# CS340: Machine Learning

# Naive Bayes classifiers

Kevin Murphy

# Classifiers

- A **classifier** is a function $f$ that maps input feature vectors $x \in \mathcal{X}$ to output class labels $y \in \{1, \ldots, C\}$
- $\mathcal{X}$ is the **feature space** eg $\mathcal{X} = \mathbb{R}^p$ or $\mathcal{X} = \{0,1\}^p$ (can mix discrete and continuous features)
- We assume the class labels are unordered (categorical) and mutually exclusive. (If we allow an input to belong to multiple classes, this is called a **multi-label** problem.)
- Goal: to learn $f$ from a **labeled training set** of $N$ input-output pairs, $(x_i, y_i)$, $i = 1 : N$.
- We will focus our attention on probabilistic classifiers, i.e., methods that return $p(y|x)$.
- Alternative is to learn a **discriminant function** $f(x) = \hat{y}(x)$ to predict the most probable label.

# Generative vs discriminative classifiers

- Discriminative: directly learn the function that computes the class posterior $p(y|x)$. It discriminates between different classes *given* the input.
- Generative: learn the **class-conditional density** $p(x|y)$ for each value of $y$, and learn the **class priors** $p(y)$; then one can apply Bayes rule to compute the posterior

$$p(y|x) = \frac{p(x,y)}{p(x)} = \frac{p(x|y)p(y)}{p(x)}$$

where $p(x) = \sum_{y'=1}^{C} p(y'|x)$.

# Generative classifiers

We usually use a plug-in approximation for simplicity

$$p(y = c|x, D) \approx p(y = c|x, \hat{\theta}, \hat{\pi}) = \frac{p(x|y = c, \hat{\theta}_C)p(y = c|\hat{\pi})}{\sum_{C'} p(x|y = c', \hat{\theta}_{C'})p(y = c'|\hat{\pi})}$$

where $D$ is the training data, $\pi$ are the parameters of the class prior $p(y)$ and $\theta$ are the paramters of the class-conditional densities $p(x|y)$.

# Class prior

- Class prior

$$p(y = c|\pi) = \pi_c$$

- MLE

$$\hat{\pi}_c = \frac{\sum_{i=1}^{N} I(y_i = c)}{N} = \frac{N_c}{N}$$

where $N_c$ is the number of training examples that have class label $c$.

- Posterior mean (using Dirichlet prior)

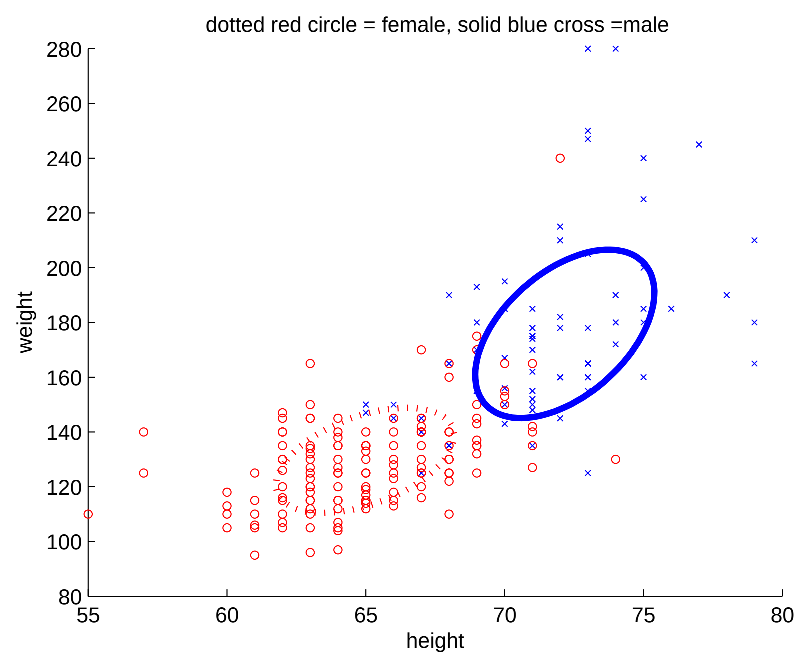
$$\hat{\pi}_c = \frac{N_c + 1}{N + C}$$

# Gaussian class-conditional densities

Suppose $x \in \mathbb{R}^2$ representing the height and weight of adult Westerners (in inches and pounds respectively), and $y \in \{1, 2\}$ represents male or female. A natural choice for the class-conditional density is a two-dimensional Gaussian

$$p(x|y = c, \theta_c) = \mathcal{N}(x|\mu_c, \Sigma_c)$$

where the mean and covariance matrix depend on the class $c$

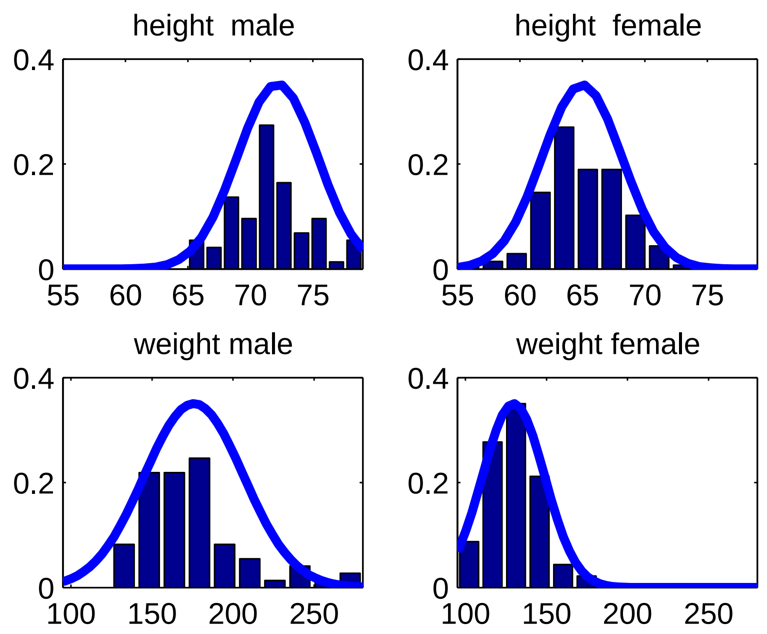


## Naive Bayes assumption

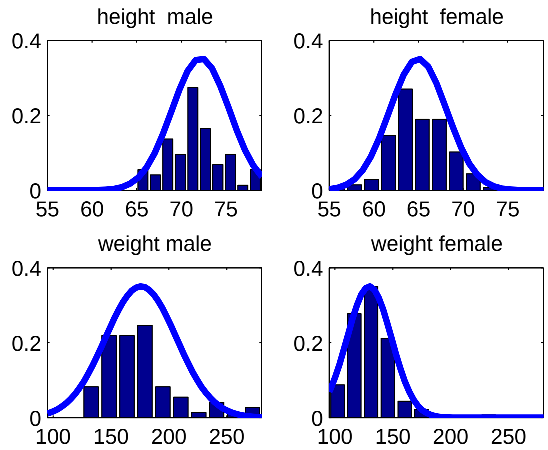
Assume features are conditionally independent given class.

$$p(x|y=c,\theta_c) = \prod_{d=1}^{p} p(x_d|y=c,\theta_c) = \prod_{d=1}^{p} \mathcal{N}(x_d|\mu_{cd},\sigma_{cd})$$

This is equivalent to assuming that $\Sigma_c$ is diagonal.



# Training




| h | w | y |
|---|---|---|
| 67 | 125 | m |
| 79 | 210 | m |
| 71 | 150 | m |
| 68 | 140 | f |
| 67 | 142 | f |
| 60 | 120 | f |

| | h | w |
|---|---|---|
| m | $\mu = 71.66,\ \sigma = 3.13$ | $\mu = 175.62,\ \sigma = 32.40$ |
| f | $\mu = 65.07,\ \sigma = 3.19$ | $\mu = 129.69,\ \sigma = 18.67$ |

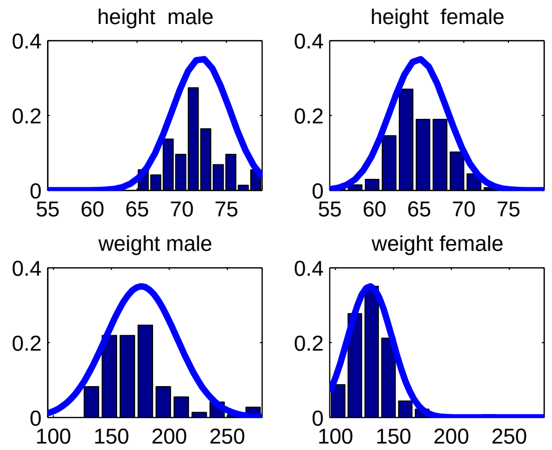
# Testing

$$p(y=m|x) = \frac{p(x|y=m)p(y=m)}{p(x|y=m)p(y=m)+p(x|y=f)p(y=f)}$$

Let us assume $p(y=m)=p(y=f)=0.5$

$$\begin{aligned} p(y=m|x) &= \frac{p(x|y=m)}{p(x|y=m)+p(x|y=f)} \\ &= \frac{p(x_h|y=m)p(x_w|y=m)}{p(x_h|y=m)p(x_w|y=m)+p(x_h|y=f)p(x_w|y=f)} \\ &= \frac{\mathcal{N}(x_h;\mu_{mh},\sigma_{mh})\times\mathcal{N}(x_h;\mu_{mw},\sigma_{mw})}{``+\mathcal{N}(x_h;\mu_{fh},\sigma_{fh})\times\mathcal{N}(x_h;\mu_{fw},\sigma_{fw})} \end{aligned}$$

# Testing

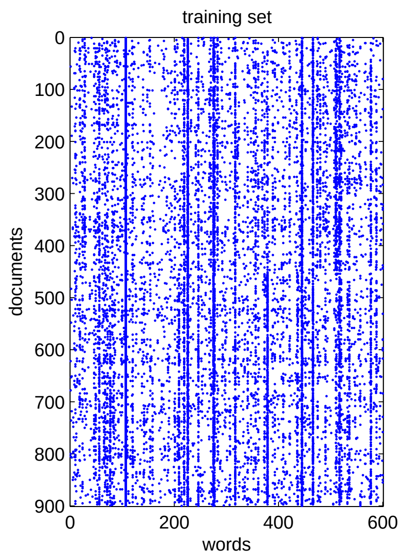



| | h | w |
|---|---|---|
| m | $\mu = 71.66,\ \sigma = 3.13$ | $\mu = 175.62,\ \sigma = 32.40$ |
| f | $\mu = 65.07,\ \sigma = 3.19$ | $\mu = 129.69,\ \sigma = 18.67$ |

| h | w | $p(y = m \vert x)$ |
|---|---|---|
| 72 | 180 | |
| 60 | 100 | |
| 68 | 155 | |

# Binary data

- Suppose we want to classify email into spam vs non-spam.
- A simple way to represent a text document (such as email) is as a **bag of words**.
- Let $x_d = 1$ if word $d$ occurs in the document and $x_d = 0$ otherwise.



# Parameter estimation

Class-conditional denstity becomes a product of Bernoullis

$$p(\vec{x}|Y = c, \theta) = \prod_{d=1}^{p} \theta_{cd}^{x_d}(1 - \theta_{cd})^{1-x_d}$$

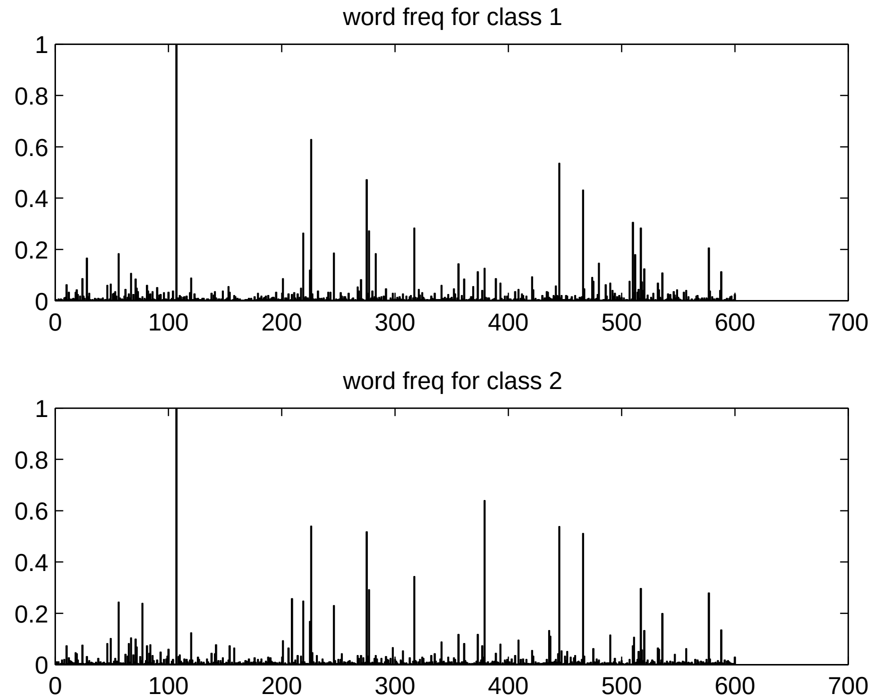
MLE

$$\hat{\theta}_{cd} = \frac{N_{cd}}{N_c}$$

Posterior mean (with Beta(1,1) prior)

$$\hat{\theta}_{cd} = \frac{N_{cd} + 1}{N_c + 2}$$

# Fitted class conditional densities $p(x = 1|y = c)$



## Numerical issues

When computing

$$P(Y = c|\vec{x}) = \frac{P(\vec{x}|Y = c)P(Y = c)}{\sum_{c'=1}^{C} P(\vec{x}|Y = c')P(Y = c')}$$

you will oftne encounter **numerical underflow** since $p(\vec{x}, y = c)$ is very small.
Take logs

$$\begin{aligned} b_c &\stackrel{\text{def}}{=} \log[P(\vec{x}|Y = c)P(Y = c)] \\ \log P(Y = c|\vec{x}) &= b_c - \log \sum_{c'=1}^{C} e^{b_{c'}} \end{aligned}$$

but $e^{b_c}$ will underflow!

## Log-sum-exp trick

$$\log(e^{-120} + e^{-121}) = \log\left(e^{-120}(e^0 + e^{-1})\right) = \log(e^0 + e^{-1}) - 120$$

In general

$$\begin{aligned}
\log \sum_c e^{b_c} &= \log\left[(\sum_c e^{b_c})e^{-B}e^B\right] \\
&= \log\left[(\sum_c e^{b_c - B})e^B\right] \\
&= \left[\log(\sum_c e^{b_c - B})\right] + B
\end{aligned}$$

where $B = \max_c b_c$.

In matlab, use `logsumexp.m`.

# Softmax function

$$
\begin{aligned}
p(y=c|\vec{x},\theta,\pi) &= \frac{p(x|y=c)p(y=c)}{\sum_{c'} p(x|y=c')p(y=c')} \\
&= \frac{\exp[\log p(x|y=c)+\log p(y=c)]}{\sum_{c'} \exp[\log p(x|y=c')+\log p(y=c')]} \\
&= \frac{\exp\left[\log \pi_c + \sum_d x_d \log \theta_{cd}\right]}{\sum_{c'} \exp\left[\log \pi_{c'} + \sum_d x_d \log \theta_{c'd}\right]}
\end{aligned}
$$

Now define vectors

$$
\begin{aligned}
\vec{x} &= [1, x_1, \ldots, x_{1p}] \\
\beta_c &= [\log \pi_c, \log \theta_{c1}, \ldots, \log \theta_{cp}]
\end{aligned}
$$

Hence

$$
p(y=c|\vec{x},\beta) = \frac{\exp[\beta_c^T \vec{x}]}{\sum_{c'} \exp[\beta_{c'}^T \vec{x}]}
$$

# Logistic function

If $y$ is binary

$$\begin{aligned} p(y=1|x) &= \frac{e^{\beta_1^T x}}{e^{\beta_1^T x} + e^{\beta_2^T x}} \\ &= \frac{1}{1 + e^{(\beta_2 - \beta_1)^T x}} \\ &= \frac{1}{1 + e^{w^T x}} \\ &= \sigma(w^T x) \end{aligned}$$

where we have defined $w = \beta_2 - \beta_1$ and $\sigma(\cdot)$ is the **logistic** or **sigmoid** function

$$\sigma(u) = \frac{1}{1 + e^{-u}}$$