Team 1: Alex A.; Madeline C.; Patrick M; Dylan P.
Artificial Intelligence (CS 534) Spring 2023
Group Project 4
22 April 2023

Question 1 (30 Points)

Part 1. Consider a 1-dimensional time-series with values [2, 1, 3, 4, 7]. Perform a convolution with a 1- dimensional filter [1, 0, 1] without padding. Show all the computational steps to get the full credit.

Here are the computational steps:

1. First convolution: [2, 1, 3] * [1, 0, 1] = (21) + (10) + (3*1) = 5
2. Second convolution: [1, 3, 4] * [1, 0, 1] = (11) + (30) + (4*1) = 5
3. Third convolution: [3, 4, 7] * [1, 0, 1] = (31) + (40) + (7*1) = 10

Therefore, the final result of the convolution is [5, 5, 10]

Part 2. Compute the convolution of the below input with the horizontal edge detection filter. Use a stride of 1 without padding. Show all the computational steps to get the full credit.

Input Matrix:

```
6 3 4 4 5 0 3
4 7 4 0 4 0 4
7 0 2 3 4 5 2
3 7 5 0 3 0 7
5 8 1 2 5 4 2
8 0 1 0 6 0 0
6 4 1 3 0 4 5
```

Horizontal Edge Detecting Filter:

```
 1  1  1
 0  0  0
-1 -1 -1
```

Convolution 1 ⇒ (6 + 3 + 4)(1) + (4 + 7 + 4)(0) + (7 + 0 + 2)(-1) = 13 + 0 - 9 = 4
Convolution 2 ⇒ (3 + 4 + 4)(1) + (7 + 4 + 0)(0) + (0 + 2 + 3)(-1) = 11 + 0 - 5 = 6
Convolution 3 ⇒ (4 + 4 + 5)(1) + (4 + 0 + 4)(0) + (2 + 3 + 4)(-1) = 13 + 0 - 9 = 4
Convolution 4 ⇒ (4 + 5 + 0)(1) + (0 + 4 + 0)(0) + (3 + 4 + 5)(-1) = 9 + 0 - 12 = -3
Convolution 5 ⇒ (5 + 0 + 3)(1) + (4 + 0 + 4)(0) + (4 + 5 + 2)(-1) = 8 + 0 - 11 = -3

Convolution 6 ⇒ (4 + 7 + 4)(1) + (7 + 0 + 2)(0) + (3 + 7 + 5)(-1) = 15 + 0 - 15 = 0
Convolution 7 ⇒ (7 + 4 + 0)(1) + (0 + 2 + 3)(0) + (7 + 5 + 0)(-1) = 11 + 0 - 12 = -1
Convolution 8 ⇒ (4 + 0 + 4)(1) + (2 + 3 + 4)(0) + (5 + 0 + 3)(-1) = 8 + 0 - 8 = 0
Convolution 9 ⇒ (0 + 4 + 0)(1) + (3 + 4 + 5)(0) + (0 + 3 + 0)(-1) = 4 + 0 - 3 = 1
Convolution 10 ⇒ (4 + 0 + 4)(1) + (4 + 5 + 2)(0) + (3 + 0 + 7)(-1) = 8 + 0 - 10 = -2

Convolution 11 ⇒ (7 + 0 + 2)(1) + (3 + 7 + 5)(0) + (5 + 8 + 1)(-1) = 9 + 0 - 14 = -5
Convolution 12 ⇒ (0 + 2 + 3)(1) + (7 + 5 + 0)(0) + (8 + 1 + 2)(-1) = 5 + 0 - 11 = -6
Convolution 13 ⇒ (2 + 3 + 4)(1) + (5 + 0 + 3)(0) + (1 + 2 + 5)(-1) = 9 + 0 - 8 = 1
Convolution 14 ⇒ (3 + 4 + 5)(1) + (0 + 3 + 0)(0) + (2 + 5 + 4)(-1) = 12 + 0 - 11 = 1
Convolution 15 ⇒ (4 + 5 + 2)(1) + (3 + 0 + 7)(0) + (5 + 4 + 2)(-1) = 11 + 0 - 11 = 0

Convolution 16 ⇒ (3 + 7 + 5)(1) + (5 + 8 + 1)(0) + (8 + 0 + 1)(-1) = 15 + 0 - 9 = 6
Convolution 17 ⇒ (7 + 5 + 0)(1) + (8 + 1 + 2)(0) + (0 + 1 + 0)(-1) = 12 + 0 - 1 = 11
Convolution 18 ⇒ (5 + 0 + 3)(1) + (1 + 2 + 5)(0) + (1 + 0 + 6)(-1) = 8 + 0 - 7 = 1
Convolution 19 ⇒ (0 + 3 + 0)(1) + (2 + 5 + 4)(0) + (0 + 6 + 0)(-1) = 3 + 0 - 6 = -3
Convolution 20 ⇒ (3 + 0 + 7)(1) + (5 + 4 + 2)(0) + (6 + 0 + 0)(-1) = 10 + 0 - 6 = 4

Convolution 21 ⇒ (5 + 8 + 1)(1) + (8 + 0 + 1)(0) + (6 + 4 + 1)(-1) = 14 + 0 - 11 = 3
Convolution 22 ⇒ (8 + 1 + 2)(1) + (0 + 1 + 0)(0) + (4 + 1 + 3)(-1) = 11 + 0 - 8 = 3
Convolution 23 ⇒ (1 + 2 + 5)(1) + (1 + 0 + 6)(0) + (1 + 3 + 0)(-1) = 8 + 0 - 4 = 4
Convolution 24 ⇒ (2 + 5 + 4)(1) + (0 + 6 + 0)(0) + (3 + 0 + 4)(-1) = 11 + 0 - 7 = 4
Convolution 25 ⇒ (5 + 4 + 2)(1) + (6 + 0 + 0)(0) + (0 + 4 + 5)(-1) = 11 + 0 - 9 = 2

After calculating each convolution, we get the following output:

```
 4  6  4 -3 -3
 0 -1  0  1 -2
-5 -6  1  1  0
 6 11  1 -3  4
 3  3  4  4  2
```

Part 3. Perform a 4 × 4 max pooling at stride 1 of the above same input. Show all the computational steps to get the full credit.

Pool 1 ⇒ max(6, 3, 4, 4, 4, 7, 4, 0, 7, 0, 2, 3, 3, 7, 5, 0) = 7
Pool 2 ⇒ max(3, 4, 4, 5, 7, 4, 0, 4, 0, 2, 3, 4, 7, 5, 0, 3) = 7
Pool 3 ⇒ max(4, 4, 5, 0, 4, 0, 4, 0, 2, 3, 4, 5, 5, 0, 3, 0) = 5
Pool 4 ⇒ max(4, 5, 0, 3, 0, 4, 0, 4, 3, 4, 5, 2, 0, 3, 0, 7) = 7

Pool 5 ⇒ max(4, 7, 4, 0, 7, 0, 2, 3, 3, 7, 5, 0, 5, 8, 1, 2) = 8
Pool 6 ⇒ max(7, 4, 0, 4, 0, 2, 3, 4, 7, 5, 0, 3, 8, 1, 2, 5) = 8
Pool 7 ⇒ max(4, 0, 4, 0, 2, 3, 4, 5, 5, 0, 3, 0, 1, 2, 5, 4) = 5
Pool 8 ⇒ max(0, 4, 0, 4, 3, 4, 5, 2, 0, 3, 0, 7, 2, 5, 4, 2) = 7

Pool 9 ⇒ max(7, 0, 2, 3, 3, 7, 5, 0, 5, 8, 1, 2, 8, 0, 1, 0) = 8
Pool 10 ⇒ max(0, 2, 3, 4, 7, 5, 0, 3, 8, 1, 2, 5, 0, 1, 0, 6) = 8
Pool 11 ⇒ max(2, 3, 4, 5, 5, 0, 3, 0, 1, 2, 5, 4, 1, 0, 6, 0) = 6
Pool 12 ⇒ max(3, 4, 5, 2, 0, 3, 0, 7, 2, 5, 4, 2, 0, 6, 0, 0) = 7

Pool 13 ⇒ max(3, 7, 5, 0, 5, 8, 1, 2, 8, 0, 1, 0, 6, 4, 1, 3) = 8
Pool 14 ⇒ max(7, 5, 0, 3, 8, 1, 2, 5, 0, 1, 0, 6, 4, 1, 3, 0) = 8
Pool 15 ⇒ max(5, 0, 3, 0, 1, 2, 5, 4, 1, 0, 6, 0, 1, 3, 0, 4) = 6
Pool 16 ⇒ max(0, 3, 0, 7, 2, 5, 4, 2, 0, 6, 0, 0, 3, 0, 4, 5) = 7

And here is the output in matrix form:

```
7 7 5 7
8 8 5 7
8 8 6 7
8 8 6 7
```