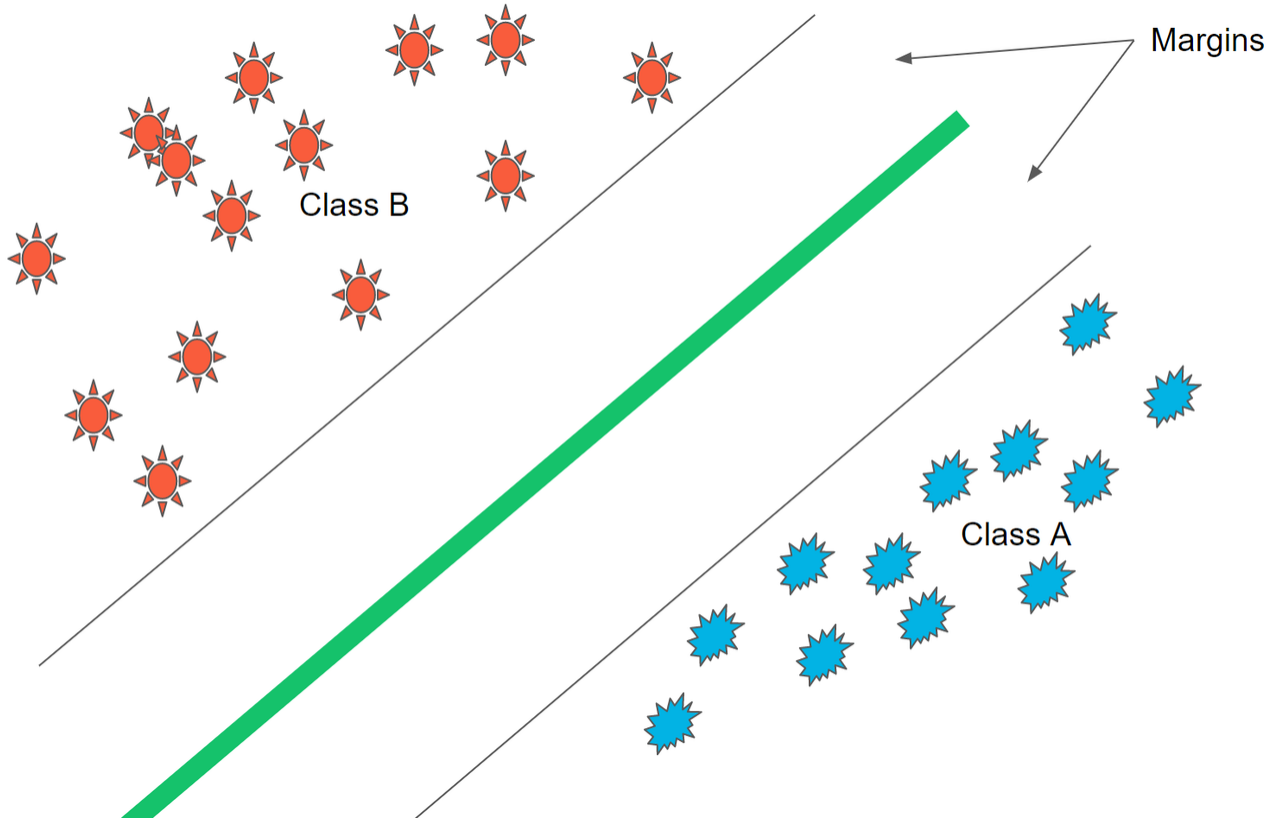
Support Vector Machines

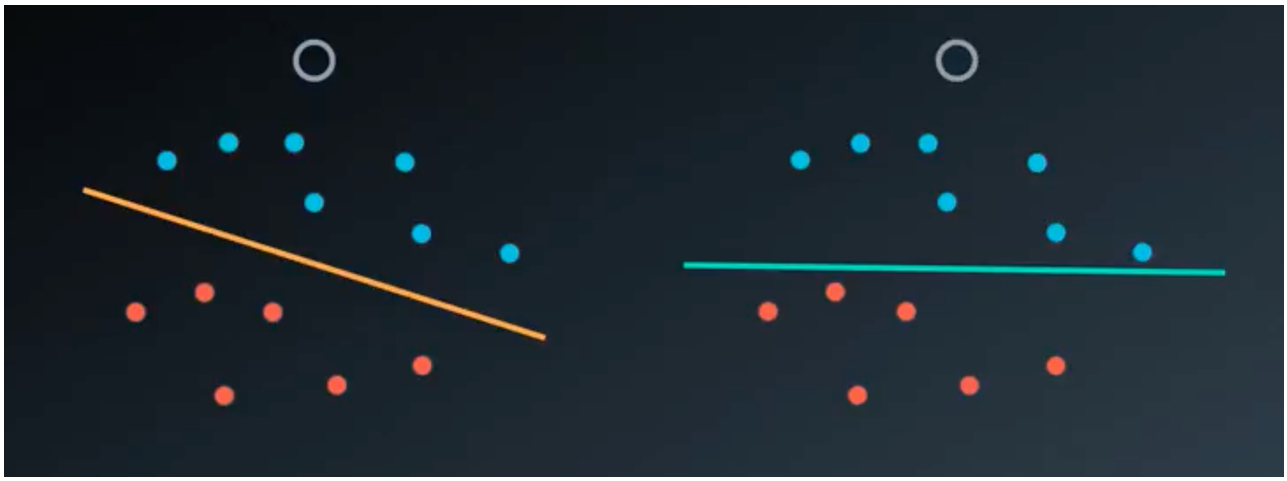
Support vector machines (SVM)

- Support Vector Machines (SVMs) is a powerful algorithm for classification which also finds the best boundary.
- Same as logistic regression algorithm, but it finds the best model parameters (W and b) that is good enough to make the boundary line has a large distance from the points.
- At a high level, SVM is splitting the data with a line, but adds margins into the equation as a tool to further improve the accuracy of the training model. There are several tricks, errors, and strategies we will learn to maximize these margins.



Simple question

- What is the best line to split the dataset based on their classes?



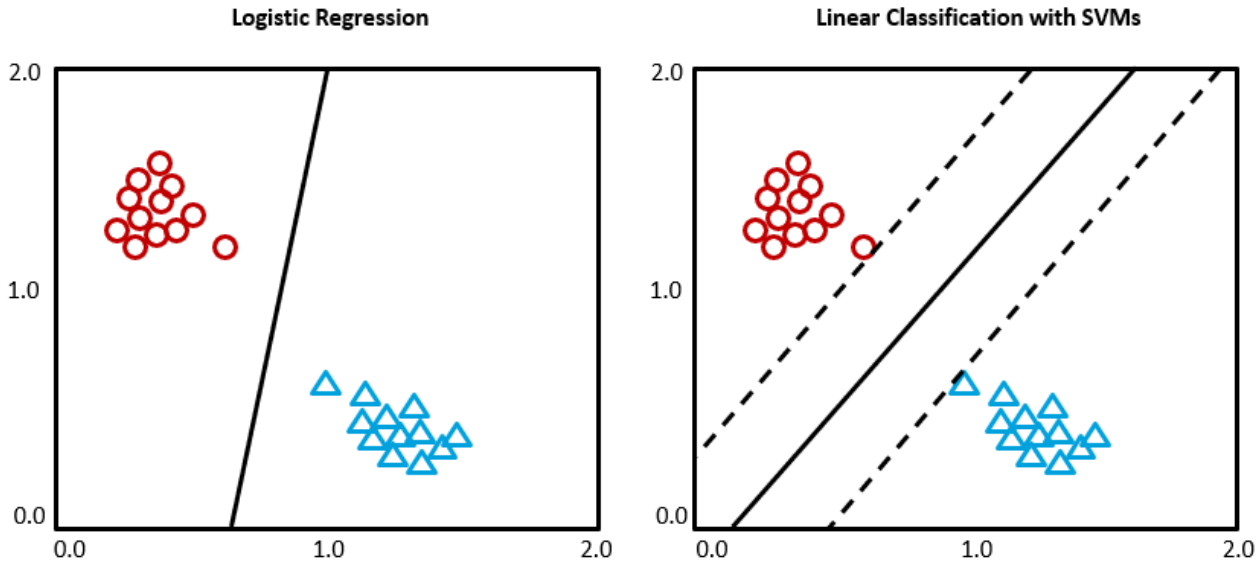
The solution

- The yellow line is the best line.
- As you can see, the yellow line is pretty well-spaced between the blue points and the red points. The green line is not, because it's a little too close to some of the points on both sides.
- Why the distance is matter?
  More distance = more sure to predict new data.
- Look at all the distances between the closest point to one of the lines and compare them to the distances between the closest points to the other line. Then, we check which one of these closest points is actually farther from the line.
- This is the SVM.

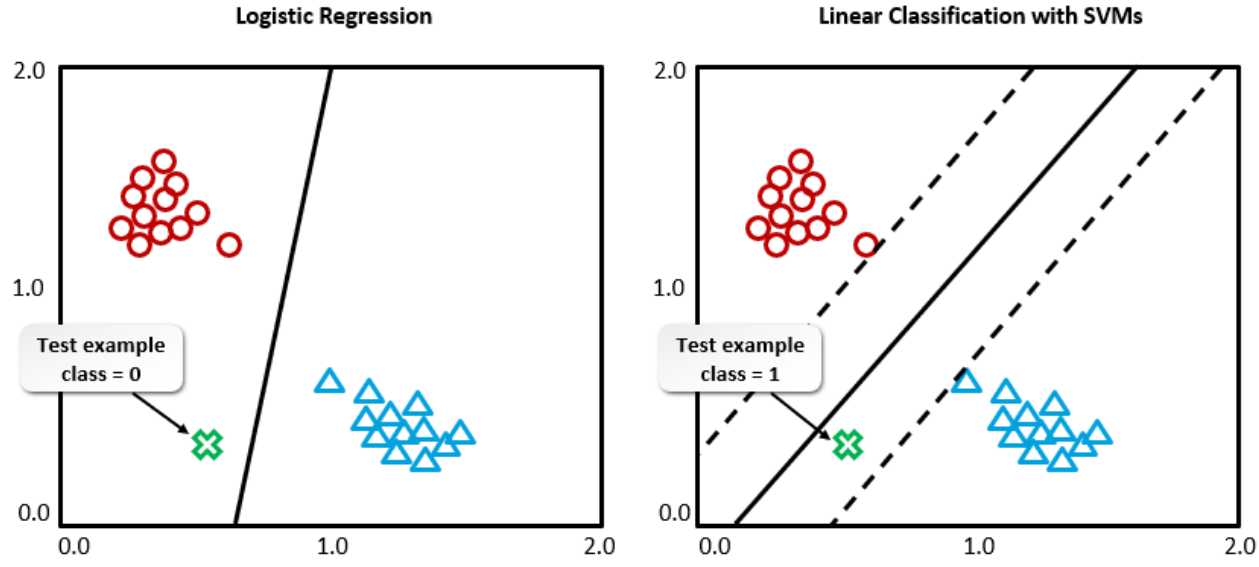
Example

- Let us see the difference between logistic regression and SVM

  1 - Training the models



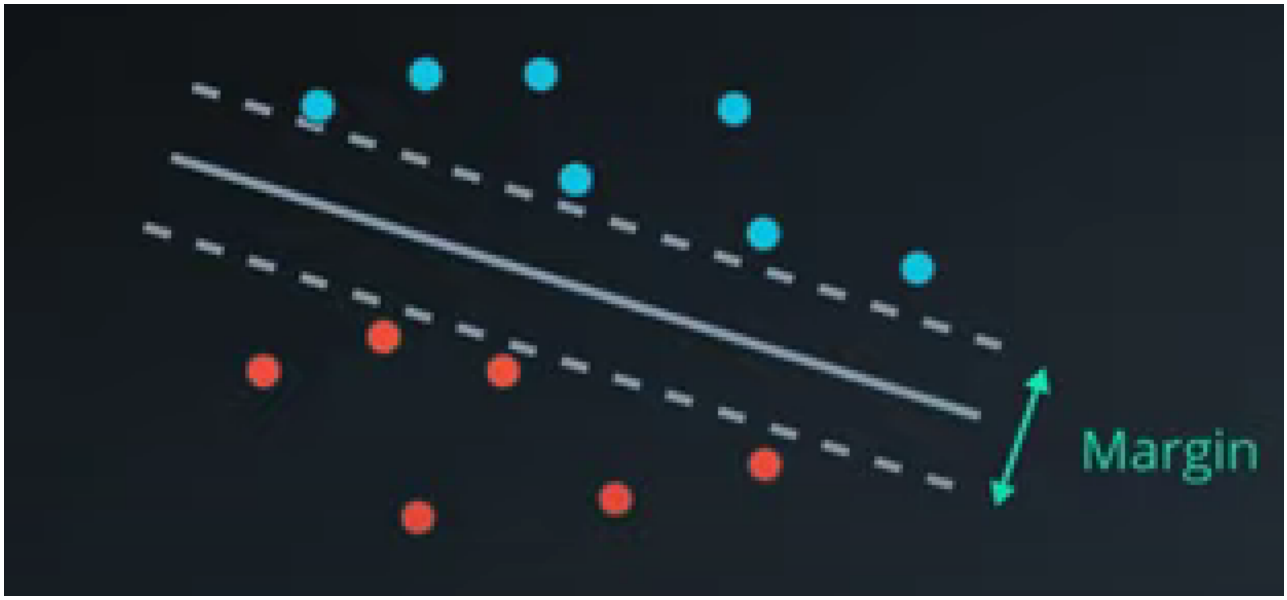
2 - Predicting unseen data using the two models



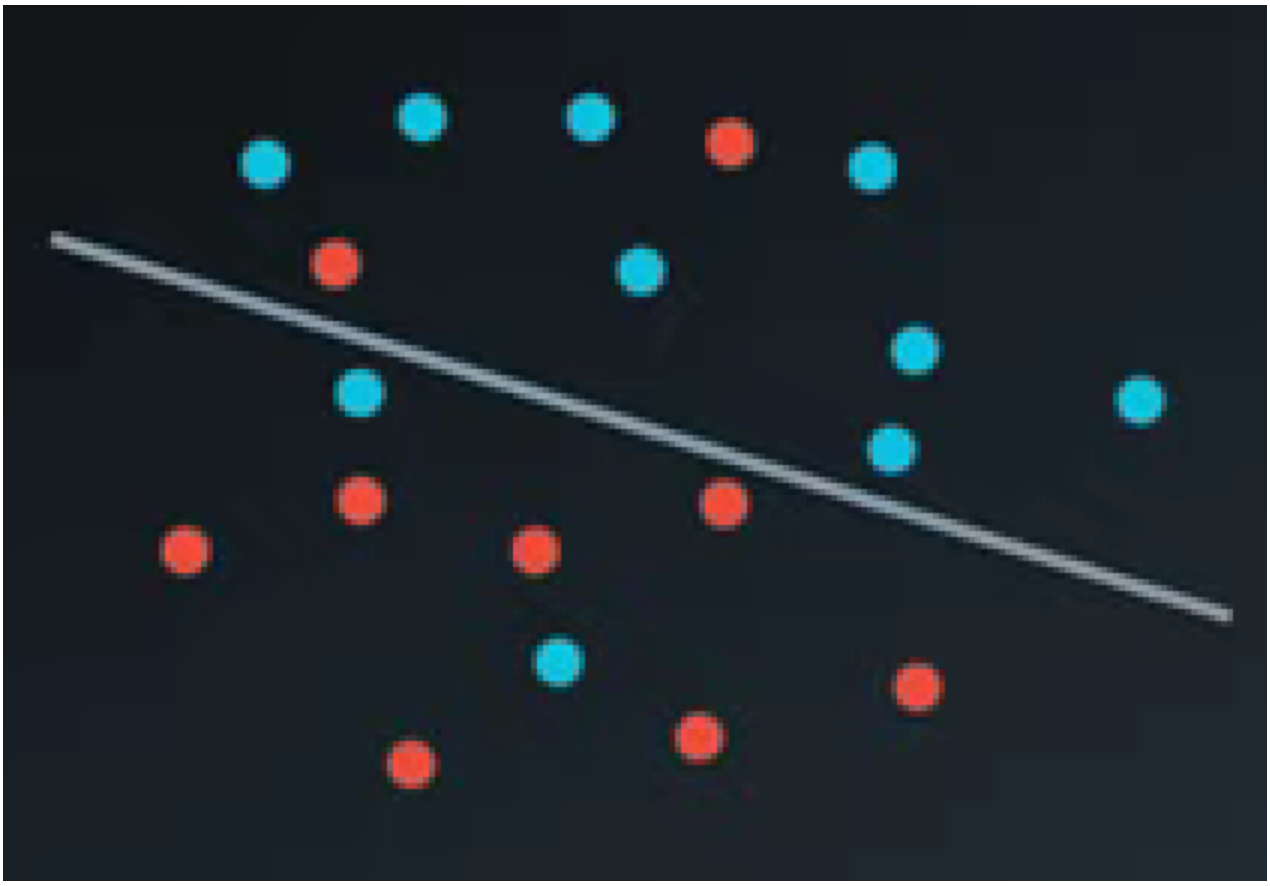
- Because SVM try to find the best line that is good enough to has a large distance from the points, it seems more powerful model than logistic regression in the case above.
- Thus, SVM called a deterministic algorithm, while logistic regression called a probabilistic algorithm.

Recall Classification Methods

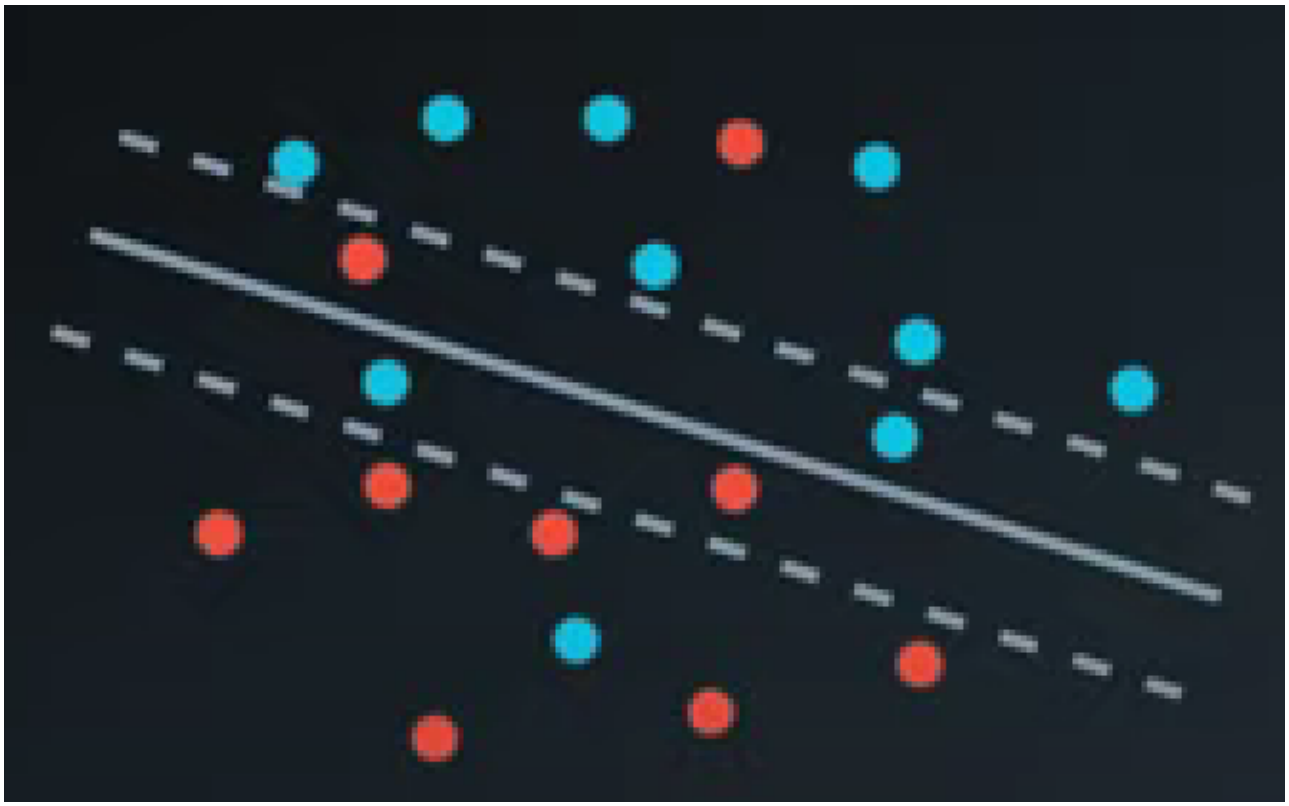
- Classification means we have some blue points and some red points, and we want to find a boundary that separates them.
- But now we want the boundary to be as far away from the points as possible. To do this we add two boundaries, and try to maximize the distance between the two.



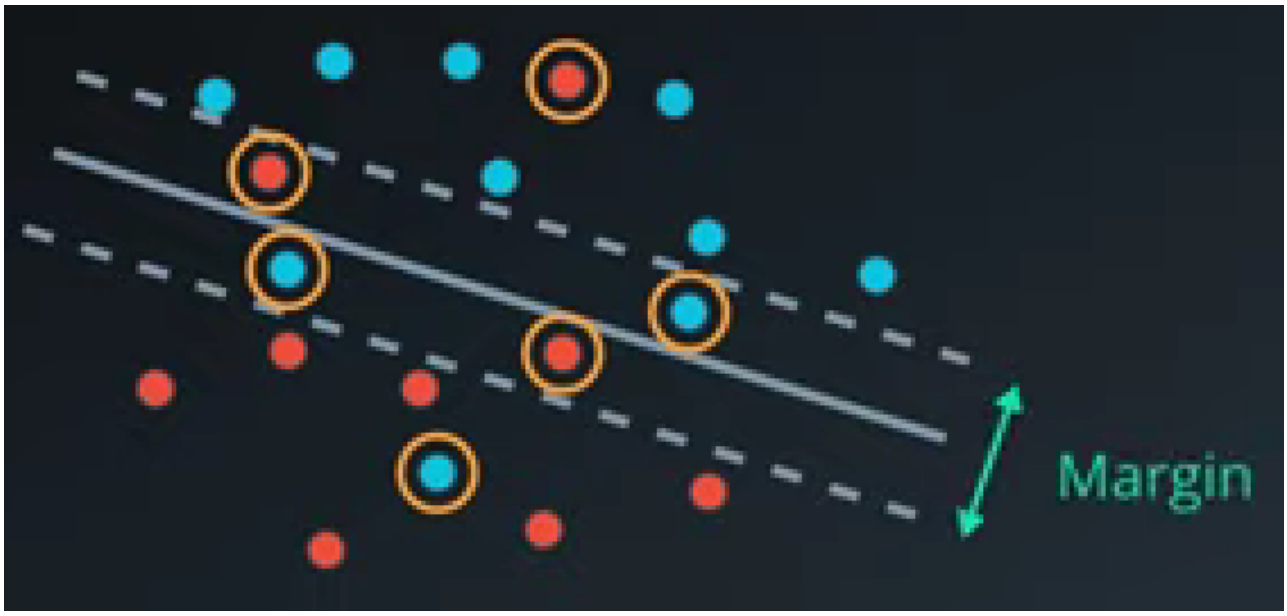
- We can measure the performance of the model by maximize the margin.
- But what about a more complex problems? Let us take another example



- Any classification method will split the dataset in this way, but SVM will split them in a different way. Let us see the SVM method



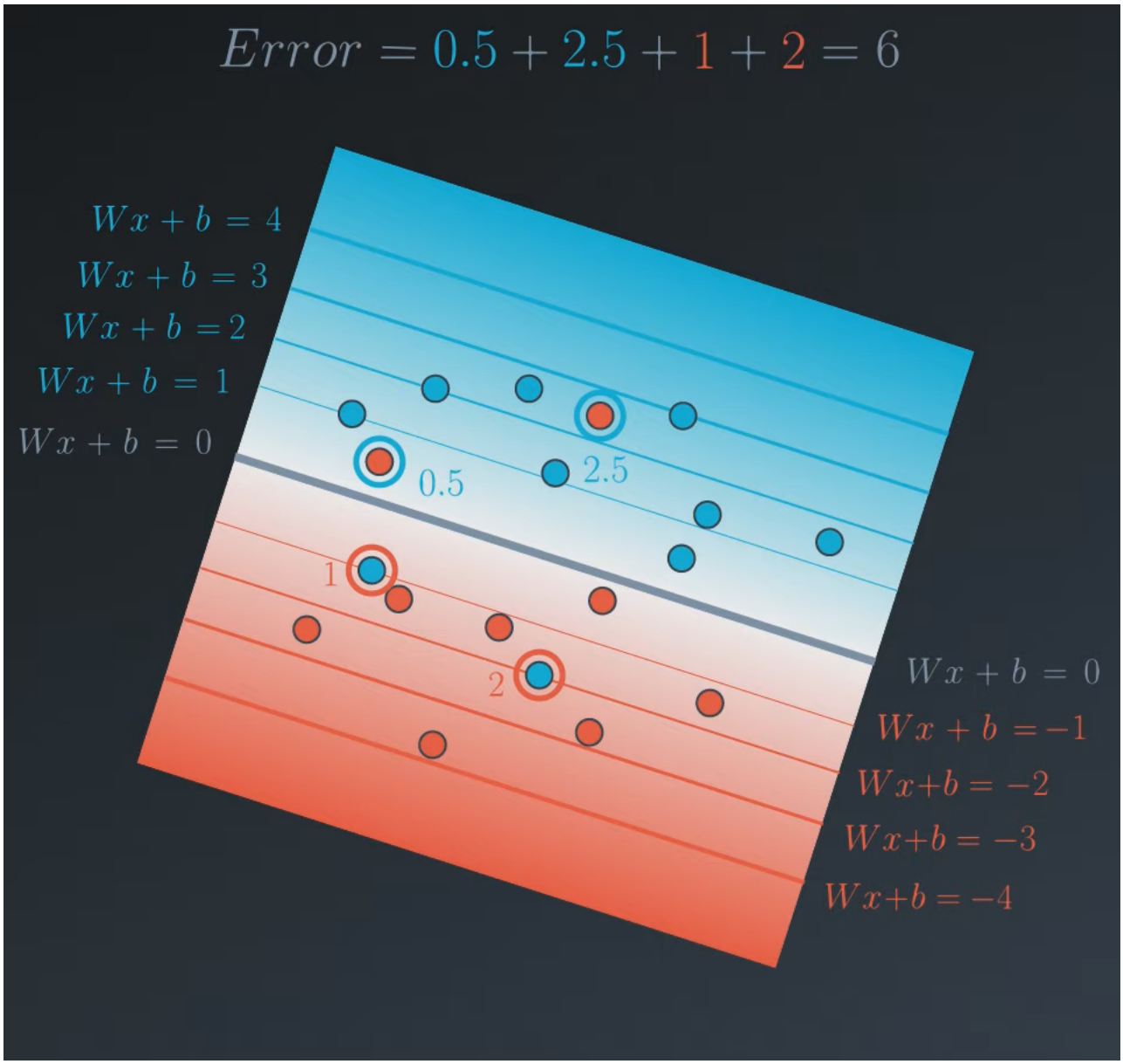
- SVM will add two more lines, one in the class 0 area, and the other one in the class 1 area.
- After that, it will maximize the distance between these two lines and the boundary line, this will maximize the margin.
- If a lot of points in the two classes in the save area (under or above their line), then stop maximize the margin, because we have a good margin.
- But what about the misclassified points? that the data inside the margin?



- We make them part of our classification error.
- We also do not want any points in the margin, so we would include them in the classification error.
- Therefore, we can calculate how our model is good by find how many misclassified points and how wide the margin is?
- Now, how to find the best boundary line that has a maximum possible margin with least number of misclassified points?

How to Calculate the misclassified error

- First, add line with the following equation
  WX + b = 0
  In the example above, we have two features, so the equation is
  w1 x1 + w2 x2 + b = 0
- Add lines above and below the boundary line, to find the errors.
- The error is, any misclassified points (points that are not in their areas) should has a value that represents the distance between the point itself and the line of the equation.

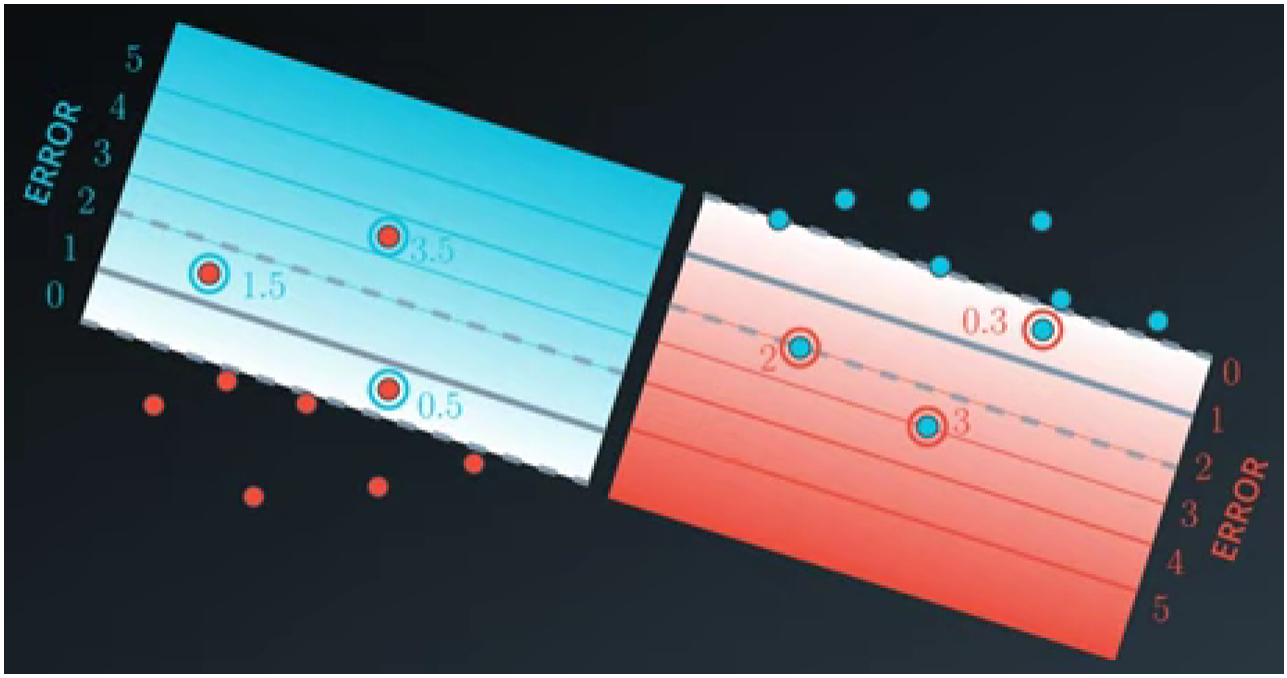


Classification error with SVM

- SVM Algorithm do the same method in the above, but except we add two extra lines to find the error.
- We have the equation WX + b = 0, that is the boundary line.
- Now, we take as the same method above, the two lines WX + b = 1 and WX + b = -1.
- Same as the method above, we will find the distance of the misclassified points that are far from the line. But infact we will calculate the distance of misclassified

points for each line except the boundary line.

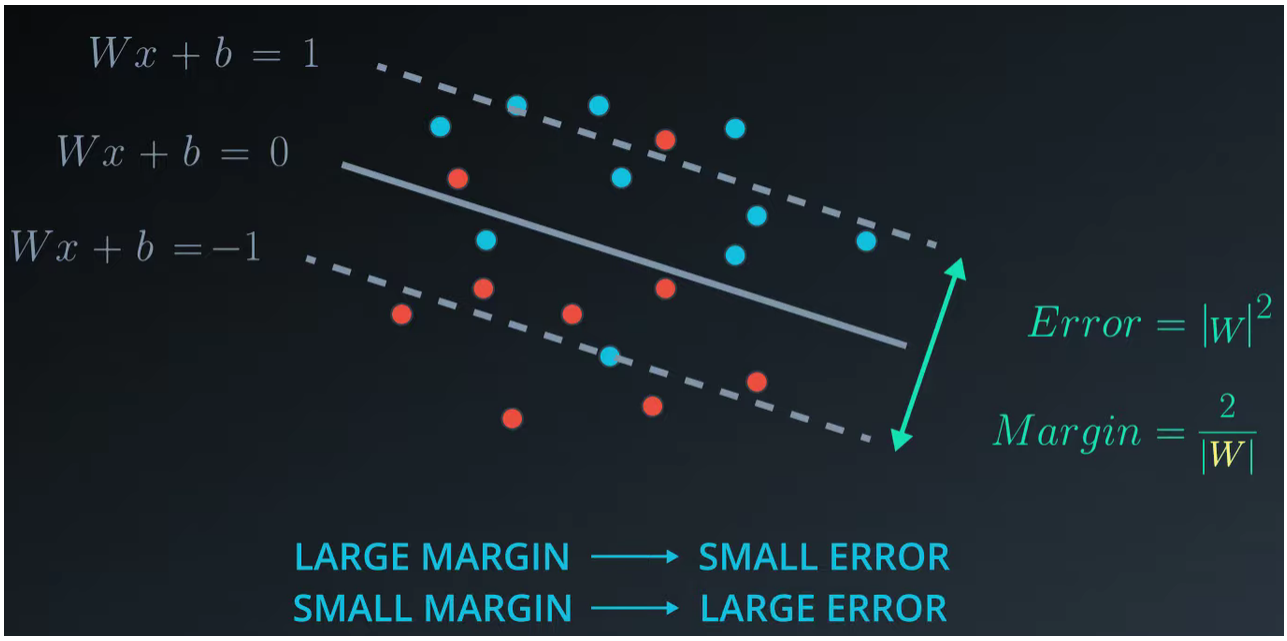
- For each of these two lines, we will split the dataset into two spaces, and after that, any point that is not in its area, it will take some value. After that, we will add these values together to get the total error.
- We look at the values of $Wx+b$. As we go up, it's going to be 1, 2, 3, 4, etc . And as we go down, it's going to be -1, -2, -3, etc. In order to build the error, we take the absolute value of those errors and translate it by one.
- That means, every line will start with the equation WX + b = 0.



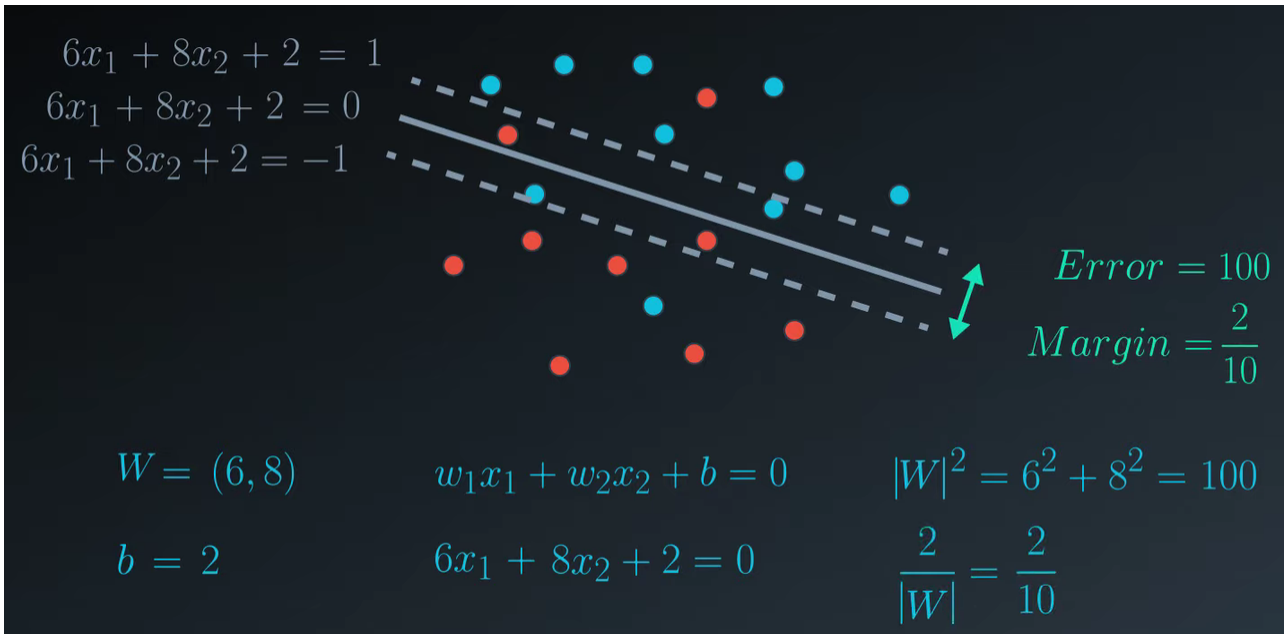
- Classification error = 1.5 + 3.5 + 0.5 + 2 + 0.3 + 3
- Repeat the step until you get the best two lines with the least error.

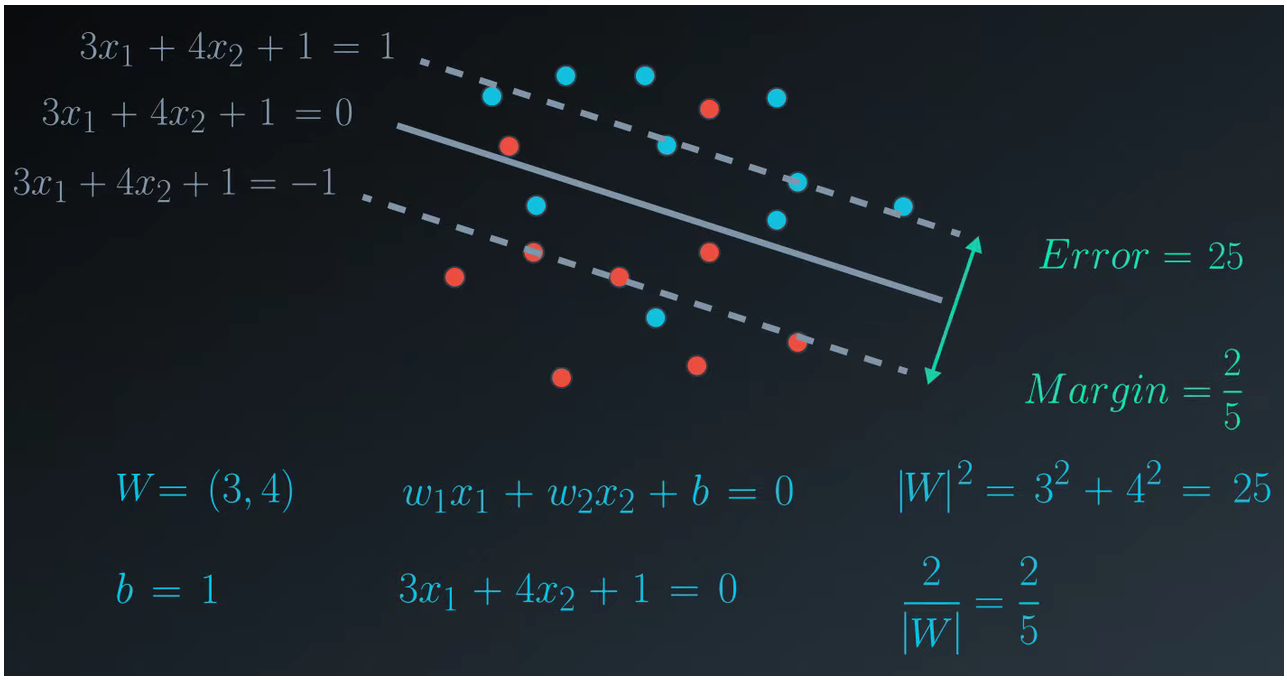
Margin Error with SVM

- The margin error is depends on the parameters W and b, if you increase them, you will get large error and small margin.
- You can find the margin distance by dividing 2 by the square root of the sum of the weights.
- You can find the margin error by summing the weights and after that, take the the square of the result.


$Wx + b = 1$
$Wx + b = 0$
$Wx + b = -1$
$Error = |W|^2$
$Margin = \frac{2}{|W|}$
LARGE MARGIN ⟶ SMALL ERROR
SMALL MARGIN ⟶ LARGE ERROR

Examples


$6x_1 + 8x_2 + 2 = 1$
$6x_1 + 8x_2 + 2 = 0$
$6x_1 + 8x_2 + 2 = -1$
$Error = 100$
$Margin = \frac{2}{10}$
$W = (6, 8)$
$b = 2$
$w_1x_1 + w_2x_2 + b = 0$
$6x_1 + 8x_2 + 2 = 0$
$|W|^2 = 6^2 + 8^2 = 100$
$\frac{2}{|W|} = \frac{2}{10}$

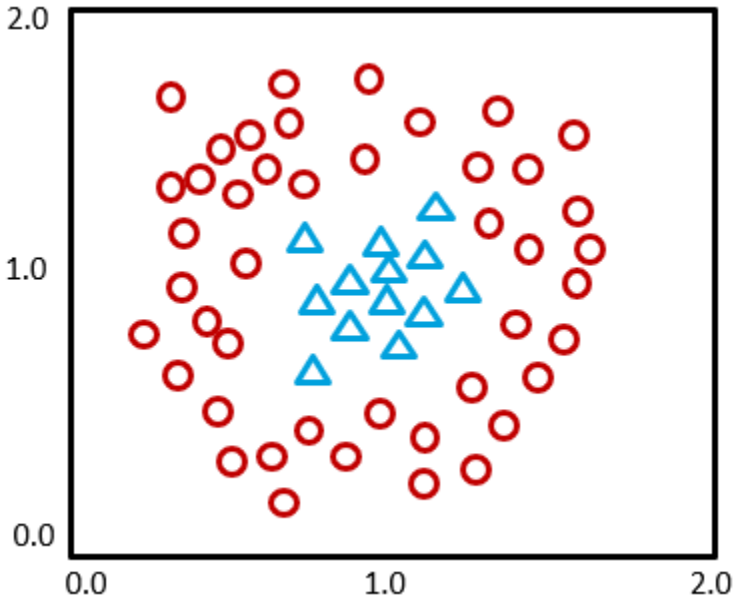


Error Function

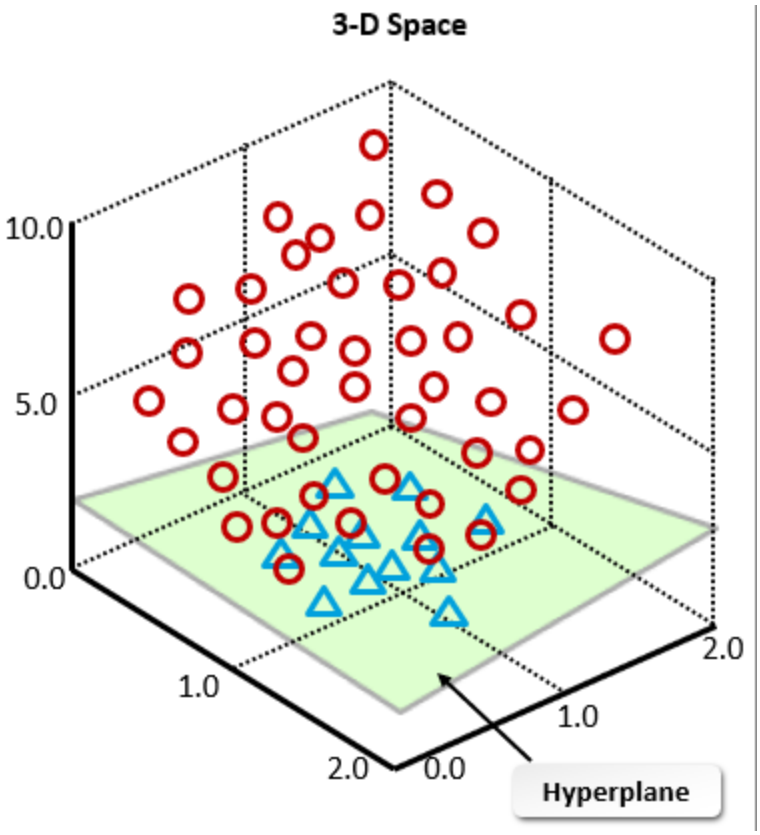
- Now, we know the classification error and the margin error.
- To find the Total error, just add them together to get the total error.
- Large error means bad model, same as logistic regression..
- What does the same as logistic regression means? That means we can minimize the error by using gradient descent algorithm.
  - You have the equation Line WX + b = 0.
  - Try to get the best lines by using different model parameters.
  - Take in account the classification error and margin error.
  - Calculate these two errors, and then calculate the total error.
  - Repeat the process until you stop at the local minima.

SVM is a Powerful algorithm!!

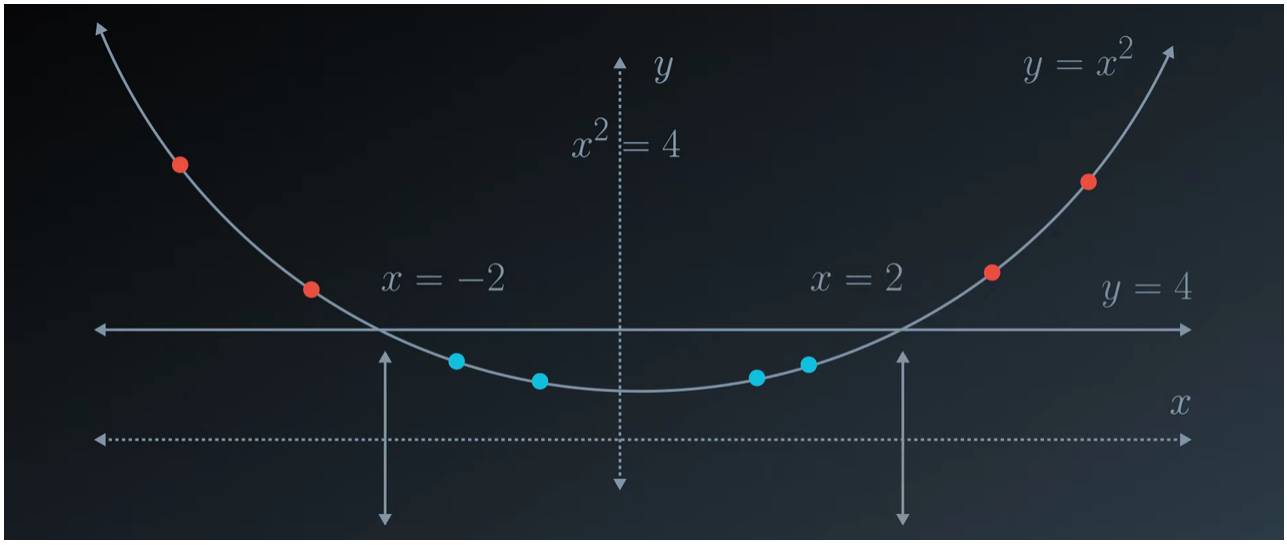
- What if we have non linear problem? Like this one



- How can we split the classes using the equation line?
- There is a parameter called a kernel trick or Polynomial trick.
- It is a group of mathematical methods for efficiently representing non-linearly separable data in higher-dimensional space.
- So, it convert the current space into a higher one.
- In the example above, we have 2D-space that is non-linear. We can convert it into a 3D-Space. Thus we put the points of class zero below, and the points of class 1 above. We get this.



- Now, let us convert it again into its original space (2D). Now you can split it easly, even you have some missclassified points.
- But, how it is working? How SVM convert the spaces?
- It depends on the state of the problem. For example, what about the problem below?



- Note, there are another kernel tricks used to convert the spaces, but all of them follow the same method.

Resources

- Intro to Machine learning with TensorFlow Nanodegree : https://www.udacity.com/course/intro-to-machine-learning-with-tensorflow-nanodegree--nd230
- CertNexus Certified AI Practitioner Certificate :
  1 - https://www.coursera.org/professional-certificates/certified-artificial-intelligence-practitioner
  2 - https://certnexus.com/certified-artificial-intelligence-practitioner-caip/