Bonn-Aachen
International Center for
Information Technology

Introduction to Machine Learning Tutorial

Another Supervised Learning Method

Support Vector Machines (SVMs) is another **supervised** ML algorithm.

It is usually used for **regression** and **classification** tasks.

We want to apply it to the diabetes.csv and heart.csv file for a classification task. The algorithm should solve the same problem as the K-Nearest Neighbors algorithm.

Which algorithm performs better on each data file?

Before solving this problem, let’s have a quick introduction to Support Vector machines (SVM).

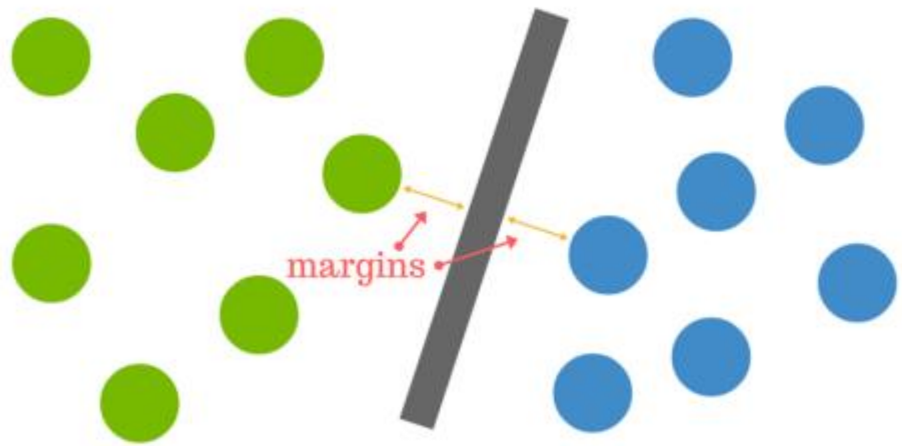
Support Vector Machines (SVM)

This explanation of SVM will be based on a classification problem.

In our case we want to classify data, if patients have or do not have diabetes.

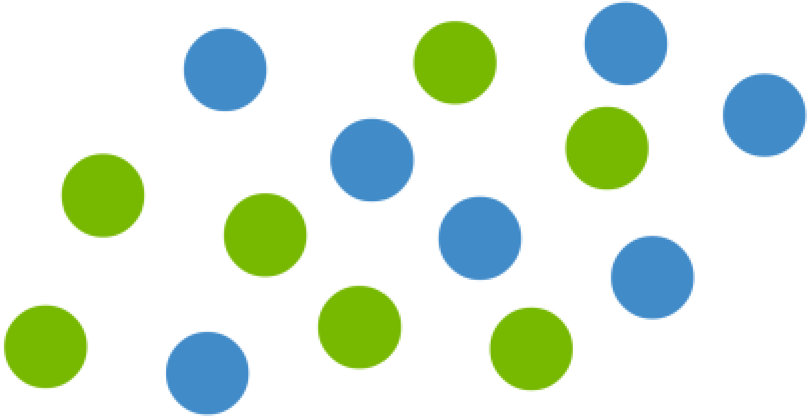
SVM are based on the assumption that it is possible to find a so called **hyperplane** that divides the dataset into two classes.

Hyperplane



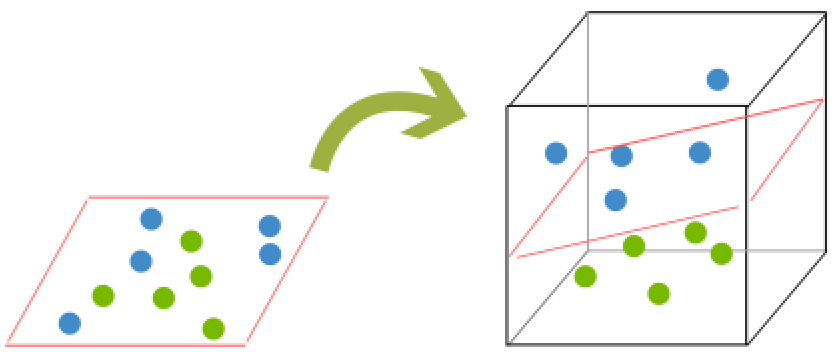
Or, in other words, how do we best segregate the two classes within the data?
The distance between the hyperplane and the nearest data point from either set is known as the **margin**. The goal is to choose a **hyperplane with the greatest possible margin** between the hyperplane and any point within the training set, giving a greater chance of **new data being classified correctly**.

Support Vector Machines (SVM)



But most likely it will not look like as in the previous slide. The data will most likely look as seen above. Where can one fit a hyperplane in this dataset?

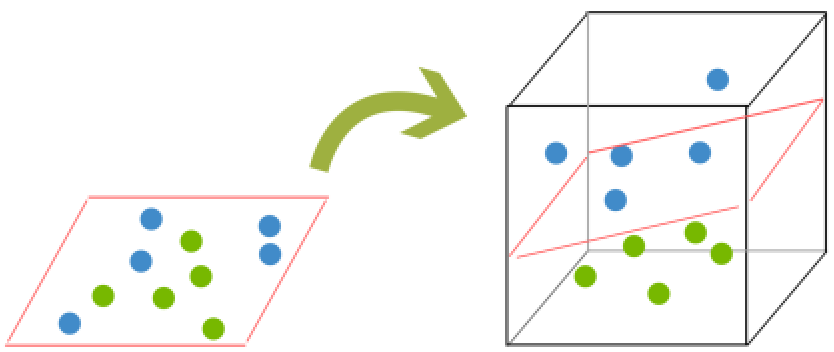
Support Vector Machines (SVM)



In order to classify a dataset like the one above it's necessary to move away from a 2d view of the data to a 3d view. You **transform your 2d dataset into a 3d dataset** and fit a plane (we are not anyone more in a plane!) in the 3d dataset that best separates the data.

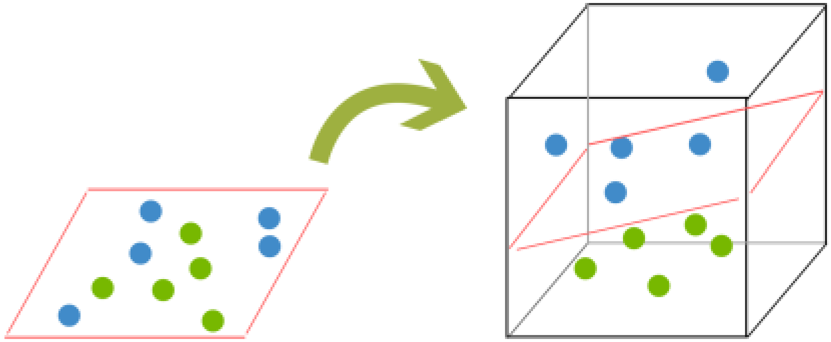
Bonn-Aachen International Center for Information Technology

Support Vector Machines (SVM)



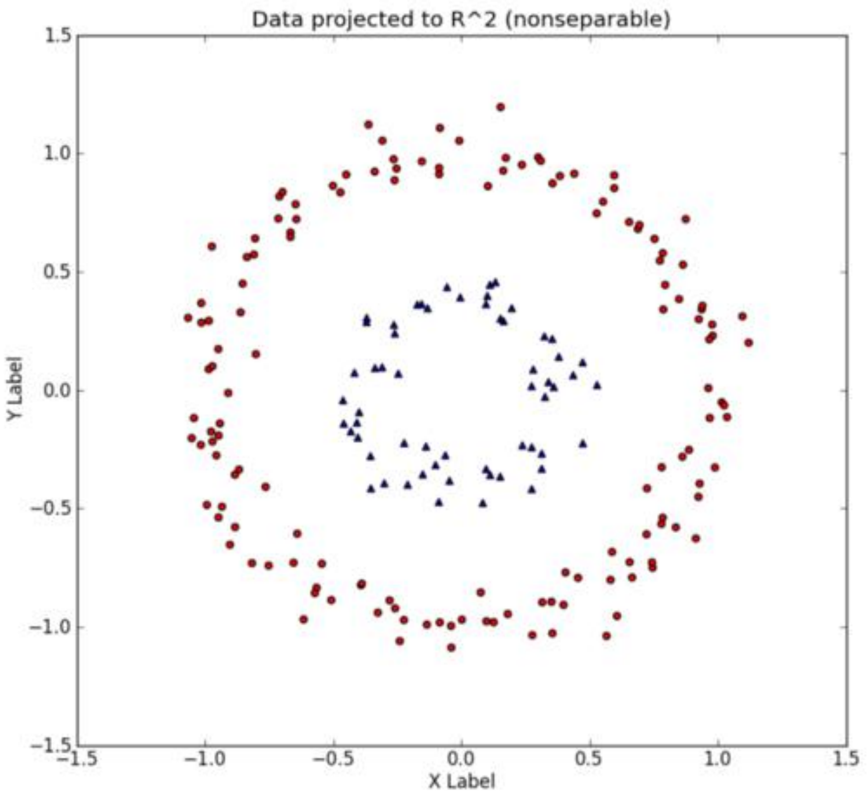
The method of **transforming the 2d data set into a 3d dataset** (multidimensional dataset) is called **kernelling**.

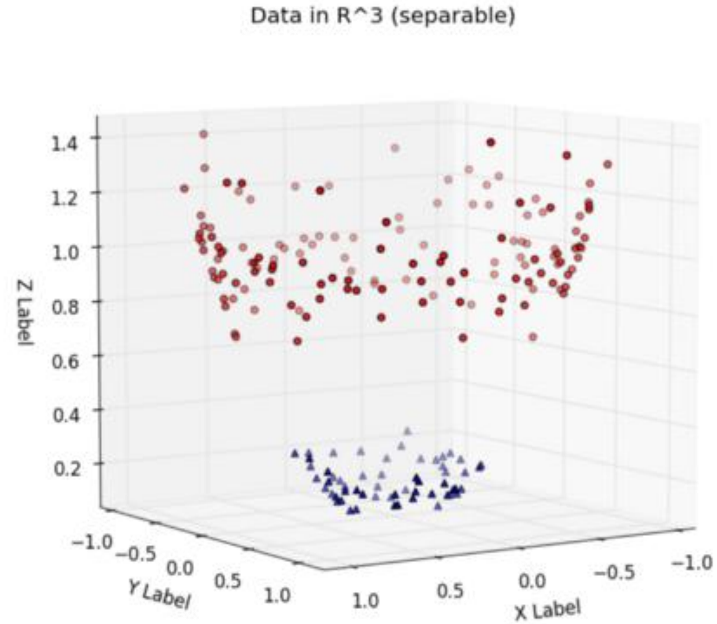
Support Vector Machines (SVM)



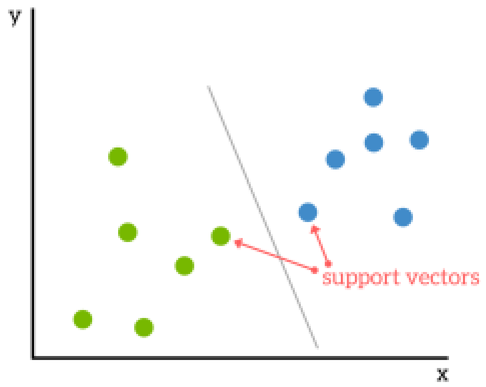
Because we are now in three dimensions, our hyperplane can **no longer be a line**. It must now **be a plane** as shown in the example above. The idea is that the data will continue to be **mapped into higher and higher dimensions** until a hyperplane can be formed to segregate it.

Separate Data by Transformation



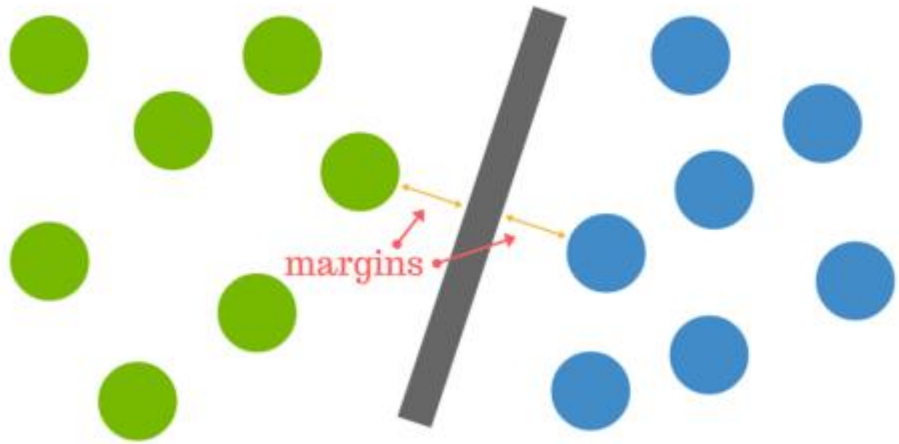


And what is a support vector?



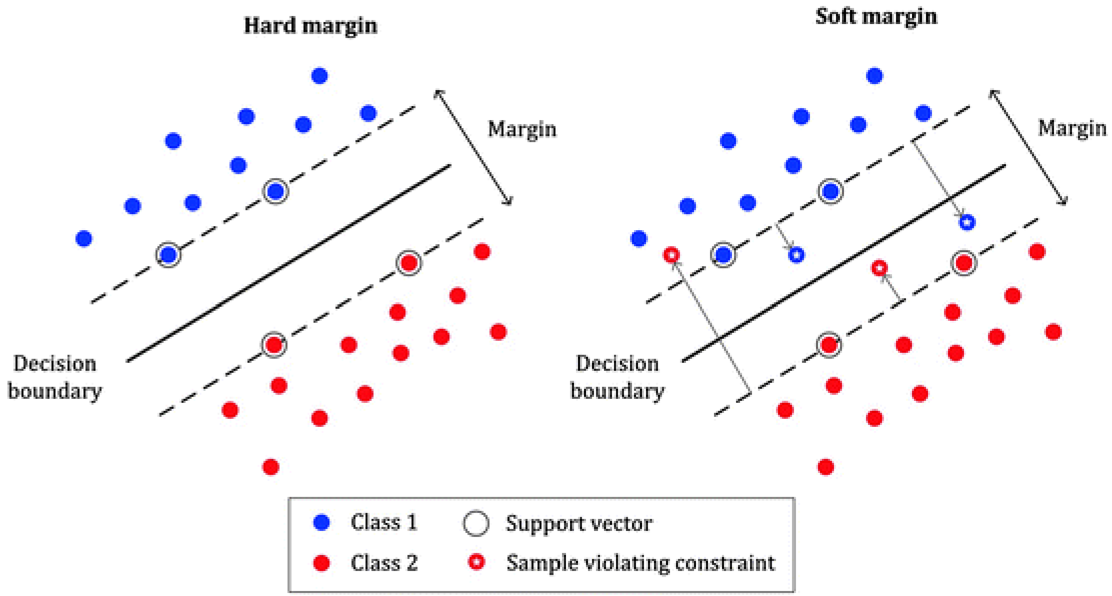
Support vectors are the data points nearest to the hyperplane, the points of a data set that, if removed, would alter the position of the dividing hyperplane. Because of this, they can be considered the critical elements of a data set.

And what is margin?



The distance between the hyperplane and the nearest data point from either set is known as the **margin**. The goal is to choose a hyperplane with the **greatest possible margin between** the hyperplane and any point within the training set, giving a greater chance of new data being classified correctly.

Soft and Hard Margin Classification



Hard margin classification: all instance have to be off the street (margin).
Soft margin: Some instances are allowed to be on the street and some can be on the margin line.

Hard Margin Classification

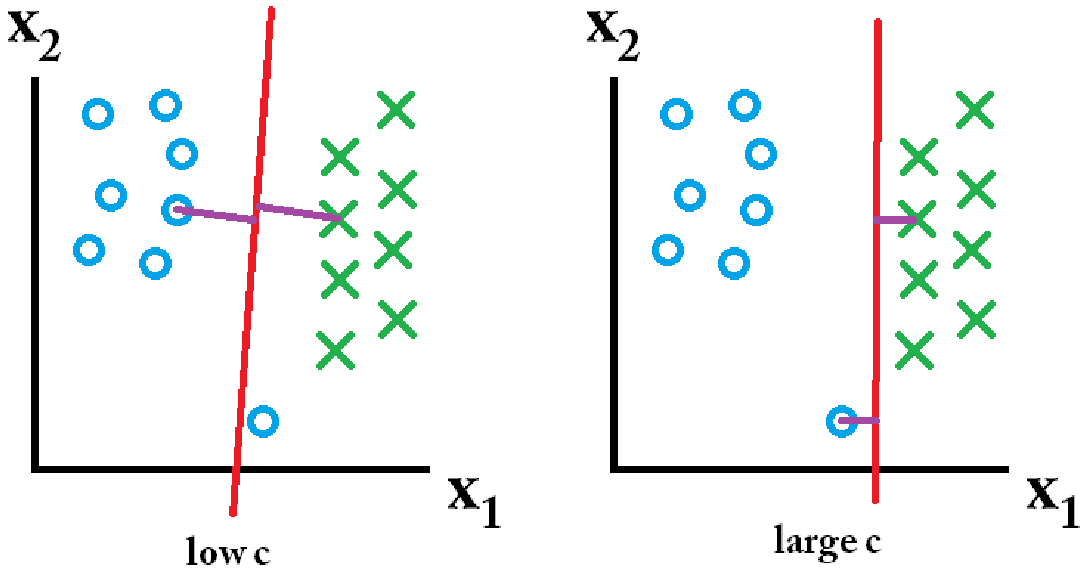
- Hard margin classification works only of the classes are easy to separate
- Sensitive to outliers (particularly with a narrow margin)
- Lack generalization capability
- Danger of overfitting
- Will only work with linear separable data

Soft Margin Classification

- Allows data to be “on the street” (within the margin)
- Less prone to outliers
- Some data on the street can be wrongly classified
- Can be mad more robust by limiting outliers
- Danger of underfitting

The over all goal is to **keep the margin** as **wide as possible** and **limiting the margin violations** (i. e. the instances that end up in the “middle of the street” or even on the wrong side.)

Finetuning of SVMs

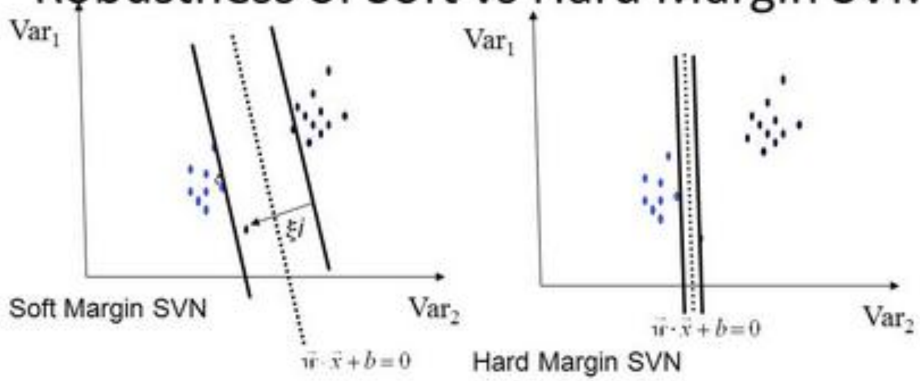


In Scikit Learn this balance can be controlled by using the **hyperparameter C**.

low C = wider street and more margin violations
larger C = narrower street and fewer margin violations

Soft vs. Hard Margin

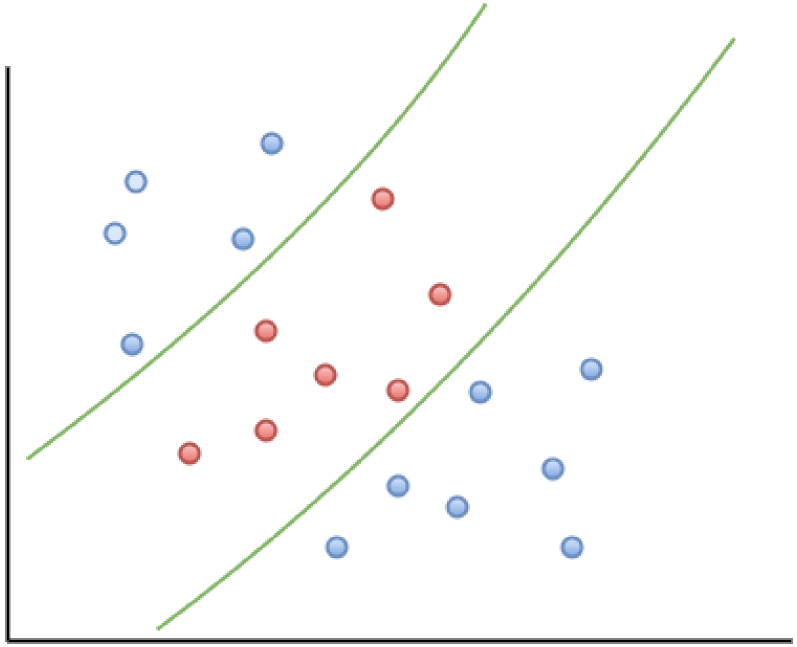
Robustness of Soft vs Hard Margin SVMs



- Soft margin – underfitting
- Hard margin – overfitting

Trade-off: width of the margin vs. no. of training errors committed by the linear decision boundary b-it Bonn-Aachen International Center for Information Technology

Non-linear SVM Classification



If the dataset set consist of not linearly separable data, it is possible to add non-linear features, a non-linear kernel such as the polynomial or the Gaussian RBF kernel (default in Scikit Learn).

Bonn-Aachen International Center for Information Technology

Support Vector Machines (SVM)

Pros

- Accuracy
- Works well on smaller and medium-sized datasets
- More efficient

Cons

- Less suited for larger data sets
- Less effective on noisier data sets with overlapping classes

Typical Use-Cases for SVMs

- Text classification tasks (e. g. spam filter, sentiment analysis)
- Image recognition challenges
- Handwritten digital recognition (postal recognition service)
- Classification tasks in medical/Life Science data

Kernel Choices

Which kernels are available in Sckit Learn:

- Linear kernel (we are going to use it in this taks)
- RBF kernel (Radial Basis Function, default in SciKit Learn)
- Poly (polynomial kernel)
- Sigmoid (sigmoid kernel)
- Precomputed kernels

You will mostly encounter and use the first three options in this course.

Bonn-Aachen International Center for Information Technology

Hints for Code

```
# Import SVM classifier
from sklearn.svm import SVC
classifier =  SVC(kernel ='linear')# linear kernel
```

Project 1

Your task: Use the diabetes and heart dataset you have been provided with in the previous project and apply SVM on this.

First apply the default kernel and then use the linear kernel.

1. Which kernel performs best?
2. Which ML algorithm perform best on the provided data sets, K-Nearest Neighbor or SVM?