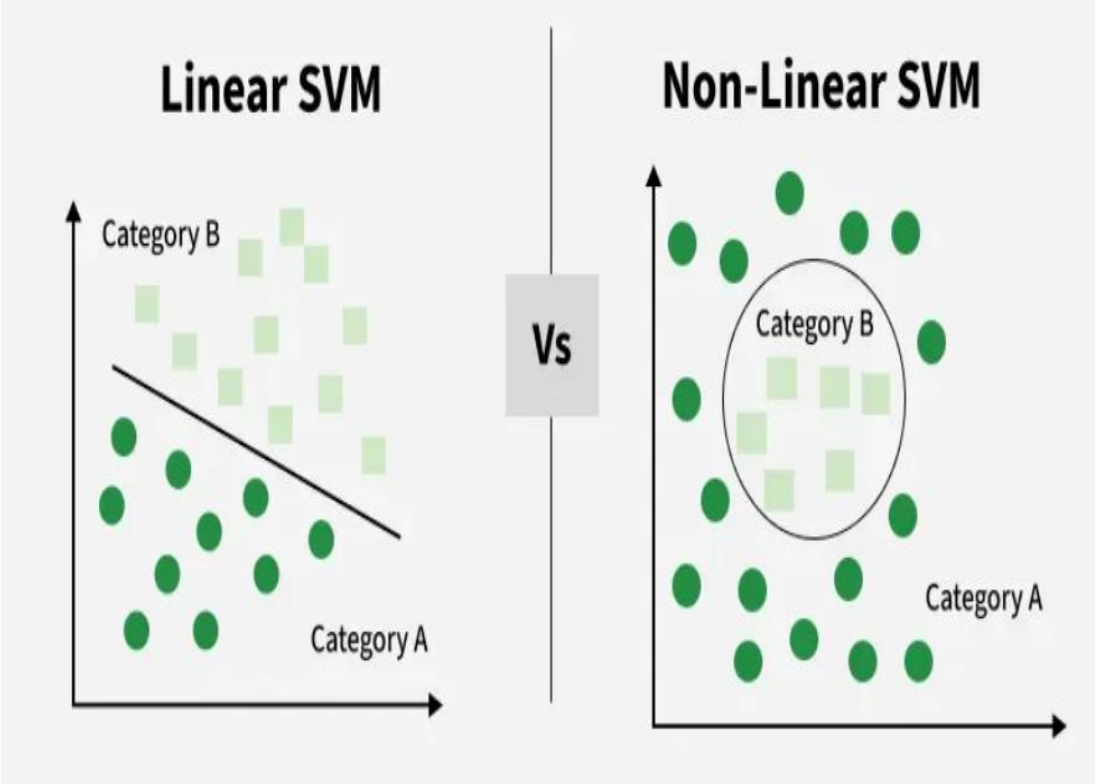
SUPPORT VECTOR MACHINE (SVM) ALGORITHM

- Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification and regression tasks.
- It tries to find the best boundary known as hyperplane that separates different classes in the data.
- It is useful when you want to do binary classification like spam vs. not spam or cat vs. dog.
- The main goal of SVM is to maximize the margin between the two classes.
- The larger the margin the better the model performs on new and unseen data.





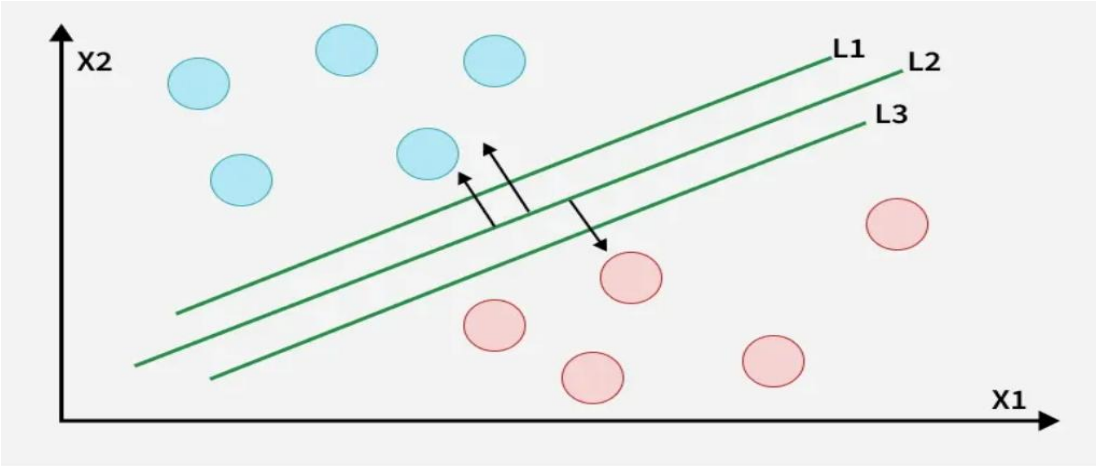
Key Concepts

- **Hyperplane**: A decision boundary separating different classes in feature space and is represented by the equation wx + b = 0 in linear classification.
- **Support Vectors**: The closest data points to the hyperplane, crucial for determining the hyperplane and margin in SVM.
- **Margin**: The distance between the hyperplane and the support vectors. SVM aims to maximize this margin for better classification performance.
- **Kernel**: A function that maps data to a higher-dimensional space enabling SVM to handle non-linearly separable data.

- **Hard Margin**: A maximum-margin hyperplane that perfectly separates the data without misclassifications.
- **Soft Margin**: Allows some misclassifications by introducing slack variables, balancing margin maximization and misclassification penalties when data is not perfectly separable.
- **C**: A regularization term balancing margin maximization and misclassification penalties. A higher C value forces stricter penalty for misclassifications.
- **Hinge Loss**: A loss function penalizing misclassified points or margin violations and is combined with regularization in SVM.
- **Dual Problem**: Involves solving for Lagrange multipliers associated with support vectors, facilitating the kernel trick and efficient computation.

Working of Support Vector Machine Algorithm

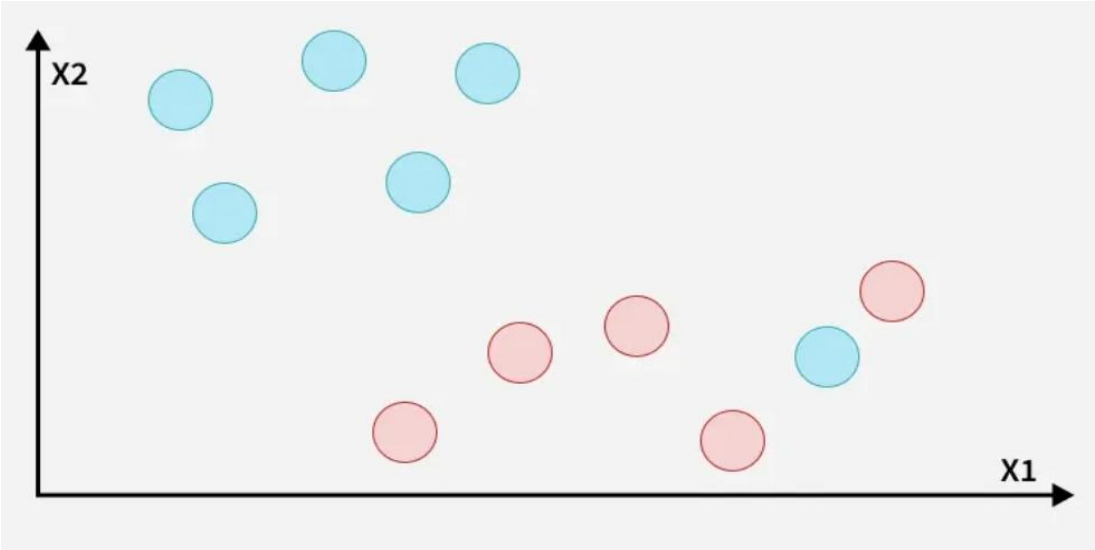
- The key idea behind the SVM algorithm is to find the hyperplane that best separates two classes by maximizing the margin between them.
- This margin is the distance from the hyperplane to the nearest data points (support vectors) on each side.



Multiple hyperplanes separate the data from two classes

- The best hyperplane also known as the hard margin is the one that maximizes the distance between the hyperplane and the nearest data points from both classes.

- This ensures a clear separation between the classes. So from the above figure, we choose L2 as hard margin.
- Let's consider a scenario like shown below:

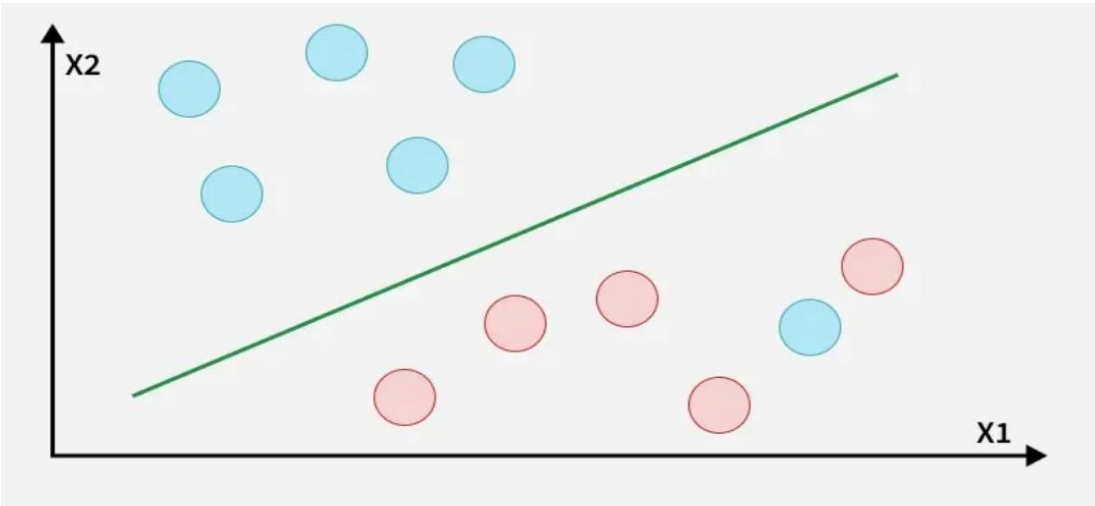


Selecting hyperplane for data with outlier

Here, we have one blue ball in the boundary of the red ball.

How does SVM classify the data?

- The blue ball in the boundary of red ones is an outlier of blue balls.
- The SVM algorithm has the characteristics to ignore the outlier and finds the best hyperplane that maximizes the margin.
- SVM can be sensitive to outliers, especially in the case of a hard margin, while soft margin SVM helps reduce their impact by allowing some misclassifications.



Hyperplane which is the most optimized one

- A soft margin allows for some misclassifications or violations of the margin to improve generalization.

- The SVM optimizes the following equation to balance margin maximization and penalty minimization:

$$\text{Objective Function} = \frac{1}{\text{margin}} + \lambda \sum \text{penalty}$$

How it connects to the formal SVM formulation

- $\frac{1}{\text{margin}}$ → In practice, this is represented as $\frac{1}{2}\|w\|^2$.
  - Since the margin is proportional to $\frac{1}{\|w\|}$, minimizing $\|w\|^2$ is equivalent to maximizing the margin.
- $\lambda \sum \text{penalty}$ → This corresponds to $C \sum_{i=1}^{n} \xi_i$.
  - The slack variables ξ_i measure violations of the margin.
  - The parameter C (your λ) controls the trade-off between margin maximization and penalty minimization.

So the more precise optimization problem is:

$$\min_{w,b,\xi} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i$$

subject to:

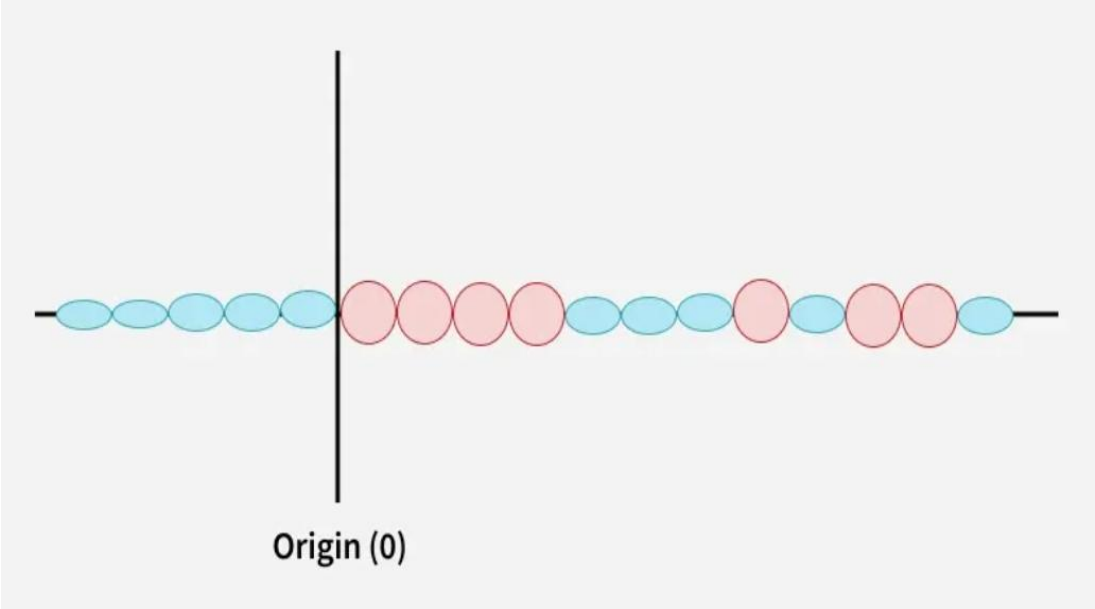
$$y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0$$

- The penalty used for violations is often hinge loss which has the following behavior:
- If a data point is correctly classified and lies outside the margin, there is no penalty (loss = 0). If it is correctly classified but lies within the margin or is misclassified, the hinge loss is greater than 0.
- If a point is incorrectly classified or violates the margin the hinge loss increases proportionally to the distance of the violation.

What if data is not linearly separable?

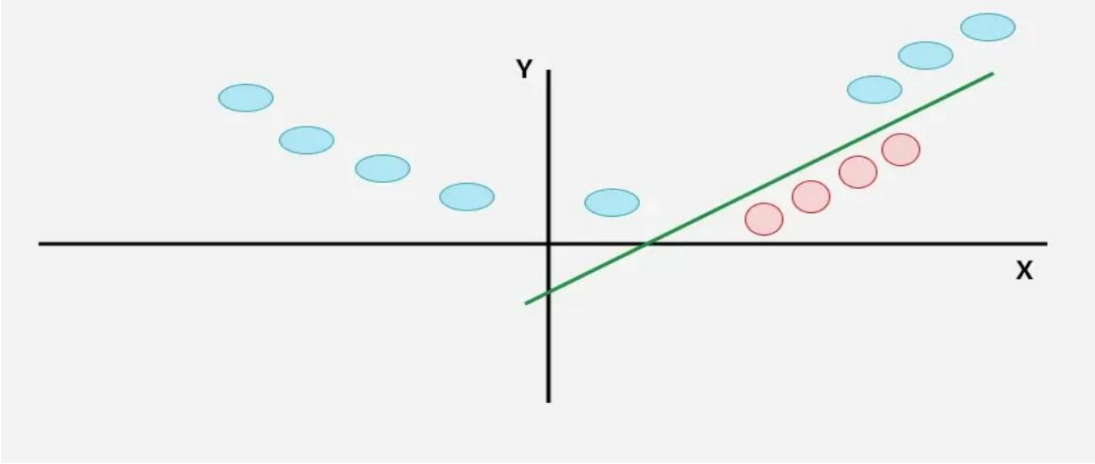
- When data is not linearly separable i.e it can't be divided by a straight line, SVM uses a technique called kernels to map the data into a higher-dimensional space where it becomes separable.

- This transformation helps SVM find a decision boundary even for non-linear data.



Original 1D dataset for classification

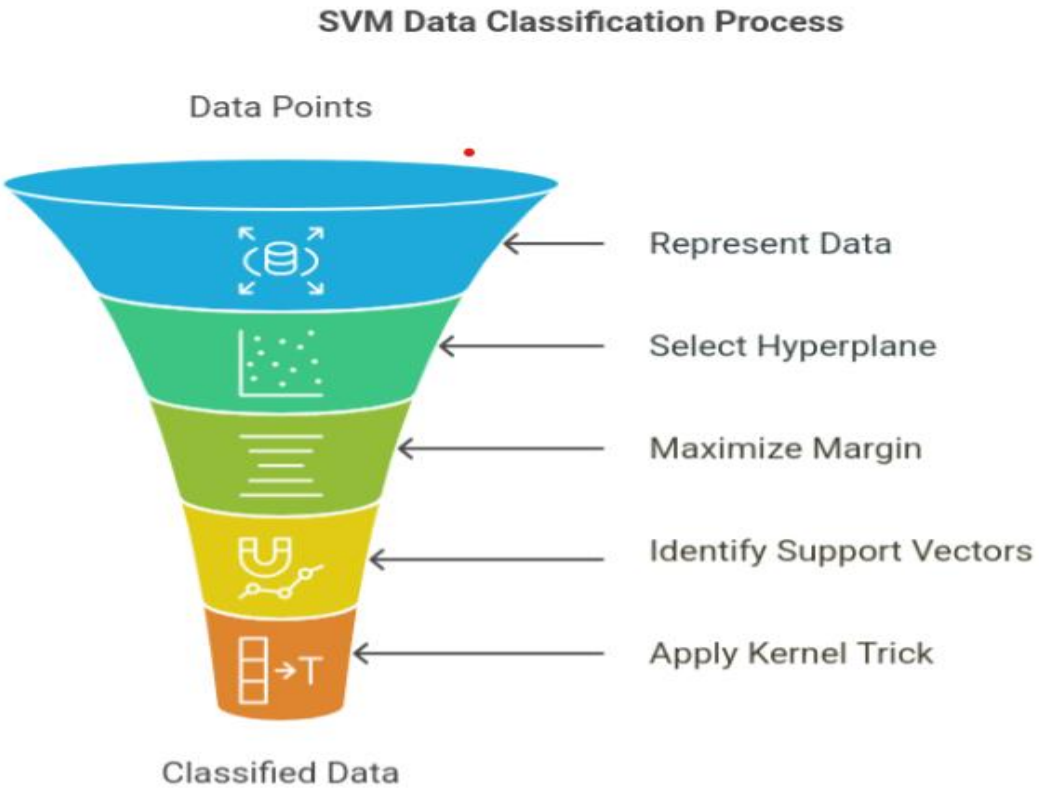
- A kernel is a function that maps data points into a higher-dimensional space without explicitly computing the coordinates in that space.
- This allows SVM to work efficiently with non-linear data by implicitly performing the mapping.
- For example consider data points that are not linearly separable. By applying a kernel function SVM transforms the data points into a higher-dimensional space where they become linearly separable.
- Linear Kernel: For linear separability.
- Polynomial Kernel: Maps data into a polynomial space.
- Radial Basis Function (RBF) Kernel: Transforms data into a space based on distances between data points.



Mapping 1D data to 2D to become able to separate the two classes

In this case the new variable y is created as a function of distance from the origin.

How Does SVM Classify Data?



- SVM follows these steps to classify data:
- Data Representation: Each data point is represented as a vector in an n-dimensional space.
- Hyperplane Selection: SVM finds the optimal hyperplane that separates the data into distinct classes.
- Maximizing the Margin: It ensures that the separation margin between classes is as wide as possible to improve generalization.
- Using Support Vectors: The closest points to the hyperplane (support vectors) influence its placement.
- Kernel Trick (for Non-Linearity): If the data is not linearly separable, SVM transforms it into a higher dimension using kernel functions.

Mathematical Foundation of SVM Algorithm

For two-class classification, given a dataset (xi, yi), where xi are feature vectors and yi are class labels (either +1 or -1), SVM aims to minimize the function:

minw,b (1/2) ||w||2

Subject to:

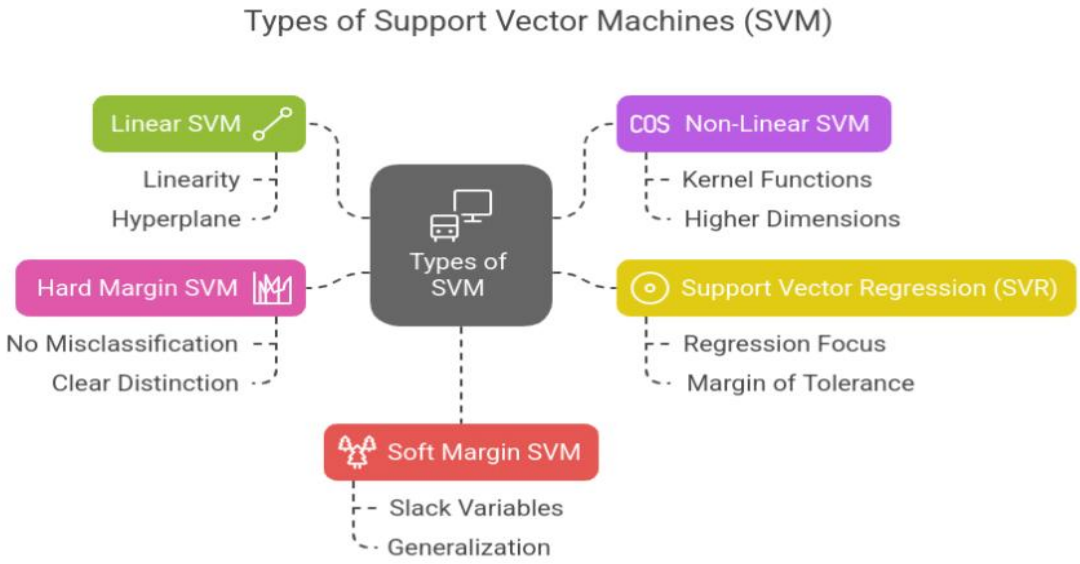
$y_i (w \cdot x_i + b) \geq 1$ for all I

where:
w represents the weight vector,
b is the bias term,
$||w||_2$ ensures the maximization of the margin.

- In cases where data points are not entirely separable, slack variables ξ are introduced, modifying the function as follows:
- $\min_{w,b,\xi} (1/2) ||w||_2 + C \Sigma \xi_i$
- Here, C controls the trade-off between maximizing the margin and minimizing classification errors.

Types of SVM

- SVM can be classified into different types based on the nature of the dataset:



- **Linear SVM**: Used when data can be separated using a straight hyperplane. It is suitable for datasets where classes are linearly separable. The decision boundary is a straight line (in 2D) or a plane (in higher dimensions).
- **Non-Linear SVM**: When data is not linearly separable, kernel functions are used to map data into a higher-dimensional space where separation is possible. Kernel functions like RBF and polynomials are commonly used in non-linear SVM.
- **Support Vector Regression (SVR)**: A variation of SVM used for regression problems instead of classification. It works similarly to classification SVM but tries to fit a function within a margin of tolerance rather than finding a strict boundary between categories.
- **Hard Margin SVM**: Assumes that the data is perfectly separable and aims to find a hyperplane that classifies all data points correctly with no tolerance for

misclassification. This works well when there is a clear distinction between classes.

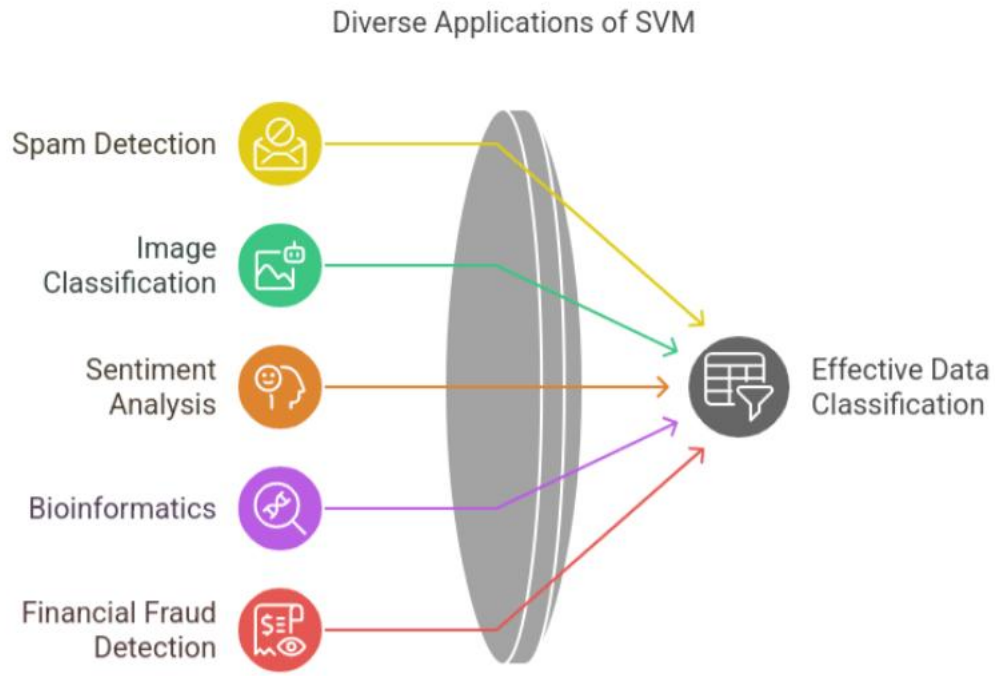
- **Soft Margin SVM:** Introduces slack variables to handle cases where classes may overlap slightly. It allows some misclassifications to improve generalization and prevent overfitting.

THE KERNEL TRICK: HANDLING NON-LINEARITY

- Real-world data is often non-linearly separable. SVM uses kernel functions to transform data into a higher-dimensional space where it becomes separable. Some popular kernels include:
- Linear Kernel: K(x, y) = x · y (used in linear support vector machine)
- Polynomial Kernel: K(x, y) = (x · y + c)d
- Radial Basis Function (RBF) Kernel: K(x, y) = e−γ||x − y||2

Real-World Applications of SVM

- SVM has practical use cases across different domains:
-



- Spam Detection: In this, we classify emails as spam or not spam.
- Image Classification: Recognizing objects, faces, or handwritten digits.
- Sentiment Analysis: Determining if a review is positive or negative.
- Bioinformatics: Identifying diseases based on genetic data.
- Financial Fraud Detection: Identifying unusual transaction patterns.

EXAMPLE PROBLEM

- Consider the following dataset:

- Using the Support Vector Machine (SVM) approach:
- Derive the equation of the optimal separating hyperplane.
- Identify the support vectors.
- Compute the margin width.
- Write the decision function for classification.

Point	x_1	x_2	Class (y)
A	2	2	+1
B	4	4	+1
C	4	0	-1
D	0	0	-1

3. Optimization

- We minimize:
- subject to the above inequalities.

$$\frac{1}{2}(w_1^2 + w_2^2)$$

4. Solution (Geometric Insight)

- The optimal separating line lies between the positive cluster (near (2,2), (4,4)) and the negative cluster (near (0,0), (4,0)).
- One valid solution is:
- $w=(1,-1), b=0$
- So the hyperplane is:
- $x1-x2=0$

1. Visual Inspection

- Positive points: (2,2), (4,4) → lie along the diagonal line x1=x2.
- Negative points: (0,0), (4,0) → lie below that diagonal.
- So geometrically, the separating boundary should be close to the line x1=x2.
-

2. Hyperplane Form

- We want a line:
- $w1x1+w2x2+b=0$
-

3. Choosing w and b

- If we set $w=(1,-1)$, then the equation becomes:
- $x1-x2+b=0$
- Try b=0. That gives the line:
- $x1-x2=0$

4. Checking Constraints

- For each point:
  $yi(w1xi1+w2xi2+b)\geq 1$
- So:
- A: $+1(2w1+2w2+b)\geq 1$
- B: $+1(4w1+4w2+b)\geq 1$
- C: $-1(4w1+0w2+b)\geq 1$
- D: $-1(0w1+0w2+b)\geq 1$

- For A (2,2,+1): $+1((2-2)+0)=0$ → lies exactly on the boundary (support vector).
- For B (4,4,+1): $+1((4-4)+0)=0$ → also on the boundary.
- For C (4,0,-1): $-1((4-0)+0)=-4$ → correctly classified as negative.
- For D (0,0,-1): $-1((0-0)+0)=0$ → on the boundary.
- Thus, the line $x1-x2=0$ separates positives (above diagonal) and negatives (below diagonal).

5. Decision Function

- $f(x)=\text{sign}(x1-x2)$
- If $x1>x2$ → class -1
- If $x1<x2$ → class +1
-

6. Margin

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

Support vectors are the points closest to the line (A and C).

