Homework 0

CS 5785 Applied Machine Learning

Xialin Shen <xs293@cornell.edu>

An Le <aql6@cornell.edu>

24th August, 2017

Question 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?

ANSWERS:

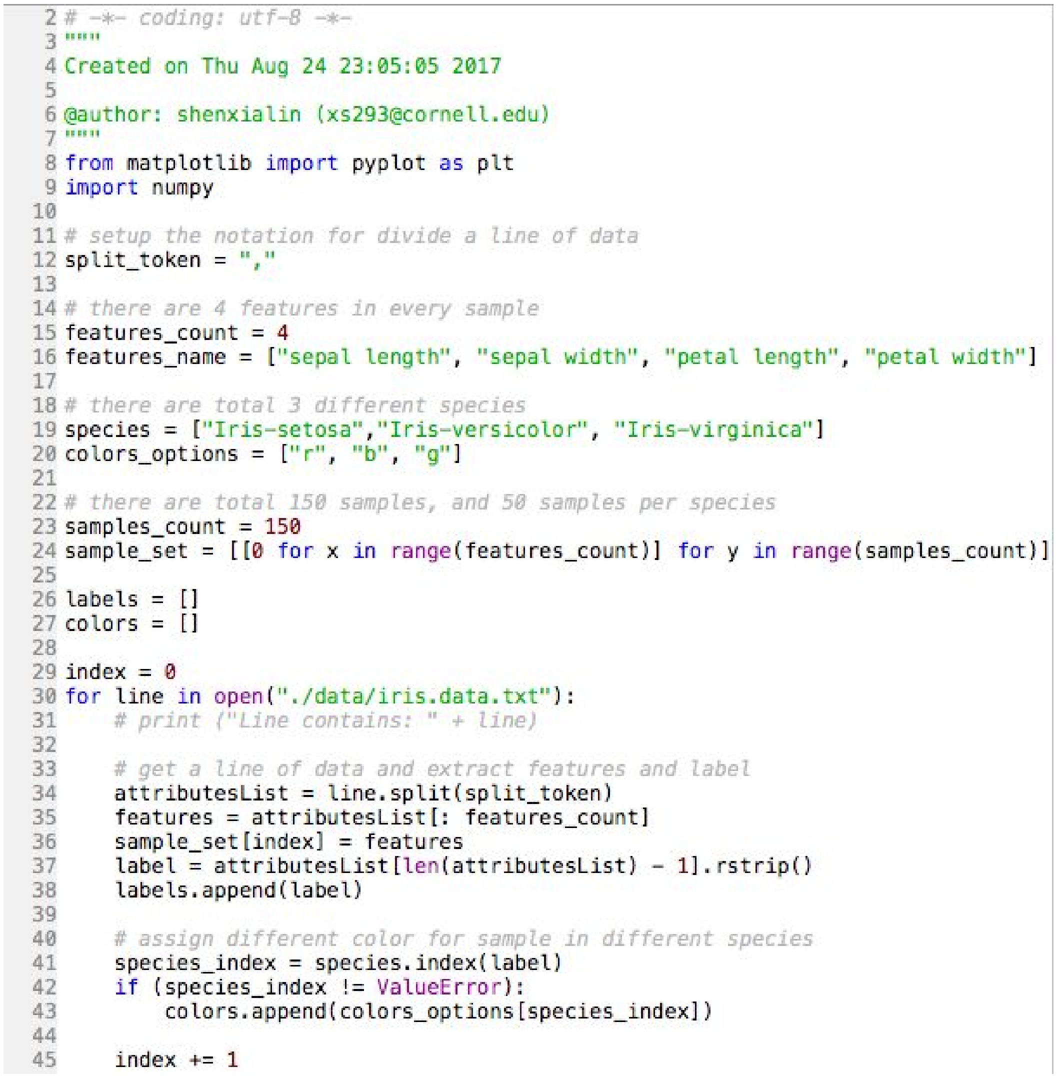
1. There are **4** attributes per sample, including “sepal length in cm”, “sepal width in cm”, “petal length in cm” and “petal width in cm”
2. There are **3** different species, including “Iris Setosa”, “Iris Versicolour” and “Iris Virginica”
3. Anderson recorded **50** samples for each species.

Question 2

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

Xialin:

CODES:



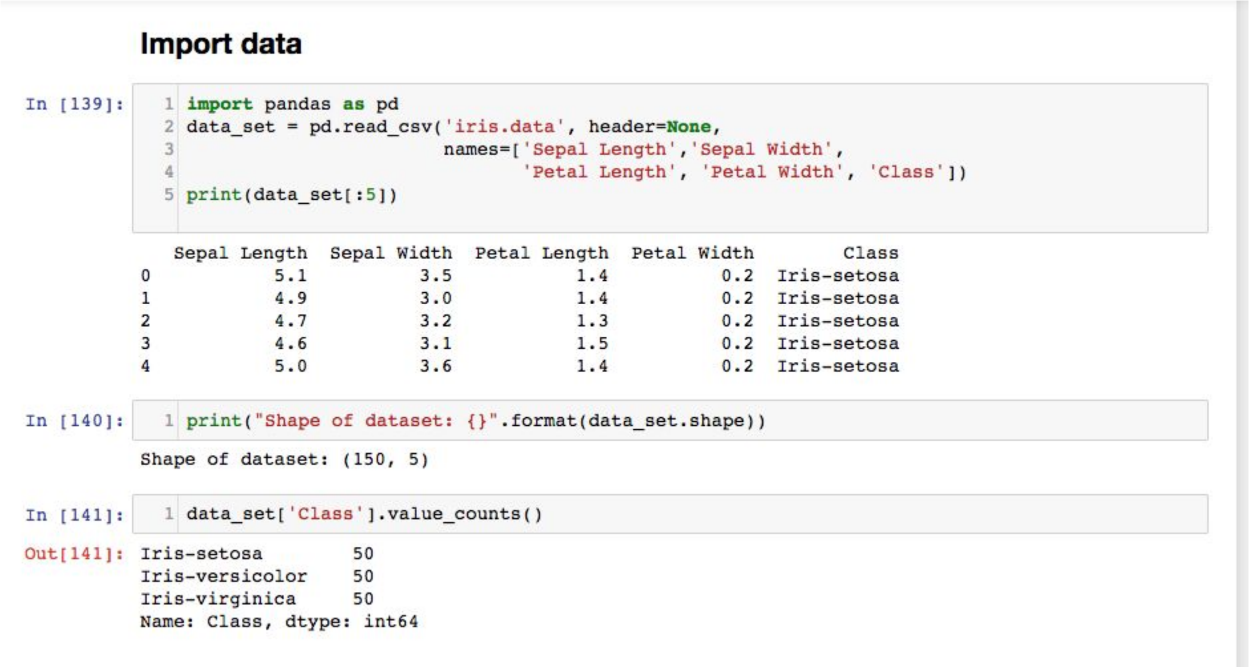
```python
# -*- coding: utf-8 -*-
"""
Created on Thu Aug 24 23:05:05 2017

@author: shenxialin (xs293@cornell.edu)
"""
from matplotlib import pyplot as plt
import numpy

# setup the notation for divide a line of data
split_token = ","

# there are 4 features in every sample
features_count = 4
features_name = ["sepal length", "sepal width", "petal length", "petal width"]

# there are total 3 different species
species = ["Iris-setosa","Iris-versicolor", "Iris-virginica"]
colors_options = ["r", "b", "g"]

# there are total 150 samples, and 50 samples per species
samples_count = 150
sample_set = [[0 for x in range(features_count)] for y in range(samples_count)]

labels = []
colors = []

index = 0
for line in open("./data/iris.data.txt"):
    # print ("Line contains: " + line)

    # get a line of data and extract features and label
    attributesList = line.split(split_token)
    features = attributesList[: features_count]
    sample_set[index] = features
    label = attributesList[len(attributesList) - 1].rstrip()
    labels.append(label)

    # assign different color for sample in different species
    species_index = species.index(label)
    if (species_index != ValueError):
        colors.append(colors_options[species_index])

    index += 1
```

RESULTS:

<sample_set>: a 150 x 4 dimensions matrix

<labels>: a list with 150 labels

An:



Import data

In [139]:

```
import pandas as pd
data_set = pd.read_csv('iris.data', header=None,
                       names=['Sepal Length','Sepal Width',
                              'Petal Length', 'Petal Width', 'Class'])
print(data_set[:5])
```

```
   Sepal Length  Sepal Width  Petal Length  Petal Width        Class
0           5.1          3.5           1.4          0.2  Iris-setosa
1           4.9          3.0           1.4          0.2  Iris-setosa
2           4.7          3.2           1.3          0.2  Iris-setosa
3           4.6          3.1           1.5          0.2  Iris-setosa
4           5.0          3.6           1.4          0.2  Iris-setosa
```

In [140]:

```
print("Shape of dataset: {}".format(data_set.shape))
```

```
Shape of dataset: (150, 5)
```

In [141]:

```
data_set['Class'].value_counts()
```

Out[141]:

```
Iris-setosa        50
Iris-versicolor    50
Iris-virginica     50
Name: Class, dtype: int64
```

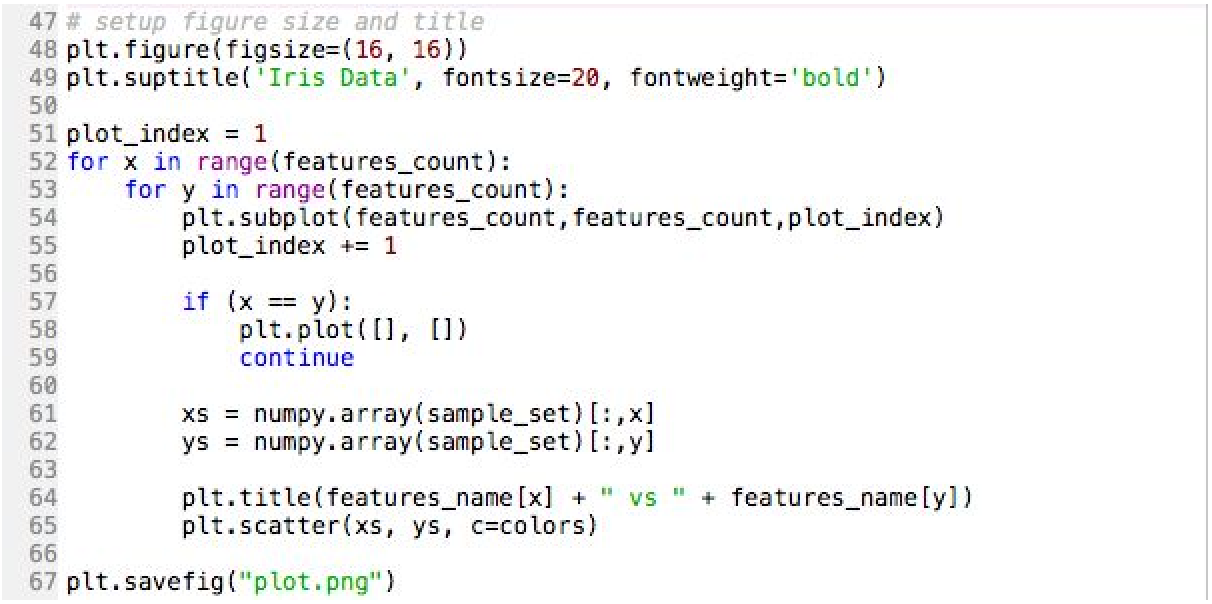
Question 3

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.

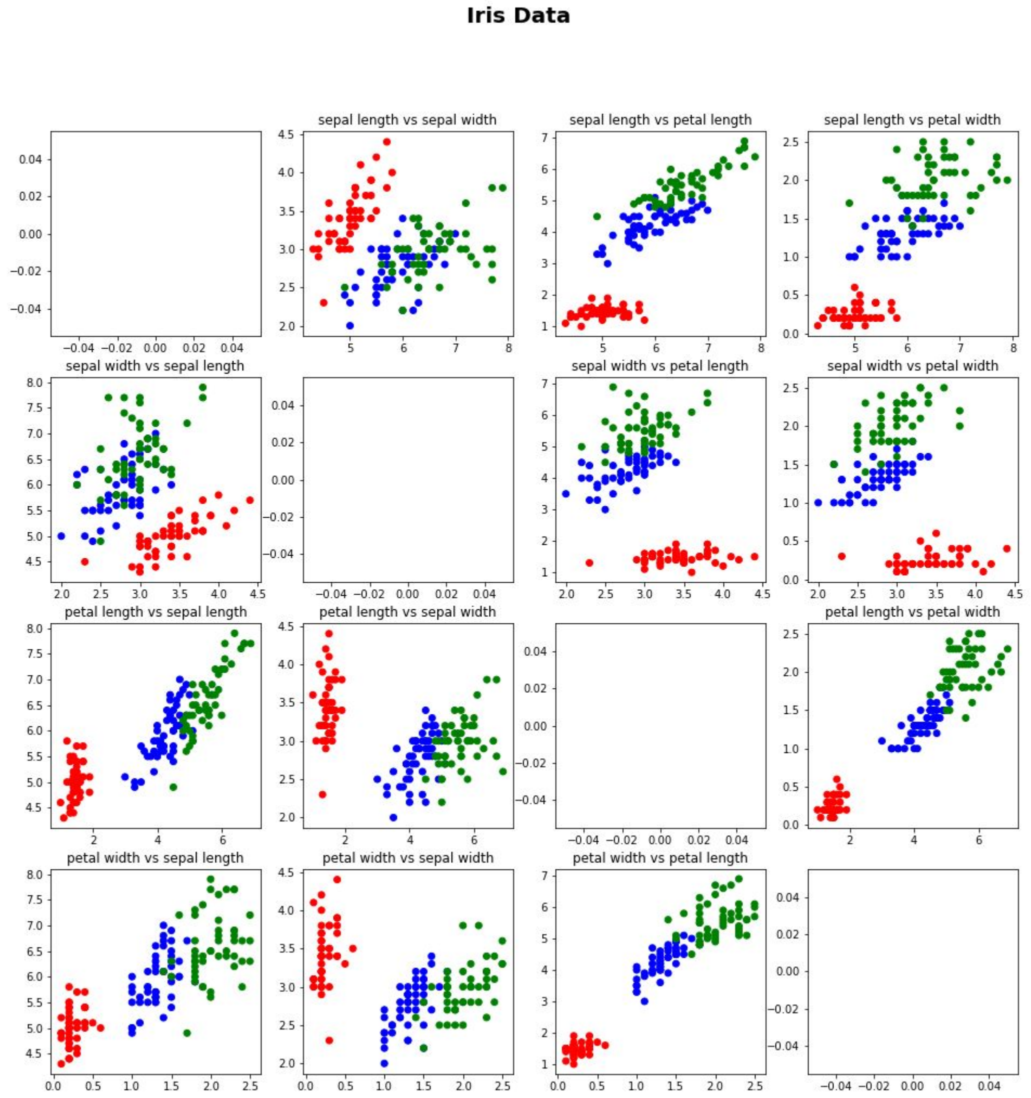
Xialin:

CODES (CON'T):



```
# setup figure size and title
plt.figure(figsize=(16, 16))
plt.suptitle('Iris Data', fontsize=20, fontweight='bold')

plot_index = 1
for x in range(features_count):
    for y in range(features_count):
        plt.subplot(features_count,features_count,plot_index)
        plot_index += 1

        if (x == y):
            plt.plot([], [])
            continue

        xs = numpy.array(sample_set)[:,x]
        ys = numpy.array(sample_set)[:,y]

        plt.title(features_name[x] + " vs " + features_name[y])
        plt.scatter(xs, ys, c=colors)

plt.savefig("plot.png")
```

RESULTS:



Iris Data

sepal length vs sepal width

sepal length vs petal length

sepal length vs petal width

sepal width vs sepal length

sepal width vs petal length

sepal width vs petal width

petal length vs sepal length

petal length vs sepal width

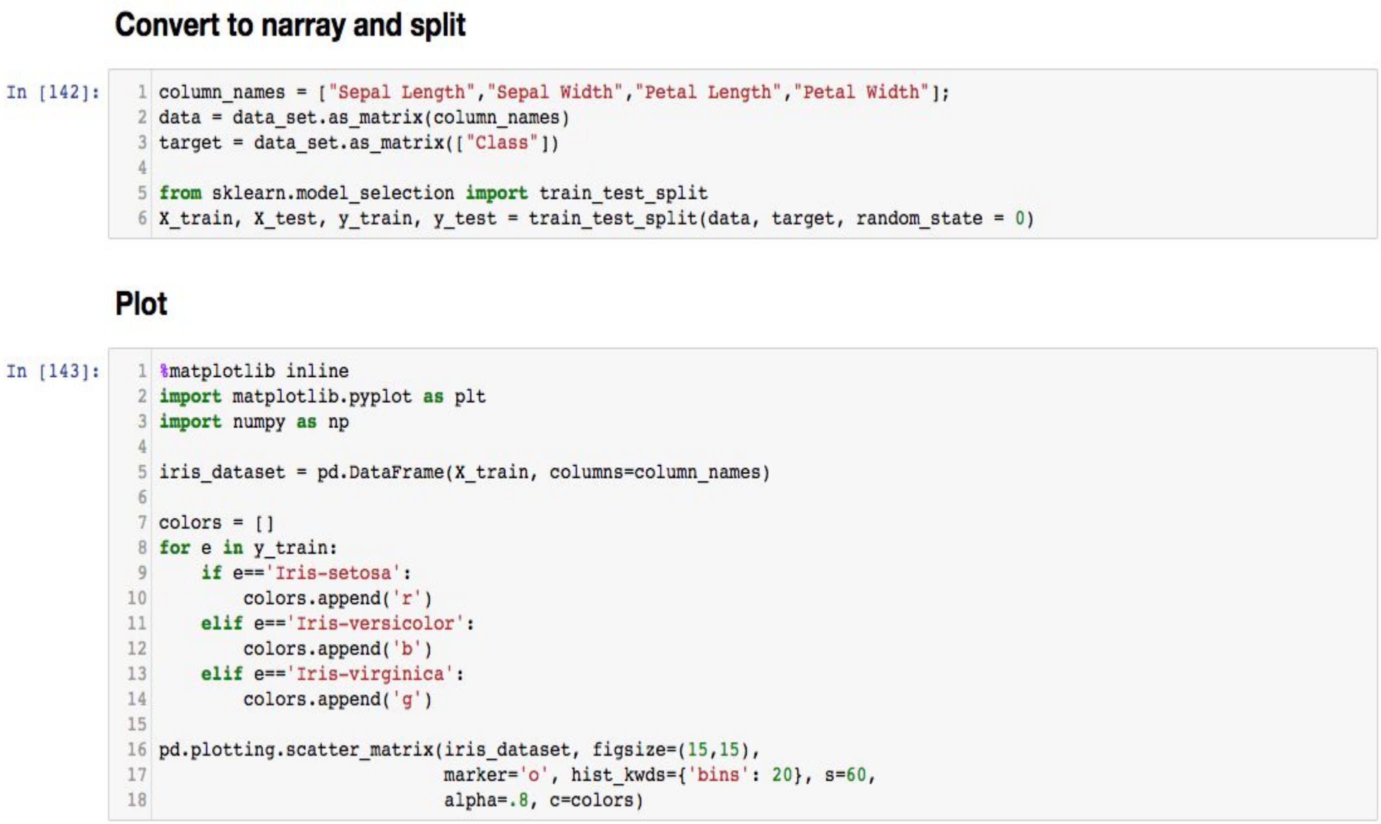
petal length vs petal width

petal width vs sepal length

petal width vs sepal width

petal width vs petal length

An:



Convert to narray and split

In [142]:

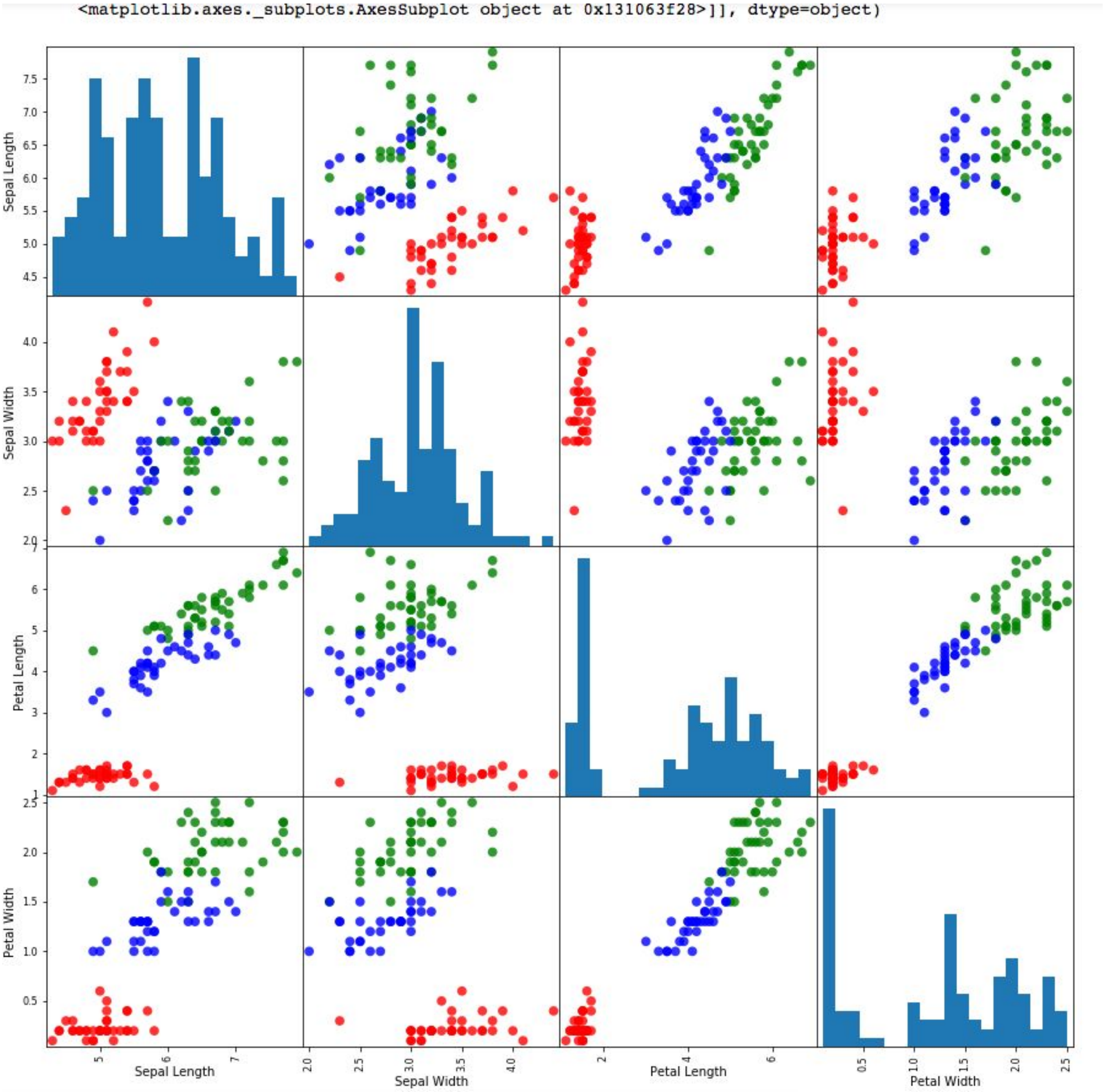
```
column_names = ["Sepal Length","Sepal Width","Petal Length","Petal Width"];
data = data_set.as_matrix(column_names)
target = data_set.as_matrix(["Class"])

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(data, target, random_state = 0)
```

Plot

In [143]:

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

iris_dataset = pd.DataFrame(X_train, columns=column_names)

colors = []
for e in y_train:
    if e=='Iris-setosa':
        colors.append('r')
    elif e=='Iris-versicolor':
        colors.append('b')
    elif e=='Iris-virginica':
        colors.append('g')

pd.plotting.scatter_matrix(iris_dataset, figsize=(15,15),
                           marker='o', hist_kwds={'bins': 20}, s=60,
                           alpha=.8, c=colors)
```

RESULTS:



```
<matplotlib.axes._subplots.AxesSubplot object at 0x131063f28>]], dtype=object)
```

CODE

Xialin

An Le

- END -