

ITCS 6156/8156 Fall 2024 Machine Learning

Support Vector Machine

Instructor: Hongfei Xue

Email: hongfei.xue@charlotte.edu

Class Meeting: Tue & Thu, 4:00 PM – 5:15 PM, WWH 130

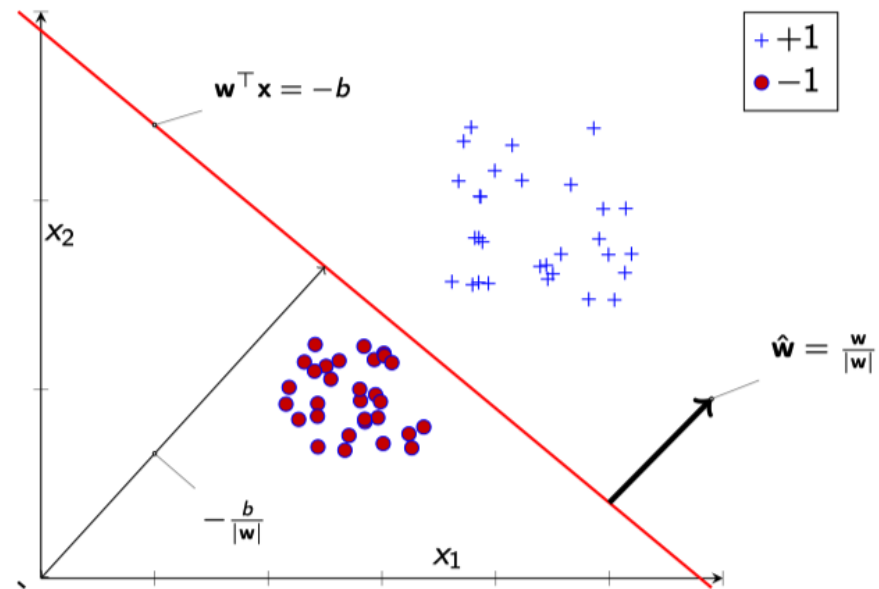


Some content in the slides is based on Dr. Varun's lecture

Maximum Margin Classifiers

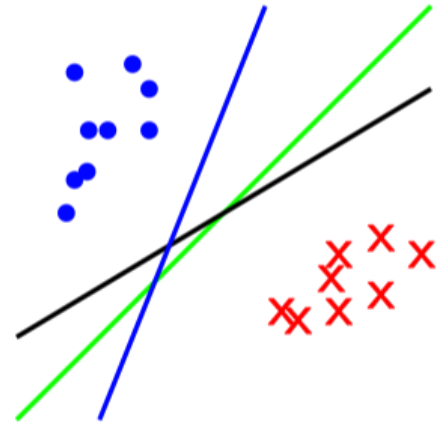
$$y = \mathbf{w}^\top \mathbf{x} + b$$

- ▶ Remember the Perceptron!
- ▶ If data is linearly separable
 - ▶ Perceptron training guarantees learning the decision boundary
- ▶ There can be other boundaries
 - ▶ Depends on initial value for \mathbf{w}
- ▶ **But what is the best boundary?**



Best Hyperplane Separator

- ▶ **Perceptron** can find a hyperplane that separates the data
 - ▶ ... if the data is linearly separable
- ▶ But there can be many choices!
- ▶ Find the one with best separability (largest margin)
- ▶ Gives better generalization performance



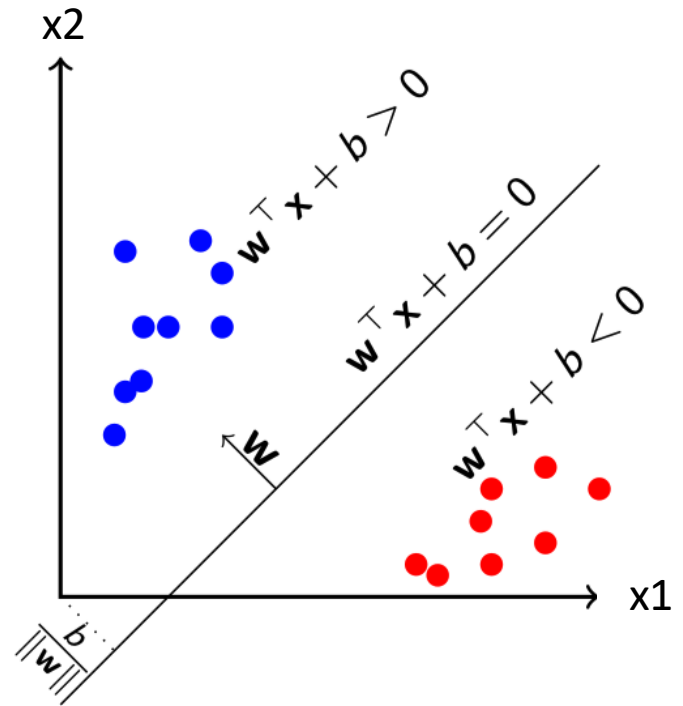
Line as a Decision Surface

- ▶ Decision boundary represented by the hyperplane \mathbf{w}
- ▶ For binary classification, \mathbf{w} points **towards** the positive class

Decision Rule

$$y = \text{sign}(\mathbf{w}^T \mathbf{x} + b)$$

- ▶ $\mathbf{w}^T \mathbf{x} + b > 0 \Rightarrow y = +1$
- ▶ $\mathbf{w}^T \mathbf{x} + b < 0 \Rightarrow y = -1$



Concept of Margin

- ▶ **Margin** is the distance between an example and the decision line
- ▶ Denoted by γ
- ▶ For a positive point:

$$\gamma = \frac{\mathbf{w}^\top \mathbf{x} + b}{\|\mathbf{w}\|} \quad (1)$$

- ▶ For a negative point:

$$\gamma = -\frac{\mathbf{w}^\top \mathbf{x} + b}{\|\mathbf{w}\|} \quad (2)$$

Functional Interpretation

- ▶ Margin **positive** if prediction is **correct**; **negative** if prediction is **incorrect**

A Geometric Perspective

To understand the margin from a geometric perspective, consider the projection of the vector connecting the origin to a point \mathbf{x} on the decision line. Let the point be denoted as \mathbf{x}' . Obviously the vector \mathbf{r} connecting \mathbf{x}' and \mathbf{x} is given by:

$$\mathbf{r} = \gamma \hat{\mathbf{w}} = \gamma \frac{\mathbf{w}}{\|\mathbf{w}\|}$$

if \mathbf{x} lies on the positive side of \mathbf{w} . But the same vector can be computed as:

$$\mathbf{r} = \mathbf{x} - \mathbf{x}'$$

Equating above two gives us \mathbf{x}' as:

$$\mathbf{x}' = \mathbf{x} - \gamma \frac{\mathbf{w}}{\|\mathbf{w}\|}$$

Noting that, since \mathbf{x}' lies on the hyperplane and hence:

$$\mathbf{w}^\top \mathbf{x}' + b = 0$$

Substituting \mathbf{x}' from above:

$$\mathbf{w}^\top \mathbf{x} - \gamma \frac{\mathbf{w}^\top \mathbf{w}}{\|\mathbf{w}\|} + b = 0$$

Noting that $\frac{\mathbf{w}^\top \mathbf{w}}{\|\mathbf{w}\|} = \|\mathbf{w}\|$, we get γ as:

$$\gamma = \frac{\mathbf{w}^\top \mathbf{x} + b}{\|\mathbf{w}\|} \quad (1)$$

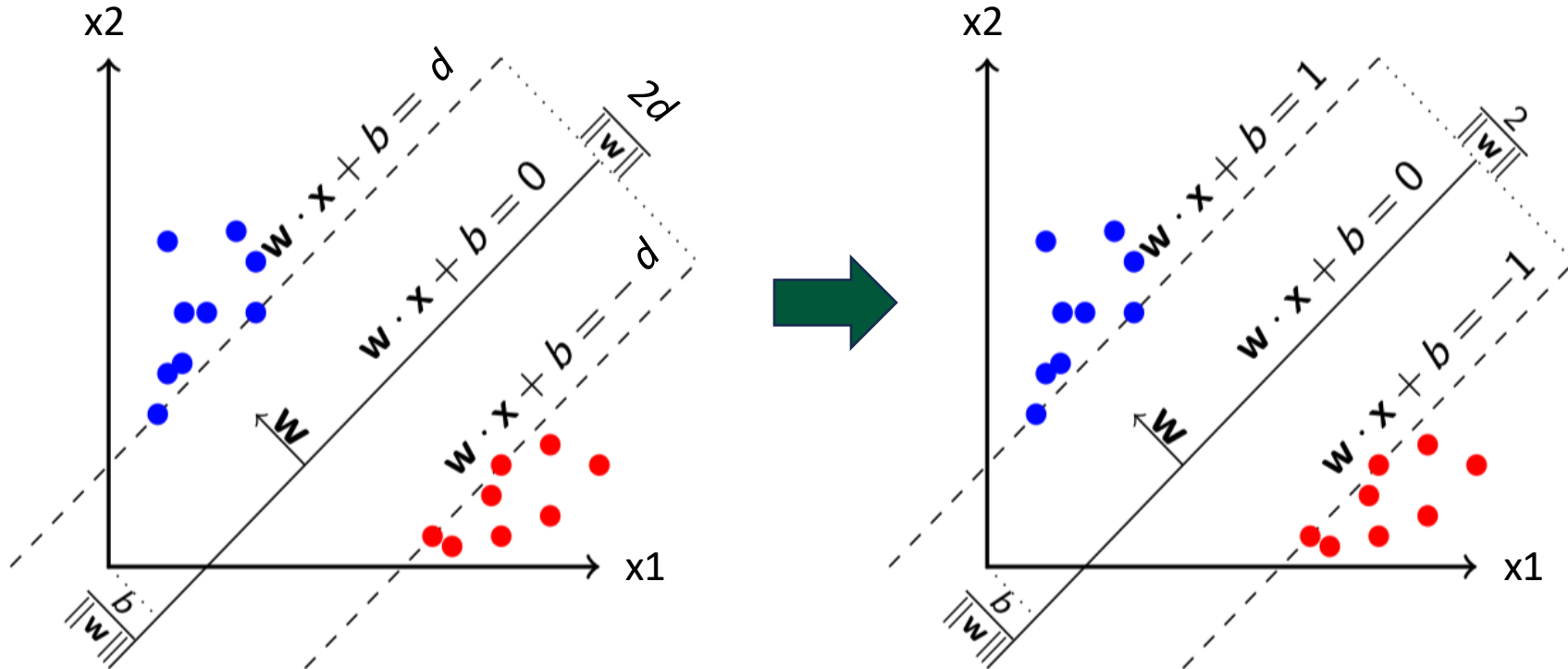
Similar analysis can be done for points on the negative side of \mathbf{x} . In general, one can write the expression for the margin as:

$$\gamma = y \frac{\mathbf{w}^\top \mathbf{x} + b}{\|\mathbf{w}\|} \quad (2)$$

where $y \in \{-1, +1\}$.

Maximum Margin Principle

- Normalization:



From the figure one can note that the size of the margin is $\frac{2}{\|w\|}$. We can show this as follows. Since the data is separable, we can get two parallel lines represented by $w^\top x + b = +1$ and $w^\top x + b = -1$. Using result from (1) and (2), the distance between the two lines is given by $2\gamma = \frac{2}{\|w\|}$.

Support Vector Machines

- ▶ A hyperplane based classifier defined by \mathbf{w} and b
- ▶ Like perceptron
- ▶ Find hyperplane with *maximum separation margin* on the training data
- ▶ Assume that data is linearly separable (will relax this later)
 - ▶ Zero training error (loss)

SVM Prediction Rule

$$y = \text{sign}(\mathbf{w}^\top \mathbf{x} + b)$$

SVM Learning

- ▶ **Input:** Training data $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N)\}$
- ▶ **Objective:** Learn \mathbf{w} and b that maximizes the margin

SVM Learning

- ▶ SVM learning task as an optimization problem
- ▶ Find \mathbf{w} and b that gives zero training error
- ▶ Maximizes the margin $(= \frac{2}{\|\mathbf{w}\|})$
- ▶ Same as minimizing $\|\mathbf{w}\|$

Optimization Formulation

$$\begin{array}{ll}\text{minimize}_{\mathbf{w}, b} & \frac{\|\mathbf{w}\|^2}{2} \\ \text{subject to} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \quad i = 1, \dots, n.\end{array}$$

- ▶ **Optimization** with N linear inequality constraint

A Different Interpretation of Margin

- ▶ What impact does the margin have on \mathbf{w} ?
- ▶ Large margin \Rightarrow Small $\|\mathbf{w}\|$
- ▶ Small $\|\mathbf{w}\| \Rightarrow$ regularized/simple solutions
- ▶ Simple solutions \Rightarrow Better generalizability (*Occam's Razor*)

Optimization Problem

Optimization Formulation

$$\begin{array}{ll}\underset{\mathbf{w}, b}{\text{minimize}} & \frac{\|\mathbf{w}\|^2}{2} \\ \text{subject to} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \quad i = 1, \dots, n.\end{array}$$

- ▶ There is an quadratic objective function to minimize with N inequality constraints
- ▶ “Off-the-shelf” packages - quadprog (MATLAB), CVXOPT
- ▶ Is that the best way?

An Optimization Problem

- An optimization problem without constraint:

$$\underset{x,y}{\text{minimize}} \quad f(x, y) = x^2 + 2y^2 - 2$$

- An optimization problem with constraint:

$$\begin{aligned} &\underset{x,y}{\text{minimize}} \quad f(x, y) = x^2 + 2y^2 - 2 \\ &\text{subject to} \quad h(x, y) = x + y - 1 = 0. \end{aligned}$$

An Optimization Problem

- ▶ Tool for solving constrained optimization problems of differentiable functions

$$\begin{array}{ll} \underset{x,y}{\text{minimize}} & f(x,y) = x^2 + 2y^2 - 2 \\ \text{subject to} & h(x,y) : x + y - 1 = 0. \end{array}$$

- ▶ A Lagrangian multiplier (β) lets you combine the two equations into one

$$\underset{x,y,\beta}{\text{minimize}} \quad L(x,y,\beta) = f(x,y) + \beta h(x,y)$$

An Optimization Problem

Solution 1. *Writing the objective as Lagrangian.*

$$L(x, y, \beta) = x^2 + 2y^2 - 2 + \beta(x + y - 1)$$

Setting the gradient to 0 with respect to x, y and β will give us the optimal values.

$$\frac{\partial L}{\partial x} = 2x + \beta = 0$$

$$\frac{\partial L}{\partial y} = 4y + \beta = 0$$

$$\frac{\partial L}{\partial \beta} = x + y - 1 = 0$$

Multiple Constraints

$$\begin{array}{ll}\text{minimize}_{x,y,z} & f(x, y, z) = x^2 + 4y^2 + 2z^2 + 6y + z \\ \text{subject to} & h_1(x, y, z) : \quad \quad \quad x + z^2 - 1 = 0 \\ & h_2(x, y, z) : \quad \quad \quad x^2 + y^2 - 1 = 0.\end{array}$$

$$L(x, y, z, \boldsymbol{\beta}) = f(x, y, z) + \sum_i \beta_i h_i(x, y, z)$$

Handling Inequality Constraints

$$\begin{array}{ll} \underset{x,y}{\text{minimize}} & f(x,y) = x^3 + y^2 \\ \text{subject to} & g(x) : x^2 - 1 \leq 0. \end{array}$$

- Inequality constraints are **transferred** as constraints on the Lagrangian, α

The Lagrangian in the above example becomes:

$$\begin{aligned} L(x,y,\alpha) &= f(x,y) + \alpha g(x,y) \\ &= x^3 + y^2 + \alpha(x^2 - 1) \end{aligned}$$

Handling Inequality Constraints

Solving for the gradient of the Lagrangian gives us:

$$\frac{\partial}{\partial x} L(x, y, \alpha) = 3x^2 + 2\alpha x = 0$$

$$\frac{\partial}{\partial y} L(x, y, \alpha) = 2y = 0$$

$$\frac{\partial}{\partial \alpha_1} L(x, y, \alpha) = x^2 - 1 = 0$$

Furthermore we require that:

$$\alpha \geq 0$$

From above equations we get $y = 0$, $x = \pm 1$ and $\alpha = \pm \frac{3}{2}$. But since $\alpha \geq 0$, hence $\alpha = \frac{3}{2}$. This gives $x = 1$, $y = 0$, and $f = 1$.

A Toy Example

Optimization Formulation

$$\begin{array}{ll}\underset{\mathbf{w}, b}{\text{minimize}} & \frac{\|\mathbf{w}\|^2}{2} \\ \text{subject to} & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \quad i = 1, \dots, n.\end{array}$$

A Toy Example

- $\mathbf{x} \in \Re^2$

- Two training points:

$$\mathbf{x}_1, y_1 = (1, 1), -1$$

$$\mathbf{x}_2, y_2 = (2, 2), +1$$

- Find the best hyperplane $\mathbf{w} = (w_1, w_2)$

A Toy Example

Optimization problem for the toy example

$$\begin{array}{ll} \underset{\mathbf{w}}{\text{minimize}} & f(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2 \\ \text{subject to} & g_1(\mathbf{w}, b) = y_1(\mathbf{w}^\top \mathbf{x}_1 + b) - 1 \geq 0 \\ & g_2(\mathbf{w}, b) = y_2(\mathbf{w}^\top \mathbf{x}_2 + b) - 1 \geq 0. \end{array}$$

- Substituting actual values for \mathbf{x}_1, y_1 and \mathbf{x}_2, y_2 .

$$\begin{array}{ll} \underset{\mathbf{w}}{\text{minimize}} & f(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2 \\ \text{subject to} & g_1(\mathbf{w}, b) = -(\mathbf{w}^\top \mathbf{x}_1 + b) - 1 \geq 0 \\ & g_2(\mathbf{w}, b) = (\mathbf{w}^\top \mathbf{x}_2 + b) - 1 \geq 0. \end{array}$$

The above problem can be also written as:

$$\begin{array}{ll} \underset{w_1, w_2, b}{\text{minimize}} & f(w_1, w_2) = \frac{1}{2}(w_1^2 + w_2^2) \\ \text{subject to} & g_1(w_1, w_2, b) = -(w_1 + w_2 + b) - 1 \geq 0 \\ & g_2(w_1, w_2, b) = (2w_1 + 2w_2 + b) - 1 \geq 0. \end{array}$$

A Toy Example

To solve the toy optimization problem, we rewrite it in the Lagrangian form:

$$L(w_1, w_2, b, \alpha) = \frac{1}{2}(w_1^2 + w_2^2) + \alpha_1(w_1 + w_2 + b + 1) - \alpha_2(2w_1 + 2w_2 + b - 1)$$

Setting $\nabla L = 0$, we get:

$$\frac{\partial}{\partial w_1} L(w_1, w_2, b, \alpha) = w_1 + \alpha_1 - 2\alpha_2 = 0$$

$$\frac{\partial}{\partial w_2} L(w_1, w_2, b, \alpha) = w_2 + \alpha_1 - 2\alpha_2 = 0$$

$$\frac{\partial}{\partial b} L(w_1, w_2, b, \alpha) = \alpha_1 - \alpha_2 = 0$$

$$\frac{\partial}{\partial \alpha_1} L(w_1, w_2, b, \alpha) = w_1 + w_2 + b + 1 = 0$$

$$\frac{\partial}{\partial \alpha_2} L(w_1, w_2, b, \alpha) = 2w_1 + 2w_2 + b - 1 = 0$$

Solving the above equations, we get, $w_1 = w_2 = 1$ and $b = -3$.

Generalized Lagrangian

Handling Both Types of Constraints

$$\begin{array}{ll} \underset{\mathbf{w}}{\text{minimize}} & f(\mathbf{w}) \\ \text{subject to} & g_i(\mathbf{w}) \leq 0 \quad i = 1, \dots, k \\ \text{and} & h_i(\mathbf{w}) = 0 \quad i = 1, \dots, l. \end{array}$$

Generalized Lagrangian

$$L(\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = f(\mathbf{w}) + \sum_{i=1}^k \alpha_i g_i(\mathbf{w}) + \sum_{i=1}^l \beta_i h_i(\mathbf{w})$$

subject to, $\alpha_i \geq 0, \forall i$

Primal and Dual Formulations

Primal Optimization

- Let θ_P be defined as:

$$\theta_P(\mathbf{w}) = \max_{\alpha, \beta: \alpha_i \geq 0} L(\mathbf{w}, \alpha, \beta)$$

- One can prove that the optimal value for the original constrained problem is same as:

$$p^* = \min_{\mathbf{w}} \theta_P(\mathbf{w}) = \min_{\mathbf{w}} \max_{\alpha, \beta: \alpha_i \geq 0} L(\mathbf{w}, \alpha, \beta)$$

Consider

$$\begin{aligned} \theta_P(\mathbf{w}) &= \max_{\alpha, \beta: \alpha_i \geq 0} L(\mathbf{w}, \alpha, \beta) \\ &= \max_{\alpha, \beta: \alpha_i \geq 0} f(\mathbf{w}) + \sum_{i=1}^k \alpha_i g_i(\mathbf{w}) + \sum_{i=1}^l \beta_i h_i(\mathbf{w}) \end{aligned}$$

It is easy to show that if any constraints are not satisfied, i.e., if either $g_i(\mathbf{w}) > 0$ or $h_i(\mathbf{w}) \neq 0$, then $\theta_P(\mathbf{w}) = \infty$. Which means that:

$$\theta_P(\mathbf{w}) = \begin{cases} f(\mathbf{w}) & \text{if primal constraints are satisfied} \\ \infty & \text{otherwise,} \end{cases}$$

Primal and Dual Formulations

Dual Optimization

- Consider θ_D , defined as:

$$\theta_D(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \min_{\mathbf{w}} L(\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

- The **dual** optimization problem can be posed as:

$$d^* = \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}: \alpha_i \geq 0} \theta_D(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}: \alpha_i \geq 0} \min_{\mathbf{w}} L(\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

$$d^* == p^*?$$

- Note that $d^* \leq p^*$
- “Max min” of a function is always less than or equal to “Min max”
- When will they be equal?
 - $f(\mathbf{w})$ is convex
 - Constraints are affine

Relation between Primal and Dual

- In general $d^* \leq p^*$, for SVM optimization the equality holds
- Certain conditions should be true
- Known as the **Kahrun-Kuhn-Tucker** conditions
- For $d^* = p^* = L(\mathbf{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$:

$$\frac{\partial}{\partial \mathbf{w}} L(\mathbf{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = 0$$

$$\frac{\partial}{\partial \beta_i} L(\mathbf{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = 0, \quad i = 1, \dots, l$$

$$\alpha_i^* g_i(\mathbf{w}^*) = 0, \quad i = 1, \dots, k$$

$$g_i(\mathbf{w}^*) \leq 0, \quad i = 1, \dots, k$$

$$\alpha_i^* \geq 0, \quad i = 1, \dots, k$$

Review

Optimization Formulation

$$\begin{aligned} & \underset{\mathbf{w}, b}{\text{minimize}} && \frac{\|\mathbf{w}\|^2}{2} \\ & \text{subject to} && y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \quad i = 1, \dots, n. \end{aligned}$$

- Introducing **Lagrange Multipliers**, α_i , $i = 1, \dots, n$

Rewriting as a (primal) Lagrangian

$$\begin{aligned} & \underset{\mathbf{w}, b, \alpha}{\text{minimize}} && L_P(\mathbf{w}, b, \alpha) = \frac{\|\mathbf{w}\|^2}{2} + \sum_{i=1}^n \alpha_i \{1 - y_i(\mathbf{w}^\top \mathbf{x}_i + b)\} \\ & \text{subject to} && \alpha_i \geq 0 \quad i = 1, \dots, n. \end{aligned}$$

Solving the Lagrangian

- Set gradient of L_P to 0

$$\frac{\partial L_P}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i$$

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

- Substituting in L_P to get the dual L_D

Review

Dual Lagrangian Formulation

$$\begin{aligned} & \underset{b, \alpha}{\text{maximize}} && L_D(\mathbf{w}, b, \alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{m,n=1}^N \alpha_m \alpha_n y_m y_n (\mathbf{x}_m^\top \mathbf{x}_n) \\ & \text{subject to} && \sum_{i=1}^n \alpha_i y_i = 0, \alpha_i \geq 0 \quad i = 1, \dots, n. \end{aligned}$$

- Dual Lagrangian is a *quadratic programming problem* in α_i 's
 - Use “off-the-shelf” solvers

Investigating Karush Kuhn Tucker Conditions

- For the primal and dual formulations
- We can optimize the dual formulation (as shown earlier)
- Solution should satisfy the **Karush-Kuhn-Tucker** (KKT) Conditions

KKT Conditions

$$\frac{\partial}{\partial \mathbf{w}} L_P(\mathbf{w}, b, \alpha) = \mathbf{w} - \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i = 0 \quad (1)$$

$$\frac{\partial}{\partial b} L_P(\mathbf{w}, b, \alpha) = - \sum_{i=1}^n \alpha_i y_i = 0 \quad (2)$$

$$1 - y_i \{\mathbf{w}^\top \mathbf{x}_i + b\} \leq 0 \quad (3)$$

$$\alpha_i \geq 0 \quad (4)$$

$$\alpha_i (1 - y_i \{\mathbf{w}^\top \mathbf{x}_i + b\}) = 0 \quad (5)$$

$$b = - \frac{\max_{n: y_i = -1} \mathbf{w}^\top \mathbf{x}_i + \min_{n: y_i = 1} \mathbf{w}^\top \mathbf{x}_i}{2}$$

Support Vectors

Most α_i 's are 0

- ▶ KKT condition #5:

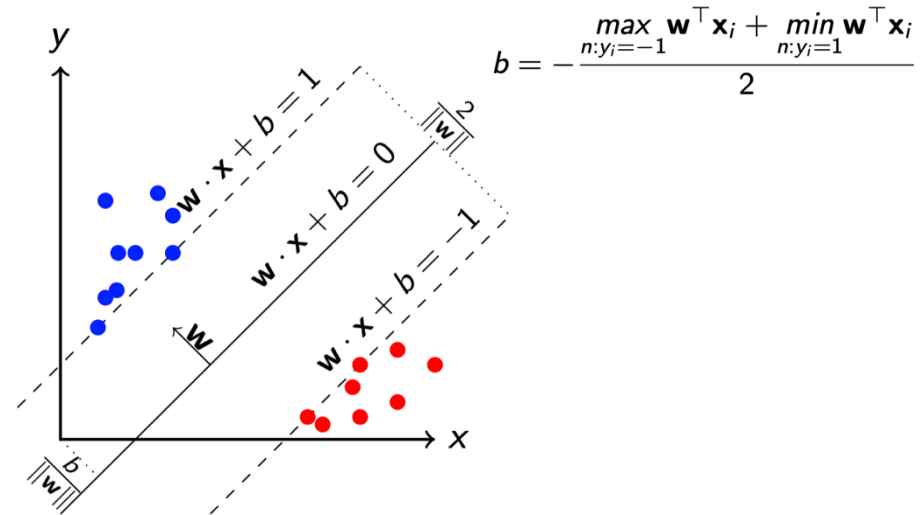
$$\alpha_i(1 - y_i\{\mathbf{w}^\top \mathbf{x}_i + b\}) = 0$$

- ▶ If \mathbf{x}_i **not** on margin

$$y_i\{\mathbf{w}^\top \mathbf{x}_i + b\} > 1$$

$$\Rightarrow \alpha_i = 0$$

- ▶ $\alpha_i \neq 0$ only for \mathbf{x}_i on margin
- ▶ These are the **support vectors**
- ▶ Only need these for prediction



- Also recall the prediction using SVMs

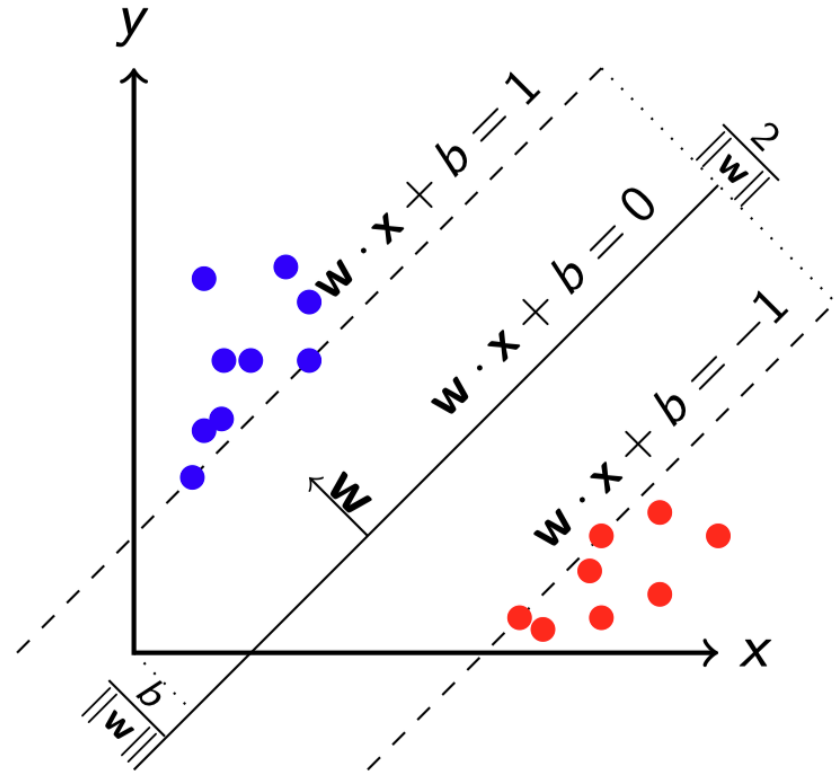
$$\begin{aligned} y^* &= \text{sign}(\mathbf{w}^\top \mathbf{x}^* + b) \\ &= \text{sign}\left(\left(\sum_{n=1}^N \alpha_n y_n \mathbf{x}_n\right)^\top \mathbf{x}^* + b\right) \\ &= \text{sign}\left(\sum_{n=1}^N \alpha_n y_n (\mathbf{x}_n^\top \mathbf{x}^*) + b\right) \end{aligned}$$

- The dot products are to be replaced with kernel functions

- RBF (Radial Basis Function) kernel: $k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{1}{2\gamma^2} \|\mathbf{x}_i - \mathbf{x}_j\|^2\right)$

SVM for Linearly Separable Data

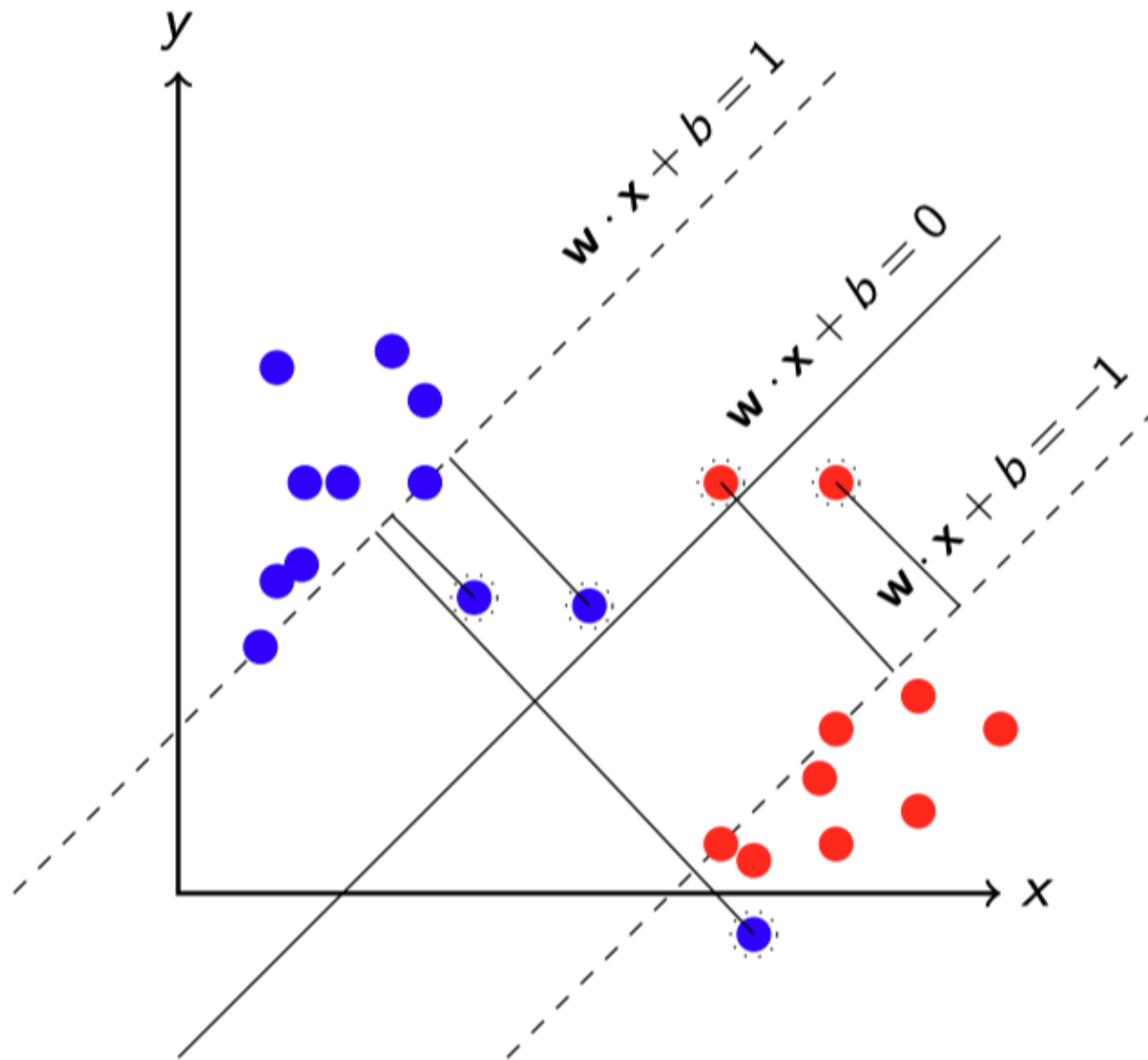
- ▶ For linearly separable data, SVM learns a weight vector \mathbf{w}
- ▶ Maximizes the margin
- ▶ SVM training is a **constrained optimization problem**
 - ▶ Each training example should lie outside the margin
 - ▶ N constraints



What if data is not linearly separable?

- ▶ Cannot go for zero training error
- ▶ Still learn a maximum margin hyperplane
 1. Allow some examples to be misclassified
 2. Allow some examples to fall **inside** the margin
- ▶ How do you set up the optimization for SVM training

Cutting Some Slack



Slack Variables

- ▶ **Separable Case:** To ensure zero training loss, constraint was

$$y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 \quad \forall i = 1 \dots n$$

- ▶ **Non-separable Case:** Relax the constraint

$$y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \quad \forall i = 1 \dots n$$

- ▶ ξ_i is called **slack variable** ($\xi_i \geq 0$)
- ▶ For misclassification, $\xi_i > 1$

Relaxing the Constraint

- ▶ It is OK to have some misclassified training examples
 - ▶ Some ξ_i 's will be non-zero
- ▶ Minimize the number of such examples

- ▶ Minimize $\sum_{i=1}^n \xi_i$

- ▶ Optimization Problem for Non-Separable Case

$$\begin{aligned} & \underset{\mathbf{w}, b}{\text{minimize}} && f(\mathbf{w}, b) = \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i \\ & \text{subject to} && y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0 \quad i = 1, \dots, n. \end{aligned}$$

- ▶ C controls the impact of margin and the margin error.

Parameter C

- C determining the trade-off between maximizing the margin and minimizing the classification error on the training data.
 - High C:
 - more emphasis on correctly classifying the training examples, even at the risk of overfitting.
 - less tolerant of noise and outliers.
 - result in a smaller margin.
 - Low C:
 - prioritize maximizing the margin, which might lead to some misclassifications on the training set (or even potentially underfit the data).
 - more tolerant of noise and outliers.
 - result in a wider margin.

Estimating Weights

- ▶ Similar optimization procedure as for the separable case (QP for the dual)
- ▶ Weights have the same expression

$$\mathbf{w} = \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i$$

- ▶ Support vectors are slightly different
 1. Points on the margin ($\xi_i = 0$)
 2. Inside the margin but on the correct side ($0 < \xi_i < 1$)
 3. On the wrong side of the hyperplane ($\xi_i \geq 1$)

Questions?