In-class assignments for Section 1

Subhasis Ray*

Startup

1. Open your terminal, activate the `dsai` environment in Anaconda:

   ```
   conda activate dsai
   ```

2. Create a directory called W03P for this class, and change to this directory

   ```
   mkdir W03P
   cd W03P
   ```

3. Download the archive `W03P1_data.zip` from LMS to your local computer This is originally from Kaggle, uploaded by Ruchi Bhatia: `https://www.kaggle.com/datasets/ruchi798/data-science-job-salaries`. After downloading, unzip the archive in the `W03P1` folder. This should create a file named `ds_salaries.csv`.

4. Open the data using your spreadsheet software. Specify comma (,) as the separator. Here is the description of the columns from Kaggle. Read it carefully.

*subhasis.ray@plaksha.edu.in

Column	Description
work_year	The year the salary was paid.
experience_level	The experience level in the job during the year with the following possible values: EN Entry-level / Junior MI Mid-level / Intermediate SE Senior-level / Expert EX Executive-level / Director
employment_type	The type of employement for the role: PT Part-time FT Full-time CT Contract FL Freelance
job_title	The role worked in during the year.
salary	The total gross salary amount paid.
salary_currency	The currency of the salary paid as an ISO 4217 currency code.
salary_in_usd	The salary in USD (FX rate divided by avg. USD rate for the respective year via fxdata.foorilla.com).
employee_residence	Employee's primary country of residence in during the work year as an ISO 3166 country code.
remote_ratio	The overall amount of work done remotely, possible values are as follows: 0 No remote work (less than 20%) 50 Partially remote 100 Fully remote (more than 80%)
company_location	The country of the employer's main office or contracting branch as an ISO 3166 country code.
company_size	The average number of people that worked for the company during the year: S less than 50 employees (small) M 50 to 250 employees (medium) L more than 250 employees (large)

5. Now start Spyder from your terminal.

6. Use Spyder to create a Python file on your computer, save it as `W03P1_classwork.py` in the same folder.

7. Create code-cells for each step below. Spyder uses Jupyter as Spyder recognizes a comment line starting with #%% as the start of a code cell.

   ```
   #%% this starts a code cell
   print('Hello')
   #%% This ends the previous code cell and starts a new one
   ```

Import pandas and numpy into your Python session

```
#%% Imports
import numpy as np
import pandas as pd
```

Load data

Look up documentation for `read_csv` function in Pandas Use this function to read the data into a DataFrame variable `df`. Take a look at the first few rows, use `DataFrame.head()` function. Check the summary statistics using `DataFrame.describe()` function.

Select columns

Now select the columns for salary in USD, and experience from this dataframe, and assign these to the Python variables `salary` and `experience`.

Find some information: 5 points

Check out documentation on DataFrame / Series with single entry, `Series.item()` function.

- Create a variable `salary_range` with the minimum salary and maximum salary as a 2-tuple.
- Create a dictionary, `experience_count`. Insert the following key-value pairs:
  - `'EN'`: number of employees with entry level experience
  - `'MI'`: number of employees with mid level experience
  - `'SE'`: number of employees with senior level experience
  - `'EX'`: number of employees with executive experience

Compute some statistics: 15 points

- Create a dict `avg_salary_by_exp` with same keys as `experience_count` above, but with value = average salary for that experience group.
- Create a dict `sd_salary_by_experience` with same keys as `experience_count` above, but with value = standard deviation in salary for that experience group.
- Create a dict `avg_remote_ratio` with same keys as above, but average remote ratio for value.

Find job-titles with highest salary: 5 points

Use pandas `DataFrame.groupby` to find the top three job titles with the highest average salary, store them in a list called `highest_salary_titles`.

Economic recession: 3 points

There is a recession, and evrybody must take a paycut. Modify the dataframe `df` to give everybody a paycut of 10% (so if they earned 100 USD, their pay should now be 90 USD). Check the `DataFrame.loc` function to figure out how to correctly modify a dataframe.

Remove some entries: : 2 points

Now modify `df` to remove all executives with salary < 100,000 USD. Lookup `DataFrame.drop`, pay attention to the `inplace` keyword argument.

Upload your work on codePost