ch4: Bayesian Classification

Bayes Rule(P9)

$$P(C_i|x) = \frac{p(x|C_i)P(C_i)}{p(x)}, \qquad i = 1,2$$

$P(C_i)$: **prior probability** of C_i (before observing x)

$p(C_i|x)$: **posterior probability** of C_i (after observing x)

$p(x|C_i)$: probability of x given C_i (likelihood)

$p(x)$: probability that x will be observed (evidence)

$$p(x) = P(C_1)p(x|C_1) + P(C_2)p(x|C_2)$$

Bayes Decision Rule(P11)

$$P(\text{error}|x) = \begin{cases} P(C_1|x) & \text{if we decide } C_2 \\ P(C_2|x) & \text{if we decide } C_1 \end{cases}$$

- the average probability of error:

  $$P(\text{error}) = \int_{-\infty}^{\infty} P(\text{error}|x)p(x)dx$$

  This is minimized if for every x we ensure that P(error|x) is as small as possible

- the decision rule:

  Classify x into C_1 if $P(C_1|x) > P(C_2|x)$

  $$P(\text{error}|x) = \min(P(C_1|x), P(C_2|x))$$

- equivalently, classify x into C_1 if

  $$\frac{p(x|C_1)p(C_1)}{p(x)} > \frac{p(x|C_2)p(C_2)}{p(x)}$$

  $$p(x|C_1)p(C_1) > p(x|C_2)p(C_2)$$

- **Special Cases(P12)**

$$\boxed{p(x|C_1)P(C_1) > p(x|C_2)P(C_2)}$$

Special case 1: $p(x|C_1) = p(x|C_2)$

- the observation gives us no information about the state of nature
- decision based entirely on the prior probabilities

Special case 2: $P(C_1) = P(C_2)$

- decision based entirely on the likelihoods

Bayes Rule for K>2 Classes(P13)

- Bayes rule for general case (K mutually exclusive and exhaustive classes):

$$P(C_i \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid C_i)P(C_i)}{p(\mathbf{x})} = \frac{p(\mathbf{x} \mid C_i)P(C_i)}{\sum_{k=1}^{K} p(\mathbf{x} \mid C_k)P(C_k)}$$

- Optimal decision rule for Bayes classifier:

$$\text{Choose } C_i \text{ if } P(C_i \mid \mathbf{x}) = \max_k P(C_k \mid \mathbf{x})$$

Example(P14)

Question

A patient takes a lab test and the result comes back positive. The test returns a correct positive result in only 98% of the cases in which the disease is actually present, and a correct negative result in only 97% of the cases in which the disease is not present. Furthermore, 0.008 of the entire population have this cancer.

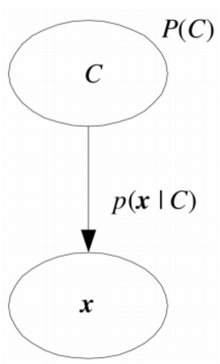
Does the patient have cancer or not?

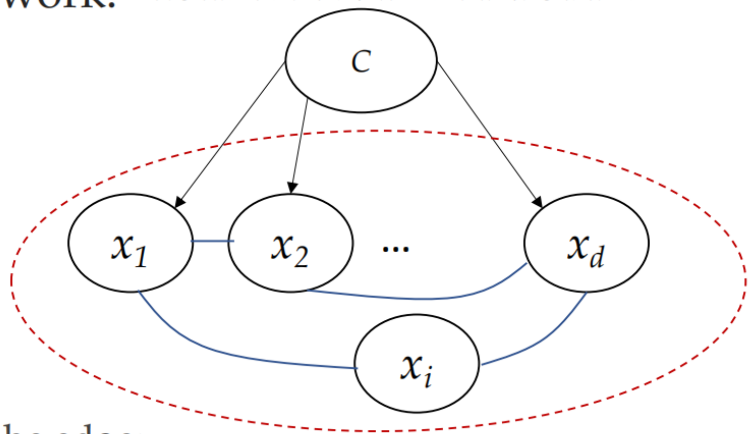
$$P(cancer) = 0.008 \quad P(\neg cancer) = 0.992$$
$$P(+|cancer) = 0.98 \quad P(-|cancer) = 0.02$$
$$P(+|\neg cancer) = 0.03 \quad P(-|\neg cancer) = 0.97$$
$$P(+|cancer)P(cancer) = 0.98(0.008) = 0.0078$$
$$P(+|\neg cancer)P(\neg cancer) = 0.03(0.992) = 0.0298$$

Answer: $\neg cancer$

Bayesian Networks for Classification(P17-18)

Classification amounts to infer the posterior of class **C** under a Bayes network. 分类相当于在贝叶斯网络下推断C类的后验。





- Bayes rule inverts the edge:

$$P(C \mid \mathbf{x}) = \frac{P(\mathbf{x} \mid C)P(C)}{P(\mathbf{x})} \Rightarrow P(C|x_1, x_2, \dots, x_d) = \frac{P(x_1, x_2, \dots, x_d|C)P(C)}{P(x_1, x_2, \dots, x_d)}$$

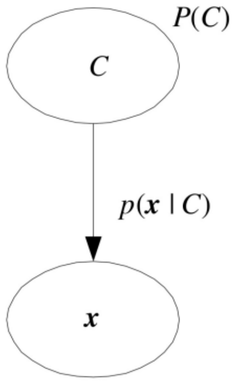
Input:

- a data sample $x = (x_1, x_2, \dots, x_d)$
- a fixed set of classes $C = \{C_1, \dots, C_j\}$.

Output:

- the most probable class $c \in C$:

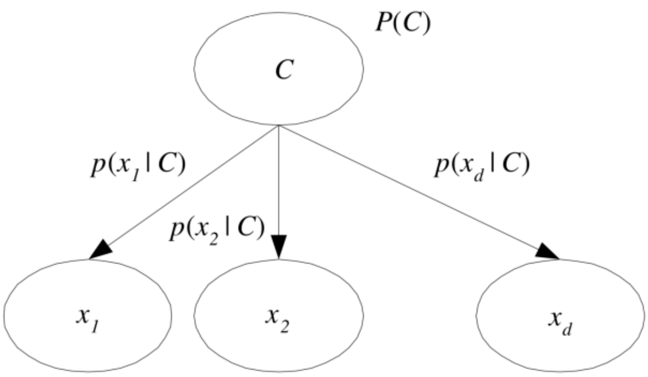
$$
\begin{aligned}
c_{\mathrm{MAP}} &= \arg\max_{c \in C} P(c|x) \\
&= \arg\max_{c \in C} \frac{P(x|c)P(c)}{P(x)} \\
&= \arg\max_{c \in C} P(x|c)P(c) \\
&= \arg\max_{c \in C} p(x_1, x_2, \dots, x_d|c)\, P(c)
\end{aligned}
$$



Naïve Bayes Independent Assumption:

Conditional Independence: assume the input features x_j are **independent** given the class c

$$P(x_1,\dots,x_n|c) = P(x_1|c) \bullet P(x_2|c) \bullet P(x_3|c) \bullet \dots \bullet P(x_n|c)$$



$$c_{\mathrm{MAP}} = \arg\max_{c \in C} p(x_1, x_2, \dots, x_d | c) \mathrm{P}(c)$$

$$c_{\mathrm{NB}} = \arg\max_{c \in C} P(c) \prod_{i=1}^{d} p(x_i|c)$$

Training the Naïve Bayes Classifier(P22)

$$c_{\mathrm{NB}} = \arg\max_{c \in C} P(c) \prod_{i=1}^{d} p(x_i|c)$$

Training amounts 相当于 to estimating parameters: $P(c)$'s, $P(x_1|c)$, … , $P(x_d|c)$ from data.

How to estimate each P(c)?

- Straightforward

How to estimate $P(x_i|c)$ for each c?

$$\hat{P}(x_i|c) \leftarrow \frac{\mathrm{count}(x_i, c)}{\sum_{x \in |x|} \mathrm{count}(x, c)}$$

\# training samples for which C=c and x = x_i

\# training samples for which C=c

Question

What if none of the training instances with class c have attribute x_i?

$$\hat{P}(x_i|c) = 0 \quad \rightarrow \quad \hat{P}(c) \prod_i \hat{P}(x_i|c) = 0$$

no chance to be classified as c, even
if all other attributes values suggest c

- Laplace smoothing: add a virtual count of 1 to each attribute value.

$$\hat{P}(x_i|c) \leftarrow \frac{\mathrm{count}(x_i, c) + 1}{\sum_{x \in |x|} (\mathrm{count}(x, c) + 1)}$$

|x| = Vocabulary, which denotes the number of different values of attribute x.

The Naïve Bayes Algorithm(P24)

Naive_Bayes_Learn(*examples*)

```
begin
    for each class c do
        p̂(c) ← estimate p(c)
        for each attribute value x_i of each attribute x do
            p̂(x_i|c) ← estimate p(x_i|c);
        end
    end
end
```

Classify_New_Instance(x)

```
begin
        c_NB = argmax_{c∈C} P(c) ∏_{i=1}^{d} p(x_i|c)
end
```

$$c_{\mathrm{NB}} = \arg\max_{c \in C} P(c) \prod_{i=1}^{d} p(x_i|c)$$

- **Example: Play Tennis(P24-25)**

Advantages(P29)

- **Fast**
  - ▻ on training, requires only a single pass over the training set
  - ▻ on testing, also fast
- **Competitive performance**
  - ▻ when assumption of independence holds, NB performs better
  - ▻ it also perform well in multi class prediction
- **Simple to update upon additions or deletions of training examples**
  - ▻ easy to maintain

Disadvantages(P30-31)

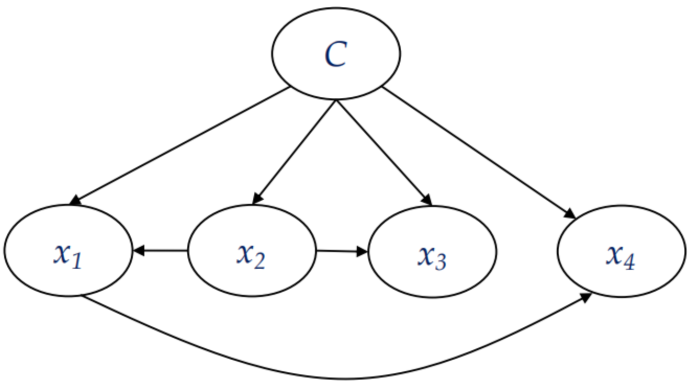
Conditional Independence Assumption

- Often violated
- But it works surprisingly well anyway!
- Don't need estimated posterior $\hat{P}(c|x)$ to be correct
- Need only that

$$\arg\max_{c\in C} \hat{P}(c)\prod_i \hat{P}(x_i|c) = \arg\max_{c\in C} P(c)P(x_1,\dots,x_d|c)$$

Underfitting

- The complexity of Naïve Bayes classifier is fixed and low.
- Bayesian (belief) network classifier can relax the assumption.



Applications(P32)

- **Real time Prediction:** NB is an eager learning classifier and it is sure fast.
- **Multi class Prediction:** NB can predict the probability of multiple classes of target variable.
- **Text classification** (e.g., spam filtering/sentiment analysis): NB is mostly used in text classification (due to better result in multi-class problems and independence rule) have higher success rate as compared to other algorithms.

Example: Spam Filtering(P35-39)