

6.867 HW2: Binary Classification Methods

October 13, 2015

When making predictions and models, sometimes we would rather have discrete labels instead of continuous values. For example, if we were looking exam results, we might only care if students pass or fail. Alternatively, we may want to study electricity patterns, and at any given time lights can either be on or off. This idea of discrete labels is known as classification and in the case that there are only two options, binary classification. For the following work, we will consider the class labels $\{-1, +1\}$.

1 Logistic Regression

The first method we will use to implement binary classification will be logistic regression. The use of logistic functions, specifically the sigmoid, allow us to map continuous values of input values to a corresponding probability between 0 and 1, which then can be used to select a discrete label. Considering the maximum likelihood estimation, logistic regression can be written as

$$\text{MLE}(w) = \sum_{i=1}^n \log(1 + e^{-y^{(i)} w^T \Phi(x^{(i)})}) \quad (1)$$

Where w are the weights defining our model, $x(i)$, $y(i)$ given data points, and Φ the basis functions from which we construct our model. Letting the basis function be a simple linear formulation, $\Phi(x) = [1 \ x_1 \ x_2 \ \dots]^T$, the above equation can be simplified as

$$\text{NNL}(w) = \sum_{i=1}^n \log(1 + e^{-y^{(i)}(w_0 + w^T x^{(i)})}) \quad (2)$$

However to prevent overfitting, we also want to introduce a regularization term into the logistic regression error function. Specifically we will consider a quadratic regularizer, which results in the finalized error function,

$$E_{LR}(w) = \text{NNL}(w) + \lambda w^T w \quad (3)$$

Unlike some of the regression formulations we looked at last time, such as ridge or linear, there is no closed form solution to minimizing the error. Thus, we must employ optimization methods to select the correct weights. As far as optimization methods, two options are considered below, a simple gradient descent and the built in MATLAB optimization, `fminunc`.

Once weights are calculated from minimizing the error on the training data our classifier is simply defined as,

$$h = \text{sign}(w^T \Phi(x)) \quad (4)$$

Which simply creates a hyperplane in which for any input, x . This classification returns -1 if $w^T \Phi(x) < 0$ and +1 if $w^T \Phi(x) \geq 0$

As the magnitude of w increases the steepness of the hyperplane increases. If w goes to ∞ the separator becomes a step function. The regularizer helps to limit this by penalizing the magnitude of the weights.

2 Support Vector Machine

Another implementation method of binary classification is the Support Vector Machine. Instead of dual form of linear SVMs with slack variables