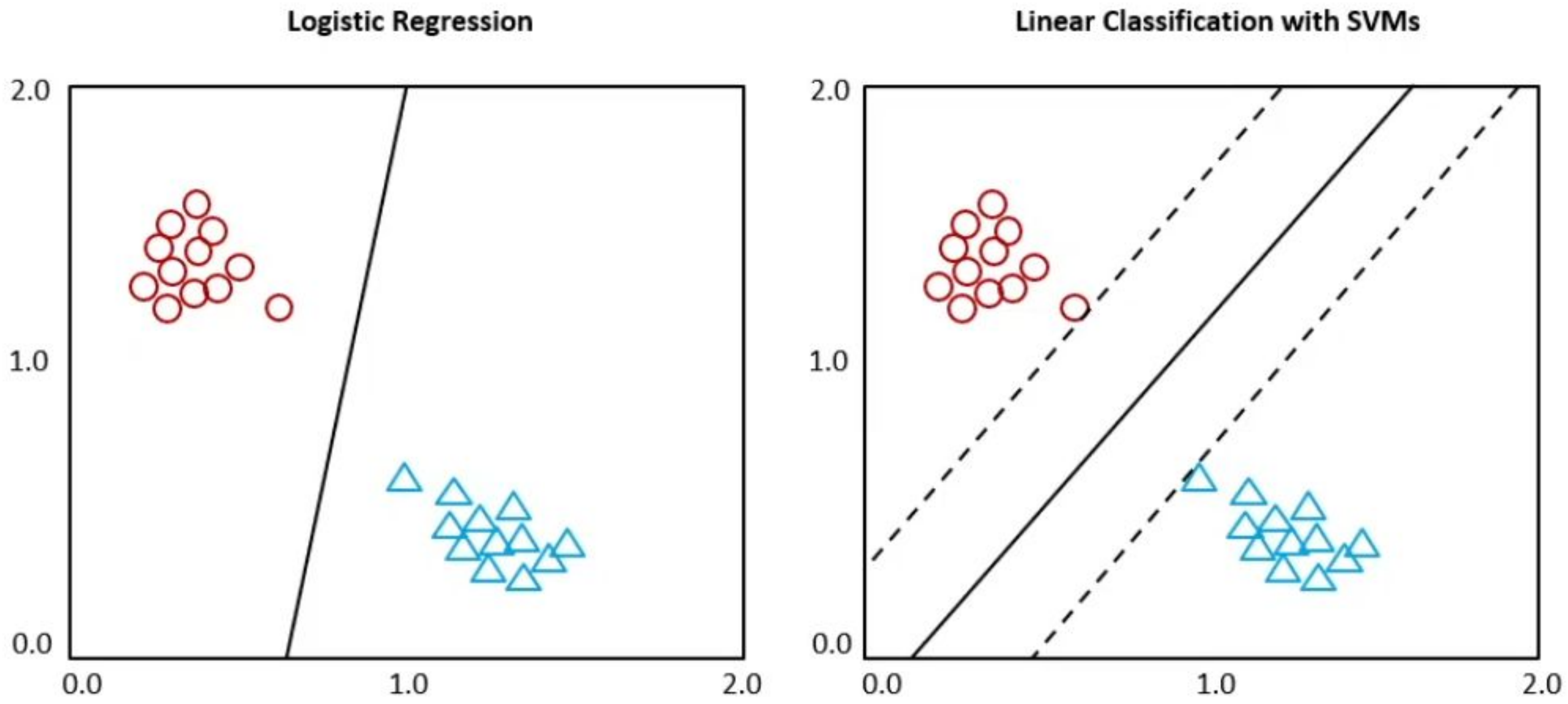
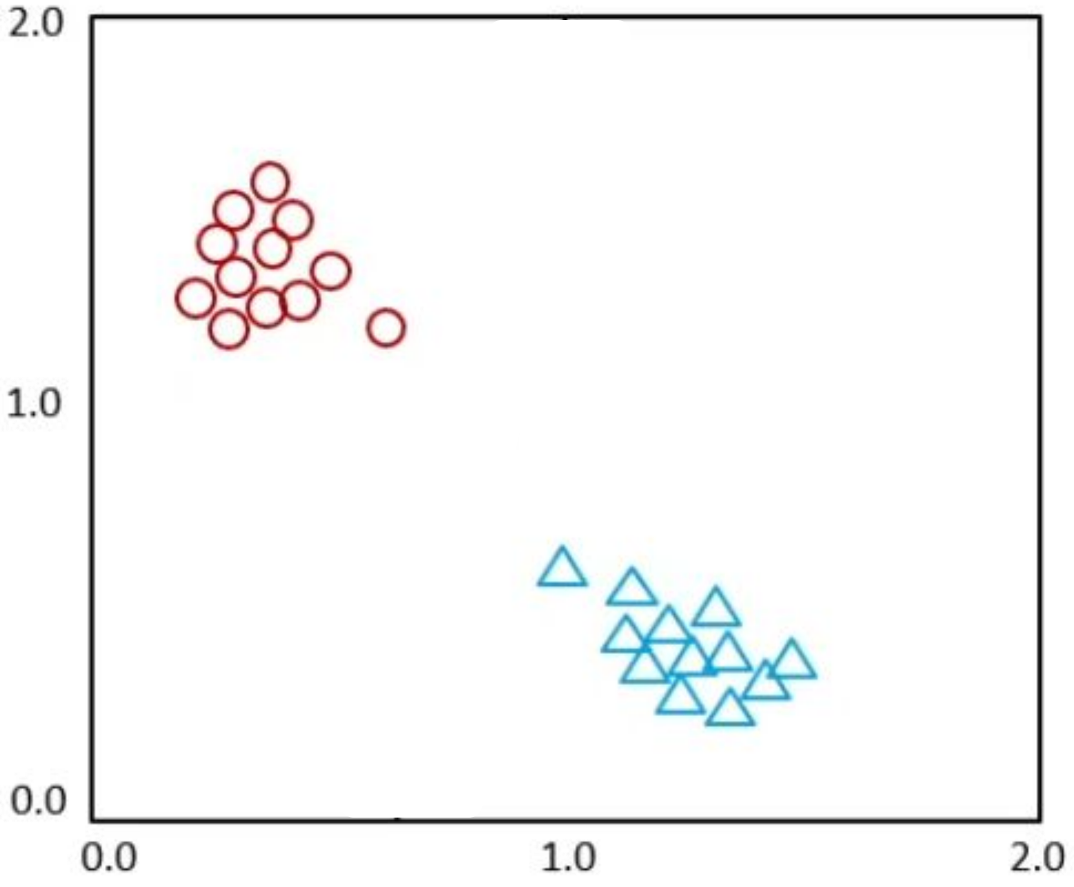
Support Vector Machine

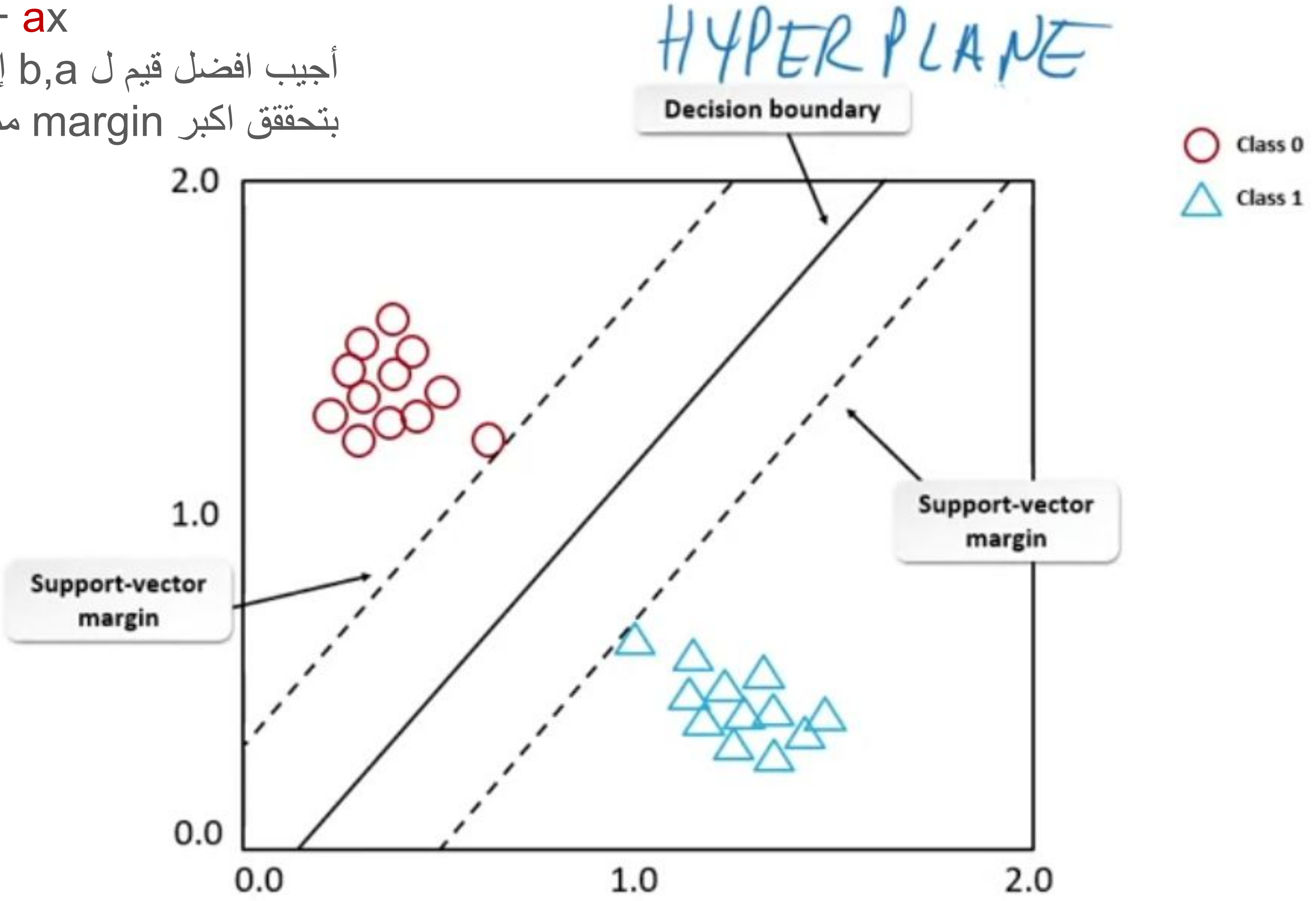
- outliers, Noise بتتعامل مع
- Curse of dimensionality
- Non linear date

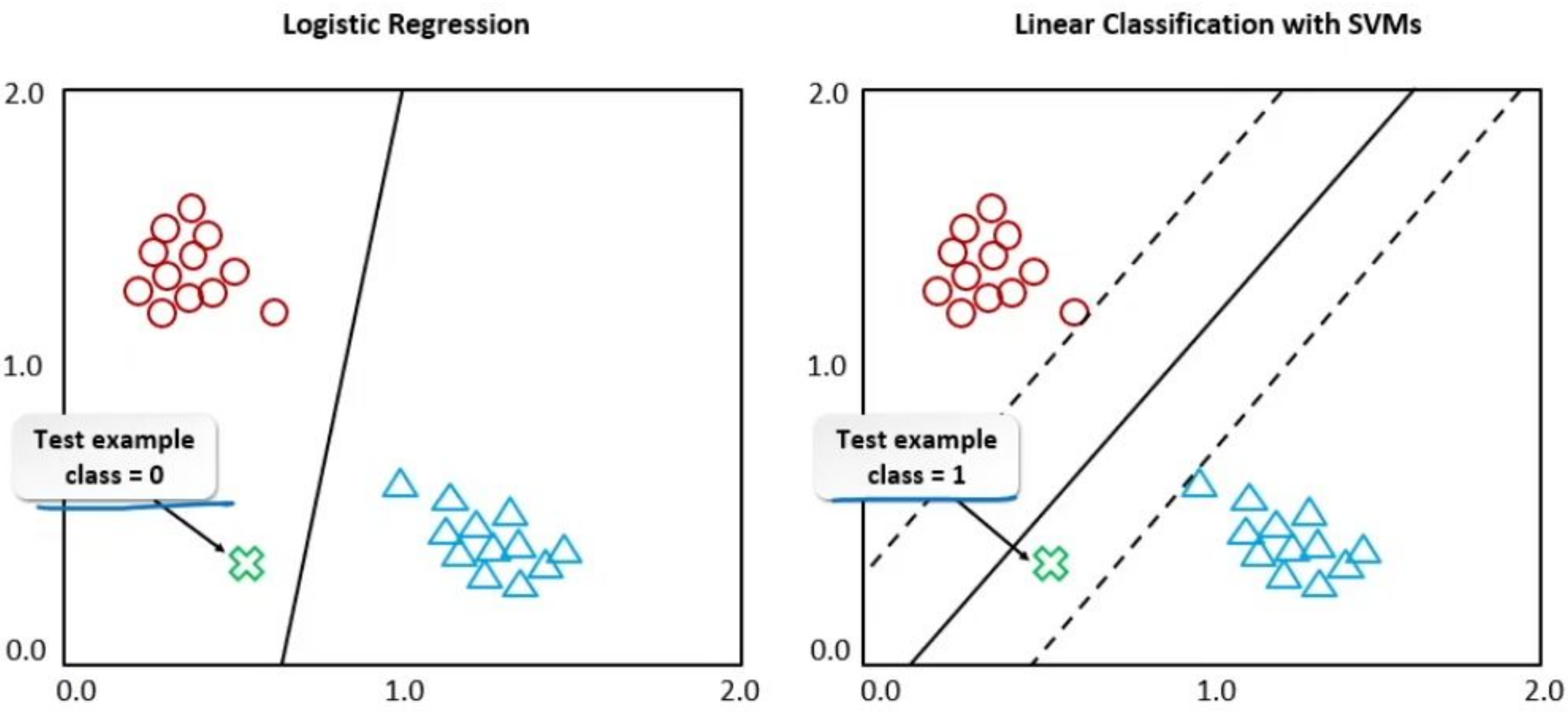

Logistic Regression
Linear Classification with SVMs
2.0
1.0
0.0
0.0
1.0
2.0
2.0
1.0
0.0
0.0
1.0
2.0


2.0
1.0
0.0
0.0
1.0
2.0

$Y = b + ax$

أجيب افضل قيم ل b,a إلي
بتحقق اكبر margin ممكن




Logistic Regression
Linear Classification with SVMs
2.0
1.0
0.0
0.0
1.0
2.0
Test example
class = 0
2.0
1.0
0.0
0.0
1.0
2.0
Test example
class = 1

What Is the Margin Exactly?

The **margin** is the distance between the hyperplane and the closest data points from each class — those are your support vectors.

Mathematically, the margin is:

$$\text{Margin} = \frac{2}{||w||}$$

So, if we want a larger margin, we need a **smaller weight magnitude (||w||)**.

That's why the optimization goal is to **minimize ||w||** while still classifying all points correctly.

The Optimization Problem

The SVM problem is formulated as an **optimization task:**

> **Minimize:**
>
> $$\frac{1}{2}||w||^2$$
>
> **Subject to:**
>
> $$y_i(w \cdot x_i + b) \geq 1$$

Where:

- y_i is the true class label (+1 or -1)
- x_i is the data point

The constraint ensures that every point lies on the correct side of the boundary.

So basically:
👉 We're searching for the **smallest w (largest margin)** that still classifies every point correctly.

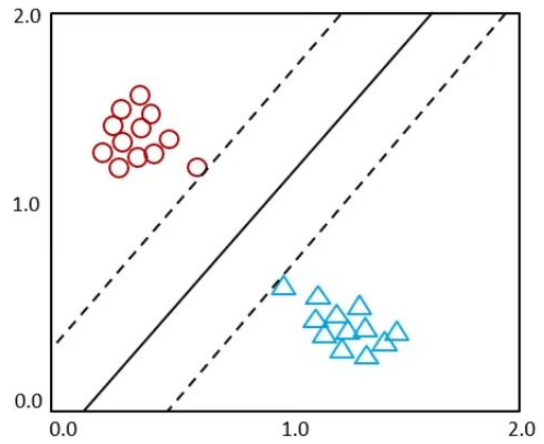
The algorithm doesn't just "draw lines and guess".
It uses mathematical optimization techniques like:

- **Quadratic Programming**
- **Lagrange Multipliers**

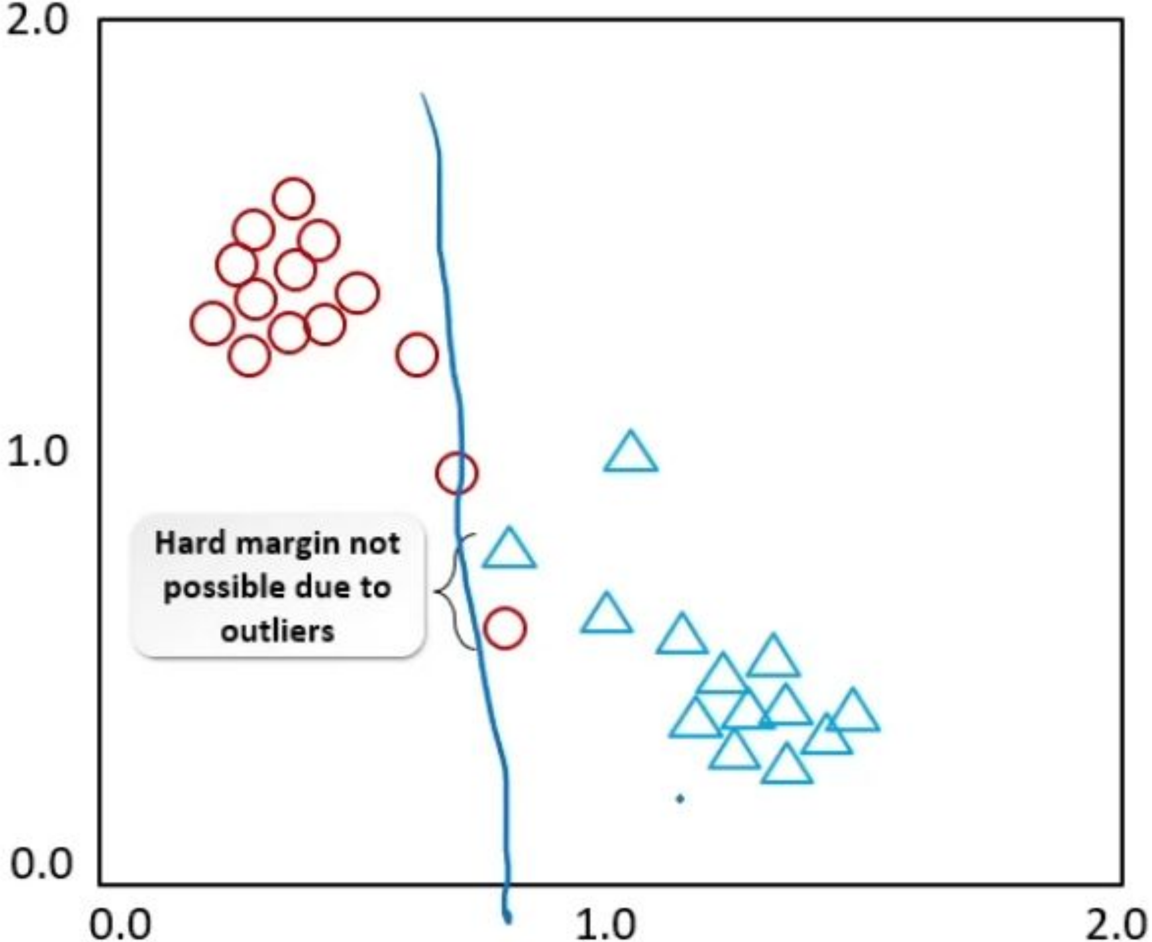
>>> to find the precise values of **w** and **b** that maximize the margin.

So, it keeps adjusting these parameters —
testing margins, calculating distances —
until it finds the line that perfectly balances both sides.

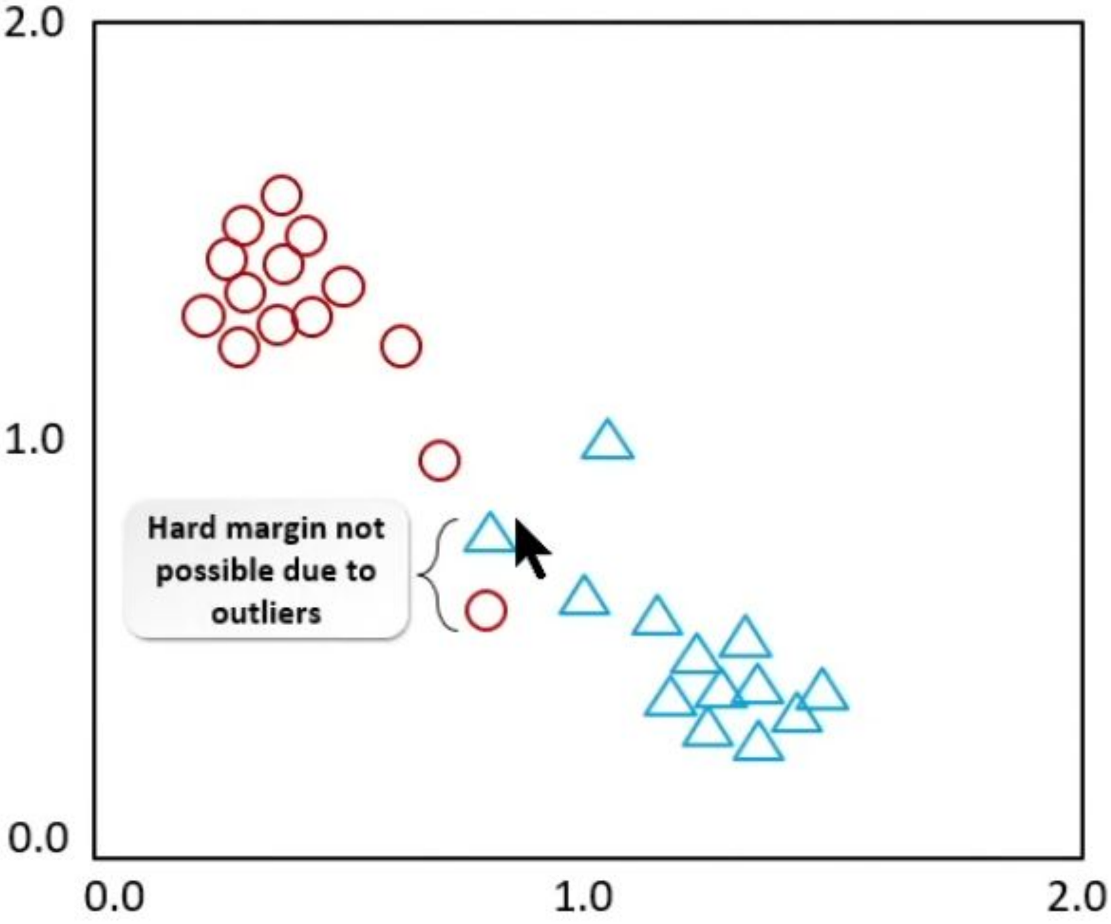
Linear Classification with SVMs

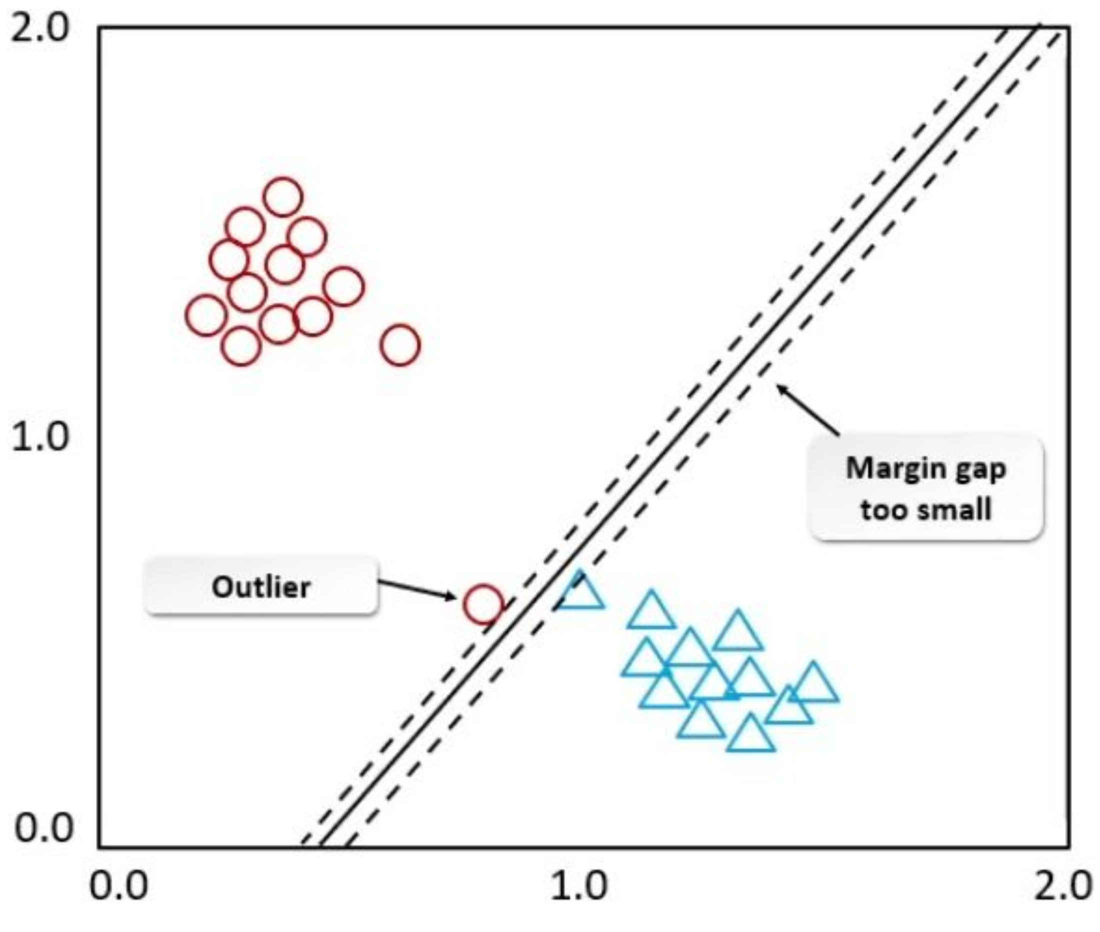


hard Margins

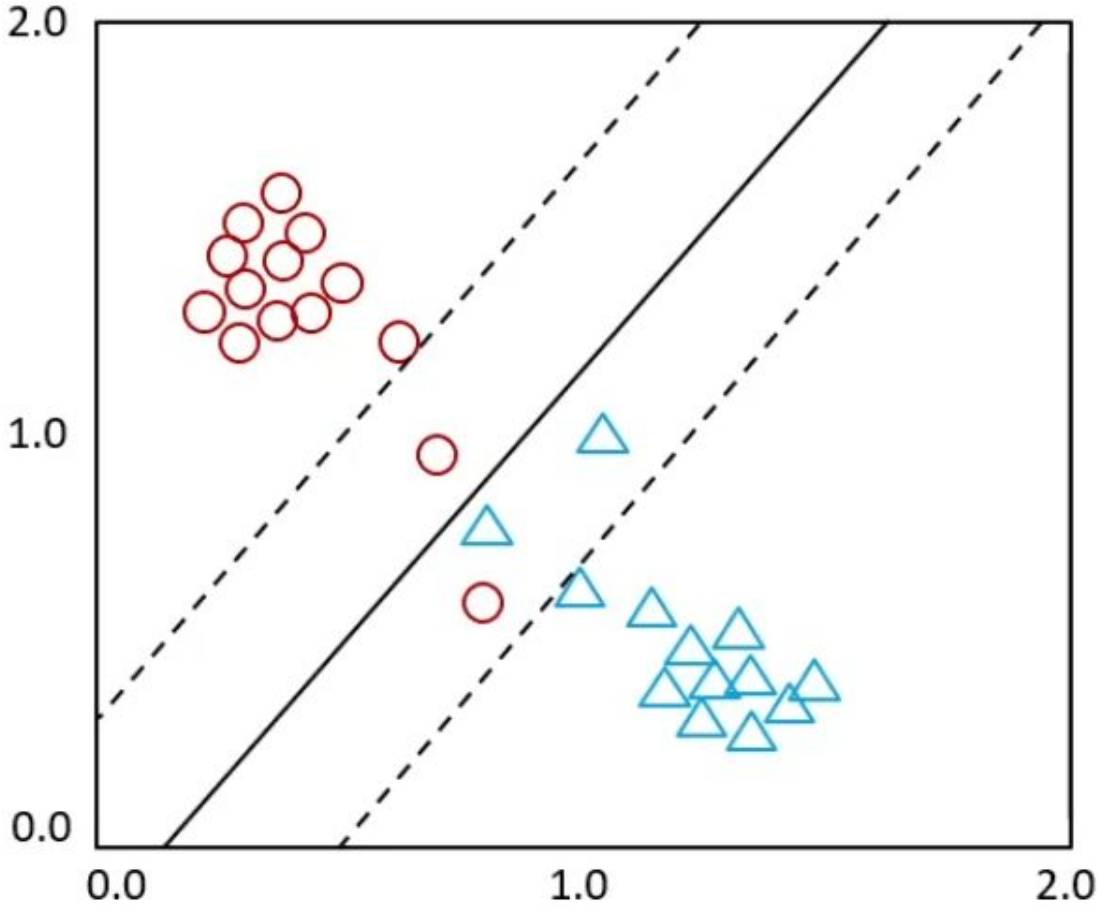


hard Margins




2.0
1.0
0.0
0.0
1.0
2.0
Margin gap
too small
Outlier

Soft Margins

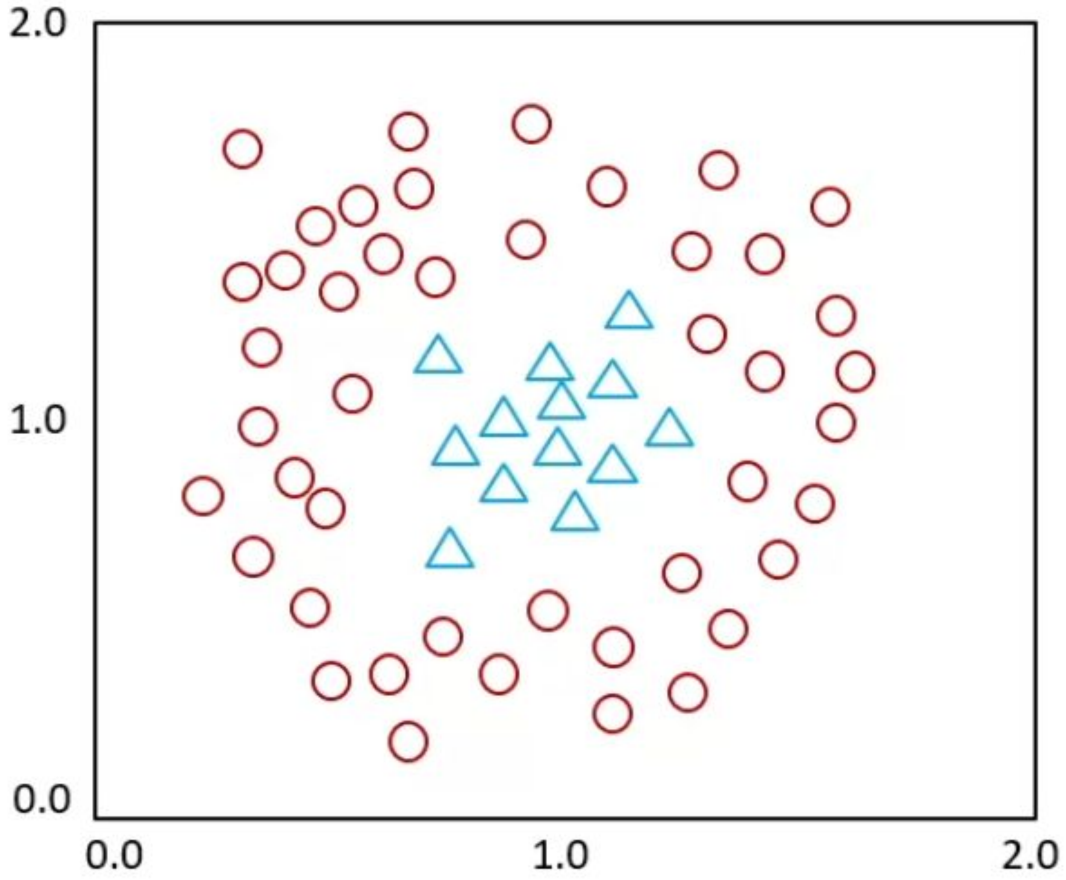


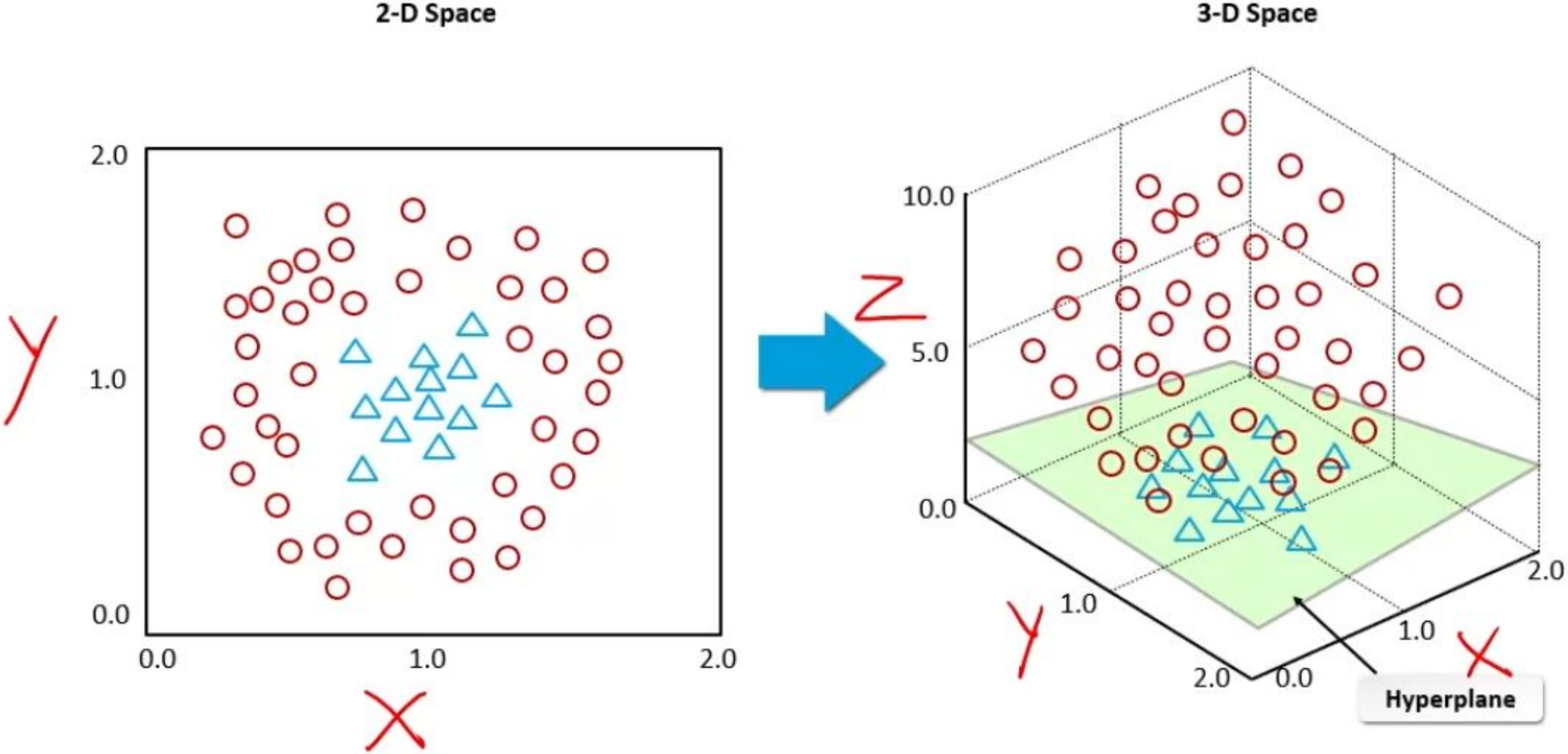
Regularization Penalty in SVM

As hyperparameter > **C**

low C >> soft margins
High C >> Hard margins >> overfitting

Nonlinear Classification




2-D Space
3-D Space
2.0
1.0
0.0
0.0
1.0
2.0
Y
X
Z
10.0
5.0
0.0
Y
1.0
2.0
0.0
1.0
2.0
X
Hyperplane

Feature Expansion / Feature Mapping / Basis Expansion

- Explicitly transform input into higher-dimensional features

$$K(x, y) = \langle f(x), f(y) \rangle$$

Assume:

- **x = (x_1, x_2, x_3)**
- **y = (y_1, y_2, y_3)**

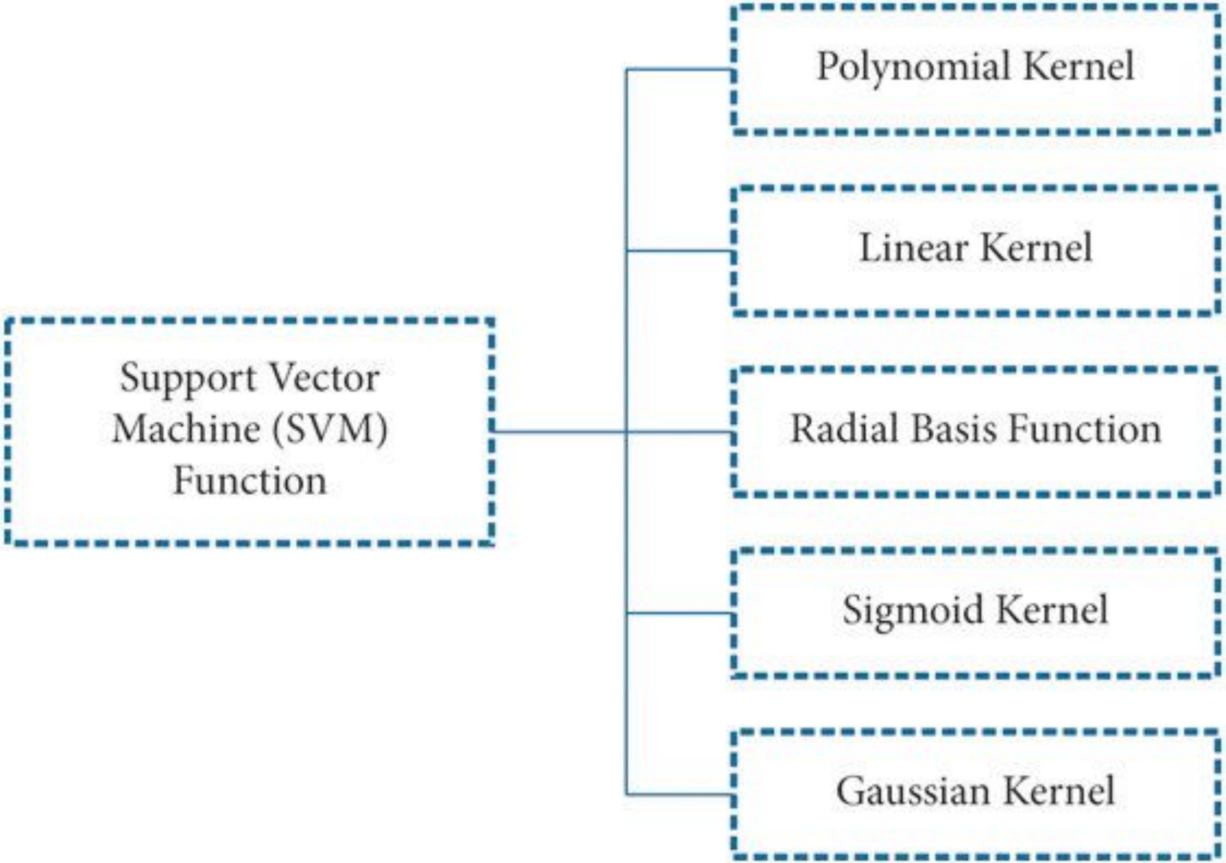
$$(x_1x_1, x_1x_2, x_1x_3, x_2x_1, x_2x_2, x_2x_3, x_3x_1, x_3x_2, x_3x_3)$$

Kernel Trick

$x = (0, 1, 2)$

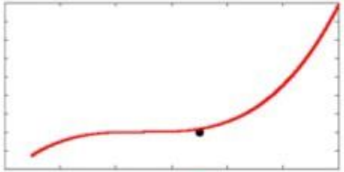
$y = (3, 4, 5)$

$$K(x, y) = (\underline{0} + \underline{4} + \underline{10})^2 = 196$$

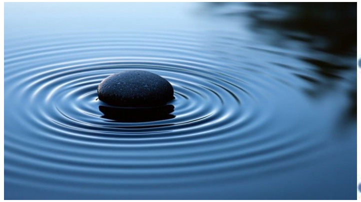

Support Vector Machine (SVM) Function
Polynomial Kernel
Linear Kernel
Radial Basis Function
Sigmoid Kernel
Gaussian Kernel

Common kernel functions

- Some commonly used kernel functions & their shape:

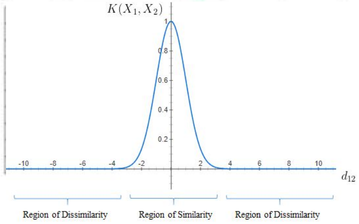


- Polynomial $K(a,b) = (1 + \sum_j a_j b_j)^d$



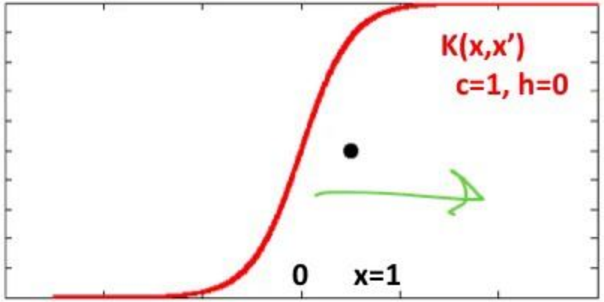
- Radial Basis Functions & Gaussian

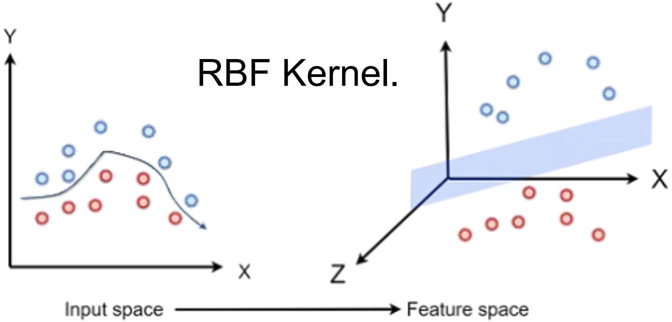
$$K(a,b) = \exp(-(a-b)^2/2\sigma^2)$$



- Saturating, sigmoid-like:

$$K(a,b) = \tanh(ca^T b + h)$$





Guidelines for Building SVM Models for Classification

- Consider using an SVM model when the problem you are trying to solve is sensitive to outliers.
- Consider using an SVM model when working with a high-dimensional dataset.
- In classification, recognize that the goal of an SVM model is to widen the margins as much as is feasible, while at the same time keeping data examples outside of the margins.
- Tune the (c)regularization hyperparameter to adjust the size of the margins.

Guidelines for Building SVM Models for Classification

- Consider that narrowing the margins too much to keep all examples outside of those margins may lead to complications (e.g., overfitting).Consider softening the margins to avoid hard-margin overfitting issues.
- Recognize that softening the margins will likely place some examples within those margins, which is often a necessary tradeoff.
- Apply a kernel trick method to SVM models whose training data is not linearly separable.
- Consider the different types of kernel methods and how one might be more applicable to your current problem.
- gamma > قديش المودل مركز على الجيران القريبين ولا مبسط اكثر؟