German Research Center for Artificial Intelligence

Machine Learning (DLMDSML01)

Nghia Duong-Trung[1]
[1] **German Research Center for Artificial Intelligence (DFKI GmbH)**
Internal Use @ IU International University of Applied Science, Berlin Campus

SPONSORED BY THE

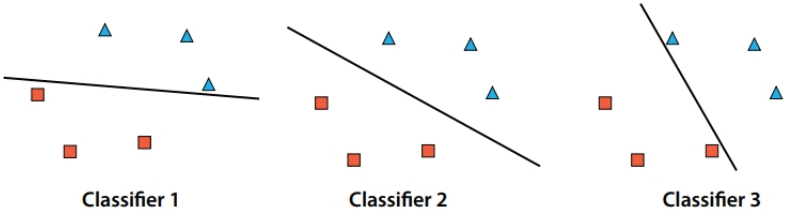
Federal Ministry of Education and Research

Section 4: Support Vector Machine (SVM)

German Research Center for Artificial Intelligence

- Textbook: check it in MyCampus
- Basic concepts
- Hyperplane
- Error function
- C parameter
- RBF kernel
- Gamma parameter

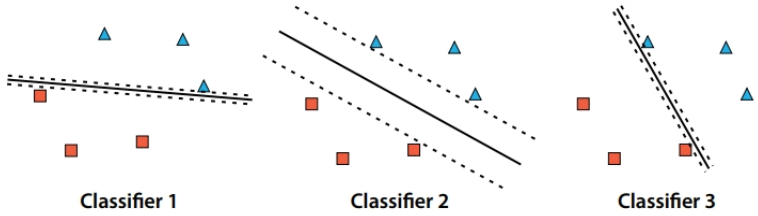
Basic concepts


Classifier 1
Classifier 2
Classifier 3

Parallel lines

German Research Center for Artificial Intelligence

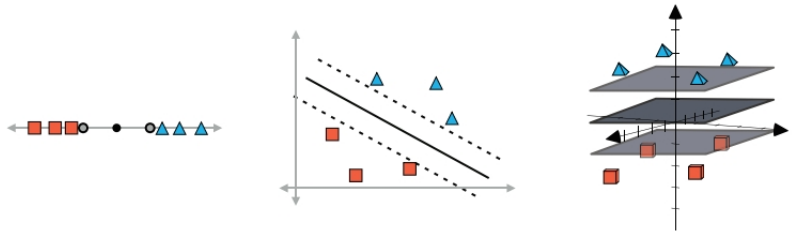
- An SVM classifier uses two parallel lines instead of one line
- The goal of the SVM is twofold
  - it tries to classify the data correctly and
  - also tries to space the lines as much as possible



Hyperplane

German Research Center for Artificial Intelligence

- SVMs work equally well in datasets of higher dimensions
- The linear boundaries in one dimension are points and in two dimensions are lines
- Likewise, the linear boundaries in three dimensions are planes, and in higher dimensions, they are hyperplanes of one dimension less than the space in which the points live
- In each of these cases, we try to find the boundary that is the farthest from the points



Error function of SVMs

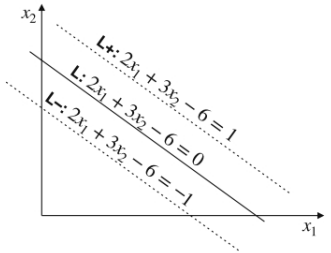
German Research Center for Artificial Intelligence

- The Error function of SVMs tries to maximize two things at the same time: the classification of the points and the distance between the lines
- The following are the two things we want to achieve
  - Each of the two lines should classify the points as best as possible
  - The two lines should be as far away from each other as possible
- Therefore, our error function can look like this:
  Error = Classification Error + Distance Error

Classification error function

German Research Center for Artificial Intelligence

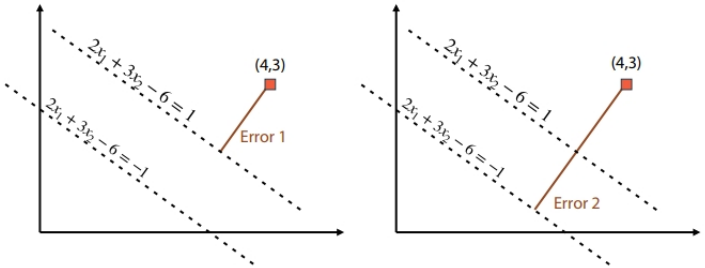
- The SVM uses two parallel lines that have similar equations; they have the same weights but a different bias
- We use the central line as a frame of reference L with equation $w_1 x_1 + w_2 x_2 + b = 0$, and construct two lines
  - $L+ : w_1 x_1 + w_2 x_2 + b = 1$
  - $L- : w_1 x_1 + w_2 x_2 + b = -1$



Classification error function

German Research Center for Artificial Intelligence

- Our classifier consists of the lines $L+$ and $L-$ as two independent classifiers

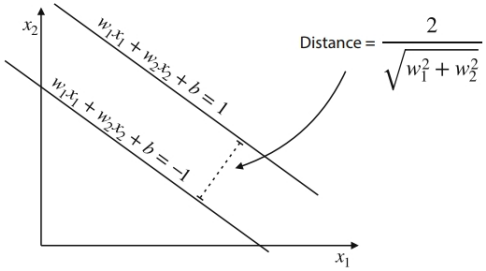


Error = Error 1 + Error 2

Distance error function

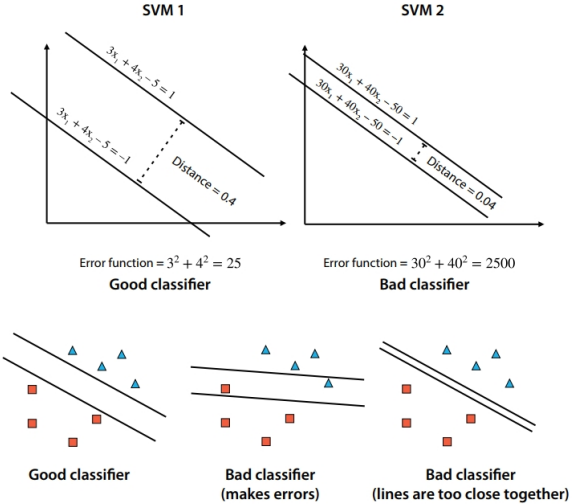
German Research Center for Artificial Intelligence

- Now that we have created an error function that measures classification errors, we need to build one that looks at the distance between the two lines and raises an alarm if this distance is small
- An error function that is large when the two lines are close and small when they are far



Distance error function

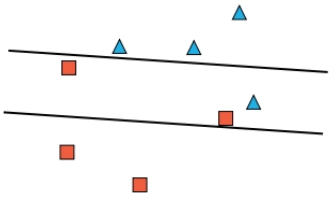
German Research Center for Artificial Intelligence



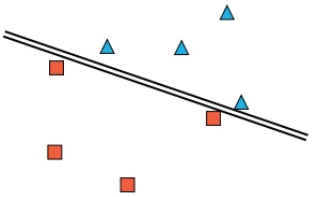
The C parameter

German Research Center for Artificial Intelligence

- The C parameter is used in cases where we want to train an SVM that pays more attention to classification than to distance (or the other way around)
  - Error formula = C . (Classification Error) + (Distance Error)



Lines are far apart (good).
Makes errors (bad).

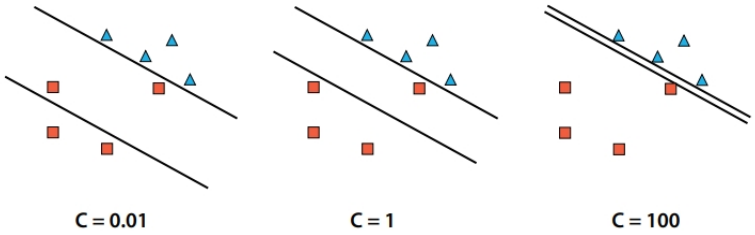


Lines are too close (bad).
Doesn't make errors (good).

The C parameter

German Research Center for Artificial Intelligence

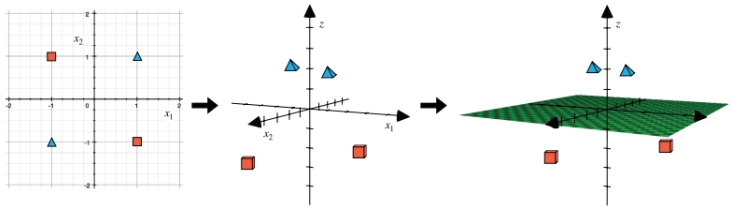
- If C is large, then the error formula is dominated by the classification error, so our classifier focuses more on classifying the points correctly
- If C is small, then the formula is dominated by the distance error, so our classifier focuses more on keeping the lines far apart



Training SVMs with nonlinear boundaries

German Research Center for Artificial Intelligence

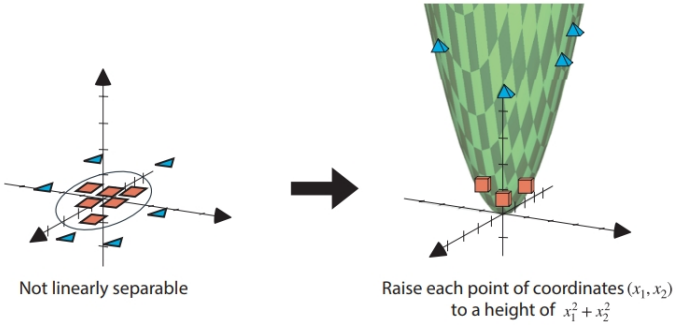
- The kernel method
  - If we have a dataset and find that we can't separate it with a linear classifier
  - One idea is to add more columns to this dataset and hope that the richer dataset is linearly separable
  - The kernel method consists of adding more columns in a clever way, building a linear classifier on this new dataset and later removing the columns we added while keeping track of the (now nonlinear) classifier



The kernel method

German Research Center for Artificial Intelligence

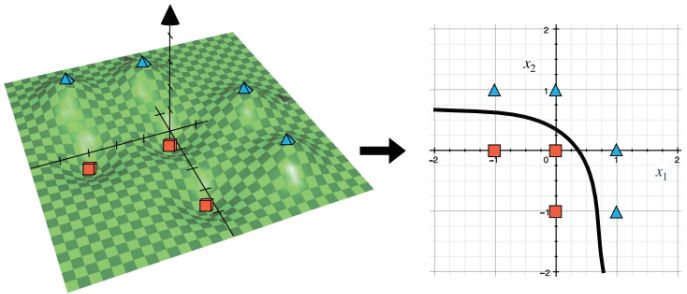
- The kernel method as a way of adding columns to our dataset to make the points separable
  - For example, the dataset has two columns, x : 1 and x_2, and we have added the third column with the value $x_1 x_2$.



Going beyond quadratic equations:: Polynomial kernel

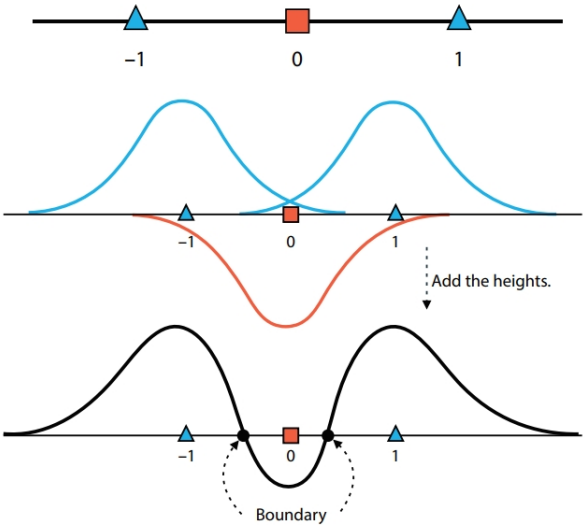
German Research Center for Artificial Intelligence

- The polynomial kernel of degree 2: $(x_1, x_2, x_1^2, x_1x_2, x_2^2)$
- The polynomial kernel of degree 3: $(x_1, x_2, x_1^2, x_1x_2, x_2^2, x_1^3, x_1^2x_2, x_1x_2^2, x_2^3)$



The radial basis function (RBF) kernel

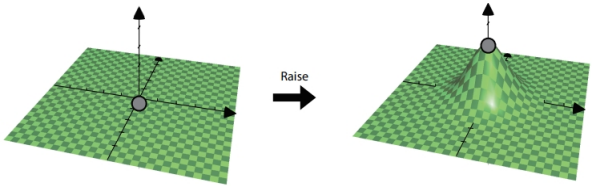
German Research Center for Artificial Intelligence



RBF kernel

German Research Center for Artificial Intelligence

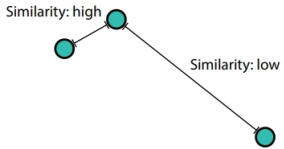
- To build the mountains and valleys on the plane, imagine the plane as a blanket
  - If we pinch the blanket at that point and raise it, we get the mountain
  - If we push it down, we get the valley
  - These mountains and valleys are radial basis functions
  - They are called radial basis functions because the value of the function at a point is dependent only on the distance between the point and the center
  - We can raise the blanket at any point we like, and that gives us one different radial basis function for each point



Similarity

German Research Center for Artificial Intelligence

- To build an SVM using the RBF kernel, we need one notion: the notion of similarity
- We say that two points are similar if they are close to each other, and not similar if they are far away
  - the similarity between two points is high if they are close to each other and low if they are far away from each other
  - If the pair of points are the same point, then the similarity is 1
  - In theory, the similarity between two points that are an infinite distance apart is 0
  - $\text{similarity}(p, q) = e^{-\text{distance}(p,q)^2}$



Training an SVM with the RBF kernel

German Research Center for Artificial Intelligence

Point	x	y (label)
1	–1	1
2	0	0
3	1	1

Point	x	Sim1	Sim2	Sim3	y
1	–1	1	0.368	0.018	1
2	0	0.368	1	0.368	0
3	1	0.018	0.368	1	1

Training an SVM with the RBF kernel

German Research Center for Artificial Intelligence

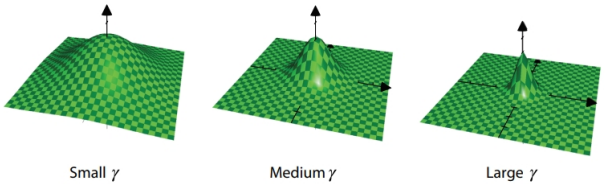
Point	x_1	x_2	y
1	0	0	0
2	−1	0	0
3	0	−1	0
4	0	1	1
5	1	0	1
6	−1	1	1
7	1	−1	1

Point	x_1	x_2	Sim1	Sim2	Sim3	Sim4	Sim5	Sim6	Sim7	y
1	0	0	1	0.368	0.368	0.368	0.368	0.135	0.135	0
2	−1	0	0.368	1	0.135	0.135	0.018	0.368	0.007	0
3	0	−1	0.368	0.135	1	0.018	0.135	0.007	0.368	0
4	0	1	0.368	0.135	0.018	1	0.135	0.368	0.007	1
5	1	0	0.368	0.018	0.135	0.135	1	0.007	0.368	1
6	−1	1	0.135	0.368	0.007	0.367	0.007	1	0	1
7	1	−1	0.135	0.007	0.368	0.007	0.368	0	1	1

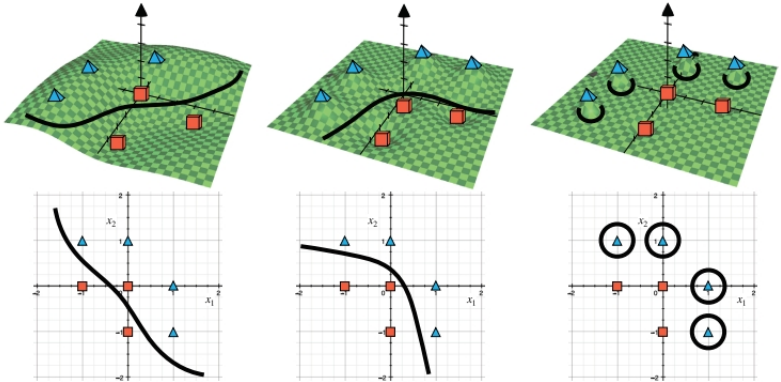
Overfitting and underfitting with the RBF kernel

German Research Center for Artificial Intelligence

- The gamma parameter: $\text{similarity}(p, q) = e^{-\gamma \text{distance}(p,q)^2}$

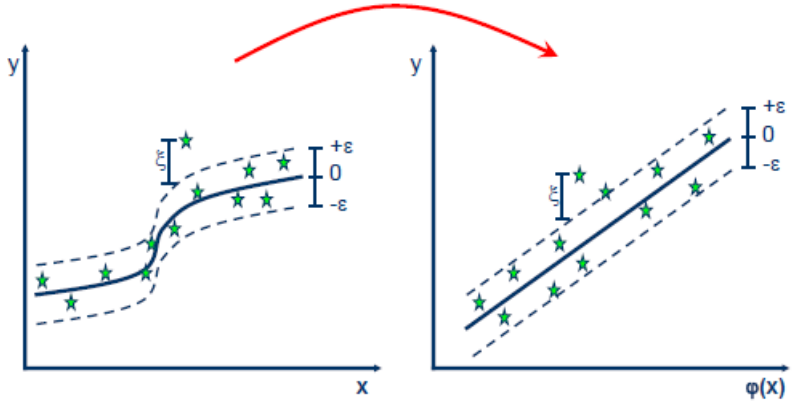


Overfitting and underfitting with the RBF kernel



SVR

German Research Center for Artificial Intelligence



Reference

German Research Center for Artificial Intelligence

- Serrano, L. (2021). Grokking Machine Learning. Simon and Schuster.
- https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html
- https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html
- https://www.datacamp.com/tutorial/svm-classification-scikit-learn-python
- https://www.tutorialspoint.com/scikit_learn/scikit_learn_support_vector_machines.htm

Questions?