Chapter 2 Statistical Learning
– Math 313 Statistics for Data Science

Guangliang Chen

Associate Professor

cheng@hope.edu

Hope College, Fall 2023

Presentation Overview

A motivating example

Consider the following scenario:

Suppose that you are teaching an introductory statistics class for the third time and just gave the last midterm of the semester (there is still a final left).

After getting the test back, a student came to you, very concerned about her overall grade in the end. She would like you to help her predict her final exam score, if possible.

Questions

Do you think whether you can help her with this? What other kind of information might you want to collect as well?

This is an example of a regression problem:

- **Response**: final example score
- **Predictors**:
  - Midterm scores
  - Current homework score
  - Attendance
  - Major
  - Academic year
  - Current GPA
  - Work status
  - etc.

To be able to make a prediction, you also need the historical data of the classes you taught in the past.

The mathematical perspective of regression

Suppose that we observe

- a set of predictors $\vec{X} = (X_1, X_2, \ldots, X_p)$, also called features or independent variables;
- a quantitative response Y, also called dependent variable

and that their relationship is described by a fixed (but unknown) function f:

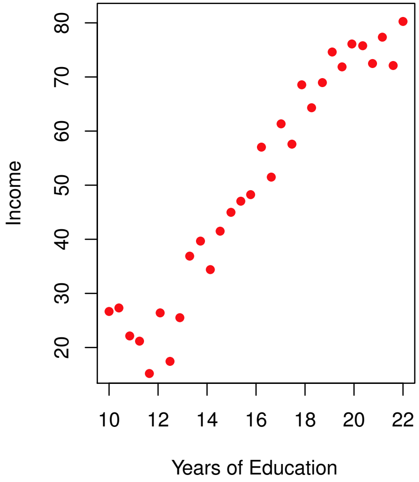
$$Y = f(\vec{X}) + \varepsilon$$

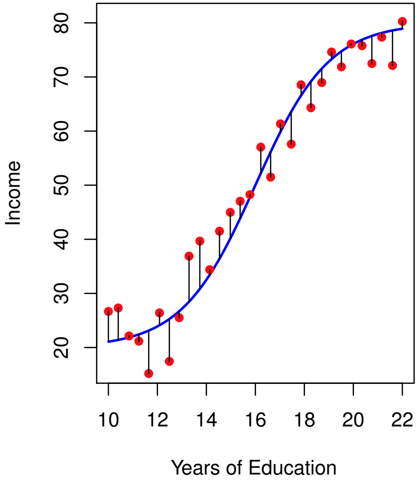
where ε is a random error term that is independent of $\vec{X}$ and has zero mean: $\mathrm{E}(\varepsilon) = 0$.

The goal of regression is to estimate f based on a sample from the model, $(\vec{x}_1, y_1), (\vec{x}_2, y_2), \ldots, (\vec{x}_n, y_n)$, often called training data.

An example

Left: training data (true *f* invisible);   Right: one estimate of *f*





Why estimate f?

Let $\hat{f}$ be an estimate of f (with an explicit or implicit form). The resulting prediction of the response Y, given the predictors in $\vec{X}$, is $\hat{Y} = \hat{f}(\vec{X})$.

There are two kinds of tasks:

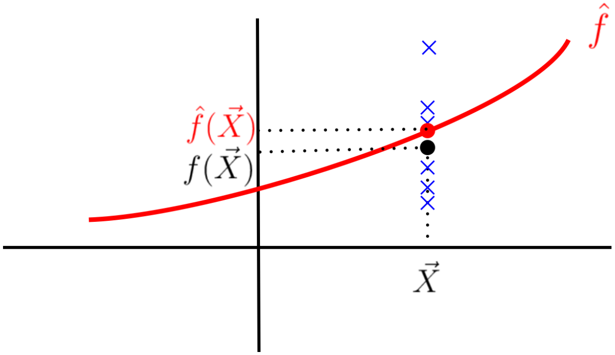
- **Prediction** (model accuracy): What is the response at a given location?
- **Inference** (model simplicity and interpretability): Which variables are important to the response and in which way?

The prediction task

The prediction of Y at a fixed location $\vec{X}$ is obviously $\hat{Y} = \hat{f}(\vec{X})$.

The average prediction error is

$$\mathrm{E}_Y\left[(Y - \hat{Y})^2 \mid \vec{X}, \hat{f}\right] = \underbrace{\left(f(\vec{X}) - \hat{f}(\vec{X})\right)^2}_{\text{reducible error}} + \underbrace{\mathrm{Var}(\varepsilon)}_{\text{irreducible error}}$$



We can work to improve the reducible error (by varying $\hat{f}$) but not the irreducible error because it is independent of the choice of $\hat{f}$.

The focus is thus on the reducible error but minimizing it can be very challenging because we do not directly observe $f(\hat{X})$ but only a corrupted version of it, $Y = f(\vec{X}) + \varepsilon$.

Another reason is that the space of possible functions $\hat{f}$ is extremely large.

The inference task

Inference concerns answering the following questions:

- *Which predictors are associated with the response?*
- *What is the direction of the association between the response and each predictor?*
- *What kind of relationship (linear or nonlinear) exists between the response and each predictor?*
- *What predictors are the most important for determining the response?*

How to estimate *f*?

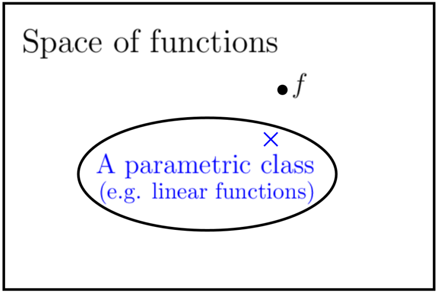
Methods for estimating *f* can be divided into the following two categories:

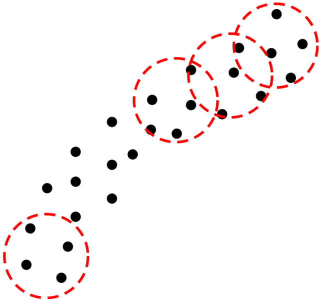
- **Parametric**: Suppose *f* has a particular functional form, such as linear:

  $$f(\vec{X}) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p$$

  Then the problem of estimating *f* reduces to estimating the coefficients $\beta_0, \beta_1, \ldots, \beta_p$ (called parameters of the model), based on a set of observations.
- **Nonparametric**: No explicit assumptions about the functional form of *f*; often relies on local approximation (globally any shape can be produced).

Parametric vs nonparametric



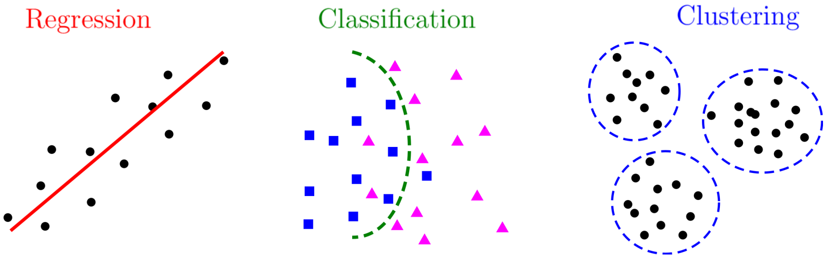


Remark

Parametric models (especially linear) tend to be more restrictive (but faster to compute and easier to interpret). In contrast, nonparametric methods are more flexible (but more complicated in computing and harder to interpret).

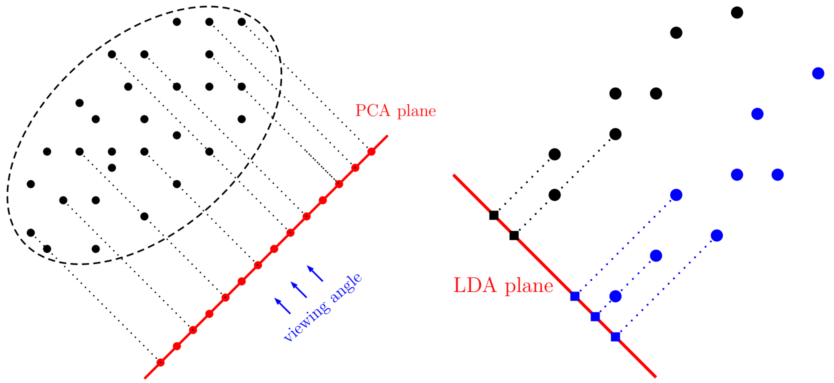
What are other statistical learning tasks?

Regression vs classification: Both are supervised (as each has a response), but in classification response is qualitative (categorical). *Example: Predicting course grade based on midterm scores and other information*.



Supervised vs unsupervised: In contrast, clustering is unsupervised (no response). The goal is to divide the data into several disjoint groups according to some similarity measure. *Example: Assigning course grades is actually a clustering task*.

Dimensionality reduction is another statistical learning task (focusing on the relationship among the variables), but it can be either unsupervised (PCA) or supervised (LDA).



Semi-supervised learning deals with a mixture of data points some of which have response values but the others do not.

Discussions

Questions to be discussed:

- What is the distinction among the three learning tasks - regression, classification and clustering?
- Can some predictors be categorical for each task?
- Which statistical learning task do you think is the most challenging? Why?

The No Free Lunch Theorem

It is important to know that "no one method is always the best over all data sets".

On a particular data set, one specific method may work best, but some other method may work better on a similar but different data set.

Selecting the best approach can be one of the most challenging parts of performing statistical learning in practice.

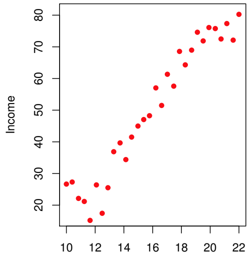
Measuring the quality of fit in regression

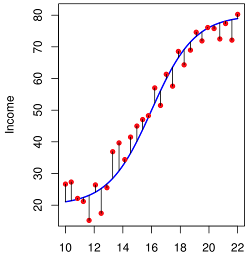
In the regression setting, the most commonly-used measure is the training MSE (mean squared error), given by

$$\mathrm{MSE}_{\mathrm{train}} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(\vec{x}_i)\right)^2$$

It is evaluated on the training data set and can be thought of as an empirical estimate of the population MSE:

$$\mathrm{E}_{\vec{X},Y}\left[(Y-\hat{Y})^2\right] = \mathrm{E}_{\vec{X}}\left[\mathrm{E}_Y(Y-\hat{Y})^2 \mid \vec{X}\right]$$





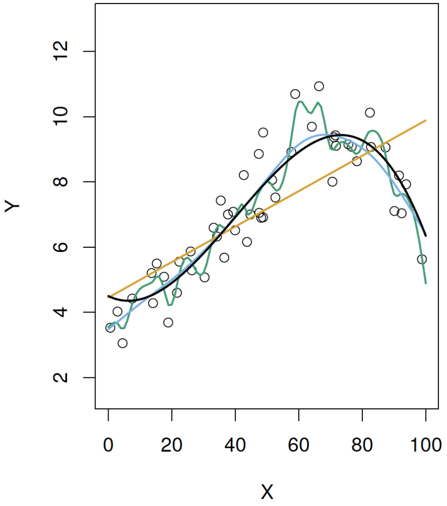
Remark.

- Our true target is the population MSE (however, it is too hard)
- The training MSE is a convenient surrogate for the population MSE (based on a specific sample)
- There is a tendency to work too hard on the training MSE. This is when overfitting happens (too focused on details and losing the big picture).

Say now we have several reasonable candidate estimates of the function *f* (e.g., the black curve in the left figure below).

Questions to think about:

- Do we know which one leads to the smallest training MSE?
- How do we know which one would lead to the smallest population MSE?
- Is it necessarily the one with the smallest training MSE?



(Data simulated from f, shown in black, as well as 3 estimates of *f*)

The test MSE in regression

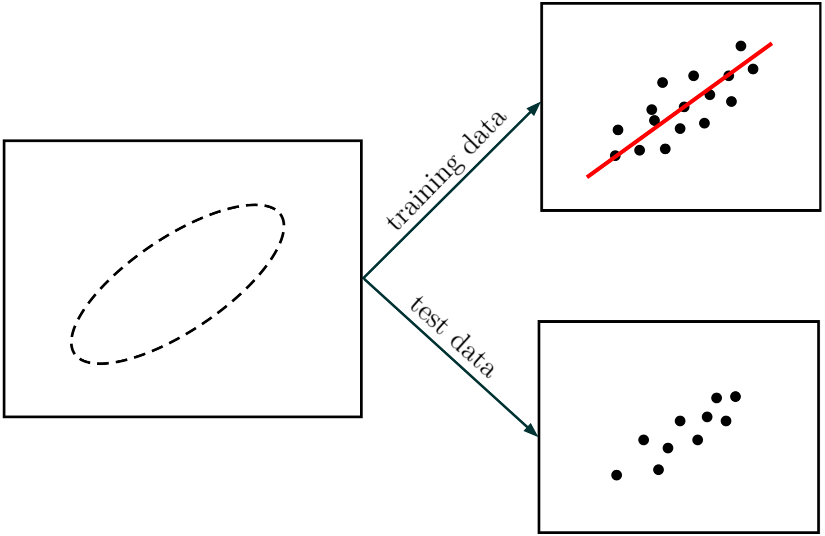
One technique is to use a second, independent (from training) sample from the same distribution, $(\vec{x}_{n+1}, y_{n+1}), \ldots, (\vec{x}_{n+m}, y_{n+m})$, called the test set, and evaluate the test MSE for each $\hat{f}$:

$$\mathrm{MSE}_{\mathrm{test}} = \frac{1}{m} \sum_{i=n+1}^{n+m} \left(y_i - \hat{f}(\vec{x}_i) \right)^2$$

This test set supposedly has the same general pattern with the training set (but likely different details). The test MSE thus provides an accurate evaluation of the population MSE for the given $\hat{f}$.

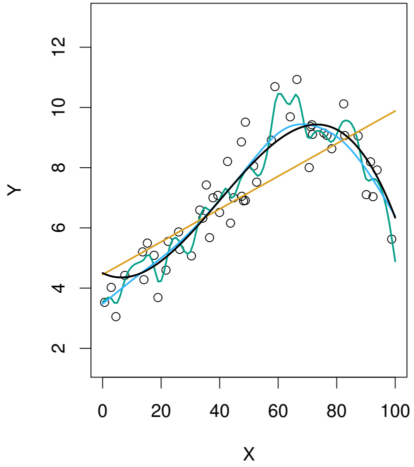
In practice test data is typically unavailable or missing the response. We will resort to methods like cross validation to select the best $\hat{f}$.

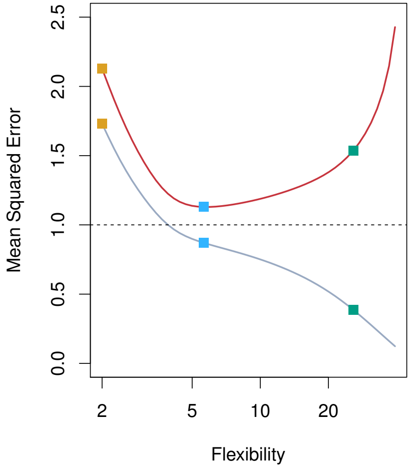
An illustration

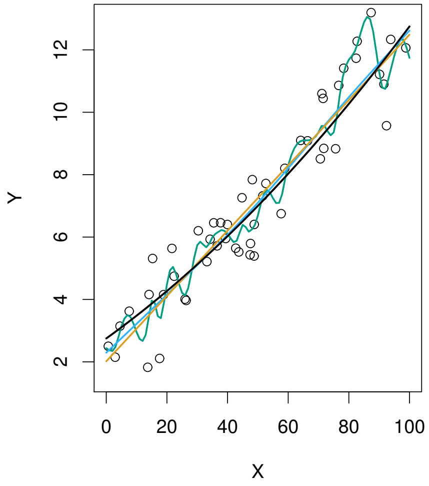


Some examples

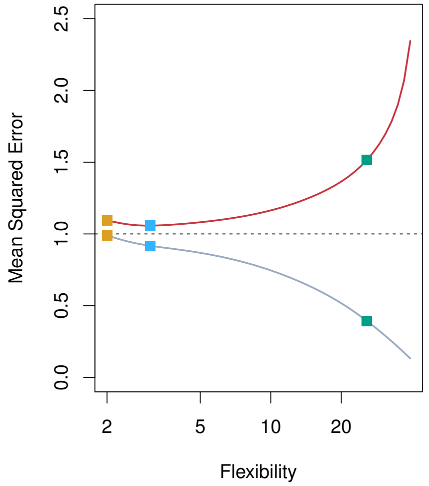
Right figure shows training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line).



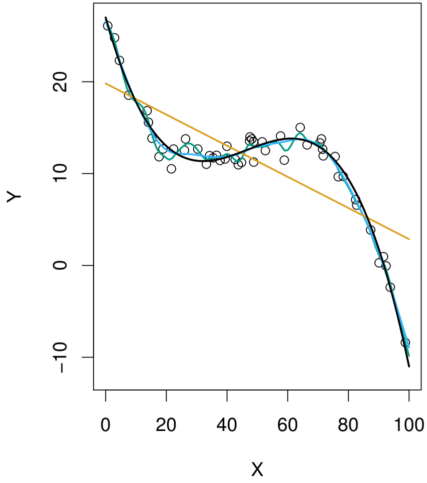


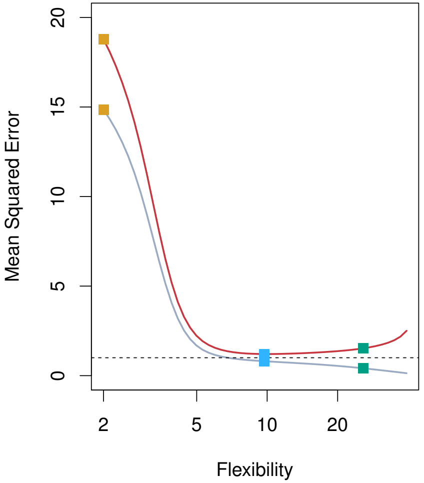


Y
X
0
20
40
60
80
100
2
4
6
8
10
12



Mean Squared Error
Flexibility
0.0
0.5
1.0
1.5
2.0
2.5
2
5
10
20


Y
X
0
20
40
60
80
100
−10
0
10
20


Mean Squared Error
Flexibility
0
5
10
15
20
2
5
10
20

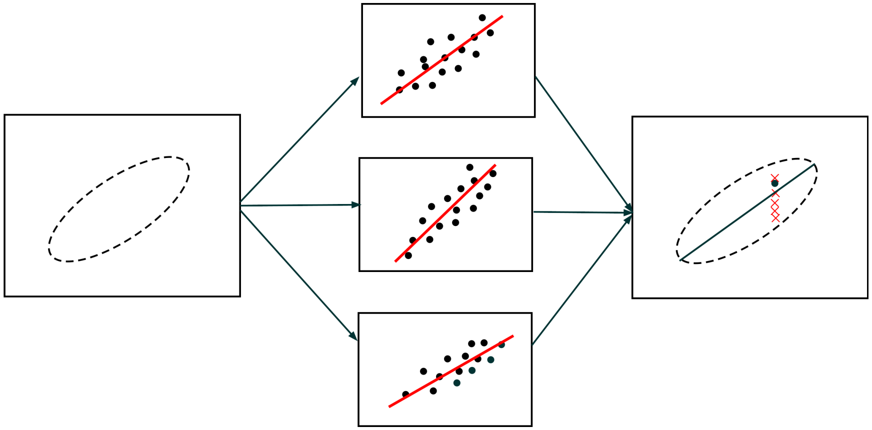
Discussions

Question

- When defining the training MSE, test MSE and population MSE, is $\hat{f}$ fixed? Are the training and test sets fixed?
- What are the distinctions and connections among the three kinds of MSE?

The expected test MSE

Suppose that we sample many training sets from the same distribution and use each of them to estimate f for predicting at a fixed location $\vec{x}_0$. What is the average test MSE at $\vec{x}_0$?



Let $y_0 = f(\vec{x}_0) + \varepsilon_0$ be the response at $\vec{x}_0$.

It can be shown that

$$\begin{aligned}
\mathrm{E}\left[(y_0 - \hat{f}(\vec{x}_0))^2\right] &= \mathrm{E}\left[(\varepsilon_0 + f(\vec{x}_0) - \hat{f}(\vec{x}_0))^2\right] \\
&= \mathrm{E}\left[(\varepsilon_0 + f(\vec{x}_0) - \mathrm{E}(\hat{f}(\vec{x}_0)) + \mathrm{E}(\hat{f}(\vec{x}_0)) - \hat{f}(\vec{x}_0))^2\right] \\
&= \underbrace{\mathrm{Var}(\epsilon_0)}_{\text{irreducible}} + \underbrace{\mathrm{Bias}(\hat{f}(\vec{x}_0))^2 + \mathrm{Var}(\hat{f}(\vec{x}_0))}_{\text{reducible error}}
\end{aligned}$$

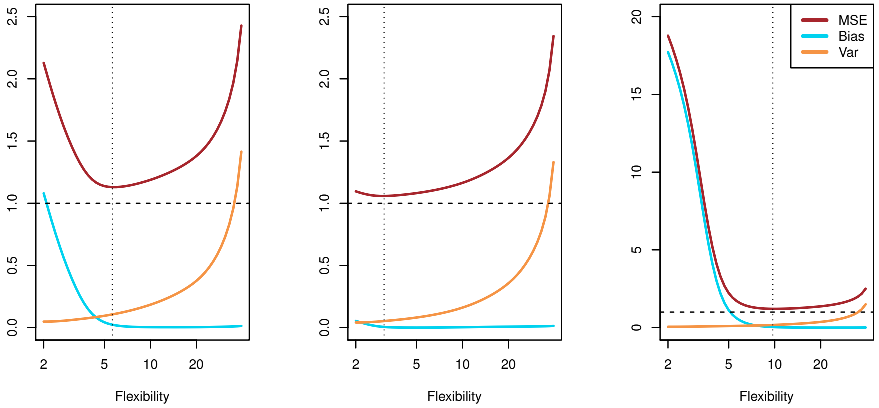
where

- $\mathrm{Bias}(\hat{f}(\vec{x}_0))^2$: On average (over all possible training sets), how much does $\hat{f}(x_0)$ differ from $f(x_0)$?
- $\mathrm{Var}(\hat{f}(\vec{x}_0))$: How much does $\hat{f}(x_0)$ vary from sample to sample?

The bias-variance tradeoff

As we use more flexible models,

- In general, the variance will increase and the bias will decrease. Think about the two extremes:
    - Very restrictive models (like linear): high bias, low variance
    - Very flexible methods (like connecting every point): high variance, low bias
- Whether the overall test MSE will increase or decrease depends on how bias and variance change together.
- A tradeoff thus needs to be made between the two in order to control them together (i.e., the reducible error).



Squared bias (blue curve), variance (orange curve), Var(ϵ) (dashed line), and test MSE (red curve) for the three data sets used earlier. The vertical dotted line indicates the flexibility level corresponding to the smallest test MSE.

Discussions

Questions to be discussed:

Consider the regression setting:

- What is the expected test MSE? And what is it averaged over?
- In the the bias-variance tradeoff, what is the bias about? Variance?
- Why do we need to consider the bias-variance tradeoff?

Measuring the quality of fit in classification

In the classification setting, responses are qualitative (categorical) and any estimate $\hat{f}$ of f is called a **classifier**.

Given training data $(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)$, the most common approach for quantifying the accuracy of a classifier $\hat{f}$ is the training error rate, defined as follows:

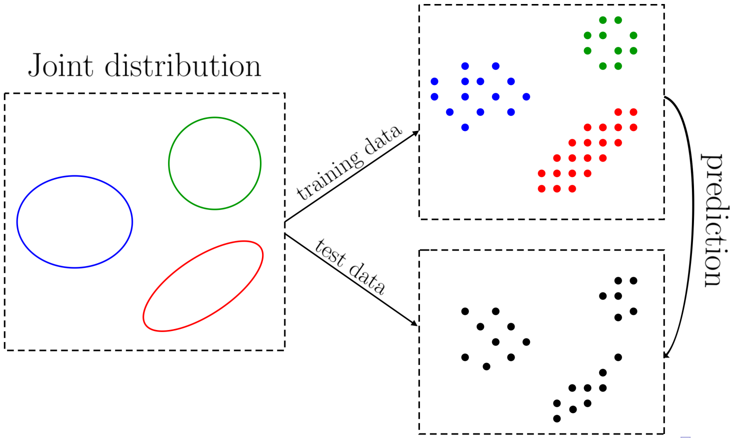
$$\frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$$

where

- $\hat{y}_i$: The predicted class label for $\vec{x}_i$ using the classifier $\hat{f}$;
- $I(y_i \neq \hat{y}_i)$: indicator function, binary valued: 0 (false), 1 (true);
- $\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$: total number of misclassified points.

As in the regression setting, the focus should be on the test error rate associated to a test data set $(\vec{x}_{n+1}, y_{n+1}), \ldots, (\vec{x}_{n+m}, y_{n+m})$:

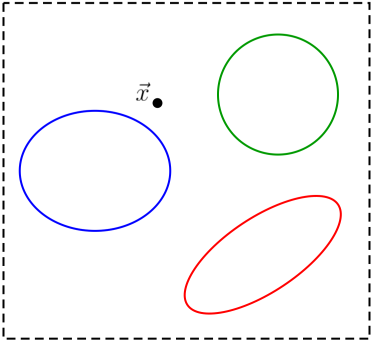
$$\frac{1}{m}\sum_{i=n+1}^{n+m} I(y_i \neq \hat{y}_i) \qquad \longrightarrow \qquad \mathrm{E}_{\vec{X}_0, y_0}(I(y_0 \neq \hat{y}_0))$$



Bayes classifiers

Assume a **joint distribution** between the predictors $\vec{X}$ (location) and the response Y (label). We can talk about the following probabilities:

- **Marginal densities**
  - $f(\vec{x})$: distribution of location
  - $P(Y = j)$: prior probabilities of labels
- **Conditional densities**
  - $f(\vec{x} \mid Y = j)$: component densities
  - $f(Y = j \mid \vec{x})$: posterior probabilities



Given a new data point x_0 (with unkown label y_0), the Bayes decision rule assigns the label based on the largest posterior probability:

$$\hat{y}_0 = \operatorname{argmax}_j P(Y_0 = j \mid x_0)$$

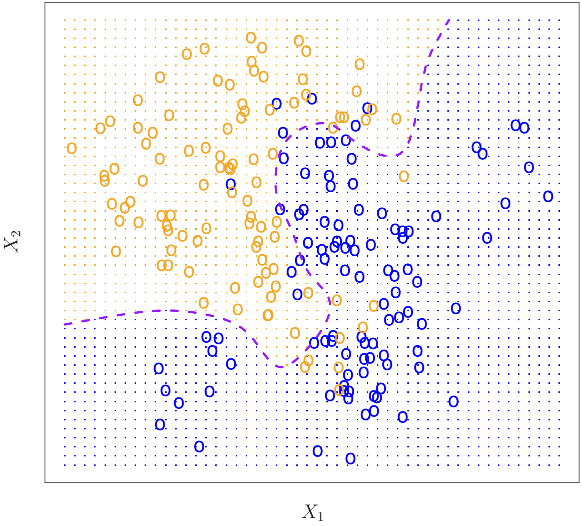
The Bayes classifier is optimal with Bayes error rate

$$1 - E_{x_0}\left(\max_j P(Y_0 = j \mid x_0)\right)$$

This is the lowest possible test error rate that may be achieved by a classifier.

However, in reality, one rarely knows the true Bayes error rate because the exact posterior probabilities are typically unknown.

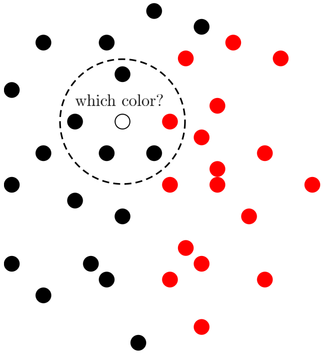
In the binary setting (two classes), the Bayes decision boundary is $\{x_0 : P(Y_0 = 1 \mid x_0) = P(Y_0 = 2 \mid x_0)\}$. In reality, it is also unknown.



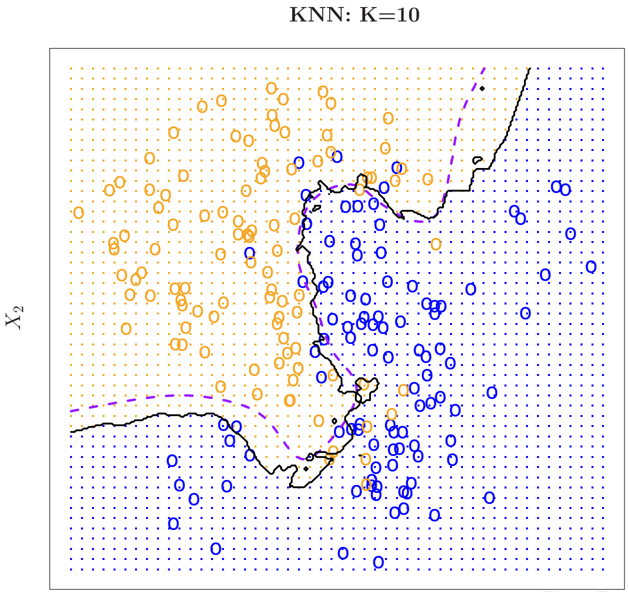
The k Nearest Neighbors (kNN) classifier

kNN assigns the label at x_0 based on a majority vote of the nearest neighbors. It is an approximate Bayes classifier:

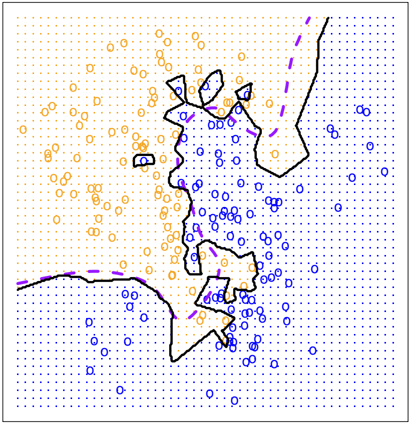
$$P(Y_0 = j \mid x_0) \approx \frac{\text{\#nearest neighbors from class } j}{\text{\#nearest neighbors examined}}$$



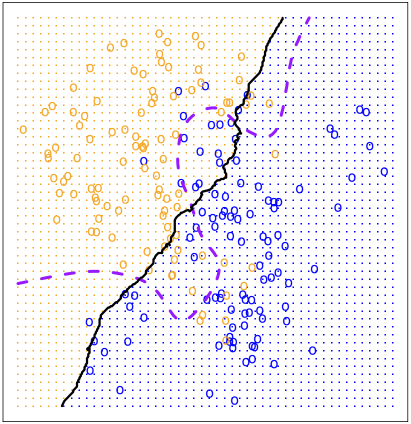
Effects of different k

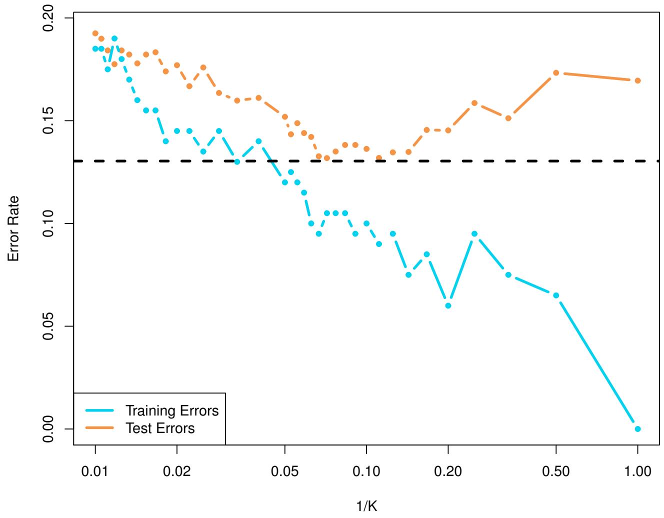


KNN: K=1



KNN: K=100





The KNN training error rate (blue, 200 observations) and test error rate (orange, 5,000 observations), as the level of flexibility (assessed using $1/K$ on the log scale) increases, or equivalently as the number of neighbors K decreases. The black dashed line indicates the Bayes error rate.

Lab: Introduction to Python