Support Vector Machine (SVM) – Introduction

1. What is SVM?

Support Vector Machine (SVM) is one of the most popular **Supervised Learning algorithms** used for both:

- **Classification** (primary use)
- **Regression** (less common, called Support Vector Regression – SVR)

The main goal of the SVM algorithm is to **find the best decision boundary** (called a **hyperplane**) that separates data points into different classes in an *n*-dimensional space.

---

2. Hyperplane

- A **hyperplane** is the decision boundary that separates data into classes.
- The dimension of the hyperplane depends on the number of features in the dataset:
    - If there are **2 features** → the hyperplane is a **straight line**.
    - If there are **3 features** → the hyperplane is a **2D plane**.
    - If there are **n features** → the hyperplane will be an *(n–1)* dimensional space.

Mathematically, a hyperplane in *n*-dimensional space can be represented as:

w·x+b=0w \cdot x + b = 0w·x+b=0

where:

- www = weight vector (normal to the hyperplane)
- xxx = input feature vector
- bbb = bias term

---

3. Support Vectors

- **Support Vectors** are the data points that are closest to the hyperplane.
- These points directly affect the position and orientation of the hyperplane.
- Since they “support” the hyperplane, they are called **Support Vectors**.

The distance between the hyperplane and the nearest support vectors is called the **margin**.
SVM aims to maximize this margin (called **Maximum Margin Classifier**).

---

4. Types of SVM

SVM can be broadly divided into two categories:

4.1 Linear SVM

- Used when data is **linearly separable** (can be divided using a straight line).
- Example: If data points of two classes can be separated by a single straight line in 2D.

Decision function:

f(x)=sign(w·x+b)f(x) = \text{sign}(w \cdot x + b)f(x)=sign(w·x+b)

4.2 Non-Linear SVM

- Used when data is **not linearly separable**.
- A straight line (or plane) cannot separate the classes properly.
- To handle this, SVM uses a technique called the **Kernel Trick**.

Kernel Trick:

- The idea is to map the input data into a **higher-dimensional space** where it becomes linearly separable.
- Example: For two features xxx and yyy, we add a third dimension zzz:

z=x2+y2z = x^2 + y^2z=x2+y2

- Now, the data that was non-linear in 2D becomes linear in 3D.

Common Kernels:

1. **Linear Kernel**: K(xi,xj)=xi·xjK(x_i, x_j) = x_i \cdot x_jK(xi,xj)=xi·xj
2. **Polynomial Kernel**: K(xi,xj)=(xi·xj+c)dK(x_i, x_j) = (x_i \cdot x_j + c)^dK(xi,xj)=(xi·xj+c)d
3. **Radial Basis Function (RBF) Kernel**:

K(xi,xj)=exp(−γ||xi−xj||2)K(x_i, x_j) = \exp\left(-\gamma ||x_i - x_j||^2\right)K(xi,xj)=exp(−γ||xi−xj||2)

---

5. Optimization Objective

The optimization goal of SVM is to:

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

subject to the constraint:

$$y_i (w \cdot x_i + b) \geq 1, \quad \forall i$$

where:

- $y_i \in \{-1, +1\}$ = class labels
- x_i = feature vectors
- w, b = parameters of the hyperplane

This ensures the **maximum margin** between classes.

---

6. Applications of SVM

- Text classification (spam detection, sentiment analysis)
- Image recognition (face detection, object recognition)
- Bioinformatics (gene classification, protein analysis)
- Handwriting recognition

✅ Key Takeaways

- SVM is mainly used for **classification tasks**.
- The **hyperplane** is the decision boundary.
- **Support vectors** are the critical points influencing the boundary.
- SVM tries to maximize the **margin** for better generalization.
- For **non-linear data**, the **kernel trick** is used to project data into higher dimensions.

Would you like me to also **add a small diagram explanation (in text form)** — like how a hyperplane looks in 2D vs 3D — so it's easier when you paste into Word, or do you just want the formatted theory + formulas?