PolyRegression

March 26, 2020

1 Get the dataset and preprocessing

[45]:

```
import pandas as pd
url = "https://www.fraserinstitute.org/api/economic-data?
 ↪start_year=2017&end_year=2017&sub=false"
freedom = pd.read_csv(url, skiprows=4)
freedom = freedom.drop(["Unnamed: 0"], axis=1)
freedom
```

[45]:

```
     Year ISO_Code  ... Freedom to Trade Internationally  Regulation
0    2017      AGO  ...                             3.21        5.66
1    2017      ALB  ...                             8.34        7.77
2    2017      ARE  ...                             8.05        7.24
3    2017      ARG  ...                             6.55        5.65
4    2017      ARM  ...                             8.20        7.54
..    ...      ...  ...                              ...         ...
157  2017      VNM  ...                             6.27        6.73
158  2017      YEM  ...                             5.93        5.04
159  2017      ZAF  ...                             6.87        7.20
160  2017      ZMB  ...                             7.10        6.46
161  2017      ZWE  ...                             3.75        6.73

[162 rows x 11 columns]
```

[46]:

```
freedom = freedom.drop(["ISO_Code", "Year", "Countries"], axis=1)
freedom
```

[46]:

```
     Economic Freedom Summary Index  ...  Regulation
0                              4.83  ...        5.66
1                              7.67  ...        7.77
2                              7.17  ...        7.24
3                              5.67  ...        5.65
4                              7.70  ...        7.54
..                              ...  ...         ...
157                            6.27  ...        6.73
158                            5.84  ...        5.04
159                            6.61  ...        7.20
160                            6.84  ...        6.46
```

```
161                            5.69  ...        6.73

[162 rows x 8 columns]
```

2 Split data

[47]:

```
cols = list(freedom.columns)
cols.remove("Economic Freedom Summary Index")
X_df = freedom[cols]
X = X_df.to_numpy()
X
```

[47]:

```
array([[158.  ,   4.  ,   6.76, ...,   5.57,   3.21,   5.66],
       [ 30.  ,   1.  ,   7.53, ...,   9.65,   8.34,   7.77],
       [ 61.  ,   2.  ,   5.85, ...,   9.06,   8.05,   7.24],
       ...,
       [101.  ,   3.  ,   5.77, ...,   8.17,   6.87,   7.2 ],
       [ 83.  ,   3.  ,   6.61, ...,   8.8 ,   7.1 ,   6.46],
       [145.  ,   4.  ,   5.49, ...,   8.42,   3.75,   6.73]])
```

[48]:

```
y = freedom['Economic Freedom Summary Index']
y = y.to_numpy()
y
```

[48]:

```
array([4.83, 7.67, 7.17, 5.67, 7.7 , 8.07, 7.71, 6.34, 6.17, 7.51, 6.08,
       6.07, 6.18, 7.54, 7.35, 7.25, 6.69, 6.64, 6.78, 6.14, 6.23, 6.45,
       6.62, 6.77, 7.37, 5.23, 8.08, 8.4 , 7.89, 6.42, 5.97, 5.84, 5.  ,
       5.08, 6.68, 7.16, 7.41, 7.68, 7.75, 7.82, 7.89, 6.92, 4.77, 6.28,
       5.05, 7.55, 7.89, 5.72, 7.8 , 6.74, 7.35, 5.88, 8.09, 7.94, 6.58,
       5.86, 7.23, 5.23, 6.59, 7.57, 6.32, 8.91, 7.16, 7.26, 6.49, 7.27,
       7.27, 6.91, 8.13, 5.72, 5.21, 7.74, 7.53, 7.41, 7.17, 7.44, 7.86,
       7.1 , 7.05, 6.92, 7.44, 7.59, 6.41, 6.67, 6.97, 6.56, 4.45, 6.57,
       6.5 , 7.88, 7.86, 7.73, 6.69, 6.66, 6.14, 6.93, 7.02, 5.91, 7.97,
       5.63, 6.84, 7.43, 5.6 , 6.11, 8.07, 6.06, 7.34, 6.53, 5.83, 6.86,
       7.01, 7.72, 7.62, 6.49, 8.5 , 6.76, 5.91, 7.66, 7.49, 7.32, 6.36,
       7.24, 7.52, 7.03, 7.07, 7.68, 6.78, 7.26, 6.52, 4.67, 6.17, 8.71,
       5.63, 7.16, 6.89, 6.65, 7.51, 7.15, 7.56, 6.52, 7.16, 5.05, 5.42,
       6.21, 6.86, 6.05, 5.99, 6.7 , 6.2 , 6.67, 7.85, 6.77, 7.39, 5.96,
       7.05, 8.19, 2.58, 6.27, 5.84, 6.61, 6.84, 5.69])
```

[49]:

```
from sklearn.preprocessing import PolynomialFeatures
# [x_1, x_2] to [1, x_1, x_2, x_1^2, x_1 x_2, x_2^2]
X = PolynomialFeatures(degree=2).fit_transform(X)
X
```

[49]:

```
array([[  1.    , 158.    ,   4.    , ...,  10.3041,  18.1686,  32.0356],
       [  1.    ,  30.    ,   1.    , ...,  69.5556,  64.8018,  60.3729],
       [  1.    ,  61.    ,   2.    , ...,  64.8025,  58.282 ,  52.4176],
       ...,
       [  1.    , 101.    ,   3.    , ...,  47.1969,  49.464 ,  51.84  ],
       [  1.    ,  83.    ,   3.    , ...,  50.41  ,  45.866 ,  41.7316],
       [  1.    , 145.    ,   4.    , ...,  14.0625,  25.2375,  45.2929]])
```

[50]:

```python
from sklearn.linear_model import LinearRegression

model = LinearRegression()
model.fit(X, y)

freedom['predicted'] = model.predict(X)
freedom
```

[50]:

```
     Economic Freedom Summary Index  Rank  ...  Regulation  predicted
0                              4.83   158  ...        5.66   4.829106
1                              7.67    30  ...        7.77   7.667934
2                              7.17    61  ...        7.24   7.173694
3                              5.67   146  ...        5.65   5.677430
4                              7.70    27  ...        7.54   7.694836
..                              ...   ...  ...         ...        ...
157                            6.27   119  ...        6.73   6.282438
158                            5.84   140  ...        5.04   5.840127
159                            6.61   101  ...        7.20   6.615632
160                            6.84    83  ...        6.46   6.839420
161                            5.69   145  ...        6.73   5.690788

[162 rows x 9 columns]
```

[51]:

```python
freedom[["Economic Freedom Summary Index", 'predicted']]
```

[51]:

```
     Economic Freedom Summary Index  predicted
0                              4.83   4.829106
1                              7.67   7.667934
2                              7.17   7.173694
3                              5.67   5.677430
4                              7.70   7.694836
..                              ...        ...
157                            6.27   6.282438
158                            5.84   5.840127
159                            6.61   6.615632
160                            6.84   6.839420
161                            5.69   5.690788

[162 rows x 2 columns]
```

3 Analysis

Podemos ver que una regresión polinomial es demasiado buena para interpolar estos datos, de hecho lo logra hacer con varios dígitos de precisión, a pesar de que la regresión fue hecha sobre un polinomio de grado dos.

Por lo tanto concluimos que para este problema este método fue muy efectivo.