Introduction to Support Vector Machines

L FLEX, Machine Learning, Paul Sabatier University

Romain Thoreau
romain.thoreau@cnes.fr

February 2025

Binary Classification: Introductory Examples

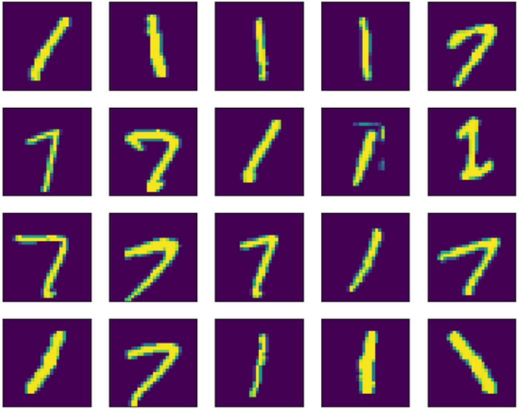


Figure: Images (28px × 28px) of hand-written digits (1 & 7)

Binary Classification: Introductory Examples

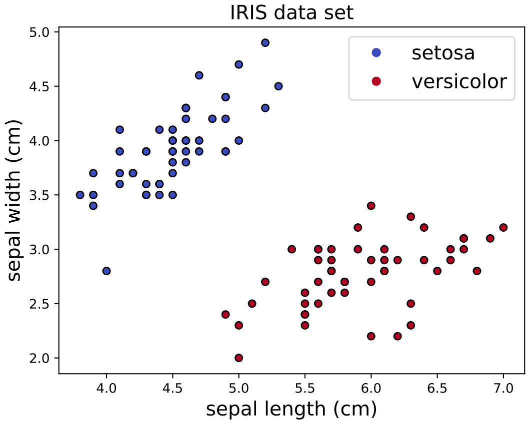
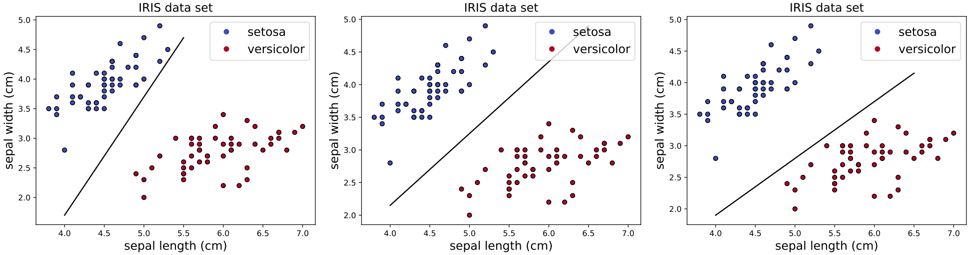


Figure: Samples of the IRIS data set [Fisher, 1936]

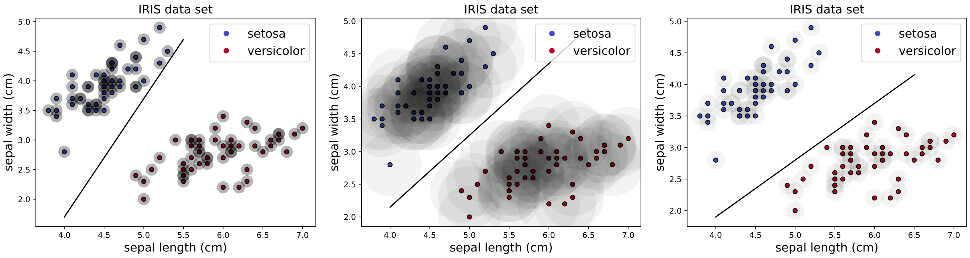
Linear Classification with a Hyperplane

- A priori, which hyperplane generalizes best?



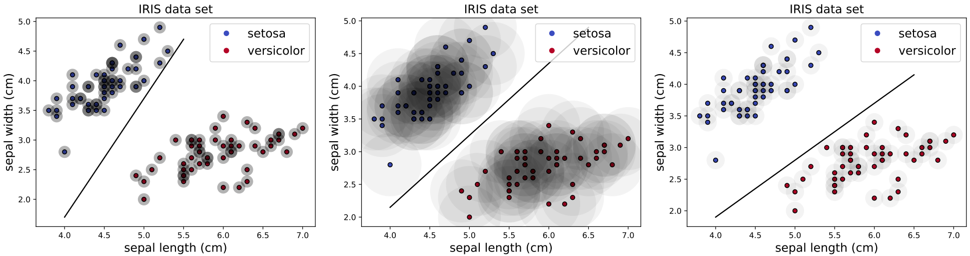
Linear Classification with a Hyperplane

- A priori, which hyperplane generalizes best?
- Which hyperplane is the most robust to gaussian noise in the data set?



Linear Classification with a Hyperplane

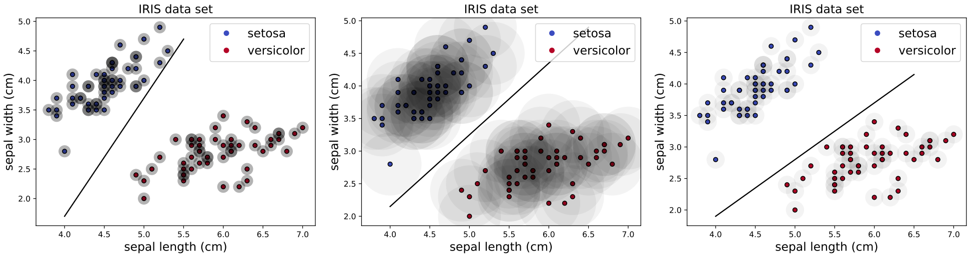
- A priori, which hyperplane generalizes best?
- Which hyperplane is the most robust to gaussian noise in the data set?



- An infinity of hyperplanes perfectly separates the data.

Linear Classification with a Hyperplane

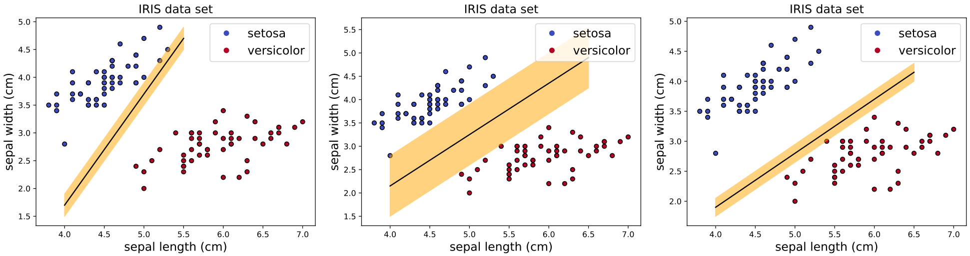
- A priori, which hyperplane generalizes best?
- Which hyperplane is the most robust to gaussian noise in the data set?



- An infinity of hyperplanes perfectly separates the data.
- We need **inductive biases**: assumptions about the data / model / training algorithm in order to select a way to generalize from the training data [Mitchell, 1980, Zhao et al., 2018].

Linear Classification with a Hyperplane

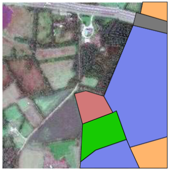
- A priori, which hyperplane generalizes best?
- Which hyperplane is the most robust to gaussian noise in the data set?



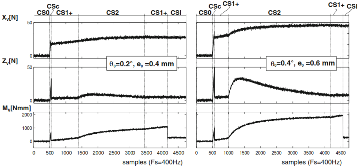
- An infinity of hyperplanes perfectly separates the data.
- We need **inductive biases**: assumptions about the data / model / training algorithm in order to select a way to generalize from the training data [Mitchell, 1980, Zhao et al., 2018].
- **Support Vector Machines**: select the hyperplane the maximizes the margin.

A few applications of SVM

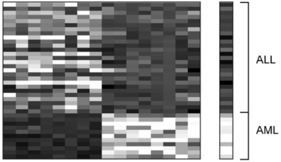
(a) Medical image classification,
e.g. [Camlica et al., 2015]



(b) Hyperspectral image segmentation,
e.g. [Mercier and Lennon, 2003]



(c) Contact states recognition in robotics,
e.g. [Jakovljevic et al., 2012]



(d) Gene selection, *e.g.* [Guyon et al., 2002]

Objectives of the lecture

- Develop a geometric intuition about classification problems,
- Understand the fundamentals of **supervised max-margin** classification models,
- Understand the need for model **regularization**,
- Understand the need for **inductive biases** in machine learning.

Bibliography

Those slides are inspired by numerous great resources including:

- Andrew Ng's lecture notes: `https://see.stanford.edu/materials/aimlcs229/cs229-notes3.pdf`
- Pattern Recognition and Machine Learning, Christopher M. Bishop, `https://www.cs.uoi.gr/~arly/courses/ml/tmp/Bishop_book.pdf`
- Kévin Bailly's lectures on SVM, `https://sites.google.com/view/bailly/`

Overview

- Linearly separable data
- Non-separable Data
- Non-Linear Classification

Support Vector Machines: Binary Linear Classification

We consider a labeled training data set $\mathcal{D}_{train} = \{(\boldsymbol{x}^{(i)}, y^{(i)}) | i \in \{1, \dots, N\}\}$ where:

$$\forall i \in \{1, \dots, N\}, \begin{cases} \boldsymbol{x}^{(i)} \in \mathcal{X} \subset \mathbb{R}^{D \times 1} \\ y^{(i)} \in \mathcal{Y} = \{-1, 1\} \end{cases}$$

Assuming that the data is linearly separable, how can we define a function $f_\theta : \mathcal{X} \longrightarrow \mathcal{Y}$, parameterized by θ, such that:

$$\forall i \in \{1, \dots, N\}, f_\theta(\boldsymbol{x}^{(i)}) = y^{(i)} \quad \text{for an optimal } \theta?$$

Support Vector Machines: Binary Linear Classification

We consider a labeled training data set $\mathcal{D}_{train} = \{(\boldsymbol{x}^{(i)}, y^{(i)}) | i \in \{1, \ldots, N\}\}$ where:

$$\forall i \in \{1, \ldots, N\}, \begin{cases} \boldsymbol{x}^{(i)} \in \mathcal{X} \subset \mathbb{R}^{D \times 1} \\ y^{(i)} \in \mathcal{Y} = \{-1, 1\} \end{cases}$$

Assuming that the data is linearly separable, how can we define a function $f_\theta : \mathcal{X} \longrightarrow \mathcal{Y}$, parameterized by θ, such that:

$$\forall i \in \{1, \ldots, N\}, f_\theta(\boldsymbol{x}^{(i)}) = y^{(i)} \quad \text{for an optimal } \theta?$$

Linear Classifier

Let's denote $\theta = \begin{bmatrix} \boldsymbol{w} \\ b \end{bmatrix}$ where $\boldsymbol{w} \in \mathbb{R}^{D \times 1}$ and $b \in \mathbb{R}$. Besides, let's define the hyperplane $\mathcal{H} : \boldsymbol{w}^T \boldsymbol{x} + b = 0, \boldsymbol{x} \in \mathbb{R}^{D \times 1}$. We define a linear classifier f_θ as follows:

$$f_\theta : \mathcal{X} \longrightarrow \mathcal{Y}$$

$$f_\theta(\boldsymbol{x}) = \sigma(\boldsymbol{w}^T \boldsymbol{x} + b) \quad \text{where } \forall v \in \mathbb{R}, \begin{cases} \sigma(v) = 1 \text{ if } v \geqslant 0 \\ \sigma(v) = -1 \text{ otherwise} \end{cases}$$

Support Vector Machines: Margin Definition

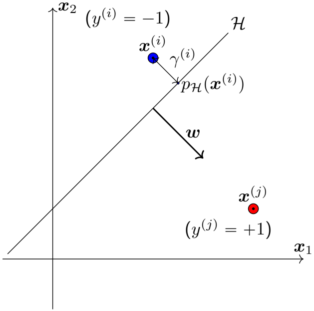
We aim to find the model parameters that maximize the margin between the hyperplane $\mathcal{H}$ and its closest training data points.

- First, let us show that the distance $\gamma^{(i)} := d(\boldsymbol{x}^{(i)}, \mathcal{H}) = d(\boldsymbol{x}^{(i)}, p_{\mathcal{H}}(\boldsymbol{x}^{(i)}))$ can be expressed as follows:

$$\gamma^{(i)} = \frac{1}{\|\boldsymbol{w}\|} y^{(i)}(\boldsymbol{w}^T \boldsymbol{x}^{(i)} + b)$$

Support Vector Machines: Margin Definition

We aim to find the model parameters that maximize the margin between the hyperplane $\mathcal{H}$ and its closest training data points.



- First, let us show that the distance $\gamma^{(i)} := d(\boldsymbol{x}^{(i)}, \mathcal{H}) = d(\boldsymbol{x}^{(i)}, p_{\mathcal{H}}(\boldsymbol{x}^{(i)}))$ can be expressed as follows:

$$\gamma^{(i)} = \frac{1}{\|\boldsymbol{w}\|} y^{(i)} (\boldsymbol{w}^T \boldsymbol{x}^{(i)} + b)$$

Support Vector Machines: Functional and Geometric Margins

- Let's define the **geometric margin** γ:

$$\gamma = \min_{i \in \{1,\ldots,N\}} \gamma^{(i)} = \min_{i \in \{1,\ldots,N\}} \frac{1}{\|\boldsymbol{w}\|} \hat{\gamma}^{(i)} = \frac{1}{\|\boldsymbol{w}\|} \hat{\gamma}$$

Support Vector Machines: Functional and Geometric Margins

- Let's define the **geometric margin** γ:

$$\gamma = \min_{i \in \{1,\dots,N\}} \gamma^{(i)} = \min_{i \in \{1,\dots,N\}} \frac{1}{\|\boldsymbol{w}\|} \hat{\gamma}^{(i)} = \frac{1}{\|\boldsymbol{w}\|} \hat{\gamma}$$

where $\hat{\gamma}$, called the **functional margin**, is defined as follows:

$$\hat{\gamma} = \min_{i \in \{1,\dots,N\}} \hat{\gamma}^{(i)} = \min_{i \in \{1,\dots,N\}} y^{(i)}(\boldsymbol{w}^T \boldsymbol{x}^{(i)} + b)$$

Support Vector Machines: Functional and Geometric Margins

- Let's define the **geometric margin** γ:

$$\gamma = \min_{i \in \{1,\ldots,N\}} \gamma^{(i)} = \min_{i \in \{1,\ldots,N\}} \frac{1}{\|\boldsymbol{w}\|} \hat{\gamma}^{(i)} = \frac{1}{\|\boldsymbol{w}\|} \hat{\gamma}$$

where $\hat{\gamma}$, called the **functional margin**, is defined as follows:

$$\hat{\gamma} = \min_{i \in \{1,\ldots,N\}} \hat{\gamma}^{(i)} = \min_{i \in \{1,\ldots,N\}} y^{(i)}(\boldsymbol{w}^T \boldsymbol{x}^{(i)} + b)$$

What does the functional margin represent? Is it a good metric for model uncertainty? Should we maximize the functional margin or the geometric margin?

Support Vector Machines: Functional and Geometric Margins

- Let's define the **geometric margin** γ:

$$\gamma = \min_{i\in\{1,\dots,N\}} \gamma^{(i)} = \min_{i\in\{1,\dots,N\}} \frac{1}{\|\boldsymbol{w}\|}\hat{\gamma}^{(i)} = \frac{1}{\|\boldsymbol{w}\|}\hat{\gamma}$$

where $\hat{\gamma}$, called the **functional margin**, is defined as follows:

$$\hat{\gamma} = \min_{i\in\{1,\dots,N\}} \hat{\gamma}^{(i)} = \min_{i\in\{1,\dots,N\}} y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b)$$

What does the functional margin represent? Is it a good metric for model uncertainty? Should we maximize the functional margin or the geometric margin?

Notice that the functional margin can be arbitrarily increased by a factor $\alpha > 0$ by substituting $\boldsymbol{w}$ and b with $\alpha\boldsymbol{w}$ and αb while f_θ is kept unchanged:

$$\begin{cases} f_{\theta'}(\boldsymbol{x}) & = \sigma(\boldsymbol{w}'^T\boldsymbol{x} + b') = \sigma(\alpha(\boldsymbol{w}^T\boldsymbol{x} + b)) = \sigma(\boldsymbol{w}^T + b) = f_\theta(\boldsymbol{x}) \\ \hat{\gamma}'^{(i)} & = \alpha\hat{\gamma}^{(i)} \end{cases}$$

Support Vector Machines: Objective Function

We aim to maximize the geometric margin: $\max_{\boldsymbol{w},b} \ \min_{i\in\{1,\dots,N\}} \frac{1}{\|\boldsymbol{w}\|}\hat{\gamma}^{(i)}$.

▸ *Can we turn this optimization problem into a quadratic optimization problem with linear constraints, such that standard solvers could solve it?*

SVM as a Quadratic Optimization Problem

$$\begin{aligned} \min_{\boldsymbol{\theta}\in\mathbb{R}^D} \quad & \frac{1}{2}\boldsymbol{\theta}^T Q\boldsymbol{\theta} + \boldsymbol{p}^T\boldsymbol{\theta} \\ \text{s.t.} \quad & A\boldsymbol{\theta} \geqslant \boldsymbol{c} \end{aligned}$$

Support Vector Machines: Objective Function

We aim to maximize the geometric margin: $\max_{\boldsymbol{w},b} \ \min_{i\in\{1,\dots,N\}} \frac{1}{\|\boldsymbol{w}\|}\hat{\gamma}^{(i)}$.

▸ *Can we turn this optimization problem into a quadratic optimization problem with linear constraints, such that standard solvers could solve it?*

SVM as a Quadratic Optimization Problem

$$\begin{aligned} \min_{\boldsymbol{\theta}\in\mathbb{R}^D} \quad & \frac{1}{2}\boldsymbol{\theta}^T Q\boldsymbol{\theta} + \boldsymbol{p}^T\boldsymbol{\theta} \\ \text{s.t.} \quad & A\boldsymbol{\theta} \geqslant \boldsymbol{c} \end{aligned}$$

SVM Formulation

Since f_θ is invariant to the scale of the functional margin, we can arbitrarily set $\hat{\gamma} = 1$:

$$\begin{aligned} \max_{\boldsymbol{w},b} \ & \frac{1}{\|\boldsymbol{w}\|} \\ \text{s.t.} \ & \min_{i\in\{1,\dots,N\}} \underbrace{y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b)}_{\hat{\gamma}^{(i)}} = 1 \end{aligned}$$

Support Vector Machines: Objective Function

We aim to maximize the geometric margin: $\max_{\boldsymbol{w},b} \; \min_{i\in\{1,\dots,N\}} \frac{1}{\|\boldsymbol{w}\|}\hat{\gamma}^{(i)}$.

▸ *Can we turn this optimization problem into a quadratic optimization problem with linear constraints, such that standard solvers could solve it?*

SVM as a Quadratic Optimization Problem

$$\begin{aligned} \min_{\boldsymbol{\theta}\in\mathbb{R}^D} \quad & \frac{1}{2}\boldsymbol{\theta}^T Q\boldsymbol{\theta} + \boldsymbol{p}^T\boldsymbol{\theta} \\ \text{s.t.} \quad & A\boldsymbol{\theta} \geqslant \boldsymbol{c} \end{aligned}$$

SVM Formulation

Since f_θ is invariant to the scale of the functional margin, we can arbitrarily set $\hat{\gamma} = 1$:

$$\begin{aligned} \min_{\boldsymbol{w},b} \; & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} \\ \text{s.t.} \; & \min_{i\in\{1,\dots,N\}} \underbrace{y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b)}_{\hat{\gamma}^{(i)}} = 1 \end{aligned}$$

Support Vector Machines: Objective Function

We aim to maximize the geometric margin: $\max_{\boldsymbol{w},b} \ \min_{i\in\{1,\ldots,N\}} \frac{1}{\|\boldsymbol{w}\|}\hat{\gamma}^{(i)}$.

- *Can we turn this optimization problem into a quadratic optimization problem with linear constraints, such that standard solvers could solve it?*

SVM as a Quadratic Optimization Problem

$$\begin{aligned} \min_{\boldsymbol{\theta}\in\mathbb{R}^D} \quad & \frac{1}{2}\boldsymbol{\theta}^T Q\boldsymbol{\theta} + \boldsymbol{p}^T\boldsymbol{\theta} \\ \text{s.t.} \quad & A\boldsymbol{\theta} \geqslant \boldsymbol{c} \end{aligned}$$

SVM Formulation

Since f_θ is invariant to the scale of the functional margin, we can arbitrarily set $\hat{\gamma} = 1$:

$$\begin{aligned} \min_{\boldsymbol{w},b} \ & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} \\ \text{s.t.} \ & \forall i \in \{1,\ldots,N\}, y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b) \geqslant 1 \end{aligned}$$

SVM as a Quadratic Optimization Problem

- *Write the corresponding values of* $\boldsymbol{\theta}$, Q, $\boldsymbol{p}$, A *and* $\boldsymbol{c}$.

SVM as a Quadratic Optimization Problem

- *Write the corresponding values of $\boldsymbol{\theta}$, Q, $\boldsymbol{p}$, A and $\boldsymbol{c}$.*

$$\boldsymbol{\theta} = \begin{bmatrix} b \\ \boldsymbol{w} \end{bmatrix}$$

$$Q = \begin{bmatrix} 0 & \mathbf{0}_{1\times D} \\ \mathbf{0}_{D\times 1} & \boldsymbol{I}_D \end{bmatrix}$$

$$\boldsymbol{p} = \mathbf{0}_{(1+D)\times 1}$$

$$A = \begin{bmatrix} \boldsymbol{a}_1^T \\ \vdots \\ \boldsymbol{a}_N^T \end{bmatrix} \text{ with:}$$

$$\boldsymbol{a}_i^T = \begin{bmatrix} y^{(i)} & y^{(i)}\boldsymbol{x}^{(i)T} \end{bmatrix}$$

$$\boldsymbol{c} = \mathbf{1}_{N\times 1}$$

Support Vector Machines: Example

- Let's consider the following data set:

$$\mathcal{D}_{train} = \{([0\,0]^T, -1), ([2\,2]^T, -1), ([2\,0]^T, 1), ([3\,0]^T, 1)\}$$

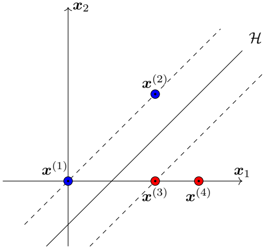
1. Write the linear constraints for this data set,
2. Compute the optimal hyperplane parameters,
3. Compute the optimal margin,
4. Draw the hyperplane, margin, samples. Which samples are on the margin?

Support Vector Machines: Example

- Let's consider the following data set:

$$\mathcal{D}_{train} = \{([0\,0]^T, -1), ([2\,2]^T, -1), ([2\,0]^T, 1), ([3\,0]^T, 1)\}$$

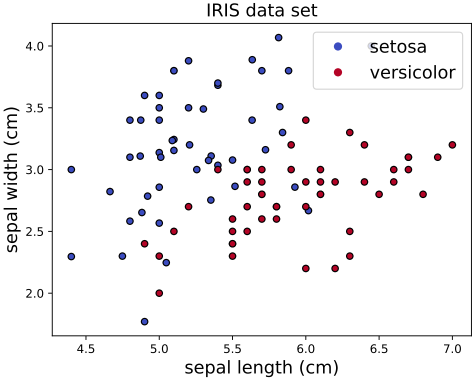
1. Write the linear constraints for this data set,
2. Compute the optimal hyperplane parameters,
3. Compute the optimal margin,
4. Draw the hyperplane, margin, samples. Which samples are on the margin?



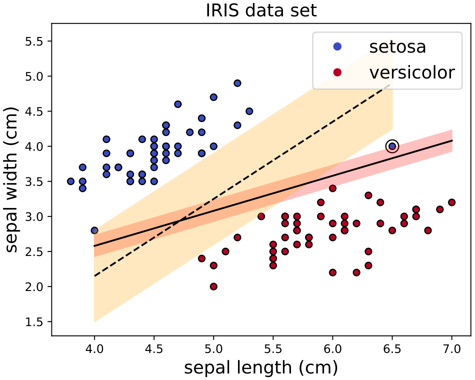
What happens if we remove $\boldsymbol{x}^{(4)}$ from the data set? $\boldsymbol{x}^{(3)}$?

- Linearly separable data
- Non-separable Data
- Non-Linear Classification

Non-separable data: examples of data with noise and outliers



(a) Samples of a noisy version of the IRIS data set



(b) Samples of the IRIS data set with an outlier

Regularization of the SVM problem

- With the current formulation of the SVM problem, the optimization algorithm will not converge.

SVM Formulation

$$\min_{\boldsymbol{w},b} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}$$

$$\text{s.t. } \forall i \in \{1,\dots,N\}, y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b) \geqslant 1$$

- *Can we modify the optimization problem to handle non-separable data?*

Regularization of the SVM problem

- With the current formulation of the SVM problem, the optimization algorithm will not converge.

SVM Formulation

$$\min_{\boldsymbol{w},b} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}$$
$$\text{s.t. } \forall i \in \{1,\dots,N\}, y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b) \geqslant 1$$

- *Can we modify the optimization problem to handle non-separable data?*

Relaxed SVM Formulation

$$\min_{\boldsymbol{w},b,\xi} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\xi_i$$
$$\text{s.t. } \forall i \in \{1,\dots,N\}, y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b) \geqslant 1-\xi_i$$
$$\forall i \in \{1,\dots,N\}, \xi_i \geqslant 0$$

Regularization of the SVM problem

Relaxed SVM Formulation

$$\min_{\boldsymbol{w},b,\xi} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\xi_i$$

$$\text{s.t.} \quad \forall i \in \{1,\ldots,N\}, y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b) \geqslant 1 - \xi_i$$

$$\forall i \in \{1,\ldots,N\}, \xi_i \geqslant 0$$

Regularization of the SVM problem

Relaxed SVM Formulation

$$\min_{\boldsymbol{w},b,\xi} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\xi_i$$

$$\text{s.t. } \forall i \in \{1,\dots,N\}, y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b) \geqslant 1 - \xi_i$$

$$\forall i \in \{1,\dots,N\}, \xi_i \geqslant 0$$

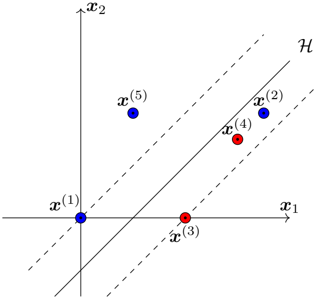
- *What happens if* $C = 0$*?* $C = \infty$*?*
- *Give the formulation of the relaxed problem to match the quadratic formulation:*

$$\min_{\boldsymbol{\theta}\in\mathbb{R}^D} \frac{1}{2}\boldsymbol{\theta}^T Q\boldsymbol{\theta} + \boldsymbol{p}^T\boldsymbol{\theta}$$

$$\text{s.t. } A\boldsymbol{\theta} \geqslant \boldsymbol{c}$$

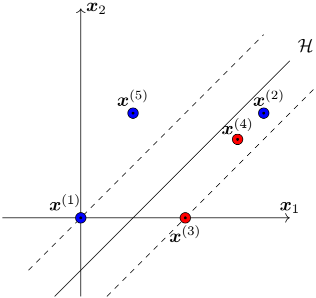
Regularization of the SVM problem

- *What are the values of* $\xi_i, i \in \{1, \ldots, 5\}$*?*



Regularization of the SVM problem

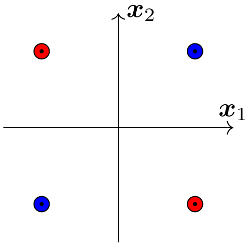
- *What are the values of $\xi_i, i \in \{1, \ldots, 5\}$?*



$$\xi_1 = 0, \quad \xi_2 > 1, \quad \xi_3 = 0, \quad 0 < \xi_4 < 1, \quad \xi_5 = 0$$

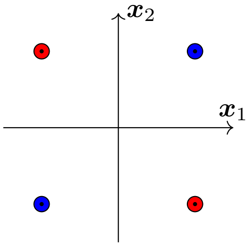
- Linearly separable data
- Non-separable Data
- Non-Linear Classification

Non-Linear Classification: the XOR example



Can you find a separating hyperplane?

Non-Linear Classification: the XOR example

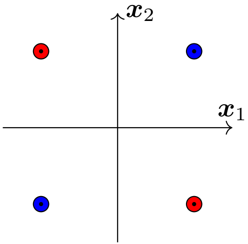


$$\phi : \mathbb{R}^2 \longrightarrow ?$$
$$\phi(\boldsymbol{x}) = ?$$

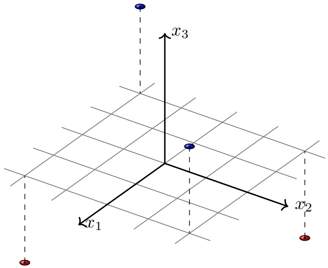
Can you find a separating hyperplane?

- *Could we map the data in a higher dimensional space where it would be linearly separable?*

Non-Linear Classification: the XOR example



$$\phi : \mathbb{R}^2 \longrightarrow \mathbb{R}^3$$
$$\phi(\boldsymbol{x}) = [\boldsymbol{x}_1 \ \ \boldsymbol{x}_2 \ \ \boldsymbol{x}_1\boldsymbol{x}_2]^T$$



Can you find a separating hyperplane?

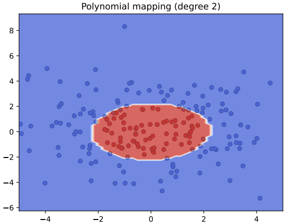
- *Could we map the data in a higher dimensional space where it would be linearly separable?*

Non-Linear Mapping before SVM

- So, when data is not linearly separable, we can find a mapping $\phi : \mathcal{X} \longrightarrow \mathcal{Z}$ from the input space $\mathcal{X}$ to a feature space $\mathcal{Z}$ of higher dimension, where features are linearly separable.
- Then, we apply the SVM on $\boldsymbol{z} = \phi(\boldsymbol{x})$ instead of $\boldsymbol{x}$.

Another example is a polynomial mapping:

$$\phi : \mathbb{R}^2 \longrightarrow \mathbb{R}^5$$
$$\phi(\boldsymbol{x}) = [\boldsymbol{x}_1 \;\; \boldsymbol{x}_2 \;\; \boldsymbol{x}_1\boldsymbol{x}_2 \;\; \boldsymbol{x}_1^2 \;\; \boldsymbol{x}_2^2]^T$$



- However, computing $\phi(\boldsymbol{x})$ can be computationally intensive.
- In order to circumvent this issue, we need to consider another formulation of the SVM optimization problem and to digress for a moment on Lagrange duality!

Lagrangian Duality

Primal Optimization Problem

The SVM optimization problem can be put in the following form, called the primal problem:

$$\min_{\boldsymbol{\theta}} \; f(\boldsymbol{\theta})$$
$$\text{s.t. } g_i(\boldsymbol{\theta}) \leqslant 0, i \in \{1, \dots, N\}$$

Lagrangian of the Primal Optimization Problem

$$\mathcal{L}(\boldsymbol{\theta}, \alpha) = f(\boldsymbol{\theta}) + \sum_{i=1}^{N} \alpha_i g_i(\boldsymbol{\theta})$$

- *Prove that the following optimization problem has the same solution than the primal problem:*

$$\min_{\boldsymbol{\theta}} \max_{\alpha \geqslant 0} \mathcal{L}(\boldsymbol{\theta}, \alpha)$$

Lagrangian Duality

Primal Optimization problem

$$\min_{\boldsymbol{\theta}} \max_{\alpha \geqslant 0} \mathcal{L}(\boldsymbol{\theta}, \alpha)$$

Dual Optimization problem

$$\max_{\alpha \geqslant 0} \min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \alpha)$$

Strong Duality

If the optimization problem is convex, then the solutions of the **primal** and **dual** problems are equal:

$$\min_{\boldsymbol{\theta}} \max_{\alpha \geqslant 0} \mathcal{L}(\boldsymbol{\theta}, \alpha) = \max_{\alpha \geqslant 0} \min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \alpha)$$

Karush-Kuhn-Tucker (KKT) Conditions

$(\boldsymbol{\theta}^*, \alpha^*)$ are the optimal solutions of the primal / dual problem if and only if they satisfy the **Karush-Kuhn-Tucker conditions**:

$$\begin{aligned}
\textbf{Stationarity} \quad & \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \alpha)|_{\boldsymbol{\theta}=\boldsymbol{\theta}^*, \alpha=\alpha^*} = 0 && (1)\\
\textbf{Primal feasibility} \quad & \forall i \in \{1, \dots, N\}, g_i(\boldsymbol{\theta}^*) \leqslant 0 && (2)\\
\textbf{Dual feasibility} \quad & \forall i \in \{1, \dots, N\}, \alpha_i^* \geqslant 0 && (3)\\
\textbf{Complementary slackness} \quad & \forall i \in \{1, \dots, N\}, \alpha_i^* g_i(\boldsymbol{\theta}^*) = 0 && (4)
\end{aligned}$$

Condition (4) states that if $\alpha_i^* > 0$, then $g_i(\boldsymbol{\theta}^*) = 0$, meaning that the constraint $g_i(\boldsymbol{\theta}^*) \leqslant 0$ is active.

Why do we care about the Dual Problem?

- *Write the Lagrangian of the SVM optimization problem:*

Why do we care about the Dual Problem?

- *Write the Lagrangian of the SVM optimization problem:*

$$\mathcal{L}(\boldsymbol{w}, b, \alpha) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} - \sum_{i=1}^{N} \alpha_i \big(y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b) - 1\big)$$

Why do we care about the Dual Problem?

- *Write the Lagrangian of the SVM optimization problem:*

$$\mathcal{L}(\boldsymbol{w}, b, \alpha) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} - \sum_{i=1}^{N} \alpha_i \big(y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b) - 1\big)$$

- *Find optimal conditions for $\boldsymbol{w}^*$ and b^* (note that the Lagrangian is convex):*

Why do we care about the Dual Problem?

- *Write the Lagrangian of the SVM optimization problem:*

$$\mathcal{L}(\boldsymbol{w}, b, \alpha) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} - \sum_{i=1}^{N} \alpha_i \big(y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b) - 1\big)$$

- *Find optimal conditions for $\boldsymbol{w}^*$ and b^* (note that the Lagrangian is convex):*

$$\nabla_{\boldsymbol{w}}\mathcal{L}(\boldsymbol{w}, b, \alpha) = 0 \iff w - \sum_{i=1}^{N} \alpha_i y^{(i)}\boldsymbol{x}^{(i)} = 0 \iff \boldsymbol{w} = \sum_{i=1}^{N} \alpha_i y^{(i)}\boldsymbol{x}^{(i)}$$

Why do we care about the Dual Problem?

- *Write the Lagrangian of the SVM optimization problem:*

$$\mathcal{L}(\boldsymbol{w}, b, \alpha) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} - \sum_{i=1}^{N} \alpha_i \big(y^{(i)}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b) - 1\big)$$

- *Find optimal conditions for $\boldsymbol{w}^*$ and b^* (note that the Lagrangian is convex):*

$$\nabla_{\boldsymbol{w}}\mathcal{L}(\boldsymbol{w}, b, \alpha) = 0 \iff w - \sum_{i=1}^{N} \alpha_i y^{(i)}\boldsymbol{x}^{(i)} = 0 \iff \boldsymbol{w} = \sum_{i=1}^{N} \alpha_i y^{(i)}\boldsymbol{x}^{(i)}$$

$$\nabla_b\mathcal{L}(\boldsymbol{w}, b, \alpha) = 0 \iff \sum_{i=1}^{N} \alpha_i y^{(i)} = 0$$

Why do we care about the Dual Problem?

Hence, $\mathcal{L}(\boldsymbol{w}^*, b^*, \alpha) =$

Why do we care about the Dual Problem?

Hence, $$\mathcal{L}(\boldsymbol{w}^*, b^*, \alpha) = \frac{1}{2}\Big(\sum_{i=1}^{N} \alpha_i y^{(i)} \boldsymbol{x}^{(i)}\Big)^T \Big(\sum_{j=1}^{N} \alpha_j y^{(j)} \boldsymbol{x}^{(j)}\Big) - \sum_{i=1}^{N} \alpha_i y^{(i)} \Big(\sum_{j=1}^{N} \alpha_j y^{(j)} \boldsymbol{x}^{(j)}\Big)^T \boldsymbol{x}^{(i)} - b\sum_{i=1}^{N} \alpha_i y^{(i)} + \sum_{i=1}^{N} \alpha_i$$

Why do we care about the Dual Problem?

Hence,

$$\begin{aligned}\mathcal{L}(\boldsymbol{w}^*, b^*, \alpha) &= \frac{1}{2}\Big(\sum_{i=1}^{N} \alpha_i y^{(i)} \boldsymbol{x}^{(i)}\Big)^T \Big(\sum_{j=1}^{N} \alpha_j y^{(j)} \boldsymbol{x}^{(j)}\Big) - \sum_{i=1}^{N} \alpha_i y^{(i)} \Big(\sum_{j=1}^{N} \alpha_j y^{(j)} \boldsymbol{x}^{(j)}\Big)^T \boldsymbol{x}^{(i)} \\ &- b \sum_{i=1}^{N} \alpha_i y^{(i)} + \sum_{i=1}^{N} \alpha_i \\ &= \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y^{(i)} y^{(j)} \boldsymbol{x}^{(i)T} \boldsymbol{x}^{(j)}\end{aligned}$$

Why do we care about the Dual Problem?

Hence, $\mathcal{L}(\boldsymbol{w}^*, b^*, \alpha) = \frac{1}{2}\Big(\sum_{i=1}^{N} \alpha_i y^{(i)} \boldsymbol{x}^{(i)}\Big)^T \Big(\sum_{j=1}^{N} \alpha_j y^{(j)} \boldsymbol{x}^{(j)}\Big) - \sum_{i=1}^{N} \alpha_i y^{(i)} \Big(\sum_{j=1}^{N} \alpha_j y^{(j)} \boldsymbol{x}^{(j)}\Big)^T \boldsymbol{x}^{(i)}$

$$- b \sum_{i=1}^{N} \alpha_i y^{(i)} + \sum_{i=1}^{N} \alpha_i$$

$$= \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y^{(i)} y^{(j)} \boldsymbol{x}^{(i)T} \boldsymbol{x}^{(j)}$$

SVM Dual Problem Formulation

$$\max_{\alpha} \; \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y^{(i)} y^{(j)} \boldsymbol{x}^{(i)T} \boldsymbol{x}^{(j)}$$

s.t. $\forall i \in \{1, \ldots, N\}, \alpha_i \geqslant 0,$

$$\sum_{i=1}^{N} \alpha_i y^{(i)} = 0$$

Why do we care about the Dual Problem?

- We can solve the SVM dual problem with a standard solver for quadratic optimization with linear constraints in order to find α^*.
- It follows that the optimal classifier is defined by:

$$\boldsymbol{w}^* = \sum_{i=1}^{N} \alpha_i^* y^{(i)} \boldsymbol{x}^{(i)} \qquad b^* = ?$$

Why do we care about the Dual Problem?

- We can solve the SVM dual problem with a standard solver for quadratic optimization with linear constraints in order to find α^*.
- It follows that the optimal classifier is defined by:

$$\boldsymbol{w}^* = \sum_{i=1}^{N} \alpha_i^* y^{(i)} \boldsymbol{x}^{(i)} \qquad b^* = ?$$

- Recall that, from the KKT conditions, we have that if $\alpha_i^* > 0$, then $y^{(i)}(\boldsymbol{w}^{*T}\boldsymbol{x}^{(i)} + b^*) = 1$. In other words, the only samples $\boldsymbol{x}^{(i)}$ that contribute to $\boldsymbol{w}^*$ are the support vectors lying on the margin.

Why do we care about the Dual Problem?

- We can solve the SVM dual problem with a standard solver for quadratic optimization with linear constraints in order to find α^*.
- It follows that the optimal classifier is defined by:

$$\boldsymbol{w}^* = \sum_{i=1}^{N} \alpha_i^* y^{(i)} \boldsymbol{x}^{(i)} \qquad b^* = ?$$

- Recall that, from the KKT conditions, we have that if $\alpha_i^* > 0$, then $y^{(i)}(\boldsymbol{w}^{*T}\boldsymbol{x}^{(i)} + b^*) = 1$. In other words, the only samples $\boldsymbol{x}^{(i)}$ that contribute to $\boldsymbol{w}^*$ are the support vectors lying on the margin.
- For a test sample $\boldsymbol{x}^*$, the prediction is given by:

$$\hat{y} = \sigma\Big(\sum_{\alpha_i^* > 0} \alpha_i^* y^{(i)} \boldsymbol{x}^{(i)T} \boldsymbol{x}^* + b^* \Big)$$

The Kernel Trick

- For non-linear classification problem, the prediction becomes:

$$\hat{y} = \sigma\Big(\sum_{\alpha_i^* > 0} \alpha_i^* y^{(i)} \phi(\boldsymbol{x}^{(i)})^T \phi(\boldsymbol{x}^*) + b^* \Big)$$

The Kernel Trick

- For non-linear classification problem, the prediction becomes:

$$\hat{y} = \sigma\Big(\sum_{\alpha_i^* > 0} \alpha_i^* y^{(i)} K_\phi(\boldsymbol{x}^{(i)}, \boldsymbol{x}^*) + b^* \Big)$$

Kernel Trick

We do not need to explicitly compute the mapping $\phi(\boldsymbol{x})$, that can be costly, if we know a formulation of kernel K_ϕ that does not depend on ϕ:

$$K_\phi(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^T \phi(\boldsymbol{x}')$$

The Kernel Trick

- For non-linear classification problem, the prediction becomes:

$$\hat{y} = \sigma\Big(\sum_{\alpha_i^* > 0} \alpha_i^* y^{(i)} K_\phi(\boldsymbol{x}^{(i)}, \boldsymbol{x}^*) + b^* \Big)$$

Kernel Trick

We do not need to explicitly compute the mapping $\phi(\boldsymbol{x})$, that can be costly, if we know a formulation of kernel K_ϕ that does not depend on ϕ:

$$K_\phi(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^T \phi(\boldsymbol{x}')$$

- *Find the kernel corresponding to the following polynomial mapping:*

$$\phi : \mathbb{R}^2 \longrightarrow \mathbb{R}^5$$

$$\phi(\boldsymbol{x}) = [\boldsymbol{x}_1 \;\; \boldsymbol{x}_2 \;\; \sqrt{2}\boldsymbol{x}_1\boldsymbol{x}_2 \;\; \boldsymbol{x}_1^2 \;\; \boldsymbol{x}_2^2]^T$$

The Kernel Trick

- The Kernel function can be thought as a similarity measure: the more similar $\boldsymbol{x}$ and $\boldsymbol{x}'$, the larger $K_\phi(\boldsymbol{x}, \boldsymbol{x}')$.

- Radial Basis Function (RBF) Kernel:

$$K_\phi(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^T \phi(\boldsymbol{x}') = \exp\Big(-\gamma \|\boldsymbol{x} - \boldsymbol{x}'\|^2\Big), \quad \gamma > 0$$

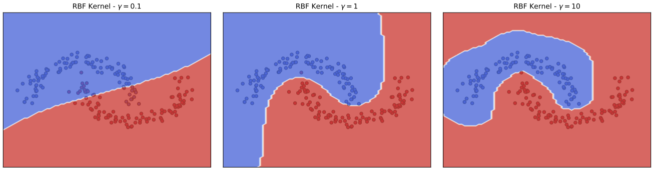


Figure: SVM decision boundaries on the Two Moons data set with various RBF kernels

- We don't need to explicitly know ϕ!

Camlica, Z., Tizhoosh, H. R., and Khalvati, F. (2015).
Medical image classification via svm using lbp features from saliency-based folded data.
In *2015 IEEE 14th international conference on machine learning and applications (ICMLA)*, pages 128–132. IEEE.

Fisher, R. A. (1936).
The use of multiple measurements in taxonomic problems.
Annals of eugenics, 7(2):179–188.

Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002).
Gene selection for cancer classification using support vector machines.
Machine learning, 46:389–422.

Jakovljevic, Z., Petrovic, P. B., and Hodolic, J. (2012).
Contact states recognition in robotic part mating based on support vector machines.
The International Journal of Advanced Manufacturing Technology, 59:377–395.

Mercier, G. and Lennon, M. (2003).
Support vector machines for hyperspectral image classification with spectral-based kernels.
In *IGARSS 2003. 2003 IEEE International Geoscience and Remote Sensing Symposium. Proceedings (IEEE Cat. No. 03CH37477)*, volume 1, pages 288–290. IEEE.

Mitchell, T. M. (1980).
The need for biases in learning generalizations (rutgers computer science tech. rept. cbm-tr-117).
Rutgers University.

Zhao, S., Ren, H., Yuan, A., Song, J., Goodman, N., and Ermon, S. (2018).
Bias and generalization in deep generative models: An empirical study.
Advances in Neural Information Processing Systems, 31.