Classifiers –II (Naïve Bayesian Classification)

Bayes classification Methods

- Bayesian classifiers are <u>statistical classifiers</u> helps to predict class membership probabilities (the probability that a given tuple belongs to a particular class).
- **<u>Foundation:</u>** Based on Bayes' Theorem.
- **<u>Performance:</u>** A simple Bayesian classifier called as naïve Bayesian classifier, has comparable performance with decision tree and selected neural network classifiers
- **<u>Conditional Independence:</u>** Assumes the effect of an attribute value on a given class is independent of the value of other attributes.

Bayes' Theorem: Basics

- **Bayes' Theorem:**
  - Named after Thomas Bayes
  - Let X be a data sample ("evidence"): class label is unknown
  - Let H be some hypothesis such that X belongs to specified class C.
  - Classification is to determine P(H|X), (i.e., posteriori probability): the probability that the hypothesis holds given the observed data sample X
  - **X: 35 year customer and income $40,000**
  - **H: Customer X will buy a computer**
  - The P(H|X) reflects the probability the tuple X will buy a computer (class) given when customer's age and income are known (description of X)

Bayes' Theorem: Basics

- **P(H) (*prior probability of H*)**:
  - E.g., it is the probability that any given customer will buy a computer regardless of age, income and other information.
  - The prior probability is independent of X

- **P(X) (*prior probability* of X)**: It is the probability that a person from set of customers is 35 years and earns $40,000

- **P(X|H) (is the posterior probability of X conditioned on H):** It is the probability that a customer X is 35 years old and earns $40.000, given that the hypothesis holds (customer will buy computer)

- Calculate the posterior probability

  P(H|X)=P(X|H)P(H)/P(X)

Naiive Bayesian Classifier using Bayes' Theorem

- Let D be a training set of tuples represented by an n-D attribute vector $X = (x_1, x_2, \ldots, x_n)$
- Suppose there are m classes $C_1, C_2, \ldots, C_m$.
- Given a tuple X, the classifier will predict X belongs to the class having the highest posterior probability conditioned on X.
- The classifier predicts the X belongs to C_i if and only if
  $P(C_i|X) > P(C_j|X)$ for $1 <= j <= m$, j not equal to i
- Classification is to derive the maximum posteriori, i.e., the maximal $P(C_i|X)$ called as **maximum posteriori hypothesis**
- This can be derived from Bayes' theorem

$$P(C_i|X) = P(X_k|C_i)\,P(C_i)/P(X)$$

- Since P(X) is constant for all classes, maximize the numerator

Prediction Based on Bayes’ Theorem

- As $P(X)$ is constant for all classes only $P(X|Ci)P(Ci)$ needs to be maximized
- If $P(Ci)$ is not known assume all the classes are equally likely so maximimze $P(X|Ci)$
- Class prior probablites are estimated as $P(Ci)=|Ci,D|/|D|$
- Computing $P(X|Ci)$ is expensive.
- To reduce the computation the naive assumption of class-conditional independence is considered.
- To predict the class label of X, the classiifer predicts the class label of tuple X is Ci iff
  - $P(X|Ci)P(Ci)> P(X|Cj)P(Cj)$

Naïve Bayes Classifier

- Attributes are class-conditional independence
- Evaluate P(X|Ci) as

$$P(X|C_i)=\prod_{k=1}^{n} P(x_k|C_i)$$

- Estimate the probabilites from the training tuples and x_k refers to the value of the attributes A_k
- If A_k is categorical, $P(x_k|C_i)$ is the # of tuples of class C_i having value x_k for $A_{k,}$ divided by $|C_{i, D}|$ (# of tuples of C_i in D)

- If A_k is continous-valued, $P(x_k|C_i)$ is usually computed based on Gaussian distribution with a mean μ and standard deviation σ

and $P(x_k|C_i)$ is

$$g(x,\mu,\sigma)=\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$P(X|C_i)=g(x_k,\mu_{C_i},\sigma_{C_i})$$

Naïve Bayes Classifier: Training Dataset

Class:

C1:buys_computer = 'yes'

C2:buys_computer = 'no'

Data to be classified:

X = (age <=30, Income = medium,

Student = yes,

Credit_rating = Fair)

age	income	student	credit rating	com
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Naïve Bayes Classifier: An Example

- **P(C_i):** P(buys_computer = "yes") = 9/14 = 0.643

  P(buys_computer = "no") = 5/14= 0.357

- Compute **P(X|C_i)** for each class and attribute

- P(age = "<=30"|buys_computer = "yes") = 2/9 = 0.222

  P(age = "<= 30"|buys_computer = "no") = 3/5 = 0.6

  P(income = "medium" | buys_computer = "yes") = 4/9 = 0.444

  P(income = "medium" | buys_computer = "no") = 2/5 = 0.4

  P(student = "yes" | buys_computer = "yes) = 6/9 = 0.667

age	income	student	credit_rating	com
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Naïve Bayes Classifier: An Example

P(student = “yes” | buys_computer = “no”) = 1/5 = 0.2

P(credit_rating = “fair” | buys_computer = “yes”) = 6/9 = 0.667

- P(credit_rating = “fair” | buys_computer = “no”) = 2/5 = 0.4X = (age <= 30 , income = medium, student = yes, credit_rating = fair)

$P(X|C_i)$: P(X|buys_computer = “yes”) = 0.222 x 0.444 x 0.667 x 0.667 = 0.044

P(X|buys_computer = “no”) = 0.6 x 0.4 x 0.2 x 0.4 = 0.019

$P(X|C_i)*P(C_i)$: P(X|buys_computer = “yes”) * P(buys_computer = “yes”) = 0.028

P(X|buys_computer = “no”) * P(buys_computer = “no”) = 0.007

Therefore, **highest probability X belongs to class (“buys_computer = yes”)**

Avoiding the Zero-Probability Problem

- Naïve Bayesian prediction requires each conditional prob. be non-zero. Otherwise, the predicted prob. will be zero

$$P(X|C_i)=\prod_{k=1}^{n} P(x_k|C_i)$$

- Ex. Suppose a dataset with 1000 tuples, income=low (0), income= medium (990), and income = high (10)
- Use Laplacian correction (or Laplacian estimator)
  - Adding 1 to each case

Prob(income = low) = 1/1003

Prob(income = medium) = 991/1003

Prob(income = high) = 11/1003

  - The "corrected" prob. estimates are close to their "uncorrected" counterparts

Naïve Bayes Classifier: Comments

- **Advantages**
  - Easy to implement
  - Good results obtained in most of the cases
- **Disadvantages**
  - Assumption: class conditional independence, therefore loss of accuracy
  - Practically, dependencies exist among variables
    - E.g., hospitals: patients: Profile: age, family history, etc.

Symptoms: fever, cough etc., Disease: lung cancer, diabetes, etc.

    - Dependencies among these cannot be modeled by Naïve Bayes Classifier
- How to deal with these dependencies? **Bayesian Belief Networks**