Submission Weather Project Udacity

Yannick Mariman

Outline

- For the data extraction step I used SQL.

*SELECT * FROM global_data*

*SELECT * FROM city_list WHERE city = 'Amsterdam'*

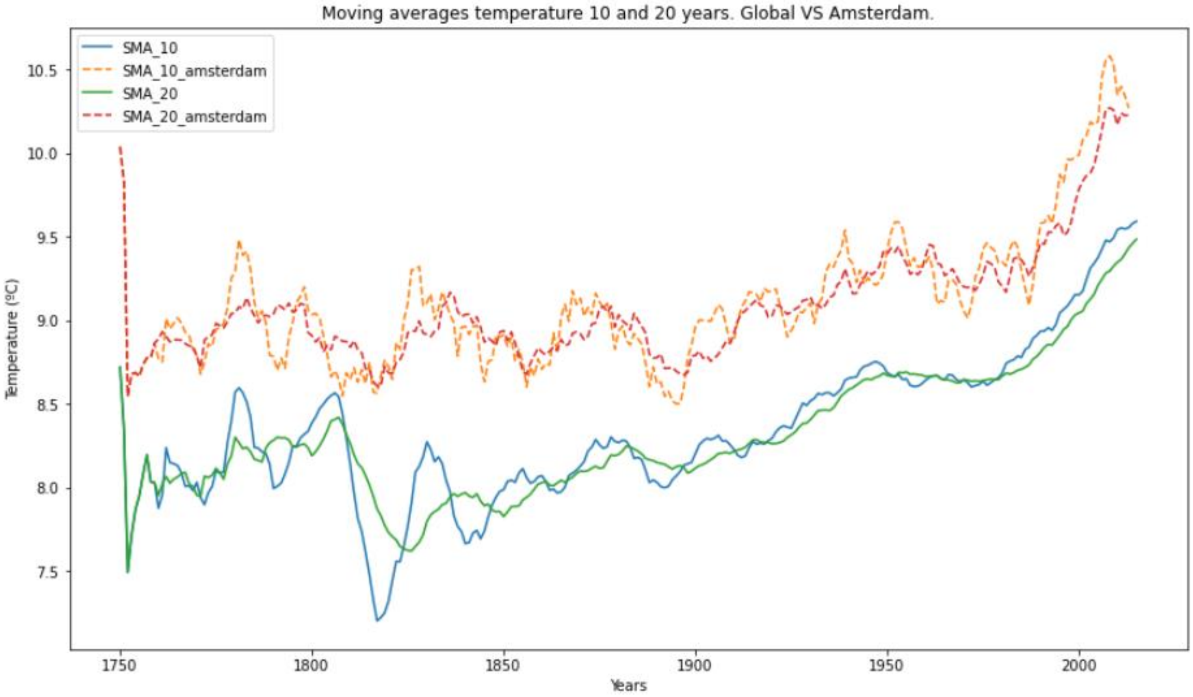
Respectively named globaldata.csv and citydata.csv. For the rest I used Python (see code at the end).

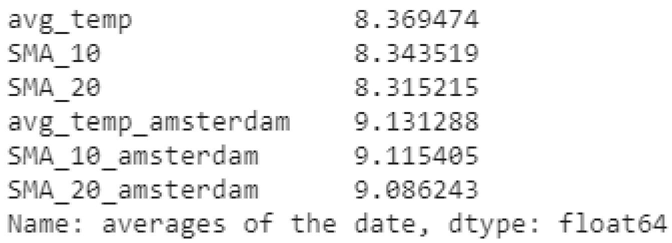
- I calculated the moving average using this function built in function: *.avg_temp.rolling(period, min_periods=1).mean()*
- My key considerations when deciding how to visualize the trends were on a coding level my proficiency in Python. I'm better at Python than Excel. Matplotlib is the to go visualization in Python.
    - I used the Moving averages from 10, 20, 30 and 40 years because those would smooth out better. I started with 5, 10, 15 and 20 but those were to 'spikey' in my opinion.

Changed after feedback:

- I only used the moving averages 10 and 20. Since they still contain the most information and don't clutter the view as much. I streamlined my code a little bit to accommodate for easier changing my SMA. Taking 30 or 40 years would smoothe out the lines too much in my opinion. The dip around 1815/1820 is not visible as much in the 30/40 years moving average.
- I added axis to the visualization.
- Added some complementary information in the conclusion and elaborated.

Visualization

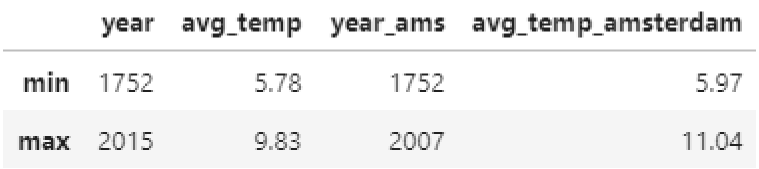




```
avg_temp                8.369474
SMA_10                  8.343519
SMA_20                  8.315215
avg_temp_amsterdam      9.131288
SMA_10_amsterdam        9.115405
SMA_20_amsterdam        9.086243
Name: averages of the date, dtype: float64
```

Observations

1. Amsterdam is always warmer when comparing moving averages than globally. As seen in the picture right above this. Where the means of the temperature and the moving averages of 10 and 20 years are visible.
2. In the beginning the lines are much more alike. This is caused because the moving averages aren't considering 10, 20, 30 or 40 years yet. This starts from 1760, 1770, 1780 and 1790 respectively.
3. Around 1815 there is a temperature dip globally which is not seen in Amsterdam.



	year	avg_temp	year_ams	avg_temp_amsterdam
min	1752	5.78	1752	5.97
max	2015	9.83	2007	11.04

4. Both globally and Amsterdam temperatures are increasing slowly. One readily observes in the table above that the minimum temperature global and in Amsterdam were both in 1752. While the maximum temperature globally was in 2015 and in Amsterdam was in 2007.
5. Temperature in 1750 makes a weird dip. But this is probably caused by one or two outliers. As mentioned in number 2 the moving averages are not yet averaging the outliers out. (Looking at the data there is a significant deep in 1752).
6. Some lines seem to start later, this is also caused by number 2. The curves weren't available at an earlier year as we use previous years for the calculation of the moving average. The moving averages using a higher number of years for the calculation are available later.

Sources of information/inspiration:

https://towardsdatascience.com/moving-averages-in-python-16170e20f6c

```
import pandas as pd
from pathlib import Path
import matplotlib.pyplot as plt

# Load the data
PROJECT = Path('D:/OneDrive -
MVGM/WerkbestandenYannick/Werkmap_Python/Projecten/Udacity/project1')
df_ams = pd.read_csv(PROJECT / 'citydata.csv')
df_glob = pd.read_csv(PROJECT / 'globaldata.csv')

#Set index and adjust dataframe
df_ams = df_ams[['year','avg_temp']]
df_ams = df_ams[df_ams.year >= 1750].set_index('year')
df_glob = df_glob.set_index('year')

# Add the Simple Moving Averages
def add_sma(df, period):
    df[f'SMA_{period}'] = df.avg_temp.rolling(period, min_periods=1).mean()
    return df

sma_periods = [10*y for y in range(1,3)]
for sma_period in sma_periods:
    df_ams = add_sma(df_ams,sma_period)
    df_glob = add_sma(df_glob,sma_period)

#merge the dfs
df = df_glob.join(df_ams, how='outer', rsuffix='_amsterdam')
df = df.reset_index()

print(df.mean().rename('averages of the date')[1:])

minmax_glob = pd.concat([df[df.avg_temp ==
df.avg_temp.min()][['year','avg_temp']].rename(index={2:'min'}),
        df[df.avg_temp == df.avg_temp.max()][['year','avg_temp']].rename(index={265:'max'})])
minmax_ams = pd.concat([df[df.avg_temp_amsterdam ==
df.avg_temp_amsterdam.min()][['year','avg_temp_amsterdam']].rename(index={2:'min'}),
          df[df.avg_temp_amsterdam ==
df.avg_temp_amsterdam.max()][['year','avg_temp_amsterdam']].rename(index={257:'max'})])

display(minmax_glob.join(minmax_ams, rsuffix='_ams'))

#Visualize
plt.figure(figsize=(14,8))
plt.title('Moving averages temperature 10 and 20 years. Global VS Amsterdam.')
plt.xlabel("Years")
plt.ylabel("Temperature (ºC)")
for sma_period in sma_periods:
    plt.plot('year', f"SMA_{sma_period}", data=df)
    plt.plot('year', f"SMA_{sma_period}_amsterdam", data=df, linestyle='dashed')
plt.legend()
plt.show()
```