Machine Learning Homework 3

Hunter Carroll

February 2024

Problem 1

Problem Statement

Prove Equation (6) in Lecture 5, i.e.,

$$F1 = \frac{TP}{TP + \frac{FN+FP}{2}}$$

Solution

Scratch Work:

The F1 score is a single score that represents both Precision (P) and Recall (R) to find a formula for this we can take their harmonic mean,

$$\text{harmonic} = \frac{2xy}{x+y}$$

Such that,

$$F1 = 2\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Let us define Precision and Recall,

$$\text{Precision} = \frac{TP}{TP+FP} \text{ and Recall } = \frac{TP}{TP+FN}$$

Let us work with the top,

$$\Rightarrow (\frac{TP}{TP+FP})(\frac{TP}{TP+FN})$$

$$\Rightarrow \frac{TP^2}{(TP+FP)(TP+FN)}$$

Let us work with the bottom,

$$\Rightarrow \frac{TP}{TP+FP} + \frac{TP}{TP+FN}$$

$$\Rightarrow \frac{TP(TP+FN)}{(TP+FP)(TP+FN)} + \frac{TP(TP+FP)}{(TP+FP)(TP+FN)}$$

$$\Rightarrow \frac{TP(TP+FN)+TP(TP+FP)}{(TP+FP)(TP+FN)}$$

Altogether,

$$\Rightarrow \frac{\frac{TP^2}{(TP+FP)(TP+FN)}}{\frac{TP(TP+FN)+TP(TP+FP)}{(TP+FP)(TP+FN)}}$$

Multiply by the reciprocal,

$$\Rightarrow \frac{TP^2(TP+FP)(TP+FN)}{(TP+FP)(TP+FN)(TP(TP+FN)+TP(TP+FP))}$$

$$\Rightarrow \frac{TP^2}{TP(TP+FN+TP+FP)}$$

Multiply by 2 and simplify the denominator,

$$\Rightarrow \frac{2TP}{2TP+FN+FP}$$

Divide both the numerator and denominator by 2,

$$\Rightarrow \frac{TP}{TP+\frac{FN+FP}{2}}$$

As desired.

Proof. Let $P = \frac{TP}{TP+FP}$ and $R = \frac{TP}{TP+FN}$. Then if we take the harmonic mean of P and R we get,

$$F1 = 2\frac{PR}{P+R}$$

Then using P and R,

$$F1 = 2(\frac{(\frac{TP}{TP+FP})(\frac{TP}{TP+FN})}{\frac{TP}{TP+FP}+\frac{TP}{TP+FN}})$$

This simplifies to,

$$F1 = \frac{2TP}{2TP + FN + FP}$$

If we divide the numerator and denominator by 2,

$$F1 = \frac{TP}{TP + \frac{FN+FP}{2}}$$

Therefore, we have arrived at the definition of the F1 score by the harmonic mean. □

Problem 2

Consider the linear decision function,

$$s(\mathbf{x}) = \boldsymbol{w}^T \boldsymbol{x} + b$$

As defined in lecture 6 which deals with binary classification problems.

Part a:

Show that $\boldsymbol{w}$ is perpendicular to every vector within the decision boundary.
Scratch Work:

Note. If we take the difference of any two points that exist in the hyperplane then their difference and sum also exist in the hyperplane.
We know that $b = \beta_0$ (bias) and $\boldsymbol{w} = (\beta_1, \beta_2, ..., \beta_n)^T$ (weights) and lastly $\boldsymbol{x} = (x_1, x_2, ..., x_n)^T$ where we have n instances. Then our hyperplane function is,

$$\begin{aligned} s(\boldsymbol{x}) &= \boldsymbol{w}^T \boldsymbol{x} + b \\ &= \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n + b \end{aligned}$$

Therefore, for any points that are on the hyperplane,

$$s(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{x} + b = 0$$

Let $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ be two arbitrary points in the hyperplane (in the feature space our points have (height, weight)),

$$s(\boldsymbol{x}_1) = \boldsymbol{w}^T \boldsymbol{x}_1 + b = 0$$

$$s(\boldsymbol{x}_2) = \boldsymbol{w}^T \boldsymbol{x}_2 + b = 0$$

If we subtract our two linear decision functions for our arbitrary points (given they both equal zero),

$$\boldsymbol{w}^T\boldsymbol{x}_1 + b - \boldsymbol{w}^T\boldsymbol{x}_2 - b = 0$$

We get,

$$\boldsymbol{w}^T(\boldsymbol{x}_1 - \boldsymbol{x}_2) = 0$$

This implies that $\boldsymbol{w}$ is orthogonal to the hyperplane because it is orthogonal to any arbitrary vector $(\boldsymbol{x}_1 - \boldsymbol{x}_2)$ on the hyperplane.

Proof. Let $s(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b$ be the hyperplane function (linear decision function).
Then for any points that are on the hyperplane,

$$s(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b = 0$$

Consider two arbitrary points in the hyperplane $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ then,

$$s(\boldsymbol{x}_1) = \boldsymbol{w}^T\boldsymbol{x}_1 + b = 0$$
$$s(\boldsymbol{x}_2) = \boldsymbol{w}^T\boldsymbol{x}_2 + b = 0$$

Then given $s(\boldsymbol{x}_1) = s(\boldsymbol{x}_2) = 0$ we can subtract the two hyperplane functions to arrive at,

$$\boldsymbol{w}^T\boldsymbol{x}_1 + b - \boldsymbol{w}^T\boldsymbol{x}_2 - b = 0$$

Which simplifies to,

$$\boldsymbol{w}^T(\boldsymbol{x}_1 - \boldsymbol{x}_2) = 0$$

Then $\boldsymbol{w}$ is orthogonal to the hyperplane and is orthogonal to any arbitrary difference vector $(\boldsymbol{x}_1 - \boldsymbol{x}_2)$ on the hyperplane.
Therefore, the weight vector $\boldsymbol{w}$ is orthogonal to every vector within the decision boundary. □

Part b:

Show that the distance from the origin to the decision boundary is given by,

$$\frac{|b|}{||\boldsymbol{w}||_2}$$

Proof. Let the hyperplane function be $s(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b$ and let $\boldsymbol{x}_H$ be the orthogonal projection of arbitrary point $\boldsymbol{x}$ on the hyperplane. Then for any point on the hyperplane $s(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b = 0$. Therefore, $d = \boldsymbol{x}_h - \boldsymbol{x}$ is a scalar that represents the directed distance along $\boldsymbol{w}$ from $\boldsymbol{x}$ to $\boldsymbol{x}_H$. Such that,

$$\boldsymbol{x} = \boldsymbol{x}_H + d\frac{\boldsymbol{w}}{||\boldsymbol{w}||_2}$$

Where $\frac{\boldsymbol{w}}{||\boldsymbol{w}||_2}$ is the unit weight vector. Now we can use such $\boldsymbol{x}$,

$$s(\boldsymbol{x}) = \boldsymbol{w}^T(\boldsymbol{x}_H + d\frac{\boldsymbol{w}}{||\boldsymbol{w}||_2}) + b$$

Rearranging our terms,

$$s(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x}_H + b + d\frac{\boldsymbol{w}^T\boldsymbol{w}}{||w||_2}$$

Where $\boldsymbol{w}^T\boldsymbol{x}_H + b = 0$ and $d\frac{\boldsymbol{w}^T\boldsymbol{w}}{||w||_2} = d\frac{||\boldsymbol{w}||_2^2}{||\boldsymbol{w}||_2} = d||\boldsymbol{w}||_2$
Then solving for d,

$$d = \frac{s(\boldsymbol{x})}{||\boldsymbol{w}||_2}$$

However, the above gives us the signed distance so to get absolute distance we can use,

$$|d| = \frac{|s(\boldsymbol{x})|}{||\boldsymbol{w}||_2}$$

Now at the origin we have $\boldsymbol{x} = 0$ and thus $\boldsymbol{w}^T 0 = 0$ then the absolute distance is,

$$|d| = \frac{|s(\mathbf{0})|}{||\boldsymbol{w}||_2} = \frac{|\boldsymbol{w}^T 0 + b|}{||\boldsymbol{w}||_2} = \frac{|b|}{||\boldsymbol{w}||_2}$$

That is, at the origin the only contributing factor to the distance from the decision boundary is our bias term $b = \beta_0$ scaled by the magnitude of the weight vector $\boldsymbol{w}$.

Therefore, the distance from the origin to the decision boundary is given by $|d| = \frac{|b|}{||\boldsymbol{w}||_2}$ as desired. □

Part c:

Let $\boldsymbol{x}$ be an arbitrary point. Show that the distance from $\boldsymbol{x}$ to the decision boundary is,

$$\frac{|s(\boldsymbol{x})|}{||\boldsymbol{w}||_2}$$

Proof. Let the hyperplane function be $s(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b$ and let $\boldsymbol{x}_H$ be the orthogonal projection of arbitrary point $\boldsymbol{x}$ on the hyperplane. Then for any point on the hyperplane $s(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b = 0$. Therefore, $d = \boldsymbol{x}_h - \boldsymbol{x}$ is a scalar that represents the directed distance along $\boldsymbol{w}$ from $\boldsymbol{x}$ to $\boldsymbol{x}_H$. Such that,

$$\boldsymbol{x} = \boldsymbol{x}_H + d\frac{\boldsymbol{w}}{||\boldsymbol{w}||_2}$$

Where $\frac{\boldsymbol{w}}{||\boldsymbol{w}||_2}$ is the unit weight vector. Now we can use such $\boldsymbol{x}$,

$$s(\boldsymbol{x}) = \boldsymbol{w}^T(\boldsymbol{x}_H + d\frac{\boldsymbol{w}}{||\boldsymbol{w}||_2}) + b$$

Rearranging our terms,

$$s(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x}_H + b + d\frac{\boldsymbol{w}^T\boldsymbol{w}}{||w||_2}$$

Where $\boldsymbol{w}^T\boldsymbol{x}_H + b = 0$ and $d\frac{\boldsymbol{w}^T\boldsymbol{w}}{||w||_2} = d\frac{||\boldsymbol{w}||_2^2}{||\boldsymbol{w}||_2} = d||\boldsymbol{w}||_2$
Then solving for d,

$$d = \frac{s(\boldsymbol{x})}{||\boldsymbol{w}||_2}$$

However, the above gives us the signed distance so to get absolute distance we can use,

$$|d| = \frac{|s(\boldsymbol{x})|}{||\boldsymbol{w}||_2}$$

Therefore, given an arbitrary point $\boldsymbol{x}$ then the distance of $\boldsymbol{x}$ to the decision boundary is $|d| = \frac{|s(\boldsymbol{x})|}{||\boldsymbol{w}||_2}$ as desired. □

Problem 3

Part a:

What is t_i^2?

My Solution: Note that t_i is -1 for negative instances ($y_i = 0$) and is 1 for positive instances $y_i = 1$ with the distance of a point x_i to the decision surface being $\frac{t_i s(\boldsymbol{x}_i)}{||\boldsymbol{w}||_2}$ therefore t_i^2 will be 1.

Part b:

Use $t_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) = 1$ and $\boldsymbol{w} = \sum_{i=1}^N a_i t_i \boldsymbol{x}_i$ and part (a) to show that

$$b = \frac{1}{N_S}\sum_{i\in S}\left(t_i - \sum_{j=1}^N a_j t_j k(\mathbf{x}_i, \mathbf{x}_j)\right)$$

where S denotes the set of support vectors and N_S is the total number of support vectors. Further, the kernel function is $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^T\boldsymbol{x}_j$.

My Solution:

Scratch Work:
Information given,

(1.) $t_i(\mathbf{w}^T\mathbf{x}_i + b) = 1$
(2.) $\mathbf{w} = \sum_{i=1}^{N} a_i t_i \mathbf{x}_i$
(3.) Kernel Function: $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^T \boldsymbol{x}_j$
(4.) $t_i^2 = 1$

Proof. Let $t_i^2 = 1$ so multiply both sides of **(1.)** by t_i to get,

$$t_i^2(\mathbf{w}^T\mathbf{x}_i + b) = t_i$$

$$\mathbf{w}^T\mathbf{x}_i + b = t_i$$

Isolating for b we get,

$$b_i = t_i - \boldsymbol{w}^T\boldsymbol{x}_i$$

Note above this accounts for a single support vector $i \in S$ where S is the set of all support vectors. Now we can use **(2.)** and substitute this in our expression for b_i where i indexes the support vector and j indexes over all of the training samples,

$$b_i = t_i - (\sum_{j=1}^{N} a_j t_j \boldsymbol{x}_j^T)\boldsymbol{x}_i$$

Using the kernel function defined in **(3.)** and the fact that the dot product is commutative we have,

$$b_i = t_i - \sum_{j=1}^{N} a_j t_j k(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

The above is the bias for one support vector $i \in S$. We can take b as the average bias across all support vectors. Therefore, let N_S represent the total number of support vectors, the bias is as follows,

$$b = \frac{1}{N_S} \sum_{i \in S} (t_i - \sum_{j=1}^{N} a_j t_j k(\boldsymbol{x}_i, \boldsymbol{x}_j))$$

as desired. □
