Regression

Statistics: describing data, inferring conclusions

Machine learning: predicting future data (out-of-sample)

Assumption for linear regression: data can be modeled by

$$y_i = \alpha + \beta x_i + \epsilon_i$$

First algorithmic question for us: how to find α and β ?

Least Squares

Define $\overline{x}$ and $\overline{y}$ as usual from our sample data. Now define:

$$S_{xx} = \sum_{i=1}^{n} (x_i - \overline{x})^2$$

$$S_{yy} = \sum_{i=1}^{n} (y_i - \overline{y})^2$$

$$S_{xy} = \sum_{i=1}^{n} (x_i - \overline{x})(y_i - \overline{y})$$

Let's fit a line to the data as best as we can. How do we define this? Residual sum of squares (RSS)

$$\sum_{i=1}^{n} (y_i - (c + dx_i))^2$$

Now, find a and b, estimators of α and β, such that:

$$\min_{c,d} \sum_{i=1}^{n} (y_i - (c + dx_i))^2 = \sum_{i=1}^{n} (y_i - (a + bx_i))^2$$

For any fixed value of d, the minimizing value of c can be found as follows.

$$\sum_{i=1}^{n} (y_i - (c + dx_i))^2 = \sum_{i=1}^{n} ((y_i - dx_i) - c)^2$$

Turns out the right side is minimized at

$$c = \frac{1}{n} \sum_{i=1}^{n} (y_i - dx_i)$$

$$= \overline{y} - d\overline{x}$$

Why?

$$\min_{a} \sum_{i=1}^{n} (x_i - a)^2 = \min_{a} \sum_{i=1}^{n} (x_i - \overline{x} + \overline{x} - a)^2$$

$$= \sum_{i=1}^{n} (x_i - \overline{x})^2 + 2 \sum_{i=1}^{n} (x_i - \overline{x})(\overline{x} - a) + \sum_{i=1}^{n} (\overline{x} - a)^2$$

Second term drops out, basically giving us our result

For a given value of d, the minimum value of RSS is then

$$\sum_{i=1}^{n} ((y_i - dx_i) - (\overline{y} - d\overline{x}))^2$$

$$= \sum_{i=1}^{n} ((y_i - \overline{y}) - d(x_i - \overline{x}))^2$$

$$= S_{yy} - 2dS_{xy} + d^2 S_{xx}$$

Take the derivative with respect to d and set to 0

$$-2S_{xy} + 2dS_{xx} = 0$$

$$\Rightarrow d = \frac{S_{xy}}{S_{xx}}$$

Multivariate Regression

Hypothesis space? Characterized by the vector $\mathbf{w} = (w_0, w_1, \ldots w_n)$, where

$$h(\mathbf{x}^j) = w_0 + w_1 x_1^j + w_2 x_2^j + \ldots w_n x_n^j$$

w_0 is the intercept term. Can just "add on" a feature that is always 1. Then $h(\mathbf{x}^j) = \mathbf{w} \cdot \mathbf{x}_j$

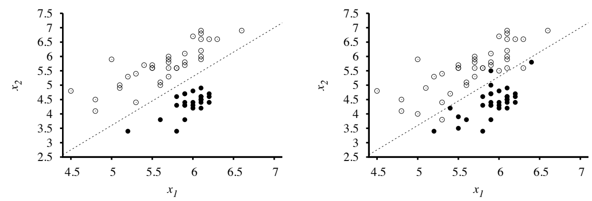
Find $\mathbf{w}^* = \underset{\mathbf{w}}{\text{argmin}} \sum_j (y^j - \mathbf{w} \cdot \mathbf{x}^j)^2$

Gradient descent will find the (unique) min of the loss function:

$$w_i \leftarrow w_i + \alpha \sum_j x_i^j (y^j - h_{\mathbf{w}}(\mathbf{x}^j))$$

Classification Using Linear Models

Two examples of data from earthquakes (white circles) and nuclear explosions (black circles). x_1 and x_2 are body wave magnitude and surface wave magnitude, respectively.



Linear separator: $-4.9 + 1.7x_1 - x_2 = 0$.
$h_{\mathbf{w}}(\mathbf{x}) = 1$ if $\mathbf{w} \cdot \mathbf{x} \geq 0$ and 0 otherwise.

In higher dimensions, the separator is called a hyperplane

Perceptron Learning Rule

$$w_i \leftarrow w_i + \Delta w_i$$

where

$$\Delta w_i = \alpha(y - h_{\mathbf{w}}(\mathbf{x}))x_i$$

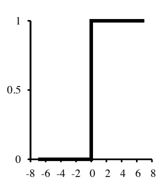
Nice fact: will provably converge to a linear separator if one exists.

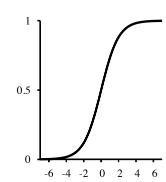
Not so nice: behaves unpredictably while training, and behavior is poor if the data are not linearly separable (although can be improved upon with some tricks)

Much of this is because of the hard threshold in 0/1 classification.

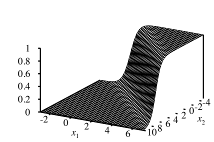
Logistic Regression

Use the logistic function $1/(1 + e^{-z})$ to map a real-valued output to a probability. Now we've got *soft* thresholds that can be converted into predictions as needed.





$$h_{\mathbf{w}}(\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w} \cdot \mathbf{x}}}$$



Weight updates can be derived using gradient descent. For square loss:

$$\frac{\partial}{\partial w_i}\mathsf{Loss}(\mathbf{w}) = \frac{\partial}{\partial w_i}(y - h_{\mathbf{w}}(\mathbf{x}))^2$$

Very useful, and the standard in the literature for prediction from an economics / statistics perspective. Also the baseline for probability estimation.