COMP 1012 Fall 2017 Assignment 4

Due Date: Friday, August 03, 2018, 11:59 PM

Material Covered

- Numpy
- Function

Notes:

- Name your script file as follows: <LastName><FirstName>A2Q1.py. For example, LiJaneA2Q1.py is a valid name for a student named Jane Li. If you wish to add a version number to your file, you may add it to the end of the file name. For example, SmithRobA2Q1V2.py is a valid name for Rob Smith.

  Spyder automatically adds the .py part (called the extension) to your file name, so you don't have to type that part when saving the file. However, please ensure that your file name DOES include the .py extension, since our marking software looks for it.
- Name your output file as follows: <LastName><FirstName>A2Q1output.txt. For example, LiJaneA2Q1output.txt is a valid name for a student named Jane Li.

  In order to prevent Spyder from automatically adding the .py extension to your output file name, you must type the complete file name, including the .txt extension, when saving your output file.
- Follow the posted programming standards to avoid losing marks. Check your script for adherence to the programming standards by using CheckStandardsV2.py, which will show you exactly where your script does not comply. We will use the same script to assign marks for your assignment, so you have NO EXCUSE for losing these marks.
- You must complete the ***Blanket Honesty Declaration*** checklist in order to submit your assignment. This applies to all assignments in COMP 1012.
- To submit the assignment follow the instructions on the course website carefully. You will upload both script and output files, via the course website. We will demonstrate the assignment hand-in procedure in lectures. There will be a period of about a week before the due date when you can submit your assignment. ***Do not be late!*** If you try to submit your assignment after the late submission deadline, you will get a message indicating that the deadline has passed.

1. tutorial.math.lamar.edu/Classes/CalcII/SurfaceArea.aspx

Question 1—Statistical analysis of large datasets [100 marks]

In this assignment, we are going to find the length and area under a curve. Suppose, our curve can be represented with the function: $y = f(x) = \frac{2}{3}(x-1)^{5/2}$. So for each value of x, we will get a corresponding y value. Now, if we plot these values, we will get a figure like the one we showed here. To calculate the length of the curve, we can consider the curve between two neighboring points is a straight line and so we need to calculate the Euclidean distance between the two neighboring points and then take the summation of those distances. So, if we have n points, the length of the curve will be:

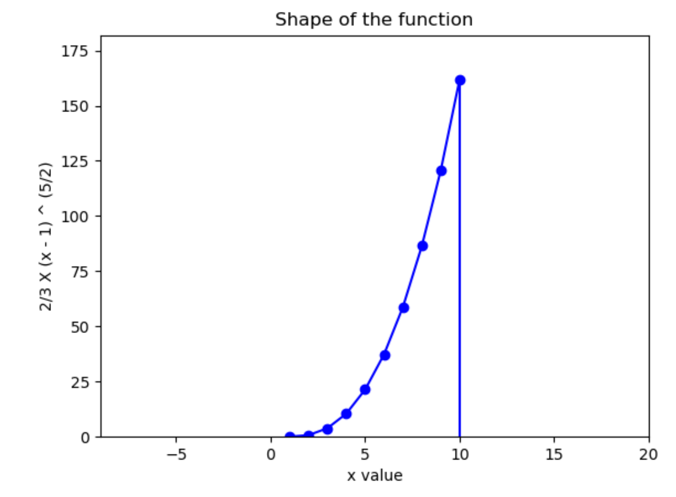


Figure 1

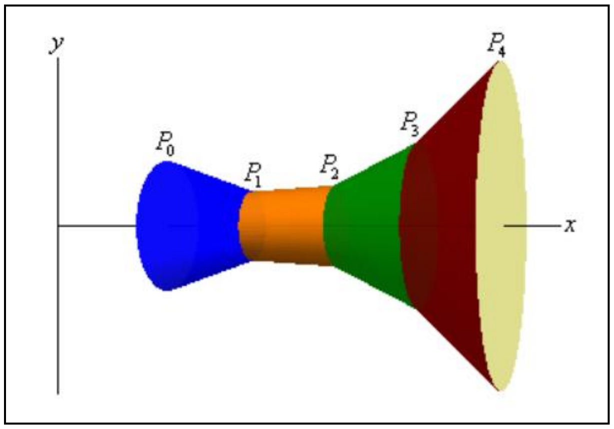
$$length = \sum_{i=1}^{n-1} |P_{i+1}, P_i|$$

Where $|P_{i-1}, P_i|$ is indicating the Euclidean distance between two points. The Euclidean distance can be calculated using the following formula:

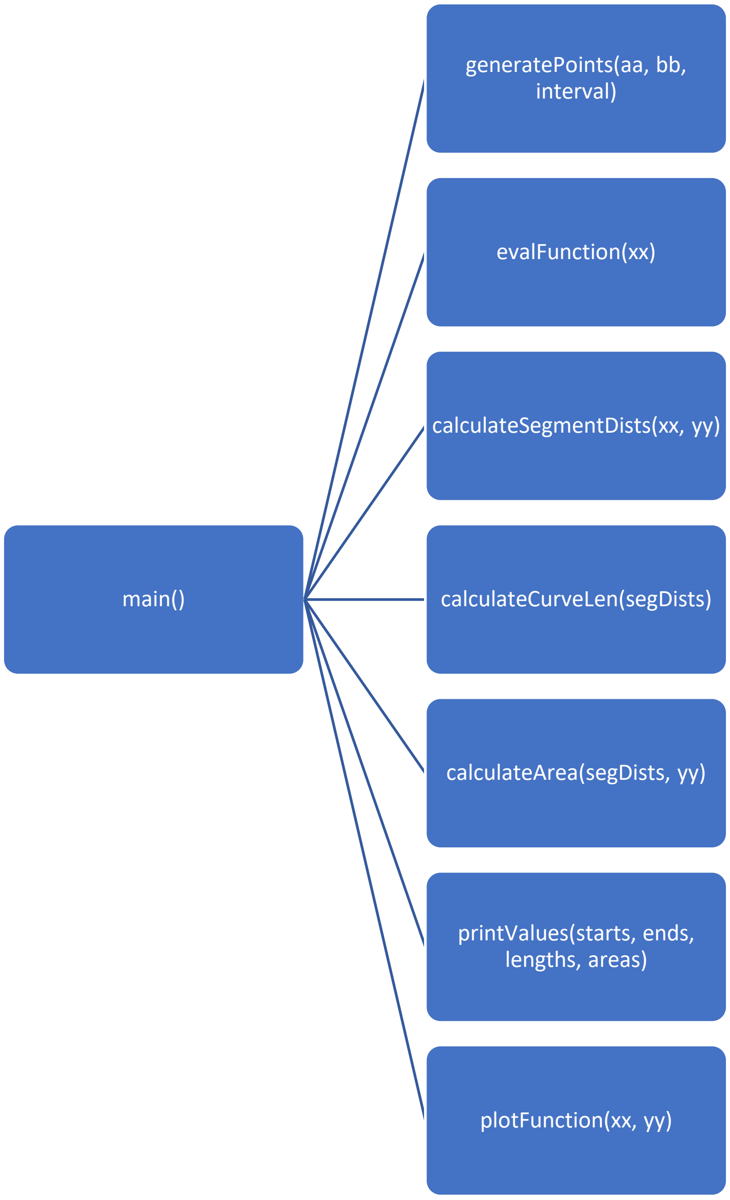
$$|P_{i-1}, P_i| = \sqrt{(x_{i-1} - x_i)^2 + (y_{i-1} - y_i)^2)}$$

The area of the curve can be calculated as the way we find the length of the curve. We can approximate the area of a small part of the curves and then take the summation of those small parts to finally get the approximate area. This can be compared to frustums [1]. The area of a frustum: $area = 2\pi rl$. This equation can be rewritten as:

$$area = \sum_{i=1}^{n-1} 2\pi \left(\frac{f(x_i) + f(x_{i+1})}{2}\right) |P_i, P_{i+1}|$$



1. tutorial.math.lamar.edu/Classes/CalcII/SurfaceArea.aspx


generatePoints(aa, bb, interval)
evalFunction(xx)
calculateSegmentDists(xx, yy)
main()
calculateCurveLen(segDists)
calculateArea(segDists, yy)
printValues(starts, ends, lengths, areas)
plotFunction(xx, yy)

generatePoints(aa, bb, interval):

This function will generate some points using arange function from numpy. *aa* is the start and *bb* is the end and interval is the step. This function will return the generated values to the main function.

evalFunction(xx):

This function takes one parameter as input. The parameter *xx* is the returned list of *generatePoints* function. It evaluates the function mentioned in the description and returns the *y* values.

1. tutorial.math.lamar.edu/Classes/CalcII/SurfaceArea.aspx

calculateSegmentDists(xx, yy):

This function takes two numpy array as input. The values in the *xx* array indicate x axis and values in *yy* array indicates y axis. It then returns the distances between each segment as a list. Here the segment means the Euclidean distance between the neighboring points.

calculateCurveLen(segDists):

This function takes segment distances as input and returns the curve length.

calculateArea(segDists, yy):

This function takes segment distances and y values as input and return the area under the curve.

main():

The program will begin from this function. In this function, you need to use a for loop that starts at 1 and ends at 100 with step 10. Inside the for loop you need to call the ***generatePoints*** function where the first parameter will be equal to the value of each iteration of for loop. The second parameter will be *(the value of each iteration of for loop + 10) and the interval* is equal to 1. Then you need to call other functions inside the main function to get areas and length of the curve. You also need to print the values and plot one of the curves. Remember, you just need to plot one curve not all of them.

printValues(starts, ends, lengths, areas):

This function takes a list of start points, end points, lengths and areas as input and show them in the output screen as given below.

plotFunction(xx, yy):

This function takes x axis and y axis as input and plot the figure as figure 1.

1. tutorial.math.lamar.edu/Classes/CalcII/SurfaceArea.aspx

Sample Output:

```
start/end |length (m)|area (m^2)
----------+----------+----------
 1 ...  10 |1.629E+02 |8.250E+04
----------+----------+----------
11 ...  20 |8.383E+02 |3.318E+06
----------+----------+----------
21 ...  30 |1.827E+03 |2.417E+07
----------+----------+----------
31 ...  40 |3.046E+03 |9.205E+07
----------+----------+----------
41 ...  50 |4.458E+03 |2.514E+08
----------+----------+----------
51 ...  60 |6.040E+03 |5.619E+08
----------+----------+----------
61 ...  70 |7.775E+03 |1.098E+09
----------+----------+----------
71 ...  80 |9.650E+03 |1.950E+09
----------+----------+----------
81 ...  90 |1.166E+04 |3.222E+09
----------+----------+----------
91 ... 100 |1.378E+04 |5.034E+09
----------+----------+----------
```

Hand-in

You will hand in your program script file. Also hand in the output produced by your code that looks like the example. You also need to submit the figure that you created. You can select your output in the IPython console window, then right-click and Copy (Raw Text) and paste it into a new document.

1. tutorial.math.lamar.edu/Classes/CalcII/SurfaceArea.aspx