CSC110Y1-F Fall 2020 - Fundamentals of Computer Science 1
Course Project Proposal

Ching Chang   Letian Cheng   Arkaprava Choudhury   Hanrui Fan

November 4, 2020

1. **Part 1**

Solution. ∎

Ching Chang, Letian Cheng
Arkaprava Choudhury, Hanrui Fan

2. **Part 2**

- Give an overview of any background knowledge necessary for the reader to understand the problem you are studying.

- Provide context for the problem and motivate why you have chosen your research question.

- Your research question should be in *bold*; it should be fairly concise, but can be more than one sentence.

Solution. ∎

3. **Part 3**

- State the source (e.g., government/organization website) and format (e.g., text, csv, json, image) of the dataset, and give some sample data contained inside that dataset.

- Don't be afraid to cobble together your own dataset, such as creating a collection of images that are related. Or to combine two datasets from different sources.

- You will also submit a small sample of your dataset to MarkUs along with your project proposal document. (See more below)

Solution. ∎

CSC110Y1-F Fall 2020
Ching Chang, Letian Cheng   Fundamentals of Computer Science 1
Arkaprava Choudhury, Hanrui Fan   Course Project Proposal

4. # Part 4

- Describe the kinds of computations you plan to perform, such as: data transformation/filtering/aggregation, computational models, and/or algorithms.

- Explain how your program will report the results of your computation in a visual and/or interactive way. You don't need to go into a lot of details here, but it should be clear what you plan to do.

Technical requirement: for your project, you must use at least one Python library/module that we have not covered in this course, *or* use <u>plotly</u> or <u>pygame</u> to a much larger extent than what what have given you so far in this course. (See examples and note in the next section).

- In this part of your proposal, you should also describe one new library you intend to use, how you will use it, and why it is appropriate. Refer to specific functions, data types, and/or capabilities of the library that make it relevant for solving the problem you wish to solve.

Solution.

We first create a function that parses the `html` element of the stats on the website as a string, and converts it to a `nested array` so that it's easier to work with. This will involve using a `for loop`, `if statements`, and an `accumulator` keeping track of the data parsed so far. Using this function, we will collect our data for deforestation in the Amazon rain-forest over the past few decades. With the data now converted into a form that we can easily manipulate, we shall focus on analysing the data using our own functions.

For this project, we use smooth polynomial fitting to relate two of the variables in our `nested list`. Now, although there exist readily available functions that would do the same in the module scipy, we try implementing our own functions for the same, to test our learning from the course.

We split the mathematical algorithm for this problem using top-down design. Firstly, the main function would have two lists, `l_x`, `l_y`, of same length as input (for the two variables), along with an integer n (where $n \leq$ `len(l_x)`; representing degree of intended polynomial). The function body would have calls to helper functions. Note: this is only a rough outline and the exact technical details may be changed based on the results after testing the functions.

Firstly, we have a function to calculate the perpendicular distance of one point from a given polynomial. As opposed to the naive approach to the problem, we use Newton-Raphson method *repeatedly* to estimate a solution for the derivative of the expression for the difference between the point and the polynomial, hence finding the coordinates of the foot of perpendicular, and consequently, the length of perpendicular.

The first estimate for the polynomial will be trivial, and we will then run the simplex algorithm to minimize the sum of the squares of the perpendiculars using two more

helper functions, yielding the polynomial regression model. We now move towards plotting the resulting graph using the matplotlib library.

We also plan to write a function to calculate the coefficient of determination to check whether the graph shows an appropriate relationship between the independent variable (forest cover) and the dependent variable.

In addition to the graph, we plan to create an interactive text-based report of our data, where the user inputs a value for the independent or dependent variable, and the program will provide the corresponding dependent or independent value, coefficient of determination, or the slope of the tangent at the point, depending on which one the user asks for. The output will be text-based, and will require string concatenation, and if statements to check whether to add trivial information to the report.

The input/output model will use while loops and input prompts to keep the program interactive. We also extrapolate the data to yield the predictions about future data using the interactive i/o model. Finally, we use the extrapolated data to summarize the upcoming significant years where the dependent variable will reach a certain milestone.

∎