Support Vector Machine
Hard Margin SVM: Dual Form

M. R. Hasan
Readings

• Alpaydin: 10.3, 13.1, 13.2
• Murphy: 14.5.2.2
• Geron: chapter 5, appendix C
What We Will Cover

• Constrained Optimization Problem
• Unconstrained Optimization Problem
• Lagrange Multiplier
• KKT Conditions
• Primal & Dual Unconstrained Optimization
Hard Margin Linear SVM Classification

- The dual problem would **benefit us in two ways**.

**Benefit 1**: Its complexity depends on $N$ (**not on $d$**)

$$
\begin{align*}
\text{Primal} & \quad \min_{\overline{w}, b} \frac{1}{2} \|\overline{w}\|^2 \equiv \frac{1}{2} \overline{w}^T \overline{w} \\
\text{subject to} & \quad y_i (\overline{w}^T \overline{x}_i + b) \geq 1 \quad \forall i
\end{align*}
$$

**Dual**

$$
\begin{align*}
& \max_{\alpha_1, \ldots, \alpha_N} \overline{w}^T \overline{1} - \frac{1}{2} \overline{w}^T \mathbf{H} \overline{w} \\
& \text{subject to} \quad \sum_{i=1}^{N} \alpha_i y_i = 0 \\
& \quad \alpha_i \geq 0
\end{align*}
$$

$$
\overline{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \overline{x}_i
$$

**Benefit 2**: It will allow us to solve a high-dimensional optimization problem **without even projecting the data** in high-dimension (for **nonlinear** case)

This will be possible by a “magical” technique known as **Kernel trick**!
Hard Margin Linear SVM Classification

**Primal**
\[
\min_{\vec{w}, b} \frac{1}{2} \|\vec{w}\|^2 \equiv \frac{1}{2} \vec{w}^T \vec{w}
\]

subject to
\[
y_i (\vec{w}^T \vec{x}_i + b) \geq 1 \forall i
\]

Thus, although we can directly solve the **primal problem** (by using CVXOPT), we are interested to convert it into its dual form.

**Dual form** (complexity depends on N)
\[
\max_{\alpha_1, \ldots, \alpha_N} \vec{\alpha}^T \vec{1} - \frac{1}{2} \vec{\alpha}^T H \vec{\alpha}
\]

subject to
\[
\sum_{i=1}^{N} \alpha_i y_i = 0
\]
\[
\alpha_i \geq 0
\]

where
\[
H_{ij} = y_i y_j \vec{x}_i^T \vec{x}_j
\]

**CVXOPT**

Python Software for Convex Optimization

CVXOPT is a free software package for convex optimization based on the Python programming language. It can be used with the interactive Python interpreter, on the command line by executing Python scripts, or integrated in other software via Python extension modules. Its main purpose is to make the development of software for convex optimization applications straightforward by building on Python’s extensive standard library and on the strengths of Python as a high-level programming language.
Hard Margin Linear SVM Classification

Primal

$$\min_{\vec{w},b} \frac{1}{2} \|\vec{w}\|^2 \equiv \frac{1}{2} \vec{w}^T \vec{w}$$

subject to \( y_i (\vec{w}^T \vec{x}_i + b) \geq 1 \ \forall i \)

How do we **transform the primal problem** into its dual form?

We can **derive the dual form** by analytically solving the primal problem.

**Dual form** (complexity depends on N)

$$\max_{\alpha_1, \ldots, \alpha_N} \alpha^T \vec{1} - \frac{1}{2} \alpha^T H \alpha$$

subject to \( \sum_{i=1}^{N} \alpha_i y_i = 0 \)

$$\alpha_i \geq 0$$

where \( H_{ij} = y_i y_j \vec{x}_i^T \vec{x}_j \)
Hard Margin Linear SVM Classification

• How do we analytically solve the primal problem?
• It involves the minimization of the objective function $\frac{1}{2} w^T w$.
• The minimization could have been trivial if we didn’t have the constraint.
• Is it possible to get rid of the constraint and solve an unconstrained optimization problem?

$$\min_{\vec{w},b} \frac{1}{2} \|\vec{w}\|^2 \equiv \frac{1}{2} w^T \vec{w}$$

subject to $y_i(w^T \hat{x}_i + b) \geq 1 \ \forall i$
Hard Margin Linear SVM Classification

- **Good news!** There is a technique that we can use to transform this problem into an **unconstrained optimization problem**.

\[
\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \equiv \frac{1}{2} \mathbf{w}^T \mathbf{w}
\]

subject to \( y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \ \forall i \)

\[
\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^{N} \alpha_i [1 - y_i (\mathbf{w}^T \mathbf{x}_i + b)]
\]
Hard Margin Linear SVM Classification

• The **method of Lagrange multipliers**!
• It is a strategy for finding the local maxima and minima of a function **subject to equality constraints**.
• We can generalize this method by using **Karush-Kuhn-Tucker (KKT) conditions** to allow **inequality constraints**.
Hard Margin Linear SVM Classification

- Based on the nature of the constraints, there are two techniques to solve constrained optimization problems.
  - Equality constraints: Method of Lagrange multiplier
  - Equality and Inequality constraints: Generalized Lagrange multiplier with the KKT conditions

See the slides from “Constrained Optimization Problem” for an introduction to these two techniques for solving constrained optimization problems.

We will introduce the method of Lagrange multiplier briefly.
The Method of Lagrange Multipliers: Equality Constraints
The Method of Lagrange Multiplier

• Consider the following constrained optimization problem.

Minimize \( f(x_1, x_2) = 2x_1^2 + x_2^2 \)
subject to \( g(x_1, x_2) = 1 \)
where \( g(x_1, x_2) = x_1 + x_2 \)
The Method of Lagrange Multiplier

- Rewrite the constraint as: \( x_1 + x_2 - 1 = 0 \)
- At the minimum point, the gradients of both the objective function \( f(x_1, x_2) \) and the constraint \( x_1 + x_2 - 1 = 0 \) will be parallel (contour lines are tangent).
- Thus, at the minimum, following condition must hold.

\[
\nabla f(x_1, x_2) = \alpha \nabla (x_1 + x_2 - 1)
\]

It means the gradients of \( f(x_1, x_2) \) and \( (x_1 + x_2 - 1) \) both point in the same direction, and differ at most by a scalar \( \alpha \) (because of their varying magnitude).
The Method of Lagrange Multiplier

- This **tangency condition must hold** when we have reached the minimum of $f(x_1, x_2)$ subject to the constraint.

Minimize $f(x_1, x_2) = 2x_1^2 + x_2^2$
subject to $g(x_1, x_2) = 1$
where $g(x_1, x_2) = x_1 + x_2$

$$\nabla f(x_1, x_2) = \alpha \nabla (x_1 + x_2 - 1)$$

$$\nabla f(x_1, x_2) - \alpha \nabla (x_1 + x_2 - 1) = 0$$
The Method of Lagrange Multiplier

• We can construct a new objective function to capture this idea.

\[ \nabla f(x_1, x_2) - \alpha \nabla (x_1 + x_2 - 1) = 0 \]

Here \( \mathcal{L} \) is the Lagrangian (unconstrained optimization).

\( \alpha \) is the Lagrange multiplier

\[ \mathcal{L}(x_1, x_2, \alpha) = f(x_1, x_2) - \alpha(x_1 + x_2 - 1) \]

Rewrite \( \mathcal{L} \)

\[ \mathcal{L}(x_1, x_2, \alpha) = f(x_1, x_2) + \alpha(1 - x_1 + x_2) \]

Minimize \( f(x_1, x_2) = 2x_1^2 + x_2^2 \)
subject to \( g(x_1, x_2) = 1 \)
where \( g(x_1, x_2) = x_1 + x_2 \)
The Method of Lagrange Multiplier

• So, we have **transformed a constrained optimization objective** into an **unconstrained one**, by moving the constraints into the objective function.

\[
\mathcal{L}(x_1, x_2, \alpha) = f(x_1, x_2) + \alpha(1 - x_1 + x_2)
\]

Minimize \( f(x_1, x_2) = 2x_1^2 + x_2^2 \)
subject to \( g(x_1, x_2) = 1 \)
where \( g(x_1, x_2) = x_1 + x_2 \)
The Method of Lagrange Multiplier

- Joseph-Louis Lagrange showed that if \((x_1^*, x_2^*)\) is a solution to the constrained optimization problem,

  then there **must exist an** \(\alpha\) such that \((x_1^*, x_2^*, \alpha^*)\) is a **stationary point** of the Lagrangian.

- A stationary point is a point where **all partial derivatives are equal to zero**.

\[
\nabla \mathcal{L} = \begin{bmatrix}
\frac{\partial f}{\partial x_1} - \alpha \frac{\partial g}{\partial x_1} \\
\frac{\partial f}{\partial x_2} - \alpha \frac{\partial g}{\partial x_2} \\
\frac{\partial f}{\partial \alpha} - \alpha \frac{\partial g}{\partial \alpha}
\end{bmatrix} = 0
\]

\[
\mathcal{L}(x_1, x_2, \alpha) = f(x_1, x_2) + \alpha(1 - x_1 + x_2)
\]
The Method of Lagrange Multiplier

- In other words, we can compute the **partial derivatives** of $\mathcal{L}(x_1, x_2, \alpha)$ with regards to $x_1$, $x_2$, and $\alpha$.
- We can **find the points** where these derivatives are all equal to zero.
- The **solutions** to the constrained optimization problem (if they exist) must be **among these stationary points**.

$$\nabla \mathcal{L} = \begin{bmatrix} \frac{\partial f}{\partial x_1} - \alpha \frac{\partial g}{\partial x_1} \\ \frac{\partial f}{\partial x_2} - \alpha \frac{\partial g}{\partial x_2} \\ \frac{\partial f}{\partial \alpha} - \alpha \frac{\partial g}{\partial \alpha} \end{bmatrix} = 0$$

$$\mathcal{L}(x_1, x_2, \alpha) = f(x_1, x_2) + \alpha(1 - x_1 + x_2)$$
The Method of Lagrange Multiplier

• Observe that at the minimum point, the constraint vanishes (satisfied).

\[
\frac{\partial \mathcal{L}(x_1, x_2, \alpha)}{\partial \alpha} = 0
\]

\[
1 - x_1 + x_2 = 0
\]

\[
x_1 + x_2 = 1
\]

Minimize \( f(x_1, x_2) = 2x_1^2 + x_2^2 \)
subject to \( g(x_1, x_2) = 1 \)
where \( g(x_1, x_2) = x_1 + x_2 \)

\[
\nabla \mathcal{L}(x_1, x_2, \alpha) = 0
\]

\[
\mathcal{L}(x_1, x_2, \alpha) = f(x_1, x_2) + \alpha(1 - x_1 + x_2)
\]

The constraint is satisfied at the minimum of \( \mathcal{L}(x_1, x_2, \alpha) \).
The Method of Lagrange Multiplier

- What is the **interpretation** of the Lagrange multiplier term \( \alpha \)?
- The multiplier \( \alpha \) is used to use the **right amount of resource** \((x_1 \text{ and } x_2)\) that satisfies the constraint.

\[
\mathcal{L}(x_1, x_2, \alpha) = f(x_1, x_2) + \alpha(1 - x_1 + x_2)
\]

Minimize \( f(x_1, x_2) = 2x_1^2 + x_2^2 \)
subject to \( g(x_1, x_2) = 1 \)
where \( g(x_1, x_2) = x_1 + x_2 \)
The Lagrange Multipliers: 
Suggested Reading

Lagrange multiplier – main idea:

Interpretation of Lagrange multipliers:
Hard Margin Linear SVM Classification

• The **hard margin linear SVM** classification primal problem is a constrained optimization problem with **inequality constraints**.

• We will use the **generalized method of Lagrange multiplier** with the **Karush-Kuhn-Tucker (KKT)** conditions to solve this problem.

**Equality constraint**: Method of Lagrange Multiplier

Minimize $f(x_1, x_2) = 2x_1^2 + x_2^2$

subject to $x_1 + x_2 = 1$

**Inequality constraint**: Generalized Method of Lagrange Multiplier (KKT conditions)

$$\min_{\vec{w}, b} \frac{1}{2} \|\vec{w}\|^2 = \frac{1}{2} \vec{w}^T \vec{w}$$

subject to $y_i(\vec{w}^T \hat{x}_i + b) \geq 1 \forall i$
Hard Margin Linear SVM Classification

• To get an intuitive understanding of the problem let’s consider a simpler version of it.

\[
\min_x x^2 \\
subject to x \geq 1
\]

Say that we don’t have the constraint, then the minimum is at \( x = 0 \).

But this does not satisfy the inequality constraint.

We will solve it using Lagrange multiplier with the KKT conditions.
Hard Margin Linear SVM Classification

• With the constraint \( x \geq 1 \), we have to minimize \( x^2 \) by satisfying the constraint.

\[
\min_x x^2 \quad \text{subject to } x \geq 1
\]

Thus, the optimal solution is found at 1: \( x^* = 1 \)
Hard Margin Linear SVM Classification

- Notice that at the optimal $x^* (= 1)$ the inequality constraint is tight (becomes an equality).

\[
\min_x x^2 \\
\text{subject to } x \geq 1
\]

Thus, we find the optimal solution by tightening the constraint.
Hard Margin Linear SVM Classification

• Similarly for the SVM primal problem, the optimal $\vec{w}^*$ & $b^*$ exists for the data points $x_i$ that are on the margin (constraint is tightened to become equality).

These $x_i$ are the support vectors that constructs optimal hyperplane by finding $\vec{w}^*$ and $b^*$.

\[
\min_{\vec{w}, b} \frac{1}{2} ||\vec{w}||^2 \equiv \frac{1}{2} \vec{w}^T \vec{w}
\]

subject to $y_i(\vec{w}^T \vec{x}_i + b) \geq 1 \forall i$

Let’s solve it using Lagrange multiplier with the KKT conditions.

\[
\min x^2 \quad \text{subject to} \quad x \geq 1
\]
Hard Margin Linear SVM Classification

• We can **recast** this constrained optimization problem as an **unconstrained optimization** problem by including the constraint in a new objective function.

• It’s called the Lagrangian.

\[
\mathcal{L}(x, \alpha) = x^2 - \alpha(x - 1)
\]

\(\alpha\) is the **Lagrange multiplier**

Rewrite \(\mathcal{L}\)

\[
\mathcal{L}(x, \alpha) = x^2 + \alpha(1 - x)
\]
Hard Margin Linear SVM Classification

• We find an optimal solution for the Lagrangian by solving a **min-max problem**.

\[ \min_{\alpha, b} \frac{1}{2} \|w\|^2 \equiv \frac{1}{2} w^T w \]

subject to \( y_i (w^T \tilde{x}_i + b) \geq 1 \ \forall i \)

\[ \min_{x} x^2 \]

subject to \( x \geq 1 \)

\[ \mathcal{L}(x, \alpha) = x^2 + \alpha (1 - x) \]

Minimize the objective function.

Maximize the constraint term (to **tighten** the constraint).
Hard Margin Linear SVM Classification

• The constraint term \((1 - x)\) is **negative** for \(x > 1\).
• At the optimal \(x^*(= 1)\) the inequality **constraint is tight** (becomes equality).
• Thus, we maximize the constraint term.

\[
\mathcal{L}(x, \alpha) = x^2 + \alpha(1 - x)
\]

Minimize the objective function.

Maximize the constraint term (to **tighten** the constraint).

\[
\min_{x, \alpha} \frac{1}{2} ||\mathbf{w}||^2 \equiv \frac{1}{2} \mathbf{w}^T \mathbf{w}
\]

subject to \(y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \forall i\)

\[
\min x^2
\]

subject to \(x \geq 1\)
Hard Margin Linear SVM Classification

• To **tighten the constraint** we have to **maximize** the 2\textsuperscript{nd} term.

There are **two regions** for the constraint: $x > 1 \ & \ x = 1$

$x > 1$: The **constraint term is negative**, hence the optimizer will maximize it by setting $\alpha$ to zero.

$x = 1$: The constraint is **tight** (equality), hence to maximize the 2\textsuperscript{nd} term $\alpha$ will be **non-zero**.

\[
\min_{\vec{w},b} \frac{1}{2} \|\vec{w}\|^2 \equiv \frac{1}{2} \vec{w}^T \vec{w} \\
\text{subject to } y_i(\vec{w}^T \vec{x}_i + b) \geq 1 \ \forall i
\]

\[
\mathcal{L}(x, \alpha) = x^2 + \alpha(1 - x)
\]

\[
\min_x x^2 \\
\text{subject to } x \geq 1
\]
Hard Margin Linear SVM Classification

There are two regions for the constraint: $x > 1$ & $x = 1$

$x > 1$: The constraint term is negative, hence the optimizer will maximize it by setting $\alpha$ to zero.

$x = 1$: The constraint is tight (equality), hence to maximize the 2nd term $\alpha$ will be non-zero.

$\mathcal{L}(x, \alpha) = x^2 + \alpha(1 - x)$

$$\min_{x} x^2$$

subject to $x \geq 1$

These two conditions can be succinctly formulated as:

$$\alpha(1 - x) = 0$$

It’s called the Complimentary Slackness condition.
Lagrange Multiplier

• If the objective function is **convex** then the **optimal solution** $x^*$ and $\alpha^*$ for this Lagrangian function satisfy the following conditions.

• These conditions are known as the **KKT conditions**.

\[
\mathcal{L}(x, \alpha) = x^2 + \alpha(1 - x)
\]

\[
\begin{align*}
\frac{\partial \mathcal{L}(x, \alpha)}{\partial x} &= 0 \\
\alpha &\geq 0 \\
\alpha(1 - x) &= 0
\end{align*}
\]

**Complimentary Slackness**

By using the **KKT conditions** we can find the optimal solution $x^*$ and $\alpha^*$.
Lagrange Multiplier

• Let’s understand the **KKT conditions**.

\[ \mathcal{L}(x, \alpha) = x^2 + \alpha(1 - x) \]

**First condition**: it is justified because if a minimum exists then the derivative of \( \mathcal{L} \) should be zero at the minimum.

\[ \frac{\partial \mathcal{L}(x, \alpha)}{\partial x} = 0 \]

\[ \alpha \geq 0 \]

\[ \alpha(1 - x) = 0 \]

**Second condition**: the constraint penalty should be non-negative.

Complimentary Slackness
Lagrange Multiplier

Let’s explain the 3rd KKT condition.

\[ \mathcal{L}(x, \alpha) = x^2 + \alpha(1 - x) \]

\[ \frac{\partial \mathcal{L}(x, \alpha)}{\partial x} = 0 \]

\[ \alpha \geq 0 \]

\[ \alpha(1 - x) = 0 \]

Complimentary Slackness

To maximize the inequality constraint [i.e., \((1 - x)\) is negative] \(\alpha\) will be zero.

The tight constraint will be at the equality and \((1 - x) = 0\)

We use this condition to find the feasible solution (that satisfies the constraint).
Lagrange Multiplier

\[ \mathcal{L}(x, \alpha) = x^2 + \alpha(1 - x) \]

\[ \min_x x^2 \quad \text{subject to } x \geq 1 \]

Let's solve the problem.

\[ \frac{\partial \mathcal{L}(x, \alpha)}{\partial x} = 0 \]

\[ 2x - \alpha = 0 \]

Complimentary Slackness

\[ \alpha(1 - x) = 0 \]

\[ \alpha = 0 \]

\[ (1 - x) = 0 \quad \Rightarrow x = 1 \]

\[ 2x - 0 = 0 \quad \Rightarrow x = 0 \]

\[ 2.1 - \alpha = 0 \quad \Rightarrow \alpha = 2 \]

Feasible solution, hence \( x = 1 \) and \( \alpha = 2 \)

Not feasible
Lagrange Multiplier

\[ \mathcal{L}(x, \alpha) = x^2 + \alpha(1 - x) \]

- Thus, we see that the KKT conditions are **instrumental** for finding the feasible solution.

\[
\begin{align*}
\frac{\partial \mathcal{L}(x, \alpha)}{\partial x} & = 0 \\
\alpha & \geq 0 \\
\alpha(1 - x) & = 0
\end{align*}
\]

The KKT conditions were originally named after **Harold W. Kuhn, and Albert W. Tucker**, who first published the conditions in **1951**.

Later scholars discovered that the necessary conditions for this problem had been stated by **William Karush** in his **Master’s thesis** in **1939**.
Hard Margin Linear SVM Classification

• We will solve the **SVM primal problem** using this technique.
• We write the **unconstrained problem** using the **Lagrange multiplier**:

\[
\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \equiv \frac{1}{2} \mathbf{w}^T \mathbf{w}
\]

subject to \( y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \ \forall i \)

\[
\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^{N} \alpha_i [1 - y_i (\mathbf{w}^T \mathbf{x}_i + b)]
\]

Here \( \alpha \) is the **the** Lagrange multiplier
Hard Margin Linear SVM Classification

• Note that at the optimal point the inequality becomes tight (equality).
• Our goal is to maximize the inequality to make it as tight as possible.
• For maximizing the inequality constraint, the Lagrange multiplier $\alpha$ is used.

$$
\mathcal{L}(\vec{w}, b, \alpha) = \frac{1}{2} \vec{w}^T \vec{w} + \sum_{i=1}^{N} \alpha_i \left[ 1 - y_i (\vec{w}^T \vec{x}_i + b) \right]
$$

Here $\alpha$ is the Lagrange multiplier

$$
\min_{\vec{w}, b} \frac{1}{2} \|\vec{w}\|^2 \equiv \frac{1}{2} \vec{w}^T \vec{w}
subject to \quad y_i (\vec{w}^T \vec{x}_i + b) \geq 1 \ \forall i
$$
Lagrange Multiplier

- The **optimal solution** for this Lagrangian function exists for the following conditions.
- These conditions are known as the **KKT conditions**.

\[
\frac{\partial}{\partial \vec{w}} \mathcal{L}(\vec{w}, b, \alpha) = 0
\]

\[
\frac{\partial}{\partial b} \mathcal{L}(\vec{w}, b, \alpha) = 0
\]

\[
\alpha_i \geq 0 \ \forall i
\]

\[
\alpha_i [1 - y_i (\vec{w}^T \vec{x}_i + b)] = 0 \ \forall i
\]

Complimentary Slackness
Lagrange Multiplier

\[
\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^{N} \alpha_i [1 - y_i (\mathbf{w}^T \mathbf{x}_i + b)]
\]

• Let’s try to understand these three KKT conditions for the optimality of the solution.

First condition: it is justified because if a minimum exists then the derivative of \( \mathcal{L} \) should be zero at the minimum.

Second condition: the constraint penalty should be non-negative.
The third condition is known as the “Complimentary Slackness” condition and needs some justification.

\[ \alpha_i [1 - y_i (\overrightarrow{w}^T \overrightarrow{x}_i + b)] = 0 \ \forall i \]

We need to understand why this constraint is required for an optimal solution to exist.
Hard Margin Linear SVM Classification

• Observe that the unconstrained optimization function or the Lagrangian function gives us a min-max optimization.

\[ \mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^{N} \alpha_i [1 - y_i (\mathbf{w}^T \mathbf{x}_i + b)] \]

Minimize the objective function.

This term is negative (mostly).

Thus we maximize the inequality to make it as tight as possible.

At the optimal point the inequality becomes tight (equality).

\[ \min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \equiv \frac{1}{2} \mathbf{w}^T \mathbf{w} \]

subject to \( y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \forall i \)
Hard Margin Linear SVM Classification

Thus, the Lagrangian function gives us a min-max optimization.

\[ \mathcal{L}(\vec{w}, b, \alpha) = \frac{1}{2} \vec{w}^T \vec{w} + \sum_{i=1}^{N} \alpha_i \left[ 1 - y_i (\vec{w}^T \vec{x}_i + b) \right] \]

**Minimize** the objective function.

**Maximize** the constraint term.

\[ \min_{\vec{w},b} \frac{1}{2} ||\vec{w}||^2 \equiv \frac{1}{2} \vec{w}^T \vec{w} \]

subject to \( y_i (\vec{w}^T \vec{x}_i + b) \geq 1 \ \forall i \)
Let’s explain further why we need to **maximize the constraint**.

Observe that some of the inequalities are **tight** \([y_i(\vec{w}^T\vec{x}_i + b) = 1]\) and some **are not** \([y_i(\vec{w}^T\vec{x}_i + b) \geq 1]\)

Since we care only about the optimal solution, we **ignore inequalities that are not tight** \([y_i(\vec{w}^T\vec{x}_i + b) \geq 1]\) which corresponds to \(\alpha = 0\).

This justifies the 3\(^{rd}\) condition.

\[\alpha_i[1 - y_i(\vec{w}^T\vec{x}_i + b)] = 0 \forall i\] Complimentary Slackness

\[
\min_{\vec{w}, b} \frac{1}{2} ||\vec{w}||^2 \equiv \frac{1}{2} \vec{w}^T\vec{w}
\]

subject to \(y_i(\vec{w}^T\vec{x}_i + b) \geq 1 \forall i\)

\[
\mathcal{L}(\vec{w}, b, \alpha) = \frac{1}{2} \vec{w}^T\vec{w} + \sum_{i=1}^{N} \alpha_i [1 - y_i(\vec{w}^T\vec{x}_i + b)]
\]

**Minimize** the objective function.  **Maximize** the constraint term.
Hard Margin Linear SVM Classification

\[ \mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^{N} \alpha_i \left[ 1 - y_i (\mathbf{w}^T \mathbf{x}_i + b) \right] \]

For instance, when an inequality is not tight, the term \( [1 - y_i (\mathbf{w}^T \mathbf{x}_i + b)] \) will be negative.

In this case how do we maximize the negative constraint?

Maximum of a negative constraint should be zero.

Hence, the negative constraint is maximized by setting \( \alpha_i = 0 \).
Hard Margin Linear SVM Classification

\[ \mathcal{L}(\vec{w}, b, \alpha) = \frac{1}{2} \|\vec{w}\|^2 + \sum_{i=1}^{N} \alpha_i [1 - y_i (\vec{w}^T \vec{x}_i + b)] \]

For those inequalities that **are tight** (active constraint), we can treat them as **equalities**: 
\[ [1 - y_i (\vec{w}^T \vec{x}_i + b)] = 0 \]

In other words, **either** \( \alpha_i = 0 \) 

or

\[ [1 - y_i (\vec{w}^T \vec{x}_i + b)] = 0 \]

Thus, the **3rd KKT condition** is justified.

Complimentary Slackness

\[ \alpha_i [1 - y_i (\vec{w}^T \vec{x}_i + b)] = 0 \forall i \]
Hard Margin Linear SVM Classification

- Why are the **three KKT conditions** useful?

\[
\begin{align*}
\frac{\partial}{\partial \vec{w}} \mathcal{L}(\vec{w}, b, \alpha) &= 0 \\
\frac{\partial}{\partial b} \mathcal{L}(\vec{w}, b, \alpha) &= 0 \\
\alpha_i &\geq 0 \quad \forall i \\
\alpha_i [1 - y_i (\vec{w}^T \vec{x}_i + b)] &= 0 \quad \forall i
\end{align*}
\]

Primal Lagrangian

\[
\mathcal{L}(\vec{w}, b, \alpha) = \frac{1}{2} \vec{w}^T \vec{w} + \sum_{i=1}^{N} \alpha_i [1 - y_i (\vec{w}^T \vec{x}_i + b)]
\]

Complimentary Slackness

\[
\min_{\vec{w}, b} \frac{1}{2} \|\vec{w}\|^2 \equiv \frac{1}{2} \vec{w}^T \vec{w} \\
\text{subject to} \quad y_i (\vec{w}^T \vec{x}_i + b) \geq 1 \quad \forall i
\]
Hard Margin Linear SVM Classification

- The KKT conditions allow to transform an unconstrained optimization **primal problem** (dependent on d) into a **dual problem** (dependent on N).

Let’s see how this works…

Primal Lagrangian

$$\mathcal{L}(\overline{w}, b, \alpha) = \frac{1}{2} \overline{w}^T \overline{w} + \sum_{i=1}^{N} \alpha_i [1 - y_i (\overline{w}^T \overline{x}_i + b)]$$
Hard Margin Linear SVM Classification

• Solving the first KKT condition with respect to $\mathbf{w}$ we get:

\[
\frac{\partial}{\partial \mathbf{w}} \mathcal{L}(\mathbf{w}, b, \alpha) = 0
\]

\[
\frac{\partial}{\partial b} \mathcal{L}(\mathbf{w}, b, \alpha) = 0
\]

\[
\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^{N} \alpha_i [1 - y_i (\mathbf{w}^T \mathbf{x}_i + b)]
\]

\[
\frac{\partial}{\partial \mathbf{w}} \mathcal{L}(\mathbf{w}, b, \alpha) = \mathbf{w}^T + \sum_{i=1}^{N} \alpha_i [-y_i \mathbf{x}_i^T] = 0
\]

\[
\mathbf{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \mathbf{x}_i
\]

It reveals something very useful!

The optimal weight vectors are linear combination of the data!
Now solving the first KKT condition with respect to $b$ we get:

$$\frac{\partial}{\partial b} L(\overrightarrow{w}, b, \alpha) = \sum_{i=1}^{N} \alpha_i \ [-y_i] = 0$$

This gives a constraint on the Lagrange multiplier $\alpha$

$$\sum_{i=1}^{N} \alpha_i^* y_i = 0$$

$$\frac{\partial}{\partial \overrightarrow{w}} L(\overrightarrow{w}, b, \alpha) = 0$$

$$\frac{\partial}{\partial b} L(\overrightarrow{w}, b, \alpha) = 0$$

$$L(\overrightarrow{w}, b, \alpha) = \frac{1}{2} \overrightarrow{w}^T \overrightarrow{w} + \sum_{i=1}^{N} \alpha_i [1 - y_i (\overrightarrow{w}^T \overrightarrow{x}_i + b)]$$
Hard Margin Linear SVM Classification

• Thus, solving the **first KKT condition** with respect to $\mathbf{w}$ and $b$ we get:

$$
\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^{N} \alpha_i \left[ 1 - y_i (\mathbf{w}^T \mathbf{x}_i + b) \right]
$$

Using these expressions in the primal Lagrangian we will derive the **dual Lagrangian**.
Hard Margin Linear SVM Classification

- We plug in $w^*$ in the **primal Lagrangian** (that depends on $d$):

\[
\mathcal{L}(\overline{w}^*, b^*, \alpha) = \frac{1}{2} \overline{w}^* T \overline{w}^* + \sum_{i=1}^{N} \alpha_i \left[ 1 - y_i((\overline{w}^*)^T \overline{\mathbf{x}}_i + b^*) \right]
\]

\[
\mathcal{L}(\overline{w}^*, b^*, \alpha) = \frac{1}{2} \left[ \sum_{i=1}^{N} \alpha_i y_i \overline{\mathbf{x}}_i \right]^T \left[ \sum_{j=1}^{N} \alpha_j y_j \overline{\mathbf{x}}_j \right] + \sum_{i=1}^{N} \alpha_i \left[ 1 - y_i \left( \left( \sum_{j=1}^{N} \alpha_j y_j \overline{\mathbf{x}}_j \right)^T \overline{\mathbf{x}}_i + b^* \right) \right]
\]

\[
\mathcal{L}(\overline{w}^*, b^*, \alpha) = \frac{1}{2} \left[ \sum_{i=1}^{N} \alpha_i y_i \overline{\mathbf{x}}_i \right]^T \left[ \sum_{j=1}^{N} \alpha_j y_j \overline{\mathbf{x}}_j \right] + \sum_{i=1}^{N} \alpha_i - \left[ \sum_{i=1}^{N} \alpha_i y_i \overline{\mathbf{x}}_i \right]^T \left[ \sum_{j=1}^{N} \alpha_j y_j \overline{\mathbf{x}}_j \right] - \sum_{i=1}^{N} \alpha_i y_i b^*
\]

\[
\mathcal{L}(\overline{w}^*, b^*, \alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \left[ \sum_{i=1}^{N} \alpha_i y_i \overline{\mathbf{x}}_i \right]^T \left[ \sum_{j=1}^{N} \alpha_j y_j \overline{\mathbf{x}}_j \right]
\]

Last term is zero using the constraint on $\alpha$

\[
\sum_{i=1}^{N} \alpha_i y_i = 0
\]
Hard Margin Linear SVM Classification

• See the **primal Lagrangian** (dependent on d) is transformed into the **dual Lagrangian** (depends on N):

\[
\mathcal{L}(\vec{w}^*, b^*, \alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \left[ \sum_{i=1}^{N} \alpha_i y_i \hat{x}_i \right]^T \left[ \sum_{j=1}^{N} \alpha_j y_j \hat{x}_j \right]
\]

**Dual:** Only **maximize** wrt \( \alpha \) (\( w \) is gone!)

**Primal:** **minimize** wrt \( w \) and maximize wrt \( \alpha \)

\[
\mathcal{L}(\vec{w}, b, \alpha) = \frac{1}{2} \vec{w}^T \vec{w} + \sum_{i=1}^{N} \alpha_i [1 - y_i (\vec{w}^T \hat{x}_i + b)]
\]

\[
\vec{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \hat{x}_i
\]
Hard Margin Linear SVM Classification

- So the **SVM dual problem** can be formulated as:

  \[
  \max_{\alpha_1, \ldots, \alpha_N} \alpha^T \mathbf{1} - \frac{1}{2} \alpha^T H \alpha
  \]

  subject to \( \sum_{i=1}^{N} \alpha_i y_i = 0 \)

  \( \alpha_i \geq 0 \)

  \( \mathbf{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \mathbf{x}_i \)

  \[
  \min_{w,b} \frac{1}{2} \|\mathbf{w}\|^2 \equiv \frac{1}{2} \mathbf{w}^T \mathbf{w}
  \]

  subject to \( y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \ \forall i \)

  \[
  \mathbf{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \mathbf{x}_i
  \]

  upper bound of time-complexity \( O(N^3) \)

  where \( H_{ij} = y_i y_j \mathbf{x}_i^T \mathbf{x}_j \)

  The size of dual depends on \( N \) (not on \( d \))
Hard Margin Linear SVM Classification

• Notice that in the dual objective function, data (x) appears only as a **dot product**.
• Thus the dimension of the data does not influence the complexity of this optimization.
• Data could be **infinite dimensional**, yet the complexity of this optimization is only determined by \( N \).
• That’s why the dual problem formulation is useful to handle the **complexity of high-dimensional data** (detail later).

\[
\mathcal{L}(\mathbf{w}^*, b^*, \alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j
\]
Hard Margin Linear SVM Classification

• The goal now is to **find the vector \( \alpha \) that maximizes** this function, with \( \alpha_i \geq 0 \) for all instances.
• We can use a QP solver (e.g., CVXOPT).
• Then, we can **compute** \( w \) using the optimal solution for \( w \).
• What about \( b \)?

\[
\mathbf{w}^* = \sum_{i=1}^{N} \alpha_i^* \mathbf{y}_i \mathbf{x}_i
\]

\[\alpha_i \geq 0 \ \forall i\]

\[
\mathcal{L}(\mathbf{w}^*, b^*, \alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j
\]
Hard Margin Linear SVM Classification

• What about $b$?
• Well, we could simply use the equation of the margin to find $b$.

\[ y_i (\vec{w}^T \vec{x}_i + b) = 1 \]

\[ b^* = 1 - y_i (\vec{w}^T \vec{x}_i) \]

\[ \vec{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \vec{x}_i \]

\[ \alpha_i \geq 0 \ \forall i \]
Hard Margin Linear SVM Classification

• What about $b$?
• But there is something else that we need to consider first.
• The parameter $b$ doesn’t depend on all $x$.
• Let’s see why.

$$y_i(\overrightarrow{w}^T \overrightarrow{x}_i + b) = 1$$

$$b^* = 1 - y_i(\overrightarrow{w}^T \overrightarrow{x}_i)$$

$$\overrightarrow{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \overrightarrow{x}_i$$

$$\alpha_i \geq 0 \ \forall i$$
Hard Margin Linear SVM Classification

• Once we solve for $\alpha$ (explained later), we see that though there are $N$ $\alpha$’s, most will vanish with $\alpha_i = 0$ and only a small percentage have $\alpha_i > 0$.

• Points that are away from the decision boundary, their $y_i(w^T x_i + b) >> 1$.

• For these out-of-boundary points $[1 - y_i(w^T x_i + b)]$ will be negative, and hence $\alpha$ is maximized to 0.

$$L(w, b, \alpha) = \frac{1}{2} w^T w + \sum_{i=1}^{N} \alpha_i [1 - y_i(w^T x_i + b)]$$

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

$$\alpha_i \geq 0 \ \forall i$$
Hard Margin Linear SVM Classification

• Now consider the **points on the boundary**.
• For them: \( y_i(w^T x_i + b) = 1 \)
• This makes the \([1 - y_i(w^T x_i + b)]\) term zero, hence hence **\( \alpha \) is maximized** to > 0.
• But the **points on the boundary are few**.
• That’s why **only a small percentage of points** have \( \alpha_i > 0 \).

\[
\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^{N} \alpha_i \left[ 1 - y_i \mathbf{w}^T \mathbf{x}_i + b \right]
\]

\[
\mathbf{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \mathbf{x}_i
\]

\( \alpha_i \geq 0 \ \forall i \)
Hard Margin Linear SVM Classification

• The set of \( x_i \) whose \( \alpha_i > 0 \) are the support vectors.
• We see in the equation of \( w \) that is written as the weighted sum of these training instances that are selected as the support vectors.
• These are the \( x_i \) that satisfy the following equation and lie on the margin.

\[
y_i (\mathbf{w}^T \mathbf{x}_i + b) = 1
\]

\[
\mathbf{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \mathbf{x}_i
\]

\[
\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^{N} \alpha_i [1 - y_i (\mathbf{w}^T \mathbf{x}_i + b)]
\]
Hard Margin Linear SVM Classification

• Using this observation ($\alpha_i > 0$ for the support vectors), we can calculate $b$.

$$y_i(\vec{w}^T \vec{x}_i + b) = 1$$

$$b^* = 1 - y_i(\vec{w}^T \vec{x}_i)$$

For **numerical stability**, it is advised that this be done for all support vectors and an average be taken.

$$b^* = \frac{1}{N_{\text{support vectors}}} \sum_{i=1}^{N} \left[ 1 - y_i(\vec{w}^T \vec{x}_i) \right]_{\alpha_i^* > 0}$$
Hard Margin Linear SVM Classification

- The **majority of the** \( \alpha_i \) **are 0**, for which \( y_i(w^T x_i + b) > 1 \).
- These are the \( x_i \) points that lie more than sufficiently away from the discriminant, and they have **no effect on the hyperplane**.
- The instances that are **not support vectors carry no information**.

Even if any subset of them **are removed**, we would **still get the same solution**.
Hard Margin Linear SVM Classification

• Let's look at the SVM dual problem:

\[
\begin{align*}
\max_{\alpha_1, \ldots, \alpha_N} & \quad \mathbf{\alpha}^T \mathbf{1} - \frac{1}{2} \mathbf{\alpha}^T H \mathbf{\alpha} \\
\text{subject to} & \quad \sum_{i=1}^{N} \alpha_i y_i = 0 \\
& \quad \alpha_i \geq 0
\end{align*}
\]

where \( H_{ij} = y_i y_j \mathbf{x}_i^T \mathbf{x}_j \)

Is this a **convex** QP?

\[
L(\mathbf{w}^*, b^*, \mathbf{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j
\]

64
Hard Margin Linear SVM Classification

- We can show that $H$ is convex.

$$H = \begin{bmatrix} y_i \vec{x}_i^T & y_j \vec{x}_j & \vdots & y_N \vec{x}_N \end{bmatrix} \begin{bmatrix} \begin{bmatrix} y_i \vec{x}_i \end{bmatrix} & \begin{bmatrix} y_j \vec{x}_j \end{bmatrix} & \cdots & \begin{bmatrix} y_N \vec{x}_N \end{bmatrix} \end{bmatrix}$$

$$H_{ij} = y_i y_j \vec{x}_i \vec{x}_j$$

For an arbitrary vector $\alpha$

$$\alpha^T H \alpha = \alpha^T A^T A \alpha = \|A \alpha\|^2 \geq 0$$

$$H \geq 0$$
Hard Margin Linear SVM Classification

• Therefore, \( H \) is **positive semidefinite**, hence convex.

• Thus, the dual QP objective is a **convex optimization problem**.

• Note that original objective function or the **primal QP was also convex**.

\[
\max_{\alpha_1, \ldots, \alpha_N} \hat{a}^T \mathbf{1} - \frac{1}{2} \hat{a}^T H \hat{a}
\]

where \( H_{ij} = y_i y_j \vec{x}_i \vec{x}_j \)
Hard Margin Linear SVM Classification

- We can use a standard QP solver to find $\alpha$ from the dual OP objective convex optimization problem.

\[
\max_{\alpha_1, \ldots, \alpha_N} \alpha^T \bar{1} - \frac{1}{2} \bar{\alpha}^T H \bar{\alpha}
\]

subject to $\sum_{i=1}^{N} \alpha_i y_i = 0$

$\alpha_i \geq 0$

where $H_{ij} = y_i y_j \bar{x}_i^T \bar{x}_j$

Then, we use the following two equations (from the primal) to find the weight vectors.

\[
\mathbf{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \bar{x}_i
\]

\[
b^* = \frac{1}{N_{\text{support vectors}}} \sum_{i=1}^{N} \left[ 1 - y_i (\mathbf{w}^T \bar{x}_i) \right]\]
Prediction In Hard Margin Linear SVM

\[ \overrightarrow{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \overrightarrow{x}_i \]

\[ b^* = \frac{1}{N_{support\ vectors}} \sum_{i=1}^{N} [1 - y_i (\overrightarrow{w}^T \overrightarrow{x}_i)] \]
Hard Margin Linear SVM Classification

• During testing, we **do not enforce a margin**.

• For a test input \( \hat{x} \), we compute \( \hat{y}(\hat{x}) \) as follows.
\[
\hat{y}(\hat{x}) = \text{sign}(\hat{w}^T \cdot \hat{x} + \hat{b})
\]

• Then, we choose the class of the data according to the sign of \( \hat{y}(\hat{x}) \):
  - Choose **Class 1** if \( \hat{y}(\hat{x}) > 0 \) and **Class 2** otherwise.

• The complexity is \( \mathcal{O}(sd) \), where \( s < N \) is the number of support vectors.

\[
\hat{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \hat{x}_i
\]

\[
b^* = \frac{1}{N_{\text{support vectors}}} \sum_{i=1}^{N} \left[ 1 - y_i (\hat{w}^T \hat{x}_i) \right]
\]
Hard Margin Linear SVM Classification

- Observe that unlike Logistic Regression, we *don’t compute probabilistic output*. The SVM predictions are *not calibrated*!

For a test input \( \tilde{x} \), we compute \( \hat{y}(\tilde{x}) \) as follows:

\[
\hat{y}(\tilde{x}) = \text{sign}(\overline{w}^* T \cdot \tilde{x} + b^*)
\]

Then, we choose the class of the data according to the \text{sign} of \( \hat{y}(\tilde{x}) \):
- Choose **Class 1** if \( \hat{y}(\tilde{x}) > 0 \) and **Class 2** otherwise.

\[
\begin{align*}
\overline{w}^* &= \sum_{i=1}^{N} \alpha_i^* y_i \tilde{x}_i \\
 b^* &= \frac{1}{N_{support \ vectors}} \sum_{i=1}^{N} [1 - y_i (\overline{w}^T \tilde{x}_i)]
\end{align*}
\]
Linear SVM: Significance of the Dual Representation
Hard Margin Linear SVM Classification

• Why do we care about the **dual representation** of the optimization problem?

• We can give at least three reasons.

\[
\min_{\vec{w},b} \frac{1}{2} \|\vec{w}\|^2 \equiv \frac{1}{2} \vec{w}^T \vec{w}
\]

subject to \( y_i (\vec{w}^T \vec{x}_i + b) \geq 1 \ \forall i \)

\[
\max_{\alpha_1, \ldots, \alpha_N} \alpha^T \vec{1} - \frac{1}{2} \alpha^T H \alpha
\]

subject to \( \sum_{i=1}^{N} \alpha_i y_i = 0 \)

\( \alpha_i \geq 0 \)

where \( H_{ij} = y_i y_j \vec{x}_i^T \vec{x}_j \)
Hard Margin Linear SVM Classification

• **Reason 1:**

• The dual QP problem **exposes the structure about the problem!**

• Recall the KKT conditions (1st one):

\[ \vec{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \vec{x}_i \]

The optimal weight vectors are **linear combination of the data!** Wow!
Hard Margin Linear SVM Classification

- Reason 1:
- The equation for w raises **two concerns**.

Does this mean that we need all training data to find w?

Does this mean the number of unknown parameters $\alpha$ is equal to the size of training data?

We can show that a large majority of the training data will have a **zero value** for $\alpha$, so w is defined by a few training samples (**support vectors**).
Hard Margin Linear SVM Classification

• Reason 1:
• Recall the complementary slackness condition:

\[ \alpha_i^* [1 - y_i ((\vec{w}^*)^T \vec{x}_i + b^*)] = 0 \forall i \]

Thus

\[ \alpha_i^* > 0 \quad \text{yields} \quad y_i ((\vec{w}^*)^T \vec{x}_i + b^*) = 1 \]

\[ \alpha_i^* > 0: \text{leads to the fact that corresponding } x_i \text{ lies on the Margin.} \]

These are the support vectors!
Hard Margin Linear SVM Classification

- **Reason 1:** Hence, weights for the decision boundary is determined **only by the support vectors.**

\[
\mathbf{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \mathbf{x}_i = \sum_{i \in \text{support vectors}} \alpha_i^* y_i \mathbf{x}_i
\]

It’s a mathematically precise way of saying “other data points don’t matter”!
Hard Margin Linear SVM Classification

- **Reason 2:**
  - Dual QP has only **one non-trivial constraint.**
  - Unconstrained or “nearly unconstrained” QPs are sometimes **easier to solve.**

\[
\begin{align*}
\max_{\alpha_1,\ldots,\alpha_N} & \quad \tilde{\alpha}^T \mathbf{1} - \frac{1}{2} \tilde{\alpha}^T H \tilde{\alpha} \\
\text{subject to} & \quad \sum_{i=1}^{N} \alpha_i y_i = 0 \\
& \quad \alpha_i \geq 0
\end{align*}
\]

where \( H_{ij} = y_i y_j \tilde{x}_i^T \tilde{x}_j \)
Hard Margin Linear SVM Classification

- **Reason 2:**
  - The dual problem is **faster to solve** than the primal when the **number of training instances is smaller** than the number of features.
  - E.g., Platt’s SMO (Sequential Minimal Optimization) algorithm solves SVM dual QP in $O(N^2)$ time.

\[
\begin{align*}
\max_{\alpha_1,\ldots,\alpha_N} & \quad \alpha^T 1 - \frac{1}{2} \alpha^T H \alpha \\
\text{subject to} & \quad \sum_{i=1}^{N} \alpha_i y_i = 0 \\
& \quad \alpha_i \geq 0
\end{align*}
\]

where $H_{ij} = y_i y_j \vec{x}_i \vec{x}_j$
Hard Margin Linear SVM Classification

• Reason 3:

• Dual QP makes the kernel trick possible, while the primal does not.