Fundamentals of Machine Learning:

Kernel Machines I: The Linear Support Vector Machine

Prof. Andrew D. Bagdanov (andrew.bagdanov AT unifi.it)

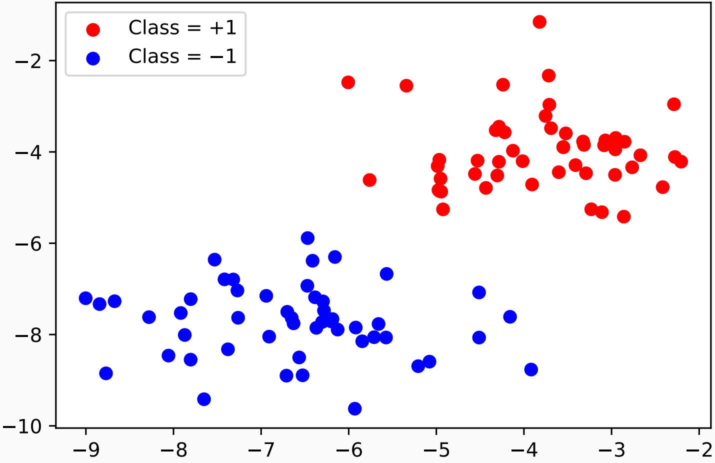
UNIVERSITÀ DEGLI STUDI FIRENZE
DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Outline

Introduction

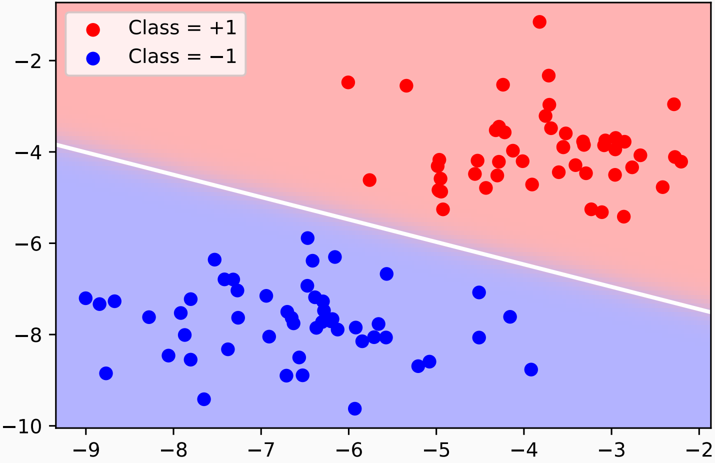
Motivations

- Let's consider a simple, linearly-separable classification problem:



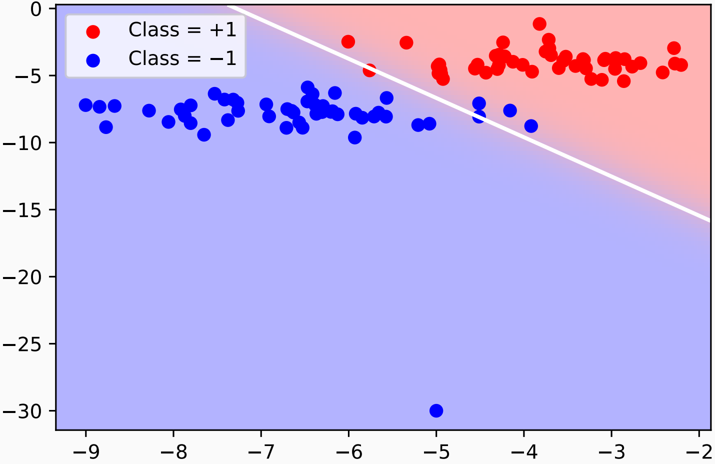
Motivations: a probabilistic approach

- We have tools for these problems, e.g. a generative linear discriminant:



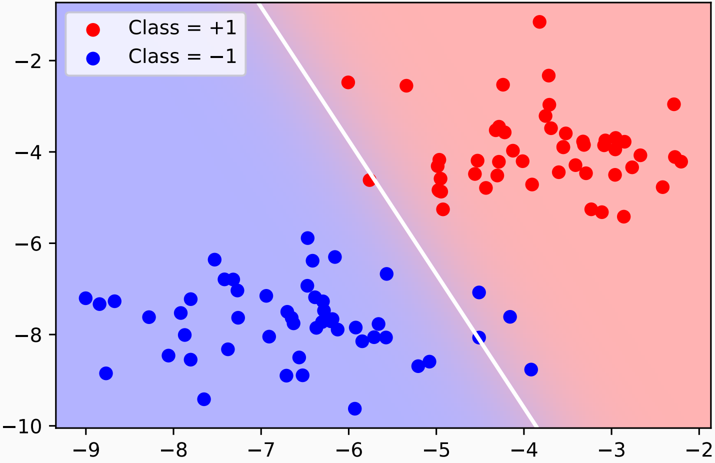
Motivations: sensitivity to outliers

- A problem with many probabilistic approaches is sensitivity to outliers:



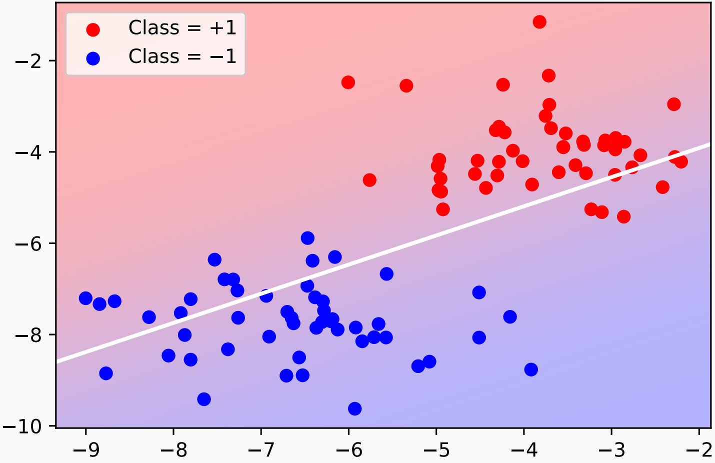
Motivations: sensitivity to outliers

- The effect on the separating hyperplane is more evident up close:



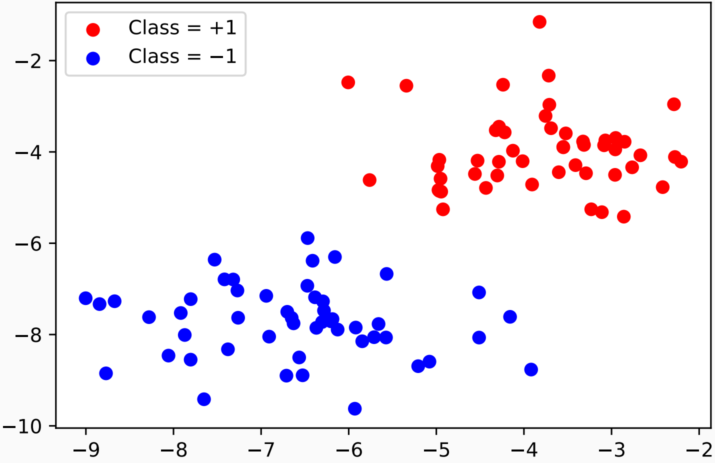
Motivations: some outliers are worse than others...

- Methods that treat all samples equally can quickly degrade:



Motivations: margin classifiers, some intuition

- Can we reformulate a classification objective in terms of only the margin?



Lecture objectives

After this lecture you will:

- Have gained a deeper understanding of the geometry of classification and how margin scaling is related to the linear discriminant parameter $\mathbf{w}$.
- Understand the primal form of the Maximum Margin Classifier – also known as the Support Vector Machine (SVM).
- Understand how the dual form of the Support Vector Machine is derived from the Lagrangian of the maximum margin formulation.
- Understand how slack variables can be introduced into the SVM formulation to account for datasets that are not linearly separable.
- Be able to interpret the dual variables and how they identify support vectors in the training set.

The Margin

Definition (Bilinear Map)

A function $\Omega : V \times V \to \mathbb{R}$ is a *bilinear map* from vector space V to $\mathbb{R}$ iff:

$$\begin{aligned} \Omega(\lambda \mathbf{x} + \psi \mathbf{y}, \mathbf{z}) &= \lambda \Omega(\mathbf{x}, \mathbf{z}) + \psi \Omega(\mathbf{y}, \mathbf{z}) \\ \Omega(\mathbf{x}, \lambda \mathbf{y} + \psi \mathbf{z}) &= \lambda \Omega(\mathbf{x}, \mathbf{y}) + \psi \Omega(\mathbf{x}, \mathbf{z}) \end{aligned}$$

for any $\mathbf{x}, \mathbf{y}, \mathbf{z} \in V$.

- Ω is called symmetric if $\Omega(\mathbf{x}, \mathbf{y}) = \Omega(\mathbf{y}, \mathbf{x})$ for all $\mathbf{x}, \mathbf{y} \in V$.
- Ω is called positive definite if:

$$\Omega(\mathbf{x}, \mathbf{x}) \geq 0 \text{ for all } \mathbf{x}, \ \text{ and } \Omega(\mathbf{x}, \mathbf{x}) = 0 \text{ iff } \mathbf{x} = \mathbf{0}$$

Definition (Inner Product and Inner Product Space)

Let V be any vector space and $\Omega : V \times V \to \mathbb{R}$ any bilinear map from V to $\mathbb{R}$. Then:

- If Ω is symmetric and positive definite, Ω is called an inner product on V. We usually write $\langle \mathbf{x}, \mathbf{y} \rangle$ instead of $\Omega(\mathbf{x}, \mathbf{y})$.
- The pair (V, Ω) (or $(V, \langle \cdot, \cdot \rangle)$) for inner product Ω is called an inner product space or vector space with inner product. If $\Omega(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \mathbf{y}$, (V, Ω) is called a Euclidean vector space.

Inner products allow us to formalize our geometrical intuitions about length, orthogonality, and distance.

Maximum Margin Classifiers

The (geometric) classification problem

- A useful way to think about classification is that we:
  1. Represent data in $\mathbb{R}^D$.
  2. Partition $\mathbb{R}^D$ in such a way that samples with the same label (and no samples with different labels) fall into the same partition.
- We will consider a convenient partitioning – that of separating $\mathbb{R}^D$ into two halves using a *separating hyperplane*.
- Consider a function $f : \mathbb{R}^D \to \mathbb{R}$ defined as:

$$f(\mathbf{x}; \mathbf{w}, b) = \langle \mathbf{w}, \mathbf{x} \rangle + b.$$

- We define a hyperplane partitioning our space using f as:

$$H = \{ \mathbf{x} \mid f(\mathbf{x}; \mathbf{w}, b) = \langle \mathbf{w}, \mathbf{x} \rangle + b = 0 \}$$

The relationship between $\mathbf{w}$ and H

- The hyperplane defined by $\mathbf{w}$ and b is perpendicular to $\mathbf{w}$.
- To see this, pick any $\mathbf{x}_1$ and $\mathbf{x}_2$ in H and consider:

$$\begin{aligned} f(\mathbf{x}_1) - f(\mathbf{x}_2) &= \langle \mathbf{w}, \mathbf{x}_1 \rangle + b - \langle \mathbf{w}, \mathbf{x}_2 \rangle - b \\ &= \langle \mathbf{w}, \mathbf{x}_1 - \mathbf{x}_2 \rangle \end{aligned}$$

How we use w and b

- When we are presented with a test sample $\mathbf{x}$, we will classify it according to which side of the hyperplane it lies:

$$\text{class}(\mathbf{x}) = \begin{cases} +1 & \text{if } f(\mathbf{x}; \mathbf{w}, b) \geq 0 \\ -1 & \text{if } f(\mathbf{x}; \mathbf{w}, b) < 0 \end{cases}$$

- When training on data $\{ (\mathbf{x}_i, y_i) \mid i = 1, \ldots, N \}$, we are searching for $\mathbf{w}$ and b such that all samples fall on the correct side of the hyperplane:

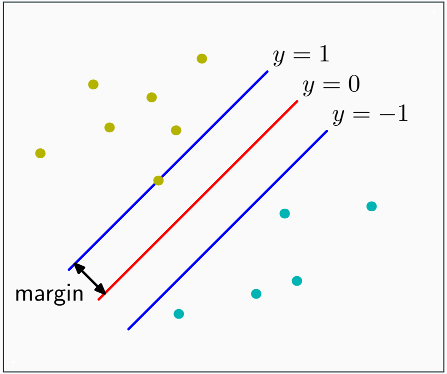
$$\begin{aligned} \langle \mathbf{w}, \mathbf{x}_i \rangle + b \geq 0 \quad & \text{when} \quad y_i = +1 \\ \langle \mathbf{w}, \mathbf{x}_i \rangle + b < 0 \quad & \text{when} \quad y_i = -1 \end{aligned}$$

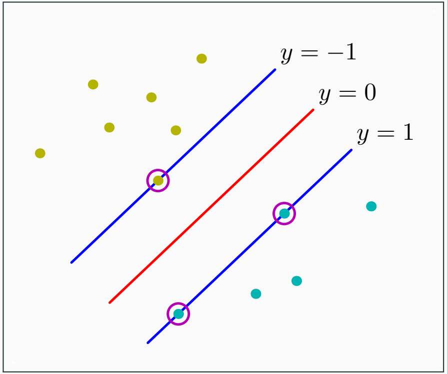
- These conditions are often combined into the more compact:

$$y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 0$$

The margin

- The margin is defined as the distance between a separating hyperplane and the closest point to it.
- Our goal is to maximize this distance, but what is it?





Maximizing the margin

- The perpendicular distance between any point $\mathbf{x}_n$ and a hyperplane defined by $\langle \mathbf{w}, \mathbf{x} \rangle + b = 0$ is:

$$\frac{y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + b)}{||\mathbf{w}||}$$

- We want to maximize the minimum such distance:

$$\arg\max_{\mathbf{w}, b} \left\{ \frac{1}{||\mathbf{w}||} \min_n \left[y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + b) \right] \right\}$$

subject to the constraints that $y_n(\langle \mathbf{w}, \mathbf{x} \rangle + b) \geq 0$.

- This is an inconvenient optimization problem due to the max/min and changing closest point.

Maximizing the margin

- Note, however, that we can freely scale $\mathbf{w}$ and b without changing the distance between points and the hyperplane.
- So, we can scale so that $\langle \mathbf{w}, \mathbf{x}_a \rangle + b = 1$ for the closest point $\mathbf{x}_a$.
- Let r be the orthogonal distance from $\mathbf{x}_a$ to the hyperplane.
- Then, the orthogonal projection of $\mathbf{x}_a$ onto the hyperplane is:

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

- Let's plug this into the fact that $\mathbf{x}_a'$ lies on the hyperplane:

$$\langle \mathbf{w}, \mathbf{x}_a - r\frac{\mathbf{w}}{||\mathbf{w}||} \rangle + b = 0$$

Maximizing the margin

- Let's plug this into the fact that $\mathbf{x}'_a$ lies on the hyperplane:

$$\langle \mathbf{w}, \mathbf{x}_a - r\frac{\mathbf{w}}{||\mathbf{w}||} \rangle + b = 0$$

- Now, exploiting bilinearity of the inner product:

$$\langle \mathbf{w}, \mathbf{x}_a \rangle + b - r\frac{\langle \mathbf{w}, \mathbf{w} \rangle}{||\mathbf{w}||} = 0$$

- Recalling that $\mathbf{x}_a$ lies on the margin, we arrive at:

$$r = \frac{1}{||\mathbf{w}||}$$

The canonical form of the Hard Margin SVM

- Combining margin maximization with the constraints we have arrive at:

$$\begin{aligned} \max_{\mathbf{w},b} \quad & \frac{1}{||\mathbf{w}||} \\ \text{subject to} \quad & y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + b) \geq 1 \text{ for all } n = 1, \ldots, N \end{aligned}$$

- Or, the more common canonical representation of the Hard Margin SVM:

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}||\mathbf{w}||^2 \\ \text{subject to} \quad & y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + b) \geq 1 \text{ for all } n = 1, \ldots, N \end{aligned}$$

Finding $\mathbf{w}$ and b

- We have a convex quadratic programming problem in D variables with linear constraints.
- To solve such a problem, we can form the Lagrangian function:

$$L(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{n=1}^{N} a_n \left\{ y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + b) - 1 \right\}$$

- Setting $\frac{\partial}{\partial \mathbf{w}} L = 0$ and $\frac{\partial}{\partial b} L = 0$ we obtain:

$$\begin{aligned} \mathbf{w} &= \sum_{n=1}^{N} a_n y_n \mathbf{x}_n \\ 0 &= \sum_{n=1}^{N} a_n y_n \end{aligned}$$

Finding $\mathbf{w}$ and b

- Substituting this value of $\mathbf{w}$ and the constraint on $\sum_n a_n y_n$ into the Lagrangian:

$$\max_{\mathbf{a}} \left\{ \sum_{n=1}^{N} a_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a_n a_m y_n y_m \langle \mathbf{x}_n, \mathbf{x}_m \rangle \right\}$$

$$\text{subject to} \quad a_n \geq 0, \;\; \text{for } n = 1, \ldots, N$$

$$\sum_{n=1}^{N} a_n y_n = 0$$

- This is the dual representation of the Hard Margin SVM, again a quadratic programming problem, but in N variables
- The complexity of solving quadratic problems in N variables is $O(N^3)$.

Using the SVM: the "Support" in Support Vector Machines

- To use the classifier we again substitute our $\mathbf{w}$ into the decision function:

$$f(\mathbf{x}) = \sum_{n=1}^{N} a_n y_n \langle \mathbf{x}, \mathbf{x}_n \rangle + b$$

- The Karush-Kuhn-Tucker (KKT) conditions mean that the solution satisfies:

$$\begin{aligned} a_n &\geq 0 \\ y_n f(\mathbf{x}_n) - 1 &\geq 0 \\ a_n \left\{ y_n f(\mathbf{x}_n) - 1 \right\} &= 0 \end{aligned}$$

- So, for all n either $a_n = 0$ or $y_n f(\mathbf{x_n}) = 1$.
- The $\mathbf{x}_n$ for which $a_n > 0$ and $y_n f(\mathbf{x_n}) = 1$ are called support vectors.

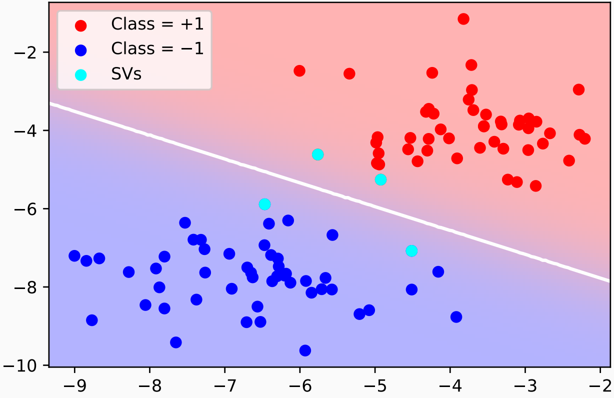
Sparse Kernel Machines (aka SVMs)

- Note that only the support vectors contribute to classification:

$$f(\mathbf{x}) = \sum_{n=1}^{N} a_n y_n \langle \mathbf{x}, \mathbf{x}_n \rangle + b = \sum_{m \in \mathrm{SV}} a_m y_m \langle \mathbf{x}, \mathbf{x}_m \rangle + b$$

- This is why SVMs are also more generally known as Sparse Kernel Machines.
- (We will see where the kernel comes from in the next lecture...)

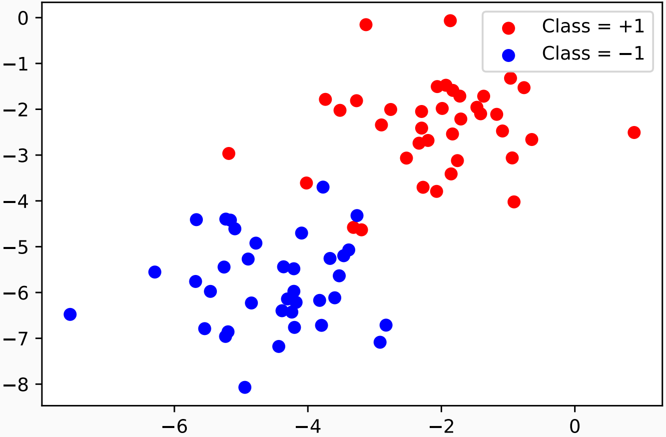
- We have a linear classifier that is now robust to outliers:



The Soft Margin Classifier

Overlapping class distributions

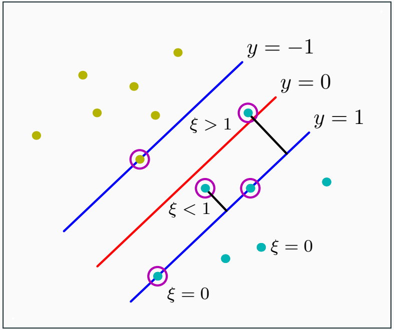
- Until now we have assumed that our problem is linearly separable.
- This is, clearly, almost never the case.



Allowing for misclassifications of training samples

- To allow for the possibility of some training samples to be misclassified, we introduce slack variables ξ_n:

$$\xi_n = \begin{cases} 0 & \text{if } \mathbf{x}_n \text{ is on or on the correct side of margin} \\ |y_n - \langle \mathbf{w}, \mathbf{x}_n \rangle + b| & \text{otherwise} \end{cases}$$



The optimization problem with slack

- The new optimization problem becomes:

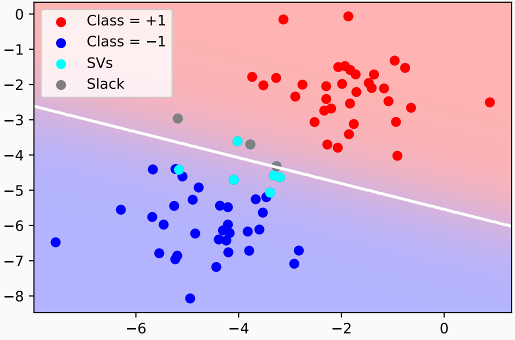
$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}||\mathbf{w}||^2 + C\sum_{n=1}^{N}\xi_n \\ \text{subject to} \quad & y_n(\langle \mathbf{w}, \mathbf{x}_n\rangle + b) \geq 1 - \xi_n \text{ for all } n = 1, \ldots, N \end{aligned}$$

- And after forming the Lagrangian and solving for the dual variables (Bishop pages 332-334):

$$\begin{aligned} \max_{\mathbf{a}} & \left\{ \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m y_n y_m \langle \mathbf{x}_n, \mathbf{x}_m\rangle \right\} \\ \text{subject to} \quad & 0 \leq a_n \leq C, \text{ for } n = 1, \ldots, N \\ & \sum_{n=1}^{N} a_n y_n = 0 \end{aligned}$$

The Soft Margin solution

- The form of the result is nearly identical to the hard margin case.
- Note, however, that the support vectors now include misclassified samples.
- Since the penalty for misclassification scales linearly with ξ the soft margin SVM is not robust to outliers.



Concluding Remarks

The Support Vector Machine

- The linear SVM is a powerful classifier that is robust to outliers (in the hard margin case).
- It can be adapted to handle problems that are not linearly separable.
- But this comes at the cost of introducing a hyperparameter C that trades-off the cost of misclassification with maximizing the margin.
- The real advantage of the SVM is that it is a convex quadratic problem which has a unique solution and admits efficient algorithms.
- In the next lecture we will see how we can extend this theory to nonlinear decision boundaries.

Reading and Homework Assignments

Reading Assignment:

- Bishop: Chapter 7 (7.1), Chapter 6 (6.1, 6.2)

Homework:

- Show that $\Omega(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T\mathbf{y}$ is an inner product.
- Show that, for $V = \mathbb{R}^2$, $\Omega(\mathbf{x}, \mathbf{y}) = x_1y_1 - (x_1y_2 + x_2y_1) + 2x_2y_2$ is an inner product.
- Show that we can scale the margin be an arbitrary constant γ (i.e. $y_n(\langle \mathbf{w}, \mathbf{x_n} \rangle + b) \geq \gamma$) and the solution to the maximum margin hyperplane does not change.