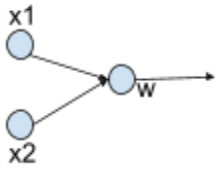
Suppose data is in two-dimensional space.



The objective for a single datapoint is given by:

$$f(w_1, w_2, x, y) = (w^T x - y)^2 = ((w_1, w_2)^T (x_1, x_2) - y)^2 = (w_1 x_1 + w_2 x_2 - y)^2$$

The objective for all datapoints (also known as the empirical risk) is given by

$$L = \sum_{i=0}^{n-1} f(w_1, w_2, x_i, y_i)$$

We want to find w_1 and w_2 that minimize L above. We can do this with a simple gradient descent procedure. The gradient descent approach is to start with random initial values for all weights and modify each weight by moving it in the direction of the negative derivative.

1. Initialize weights w1 and w2 to a random value between [-.01,+.01].
2. Calculate initial_objective $L = f(w_1, w_2, x\ , y\ )$
3. Set previous_objective to L+10
4. while(previous_objective - L > .01):
    a. Set previous_objective to L
    b. Update w_1 by moving it slightly in the direction of the negative derivative. We have $df/dw_1 = 2(w_1 x_1 + w_2 x_2 - y)\ x_1$. Our update for w_1 will be $w_1 = w_1 - \eta df/dw_1$.
    c. Update w_2 by moving it slightly in the direction of the negative derivative. We have $df/dw_2 = 2(w_1 x_1 + w_2 x_2 - y)\ x_2$. Our update for w_2 will be $w_2 = w_2 - \eta df/dw_2$.
    d. We can write the derivative of a function with respect to a vector as the derivative of each component of the vector. So $df/dw = (df/dw_1, df/dw_2) =$ $(2(w_1 x_1 + w_2 x_2 - y)\ x_1, 2(w_1 x_1 + w_2 x_2 - y)\ x_2) = 2(w_1 x_1 + w_2 x_2 - y)(x_1, x_2)$
    e. We can now write our update rule simply as $w = w - \eta df/dw$
    f. Recalculate objective L