Changelog: 15 Oct; 12 Dec 2016

Models' Characterization (Supervised)

Machine Learning for Language Technology
ML4LT (2016)

Marina Santini
Department of Linguistics and Philology
Uppsala University

Outline

- Discriminative vs Generative
- Linearity
- Parametric & Non-Parametric
- Summary Table

Intro: Generative vs. Discriminative Methods

- A generative method only applies to probabilistic models. A model is generative if it gives us the model of the joint distribution of x and y together. It is called generative because you can generate with the correct probability distribution data points.

- Discriminative methods do not model probabilities at all, but they map the input to the output directly.
  - Conditional methods model the conditional distribution of the output given the input: $P(Y \mid X)$.
  - Linear

Terminology

- Probabilistic (Joakim explains very well that this word can be ambinguous, see videoclip 1 lect 07).
- Generative = based on JOINT probabilities
- Discriminative = either based on CONDITIONAL probabilities or based on other approaches.
- (*a conditional model is probabilistic, but not generative*)
- Linear: block feature vectors, inner (dot) product and weights.

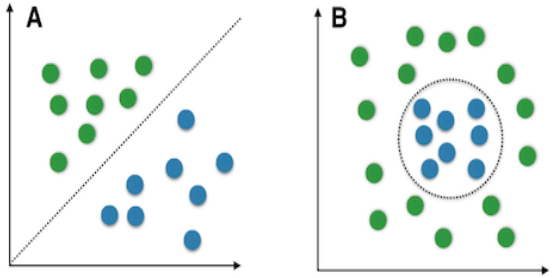
Model Types

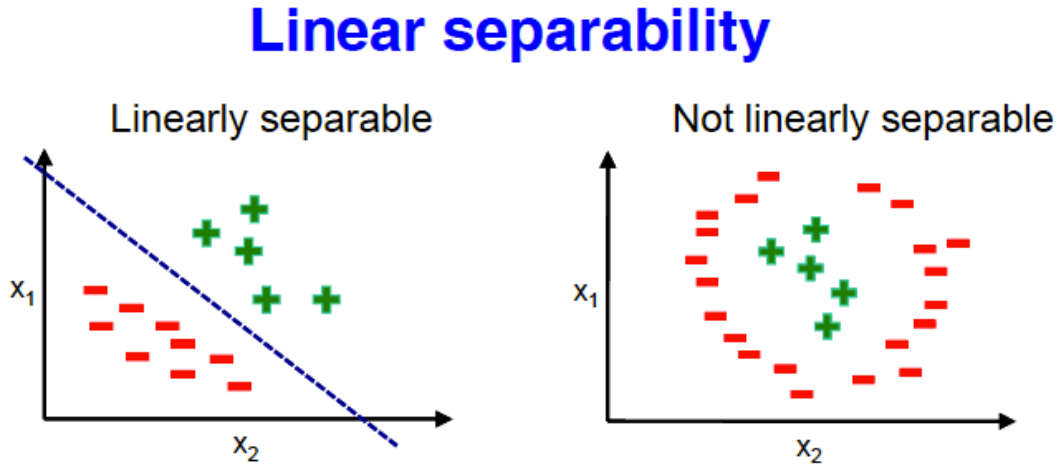
- **Generative Probabilistic Models**
  - Joint probabilities: $P(x,y)$
  - Ex: ***Naïve Bayes***, Hidden Markov Models, etc.

- **Discriminative Models**
  - Conditional probabilistic models
    - Conditional probabilities: $P(x \mid y)$
    - Ex: Logistic Regression, etc.
  - Discriminative models
    - Direct mapping input-output (no explicit probability modelling: $X : \{0,1\}$
    - Ex: ***Decision Trees, k-NN, Perceptron***, SVM, etc.

Not all problems are linearly separable

2-class problem: a *linear classifier* decides class membership by comparing a linear combination of the features to a threshold.





Two classes are **linearly separable**, if you can perfectly classify them with a linear decision boundary.
Why is this important? Linear classifiers can only draw linear decision boundaries.

Parametric vs. Non-parametric

- This definition of **parametric and non-parametric** is taken from Artificial Intelligence: A Modern Approach (3rd Edition) 3rd Edition by Stuart Russell (Author), Peter Norvig:
- **parametric**: "A learning model that summarizes data with a set of parameters of fixed size (independent of the number of training examples) is called a parametric model. No matter how much data you throw at a parametric model, it won't change its mind about how many parameters it needs." Basically it includes normal distribution and other known distributions.
- **non-parametric**: "Nonparametric methods are good when you have a lot of data and no prior knowledge, and when you don't want to worry too much about choosing just the right features."

Our models

- DTs and kNN are non-parametric (remember that Daume' talks about "unknown distribution").

- NB is parametric, but this trait can be overidden by using parameters (remember the –k parameter in weka's implementation; and remember the different behavior of NaiveBayesSimple and NaiveBayes in weka).
- Perceptron (simple version) is non-parametric (see a nice description here: < http://homepages.gold.ac.uk/nikolaev/311perc.htm >).

Summary Table

- Remember the distinction that Joakim made in lecture 7 about "being probabilistic". Here we refer to the models themselves, and not to internal calculations.
- Take all characterizations with a grain of salt (see slides 13 and 14 at the end of this presentation).

Classification algorithms	**Probabilistic**	**Generative**	**Discriminative (direct mapping)**	**Discriminative (probabilistic/ conditional)**	**Linear**	**Parametric/ Non-Parametric**	**Eager/ Lazy learning**		**Online/ Batch processing**
Decision Trees	no	no	**yes**	no	no	*non-parametric*	eager		online
k-NN	no	no	yes*	no	no	*non-parametric*	lazy		batch
Naïve Bayes	**yes**	**yes**	no	no	yes*	parametric	eager		online
Perceptron	no	no	*yes*	no	**yes**	*non-parametric*	eager		online

Modelling

- ML uses formal methods that might perform well on our data.
- The choice of using one method rather than another is *our choice*, and this choice should be based on evidences provided by the evaluation of the performance.
- A method tells us what sort of things we can learn.
- A model tells us what our inductive bias is (but some traits of the inductive bias can be overriden by the use of parameters).

Key Points

- Our goal when we choose a machine learning model is: the model should perform well on future, unseen data.
- The way in which we measure performance should depend on the problem we are trying to solve.
- There should be a strong relationship between the data that our algorithm sees at training time and the data it sees at test time (ie. training and test sets should have the same statistical distribution).

Models characterization can be controversial ! (1)
Study the algorithms and motivate your own stance !

A "middle-ground" of considering models with some degree of flexibility is preferred.

- Controversial (perceptron is parametric)
  - Parametric and Nonparametric Machine Learning Algorithms < http://machinelearningmastery.com/parametric-and-nonparametric-machine-learning-algorithms/ >

- Controversial (k-NN is probabilistic vs geometric) (see overleaf)

Models characterization can be controversial ! (2)

- Analysis of Parametric & Non Parametric Classifiers for Classification Technique using WEKA < http://www.mecs-press.org/ijitcs/ijitcs-v4-n7/IJITCS-V4-N7-6.pdf >;
- Is KNN a discriminative learning algorithm?
  < http://stats.stackexchange.com/questions/105979/is-knn-a-discriminative-learning-algorithm >

- About the linearity of NB
  - How is Naive Bayes a Linear Classifier?

  < http://stats.stackexchange.com/questions/142215/how-is-naive-bayes-a-linear-classifier >

The End