CS5785 Homework 0

Due date: Thursday, September 11

Overview

Welcome to CS 5785! After completing this homework, you should be able to find a teammate, set up your preferred development environment, download and parse a dataset, and use visualization tools to help you understand what that dataset contains. Completing this homework will give you the scaffolding you need for the rest of the course.

If you need help

There are several strategies available to you. If you ever get stuck, the best way is to ask your teammates on Piazza. That way, your solutions will be available to the other students in the class. If you use the Web for help, be sure to *cite all of your sources*. Finally, your TAs will offer office hours, which are a great way to get some one-on-one help.

Submission instructions

The homework is generally split into programming exercises and written exercises. You should turn in an electronic copy of your solutions to the homework. Please submit **all your code** and **a detailed, organized write-up** to CMS. If there are any questions about submission please refer to Piazza for clarification or further details. You are responsible for submitting clear, organized answers to the questions. Please include all relevant information for a question, including text response, equations, figures, graphs, etc. Also, please pay attention to the discussion board for relevant information regarding updates, tips, and policy changes. You are encouraged (but not required) to work in groups of 2.
The write-up should contain a general summary of what you did, how well your solution works, any insights you found, etc. It need not be excessive, but we certainly enjoy receiving write-ups in a full lab report format. On the cover page, include the class name, homework number, and team member names.

1 Setting up Python

1. Find your teammate. You are encouraged (but not required) to work in groups of 2. If you do decide to work in a team, inlcude both teammates' names on the report and in your submission email. One good way of finding teammates is to check the "*Search for Teammates!*" post on Piazza.

2. Sign up for an account on Kaggle (`http://www.kaggle.com/`). Some future homeworks may analyze datasets from Kaggle and you will need an account to download them, so it is best to get this out of the way now.

3. Set up a working environment for machine learning. See the companion document, "*How to set up Python using Enthought Canopy*," for instructions; we will post a link to it on Piazza. We recommend using Python with one of tthese two IDEs:

   Enthought Canopy

   Pycharm

   Or, feel free to use other languages and tools such as Matlab and Java, but note that the TAs will only give technical support for Python—you will be on your own.

   Find out how to install packages within your Python environment and install the following packages: scikits.image, scikit-learn, matplotlib, and numpy. Some of them might already be installed if you set up Canopy correctly. You can verify that you have these packages by running the following Python code:

   ```
   import sklearn
   import skimage
   import numpy
   from matplotlib import *
   ```

   If you get an `ImportError`, that means the package is not installed.

2 IRIS FLOWERS

In 1935, Edgar Anderson went to his favourite pasture and recorded the length and width of the sepals and petals on several flowers in the field. For whatever reason, this dataset became one of the oldest and most well-known "sanity-check" datasets around, being cited by countless papers. This class continues this time-honored tradition by using *Iris Flowers* to sanity-check your Python environment and plotting libraries.

1. Find and download the Iris Flowers dataset from the UC Irvine Machine Learning datasets archive at `http://archive.ics.uci.edu/ml/datasets.html` Hint: The `iris.names` file describes the structure of the dataset. How many features/attributes are there per sample? How many different species are there, and how many samples of each species did Anderson record?

2. Figure out how to parse the dataset you downloaded. Load the samples into an $N \times p$ array, where N is the number of samples and p is the number of attributes per sample. Aditionally, create a N-dimensional vector containing each sample's label (species).

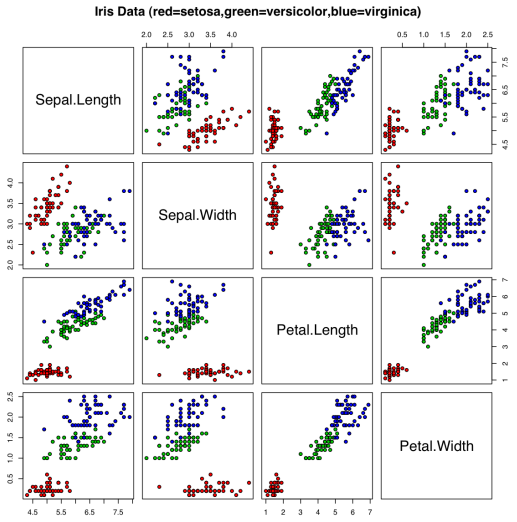
   Hint: Python has a built-in CSV parser in the `csv` library, or you can use the `"string".split(...)` method.

   Hint 2: Here is some code that prints each line in a file:

   ```
   for line in open("/path/to/filename.txt"):
       print "Line contains: "+line
   ```

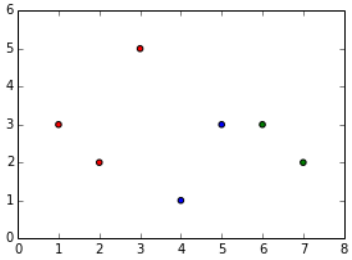
3. To visualize this dataset, we would have to build a p-dimensional scatterplot. Unfortunately, we only have 2D displays so we must reduce the dataset's dimensionality. The easiest way to view the set is to plot two attributes of the data against one another and repeat for each pair of attributes.

Create every possible scatterplot from all pairs of two attributes. (For example, one scatterplot would graph petal length vs sepal width, another would graph petal length vs. sepal length, and so on). Within each scatterplot, the color of each dot should correspond with the sample species. Ideally, we're looking for something like this figure from Wikipedia:



But your results do not have to be this ornate. Presenting six separate figures in your report is certainly fine. Be sure to include the source code for all plots!

Hint: This is one way to draw a scatterplot. Use whatever works for you.



```
xs      = numpy.array([1, 2, 3, 4, 5, 6, 7])
ys      = numpy.array([3, 2, 5, 1, 3, 3, 2])
colors = ["r","r","r","b","b","g","g"]
scatter(xs, ys, c=colors)
savefig("plot.png")
```

Good luck!