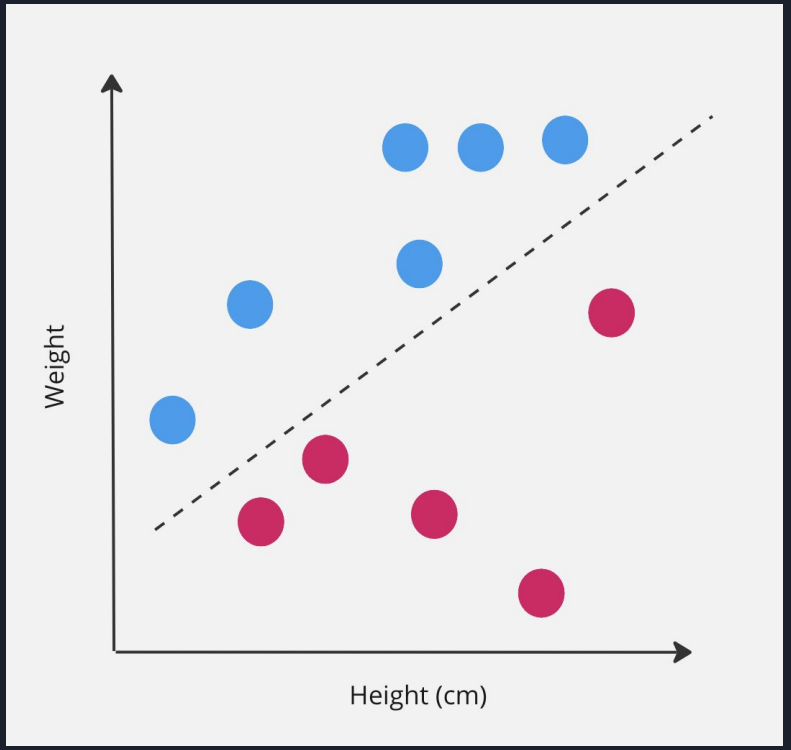
Support Vector Machines

Av Eirik[2]

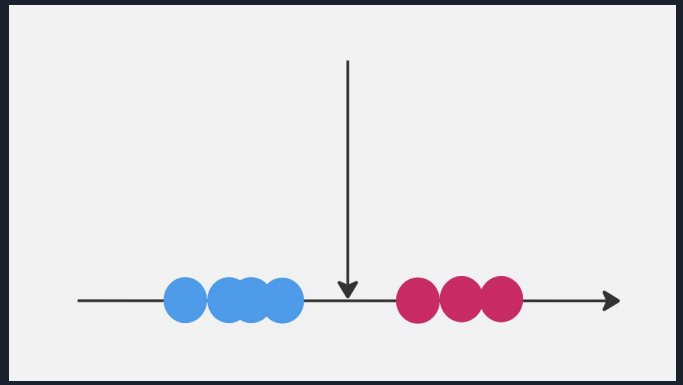
What?!

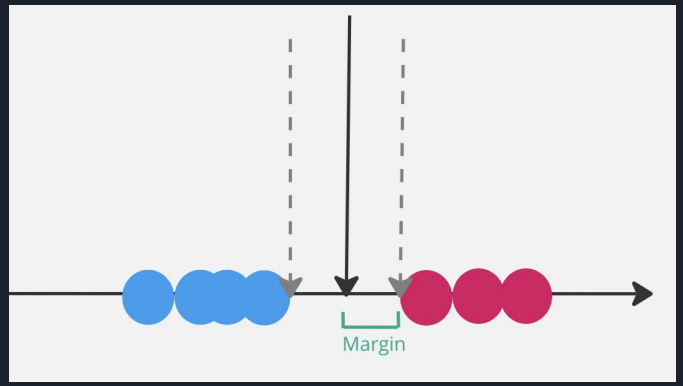
- Linear classifier
  - Takes an input and gives a class
  - Creates a straight line based on the training data. Separates points above or below the line
- We will focus on main ideas and purpose



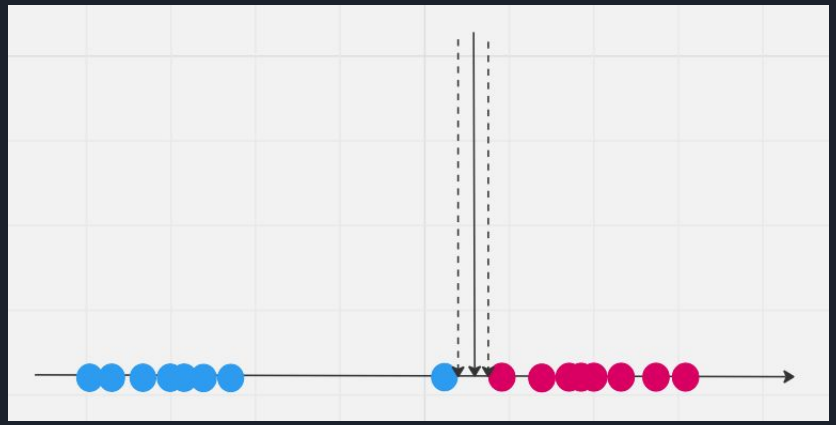
One Dimension

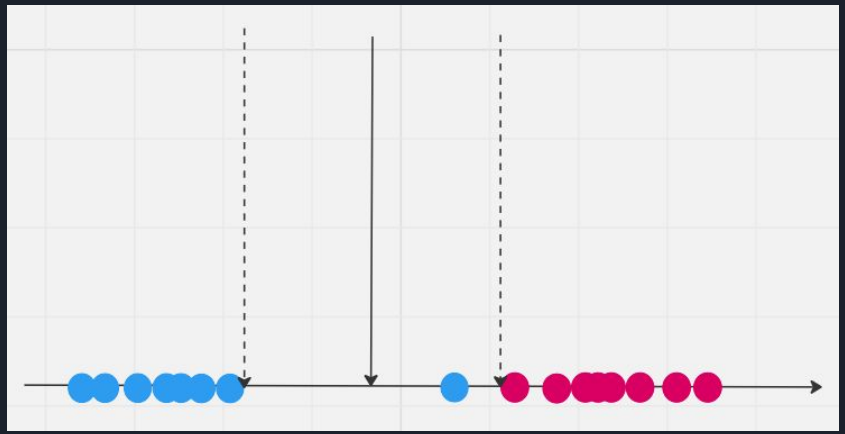
- Example is linearly separable
- Margin distance from closest point to the separator
- Maximize margin to separator
- Support vectors are used find the decision boundary





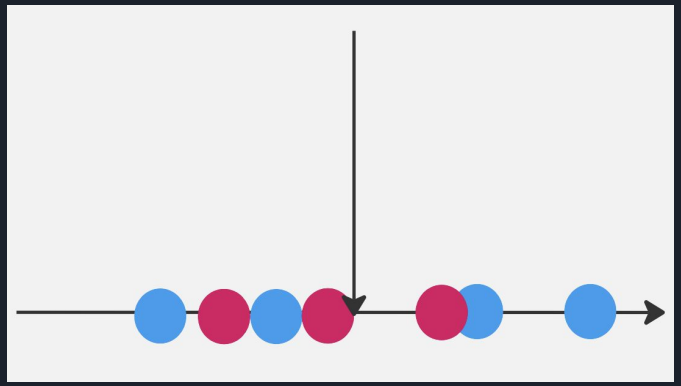
- Outliers are bad
- Sometimes we allow misclassification to maximize the margin





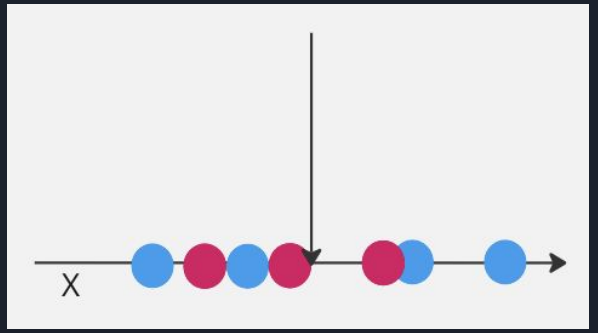
Problem

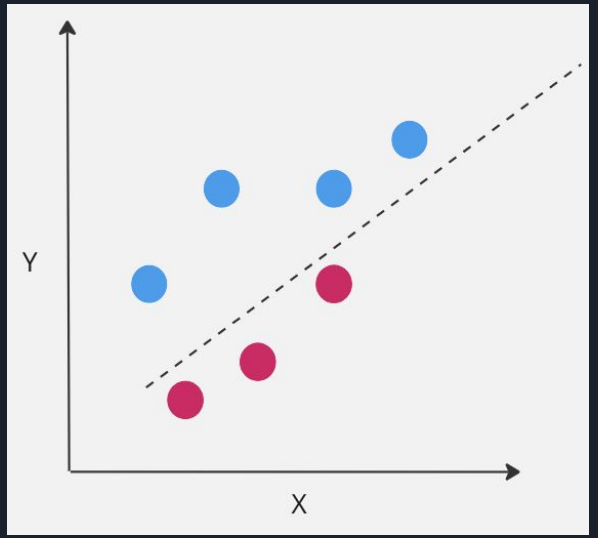
- What if the data is not linearly separable?



Solution

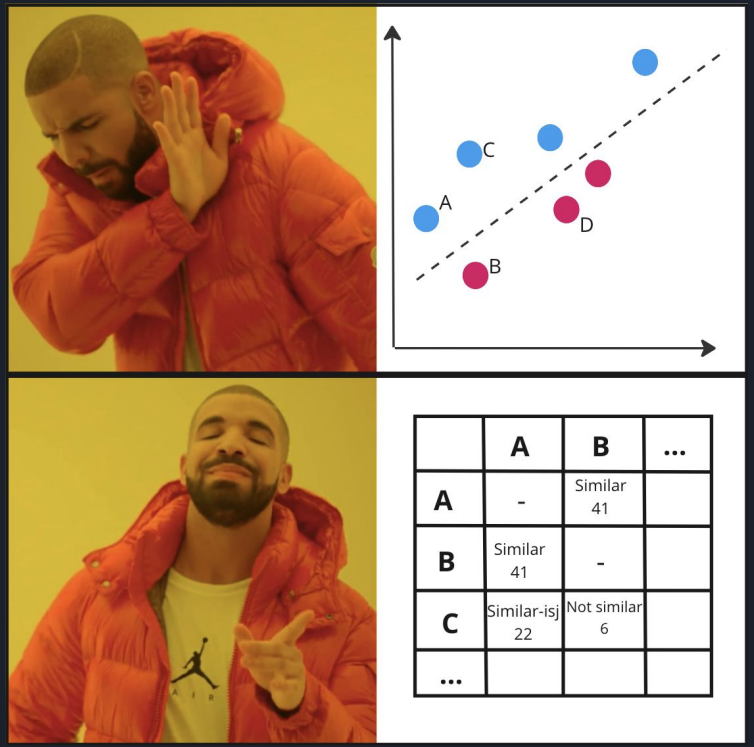
- What if there was a way to increase the number of dimensions, so we could find a decision boundary?
- The kernel trick goes brr





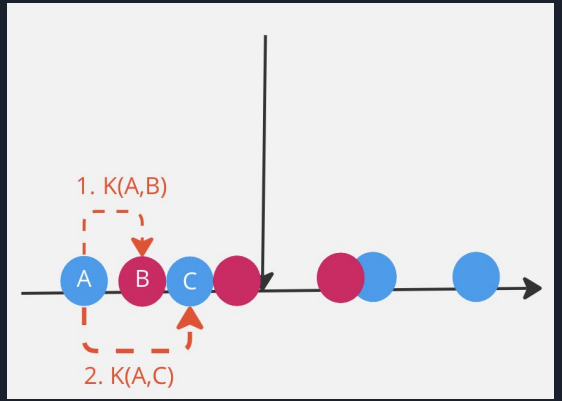
Kernel functions

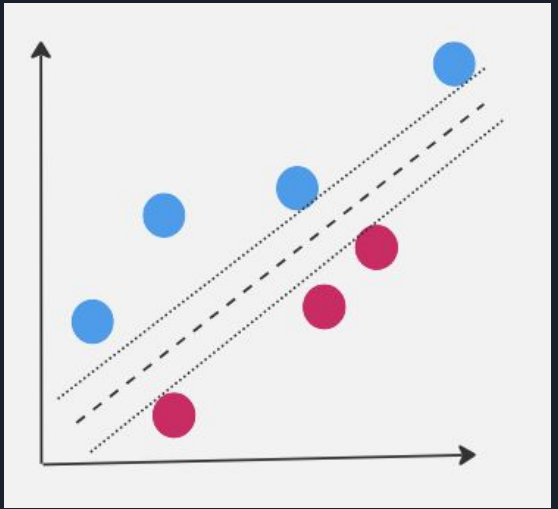
- Kernel functions does not actually transform the data into a new vector space
- Matrix with similarity between K(x,y) for for every pair of datapoints
- When training: decision boundary (and support vectors) are decided based on matrix



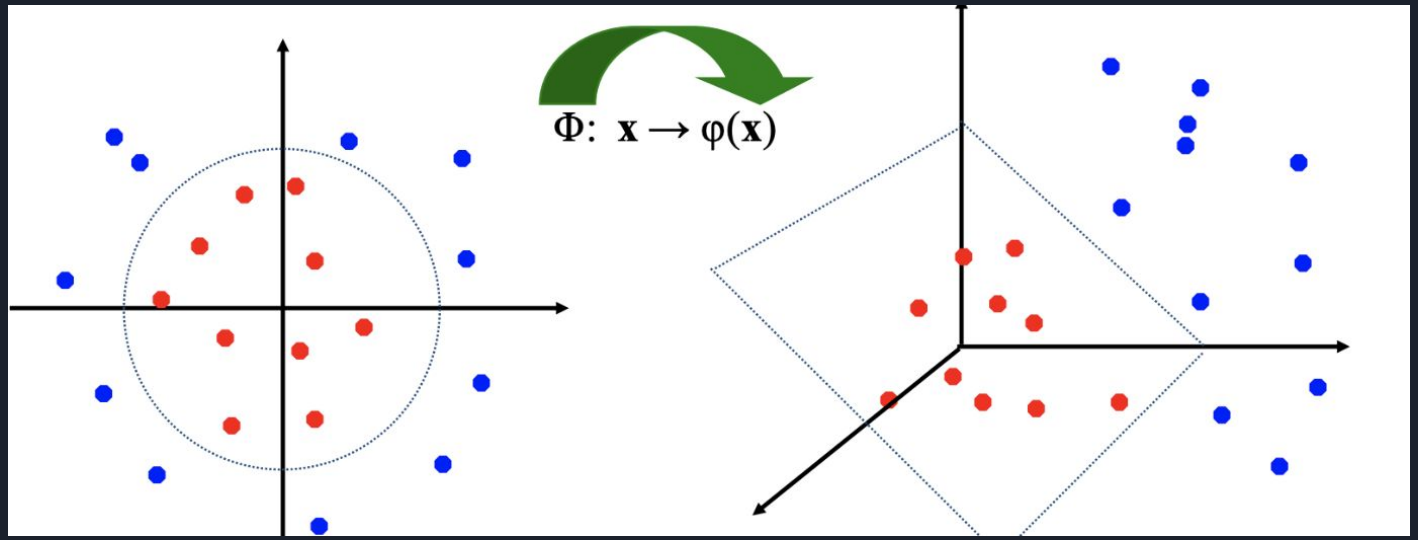
Types of functions

- Polynomial Kernel functions
  - computes relationships between each pairs in *d* dimensions
  - $K(x,y) = (x \cdot y+c)^d$
  - Constant value and dimensional variable
- Radial kernel functions
  - Theoretically works in infinite dimension
  - Weights points similarly to nearest neighbour algorithm, useful for clustered datasets
  - $K(x,y) = e^{(-\gamma \| x-y \| ^2)}$





Multiple dimensions



Pros for Support Vector Machines

- Works well on small training sets
- Works well on high dimensional classifying problems
- Work on data which is not linearly separable

Cons

- Sensitivity to Noise and Outliers:
  - Outliers can affect margins
- Computational Complexity:
  - The time complexity often cubic based on number of datapoints
- Binary Classifier:
  - Multiple classes may require methods like one vs rest
  - There are better algorithms for huge datasets

Thanks for your attention!