haberman_notebook

July 18, 2017

1 HABERMAN'S SURVIVAL DATA ANALYSIS

```
In [1]: # First, we'll import pandas, a data processing and CSV file I/O library
        import pandas as pd
        # We'll also import seaborn, a Python graphing library
        import seaborn as sns
        import matplotlib.pyplot as plt
        # Next, we'll load the Habermans's Survival dataset, which is in /
        #the current directory
        hman = pd.read_csv("haberman.csv")
        # Let's see what's in the Habermans's Survival data - Jupyter notebooks /
        #print the result of the last thing you do
        hman.head()
        # Press shift+enter to execute this cell
```

```
Out[1]:    Age  Year  Axillary                          Survived
        0   30    64         1  Patient survived 5 years or longer
        1   30    62         3  Patient survived 5 years or longer
        2   30    65         0  Patient survived 5 years or longer
        3   31    59         2  Patient survived 5 years or longer
        4   31    65         4  Patient survived 5 years or longer
```

```
In [2]: # (Q) how many data-points and featrues are there?
        print (hman.shape)
```

```
(306, 4)
```

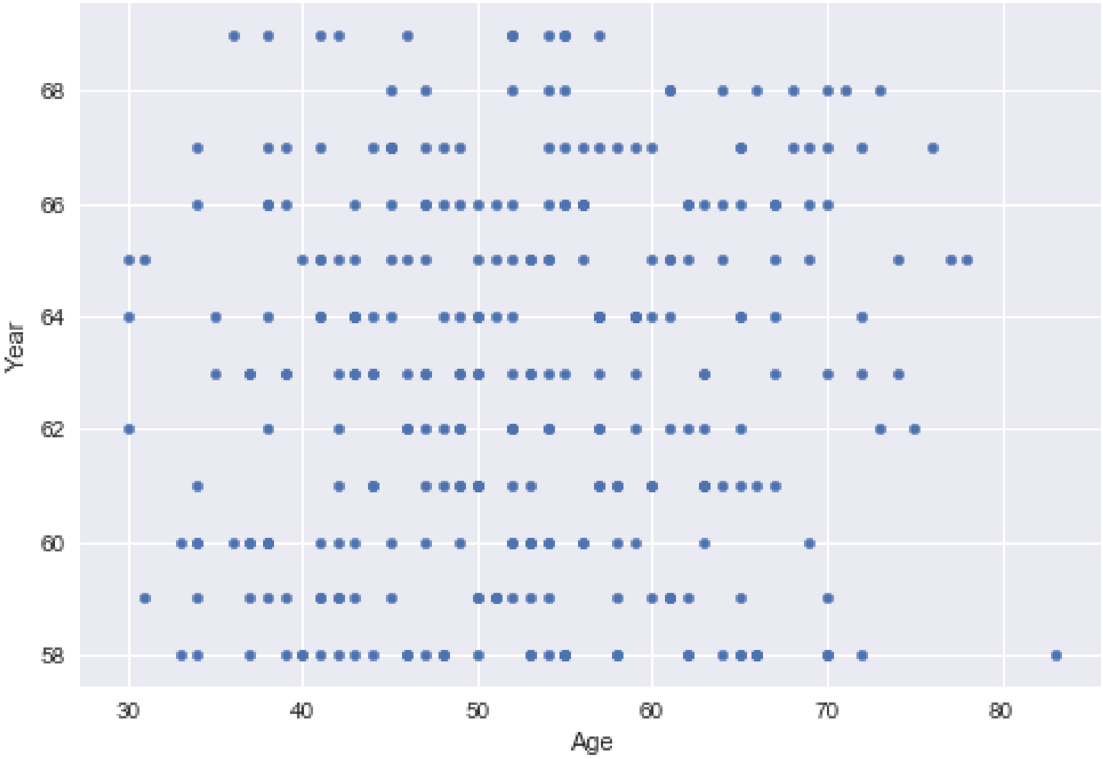
```
In [3]: #(Q) What are the column names in our dataset?
        print (hman.columns)
```

```
Index([u'Age', u'Year', u'Axillary', u'Survived'], dtype='object')
```

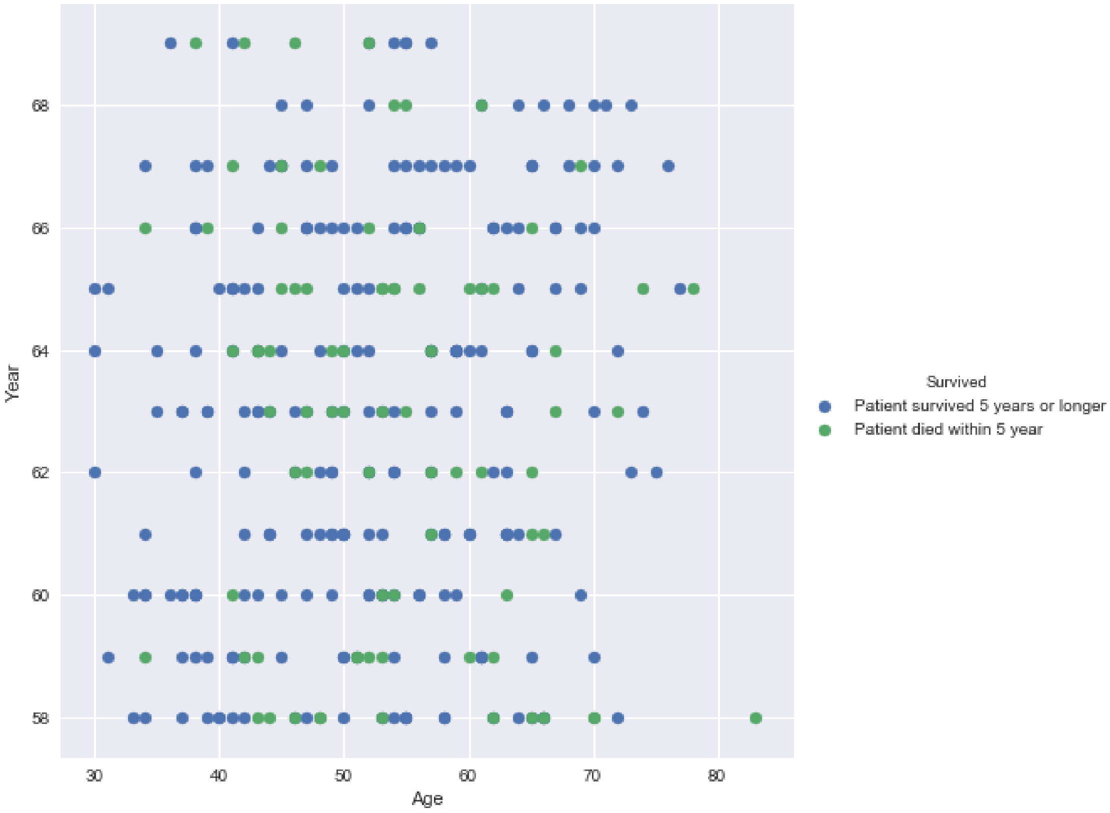
```
In [5]: # Let's see how many examples we have of each survival's.
        hman["Survived"].value_counts()
```

```
Out[5]: Patient survived 5 years or longer    225
        Patient died within 5 year             81
        Name: Survived, dtype: int64
```

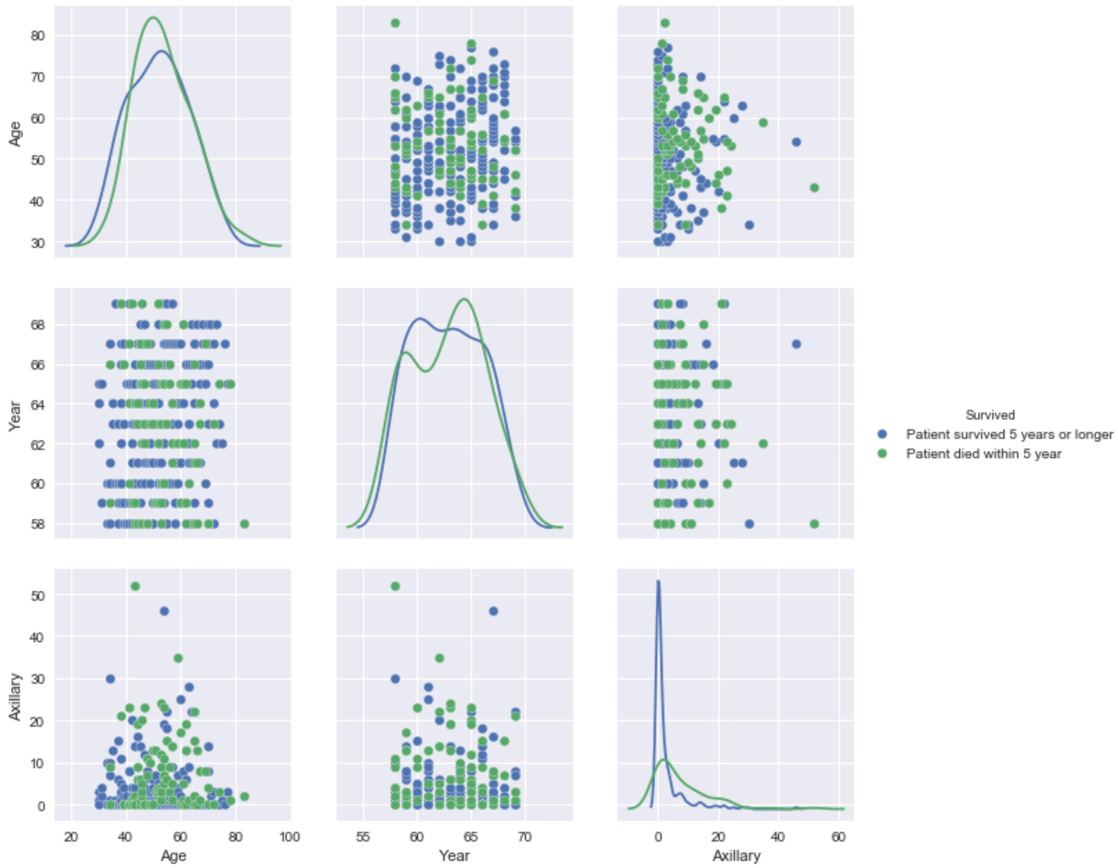
```
In [6]: # The first way we can plot things is using the .plot extension from Pandas /
        #dataframes
        # We'll use this to make a scatterplot of the Haberman's features.
        hman.plot(kind="scatter", x="Age", y="Year")
        plt.show()
```



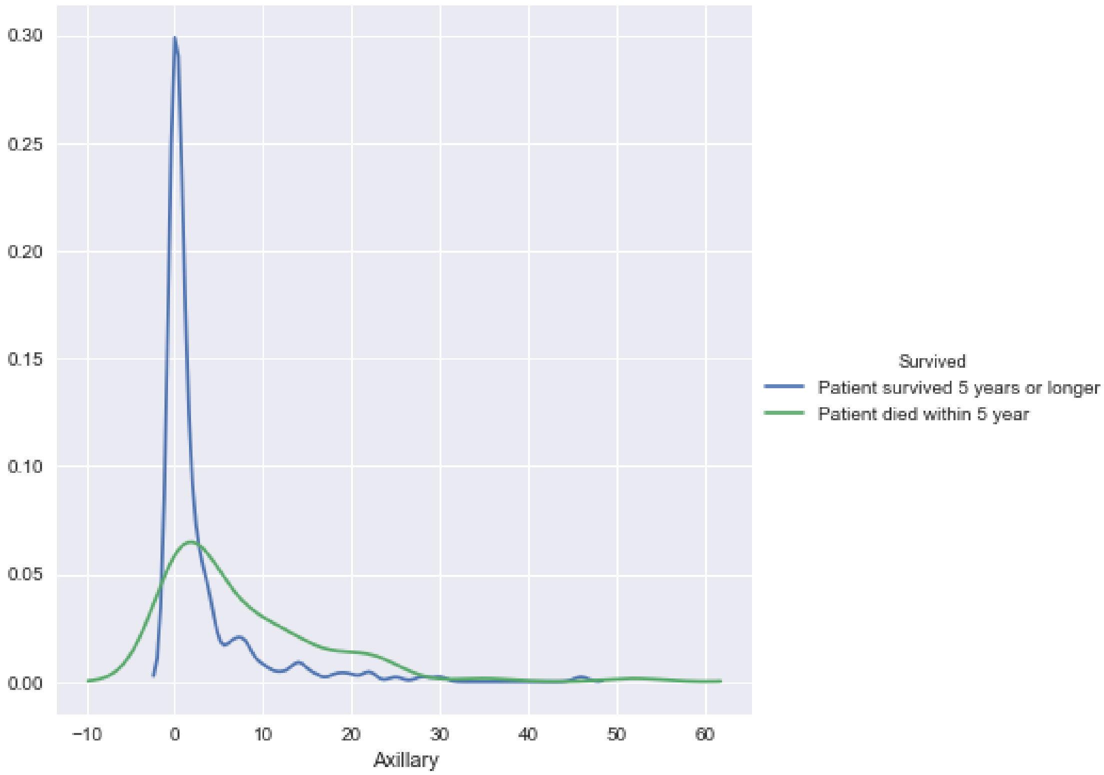
```
In [7]: # One piece of information missing in the plots above is if the /
        #patient Survived or not.
        # We'll use seaborn's FacetGrid to color the scatterplot by Survival.

        plt.close()
        sns.FacetGrid(hman, hue="Survived", size=7) \
           .map(plt.scatter, "Age", "Year") \
           .add_legend()
        plt.show()
```



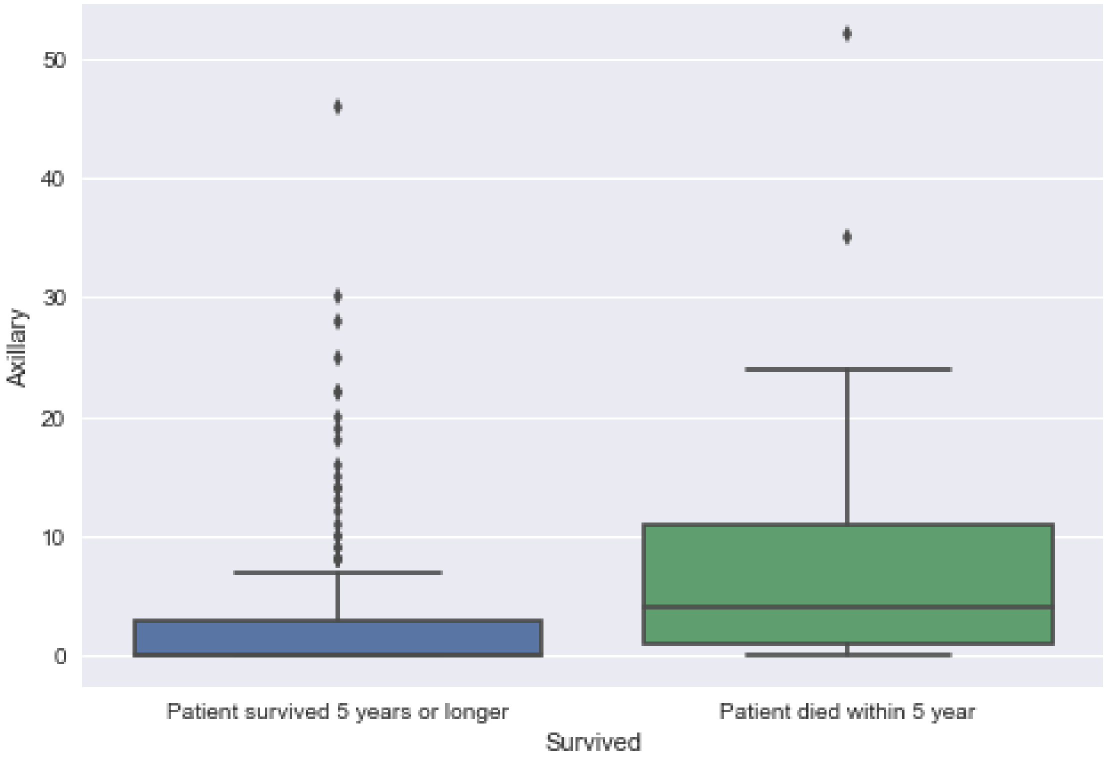
```
In [8]: # Another useful seaborn plot is the pairplot, which shows the /
        #bivariate relation between each pair of features

        plt.close()
        sns.pairplot(hman, hue="Survived", size=3, diag_kind="kde")
        plt.show()
```



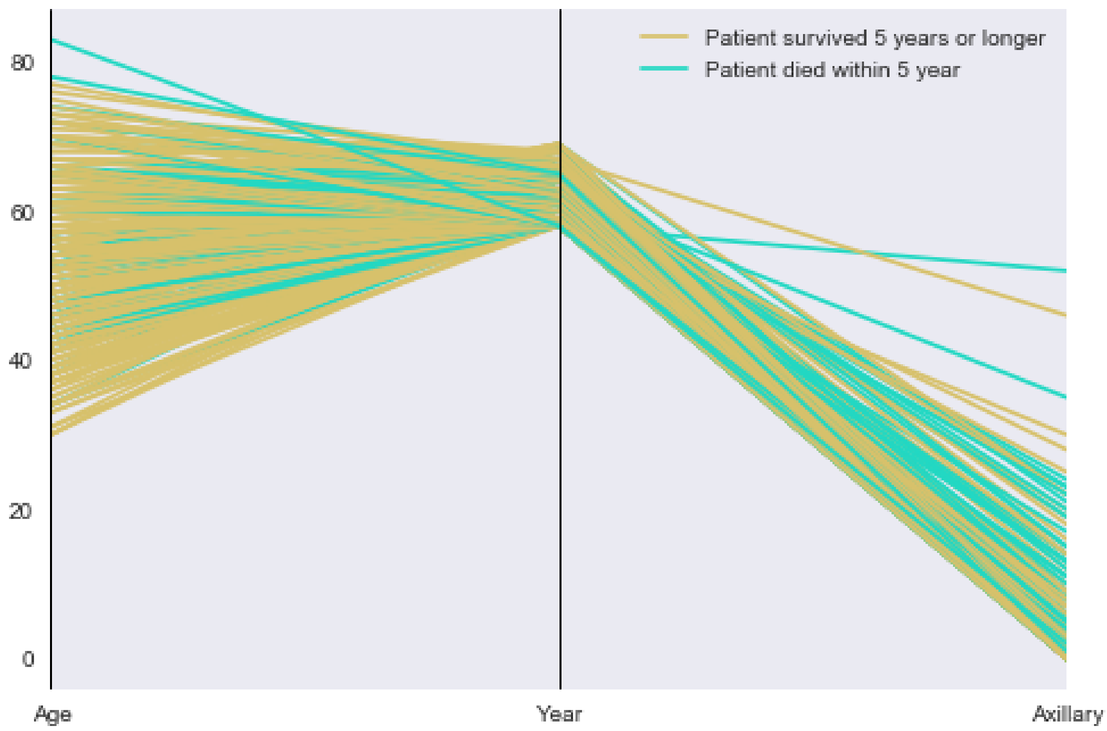
```
In [2]: # A seaborn plot useful for looking at univariate relations is the /
        #kdeplot, which creates and visualizes a kernel density estimate of /
        #the underlying feature
        plt.close()
        sns.FacetGrid(hman, hue="Survived", size=6) \
           .map(sns.kdeplot, "Axillary") \
           .add_legend()
        plt.show()
        #OBSERVATION: It is a long tail distribution, not a gaussian distribution
```



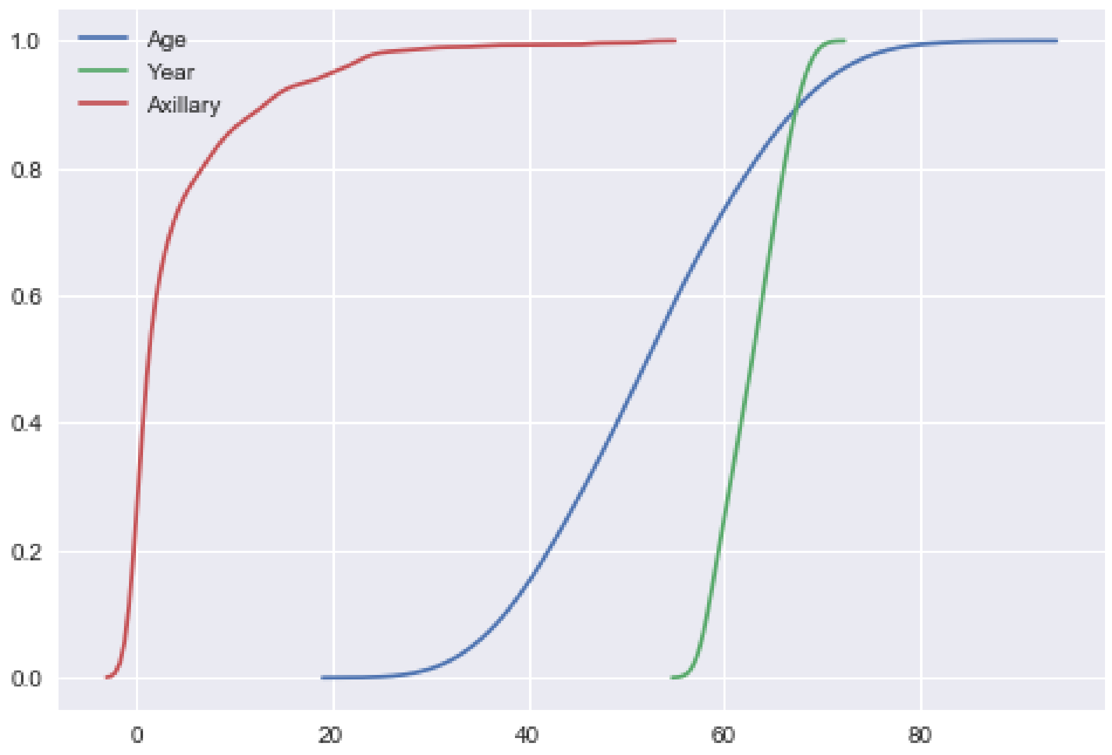
```
In [9]: # We can look at an individual feature in Seaborn through a boxplot
        plt.close()
        sns.boxplot(x='Survived',y='Axillary', data=hman, palette="deep")
        plt.show()
        #OBSERVATION: Patients with more than or equal to 5 Axillary Nodes /
        #are more likely to die within 5 years.
```



```
In [3]: # Another multivariate visualization technique pandas has is /
        #parallel_coordinates
        # Parallel coordinates plots each feature on a separate column & /
        #then draws lines connecting the features for each data sample

        from pandas.plotting import parallel_coordinates
        parallel_coordinates(hman, "Survived");
        plt.show();
```



```
In [4]: ax = sns.kdeplot(hman['Age'], cumulative=True)
        ax = sns.kdeplot(hman['Year'], cumulative=True)
        ax = sns.kdeplot(hman['Axillary'], cumulative=True)
        plt.show()
```



2 # # Summary:

From the given Haberman's data we have made above graphical plots. We can conclude that the feature 'Axillary', can be used in order to tell the life expectancy of a patient after the operation. From the analysis done using Haberman's data, I can conclude that the patients with more than or equal to 5 Axillary nodes are more likely to die within 5 years.