CS57800: Statistical Machine Learning

HW0: Self tests & Homework Submission Instructions

HW0 submission

Please read this document carefully and play with the submission system with your hw0. You hw0 will not be graded, but this is still a mandatory homework for you to prepare yourself for this course. Unable to use the *turnin* system described in this homework in your future formal submissions will cause your homework not to be graded, and failing to follow the requirements will cause large penalty on grading your future homework.

1 Problem Solving

Mathmatical Foundations

Please test yourself on these questions and try to type the answers in LaTeX for some practice to measure whether you will be comfortable with this course. If you feel uncomfortable with these question, you may need to consider strengthening your mathematical and statistical background before you take this course.

1. Consider the planes $x_1+x_2+3x_3 = 4$ and $x_1+2x_2+4x_3 = 5$ in $\mathbb{R}^3$. Find parametric equations for the line of intersection of these two planes.

2. Given three points $P(0,0,0)$, $Q(1,-1,1)$, $R(4,3,7)$, find a vector which is orthogonal to the plane through P, Q, and R.

3. Differentiate the following equations.

   (a) $f(x) = (3x^2)(x^{\frac{1}{2}})$
   (b) $f(x) = (e^{2x} + e)^{\frac{1}{2}}$
   (c) $f(x) = [\ln(5x^2 + 9)]^3$

4. Find $\frac{\partial f}{\partial x}$, $\frac{\partial f}{\partial y}$.

   (a) $f(x,y) = xy^3 + x^2y^2$
   (b) $f(x,y) = xe^{2x+3y}$

5. We say that $f(n) \prec g(n)$ if $g(n)$ grows faster than $f(n)$. Order the following functions by $\prec$ from the the lowest to the highest:
   $(\frac{5}{3})^{2n}$, 10^8, $\sqrt{n^3}\log^2 n$, $2^{\log_2 n}$, $\log^4\sqrt{n}$, $2^{3\log_2 n}$, 2^n

6. Suppose you roll three dice. Compute the followings: (a) the expected value of the sum of the rolls, (b) the expected value of the product of the rolls, and (c) the variance of the sum of the rolls.

Python Programming

Please test yourself on this simple programming problem. If you feel uncomfortable with it, you may need to consider strengthening your programming background before you take this course.

Please find the attached "nodes.csv" file. The format of this file is as following:

```
<0:100>, <1:1000>
<1:1000>, <2:10000>
<0:100>, <2:10000>
...
```

This *csv* file represents a graph (adapted from a real Facebook social network graph), where each line is a pair of $\langle nodeID : nodeValue\rangle, \langle nodeID : nodeValue\rangle$, indicating an edge. Please use python2.7 to implement a recursive BFS (Breadth-first search) algorithm to traverse the graph and find the node with the smallest value (if there are two nodes of the same smallest value, your algorithm can return either one). Please test your program on *data.cs.purdue.edu*. Your program should be fairly fast.

2 Homework Requirements

General Programming Requirements

You should implement your solution using **Python 2.7**. You may **not** use anybody else's code; you **must** implement it by yourself. We will use **MOSS** (Measure Of Software Similarity) from Stanford University to check for plagiarism. You should submit your source code files along with your typed HW report. External packages may be allowed in the homework but only as specified, which vary among different homework. All python generic packages (those shipped with python2.7, e.g., *math, OS, system*) are allowed. You need to name your main file as "`hw⋆.py`", where "⋆" is the index of the homework. The TAs should be able to compile and run your code.

Submission Instructions:

You are required to use LaTeX to type your solutions to questions, and report of your programming as well. `https://www.overleaf.com/` is a website you can use freely as a Purdue student. Other formats of submission will **not** be accepted. A template named "homework_template.tex" is also provided for your convenience.

Your code will be tested on *data.cs.purdue.edu*, where you submit your homework as well. After logging into *data.cs.purdue.edu* (physically go to the lab or use ssh remotely, as you are all granted the accounts to CS data machines during this class), please follow these steps to submit your assignment:

1. Make a directory named *'yourname_yoursurname'* (all letters in lower case) and copy all of your files there, including your report in *.pdf* format generated using LaTeX, and the 'hw*.py' shall reside directly in this directory.
2. While in the upper level directory (if the files are in /homes/dan/dan_goldwasser, go to/homes/dan), execute the following command (for instance, hw1):

   ```
   turnin -c cs578 -p HW1 *your_folder_name*
   ```

   (e.g. your instructor would use: `turnin -c cs578 -p HW1 dan_goldwasser` to submit his homework)

   Keep in mind that old submissions are overwritten with new ones whenever you execute this command.

3. You can verify the contents of your submission by executing the following command:

   ```
   turnin -v -c cs578 -p HW1
   ```

   Do **not** forget the -v flag here, as otherwise your submission would be replaced with an empty one.

Your submission should include the following files:

1. The source code in python2.7.
2. Your evaluation & analysis in .pdf format (together with your HW solutions in it).
3. (optional) A README file containing your name, instructions to run your code and anything if you would like us to know about your program (like errors, special conditions, etc).

Failure to follow the above instructions will incur the penalty when your homework is being graded.

Homework Grading Rubrics:

In the past, some students asked about code grading rubrics. Though it is an advanced graduate-level course assuming students are equipped with strong programming skills, we still post some code grading rubrics for rough reference:

1. The program is executable: 20%.
2. You implemented the required algorithms correctly: 35%.
3. The program has no spam on the screen: 5%.
4. The program print the results in a correct/readable (user-friendly) format on the shell: 5%.
5. The program can finish running within a reasonable time duration: 15%.
6. You describe the task, the algorithm(s) clearly and show the results in the report by the table(s) or/and graphic(s): 20%.

The percentages are for one programming question in one homework.

Late Policies

You lose 10% of the grade for every 24 hours, for the first 48 hours. After that, you will not be able to submit.

Plagiarism Policies

Seriuosly, no cheating. If plagiarism was found, both you and the one whose homework you “referred to” receive 0 point on that homework. If you were found to have plagiarised more than once, we will report to the instructor and the department (and your home department if you are not a CS student).