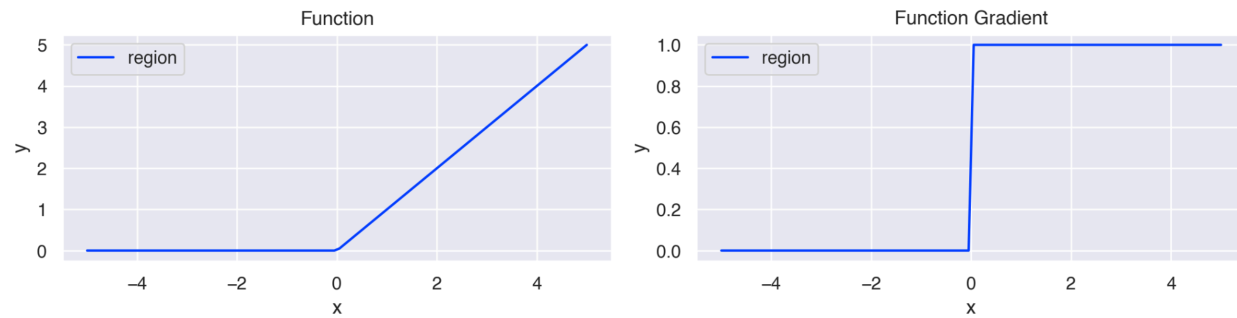
Homework #1 – Neural Network Overview – Activation Functions and Back Propagation

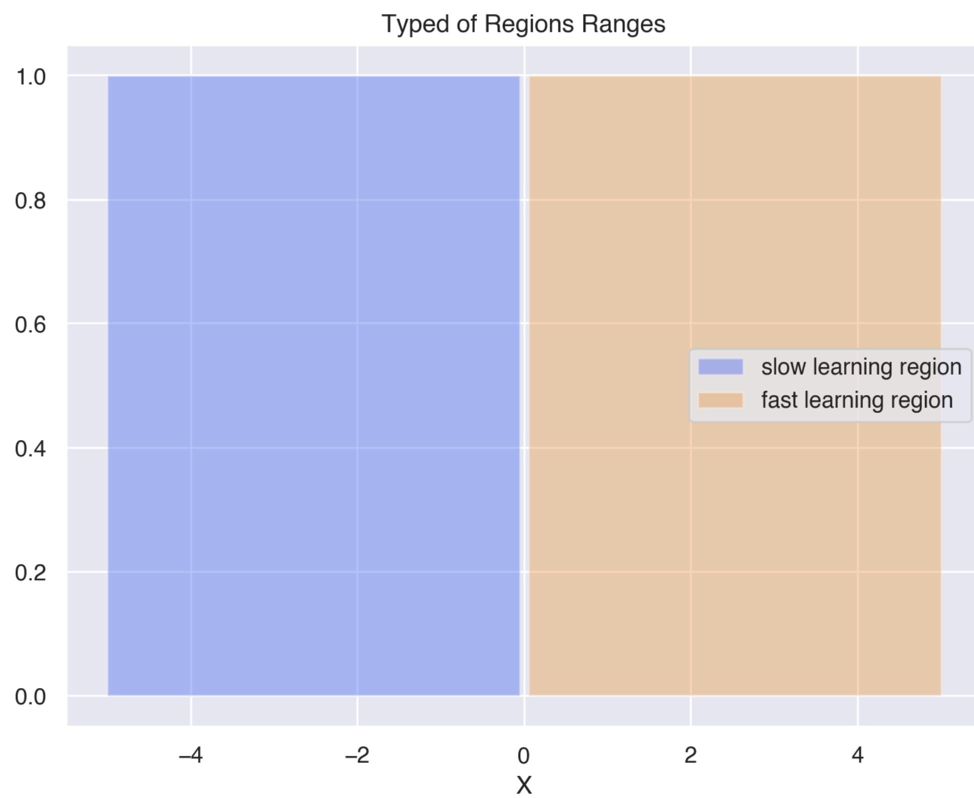
Ivan Liuliaev

Problem 1

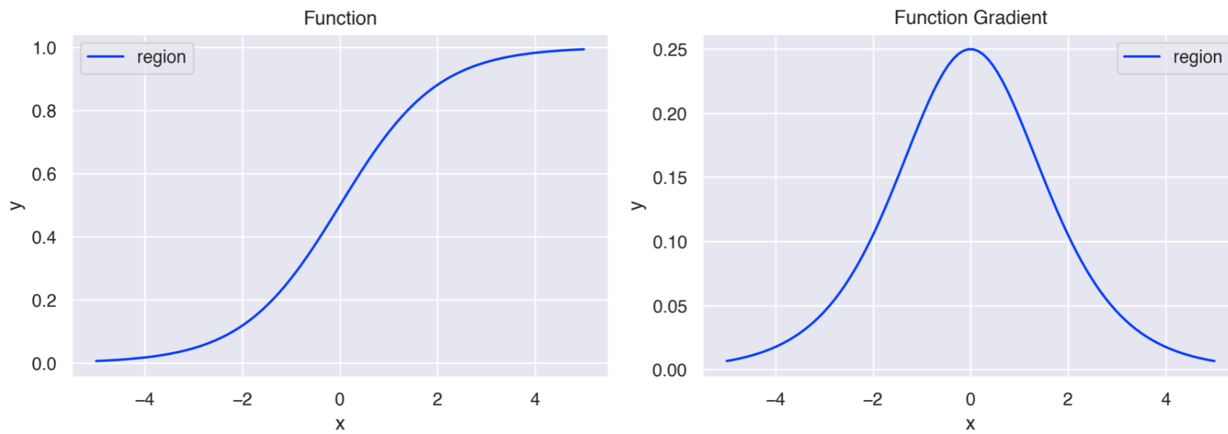
For each of the following activation functions, plot its gradient in the range from -5 to 5 of the input and then list the four types of regions.

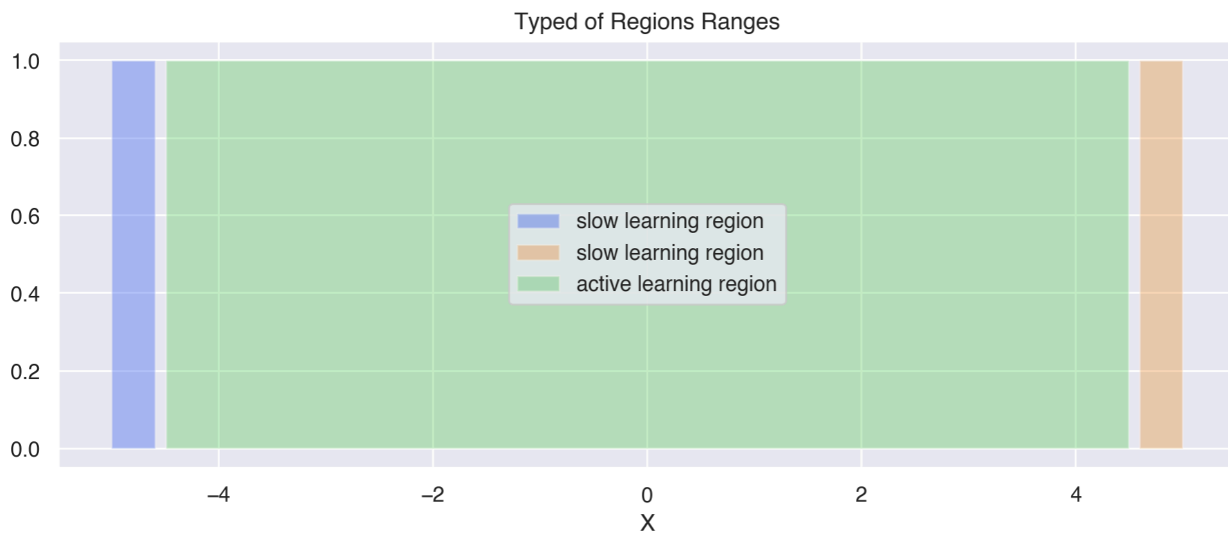
ReLU



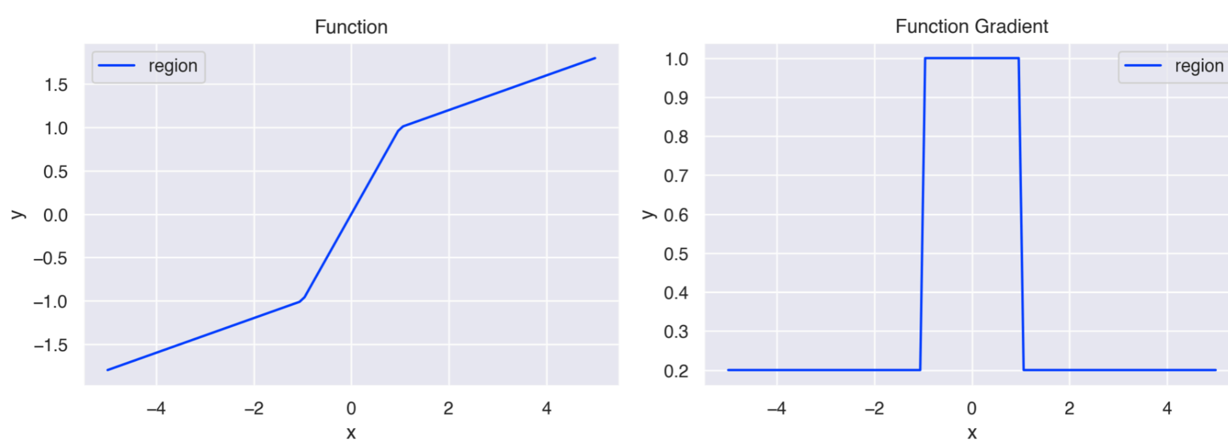


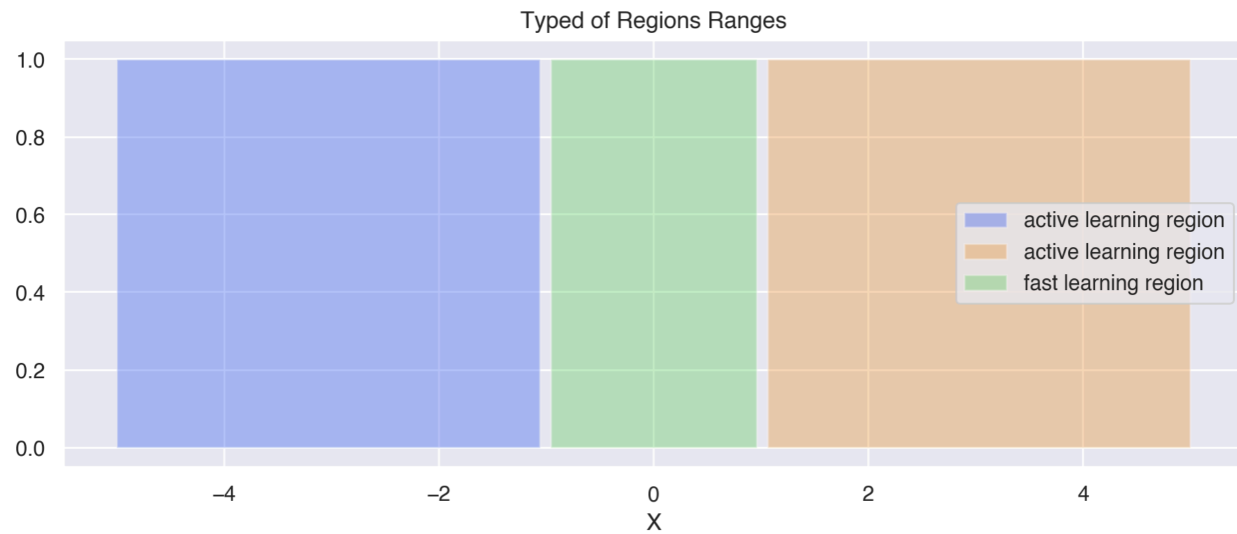
Logistic Sigmoid



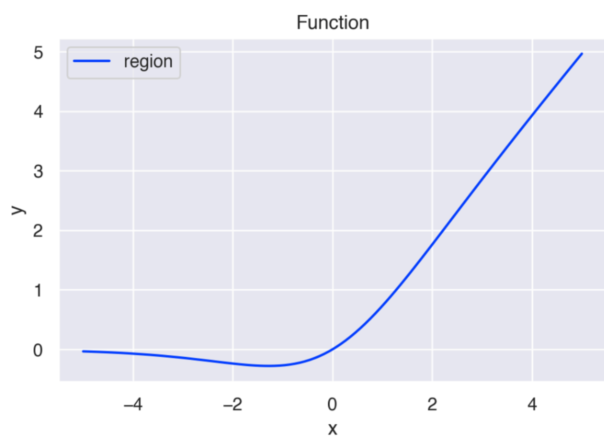


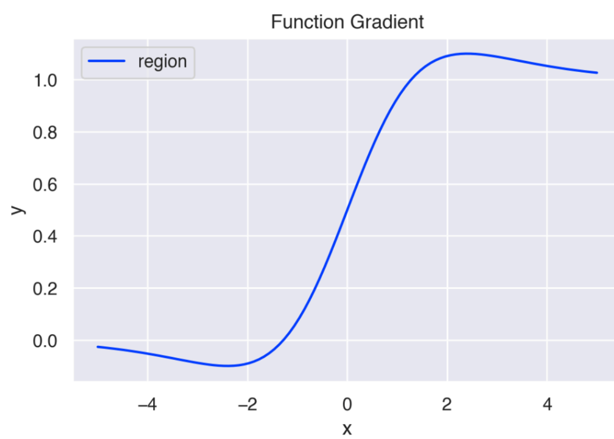
Piecewise Linear

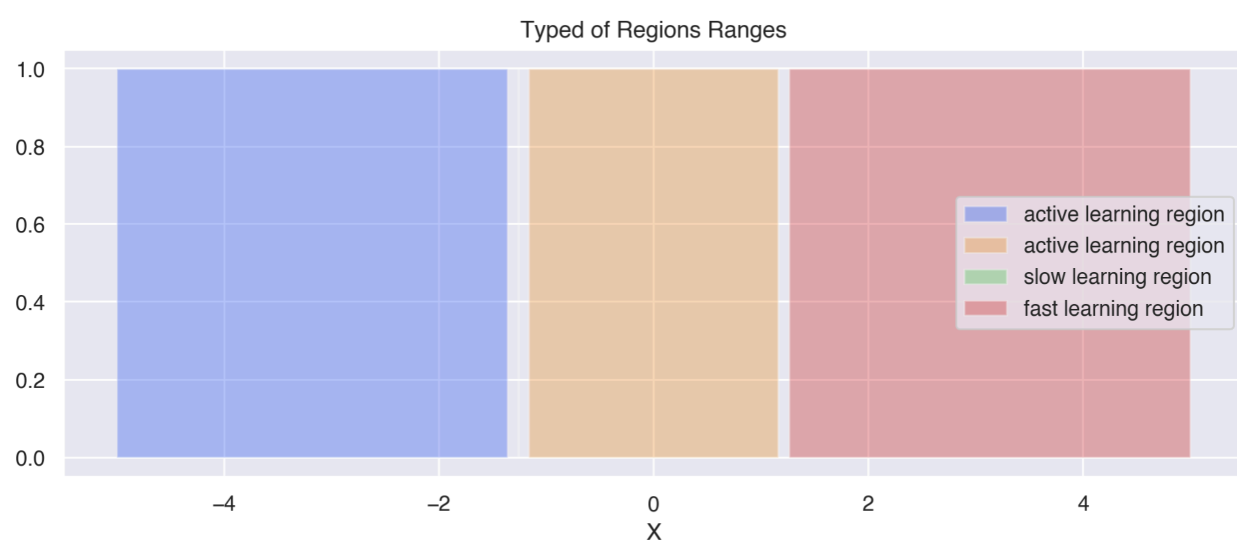



Typed of Regions Ranges
active learning region
active learning region
fast learning region
X

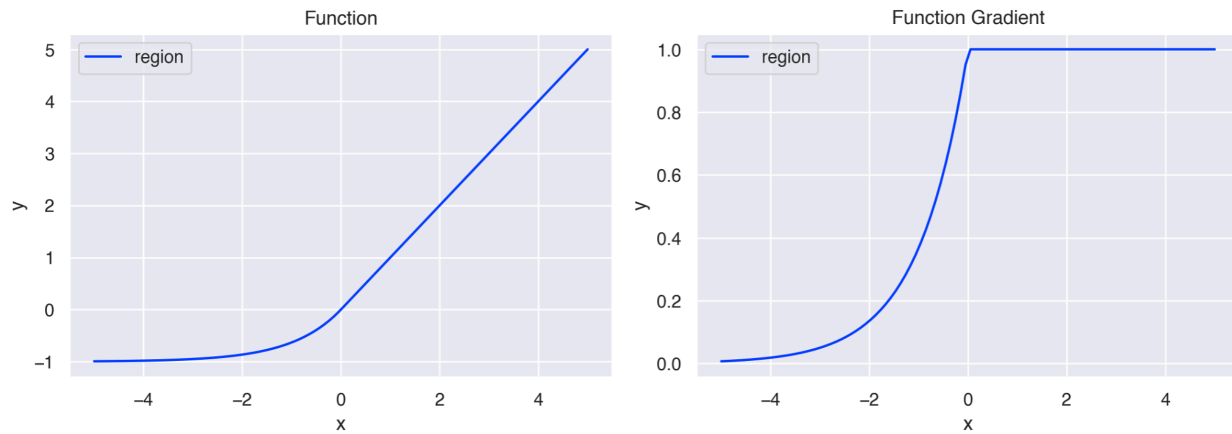
Swish

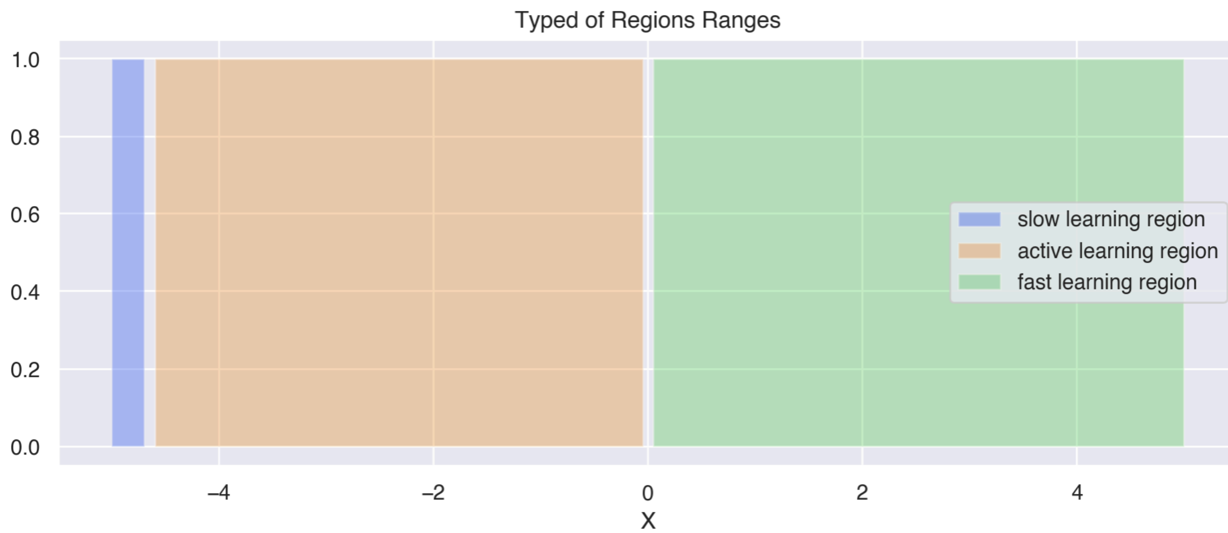

Function
region
y
x


Function Gradient
region
y
x

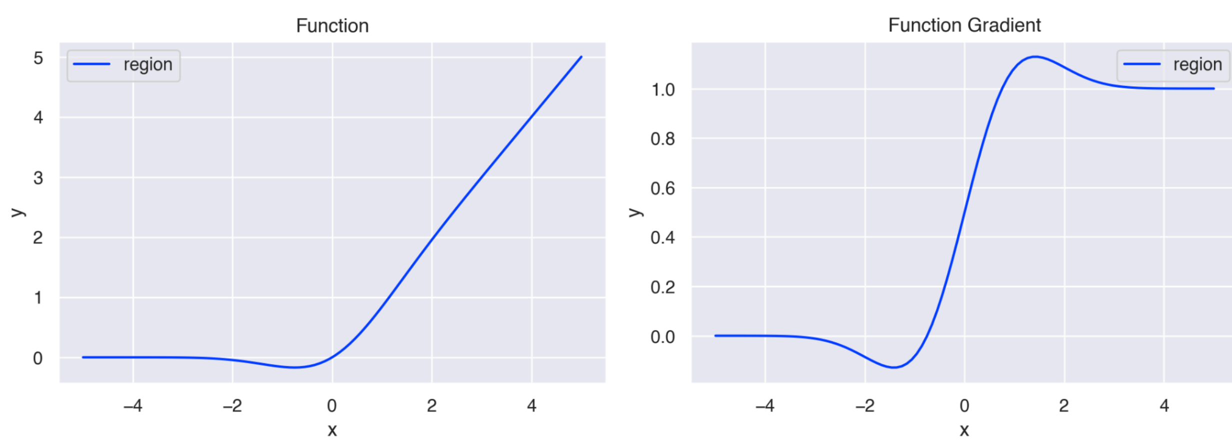

Typed of Regions Ranges
active learning region
active learning region
slow learning region
fast learning region
X

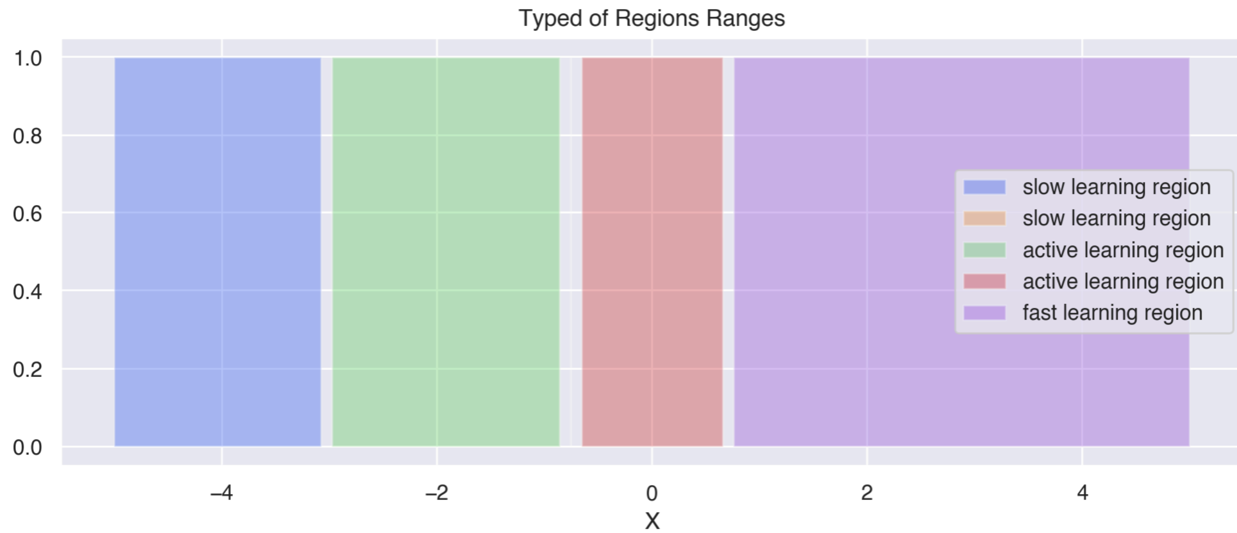
ELU


Function
Function Gradient
region
region
y
x
y
x


Typed of Regions Ranges
slow learning region
active learning region
fast learning region
X

GELU


Function
Function Gradient
region
region
y
x
y
x



Typed of Regions Ranges
1.0
0.8
0.6
0.4
0.2
0.0
−4
−2
0
2
4
X
slow learning region
slow learning region
active learning region
active learning region
fast learning region

Problem 2

2.1

```
Sample 1: x = [0. 0.], target = 0
  y_hat (sigmoid output): 0.3775
  Loss: 0.4741

Sample 2: x = [0. 1.], target = 1
  y_hat (sigmoid output): 0.6225
  Loss: 0.4741

Sample 3: x = [1. 0.], target = 1
  y_hat (sigmoid output): 0.6225
  Loss: 0.4741

Sample 4: x = [1. 1.], target = 0
  y_hat (sigmoid output): 0.3775
  Loss: 0.4741
```

2.2

```
Sample 1: x = [0. 0.], target = 0
  grad_w: [0. 0.]
  grad_b: 0.3775406687981454
  grad_W:
[[ 0.  0.]
 [-0. -0.]]
  grad_c: [ 0. -0.]

Sample 2: x = [0. 1.], target = 1
  grad_w: [-0.37754067 -0.        ]
  grad_b: -0.3775406687981454
  grad_W:
[[-0.         -0.37754067]
 [ 0.          0.        ]]
  grad_c: [-0.37754067  0.        ]
```

```
Sample 3: x = [1. 0.], target = 1
  grad_w: [-0.37754067 -0.        ]
  grad_b: -0.3775406687981454
  grad_W:
[[-0.37754067 -0.        ]
 [ 0.          0.        ]]
  grad_c: [-0.37754067  0.        ]

Sample 4: x = [1. 1.], target = 0
  grad_w: [0.75508134 0.37754067]
  grad_b: 0.3775406687981454
  grad_W:
[[ 0.37754067  0.37754067]
 [-0.75508134 -0.75508134]]
  grad_c: [ 0.37754067 -0.75508134]
```

Problem 3

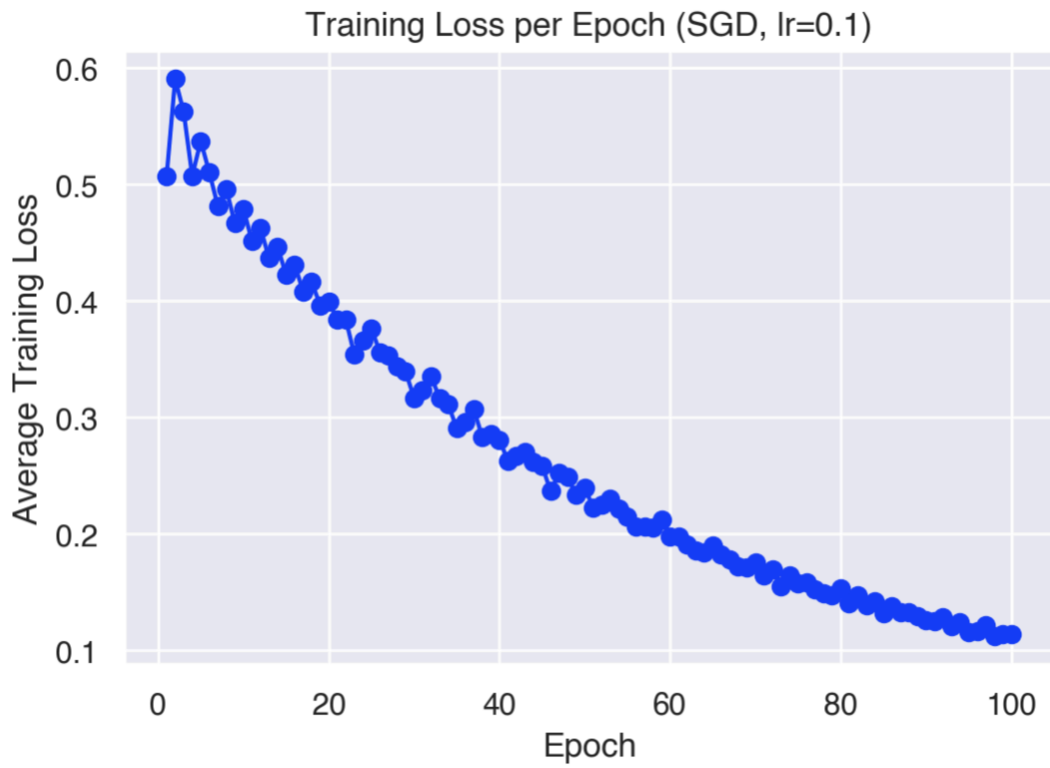
3.1

```
Sample 1: input [0. 0.], target 0.0
  a_hidden: [[ 0. -1.]]
  h (ReLU): [[0. 0.]]
  a_out: [[-0.5]]
  y_hat: 0.3775
  Loss: 0.4741

Sample 2: input [0. 1.], target 1.0
  a_hidden: [[1. 0.]]
  h (ReLU): [[1. 0.]]
  a_out: [[0.5]]
  y_hat: 0.6225
  Loss: 0.4741

Sample 3: input [1. 0.], target 1.0
  a_hidden: [[1. 0.]]
  h (ReLU): [[1. 0.]]
  a_out: [[0.5]]
  y_hat: 0.6225
  Loss: 0.4741

Sample 4: input [1. 1.], target 0.0
  a_hidden: [[2. 1.]]
  h (ReLU): [[2. 1.]]
  a_out: [[-0.5]]
  y_hat: 0.3775
  Loss: 0.4741
```

Verified.

3.2

```
=== (2) Training ===
Epoch  10: Avg Loss = 0.4794
Epoch  20: Avg Loss = 0.3996
Epoch  30: Avg Loss = 0.3168
Epoch  40: Avg Loss = 0.2809
Epoch  50: Avg Loss = 0.2393
Epoch  60: Avg Loss = 0.1973
Epoch  70: Avg Loss = 0.1756
Epoch  80: Avg Loss = 0.1528
Epoch  90: Avg Loss = 0.1258
Epoch 100: Avg Loss = 0.1137
```



Comment on the effectiveness of gradient descent
Gradient descent consistently reduces the loss with a reasonable speed.

3.3

```
Initial OAE: (0.36000022292137146, (tensor([0., 1.]), tensor([-0.2546,  0.7454])))
Trained OAE: (0.3100009262561798, (tensor([1., 0.]), tensor([1.2224, 0.2160])))
```

Problem 4 (Extra Credit)

4.1

Explanation

By replacing ReLU with JumpReLU small perturbations that do not push the pre-activation past θ. Essentially, when inputs are near the threshold, small adversarial perturbations are ignored rather than causing gradual changes.

4.2

```
JumpReLU OAE:
  Original sample: [1. 0.]
  Adversarial sample: [1.1849158  0.18277444]
  L2 perturbation: 0.26000064611434937
```

Explanation

Minimal (L2 measured) perturbation required to flip the classification is larger compared to a standard ReLU network. This indicates that the JumpReLU variant is less sensitive to small adversarial perturbations.