1 Widrow-Hoff Algorithm

First let’s review the Widrow-Hoff algorithm that was covered from last lecture:

**Algorithm 1: Widrow-Hoff Algorithm**

| Initialize parameter $\eta > 0$, $w_1 = 0$ |
| for $t = 1 \ldots T$ |
| get $x_t \in \mathbb{R}^n$ |
| predict $\hat{y}_t = w_t \cdot x_t \in \mathbb{R}$ |
| observe $y_t \in \mathbb{R}$ |
| update $w_{t+1} = w_t - \eta (w_t \cdot x_t - y_t) \cdot x_t$ |

And we define the loss functions as $L_A = \sum_{t=1}^{T} (\hat{y}_t - y_t)^2$. And $L_u = \sum_{t=1}^{T} (u \cdot x_t - y_t)$. What we want is

$$L_A \leq \min_u L_u + \text{small}$$

There are 2 goals in choosing the update function to be $w_{t+1} = w_t - \eta (w_t \cdot x_t - y_t) \cdot x_t$: (1) Want loss of $w_{t+1}$ on $x_t, y_t$ to be small. This means we want to minimize $(w_{t+1} \cdot x_t - y_t)^2$ (2) Want $w_{t+1}$ close to $w_t$ so that we do not forget everything we learnt so far. And this means we want to minimize $\|w_{t+1} - w_t\|^2$.

Therefore to sum up, we want to minimize

$$\eta (w_{t+1} \cdot x_t - y_t)^2 + \|w_{t+1} - w_t\|^2$$

If we take the derivative of the above equation and set it to zero, we have

$$w_{t+1} = w_t - \eta (w_{t+1} \cdot x_t - y_t) \cdot x_t$$

Instead of solving $w_{t+1}$, we approximate the term $w_{t+1}$ inside the parenthesis and change it to $w_t$. The reason we can do this is because $w_{t+1}$ does not change much from $w_t$. Therefore we have

$$w_{t+1} = w_t - \eta (w_t \cdot x_t - y_t) \cdot x_t$$

which is the update function stated in the algorithm.

Now let’s state a theorem:

**Theorem 1.1** If we assume on every round $t$, $\|x_t\|_2 \leq 1$, then:

$$L_{WH} \leq \min_{u \in \mathbb{R}^n} \left[ \frac{L_u}{1 - \eta} + \frac{\|u\|^2}{\eta} \right]$$
From this theorem, we have ∀u:

\[ LW_H \leq \frac{1}{1 - \eta} \cdot L_u + \frac{\|u\|^2}{\eta} \]

If we divide T on both side, we have:

\[ \frac{L_{WH}}{T} \leq \frac{1}{1 - \eta} \cdot \frac{L_u}{T} + \frac{\|u\|^2}{\eta T} \]

The term \( \frac{\|u\|^2}{\eta T} \) goes to 0 when T gets large. And we can choose \( \eta \) small enough to make \( \frac{1}{1 - \eta} \) to be close to 1. Therefore we have the rate that the algorithm is suffering loss is close to rate that \( L_u \) is suffering loss.

Now let’s prove the theorem:

**Proof**: Pick any \( u \in \mathbb{R}^n \). First let’s define some terms:

\[
\begin{align*}
\Phi_t &= \|w_t - u\|^2_2 \quad \text{(measure of progress)} \\
l_t &= w_t \cdot x_t - y_t = \hat{y}_t - y_t \quad \text{(notice } l_t^2 \text{ is the loss of WH on round } t) \\
g_t &= u \cdot x_t - y_t \quad \text{(} g_t^2 \text{ is the loss of } u \text{ on round } t) \\
\Delta_t &= \eta(w_t \cdot x_t - y_t) \cdot x_t = \eta l_t x_t \\
w_{t+1} &= w_t - \Delta_t
\end{align*}
\]

Our main claim is that the change of potential is:

\[ \Phi_{t+1} - \Phi_t \leq -\eta l_t^2 + \frac{\eta}{1 - \eta} \cdot g_t^2 \quad (1) \]

This shows that \( l_t^2 \) tends to drive potential down while \( g_t^2 \) tends to drive potential up.

Now assume (1) holds. Notice that total change in potential should be non-negative. And also we initialize \( w_1 = 0 \). So we have the following inequality:

\[
-\|u\|^2_2 = -\Phi_1 \leq \Phi_{T+1} - \Phi_1 = \Phi_{T+1} - \Phi_t + \Phi_t - \Phi_{t-1} + \cdots + \Phi_2 - \Phi_1 = \sum_{t=1}^{T} (\Phi_{t+1} - \Phi_t) \leq \sum_{t=1}^{T} [-\eta l_t^2 + \frac{\eta}{1 - \eta} g_t^2] = -\eta \sum_t l_t^2 + \frac{\eta}{1 - \eta} \sum_t g_t^2 = -\eta LW_H + \frac{\eta}{1 - \eta} L_u
\]

Now we solve for \( LW_H \), we get

\[ LW_H \leq \frac{1}{1 - \eta} \cdot L_u + \frac{\|u\|^2}{\eta} \]
And since this inequality holds for all \( u \), we have:

\[
L_{WH} \leq \min_{u \in \mathbb{R}} \left[ \frac{1}{1 - \eta} L_u + \frac{\|u\|^2}{\eta} \right]
\]

which is the theorem.

Now let’s go back to prove (1):

\[
\Phi_{t+1} - \Phi_t = \|w_{t+1} - u\|^2 - \|w_t - u\|^2
= \|w_t - u - \Delta_t\|^2 - \|w_t - u\|^2
= \|\Delta_t\|^2 - 2(w_t - u) \cdot \Delta_t + \|w_t - u\|^2 - \|w_t - u\|^2
= \|\Delta_t\|^2 - 2(w_t - u) \cdot \Delta_t
= \|\Delta\|^2 - 2(w - u) \cdot \Delta \quad \text{(dropping subscript} \ t \ \text{since it doesn’t affect the proof)}
\]

\[
= \eta l^2 \|x\|^2 - 2\eta l x \cdot (w - u)
= \eta l^2 \|x\|^2 - 2\eta l (w \cdot x - u \cdot x - y + y)
= \eta l^2 \|x\|^2 - 2\eta l [(w \cdot x - y) - (u \cdot x - y)]
= \eta l^2 \|x\|^2 - 2\eta l^2 + 2\eta l g
\]

\[
\leq \eta l^2 - 2\eta l^2 + 2\eta l g \quad (\|x\|^2 \leq 1)
\]

\[
\leq (\eta^2 - 2\eta)l^2 + \frac{2\eta[\frac{g^2}{1 - \eta} + l^2(1 - \eta)]}{2} \quad (ab \leq \frac{a^2 + b^2}{2})
\]

\[
= (\eta^2 - 2\eta)^2 + \eta \left[ \frac{g^2}{1 - \eta} + l^2(1 - \eta) \right]
\]

\[
= -\eta l^2 + \frac{\eta}{1 - \eta} g^2
\]

2 Families of Online Algorithm

The two goals of the learning algorithm are minimizing the loss of \( w_{t+1} \) on \( x_t \) and \( y_t \), and minimizing the distance between \( w_{t+1} \) and \( w_t \). So to generalize, we are trying to minimize

\[
\eta L(w_{t+1}, x_t, y_t) + d(w_{t+1}, w_t)
\]

So if we use the Euclidean norm as our distance measurement, then the above function becomes:

\[
\eta L(w_{t+1}, x_t, y_t) + \|w_t - w_{t+1}\|^2
\]

So if we try to optimize the above function, we have the update equation:

\[
w_{t+1} = w_t - \eta \nabla w L(w_{t+1}, x_t, y_t)
\approx w_t - \eta \nabla w L(w_t, x_t, y_t)
\]

Notice that we use \( w_t \) to approximate \( w_{t+1} \) when we calculate \( w_{t+1} \). This is called the Gradient Descent Algorithm.
Alternatively, we can use relative entropy as a measure of distance. So $d(w_t, w_{t+1}) = RE(w_t \| w_{t+1})$. Now we can have the update function as

$$w_{t+1,i} = \frac{w_{t,i} \cdot \exp(\eta \frac{\partial L(w_{t+1}, x_t, y_t)}{\partial w_i})}{Z_t}$$

This is called the Exponentiated Gradient Algorithm, or “EG” algorithm. We need to change the norm: $\|x_t\|_\infty \leq 1$ and $\|u\|_1 = 1$. It’s also possible to prove a bound on this update equation, but we skip it in this class.

### 3 Online Algorithm in a Batch Setting

We can modify the online algorithms slightly so that we can use them in the batch learning settings. Let’s take a look at one example in a linear regression setting. In a linear regression setting, training and test samples are drawn i.i.d from a fixed distribution $\mathcal{D}$. So we have $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ where $(x_i, y_i) \sim \mathcal{D}$. Our goal is to find $v$ with low risk, where risk is defined to be

$$R_v = E_{(x,y) \sim \mathcal{D}}[(v \cdot x - y)^2]$$

We want to find $v$ such that $R_v$ is small compared to $\min_u R_u$.

Now we can apply WH algorithm to the data as follows:

1. run WH on $(x_1, y_1), \ldots, (x_m, y_m)$, and calculate $w_1, w_2, \ldots, w_m$.

2. Combine the vectors:

$$v = \frac{1}{m} \sum_{t=1}^{m} w_t$$

and output $v$. We choose to output the average of all the $w_t$’s because we can prove something theoretically good about it, which is not necessarily the case for the last vector $w_m$.

Now let’s state another theorem:

**Theorem 3.1**

$$E_S[R_v] \leq \min_{u \in \mathbb{R}^n} \left\{ \frac{R_u}{1 - \eta} + \frac{\|u\|^2}{\eta m} \right\}$$

If we divide $T$ on both side of the equation above and if $\eta$ is chosen to be small, we can see that $\frac{R_v}{T}$ will be close to $\frac{R_u}{T}$ when $T$ is large. **Proof:**

There are three observations needed in the proof:

1. Let $x, y$ be a random test example from $\mathcal{D}$. Then we have

$$\left( v \cdot x - y \right)^2 \leq \frac{1}{m} \sum_{t=1}^{m} \left( w_t \cdot x_t - y \right)^2$$
Proof for (1):

\[(v \cdot x - y)^2 = \left[ \frac{1}{m} \sum_{t=1}^{m} w_t \cdot x - y \right]^2 \]
\[= \left[ \frac{1}{m} \sum_{t=1}^{m} w_t \cdot x \right] - y \right]^2 \]
\[= \left[ \frac{1}{m} \sum_{t=1}^{m} (w_t \cdot x - y) \right]^2 \]
\[\leq \frac{1}{m} \sum_{t} (w_t \cdot x - y)^2 \quad \text{(convexity of } f(x) = x^2) \]

(2):

\[E[(u \cdot x_t - y_t)^2] = E[(u \cdot x - y)^2] \]

The above expectation is with respect to the random choice of \((x_1, y_1), \ldots, (x_m, y_m)\) and \((x, y)\). This is because \((x_t, y_t)\) and \((x, y)\) are from the same distribution.

(3):

\[E[(w_t \cdot x_t - y_t)^2] = E[(w_t \cdot x - y)^2] \]

This is because \(w_t\) only depends on the first \(t - 1\) samples but doesn’t depend on \((x_t, y_t)\).

Now let’s start the proof:

\[E_S[R_v] = E_{S,(x,y)}[(v \cdot x - y)^2] \]
\[\leq E\left[ \frac{1}{m} \sum_{t} (w_t \cdot x - y)^2 \right] \quad \text{(using observation (1))} \]
\[= \frac{1}{m} \sum_{t} E[(w_t \cdot x - y)^2] \]
\[= \frac{1}{m} \sum_{t} E[(w_t \cdot x_t - y_t)^2] \quad \text{(observation (3))} \]
\[= \frac{1}{m} E\left[ \sum_{t} (w_t \cdot x_t - y_t)^2 \right] \]
\[\leq \frac{1}{m} E\left[ \sum_{t} \left( \frac{u \cdot x_t - y_t}{1 - \eta} \right)^2 + \frac{||u||^2}{\eta} \right] \quad \text{(by WH bound)} \]
\[= \frac{1}{m} \left[ \sum_{t} E[(u \cdot x_t - y_t)^2] \right] + \frac{||u||^2}{\eta} \]
\[= \frac{1}{m} \left[ \sum_{t} E[(u \cdot x - y)^2] \right] + \frac{||u||^2}{\eta m} \quad \text{(by observation (2))} \]
\[= \frac{R_u}{1 - \eta} + \frac{||u||^2}{\eta m} \]

and we have completed the proof.