Rohan Mahendra Chaudhari

USCID : 6675-5653-85

1 Problem 1 - Support Vector Machines

Consider a dataset consisting of points in the form of (x, y), where x is a real value, and y $\in \{1, 1\}$ is the class label. There are only three points (x_1, y_1) = (-1, -1), (x_2, y_2) = (1, -1), and (x_3, y_3) = (0, 1), shown in Figure 1.

1.1 Can these three points in their current one-dimensional feature space be perfectly separated with a linear classifier? Why or why not?

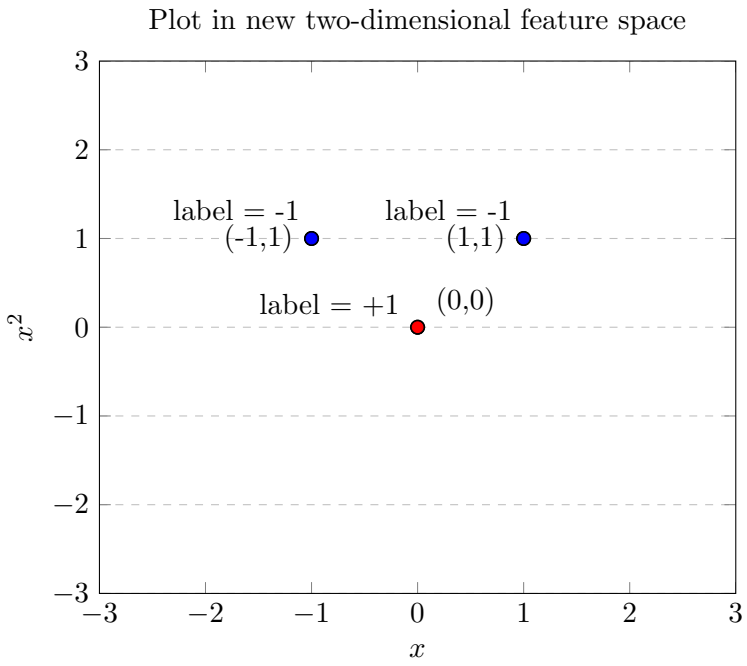
Solution

These 3 points, in its current one-dimensional feature space, cannot be perfectly separated using a linear separator. The data changes class twice along only one dimension; a linear classifier in one dimension can only represent a single split and thus the 3 points given cannot be perfectly separated with a linear classifier.

1.2 Now we define a simple feature mapping $\phi(x) = [x, x^2]^T$ to transform the three points from one-dimensional to two-dimensional feature space. Plot the transformed points in the new two-dimensional feature space. Is there a linear model $w^T\mathbf{x} + \mathbf{b}$ for some $\mathbf{w} \in R^2$ and $b \in R$ that can correctly separate the three points in this new feature space? Why or why not?

Solution

We can simply apply the function map $\phi(x)$ to each of the 3 points to obtain the below graph. Notice how the feature map wraps the original space.



As per the above graph, we have plotted the 3 points in a two-dimensional space. We can conclude that yes, we can correctly separate the 3 points in the new feature space. Any point between y = 0 and y = 1 can correctly separate the data.

1.3 Given the feature mapping $\phi(x) = [x, x^2]^T$, write down the 3 x 3 kernel/Gram matrix K for this dataset.

Solution

The kernel function is k(x,$x^{'}$) = $\boldsymbol{\phi}(x)^T\boldsymbol{\phi}(x)$ = x$x^{'}$ + (x$x^{'}$)^2

The Gram Matrix is

$$\boldsymbol{K} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

1.4 Now write down the primal and dual formulations of SVM for this dataset in the two-dimensional feature space. Note that when the data is separable, we set the hyperparameter C to be $+\infty$ which makes sure that all slack variables (ξ) in the primal formulation have to be 0 (and thus can be removed from the optimization)

Solution

Primal formulation of SVM for separable data

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2$$

$$\text{such that, } y_n[\boldsymbol{w}^T\boldsymbol{\phi}(\boldsymbol{x}_n) + b] \geq 1, \forall n$$

Substituting the given values,

$$\min_{w_1,w_2,b} \frac{1}{2}(w_1^2 + w_2^2)$$

$$\text{such that, } w_1 + w_2 + b \leq -1$$
$$w_1 - w_2 - b \geq 1$$
$$b \geq 1$$

Dual Formulation of SVM for separable data

$$\max_{\alpha} \sum_n \alpha_n - \frac{1}{2}\sum_{m,n} y_m y_n \alpha_m \alpha_n k(\boldsymbol{x}_m, \boldsymbol{x}_n)$$

$$\text{such that, } \alpha_n \geq 0, \forall n$$

$$\sum_n \alpha_n y_n = 0$$

Substituting the given values,

$$\max_{\alpha_1,\alpha_2,\alpha_3 \geq 0} \alpha_1 + \alpha_2 + \alpha_3 - \alpha_1^2 - \alpha_2^2$$

$$\text{such that, } \alpha_1 + \alpha_2 = \alpha_3$$

1.5 Next, solve the dual formulation exactly (note: while this is not generally feasible as discussed in the lecture, the simple form of this dataset makes it possible). Based on that, calculate the primal solution.

Solution

To solve the dual formulation, we use the constraint $\alpha_1 + \alpha_2 = \alpha_3$ to eliminate α_3,

$$\max_{\alpha_1,\alpha_2\geq 0} 2\alpha_1 - \alpha_1^2 + 2\alpha_2 - \alpha_2^2$$

We then maximize over α_1 and α_2 independently thus getting, $\alpha_1^* = 1$ & $\alpha_2^* = 1$ and thus by using the relation $\alpha_1 + \alpha_2 = \alpha_3$ we get, $\alpha_3^* = 2$

The primal solution,

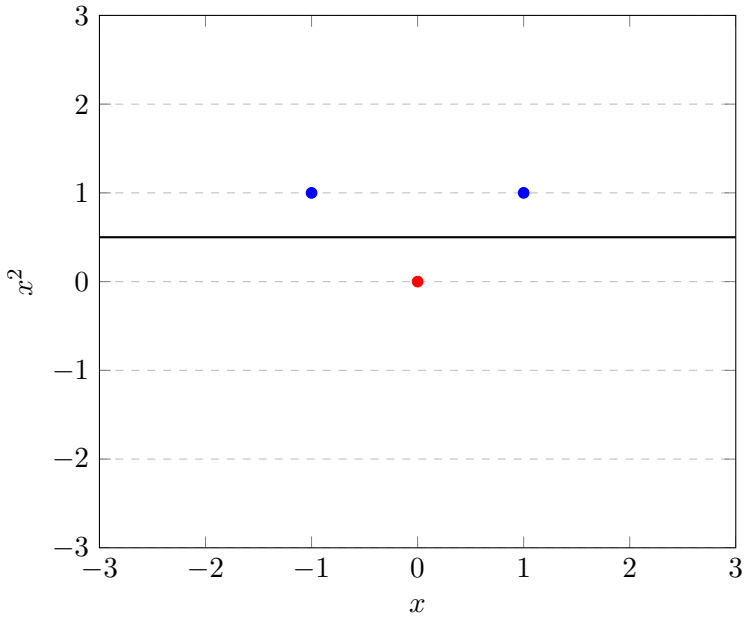
$$(w_1^*, w_2^*)^T = \sum_{n=1}^{3} y_n \alpha_n^* \boldsymbol{\phi}(\boldsymbol{x}_n) = (0, -2)^T$$
$$b^* = y_1 - \boldsymbol{w}^{*T}\boldsymbol{\phi}(\boldsymbol{x}_1) = 1$$

1.6 Plot the decision boundary (which is a line) of the linear model $\boldsymbol{w}^{*T}\boldsymbol{x} + b^*$ in the two-dimensional feature space, where w^* and b^* are the primal solution you got from the previous question. Then circle all support vectors. Finally, plot the corresponding decision boundary in the original one-dimensional space (which are just all the points x such that $\boldsymbol{w}^{*T}\boldsymbol{\phi}(x) + b^* = 0$).

Solution

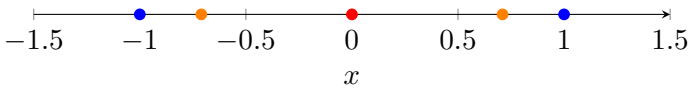
The decision boundary of the linear model in the two-dimensional feature space is a horizontal line at y = $\frac{1}{2}$. All points classified below it belong to the same class (positive : +1) and everything above it is classified to the same class (negative : -1). In this case, all the 3 points are support vectors.

Plot in new two-dimensional feature space



On solving $\boldsymbol{w}^{*T}\boldsymbol{\phi}(\boldsymbol{x}) + b^* = -2x^2 + 1 = 0$ we obtain the decision boundary in the original one-dimensional space that consists of 2 points, $\frac{1}{\sqrt{2}}$ and -$\frac{1}{\sqrt{2}}$. Any point between $\frac{1}{\sqrt{2}}$ and -$\frac{1}{\sqrt{2}}$ are classified as positive else negative.

Plot in new one-dimensional feature space



2 Problem 2 - Decision Trees

Consider a binary dataset with 400 examples, where half of them belongs to class A and another half belongs to class B.
Next consider two decision stumps (i.e. trees with depth 1) T_1 and T_2, each with two children. For T_1, its left child has 150 examples in class A and 50 examples in class B; for T_2, its left child has 0 example in class A and 100 examples in class B. (You can infer what are in the right child.)

2.1 For each leaf of T_1 and T_2, compute the corresponding classification error, entropy (base e) and Gini impurity. You can either exactly express the final numbers in terms of fractions and logarithms, or round them to two decimal places. (Note: the value/prediction of each leaf is the majority class among all examples that belong to that leaf.)

Solution
Classification Error:
$error_{1,left} = \frac{50}{150+50} = 0.25$
$error_{1,right} = \frac{50}{150+50} = 0.25$

$error_{2,left} = \frac{0}{0+100} = 0$
$error_{2,right} = \frac{100}{200+100} \approx 0.33$

Entropy:
$entropy_{1,left} = -\frac{150}{150+50} ln(\frac{150}{150+50}) - \frac{50}{150+50} ln(\frac{50}{150+50}) \approx 0.56$
$entropy_{1,right} = -\frac{50}{150+50} ln(\frac{50}{150+50}) - \frac{150}{150+50} ln(\frac{150}{150+50}) \approx 0.56$
$entropy_{2,left} = -\frac{0}{0+100} ln(\frac{0}{0+100}) - \frac{100}{0+100} ln(\frac{100}{0+100}) = 0$
$entropy_{2,right} = -\frac{200}{200+100} ln(\frac{200}{200+100}) - \frac{100}{100+200} ln(\frac{100}{100+200}) \approx 0.64$

Gini Impurity:
$gini_{1,left} = 1 - (\frac{150}{150+50})^2 - (\frac{50}{150+50})^2 = 0.375 \approx 0.38$
$gini_{1,right} = 1 - (\frac{50}{150+50})^2 - (\frac{150}{150+50})^2 = 0.375 \approx 0.38$
$gini_{2,left} = 1 - (\frac{0}{0+100})^2 - (\frac{100}{0+100})^2 = 0$
$gini_{2,right} = 1 - (\frac{200}{200+100})^2 - (\frac{100}{200+100})^2 \approx 0.44$

2.2 Compare the quality of T_1 and T_2 (that is, the two different splits of the root) based on classification error, conditional entropy (base e), and weighted Gini impurity respectively.

Solution
Fraction of examples that belong to the left leaf of T_1 : $p_1 = \frac{150+50}{400} = 0.5$
Fraction of examples that belong to the left leaf of T_2 : $p_2 = \frac{0+100}{400} = 0.25$

Then the total classification error for T_1 and T_2 are respectively:
$error_1 = p_1 error_{1,left} + (1 - p_1) error_{1,right} = 0.25$
$error_2 = p_2 error_{2,left} + (1 - p_2) error_{2,right} = 0.25$
So, we can conclude that they are equally good in terms of classification error.

The conditional entropy for T_1 and T_2 are respectively:
$entropy_1 = p_1 entropy_{1,left} + (1 - p_1) entropy_{1,right} \approx 0.56$
$entropy_2 = p_2 entropy_{2,left} + (1 - p_2) entropy_{2,right} = 0.48$
So, we can conclude that T_2 is better in terms of conditional entropy

The weighted Gini Impurity for T_1 and T_2 are respectively:
$gini_1 = p_1 gini_{1,left} + (1 - p_1) gini_{1,right} \approx 0.38$
$gini_2 = p_2 gini_{2,left} + (1 - p_2) gini_{2,right} \approx 0.33$
So, we can conclude that T_2 is better in terms of Gini Impurity.

3 Problem 3 - Boosting

3.1 We discussed in class that AdaBoost minimizes the exponential loss greedily. In particular,the derivation of β_t is by finding the minimizer of $\epsilon_t(e^{\beta_t} - e^{-\beta_t}) + e^{-\beta_t}$ where ϵ_t is the weighted classification error of h_t and is fixed. Show that $\beta_t^* = \frac{1}{2}\ln(\frac{1-\epsilon_t}{\epsilon_t})$ is the minimizer. (You can use the fact that the function above is convex.)

Solution

Set the derivative to 0:

$\epsilon_t(e^{\beta_t} + e^{-\beta_t}) - e^{-\beta_t} = 0$

Multiplying both sides by e^{β_t} gives,

$e^{2\beta_t} = \frac{1}{\epsilon_t} - 1$

Taking ln on both sides,

$2\beta_t = ln(\frac{1-\epsilon_t}{\epsilon_t})$

$\beta_t^* = \frac{1}{2}ln(\frac{1-\epsilon_t}{\epsilon_t})$

Hence Proved.

3.2 Recall that at round t of AdaBoost, a classifier h_t is obtained and the weighting over the training set is updated from D_t to D_{t+1}. Prove that h_t is only as good as random guessing in terms of classification error weighted by D_{t+1}. That is

$$\sum_{n:h_t(x_n)\neq y_n} D_{t+1}(n) = \frac{1}{2}$$

In other words, the update is so that D_{t+1} is the "hardest" weighing for h_t.

Solution

$$\sum_{n:h_t(x_n)\neq y_n} D_{t+1}(n) \propto \sum_{n:h_t(x_n)\neq y_n} D_t(n)e^{\beta_t} = \epsilon_t e^{\beta_t} = \sqrt{\epsilon_t(1-\epsilon_t)}$$

Similarly,

$$\sum_{n:h_t(x_n)=y_n} D_{t+1}(n) \propto \sum_{n:h_t(x_n)=y_n} D_t(n)e^{-\beta_t} = (1-\epsilon_t)e^{-\beta_t} = \sqrt{(1-\epsilon_t)\epsilon_t}$$

$$\sum_{n:h_t(x_n)\neq y_n} D_{t+1}(n) + \sum_{n:h_t(x_n)=y_n} D_{t+1}(n) = 1$$

Thus,

$$\sum_{n:h_t(x_n)\neq y_n} D_{t+1}(n) = \sum_{n:h_t(x_n)=y_n} D_{t+1}(n) = \frac{1}{2}$$