# Problem 1

Suppose you want to find the value of x that minimizes the function, $f(x) = x^2 + 4x$.

The derivative of this function is: $f'(x) = 2x + 4$.

a) If x is initialized to 0, and the learning rate is 1.0, what is the new value of x after 1 iteration of gradient descent?

b) It is possible that gradient descent could find a local minimum after a single iteration, depending on the choice of learning rate. Find a learning rate such that the function is minimized after 1 iteration of gradient descent when x is initialized to 0.

# Problem 1

a) $f'(0) = 2(0) + 4 = 4$

$x = 0 - 1.0*4 = -4$

# Problem 1

b) First, find minimum:

$f'(x) = 0$ when $x = -2$

$f(x)$ is minimized at $x=-2$

Now, find $\eta$ such that the gradient descent update would result in $x=-2$:

$-2 = 0 - \eta*4$

$\eta = ½$

# Problem 2

Suppose your dataset contains 3 binary features and you learn a perceptron with the following weights:

- $w_1$ = -1.0, $w_2$ = 0.5, $w_3$ = 1.0, b = -0.3

Determine the predictions perceptron makes:

| $x_1$ | $x_2$ | $x_3$ | y |
|---|---|---|---|
| 1 | 1 | 1 | |
| 0 | 1 | 1 | |
| 0 | 0 | 0 | |
| 1 | 0 | 0 | |
| 0 | 1 | 0 | |

# Problem 2

| $x_1$ | $x_2$ | $x_3$ | $y$ |
|---|---|---|---|
| 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 |
| 0 | 0 | 0 | -1 |
| 1 | 0 | 0 | -1 |
| 0 | 1 | 0 | 1 |

Explanation:

- $-1.0 + 0.5 + 1.0 - 0.3 = 0.2 > 0$
- $0.5 + 1.0 - 0.3 = 1.2 > 0$
- $-0.3 = -0.3 < 0$
- $-1.0 - 0.3 = -1.3 < 0$
- $0.5 - 0.3 = 0.2 > 0$

# Problem 3

Suppose you train a classifier on the instances below, using the rule that y=1 if $\mathbf{w}^T\mathbf{x} \geq 0$ (e.g., a perceptron). Come up with weights **w** that would correctly classify the five instances.

| **x1** | **x2** | **x3** | **y** |
|---|---|---|---|
| 1.0 | -0.5 | 1.0 | 1 |
| 1.0 | 1.0 | 1.0 | -1 |
| -1.0 | -1.0 | 0.0 | 1 |
| 0.0 | 1.0 | 5.0 | -1 |
| 3.0 | 1.0 | 2.0 | 1 |

# Problem 3

Answer(s):

| **w1** | **w2** | **w3** |
|---|---|---|
| 1 | -1 | -1 |
| 0.5 | -1 | 0 |
| 1.5 | -2 | -1 |

There are many possible answers. One way to solve this would be to run through the perceptron algorithm on paper, but you could also solve this by trying different values, seeing the mistakes they would make, and adjusting them to fix those mistakes. For example, you might start by setting $w_2$ to a negative value because y=0 whenever $w_2$ is positive, then adjust the other weights so that they give the correct y values.