Least Square Fits for Modelling Population Growth Code

Valentina Watanabe
21367661

October 2024

```
    #!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Thu Oct 10 14:00:02 2024

@author: vwitch
"""
from matplotlib import pyplot as plt
import numpy as np
import scipy.optimize as optimization

gapminder_years =  [1750, 1800, 1820, 1850, 1870, 1875, 1900, 1910, 1913,
1920, 1925, 1930, 1940]
gapmind = [770, 985, 1093,  1278, 1347, 1383, 1645, 1790, 1829, 1924,
2007, 2100, 2324]

census_years = list(range(1949,2016,1))
census = [2557.63, 2594.94, 2636.77,  2682.05,  2730.23,  2782.1,  2835.3,
2891.35,  2948.14,  3000.72,  3043,  3083.97,  3140.09,  3209.83,  3281.2,
3350.43, 3420.68,  3490.33,  3562.31,  3637.16,  3712.7,  3790.33, 3866.57,  3942.1,
4016.61,  4089.08,  4160.19,  4232.08,  4304.11, 4379.01,  4451.36, 4534.41,  4614.57,
4695.74,  4774.57,  4856.46,  4940.57,  5027.2, 5114.56,  5201.44,  5288.96,  5371.59,
5456.14, 5538.27,  5618.68,  5699.2, 5779.44,  5857.97,  5935.21,  6012.07, 6088.57,
6165.22,  6242.02,  6318.59,  6395.7,
6473.04,  6551.26, 6629.91,  6709.05,  6788.21,
6858.58, 6936,  7013.87,  7092.13,  7169.97, 7247.89,  7326]

#Plotting data
plt.figure(figsize=(10, 6))
plt.scatter(gapminder_years,gapmind)
plt.scatter(census_years,census)
plt.title("World Population Growth (1750-2016)")
```

```
plt.xlabel("Years")
plt.ylabel("World Population (millions)")
plt.legend(["Years 1750-1940", "Years 1950-2016"], loc="lower right")
plt.grid(True)
plt.show()

#Log plotssss
log_gapminder = np.log(gapmind)
log_census = np.log(census)

plt.figure(figsize=(10, 6))
plt.scatter(gapminder_years,log_gapminder)
plt.scatter(census_years,log_census)
plt.title("Logarithm of World Population (1750-2016)")
plt.xlabel("Years")
plt.ylabel("ln(World Population (millions))")
plt.legend(["Years 1750-1940", "Years 1950-2016"], loc="lower right")
plt.grid(True)
plt.show()


#FITTING SECTIONNN


def func(x,a,b):
    return a + b*x

#Rangesss
gapminder_fit_range = [0, 12]  # Years 1820 - 1940 for fitting
census_fit_range = [0, 66]


# Fit the chosen time range for Gapminder data (actual years, no shifting)
gapminder_fit_params, _ = optimization.curve_fit(
    func, np.array(gapminder_years[gapminder_fit_range[0]:gapminder_fit_range[1]+1]),
    log_gapminder[gapminder_fit_range[0]:gapminder_fit_range[1]+1]
)

# Fit the chosen time range for Census data (actual years, no shifting)
census_fit_params, _ = optimization.curve_fit(
    func, np.array(census_years[census_fit_range[0]:census_fit_range[1]+1]),
    log_census[census_fit_range[0]:census_fit_range[1]+1]
)

# We get the parameters a and b for each
a_gapminder = gapminder_fit_params[0]
```

```
b_gapminder = gapminder_fit_params[1]
a_census = census_fit_params[0]
b_census = census_fit_params[1]

# Output the fit parameters
print(f"Fitted parameters: a = {a_gapminder}, b = {b_gapminder}")
# Output the fit parameters
print(f"Fitted parameters: a = {a_census}, b = {b_census}")


# Calculate n0 and  for both fits
n0_gapminder = np.exp(a_gapminder)
lambda_gapminder = b_gapminder
n0_census = np.exp(a_census)
lambda_census = b_census

# Output the fit parameters
print(f" n0 and : n0 = {n0_gapminder},  = {lambda_gapminder}")
# Output the fit parameters
print(f" n0 and : n0 = {n0_census},  = {lambda_census}")


def fitted(n0, lambdaa, t, t0):
    return n0 * np.exp(lambdaa * (t - t0))


# Plot the original population data with the fitted exponential curves
plt.figure(figsize=(10, 6))
plt.scatter(gapminder_years, gapmind, color='blue', label="Years 1750-1940 (Gapminder)")
plt.scatter(census_years, census, color='red', label="Years 1950-2016 (US Census)")

#Values for the fitted lines

gapminder_years = np.array(gapminder_years)
census_years = np.array(census_years)

t0_gapminder = gapminder_years[0]
t0_census = census_years[0]


fitted_gapminder_population = n0_gapminder * np.exp(lambda_gapminder *
(np.array(gapminder_years)))
fitted_census_population = n0_census * np.exp(lambda_census *
(np.array(census_years)))

# Plot the fitted lines on the original data
```

```
plt.plot(gapminder_years, fitted_gapminder_population, color='blue', linestyle='--',
label=f"Fit 1750-1940: ={lambda_gapminder:.4f}, n0={n0_gapminder:.2f}")

plt.plot(census_years, fitted_census_population, color='red',
linestyle='--', label=f"Fit 1950-2016: ={lambda_census:.4f}, n0=
{n0_census:.2f}")

plt.title("World Population Growth with Fitted Exponential Curves (1750-2016)")
plt.xlabel("Years")
plt.ylabel("World Population (millions)")
plt.legend(loc="upper left")
plt.grid(True)
plt.show()
```