Pandas Worksheet

February 22, 2022

Before you attempt these problems, please ensure that you have worked through the **NumPy worksheet** from Week 4 and have read and understood the **pandas Notes** on the course website. Throughout, you may need to consult the pandas online documentation available at https://pandas.pydata.org.

1. pandas Series arithmetic operations

Create a pandas Series, `s`, with the indices A to E and data values 2^n for $n = 0, \ldots, 4$. Compute the following quantities:

(i) `s + s`;

(ii) `s` squared;

(iii) exponential of `s`;

(iv) the first four entries of `s` added to the last four entries of `s`.

(Question (iv) demonstrates that operations between Series automatically align the data based on *label*. Per the pandas documentation:

"The result of an operation between unaligned Series will have the union of the indexes involved. If a label is not found in one Series or the other, the result will be marked as missing NaN. Being able to write code without doing any explicit data alignment grants immense freedom and flexibility in interactive data analysis and research. The integrated data alignment features of the pandas data structures set pandas apart from the majority of related tools for working with labeled data."

(v) Now change the data entry with the index C to a string (a word) of your choosing. Repeat (i) and (ii).

2. Create a DataFrame from a list of Python dictionaries

Create the following pandas DataFrame, `d`, from a list of Python dictionaries.

Using the `.mean()` function in pandas, calculate the average score for each person.

[36]:

```
d
```

[36]:

```
        Sara  Jeff  Phil  Lauren  Emma  Harriet
Test B  95.0  78.0  65.0     NaN   NaN      NaN
Test A   NaN   NaN   NaN    45.0  89.0     55.0
Test C  85.0  45.0  39.0    75.0  66.0     54.0
```

3. Basic DataFrame operations: column selection, addition, and deletion

Create the following DataFrame:

[54]:
```
df
```

[54]:
```
             A    B
apple      6.0  NaN
banana     2.0  9.0
kiwi       NaN  8.0
orange     9.0  5.0
pineapple  NaN  7.0
```

(i) Using the command starting `df['C'] = ...`, add a third column to `df` that is the sum of the first two columns.

(ii) Add another column to `df` named T/F that indicates whether the entries in column C are greater than 6.

(iii) Using the function `.pop()`, remove column B.

(iv) Using the function `.insert()`, insert a copy of column C that is the new first column of `df`.

(v) Finally, add a column in any position composed of the first three entries in column A.

4. **Descriptive statistics**

For this question, you will need to download the `NBA_Stats.csv` file from the course website. Throughout, you may find it useful to query the data types contained within each column of the database.

[55]:
```
NBA = pd.read_csv('./NBA_Stats.csv', sep = ',')
```

(i) Find the minimum, maximum, mean, and standard deviation of `player_height` of the players listed in the database.

Using the functions `.idxmax()` and `idxmin()` who is (are) the tallest/shortest player(s) in the database? What is the difference in their heights?

(ii) Create a new DataFrame containing the columns `player_height`, offensive rebound percentage (`oreb_pct`), and defensive rebound percentage (`dreb_pct`).

Using the function `.corr()`, analyze whether there exists a correlation between each of these three quantites for the players listed in the DataFrame.

(iii) Write a function that takes as input a *year* (`draft_year`) and *season* (`season`) and returns the player who was drafted in the year 2017 and scored the most points per game (`pts`) in the season 2018-2019?

(**Hint**: be careful - the data types for the columns `draft_year` and `season` are *not* integers or floats.)

(iv) How many colleges (`college`) are listed in the database? (*Hint*: you may find the `.value_counts()` function useful.)

Which college has had the most players drafted (`draft_round`) in the first round? In the top 10?

For the college with the most players drafted in the first round, what is the average `net_rating` of these players and how many players having a `net_rating` above the average?

How does this average rating compare to the average rating of all other colleges combined?

[ ]:

```
```