seaLevelPredictor

July 11, 2023

1 Sea Level Predictor

[2]:

```
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats import linregress
```

[3]:

```
df = pd.read_csv("epa-sea-level.csv")
```

1.1 Data Summary

[4]:

```
df
```

[4]:

```
     Year  CSIRO Adjusted Sea Level  Lower Error Bound  Upper Error Bound  \
0    1880                  0.000000          -0.952756           0.952756
1    1881                  0.220472          -0.732283           1.173228
2    1882                 -0.440945          -1.346457           0.464567
3    1883                 -0.232283          -1.129921           0.665354
4    1884                  0.590551          -0.283465           1.464567
..    …                        …                  …                 …
129  2009                  8.586614           8.311024           8.862205
130  2010                  8.901575           8.618110           9.185039
131  2011                  8.964567           8.661417           9.267717
132  2012                  9.326772           8.992126           9.661417
133  2013                  8.980315           8.622047           9.338583

     NOAA Adjusted Sea Level
0                        NaN
1                        NaN
2                        NaN
3                        NaN
4                        NaN
..                       …
129                 8.046354
130                 8.122973
131                 8.053065
132                 8.457058
133                 8.546648
```

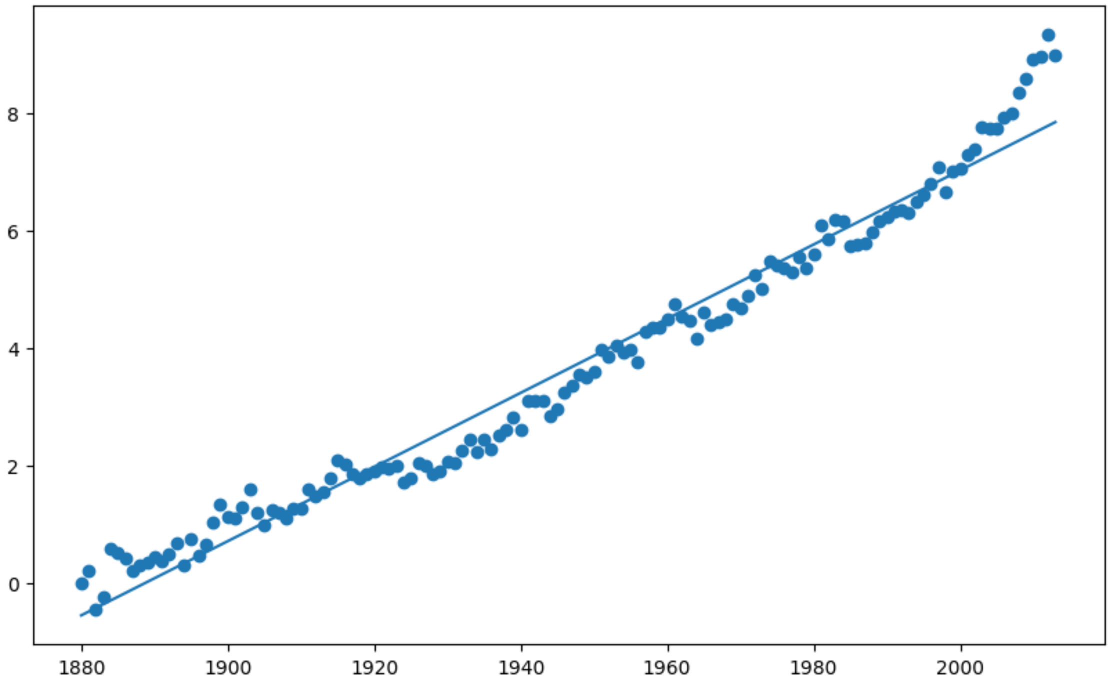
```
[134 rows x 5 columns]
```

1.2 Sea Level Values by years and predict line

1.2.1 Based on 1880 and 2013

[5]:

```
slope, intercept, r, p, std_err = linregress(df["Year"], df["CSIRO Adjusted Sea␣
  ↪Level"])

def myfunc(x):
  return slope * x + intercept

seamodel = list(map(myfunc, df["Year"]))
fig = plt.figure(figsize=(10,6))

plt.scatter(x=df["Year"], y=df["CSIRO Adjusted Sea Level"])
plt.plot(df["Year"], seamodel)
plt.show()
```

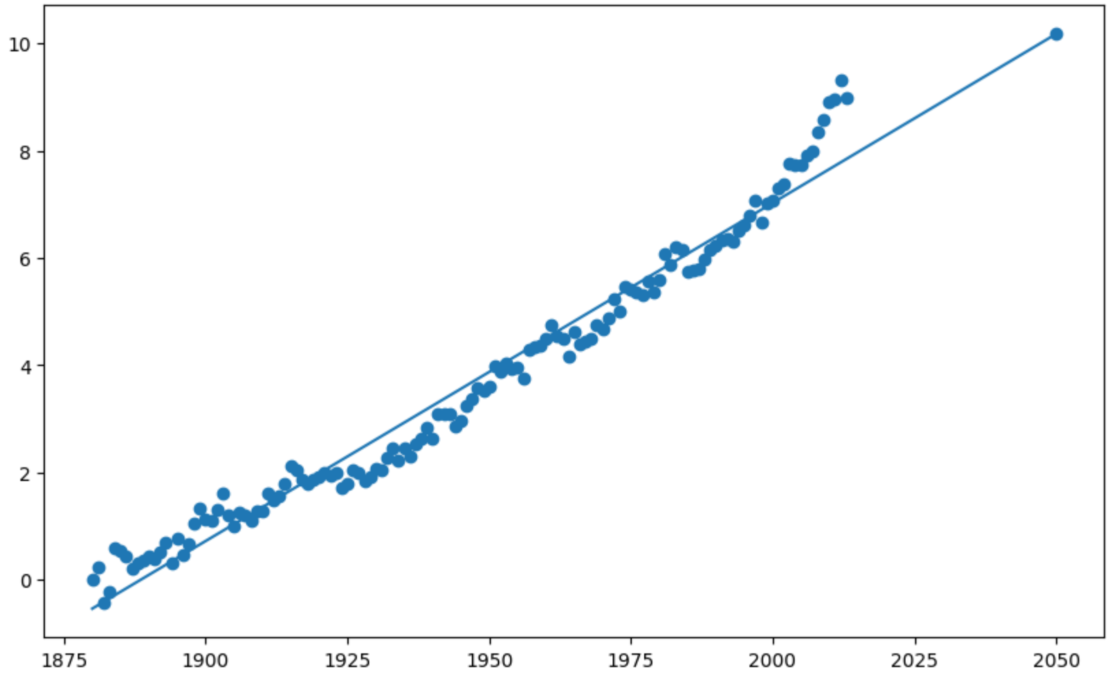


1.3 2050 sea level prediction

1.3.1 This prection calculated based on the data between 1880 and 2013

[6]:

```
slope, intercept, r, p, std_err = linregress(df["Year"], df["CSIRO Adjusted Sea Level"])

def myfunc(x):
  return slope * x + intercept

x_extended = pd.concat([df["Year"],pd.Series([2050])])
y_extended = pd.concat([df["CSIRO Adjusted Sea Level"], pd.Series([myfunc(2050)])])

slope, intercept, r, p, std_err = linregress(x_extended, y_extended)

seamodel = list(map(myfunc, x_extended))
fig = plt.figure(figsize=(10,6))

plt.scatter(x=x_extended, y=y_extended)
plt.plot(x_extended, seamodel)
plt.show()
```

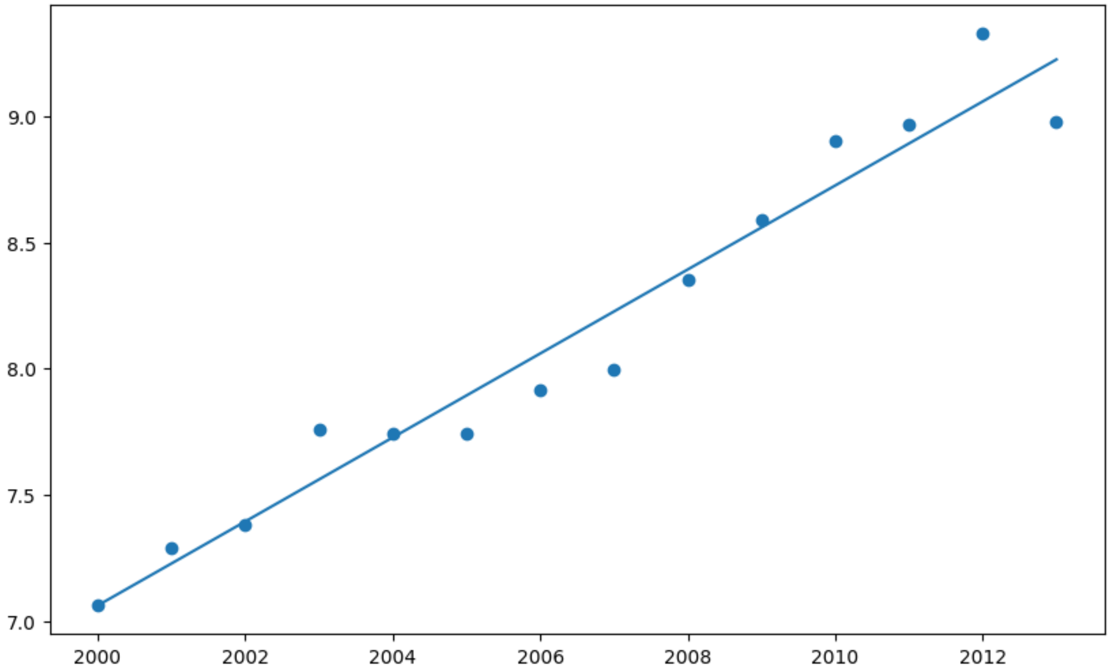


1.4 Sea Level Values by years and predict line

1.4.1 Based on 2000 and 2013

[9]:

```
df2k = df[df["Year"] >= 2000]
slope, intercept, r, p, std_err = linregress(df2k["Year"], df2k["CSIRO Adjusted␣
 ↪Sea Level"])

def myfunc(x):
  return slope * x + intercept

seamodel = list(map(myfunc, df2k["Year"]))
fig = plt.figure(figsize=(10,6))

plt.scatter(x=df2k["Year"], y=df2k["CSIRO Adjusted Sea Level"])
plt.plot(df2k["Year"], seamodel)
plt.show()
```



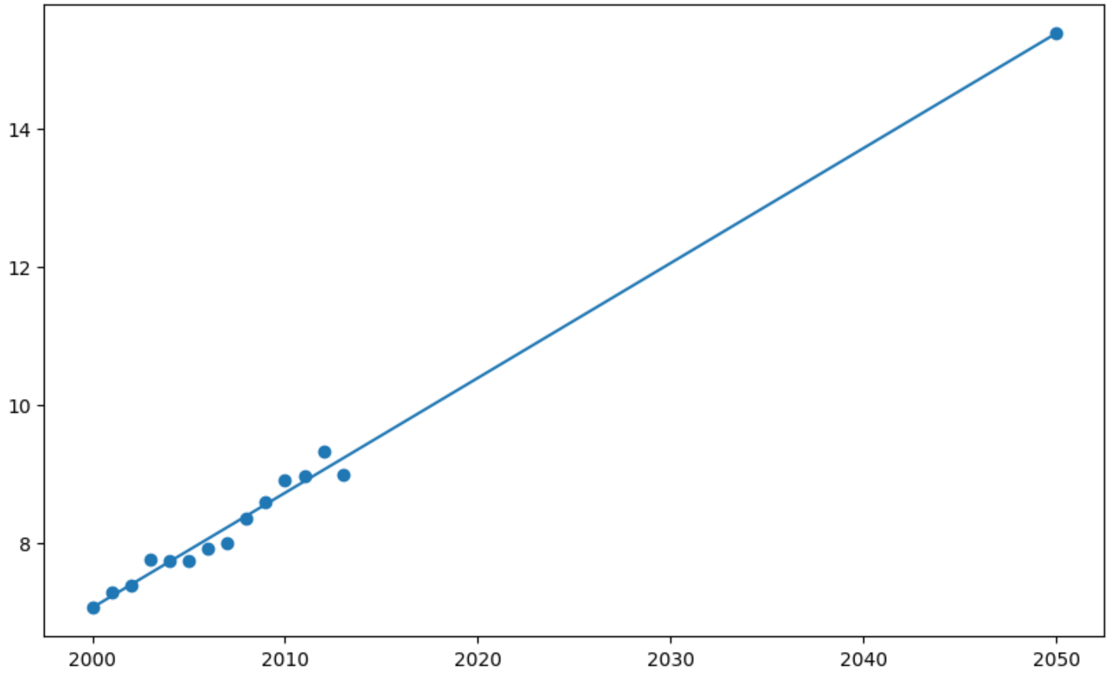
1.5 2050 sea level prediction

1.5.1 This prection calculated based on the data between 2000 and 2013

[10]:

```
df2k = df[df["Year"] >= 2000]
slope, intercept, r, p, std_err = linregress(df2k["Year"], df2k["CSIRO Adjusted␣
 ↪Sea Level"])
```

```
def myfunc(x):
  return slope * x + intercept

x_extended = pd.concat([df2k["Year"],pd.Series([2050])])
y_extended = pd.concat([df2k["CSIRO Adjusted Sea Level"], pd.
 ↪Series([myfunc(2050)])])

slope, intercept, r, p, std_err = linregress(x_extended, y_extended)

seamodel = list(map(myfunc, x_extended))
fig = plt.figure(figsize=(10,6))

plt.scatter(x=x_extended, y=y_extended)
plt.plot(x_extended, seamodel)
plt.show()
```



```
[ ]:
```