Naïve Bayesian Learning

Sven Koenig, USC

Russell and Norvig, 3rd Edition, Sections 13.5.2 and 20.2.2

These slides are new and can contain mistakes and typos.
Please report them to Sven (skoenig@usc.edu).

1

Naïve Bayesian Learning

- We now apply what we have learned to machine learning.

2

Inductive Learning for Classification

- Labeled examples

Feature_1	Feature_2	Class
true	true	true
true	false	false
false	true	false

Learn f(Feature_1, Feature_2) = Class from
f(true, true) = true
f(true, false) = false
f(false, true) = false
The function needs to be consistent with all labeled examples
and should make the fewest mistakes on the unlabeled examples.

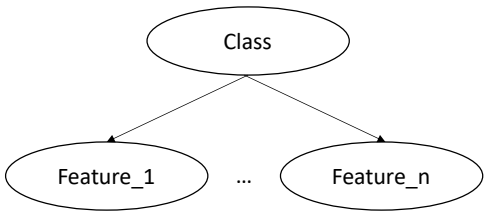
- Unlabeled examples

Feature_1	Feature_2	Class
false	false	?
true	true	?

3

Naïve Bayesian Learning

- Assume that the features are conditionally independent of each other given the class.
- This naïve (= potentially wrong) assumption keeps the number of parameters to be learned small.



Naïve Bayesian Network

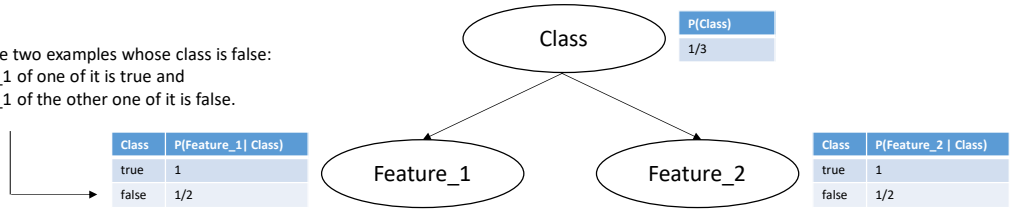
4

Naïve Bayesian Learning

- Use maximum-likelihood estimates to learn the probabilities in the conditional probability tables from the labeled examples, that is, use frequencies to estimate the probabilities.

Feature_1	Feature_2	Class
true	true	true
true	false	false
false	true	false

There are two examples whose class is false:
Feature_1 of one of it is true and
Feature_1 of the other one of it is false.

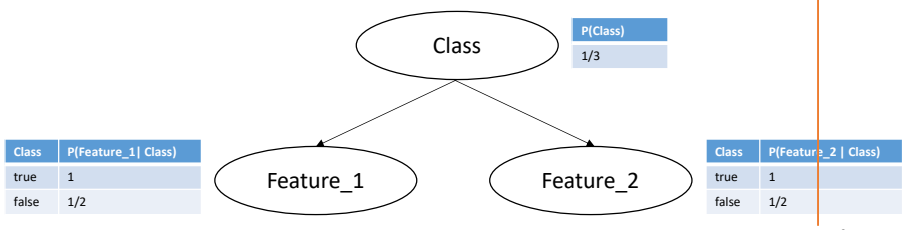


5

Naïve Bayesian Learning

- Calculate the probabilities of the class values given the feature values for unlabeled examples

Feature_1	Feature_2	Class
false	false	?



- Either make a probabilistic prediction by outputting P(Class | NOT Feature_1, NOT Feature_2) or a deterministic prediction by outputting the more likely class.

6

Naïve Bayesian Learning

- P(Class, NOT Feature_1, NOT Feature_2) = P(Class) P(NOT Feature_1 | Class) P(NOT Feature_2 | Class) = 1/3 0 0 = 0
- P(NOT Class, NOT Feature_1, NOT Feature_2) = P(NOT Class) P(NOT Feature_1 | NOT Class) P(NOT Feature_2 | NOT Class) = 2/3 1/2 1/2 = 1/6
- P(NOT Feature_1, NOT Feature_2) = P(Class, NOT Feature_1, NOT Feature_2) + P(NOT Class, NOT Feature_1, NOT Feature_2) = 0 + 1/6 = 1/6
- P(Class | NOT Feature_1, NOT Feature_2) = P(Class, NOT Feature_1, NOT Feature_2) / P(NOT Feature_1, NOT Feature_2) = 0 / (1/6) = 0
- P(NOT Class | NOT Feature_1, NOT Feature_2) = P(NOT Class, NOT Feature_1, NOT Feature_2) / P(NOT Feature_1, NOT Feature_2) = (1/6) / (1/6) = 1

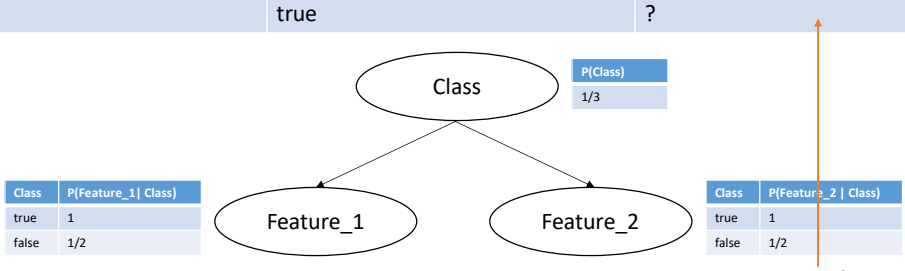
Feature_1	Feature_2	Class
false	false	P(Class | NOT Feature_1, NOT Feature_2) = 0 or false

7

Naïve Bayesian Learning

- Calculate the probabilities of the class values given the feature values for unlabeled examples

Feature_1	Feature_2	Class
true	true	?



- Either make a probabilistic prediction by outputting P(Class | Feature_1, Feature_2) or a deterministic prediction by outputting the more likely class.

8

Naïve Bayesian Learning

- P(Class, Feature_1, Feature_2) = P(Class) P(Feature_1 | Class) P(Feature_2 | Class) = 1/3 1 1 = 1/3
- P(NOT Class, Feature_1, Feature_2) = P(NOT Class) P(Feature_1 | NOT Class) P(Feature_2 | NOT Class) = 2/3 1/2 1/2 = 1/6
- P(Feature_1, Feature_2) = P(Class, Feature_1, Feature_2) + P(NOT Class, Feature_1, Feature_2) = 1/3 + 1/6 = 1/2
- P(Class | Feature_1, Feature_2) = P(Class, Feature_1, Feature_2) / P(Feature_1, Feature_2) = (1/3) / (1/2) = 2/3
- P(NOT Class | Feature_1, Feature_2) = P(NOT Class, Feature_1, Feature_2) / P(Feature_1, Feature_2) = (1/6) / (1/2) = 1/3

Feature_1	Feature_2	Class
true	true	P(Class | Feature_1, Feature_2) = 2/3 or true

9

Naïve Bayesian Learning

- For inductive learning, we typically demand that the learned function is consistent with all labeled examples (if possible). However, then we should have calculated P(Class | Feature_1, Feature_2) = 1.
- This is not possible because the naïve Bayesian assumption does not hold for the labeled examples (see next slide).
- Thus, a naïve Bayesian network cannot represent the labeled examples correctly and thus cannot represent all Boolean functions correctly.
- Just like for single perceptrons, this does not mean that they should not be used. They will make some mistakes for some Boolean functions but they often work well, that is, make few mistakes on the labeled and unlabeled examples.

10

Naïve Bayesian Learning

- The assumption that the features are conditionally independent of each other given the class does not hold for the labeled examples.

Feature_1	Feature_2	Class
true	true	true
true	false	false
false	true	false

- For example,
  P(Feature_1 | NOT Class) = 1/2 but
  P(Feature_1 | Feature_2, NOT Class) = 0.

11

Naïve Bayesian Learning

- Properties (some versus decision trees)
  - Are very tolerant of noise in feature and class values of examples
  - Can make deterministic or probabilistic predictions
  - Learn quickly even for large problems
  - Cannot represent all Boolean functions (since the naïve Bayesian assumption does not hold for all of them)
- Early application
  - Email spam detectors (where Feature_i = "How often does the *i*th word in a dictionary appear in the email?" and Class = "Is the email spam?")

12