COMP20008 Elements of Data Processing
Project 1

March 25, 2020

Due date

The assignment is worth 25 marks, (25% of subject grade) **and is due 8:00am 27 April 2020 Australia/Melbourne time**.

Background

A web server has been setup at *http://comp20008-jh.eng.unimelb.edu.au:9889/main/* containing a number of media reports on Tennis matches between 2004 and 2005. As data scientists, we would like to extract information from those reports and use that information to improve our understanding of player performance. A JSON data file containing player names and key statistics is also available here.

Tennis scores

Understanding the tennis scoring system is important in order to be able to extract scores from match reports. The tennis scoring system is relatively complicated. A tennis match is divided into either 3 or 5 sets, with a player needing to win 2/3 or 3/5 sets to win the match. A set is won by the first player to win 6 games, with a margin of at least 2 games over the other player (e.g. 6–3 or 7–5). If the set is tied at 6 games each, a tie-break is played to decide the set. If a set is decided with a tie-break, the score in the tie break is typically reported in brackets next to the score for the set. For example, any of the following are valid final scores for a tennis match:

```
6-0 6-0
6-3 2-6 7-5
7-6 (7-5) 7-6 (7-4)
6-3 2-6 7-5 1-6 6-3
```

You can read more about the tennis scoring system on Wikipedia

Learning outcome

The learning objectives of this assignment are:

- To gain practical experience in written communication skills for data science projects.
- To practice a selection of processing and exploratory analysis techniques through visualisation discussed in lectures and workshops.
- To practice crawling and scraping data from the Internet.
- To practice using widely used Python library for data processing and gain experience using library functions which may be unfamiliar and which require consultation of additional documentation from resources on the Web.

Your tasks

You are to perform a small data science project including some data processing and analysis using Python. Your responses to Tasks 1-5 must be contained in a single **.py** file. Specifically, you have the following tasks:

Task 1 *(2 marks)*

Crawl the *http://comp20008-jh.eng.unimelb.edu.au:9889/main/* website to find a complete list of articles available.

Produce a csv file containing the URL and headline of each the articles your crawler has found found. The CSV file should have two column headings **url** and **headline** and be called **task1.csv**.

Note: You might want to start with a smaller website to test your crawling implementation with this site (*http://comp20008-jh.eng.unimelb.edu.au:9889/sample/*).

Task 2 *(4 marks)*

For each article found in Task 1,

a) extract the name of the first player mentioned in the article. You can find a list of player names as part of the **tennis.json** file provided. We will assume the article is written about that player (and only that player). **(2 marks)**

b) extract the first complete match score identified in the article. You will need to use regular expressions to accomplish this. We will assume this score relates to the first named player in the article. **(2 marks)**

Produce a csv file containing the URL, headline, first player mentioned and first complete match score of each the articles your crawler has found. The csv file should have four column headings **url**, **headline**, **player** and **score** and be called **task2.csv**.

Note: Some articles may not contain a player name and/or a match score. These articles can be discarded.

Task 3 *(2 marks)*

For each article used in Task 2, identify the absolute value of the game difference. E.g. a 6-2 6-2 score has a game difference of 8, while a 6-4 4-6 6-4 score has a game difference of 2. The value is referred to as the game_difference

Produce a csv file containing the player name and *average* game_difference for each player that at least one article has been written about. The csv file should have two column headings **player** and **avg_game_difference** and be called **task3.csv**.

Task 4 *(2 marks)*

Generate a suitable plot showing five players that articles are most frequently written about and the number of times an article is written about that player.

Save this plot as a png file called **task4.png**

Task 5 *(2 marks)*

Generate a suitable plot showing the average game difference for each player that at least one article has been written about and their win percentage. You can find a player's win percentage in the **tennis.json** file.

Save this plot as a png file called **task5.png**

Task 6 *(13 marks)*

Write a 3-4 page report to communicate the process and activities undertaken in the project, the analysis, and some limitations. Specifically, the report should contain the following information:

- A description of the crawling method and a brief summary the output for Task 1. **(2 marks)**
- A description of how you scraped data from each page, including any regular expressions used for Task 2 and a brief summary of the output. **(3 marks)**
- An analysis of the information shown in the two plots produced for Tasks 4 & 5, including a brief summary of the data used. The plots are to be shown (included) along with your analysis. **(4 marks)**
- A discussion of the appropriateness of associating the first named player in the article with the first match score. **(2 marks)**
- At least one suggested method for how you could figure out from the contents of the article whether the first named player won or lost the match being reported on. **(1 mark)**
- A discussion of what other information could be extracted from the articles to better understand player performance and a brief suggestion for how this could be done. **(1 mark)**

Submission instructions

Your responses to Tasks 1 - 5 must be contained in a single python script (**.py**) file. As the output of this file will be verified automatically, it is essential that the program runs without producing errors. Submission is via the LMS. Two submission links will be provided, one for the **.py** file containing your responses to Tasks 1 - 5 and a second for a **.pdf** or **.docx** file containing your response to Task 6.

Please ensure you get submission receipts via email. If you don't receive a receipt via email, this means your submission has not been received and hence cannot be marked. Late penalty structure is described at the end of this document.

Extensions and late submission penalties

If requesting an extension due to illness, please submit a medical certificate to the lecturer. If there are any other exceptional circumstances, please contact the lecturer with plenty of notice. Late submissions without an approved extension will attract the following penalties

- $0 < hourslate <= 24$ (2 marks deduction)
- $24 < hourslate <= 48$ (4 marks deduction)
- $48 < hourslate <= 72$: (6 marks deduction)
- $72 < hourslate <= 96$: (8 marks deduction)
- $96 < hourslate <= 120$: (10 marks deduction)
- $120 < hourslate <= 144$: (12 marks deduction)
- $144 < hourslate$: (20 marks deduction)

where *hourslate* is the elapsed time in hours (or fractions of hours).

This project is expected to require 30-35 hours work.

Academic honesty

You are expected to follow the academic honesty guidelines on the University website
`https://academichonesty.unimelb.edu.au`

Further information

A project discussion forum has also been created on the Ed forum. Please use this in the first instance if you have questions, since it will allow discussion and responses to be seen by everyone. There will also be a list of frequently asked questions on the project page.