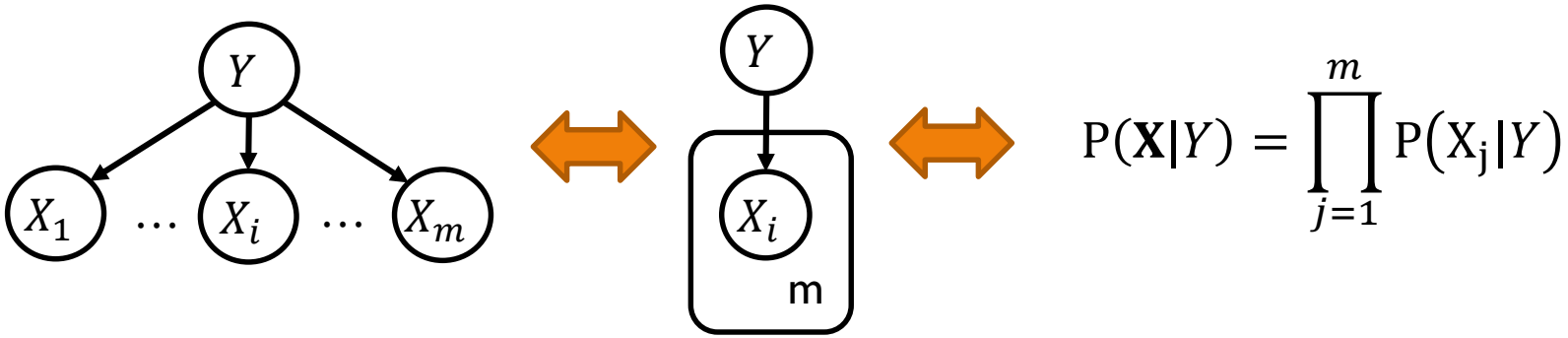
Bayesian Classification: Exemple of Naive Bayes

Naive Bayes:
a very simple Bayesian classification method

(Naive) hypothesis:

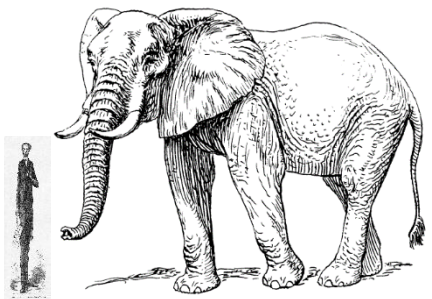
Features X_i are independent given class Y $(\forall i, \forall j \neq i, X_i \perp X_j | Y)$



Ignore information redundancy → requires preprocessing (PCA, etc)

Interpretation: the class mostly determines the characteristics distribution

e.g. : animal species mostly determine size and weight



A small elephant will always be bigger than a tall man

Naive Bayes: parameterization & prediction

Parameters (in the discrete case):

- CPT of Y: $(\mathrm{p}_c)_{1 \le c \le C} = P_Y(c)$
- CPT of $(\mathrm{X_i})_{1 \le i \le M}$: $\left(\mathrm{p_{j,v_j,c}}\right)_{\substack{1 \le c \le C, \\ 1 \le j \le m, \\ 1 \le v_j \le n_j}} = P\left(\mathrm{X_j} = v_j | Y = c\right)$

Prediction of Y given $\mathbf{X}, \mathcal{O}$:

$$\mathrm{P}(\mathrm{Y}|\mathbf{X}, \mathcal{O}) = \mathrm{P}(\mathbf{X}|\mathrm{Y}, \mathcal{O}) \cdot \frac{P(Y|\mathcal{O})}{P(\boldsymbol{X}|\mathcal{O})} \propto \left(\prod_{j=1}^{m} \mathrm{P}\left(\mathrm{X_j} = x_j | Y, \mathcal{O}\right) \right) \times P(Y|\mathcal{O})$$

$$\rightarrow \boxed{\hat{\mathrm{y}}(\boldsymbol{X}) = \underset{\mathrm{c}}{\operatorname{argmax}} \left(\left(\prod_{j=1}^{m} p_{j,x_j,c} \right) \times p_c \right)}$$

Classical Naive Bayes: Maximum likelihood estimation

Estimation of parameters $\mathrm{P}\left(\mathrm{X_j}|Y,\mathcal{O}\right)$ and $P(Y|\mathcal{O})$ by counting

Given observations $\mathcal{O} = \{(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_n}, y_n)\}$ with $\boldsymbol{x_i} = \left(x_{i,1}, \ldots x_{i,m}\right)$

$$\mathrm{p_{j,v_j,c}} = \hat{P}\left(\mathrm{X_j} = v|Y = c, \mathcal{O}\right) = \frac{N(y_i = c \; and \; x_{i,j} = v)}{N(y_i = c)}$$

$$\mathrm{p_c} = \hat{P}(Y = c|\mathcal{O}) = \frac{N(y_i = c)}{n}$$

Weka demo: mushroom classification

Naive Bayes: Example of Spam detection

Given 10 examples of emails from the banking sector:

earn	million	account	password	Class y
1	1	0	0	Spam
0	0	1	1	Spam
0	1	1	0	Not spam
1	1	0	0	Spam
0	0	0	0	Not spam
1	0	0	0	Spam
1	0	0	0	Not spam
0	0	0	1	Spam
1	0	1	1	Spam
0	1	1	1	Not Spam

Problem: predict class of messages "earn million", "million account" and "account password"

Naive Bayes: Solution

Feature x_i	$P(x_i|\text{spam})$	$P(x_i|\neg\text{spam})$
earn	$4/6$	$1/4$
million	$2/6$	$2/4$
account	$2/6$	$2/4$
password	$3/6$	$1/4$

$P(\text{spam})$	$P(\neg\text{spam})$
$6/10$	$4/10$

Message	Score if spam	Score if not spam	Prediction
earn + million	$\frac{2}{3}\cdot\frac{1}{3}\cdot\frac{2}{3}\cdot\frac{1}{2}\cdot\frac{6}{10}=\frac{2}{45}$	$\frac{1}{4}\cdot\frac{1}{2}\cdot\frac{1}{2}\cdot\frac{3}{4}\cdot\frac{4}{10}=\frac{3}{160}$	Spam
million + account	$\frac{1}{3}\cdot\frac{1}{3}\cdot\frac{1}{3}\cdot\frac{1}{2}\cdot\frac{6}{10}=\frac{1}{90}$	$\frac{3}{4}\cdot\frac{1}{2}\cdot\frac{1}{2}\cdot\frac{3}{4}\cdot\frac{4}{10}=\frac{9}{160}$	Not spam
account + password	$\frac{1}{3}\cdot\frac{2}{3}\cdot\frac{1}{3}\cdot\frac{1}{2}\cdot\frac{6}{10}=\frac{1}{45}$	$\frac{3}{4}\cdot\frac{1}{2}\cdot\frac{1}{2}\cdot\frac{1}{4}\cdot\frac{4}{10}=\frac{3}{160}$	Spam

Naive Bayes:
Extension to continuous features

Learning: estimating $\mathrm{P}(\mathrm{x_j}|y, \mathcal{O})$ from obs. $\mathcal{O} = \{(X_1, y_1), \ldots, (X_n, y_n)\}$:

1. Assume a distribution shape for each continuous $\mathrm{x_j}$ and value of y.

$$\text{e.g. } \mathrm{P}(\mathrm{x_j}|y = k, \mathcal{O}) \sim \mathcal{N}(\mu_{jk}, \sigma_{jk}^2)$$

2. Estimate parameters for each distribution from $\mathcal{O}$

$$\text{e.g. } \boxed{\hat{\mu}_{jk} = \frac{\sum_{i=1}^{n} \delta(y_i{=}k) \cdot x_{ij}}{\sum_{i=1}^{n} \delta(y_i{=}k)}}, \boxed{\hat{\sigma}_{jk}^2 = \frac{\sum_{i=1}^{n} \delta(y_i{=}k) \cdot (x_{ij} - \hat{\mu}_{jk})^2}{(\sum_{i=1}^{n} \delta(y_i{=}k)) - 1}}$$

Prediction:

$$\hat{\mathrm{y}}(X) = \underset{\mathrm{k}}{\operatorname{argmax}} \left(\left(\prod_{j=1}^{m} \mathrm{P}(\mathrm{x_j} = X_j | y = k, \mathcal{O}) \right) \times P(y = k|\mathcal{O}) \right)$$

Naive Bayes: Extension to continuous features

Length in words	Class y
30	Spam
40	Spam
100	Not spam
100	Spam
60	Not spam
30	Spam
200	Not spam
90	Spam
70	Spam
40	Not Spam

$\mathbf{P}(length\|\mathbf{spam})$	$\mathbf{P}(length\|\neg\mathbf{spam})$
$\mathcal{N}(\mu_s, \sigma_s^2)$ $\mu_s = 60$ $\sigma_s^2 = 960$	$\mathcal{N}(\mu_n, \sigma_n^2)$ $\mu_n = 100$ $\sigma_n^2 = 5067$

Naive Bayes: Extension to continuous features

Feature x_i	$P(x_i \mid spam)$	$P(x_i \mid \neg spam)$
earn	$4/6$	$1/4$
million	$2/6$	$2/4$
account	$2/6$	$2/4$
password	$3/6$	$1/4$
length	$\mathcal{N}(60,960)$	$\mathcal{N}(100,5067)$

$P(spam)$	$P(\neg spam)$
$6/10$	$4/10$

$$\mathcal{N}(\mu, \sigma^2) \sim \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(L-\mu)^2}{2\sigma^2}}$$

Message	Score if spam	Score if not spam	Prediction
earn + million $L = 80$	$\frac{2}{45} \times 0.004$	$\frac{1}{160} \times 0.0054$	Spam
million + account $L = 120$	$\frac{1}{90} \times 0.003$	$\frac{9}{160} \times 0.0054$	Not spam
account + password $L = 40$	$\frac{1}{45} \times 0.004$	$\frac{3}{160} \times 0.004$	Spam

Naive Bayes:
Strong Bayesian version and MAP estimator

Strong Bayesian version introduce priors for each distribution:

- Dirichlet prior for CPT of Y
- Dirichlet prior for discrete variable X_j
- Gaussian prior for continuous variable X_j with Gaussian distribution

MAP estimator represents prior knowledge with faked examples:

For target class Y :

$$\hat{P}_Y(c) = \frac{N\left(y^{(i)} = c\right) + \alpha_c^y - 1}{n + \sum_i \alpha_i^y - C}$$

Same for for discrete variable X_j:

$$p_{j,v,c} = \frac{N\left(x_j^{(i)} = v \; et \; y^{(i)} = c\right) + \alpha_v^{j,c} - 1}{N(y^{(i)} = c) + \sum_v \alpha_v^{j,c} - V}$$

Naive Bayes: A summary

- Most simple model
- Advantages:
  - Robust: low number of parameters $\rightarrow$ low risk of overfitting
  - Fast and simple to compute
  - Deal with mixture of discrete/continuous variables
  - Fully Bayesian (can integrate Dirichlet prior)
- Drawbacks:
  - Independence hypothesis too naive
  - Not accurate for complex classification problems
- Applications:
  - Document classification (e.g. spam detection)