Lecture 3 Notes

Dan Sheldon

September 17, 2012

0 Errata

- Section 4, Equation (2): y_N^2 should be x_N^2. Fixed 9/17/12
- Section 5.3, Example 3: should read $w_0 = 0, w_1 = -1$. Fixed 9/17/12.

1 Review: Linear Regression Setup

- Training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$, where $x_i, y_i \in \mathbb{R}$ (i.e., real numbers: $0.3, 1.56, \pi$, etc.)
- Hypothesis $h_{\mathbf{w}}(x) = w_0 + w_1 x$ (linear function)
- Parameters w_0, w_1: each different value of parameters gives a different hypothesis
- **Goal**: find hypothesis $h_{\mathbf{w}}$ that is "best" fit to training data
- Cost function (aka loss function)
  - Numerical measure of fit between hypothesis and training data
  - Higher cost $\Rightarrow$ worse fit
  - Squared error cost function (Gauss)

$$\text{cost}(h_{\mathbf{w}}) = \sum_{i=1}^{N} (h_{\mathbf{w}}(x_i) - y_i)^2$$

  - Substitute form of linear hypothesis $h_{\mathbf{w}}(x) = w_1 x + w_0$ into cost function:

$$J(w_0, w_1) = \sum_{i=1}^{N} (w_0 + w_1 x_i - y_i)^2$$

  - **Simplification** (for now): assume $w_0 = 0 \;\Rightarrow\; h_w(x) = w_1 x$. New cost function is

$$J(w_1) = \sum_{i=1}^{N} (w_1 x_i - y_i)^2$$

  - For any given training set, *the cost function is a function of parameters only.*

Example 1 (Assuming $w_0 = 0$). *Consider the following training set*

x	y
1	2
2	3

The cost function is

$$\begin{aligned} J(w_1) &= (w_1 x_1 - y_1)^2 + (w_1 x_2 - y_2)^2 \\ &= (w_1 - 2)^2 + (2w_1 - 2)^2 \\ &= 5w_1^2 - 16w_1 + 13 \end{aligned}$$

This is a quadratic function of w_1*, so we can find the minimum by optimization.*

2 Illustration: Hypothesis vs. Cost Function

- Each hypothesis equated with numerical parameters
- *Parameter space* - set of all possible parameters

- To find best hypothesis: minimze cost function over parameter space.

3 Derivatives: What You Need To Know

- For a function $f(x)$, denote the *derivative* of f by $\frac{d}{dx} f(x)$ (sometimes $f'(x)$)
- Derivative = slope of the tangent line to f at x
- Derivative is equal to zero at a minimum of $f(x)$

- Illustration: minima, maxima, local minima, convex function (bowl-shaped)

4 Minimizing $J(w_1)$

One way to find a minimum (which works for linear regression, but not every problem) is to find the derivative, set it equal to zero, and solve the resulting equation.

Example 2 (Continuation of Example 1)**.** *To minimize* $J(w_1) = 5w_1^2 - 16w_1 + 13$*, set the derivative equal to zero and solve for* w_1*:*

$$\begin{aligned} 0 &= \frac{d}{dw_1} J(w_1) = 10w_1 - 16 \\ 10w_1 &= 16 \\ w_1 &= \frac{8}{5} = 1.6. \end{aligned}$$

General Case: For the general problem, we can solve for w_1 in terms of $(x_1, y_1), (x_2, y_2), \ldots (x_N, y_N)$. We need the following fact (which you can verify if you know calculus):

$$\frac{d}{dw_1} J(w_1) \;=\; \frac{d}{dw_1} \sum_{i=1}^{N} (w_1 x_i - y_i)^2 \;=\; 2 \sum_{i=1}^{N} (w_1 x_i - y_i) x_i \;=\; 2 \sum_{i=1}^{N} (w_1 x_i^2 - x_i y_i). \tag{1}$$

Then, we can set the derivative to zero and solve, to get

$$\begin{aligned} 0 &= 2[(w_1 x_1^2 - x_1 y_1) + \ldots + (w_1 x_N^2 - x_N y_N)] \\ w_1(x_1^2 + \ldots + x_N^2) &= (x_1 y_1 + \ldots + x_N y_N) \\ w_1 &= \frac{x_1 y_1 + \ldots + x_N y_N}{x_1^2 + \ldots + x_1^2} \end{aligned} \tag{2}$$

We can apply this formula for w_1 to any training set to get the best fit line. It is our first ML algorithm!

5 Gradient Descent

- But we want to minimize $J(w_0, w_1)$, not $J(w_1)$. For general ML problems, not always possible to minimize cost function by setting derivatives to zero (In this case it is possible, but laborious. See Equation (18.3) on p. 719 of R&N for the answer).
- Gradient descent: simple and *very broadly applicable* algorithm to minimize *any* function of $J(w_0, w_1, \ldots, w_d)$ of multiple variables. Requirement: be able to compute *partial derivatives* $\frac{\partial}{\partial w_j} J(w_1, \ldots, w_d)$
- Mathemetical definition of algorithm ($d = 2$):
  1. Initialize w_0, w_1 arbitrarily (e.g. $w_0 = 0$, $w_1 = 0$)
  2. Repeat until convergence

$$\begin{aligned} w_0 &= w_0 - \alpha \frac{\partial}{\partial w_0} J(w_0, w_1) \\ w_1 &= w_1 - \alpha \frac{\partial}{\partial w_1} J(w_0, w_1) \end{aligned}$$

- Implementation note: make updates *simultaneously*

1. Initialize w_0, w_1 arbitrarily
2. Repeat until convergence

$$\begin{aligned}
\Delta_0 &\leftarrow \frac{\partial}{\partial w_0} J(w_0, w_1) \\
\Delta_1 &\leftarrow \frac{\partial}{\partial w_1} J(w_0, w_1) \\
w_1 &\leftarrow w_1 - \alpha \Delta_0 \\
w_2 &\leftarrow w_2 - \alpha \Delta_1
\end{aligned}$$

5.1 Illustration In One Dimension

$$\text{Repeat:} \quad w_1 \leftarrow w_1 - \alpha \frac{d}{dw_1} J(w_1)$$

.

5.2 Illustration In Two Dimensions

5.3 Gradient Descent for Linear Regression

To solve the linear regression problem using gradient descent, the only thing we need to know are the partial derivatives for our cost function:

$$\frac{\partial}{\partial w_j} J(w_0, w_1) \;=\; \frac{\partial}{\partial w_j} \sum_{i=1}^{N} (h_{\mathbf{w}}(x_i) - y_i)^2 \;=\; \frac{\partial}{\partial w_j} \sum_{i=1}^{N} (w_1 x_i + w_0 - y_i)^2, \qquad j = 1, 2.$$

Here they are:

$$\frac{\partial}{\partial w_0} J(w_0, w_1) = 2 \sum_{i=1}^{N} (h_{\mathbf{w}}(x_i) - y_i)$$
$$\frac{\partial}{\partial w_1} J(w_0, w_1) = 2 \sum_{i=1}^{N} (h_{\mathbf{w}}(x_i) - y_i) \cdot x_i$$

- Note that we can drop the constant 2 and absorb it into the learning rate α
- Work these out on your own if you are comfortable with partial derivatives.

Example 3 (Continuation of Example 1)**.** *Recall the training set:*

x	y
1	2
2	3

Initialize $w_0 = 0, w_1 = -1$, and take one step of gradient descent. (Drop the factor of 2 in the partial derivatives).

First, compute

$$h_{\mathbf{w}}(x_1) = -1, \quad h_{\mathbf{w}}(x_2) = -2.$$

Then,

$$\frac{\partial}{\partial w_0} J(w_0, w_1) = (-1 - 2) + (-2 - 3) = -8$$
$$\frac{\partial}{\partial w_0} J(w_0, w_1) = (-1 - 2) \cdot 1 + (-2 - 3) \cdot 2 = -13.$$

So the new values (w_0', w_1') are

$$w_0' = 0 - (0.1)(-8) = 0.1$$
$$w_1' = -1 - (0.1)(-13) = 0.3$$

A Derivatives: Optional Background

For common functions (polynomials, exponentials, log, etc.) there are rules to find their derivatives. Here are a few of the most important rules.

- Linear
$$\frac{d}{dx}x = 1.$$
- Quadratic (important!):
$$\frac{d}{dx}x^2 = 2x.$$
- General polynomial:
$$\frac{d}{dx}x^k = kx^{k-1}.$$
- Linearity:
$$\frac{d}{dx}(af(x) + bg(x)) = a\frac{d}{dx}f(x) + b\frac{d}{dx}g(x).$$
- Chain rule:
$$\frac{d}{dx}f(g(x)) = f'(g(x))\frac{d}{dx}g(x)$$

Examples:

1. $\frac{d}{dx}3x = 3$
2. $\frac{d}{dx}(3x + 4) = 3$
3. $\frac{d}{dx}3x^2 = 6x$
4. $\frac{d}{dx}(3x - 4)^2 = 2(3x - 4)\frac{d}{dx}(3x - 4) = 2 \cdot (3x - 4) \cdot 3 = 18x - 24$
5. (another way) $\frac{d}{dx}(3x - 4)^2 = \frac{d}{dx}(9x^2 - 24x + 16) = 18x - 24$

A.1 Partial Derivatives

For a function $f(x, y)$ of two variables, the *partial derivative with respect to* x is denoted $\frac{\partial}{\partial x}f(x, y)$. It is calculated by following the same rules, except y *is treated as a constant.* **Example:**

$$\frac{\partial}{\partial x}3x^2y = \frac{\partial}{\partial x}(3y)x^2 = 6yx.$$

Similarly, the partial derivative with respect to y, denoted $\frac{\partial}{\partial y}f(x, y)$ is computed by treating x as a constant and differentiating with respect to y. **Example:**

$$\frac{\partial}{\partial x}3x^2y = \frac{\partial}{\partial x}(3x^2)y = 3x^2.$$

For a function $f(x_1, x_2, \ldots, x_d)$ of many variables, the partial derivative with respect to x_i, denoted $\frac{\partial}{\partial x_i}f(x_1, x_2, \ldots, x_d)$, is computed by treating *all variables expect* x_i *as constants*, and computing the derivative with respect to x_i.