Chapter 9

Classification Continued

Classification Algorithms

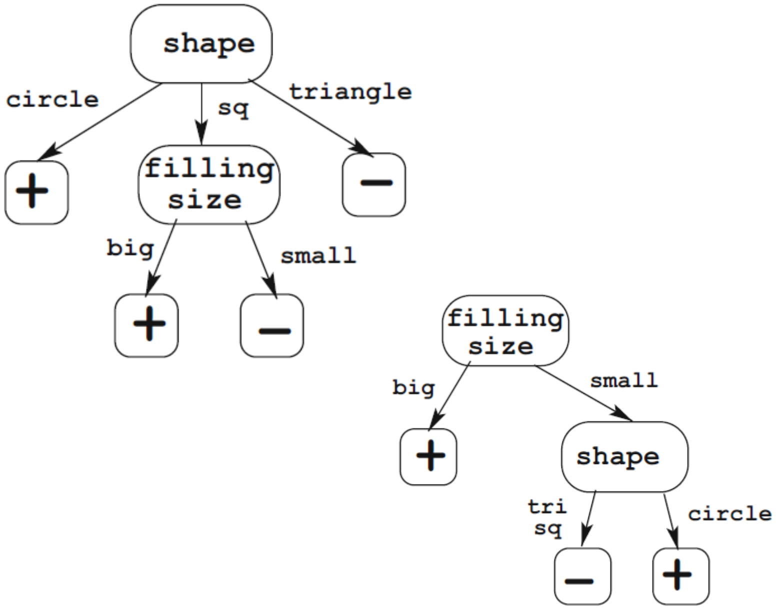
- Dozens of algorithms exist and a lot of them have many variations
- The algorithms can be *classified* into 4 categories
  - Distance-based algorithms
  - Probability-based algorithms
  - **Search-based algorithms**
    - **Decision Tree**
  - **Optimization-based algorithms**
    - **Support Vector Machine**

Decision Tree

What is it?

- A technique to create easily interpretable flowchart-like models
- The new (classifiable) object starts at the root node
- At each node, the object travels down based on the value of **one** of it's attribute
- The problem space is *split* by the node along the axis of the attribute
- Leaf nodes are output nodes, at each leaf node we have an assigned output value

Example



Example	crust size	shape	filling size	Class
$e1$	big	circle	small	**pos**
$e2$	small	circle	small	**pos**
$e3$	big	square	small	**neg**
$e4$	big	triangle	small	**neg**
$e5$	big	square	big	**pos**
$e6$	small	square	small	**neg**
$e7$	small	square	big	**pos**
$e8$	big	circle	big	**pos**

Example from *An Introduction to Machine Learning* by *Miroslav Kubat*

Decision Tree Induction

High level algorithm overview

Let T be the training set.

grow(T):

(1) Find the attribute, *at*, that contributes the maximum information about the class labels.
(2) Divide T into subsets, T_i, each characterized by a different value of *at*.
(3) For each T_i:
If all examples in T_i belong to the same class, then create a leaf labeled with this class; otherwise, apply the same procedure recursively to each training subset: *grow(T_i)*.

from *An Introduction to Machine Learning* by *Miroslav Kubat*

Decision Tree

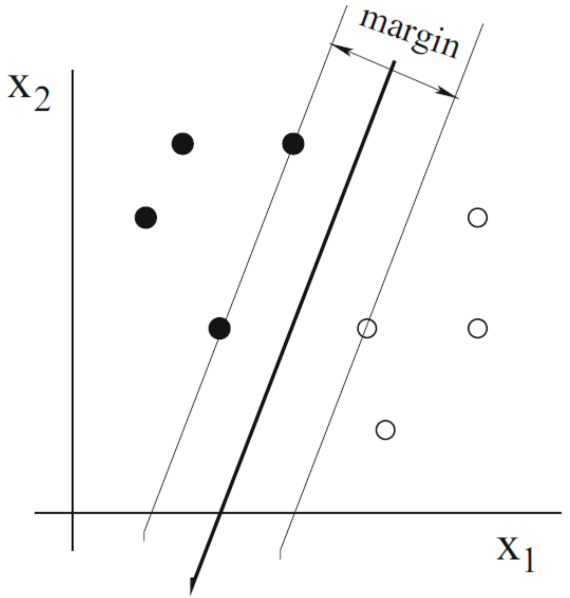
- **Pros**
  - Its simple and interpretable as flowchart or a set of rules
  - Very robust: Can handle outliers and missing values, no need to normalize, does not care about attribute correlation (all thanks to handling one attribute at a node)
- **Cons**
  - Fails at complex models where attribute interrelations are important
  - Can only split along an axis
  - Only able to learn $x_i <= a$ rules, where x_i is a predictive attribute and a is a constant

Support Vector Machine

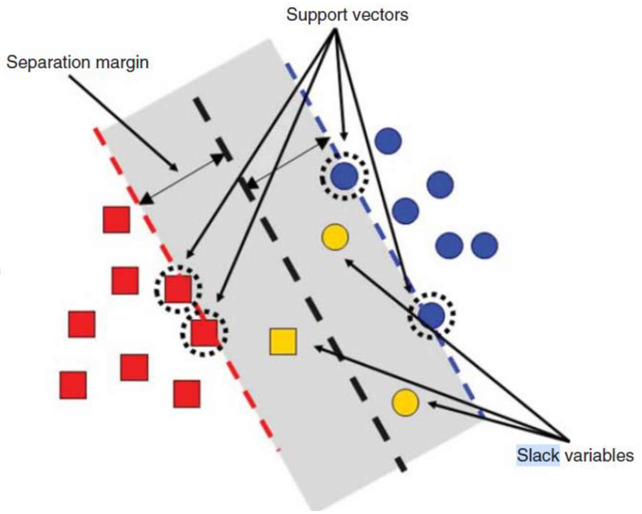
What is it?

- A technique allowing us to create good generalizing models that separate the problem space
- Unlike logistic regression the model clearly decides the class label instead of probabilistic result
- Unlike Neural Networks we find the most optimal solution to split the data by finding the line that maximizes the margin with respect to the support vectors
- Introduces the **kernel trick** to transform data into linearly separable representation

Support Vector Machine



from *An Introduction to Machine Learning* by *Miroslav Kubat*



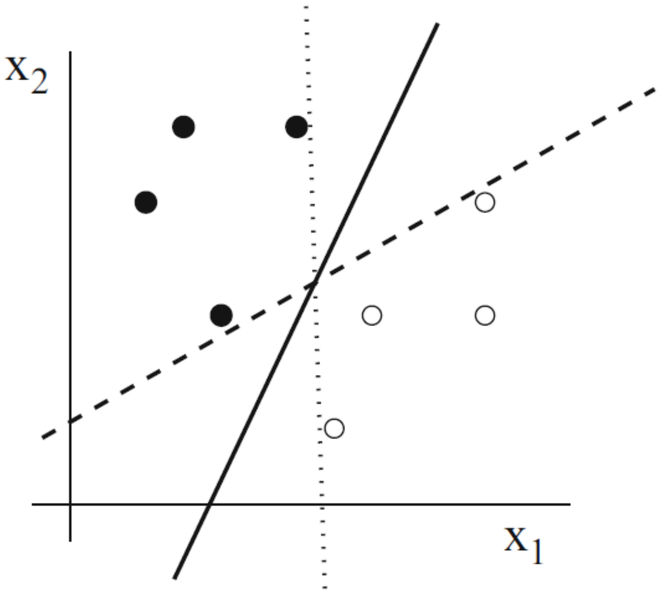
from *A General Introduction to Data Analytics* by *by João Moreira et al.*

Illustration

- The thick line is the class separator, the thin lines are the support vectors for each class
- The class separator is the best fit for maximizing margin size
- We can allow some *slack variables* inside the margin zone to increase margin size

Support Vector Machine

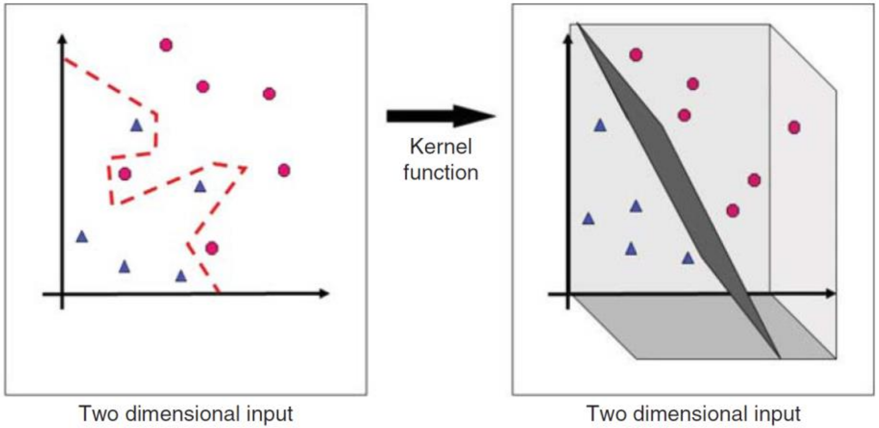
- Out of the many possible separators SVM will find the most optimal one
- Greater margin means better generalized model



Support Vector Machine

Kernel function

- A technique to increase dimensionality in order to transform a non-linear problem into a linear one
- There are also more advanced kernels that can solve non-linear problems these are Radial Basis Function (RBF) and Polynomial kernel



from *A General Introduction to Data Analytics* by *by João Moreira et al.*

Support Vector Machine

- **Pros**
  - Not random, same results achieved between runs (deterministic)
  - Good performance in many problems
  - Good theoretical foundations
- **Cons**
  - Very sensitive to hyperparameter values
  - Sensitive to outliers, magnitude difference between variables (needs normalization)
  - Training time grows at least quadratically with increased training samples

References, Literature, further reading

- Chapter 6 of *Introduction to Machine Learning* by *Miroslav Kubat*
- https://towardsdatascience.com/understanding-support-vector-machine-part-1-lagrange-multipliers-5c24a52ffc5e
- https://towardsdatascience.com/understanding-support-vector-machine-part-2-kernel-trick-mercers-theorem-e1e6848c6c4d
- Further methods and comparison https://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html
- https://www.deeplearningbook.org/contents/ml.html

Questions?