CIS5200: Machine Learning Spring 2023

Lecture 8: Support Vector Machines

Date: February 7, 2023 *Author: Surbhi Goel*

Acknowledgements. These notes are heavily inspired by material from CIS5200 Spring 2022 and Cornell University's CS 4/5780 — Spring 2022.

Disclaimer. These notes have not been subjected to the usual scrutiny reserved for formal publications. If you notice any typos or errors, please reach out to the author.

1 Support Vector Machines

In this lecture, we will go back to the binary classification paradigm with the Perceptron algorithm that we studied in Lecture 2. Recall that the Perceptron algorithm guaranteed to find a hyperplane that separates the data as long as one exists. In fact, since we assumed separation with margin, there are infinitely many hyperplanes that separate the data, and the Perceptron may find any one of these. In particular, Perceptron algroithm did not provide any guarantees on the margin of the hyperplane found, even though we assumed that the training dataset had margin γ with respect to the true separator. In this lecture, we will look at a machine learning method that finds the hyperplane with the maximum margin.

Note: In Homework 1, we saw a simple modification of the Perceptron to the Margin Perceptron which guaranteed an approximate margin of $\gamma/3$. Here we will guarantee exact maximum margin of γ.

1.1 Setting

We will study binary classification with linear classifiers,

- Input space $\mathcal{X} = \mathbb{R}^d$
- Label space $\mathcal{Y} = \{-1, 1\}$[1]
- Hypothesis class $\mathcal{F} := \{x \mapsto \text{sign}(w^\top x + b) \mid w \in \mathbb{R}^d, b \in \mathbb{R}\}$[2] where $\text{sign}(a) = \begin{cases} 1 & \text{if } a \geq 0, \\ -1 & \text{otherwise.} \end{cases}$.
- Loss function $\ell(y, y) = \begin{cases} 0 & \text{if } y = y, \\ 1 & \text{otherwise.} \end{cases}$

[1] In the last lecture we used $\{0, 1\}$. For technical reasons, it will be easier to use $\{-1, 1\}$.

[2] Here we will keep the bias term, it will become clear why this is important as we go through the analysis.

Margin. Recall that the margin of a hyperplane $w^\top x + b = 0$ is the minimum distance of any point in the dataset to the hyperplane. Recall in Homework 0, you computed the distance of point x to the hyperplane to be $\frac{|w^\top x+b|}{\|w\|_2}$. This gives the margin as,

$$\gamma(w, b) = \min_{i\in[m]} \frac{|w^\top x_i + b|}{\|w\|_2}.$$

1.2 Max-margin classifier: separable case

Let us assume that the data is separable by a hyperplane $w_*^\top x + b_* = 0$. The goal here is to find the classifier that achieves maximum margin. We can formulate this as the following optimization problem:

$$\begin{aligned} \text{maximize over } w, b \quad & \underbrace{\gamma(w, b)}_{\text{margin}} \\ \text{such that} \quad & \underbrace{\forall i \in [m], y_i(w^\top x_i + b) \geq 0}_{(w,b) \text{ is a separating hyperplane}}. \end{aligned}$$

This gives us,

$$\begin{aligned} \text{maximize over } w, b \quad & \underbrace{\frac{1}{\|w\|_2} \min_{i\in[m]} |w^\top x_i + b|}_{\text{margin}} \\ \text{such that} \quad & \underbrace{\forall i \in [m], y_i(w^\top x_i + b) \geq 0}_{(w,b) \text{ is a separating hyperplane}}. \end{aligned}$$

Note that in the above problem, if (w, b) is a solution then $(\alpha w, \alpha b)$ is also a solution. This is because the scale of the hyperplane does not affect the prediction (we are using only the sign of the value). To make this solution unique, we can fix the scale of this problem. One way to do this is to set the following constraint,

$$\min_{i\in[m]} |w^\top x_i + b| = 1.$$

Substituting this constraint in our optimization gives us,

$$\begin{aligned} \text{maximize over } w, b \quad & \frac{1}{\|w\|_2} \\ \text{such that} \quad & \forall i \in [m], y_i(w^\top x_i + b) \geq 0 \\ & \min_{i\in[m]} |w^\top x_i + b| = 1. \end{aligned}$$

This is equivalent to,

$$\begin{aligned} \text{minimize over } w, b \quad & \frac{1}{2}\|w\|_2^2 \\ \text{such that} \quad & \forall i \in [m], y_i(w^\top x_i + b) \geq 0 \\ & \min_{i\in[m]} |w^\top x_i + b| = 1. \end{aligned}$$

This problem is a quadratic optimization problem (we saw before) but it is non-convex because of the last constraint.[3] There is a clever way to rewrite this to make it convex.

$$\begin{aligned} \text{minimize over } w, b \qquad & \frac{1}{2}\|w\|_2^2 \\ \text{such that} \qquad & \forall i \in [m], y_i(w^\top x_i + b) \geq 1. \end{aligned}$$

Now this is a convex objective with convex constraints making this problem a convex quadratic optimization problem. It's a good exercise to check why these are equivalent. *Hint: Try to prove using contradiction.*

Dual formulation. It is often helpful to view the constrained optimization in its dual form. The constrained optimization problem for the SVM satisfies strong duality, implies that the solution to the primal is the same as the dual. Now let us write the dual formulation. First we will define the Lagrangian

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|_2^2 + \sum_{i=1}^{m} \alpha_i (1 - y_i(w^\top x_i + b))$$

where $\alpha_i \geq 0$ are the dual variables. Now, the Lagrange (dual) function is

$$D(\alpha) = \min_{w,b} \mathcal{L}(w, b, \alpha).$$

To get a solution for the above, since it is unconstrained, we can take the derivative with respect to w and b and set them to 0.

$$\nabla_w \mathcal{L}(w, b, \alpha) = w - \sum_{i=1}^{m} \alpha_i y_i x_i = 0 \implies w = \sum_{i=1}^{m} \alpha_i y_i x_i$$

$$\nabla_b \mathcal{L}(w, b, \alpha) = -\sum_{i=1}^{m} \alpha_i y_i = 0 \implies \sum_{i=1}^{m} \alpha_i y_i = 0.$$

Substituting this back and doing some algebra (*you should work this out*), we have

$$D(\alpha) = -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j (x_i^\top x_j) + \sum_{i=1}^{m} \alpha_i.$$

under the constraint that $\sum_{i=1}^{m} \alpha_i y_i = 0$. The dual is then

$$\begin{aligned} \text{maximize over } \alpha \qquad & -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j (x_i^\top x_j) + \sum_{i=1}^{m} \alpha_i \\ \text{such that} \qquad & \sum_{i=1}^{m} \alpha_i y_i = 0 \\ & \forall i \in [m], \alpha_i \geq 0. \end{aligned}$$

[3] Try to show why this constraint is not convex. Recall that a constraint $c(w) = 0$ is convex if for any w, w' such that $c(w) = c(w') = 0$, $c(\alpha w + (1-\alpha)w') = 0$ for all $\alpha \in [0, 1]$. This essentially means that the set $S = \{w : c(w) = 0\}$ is convex.

Here we have reduced the problem from solving for m variables instead of $d+1$ that we had before. It might seem wasteful if $m > d+1$. In the next class, we will show that in fact this dual form turns out to be pretty useful when we are learning a linear function over a feature map $\phi(x)$ instead of directly on x, and ϕ can map the d-dimensional input into a much larger space. Also, this is useful when the number of samples is actually less than d.

Suppose α is the solution we find for the above optimization, then we get

$$w = \sum_{i=1}^{m} \alpha_i y_i x_i.$$

Something interesting to notice here is that w is essentially a linear combination of x_i's (our datapoints). So it is always in the subspace spanned by $x_1, \ldots, x_m$. The dual formulation does not have b, so how do we estimate that? This bring us to an important concept of s**upport vector**, where the name of this approach comes from.

Support Vectors. Any datapoints for which $\alpha_i > 0$, then we call it a support vector. Observe that w only depends on the support vectors, and is independent of any other datapoints. Let us call this set $SV = \{i \in [m] : \alpha_i > 0\}$.

By complementary slackness condition in the KKT conditions (from last lecture), we have, for all $i \in SV$

$$1 - y_i(w^\top x_i + b) = 0 \implies y_i - y_i^2(w^\top x_i + b) = 0 \implies b = y_i - w^\top x_i.$$

Now we can estimate b using any of the support vectors. We could also average over them to get a more stable estimator in case of noise in estimators.

Note that the above also implies that all support vectors are equidistant from the decision boundaries, and in fact the points closest to the decision boundary.

1.3 Soft-margin classifier: non-separable case

Suppose the data was not separable, then the constraint in the optimization problem in the previous section, $\forall i \in [m], y_i(w^\top x_i + b) \geq 1$ would be infeasible. In order to accommodate non-separable data, one way is to add some slack in the constraints making them soft constraints. In particular, we can relax the constraints to be,

$$\forall i \in [m], y_i(w^\top x_i + b) \geq 1 - \xi_i.$$

Here $\xi_i \geq 0$ is the slack variable corresponding to datapoint x_i. Observe that $\xi_i = 0$ implies that the datapoint i is correctly classified and satisfies the large margin constraint. $\xi < 1$ implies that the point is correctly classified but with margin but does not satisfy the large margin constraint. Lastly, $\xi_i > 1$ implies that the point is incorrectly classified.

Given these slack variables, we would ideally want the slack to not be too large. This means that

we want to assign some cost to having large slack. This gives us the following objective,

$$\begin{aligned} \text{minimize over } w, b, \xi_1, \ldots, \xi_m \qquad & \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{m} \xi_i \\ \text{such that} \qquad & \forall i \in [m], y_i(w^\top x_i + b) \geq 1 - \xi_i \\ & \forall i \in [m], \xi_i \geq 0. \end{aligned}$$

Here $C \geq 0$ controls how much the slack variables are penalized, and is generally a hyperparameter of this problem that needs to be tuned. Note that this is still a convex quadratic optimization problem. We can write the dual formulation of this as well:

$$\begin{aligned} \text{maximize over } \alpha \qquad & -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j (x_i^\top x_j) + \sum_{i=1}^{m} \alpha_i \\ \text{such that} \qquad & \sum_{i=1}^{m} \alpha_i y_i = 0 \\ & \forall i \in [m], 0 \leq \alpha_i \leq C. \end{aligned}$$

Exercise: Work out this dual formulation for the soft-margin SVMs.

Loss minimization view. Soft-SVM can alternately be viewed as minimizing a regularized hinge loss.

$$\ell_{\mathsf{hinge}}(y, y) = \max(0, 1 - yy).$$

Then we can view this as,

$$\min_{w,b} \frac{1}{m}\sum_{i=1}^{m} \max(0, 1 - y_i(w^\top x_i + b)) + \lambda\|w\|^2.$$

Introducing slack variables, we can rewrite this as.

$$\begin{aligned} \text{minimize over } w, b, \xi_1, \ldots, \xi_m \qquad & \lambda\|w\|_2^2 + \frac{1}{m}\sum_{i=1}^{m} \xi_i \\ \text{such that} \qquad & \forall i \in [m], \xi_i \geq 1 - y_i(w^\top x_i + b) \\ & \forall i \in [m], \xi_i \geq 0. \end{aligned}$$

This is essentially the soft-SVM formulation with $C = \frac{1}{2\lambda m}$!