Introduction to Statistical Machine Learning with Applications in Econometrics

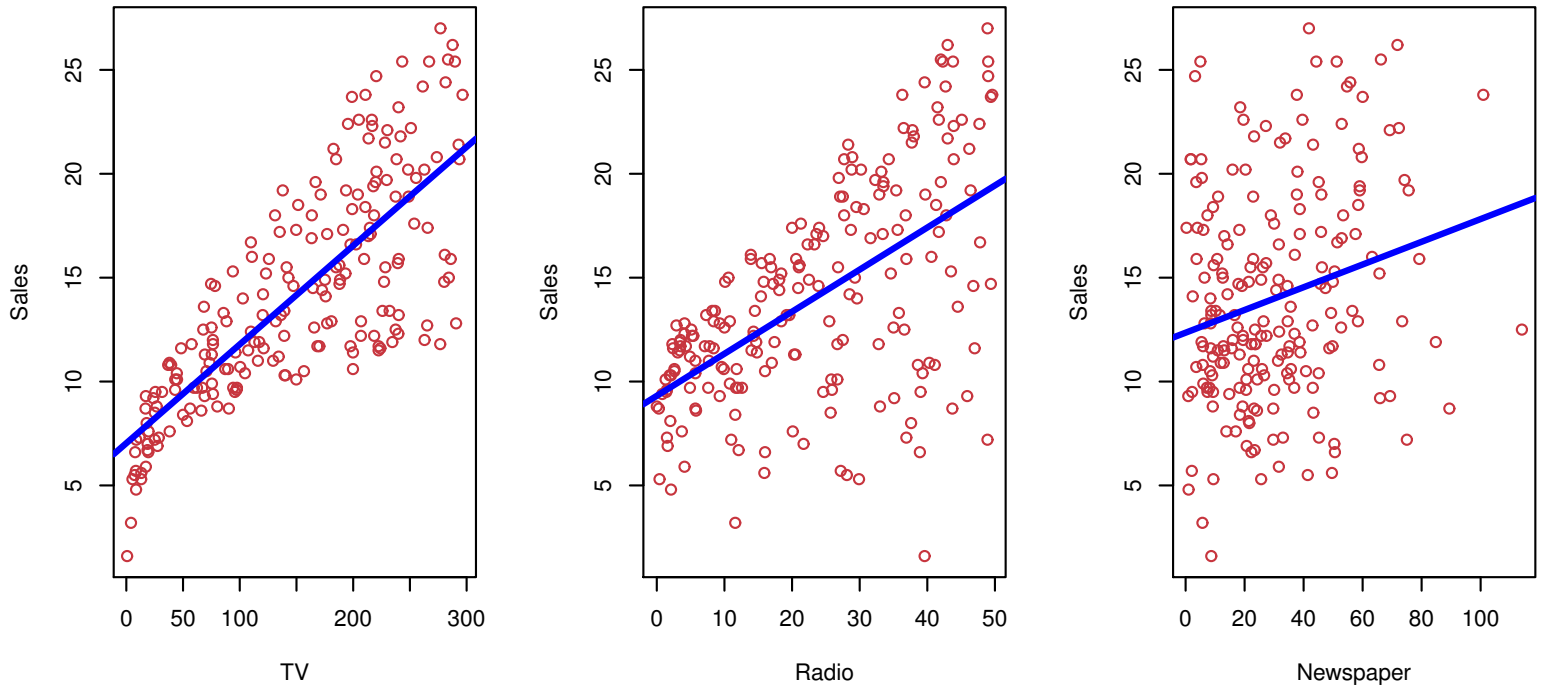
Lecture 2: Statistical Learning (ISL ch. 2)

Instructor: Ma, Jun

Renmin University of China

September 16, 2021

What is statistical learning used for?



ISL Figure 2.1

- The Advertising data set consists of the sales in 200 different markets, with advertising budgets for TV, radio and newspaper.
- Our goal: let the computer use an algorithm to predict sales on the basis of the three media budgets.
- Figure 2.1: regression of Sales against each of TV, Radio and Newspaper.

- Sales is a response or target that we wish to predict. We generically denote the response by Y.
- Each of TV, radio and newspaper is a feature, or input, or predictor. We denote the features by the input vector $X = (X_1, X_2, X_3)^\top$.

A statistical model

- More generally, suppose that we observe a quantitative response Y and p different predictors $X = \left(X_1, X_2, ..., X_p\right)^\top$.
- Assume that there is some relationship between Y and X:

$$Y = f\left(X\right) + \epsilon,$$

  where f is some fixed but unknown function of X, and ϵ is a random error term, which is independent of X and $\mathrm{E}\left[\epsilon\right] = 0$.
- If the model is viewed as a structural model, ϵ is viewed as (aggregation of) unobserved factors that generate Y.
  - As an example, X are characteristics of a patient's blood sample; Y is the patient's risk for a severe adverse reaction to a particular drug; ϵ can be manufacturing variation in the drug or the patient's general feeling of well-being.
  - Caution: this model assumes no endogeneity, which may not be true, from an econometric perspective.

- Alternatively, the model can be viewed as a prediction model. Taking $f(X) = \mathrm{E}[Y \mid X]$ and $\epsilon = Y - f(X)$, it automatically holds that $\mathrm{E}[\epsilon \mid X] = 0$. The additional model assumption is that ϵ and X are independent.
- This prediction modelling approach suffices if our objective is out-of-sample prediction. But it does not lead to causality interpretation.
- Mathematically, in either case, $f(x) = \mathrm{E}[Y|X = x]$.
- Suppose that there is an unseen data point (X_0, Y_0) with $Y_0 = f(X_0) + \epsilon_0$ for some unseen error ϵ_0, which is a random draw from the distribution of ϵ. We do not see Y_0 (or ϵ_0) and wish to let the computer predict what Y_0 should be, after it receives X_0.

Recap of conditional expectation

- Conditional PDF (when (Y, X) are continuous): $f_{Y|X=x}(y \mid x) = f_{X,Y}(x, y) / f_X(x)$. If X and Y are independent, $f_{Y|X}(y \mid x) = f_Y(y)$ for all x.
- Suppose you know that $X = x$. You can update your expectation of Y by conditional expectation. We define conditional expectationfrom conditional PDF: $\mathrm{E}[Y \mid X = x] = \int y f_{Y|X}(y \mid x)\, \mathrm{d}y$. $\mathrm{E}[Y \mid X = x]$ is a constant.
- A conditional expectation $\mathrm{E}[Y \mid X = x]$ is a number not a random variable. $\mathrm{E}[Y \mid X = x]$ is not random, not a function of Y. It is a function of the observed "realized" value x of the random variable X.
- We denote this function by $g(x) = \mathrm{E}[Y \mid X = x]$. Notice that g is an ordinary function of x, which is just a number.
- $g(X)$ is a random variable. If denoting $\mathrm{E}[Y \mid X] = g(X)$, $\mathrm{E}[Y \mid X]$ is a random variable and a function of X (Uncertainty about X has not been realized yet).

- Conditional expectations satisfies all properties of unconditional expectation. E.g.

$$\mathrm{E}\left[Y+Z \mid X\right]=\mathrm{E}\left[Y \mid X\right]+\mathrm{E}\left[Z \mid X\right].$$

- Once you condition on X, you can treat any function of X as a constant:

$$\mathrm{E}\left[h_1\left(X\right)+h_2\left(X\right)Y \mid X\right]=h_1\left(X\right)+h_2\left(X\right)\mathrm{E}\left[Y \mid X\right],$$

  for any functions h_1 and h_2.
- Law of Iterated Expectation (LIE):

$$\begin{aligned}\mathrm{E}\left[\mathrm{E}\left[Y \mid X\right]\right] &= \mathrm{E}\left[Y\right],\\ \mathrm{E}\left[\mathrm{E}\left[Y \mid X,Z\right] \mid X\right] &= \mathrm{E}\left[Y \mid X\right]\\ \mathrm{E}\left[\mathrm{E}\left[Y \mid X\right] \mid X,Z\right] &= \mathrm{E}\left[Y \mid X\right].\end{aligned}$$

- Mean independence: Y and X are mean independent if

$$\mathrm{E}\left[Y \mid X\right]=\mathrm{E}\left[Y\right]=\text{constant}.$$

$$
\begin{array}{c}
X \text{ and } Y \text{ are independent} \\
\Downarrow \\
\mathrm{E}\left[Y \mid X\right] = \text{constant (mean independence)} \\
\Downarrow \\
\mathrm{Cov}\left[X, Y\right] = 0 \text{ (uncorrelatedness)}
\end{array}
$$

Why estimate f?

- $f(X_0)$ is an ideal or optimal predictor of Y_0.
- $f(X_0)$ minimizes the mean square prediction error

$$\mathrm{E}\left[(Y_0 - f(X_0))^2 \mid X_0\right] \leq \mathrm{E}\left[(Y_0 - g(X_0))^2 \mid X_0\right].$$

- $\epsilon_0 = Y_0 - f(X_0)$ is the irreducible error. Even if we knew $f(X_0)$, we would still make errors in prediction, since when X_0 is given there is still a distribution of possible Y_0 values.
- We predict Y_0 by using $\hat{Y}_0 = \hat{f}(X_0)$. Assume for now that $\hat{f}(X_0)$ is non-random given X_0, then,

$$\begin{aligned}\mathrm{E}\left[\left(Y_0 - \hat{Y}_0\right)^2 \mid X_0\right] =& \mathrm{E}\left[\left\{f(X_0) + \epsilon_0 - \hat{f}(X_0)\right\}^2 \mid X_0\right] \\ =& \underbrace{\left[f(X_0) - \hat{f}(X_0)\right]^2}_{\text{Reducible}} + \underbrace{\mathrm{Var}\left[\epsilon_0\right]}_{\text{Irreducible}}.\end{aligned}$$

- Our focus is on minimizing the reducible error.

- There are situations when we are interested in the form of f ("inference problem" in ISL) rather than estimating $f(X_0)$ given X_0 for prediction (e.g., linearity, partial derivatives...).
  - In the context of causal inference ($Y = f(X) + \epsilon$ is interpreted as the structural model), partial derivatives of f are interpreted as causal effects.
  - We are interested in identifying the subset of X that has nonzero partial derivatives and also signs of them (positive or negative effects?).
  - Is it adequate to specify a linear model, i.e., assuming $f(X) = X^\top \beta$ for some β, or this specification assumption is false?
- In a real estate setting, Y: values of homes; X: crime rate, zoning, distance from a river, air quality, schools, income level of community, size of houses, and so forth.
  - How much extra will a house be worth if it has a view of the river? This is an inference problem.
  - Predicting the value of a house newly put on the market given its characteristics. This is a prediction problem.

How do we estimate f?

- Observed a set of n data points.
- $X_{j,i}$: the value of the j-th predictor, or input, for observation i, where $i = 1, 2, ..., n$ and $j = 1, 2, ..., p$.
- Y_i: the response variable for the i-th observation.
- Our training data consist of $\{(X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n)\}$, where $X_i = \left(X_{1,i}, X_{2,i}, ..., X_{p,i}\right)^\top$.
- Each Y_i is generated by the model: $Y_i = f(X_i) + \epsilon_i$.
- Our goal is to apply a statistical learning method to the training data to estimate the unknown function f.
- An unseen data point: (X_0, Y_0), where X_0 is observed (received by the computer) but Y_0 is not.
- Predict Y_0 by $\hat{Y}_0 = \hat{f}(X_0)$, where $\hat{f}(X_0)$ depends on the training data.

Parametric approach: prediction perspective

- Linear model (as an example of parametric models):

$$f(X_i) = \beta_0 + \beta_1 X_{1,i} + \beta_2 X_{2,i} + \ldots + \beta_p X_{p,i}.$$

  Instead of having to estimate an entirely arbitrary p-dimensional function f, one only needs to estimate the $p + 1$ coefficients $\beta_1, \beta_2, ..., \beta_p$.
- After a model has been selected, we need a procedure that uses the training data to fit or train the model. In the case of the linear model, we want to find values of these parameters such that

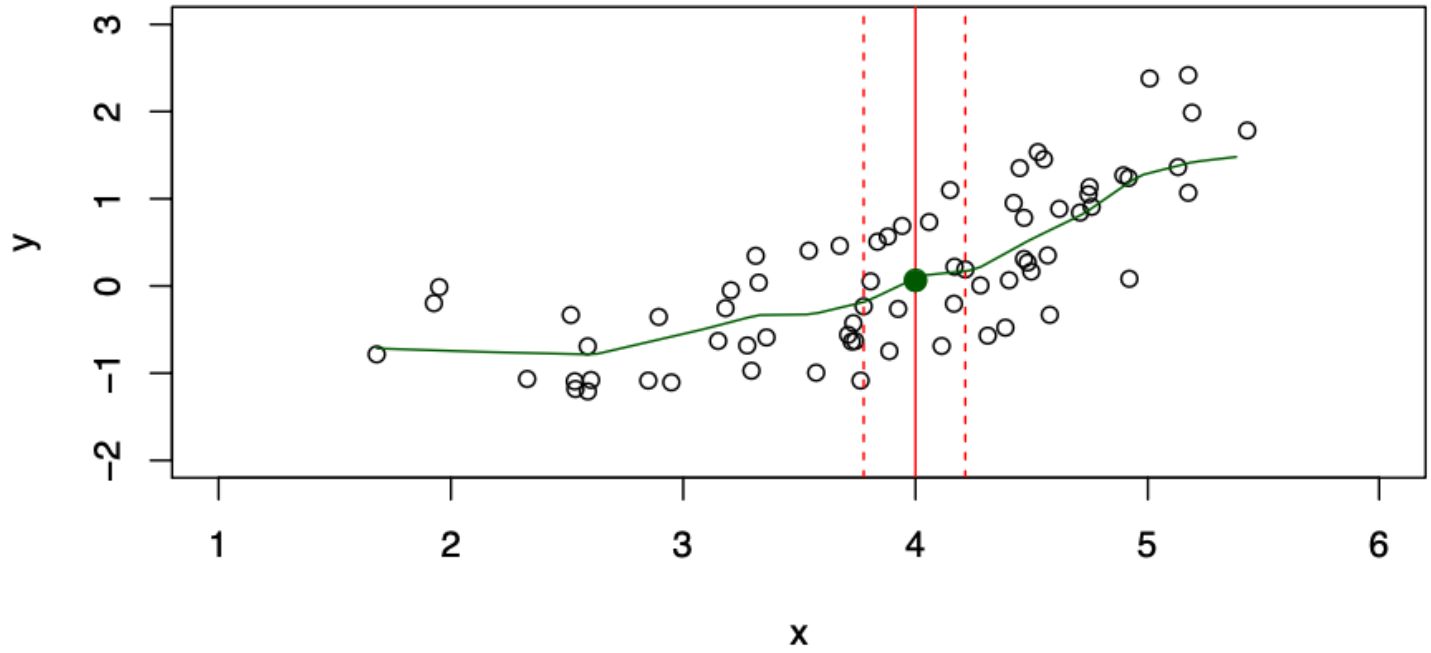
$$Y_i \approx \beta_0 + \beta_1 X_{1,i} + \beta_2 X_{2,i} + \ldots + \beta_p X_{p,i}.$$

  The most common approach is OLS.

- For prediction, we care about the expected test mean square (prediction) error MSE $= \mathrm{E}\left[\left(Y_0 - \hat{f}\left(X_0\right)\right)^2\right]$ and hope to make it as small as possible.
- Later we show that MSE $= \text{Bias}^2 + \text{Variance} + \text{Noise}$, where Noise $= \text{Var}\left[\epsilon\right]$.
- The potential disadvantage of a parametric approach is that the model we choose will usually not match the true unknown form of f.
- If the chosen model is too far from the true f, then our estimate will be poor, since the absolute bias is large.
- We can try to address this problem by choosing flexible models that can fit many different possible functional forms for f. But fitting a more flexible model can lead to overfitting, since the variance is large.

Non-parametric approach: prediction perspective

- Non-parametric methods do not make explicit assumptions about the functional form of $f(x) = \mathrm{E}[Y \mid X = x]$.
- Without functional form assumption for f, they have the potential to accurately fit a wider range of possible shapes for f (i.e., low bias).
- A very large number of observations (far more than what is typically needed for a parametric approach) is required to get an accurate estimate for f.
- A large class of old-generational nonparametric regression methods for estimating $f(x) = \mathrm{E}[Y \mid X = x]$ is based on local averaging.



- Suppose we wish to estimate $\mathrm{E}[Y \mid X = 4]$. No data point with $X = 4$ exactly.
- Let $\mathcal{N}(x)$ be some neighborhood of x,

$$\hat{f}(x) = \frac{\sum_{i=1}^{n} Y_i 1(X_i \in \mathcal{N}(x))}{\sum_{i=1}^{n} 1(X_i \in \mathcal{N}(x))},$$

where

$$1(X_i \in \mathcal{N}(x)) = \begin{cases} 1 & \text{if } X_i \in \mathcal{N}(x) \\ 0 & \text{if } X_i \notin \mathcal{N}(x) \end{cases}.$$

- The neighborhood $\mathcal{N}(x)$ is usually selected in a data-driven manner.
- The performance of local averaging non-parametric method deteriorates substantially as p increases. This is known as "curse of dimensionality".
- You need a incredibly huge n to get a reasonably accurate estimate. When p is large, to get a reasonably large "effective sample size" $\sum_{i=1}^{n} 1(X_i \in \mathcal{N}(x))$, n should be an incredibly huge number.
- Local averaging works well only when p is small ($p \leq 4$).

Parametric vs non-parametric: inference perspective

- Linear models are easy to interpret: marginal/causal effects are just regression coefficients.
- Linear models can be misspecified: the true relationship may not be linear. In this case, interpreting the regression coefficients as causal effects is not correctly.
- Nonparametric models are robust to misspecification. However, the regression results are hard to interpret.

Assessing model accuracy

- No free lunch in statistics: no one method dominates all others over all possible data sets.
- How to decide for any given set of data which method produces the best results?
- We could compute and compare the average training mean square (prediction) error using the training data:

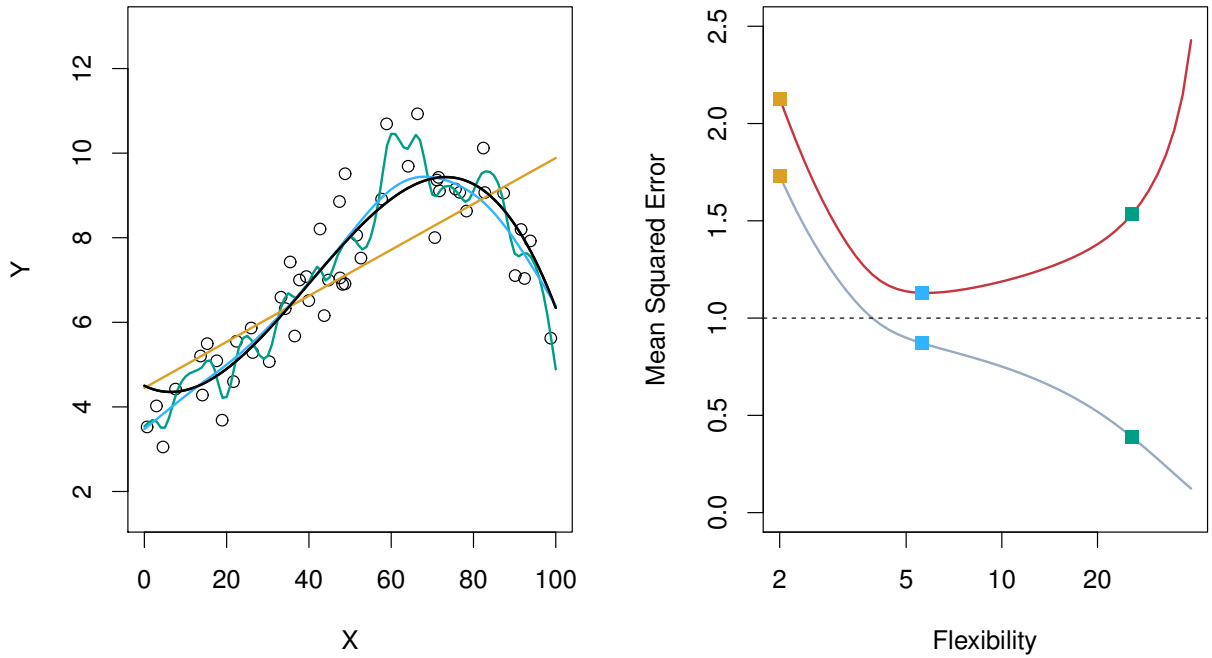
$$\mathrm{MSE}_{\mathsf{Tr}} = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{f}\left(X_i\right)\right)^2.$$

- For comparison of different models, $\mathrm{MSE}_{\mathsf{Tr}}$ is biased towards more flexible model.

- We do not care how well the method works on the training data. We are interested in its prediction accuracy when applied to previously unseen test data.
  - As an example, suppose that we have clinical measurements for a number of patients, as well as information about whether each patient has diabetes. We wish to accurately predict diabetes risk for future patients based on their clinical measurements. We are not interested in whether or not the method accurately predicts diabetes risk for patients used to train the model, since we already know which of those patients have diabetes.
- Suppose we have a large number of test data: $(Y_{n+1}, X_{n+1}), ..., (Y_{n+m}, X_{n+m})$. We compute the average test mean square (prediction) error

$$\text{MSE}_{\text{Te}} = \frac{1}{m} \sum_{i=1}^{m} \left(Y_{n+i} - \hat{f}\left(X_{n+i} \right) \right)^2 .$$

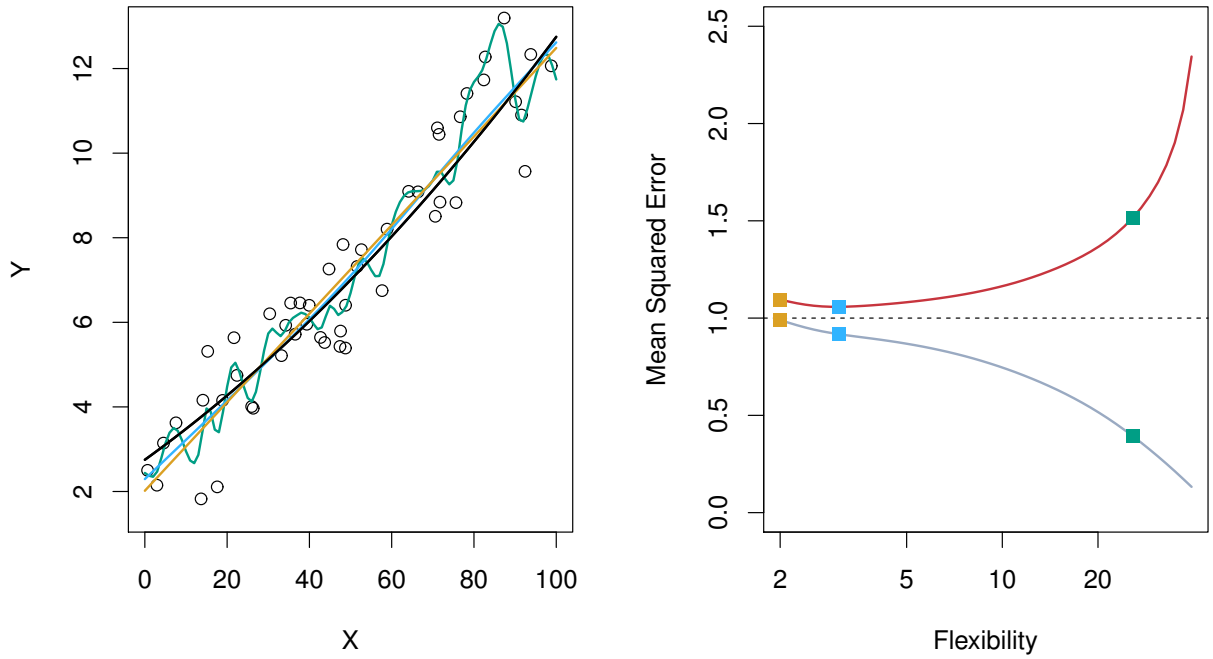
- In practice, we can use part of our data to train the model and use the rest as test data to compare different models.
- A more effective method is cross-validation.



ISL Figure 2.9

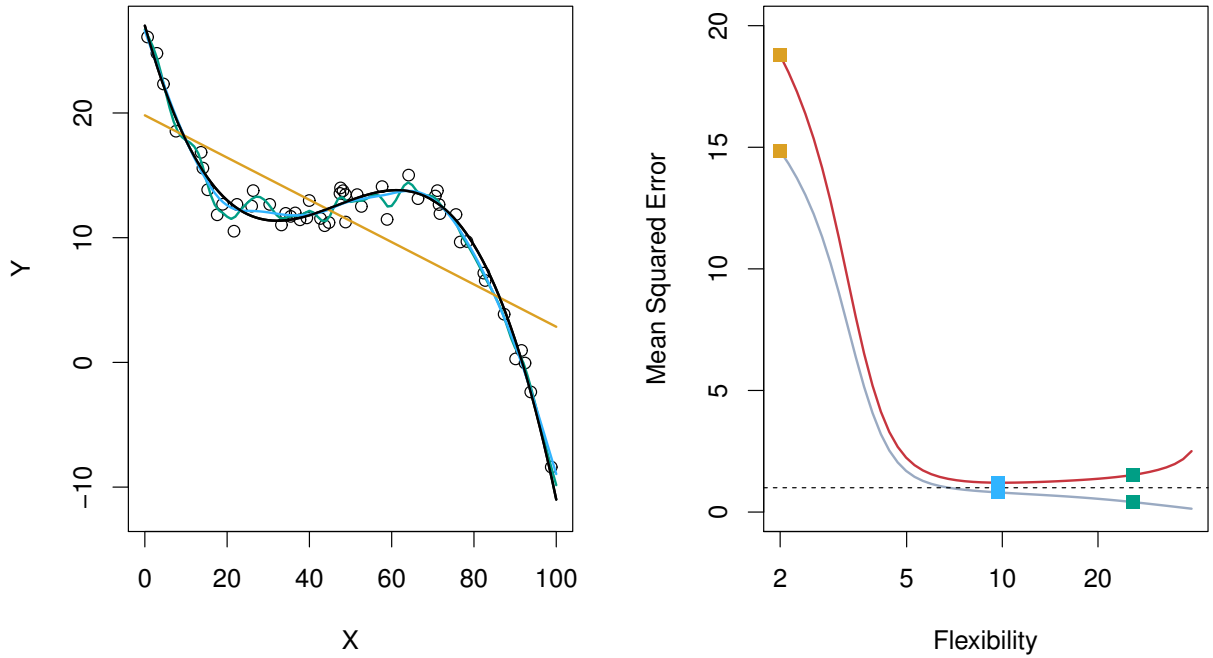
- Figure 2.9: simulated data.
- Left: true f (black), linear regression (orange curve) and two nonparametric regression curves (blue and green curves).
- Right: Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line).
- Squares represent the training and test MSEs for the three fits shown in the left-hand panel.

- As model flexibility increases, training MSE will decrease, but the test MSE may not.
- When a given method yields a small training MSE but a large test MSE, we are said to be overfitting the data.
- Overfitting happens because our procedure is working too hard to find patterns in the training data, and may be picking up some patterns that are just caused by random noise rather than the signal $f\left(X_i\right)$.



ISL Figure 2.10

- f is smoother. Variation of the data along the Y-axis as X increases is driven by random noise.
- The most flexible model (green curve) overfits the data by picking up patterns driven by the noise.
- An unseen data point is generated by a new error term, which is independent from those error terms that generate the training data.



ISL Figure 2.11

- f is wiggly and the noise has low variation. Variation of the data along the Y-axis is driven by change in the signal $f(X_i)$.
- The more flexible fits work the best.

Bias-variance trade-off

- Suppose we have fit a model and get $\hat{f}(x)$ with our training data. Let (X_0, Y_0) be a test observation that is drawn from the population and independent from the training data.
- The conditional expected test MSE can be decomposed into

$$\mathrm{E}\left[\left(Y_0 - \hat{f}(X_0)\right)^2 \mid X_0\right] = \mathrm{Var}[\epsilon] + \mathrm{Bias}(X_0)^2 + \mathrm{Variance}(X_0)$$

  where $\mathrm{Bias}(X_0) = \mathrm{E}\left[\hat{f}(X_0) \mid X_0\right] - f(X_0)$ and $\mathrm{Variance}(X_0) = \mathrm{Var}\left[\hat{f}(X_0) \mid X_0\right]$.
- The unconditional expected test MSE is just

$$\mathrm{E}\left[\left(Y_0 - \hat{f}(X_0)\right)^2\right] = \mathrm{E}\left[\mathrm{E}\left[\left(Y_0 - \hat{f}(X_0)\right)^2 \mid X_0\right]\right],$$

  by law of iterated expectations.
- The expected test MSE can never lie below $\mathrm{Var}[\epsilon]$, the irreducible error.

- Typically as the flexibility increases, variance increases and bias decreases.
- Variance refers to the amount by which $\hat{f}$ would change if we estimated it using a different training data set.
- Bias refers to the error that is introduced by approximating the unknown f, which may be extremely complicated, by a much simpler model.
- To minimize the expected test MSE, we need to select a statistical learning method that simultaneously achieves low variance and low bias.
- Choosing the flexibility based on average test MSE amounts to a bias-variance trade-off.

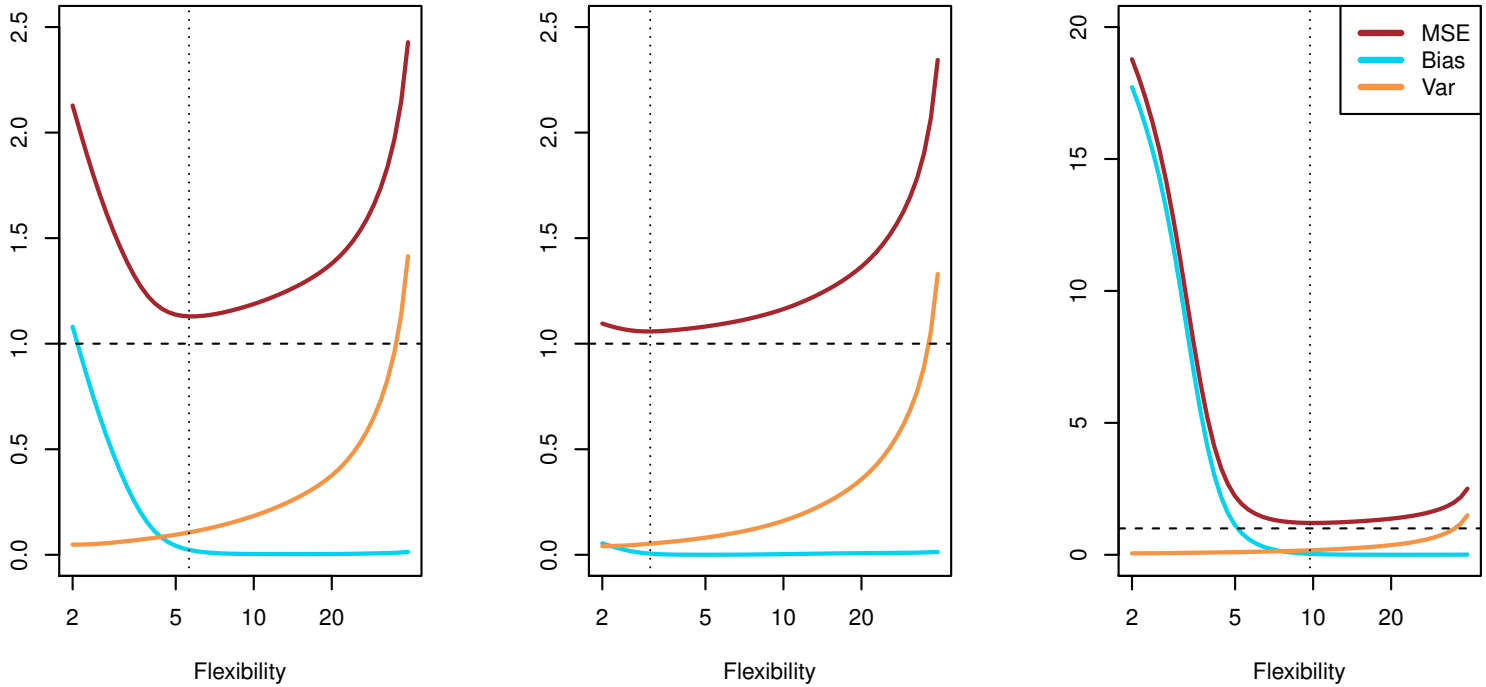


Figure 2.12

- Figure 2.12: bias-variance trade-off for the three examples.
- The flexibility level corresponding to the optimal expected test MSE differs considerably among the three data sets.

Classification problems

- The response variable Y is qualitative .
  - Email is one of $\mathscr{C} = \{\text{spam}, \text{ham}\}$, in a spam email detection problem.
  - Digit class is one of $\mathscr{C} = \{0, 1, ..., 9\}$, in a handwritten image recognition problem.
- (X_0, Y_0) is an unseen data point, X_0 is received by the computer and we let the computer to predict what Y_0 should be.
- The computer uses the training data to build a classifier $\hat{f}(X_0) \in \mathscr{C}$ that assigns a label from $\mathscr{C}$ as an estimate of Y_0.
- The training error rate (the fraction of incorrect classifications):

$$\frac{1}{n} \sum_{i=1}^{n} 1\left(Y_i \neq \hat{Y}_i\right),$$

where $\hat{Y}_i = \hat{f}(X_i)$.

- With a test data: $(Y_{n+1}, X_{n+1}), ..., (Y_{n+m}, X_{n+m})$, we evaluate the performance of the classifier $\hat{f}$ using the test misclassification error rate

$$\frac{1}{m}\sum_{i=1}^{m} 1\left(Y_{n+i} \neq \hat{Y}_{n+i}\right),$$

  where $\hat{Y}_{n+i} = \hat{f}(X_{n+i})$.
- As in the regression setting, we are most interested in the error rates that result from applying our classifier to test observations that were not used in training.
- The (conditional) expected test error rate: $\Pr\left[Y_0 \neq \hat{Y}_0 \mid X_0\right]$, where $\hat{Y}_0 = \hat{f}(X_0)$, and the unconditional version is $\mathrm{E}\left[\Pr\left[Y_0 \neq \hat{Y}_0 \mid X_0\right]\right]$.
- A good classifier is one for which the test error rate is smallest.
- Suppose $\mathscr{C}$ are labelled: $\mathscr{C} = \{1, 2, ..., K\}$. Let $p_k(X) = \Pr[Y = k \mid X]$, $k = 1, 2, ..., K$.
- The Bayes optimal classifier:

$$C(X) = \mathrm{argmax}_j\, p_j(X).$$

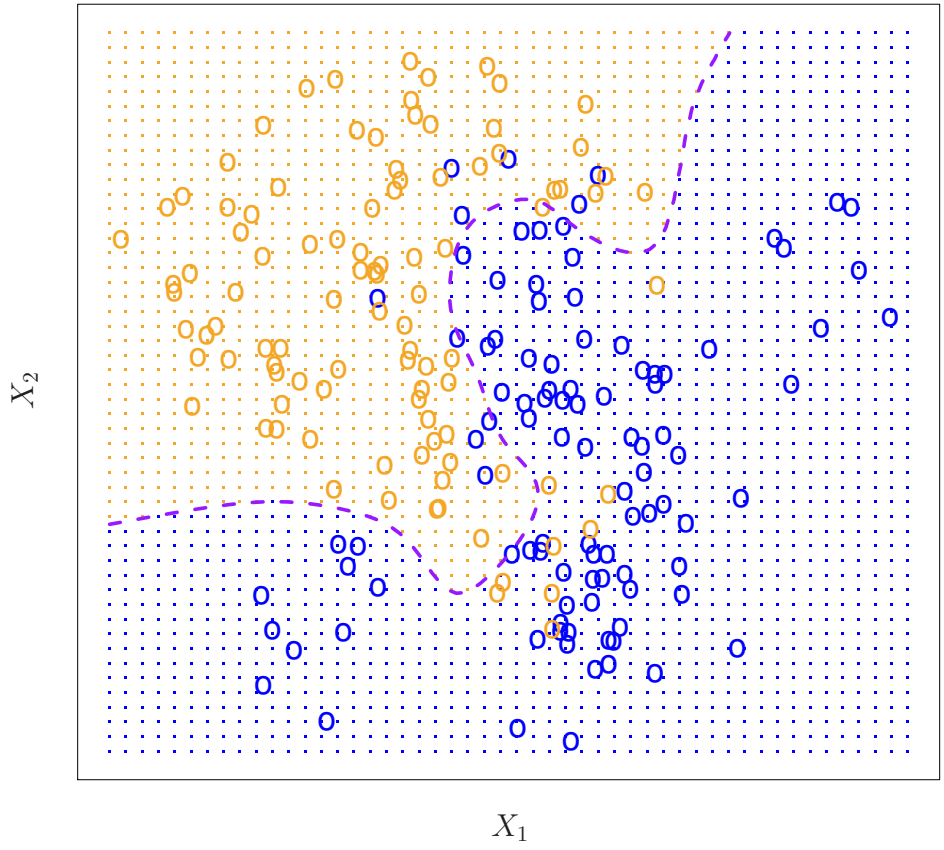


Figure 2.13

- The purple dashed line represents the Bayes decision boundary.
- The orange background grid indicates the region in which a test observation will be assigned to the orange class, and the blue background grid indicates the region in which a test observation will be assigned to the blue class.

- The Bayes classifier produces the lowest possible test error rate, called the Bayes error rate:

$$\begin{aligned}\Pr\left[Y \neq C\left(X\right) \mid X\right] &= \Pr\left[Y \neq \operatorname{argmax}_j p_j\left(X\right) \mid X\right] \\ &= 1 - \max_{j=1,\ldots,K} p_j\left(X\right).\end{aligned}$$

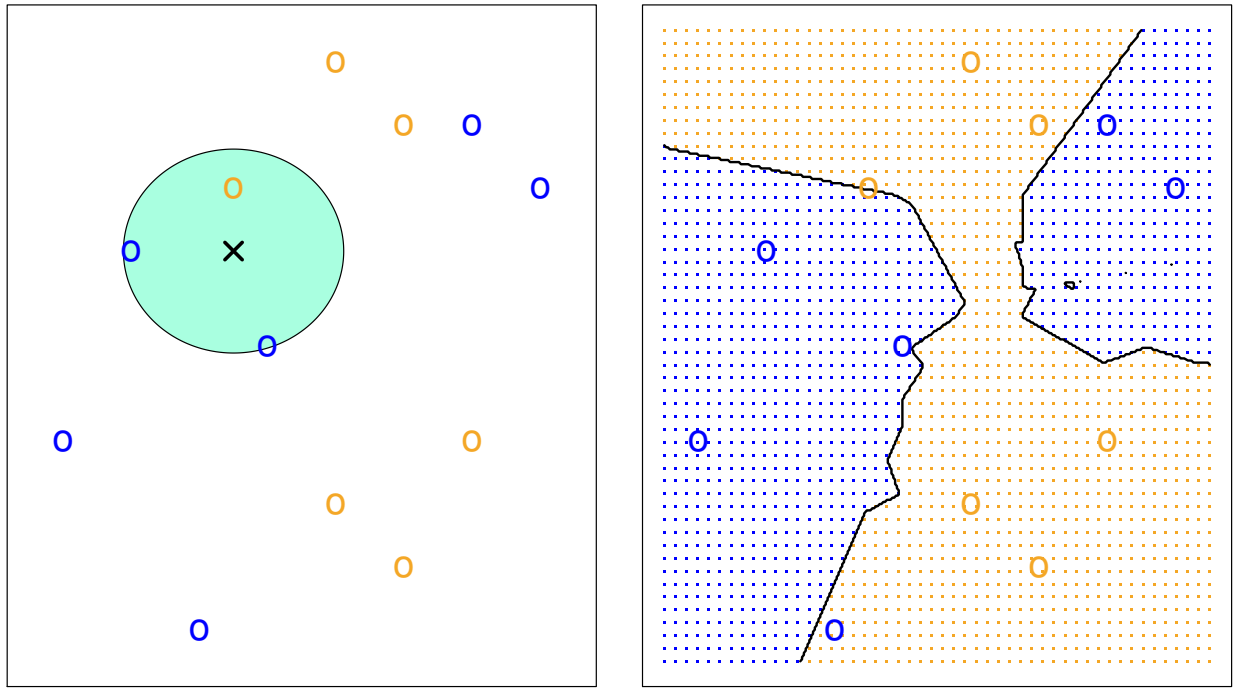
- The overall Bayes error rate $\Pr\left[Y \neq C\left(X\right)\right] = 1 - \mathrm{E}\left[\max_{j=1,\ldots,K} p_j\left(X\right)\right]$ is analogous to the irreducible error in the regression context.

KNN Classifier

- For real data, we do not know the conditional distribution of Y given X, and so computing the Bayes classifier is impossible.
- Many approaches attempt to estimate the conditional distribution of Y given X, and then classify a given observation to the class with highest estimated probability.
- One such method is the K-nearest neighbors (KNN) classifier. It uses the KNN nonparametric estimator of $p_j(x)$:

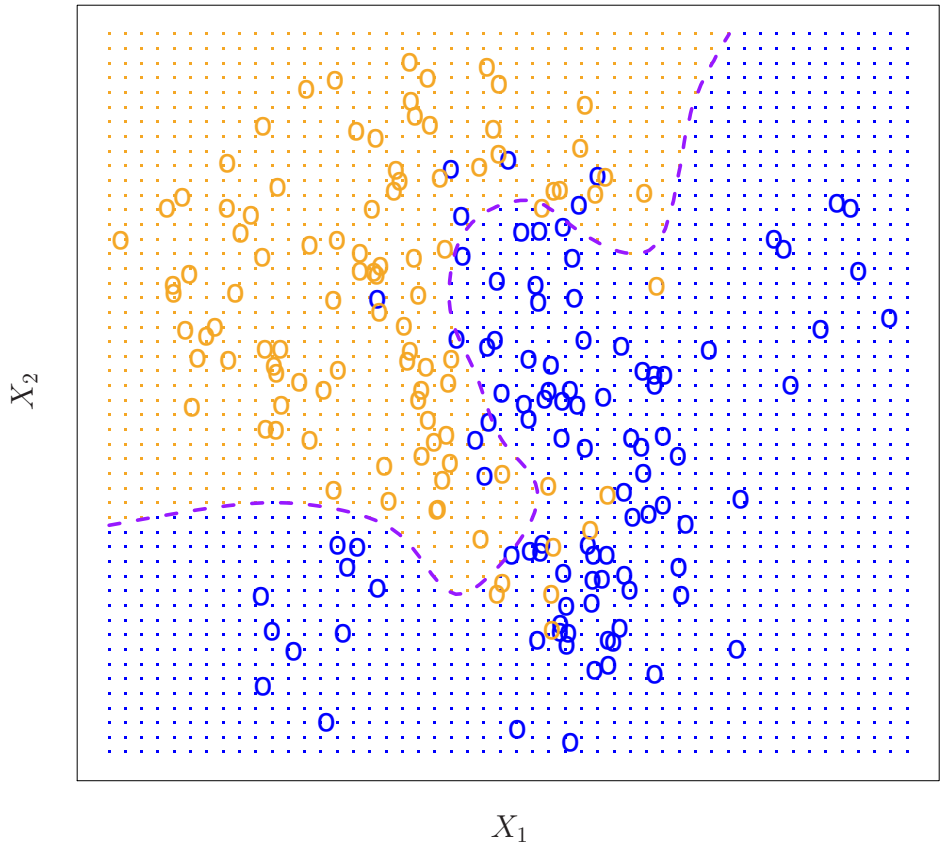
$$\widehat{p}_j(x) = \frac{1}{K} \sum_{i \in \mathcal{N}_K(x)} 1(Y_i = j),$$

  where $\mathcal{N}_K(x)$ denotes the K points in the training data that are closest to x.
- KNN classifies the test observation X_0 to the class with the largest probability from $\left\{\widehat{p}_j(X_0) : j = 1, ..., K\right\}$.



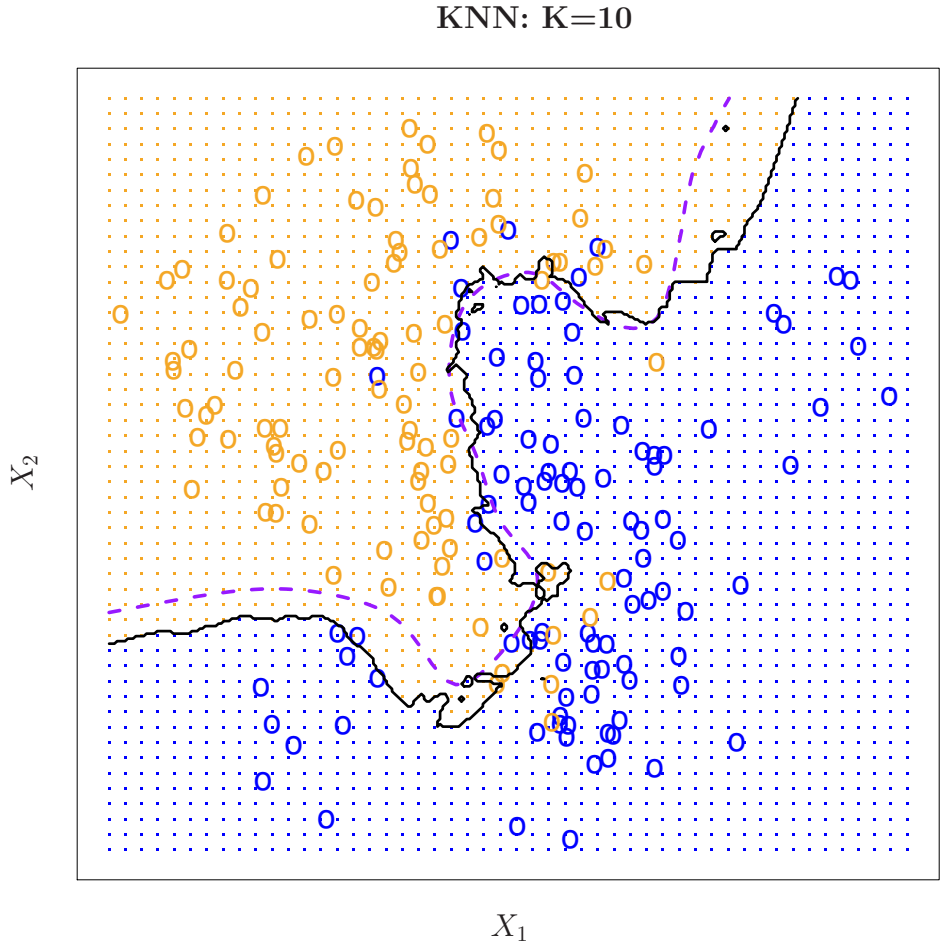
ISL Figure 2.14

- KNN with $K = 3$.
- Left: a test observation at which a predicted class label is desired is shown as a black cross.
- Right: The KNN decision boundary for this example is shown in black.



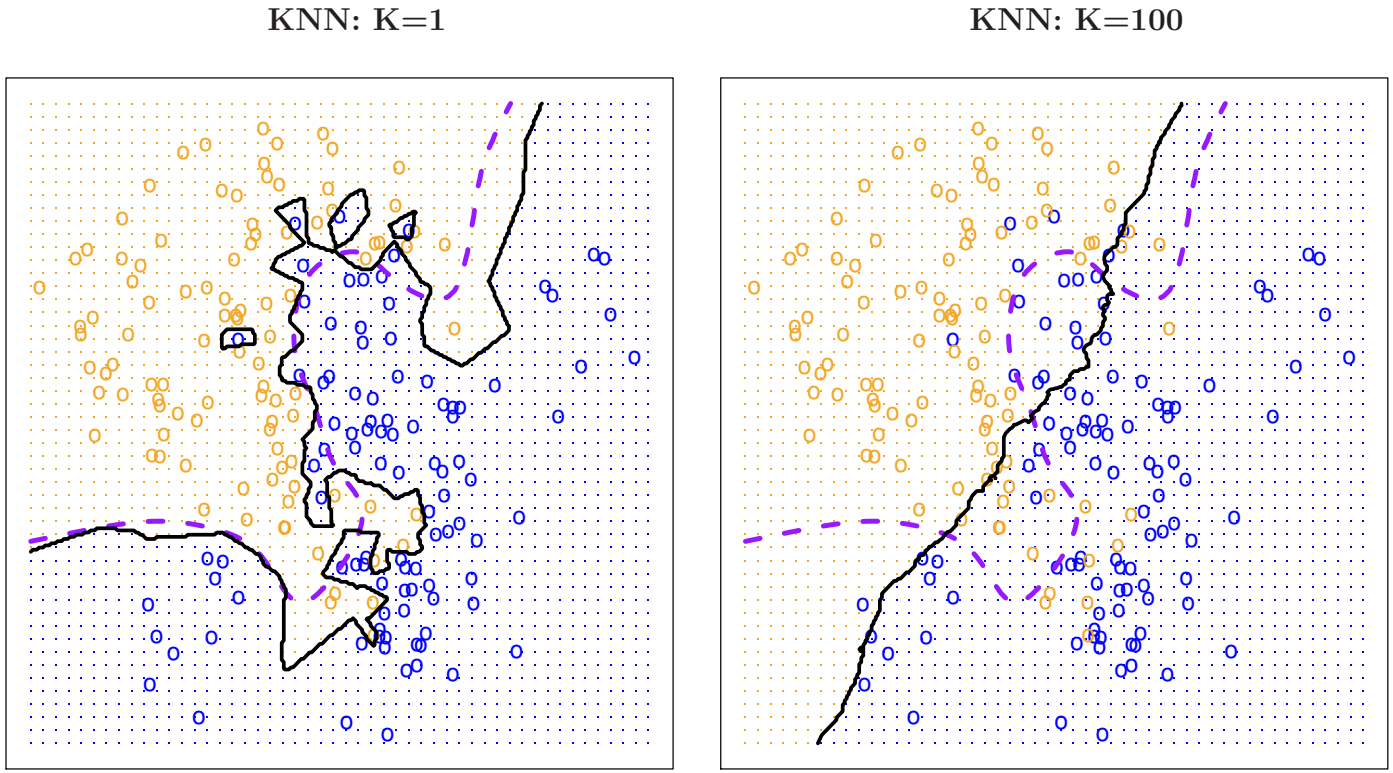
ISL Figure 2.13

- A simulated data set consisting of 100 observations in each of two groups, indicated in blue and in orange.
- The purple dashed line represents the Bayes decision boundary.



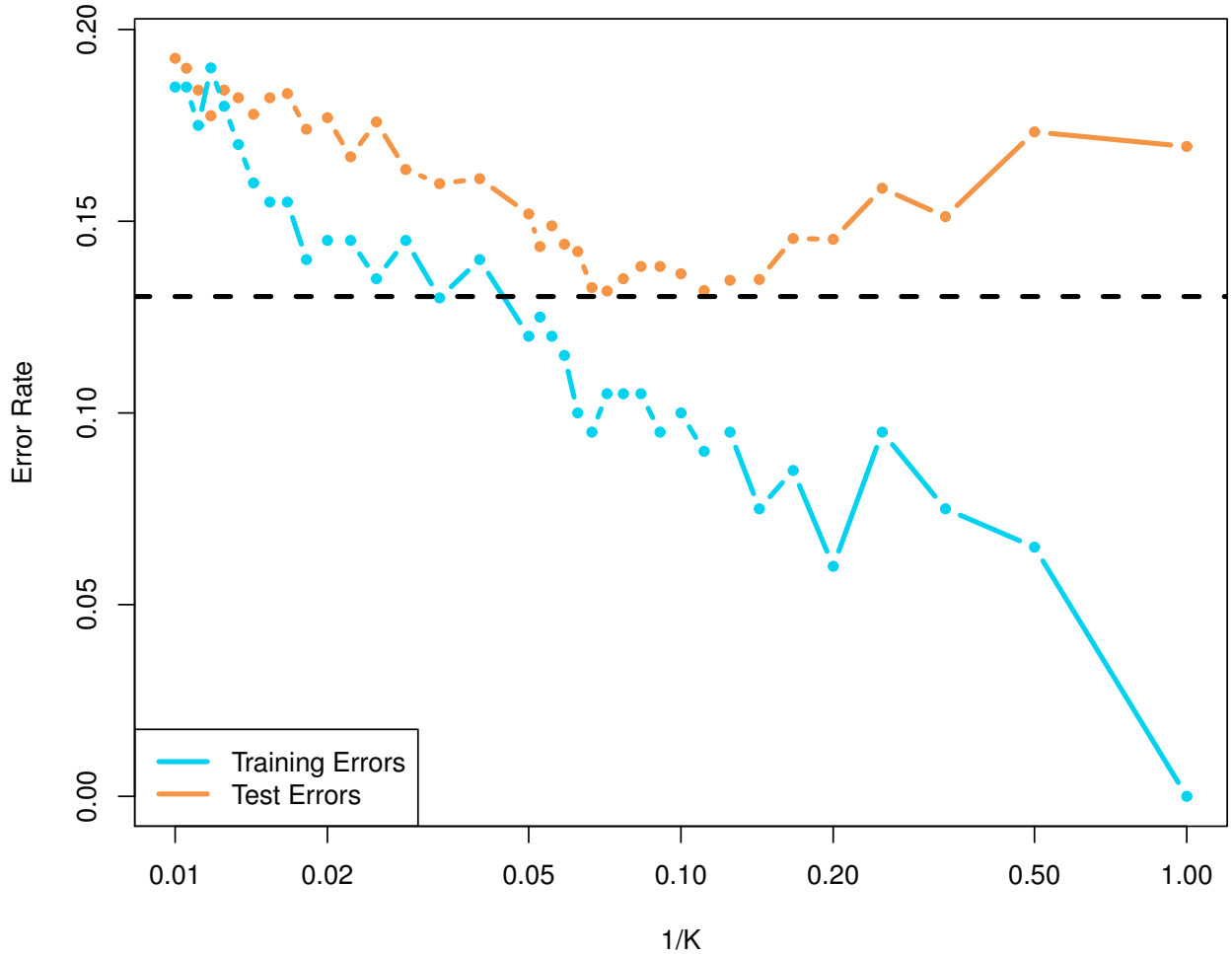
ISL Figure 2.15

- KNN with $K = 10$.
- The KNN (black) and Bayes decision (purple dashed) boundaries are very similar.
- The test error rate using KNN is 0.1363, which is close to the Bayes error rate of 0.1304.



ISL Figure 2.16

- $K = 1$, the decision boundary is overly flexible: low bias but very high variance.
- $K = 100$, decision boundary that is overly smooth: low variance but high bias.
- Neither $K = 1$ nor $K = 100$ give good predictions: they have test error rates of 0.1695 and 0.1925, respectively.



ISL Figure 2.16

- As $1/K$ increases, the method becomes more flexible.
- The training error rate consistently declines as the flexibility increases.
- The method overfits the data when $1/K$ is large.