

## Derivation of the Normal Equation for linear regression - Eli Bendersky's website

I was going through the Coursera "Machine Learning" course, and in the section on multivariate linear regression something caught my eye. Andrew Ng presented the Normal Equation as an analytical solution to the linear regression problem with a least-squares cost function. He mentioned that in some cases (such as for small feature sets) using it is more effective than applying gradient descent; unfortunately, he left its derivation out.

Here I want to show how the normal equation is derived.

First, some terminology. The following symbols are compatible with the machine learning course, not with the exposition of the normal equation on Wikipedia and other sites - semantically it's all the same, just the symbols are different.

Given the hypothesis function:

$$h_{\theta}(x) = \theta_0 x_0 + \theta_1 x_1 + \dots + \theta_n x_n$$

We'd like to minimize the least-squares cost:

$$J(\theta_{0\dots n}) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Where  $x^{(i)}$  is the  $i$ -th sample (from a set of  $m$  samples) and  $y^{(i)}$  is the  $i$ -th expected result.

To proceed, we'll represent the problem in matrix notation; this is natural, since we essentially have a system of linear equations here. The regression coefficients  $\theta$  we're looking for are the vector:

$$\begin{pmatrix} \theta_0 \\ \theta_1 \\ \dots \\ \theta_n \end{pmatrix} \in \mathbb{R}^{n+1}$$

Each of the  $m$  input samples is similarly a column vector with  $n+1$  rows,  $x_0$  being 1 for convenience. So we can now rewrite the hypothesis function as:

$$h_{\theta}(x) = \theta^T x$$

.... ..

When this is summed over all samples, we can dip further into matrix notation. We'll define the "design matrix"  $X$  (uppercase X) as a matrix of  $m$  rows, in which each row is the  $i$ -th sample (the vector  $x^{(i)}$ ). With this, we can rewrite the least-squares cost as following, replacing the explicit sum by matrix multiplication:

$$J(\theta) = \frac{1}{2m} (X\theta - y)^T (X\theta - y)$$

Now, using some matrix transpose identities, we can simplify this a bit. I'll throw the  $\frac{1}{2m}$  part away since we're going to compare a derivative to zero anyway:

$$J(\theta) = ((X\theta)^T - y^T)(X\theta - y)$$

$$J(\theta) = (X\theta)^T X\theta - (X\theta)^T y - y^T (X\theta) + y^T y$$

Note that  $X\theta$  is a vector, and so is  $y$ . So when we multiply one by another, it doesn't matter what the order is (as long as the dimensions work out). So we can further simplify:

$$J(\theta) = \theta^T X^T X\theta - 2(X\theta)^T y + y^T y$$

Recall that here  $\theta$  is our unknown. To find where the above function has a minimum, we will derive by  $\theta$  and compare to 0. Deriving by a vector may feel uncomfortable, but there's nothing to worry about. Recall that here we only use matrix notation to conveniently represent a system of linear formulae. So we derive by each component of the vector, and then combine the resulting derivatives into a vector again. The result is:

$$\frac{\partial J}{\partial \theta} = 2X^T X\theta - 2X^T y = 0$$

Or:

$$X^T X\theta = X^T y$$

[Update 27-May-2015: I've written [another post](#) that explains in more detail how these derivatives are computed.]

Now, assuming that the matrix  $X^T X$  is invertible, we can multiply both sides by  $(X^T X)^{-1}$  and get:

$$\theta = (X^T X)^{-1} X^T y$$

Which is the normal equation.