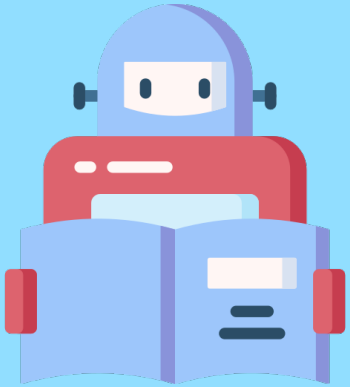
Mathematics for Machine Learning



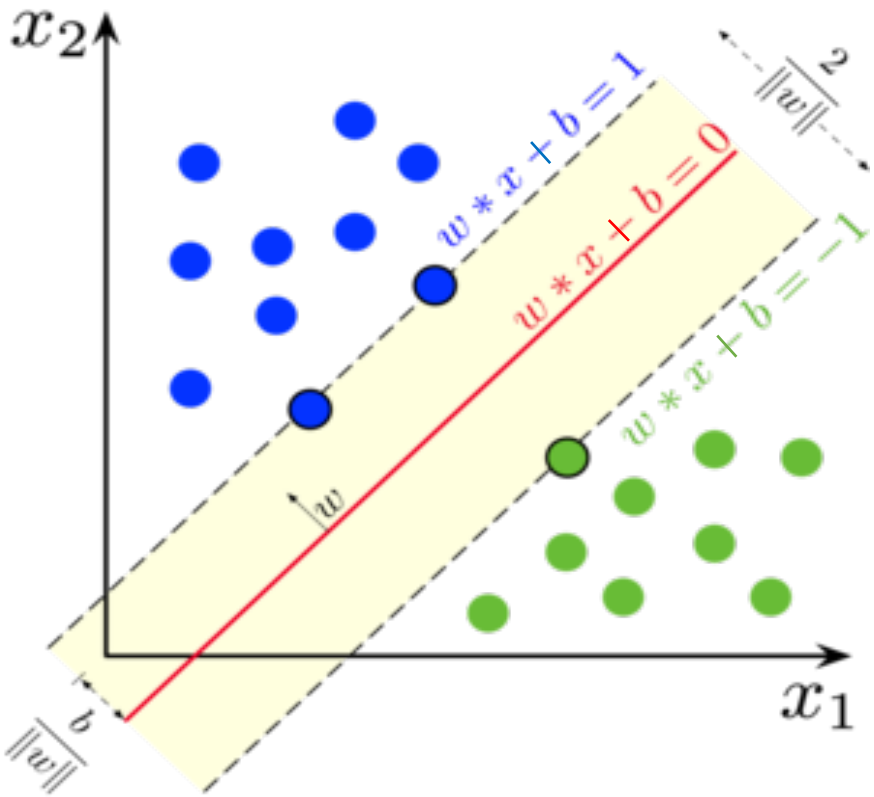
Special Mathematics Lecture
Nagoya University, Fall 2020

Lecture 8
Support vector machines:
Primal & Dual problem, Kernels

https://www.henrikbachmann.com/mml_2020.html

Support vector machines

Feature space: $\mathcal{X} = \mathbb{R}^d$
Label space: $\mathcal{Y} = \{-1, 1\}$
Training set: $\mathcal{T} = \big((x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\big) \in (\mathcal{X} \times \mathcal{Y})^n$



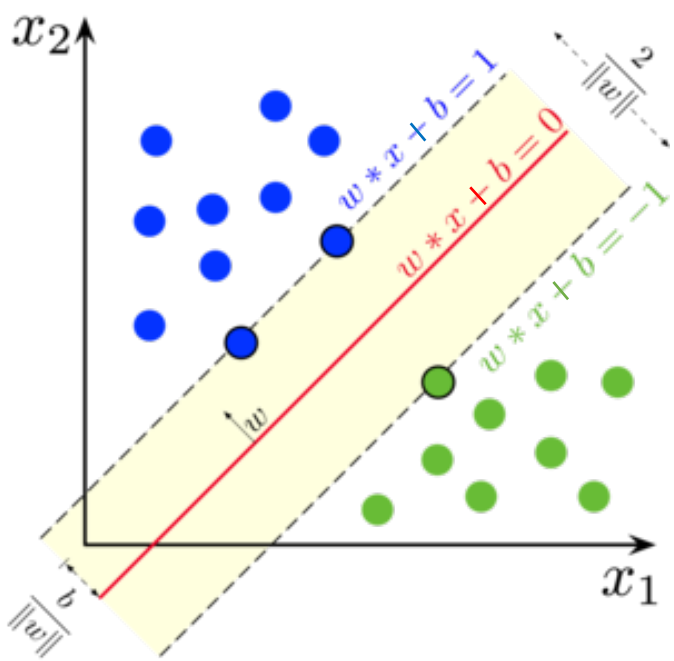
Support vector machines: Primal problem

Feature space: $\mathcal{X} = \mathbb{R}^d$
Label space: $\mathcal{Y} = \{-1, 1\}$
Training set: $\mathcal{T} = \big((x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\big) \in (\mathcal{X} \times \mathcal{Y})^n$

Goal: Find a hyperplane $H(w, b)$, such that

i) $\|w\|$ is minimal.

ii) For $j = 1, \ldots, n$ we have $y^{(j)}(w^T x^{(j)} + b) \geq 1$.



$$H(w, b) = \{x \in \mathbb{R}^d \mid w^T x + b = 0\}$$

Sorry: The "–" is now a "+" in the definition of H.
(Want to be consistent with the literature)

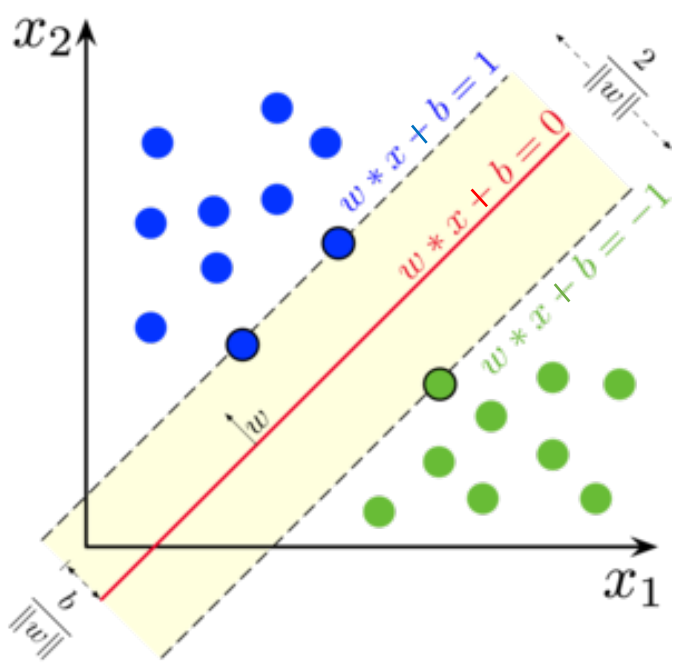
Support vector machines: Primal problem

Feature space: $\mathcal{X} = \mathbb{R}^d$
Label space: $\mathcal{Y} = \{-1, 1\}$
Training set: $\mathcal{T} = \big((x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\big) \in (\mathcal{X} \times \mathcal{Y})^n$

The **primal optimization problem** is to find $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$ such that

i) $f(w) = \frac{1}{2}\|w\|^2$ is minimal.

ii) For $j = 1, \ldots, n$ we have $g_j(w, b) = -y^{(j)}(w^T x^{(j)} + b) + 1 \leq 0.$ (**constraints**)



Rewriting the primal problem

The **primal optimization problem** is to find $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$ such that

i) $f(w) = \frac{1}{2}\|w\|^2$ is minimal.

ii) For $j = 1, \ldots, n$ we have $g_j(w, b) = -y^{(j)}(w^T x^{(j)} + b) + 1 \leq 0.$ (**constraints**)

Define the **Lagrangian** for **Lagrange multipliers** $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n_{\geq 0}$ by

$$\mathcal{L}(w, b, \alpha) = f(w) + \sum_{j=1}^{n} \alpha_j g_j(w, b)\,.$$

Rewriting the primal problem

The **primal optimization problem** is to find $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$ such that

i) $f(w) = \frac{1}{2}\|w\|^2$ is minimal.

ii) For $j = 1, \ldots, n$ we have $g_j(w, b) = -y^{(j)}(w^T x^{(j)} + b) + 1 \leq 0.$ (**constraints**)

Define the **Lagrangian** for **Lagrange multipliers** $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n_{\geq 0}$ by

$$\mathcal{L}(w, b, \alpha) = f(w) + \sum_{j=1}^{n} \alpha_j g_j(w, b)\,.$$

Now define

$$\Theta_P(w, b) = \max_{\alpha_1, \ldots, \alpha_n \geq 0} \mathcal{L}(w, b, \alpha)\,.$$

Observe that this functions satisfies the following

$$\Theta_P(w, b) = \begin{cases} f(w) & \text{, if } g_j(w, b) \leq 0 \text{ for all } j = 1, \ldots, n \qquad \text{(constraints are satisfied)} \\ \infty & \text{, otherwise} \end{cases}.$$

We can therefore focusing on finding w and b such that $\Theta_P(w, b)$ is minimal!

Primal problem vs. Dual problem

Primal problem: Find $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$ which minimize $\Theta_P(w, b)$

$$\min_{w,b} \Theta_P(w, b) = \min_{w,b} \max_{\alpha_1,\ldots,\alpha_n \geq 0} \mathcal{L}(w, b, \alpha)\,.$$

One can show that in our case there exists a unique solution to our problem and that

$$\min_{w,b} \max_{\alpha_1,\ldots,\alpha_n \geq 0} \mathcal{L}(w, b, \alpha) = \max_{\alpha_1,\ldots,\alpha_n \geq 0} \min_{w,b} \mathcal{L}(w, b, \alpha)\,.$$

Primal problem vs. Dual problem

Primal problem: Find $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$ which minimize $\Theta_P(w, b)$

$$\min_{w,b} \Theta_P(w, b) = \min_{w,b} \max_{\alpha_1,\ldots,\alpha_n \geq 0} \mathcal{L}(w, b, \alpha)\,.$$

One can show that in our case there exists a unique solution to our problem and that

$$\min_{w,b} \max_{\alpha_1,\ldots,\alpha_n \geq 0} \mathcal{L}(w, b, \alpha) = \max_{\alpha_1,\ldots,\alpha_n \geq 0} \min_{w,b} \mathcal{L}(w, b, \alpha)\,.$$

Motivated by this we define for $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n_{\geq 0}$

$$\Theta_D(\alpha) = \min_{w,b} \mathcal{L}(w, b, \alpha)\,.$$

Dual problem: Find $\alpha \in \mathbb{R}^n_{\geq 0}$ which maximizes $\Theta_D(\alpha)$.

Dual problem

Dual problem: Find $\alpha \in \mathbb{R}^n_{\geq 0}$ which maximizes $\Theta_D(\alpha)$.

$$\Theta_D(\alpha) = \min_{w,b} \mathcal{L}(w, b, \alpha)\,.$$

Recall that we have

$$\mathcal{L}(w, b, \alpha) = f(w) + \sum_{j=1}^{n} \alpha_j g_j(w, b)$$

with $f(w) = \frac{1}{2}\|w\|^2 = \frac{1}{2}w^T w$ and $g_j(w, b) = -y^{(j)}(w^T x^{(j)} + b) + 1$.

Dual problem

Dual problem: Find $\alpha \in \mathbb{R}^n_{\geq 0}$ which maximizes $\Theta_D(\alpha)$.

$$\Theta_D(\alpha) = \min_{w,b} \mathcal{L}(w, b, \alpha)\,.$$

Recall that we have

$$\mathcal{L}(w, b, \alpha) = f(w) + \sum_{j=1}^{n} \alpha_j g_j(w, b)$$

with $f(w) = \frac{1}{2}\|w\|^2 = \frac{1}{2}w^T w$ and $g_j(w, b) = -y^{(j)}(w^T x^{(j)} + b) + 1$.

We obtain

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}w^T w - \sum_{j=1}^{n} \alpha_j \left(y^{(j)}(w^T x^{(j)} + b) + 1 \right)\,.$$

This is a quadratic function in (w, b) with a single minima, which we can find by setting the gradients zero. (Think of it as $d + 1$ quadratic functions in variables $w_1, \ldots, w_d, b$.)

Dual problem

Want to calculate: $\Theta_D(\alpha) = \min_{w,b} \mathcal{L}(w, b, \alpha)$.

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2} w^T w - \sum_{j=1}^{n} \alpha_j \left(y^{(j)} (w^T x^{(j)} + b) + 1 \right) .$$

The gradients are given by

$$\nabla_w \mathcal{L}(w, b, \alpha) = w - \sum_{j=1}^{n} \alpha_j y^{(j)} x^{(j)}$$

$$\nabla_b \mathcal{L}(w, b, \alpha) = - \sum_{j=1}^{n} \alpha_j y^{(j)} .$$

Dual problem

Want to calculate: $\Theta_D(\alpha) = \min_{w,b} \mathcal{L}(w, b, \alpha)\,.$

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2} w^T w - \sum_{j=1}^{n} \alpha_j \left(y^{(j)} (w^T x^{(j)} + b) + 1 \right)\,.$$

The gradients are given by

$$\nabla_w \mathcal{L}(w, b, \alpha) = w - \sum_{j=1}^{n} \alpha_j y^{(j)} x^{(j)}$$

$$\nabla_b \mathcal{L}(w, b, \alpha) = -\sum_{j=1}^{n} \alpha_j y^{(j)}\,.$$

Finding the minima by setting the gradients zero:

$$w = \sum_{j=1}^{n} \alpha_j y^{(j)} x^{(j)}\,, \quad \text{and} \quad \sum_{j=1}^{n} \alpha_j y^{(j)} = 0\,.$$

Dual problem

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2} w^T w - \sum_{j=1}^{n} \alpha_j \left(y^{(j)} (w^T x^{(j)} + b) + 1 \right)$$

We get the minima (w.r.t w and b) if: $$w = \sum_{j=1}^{n} \alpha_j y^{(j)} x^{(j)}, \quad \text{and} \quad \sum_{j=1}^{n} \alpha_j y^{(j)} = 0 .$$

Plugging this into Θ_D gives

$$\begin{aligned}
\Theta_D(\alpha) &= \min_{w,b} \mathcal{L}(w, b, \alpha) \\
&= \frac{1}{2} \left(\sum_{j=1}^{n} \alpha_j y^{(j)} x^{(j)} \right)^T \left(\sum_{i=1}^{n} \alpha_i y^{(i)} x^{(i} \right) - \sum_{j=1}^{n} \alpha_j \left(y^{(j)} (\left(\sum_{i=1}^{n} \alpha_i y^{(i)} x^{(i)} \right)^T x^{(j)} + b) + 1 \right) \\
&= \sum_{j=1}^{n} \alpha_j - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} \left(x^{(i)} \right)^T x^{(j)} .
\end{aligned}$$

Here we used $\sum_{j=1}^{n} \alpha_j y^{(j)} = 0$ in the last step.

Dual problem

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2} w^T w - \sum_{j=1}^{n} \alpha_j \left(y^{(j)} (w^T x^{(j)} + b) + 1 \right)$$

We get the minima (w.r.t w and b) if: $$w = \sum_{j=1}^{n} \alpha_j y^{(j)} x^{(j)}, \quad \text{and} \quad \sum_{j=1}^{n} \alpha_j y^{(j)} = 0 .$$

Plugging this into Θ_D gives

$$\begin{aligned}
\Theta_D(\alpha) &= \min_{w,b} \mathcal{L}(w, b, \alpha) \\
&= \frac{1}{2} \left(\sum_{j=1}^{n} \alpha_j y^{(j)} x^{(j)} \right)^T \left(\sum_{i=1}^{n} \alpha_i y^{(i)} x^{(i} \right) - \sum_{j=1}^{n} \alpha_j \left(y^{(j)} (\left(\sum_{i=1}^{n} \alpha_i y^{(i)} x^{(i)} \right)^T x^{(j)} + b) + 1 \right) \\
&= \sum_{j=1}^{n} \alpha_j - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} \left(x^{(i)} \right)^T x^{(j)} .
\end{aligned}$$

Here we used $\sum_{j=1}^{n} \alpha_j y^{(j)} = 0$ in the last step.

Dual problem

The **dual optimization problem** is to find $\alpha \in \mathbb{R}^d_{\geq 0}$ such that

i) $\Theta_D(\alpha) = \sum_{j=1}^n \alpha_j - \frac{1}{2}\sum_{i,j=1}^n \alpha_i \alpha_j y^{(i)} y^{(j)} \left(x^{(i)} \bullet x^{(j)}\right)$ is maximal.

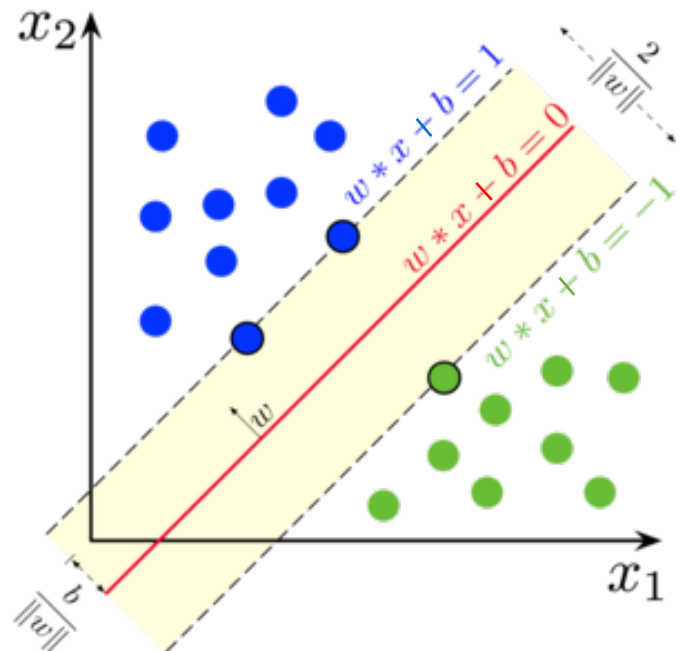
ii) $\sum_{j=1}^n \alpha_j y^{(j)} = 0$

If we can solve this problem then the w can be obtained by

$$w = \sum_{j=1}^{n} \alpha_j y^{(j)} x^{(j)}$$

and the b is given by (Exercise!)

$$b = -\frac{1}{2}\left(\min_{\substack{j=1,\ldots,n \\ y^{(j)}=1}} w^T x^{(j)} + \max_{\substack{j=1,\ldots,n \\ y^{(j)}=-1}} w^T x^{(j)} \right)$$



Dual problem – How to solve?

The **dual optimization problem** is to find $\alpha \in \mathbb{R}^d_{\geq 0}$ such that

i) $\Theta_D(\alpha) = \sum_{j=1}^n \alpha_j - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y^{(i)} y^{(j)} \left(x^{(i)} \bullet x^{(j)}\right)$ is maximal.

ii) $\sum_{j=1}^n \alpha_j y^{(j)} = 0$

- This is an example of a “Quadratic Programming” (QP) problem.
- It can be solved by the SMO (sequential minimal optimization) algorithm.
- There are various implementations for QP Solver in Python.
- Due to lack of time, we will not describe them in detail in this course.

Dual problem - Observation

The **dual optimization problem** is to find $\alpha \in \mathbb{R}^d_{\geq 0}$ such that

i) $\Theta_D(\alpha) = \sum_{j=1}^n \alpha_j - \frac{1}{2}\sum_{i,j=1}^n \alpha_i \alpha_j y^{(i)} y^{(j)} \left(x^{(i)} \bullet x^{(j)}\right)$ is maximal.

ii) $\sum_{j=1}^n \alpha_j y^{(j)} = 0$

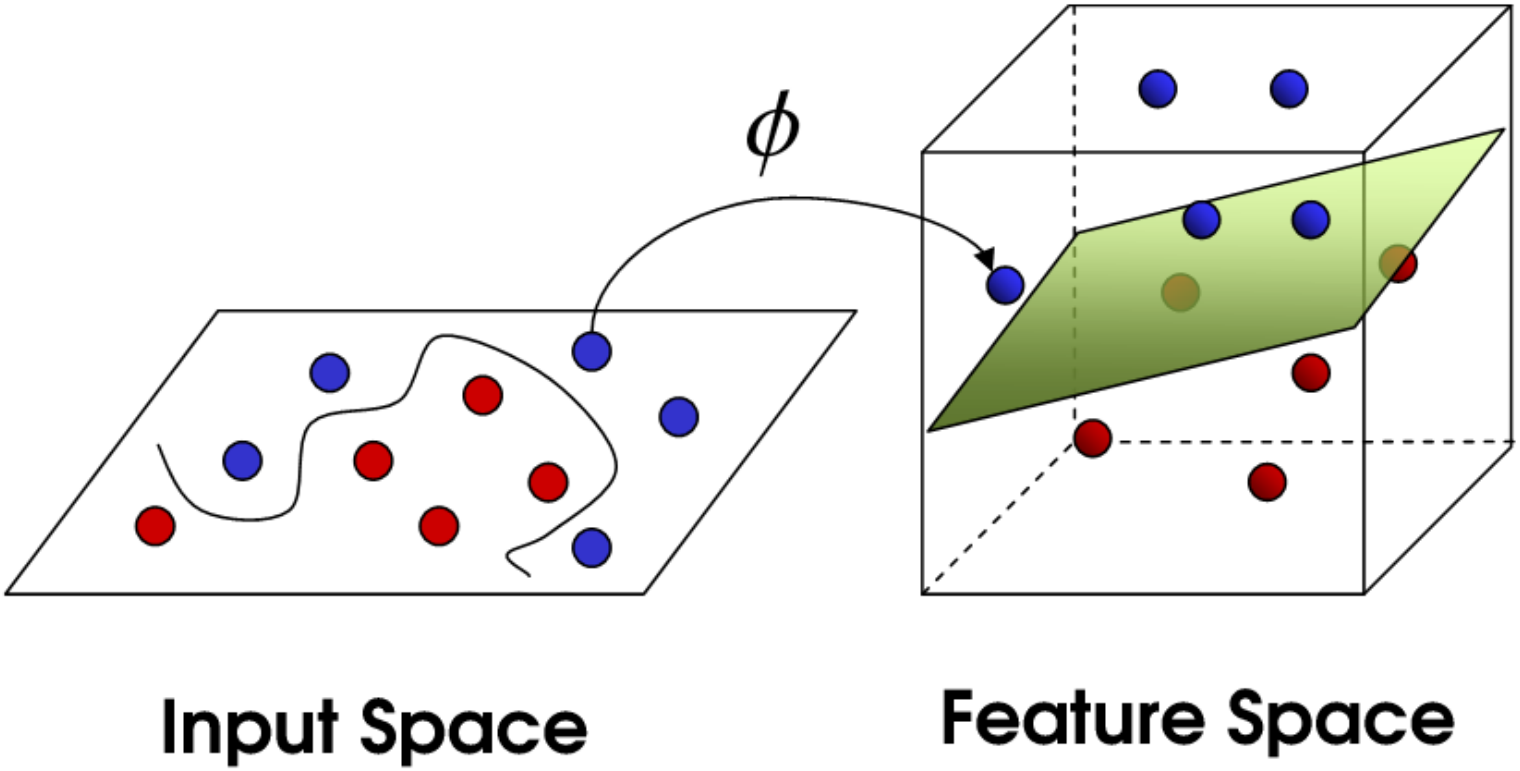
$$w = \sum_{j=1}^n \alpha_j y^{(j)} x^{(j)} \qquad b = -\frac{1}{2}\left(\min_{\substack{j=1,\dots,n \\ y^{(j)}=1}} w^T x^{(j)} + \max_{\substack{j=1,\dots,n \\ y^{(j)}=-1}} w^T x^{(j)}\right)$$

But actually we do not need w. We usually just need to calculate the following:

$$\begin{aligned} w^T x + b &= \left(\sum_{j=1}^n \alpha_j y^{(j)} x^{(j)}\right)^T x + b \\ &= \sum_{j=1}^n \alpha_j y^{(j)} (x^{(j)} \bullet x) + b\,. \end{aligned}$$

Conclusion: For everything we just need the dot product of elements in the feature space!

Going to higher dimensions



Kernels

If we can not separate our data by a hyperplane in our Feature space,
we go to higher dimensions.

Kernel trick:

- Write your algorithm in terms of dot product of elements in your feature space $\mathcal{X}$.
- Find a map from your input space $\mathcal{I}$ to a (probably higher dimensional) Feature space $\mathcal{X}$

$$\Phi : \mathcal{I} \to \mathcal{X} \,.$$

- Find a **Kernel function**

$$K : \mathcal{I} \times \mathcal{I} \to \mathbb{R}$$

such that

$$K(x, x') = \Phi(x) \bullet \Phi(x')$$

for $x, x' \in \mathcal{I}$.

Advantage: We can use our algorithm in higher dimensions without needing to explicitly calculate Φ! The kernel K can be arbitrary chosen so long as the existence of an Φ is guaranteed (Mercer's condition)

Kernels: Example

Kernels: More example

Assume that the input space is $\mathcal{I} = \mathbb{R}^N$.

i) **Polynomial kernels**: For any constant $c > 0$ and $d \in \mathbb{Z}_{\geq 1}$:

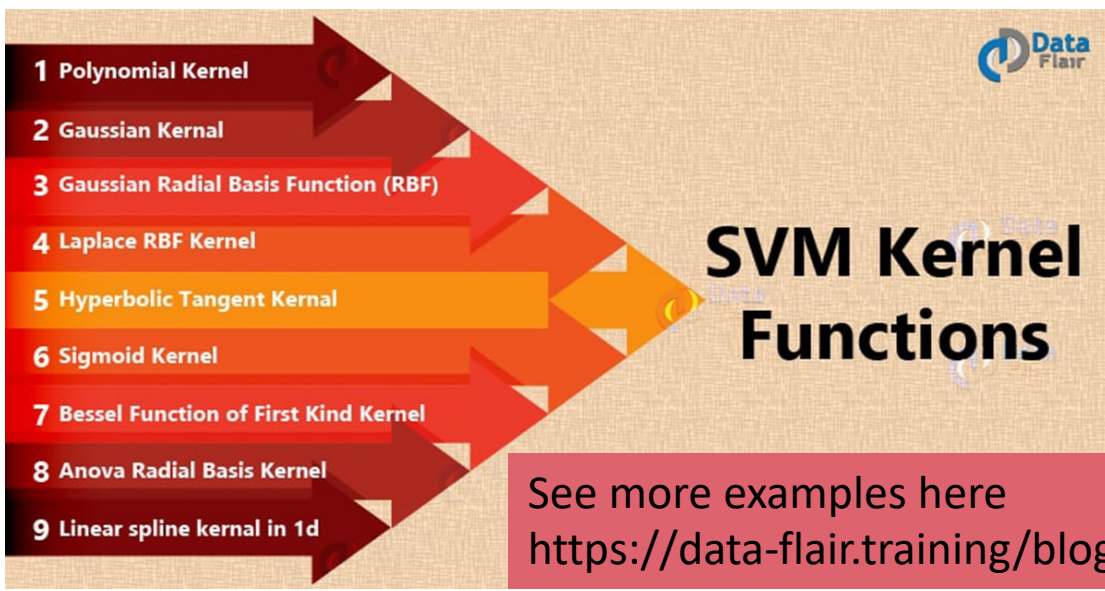
$$K(x, x') = (x \bullet x' + c)^d \,.$$

The corresponding feature space has dimension $\binom{N+d}{d}$ and the Φ sends an input vector to a vector containing all possible monomials up to degree d.

ii) **Gaussian kernels**: Define the kernel

$$K(x, x') = \exp\left(-\|x' - x\|^2\right) \,.$$

In this case the feature space is infinite dimensional!



See more examples here
https://data-flair.training/blogs/svm-kernel-functions/

Support Vector machines

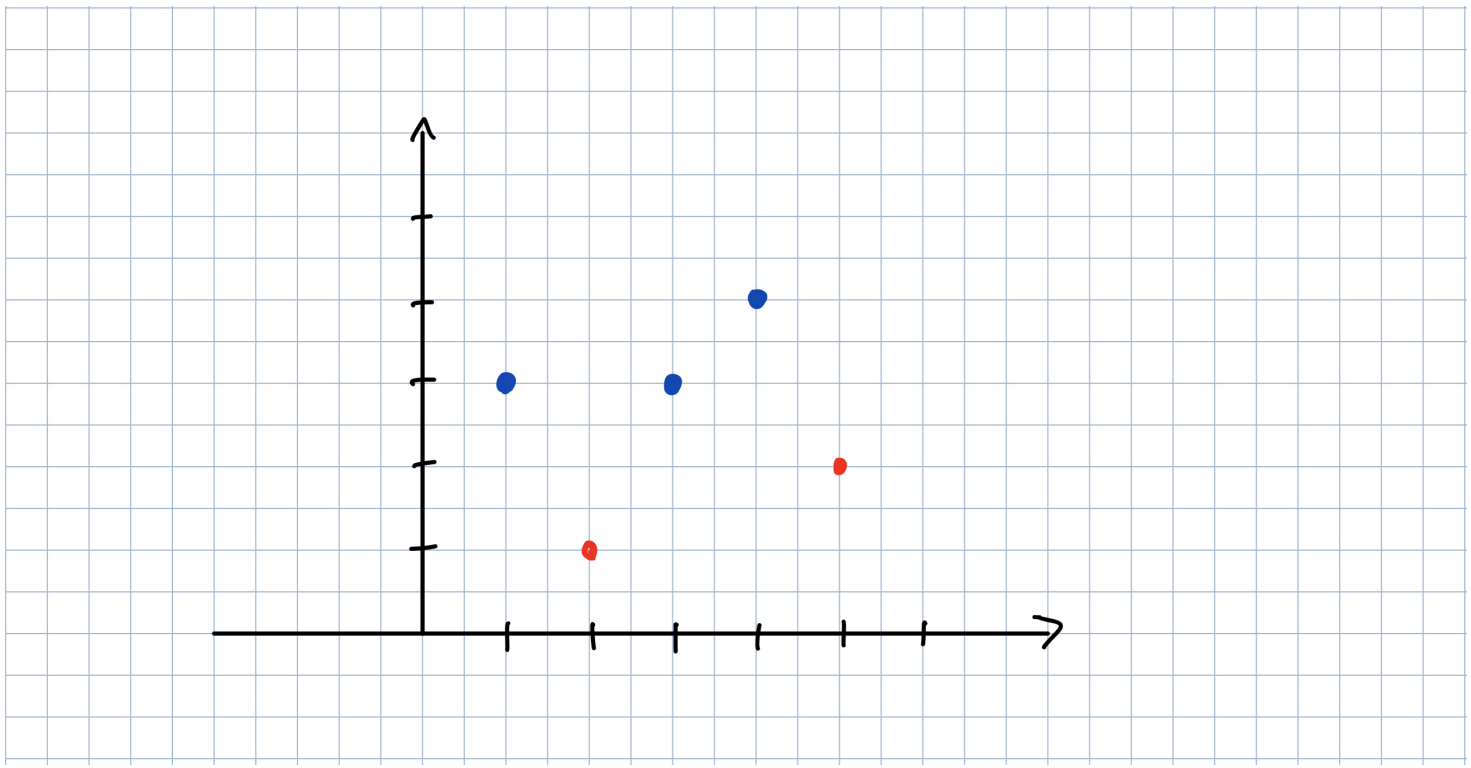
- There are various topics of SVM which we did not cover.
- For example: Often data can, even in higher dimension, not separated by a hyperplane.
- This leads to the notion of “soft margin” SVM.

For motivated students: Give a simple implementation of what we discussed today in Python. (Next slide)

Extra homework for motivated students

Let $d = 2$ and consider the training set

$$\mathcal{T} = \left(\left(\begin{pmatrix} 1 \\ 3 \end{pmatrix}, 1 \right), \left(\begin{pmatrix} 3 \\ 3 \end{pmatrix}, 1 \right), \left(\begin{pmatrix} 4 \\ 4 \end{pmatrix}, 1 \right), \left(\begin{pmatrix} 2 \\ 1 \end{pmatrix}, -1 \right), \left(\begin{pmatrix} 5 \\ 2 \end{pmatrix}, -1 \right) \right)$$



Implement the algorithm explained today in Python and find the hyperplane for the above training set. Other students (and me!) would be interested in an implementation.