University of Pittsburgh

ECE 2195: Special Topics – Computers Machine Learning

Classification Setting – Bayes Classifier and KNN

Mai Abdelhakim, PhD

ECE Department

Swanson School of Engineering

University of Pittsburgh

maia@pitt.edu

Classification Setting

- The response in is qualitative
  - y_i belongs to a finite set of possible classes: $y_i \in C, C = \{1,2,..,m\}$
    - E.g. spam/not spam:
- Build classifier that assigns class label to a future unlabeled observation
- To assess the model accuracy, we typically evaluate the **error rate**
  - **Accuracy =1 – error rate**
- $\hat{y}_o = \hat{f}(x_0)$ is the **predicted output class**
- **Test error rate** associated with **test** observations (x_0, y_0):

$$Average(I(y_o \neq \hat{y}_o))$$

$I(y_o \neq \hat{y}_o)$ is indicator variable that is equal to 1 when $y_o \neq \hat{y}_o$, and zero when $y_o = \hat{y}_o$

Classification Setting

- The error rate is minimized by a simple classifier, called **Bayes classifier**
- Bayes classifier assigns each observation to the **most likely class given the feature values**.
  - Assign x_0 to class j that has **largest** $Pr(Y=j|X=x_0)$
  - $Pr(Y=j|X=x_0)$ *is the Posterior probability*
    - *Example, spam filter: class label is Y=1 (spam), Y=2 (not spam)*
      - *Pr(Y=1) is the probability that Y is spam email – Prior probability*
      - *Pr(Y=1|X) is the conditional probability that Y is spam given features of an email, e.g. size of email*

Bayes Classifier – Decision Rule

- Assume two classes y=w1 and y=w2
  - **Decide state of nature = w1 IF** $\Pr(\boldsymbol{w1}|\boldsymbol{x}) > \Pr(\boldsymbol{w2}|\boldsymbol{x})$,
    - **otherwise (o.w.) decide w2**

- Bayes decision rule minimizes the probability of error
  - $\Pr(error|x) = \min[\Pr(\boldsymbol{w1}|\boldsymbol{x}), \Pr(\boldsymbol{w2}|\boldsymbol{x})]$
    - Unconditional error Pr(error) is obtained by integration over x

Bayes Classifier – Decision Rule (Complete Information is available)

- Assume two classes y=w1 and y=w2
  - **Decide state of nature = w1 IF** $\Pr(\boldsymbol{w1}|\boldsymbol{x}) > \Pr(\boldsymbol{w2}|\boldsymbol{x})$,
    - **otherwise (o.w.) decide w2**

- **Special cases**
  1. If priors are equal Pr(w1) = Pr(w2) ➔Decide w1 if $\Pr(x|w1) > \Pr(x|w2)$,
     o.w. choose w2
     - **Maximum likelihood**
  2. If Pr(x|w1) = Pr(x|w2); Decide w1 if Pr(w1) > Pr(w2), o.w. decide w2

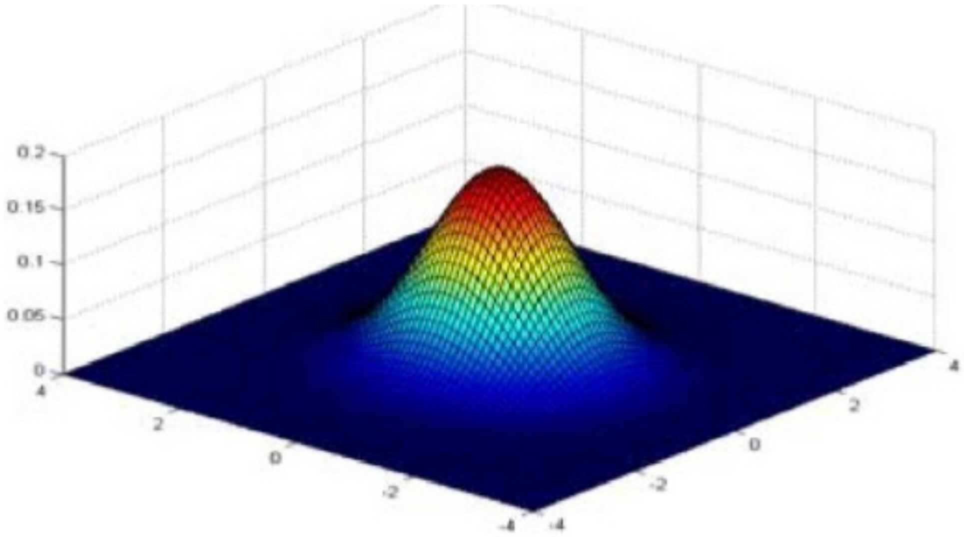
Recall Gaussian Distribution

- 1-Dimensional Gaussian

$$p(x|\mu,\sigma) = \frac{1}{(2\pi\sigma^2)^{1/2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}$$

- 2-Dimensional Gaussian

$$p(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{2\pi|\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$$



- D-Dimensional Gaussian

$$p(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$$

Example: One feature

- $P(X|w1) \sim N(\mu_1, \sigma_1)$, $P(X|w2) \sim N(\mu_2, \sigma_2)$

Example: Bayesian decision rule

$$P(X|w1) \sim N\left(\begin{bmatrix}\mu_{11}\\ \mu_{12}\end{bmatrix}, \Sigma_1\right), P(X|w2) \sim N\left(\begin{bmatrix}\mu_{21}\\ \mu_{22}\end{bmatrix}, \Sigma_2\right), and\ equal\ priors$$

Example: Bayesian Decision Boundary

$$P(X|w1) \sim N\left(\begin{bmatrix}\mu_{11} \\ \mu_{12}\end{bmatrix}, \Sigma_1\right), P(X|w2) \sim N\left(\begin{bmatrix}\mu_{21} \\ \mu_{22}\end{bmatrix}, \Sigma_2\right)$$

$$g_1(x) = -\ln(2\pi) - \frac{1}{2}\ln|\Sigma_1| - \frac{1}{2}\left(\vec{x} - \overrightarrow{\mu_1}\right)^t \Sigma_1^{-1}\left(\vec{x} - \overrightarrow{\mu_1}\right) = -\ln(2\pi) - \frac{1}{2}\ln|\Sigma_1| - \frac{1}{2}\begin{bmatrix}x_1 - \mu_{11} & x_2 - \mu_{12}\end{bmatrix}\Sigma_1^{-1}\begin{bmatrix}x_1 - \mu_{11} \\ x_2 - \mu_{12}\end{bmatrix}$$

$$g_2(x) = -\ln(2\pi) - \frac{1}{2}\ln|\Sigma_2| - \frac{1}{2}\left(\vec{x} - \overrightarrow{\mu_2}\right)^t \Sigma_2^{-1}\left(\vec{x} - \overrightarrow{\mu_2}\right) = -\ln(2\pi) - \frac{1}{2}\ln|\Sigma_2| - \frac{1}{2}\begin{bmatrix}x_1 - \mu_{21} & x_2 - \mu_{22}\end{bmatrix}\Sigma_2^{-1}\begin{bmatrix}x_1 - \mu_{21} \\ x_2 - \mu_{22}\end{bmatrix}$$

$At\ the\ boundary\ g_1(x) = g_1(x)$ ➔ function of features

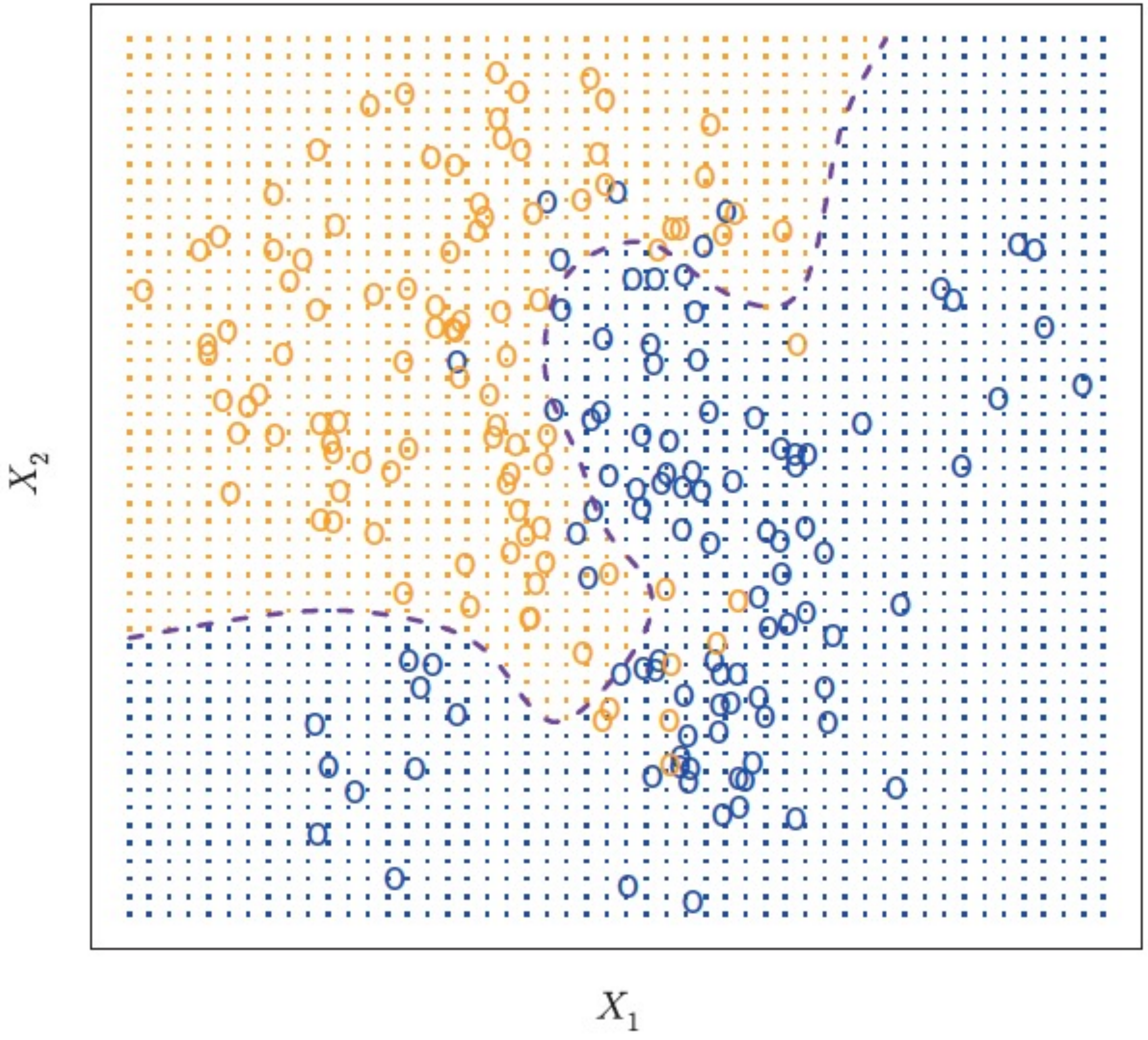
Bayes Classifier

Figure shows two features of **100** simulated observations of two classes

- For each value of X_1 and X_2 there is a probability of each classes
  - Here **conditional distribution is known**
- The dashed line is called **decision boundary**, where the probability is exactly 50%
- Decisions are based on this boundary
  - Each side of the decision boundary belongs to a different class

Feature Space 2-D

Class 1: blue
Class 2: orange



Simulated data of 100 observations

K-Nearest Neighbors

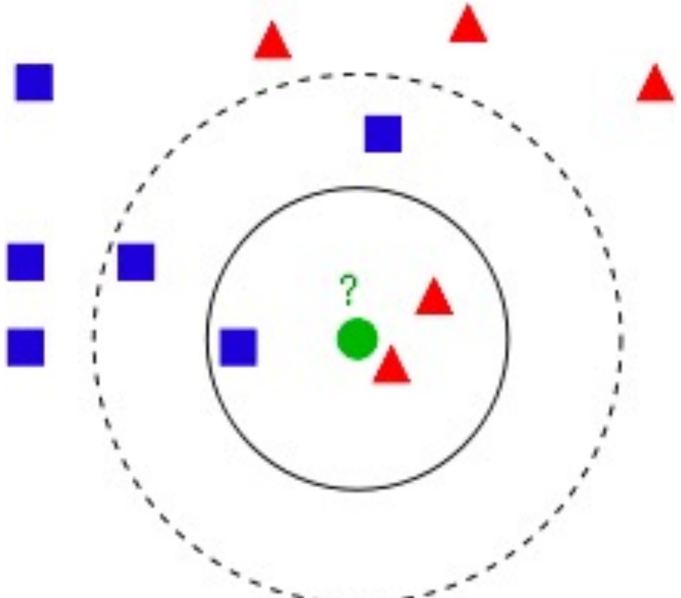
- Bayes classifiers assumes complete information about distribution
- Typically, we do not have the distribution and it is hard to get conditional probabilities
  - We have few points, if any, at each X

- Many methods tries to **estimate the conditional distribution**

K-Nearest Neighbors

- K-nearest neighbor (KNN):
  - Define a positive integer K
  - For each **test observation x_0** , identify **K points in the training data that are closest to x_0** referred to as $\mathcal{N}_0$
  - **Estimate the conditional probability** for class j as **fraction of points in $\mathcal{N}_0$ whose label values equal to j**

$$\Pr(Y = j | X = x_0) = \frac{1}{K} \sum_{i \in \mathcal{N}_0} I(y_i = j)$$



  - Then assign to the class with largest conditional probability
  - Decision depends on the choice of K

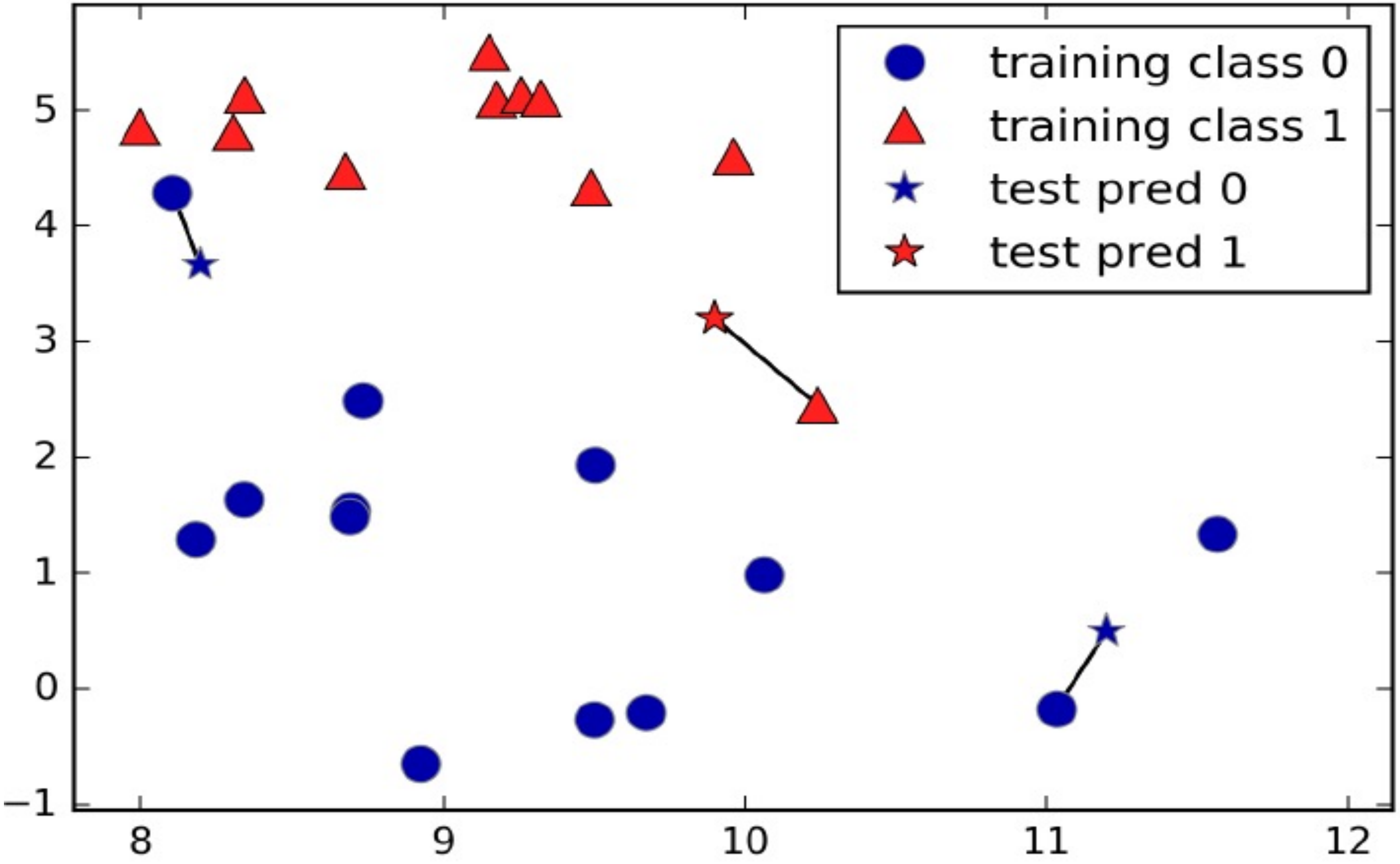
What is the estimated conditional probabilities if k=3, k=5?

K-Nearest Neighbors - Simplified

- For any given test data point, we find the **K closest neighbors** to this point in the **training data**, and examine their corresponding class (y).
  - **Euclidean distance** can used to find close neighbors
  - Assume Point 1: with feature vector $P_1= \{x_{11}, x_{12},\ldots, x_{1p}\}$
    Point 2, with feature vector $P_2= \{x_{21}, x_{22},\ldots, x_{2p}\}$
    Then the Euclidean distance between the two samples is:
    $d(P_1,P_2)=\sqrt{\sum_{j=1}^{p}(x_{1j} - x_{2j})^2}$

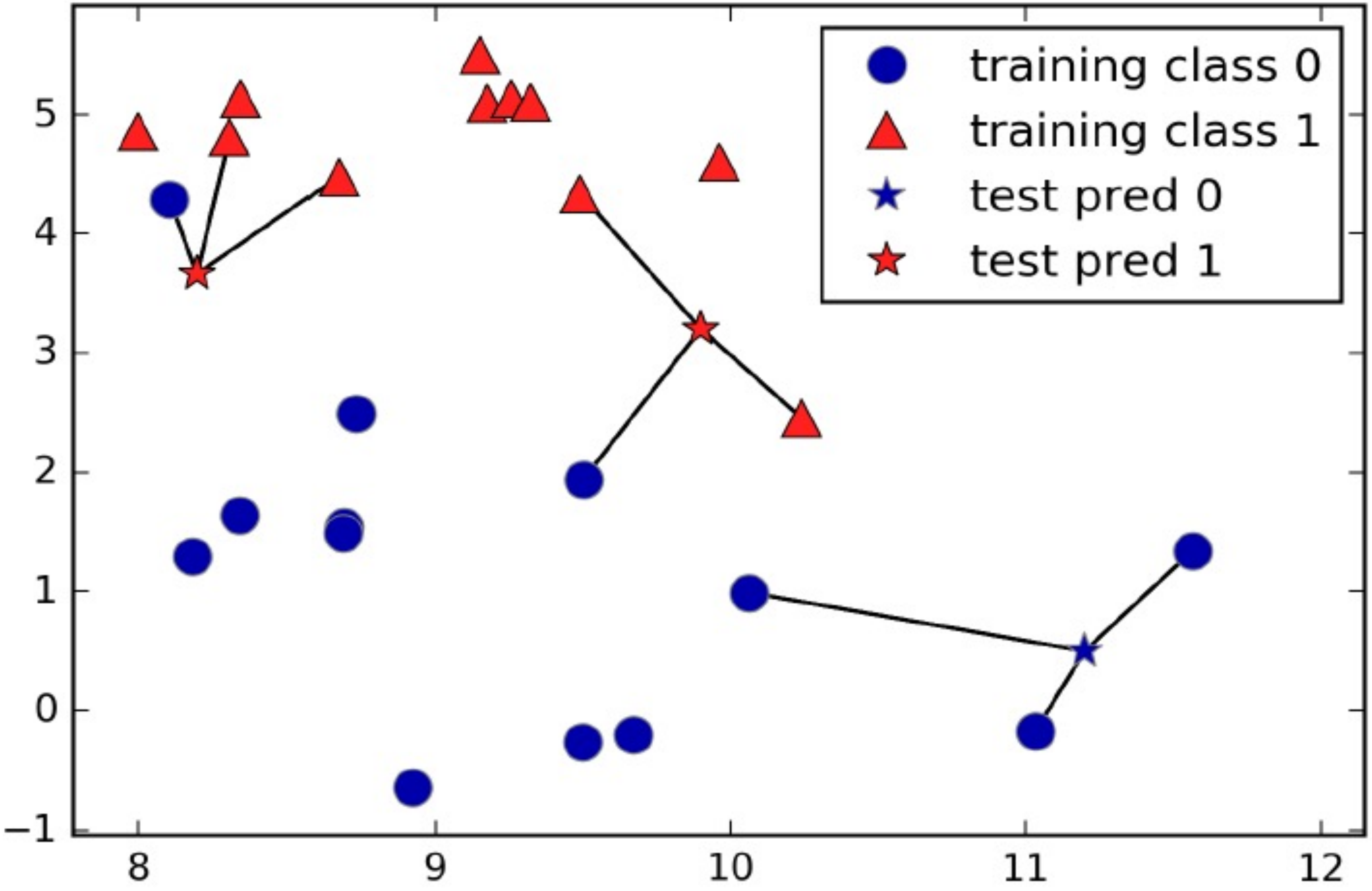
    $X_{i,j}$: the *j*th feature of *i*th data point
- Assign the data point to class from which **majority of neighbors** belong

KNN with K=1



training class 0
training class 1
test pred 0
test pred 1
5
4
3
2
1
0
−1
8
9
10
11
12

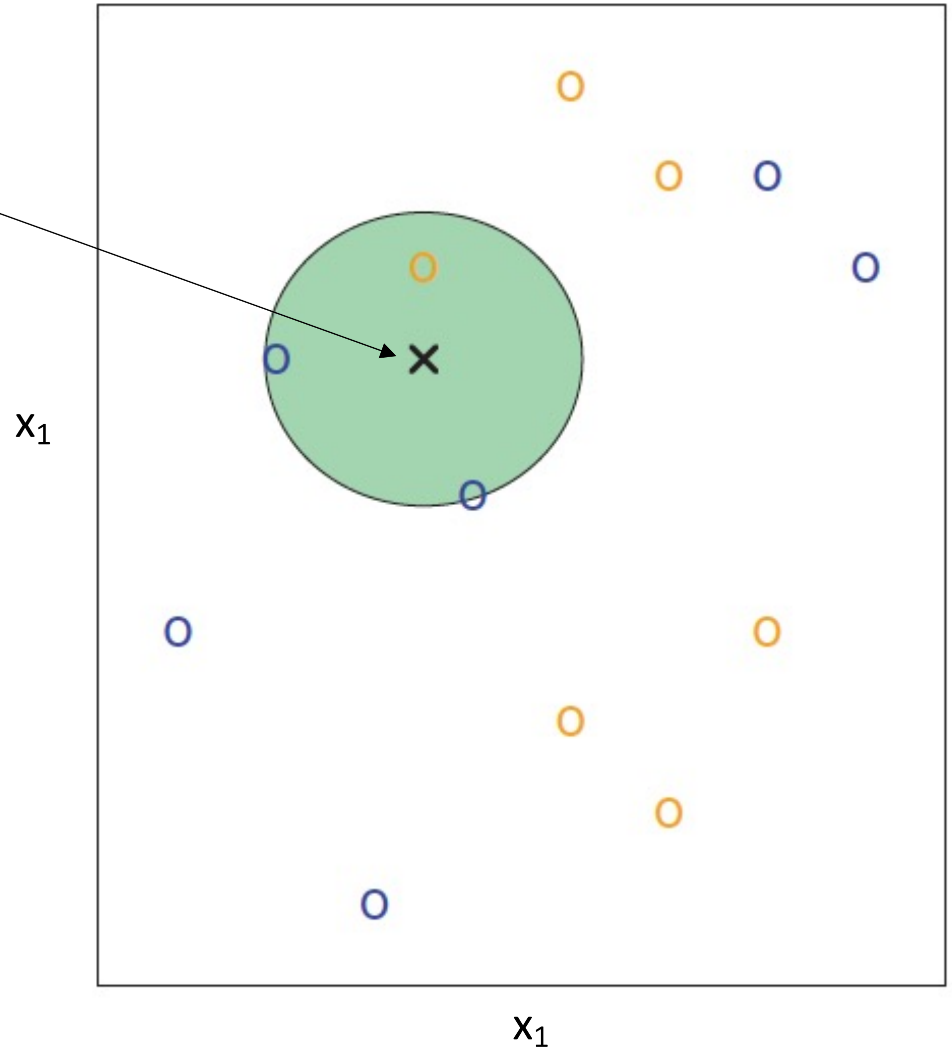
KNN with K=3

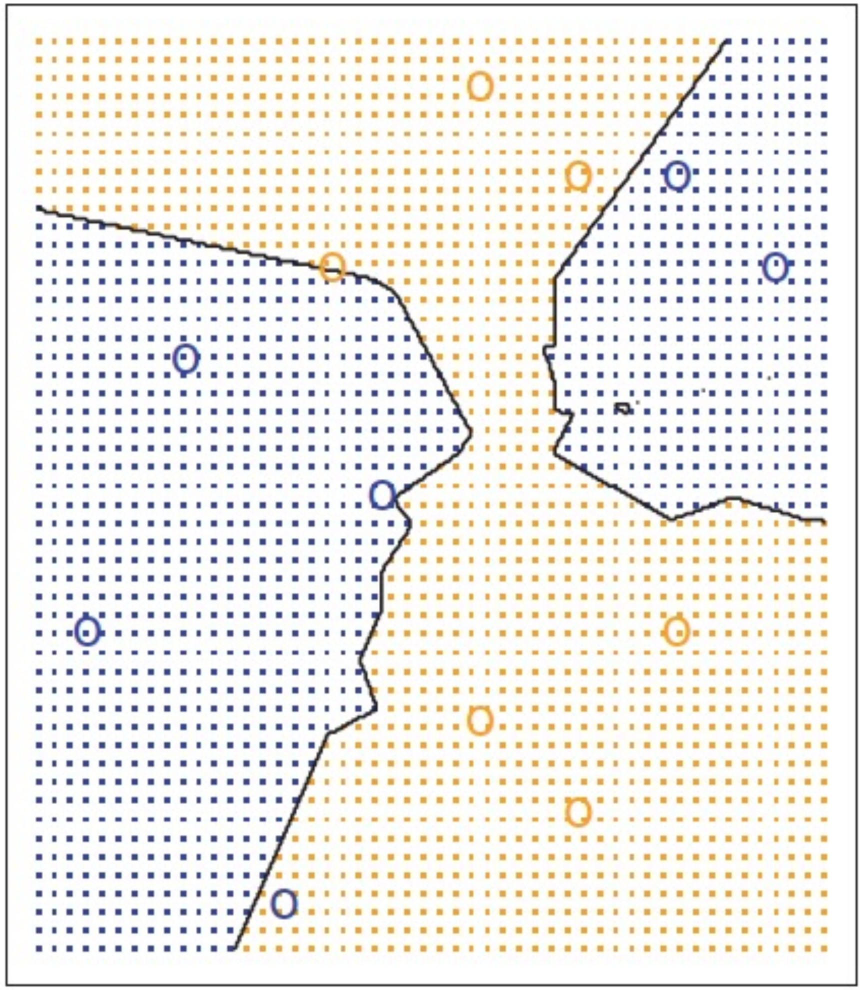


training class 0
training class 1
test pred 0
test pred 1
5
4
3
2
1
0
−1
8
9
10
11
12

K= 3, 2-D Feature Space Example

Within the 3 nearest neighbors, two of them belong to the blue class (majority). Thus, we classify x as blue





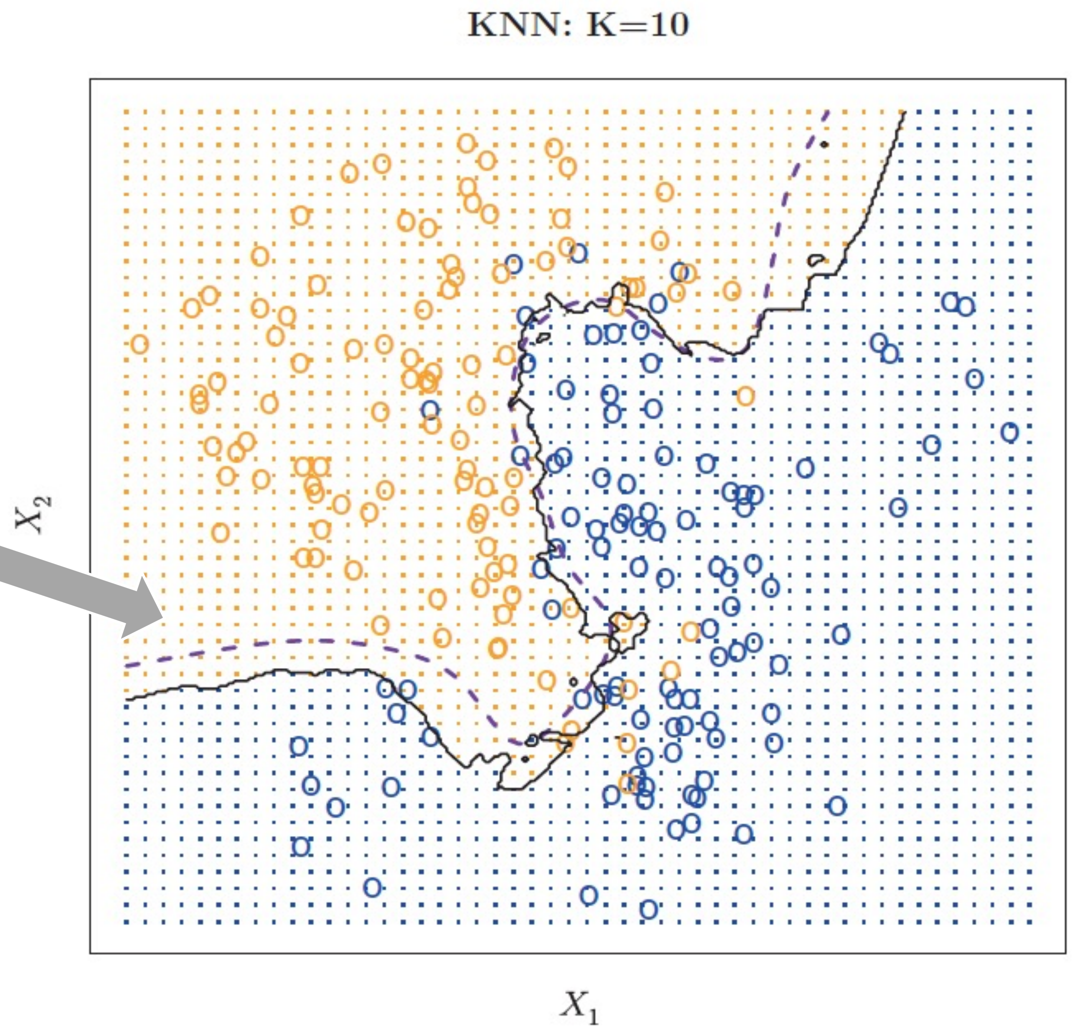
2 classes: blue and orange circle

- KNN is simple, sometimes it is close to the optimal Bayes classifier

- Fig. shows an example of using K=10

Dashed line: Bayes decision boundary
Solid line: KNN decision boundary

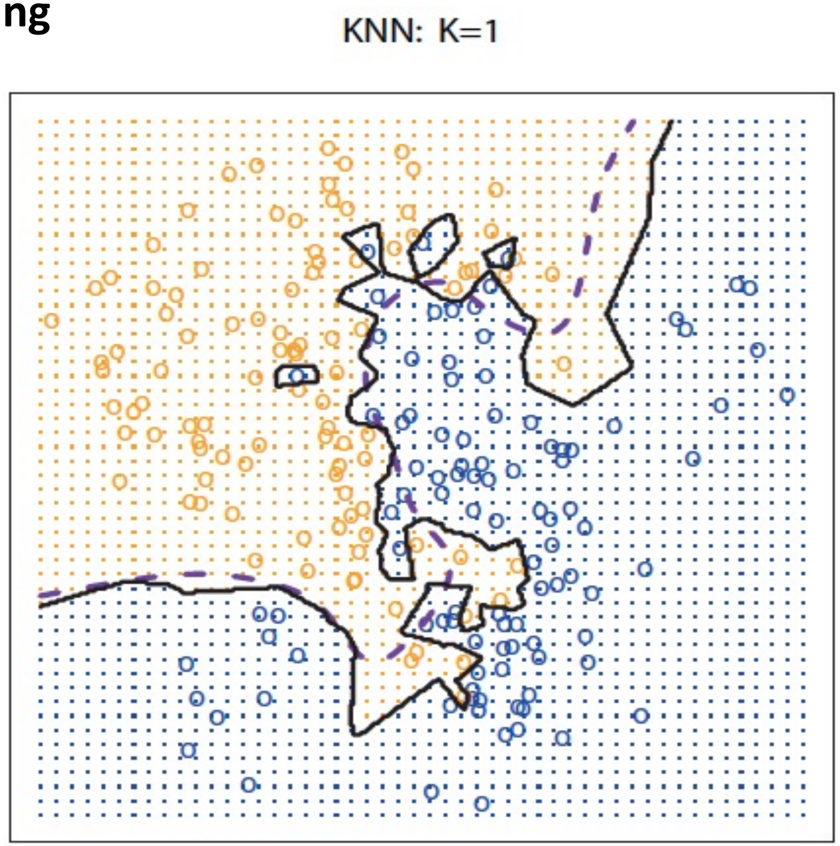
100 simulated observations

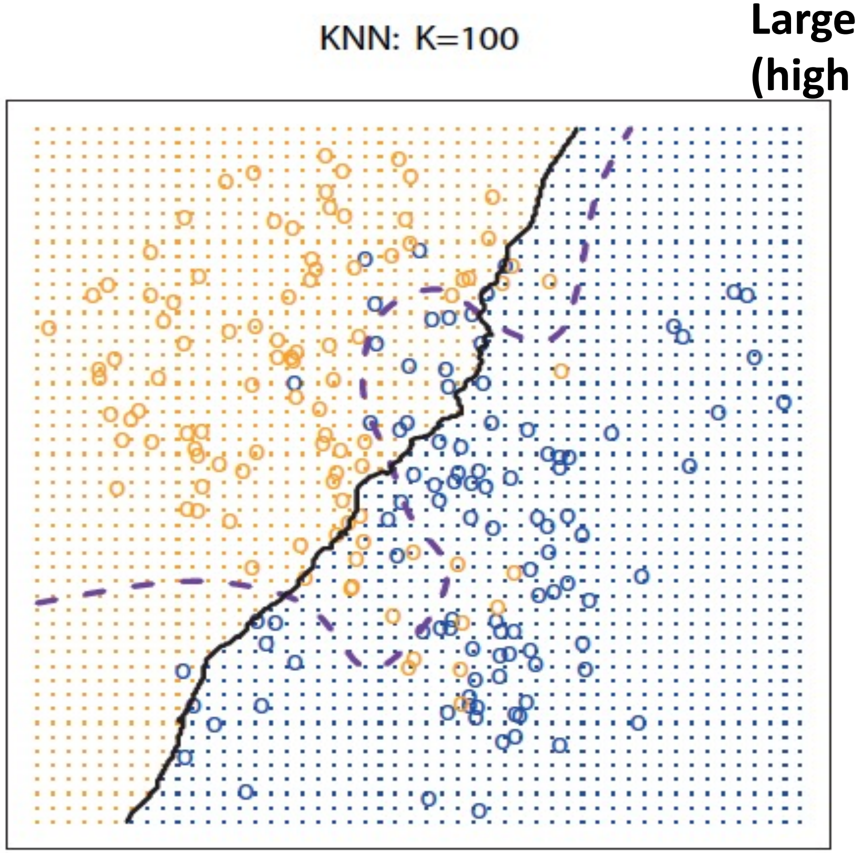


K-Nearest Neighbors

- Choice of K has huge impact on the performance
  - Bias-variance trade-off applies

K=1 => overfitting (high variance)



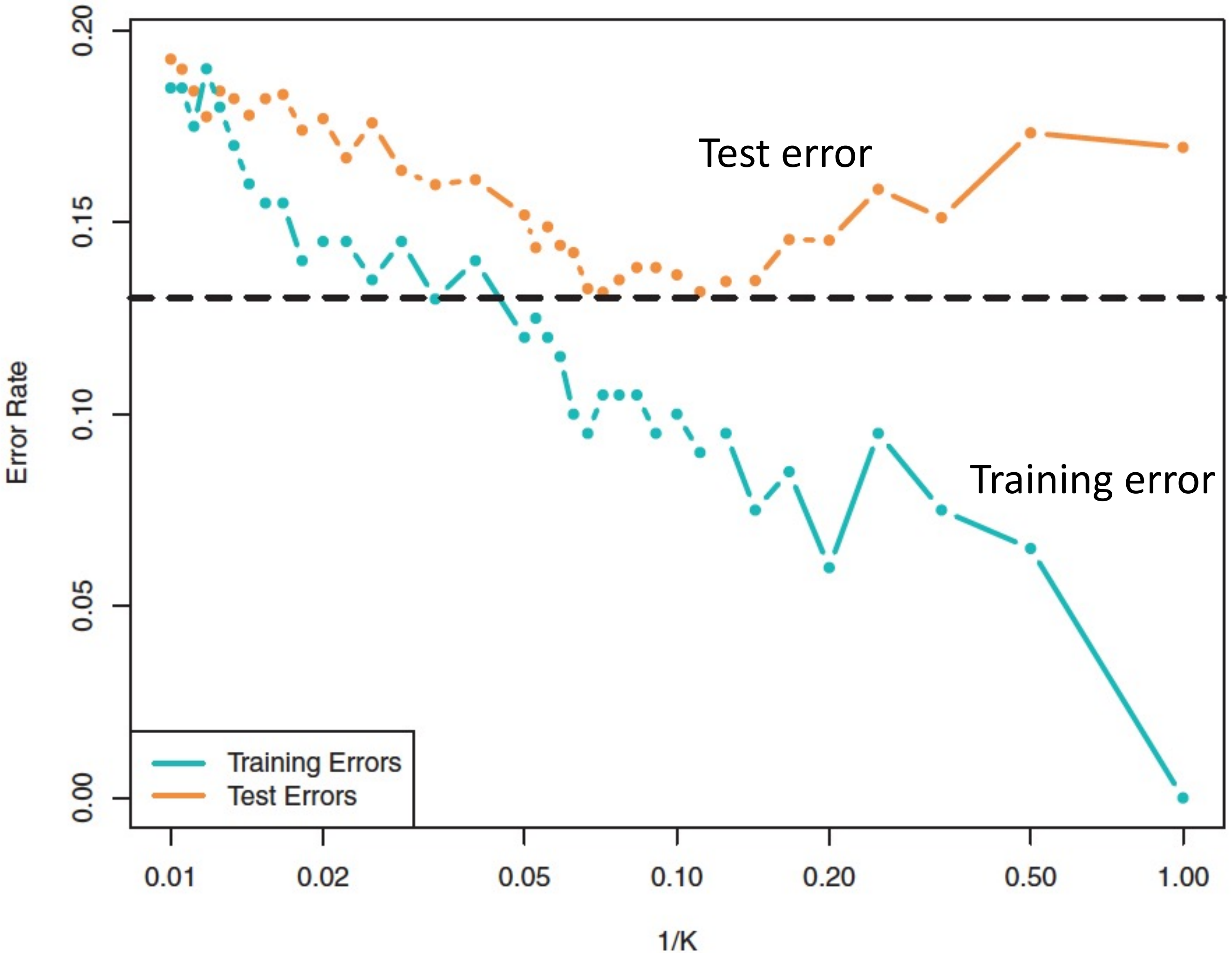


Large K =>underfitting (high bias)

Dashed line: Bayes decision boundary

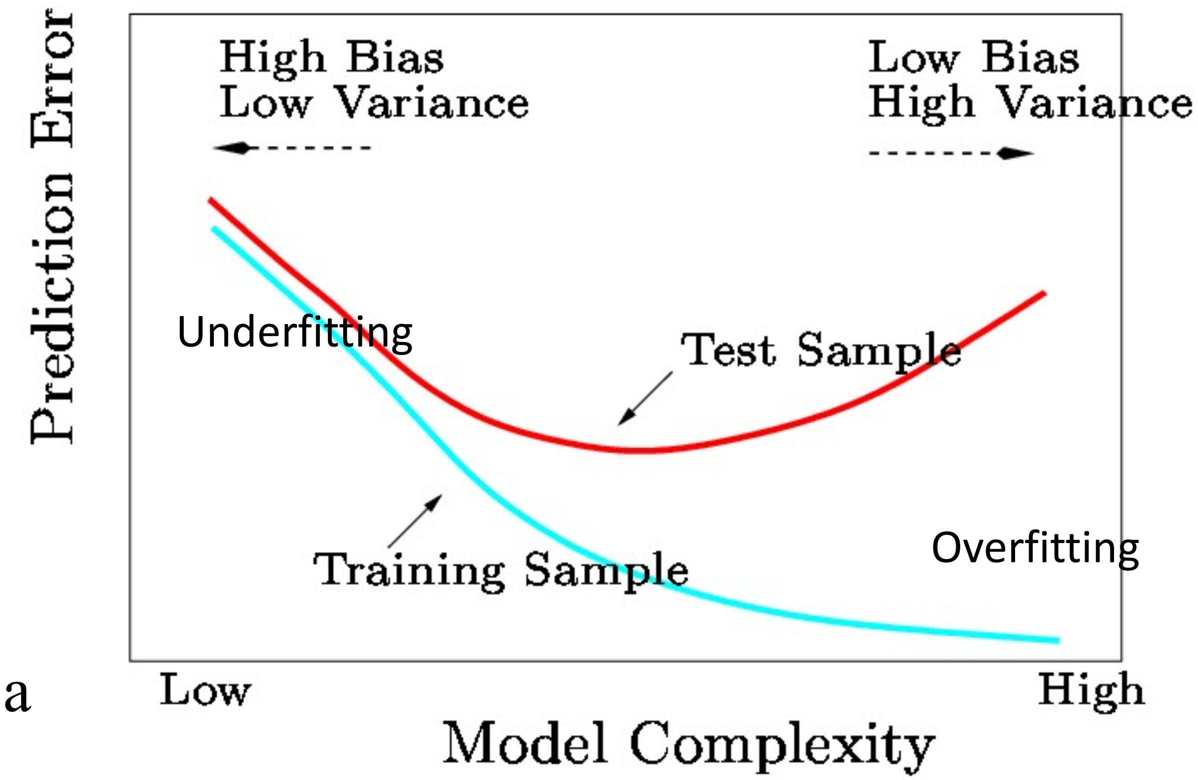
Different K different decision boundaries

- Same principles apply to classification problems



Trade-offs

- Prediction accuracy versus model complexity (flexibility)
    - Bias-variance trade-off
- Good fit versus over-fit or under-fit



Keep this picture in mind when choosing a learning method.

More flexible/complicated model is not always better!