Lab 1: D7054E

Umuthan Ercan (umuerc-2)
Sergio Serrano Hernandez (serser-1)
Group 34

February 11, 2023

Abstract

This lab report provides information on how we fulfilled the requirements of part A from Lab 1, which was mainly focused on the implementation of object-oriented programming basics on two different data sets. We were expected to work on the "World Happiness Report Dataset" [1] and "Who Coronavirus Disease (COVID-19) Dashboard" [2], and perform some operations such as reading the csv files, scoping on a certain number of rows, and executing some numerical procedures in the perspective of OOP.

Introduction

Part A required us to implement object-oriented programming concepts like inheritance and polymorphism we were taught in the lecture, on two different data sets containing numerical statistics about the world happiness index and covid-19 disease cases.

A challenge for this lab is that we could not use libraries such as pandas which would facilitate this task a lot. Unfortunately we were not able to implement part B due to lack of time but we are very looking forward to implement further parts B in other labs.

To fulfill this, we created a parent class, declared related methods of functions inside, and inherited these functions in two child classes containing two different data sets we are provided. We used Anaconda environment and Python programming language in our work, we tried to avoid certain Python packages to challenge ourselves and decided to conclude our work with part A. The following sections will provide further details of our study.

Methodology

1. In the code implementation, we started with building functions to read csv files, preprocess data, take 10 rows, and execute certain requested numerical operations on given data sets. For generalization and object-oriented programming purposes, we chose to declare mentioned functions in a parent class called "dataset" and inherit these functions from the two child classes we generated, one for each .csv file.
2. As mentioned before, we formed a parent class called "dataset" and added the necessary functions to it. In the subsequent bullet points, I will try to explain the steps we followed.
   - We initialized the parent class and the init constructor. you will be able to see it in the provided python notebook. The parent class is named dataset, very simple but intuitive name.
   - We formed a procedure to read from a given dataset and generate a data structure from 10 rows. The way that it works is that we read all the lines of the file in a list of lists, and then iterating ten times we get random rows in the final data structure that we will use for working.
   - The data sets had mostly numerical elements, but they also had headers. The headers of course happened to be the first row of the list of lists, so what we did is copying that first row in another list, and deleted that first list from the original list of lists. Then iterating ten times we added random rows from the original list of lists to another new one with which we will work. Since we are told to just use ten rows, and in one of the datasets just having zeros in the numeric values of all rows, we decided to randomize that and get rows from all over the dataset.
   - We provided separate procedures for the numbers according to the requested operations; such as searching mean, variance, standard deviation, and min/max values. For being able to work with the values that are numeric, we first have to convert them to float values. For knowing if a row has numeric values or not, we implemented a method called is_number, that basically takes the first element of the column that we want to check (since the first element will have the same type as the rest of the elements), and we try to convert it to a float, if it works, then the value is a number, and if it does not work the value is not a numeric value.

- We added printing code for the class so when you initialize a class you will see all the ten random rows selected for the dataset as well as the headers, and for the columns that are numeric values we will be displaying calculations including standard deviation, variation, mean, and minimum and maximum value.
- Using inheritance, we created child data classes for the happiness and the covid datasets. After that, we initialize the classes with the names "happyclass" for the happinness class object, and "covidicos" for the covid dataclass.

1 Results

Since we only finalized Part A, unfortunately we cannot add many graphical results, but we can observe how the implemented code does what is intended, which is printing 10 random rows from the datasets and calculating mean, variances, standard deviations, and minimum and maximum values of each category in the datasets.

It is important to mention that since each time we are executing the code we are taking ten random rows from each dataset, the results will vary from execution to execution.

```
printing the dataset:  2019.csv

 ['Overall rank', 'Country or region', 'Score', 'GDP per capita', 'Social support', 'Healthy life expectancy', 'Freedom to make life choices', 'Generosity', 'Perceptions of co
rruption']
['109', 'Cambodia', '4.700', '0.574', '1.122', '0.637', '0.609', '0.232', '0.062\n']
['38', 'Slovakia', '6.198', '1.246', '1.504', '0.881', '0.334', '0.121', '0.014\n']
['123', 'Mozambique', '4.466', '0.204', '0.986', '0.390', '0.494', '0.197', '0.138\n']
['41', 'Uzbekistan', '6.174', '0.745', '1.529', '0.756', '0.631', '0.322', '0.240\n']
['52', 'Thailand', '6.008', '1.050', '1.409', '0.828', '0.557', '0.359', '0.028\n']
['11', 'Australia', '7.228', '1.372', '1.548', '1.036', '0.557', '0.332', '0.290\n']
['103', 'Congo (Brazzaville)', '4.812', '0.673', '0.799', '0.508', '0.372', '0.105', '0.093\n']
['84', 'North Macedonia', '5.274', '0.983', '1.294', '0.838', '0.345', '0.185', '0.034\n']
['90', 'Azerbaijan', '5.208', '1.043', '1.147', '0.769', '0.351', '0.035', '0.182\n']
['131', 'Myanmar', '4.360', '0.710', '1.181', '0.555', '0.525', '0.566', '0.172\n']
the mean of the row: Overall rank is: 78.2
the standard deviation of the row: Overall rank is: 38.384371819791454
the variation of the row: Overall rank is: 1473.3600000000001
the maximum value of the row: Overall rank is: 131.0
the minimum value of the row: Overall rank is: 11.0


the mean of the row: Score is: 5.4428
the standard deviation of the row: Score is: 0.8815469131021898
the variation of the row: Score is: 0.7771249599999999
the maximum value of the row: Score is: 7.228
the minimum value of the row: Score is: 4.36


the mean of the row: GDP per capita is: 0.86
the standard deviation of the row: GDP per capita is: 0.32843325044824556
the variation of the row: GDP per capita is: 0.1078684
the maximum value of the row: GDP per capita is: 1.372
the minimum value of the row: GDP per capita is: 0.204
```

Figure 1: This figure shows some of the results for the happiness dataset.

```
printing the dataset:  WHO-COVID-19-global-data.csv

 ['ï»¿Date_reported', 'Country_code', 'Country', 'WHO_region', 'New_cases', 'Cumulative_cases', 'New_deaths', 'Cumulative_deaths']
['2020-11-02', 'US', 'United States of America', 'AMRO', '92783', '9222942', '942', '235457\n']
['2020-09-07', 'MG', 'Madagascar', 'AFRO', '50', '15319', '1', '200\n']
['2022-04-11', 'BN', 'Brunei Darussalam', 'WPRO', '269', '138530', '1', '101\n']
['2022-06-17', 'TC', 'Turks and Caicos Islands', 'AMRO', '0', '6189', '0', '36\n']
['2021-12-24', 'MR', 'Mauritania', 'AFRO', '69', '40308', '1', '860\n']
['2021-01-02', 'AG', 'Antigua and Barbuda', 'AMRO', '0', '159', '0', '5\n']
['2020-08-11', 'ZW', 'Zimbabwe', 'AFRO', '99', '4748', '0', '108\n']
['2020-02-26', 'FJ', 'Fiji', 'WPRO', '0', '0', '0', '0\n']
['2022-04-24', 'XC', 'Saba', 'AMRO', '0', '526', '0', '0\n']
['2021-06-25', 'VE', 'Venezuela (Bolivarian Republic of)', 'AMRO', '1179', '264551', '18', '3007\n']
the mean of the row: New_cases is: 9444.9
the standard deviation of the row: New_cases is: 27781.47439013992
the variation of the row: New_cases is: 771810319.2900002
the maximum value of the row: New_cases is: 92783.0
the minimum value of the row: New_cases is: 0.0


the mean of the row: Cumulative_cases is: 969327.2
the standard deviation of the row: Cumulative_cases is: 2752413.582241477
the variation of the row: Cumulative_cases is: 7575780527707.359
the maximum value of the row: Cumulative_cases is: 9222942.0
the minimum value of the row: Cumulative_cases is: 0.0


the mean of the row: New_deaths is: 96.3
the standard deviation of the row: New_deaths is: 281.94930395374274
the variation of the row: New_deaths is: 79495.41
the maximum value of the row: New_deaths is: 942.0
the minimum value of the row: New_deaths is: 0.0


the mean of the row: Cumulative_deaths is: 23977.4
the standard deviation of the row: Cumulative_deaths is: 70498.71387224026
the variation of the row: Cumulative_deaths is: 4970068657.640001
the maximum value of the row: Cumulative_deaths is: 235457.0
the minimum value of the row: Cumulative_deaths is: 0.0
```

Figure 2: This figure shows some of the results for the covid dataset.

2 Discussion

As we mentioned earlier, each execution works with different data but the results tend to be always similar.

Based on the results for the happiness dataset we could argue the data seems to be quite even, which we notice with the standard deviation of the different columns not being very high. That data is even that does not mean that it does not vary, we have to beware of the maximum and minimum values that we have for each column, and based on that we can determine if the standard deviation that we have is big or small. For example if we have that the standard deviation of something is 20, we might think that it is very high and we have uneven data, but if we say that the minimum value is 0 and the maximum is 1000, then having 20 as a standard deviation is small compared to the value range that we are working with. Whether a statistical value is "large" or "small" is relative.

Moving on to the other dataset, the one about covid cases, we can clearly notice how the standard deviation and variance increase quite a lot compared to the other dataset's variances and standard deviations. This is mainly due to how covid cases increased rapidly over time and that makes the minimum and maximum values be very different since we are taking 10 random rows that might contain data from very different times.

Although it is not about the implementation of the lab, we had some confusion about the relativity of certain columns in the happiness data set, such as "freedom to make choice" and "generosity" and how they can be measured and enumerated.

3 Conclusion

From our perspective, the first lab was parallel to the course syllabus and a good exercise for object-oriented programming in python. The data provided was interesting, and we had the chance to test our python skills and challenge ourselves. The difficulty level was also decent, we experienced some struggles but we managed to succeed

so we're satisfied with the result.

The conclusions we get from this lab are varied, going from the importance of knowing the data that you are working with to understanding the importance of inheritance. In this particular case we have only implemented 2 datasets, for which we have created 2 subclasses and we already can see how many lines of code we have saved due to inheritance, if we were to implement more classes we could just see how efficient using inheritance is.

References

[1] "who coronavirus disease (covid-19) dashboard". https://www.overleaf.com/3657847762kyhbkjykhhkc. Accessed: 01.02.2022.

[2] "world happiness report dataset". https://covid19.who.int/table. Accessed: 01.02.2022.