Training a Support Vector Machine (SVM) using the Primal Form

Jerrison Li, Adrian Barbu[1]

[1] Department of Statistics, Florida State University

Support Vector Machine is a supervised learning tool in machine learning that analyzes and recognizes pattern in binary data. We will be working with linearly separable data, so our SVM will try to find a line that separates the points from each class x_1 and x_2 when plotted together. Support vectors are the points nearest to this line and what we are trying to do is to find the line that is farthest from both of the support vectors. This line can be described by:

$$B \cdot x + B_0 = 0$$

We achieve this by solving the following problem:

$$\min_{B_0, B} \sum_{i=1}^{N} f(y_i (B_0 + \sum_{j=1}^{N} x_{i,j} \cdot B_j))$$

where φ is the loss function

We update the parameters after each iteration:

$$B_0 = B_0 + \eta * y_i * g$$

$$B = B + (\eta * y_i * g) * x_i$$

where $g = f'(y_i * (B_0 + x_i * B))$,

$\eta =$ training rate

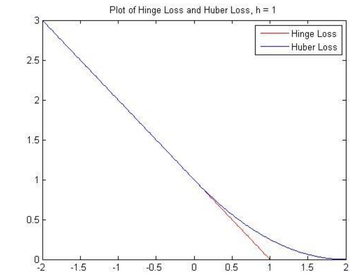
We will be using the Huber loss (a differentiable approximation to the Hinge loss). The other loss used is a cubic approximation to the logistic loss.

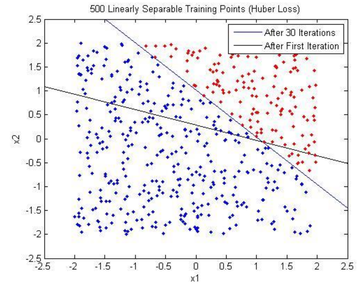
Hinge Loss

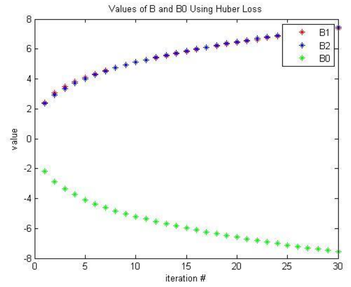
$$f(z) = \max(0, 1 - z)$$

Huber Loss

$$f(x) = \begin{cases} 0 & if \quad z > 1 + h \\ \dfrac{(1 + h - z)^2}{4h} & if \quad |1 - z| \le h \\ 1 - z & if \quad z < 1 - h \end{cases}$$





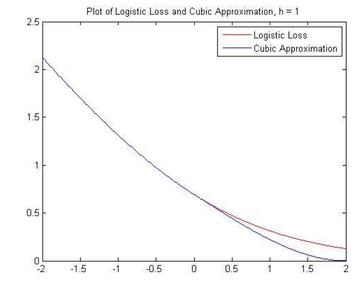


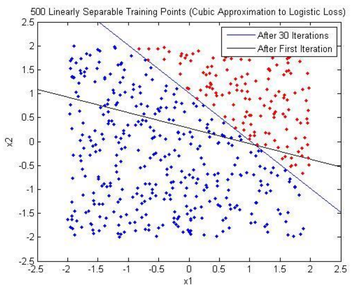
Logistic Loss

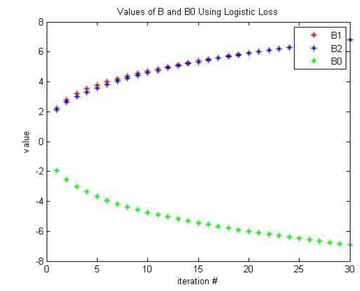
$$f(z) = \log(1 + e^{-z})$$

Cubic Approximation

$$f(x) = \begin{cases} \log(1 + e^{-z}) & if \quad z < 1 - h \\ ax^3 + bx^2 + cx + d & if \quad |1 - z| \le h \\ 0 & if \quad z > 1 + h \end{cases}$$







Results Pertaining to Both Loss Functions