SVM AND STATISTICAL LEARNING THEORY

W. RYAN LEE
CS109/AC209/STAT121 ADVANCED SECTION
INSTRUCTORS: P. PROTOPAPAS, K. RADER
FALL 2017, HARVARD UNIVERSITY

In the last chapter, we introduced GLMs and, in particular, logistic regression, which was our first model for the task of **classification**, which we will formalize below. Such models were based on probabilistic foundations, and had statistical interpretations for the predictions. We now proceed to describe two methods of classification that do not have such foundations, but are grounded in considerations of optimizing predictive power.

Classification and Statistical Learning Theory

First, we formalize the problem of classification. We assume that we are given a set of points, called the *training set*, denoted $\{(y_i, x_i)\}_{i=1}^n$. We consider the special (but common) case of binary classification, so that

$$y_i \in [-1, 1]$$

for all i, and $x_i \in \mathbb{R}^p$. As we noted previously, one possibility for modeling such data is to use logistic regression by assuming that

$$y_i | x_i \sim \text{Bern}\left(\frac{\exp(x_i^T \beta)}{1 + \exp(x_i^T \beta)}\right)$$

which yields a generalized linear model for y given x. This endows the observations with a probabilistic structure.

Given such a model, we can then make predictions on new data x^* by constructing a **discriminant function**. A discriminant $f : \mathbb{R}^p \to [-1, 1]$ is a function that takes the covariates x and outputs a predicted label ± 1. In the logistic regression case, one natural family of discriminants is to consider functions of the form

$$f(x) = \begin{cases} +1 & \text{if } P(y = +1 | x, \beta) \geq c \\ -1 & \text{otherwise} \end{cases}$$

which states that we predict that the class is $+1$ if the model-predicted probability is higher than some threshold. We showed that such a discriminant is equivalent to the following linear discriminant

$$f(x) = \begin{cases} +1 & \text{if } x^T \beta \geq \tilde{c} \\ -1 & \text{otherwise} \end{cases}$$

due to the linear relationship between the covariates and the probability.

From the perspective of discriminants, however, it is not necessary to require that a classification model be grounded in a probabilistic framework. We can consider arbitrary functions f that predict the outcome y, and optimize among

these functions based on some **loss criterion**. For binary classification, the most obvious choice of loss function is known as the **0-1 loss**, namely

$$l(f(x), y) = 1_{f(x) \neq y}$$

That is, we penalize according to the number of incorrect predictions made by our discriminant f.

When we consider all possible functions f, we can perfectly classify all points in our training set; we can simply set $f(x_i) = y_i$ for all i and let f be arbitrary elsewhere. However, what concerns us is not how well we "predict" on training data (which our classifier has already seen), but rather how well we predict on new data, namely the test set. If we need to use a highly-complex function f in order to perfectly classify our training set, it can often lead to overfitting, in which the loss on the training set (which is what we optimize) is considerably lower than that on the test set (which is what we want to optimize).

In fact, in a seminal paper founding *statistical learning theory*, Vapnik and Chervonenkis defined the celebrated **VC dimension**, which measures the capacity (or complexity) of the set of functions under consideration. Suppose we are considering a parametrized family of functions

$$\mathcal{F} \equiv \{f_\theta : \theta \in \Theta\}$$

Equivalently, we are considering a model $\mathcal{F}$ such as logistic regression, and are aiming to optimize a loss criterion to find a parameter θ that yields our discriminant or classifier f_θ. We then say that $\mathcal{F}$ **shatters** the training set if there exists some $\theta \in \Theta$ such that f_θ perfectly classifies all points in the training set. Then, we define the VC dimension of $\mathcal{F}$ as the maximum cardinality of the training set that can be shattered by $\mathcal{F}$,

$$VC(\mathcal{F}) \equiv \max\{n \in \mathbb{N} : \exists \text{ dataset } \mathcal{D}_n \text{ of size } n \text{ and } f \in \mathcal{F} \text{ s.t. } f \text{ shatters } \mathcal{D}_n\}$$

The importance of the VC dimension is that classification error on the test set can be upper bounded by the error on the training set and the VC dimension. Heuristically, the idea is that

$$\text{Test error } \leq \text{ Training error } + \text{ Model complexity}$$

where model complexity is an increasing function of VC dimension.

Support Vector Machines

These considerations lead us to consider "simpler" models that generalize well to unseen data while still preserving classification performance (though there is a natural trade-off between the two). That is, since we would like to minimize test error, our goal is to minimize training error (which we do directly or by a surrogate loss function) while also minimizing model complexity.

One possibility is to consider *linear* classifiers in x, which is equivalent to considering a hyperplane in the space of covariates to separate the points. In the **linearly separable** case, in which a hyperplane can perfectly separate (and thus classify) the training set, we can consider creating a "good" hyperplane in the sense that we maximize the distance from any of the points to the hyperplane. This approach is known as the **support vector machine (SVM)**.

That is, we consider hyperplanes of the form

$$w^T x = 0$$

where $w \in \mathbb{R}^p$ are the weights that define the hyperplane. Then, we use the discriminant

$$f_w(x) \equiv \text{sign}(w^T x)$$

which defines the family of functions $\mathcal{F} = \{f_w : w \in \mathbb{R}^p\}$ as our functions of interest.

Our goal is to maximize the minimum distance from the points to the hyperplane. One can show that the distance from the point x to the hyperplane defined above is given by

$$\frac{|w^T x|}{\|w\|}$$

Assuming that all points can be correctly classified, we must have

$$|w^T x| = y w^T x$$

Thus, our goal is to maximize

$$\max_w \frac{1}{\|w\|} \left[\min_i (y_i w^T x_i) \right]$$

Clearly, this is a very complicated optimization problem. One innovation was to turn this problem into an equivalent problem that is more easily solved. First, we have the freedom to constrain w since the margin is unchanged by scaling, so that we enforce

$$\min_i y_i (w^T x_i) = 1$$

so that every observation (y_i, x_i) satisfies

$$y_i (w^T x_i) \geq 1$$

Thus, we now only need to consider the maximization of $\|w\|^{-1}$, which is equivalent to minimizing $\|w\|^2$. Thus, we are led to the quadratic programming problem

$$\min_w \frac{1}{2} \|w\|^2$$

$$\text{s.t. } y_i (w^T x_i) \geq 1$$

This is achieved by using Lagrange multipliers and constructing the Lagrangian

$$L(w, a) \equiv \frac{1}{2} \|w\|^2 - \sum_{i=1}^n a_i \left[y_i (w^T x_i) - 1 \right]$$

We can then use the first-order conditions to eliminate w entirely to obtain the **dual representation of an SVM**:

$$\tilde{L}(a) \equiv \sum_{i=1}^n a_i - \frac{1}{2} \sum_{i,j=1}^n a_i a_j y_i y_j x_i^T x_j = \sum_{i=1}^n a_i - \frac{1}{2} \sum_{i,j=1}^n a_i a_j y_i y_j k(x_i, x_j)$$

where $k(x_i, x_j) = x_i^T x_j$ is a **kernel function**. This is subject to the constraints $a_i \geq 0$ and $\sum_{i=1}^n a_i y_i = 0$.

Given the dual parameters a, we can predict a new point x by considering the sign of the following (again eliminating w through the first-order conditions)

$$\sum_{i=1}^n a_i y_i x^T x_i = \sum_{i=1}^n a_i y_i k(x, x_i)$$

It can be shown that the dual representation satisfies the Karush-Kuhn-Tucker (KKT) conditions, which yields the following properties

$$a_i \geq 0$$
$$y_i(w^T x_i) - 1 \geq 0$$
$$a_i(y_i w^T x_i - 1) = 0$$

Thus, for every i, either $a_i = 0$ or $y_i(w^T x_i) = 1$. The points for which $a_i > 0$ are called **support vectors**. This is because these points are the only ones that impact the prediction, since when $a_i = 0$, (y_i, x_i) play no role in the dual classification rule above. In fact, the prediction rule for future x is essentially a weighted average of y_i among the support vectors, weighted by the "similarity" of the points x to the covariates x_i.

Moreover, one can see from the primal formulation that only the points for which $a_i > 0$ are on the margin; that is, these points satisfy the constraint

$$y_i w^T x_i = 1$$

Thus, this implies that the only points that influence predictions are the ones that are on the margin, and after training the SVM, we can throw away all other points not on the margin for predictive purposes.

C-SVM (Soft-Margin SVM)

In most real cases, the training set will not be linearly separable, even with a fairly sophisticated transformation of the feature space (i.e. using some $\phi(x)$ rather than x directly). However, in the SVM, we actually enforced perfect classification accuracy by adding $y_i(w^T x_i) \geq 1$ as a *constraint*, effectively putting infinite loss on points that lie on the wrong side of the hyperplane.

To get around this issue, we would like to allow for points to be on the wrong side, but to penalize the distance that the point takes inside its proper margin. That is, if a point is incorrectly classified, it should incur a higher loss if it is far on the wrong side. We thus introduce *slack variables* for each point as

$$\xi_i \equiv \begin{cases} 0 & \text{outside correct margin} \\ |y_i - w^T x_i| & \text{otherwise} \end{cases}$$

For example, if a point is inside the margin but on the correct side, $0 \leq \xi_i < 1$; if it is on the hyperplane, then $\xi_i = 1$; and if $\xi_i > 1$, then it is misclassified. Moreover, we have

$$y_i(w^T x_i) \geq 1 - \xi_i$$

Note that slack variables provide a linear measure of how far the point is from the correct side of the hyperplane, and that now it is possible for support vectors to lie inside the margins.

With these considerations, we seek to minimize

$$\min_w C \sum_{i=1}^{n} \xi_i + \frac{1}{2}\|w\|^2$$
$$\text{s.t. } \xi_i \geq 0$$
$$\text{and } y_i(w^T x_i) \geq 1 - \xi_i$$

according to the constraints, where C controls the trade-off between the slack variable penalty and the margin. As $C \to \infty$, we recover the hard-margin SVM, whereas

for $C \to 0$, we obtain a "flat" hyperplane that places no penalty on misclassification (i.e. the optimum is found when $\xi_i \to \infty$ and $w \to 0$).

Again, we consider the dual form by eliminating w using the Lagrangian and first-order conditions

$$L(w, a, bs) \equiv \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}a_i[y_i(w^T x_i) - 1 + \xi_i] - \sum_{i=1}^{n}b_i\xi_i$$

After eliminating w, ξ, μ from the Lagrangian, we obtain the **dual representation of the C-SVM** as

$$\tilde{L}(a) \equiv \sum_{i=1}^{n}a_i - \frac{1}{2}\sum_{i,j=1}^{n}a_i a_j y_i y_j (x_i^T x_j) = \sum_{i=1}^{n}a_i - \frac{1}{2}\sum_{i,j=1}^{n}a_i a_j y_i y_j k(x_i, x_j)$$

with the constraints $0 \leq a_i \leq C$, $\sum_{i=1}^{n} a_i y_i = 0$. Thus, this is identical to the separable case, except with an additional constraint on $a_i \leq C$. We also obtain similar KKT conditions, the important of which are

$$y_i(w^T x_i) - 1 + \xi_i \geq 0$$
$$a_i(y_i(w^T x_i) - 1 + \xi_i) \geq 0$$

which shows that for the support vectors with $a_i > 0$, we must have

$$y_i(w^T x_i) = 1 - \xi_i$$

Additional intuition and characterization of the support vectors based on the dual representation are possible, and can be found in further references.