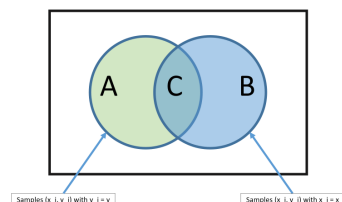


# Naive Bayes

Cornell CS 3/5780 · Spring 2026

## 2. Curse of dimensionality

- **MLE estimate:**  $\hat{P}(y|\mathbf{x}) = \frac{|C|}{|B|}$
- **Problem:** Requires many training data with *identical* features as  $\mathbf{x}$ , basically never happens in high dimensions or continuous space
  - zero denominator  $|B| \rightarrow 0$  (and  $|C| \rightarrow 0$ ), so estimate becomes unreliable
- **Idea:** Use Bayes Rule to flip the problem to "generative approach"  
 $P(y|\mathbf{x}) \propto P(\mathbf{x}|y)P(y)$ 
  - $P(y)$  is easy to estimate by counting classes (like coin tossing)
  - $P(\mathbf{x}|y)$  groups data by class, but is still high-dimensional



1

3

## 1. Estimating Distributions

- **Training data:**  $D = (\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)$  drawn i.i.d. from  $P(X, Y)$
- **MLE estimate of joint distribution:** counting occurrences

$$P(D) = \prod_{i=1}^n P(\mathbf{x}_i, y_i) \approx \hat{P}(\mathbf{x}, y) = \frac{\sum_{i=1}^n I(\mathbf{x}_i = \mathbf{x} \wedge y_i = y)}{n}$$

- **Conditional distribution:** For classification, estimate  $P(Y|X)$  directly
- Estimating each distribution by counting:

- $\hat{P}(y) = \frac{1}{n} \sum_{i=1}^n I(y_i = y)$
- $\hat{P}(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n I(\mathbf{x}_i = \mathbf{x})$
- $\hat{P}(\mathbf{x}, y) = \frac{1}{n} \sum_{i=1}^n I(\mathbf{x}_i = \mathbf{x})I(y_i = y)$

- **Conditional probability:**

$$\hat{P}(y|\mathbf{x}) = \frac{\hat{P}(y, \mathbf{x})}{\hat{P}(\mathbf{x})} = \frac{\sum_{i=1}^n I(\mathbf{x}_i = \mathbf{x})I(y_i = y)}{\sum_{i=1}^n I(\mathbf{x}_i = \mathbf{x})}$$

2

## 3. Naive Bayes

- **Naive Bayes Assumption:** Feature values are independent given the label

$$P(\mathbf{x}|y) = \prod_{\alpha=1}^d P(x_\alpha|y)$$

- Conditional independence assumption means we only need to estimate  $P(x_\alpha|y)$  for each dimension  $\alpha$  independently!

- **Naive Bayes Classifier:**

$$h(\mathbf{x}) = \operatorname{argmax}_y P(y|\mathbf{x}) \quad (1)$$

$$= \operatorname{argmax}_y \frac{P(\mathbf{x}|y)P(y)}{P(\mathbf{x})} \quad (2)$$

$$= \operatorname{argmax}_y P(\mathbf{x}|y)P(y) \quad (3)$$

$$= \operatorname{argmax}_y \prod_{\alpha=1}^d P(x_\alpha|y)P(y) \quad (4)$$

$$= \operatorname{argmax}_y \sum_{\alpha=1}^d \log(P(x_\alpha|y)) + \log(P(y)) \quad (5)$$

- **Question:** Explain each step of above derivation

4

## 4. Categorical Features

- **Features:**  $x_\alpha \in f_1, f_2, \dots, f_{K_\alpha}$  (example: demographic data)
- **Model:** Each feature follows a categorical distribution

$$P(x_\alpha = j | y = c) = [\theta_{jc}]_\alpha \quad \text{where} \quad \sum_{j=1}^{K_\alpha} [\theta_{jc}]_\alpha = 1$$

- **Generative story:** For each class, we roll  $d$  dice (one per feature)
- **MLE estimate:**

$$[\hat{\theta}_{jc}]_\alpha = \frac{\sum_{i=1}^n I(y_i = c) I(x_{i\alpha} = j)}{\sum_{i=1}^n I(y_i = c)}$$

- **MAP estimate:** add smoothing  $l_\alpha$  to numerator and  $\sum_{\alpha}^{K_\alpha} l_\alpha$  to denominator where e.g.  $l_\alpha = 1$  is Laplace prior
- **Prediction:** for  $\hat{\pi}_c = \sum_{i=1}^n I(y_i = c) / n$ , we predict

$$\operatorname{argmax}_c \hat{\pi}_c \prod_{\alpha=1}^d [\hat{\theta}_{jc}]_\alpha$$

## 6. Gaussian Naive Bayes

- **Features:**  $x_\alpha \in \mathbb{R}$  (continuous real values)
- **Model:** Each feature follows a Gaussian distribution

$$P(x_\alpha | y = c) = \mathcal{N}(\mu_{\alpha c}, \sigma_{\alpha c}^2) = \frac{1}{\sqrt{2\pi}\sigma_{\alpha c}} \exp\left(-\frac{1}{2} \left(\frac{x_\alpha - \mu_{\alpha c}}{\sigma_{\alpha c}}\right)^2\right)$$

- **Full distribution:**  $P(\mathbf{x} | y) \sim \mathcal{N}(\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y)$ , where  $\boldsymbol{\Sigma}_y$  is *diagonal* (independence assumption) with values  $\sigma_{\alpha, y}^2$

- **Mean estimation:** for  $n_c = \sum_{i=1}^n I(y_i = c)$

$$\hat{\mu}_{\alpha c} \leftarrow \frac{1}{n_c} \sum_{i=1}^n I(y_i = c) x_{i\alpha}$$

- **Variance estimation:**  $\hat{\sigma}_{\alpha c}^2 \leftarrow \frac{1}{n_c} \sum_{i=1}^n I(y_i = c) (x_{i\alpha} - \mu_{\alpha c})^2$

## 5. Multinomial Features

- **Features:** counts  $x_\alpha \in 0, 1, \dots, m$  where  $m = \sum_{\alpha=1}^d x_\alpha$  and higher count  $\implies$  stronger signal
- **Example:** Document classification,  $x_\alpha$  is count of word  $\alpha$  in document,  $m$  is total word count and  $d$  is vocabulary size
- **Model:** multinomial distribution

$$P(x_\alpha | y = c) \propto (\theta_{\alpha c})^{x_\alpha}, \quad \text{where} \quad \sum_{\alpha=1}^d \theta_{\alpha c} = 1$$

- **Parameter estimation:** MLE/MAP on multinomial distribution

$$\hat{\theta}_{\alpha c} = \frac{\sum_{i=1}^n I(y_i = c) x_{i\alpha} + l_\alpha}{\sum_{i=1}^n I(y_i = c) m_i + \sum_{\alpha=1}^d l_\alpha}$$

- **Prediction:** for  $\hat{\pi}_c = \sum_{i=1}^n I(y_i = c) / n$ , we predict

$$\operatorname{argmax}_c \hat{\pi}_c \prod_{\alpha=1}^d \hat{\theta}_{\alpha c}^{x_\alpha}$$

5

## 7. Naive Bayes is a Linear Classifier

- For many common cases, Naive Bayes produces a linear decision boundary!

- **Multinomial features** with  $y \in \{-1, +1\}$ , we can derive

$$P(y | \mathbf{x}) = \frac{1}{1 - \exp(y(\mathbf{w}^\top \mathbf{x} + b))}$$

where weight  $\mathbf{w}$  and bias  $b$  are defined in terms of  $\theta$  and  $\pi$

- weights:  $w_\alpha = \log[\theta_{\alpha+}] - \log[\theta_{\alpha-}]$
- bias:  $b = \log[P(Y = +1)] - \log[P(Y = -1)]$

- **Gaussian** with constant variance and  $y \in \{-1, +1\}$

- Similar derivation and expression but with  $w_\alpha = \mu_{\alpha,+} - \mu_{\alpha,-}$  (difference of means), as long as  $\sigma_{\alpha,+} = \sigma_{\alpha,-}$  for all  $\alpha$

7

6

8

## 8. Linear Classifier Proof (Multinomial)

- **Question:** Explain the following steps, where we let  $w_\alpha^+ = \log[\theta_{\alpha+}]$

$$\begin{aligned} \log [P(\mathbf{x}|Y = +1)] &= \log [\prod_{\alpha=1}^d P(x_\alpha|Y = +1)] \\ &= \sum_{\alpha=1}^d x_\alpha \log[\theta_{\alpha+}] \\ &= \sum_{\alpha=1}^d x_\alpha w_\alpha^+ = \mathbf{x}^\top \mathbf{w}_+. \end{aligned}$$

- A similar argument shows  $\log [P(\mathbf{x}|Y = -1)] = \mathbf{x}^\top \mathbf{w}_-$

- **Question:** Explain the following steps

$$\begin{aligned} P(Y = +1 | \mathbf{x}) &= \frac{P(\mathbf{x} | Y = +1)P(Y = +1)}{P(\mathbf{x})} \\ &= \frac{P(\mathbf{x} | Y = +1)P(Y = +1)}{P(\mathbf{x} | Y = +1)P(Y = +1) + P(\mathbf{x} | Y = -1)P(Y = -1)} \end{aligned}$$

## 9. Linear Classifier Proof Cont. (Multinomial)

- **Question:** Explain the following steps, recalling  $b = \log[P(Y = +1)] - \log[P(Y = -1)]$

$$(6) \quad P(Y = +1 | \mathbf{x}) = \frac{P(\mathbf{x} | Y = +1)P(Y = +1)}{P(\mathbf{x} | Y = +1)P(Y = +1) + P(\mathbf{x} | Y = -1)P(Y = -1)} \quad (11)$$

$$(7) \quad = \frac{e^{\mathbf{x}^\top \mathbf{w}_+} P(Y = +1)}{e^{\mathbf{x}^\top \mathbf{w}_+} P(Y = +1) + e^{\mathbf{x}^\top \mathbf{w}_-} P(Y = -1)} \quad (12)$$

$$(8) \quad = \frac{1}{1 + e^{-\mathbf{x}^\top \mathbf{w}} \frac{P(Y=-1)}{P(Y=+1)}} \quad (13)$$

$$(9) \quad = \frac{1}{1 + e^{-(\mathbf{x}^\top \mathbf{w} + b)}} \quad (14)$$

- A similar argument shows  $P(Y = -1 | \mathbf{x}) = \frac{1}{1 + e^{(\mathbf{x}^\top \mathbf{w} + b)}}$ , and combining the two expressions completes the proof.

(10)