Lecture 11

Name: Harsh Sanjay Roniyar

Roll Number: 22B3942

$x^T M x + x^T v + C$

Linear: $x^T v + c = 0$

Let's come back to 1 dimension..

The decision boundary is $W^T x + b$ and depending on the sign of the boundary for a specific value, the data-point is assigned to that class.

Suppose, we did have two classes in the single dimension, but the σ for these classes is very different. Their intersections give some sort of thresholds (boundaries) for the classifiers. In this case, we won't be able to get a linear classifier since we have two threshold conditions.

Thus, even if we have different σ's in one of the dimensions, the whole classifier for the n-dimensional data cannot retain its non-linearity.

Now, in the case when we were in 2 dimensions, and have unequal σ's, then the bayesian boundaries, wouldn't be exactly same contours. In order to find the boundaries, we assume the distributions to be linear.

Gradient Descent for Linear Classifiers

For Regression, we had loss options as - 1. MSE and 2. MAE and L2 and L1 regularization.

Now, for linear classifiers, we have the following Loss function -

1. Misclassification Rate:

$$\frac{1}{N}\sum_{i=1}^{N}\mathbb{1}\{sign(y_i) \neq t_i\}$$

   where $\mathbb{1}$ is the indicator function, $t_i \in \{-1, 1\}$ and $y_i = w^T x_i + b$. The indicator function is a binary function, where it outputs 1 if the condition evaluates to true, else 0. But, the problem with the sign function is that its is almost zero everywhere, leading to a discontinuous classifier.

2. Replacing $\text{sign}(y_i)$ with $\sigma(y_i)$ where σ is the sigmoid function given by -

$$\sigma(y_i) = \frac{1}{1+e^{-y_i}},$$

   and we also change the labels t_i from $\{-1, 1\}$ to $\{0, 1\}$.
   Now, $y_i = \sigma(w^T x_i + b)$ interpreted as $p(t = 1|x_i)$ and thus, $y_i \in (0, 1)$ and $w^T x_i + b \in (-\infty, \infty)$.
3. Measuring loss as

$$\text{loss} = \frac{1}{2N} \sum_i (y_i - t_i)^2$$

   can work but its error is gaussian, and might not give the most optimal result.
4. **KL Divergence** - BCE (Binary Cross Entropy):

$$-\sum_i \sum_c t_{c,i} \log y_{c,i} = -\sum_i \left[t_i \log y_i + (1 - t_i) \log(1 - y_i)\right]$$

   where, c is the class.

Logistic Regression - Minimize BCE of sigmoid of line lin. exp. wrt. a binary target.

Gradient Descent using BCE

$$\frac{dl_i}{dw} = \frac{dl_i}{dy_i} \frac{dy_i}{dh_i} \frac{dh_i}{dw}$$

$$= \left(\frac{t_i}{y_i} - \frac{1 - t_i}{1 - y_i}\right) \cdot y_i \cdot (1 - y_i) \cdot x_{i,k}$$