTERNATIONAL AIRPORT WA US	
2012-01-02	GHCND:USW00024233	SEATTLE TACOMA INTERNATIONAL AIRPORT WA US	
2012-01-03	GHCND:USW00024233	SEATTLE TACOMA INTERNATIONAL AIRPORT WA US	

	PRCP	SNWD	SNOW	TMAX	TMIN	AWND	WDF2	WDF5	...	WT17	WT05	\
DATE									...			
2012-01-01	0	0	0	128	50	47	100	90	...	-9999	-9999	
2012-01-02	109	0	0	106	28	45	180	200	...	-9999	-9999	
2012-01-03	8	0	0	117	72	23	180	170	...	-9999	-9999	

	WT02	WT22	WT04	WT13	WT16	WT08	WT18	WT03
DATE								
2012-01-01	-9999	-9999	-9999	-9999	-9999	-9999	-9999	-9999
2012-01-02	-9999	-9999	-9999	1	1	-9999	-9999	-9999
2012-01-03	-9999	-9999	-9999	-9999	1	-9999	-9999	-9999

[3 rows x 25 columns]

```
[17]: weather['TMIN'] /= 10
weather['TMAX'] /= 10
weather['Temp (C)'] = 0.5 * (weather['TMIN'] + weather['TMAX'])
weather['PRCP'] /= 254
weather['dry day'] = (weather['PRCP'] == 0).astype(int)
daily = daily.join(weather[['PRCP', 'Temp (C)', 'dry day']])
daily.dropna(inplace=True)
daily['annual'] = (daily.index - daily.index[0]).days / 365
```

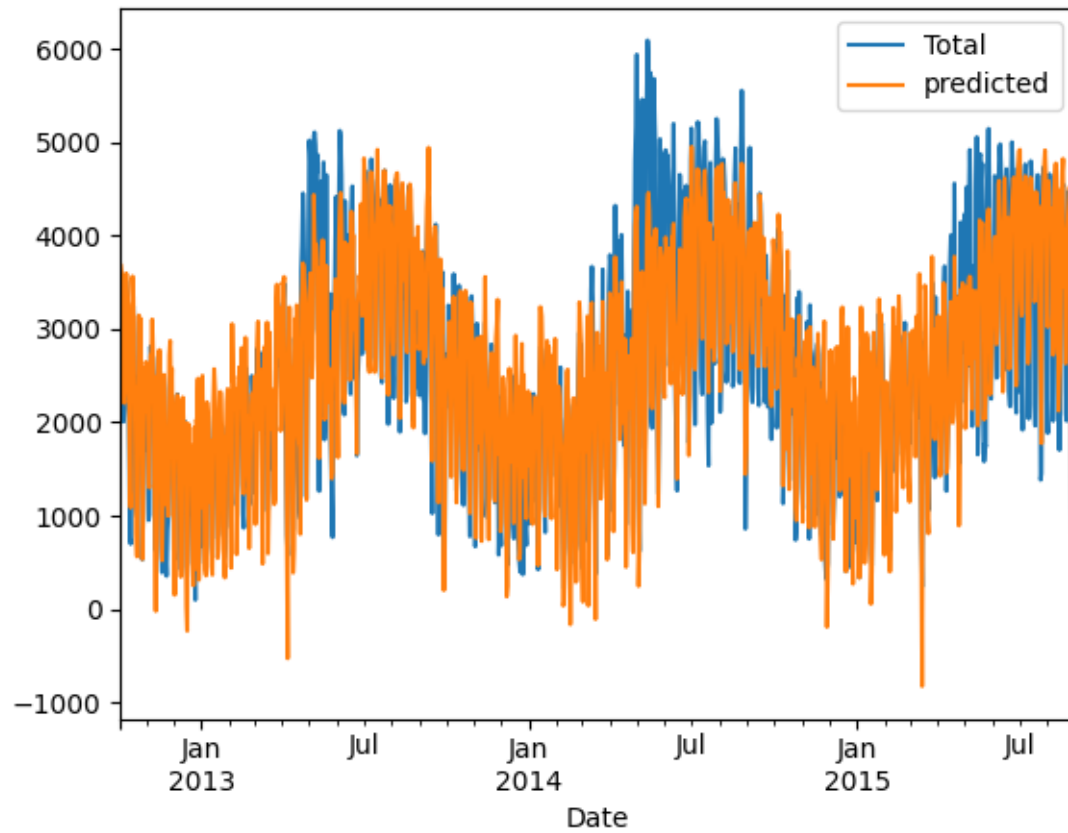
```
[18]: x = daily[['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun', 'holiday', 'PRCP', 'Temp (C)', 'dry day']]
y = daily['Total']
```

```
[19]: model = LinearRegression()
model.fit(x, y)
daily['predicted'] = model.predict(x)
```

```
[20]: r2_score = model.score(x, y)
print("R-squared:", r2_score)

# Plot results
daily[['Total', 'predicted']].plot()
plt.show()
```

R-squared: 0.8398535650641547



[]: