**Data project - Election Analysis**

In this Data Project we will be looking at data from the 2012 election. The first data set are the results of political polls. Analysing this data, we answer the following questions:

1.) Who was being polled and what was their party affiliation?  
2.) Did the poll results favor Romney or Obama?  
3.) How do undecided voters effect the poll?  
4.) Can we account for the undecided voters?  
5.) How did voter sentiment change over time?  
6.) Can we see an effect in the polls from the debates?

* We first import the standard pandas, numpy, Series and Dataframe libraries. Then, for data visualization, we import matplotlib.pyplot, seaborn and set the style to ‘whitegrid’ using sns.set\_style(). And to display it in IPython notebook, we use the command %matplotlib inline.
* To import data from the web, we use requests modules. It is a really nice API to gather information from the web. StringIO provides a convenient means of working with text in memory using the file API. To work with csv data, we use StringIO.
* Using http capabilities, we grab data from the web. We import the requests library.
* To use StringIO to work with the csv file, we import StringIO from StringIO library.
* We use the url link for the poll data in csv form and store it in an object called ‘url’
* We also use requests to get the information from the web as a text form using the command: requests.get(url).text
* To avoid IO error with pandas, use StringIO.
* Now, we set the data as a dataframe.
* To preview the dataframe, we use the .head() method.
* To answer the first question and get an overview of the affiliation for the polls, we use data visualization.
* We use factorplot to plot the different affliations: none, republican, democratic, other.
* Then we factorplot the affliations by grouping by the population. We use hue as population.
* This will answer the first question. To answer the second, we find the averages for Obama, Romney, and the polled undecided people.
* If the plots look too close together, it will be difficult to determine which is higher, so we use number. To Display the average polls and the standard deviation, we concatenate both, and then display the concatenated object.
* Now to determine how the undecided voters affect the poll.If we assume we split the undecided evenly between the two candidates the observed difference should be an unbiased estimate of the final difference. This is to account for the undecided voters, which answers the fourth question.
* We can do a quick time series analysis of the voter sentiment by plotting Obama/Romney favor versus the Poll End Dates. The time in this analysis is in reverse chronological order.
* To plot out the difference between Obama and Romney and how it changes as time moves along, we use the datetime module to create timestamps.
* To take into account the difference between Romney and Obama in the polls, define a new column in our DataFrame.
* A positive difference indicates a leaning towards Obama in the polls as the difference column is Obama-Romney.
* To notice how the sentiment in difference changes over time, we will use groupby to group the polls by their start data and then sort it by that Start Date.
* Now we get a straight forward plot between the Difference versus time.
* To answer the final question of whether we can see an effect in the polls from the debates, we can plot marker lines on the dates of the debates and see if there is any general insight to the poll results.
* If all the debates dates were in October, we use a simple for loop to find the x limits for the figure, to find out where the index for the month of October in 2012 is.
* After knowing where to set our x limits for the month of October, we plot the original figure, and then add markers to mark the debate dates.
* Thus, we have analysed all the questions that we have sought after.

The second dataset consisting of information on donations to the federal campaign.The questions we will be trying to answer while looking at this Data Set is:

1.) How much was donated and what was the average donation?  
2.) How did the donations differ between candidates?  
3.) How did the donations differ between Democrats and Republicans?  
4.) What were the demographics of the donors?  
5.) Is there a pattern to donation amounts?

* After importing the usual libraries and modules, the dataset, which is stored as a text file can be read by using the pd.read\_csv() method. We also store store this as a dataframe.
* To get an overview of the data, we use .info() method. To answer the first question, to look at the various donation amounts, we use the .value\_counts() method. To display the average, and the standard deviation, we use the .mean() and .std() function.
* Now, we make a series out of the dataframe using the .copy() method to avoid view errors. We can also sort it using .sort() method.
* If there are negative values, to get rid of those, we use the command:

<series name> = <series name>[<series name> >0]

* Now, after we sort the series, it will display the top common donations value counts.
* To check if the donations are usually made in round number amounts, we can make a histogram and check for peaks at those values. If there are spikes at round numbers, it means the donations are usually madein round number amounts.
* To separate donations by Party, we create a new a new 'Party' column. We do this by getting a list of candidates and defining a dictionary of part affliatons for each candidate. And then we map the party affliations dictionary with the dataframe.
* We also clear refunds in the contribution amounts in the dataframes.
* Now, to answer the second question, we find the total amounts received by each candidate. First, we find the total number of donations then the total amount for each candidate.
* If the values are not too clear, we can print out the values in a for loop for better clarity. We can also do a graphic presentation of the data to get a clear idea and to find the answer.
* To answer the third question, we compare Democrats versus the Republican donations.
* To find the demographics of the donors, we look at donations and who they came from. We first get the occupation information from the DataFrame and then, use a pivot\_table to make the index defined by the various occupations and then have the columns defined by the Party (Republican or Democrat).
* To add up all the contributions by anyone with the same profession, we'll pass an aggregation function in the pivot table.
* If the dataframe is too large to display as an effective visualization, we create a cut-off for total contribution amounts. Now we can effectively visualize the cut off dataframe using pandas’s .plot() method.
* In case of difficulty to read, we can also switch the axis of the plot.
* To get rid of the mislabelled or unavailable occupations, we use the .drop() method. And to combine similar rows, we use the .loc and .drop() method.
* The new plot will give us the proper pattern of donation amounts.