Mid-Term Exam

Question 1

Suppose that the error of a Neural Network against parameter x is defined as:

$f(x) = x^2 - 2x$

BY considering that the derivative of the function is: f′(x)=2x−2

Find a value of x which minimize error using the gradient descent approach with two different settings as follows:

A) starting point: 4 , learning rate (**α**): 0.4

B) starting point: 4 , learning rate (**α**): 0.7

Show iterations till x gets near 1 (a value between 0.80 and 1.20)

Discuss which of the settings is better?

(Update of x using learning rate (**α**) and gradient (derivative): $x_{new} = x_{old} - \alpha \cdot f'(x_{old})$)

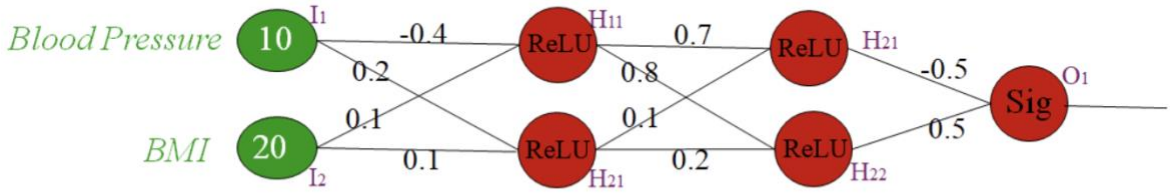
	1	2	3	4	5	6	7	8	9
0.4	1.6	1.12							
0.7	-0.2	1.48	0.808						

So, ‘A) starting point: 4 , learning rate (**α**): 0.4’ is better, use less iterations.

Question 2

Suppose that a feed forward neural network with ReLU as activation function of hidden layers' nodes and Sigmoid as activation function of output layer's node is trained to predict two classes of **Heart_Attack(1)** and **No_Heart_Attack (0)** based on Blood Pressure and BMI features.

Predict the class of the following test data with Blood Pressure of 10 and BMI of 20 :



Threshold of classification : 0.5

ReLU (X) = Max(X, 0)

ReLU(H11) = Max(10*(-0.4) + 20*0.1, 0) = Max(-2, 0) = 0

ReLU (H12) = Max(10*0.2 + 20*0.1, 0) = Max(4, 0) = 4

ReLU (H21) = Max(0*0.7 + 4*0.1, 0) = Max(0.4, 0) = 0.4

ReLU (X22) = Max(0*0.8 + 4*0.2, 0) = Max(0.8, 0) = 0.8

Sig (O1) = Sig(0.4*(-0.5) + 0.8*0.5) = Sig(0.2) = 0.54

Since 0.54 > Threshold 0.5, so we predict Heart_Attack(1)

Question 3

For tokenization using BPE (Byte-Pair Encoding) consider the following training data and min-frequency = 2

Training data = ["low", "lower", "high", "higher"]

Show iterations of BPE till there is no frequent pair of sub-words for merging based on min-frequency

(For pairs with the same frequency, the order of operation is not important).

Iteration 1: Count Pairs
Counting pairs of adjacent symbols in the dataset:
- `("l", "o")`: 2
- `("o", "w")`: 2
- `("w", "e")`: 1
- `("e", "r")`: 2
- `("h", "i")`: 2
- `("i", "g")`: 2
- `("g", "h")`: 2
- `("h", "e")`: 2

Most frequent pairs with `frequency >= 2`:
- `("l", "o")`, `("o", "w")`, `("e", "r")`, `("h", "i")`, `("i", "g")`, `("g", "h")`, `("h", "e")`

Choose `("l", "o")` to merge (order is not important).

Iteration 1: Merge `("l", "o")`
After merging `("l", "o")` to form `lo`:
1. `"lo w"`
2. `"lo w e r"`
3. `"h i g h"`
4. `"h i g h e r"`

Iteration 2: Count Pairs
- `("lo", "w")`: 2
- `("w", "e")`: 1
- `("e", "r")`: 2
- `("h", "i")`: 2
- `("i", "g")`: 2
- `("g", "h")`: 2
- `("h", "e")`: 2
Merge `("lo", "w")`.

Iteration 2: Merge `("lo", "w")`
After merging `("lo", "w")` to form `low`:
1. `"low"`
2. `"low e r"`
3. `"h i g h"`
4. `"h i g h e r"`

Iteration 3: Count Pairs
- `("e", "r")`: 2
- `("h", "i")`: 2
- `("i", "g")`: 2
- `("g", "h")`: 2
- `("h", "e")`: 2
Merge `("e", "r")`.

Iteration 3: Merge `("e", "r")`
After merging `("e", "r")` to form `er`:
1. `"low"`
2. `"low er"`
3. `"h i g h"`
4. `"h i g h er"`

Iteration 4: Count Pairs
- `("h", "i")`: 2
- `("i", "g")`: 2
- `("g", "h")`: 2
- `("h", "e")`: 2
Merge `("h", "i")`.

Iteration 4: Merge `("h", "i")`
After merging `("h", "i")` to form `hi`:
1. `"low"`
2. `"low er"`
3. `"hi g h"`
4. `"hi g h er"`

Iteration 5: Count Pairs
- `("i", "g")`: 2
- `("g", "h")`: 2
- `("h", "e")`: 2
Merge `("i", "g")`.

Iteration 5: Merge `("i", "g")`
After merging `("i", "g")` to form `ig`:
1. `"low"`
2. `"low er"`
3. `"hi gh"`
4. `"hi gh er"`

Iteration 6: Count Pairs
- `("g", "h")`: 2
- `("h", "e")`: 2
Merge `("g", "h")`.

Iteration 6: Merge `("g", "h")`
After merging `("g", "h")` to form `gh`:

1. `"low"`
2. `"low er"`
3. `"high"`
4. `"high er"`

Iteration 7: Count Pairs
- `("h", "e")`: 2

Merge `("h", "e")`.

Iteration 7: Merge `("h", "e")`
After merging `("h", "e")` to form `he`:

1. `"low"`
2. `"lower"`
3. `"high"`
4. `"higher"`

Final Result
The final subword vocabulary is:
`"low"`, `"lower"`, `"high"`, `"higher"`

Question 4
Suppose that word embeddings (with the same dimensions) were created using an approach which preserves relationships between embeddings precisely.

A) If the following equations are correct calculate the embedding of “brothers”:

E(students) - E(student) = [0,0,2]

E(father) - E(mother) = [0,1,0]

E(sister)=[-2,1,0]

E(brothers)= ?

B) Suppose that the following sentence is processed using self-attention mechanism and the context is already added to embeddings of words. What is the embedding of “sibling” after adding the context by the self-attention?

Sentence: “I have one male sibling”

E(sibling)= ?

A)
Since E(sister) = [-2,1,0] and E(father) - E(mother) = [0,1,0],
So E(brother) = E(sister) + [0,1,0] = [-2,2,0]

Since E(students) - E(student) = [0,0,2],
So E(brothers) = E(brother) + [0,0,2] = [-2,2,2]

B)
E(sibling) = [E(brother) + E(sister)] / 2 = [−2,1.5,0]

Given the presence of "male" in the sentence, the self-attention mechanism will adjust E(sibling) toward E(brother).

Adding the gender shift vector [0,1,0] for male context:
E(sibling) with context = [−2,1.5,0]+[0,0.5,0]=[−2,2,0]

Question 5

Use **Greedy** and **Beam Search** (beam width=3) as decoding strategies for a sentence which starts with "I" and the following probabilities. Show steps of both decoding strategies and the resulting sentences.

Greedy:
I + buy (0.25) + it(0.10)

Beam Search (beam width=3):

First word	Probabilities - Step1	Probabilities - Step2	
I	have (0.20)	this(0.10)	0.02
		that(0.15)	0.03
		it(0.20)	0.04
		not(0.05)	0.01
	buy (0.25)	it(0.10)	0.025
		nothing(0.05)	0.0125
		anything(0.01)	0.0025
		something(0.05)	0.0125
	had(0.15)	this(0.10)	0.015
		that(0.10)	0.015
		it(0.20)	0.03
		not(0.02)	0.003
	got (0.10)	nothing(0.05)	0.005
		it(0.20)	0.02
		something(0.10)	0.01
		this(0.10)	0.01

So Beam Search Result is "I have it" with the highest cumulative probability 0.04.