SCC.022 Making Sense of Data

Studio Worksheet: Week 2

In this week's studio, we will cover aspects from the first week of the module and build on the skills learnt in last week's studio. We will look at some data cleaning approaches and implement them in Python using the Pandas Data Analysis library.

This lab session will focus on practical implementation using Python. The preparation detailed below should be completed before the lab session.

Preparation

Please complete the following preparation items before the lab session.

It is assumed that you already have Python installed from the last session. If not, you can install Python from AppsAnywhere, or via the Python website if you have problems with AppsAnywhere.

Step 1: Create a directory for this session

We will need a new home directory (or folder) to house all of our files for this studio. In this worksheet, we will assume that you are using the address "H:\SCC022\Wk2". Please keep in mind that your home directory address might be different so, when you see this, you should substitute it for your home folder address.

Step 2: Download the week2studio.zip file

This can be downloaded from the Moodle module page. This contains the files needed for the rest of this worksheet. Save it to your home directory and extract the archive so the files are accessible (right click the file and select "Extract all..." or similar). Move the files to your home directory. This should include the files "requirements.txt" and "WNBA_ALL.csv"

Step 3: Install Pandas and the supporting libraries

We will use "pip" to install these:

1. Launch the command prompt:
    a. If you are using Windows: Open Windows Powershell from the start menu.

b. If you are using Mac: Open Terminal.app
c. If you are using Ubuntu: Open the Terminal

2. Go to the folder where the worksheet files are saved by typing “cd H:\SCC022\Wk2”
   NB: If there are spaces in your filename, you may need to use speech marks in your command. E.g. cd “H:\SCC 022\Wk2”
3. Install the required libraries by running “pip install -r requirements.txt” This will install the Pandas, Scipy and Matplotlib libraries.

Task 1: Data Statistics and Visualisation

In this task, we will take a real data file and try to understand the relationships between the variables to drawer a deeper understanding. This will introduce you to a several new functions in Pandas that are particularly useful in data science. We will use the WNBA dataset of Women’s National Basketball Association players for this task.

Step 1: Run IDLE

We will be using the interactive window in IDLE to run our program step-by-step as we learn the steps. If you want to run it all automatically later, you can write it into a new file window in IDLE and save and run it like other Python programs. If you are doing so, remember to move all the import lines to the top of the script file.

Step 2: Import the libraries

First, we need to import the needed libraries into the program, so type these separate lines into IDLE's interactive window (the one with the >>> prompt):

```
>>> import pandas as pd
>>> import matplotlib as plt
>>> import datetime as dt
>>> import numpy as np
```

Tell the plotting library **matplotlib** to pop-up any graph windows and continue on with the program instead of waiting until the windows has closed to continue. This makes it easier to interactively experiment with.

```
>>> plt.interactive(True)
```

NB: you will not see any feedback from this command.

Step 3: Load the CSV file.

Store the home directory address as a string variable and load the CSV file into a Pandas DataFrame using the read_csv function. We can then use that variable to access all the values in the file.

```
>>> hdir = "H:\\scc022\\Wk2\\"
>>> wnba = pd.read_csv(hdir + "WNBA_ALL.csv")
```

NB: If there are any backslash characters in the path then we need to type them twice to escape them so they are stored in the Python string properly - if you forget to do this you will get an error. Note also that we have used the "+" symbol to join together two strings.

The CSV file should now load without error. This will be stored in a DataFrame called wnba. Recall that we can check this by typing the variable name into the command window and it will show us some of the first and last rows and columns. It is a good idea to do this after each command below so you can see the effects.

```
>>> wnba
```

Step 4: Data Analysis

Visualisation. Your first task is to visualise the data and plot the distributions for each of the following characteristics:

- Field goal percentage (FG%)

- Total rebounds (TRB)
- Steals (STL)
- Blocks (BLK)
- Assists (AST)
- Total points (PTS)

What kind of distributions do you see?

HINT: You might find wnba.hist() useful for this task. If your figure does not appear, call plt.show()

Descriptive Statistics. You are now asked to describe the above characteristics in a report using descriptive statistics along with your plots. For each characteristic, select appropriate measures of central tendency and variability.

HINT: Recall from last week that wnba.describe() can help you to obtain the statistics that you need.

Working with Subsets. You are now asked to identify areas to focus on in training. For each of the five positions below, calculate descriptive statistics and obtain visualisations for the field goal percentage, steals, blocks and assists:

- Forward-center
- Forward-guard
- Center
- Guard
- Forward

HINT: This is similar to the above tasks but with different subsets of the data. In order to create these subsets, use the command where wnba_subset is your subset, wnba is your dataframe, colname is the name of the column you want to filter for, and x is the value to filter.

```
>>> wnba_subset = wnba[wnba['colname'] == x]
```

Correlation. Plot the Pearson correlation metrics. Determine whether any correlations are visible. If so, do these make sense and do you think causality exists?

HINT: the command wnba.corr() will give you a 2 x 2 matrix with the Pearson correlation coefficient for the corresponding two variables.

Scatter Plots. Investigate the pairings with high correlation scores further (whether they are positive or negative) using scatter plots. Take a look at the pairings with close to zero correlation. Do you notice any other type of relationship?