exercise_05_subplots

August 12, 2021

1 Exercise 05 - subplots and statistics

In notebook 05, we saw how subplots can be used to display several datasets at once.

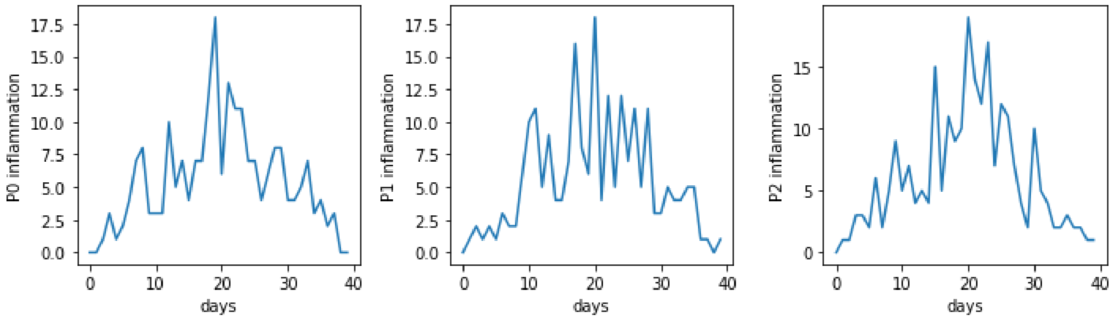
For example, run the code below. It should display the inflamation of three patients.

[1]:

```
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt

filename = '../data/inflammation-01.csv'
data = np.loadtxt(filename, delimiter=',')
```

[2]:

```
fig = plt.figure(figsize=(10.0, 3.0))

axes1 = fig.add_subplot(1, 3, 1)
axes2 = fig.add_subplot(1, 3, 2)
axes3 = fig.add_subplot(1, 3, 3)

axes1.plot(data[0,:])
axes1.set_ylabel('P0 inflammation')
axes1.set_xlabel('days')

axes2.plot(data[1,:])
axes2.set_ylabel('P1 inflammation')
axes2.set_xlabel('days')

axes3.plot(data[2,:])
axes3.set_ylabel('P2 inflammation')
axes3.set_xlabel('days')

fig.tight_layout()
```



2 Exercise

Combining the visualization tools of matplotlib with some of the functions we used with numpy, we can generate some more descriptive plots.

2.0.1 Task:

Read the documentation for numpy.max() (https://numpy.org/doc/stable/reference/generated/numpy.ndarray.ma

Use the subplot code above, but in this case plot the mean, maximum, and minimum of all patients over time using numpy functions:

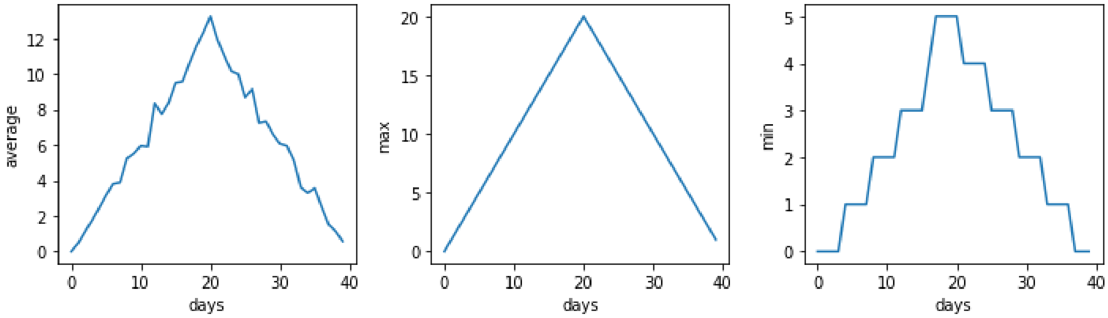
```
np.mean(data, axis = __)
np.max(data, axis = __)
np.min(data, axis = __)
```

You can create variables for each statistic, or you can put the function directly into the `plt.plot()` function, it's up to you. Label your axes appropriatly.

[3]:

```
# answer here:

%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt

filename = '../data/inflammation-01.csv'
data = np.loadtxt(filename, delimiter=',')

fig = plt.figure(figsize=(10.0, 3.0))

axes1 = fig.add_subplot(1, 3, 1)
axes2 = fig.add_subplot(1, 3, 2)
axes3 = fig.add_subplot(1, 3, 3)

axes1.set_ylabel('average')
axes1.set_xlabel('days')
```

```
axes1.plot(np.mean(data, axis=0))

axes2.set_ylabel('max')
axes2.set_xlabel('days')
axes2.plot(np.max(data, axis=0))

axes3.set_ylabel('min')
axes3.set_xlabel('days')
axes3.plot(np.min(data, axis=0))

fig.tight_layout()
```


average
days
max
days
min
days