# CS 4780/5780 Homework 7

Due: Thursday 04/12/18 11:55pm on Gradescope

## Problem 1: Derivation for Hard-margin Linear SVMs

**a)** Suppose your data is linearly separable. Why might you prefer the hard-margin linear SVM solution over the solution found by the Perceptron?

**b)** The most intuitive definition of the hard-margin linear SVM is to "maximize the margin of the hyperplane, subject to the constraint that every training point is classified correctly." Write this objective mathematically, and label which parts of your formulation correspond to each part of our intuitive definition. (Hint: it's in the notes.)

**c)** Though the maximum margin separating hyperplane is unique, the $(\mathbf{w}, b)$ we use to define it is not unique unless we do something like restrict the norm of $\mathbf{w}$. That's precisely what we do when we include the constraint $\min_i \|\mathbf{w}^\top \mathbf{x}_i + b\| = 1$. What have we restricted $\|\mathbf{w}\|_2$ to equal? (Hint: recall the definition of the margin of a hyperplane.)

**d)** At this point we can do some quick algebra to get the following formulation:

$$\begin{aligned} &\min_{\mathbf{w},b} \mathbf{w}^\top \mathbf{w} \\ \text{s.t.} \quad &\forall_i \; y_i(\mathbf{w}^T \mathbf{x}_i + b) \quad \geqslant 0 \\ &\min_i \left|\mathbf{w}^T \mathbf{x}_i + b\right| \quad = 1 \end{aligned}$$

Prove that for the optimal solution, these constraints are equivalent to $\forall_i \; y_i(\mathbf{w}^T \mathbf{x}_i + b) \geqslant 1$.

## Problem 2: Hard- vs. Soft-margin SVMs

**a)** The Perceptron algorithm does not converge on non-linearly separable data. Does the hard-margin SVM converge on non-linearly separable data? How about the soft-margin SVM? Why?

**b)** For the soft-margin linear SVM, we use the hyperparameter $C$ to tune how much we penalize misclassifications. As $C \to \infty$, does the soft-margin SVM become more similar or less similar to the hard-margin SVM? As $C \to 0^+$, what happens to the solution of the soft-margin SVM? Why?

**c)** How would you go about picking a good value for $C$?

## Problem 3: Thinking in Terms of the "Buffer"

**a)** Suppose you found a solution $(\hat{\mathbf{w}}, \hat{b})$ to the hard-margin SVM. The separating hyperplane is surrounded by a buffer defined by hyperplanes $\{\mathbf{x} : \hat{\mathbf{w}}^\top \mathbf{x} + \hat{b} = 1\}$ and $\{\mathbf{x} : \hat{\mathbf{w}}^\top \mathbf{x} + \hat{b} = -1\}$. Prove that at least one training datapoint lies on each of these buffer hyperplanes.

**b)** Your TA is unimpressed with your solution and says you didn't minimize $\|\mathbf{w}\|_2$ enough, so he tells you to use $(\mathbf{w}_{TA}, b_{TA}) = (0.9 * \hat{\mathbf{w}}, 0.9 * \hat{b})$. Prove that, though this solution corresponds to the same hyperplane, it has violated a constraint in the hard-margin SVM optimization problem. What happened to the buffer, when you rescaled (decreased) the norm of the solution?

**c)** Your TA doesn't understand your proof, so you decide to give him a pictorial explanation. Draw a reasonable, linearly separable, binary class, 2D dataset. Draw the hyperplane and buffer corresponding to the hard-margin SVM solution $(\hat{\mathbf{w}}, \hat{b})$ for this dataset. Then on another plot, draw what the hyperplane and buffer might look like if we decrease the norm of the solution, e.g. $(\mathbf{w}_{TA}, b_{TA}) = (0.9 * \hat{\mathbf{w}}, 0.9 * \hat{b})$.

**d)** Your TA is embarrassed, so you graciously say that he obviously meant that you should use a soft-margin SVM. Why is $(\mathbf{w}_{TA}, b_{TA})$ now, in the case of soft-margin SVMs, a reasonable suggestion? Are training datapoints "allowed past the buffer" for soft-margin SVMs?

## Problem 4: MLE in terms of ERM

Recall that a generative model assigns a probability to every features-label pair $(x, y)$. Suppose that we would like to learn a generative model $g$. What constitutes a "good" $g$? In class you have discussed a few learning principles. In this problem, you will show that there is a cool connection between MLE, ERM with a particular loss function, and minimizing the KL-divergence. First, some background:

The KL-divergence is a measure of how much the distribution $p$ diverges from the distribution $q$. It is defined as

$$D_{KL}(p||q) = \mathbb{E}_{(x,y)\sim p}\left[\log \frac{p(x,y)}{q(x,y)}\right].$$

True risk is defined as the expected loss of your classifier, $\mathbb{E}_{(x,y)\sim\pi}\left[\ell(h(x), y)\right]$ – where $\pi$ is the true test-time distribution, $\ell$ is the loss function of interest, and $h$ is the learned function. Ideally, we would learn a function which minimizes the true risk

$$\hat{h} = \arg\min_h \mathbb{E}_{(x,y)\sim\pi}\left[\ell(h(x), y)\right].$$

We usually can't evaluate this directly because we don't know $\pi$. Instead, we consider the empirical risk, $\sum_{(x,y)\in D} \ell(h(x), y)$ – where $D$ is a training dataset of examples. If $D$ is large and drawn iid from $\pi$, then the empirical risk approaches the true risk. In ERM, our learning algorithm attempts to find a function $\hat{h}$ which minimizes the empirical risk

$$\hat{h} = \arg\min_h \sum_{(x,y)\in D} \ell(h(x), y).$$

**a)** Suppose our learning goal is to minimize the KL-divergence from our model distribution to the true data distribution,

$$\hat{g} = \arg\min_g \mathbb{E}_{(x,y)\sim\pi}\left[\log \frac{\pi(x,y)}{g(x,y)}\right].$$

Show that this is, in fact, a type of true risk minimization. In particular: what is the predictive function $h$? what is the loss function $\ell$?

**b)** What is the empirical risk minimization problem, for the true risk minimization problem you defined in part (a)?

**c)** Show that the solution to the empirical risk minimization problem is equivalent to the solution to the maximum likelihood optimization problem

$$\hat{g} = \arg\max_g \sum_{(x,y)\in D} g(x,y).$$