CS 578: Statistical Machine Learning
Homework 0

Yin Wang
wang2485@purdue.edu

August 29, 2018

1 Problem Solving

1.1 Problem 1

1.1.1 Problem Statement

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

1.1.2 Solution

step1: The normal vector to plane 1:

$$\vec{n_1} = \begin{pmatrix} 1 \\ 1 \\ 3 \end{pmatrix}$$

step2: The normal vector to plane 2:

$$\vec{n_2} = \begin{pmatrix} 1 \\ 2 \\ 4 \end{pmatrix}$$

step3: The direction vector that go through both plane 1 and plane 2:

$$\vec{d} = \vec{n_1} \times \vec{n_2} = \begin{pmatrix} 1 \\ 1 \\ 3 \end{pmatrix} \times \begin{pmatrix} 1 \\ 2 \\ 4 \end{pmatrix} = \begin{pmatrix} -2 \\ -2 \\ 1 \end{pmatrix}$$

step4: Find a point $\vec{x}$ that lies on both plane 1 and plane 2:

$$\vec{x} = \begin{pmatrix} 3 \\ 1 \\ 0 \end{pmatrix}$$

step5: The parametric equation could be expressed as:

$$\begin{pmatrix} 3 \\ 1 \\ 0 \end{pmatrix} + t \begin{pmatrix} -2 \\ -2 \\ 1 \end{pmatrix}, t \in \mathbb{R}$$

or we could write as:

$$\begin{cases} x = 3 - 2t \\ y = 1 - 2t \\ z = t \end{cases} \quad , t \in \mathbb{R}$$

1.2 Problem 2

1.2.1 Problem Statement

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.

1.2.2 Solution

step1:

$$\vec{PQ} = \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix}$$

step2:

$$\vec{PR} = \begin{pmatrix} 4 \\ 3 \\ 7 \end{pmatrix}$$

step3: Orthogonal vector is:

$$\vec{n} = \vec{PQ} \times \vec{PR} = \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix} \times \begin{pmatrix} 4 \\ 3 \\ 7 \end{pmatrix} = \begin{pmatrix} -10 \\ -3 \\ 1 \end{pmatrix}$$

1.3 Problem 3

1.3.1 Problem Statement

Differentiate the following equations:

(a) $f(x) = (3x^2)(x^{\frac{1}{2}})$

(b) $f(x) = (e^{2x} + e)^{\frac{1}{2}}$

(c) $f(x) = [ln(5x^2 + 9]^3$

1.3.2 Solution

(a) $f'(x) = \frac{15}{2}x^{\frac{3}{2}}$

(b) $f'(x) = \frac{1}{2}(e^{2x} + e)^{-\frac{1}{2}}2e^{2x} = e^{2x}(e^{2x} + e)^{-\frac{1}{2}}$

(c) $f'(x) = [3ln(5x^2 + 9)^2](\frac{1}{5x^2+9})(10x) = \frac{30x}{5x^2+9}ln(5x^2 + 9)^2$

1.4 Problem 4

1.4.1 Problem Statement

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

(a) $f(x,y) = xy^3 + x^2y^2$

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

1.4.2 Solution

(a)

$$\frac{\partial f}{\partial x} = y^3 + 2xy^2$$
$$\frac{\partial f}{\partial y} = 3xy^2 + 2x^2y$$

(b)

$$\frac{\partial f}{\partial x} = e^{2x+3y} + 2xe^{2x+3y} = (2x+1)e^{2x+3y}$$
$$\frac{\partial f}{\partial y} = 3xe^{2x+3y}$$

1.5 Problem 5

1.5.1 Problem Statement

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

1.5.2 Solution

$\sqrt{n^3}log^2n = n^{\frac{3}{2}}(log(n))^2$
$log^4\sqrt{n} = (log\sqrt{n})^4 = (log(n)^{\frac{1}{2}})^4 = (\frac{1}{2}log(n))^4 = \frac{1}{16}(log(n))^4$
$2^{log_2 n} = n$
$2^{3log_2 n} = n^3$
We know that $10^8 \prec n \prec n^3 \prec 2^n$.
Now, let's compare 2^n and $(\frac{5}{3})^{2n}$

$$\lim_{n\to\infty} \frac{(\frac{5}{3})^{2n}}{2^n} = \lim_{n\to\infty} \frac{(\frac{5}{3})^n(\frac{5}{3})^n}{(\frac{5}{3})^n(\frac{6}{5})^n} = \lim_{n\to\infty} (\frac{25}{18})^n \to \infty$$

So $10^8 \prec n \prec n^3 \prec 2^n \prec (\frac{5}{3})^{2n}$.
Next,let's compare $\sqrt{n^3}log^2n$ and $log^4\sqrt{n}$

$$\lim_{n\to\infty} \frac{log^4\sqrt{n}}{\sqrt{n^3}log^2n} = \lim_{n\to\infty} \frac{\frac{1}{16}(log(n))^4}{n^{\frac{3}{2}}(log(n))^2} = \lim_{n\to\infty} (\frac{1}{16})\frac{log(n)^2}{n^{\frac{3}{2}}} = \lim_{n\to\infty} (\frac{1}{18})(\frac{1}{n})^{\frac{3}{2}} \to 0$$

So $log^4\sqrt{n} \prec \sqrt{n^3}log^2n$
Next, we will compare $\sqrt{n^3}log^2n$ and n^3:

$$\lim_{n\to\infty}\frac{\sqrt{n^3}log^2n}{n^3} = \lim_{n\to\infty}\frac{(log(n))^2}{n^{\frac{3}{2}}} = \lim_{n\to\infty}(\frac{4}{3})\frac{log(n)}{n^{\frac{3}{2}}} = \lim_{n\to\infty}(\frac{8}{9})\frac{1}{n^{\frac{3}{2}}} \to 0$$

So $\sqrt{n^3}log^2n \prec n^3$.
Last,let's compare $log^4\sqrt{n}$ and n:

$$\lim_{n\to\infty}\frac{log^4\sqrt{n}}{n} = \lim_{n\to\infty}\frac{\frac{1}{16}(log(n))^4}{n} = \lim_{n\to\infty}(\frac{3}{8})\frac{1}{n^2} \to 0$$

So $log^4\sqrt{n} \prec n$:

Conclusion: $10^8 \prec log^4\sqrt{n} \prec n \prec \sqrt{n^3}log^2n \prec n^3 \prec 2^n \prec (\frac{5}{3})^{2n}$.

1.6 Problem 6

1.6.1 Problem Statement

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.

(c) The variance of the sum of the rolls

1.6.2 Solution

(a) First, we define the event $\mathcal{A}$ as the number of the dice for only one roll. The expected value for $\mathcal{A}$,according to the definition,is:

$$\mathrm{E}[A] = \sum_{k=1}^{6}(\frac{1}{6})k = \frac{7}{2}$$

We know that the rolling of the three dice are independent, so that the expectation of the sum of the three rolls could be expressed as (We use $\mathcal{X}$ to represent the sum of three dice):

$$\mathrm{E}[X] = 3\mathrm{E}[A] = \frac{21}{2}$$

(b) We have already know that

$$\mathrm{E}[A] = \frac{7}{2}$$

Also the number we got when rolling each dice is independent,let $\mathcal{Y}$ represent the product of three dice, we will have:

$$\mathrm{E}[Y] = \mathrm{E}[A \times A \times A] = \mathrm{E}[A]\mathrm{E}[A]\mathrm{E}[A] = (\frac{7}{2})^3 = \frac{343}{8}$$

(c) From the definition of variance, we know that:

$$\mathrm{Var}[X] = \mathrm{E}[X^2] - (\mathrm{E}[X])^2$$
$$\mathrm{E}[X^2] = \mathrm{E}[(3A)^2] = 9\mathrm{E}[A^2] = 9(\mathrm{E}[A])^2$$
$$\mathrm{Var}[X] = 8(\mathrm{E}[A])^2 = 98$$

2 Python Programming

2.1 source code

```
#Read the csv file and store it as a dictonary

with open('/Users/camillewang/Documents/purdue_classes_material/Fall_2018/
    lines = f.readlines()
Edge_graph = {}
Val_graph = {}
previous_num = -1
for line in lines:
    node1 = line.split(",")[0].strip()[1:-1]
    node2 = line.split(",")[1].strip()[1:-1]
    node1_num = int(node1.split(":")[0])
    node1_val = int(node1.split(":")[1])
    node2_num = int(node2.split(":")[0])
    node2_val = int(node2.split(":")[1])
    Val_graph[node1_num] = node1_val
    Val_graph[node2_num] = node2_val
    if node1_num == previous_num:
        Edge_graph[node1_num] = Edge_graph[node1_num]+[node2_num]
    else:
        Edge_graph[node1_num] = [node2_num]
    previous_num = node1_num

# Raising the resusion limit to a big number
import sys
sys.setrecursionlimit(50000)

# Define the function recursive BFS
def recursive_BFS(graph, queue, seen, min_val, min_node):
    if (len(queue)!=0):
        node = queue.pop(0)
        if min_val>Val_graph[node]:
            min_val = Val_graph[node]
            min_node = node
        else:
            min_val = min_val
            min_node = node
        seen.add(node)
        if node not in graph.keys():
            neighbors = []
        else:
            neighbors = graph[node]
        for w in neighbors:
            if (w not in seen) & (w not in queue):
                queue.append(w)
```

```
            recursive_BFS(graph, queue, seen, min_val, min_node)
    else:
        print "Node %d has smallest number value %d\n" %(min_node, min_val)

# Initialize some variables in the function
queue = []
queue.append(Edge_graph.keys()[0])
seen = set()
min_val = 'Inf'
min_node = 0

# Call function
recursive_BFS(Edge_graph, queue, seen, min_val, min_node)
```

2.2 Result

The result of the programming is the node number is:3279 and the smallest value is: 3

2.3 Result Analysis

The result is fairly reasonable since we can sort the list of the node number and corresponding node value. The smallest value is 3 and the node is 3279