Mine Automation and Data Analytics

Prof. Radhakanta Koner

Department of Mining Engineering

IIT (ISM) Dhanbad

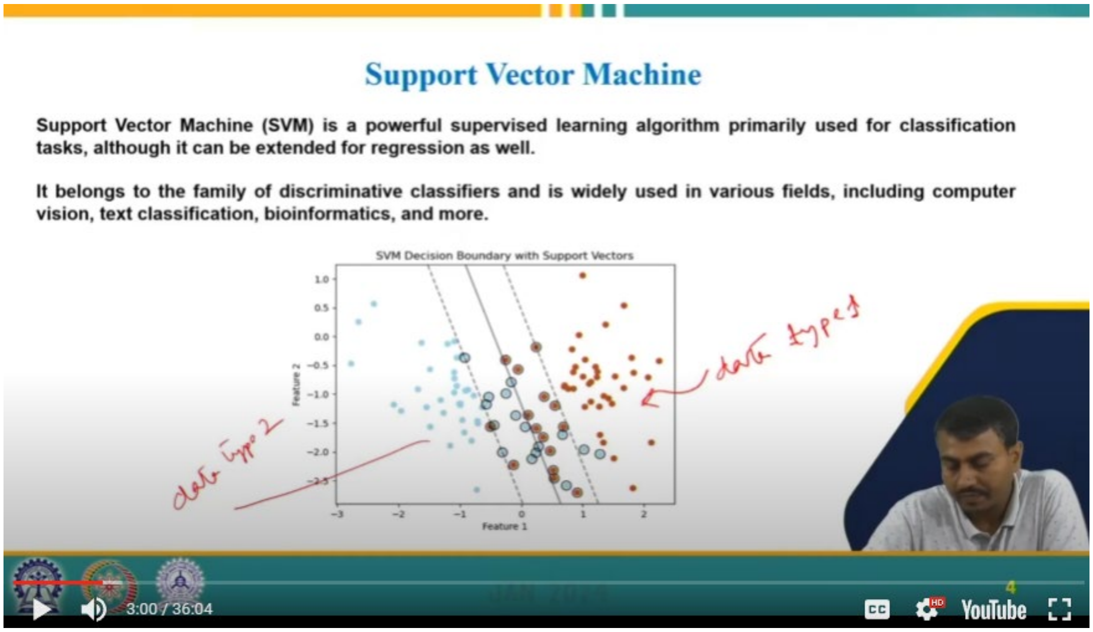
Week - 10

Lecture 49

Support Vector Machine

Welcome to my class, Mine, Automation and Data Analytics. Today, we will discuss the support vector machine. It is also a supervised machine learning method and one of the most popular classifiers machine learning communities use for feature extraction, image processing, and automated entry and exit planning at the industry gate. So, there are many applications for these support vector machines. So, in this lesson, let us discuss the theoretical aspect and the detailed features of the support vector machine. So, in this lecture, we will discuss the concept of a support vector machine because it is a unique idea and a straightforward method for classifying two data types convincingly using a margin classifier.

We will then discuss the assumptions. When these classifications are not possible in a linear space, we will use different kinds of kernel functions: linear kernel function, polynomial kernel function, and radial basis function. So, we will also cover and elaborate on the advantages and disadvantages of the support vector machine concept and methods. Finally, we will discuss some potential applications of support vector machines. So you can see that these are the data sets; this is the data sparsely distributed. This is data type 1, and it is another data type 2. And if you see, it is tough to separate all these points convincingly through some line, that will show that in that boundary, data type 1 is lying on one side, and data type 2 is lying on the other.



So, for this kind of problem, the support vector machine is one of the most powerful supervised learning algorithms for this classification. Though it is widely used for classification, many examples have also been used for regression. So, in the computer vision community, text classification, and bioinformatics, wide applications use this classifier algorithm to differentiate and subdivide the two data types convincingly through the margin classifier. So, a new term we will introduce is the hyperplane. So, this hyperplane best separates the data points of different classes in an n-dimensional space.

So think about that data are three-dimensional. It has X, Y, and Z, and data are distributed in a three-dimensional space. So, the hyperplane for this three-dimensional space data is nothing but a 2D plane. The 2D plane may be horizontal, inclined, or vertical.

So, a 2D plane using the support vector machine concept can convincingly separate the data points into two distinct categories. This can be done in a few steps only using these nearby support vectors nearest the hyperplane. So what is the advantage in that case? In that case, it creates a new hyperplane to separate the data points into two classes using only a few support vectors that lie closely to the hyperplane. That makes it a very efficient and cost-effective algorithm for segregating and classifying two data classes. And also, through this method, we are looking at maximizing the margin.

So, for 3D data points, the plane separating the data has the maximum separation, ensuring there is a maximum separation between these two data classes. Nearby points lying to the hyperplane on both sides are the support vectors. So, when the data points are not linearly separable, we will use the kernel trick to map the data into the higher dimensional space where it can be linearly separated. So this is the fun done. So, for 3D data points, we are separating using a plane.

For a 2D space XY plane, we separate the data using a hyperplane line. So, let us discuss some of the critical concepts of the support vector machine. The hyperplane is nothing, but this is the hyperplane that separates two types of data sets and the maximum margin hyperplane. And the points lying close to the hyperplane are the support vectors. The hyperplane is the decision boundary separating the data points of different classes in n-dimensional space.

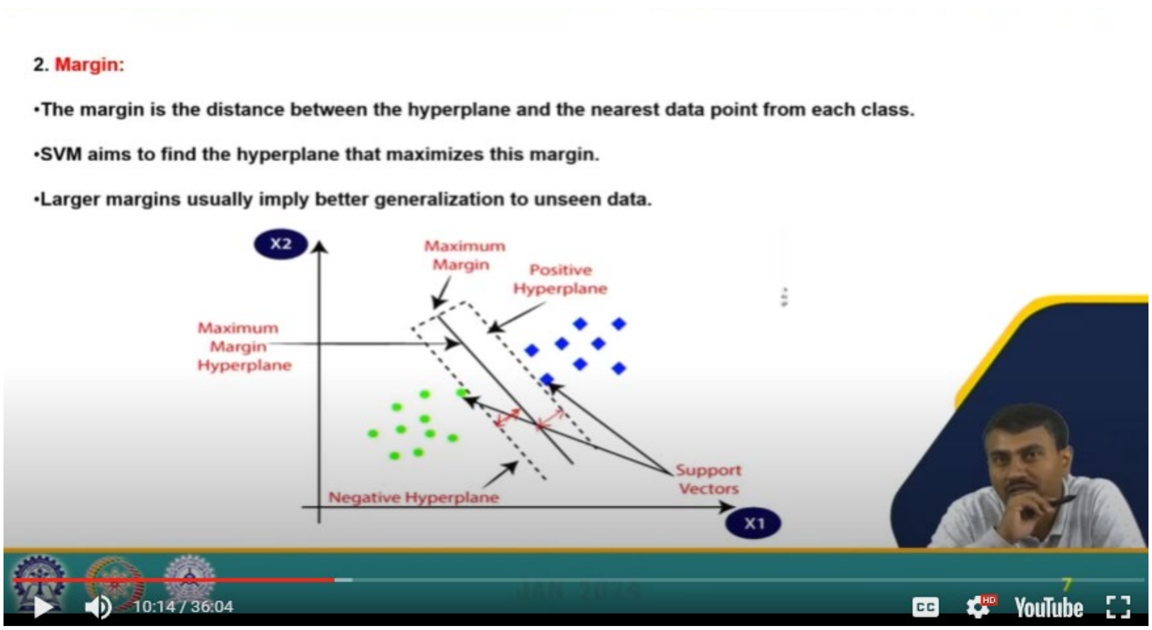
For example, this is two-dimensional space. Here, our board is 2D. So, we use 2D, two-dimensional data in our example. So similarly, I have already enumerated the 3D data points; we need a plane. So 3-1 2D plane is 2D.

For 2D data points, one line, the line is 1D. So, in the binary classification problem, the hyperplane is an n-1 dimensional subspace in the n-dimensional feature space. So, for a three-dimensional feature space, it is a two-dimensional subspace that is a plane. For a two-dimensional data feature, 2-1 is one-dimensional subspace that is a line. So that can be represented using this equation: Wx plus b is equal to zero.

$$\mathbf{w} \cdot \mathbf{x} + b = 0$$

So, W is the weight vector perpendicular to the hyperplane, and x is the input vector. This input vector represents the input vectors from the positive side of the data points, the negative side of the data points, and b is the bias term. These are the perpendicular distances that we are measuring through both sides. So, both are equal. So, iteratively, we will find precisely the maximum margin, their hyperplane, and the support vectors through iterative steps.

This margin is the distance between the hyperplane and the nearest data points from each class. So this is the margin, this is the margin. If the gamma is the margin, two gamma is the total distance between the two different sets of data. So, the SVM aims to find the hyperplane that maximizes this margin and seeks to maximize this margin. This is the objective of the support vector machine.



So, the more considerable margin usually implies better generalization to unseen data. This is the advantage. So, the support vector is the data point. So, these are the support vectors, the data points that lie closest to the decision boundary. These are the nearest points that lie to the decision boundary hyperplane.

These are critical elements in defining the decision boundary and determining the margin. These support vectors influence the position and orientation of the hyperplane. The decision boundary is solely determined by the support vectors, making SVM memory efficient exclusively determined by the support vector. The support vector solely determines the decision boundary, so we take only a few data points here.

That makes SVM memory efficient, so we will not be able to handle it when we cannot classify or separate the data points using the linear classifier or the linear boundary. We use nonlinear, sometimes linear, functions for the linear case when we cannot separate the data points through the linear lines. We will use the kernel's help. So, the kernel is nothing but the dot product between the input vectors in the higher dimensional space without explicitly transforming them.

So, three kinds of variants are the linear kernel; the second is a polynomial kernel, and the radial basis function kernel, RDF; for which kind of data, which kind of kernel will we select that we need to check the parameters? Based on that, we must choose which kernel to choose. For example, the green dot points are sparsely distributed around the red points. What will happen if I transform these data points into a higher dimensional space and increase the height of this particular side? These data points will be up, and these data points will be segregated on the down. So, we can separate these data points convincingly by plane.

So, I cannot separate these data points using a line. Still, by transforming the data points into a higher dimensional space, I can also separate these data through a hyperplane. This is the flexibility and beauty of the support vector machine and its use of kernel functions. Optimization. So, SVM aims to find the optimum hyperplane that maximizes the margin while minimizing the classification error, and this optimization problem is often formulated as a quadratic programming QP or a convex optimization problem. Techniques like gradient descent, sequential minimization, minimal optimization, SMO, or interior point methods are commonly used to solve this optimization problem efficiently.

So, what are the assumptions? Because this is a method, this is a mathematical method. So, these mathematical methods also have some critical assumptions, and based on the assumptions, we proceed to the next stage of processing. So, the first one is the linear separability for linear SVM. So, the original formulation of the SVM assumes that the classes can be separated by a linear decision boundary hyperplane in the input space. So when the classes are not linearly separable, SVM aims to find the hyperplane that maximizes margin while minimizing the classification error, which might lead to a soft margin SVM formulation or a nonlinear kernel.

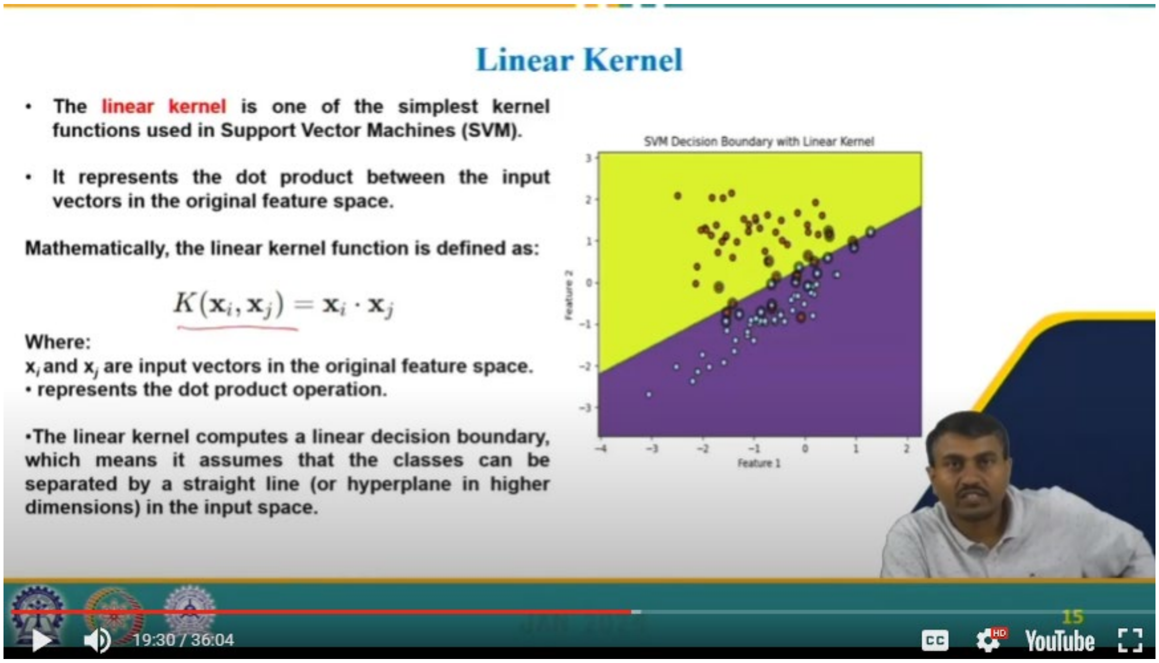
Margin maximization. SVM aims to find the hyperplane that maximizes the margin, the distance between the hyperplane and the nearest data point from each class, and the support vector. Support vectors are lying close to the hyperplane. So, maximizing this margin helps SVM generalize well to unforeseen or unseen data and enhances its robustness against outliers, such as kernel function selection. When using nonlinear

SVM, that is, polynomial kernel or RBF, the choice of kernel function and its parameters is crucial for the degree of polynomial γ for the RBF.

So, these are to be chosen judiciously and efficiently for better optimization—noisy or outlier-free data. SVM is very sensitive to noise and outliers in the data, especially in the case of hard-margin SVM. Soft margins have some mitigating measures. Outliers can significantly affect the position and orientation of the decision boundary.

So techniques such as soft margin SVM or robust kernel function RBF kernel can help mitigate the impact of outliers—feature scaling. The scale of the input features can influence SVM performance. So, it is advisable to scale the feature to a similar range using standardization or normalization. That will help this model perform better, and the decision boundary can be easily achieved.

Binary classification. SVM is inherently a binary classifier. This means that data points are separated into two classes. So, multi-class classification techniques such as one versus one or one versus all strategies can be used, and multiple binary classifiers can be trained and combined to make predictions for various classes—sparse solutions. SVM often yields sparse solutions, meaning that only a subset of training data that is a support vector contributes to defining the decision boundary.

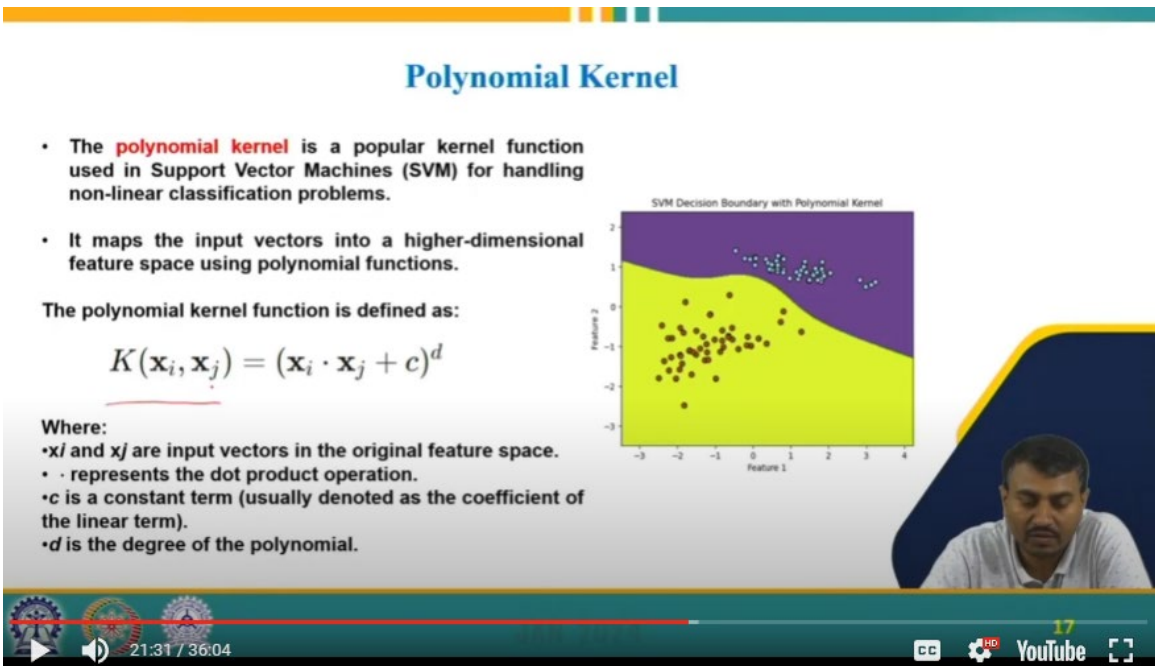


So, this property makes SVM memory efficient and allows it to handle high-dimensional data efficiently. So, these understandings are essential to apply the SVM in the problem we will deal with for industry relevance or some industry purpose. So, let us discuss the linear classifier. So, the linear classifier uses the linear kernel concept. So, linear kernel is one of the most straightforward functions in support vector machines. So, it represents the dot product between the input and input vectors in the original feature space. So mathematically, Kij is equal to xi into xj, which is just a multiplication of the input data. So, xi and xj are the original feature space's input vectors representing the dot product operation. So, the linear kernel computes a linear decision boundary, assuming that the class can be separated by a straight line or hyperplane in a higher dimension in the input space.

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \cdot \mathbf{x}_j$$

The advantage. The significant advantage is the computational efficiency. The linear kernel is computationally less expensive than nonlinear kernels like polynomial or RGF kernel. Interpretability. Linear SVM with linear kernel provides straightforward decision boundaries that are easier to interpret and separate the data point into two distinct classes. However, the linear kernel needs to be improved in its use and ability to capture complex relationships in the data.

So, if the classes are not linearly separable, using a linear kernel may result in poor classification performance. So, in such cases, we have to use the nonlinear kernel, like polynomial or RGF, which are often used in the remote sense in the ML community for higher dimensional space where nonlinear relationships can be captured. Polynomial kernel. The polynomial kernel is a popular function in support vector machines to handle nonlinear classification problems. So, it maps the input vectors into a higher dimensional feature space using a polynomial function.



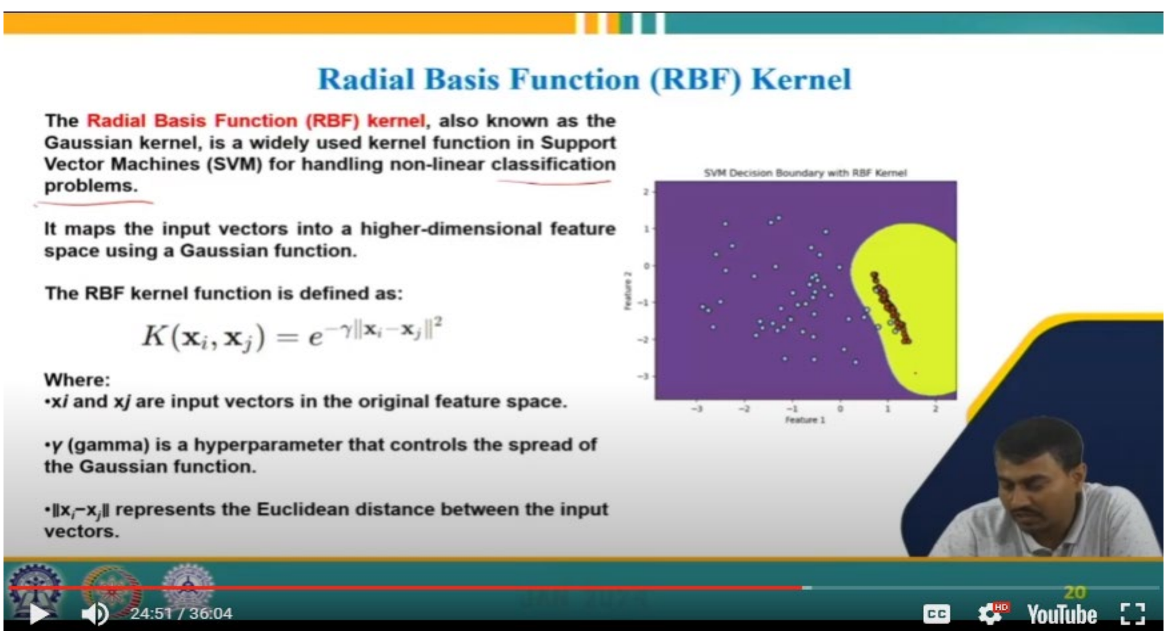
So, the kernel function K Xij Xj is equal to the Xi Xj dot product of the input vector plus C and is total to the power D. So, these D represent the degree of the polynomial. Square is the 2 degree, 3 is the cube degree like that. C indicates the constant term, usually denoted as the coefficient of the linear term. So, these data points are segregated and separated using a curve.

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + c)^d$$

So, the polynomial kernel allows SVM to capture nonlinear decision boundaries by transforming the input space into a higher dimensional space where classes might become linearly separable. So here are two critical points for the polynomial kernel. One is the degree, which is the D. The degree of the polynomial determines the complexity of the decision boundary. Higher degrees allow for more complex decision boundaries but may also lead to overfitting.

Coefficient C affects the importance of higher-degree features compared to lower-degree features. It helps control the influence of higher-order terms in the polynomial. What is the advantage? The advantage of the polynomial kernel includes its flexibility, which allows it to capture complex nonlinear relationships in the data and control over complexity. So, by adjusting the degree parameter, one can control the complexity of the decision boundary. However, choosing the appropriate degree D and the coefficient value C is crucial for achieving good classification performance.

A higher degree polynomial may lead to overfitting, while a lower degree may result in underfitting. So there should be a tradeoff, a balance between higher and lower degrees. Additionally, the computational complexity increases with a higher degree of polynomial kernel, which can impact training time, especially for large datasets. Therefore, parameter tuning and cross-validation are essential for the optimization and performance of SVM with polynomial kernels. So another polynomial or nonlinear kernel is the radial basis function, that is RBF also known as Gaussian kernel is a widely used kernel function in support vector machine for handling nonlinear classification problems.



It maps the input vector into higher-dimension feature space using a Gaussian function. So here, these are the data points; separating them using a linear classifier is tough. So, the RBM function does that X and Xj are basically the input vectors. So it makes the kernel Xij, Xi, and Xj equal to e in the power minus gamma Xi minus Xj difference in the power square. So, Xi minus Xj square represents an Euclidean distance between the input vectors.

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$$

And gamma represents the hyperparameter that controls the spread of the Gaussian function. So here, the value of the gamma is essential to choose, and its value is very, very sensitive to the performance of this RBF kernel. So, the RBF kernel allows the capture of complex nonlinear decision boundaries by implicitly mapping the input space into an infinite dimensional feature space. It is called a radial basis function because its value depends only on the distance between the input vectors, and it decreases radially as the distance from the reference point usually suffered vector increases. So, the gamma parameter determines the influence of each training sample.

For example, a smaller value of gamma results in a more extensive range of influence and a smoother decision boundary. A more considerable value makes the decision boundary more irregular and closely fitted to the training data. It controls the flexibility of the decision boundary. Implicit feature mapping, unlike the polynomial kernel, the RBF kernel implicitly maps the data into an infinite dimensional space, making it very flexible in capturing complex decision boundaries. What are the advantages? It can capture highly complex nonlinear relationships in the data, that is, its higher degree of flexibility. Second, the RBF kernel can effectively handle a wide range of data distribution and its versatility.

However, choosing the appropriate value for the gamma parameter is crucial for achieving good classification performance. A poorly chosen value of gamma can lead to overfitting or underfitting. So additionally, the computational complexity of SVM with RBF kernel increases significantly with the size of the dataset as it involves computing pairwise distances between all data points. Therefore, parameter training and model validation are essential for optimizing the performance of SVM with the RBF kernel. What are the advantages and disadvantages of SVM? One significant advantage of SVM is that it is effective in high-dimensional space.

SVM performs well even in high-dimensional space, making it suitable for tasks with many features, such as image classification or text categorization. Memory efficient, SVM uses a subset of the training points only the support vectors to define the decision boundary. So, this property makes SVM memory efficient, especially for large datasets. Versatility: SVM supports different kernel functions, allowing it to handle various data distributions and nonlinear relationships. Standard kernels include linear, polynomial, and error basis functions.

Due to its ability to maximize the margin between classes, SVM is less prone to overfitting, especially in high dimensional space. Effective with small datasets, SVM can produce accurate results with relatively small training datasets, making it suitable for tasks where data collection is expensive or time-consuming. Works well with nonlinear data using appropriate kernel functions. SVM can efficiently model complex nonlinear decision boundaries. Global optimum: SVM aims to find the global optimum solutions that are hyperplanes with the maximum margin, which leads to better generalization performance. What are the disadvantages? It is computationally intensive, SVM can be computationally intensive for large datasets, and it requires solving a quadratic programming problem. It is sensitive to parameter tuning. SVM performance is sensitive to the choice of parameters for γ, C, and D, such as the regulation parameter C and kernel parameters.

So, selecting appropriate values for these parameters can be challenging and require extensive experimentation. Limited interpretability: The SVM model can be difficult to interpret, especially when using complex kernel functions or in higher dimensional space. So, understanding the relationship between input features and the decision boundary can be complicated. Slow training time: Training an SVM model can take a long, especially for large datasets or when using a nonlinear kernel. Additionally, the complexity of training time can increase significantly the number of support vectors.

Not suitable for large datasets, SVM may not be well suited to large datasets due to its computational complexity and memory requirement. No probabilistic output, SVM does not provide direct probabilistic interpretation of class membership, unlike some other classifiers such as logistic regression. So, it is also sensitive to noise; SVM performance can degrade significantly in the presence of noisy data or outliers, especially when using hard merging SVM. So, understanding these advantages and disadvantages will help us understand how we can use this SVM, how to mitigate these problems and the limitations of a particular problem-solving situation. So, let us discuss some potential uses of these support vector machines in the mining industry and their capability to handle complex issues.

The first one is mineral identification and classification. SVM is used for mineral identification and classification in mining exploration and mineral processing. So, by analyzing spectroscopic data obtained from various sensors such as X-ray fluorescence, XRF, near-infrared NIR, and hyperspectral imaging, the SVM model can classify different minerals or mineral compositions in a rock sample. So this helps identify potential mineral deposits and optimize mineral processing operations. Fault detection and predictive maintenance because we are using costly machines. So, predicting the fault, predicting the downtime, or reducing the downtime helps to utilize this machine to its maximum.

So, SVM can be employed to detect faults and perform predictive maintenance on mining equipment and machinery. The SVM model can detect abnormal patterns indicative of equipment faults or failure by analyzing the various sensor data, such as vibration, temperature, and pressure readings from different mining equipment. So, this early detection of faults allows for proactive maintenance intervention, minimizing downtime and maximizing operational efficiency. Overgrade estimation and SVM are also utilized for overgrade estimation in mining operations. So, by analyzing geological data, drill core samples, and assay results, the SVM model can predict the grade and quality of ore deposits in different mine sites.

So, accurate overgrade estimation enables mining companies to optimize resource extraction, mine planning, and production scheduling, leading to cost savings and increased profitability through mine safety and risk assessment. SVM can play a crucial role in the safety aspect and prediction of some failure or some hazard very well using past data and classifying the potentially hazardous data that are potentially hazardous or not hazardous based on past experiences very quickly: stockpile management and inventory control. Based on the volume data on the sites, we can very well manage using the data to classify the grades and their aggregates and efficiently handle the inventory at the site level in the mine.

So, these are the few examples that we have shared. There are other applications in the support vector machines, as well. For instance, we can use a support vector machine in an automated entry-exit system in a mine. Those are not allowed. They can easily differentiate using this classifier. So, the SVM mechanism can also control automatic entry and exit.

So these are the references. So, let me summarize a few sentences covered in this lesson. So, we have covered the support vector machine, its fundamental concept, and the assumptions, and we have also discussed some kernel tricks that we often use to classify the data in linear classifying methods using linear kernel, polynomial kernel, and radial basis function kernel. We have also discussed the advantages and disadvantages of support vector machines. Lastly, we discussed its potential applications in the mining industry. Thank you.