COS 511: Foundations of Machine Learning

Rob Schapire
Scribe: Erhan Bilal

Lecture #16
April 6, 2006

1 Regression (continued)

Let's review the meteorologist problem from the previous class: A TV station wanted to hire a meteorologist out of two applicants. Applicant A predicted rain with a probability of 70% while applicant B predicted 80%. The next day it rains and the TV station has to decide which one to hire. In other words, assuming the two candidates are using hypothesis h_1 and h_2 respectively, the question is which prediction function is better.

Here is again the mathematical formulation of this model:

- x - weather conditions
- $y = \begin{cases} 1 & \text{if it rains} \\ 0 & \text{otherwise} \end{cases}$
- $(x, y) \sim D$
- $p(x) = Pr[y = 1 \mid x] = E[y \mid x]$

The idea is to come up with a penalty function in terms of $h(x)$ and y that would allow us to compare how well the two hypotheses do. A good function is obtained by computing the squared difference between the prediction and the actual outcome and then sum it for all the data points. In the end choose the smallest sum.

$$\begin{array}{ccc} x_1 & y_1 & (h_1(x_1) - y_1)^2 \\ x_2 & y_2 & (h_2(x_2) - y_2)^2 \\ \vdots & \vdots & \vdots \end{array}$$

We will further show why this idea works. The hypothesis $h = h(x)$ chosen in this way is a good estimate of $p = p(x)$.

Fix x
Claim $h = p$ minimizes the expectation $E[(h - y)^2]$

Proof

$$\begin{aligned} E = E[(h-y)^2] &= p(h-1)^2 + (1-p)h^2 \\ \frac{dE}{dh} &= 2(h-p) = 0 \Rightarrow h = p \end{aligned}$$

A similar analysis can show us why a penalty function of the form $|h - y|$ doesn't work.

$$E[|h - y|] = p(1 - h) + (1 - p)h$$

which is minimized over $h \in [0, 1]$ when

$$h = \begin{cases} 1 & \text{if } p \geq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

As you can see we don't get the right behavior here since we wanted $h = p$.

To further justify the choice of the previous loss function we will show that minimizing the observed difference between predicted probability and actual outcomes is equivalent to minimizing the difference between predicted and true probability.

Theorem 1

$$E[(\underbrace{h(x) - p(x)}_{goal})^2] = E[(\underbrace{h(x) - y}_{observed})^2] - E[(\underbrace{p(x) - y}_{intrinsic})^2]$$

Proof

Fix x
$p = p(x)$
$h = h(x)$

$$\begin{aligned} E[(h-p)^2] &= (h-p)^2 \; because \; x \; is \; fixed \\ E[(h-y)^2] - E[(p-y)^2] &= E[(h^2 - 2hy + y^2) - (p^2 - 2py + y^2)] \\ &= h^2 - 2h\underbrace{E[y]}_{p} - p^2 + 2p\underbrace{E[y]}_{p} = (h-p)^2 \end{aligned}$$

Hence we proved the claim for a fixed x. To get the more general statement we only need to average over random variable x.

Finally we need to estimate the expectation $E[(h(x) - y)^2]$ by empirical average:

$$E[(h(x) - y)^2] \approx \frac{1}{m} \sum_{i=1}^{m} (h(x_i) - y_i)^2$$

We can use Chernoff bounds to show that these will be close for a single h. If $\mathcal{H}$ is finite we can use the union bound to show that

$$Pr[\exists h \in \mathcal{H} : |E[L_h] - \hat{E}[L_h]| > \epsilon] \leq \delta$$

where $L_h(x, y) = (h(x) - y)^2$. For $\mathcal{H}$ infinite this result can be generalized using a VC-style analysis.

So far we justified the use of the least-squares loss function and in the next section we will see how this minimization problem can be solved.

1.1 Linear Regression

Formulation of the problem:
Assume $h(x) = \mathbf{w} \cdot \mathbf{x}$
Given $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_m, y_m)$, m data points
Find $\mathbf{w}$ to minimize $\Phi = \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i - y_i)^2$

This particular method for doing regression is attributed to Gauss who discovered it in 1795. It's a procedure for finding the best-fitting curve to a given set of points by minimizing the sum of the squares of the offsets of the points from the curve. The process is illustrated in Figure 1 for the one dimensional case.

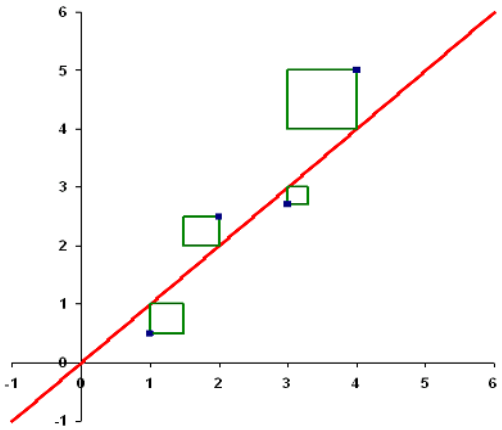

Figure 1: Linear regression.

For the multidimensional case Φ can also be rewritten in the matrix form below:

$$\Phi = \left\| \underbrace{\begin{pmatrix} \longleftarrow & \mathbf{x}_1^T & \longrightarrow \\ \longleftarrow & \mathbf{x}_2^T & \longrightarrow \\ \vdots & \vdots & \vdots \end{pmatrix}}_{M} \underbrace{\begin{pmatrix} w_1 \\ w_2 \\ \vdots \end{pmatrix}}_{\mathbf{w}} - \underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \end{pmatrix}}_{\mathbf{b}} \right\|^2$$

and then minimize it by setting the gradient equal to zero:

$$\nabla\Phi = \begin{pmatrix} \frac{\partial\Phi}{\partial w_1} \\ \frac{\partial\Phi}{\partial w_2} \\ \vdots \end{pmatrix} = 2M^T(M\mathbf{w} - \mathbf{b}) = 0 \Rightarrow \mathbf{w} = \underbrace{(M^TM)^{-1}M^T}_{pseudoinverse}\mathbf{b}$$

The quantity $(M^TM)^{-1}M^T$ is called the pseudoinverse of M and exists if M^TM is invertible. If not, it is still possible to compute an optimum $\mathbf{w}$ by using some other numerical optimization technique, *e.g.* gradient descent.

Next we will take a look at how the regression problem changes in an online setting and how it compares to the batch case.

1.2 Online linear regression

In the case of the online model the algorithm receives one example at a time based on which it has to make a prediction. The general outline of the linear regressor in this case is the following:

For $t = 1, 2, \ldots, T$

1. Get $\mathbf{x}_t \in \mathbf{R}^n$
2. Predict $\hat{y}_t = \mathbf{w}_t \cdot \mathbf{x}_t$
3. Observe $y_t \in \mathbf{R}$

4. Current loss $(y_t - \hat{y}_t)^2$

5. Update $\mathbf{w}_t$

The goal here is to minimize the cumulative loss with respect to the best loss we can obtain from the corresponding batch case. More specifically, we want to show that:

$$L_A \leq \min_{\mathbf{u}} L_{\mathbf{u}} + (small\ amount)$$

where

$$\begin{aligned} L_A &= \sum_{t=1}^{T} (y_t - \hat{y}_t)^2 \\ L_{\mathbf{u}} &= \sum_{t=1}^{T} (\mathbf{u} \cdot \mathbf{x}_t - y_t)^2 \end{aligned}$$

An example of where this method is used is echo cancellation in telephone networks. On long distance calls there is a phenomenon that causes the input signal to leak to the output, producing an echo at the sending end. To counter this, a complementary signal is sent in the opposite way to cancel the noise out. In order to successfully do this an online learning algorithm is used that predicts the future values of the noise signal. The name of the method in this particular case is Widrow-Hoff and will be presented further.

1.3 Widrow-Hoff Algorithm

The secret of this method lies in the way the weights $\mathbf{w}$ are updated:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta(\mathbf{w}_t \cdot \mathbf{x}_t - y_t)\mathbf{x}_t \quad where\, \eta > 0$$

So what is the explanation for this formula? What is the intuition behind it? One idea is to take the loss L_A and move a step in the steepest direction that minimizes it, and that would be the gradient:

$$\nabla L_A = 2(\mathbf{w}_t \cdot \mathbf{x}_t - y_t)\mathbf{x}_t$$

There is also another, more modern way to motivate this rule. There are two goals:

1. Minimize the loss $(\mathbf{w}_{t+1} \cdot \mathbf{x}_t - y_t)^2$.

2. Minimize the distance between $\mathbf{w}_{t+1}$ and $\mathbf{w}_t$.

We can combine these two goals into one:

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

and after we take the derivative we end up with the update rule:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta(\underbrace{\mathbf{w}_{t+1}}_{\approx \mathbf{w}_t} \cdot \mathbf{x}_t - y_t)\mathbf{x}_t$$

Given this update function we can now prove the following theorem:

Theorem 2

Assume $\|\mathbf{x}_t\|_2 \leq 1$ then

$$L_{WH} \leq \min_{\mathbf{u}} \left\{ \frac{L_{\mathbf{u}}}{1-\eta} + \frac{\|\mathbf{u}\|_2^2}{\eta} \right\}.$$

Alternatively if we divide by T, the total number of time steps, and for η small:

$$\frac{L_{WH}}{T} \lesssim (1+\eta)\frac{L_{\mathbf{u}}}{T} + \frac{\|\mathbf{u}\|_2^2}{\eta T}$$

we can affirm that the rate of WH loss over the total number of time steps converges to the rate of loss of the best vector $\mathbf{u}$.

Proof

First we must choose a potential function Φ. Intuitively, a good one should measure how close we are to the best vector $\mathbf{u}$:

$$\Phi = \|\mathbf{w}_t - \mathbf{u}\|_2^2$$

Notation:

$$\begin{aligned} l_t &= \mathbf{w}_t \cdot \mathbf{x}_t - y_t \\ g_t &= \mathbf{u} \cdot \mathbf{x}_t - y_t \end{aligned}$$

We will show that:

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

Once we show this the theorem will follow since:

$$\begin{aligned} \Phi_{T+1} - \Phi_1 &= (\Phi_{T+1} - \Phi_T) + (\Phi_T - \Phi_{T-1}) + \ldots + (\Phi_2 - \Phi_1) \\ &= \sum_{t=1}^{T} (\Phi_{t+1} - \Phi_t) \\ &\leq -\eta \underbrace{\sum_t l_t^2}_{L_{WH}} + \frac{\eta}{1-\eta} \underbrace{\sum_t g_t^2}_{L_{\mathbf{u}}} \end{aligned}$$

Therefore, since $\Phi_{T+1} \geq 0$ we have:

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

which, solving for L_{WH}, proves the theorem.