Python Homework 3: Report

Aman Shrivastava(as3ek)

Introduction

Superheroes have dominated pop culture for a long time now. Across different comic book universes, superheros have a plethora of different strengths and weaknesses. They have different skill statistics across categories like intelligence, strength, power, combat, speed and durability. This project aims to provide an overview about heroes and their physical as well as power characteristics, helping us identify trends and patterns.

Data Source

The data is scraped from the website https://www.superherodb.com by extracting the list of heroes from the /characters page and detailed statistics extracted from individual details page of each character.

Content

The dataset extracted has a total of 743 characters and each character has the following 27 attributes :

- Name
- Url
- Intelligence
- Strength
- Speed
- Durability
- Power
- Combat
- Full name
- Alter Egos
- Aliases
- Place of birth
- First appearance
- Creator
- Alignment
- Gender
- Race
- Height
- Weight
- Eye color
- Hair color
- Occupation
- Base
- Team Affiliation
- Relatives
- Skin color
- Total Power

Libraries Used

- Pandas
  For importing data into dataframes.

- Requests
  For making http requests to web pages to be crawled through the script.

- BeautifulSoup
  For parsing and navigating the source code of the web page to be scrapped.

- Matplotlib
  For making plots and pie charts during data visualization.

- Jupyter notebook
  For displaying the results of data analysis and visualizations made.

Approach

Step 1. Extracting names and urls to their respective details page
Using the requests library sent a request the the base url (https://www.superherodb.com/characters/) to extract the source code of the web page with list of all available characters on the website. After converting the source code to a soup object, extracted the names and url to their respective details page of the characters. Stored the collected data will all the names and personal urls to a dataframe.

Step 2. Extracting statistics and info from the individual details page
Iterated over the dataframe created in step 1. to visit the page for each character which contained information and detailed statistics. For each page used the soup object obtained to find and extract relevant information about the character across all the parameters mentioned in the content section. For every loop, appended the collected data to a dataframe.

Step 3. Creating a master dataframe and writing it to a csv file
Merged the dataframes generated from step 1 and step 2 to create a master dataframe of all the collected information about the characters across the parameters mentioned in the content section. Wrote this dataframe to a csv file for storage.

Step 4. Reading the csv file and data cleaning

Read in the data from the csv file generated in the above step to a dataframe to perform further analysis. Changed the data type of the columns to match the type of data contained in them. Took care of missing values by replacing them with relevant data.

Step 5. Generating new parameters from existing information

Used the available character power statistics to create a new column called Total Power to represent the cumulative strength of the character and used it to find the Top 10 most powerful superheroes in the dataset.

Extra Credit

Step 6. Display details about a character selected by the user

Takes the name of any character as an input from the user and displayed the characteristics and all details about that character

Step 7. Battle two user selected characters and predict winner

Takes the name of two different characters as input and display the more proficient character across all fighting statistics and predict the overall winner of the fight.

Step 8. Generating visualizations and further analysis

Used jupyter notebook (.ipynb) to extract interesting insights and visualize the data further. Have submitted the notebook for consideration and have attached the generated pdf of the visualizations at the end of this report.

Conclusion

The script extracts an exhaustive dataset of all superhero characters, presents some interesting insights in form of visualizations, displays details of requested character and simulates a battle between two user selected characters.

Scope

Apart from these functionalities, the dataset can be used to do more advanced analyses of comic book characters as the dataset contains various other attributes—about the characters' personal traits and history—that can be explored, like race, eye color, height, weight, base, alter egos, team affiliations etc.

Super Hero Data Analysis and Visualization

July 28, 2018

In [1]:
```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.plotly as py1
import plotly.offline as py
py.init_notebook_mode(connected=True)
from plotly.offline import init_notebook_mode, iplot
init_notebook_mode(connected=True)
import plotly.graph_objs as go
import plotly.offline as offline
offline.init_notebook_mode()
from plotly import tools
import plotly.graph_objs as go

pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
```

0.0.1 Reading in and cleaning data

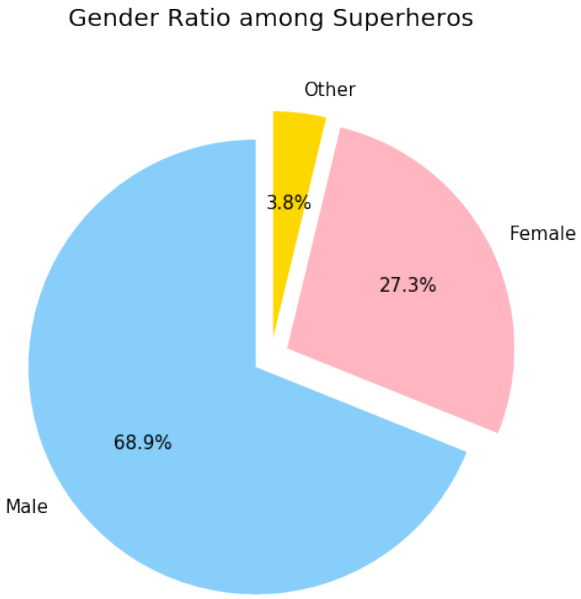
In [2]:
```
data = pd.read_csv('SuperheroDataset.csv')
data.replace(to_replace='-', value='Other', inplace=True)
data['Creator'].fillna('Other', inplace=True)
```

0.0.2 Finding out and visualising gender ratio of all collected superheros

In [3]:
```
hero_g = data.Gender.value_counts()
```

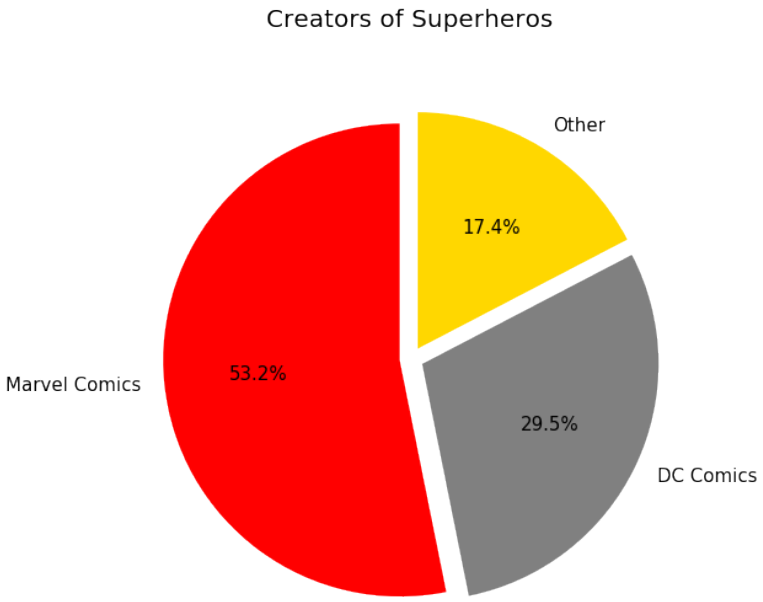
In [4]:
```
plt.figure(figsize=(16,8))
plt.title('Gender Ratio among Superheros', fontsize=20, y=1.1,)
labels = 'Male', 'Female', 'Other'
colors = ['lightskyblue', 'lightpink', 'gold']
explode=(0.08, 0.08, 0.08)
plt.rcParams['font.size'] = 15.0
plt.pie(hero_g.values, colors=colors,
        explode=explode, labels=labels,
        autopct='%1.1f%%', startangle=90)
```

```
plt.axis('equal')
plt.show()
```



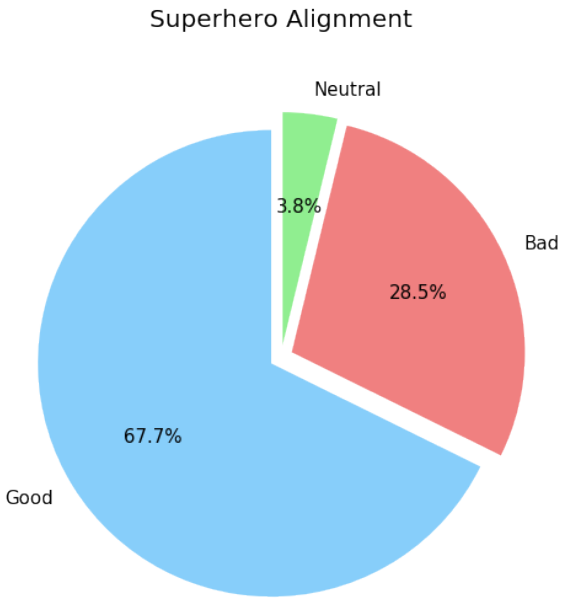
0.0.3 Visualizing Publishers

```
In [5]: hero_p = data.Creator.value_counts()
        other = hero_p.values[2:].sum()
        hero_p = hero_p[:2]
        hero_p['Other'] = other

In [6]: plt.figure(figsize=(16,8))
        plt.title('Creators of Superheros', fontsize=20, y=1.1,)
        labels = hero_p.index
        colors = ['red', 'gray', 'gold']
        explode=(0.05, 0.05, 0.05)
        plt.rcParams['font.size'] = 15.0
        plt.pie(hero_p.values, colors=colors,
                explode=explode, labels=labels,
                autopct='%1.1f%%', startangle=90)
        plt.axis('equal')
        plt.show()
```



0.0.4 Visualising Superhero Alignments

```
In [7]: hero_align = data.Alignment.value_counts()
        hero_align = hero_align[:3]

In [8]: plt.figure(figsize=(16,8))
        plt.title('Superhero Alignment', fontsize=20, y=1.1,)
        labels = ['Good', 'Bad', 'Neutral']
        colors = ['lightskyblue', 'lightcoral', 'lightgreen']
        explode=(0.05, 0.05, 0.05)
        plt.rcParams['font.size'] = 15.0
        plt.pie(hero_align.values, colors=colors,
                explode=explode, labels=labels,
                autopct='%1.1f%%', startangle=90)
        plt.axis('equal')
        plt.show()
```



0.0.5 Finding the most powerful superheroes in each universe

```
In [9]: data_marvel = data.loc[data['Creator'] == 'Marvel Comics']
        data_dc = data.loc[data['Creator'] == 'DC Comics']
        data_marvel = data_marvel.sort_values('Total Power', ascending=False)
        data_dc = data_dc.sort_values('Total Power', ascending=False)
        top_10_dc = data_dc[:10]
        top_10_marvel = data_marvel[:10]

In [10]: top_10_dc[['Name', 'Total Power']]

Out[10]:                    Name  Total Power
         668         The Presence        600.0
         275          General Zod        595.0
         471              Monarch        590.0
         653             Superman        585.0
         292      Granny Goodness        585.0
         651       Superboy-Prime        585.0
         646          Steppenwolf        585.0
         416  Lucifer Morningstar        580.0
         652            Supergirl        575.0
         529           Power Girl        575.0

In [11]: top_10_marvel[['Name', 'Total Power']]

Out[11]:                  Name  Total Power
         506     One-Above-All        600.0
```

```
87              Binary        595.0
80            Beyonder        585.0
670               Thor        570.0
522            Phoenix        565.0
161  Captain Universe        565.0
640           Stardust        565.0
341           Hyperion        560.0
231           Dormammu        555.0
335               Hulk        545.0
```

0.0.6 Comparing total strength of DC vs Marvel characters

```
In [12]: strength_dc = data_dc['Total Power'].sum()/hero_p['DC Comics']
         strength_marvel = data_marvel['Total Power'].sum()/hero_p['Marvel Comics']

In [13]: df=pd.DataFrame({'Universe':['Marvel', 'DC'],
                          'Strength':[strength_marvel, strength_dc]})
         fig = plt.figure(figsize=(12,7))
         fig.add_subplot(1,1,1)
         sns.barplot(x='Universe',y='Strength',data=df)
         plt.xticks(rotation=0)
         plt.show()
```