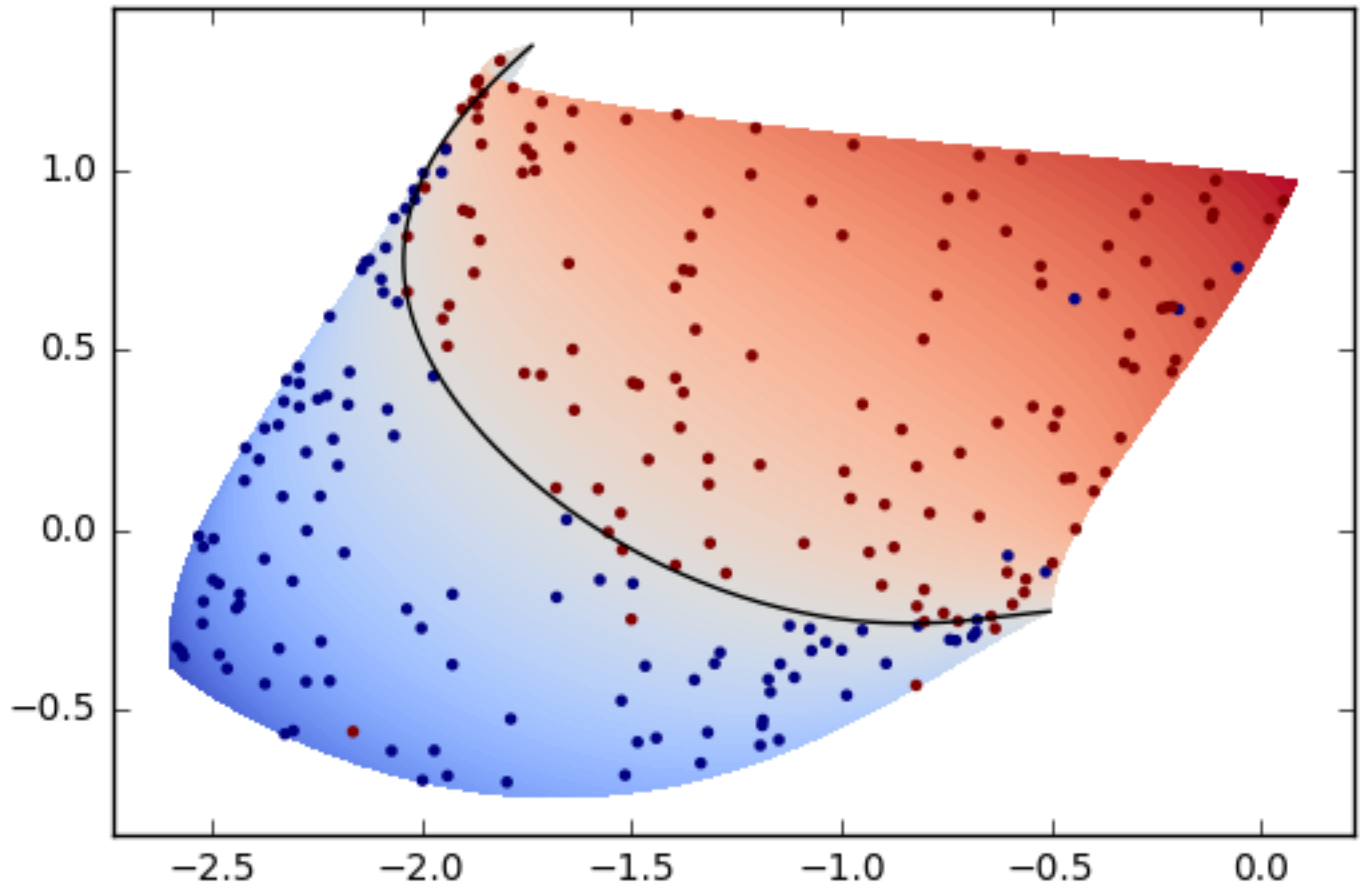
# CS 273A: Machine Learning

## Fall 2021

# Lecture 3: Bayes Classifiers

Roy Fox
Department of Computer Science
Bren School of Information and Computer Sciences
University of California, Irvine




All slides in this course adapted from Alex Ihler & Sameer Singh

# Logistics

assignment 1

- Assignment 1 is due Tuesday

recordings

- Lectures will be recorded, starting today
- Recordings from Fall'21 also available

# Today's lecture

$k$-Nearest Neighbors

Bayes classifiers

Naïve Bayes Classifiers

Bayes error

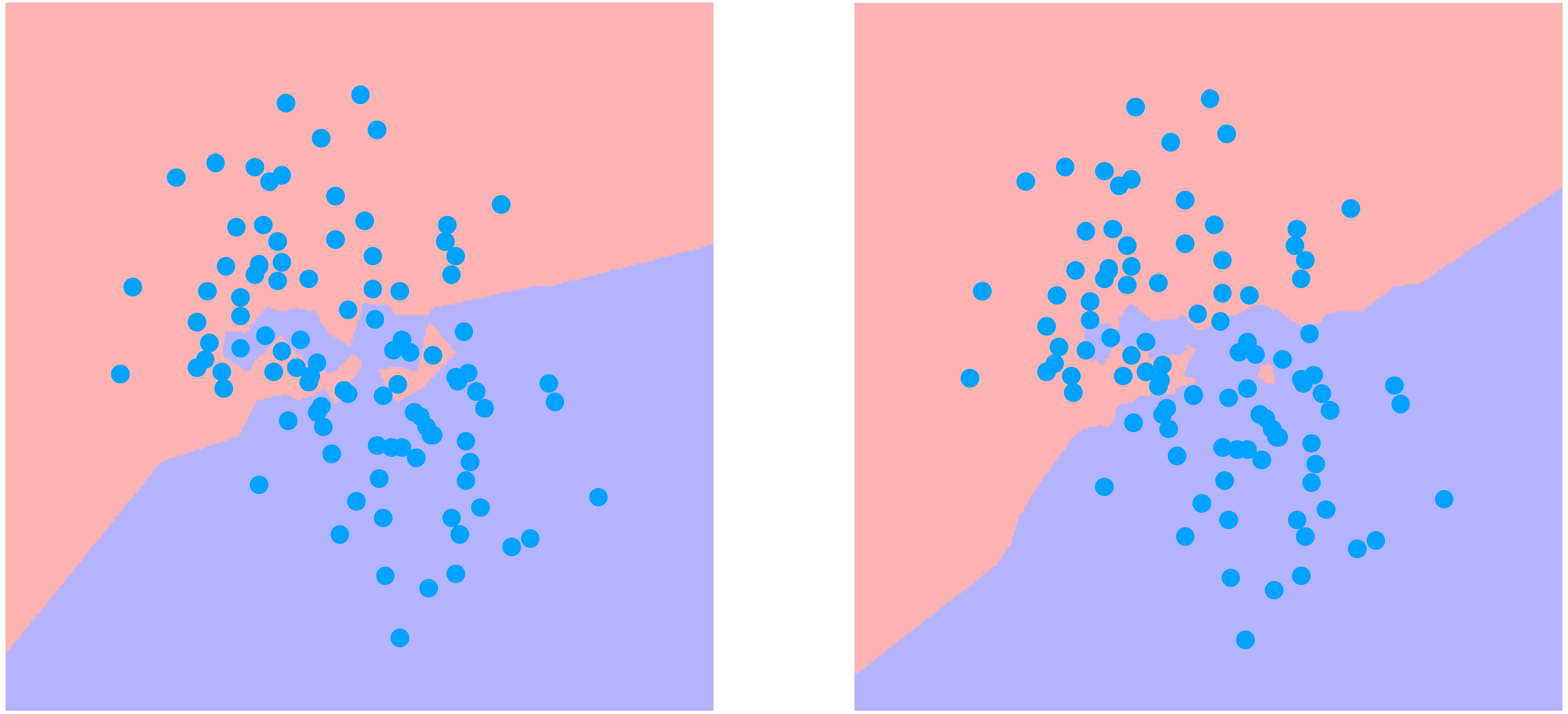
# $k$-Nearest Neighbor (kNN)

- Find the $k$ nearest neighbors to $x$ in the dataset
  - Given $x$, rank the data points by their distance from $x$, $d(x, x^{(j)})$
    - Usually, Euclidean distance $d(x, x^{(j)}) = \sqrt{\frac{1}{n}\sum_i (x_i - x_i^{(j)})^2}$
  - Select the $k$ data points which are have smallest distance to $x$
- What is the prediction?
  - Regression: average $y^{(j)}$ for the $k$ closest training examples
  - Classification: take a majority vote among $y^{(j)}$ for the $k$ closest training examples
    - No ties in 2-class problems when $k$ is odd

# kNN decision boundary

- For classification, the decision boundary is piecewise linear
- Increasing $k$ "simplifies" the decision boundary
  - Majority voting means less emphasis on individual points

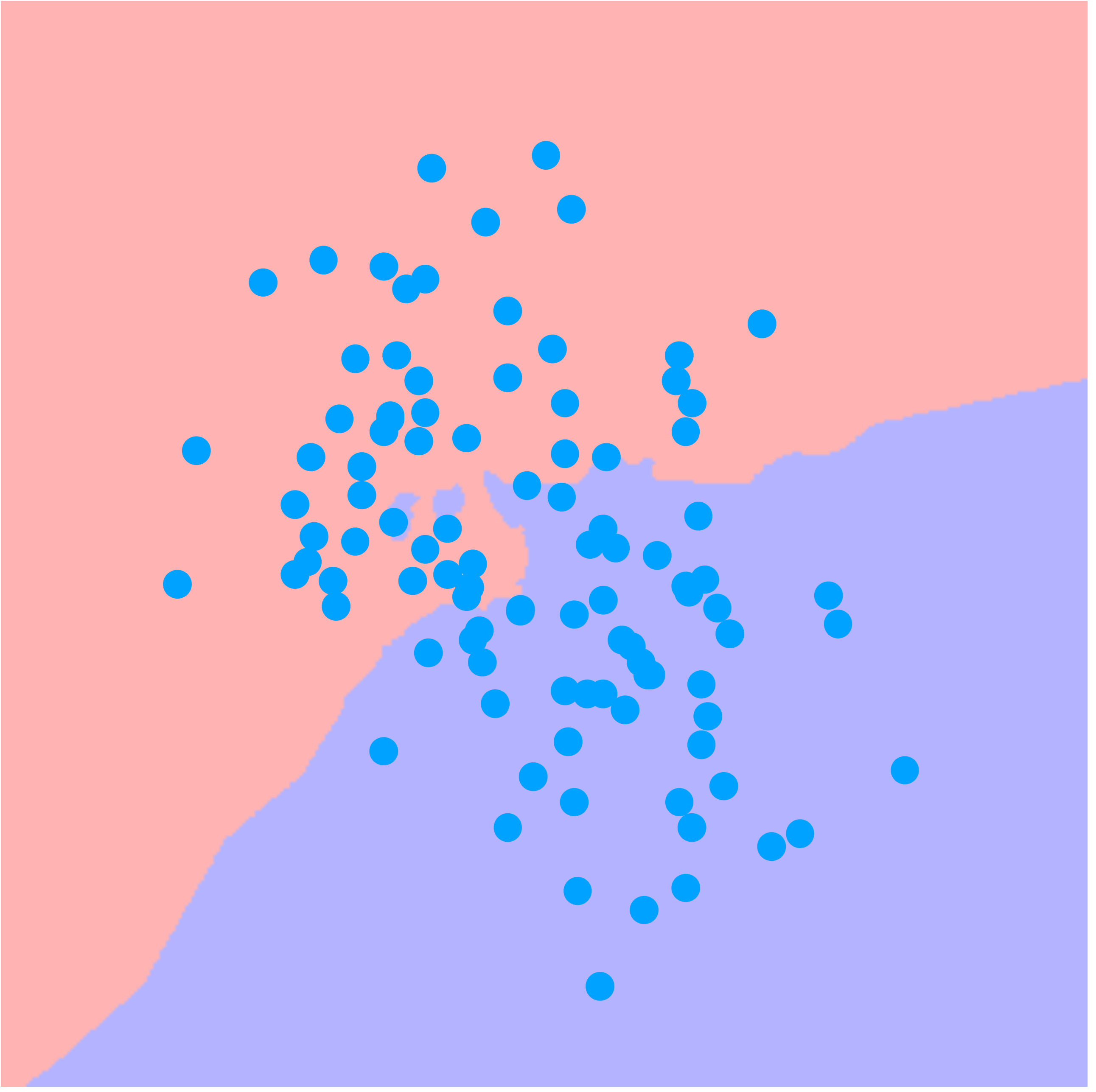
$k = 1$ $\qquad$ $k = 3$



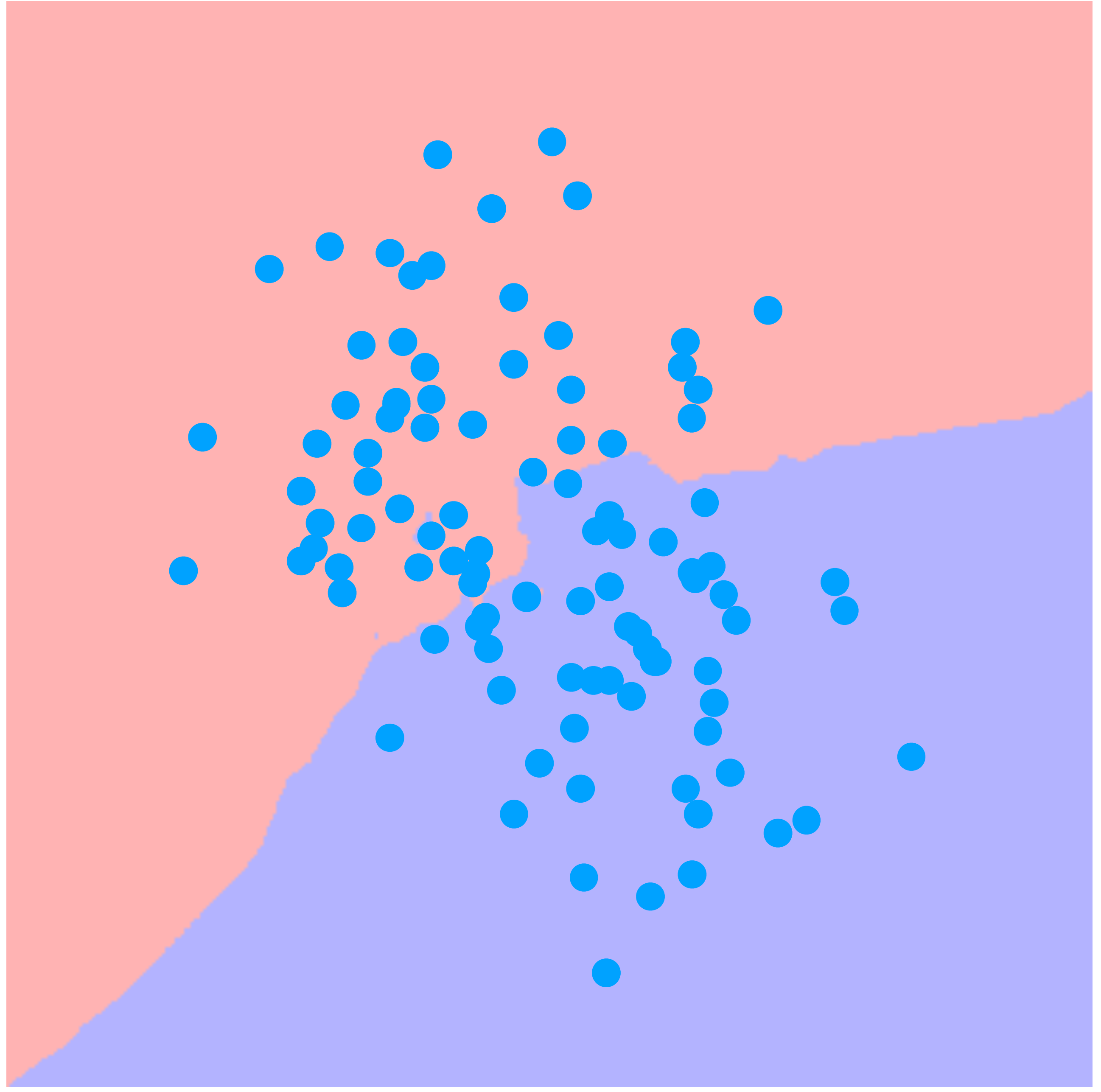
# kNN decision boundary

- For classification, the decision boundary is piecewise linear
- Increasing $k$ "simplifies" the decision boundary
  - Majority voting means less emphasis on individual points

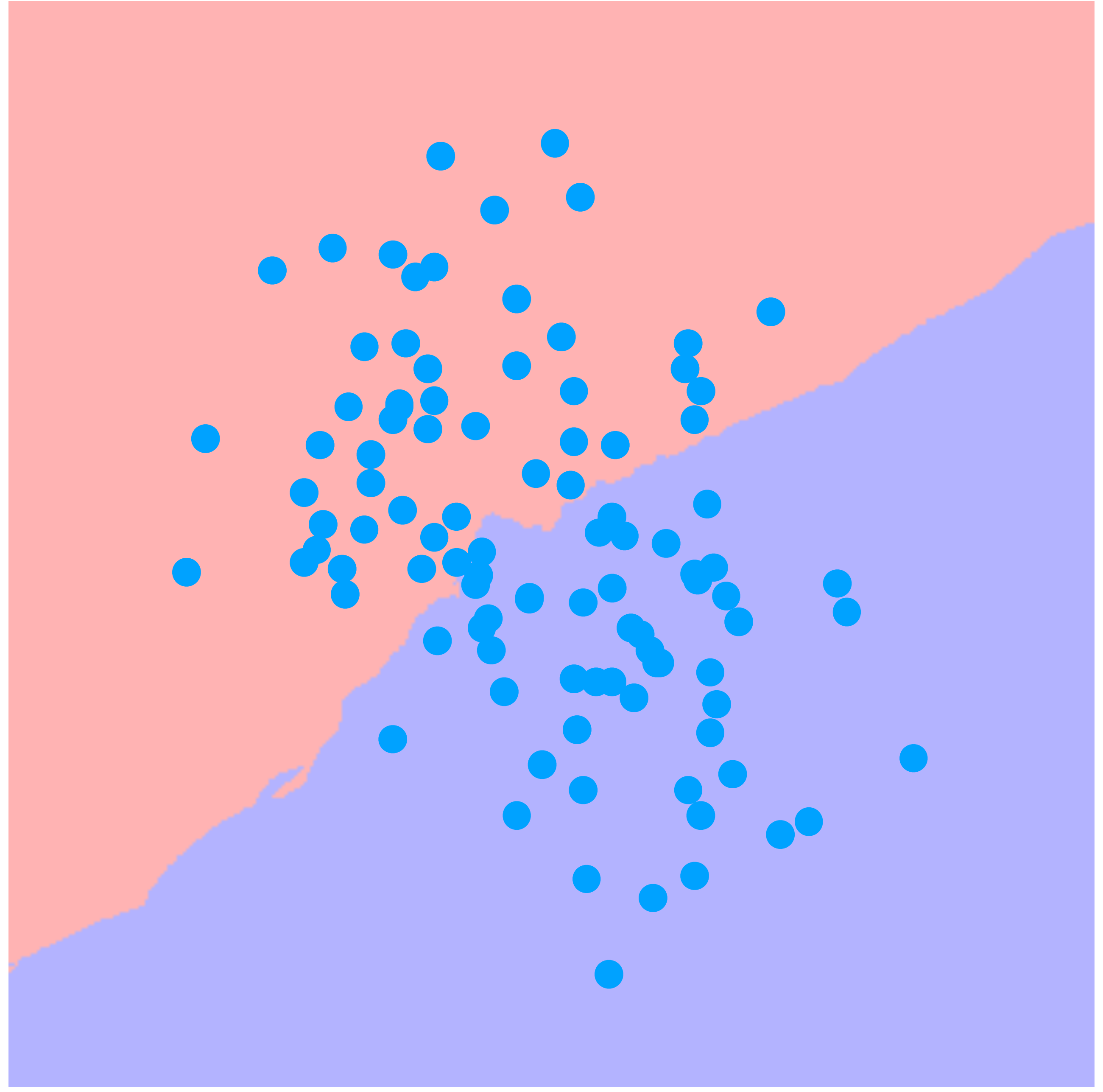
$k = 5$



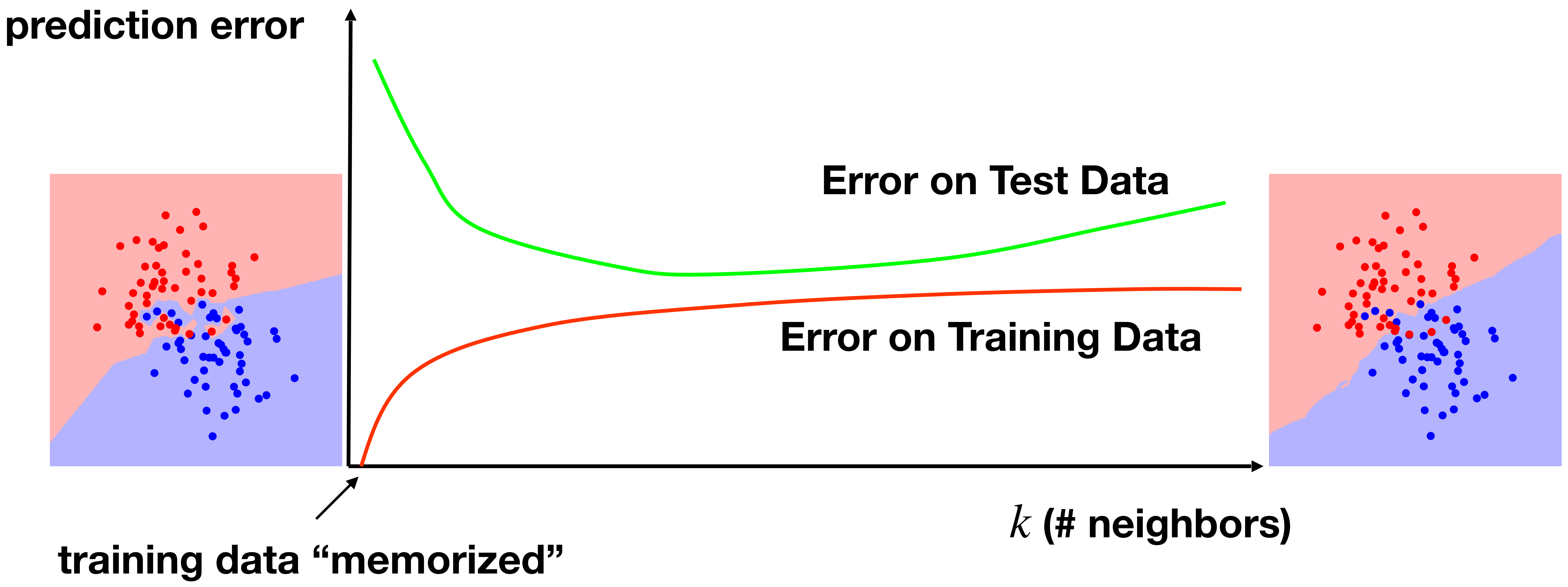
$k = 7$



$k = 25$



# Error rates and $k$




- A complex model fits training data but generalizes poorly
- $k = 1$: perfect memorization of examples = complex
- $k = m$: predict majority class over entire dataset = simple
- We can select $k$ with validation

# kNN classifier: further considerations

- Decision boundary smoothness
  - Increases with $k$, as we average over more neighbors
  - Decreases with training size $m$, as more points support the boundary
  - Generally, optimal $k$ should increase with $m$
- Extensions of $k$-Nearest Neighbors
  - Do features have the same scale? importance?
    - Weighted distance: $d(x, x') = \sqrt{\sum_i w_i (x_i - x'_i)^2}$
    - Non-Euclidean distances may be more appropriate for type of data
  - Fast search techniques (indexing) to find $k$ closest points in high-dimensional space
  - Weighted average / voting based on distance: $\hat{y} = \sum_j w(d(x, x^{(j)})) y^{(j)}$

# Recap: $k$-Nearest Neighbors

- Piecewise linear decision boundary
  - Just for analysis — the algorithm doesn't compute the boundary
- With $k > 1$:
  - Regression → (weighted) average
  - Classification → (weighted) vote
- Overfitting and complexity:
  - Model "complexity" goes down as $k$ grows
  - Use validation data to estimate test error rates and select $k$

# Today's lecture

$k$-Nearest Neighbors

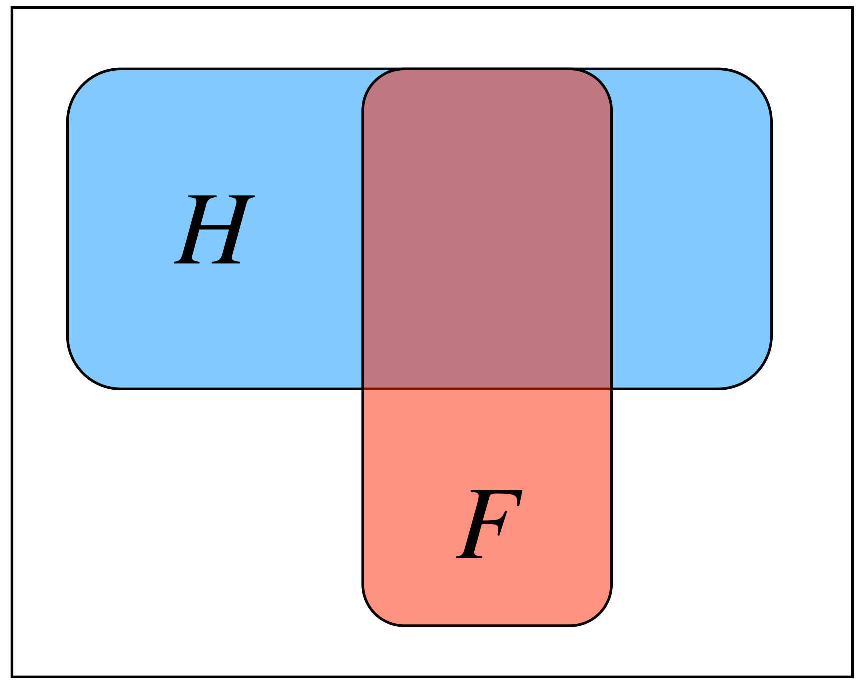
**Bayes classifiers**

Naïve Bayes Classifiers

Bayes error

# Conditional probabilities

- Two events: headache ($H$), flu ($F$)
- $p(H) = \frac{1}{10}$
- $p(F) = \frac{1}{40}$
- $p(H \mid F) = \frac{1}{2}$
- You wake up with a headache
  - What are the chances that you have the flu?




$$p(F, H) = p(F)p(H \mid F)$$
$$= \frac{1}{40} \cdot \frac{1}{2} = \frac{1}{80}$$

$$p(F \mid H) = \frac{p(F, H)}{p(H)}$$
$$= \frac{1}{80} \cdot \frac{10}{1} = \frac{1}{8}$$

Example from Andrew Moore's slides

# Probabilistic modeling of data

- Assume data with features $x$ and discrete labels $y$
- Prior probability of each class: $p(y)$
  - Prior = before seeing the features
  - E.g., fraction of applicants that have good credit
- Distribution of features given the class: $p(x \mid y = c)$
  - How likely are we to see $x$ in applicants with good credit?
- Joint distribution: $p(x, y) = p(x)p(y \mid x) = p(y)p(x \mid y)$
- Bayes' rule: posterior $p(y \mid x) = \dfrac{p(y)p(x \mid y)}{p(x)} = \dfrac{p(y)p(x \mid y)}{\sum_c p(y = c)p(x \mid y = c)}$

**models:**

$x \longrightarrow y$

$y \longrightarrow x$

**does not imply causality!**

# Bayes classifiers

- Learn a "class-conditional" model for the data
  - Estimate the probability for each class $p(y = c)$
  - Split training data by class $\mathcal{D}_c = \{x^{(j)} : y^{(j)} = c\}$
  - Estimate from $\mathcal{D}_c$ the conditional distribution $p(x \mid y = c)$
- For discrete $x$, can represent as a contingency table

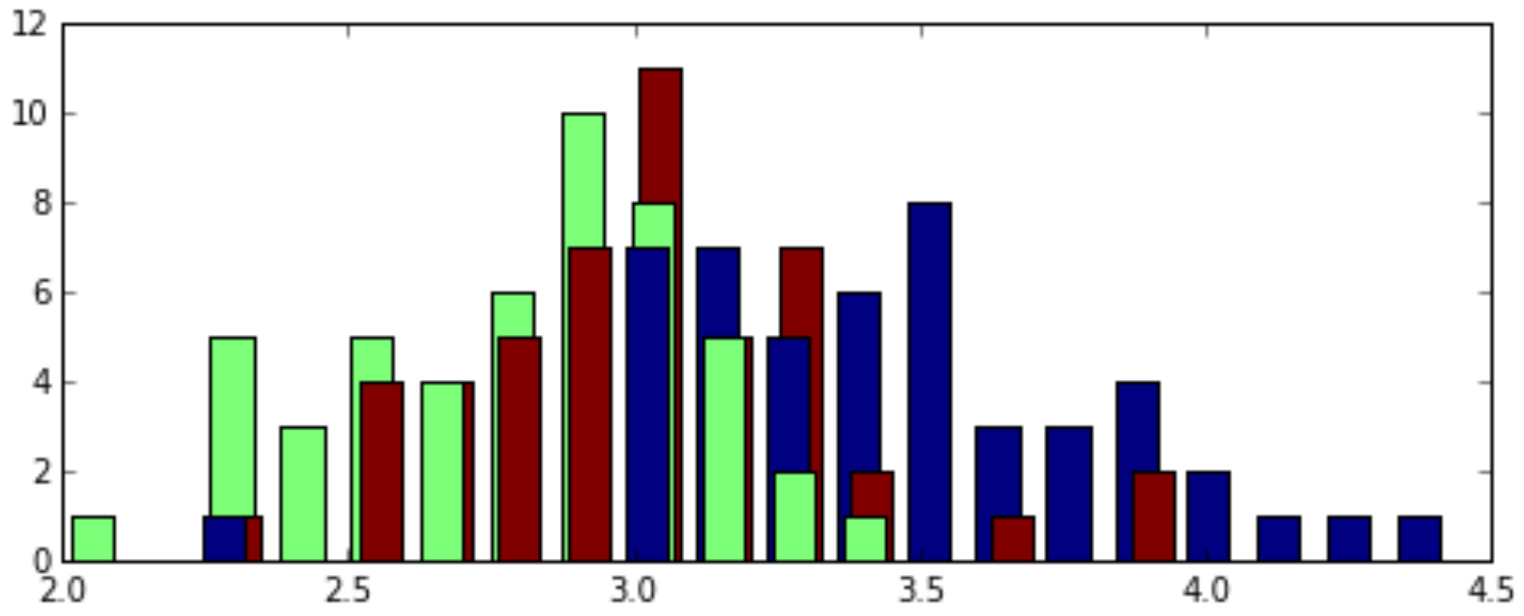
| Features | # bad | # good |
|---|---|---|
| X=0 | 42 | 15 |
| X=1 | 338 | 287 |
| X=2 | 3 | 5 |
| p(y) | 383/690 | 307/690 |

| p(x\|y=0) | p(x\|y=1) |
|---|---|
| 42/383 | 15/307 |
| 338/383 | 287/307 |
| 3/383 | 5/307 |

| p(y=0\|x) | p(y=1\|x) |
|---|---|
| .7368 | .2632 |
| .5408 | .4592 |
| .3750 | .6250 |

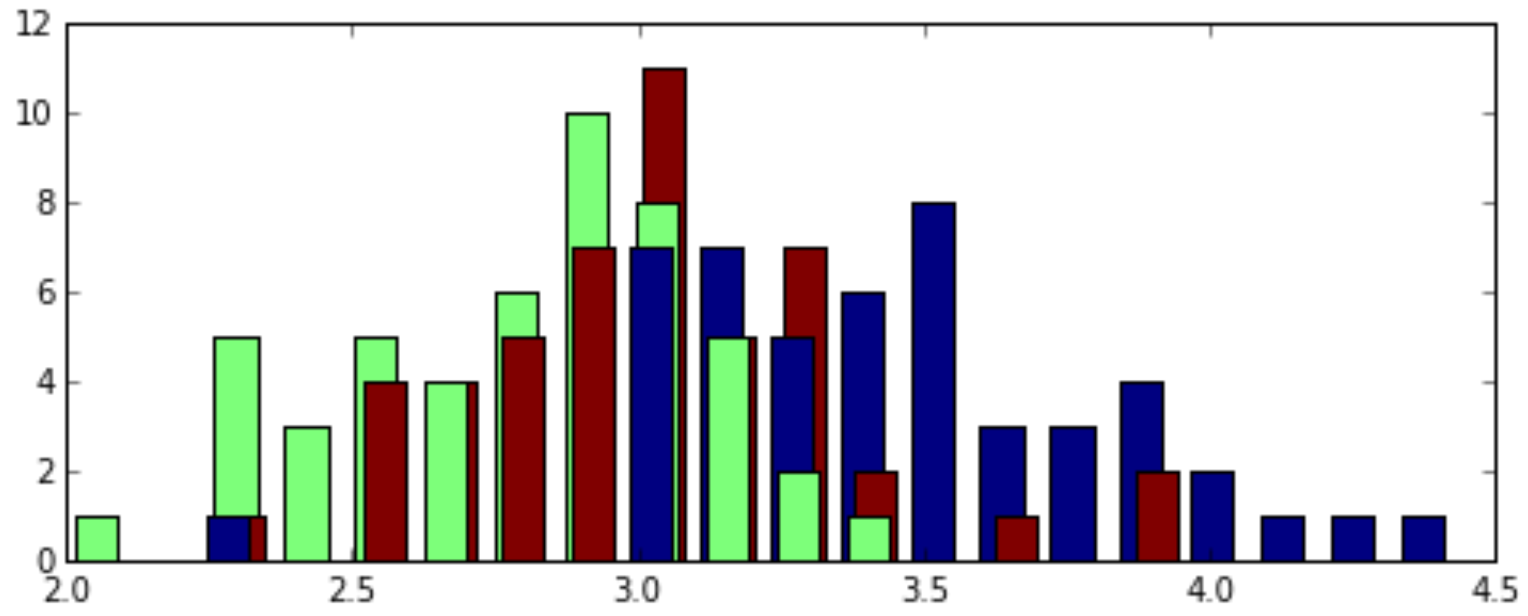
# Bayes classifiers

- Learn a "class-conditional" model for the data
  - Estimate the probability for each class $p(y = c)$
  - Split training data by class $\mathcal{D}_c = \{x^{(j)} : y^{(j)} = c\}$
  - Estimate from $\mathcal{D}_c$ the conditional distribution $p(x \mid y = c)$
- For continuous $x$, we need some other density model
  - Histogram
  - Gaussian
  - others...



# Histograms

- Split training data by class $\mathcal{D}_c = \{x^{(j)} : y^{(j)} = c\}$
- For each class, split $x$ into $k$ bins and count data points in each bin
- Normalize the $k$-dimensional count vector to get $p(x \mid y = c)$
- To use: given $x$, find its bin, output probability for that bin



# Gaussian models

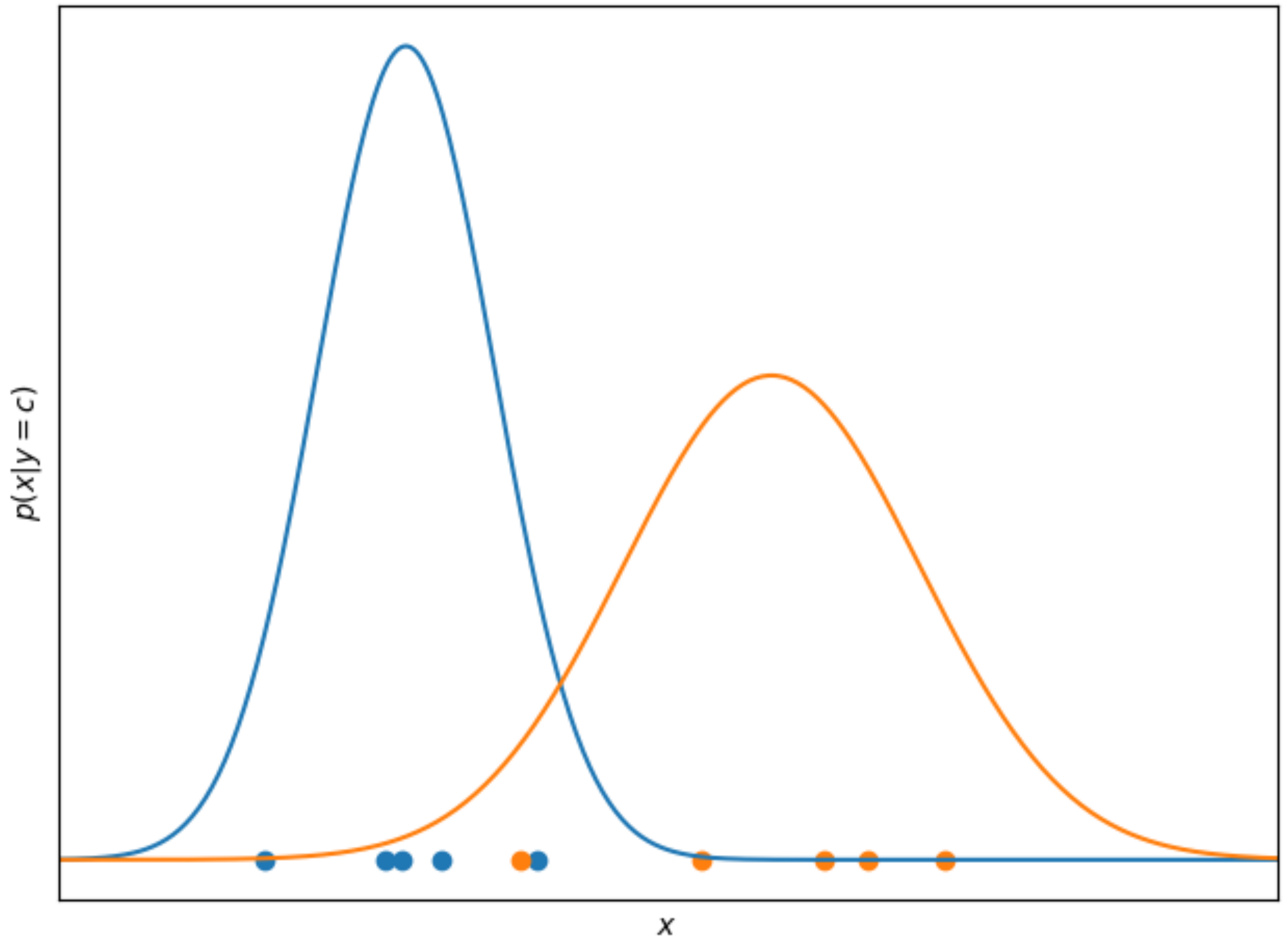
- Model instances in each class with a Gaussian $p(x \mid y = c) \sim \mathcal{N}(\mu_c, \sigma_c^2)$

- Estimate parameters of each Gaussians from the data $\mathcal{D}_c$

  - $\hat{p}(y = c) = \dfrac{m_c}{m}$ where $m_c = |\mathcal{D}_c|$

  - $\hat{\mu}_c = \dfrac{1}{m_c} \sum_{j:\ y^{(j)}=c} x^{(j)}$

  - $\hat{\sigma}_c^2 = \dfrac{1}{m_c} \sum_{j:\ y^{(j)}=c} (x^{(j)} - \hat{\mu}_c)^2$



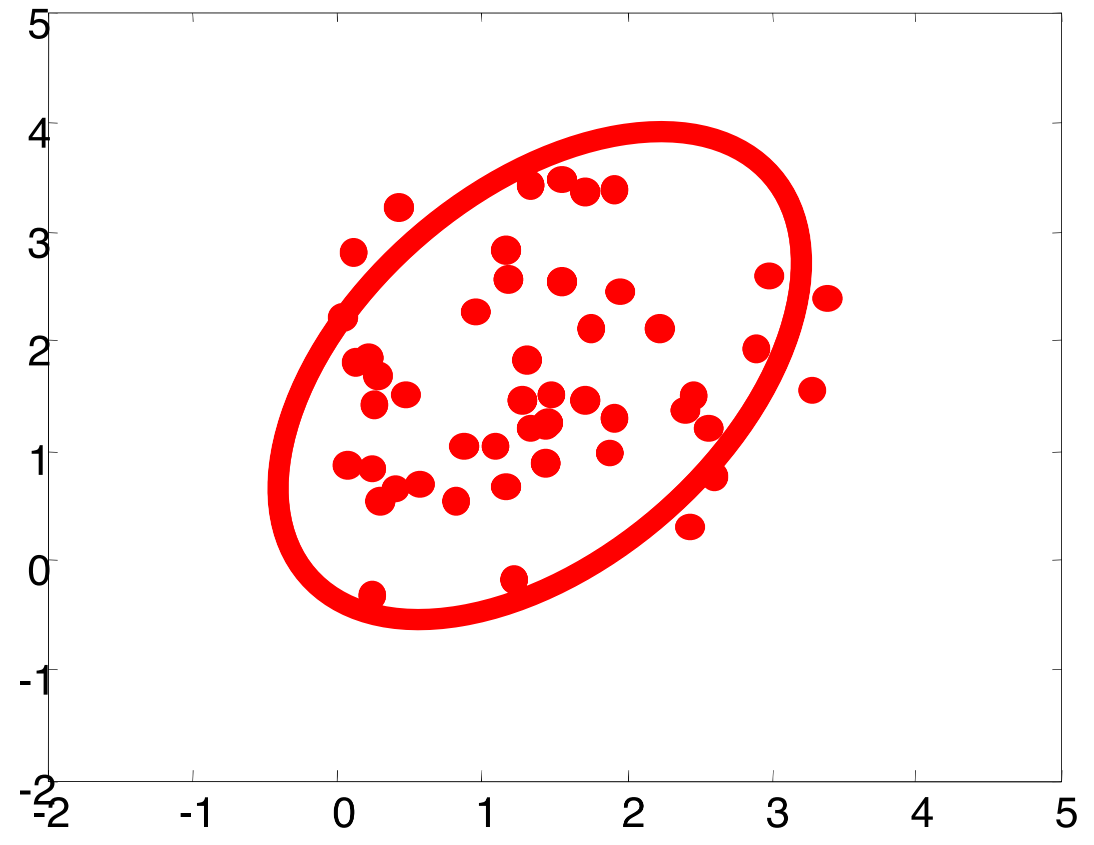
# Multivariate Gaussian models

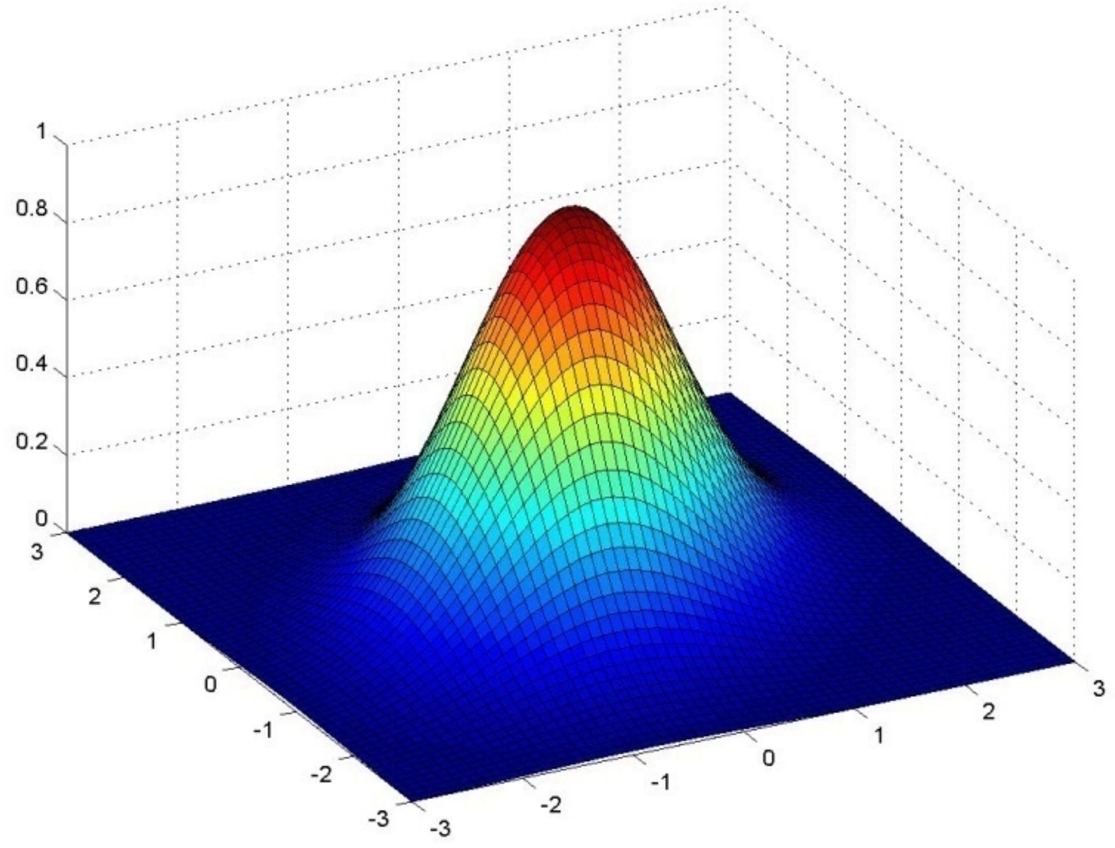
- Multivariate Gaussian: $\mathcal{N}(x; \mu, \Sigma) = (2\pi)^{-\frac{d}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(x-\mu)^\intercal \Sigma^{-1} (x-\mu)\right)$

- Estimation similar to univariate case:
  - $\hat{\mu}_c = \frac{1}{m_c} \sum_j x^{(j)}$
  - $\hat{\Sigma}_c = \frac{1}{m_c} \sum_j (x^{(j)} - \hat{\mu}_c)(x^{(j)} - \hat{\mu}_c)^\intercal$ (outer product)

**$\mu$ = mean ($d$-dimensional vector)**
**$\Sigma$ = covariance ($d \times d$ matrix)**
**$\Sigma^{-1}$ = precision ($d \times d$ matrix)**
**$|\cdot|$ = determinant (scalar)**

- How many parameters?
  - $d + d^2$

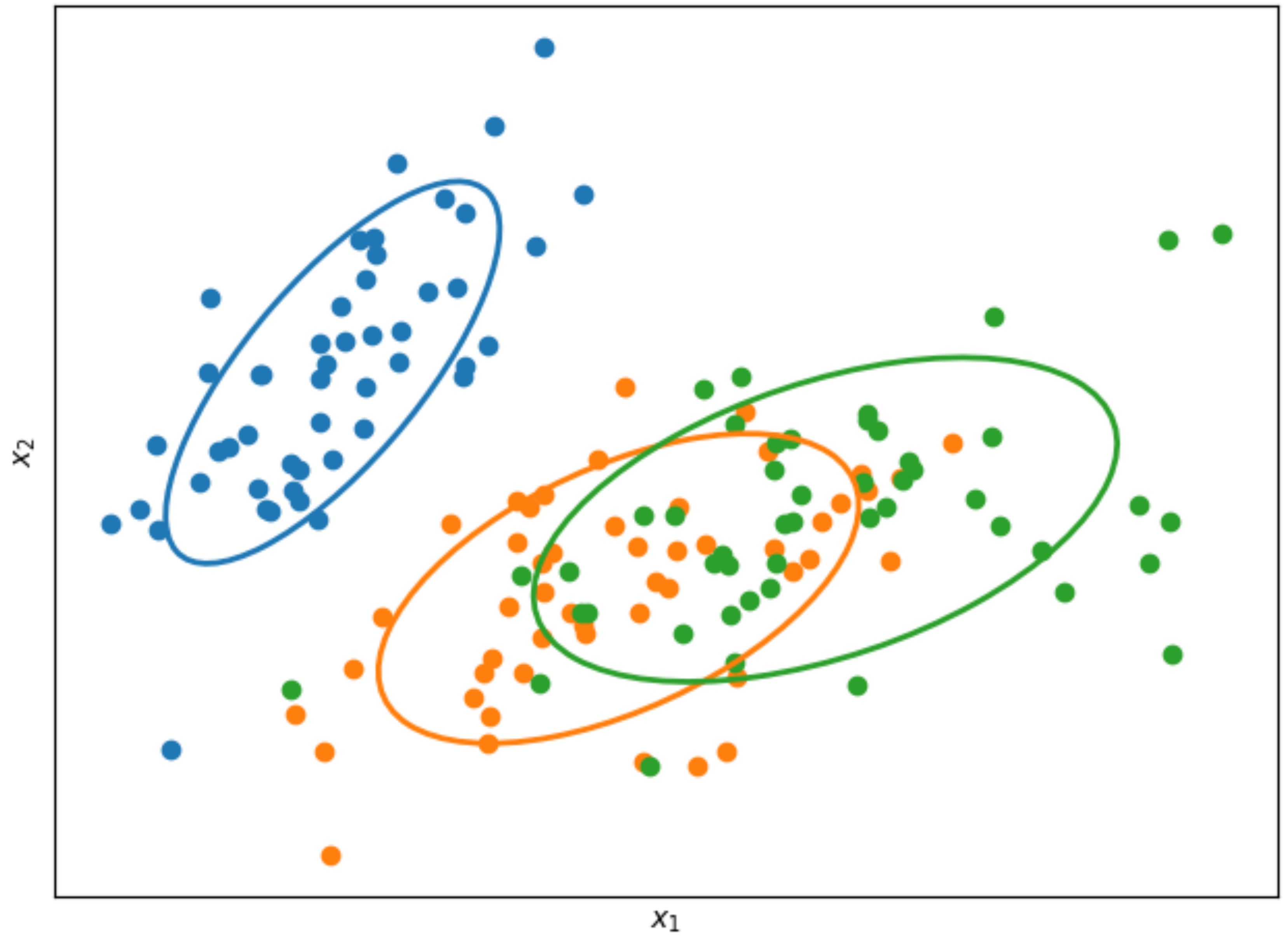


# Gaussian Bayes: Iris example

- $\hat{p}(y = c) = \frac{50}{150}; y \sim \text{Categorical}\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$

- Fit mean and covariance for each class, $\hat{p}(x \mid y = c) = \mathcal{N}(x; \hat{\mu}_c, \hat{\Sigma}_c)$




- How to use:
  - $\hat{p}(y \mid x) = \frac{\hat{p}(y)\hat{p}(x \mid y)}{\hat{p}(x)} \propto \hat{p}(y)\hat{p}(x \mid y)$
  - Maximum posterior (MAP): $\hat{y}(x) = \arg\max_y \hat{p}(y)\hat{p}(x \mid y)$

# Today's lecture

$k$-Nearest Neighbors

Bayes classifiers

**Naïve Bayes Classifiers**

Bayes error

# Representing joint distributions

- Assume data with binary features
- How to represent $p(x \mid y)$?
- Create a truth table of all $x$ values

| A | B | C |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |
| 1 | 1 | 1 |

# Representing joint distributions

- Assume data with binary features
- How to represent $p(x \mid y)$?
- Create a truth table of all $x$ values
- Specify $p(x \mid y)$ for each cell
- How many parameters?
  - $2^n - 1$

| A | B | C | p(A,B,C \| y=1) |
|---|---|---|---|
| 0 | 0 | 0 | 0.50 |
| 0 | 0 | 1 | 0.05 |
| 0 | 1 | 0 | 0.01 |
| 0 | 1 | 1 | 0.10 |
| 1 | 0 | 0 | 0.04 |
| 1 | 0 | 1 | 0.15 |
| 1 | 1 | 0 | 0.05 |
| 1 | 1 | 1 | 0.10 |

# Estimating joint distributions

- Can we estimate $p(x \mid y)$ from data?
- Count how many data points for each $x$?
  - If $m \ll 2^n$, most instances never occur
    - Do we predict that missing instances are impossible?
      - What if they occur in test data?
- Difficulty to represent and estimate go hand in hand
  - Model complexity → overfitting!

| A | B | C | p(A,B,C \| y=1) |
|---|---|---|---|
| 0 | 0 | 0 | 4/10 |
| 0 | 0 | 1 | 1/10 |
| 0 | 1 | 0 | 0/10 |
| 0 | 1 | 1 | 0/10 |
| 1 | 0 | 0 | 1/10 |
| 1 | 0 | 1 | 2/10 |
| 1 | 1 | 0 | 1/10 |
| 1 | 1 | 1 | 1/10 |

# Regularization

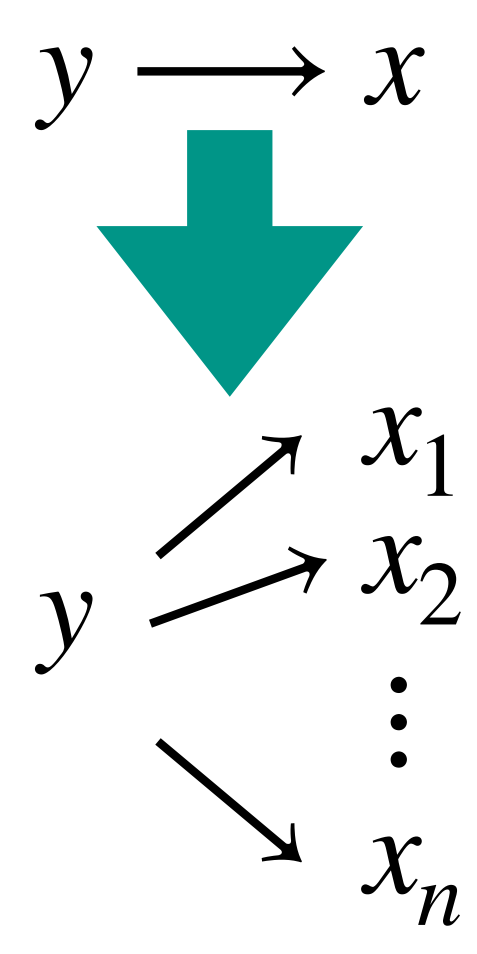
- Reduce effective size of model class
  - Hope to avoid overfitting
- One way: make the model more “regular”, less sensitive to data quirks
- Example: add small “pseudo-count” to the counts (before normalizing)
  - $\hat{p}(x \mid y = c) = \dfrac{\#_c(x) + \alpha}{m_c + \alpha \cdot 2^n}$
  - Not a huge help here, most cells will be uninformative $\dfrac{\alpha}{m_c + \alpha \cdot 2^n}$

# Simplifying the model

- Another way: reduce model complexity

- Example: assume features are independent of one another (in each class)
  - $p(x_1, x_2, \ldots, x_n \mid y) = p(x_1 \mid y) p(x_2 \mid y) \cdots p(x_n \mid y)$

- Now we only need to represent / estimate each $p(x_i \mid y)$ individually




| A | p(A \|y=1) |
|---|---|
| 0 | .4 |
| 1 | .6 |

| B | p(B \|y=1) |
|---|---|
| 0 | .7 |
| 1 | .3 |

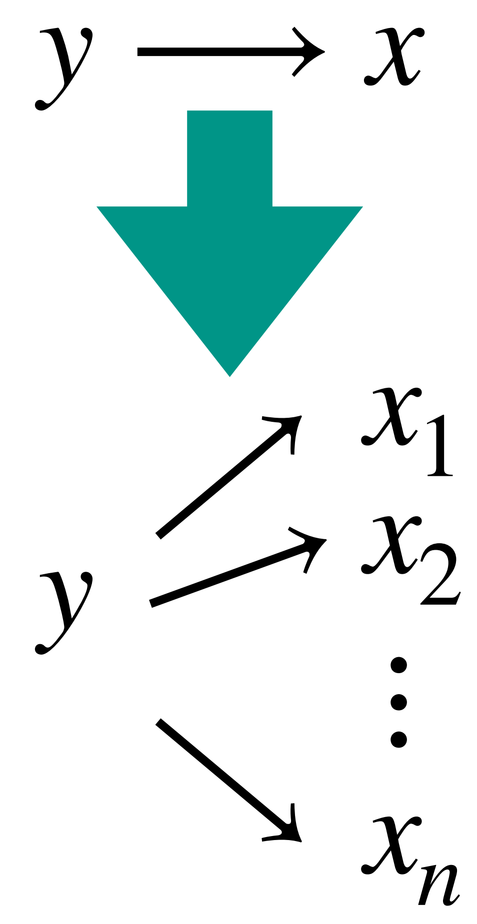
| C | p(C \|y=1) |
|---|---|
| 0 | .1 |
| 1 | .9 |

$\Rightarrow$

| A | B | C | p(A,B,C \| y=1) |
|---|---|---|---|
| 0 | 0 | 0 | .4 * .7 * .1 |
| 0 | 0 | 1 | .4 * .7 * .9 |
| 0 | 1 | 0 | .4 * .3 * .1 |
| 0 | 1 | 1 | … |
| 1 | 0 | 0 | |
| 1 | 0 | 1 | |
| 1 | 1 | 0 | |
| 1 | 1 | 1 | |

# Naïve Bayes models

- We want to predict some value $y$, e.g. auto accident next year
- We have many known indicators for $y$ (covariates) $x = x_1, \ldots, x_n$
  - E.g., age, income, education, zip code, ...
  - Learn $p(y | x_1, \ldots, x_n)$ — but cannot represent / estimate $O(2^n)$ values
- Naïve Bayes
  - Estimate prior distribution $\hat{p}(y)$
  - Assume $p(x_1, \ldots, x_n | y) = \prod_i p(x_i | y)$, estimate covariates independently $\hat{p}(x_i | y)$
  - Model: $\hat{p}(y | x) \propto \hat{p}(y) \prod_i \hat{p}(x_i | y)$




**causal structure wrong!**
**(but useful...)**

# Naïve Bayes models: example

- $y \in \{\text{spam, not spam}\}$
- $x$ = observed words in email
  - E.g., [“the” ... “probabilistic” ... “lottery” ...]
  - $x = [0,1,0,0,\ldots,0,1]$ ($1$ = word appears; $0$ = otherwise)
- Representing $p(x \mid y)$ directly would require $2^{\text{thousands}}$ parameters
- Represent each word indicator as independent (given class)
  - Reducing model complexity to thousands of parameters
- Words more likely in spam pull towards higher $p(\text{spam} \mid x)$, and v.v.

# Numeric example

- $\hat{p}(y=1) = \frac{4}{8} \quad = 1 - \hat{p}(y=0)$
- $\hat{p}(x_1, x_2 \mid y) = \hat{p}(x_1 \mid y)\hat{p}(x_2 \mid y)$
- $\hat{p}(x_1 = 1 \mid y = 0) = \frac{3}{4} \qquad \hat{p}(x_1 = 1 \mid y = 1) = \frac{2}{4}$
- $\hat{p}(x_2 = 1 \mid y = 0) = \frac{2}{4} \qquad \hat{p}(x_2 = 1 \mid y = 1) = \frac{1}{4}$
- What to predict for $x_1, x_2 = 1{,}1$? **prediction:** $\hat{y} = 0$
  - $\hat{p}(y=0)\hat{p}(x=1{,}1 \mid y=0) = \frac{4}{8} \cdot \frac{3}{4} \cdot \frac{2}{4} \qquad \hat{p}(y=1)\hat{p}(x=1{,}1 \mid y=1) = \frac{4}{8} \cdot \frac{2}{4} \cdot \frac{1}{4}$

| $x_1$ | $x_2$ | y |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 1 |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 1 | 0 |
| 0 | 0 | 1 |
| 1 | 0 | 1 |

# Numeric example

- $\hat{p}(y=1) = \frac{4}{8} \quad = 1 - \hat{p}(y=0)$
- $\hat{p}(x_1, x_2 \mid y) = \hat{p}(x_1 \mid y)\hat{p}(x_2 \mid y)$
- $\hat{p}(x_1 = 1 \mid y=0) = \frac{3}{4} \qquad \hat{p}(x_1 = 1 \mid y=1) = \frac{2}{4}$
- $\hat{p}(x_2 = 1 \mid y=0) = \frac{2}{4} \qquad \hat{p}(x_2 = 1 \mid y=1) = \frac{1}{4}$
- What is $\hat{p}(y=1 \mid x_1 = 1, x_2 = 1)$?

$$\frac{\hat{p}(y=1)\hat{p}(x=1,1 \mid y=1)}{\hat{p}(x=1,1)} = \frac{\hat{p}(y=1)\hat{p}(x=1,1 \mid y=1)}{\hat{p}(y=0)\hat{p}(x=1,1 \mid y=0) + \hat{p}(y=1)\hat{p}(x=1,1 \mid y=1)} = \frac{\frac{4}{8}\cdot\frac{2}{4}\cdot\frac{1}{4}}{\frac{4}{8}\cdot\frac{3}{4}\cdot\frac{2}{4} + \frac{4}{8}\cdot\frac{2}{4}\cdot\frac{1}{4}} = \frac{1}{4}$$

| $x_1$ | $x_2$ | y |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 1 |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 1 | 0 |
| 0 | 0 | 1 |
| 1 | 0 | 1 |

# Recap

- Bayes' rule: $p(y\,|\,x) = \dfrac{p(y)p(x\,|\,y)}{p(x)}$
- Bayes classifiers: estimate $p(y)$ and $p(x\,|\,y)$ from data
- Naïve Bayes classifiers: assume independent features $p(x\,|\,y) = \prod_i p(x_i\,|\,y)$
  - Estimate each $p(x_i\,|\,y)$ individually
- Maximum posterior (MAP): $\hat{y}(x) = \arg\max_y p(y\,|\,x) = \arg\max_y p(y)p(x\,|\,y)$
  - Normalizer $p(x)$ not needed

# Today's lecture

$k$-Nearest Neighbors

Bayes classifiers

Naïve Bayes Classifiers

**Bayes error**

# Bayes classification error

- What is the training error of the MAP prediction $\hat{y}(x) = \arg\max_y p(y \mid x)$?

| Features | # bad | # good | prediction: |
|---|---|---|---|
| X=0 | 42 | 15 | bad |
| X=1 | 338 | 287 | bad |
| X=2 | 3 | 5 | good |

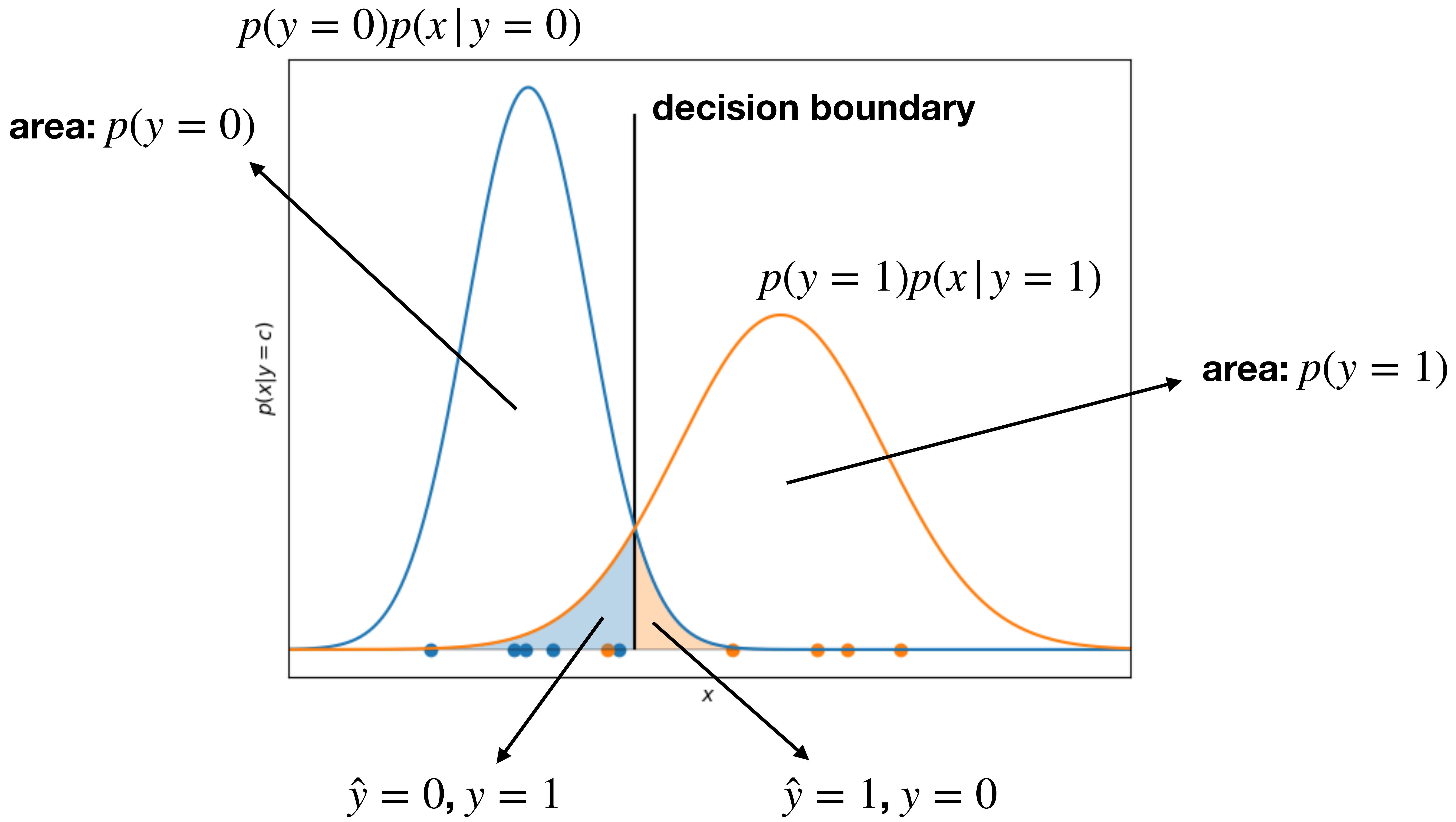

- $p(\hat{y} \neq y) = \dfrac{15 + 287 + 3}{690} = 0.442$

- Bayes error rate: probability of misclassification by MAP of true posterior

# Bayes error rate

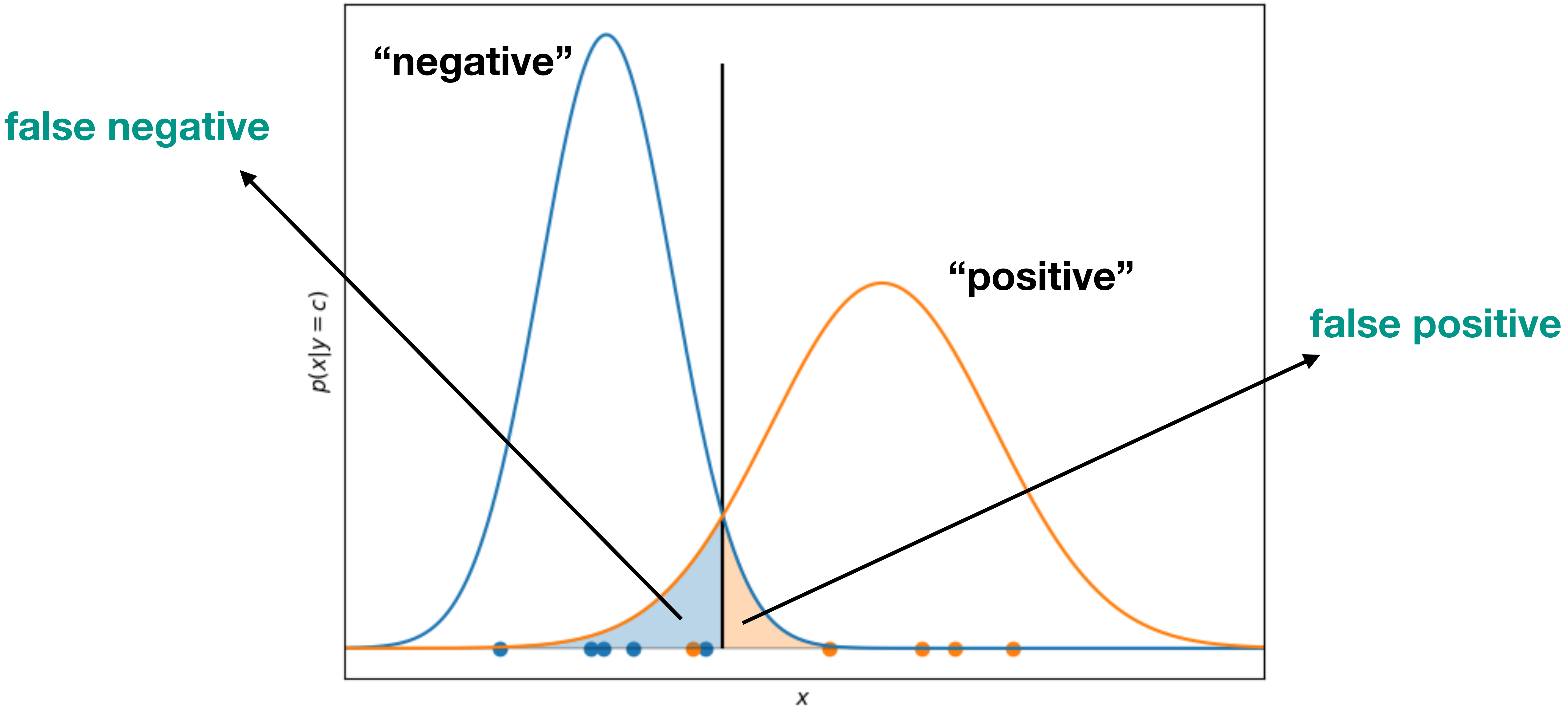
- Suppose that we know the true probabilities $p(x, y)$
  - And that we can compute prior $p(y)$ and posterior $p(y \mid x)$
- Bayes-optimal decision = MAP: $\hat{y} = \arg\max_y p(y \mid x)$
- Bayes error rate: $\mathbb{E}_{x,y\sim p}[\hat{y} \neq y] = \mathbb{E}_{x\sim p}[1 - \max_y p(y \mid x)]$
  - This is the optimal error rate of any classifier
  - Measures intrinsic hardness of separating $y$ values given only $x$
    - But may get better with more features
- Normally we cannot estimate the Bayes error rate, only approximate with good classifier

# Bayes error rate: Gaussian example



# Types of error

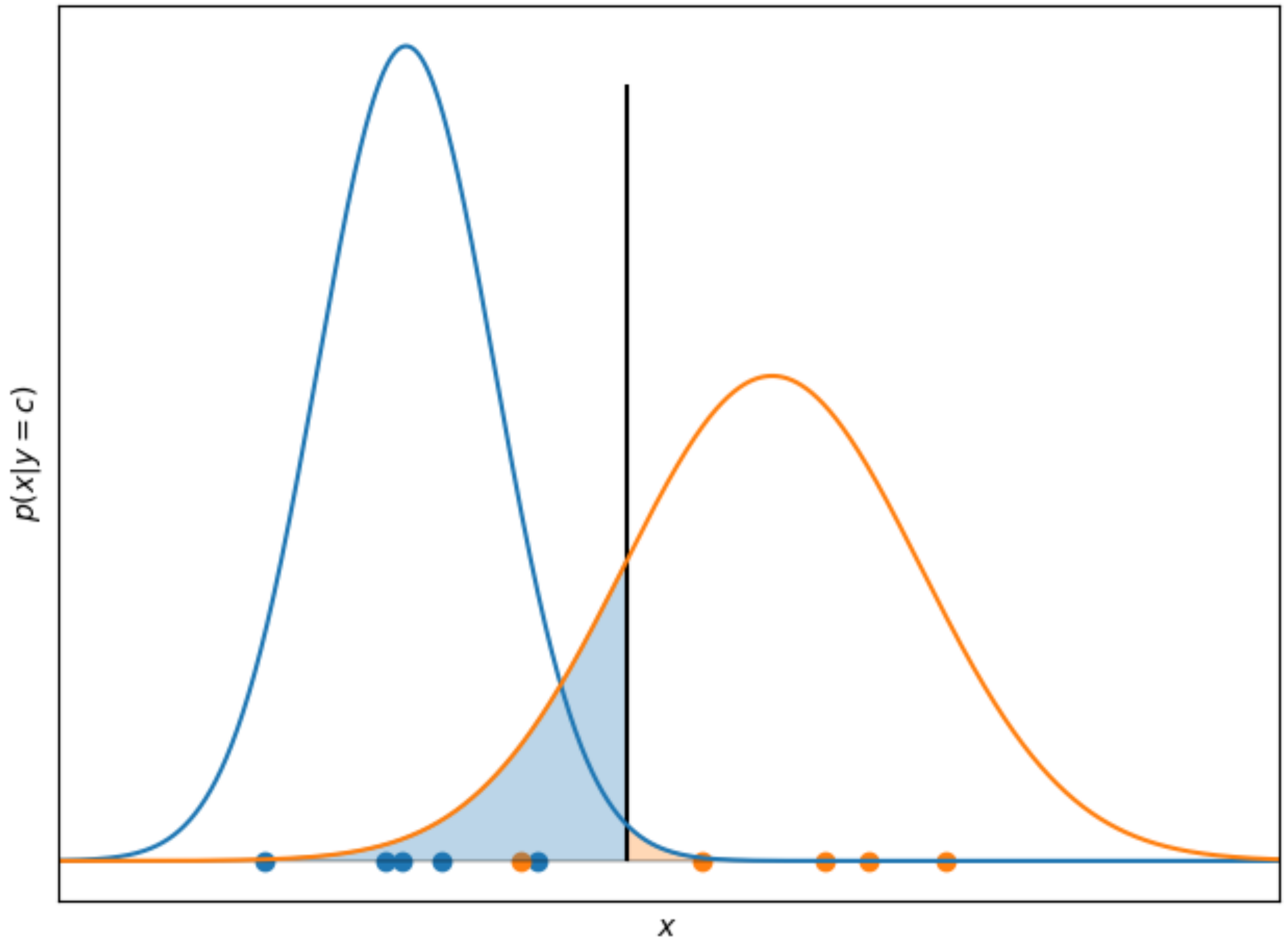
- Not all errors are equally bad
  - Do some cost more? (e.g. red / green light, diseased / healthy)




- False negative rate: $\frac{p(y=1, \hat{y}=0)}{p(y=1)}$; false positive rate: $\frac{p(y=0, \hat{y}=1)}{p(y=0)}$

# Cost of error

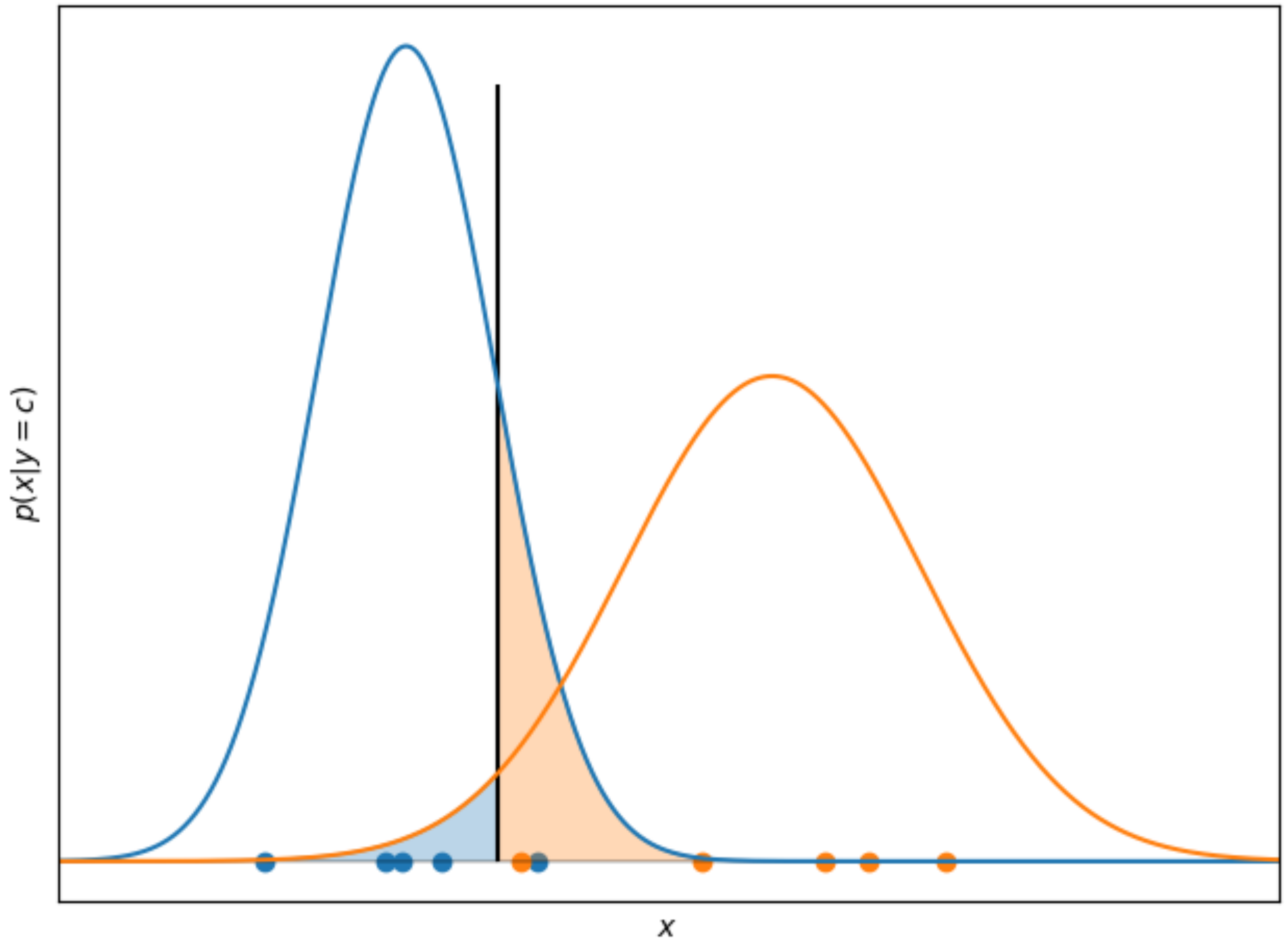
- Weight different costs differently
  - $\alpha \cdot p(y=0)p(x \mid y=0) \lessgtr p(y=1)p(x \mid y=1)$




- Increase $\alpha$ to prefer class 0

# Cost of error

- Weight different costs differently
  - $\alpha \cdot p(y=0)p(x \mid y=0) \lessgtr p(y=1)p(x \mid y=1)$




- Decrease $\alpha$ to prefer class 1

# Logistics

**assignment 1**

- Assignment 1 is due Tuesday

**recordings**

- Lectures will be recorded, starting today
- Recordings from Fall'21 also available