Naïve Bayes

Instructor, Nero Chan Zhen Yu

Forward School

Naïve Bayes

- A statistical classification technique based on Bayes Theorem
- It is one of the simplest supervised learning algorithms.
- Naive Bayes classifier is the fast, accurate and reliable algorithm.
  - Naive Bayes classifiers have high accuracy and speed on large datasets.
- Naive Bayes classifier assumes that the effect of a particular feature in a class is independent of other features
  - For example, a loan applicant is desirable or not depending on his/her income, previous loan and transaction history, age, and location. Even if these features are interdependent, these features are still considered independently.

Forward School

Conditional Independence in NB

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

- P(h): the probability of hypothesis h being true (regardless of the data). This is known as the prior probability of h.
- P(D): the probability of the data (regardless of the hypothesis). This is known as the prior probability.
- P(h|D): the probability of hypothesis h given the data D. This is known as posterior probability.
- P(D|h): the probability of data d given that the hypothesis h was true. This is known as posterior probability.

Forward School

How does it work

- Assuming only 1 feature

- Step 1: Calculate the prior probability for given class labels
- Step 2: Find Likelihood probability with each attribute for each class
- Step 3: Put these value in Bayes Formula and calculate posterior probability.
- Step 4: See which class has a higher probability, given the input belongs to the higher probability class.

Example of a Spam Filter

- You have emails – normal and spam
  - Normal = 17
  - Spam = 7
- Normal:
  - Keywords = Dear (8), Friend (5), Lunch(3), Money (1)
- Spam:
  - Keywords = Dear (2), Friend (1), Lunch(0), Money (4)

Forward School

Example of a Spam Filter

- Normal
  - Keywords = Dear (8), Friend (5), Lunch(3), Money (1)
  - Probability (N = 17)
    - P(Dear|N) =
    - P(Friend|N) =
    - P(Lunch|N) =
    - P(Money|N) =
- Spam:
  - Keywords = Dear (2), Friend (1), Lunch(0), Money (4)

Forward School

Example of a Spam Filter

- Normal
  - Keywords = Dear (8), Friend (5), Lunch(3), Money (1)
  - Probability (N = 17)
    - P(Dear|N) = 8/17 = 0.47
    - P(Friend|N) = 5/17 = 0.29
    - P(Lunch|N) = 3/17 = 0.18
    - P(Money|N) = 1/17 = 0.06
- Spam:
  - Keywords = Dear (2), Friend (1), Lunch(0), Money (4)

Forward School

Example of a Spam Filter

- Normal
  - Keywords = Dear (8), Friend (5), Lunch(3), Money (1)
  - Probability = Dear (0.47), Friend(0.29), Lunch (0.18), Money (0.06)
- Spam:
  - Keywords = Dear (2), Friend (1), Lunch(0), Money (4)
    - P(Dear|S) =
    - P(Friend|S) =
    - P(Lunch|S) =
    - P(Money|S) =

Example of a Spam Filter

- Normal
  - Keywords = Dear (8), Friend (5), Lunch(3), Money (1)
  - Probability = Dear (0.47), Friend(0.29), Lunch (0.18), Money (0.06)
- Spam:
  - Keywords = Dear (2), Friend (1), Lunch(0), Money (4)
    - P(Dear|S) = 0.29
    - P(Friend|S) = 0.14
    - P(Lunch|S) = 0.00
    - P(Money|S) = 0.57

Forward School

How does it work

- Assuming only 1 feature

- Step 1: Calculate the **prior probability** for given class labels
- Step 2: Find **Likelihood probability** with each attribute for each class
- Step 3: Put these value in Bayes Formula and calculate **posterior probability.**
- Step 4: See which class has a higher probability, given the input belongs to the higher probability class.

Forward School

Prior probability

- Initial estimation of probability of each case
  - What's the likelihood a message is Normal or Spam
- Say, in every 12 messages, 8 are normal/non-spam
- P(N) = 8/ 8+4 = 8/12 = 0.67
- P(S) = 4/ 12 = 0.33

Let’s detect spam!

- A message came in with
  - “Dear Friend”

- Keywords = Dear (8), Friend (5), Lunch(3), Money (1)
- Probability = Dear (0.47), Friend(0.29), Lunch (0.18), Money (0.06)

- Score for Normal
  - $p(N \mid \text{Dear Friend}) = P(N) \times P(\text{Dear}|N) \times P(\text{Friend}|N)$
    $= 0.67 \times 0.47 \times 0.29$
    $= 0.09$

Forward School

Let’s detect spam!

- Now do the same for SPAM

- Keywords = Dear (2), Friend (1), Lunch(0), Money (4)
- Probability = Dear (0.29), Friend(0.14), Lunch (0), Money (0.57)

- Score for SPAM
  - p(S | Dear Friend) = P(S) X P (Dear|S) X P (Friend|S)

    = 0.33 X 0.29 X 0.14

    = 0.01

Forward School

FIGHT!

- NORMAL vs SPAM

- P (N| Dear Friend) = 0.09
- P (S| Dear Friend) = 0.01

- P (N| Dear Friend) > P(S| Dear Friend)
  - Message is likely – NORMAL!

Forward School

Try out something?

- Which phrase?
- Let's calculate… and detect spam

- Normal = P(N) = 0.67
  - Keywords = Dear (8), Friend (5), Lunch(3), Money (1)
  - Probability = Dear (0.47), Friend(0.29), Lunch (0.18), Money (0.06)
- Spam = P(S) = 0.33
  - Keywords = Dear (2), Friend (1), Lunch(0), Money (4)
  - Probability = Dear (0.29), Friend(0.14), Lunch (0), Money (0.57)

Try out something? V2.0!

- Which phrase?
- Let's calculate… and detect spam

- Normal = P(N) = 0.67
  - Keywords = Dear (9), Friend (6), Lunch(4), Money (2)
  - Probability = Dear (0.43), Friend(0.29), Lunch (0.16), Money (0.1)
- Spam = P(S) = 0.33
  - Keywords = Dear (2+1), Friend (1+1), Lunch(0+1), Money (4+1)
  - Probability = Dear (0.27), Friend(0.18), Lunch (0.09), Money (0.45)

Forward School