COS 511: Theoretical Machine Learning

Lecturer: Rob Schapire
Scribe: Ksenia Sokolova

Lecture #18
April 10, 2019

Last time we started the topic of online linear regression. In this lecture we look at the Widrow-Hoff algorithm, the motivation behind it, and prove the regret bound. We then explore the connection of online and batch learning algorithms.

1 Online Linear Regression

In regression we are generally trying to predict real valued labels. For example, predicting whether or not it will rain can be treated as a classification problem, but really we want to predict the probability of rain, a regression problem.

Recall that in the online linear regression setting the learning algorithm is maintaining a weight vector $\mathbf{w}_t$, and learning happens in T rounds. On each round the learner gets an example, makes a prediction using $\mathbf{w}_t$, observes the actual outcome, computes loss and updates the weight vector:

- Initialize $\mathbf{w_1}$
- For $t = 1, ..., T$:
  - Observe $\mathbf{x}_t \in \mathbb{R}$
  - Predict $\hat{y}_t = \mathbf{w}_t \cdot \mathbf{x}_t$
  - Observe $y_t \in \mathbb{R}$
  - Calculate loss $(\hat{y}_t - y_t)^2$
  - Update $\mathbf{w}_{t+1}$

Additionally, define the cumulative loss of the algorithm as the cumulative sum of the losses for every round: $L_A = \sum_{t=1}^{T} (\hat{y}_t - y_t)^2$. The loss for the single fixed vector $\mathbf{u}$ is $L_{\mathbf{u}} = \sum_{t=1}^{T} (\mathbf{u} \cdot \mathbf{x}_t - y_t)^2$. We want to bound the loss of the algorithm in terms of the loss of the best vector $\mathbf{u}$:

$$L_A \leq \min_{\mathbf{u}} L_{\mathbf{u}} + [\text{small number}]$$

To finish the algorithm outline above, we need to provide initialization information for $\mathbf{w}_1$ and the update rule:

- **Initialize $\mathbf{w}_1 = \mathbf{0}$**
- For $t = 1, ..., T$:
  - Observe $\mathbf{x}_t \in \mathbb{R}$
  - Predict $\hat{y}_t = \mathbf{w}_t \cdot \mathbf{x}_t$
  - Observe $y_t \in \mathbb{R}$
  - Calculate loss $(\hat{y}_t - y_t)^2$
  - **Update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta(\mathbf{w}_t \cdot \mathbf{x}_t - y_t)\mathbf{x}_t$**

Here η is the leaning rate parameter, such that $0 < \eta < 1$. This algorithm is called Widrow-Hoff (WH), or sometimes Least Mean Squares (LMS).

1.1 Where does the update rule come from?

Motivation 1: Form of Gradient Descent

The first approach to explaining the update rule would be to consider our goal of minimizing the loss function:

$$\text{loss of } \mathbf{w} \text{ on } (\mathbf{x}, y) = (\mathbf{w} \cdot \mathbf{x} - y)^2 = L(\mathbf{w}, \mathbf{x}, y)$$

Recall that the gradient of a continuous and differentiable function is the direction in which it increases the fastest. So the natural way to decrease the function is to take steps in the direction of the negative gradient. The gradient of the loss function is:

$$\nabla_{\mathbf{w}} L = \begin{pmatrix} \partial L/\partial w_1 \\ \partial L/\partial w_2 \\ \vdots \\ \partial L/\partial w_n \end{pmatrix} = 2(\mathbf{w} \cdot \mathbf{x} - y)\mathbf{x}$$

Thus, the update rule is $\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{1}{2}\eta(\nabla_{w_t} L(\mathbf{w}_t, \mathbf{x}_t, y_t))$. Unfolding it from the inside, $\mathbf{w}_t$ is where we are in the process; $\mathbf{x}_t$ is the new example; $\eta \nabla_{\mathbf{w}_t} L(\mathbf{w}_t, \mathbf{x}_t, y_t)$ is the step; η is the size of the step and $1/2$ is just a factor used to cancel 2 in the gradient.

Motivation 2: what is the goal of each step?

In the algorithm, $\mathbf{w}_t$ encapsulates everything that was learned so far. So going forward, we want to minimize loss on the example just observed and at the same time keep the progress achieved. Mathematically, we want to:

- Minimize $L(\mathbf{w}_{t+1}, \mathbf{x}_t, y_t)$: $\min(\mathbf{w}_{t+1} \cdot \mathbf{x}_t - y_t)^2$
- Stay close to $\mathbf{w}_t$: small $||\mathbf{w}_{t+1} - \mathbf{w}_t||_2^2$

Since we have two items to minimize, we can combine them into a problem of minimizing the weighted sum:

$$\min \eta(\mathbf{w}_{t+1} \cdot \mathbf{x}_t - y_t)^2 + ||\mathbf{w}_{t+1} - \mathbf{w}_t||_2^2$$

The solution to this minimization problem turns out to be $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta(\mathbf{w}_{t+1} \cdot \mathbf{x}_t - y_t)\mathbf{x}_t$. This is almost the value we want, but both sides of the expression depend on $\mathbf{w}_{t+1}$. However, in this situation it is appropriate to approximate $\mathbf{w}_{t+1} \approx \mathbf{w}_t$, and we get the update rule.

2 Analysis

Theorem 1. *If for all rounds t, $||\mathbf{x}_t||_2 \leq 1$, then*

$$L_{WH} \leq \min_{\mathbf{u} \in \mathbb{R}^n} \left(\frac{L_{\mathbf{u}}}{1 - \eta} + \frac{||\mathbf{u}||_2^2}{\eta} \right)$$

where L_{WH} is the cumulative loss of the Widrow-Hoff algorithm.

Note:

- Proving the upper bound containing minimum is equivalent to stating that the bound it true for all values of $\mathbf{u}$. In other words, it will be enough to show that:

$$\forall \mathbf{u}, L_{WH} \leq \frac{L_{\mathbf{u}}}{1-\eta} + \frac{||\mathbf{u}||_2^2}{\eta}$$

  To get a better insight on the meaning of the expression, divide both sides by T:

$$\frac{L_{WH}}{T} \leq \frac{1}{1-\eta}\frac{L_{\mathbf{u}}}{T} + \frac{||\mathbf{u}||_2^2}{\eta T}$$

  For a small η, $1-\eta \to 1$. Additionally, as $T \to \infty$, $\frac{||\mathbf{u}||_2^2}{\eta T} \to 0$. Thus, under these conditions, the rate at which Widrow-Hoff suffers loss approaches the rate at which $L_{\mathbf{u}}$ does.

- The proof that follows uses the potential function. To get a better intuitive understanding of what happens, note that potential is measuring progress in some way or how much loss the algorithm can afford to suffer while still achieving a particular regret bound. The kind of potential used before usually measured similarity between the learning algorithm's weight vector $\mathbf{w}_t$ and whatever it was compared to (here $\mathbf{u}$). Since both are just vectors in Euclidean space, $\Phi_t = ||\mathbf{w}_t - \mathbf{u}||_2^2$ can be used.

Proof. Choose any $\mathbf{u} \in \mathbb{R}^n$. Decide on the potential function to use. As discussed above, define potential at round t as $\Phi_t = ||\mathbf{w}_t - \mathbf{u}||_2^2$.

Establish some notation for the proof:

- $\ell_t = \hat{y}_t - y_t = \mathbf{w}_t \cdot \mathbf{x}_t - y_t$, which means that ℓ_t^2 is loss of Widrow-Hoff at round t
- $g_t = \mathbf{u} \cdot \mathbf{x}_t - y_t$, , which means that g_t^2 is loss of weight vector $\mathbf{u}$ at round t
- $\boldsymbol{\Delta}_t = \eta(\mathbf{w} \cdot \mathbf{x}_t - y_t)\mathbf{x}_t = \eta \ell_t \mathbf{x}_t$, and so $\mathbf{w}_{t+1} = \mathbf{w}_t - \boldsymbol{\Delta}_t$

Claim

$$\Phi_{t+1} - \Phi_t \leq -\eta \ell_t^2 + \frac{\eta}{1-\eta} g_t^2$$

Note: consider the expression above. The first element is the weighted measure of loss of Widrow-Hoff at round t. The second element is related to the loss that $\mathbf{u}$ suffers. Thus, this can be thought of as a measure of how much loss the learner can incur to not fall behind $\mathbf{u}$ too much.

Proof of claim

First, rewrite the potentials using the defined potential function, plug in the expression for $\mathbf{w}_{t+1}$ as described above and expand the first term:

$$\begin{aligned}
\Phi_{t+1} - \Phi_t &= ||\mathbf{w}_{t+1} - \mathbf{u}||_2^2 - ||\mathbf{w}_t - \mathbf{u}||_2^2 \\
&= ||(\mathbf{w}_t - \mathbf{u}) - \boldsymbol{\Delta}_t||_2^2 - ||\mathbf{w}_t - \mathbf{u}||_2^2 \\
&= ||\boldsymbol{\Delta}_t||^2 - 2(\mathbf{w}_t - \mathbf{u}) \cdot \boldsymbol{\Delta}_{\mathbf{t}} + ||\mathbf{w}_t - \mathbf{u}||_2^2 - ||\mathbf{w}_t - \mathbf{u}||_2^2 \\
&= ||\boldsymbol{\Delta}_t||^2 - 2(\mathbf{w}_t - \mathbf{u}) \cdot \boldsymbol{\Delta}_{\mathbf{t}}
\end{aligned}$$

Consider the elements in the expression above:

- $||\mathbf{\Delta}_t||^2 = \eta^2 \ell_t^2 ||\mathbf{x_t}||^2$, where $||\mathbf{x}_t||^2 \leq 1$.
- $2(\mathbf{w}_t - \mathbf{u}) \cdot \mathbf{\Delta}_t = 2\eta \ell_t \mathbf{x}_t \cdot (\mathbf{w}_t - \mathbf{u})$. Note that $\mathbf{x}_t \cdot (\mathbf{w}_t - \mathbf{u}) = (\mathbf{w}_t \cdot \mathbf{x}_t - y_t) - (\mathbf{u} \cdot \mathbf{x}_t - y) = \ell_t - g_t$.

Therefore, returning to the expression,

$$\Phi_{t+1} - \Phi_t \leq \eta^2 \ell_t^2 - 2\eta \ell_t (\ell_t - g_t) = (\eta^2 - 2\eta)\ell_t^2 + 2\eta \ell_t g_t$$

To simplify the above expression, use the inequality $ab \leq \frac{a^2+b^2}{2}$, with $a = \frac{g_t}{\sqrt{1-\eta}}$ and $b = \ell_t \sqrt{1-\eta}$:

$$\Phi_{t+1} - \Phi_t \leq (\eta^2 - 2\eta)\ell_t^2 + \eta\Big(\frac{g_t^2}{1-\eta} + \ell_t^2(1-\eta)\Big) \leq -\eta \ell_t^2 + \frac{\eta}{1-\eta} g_t^2$$

Finishing proof of Theorem 1

We will now use the claim to prove the theorem. First, note that $\Phi_{T+1} - \Phi_1$ is the total change in the potential. Further note that since $\mathbf{w}_1 = \mathbf{0}$, $||\mathbf{u}||_2^2 = ||\mathbf{w}_1 - \mathbf{u}||_2^2 = \Phi_1$. Thus,

$$-||\mathbf{u}||_2^2 = -\Phi_1 \leq \Phi_{T+1} - \Phi_1$$

where the last inequality is true since $\Phi_{T+1} \geq 0$ as a norm. Then, do some algebra and use the claim:

$$\Phi_{T+1} - \Phi_1 = (\Phi_{T+1} - \Phi_T) + (\Phi_T - \Phi_{T-1}) + \cdots + (\Phi_2 - \Phi_1)$$

$$= \sum_{t=1}^{T} (\Phi_{t+1} - \Phi_t)$$

$$\leq \sum_{t=1}^{T} (-\eta \ell_t^2 + \frac{\eta}{1-\eta} g_t^2)$$

Distribute the sum:

$$= -\eta \sum_{t=1}^{T} \ell_t^2 + \frac{\eta}{1-\eta} \sum_{t=1}^{T} g_t^2$$

Observe that the cumulative loss is the sum of losses every round:

$$= -\eta L_{WH} + \frac{\eta}{1-\eta} L_{\mathbf{u}}$$

Solving for L_{WH} we get exactly the bound we wanted to prove.

□

2.1 Families of Online Algorithms

In the previous section we minimized

$$\eta(\text{ loss of } \mathbf{w}_{t+1} \text{ on } \mathbf{x}_t, y_t) + (\text{distance between } \mathbf{w}_{t+1}, \mathbf{w}_t)$$

In particular, we chose the square loss function for the first part, and Euclidean distance for the second. But we have the freedom to choose other functions. For example, for any loss function L (still using the Euclidean distance) the update rule is:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla_{\mathbf{w}} L(\mathbf{w_t}, \mathbf{x}_t, y)$$

Another interesting variation would be to use the relative entropy instead of the Euclidean distance: $RE(\mathbf{w}_t || \mathbf{w}_{t+1})$. It is important to note that $\mathbf{w}$ then needs to be a probability distribution: non-negative and with components that sum to 1. The new update rule:

$$\forall i, w_{t+1,i} = \frac{w_{t,i}}{Z_t} \cdot \exp\left(-\eta \frac{\partial L(\mathbf{w}_t, \mathbf{x}_t, y_t)}{\partial w_i}\right)$$

This is called the Exponentiated Gradient Algorithm (EG). Note that the update rule is multiplicative, and not additive (as it was before). The analysis will be based on the relative entropy, but we are not going to cover it in detail.

The following table provides a summary of the additive and multiplicative update rules we have encountered thus far:

Additive	**Multiplicative**
SVM	AdaBoost
Perceptron	Winnow/Weighted Majority Algorithm (WMA)
Gradient Descent (GD)	Exponentiated Gradient (EG)

3 Connecting Online and Batch learning

Consider the two types of learning:

Online learning: uses one example at a time and does not make independence assumptions about the data.

Batch learning: uses the whole set of random data offline.

Intuitively, online learning seems more powerful since it does not make any randomness assumptions. In this section we will look at how we can take an online learning algorithm and apply it to the batch setting, and how the analysis carries over. We will do that through an example in the linear regression setting.

We are given $S = \langle(\mathbf{x_1}, y_1), ..., (\mathbf{x}_m, y_m)\rangle$, where $(\mathbf{x}_i, y_i) \sim D$ and are iid. We then get a test point $(\mathbf{x}, y) \sim D$. The goal is to find $\mathbf{v}$ with low risk (expected loss), where risk is defined to be

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

and the minimization goal can be expressed as bounding $R_{\mathbf{v}}$: $R_{\mathbf{v}} \leq \min_{\mathbf{u}} R_{\mathbf{u}}$+[something small]

Conveniently, we can use the WH algorithm for which we already have done the analysis:

1. Run WH for $T = m$ rounds on the examples $(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_m, y_m)$ in exactly the order given in S. Calculate $\mathbf{w}_1, ..., \mathbf{w}_m$ for each round
2. Output $\mathbf{v} = \frac{1}{m} \sum_{t=1}^{m} \mathbf{w}_t$

Note that in the proposed algorithm we are using the average of the weight vectors, and not the last produced $\mathbf{w}$. We then claim that the following theorem holds:

Theorem 2.

$$\mathbb{E}_S[R_{\mathbf{v}}] \leq \min_{\mathbf{u} \in \mathbb{R}^n} \left[\frac{R_{\mathbf{u}}}{1-\eta} + \frac{||\mathbf{u}||_2^2}{\eta m} \right]$$

where the expectation is taken over the random choice of sample S.

Note: previously, we proved high probability bounds, where we said that the statement is true with high probability. Here, we are doing it over the expected value.

Proof. Fix any vector $\mathbf{u} \in \mathbb{R}^n$. Let $(\mathbf{x}, y)$ be a random test example from D. The expectations used in the proof are with respect to the random sample S and the random test point $(\mathbf{x}, y)$.

The proof will consist of 3 observations.

Observation 1:
Statement: $(\mathbf{v} \cdot \mathbf{x} - y)^2 \leq \frac{1}{m} \sum_{t=1}^{m} (\mathbf{w}_t \cdot \mathbf{x} - y)^2$
Proof:

$$(\mathbf{v} \cdot \mathbf{x} - y)^2 = \left(\frac{1}{m} \sum_{t=1}^{m} (\mathbf{w}_t \cdot \mathbf{x} - y) \right)^2$$

By Jensen's inequality and convexity of $f(z) = z^2$:

$$\leq \frac{1}{m} \sum_{t=1}^{m} (\mathbf{w}_t \cdot \mathbf{x} - y)^2$$

Observation 2:
Statement: $\mathbb{E}\big[(\mathbf{u} \cdot \mathbf{x}_t - y_t)^2\big] = \mathbb{E}\big[(\mathbf{u} \cdot \mathbf{x} - y)^2\big]$
Proof: $(\mathbf{x}_t, y_t)$ and $(\mathbf{x}, y)$ are from the same distribution.

Observation 3:
Statement: $\mathbb{E}\big[(\mathbf{w}_t \cdot \mathbf{x}_t - y_t)^2\big] = \mathbb{E}\big[(\mathbf{w}_t \cdot \mathbf{x} - y)^2\big]$
Proof: $\mathbf{w}_t$ depends on the $t-1$ observations and therefore is independent from $(\mathbf{x}_t, y_t)$ and $(\mathbf{x}, y)$. In other words, if the dataset S and the test point are generated sequentially, $\mathbf{w}_t$ is already determined by the first $t-1$ examples before either $(\mathbf{x}_t, y_t)$ or $(\mathbf{x}, y)$ are observed. Therefore, even if we condition on $\mathbf{w}_t$, $(\mathbf{x}_t, y_t)$ or $(\mathbf{x}, y)$ remain identically distributed. And lastly, $(\mathbf{x}_t, y_t)$ and $(\mathbf{x}, y)$ are from the same distribution.

Putting it all together:

$$\mathbb{E}_S[R_{\mathbf{v}}] = \mathbb{E}\Big[(\mathbf{v}\cdot\mathbf{x}-y)^2\Big] \leq \mathbb{E}\Big[\frac{1}{m}\sum_{t=1}^{m}(\mathbf{w}_t\cdot\mathbf{x}-y)^2\Big] = \frac{1}{m}\Big[\sum_{t=1}^{m}\mathbb{E}(\mathbf{w}_t\cdot\mathbf{x}-y)^2\Big]$$

by definition of risk, using the first observation and linearity of expectation.

Use the third observation:

$$= \frac{1}{m}\sum_{t=1}^{m}\mathbb{E}\Big[(\mathbf{w}_t\cdot\mathbf{x}_t-y_t)^2\Big]$$

By linearity of expectation:

$$= \frac{1}{m}\mathbb{E}\Big[\sum_{t=1}^{m}(\mathbf{w}_t\cdot\mathbf{x}_t-y_t)^2\Big]$$

Note that now inside of the expected value we have the cumulative loss of Widrow-Hoff, and we proved the upper bound for it:

$$\leq \frac{1}{m}\mathbb{E}\Big[\frac{\sum_{t=1}^{m}(\mathbf{u}\cdot\mathbf{x}_t-y_t)^2}{1-\eta}+\frac{||\mathbf{u}||_2^2}{\eta}\Big]$$

Use linearity of expectation and pull out the constants:

$$= \frac{1}{m}\frac{\sum_{t=1}^{m}\mathbb{E}(\mathbf{u}\cdot\mathbf{x}_t-y_t)^2}{1-\eta}+\frac{||\mathbf{u}||_2^2}{m\eta}$$

Finally, use observation 2 and definition of risk of $\mathbf{u}$:

$$= \frac{1}{m}\frac{\sum_{t=1}^{m}\mathbb{E}(\mathbf{u}\cdot\mathbf{x}-y)^2}{1-\eta}+\frac{||\mathbf{u}||_2^2}{m\eta}$$

$$= \frac{R_{\mathbf{u}}}{1-\eta}+\frac{||\mathbf{u}||_2^2}{m\eta}$$

□