Problem Set 1

Artificial Intelligence
Fall 2021 CS47100-AI

Student name: Yiting Gan Student PUID: 0030665034

Problem 1

Solution. If θ is the mean value of $\{x_i\}_i^n$, $f(\theta)$ could be minimized, since $(\theta - x_i)^2$ would be minimized. Since $f(\theta)$ is a quadratic function, and $\left\{w_i\right\}_i^n$ are all positive, which means the function is concave up. $f(\theta)$ is minimized when $f'(\theta)$ is equal to 0.
The second derivative $f(\theta)$ is $\sum_{i=1}^n w_i$. If some of the w_i are negative, the second derivative might be 0. If the second derivative is 0, $f(\theta)$ does not have a minimum. □

Problem 2

Solution. $f(x) \geq g(x)$
For f(x), every term will be positive since we choose the proper s for each x_i to maximize $s * x_i$. However, for g(x), we could only get $sum_{i=1}^d x_i$ or $sum_{i=1}^d - x_i$.If not all x_i have the same sign, $sum_{i=1}^d x_i$ or $sum_{i=1}^d - x_i$ is definitely less than f(x). Otherwise, $f(x) = g(x)$ □

Problem 3

Solution. There are four cases:
case 1: x_1: roll a 1 P1=$\frac{1}{6}$
case 2: x_2: roll a 2 P2=$\frac{1}{6}$
case 3: x_3: roll a 6 P3=$\frac{1}{6}$
case 4: x_4: roll 3,4,5 P4= $\frac{1}{2}$

Let E be the expected number of points before stopping. $E = \frac{1}{6} * 0 + \frac{1}{6} * (E - a) + \frac{1}{6} * (E + b) + \frac{1}{2} * E$
$E = b - a$ □

Problem 4

Solution. I expand $L(p) = p^4(1-p)^3$ get $L(p) = p^4 - 3p^5 + 3p^6 - p^7)$. Based on the coefficient of p^7, a term that has the highest power, we know L(p) is a concave down function within the domain (0,1). Thus, L(p) will be maximized when $L'(p) = 0$, p = $\frac{4}{7}$. The intuitive interpretation of value of p is that the maximum power term is 7, and compare to 3, 4 is larger, so $p = \frac{4}{7}$.

□

Problem 5

Solution. We know that $P(A \bigcup B) = 1 = P(A) + P(B) - P(A \bigcap B)$ and $P(A \bigcap B) > 0$. If we rewrite the above equation, we could get $1 + P(A \bigcap B) = P(A) + P(B) > 1$. Also, given $(A|B) = (B|A)$, we know P(A) = P(B). Thus, 2P(A) > 1 $\rightarrow P(A) > 1/2$. □
Problem 6

Solution. $f(\omega) = (\sum_{i=1}^{n} \sum_{j=1}^{n} (\mathbf{a_i}^\top * w - \mathbf{b_j}^\top * w)^2) + \lambda||w||_2^2$
$\lambda||w||_2^2 = \mathbf{w}^\top w = \lambda w^2$
$f(\omega) = w^2(\sum_{i=1}^{n} \sum_{j=1}^{n} (\mathbf{a_i}^\top - \mathbf{b_j}^\top)^2 + \lambda)$
$f(w) = (\frac{\partial f(w)}{1}, ..., \frac{\partial f(w)}{d})$
$(w^2) = (\frac{\partial f(w)}{1}, ..., \frac{\partial f(w)}{d})$
$= (2\frac{\partial f(w)}{\omega_1}, ..., 2\frac{\partial f(w)}{\omega_d})$
$= 2(\frac{\partial f(w)}{\omega_1}, ..., \frac{\partial f(w)}{\omega_d})$
$f(\omega) = 2(\frac{\partial f(w)}{\omega_1}, ..., \frac{\partial f(w)}{\omega_d})(\sum_{i=1}^{n} \sum_{j=1}^{n} (\mathbf{a_i}^\top - \mathbf{b_j}^\top)^2 + \lambda)$ □

Problem 7

Solution. The asymptotic complexity is $O(n^{24})$. I approach this problem by determining how many possible ways an rectangle can be by deciding the four sides of the rectangle. Choosing two sides from the (n+1) rows and two sides from (n+1) columns makes the possible ways an rectangle $(n+1)^4$. Thus, the possible ways to place six rectangles is $(n+1)^{26}$, since these are independent events. i.e. $O(n^{24})$. □

Problem 8

Solution.
```
procedure Problem8(list vertices, vertex source)
    for each vertex (i,j) in vertices:
        if (i,j) is source then distance[i,j] := 0
        else distance[i,j] := infinity

    for vertices (i,j) from (0,0) to (n,n):
        for vertices (a,b) on the right and down direction:
            if distance[i,j] + c(a,b) < distance[a,b]:
                distance[a,b] := distance[i,j] + c(a,b)
return distance[n,n]
```
The run-time is $O(n^2)$.
The first part is a for-loop that has run-time $O(n^2)$, since it accessed all the vertices to initialize them. The second part is a outer for-loop with a inner for-loop that has run-time $O(n^2)$, since for the outer for-loop, it goes through all the vertices, and the inner for loop only goes to the vertices that are in the right direction and the down direction, which in-total would be $O(2n^2)$. □

Problem 9

Solution. a. Using split gets all the words in the text, and then using max to get the last word occurs in the dictionary by comparing all the letters.
b. Using the equation of euclidean distance $Distance = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}$.
c. The code generated dictionary for all the words in the sentence to indicate the next possible words. Then I use recursion to generate all the possible mutated sentence. The recursion will break when the mutated sentences have the same number of words to that of the original sentence.
d. Applying for loop to get the summation of the dot product of each tuples.
e. Applying for loop too modify each tuple of v1 by doing math.
f. Applying for loop to count all the words in the text and making them a set for those count==1.
g. I created a recursion to go through the word from first letter and last letter simultaneously. If I find the same letter in the word, I decrease the length of string I will go through by 2; otherwise, I create two strings that one does not include the last letter, the other does not include the first letter. The process continues until we reach the letter in the middle index. □