"Support Vector Machines (SVMs) are supervised machine learning algorithms used for both classification and regression. In classification, the goal is to find an optimal hyperplane that separates different classes with the maximum possible margin. The margin is the distance between the hyperplane and the closest data points, known as support vectors. A 'soft margin' approach allows for some misclassification to improve generalization. In regression, SVMs aim to fit a hyperplane within a defined margin of error.

To handle non-linear relationships, SVMs employ the kernel trick. This technique implicitly maps data into a higher-dimensional space, enabling the creation of non-linear decision boundaries in the original space, without explicitly calculating the transformation. Common kernel functions include linear, polynomial, and radial basis function (RBF) kernels. The 'C' parameter is used to regulate the trade off between achieving a large margin, and minimising the number of misclassified points."

 **Hyperplane Definition:**

* For classification, the hyperplane is the decision boundary.
* For regression, the hyperplane is the central line or plane of the tube that the algorithm tries to fit the data into.

 **Margin:**

* Clarify that the "margin" is the distance between the hyperplane and the closest data points (support vectors).
* Explain that a "hard margin" SVM aims for perfect separation (not always feasible), while a "soft margin" SVM allows for some errors to improve generalization.

 **Support Vectors:**

* Explicitly mention "support vectors," which are the data points closest to the hyperplane and crucial for defining it.

 **Kernel Trick Details:**

* The kernel trick allows us to compute dot products in the high-dimensional space without explicitly calculating the transformation, which is computationally efficient.
* Other common Kernels include linear Kernel.

 **Regularization:**

* Adding the concept of the C parameter that control the trade off between maximizing the margin, and minimizing the misclassification error is important.

let's break down the Radial Basis Function (RBF) and Polynomial kernels in a way that's clear and concise for an interview.

**1. Radial Basis Function (RBF) Kernel**

* **Concept:**
  + The RBF kernel measures the similarity between data points based on their distance. Essentially, it transforms the data into an infinite-dimensional space.
  + It's particularly effective for capturing complex, non-linear relationships.
* **How it Works:**
  + It calculates the similarity between two points as a function of their distance.
  + The most common form is the Gaussian RBF, which uses the Gaussian function to determine similarity.
  + The kernel's output decreases exponentially as the distance between points increases.
* **Formula (Gaussian RBF):**
  + K(x,x′)=exp(−γ∣∣x−x′∣∣2)
    - Where:
      * K(x,x′) is the kernel function.
      * x and x′ are the two data points.
      * ∣∣x−x′∣∣2 is the squared Euclidean distance between x and x′.
      * γ (gamma) is a parameter that controls the width of the Gaussian function. A larger gamma makes the influence of a single point smaller, and a smaller gamma makes it further.
* **Key Points:**
  + Gamma (γ) is a crucial parameter. It determines the influence of each data point.
  + RBF can create very complex decision boundaries.
  + It is very commonly used.

**2. Polynomial Kernel**

* **Concept:**
  + The polynomial kernel computes the similarity between data points by raising their dot product to a certain power.
  + It allows SVMs to model polynomial relationships in the data.
* **How it Works:**
  + It essentially adds polynomial features to the data.
  + The degree of the polynomial is a parameter that controls the complexity of the model.
* **Formula:**
  + K(x,x′)=(x⋅x′+c)d
    - Where:
      * K(x,x′) is the kernel function.
      * x and x′ are the two data points.
      * x⋅x′ is the dot product of x and x′.
      * c is a constant (often 0 or 1).
      * d is the degree of the polynomial.
* **Key Points:**
  + The degree (d) is a crucial parameter. It determines the complexity of the polynomial.
  + Higher degrees can lead to more complex decision boundaries but also increase the risk of overfitting.
  + The constant c controls the influence of higher order versus lower order terms.

**In an Interview:**

When explaining these kernels, emphasize:

* **The fundamental idea:** What kind of similarity does each kernel measure?
* **The parameters:** What are the key parameters, and how do they affect the model?
* **When to use them:** What types of data or problems are each kernel best suited for?
* **The computational efficiency** that the kernel trick provides.
* **Avoid overly complex mathematical jargon:** Focus on conveying the concepts clearly.