**How To Learn Machine Learning Algorithms For Interviews**

**SVM**

Theoretical Understanding:

1. <https://www.youtube.com/watch?v=H9yACitf-KM>
2. <https://www.youtube.com/watch?v=Js3GLb1xPhc>

***1. What Are the Basic Assumption?***

There are no such assumptions

***2. Advantages***

1. SVM is more effective in high dimensional spaces.
2. SVM is relatively memory efficient.
3. SVM’s are very good when we have no idea on the data.
4. Works well with even unstructured and semi structured data like text, Images and trees.
5. The kernel trick is real strength of SVM. With an appropriate kernel function, we can solve any complex problem.
6. SVM models have generalization in practice, the risk of over-fitting is less in SVM.

***3. Disadvantages***

1. More Training Time is required for larger dataset
2. It is difficult to choose a good kernel function <https://www.youtube.com/watch?v=mTyT-oHoivA>
3. The SVM hyper parameters are Cost -C and gamma. It is not that easy to fine-tune these hyper-parameters. It is hard to visualize their impact

***4. Whether Feature Scaling is required?***

Yes

***5. Impact of Missing Values?***

Although SVMs are an attractive option when constructing a classifier, SVMs do not easily accommodate missing covariate information. Similar to other prediction and classification methods, in-attention to missing data when constructing an SVM can impact the accuracy and utility of the resulting classifier.

***6. Impact of outliers?***

It is usually sensitive to outliers <https://arxiv.org/abs/1409.0934#:~:text=Despite%20its%20popularity%2C%20SVM%20has,causes%20the%20sensitivity%20to%20outliers>.

***Types of Problems it can solve(Supervised)***

1. Classification
2. Regression

***Overfitting And Underfitting***

In SVM, to avoid overfitting, we choose a Soft Margin, instead of a Hard one i.e. we let some data points enter our margin intentionally (but we still penalize it) so that our classifier don't overfit on our training sample

<https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>

***Different Problem statement you can solve using Naive Baye's***

1. We can use SVM with every ANN usecases
2. Intrusion Detection
3. Handwriting Recognition

**Practical Implementation**

1. <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>
2. <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html>

***Performance Metrics***[***¶***](https://render.githubusercontent.com/view/ipynb?commit=655ac129548c3d8502334fbdf9322cf1bf425126&enc_url=68747470733a2f2f7261772e67697468756275736572636f6e74656e742e636f6d2f6b726973686e61696b30362f496e746572766965772d50726570617274696f6e2d446174612d536369656e63652f363535616331323935343863336438353032333334666264663933323263663162663432353132362f496e746572766965772532305072657061726174696f6e2d253230446179253230332d53564d2e6970796e62&nwo=krishnaik06%2FInterview-Prepartion-Data-Science&path=Interview+Preparation-+Day+3-SVM.ipynb&repository_id=297263348&repository_type=Repository#Performance-Metrics)

***Classification***

1. Confusion Matrix
2. Precision,Recall, F1 score

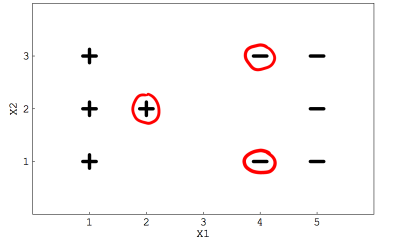
***Regression***

1. R2,Adjusted R2
2. MSE,RMSE,MAE

## Skill test Questions and Answers

**Question Context: 1 – 2**

Suppose you are using a Linear SVM classifier with 2 class classification problem. Now you have been given the following data in which some points are circled red that are representing support vectors.

[](https://cdn.analyticsvidhya.com/wp-content/uploads/2017/09/01154204/Image_18-19.png)

**1) If you remove the following any one red points from the data. Does the decision boundary will change?**

A) Yes  
B) No

[](https://id.analyticsvidhya.com/auth/login/?next=https://www.analyticsvidhya.com/blog/2017/10/svm-skilltest/?&utm_source=question-answer-blog&source=qa-blog)

**2) [True or False] If you remove the non-red circled points from the data, the decision boundary will change?**

A) True  
B) False

[](https://id.analyticsvidhya.com/auth/login/?next=https://www.analyticsvidhya.com/blog/2017/10/svm-skilltest/?&utm_source=question-answer-blog&source=qa-blog)

**3) What do you mean by generalization error in terms of the SVM?**

A) How far the hyperplane is from the support vectors  
B) How accurately the SVM can predict outcomes for unseen data  
C) The threshold amount of error in an SVM

[](https://id.analyticsvidhya.com/auth/login/?next=https://www.analyticsvidhya.com/blog/2017/10/svm-skilltest/?&utm_source=question-answer-blog&source=qa-blog)

**4) When the C parameter is set to infinite, which of the following holds true?**

A) The optimal hyperplane if exists, will be the one that completely separates the data  
B) The soft-margin classifier will separate the data  
C) None of the above

[](https://id.analyticsvidhya.com/auth/login/?next=https://www.analyticsvidhya.com/blog/2017/10/svm-skilltest/?&utm_source=question-answer-blog&source=qa-blog)

**5) What do you mean by a hard margin?**

A) The SVM allows very low error in classification  
B) The SVM allows high amount of error in classification  
C) None of the above

[](https://id.analyticsvidhya.com/auth/login/?next=https://www.analyticsvidhya.com/blog/2017/10/svm-skilltest/?&utm_source=question-answer-blog&source=qa-blog)

**6) The minimum time complexity for training an SVM is O(n2). According to this fact, what sizes of datasets are not best suited for SVM’s?**

A) Large datasets  
B) Small datasets  
C) Medium sized datasets  
D) Size does not matter

**Solution: A**

Datasets which have a clear classification boundary will function best with SVM’s.

**7) The effectiveness of an SVM depends upon:**

A) Selection of Kernel  
B) Kernel Parameters  
C) Soft Margin Parameter C  
D) All of the above

**Solution: D**

The SVM effectiveness depends upon how you choose the basic 3 requirements mentioned above in such a way that it maximises your efficiency, reduces error and overfitting.

**8) Support vectors are the data points that lie closest to the decision surface.**

A) TRUE  
B) FALSE

**Solution: A**

They are the points closest to the hyperplane and the hardest ones to classify. They also have a direct bearing on the location of the decision surface.

**9) The SVM’s are less effective when:**

A) The data is linearly separable  
B) The data is clean and ready to use  
C) The data is noisy and contains overlapping points

**Solution: C**

When the data has noise and overlapping points, there is a problem in drawing a clear hyperplane without misclassifying.

**10) Suppose you are using RBF kernel in SVM with high Gamma value. What does this signify?**

A) The model would consider even far away points from hyperplane for modeling  
B) The model would consider only the points close to the hyperplane for modeling  
C) The model would not be affected by distance of points from hyperplane for modeling  
D) None of the above

**Solution: B**

The gamma parameter in SVM tuning signifies the influence of points either near or far away from the hyperplane.

For a low gamma, the model will be too constrained and include all points of the training dataset, without really capturing the shape.

For a higher gamma, the model will capture the shape of the dataset well.

**11) The cost parameter in the SVM means:**

A) The number of cross-validations to be made  
B) The kernel to be used  
C) The tradeoff between misclassification and simplicity of the model  
D) None of the above

**Solution: C**

The cost parameter decides how much an SVM should be allowed to “bend” with the data. For a low cost, you aim for a smooth decision surface and for a higher cost, you aim to classify more points correctly. It is also simply referred to as the cost of misclassification.

**12)**

Suppose you are building a SVM model on data X. The data X can be error prone which means that you should not trust any specific data point too much. Now think that you want to build a SVM model which has quadratic kernel function of polynomial degree 2 that uses Slack variable C as one of it’s hyper parameter. Based upon that give the answer for following question.

**What would happen when you use very large value of C(C->infinity)?**

**Note: For small C was also classifying all data points correctly**

A) We can still classify data correctly for given setting of hyper parameter C  
B) We can not classify data correctly for given setting of hyper parameter C  
C) Can’t Say  
D) None of these

**Solution: A**

For large values of C, the penalty for misclassifying points is very high, so the decision boundary will perfectly separate the data if possible.

**13) What would happen when you use very small C (C~0)?**

A) Misclassification would happen  
B) Data will be correctly classified  
C) Can’t say  
D) None of these

**Solution: A**

The classifier can maximize the margin between most of the points, while misclassifying a few points, because the penalty is so low.

**14) If I am using all features of my dataset and I achieve 100% accuracy on my training set, but ~70% on validation set, what should I look out for?**

A) Underfitting  
B) Nothing, the model is perfect  
C) Overfitting

**Solution: C**

If we’re achieving 100% training accuracy very easily, we need to check to verify if we’re overfitting our data.

**15) Which of the following are real world applications of the SVM?**

A) Text and Hypertext Categorization  
B) Image Classification  
C) Clustering of News Articles  
D) All of the above

**Solution: D**

SVM’s are highly versatile models that can be used for practically all real world problems ranging from regression to clustering and handwriting recognitions.

**Question Context: 16 – 18**

Suppose you have trained an SVM with linear decision boundary after training SVM, you correctly infer that your SVM model is under fitting.

**16) Which of the following option would you more likely to consider iterating SVM next time?**

A) You want to increase your data points  
B) You want to decrease your data points  
C) You will try to calculate more variables  
D) You will try to reduce the features

**Solution: C**

The best option here would be to create more features for the model.

**17) Suppose you gave the correct answer in previous question. What do you think that is actually happening?**

1. We are lowering the bias  
2. We are lowering the variance  
3. We are increasing the bias  
4. We are increasing the variance

A) 1 and 2  
B) 2 and 3  
C) 1 and 4  
D) 2 and 4

**Solution: C**

Better model will lower the bias and increase the variance

**18) In above question suppose you want to change one of it’s(SVM) hyperparameter so that effect would be same as previous questions i.e model will not under fit?**

A) We will increase the parameter C  
B) We will decrease the parameter C  
C) Changing in C don’t effect  
D) None of these

**Solution: A**

Increasing C parameter would be the right thing to do here, as it will ensure regularized model

**19) We usually use feature normalization before using the Gaussian kernel in SVM. What is true about feature normalization?**

1. We do feature normalization so that new feature will dominate other  
2. Some times, feature normalization is not feasible in case of categorical variables  
3. Feature normalization always helps when we use Gaussian kernel in SVM

A) 1  
B) 1 and 2  
C) 1 and 3  
D) 2 and 3

**Solution: B**

Statements one and two are correct.

**Question Context: 20-22**

Suppose you are dealing with 4 class classification problem and you want to train a SVM model on the data for that you are using One-vs-all method. Now answer the below questions?

**20) How many times we need to train our SVM model in such case?**

A) 1  
B) 2  
C) 3  
D) 4

**Solution: D**

For a 4 class problem, you would have to train the SVM at least 4 times if you are using a one-vs-all method.

**21) Suppose you have same distribution of classes in the data. Now, say for training 1 time in one vs all setting the SVM is taking 10 second. How many seconds would it require to train one-vs-all method end to end?**

A) 20  
B) 40  
C) 60  
D) 80

**Solution: B**

It would take 10×4 = 40 seconds

**22) Suppose your problem has changed now. Now, data has only 2 classes. What would you think how many times we need to train SVM in such case?**

A) 1  
B) 2  
C) 3  
D) 4

**Solution: A**

Training the SVM only one time would give you appropriate results

**Question context: 23 – 24**

Suppose you are using SVM with linear kernel of polynomial degree 2, Now think that you have applied this on data and found that it perfectly fit the data that means, Training and testing accuracy is 100%.

**23) Now, think that you increase the complexity(or degree of polynomial of this kernel). What would you think will happen?**

A) Increasing the complexity will overfit the data  
B) Increasing the complexity will underfit the data  
C) Nothing will happen since your model was already 100% accurate  
D) None of these

**Solution: A**

Increasing the complexity of the data would make the algorithm overfit the data.

**24) In the previous question after increasing the complexity you found that training accuracy was still 100%. According to you what is the reason behind that?**

1. Since data is fixed and we are fitting more polynomial term or parameters so the algorithm starts memorizing everything in the data  
2. Since data is fixed and SVM doesn’t need to search in big hypothesis space

A) 1  
B) 2  
C) 1 and 2  
D) None of these

**Solution: C**

Both the given statements are correct.

**25) What is/are true about kernel in SVM?**

1. Kernel function map low dimensional data to high dimensional space  
2. It’s a similarity function

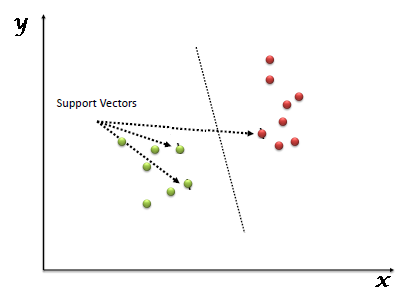
A) 1  
B) 2  
C) 1 and 2  
D) None of these

**Solution: C**

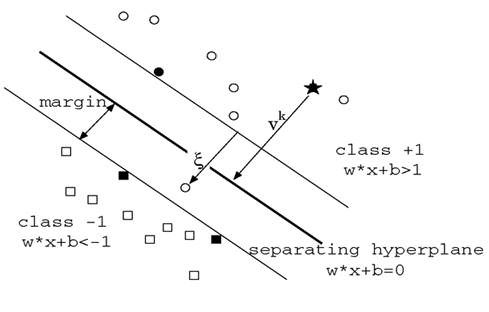
Both the given statements are correct.

1.Explain SVM machine learning algorithm in detail.

SVM stands for support vector machine, it is a supervised machine learning algorithm which can be used for both Regression and Classification. If you have n features in your training data set, SVM tries to plot it in n-dimensional space with the value of each feature being the value of a particular coordinate. SVM uses hyper planes to separate out different classes based on the provided kernel function.



2. What are support vectors in SVM.



In the above diagram we see that the thinner lines mark the distance from the classifier to the closest data points called the support vectors (darkened data points). The distance between the two thin lines is called the margin.

3. What are the different kernels functions in SVM ?

There are four types of kernels in SVM.

1. Linear Kernel
2. Polynomial kernel
3. Radial basis kernel
4. Sigmoid kernel

*4.Give some situations where you will use an SVM over a RandomForest Machine Learning algorithm and vice-versa?*

     The performance depends on many factors

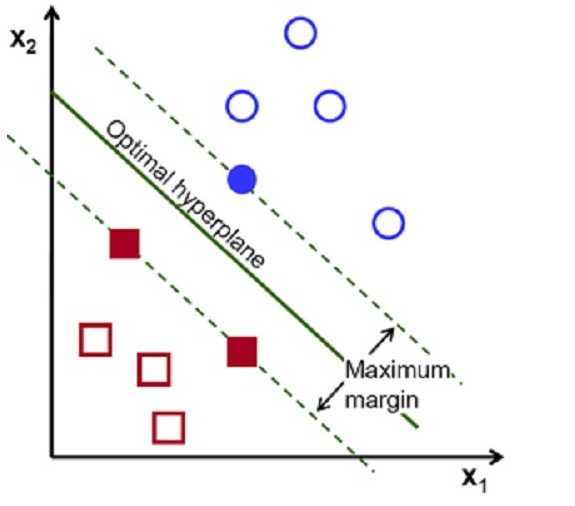
* the number of training instances
* the distribution of the data
* linear vs. non-linear problems
* input scale of the features
* the chosen hyperparameters
* how you validate/evaluate your model

In general, It is easier to train a well-performing Random Forest classifier since you have to worry less about hyperparameter optimization. Due to the nature Random Forests, you are less likely to overfit. You simply grow *n*trees on *n* bootstrap samples of the training set on feature subspaces — using the majority vote, the estimate will be pretty robust.

Using Support Vector Machines, you have “more things” to “worry” about such as choosing an appropriate kernel (poly, RBF, linear …), the regularization penalty, the regularization strength, kernel parameters such as the poly degree or gamma, and so forth.

So, in sum, We can say that Random Forests are much more automated and thus “easier” to train compared to SVMs, but there are many examples in literature where SVMs outperform Random Forests and vice versa on different datasets. So, if you like to compare these two, make sure that you run a large enough grid search for the SVM and use nested cross-validation to reduce the performance estimation bias.

5. *Why SVM is an example of a large margin classifier?*



* SVM is a type of classifier which classifies positive and negative examples, here blue and red data points
* As shown in the image, the largest margin is found in order to avoid overfitting ie,.. the optimal hyperplane is at the maximum distance from the positive and negative examples(Equal distant from the boundary lines).
* To satisfy this constraint, and also to classify the data points accurately, the margin is maximised, that is why this is called the large margin classifier.

6. *What is the role of C in SVM?*

             The C parameter tells the SVM optimization how much you want to avoid misclassifying each training example. For large values of C, the optimization will choose a smaller-margin hyperplane if that hyperplane does a better job of getting all the training points classified correctly. Conversely, a very small value of C will cause the optimizer to look for a larger-margin separating hyperplane, even if that hyperplane misclassifies more points. For very tiny values of C, you should get misclassified examples, often even if your training data is linearly separable.

7. *What is the intuition of a large margin classifier?*

           Let’s say you’ve found a hyperplane that completely separates the two classes in your training set. We expect that when new data comes along (i.e. your test set), the new data will look like your training data. Points that should be classified as one class or the other should lie near the points in your training data with the corresponding class. Now, if your hyperplane is oriented such that it is close to some of the points in your training set, there’s a good chance that the new data will lie on the wrong side of the hyperplane, even if the new points lie close to training examples of the correct class.

So we say that we want to find the hyperplane with the maximum margin. That is, find a hyperplane that divides your data properly, but is also as far as possible from your data points. That way, when new data comes in, even if it is a little closer to the wrong class than the training points, it will still lie on the right side of the hyperplane.

If your data is separable, then there are infinitely many hyperplanes that will separate it. SVM (and some other classifiers) optimizes for the one with the maximum margin, as described above.

8. *What is a kernel in SVM? Why do we use kernels in SVM?*

      SVM algorithms use a set of mathematical functions that are defined as the kernel. The function of kernel is to take data as input and transform it into the required form. Different SVM algorithms use different types of kernel functions. These functions can be different types. For example*linear, nonlinear, polynomial, radial basis function (RBF), and sigmoid.*Introduce Kernel functions for sequence data, graphs, text, images, as well as vectors. The most used type of kernel function is RBF. Because it has localized and finite response along the entire x-axis. The kernel functions return the inner product between two points in a suitable feature space. Thus by defining a notion of similarity, with little computational cost even in very high-dimensional spaces.

9. *Can we apply the kernel trick to logistic regression? Why is it not used in practice then?*

1. Classification performance is almost identical in both cases.
2. KLR (Kernal Logistic Regression) can provide class probabilities whereas SVM is a deterministic classifier.
3. KLR has a natural extension to multi-class classification whereas in SVM, there are multiple ways to extend it to multi-class classification (and it is still an area of research whether there is a version which has provably superior qualities over the others).
4. Surprisingly or unsurprisingly, KLR also has optimal margin properties that the SVMs enjoy (well in the limit at least)!

Looking at the above it almost feels like kernel logistic regression is what you should be using. However, there are certain advantages that SVMs enjoy

1. KLR is computationally more expensive than SVM — O(N3) vs O(N2k) where kk is the number of support vectors.
2. The classifier in SVM is designed such that it is defined only in terms of the support vectors, whereas in KLR, the classifier is defined over all the points and not just the support vectors. This allows SVMs to enjoy some natural speed-ups (in terms of efficient code-writing) that is hard to achieve for KLR.

10. *What is the difference between logistic regression and SVM without a kernel?*

 Only in implementation, One is much more efficient and has good optimization packages

11. *What is the difference between logistic regression and SVM*

      Logistic regression assumes that the predictors aren’t sufficient to determine the response variable, but determine a probability that is a logistic function of a linear combination of them. If there’s a lot of noise, logistic regression (usually fit with maximum-likelihood techniques) is a great technique.

On the other hand, there are problems where you have thousands of dimensions and the predictors do nearly-certainly determine the response, but in some hard-to-explicitly-program way. An example would be image recognition. If you have a grayscale image, 100 by 100 pixels, you have 10,000 dimensions already. With various basis transforms (kernel trick) you will be able to get a linear separator of the data.

Non-regularized logistic regression techniques don’t work well (in fact, the fitted coefficients diverge) when there’s a separating hyperplane, because the maximum likelihood is achieved by any separating plane, and there’s no guarantee that you’ll get the best one. What you get is an extremely confident model with poor predictive power near the margin.

SVMs get you the best separating hyperplane, and they’re efficient in high dimensional spaces. They’re similar to regularization in terms of trying to find the lowest-normed vector that separates the data, but with a margin condition that favors choosing a *good* hyperplane. A hard-margin SVM will find a hyperplane that separates all the data (if one exists) and fail if there is none; soft-margin SVMs (generally preferred) do better when there’s noise in the data.

Additionally, SVMs only consider points near the margin (support vectors). Logistic regression considers all the points in the data set. Which you prefer depends on your problem.

Logistic regression is great in a low number of dimensions and when the predictors don’t suffice to give more than a probabilistic estimate of the response. SVMs do better when there’s a higher number of dimensions, and especially on problems where the predictors do certainly (or near-certainly) determine the responses.

12. Suppose you are using RBF kernel in SVM with high Gamma value. What does this signify?

     The gamma parameter in SVM tuning signifies the influence of points either near or far away from the hyperplane.

For a low gamma, the model will be too constrained and include all points of the training dataset, without really capturing the shape.

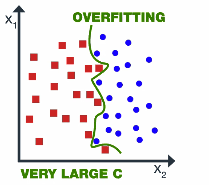
For a higher gamma, the model will capture the shape of the dataset well.

13. What is generalization error in terms of the SVM?

    Generalisation error in statistics is generally the out-of-sample error which is the measure of how accurately a model can predict values for previously unseen data.

14.**Advantages of Support Vector Machines**

**1. Avoiding Over-Fitting**



**Once the hyperplane of the vector machine has been found, apart from the points closest to the plane( support vectors), most of the other data become redundant can be omitted.**

**This implies that small changes made cannot make any significant changes to the overall data and also leave the hyper-plane also unaffected. Thus the name ‘support vector machine’ means that such algorithms tend to generalize data efficiently.**

2. Simplification of Calculation

They have comprehensive algorithms of regression which help in the classification of class data of two classes. This allows us to make our predictions and calculations simpler as the algorithm is presented in a graph image and can be used to estimate the class distinction.

Simpler visual calculation helps faster and more reliable data output rather than individually corresponding each support co-ordinate of the 2 cases.

**15.Disadvantages of Support Vector Machines**

The main disadvantages are primarily in the theory which actually only covers the determination of what the parameters will be a given set of values. Also. Kernel models can be sensitive to overfitting to the criteria of the model.

Moreover, the optimal choice of kernel often ends up to have all the data points as the supporting vector. This makes it more cumbersome to proceed with the algorithm.

Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression problems.

SVMs are particularly well suited for classification of complex but small- or medium-sized datasets.

We’ll talk about some Interview questions related to SVMs.

1. **Explain SVM to a non-technical person.**

Explanation: Suppose you have to construct a bidirectional road. Now you have to make a dividing line. The optimal approach would be to make margins on the sides and draw an equidistant line from both the margins.

Image for post



This is exactly how SVM tries to classify points by finding an optimal centre line (technically called as hyperplane).

**2. Can you explain SVM?**

Explanation: Support vector machines is a supervised machine learning algorithm which works both on classification and regression problems. It tries to classify data by finding a hyperplane that maximizes the margin between the classes in the training data. Hence, SVM is an example of a large margin classifier.

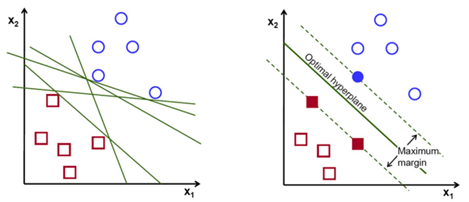
The basic idea of support vector machines:

* Optimal hyperplane for linearly separable patterns
* Extend to patterns that are not linearly separable by transformations of original data to map into new space(i.e the kernel trick)

**3. What is the geometric intuition behind SVM?**

Explanation: If you are asked to classify two different classes. There can be multiple hyperplanes which can be drawn.

Image for post

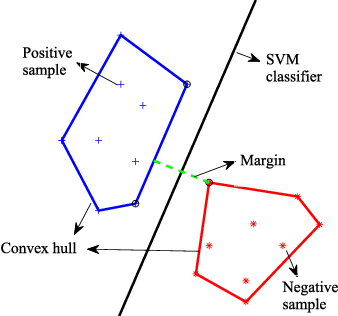


SVM chooses the hyperplane which separates the data points as widely as possible. SVM draws a hyperplane parallel to the actual hyperplane intersecting with the first point of class A (also known as Support Vectors) and another hyperplane parallel to the actual hyperplane intersecting with the first point of class B. SVM tries to maximize these margins. Eventually, this margin maximization improves the model’s accuracy on unseen data.

**4. How would explain Convex Hull in light of SVMs?**

Explanation: We simply build a convex hull for class A and class B and draw a perpendicular on the shortest distance between the closest points of both these hulls.

Image for post



**5. What do know about Hard Margin SVM and Soft Margin SVM?**

Explanation: If a point Xi satisfies the equation **Yi(WT\*Xi +b) ≥ 1,**then Xi is correctly classified else incorrectly classified. So we can see that if the points are linearly separable then only our hyperplane is able to distinguish between them and if any outlier is introduced then it is not able to separate them. So these type of SVM is called**hard margin SVM**(since we have very strict constraints to correctly classify each and every data point).

To overcome this, we introduce a term**( ξ )**(pronounced as Zeta)

Image for post

Image for post

if ξi= 0, the points can be considered as correctly classified.

if ξi> 0 , Incorrectly classified points.

**6. What is Hinge Loss?**

Explanation: Hinge Loss is a loss function which penalises the SVM model for inaccurate predictions.

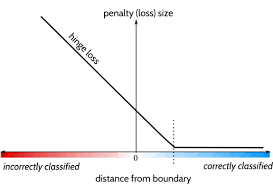
If**Yi(WT\*Xi +b) ≥ 1**, hinge loss is ‘**0**’ i.e the points are correctly classified. When

**Yi(WT\*Xi +b) < 1**, then hinge loss increases massively.

As **Yi(WT\*Xi +b)**increases with every misclassified point, the upper bound of hinge loss {**1- Yi(WT\*Xi +b)**} also increases exponentially.

Hence, the points that are farther away from the decision margins have a greater loss value, thus penalising those points.

Image for post

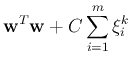


We can formulate hinge loss as **max[0, 1- Yi(WT\*Xi +b)]**

**7. Explain the Dual form of SVM formulation?**

Explanation: The aim of the Soft Margin formulation is to minimize

Image for post



subject to

Image for post

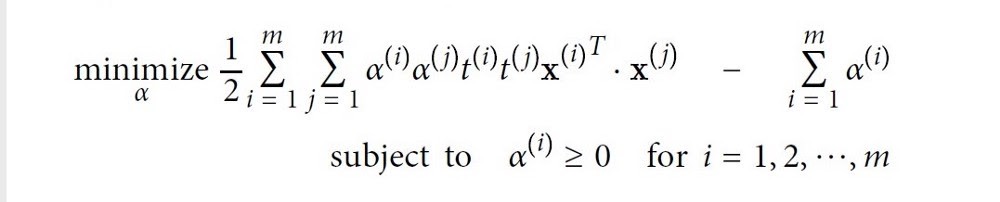
Image for post

This is also known as the primal form of SVM.

The duality theory provides a convenient way to deal with the constraints. The dual optimization problem can be written in terms of dot products, thereby making it possible to use kernel functions.

It is possible to express a different but closely related problem, called its dual problem. The solution to the dual problem typically gives a lower bound to the solution of the primal problem, but under some conditions, it can even have the same solutions as the primal problem. Luckily, the SVM problem happens to meet these conditions, so you can choose to solve the primal problem or the dual problem; both will have the same solution.

Image for post



**8. What’s the “kernel trick” and how is it useful?**

Explanation: Earlier we have discussed applying SVM on linearly separable data but it is very rare to get such data. Here, kernel trick plays a huge role. The idea is to map the non-linear separable data-set into a higher dimensional space where we can find a hyperplane that can separate the samples.

Image for post



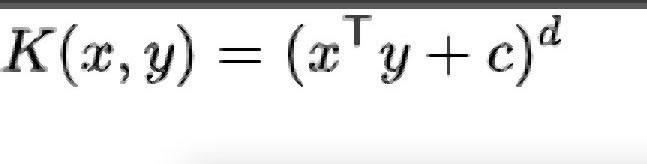
It reduces the complexity of finding the mapping function. So, **Kernel function defines the inner product in the transformed space.**Application of the kernel trick is not limited to the SVM algorithm. Any computations involving the dot products (x, y) can utilize the kernel trick.

**9. What is Polynomial kernel?**

Explanation:**Polynomial kernel** is a[kernel function](https://en.wikipedia.org/wiki/Kernel_function) commonly used with[support vector machines](https://en.wikipedia.org/wiki/Support_vector_machine) (SVMs) and other[kernelized](https://en.wikipedia.org/wiki/Kernel_trick) models, that represents the similarity of vectors (training samples) in a feature space over polynomials of the original variables, allowing learning of non-linear models.

For d-degree polynomials, the polynomial kernel is defined as:

Image for post

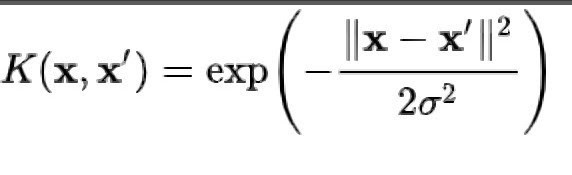


**10. What is RBF-Kernel?**

Explanation:

The RBF kernel on two samples **x** and **x’**, represented as feature vectors in some input space, is defined as

Image for post



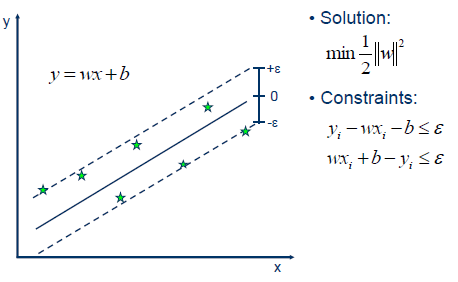
||x-x’||² recognized as the[squared Euclidean distance](https://en.wikipedia.org/wiki/Euclidean_distance#Squared_Euclidean_distance) between the two feature vectors. sigma is a free parameter.

**11. Should you use the primal or the dual form of the SVM problem to train a model on a training set with millions of instances and hundreds of features?**

Explanation: This question applies only to linear SVMs since kernelized can only use the dual form. The computational complexity of the primal form of the SVM problem is proportional to the number of training instances m, while the computational complexity of the dual form is proportional to a number between m² and m³. So, if there are millions of instances, you should use the primal form, because the dual form will be much too slow.

**12. Explain about SVM Regression?**

Explanation: The Support Vector Regression (SVR) uses the same principles as the SVM for classification, with only a few minor differences. First of all, because the output is a real number it becomes very difficult to predict the information at hand, which has infinite possibilities. In the case of regression, a margin of tolerance (epsilon) is set in approximation to the SVM



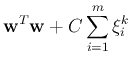
**13. Give some situations where you will use an SVM over a RandomForest Machine Learning algorithm.**

Explanation:

* The main reason to use an SVM instead is that the problem might not be linearly separable. In that case, we will have to use an SVM with a non-linear kernel (e.g. RBF).
* Another related reason to use SVMs is if you are in a higher-dimensional space. For example, SVMs have been reported to work better for text classification.

**14. What is the role of C in SVM? How does it affect the bias/variance trade-off?**

Explanation:



In the given Soft Margin Formulation of SVM, C is a hyperparameter.

**C hyperparameter**adds a penalty for each misclassified data point.

Large Value of parameter C implies a small margin, there is a tendency to overfit the training model.

Small Value of parameter C implies a large margin which might lead to underfitting of the model.

**15. SVM being a large margin classifier, is it influenced by outliers?**

Explanation: Yes, if C is large, otherwise not.

**16. In SVM, what is the angle between the decision boundary and theta?**

Explanation: Decision boundary is a plane having equation Theta1\*x1+Theta2\*x2+……+c = 0, so as per the property of a plane, it’s coefficients vector is normal to the plane. Hence, the Theta vector is perpendicular to the decision boundary.

**17. Can we apply the kernel trick to logistic regression? Why is it not used in practice then?**

Explanation:

1. Logistic Regression is computationally more expensive than SVM — O(N³) vs O(N²k) where k is the number of support vectors.
2. The classifier in SVM is designed such that it is defined only in terms of the support vectors, whereas in Logistic Regression, the classifier is defined over all the points and not just the support vectors. This allows SVMs to enjoy some natural speed-ups (in terms of efficient code-writing) that is hard to achieve for Logistic Regression.

**18. What is the difference between logistic regression and SVM without a kernel?**

Explanation: They differ only in the implementation . SVM is much more efficient and has good optimization packages.

**19. Can any similarity function be used for SVM?**

Explanation: No. It has to have to satisfy [Mercer’s theorem](https://towardsdatascience.com/understanding-support-vector-machine-part-2-kernel-trick-mercers-theorem-e1e6848c6c4d).

**20. Does SVM give any probabilistic output?**

Explanation: SVMs do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation

Reference:

1. <https://www.kdnuggets.com/2016/07/support-vector-machines-simple-explanation.html>
2. <http://web.mit.edu/6.034/wwwbob/svm-notes-long-08.pdf>
3. <https://en.wikipedia.org/wiki/Support_vector_machine>