Chapter 3, Part 2: Support Vector Machines

Advanced Topics in Statistical Machine Learning

Tom Rainforth

Hilary 2024

rainforth@stats.ox.ac.uk

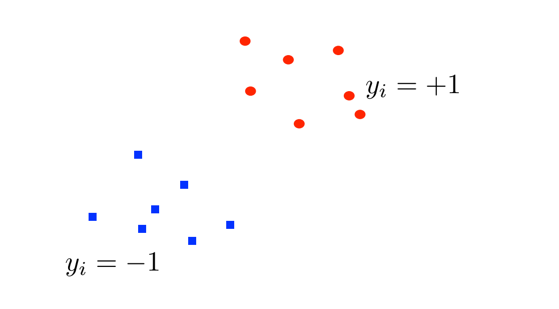
Support Vector Machines (SVMs)

- Support vector machines (SVMs) are a class of **linear** models for **classification**[1]
- The principle behind them is to a find a hyperplane that maximizes the **margin** of separation between points of different classes, while minimizing the number of points misclassified
- Their empirical risk minimization formulation satisfies **strong duality** making them easy to train
- Their real power is realized when we combine them with nonlinear features as will explain in subsequent lectures

[1] Their are a few less prominent variants that are used for regression, but we will not be covering these in the course.

Linearly Separable Points

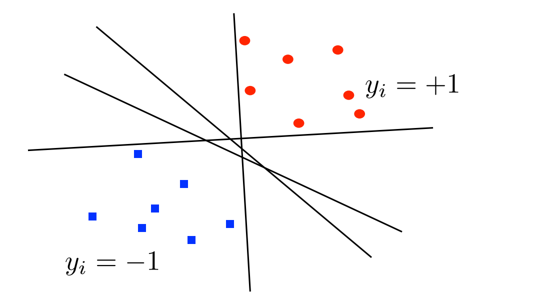
Consider classifying two clouds of points that can be perfectly separated by a linear hyperplane



Data: $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$, $x_i \in \mathbb{R}^p$, $y_i \in \{-1, +1\}$

Linearly Separable Points

Consider classifying two clouds of points that can be perfectly separated by a linear hyperplane

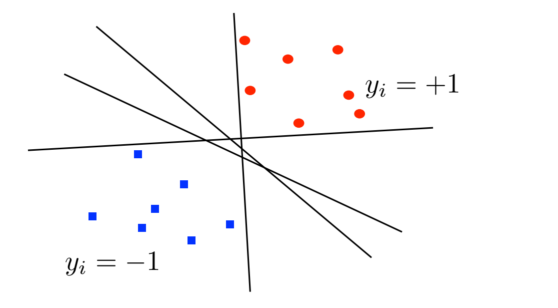


Data: $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$, $x_i \in \mathbb{R}^p$, $y_i \in \{-1, +1\}$

Predictive model: $\hat{y}(x) = \text{sign}(w^\top x + b)$

Linearly Separable Points

Consider classifying two clouds of points that can be perfectly separated by a linear hyperplane



Data: $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$, $x_i \in \mathbb{R}^p$, $y_i \in \{-1, +1\}$

Predictive model: $\hat{y}(x) = \mathrm{sign}(w^\top x + b)$

What is the best choice of hyperplane $w^\top x + b = 0$?

Classifier Margin

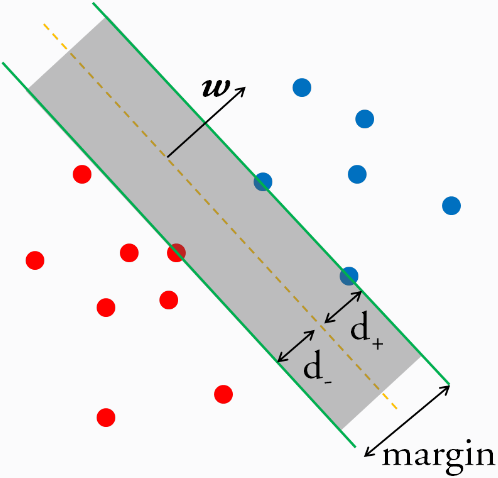
The **margin** of the hyperplane is defined as twice the distance to the closest point:

$$\text{margin} = 2 \min_i \left\| \frac{w^T}{\|w\|} x_i + \frac{b}{\|w\|} \right\|$$

SVMs are based on choosing the hyperplane that maximizes this margin, so that the points are as far as possible from the decision boundary (and thus least sensitive to changes)

The optimal hyperplane will always be halfway between the bounding points for each class, so the margin at the solution will be equal to the distance between the two closest points of each class in the direction of w

Classifier Margin



[1] Image credit: Cathy Yeh, `https://www.efavdb.com/svm-classification`. Here we have assumed that $d_+ = d_-$ so that the margin is equal to the separation.

Maximum Margin Classifier, Linearly Separable Case

This margin maximization problem can be conveniently expressed as the following **quadratic program** (derivation on whiteboard):

$$\min_{w,b} \frac{1}{2}\|w\|^2$$
$$\text{subject to} \quad y_i(w^\top x_i + b) \geq 1 \;\; \forall i$$

which can be straightforwardly be solved by a number of standard methods

Maximum Margin Classifier with Errors Allowed

Points will not generally be linearly separable, so any practical classifier needs to allow points on the wrong side of the decision boundary or within the margin

We can assign a loss to such **margin errors** and then trade-off this loss with our desire to maximize the margin

One naive choice based on a 0-1 loss would be to take a scaled sum of the number of margin errors with our previous objective:

$$\min_{w,b} \left(\frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \mathbb{I} \left[y_i \left(w^\top x_i + b \right) < 1 \right] \right),$$

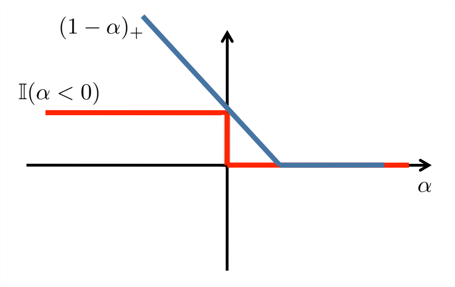
where C controls the tradeoff between maximum margin and loss.

This is impractical to optimize and fails to scale the penalization based **how far** a point is the wrong side of the margin boundary

Hinge Loss

SVMs instead use a hinge loss:

$$h(\alpha) = (1-\alpha)_+ = \begin{cases} 1-\alpha, & \alpha < 1 \\ 0, & \text{otherwise.} \end{cases}$$



This induces **sparse** solutions: w, b will be completely determined by a small number of datapoints known as the **support vectors**.

The C-SVM

Our optimization problem is given by

$$\min_{w,b}\left(\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} h\left(y_i\left(w^\top x_i + b\right)\right)\right).$$

By applying a rescaling, we can easily see that this can be viewed as a regularized empirical risk minimization problem:

$$\min_{w,b}\left(\frac{1}{2nC}\|w\|^2 + \frac{1}{n}\sum_{i=1}^{n} h\left(y_i\left(w^\top x_i + b\right)\right)\right).$$

Here the second term is an empirical risk from our margin errors, while the first term can be thought of as a regularizer that encourages large margins, with scaling $1/(2nC)$

The C-SVM

Though unconstrained, this problem is generally not convex and thus difficult to solve directly

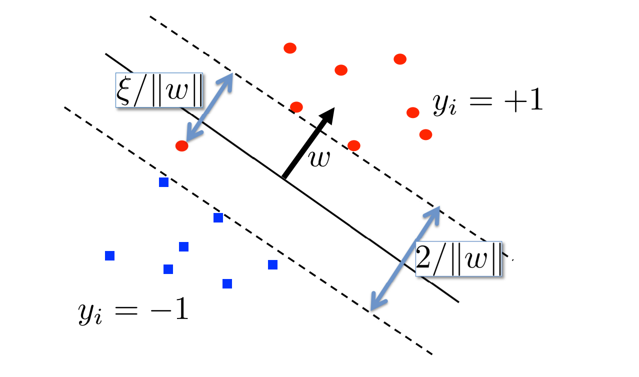
Using the substitutions $\xi_i = h\left(y_i\left(w^\top x_i + b\right)\right)$, we obtain the C-SVM, which is an equivalent formulation, but one that takes the form of a standard convex constrained optimization problem:

$$\min_{w,b,\xi} \left(\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i\right)$$

$$\text{subject to} \qquad \xi_i \geq 0 \qquad y_i\left(w^\top x_i + b\right) \geq 1 - \xi_i.$$

Proof (see notes) based on separately considering $\xi_i = 0$ and $\xi_i > 0$ and noting that in each case we must have $\xi_i = h\left(y_i\left(w^\top x_i + b\right)\right)$ at the optimum

The C-SVM



$$\min_{w,b,\xi} \left(\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i \right)$$

s.t. $\quad \xi_i \geq 0 \qquad y_i\left(w^\top x_i + b\right) \geq 1 - \xi_i$

Duality

Primal problem in standard form:

$$\begin{aligned} \text{minimize} \quad & f_0(w,b,\xi) := \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \\ \text{subject to} \quad & f_i(w,b,\xi) := 1 - \xi_i - y_i\left(w^\top x_i + b\right) \le 0,\ i = 1,\dots,n \\ & f_{n+i}(w,b,\xi) := -\xi_i \le 0,\ i = 1,\dots,n. \end{aligned}$$

As a convex optimization problem with affine constraints in w, b, ξ, **strong duality** holds (noting that it is trivial to see that a feasible solution will exist).

Note that there are no equality constraints, but it will be convenient to use separate notion for the Lagrange multiplier of the two sets of equality constraints

The Lagrangian

The Lagrangian (with Lagrange multiplier α and λ)

$$
\begin{aligned}
L(w, b, \xi, \alpha, \lambda) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \\
+ \sum_{i=1}^{n}\alpha_i\left(1 - \xi_i - y_i\left(w^\top x_i + b\right)\right) + \sum_{i=1}^{n}\lambda_i(-\xi_i)
\end{aligned}
$$

$$
= \frac{1}{2}\|w\|^2 - w^\top\sum_{i=1}^{n}\alpha_i y_i x_i - b\sum_{i=1}^{n}\alpha_i y_i + \sum_{i=1}^{n}\xi_i(C - \lambda_i - \alpha_i) + \sum_{i=1}^{n}\alpha_i
$$

with dual variable constraints

$$
\alpha_i \geq 0, \qquad \lambda_i \geq 0.
$$

Minimize wrt the primal variables w, b, and ξ.

Stationary Points of the Lagrangian

$$L = \frac{1}{2}\|w\|^2 - w^\top \sum_{i=1}^n \alpha_i y_i x_i - b \sum_{i=1}^n \alpha_i y_i + \sum_{i=1}^n \xi_i (C - \lambda_i - \alpha_i) + \sum_{i=1}^n \alpha_i$$

Derivative wrt w:

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^n \alpha_i y_i x_i = 0 \quad \Rightarrow \quad w = \sum_{i=1}^n \alpha_i y_i x_i$$

Derivative wrt b:

$$\frac{\partial L}{\partial b} = -\sum_{i=1}^n y_i \alpha_i = 0 \quad \Rightarrow \quad \sum_{i=1}^n y_i \alpha_i = 0$$

Derivative wrt ξ_i:

$$\frac{\partial L}{\partial \xi_i} = C - \alpha_i - \lambda_i = 0 \quad \Rightarrow \quad \alpha_i = C - \lambda_i$$

Since $\lambda_i \geq 0$,

$$\alpha_i \leq C$$

Dual Feasible Space

Here the derivatives with respect to b and ξ_i have led to expressions that are independent of the primal variables: $\sum_i y_i\alpha_i = 0$ and $\alpha_i = C - \lambda_i$

These essentially form equality constraints for the dual variables to be feasible: if they do not hold,

$$g(\alpha, \lambda) = \inf_{w,b,\xi} L(w, b, \xi, \alpha, \lambda) = -\infty$$

We can therefore assume they hold when calculating the dual function form, before adding them back in as constraints to the dual problem itself

The Dual Function

$$g(\alpha, \lambda) = \inf_{w,b,\xi} \left(\frac{1}{2}\|w\|^2 - w^\top \sum_{i=1}^n \alpha_i y_i x_i - b \sum_{i=1}^n \alpha_i y_i + \sum_{i=1}^n \xi_i (C - \lambda_i - \alpha_i) + \sum_{i=1}^n \alpha_i \right)$$

substituting in $w = \sum_{i=1}^n \alpha_i y_i x_i$ yields

$$= \inf_{b,\xi} \left(\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^\top x_j - \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^\top x_j - b \underbrace{\sum_{i=1}^n \alpha_i y_i}_{=0 \text{ at opt}} + \sum_{i=1}^n \xi_i \underbrace{(C - \lambda_i - \alpha_i)}_{=0 \text{ at opt}} + \sum_{i=1}^n \alpha_i \right)$$

$$= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^\top x_j.$$

SVM Training: Maximize the Dual Function

By strong duality we can now optimize the primal problem by solving the quadratic dual program (variety of efficient methods)

$$\max_{\alpha,\lambda} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^\top x_j,$$

$$\text{subject to} \qquad 0 \leq \alpha_i \leq C, \quad \sum_{i=1}^{n} y_i \alpha_i = 0, \quad \lambda_i = C - \alpha_i$$

When solving, we can ignore λ, solve for α and then set $\lambda = C - \alpha$

We can then derive the variables for our hyperplane by taking

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i, \qquad b = y_{i_{\text{margin}}} - w^\top x_{i_{\text{margin}}}$$

where i_{magin} is any index of a margin SV, i.e. any i such that $0 < \alpha_i < C$ (they will all give the same b).

The Support Vectors

Using **complementary slackness** and remembering α_i is the Lagrange multiplier for the constraint $1 - \xi_i - y_i \left(w^\top x_i + b\right) \leq 0$, we can show that all datapoints fall into one of the following:

Non-SVs (SV = support vector): $\alpha_i = 0$

1. From $\alpha_i = C - \lambda_i$, $\lambda_i > 0$, hence $\xi_i = 0$ (as $\lambda_i \xi_i = 0$).
2. Thus $y_i \left(w^\top x_i + b\right) > 1$: on the correct side of the margin

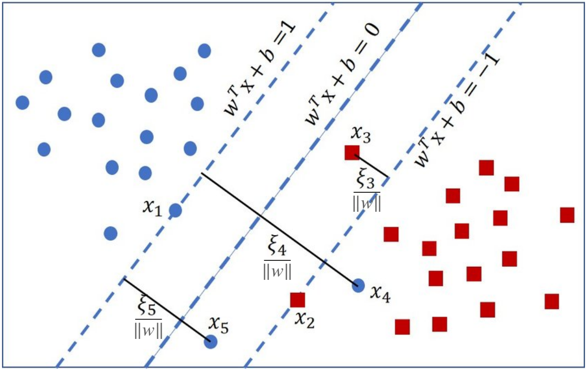
Margin SVs: $0 < \alpha_i < C$

1. We immediately have $y_i \left(w^\top x_i + b\right) = 1 - \xi_i$.
2. Again as $\alpha_i = C - \lambda_i$, we have $\lambda_i > 0$, hence $\xi_i = 0$, and $y_i \left(w^\top x_i + b\right) = 1$: on the margin boundary

Non-margin SVs (margin errors): $\alpha_i = C > 0$

1. We again have $y_i \left(w^\top x_i + b\right) = 1 - \xi_i$.
2. From $\alpha_i = C - \lambda_i$, we now have $\lambda_i = 0$, so $\xi_i \geq 0$, $y_i \left(w^\top x_i + b\right) < 1$: margin error

The Support Vectors



Margin SVs: x_1, x_2, non-margin SVs: x_3, x_4, x_5, non-SVs: $x_{>5}$

[1] Image adapted from Hung Minh Le, Toan Dinh Tran, and LANG Van Tran. "Automatic heart disease prediction using feature selection and data mining technique". In: **Journal of Computer Science and Cybernetics** (2018).

Insights from Form of Solution

Our solution for w is a linear sum of the datapoints

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

- The solution is sparse: points which are neither on the margin nor "margin errors" have $\alpha_i = 0$
- **The support vectors** are points where $\alpha_i \neq 0$: only points on the decision boundary, or which are margin errors, contribute.
- As $\alpha_i \geq 0$, points of class $y_i = +1$ have positive coefficients and points with $y_i = -1$ have negative coefficients
- Influence of any single datapoint is bounded, since the weights cannot exceed C.
- Even if $p > n$, $w \in \operatorname{span}\{x_i : i = 1, \ldots, n\}$ (i.e. w lives in the subspace spanned by the datapoints).

Multi-Class Classification

- SVMs do not directly generalize to multi-class classification because they are based on learning a single hyperplane
- They can still be applied to multi-class classification problems by reducing them into a series of binary classification problems
- For example, given K classes we can perform a **one-versus-the-rest** binary classification for each $k \in K$, yielding w_k and b_k, and then classify according to

$$\hat{y}(x) = \arg\max_k w_k^\top x + b_k$$

- Alternatively, we can also perform $K(K-1)$ **one–versus–one** classifications for each pair of classes and then use a max–wins voting strategy to choose the class

Recap

- SVMs are a class of **linear** classification models that find the hyperplane that maximizes the **margin** of separation between the classes while trying to avoid misclassifications
- They use a convex **hinge** loss that produces sparse solutions
- We find the optimal hyperplane by solving

  $$\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^\top x_j \quad \text{s.t.} \quad \begin{cases} \sum_{i=1}^{n} \alpha_i y_i = 0 \\ 0 \preceq \alpha \preceq C \end{cases}$$

  and taking $w = \sum_{i=1}^{n} \alpha_i y_i x_i,\ b = y_{i_{\text{margin}}} - w^\top x_{i_{\text{margin}}}$
- The resulting hyperplane is defined through the support vectors: the x_i for which $\alpha_i > 0$
- $0 < \alpha_i < C$ indicates a datapoint on the margin and $\alpha_i = C$ indicates a point the wrong side of the margin

Further Reading

- Youtube video with some nice visualizations and discussions on using features: `https://www.youtube.com/watch?v=efR1C6CvhmE`
- Sections 12.1 to 12.3 of Trevor Hastie, Robert Tibshirani, and Jerome Friedman. **The elements of statistical learning: data mining, inference, and prediction**. Springer Science & Business Media, 2009 (`https://web.stanford.edu/~hastie/ElemStatLearn/`)