18-661 Introduction to Machine Learning

SVM – II

Spring 2023

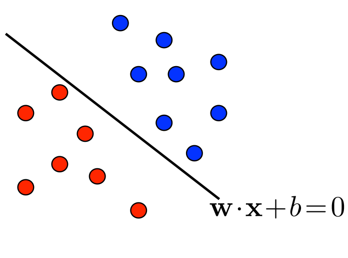
ECE – Carnegie Mellon University

Outline

1. Review of Max Margin SVM Formulation
2. SVM: Hinge Loss Formulation
3. SVM: Example
4. A Dual View of SVMs (the short version)

Review of Max Margin SVM Formulation

Intuition: Where to Put the Decision Boundary?



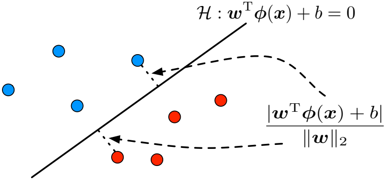
Find a decision boundary in the '*middle*' of the two classes that:

- Perfectly classifies the training data
- Is as far away from every training point as possible

Defining the Margin

Margin
Smallest distance between the hyperplane and all training points

$$\textsc{margin}(\boldsymbol{w}, b) = \min_n \frac{y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b]}{\|\boldsymbol{w}\|_2}$$



How can we use this to find the SVM solution?

Rescaled Margin

We further constrain the problem by scaling $(\boldsymbol{w}, b)$ such that

$$\min_n y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] = 1.$$

which leads to:

$$\textsc{margin}(\boldsymbol{w}, b) = \frac{\min_n y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b]}{\|\boldsymbol{w}\|_2} = \frac{1}{\|\boldsymbol{w}\|_2}$$

SVM: Max-margin Formulation for Separable Data

We thus want to solve:

$$\max_{\boldsymbol{w},b} \underbrace{\frac{1}{\|\boldsymbol{w}\|_2}}_{\text{margin}} \quad \text{such that} \quad \underbrace{\min_n y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] = 1}_{\text{scaling of } \boldsymbol{w},\, b}$$

This is equivalent to

$$\begin{aligned} \min_{\boldsymbol{w},b} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 \\ \text{s.t.} \quad & y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1, \quad \forall \quad n \end{aligned}$$

SVM for Non-separable Data

Constraints in separable setting

$$y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1, \quad \forall \quad n$$

This inherently requires all the training data are correctly separated into two sides of the boundary.

Constraints in non-separable setting
Can we modify our constraints to account for non-separability?

Specifically, we introduce slack variables $\xi_n \geq 0$:

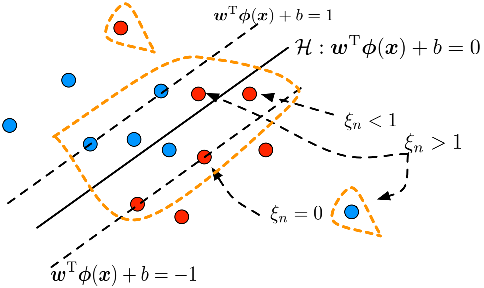
$$y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1 - \xi_n, \quad \forall \quad n$$

Soft-margin SVM Formulation

We do not want ξ_n to grow too large, and we can control their size by incorporating them into our optimization problem:

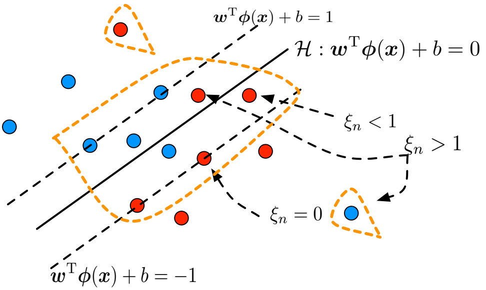
$$
\begin{aligned}
\min_{\boldsymbol{w}, b, \boldsymbol{\xi}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_n \xi_n \\
\text{s.t.} \quad & y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1 - \xi_n, \quad \forall \quad n \\
& \xi_n \geq 0, \quad \forall \ n
\end{aligned}
$$

Support Vectors: Revisit



Recall the constraints $y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1 - \xi_n$ from the soft-margin formulation. All the training points $(\boldsymbol{x}_n, y_n)$ that satisfies the constraint with "$=$" are support vectors.

Support Vectors: Revisit



In other words, support vectors satisfy $y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] = 1 - \xi_n$, which can be further divided into several categories:

- $\xi_n = 0$: $y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] = 1$, the point is on the correct side with distance $\frac{1}{\|\boldsymbol{w}\|}$.
- $0 < \xi_n \leq 1$: $y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \in [0, 1)$ on the correct side, but with distance less than $\frac{1}{\|\boldsymbol{w}\|}$.
- $\xi_n > 1$: $y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] < 0$, on the wrong side of the boundary.

SVM: Hinge Loss Formulation

SVM vs. Logistic Regression

SVM soft-margin formulation

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \quad \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_n \xi_n$$
$$\text{s.t.} \quad y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1 - \xi_n, \ \forall\ n$$
$$\xi_n \geq 0, \quad \forall\ n$$

Logistic regression formulation

$$\min_{\boldsymbol{w}} -\sum_n \{y_n \log \sigma(\boldsymbol{w}^\top \boldsymbol{x}_n) + (1 - y_n)\log[1 - \sigma(\boldsymbol{w}^\top \boldsymbol{x}_n)]\} + \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2$$

- Logistic regression defines a loss for each data point and minimizes the total loss plus a regularization term.
- This is convenient for assessing the "goodness" of the model on each data point.
- Can we write SVMs in this form as well? The Hinge Loss formulation!

Derive the Hinge Loss Formulation

Here's the soft-margin formulation again:

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_n \xi_n \;\text{ s.t. } y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1 - \xi_n,\; \xi_n \geq 0,\; \forall\, n$$

Now since $y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1 - \xi_n \iff \xi_n \geq 1 - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b]$:

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} C\sum_n \xi_n + \frac{1}{2}\|\boldsymbol{w}\|_2^2 \text{ s.t. } \xi_n \geq \max(0, 1 - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b]),\; \forall\, n$$

Now since the ξ_n should always be as small as possible, we obtain:

$$\min_{\boldsymbol{w},b} C\sum_n \max(0, 1 - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b]) + \frac{1}{2}\|\boldsymbol{w}\|_2^2$$

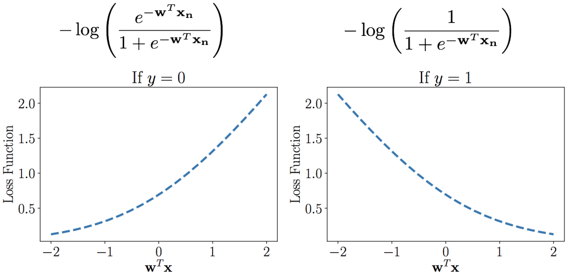
Divide by C and set $\lambda = \frac{1}{C}$, we get get Hinge Loss formulation:

$$\min_{\boldsymbol{w},b} \sum_n \underbrace{\max(0, 1 - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b])}_{\text{Hinge Loss for } x_n, y_n} + \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2$$

Logistic Regression Loss vs Hinge Loss

Given training data (x_n, y_n), the cross entropy loss was

$$-\{y_n \log \sigma(\boldsymbol{w}^\top \boldsymbol{x}_n) + (1 - y_n) \log[1 - \sigma(\boldsymbol{w}^\top \boldsymbol{x}_n)]\}$$

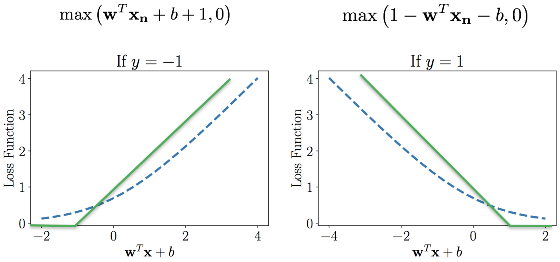


- How does the Hinge Loss Function look like?

Logistic Regression Loss vs Hinge Loss

Given training data (x_n, y_n), the Hinge loss is

$$\max(0, 1 - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b])$$



- Loss grows linearly as we move away from the boundary.
- No penalty if a point is more than 1 unit from the boundary.
- Makes the search for the boundary easier (as we will see later).

Hinge Loss SVM Formulation

Minimizing the total hinge loss on all the training data

$$\min_{\boldsymbol{w},b} \sum_n \underbrace{\max(0, 1 - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b])}_{\text{hinge loss for sample } n} + \underbrace{\frac{\lambda}{2}\|\boldsymbol{w}\|_2^2}_{\text{regularizer}}$$

Analogous to regularized least squares or logistic regression, as we balance between two terms (the loss and the regularizer).

- Can solve using gradient descent to get the optimal $\mathbf{w}$ and b
- Gradient of the first term will be either 0, $\mathbf{x}_n$ or $-\mathbf{x}_n$ depending on y_n and $\boldsymbol{w}^\top \boldsymbol{x}_n + b$.
- Much easier to compute than in logistic regression, where we need to compute the sigmoid function $\sigma(\boldsymbol{w}^\top \boldsymbol{x}_n + b)$ in each iteration.

Summary: Three SVM Formulations

Hard-margin (for separable data)

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \ \frac{1}{2}\|\boldsymbol{w}\|_2^2 \text{ s.t. } y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1, \ \xi_n \geq 0, \ \forall \ n$$

Soft-margin (add slack variables)

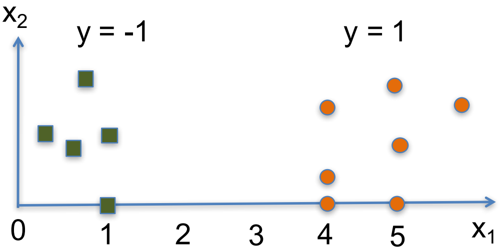
$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \ \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_n \xi_n \ \text{ s.t. } y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1 - \xi_n, \ \xi_n \geq 0, \ \forall \ n$$

Hinge loss (define a loss function for each data point)

$$\min_{\boldsymbol{w},b} \ \sum_n \max(0, 1 - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b]) + \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2$$

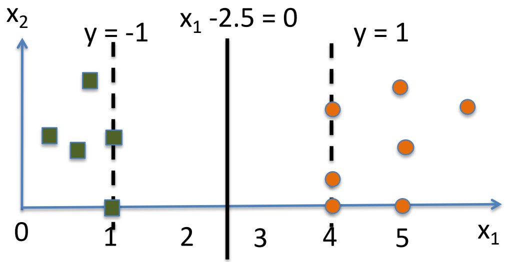
SVM: Example

Example of SVM



What will be the decision boundary learnt by solving the SVM optimization problem?

Example of SVM



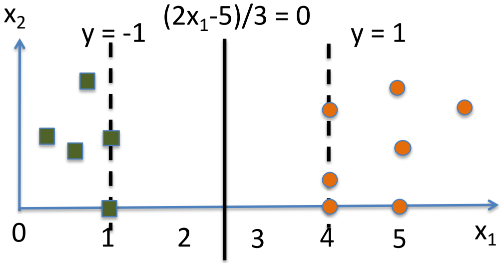
Margin $= 1.5$; the decision boundary has $\mathbf{w} = [1, 0]^\top$, and $b = -2.5$.

Is this the right scaling of $\mathbf{w}$ and b? We need $\min_n y_n(\mathbf{w}^\top \mathbf{x}_n + b) = 1$.

Not quite. For example, a support vector $\mathbf{x}_n = [1, 0]^\top$ (which achieves the above min), we have

$$y_n(\mathbf{w}^\top \mathbf{x}_n + b) = (-1)[1 - 2.5] = 1.5.$$

Example of SVM: Scaling



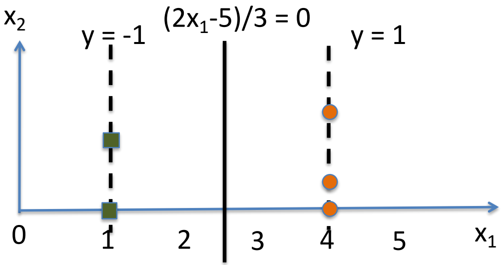
Thus, our optimization problem will re-scale $\mathbf{w}$ and b to get this equation for the same decision boundary

The correct parameter should be $\mathbf{w} = [2/3, 0]^\top$, and $b = -5/3$.

For example, for $\mathbf{x}_n = [1, 0]^\top$, we have

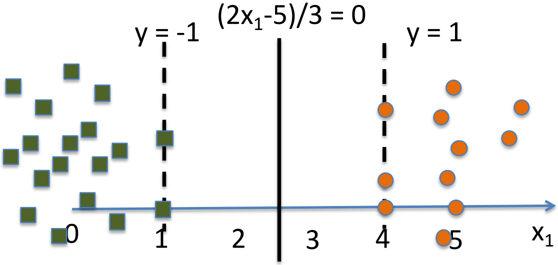
$$y_n(\mathbf{w}^\top \mathbf{x}_n + b) = (-1)[2/3 - 5/3] = 1.$$

Example of SVM: Support Vectors



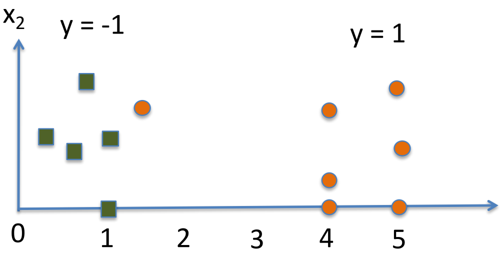
The solution to our optimization problem will be the **same** to the *reduced* dataset containing all the support vectors.

Example of SVM: Support Vectors



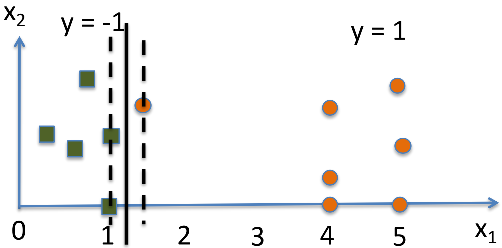
There can be many more data than the number of support vectors (so we can train on a smaller dataset).

Example of SVM: Resilience to Outliers



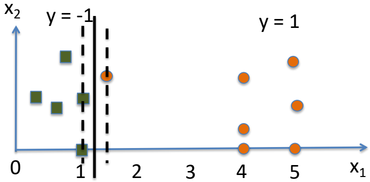
- Still linearly separable, but one of the orange dots is an "outlier".

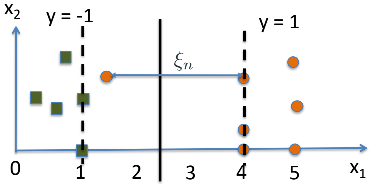
Example of SVM: Resilience to Outliers



- Naively applying the hard-margin SVM will result in a classifier with small margin.
- So, better to use the soft-margin (or equivalently, hinge loss) formulation.

Example of SVM: Resilience to Outliers





$$\min_{\boldsymbol{w}, b, \boldsymbol{\xi}} \quad \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C \sum_n \xi_n$$

$$\text{s.t.} \quad y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1 - \xi_n, \quad \forall \; n$$

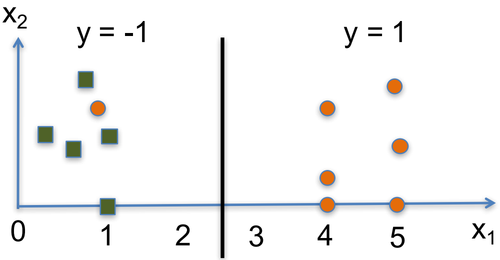
$$\xi_n \geq 0, \quad \forall \; n$$

We allow the outlier to violate the constraint by ξ_n which we penalize.

- Small $C \Rightarrow$ more constraint violation, less sensitivity to outliers; but also (potentially) worse accuracy as more points are misclassified.

- $C = +\infty$ corresponds to hard margin SVM.

Example of SVM



- Similar reasoning applies to the case when the data is not linearly separable.
- The value of C determines how much the boundary will shift: trade-off of accuracy and robustness (sensitivity to outliers).

Advantages of SVM

So far, shown SVM:

1. Maximizes distance of training data from the boundary.
2. Only requires a subset of the training points.
3. Is less sensitive to outliers.
4. Scales better with high-dimensional data.
5. Generalizes well to many nonlinear models.

We will need to use duality to show the fourth property.

Outline

1. Review of Max Margin SVM Formulation
2. SVM: Hinge Loss Formulation
3. SVM: Example
4. A Dual View of SVMs (the short version)

A Dual View of SVMs (the short version)

What Is Duality?

Consider optimization problem with single constraint

$$\min f(x) \text{ s.t. } g(x) \leq 0$$

Define Lagrangian $L(x, \lambda) = f(x) + \lambda g(x)$, where you can think of $\lambda g(x)$ as "penalty" for constraint violation.

The above (known as primal) is equivalent to $\min_x \max_{\lambda \geq 0} L(x, \lambda)$

- If $g(x) \leq 0$, $\max_{\lambda \geq 0} L(x, \lambda) = f(x)$
- If $g(x) > 0$, $\max_{\lambda \geq 0} L(x, \lambda) = +\infty$
- Effectively enforces constraint $g(x) \leq 0$.

Dual problem: swapping the order of min and max

$$\max_{\lambda \geq 0} \underbrace{\min_x L(x, \lambda)}_{\text{known as dual function}}$$

What Is Duality?

Consider the following problem with optimizer $x^* = -1$, optimal value $\frac{1}{2}$.

$$\min \frac{1}{2}x^2 \text{ s.t. } x + 1 \leq 0$$

Lagrangian $L(x, \lambda) = \frac{1}{2}x^2 + \lambda(x + 1)$

Dual problem:

$$\max_{\lambda \geq 0} \quad \underbrace{\min_x L(x, \lambda)}_{\text{known as dual function } D(\lambda)}$$

$D(\lambda) = \min_x L(x, \lambda)$ - how to compute?

- Set $\nabla_x L(x, \lambda) = x + \lambda = 0 \Rightarrow x^*(\lambda) = -\lambda$
- $D(\lambda) = L(x^*(\lambda), \lambda) = -\frac{1}{2}\lambda^2 + \lambda$

Can show $\max_{\lambda \geq 0} D(\lambda) = \frac{1}{2}$ (achieved at $\lambda^* = 1$), same as the optimal value of primal problem). Further, $x^*(\lambda^*) = -1$, recovers optimal primal solution.

What Is Duality?

Recap: for the following problem with optimizer

$$\min \frac{1}{2}x^2 \text{ s.t. } x + 1 \leq 0$$

- Primal solution $x^* = -1$ satisfies constraint $x + 1 \leq 0$ with $=$.
- Dual solution $\lambda^* = 1$ is non-zero.

Slightly change the problem:

$$\min \frac{1}{2}x^2 \text{ s.t. } x - 1 \leq 0$$

- Primal solution $x^* = 0$ satisfies constraint $x - 1 \leq 0$ with $<$.
- Can show dual solution λ^* is zero.

This is known as complimentary slackness: suppose the constraint is $g(x) \leq 0$, then $\lambda^* g(x^*) = 0$, i.e. $\lambda^* > 0$ only when the constraint is met with $=$.

What Is Duality?

Duality is a way of transforming a constrained optimization problem.

It tells us sometimes-useful information about the problem structure, and can sometimes make the problem easier to solve.

- Under strong duality condition (the details is beyond the scope...), primal and dual problems are equivalent.
- Further, due to complementary slackness, dual variables tell us whether constraints are met with $=$ or $<$
- The strong duality condition is not always true for all optimiztion problems, but is true for the soft-margin SVM problem.

Instead of solving the max margin (primal) formulation, we solve its dual problem which will have certain advantages we will see.

Derivation of the Dual

Here is a skeleton of how to derive the dual problem.

Recipe

1. Formulate the generalized Lagrangian function (we'll define this on the next slide) that incorporates the constraints and introduces dual variables
2. Minimize the Lagrangian function over the primal variables
3. Plug in the primal variables from the previous step into the Lagrangian to get the dual function
4. Maximize the dual function with respect to dual variables
5. Recover the solution (for the primal variables) from the dual variables

Deriving the Dual for SVM

Primal SVM

$$\begin{aligned} \min_{\boldsymbol{w},b,\boldsymbol{\xi}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_n \xi_n \\ \text{s.t.} \quad & y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1 - \xi_n, \quad \forall \ n \\ & \xi_n \geq 0, \quad \forall \ n \end{aligned}$$

The constraints are equivalent to the following canonical forms:

$$-\xi_n \leq 0 \quad \text{and} \quad 1 - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] - \xi_n \leq 0$$

Lagrangian

$$\begin{aligned} L(\boldsymbol{w}, b, \{\xi_n\}, \{\alpha_n\}, \{\lambda_n\}) = C\sum_n \xi_n + \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \sum_n \lambda_n \xi_n \\ + \sum_n \alpha_n \{1 - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] - \xi_n\} \end{aligned}$$

under the constraints that $\alpha_n \geq 0$ and $\lambda_n \geq 0$.

Deriving the Dual of SVM

Lagrangian

$$L(\boldsymbol{w}, b, \{\xi_n\}, \{\alpha_n\}, \{\lambda_n\}) = C\sum_n \xi_n + \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \sum_n \lambda_n \xi_n + \sum_n \alpha_n\{1 - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] - \xi_n\}$$

under the constraints that $\alpha_n \geq 0$ and $\lambda_n \geq 0$.

- Primal variables: $\boldsymbol{w}$, $\{\xi_n\}$, b; dual variables $\{\lambda_n\}$, $\{\alpha_n\}$
- Minimize the Lagrangian function over the primal variables by setting $\frac{\partial L}{\partial \boldsymbol{w}} = 0$, $\frac{\partial L}{\partial b} = 0$, and $\frac{\partial L}{\partial \xi_n} = 0$.
- Substitute primal variables from the above into the Lagrangian to get the dual function.
- Maximize the dual function with respect to dual variables
- After some further maths and simplifications, we have...

Dual Formulation of SVM

Dual is also a convex quadratic program

$$\max_{\boldsymbol{\alpha}} \quad \sum_n \alpha_n - \frac{1}{2} \sum_{m,n} y_m y_n \alpha_m \alpha_n \boldsymbol{x}_m^\top \boldsymbol{x}_n$$

$$\text{s.t.} \quad 0 \leq \alpha_n \leq C, \quad \forall\ n$$

$$\sum_n \alpha_n y_n = 0$$

- There are N dual variables α_n, one for each data point
- Independent of the size d of $\mathbf{x}$: SVM scales better for high-dimensional feature.
- May seem like a lot of optimization variables when N is large, but many of the α_n's become zero. α_n is non-zero only if the n^{th} point is a support vector

Why Do Many α_n's Become Zero?

$$\max_{\boldsymbol{\alpha}} \quad \sum_n \alpha_n - \frac{1}{2}\sum_{m,n} y_m y_n \alpha_m \alpha_n \boldsymbol{x}_m^\top \boldsymbol{x}_n$$

$$\text{s.t.} \quad 0 \leq \alpha_n \leq C, \quad \forall\ n$$

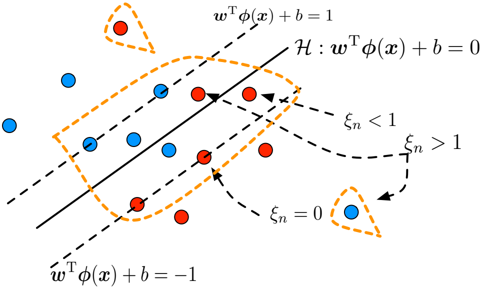
$$\sum_n \alpha_n y_n = 0$$

- By complementary slackness:

$$\alpha_n\{1 - \xi_n - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b]\} = 0 \quad \forall n$$

- This tells us that $\alpha_n > 0$ only when $1 - \xi_n = y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b]$, i.e. (x_n, y_n) is a support vector. So most of the α_n is zero, and the only non-zero α_n are for the support vectors.
- Further, $\alpha_n < C$ only when $\xi_n = 0$. (The derivation of this is beyond the scope of today's lecture)

Visualizing the Support Vectors



- $\alpha_n = 0$: non-support vector.
- $0 < \alpha_n < C$: support vector with $\xi_n = 0$, i.e. $y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] = 1$, distance to boundary $\frac{1}{\|w\|}$.
- $\alpha_n = C$: support vector with $\xi_n > 0$, hence $y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] < 1$.

How to Get w and b?

Lagrangian

$$L(\boldsymbol{w}, b, \{\xi_n\}, \{\alpha_n\}, \{\lambda_n\}) = C \sum_n \xi_n + \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \sum_n \lambda_n \xi_n + \sum_n \alpha_n \{1 - y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] - \xi_n\}$$

Recovering w

$$\frac{\partial L}{\partial \boldsymbol{w}} = 0 \rightarrow \boldsymbol{w} = \sum_n \alpha_n y_n \boldsymbol{x}_n$$

Only depends on support vectors, i.e., points with $\alpha_n > 0$!

Recovering b

Find a sample (x_n, y_n) such that $0 < \alpha_n < C$. Using $y_n \in \{-1, 1\}$,

$$y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] = 1$$

$$b = y_n - \boldsymbol{w}^\top \boldsymbol{x}_n$$

$$b = y_n - \sum_m \alpha_m y_m \boldsymbol{x}_m^\top \boldsymbol{x}_n$$

Summary of Dual Formulation

Primal Max-Margin Formulation

$$\min_{\boldsymbol{w}, b, \boldsymbol{\xi}} \quad \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_n \xi_n$$

$$\text{s.t.} \quad y_n[\boldsymbol{w}^\top \boldsymbol{x}_n + b] \geq 1 - \xi_n, \quad \forall \; n$$

$$\xi_n \geq 0, \quad \forall \; n$$

Dual Formulation

$$\max_{\boldsymbol{\alpha}} \quad \sum_n \alpha_n - \frac{1}{2}\sum_{m,n} y_m y_n \alpha_m \alpha_n \boldsymbol{x}_m^\top \boldsymbol{x}_n$$

$$\text{s.t.} \quad 0 \leq \alpha_n \leq C, \quad \forall \; n$$

$$\sum_n \alpha_n y_n = 0$$

- In dual formulation, the # of variables is independent of dimension.
- Most of the dual variables are 0, and the non-zero ones are the support vectors.
- Can easily recover the primal solution $\boldsymbol{w}, b$ from dual solution.

Advantages of SVM

We have shown SVM:

1. Maximizes distance of training data from the boundary
2. Only requires a subset of the training points.
3. Is less sensitive to outliers.
4. Scales better with high-dimensional data.
5. Generalizes well to many nonlinear models.

The last thing left to consider is non-linear decision boundaries, or kernel SVMs, which we will cover in the next lecture.