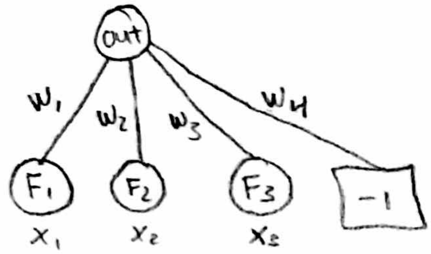
CS 540 HW 4 Solution

Problem 1

Initial Perceptron



$w_1 = w_2 = w_3 = w_0 = 0.1$

To update the weights, we use the perceptron learning rule.

$\Delta W_i = \alpha (y - O) x_i$

where α is the learning rate, set to $\alpha = 0.1$ as given in the problem, y is the true label / category of the example, O is the output of the perceptron, w is the weight vector and x is the feature vector.

- We only update the weights of the perceptron if the output is incorrect for the current example.
- To compute the output of this perceptron, we use the following formula:

$$out = step\left(\sum_i w_i x_i\right) \quad \text{where } step(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

i. $ex_1 = [1, 1, 0]$ category $= 1$

$\underline{out} = step(\overset{w_1}{0.1}(\overset{x_1}{1}) + \overset{w_2}{0.1}(\overset{x_2}{1}) + \overset{w_3}{0.1}(\overset{x_3}{0}) + \overset{w_4}{0.1}(\overset{x_4}{-1})) = step(0.1) = 1$

Output equals the category, so no update. $\boxed{w = [0.1, 0.1, 0.1, 0.1]}$

ii. $ex_2 = [1, 0, 1]$ category $= 0$

$\underline{out} = step(0.1(1) + 0.1(0) + 0.1(1) + 0.1(-1)) = 1$

Update weights since category $\neq$ output

new $W_1 = W_1 + \Delta W_1 = 0.1 + 0.1(0-1)1 = 0.1 - 0.1 = 0$

new $W_2 = W_2 + \Delta W_2 = 0.1 + 0.1(0-1)0 = 0.1$

new $W_3 = W_3 + \Delta W_3 = 0.1 + 0.1(0-1)1 = 0$

new $W_4 = W_4 + \Delta W_4 = 0.1 + 0.1(0-1)(-1) = 0.2$

Problem 1 continued

ii continued

The new weight vector is $\boxed{w = [0, 0.1, 0, 0.2]}$

continuing from previous step.

iii $ex_3 = [0, 1, 1]$ category = 1

$$\underline{out} = step(\underset{w_1}{0}(\underset{x_1}{0}) + \underset{w_2}{0.1}(\underset{x_2}{1}) + \underset{w_3}{0}(\underset{x_3}{1}) + \underset{w_4}{0.2}(\underset{x_4}{-1})) = 0$$

Update weights

$$\begin{aligned}
new\,w_1 &= w_1 + \Delta w_1 = 0 + 0.1(1-0)0 = 0 \\
new\,w_2 &= w_2 + \Delta w_2 = 0.1 + 0.1(1-0)1 = 0.2 \\
new\,w_3 &= w_3 + \Delta w_3 = 0 + 0.1(1-0)1 = 0.1 \\
new\,w_4 &= w_4 + \Delta w_4 = 0.2 + 0.1(1-0)(-1) = 0.1
\end{aligned}$$

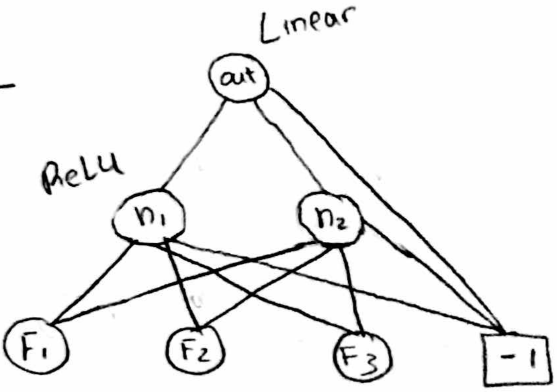
$$\boxed{w = [0, 0.2, 0.1, 0.1]}$$

iv $exNew = [0, 1, 0]$ category = ?

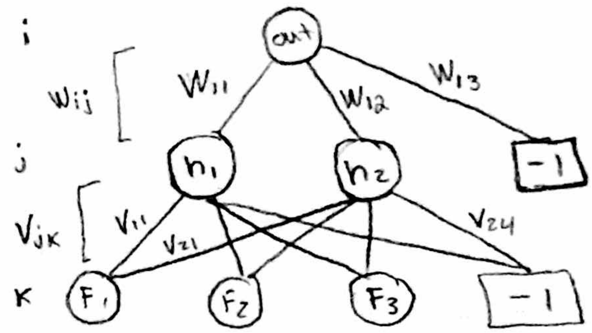
$$out = step(0(0) + 0.2(1) + 0.1(0) + 0.1(-1)) = 1$$

Problem 2

i



or



The calculations for the above are the same despite different layouts. I use weight labels as defined on right. All weights are initially 0.1.

Activation Functions

$$ReLu(x) = \begin{cases} 0 & \text{if } x \le 0 \\ x & \text{if } x > 0 \end{cases}$$

$$Linear(x) = x$$

ii example = [1, 1, 0] category = 1

$$h_j = ReLu\left(\sum_K V_{jK} X_K\right)$$

$$\underline{h_1} = ReLU(V_{11}X_1 + V_{12}X_2 + V_{13}X_3 + V_{14}(-1))$$
$$= ReLU(0.1(1) + 0.1(1) + 0.1(0) + 0.1(-1))$$
$$= \boxed{0.1}$$

$$\underline{h_2} = ReLU(V_{21}X_1 + V_{22}X_2 + V_{23}X_3 + V_{24}(-1))$$
$$= ReLU(0.1(1) + 0.1(1) + 0.1(0) + 0.1(-1))$$
$$= \boxed{0.1}$$

$$\underline{out} = Linear\left(\sum_j W_{1j} h_j\right)$$

$$= Linear(W_{11}h_1 + W_{12}h_2 + W_{13}h_3)$$
$$= Linear(0.1(0.1) + 0.1(0.1) + 0.1(-1))$$
$$= Linear(0.02 - 0.1)$$
$$= Linear(-0.08)$$
$$= \boxed{-0.08}$$

Problem 2 continued

iii

1. Compute deviation for output unit using:

$$\delta_{out} = F'(net_{out}) \cdot (y - o)$$

where F' is the derivative of the activation function used at the output unit, net_{out} is the weighted sum of edges going into the output unit, y is the true category of the example and o is the output from the output unit.

The derivative of the linear output function is 1.

$$\delta_{out} = F'(-0.08) \cdot (1 - (-0.08))$$
$$= 1(1.08) = \boxed{1.08}$$

2. Compute deviations for hidden units using:

$$\delta_j = F'(net_j) \cdot \left(\sum_i W_{ij} \cdot \delta_{out}\right)$$

where F' is the derivative of the ReLU activation function at the hidden units.

$$F'(x) = \begin{cases} 0 & \text{if } x \le 0 \\ 1 & \text{otherwise} \end{cases}$$

$$\delta_1 = F'(0.1) \cdot (W_{11} \delta_{out}) = 1(0.1)(1.08) = 0.108$$
$$\delta_2 = F'(0.1) \cdot (W_{12} \delta_{out}) = 1(0.1)(1.08) = 0.108$$

3. Update weights using the following rules

$$\Delta W_{ij} = \alpha \cdot \delta_{out} \cdot h_j$$
$$\Delta V_{jk} = \alpha \cdot \delta_j \cdot x_k$$

where $\alpha = 0.1$ is the learning rate, h_j is output from hidden unit j (computed on prev. page) and x_k is the k^{th} element of example x.

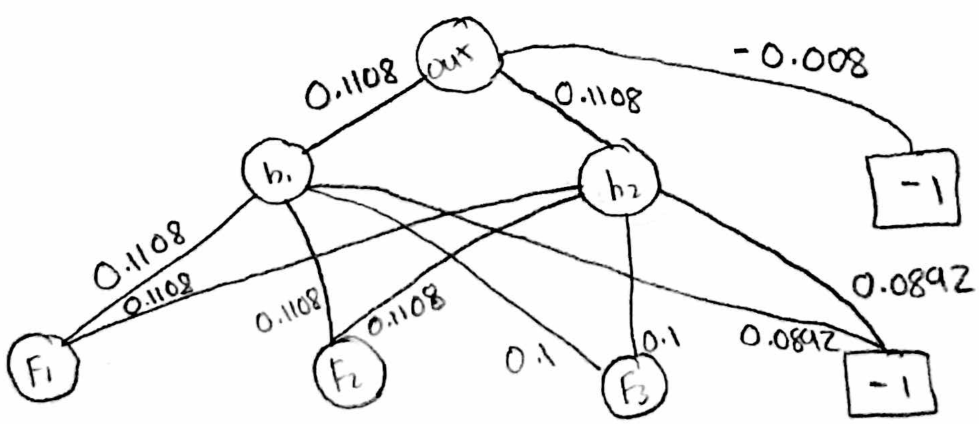
Problem 2 continued

iii continued

$$\Delta W_{11} = \alpha \cdot \delta_{out} \cdot h_1 = (0.1)(1.08)(0.1) = 0.0108$$
$$\Delta W_{12} = \alpha \cdot \delta_{out} \cdot h_2 = (0.1)(1.08)(0.1) = 0.0108$$
$$\Delta W_{13} = \alpha \cdot \delta_{out} \cdot (-1) = (0.1)(1.08)(-1) = -0.108$$

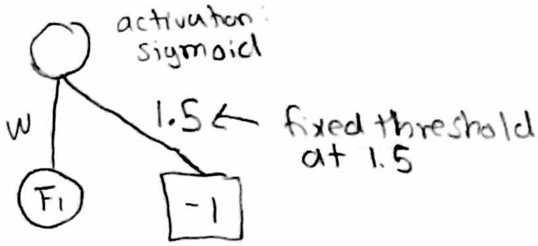
$$\Delta V_{11} = \alpha \cdot \delta_1 \cdot x_1 = (0.1)(0.108)(1) = 0.0108$$
$$\Delta V_{21} = \alpha \cdot \delta_2 \cdot x_1 = (0.1)(0.108)(1) = 0.0108$$
$$\Delta V_{12} = \alpha \cdot \delta_1 \cdot x_2 = (0.1)(0.108)(1) = 0.0108$$
$$\Delta V_{22} = \alpha \cdot \delta_2 \cdot x_2 = (0.1)(0.108)(1) = 0.0108$$
$$\Delta V_{13} = \alpha \cdot \delta_1 \cdot x_3 = (0.1)(0.108)(0) = 0$$
$$\Delta V_{23} = \alpha \cdot \delta_2 \cdot x_3 = (0.1)(0.108)(0) = 0$$
$$\Delta V_{14} = \alpha \cdot \delta_1 \cdot (-1) = (0.1)(0.108)(-1) = -0.0108$$
$$\Delta V_{24} = \alpha \cdot \delta_2 \cdot (-1) = (0.1)(0.108)(-1) = -0.0108$$

To compute the updated weights, we must add the deltas above to the current weights. All current weights are 0.1. The network below is the final network.



Problem 3

The perceptron looks like:



For each w in $\{-4, -2, -1, 0, 1, 2, 4\}$, we must compute the total error of the network, defined as $\sum_i (\text{category}_{ex_i} - \text{output}_{ex_i})^2$

For w = -4

ex₁'s F₁ value

$$\text{output}_{ex_1} = \text{Sigmoid}(w x + 1.5(-1))$$
$$= \text{Sigmoid}(-4(1) + 1.5(-1))$$
$$= \text{Sigmoid}(-5.5)$$
$$= \frac{1}{1+e^{-(-5.5)}} \approx 0.00407$$

$$\text{output}_{ex_2} = \text{sigmoid}(-4(2) + 1.5(-1)) \approx 0.00007$$
$$\text{output}_{ex_3} = \text{Sigmoid}(-4(3) + 1.5(-1)) \approx 1.4 \times 10^{-6}$$

Total error for w = -4

$$= (1 - 0.00407)^2 + (0 - 0.00007)^2 + (0 - 1.4 \times 10^{-6})^2$$

category for ex_1 · output for ex_1 · category for ex_2 · output ex_2 · category for ex_3 · output ex_3

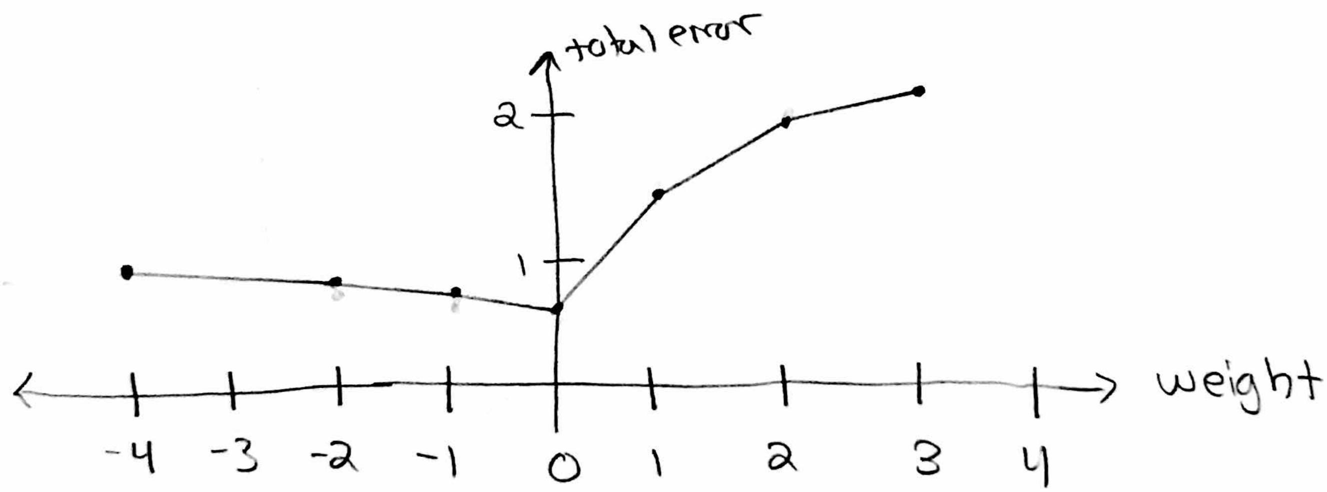
≈ 0.9919

Now compute total error for remaining w's.

Problem 3 continued

weight	total error
-4	0.9919
-2	0.9423
-1	0.8550
0	0.7350
1	1.4433
2	1.9747
4	2.0027

Graph (approximately...)



Problem 4

i

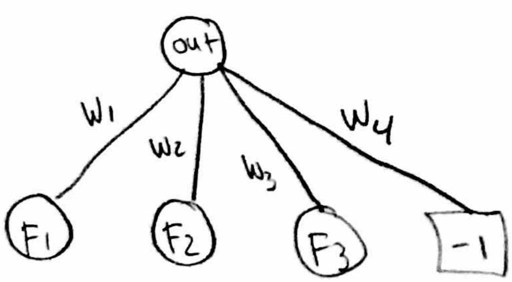
	F_1	F_2	F_3	category
ex_1	$F1(ex_1)$	$F2(ex_1)$	$F3(ex_1)$	1
ex_2	$F_1(ex_2)$	$F_2(ex_2)$	$F_3(ex_2)$	0
ex_3	$F_1(ex_3)$	$F_2(ex_3)$	$F_3(ex_3)$	1

$F1(ex_1)$ = similarity between ex_1 and ex_1 = 3
$F2(ex_1)$ = " ex_2 and ex_1 = 1
$F3(ex_1)$ = " ex_3 and ex_1 = 1

Filling out the rest of the table,

	F_1	F_2	F_3	category
ex_1	3	1	1	1
ex_2	1	3	1	0
ex_3	1	1	3	1

ii



All weights initialized to 0.1.

The weight update rule is

$$w_i = w_i + \underbrace{[\alpha\,(\text{category}(ex) - \text{output}(ex))\,\text{feature}_i(ex)]}_{\text{This is the perceptron learning rule. If category = output, this term will be 0 and not factor into the equation.}} - \alpha \cdot \lambda \cdot w_i$$

This is the perceptron learning rule. If category = output, this term will be 0 and not factor into the equation.

Problem 4 continued

ii continued

ex_1 [3, 1, 1] category = 1

$$out = step(w_1 x_1 + w_2 x_2 + w_3 x_3 + w_4(-1))$$
$$= step(0.1(3) + 0.1(1) + 0.1(1) + 0.1(-1))$$
$$= step(0.4) = 1$$

The category matches the output. Update weights

$$w_i = w_i - \alpha \cdot \lambda \cdot w_i$$
$$w_1 = w_2 = w_3 = w_4 = 0.1 - (0.1 \cdot 0.2 \cdot 0.1) = \boxed{0.098}$$

ex_2 [1, 3, 1] category = 0

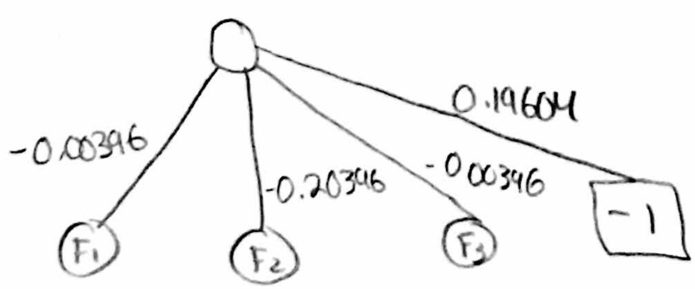
$$out = step(0.098(1) + 0.098(3) + 0.098(1) + 0.098(-1))$$
$$= step(0.392) = 1$$

Category does not match output. Update using

$$w_i = w_i + [\alpha(category(ex) - output(ex))\, feature_i(ex)] - \alpha \cdot \lambda \cdot w_i$$

$$w_1 = 0.098 + [0.1(0-1)1] - (0.1)(0.2)(0.098) = -0.00396$$
$$w_2 = 0.098 + [0.1(0-1)3] - (0.1)(0.2)(0.098) = -0.20396$$
$$w_3 = 0.098 + [0.1(0-1)1] - (0.1)(0.2)(0.098) = -0.00396$$
$$w_4 = 0.098 + [0.1(0-1)(-1)] - (0.1)(0.2)(0.098) = 0.19604$$

Perceptron after ex2:



Problem 4 continued

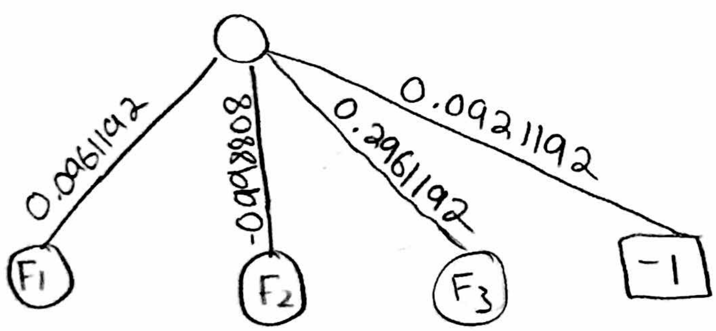
ii continued

ex3 [1, 1, 3] category = 1

$$out = step(-0.00396(1) - 0.20396(1) - 0.00396(3) - 0.19604(-1))$$
$$= step(-0.41584) = 0$$

$$w_1 = -0.00396 + [0.1(1-0)\,1] - (0.1)(0.2)(-0.00396) = 0.0961192$$
$$w_2 = -0.20396 + [0.1(1-0)\,1] - (0.1)(0.2)(-0.20396) = -0.998808$$
$$w_3 = -0.00396 + [0.1(1-0)\,3] - (0.1)(0.2)(-0.00396) = 0.2961192$$
$$w_4 = -0.19604 + [0.1(1-0)(-1)] - (0.1)(0.2)(-0.19604) = 0.0921192$$

Perceptron after ex3



iii

ex New = [2, 0, 2] ← after applying same transformation as in part i

out =

$$step(0.0961192(2) - 0.998808(0) + 0.2961192(2) + 0.0921192(-1))$$
$$= step(0.6923576) = 1$$