Project focusing on Data Analysis and Visualization via Python Matplotlib

Data Bootcamp – Homework-Matplotlib

kimberly GORDON

2018

Contents

Objectives

Terminology

Requirements

Details

* + - Plot 1
    - Plot 2
    - Plot 3
    - Plot 4

Resources

Objectives

1. Create scatter plot that shows how the tumor volume changes over time for each treatment.
2. Create a scatter plot that shows how the number of metastatic (https://en.wikipedia.org/wiki/Metastasis) (cancer spreading) sites changes over time for each treatment.
3. Create a scatter plot that shows the number of mice still alive through the course of treatment (Survival Rate)
4. Create a bar graph that compares the total % tumor volume change for each drug across the full 45 days.

Setting up plots

1. How does the DataFrame need to be structured to have the right X and Y axis?

dataframe=pd.DataFrame(example, columns =[‘Height’, ‘Latitude’])

* + 1. In the series within the panda DataFrame, assign x = Height and y = Latitude.
       - X = dataframe.Height
       - Y = dataframe.Latitude
    2. Plot them as a scatter chart by adding:
       - plt .scatter(x,y)
    3. Show or save the chart with the following:
       - plt.show()
       - plt.savefig(“chart-name.png”)

1. How do I build a basic scatter plot?

Where s is the size of the dots.

Import matplotlib.pyplot as plt

plt.title(“Scatter plot”, fontsize=24)

x\_values = [2, 5, 8, 11, 14]

y\_values = [2, 6, 10, 14, 18]

plt.scatter(x\_values, y\_values, s=100)

plt.show()

1. How do I add a label to that scatter plot?

plt.xlabel(“Height (m)”, fontsize=20)

plt.ylabel(“Latitude”, fontsize=20)

Font size of the number labels on the axes

plt.xticks(fontsize=16)

plt.yticks(fontsize=16)

1. Where would the labels for that scatter plot originate?

The labels are identified by xlabel, ylabel, etc., in the code along with the fontsize

Terminology

* Matplotlib
  + Python package used for data plotting and visualization.
* Metastasis
  + Pathogenic agent’s spread from an initial or primary site to a different or secondary site within the host’s body; it is typically spoken of as such spread by a cancerous tumor. The newly pathological sites are metastases (mets).

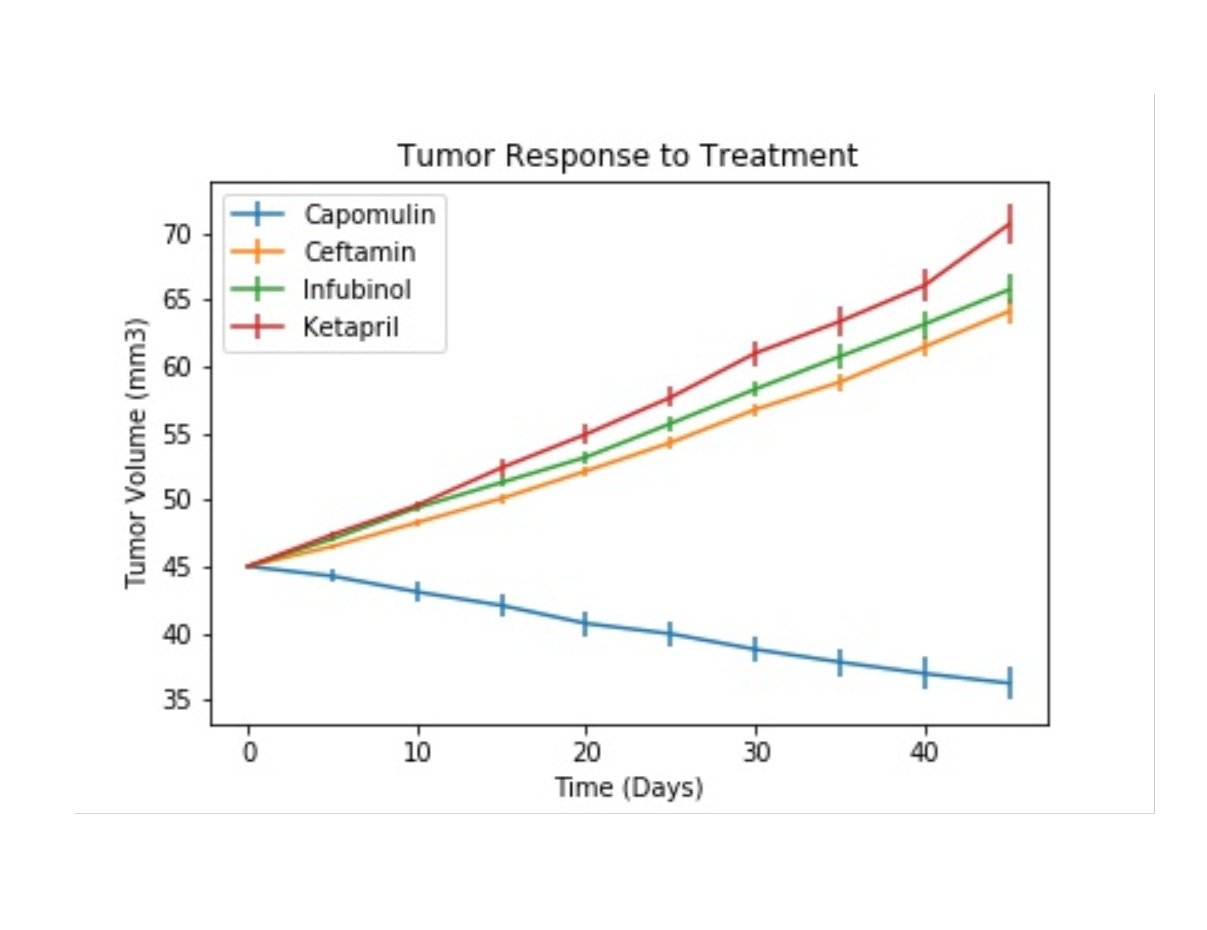
Requirements

(*Assumptions and Considerations)*

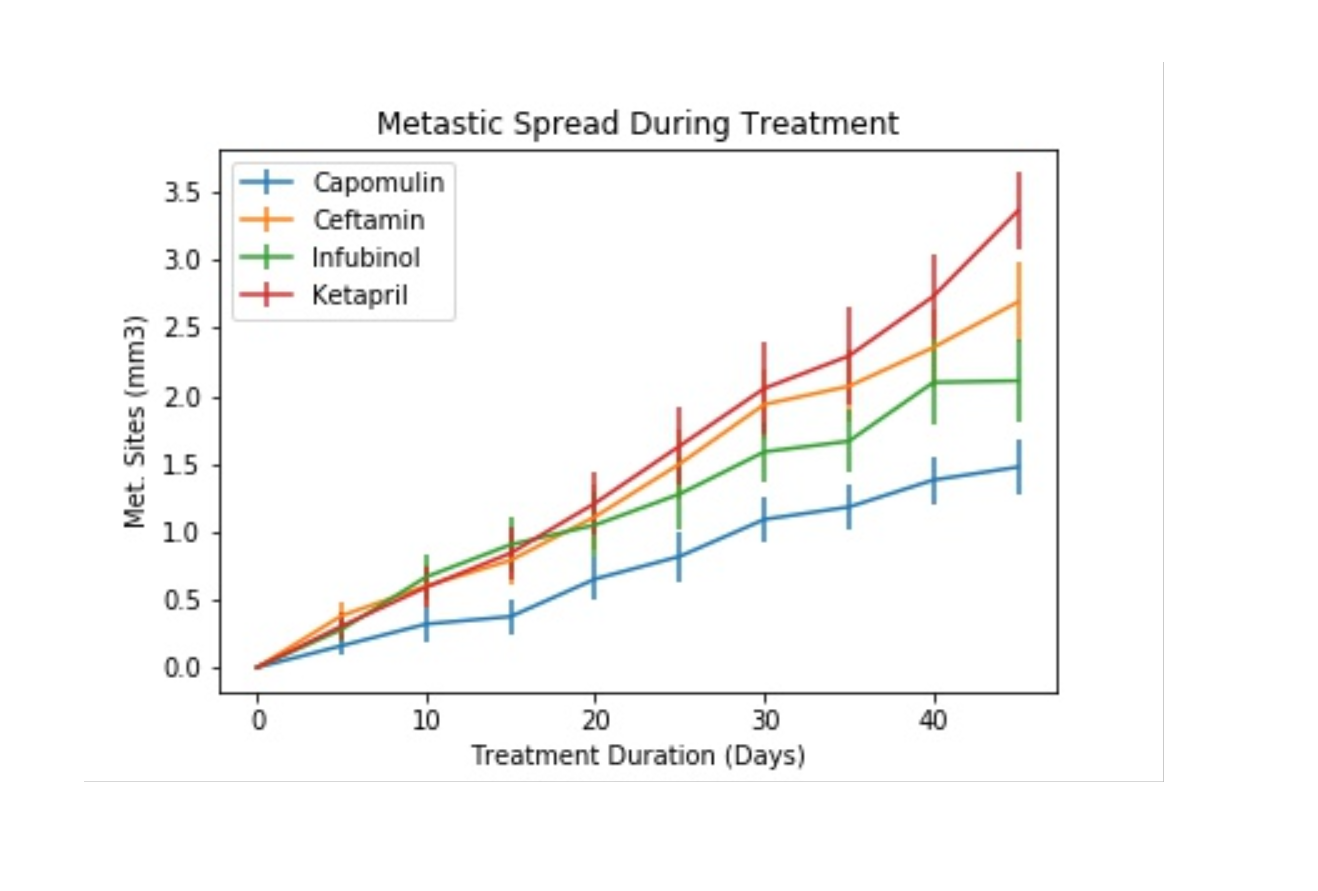
* Use the Pandas Library, Jupyter Notebook and Matplotlib library.
* Include a written description of three observable trends based on the data.
* Plot Titles, Axes Labels, Legend Labels, X and Y Axis Limits, etc.
* Scatter plots must include error bars (<https://en.wikipedia.org/wiki/Error_bar>). This will allow the company to account for variability between mice.
* Look into pandas.DataFrame.sem (<http://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.sem.html>) for ideas on how to calculate error bars.
* Consider aesthetics!
* Legends should not overlay any data.
* Bar graph should indicate tumor growth as red and tumor reduction as green. It should also include a label with the percentage change for each bar.
* Consult tutorial (http://composition.al/blog/2015/11/29/a-better-way-to-add-labels-to-bar-chartswith-matplotlib/) for relevant code snippets.
* See Starter Workbook (Pymaceuticals/pymaceuticals\_starter.ipynb) for a reference on expected format.
* Matching the tables or data frames included in starter package is not required.
* The only goal is to build the scatter plots and bar graphs.

Details

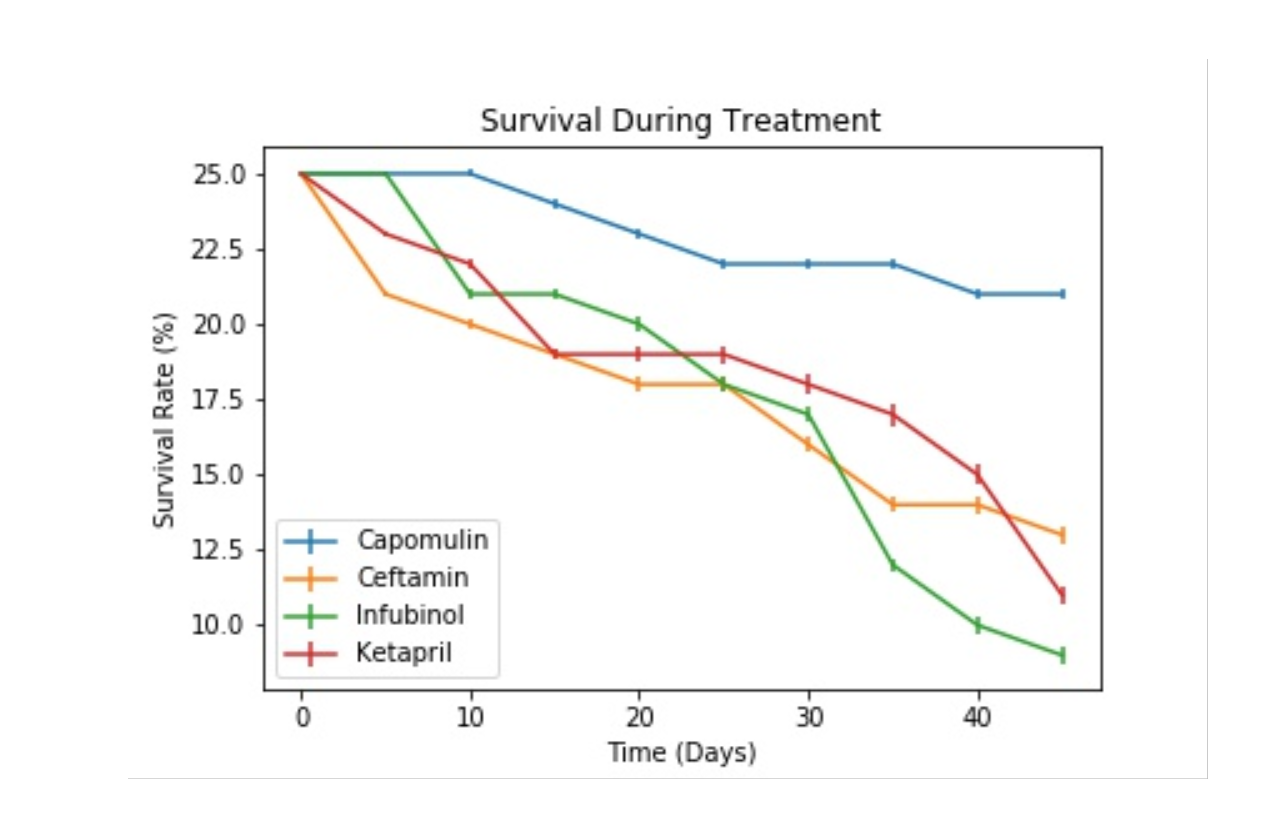
1. Scatter plot that shows how the tumor volume changes over time for each treatment.



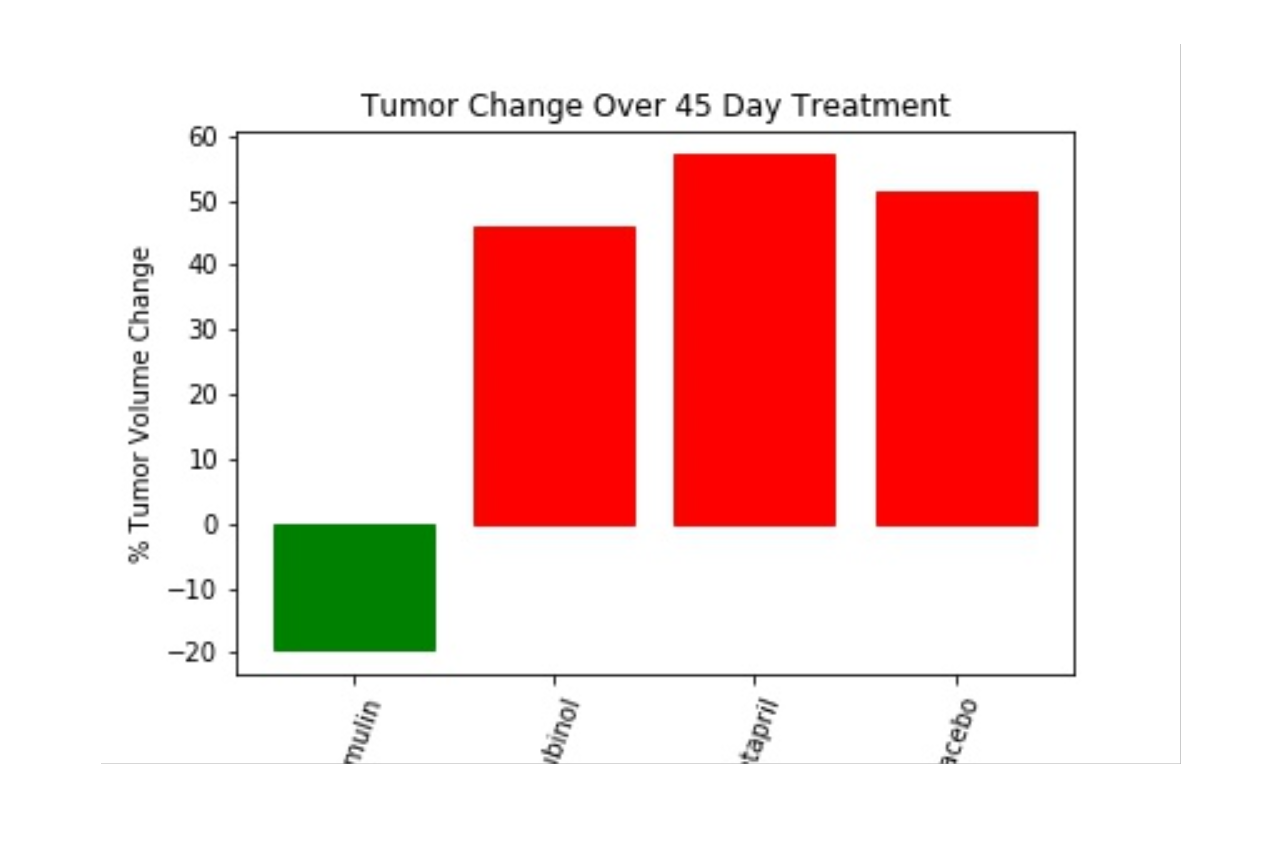
1. Create a scatter plot that shows how the number of metastatic (https://en.wikipedia.org/wiki/Metastasis) (cancer spreading) sites changes over time for each treatment.



1. Create a scatter plot that shows the number of mice still alive through the course of treatment (Survival Rate)



1. Create a bar graph that compares the total % tumor volume change for each drug across the full 45 days.



Resources

Skills Developed

* Understand Matplotlib's pyplot interface
* Ability to:
  1. create line; bar; scatter; and pie charts
  2. use basic plot configuration options, such as `xlim` and `ylim`
  3. create plots using the `DataFrame.plot()` method
  4. understand the advantages and disadvantages of creating charts using the `DataFrame.plot()` method
  5. work through a complex data set using Pandas and then chart some visualizations based upon the cleaned DataFrame
  6. define \*\*mean\*\*, \*\*median\*\*, and \*\*mode\*\*, and choose which one is most appropriate to describe a given data set
  7. explain the meaning of variance and standard deviation
  8. describe standard error and the difference between a sample and a population
  9. add error bars to their plots
  10. fit lines to their data

Activities

* [Basic Plot](1/Activities/02-Stu\_NJTemp)
* [Bar Charts](1/Activities/08-Stu\_PyBars)
* [Scatter Plots](1/Activities/12-Stu\_ScatterPy)
* [Pandas Plotting](2/Activities/03-Stu\_BattlingKings)
* [Plotting Groups](2/Activities/05-Stu\_BikeTrippin)
* [Plotting Multi-lines](2/Activities/07-Ins\_PandasMultiLine)
* [Mean, Median, Mode](3/Activities/01-Ins\_Mean\_Median\_Mode)
* [Variance](3/Activities/02-Ins\_Variance\_and\_Z\_Score)
* [Quartiles](3/Activities/04-Stu\_Quartiles\_and\_Outliers)
* [Standard Error](3/Activities/06-Stu\_Standard\_Error)

Links

* Numpy <http://www.numpy.org/>
* Matplotlib Gallery <https://matplotlib.org/gallery.html>
* Pandas Plotting <https://pandas.pydata.org/pandas-docs/stable/visualization.html>
* Alternatives to Pie Charts <http://www.storytellingwithdata.com/blog/2014/06/alternatives-to-pies>
* Creating a Twitter Dev Account Supplemental/Dev\_Account\_Creation.pdf

# Mean, Median, Mode

## Instructions

For each activity below, assume the input is a simple list.

\* Write a function to calculate the mean of a data set, provided as a list.

\* Write a function to calculate the median of a data set. Ensure your function works if the length of the input list is even. You can simply return either "middle" element, or return the mean of the two.

\* Write a function to calculate the mode of a data set. It should return the most common element, and how many times it appears. If your data set has multiple modes, simply return one. Use a [Counter](https://docs.python.org/2/library/collections.html#collections.Counter) from the collections module to implement this function.

\* Consider the lists provided in the [samples file](../Activities/01-Ins\_Mean\_Median\_Mode/Solved/samples.ipynb). Which measure of central tendency best describes each list? Why?

\* As a \*\*bonus\*\*, write a function that returns all modes of a data set. For the list `[1, 1, 1, 2, 2, 2, 3, 4]`, this function should return `[(1, 3), (2, 3)]`; or just `[1, 2]`.

<https://en.wikipedia.org/wiki/Metastasis>