POLITECNICO DI MILANO

Pattern Analysis and Machine Intelligence

Assessing Model Accuracy

Prof. Matteo Matteucci

Quality of Fit

- Suppose we have a regression problem.
  - A common accuracy measure is mean squared error (MSE)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

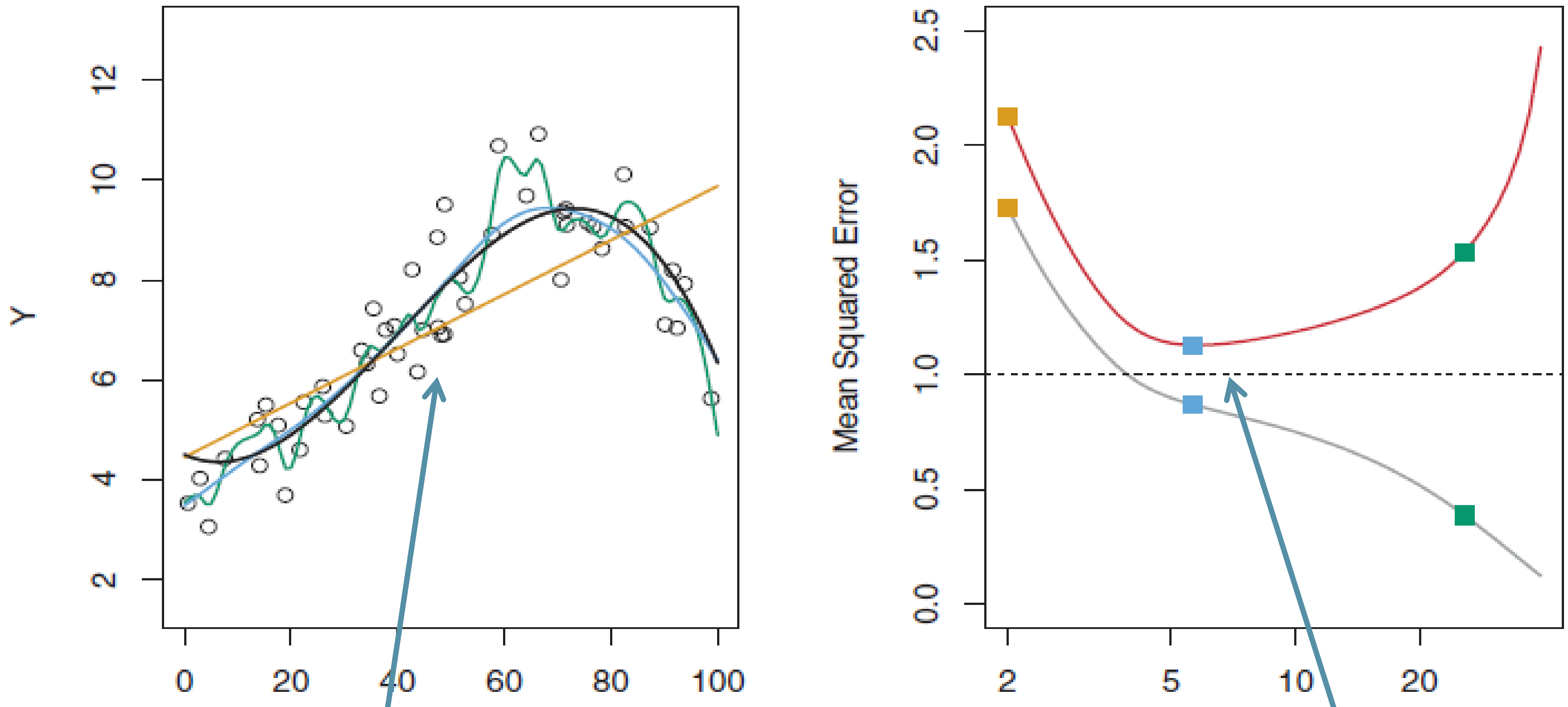
  - Where $\hat{y}_i$ is the prediction our method gives for the observation in our training data.

- Training is designed to make MSE small on training data, but …
  - What we really care about is how well the method works on new data. We call this new data "**Test Data**".
  - There is no guarantee that the method with the smallest training MSE will have the smallest test (i.e., new data) MSE.

Training vs. Test Mean Squared Error

- The more flexible a method is, the lower its training MSE will be i.e., it will "fit" or explain the training data very well.
  - *Side Note*: More Flexible methods (such as splines) can generate a wider range of possible shapes to estimate *f* as compared to less flexible and more restrictive methods (such as linear regression). The less flexible the method, the easier to interpret the model. Thus, there is a trade-off between flexibility and model interpretability.
- However, the test MSE may in fact be higher for a more flexible method than for a simple approach like linear regression

Example I

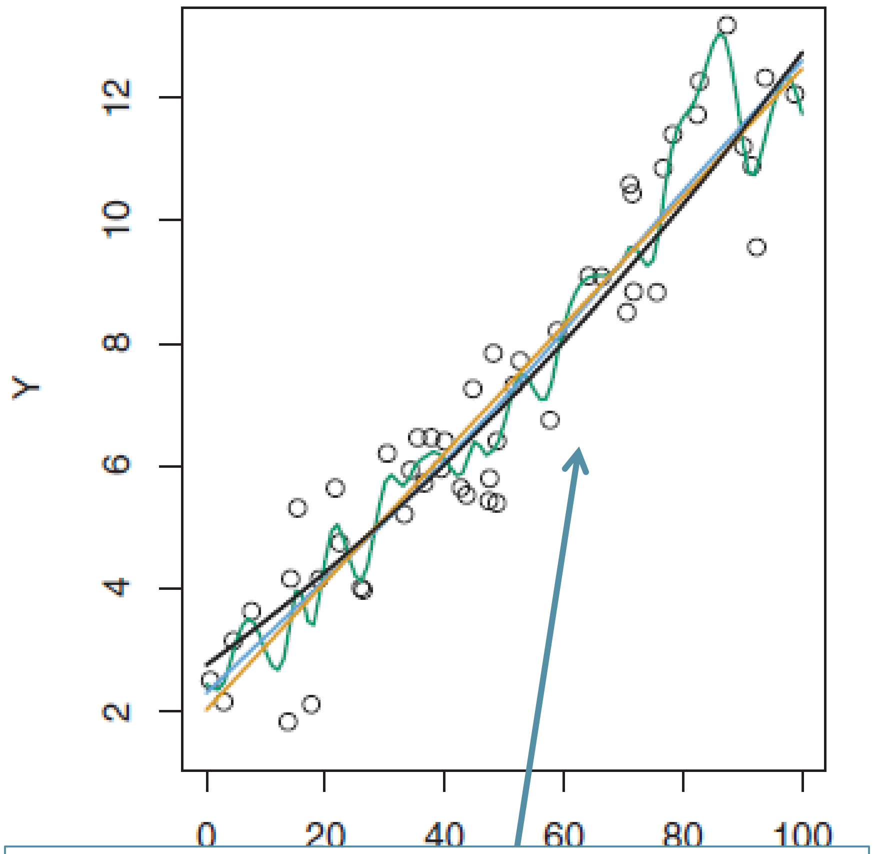


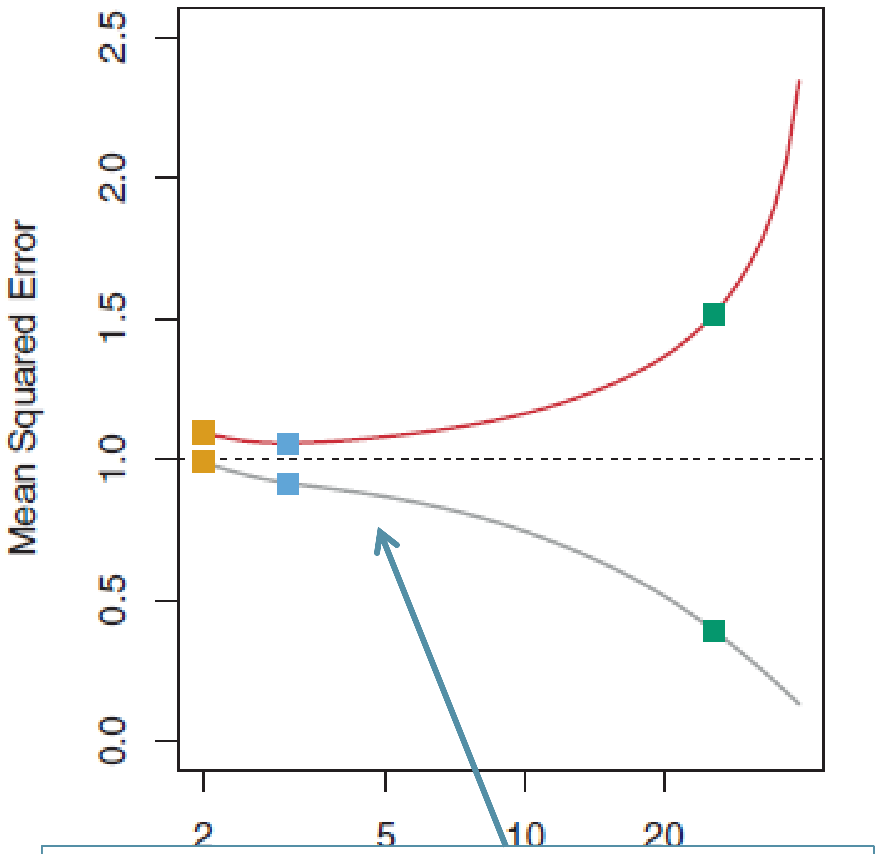
Black: Truth
Orange: Linear Estimate
Blue: smoothing spline
Green: smoothing spline

RED: Test MES
Grey: Training MSE
Dashed: Minimum possible test MSE (irreducible error)

represent the training and test MSEs for the three fits shown in the left-hand panel.

Example 2

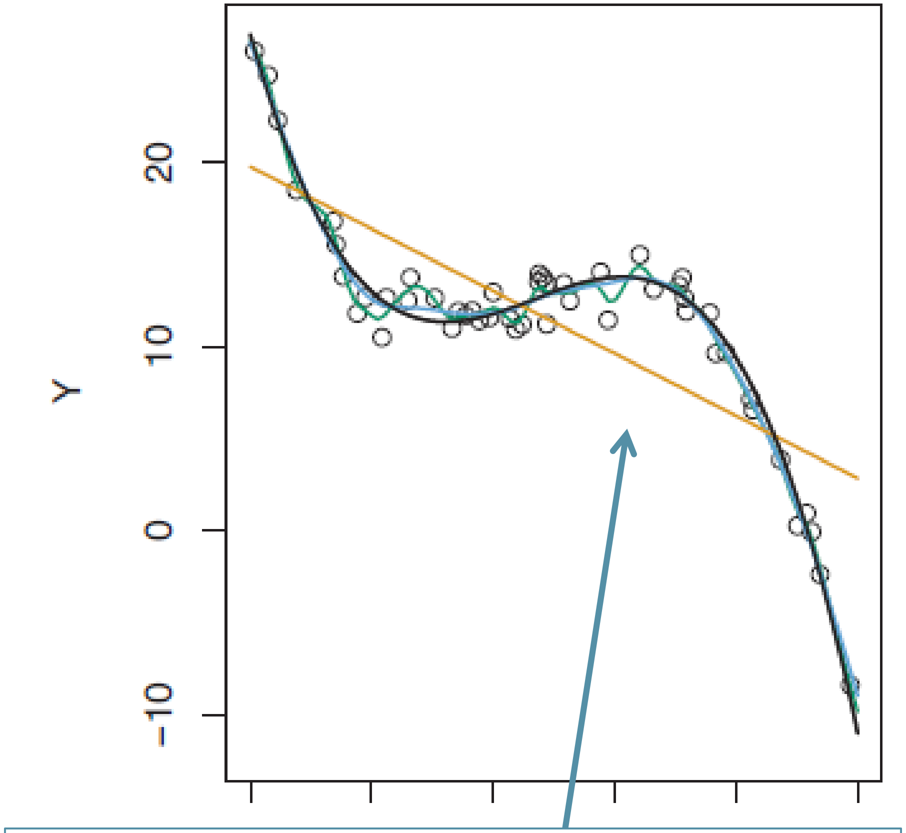


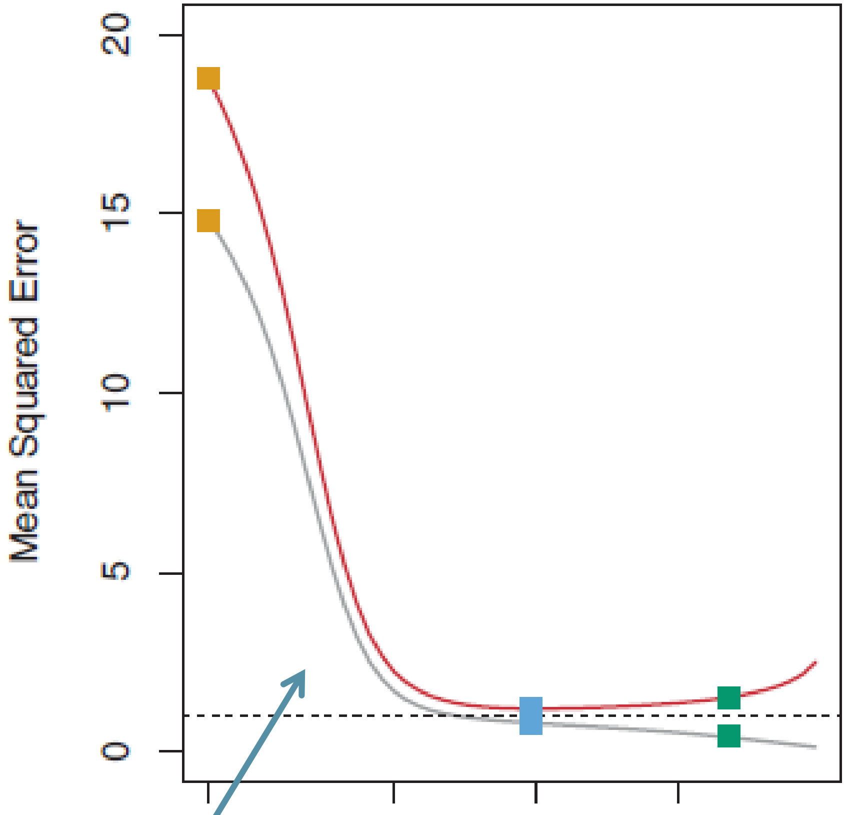


Black: Truth
Orange: Linear Estimate
Blue: smoothing spline
Green: smoothing spline

RED: Test MES
Grey: Training MSE
Dashed: Minimum possible test MSE (irreducible error)

Example 3





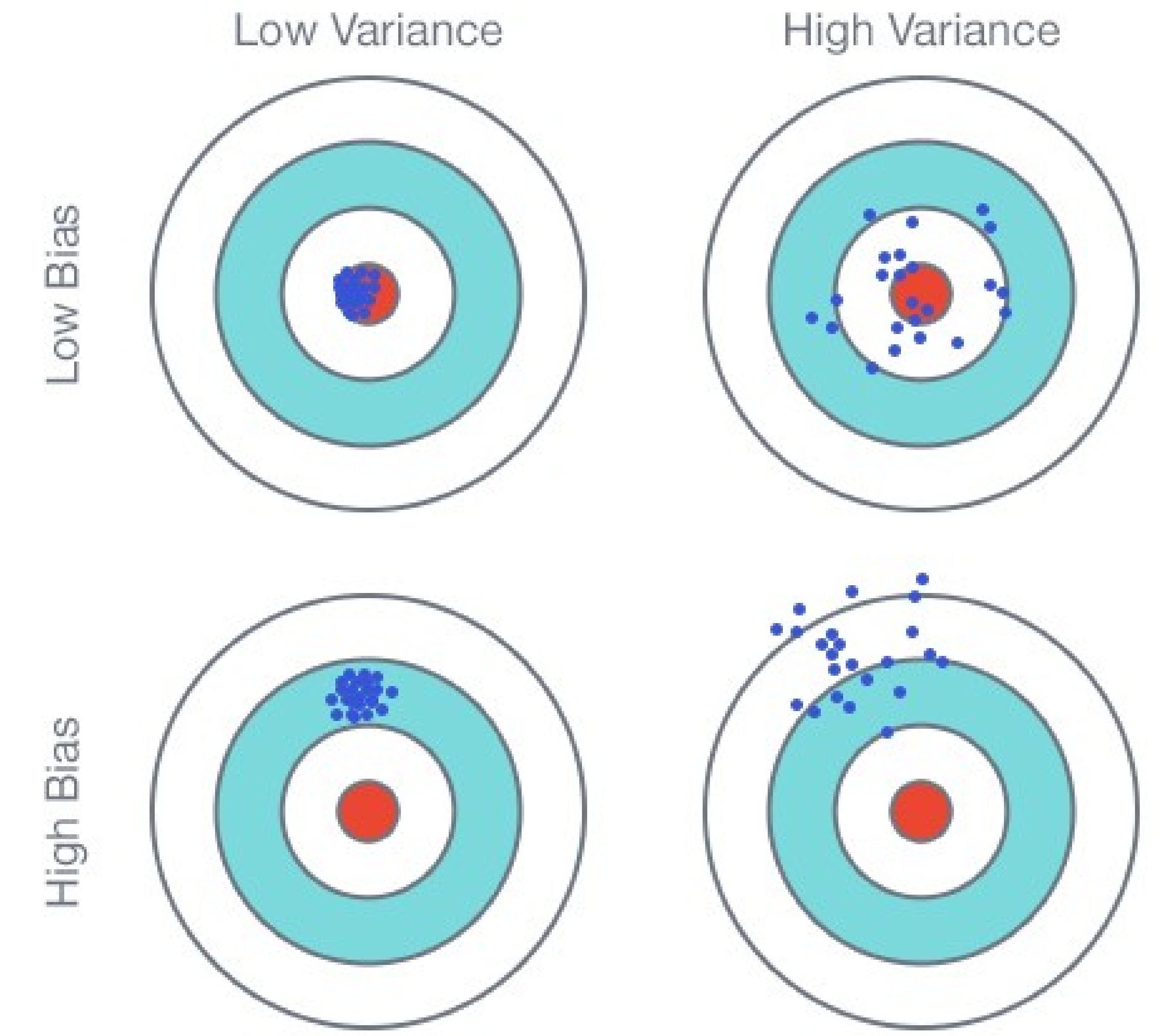
Black: Truth
Orange: Linear Estimate
Blue: smoothing spline
Green: smoothing spline

RED: Test MES
Grey: Training MSE
Dashed: Minimum possible test MSE (irreducible error)

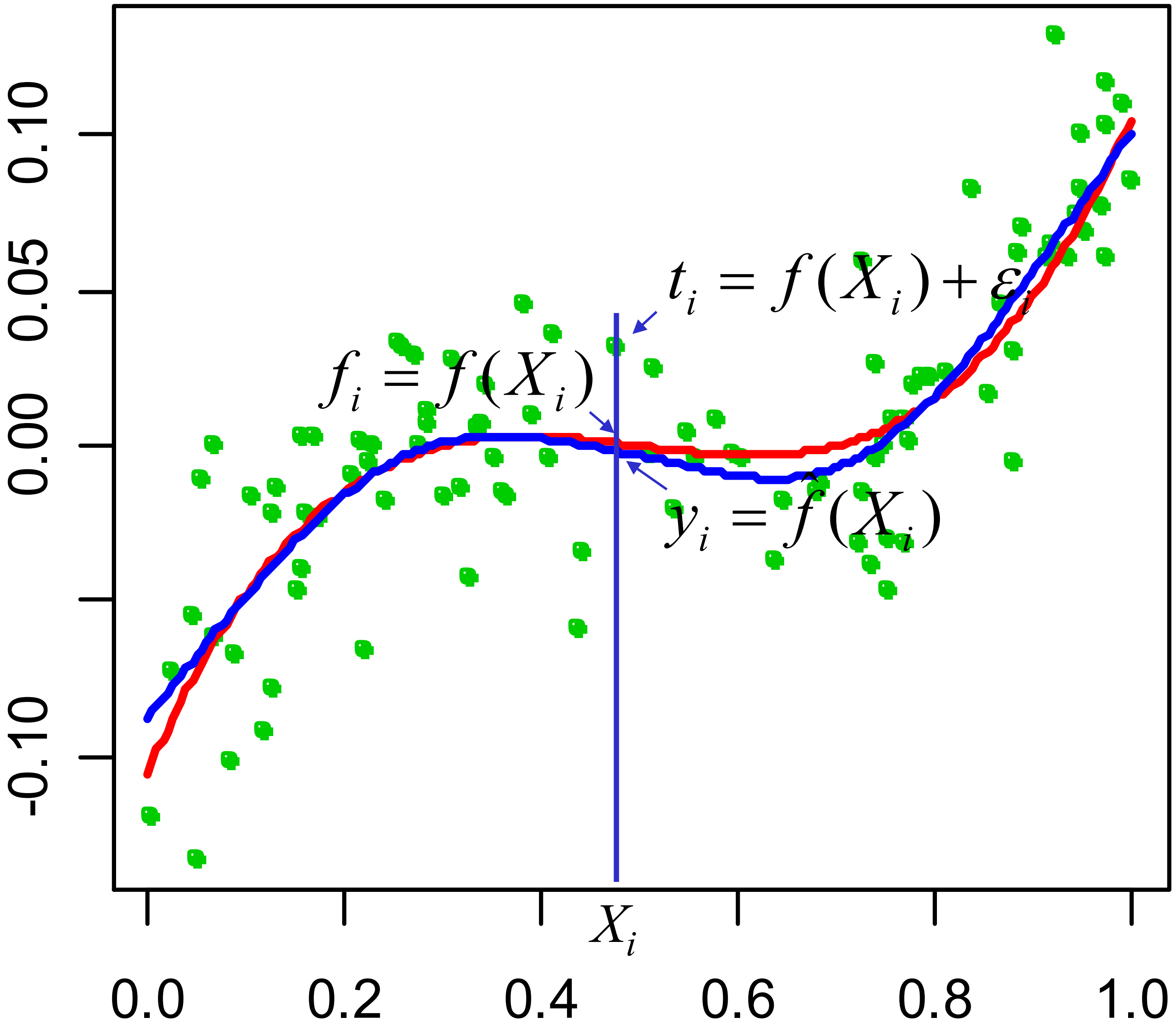
Bias/ Variance Tradeoff

- Test vs. Training MSE's illustrates a very important tradeoff that governs the choice of statistical learning methods; two competing forces that govern the choice of learning method
  - **Bias** refers to the error that is introduced by modeling a real life problem by a much simpler problem
    - E.g., linear regression assumes that there is a linear relationship between Y and X. In real life, some bias will be present
    - The more flexible/complex a method is the less bias it will have
  - **Variance** refers to how much your estimate for *f* would change by if you had a different training data set
    - Generally, the more flexible a method is the more variance it has.

Bias/ Variance Tradeoff



New Notation (from ESL)



Bias-Variance in Regression (Part 1)

- Let consider the Expected Squared Prediction Error (over any possible data)

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(t_i - y_i)^2$$

$$E\{MSE\} = E\left\{\frac{1}{N}\sum_{i=1}^{N}(t_i - y_i)^2\right\} = \frac{1}{N}\sum_{i=1}^{N}E\left\{(t_i - y_i)^2\right\}$$

- Let apply an “augmentation trick” to the expectation

$$\begin{aligned}
E\left\{(t_i - y_i)^2\right\} &= E\left\{(t_i - f_i + f_i - y_i)^2\right\} \\
&= E\left\{(t_i - f_i)^2\right\} + E\left\{(f_i - y_i)^2\right\} + 2E\left\{(f_i - y_i)(t_i - f_i)\right\} \\
&= E\left\{\varepsilon^2\right\} + E\left\{(f_i - y_i)^2\right\} + \cancel{2\left(E\left[f_i t_i\right] - E\left[f_i^2\right] - E\left[y_i t_i\right] + E\left[y_i f_i\right]\right)}
\end{aligned}$$

- Let notice that
  - Being *f* deterministic we have $E\{f_i t_i\} = f_i^2$, $E\{t_i\} = f_i$ and $E\{f_i^2\} = f_i^2$
  - Noise is independence $E\{y_i t_i\} = E\{y_i(f_i + \varepsilon)\} = E\{y_i f_i + y_i \varepsilon\} = E\{y_i f_i\} + 0$

Bias-Variance in Regression (Part 2)

- From the previous we get something already know

$$E\left\{(t_i - y_i)^2\right\} = E\left\{\varepsilon^2\right\} + E\left\{(f_i - y_i)^2\right\}$$

- Lets check the second expected value

$$E\left\{(f_i - y_i)^2\right\} = E\left\{\left(f_i - E\{y_i\} + E\{y_i\}_i - y_i\right)^2\right\}$$

$$= E\left\{(f_i - E\{y_i\})^2\right\} + E\left\{(E\{y_i\} - y_i)^2\right\} + 2E\left\{(E\{y_i\} - y_i)(f_i - E\{y_i\})\right\}$$

$$= bias^2 + Var\{y_i\} + \cancel{2\left(E\{f_i E\{y_i\}\} - E\left\{E\{y_i\}^2\right\} - E\{y_i f_i\}_i + E\{y_i E\{y_i\}\}\right)}$$

- Where we have, Because *f* is deterministic and $E\{E\{z\}\} = z$:
  - $E\{f_i E\{y_i\}\} = f_i E\{y_i\}$
  - $E\left\{E\{y_i\}^2\right\} = E\{y_i\}^2$
  - $E\{y_i f_i\} = f_i E\{y_i\}$
  - $E\{y_i E\{y_i\}\} = E\{y_i\}^2$

$$E\left\{(f_i - y_i)^2\right\} = bias^2 + Var\{y_i\}$$

$$E\left\{(t_i - y_i)^2\right\} = Var\{noise\} + bias^2 + Var\{y_i\}$$

The Trade-off

- For any given, X=*x*, the expected test MSE for a new Y will be

$$\underbrace{E\left\{\left(t_i - y_i\right)^2\right\}}_{\text{Expected Prediction Error}} = \overbrace{Var\{noise\}}^{\text{Irreducible Error}} + \underbrace{bias^2}_{\text{Model Bias}} + \overbrace{Var\{y_i\}}^{\text{Model Variance}}$$

- I.e., as a method gets more complex
  - Bias will decrease
  - Variance will increase
  - Expected Prediction Error may go up or down!

Can we actually compute those?

- Knowing the model we can compute the value of EPE
  - For a Linear Model

$$\mathrm{Err}(x_0) = \mathrm{E}[(Y - \hat{f}_\lambda)^2 | X = x_0]$$

$$\sigma^2 + \left[f(x_0) - \mathrm{E}\hat{f}(x_i) \right]^2 + \|\mathbf{h}(x_0)\|^2 \sigma^2$$

$$\frac{1}{N} \sum_{i=1}^{N} \mathrm{Err}(\mathrm{x_i}) = \sigma^2 + \frac{1}{N} \sum_{i=1}^{N} [f(x_i) - \mathrm{E}\hat{f}(x_i)]^2 + \frac{p}{N}\sigma^2$$

  - For the KNN regression fit

$$\mathrm{Err}(x_0) = \mathrm{E}[(Y - \hat{f}_\lambda)^2 | X = x_0]$$

$$= \sigma^2 + \left[f(x_0) - \frac{1}{k} \sum_{l=1}^{k} f(x_l) \right]^2 + \frac{\sigma^2}{k}$$

Test MSE, Bias and Variance

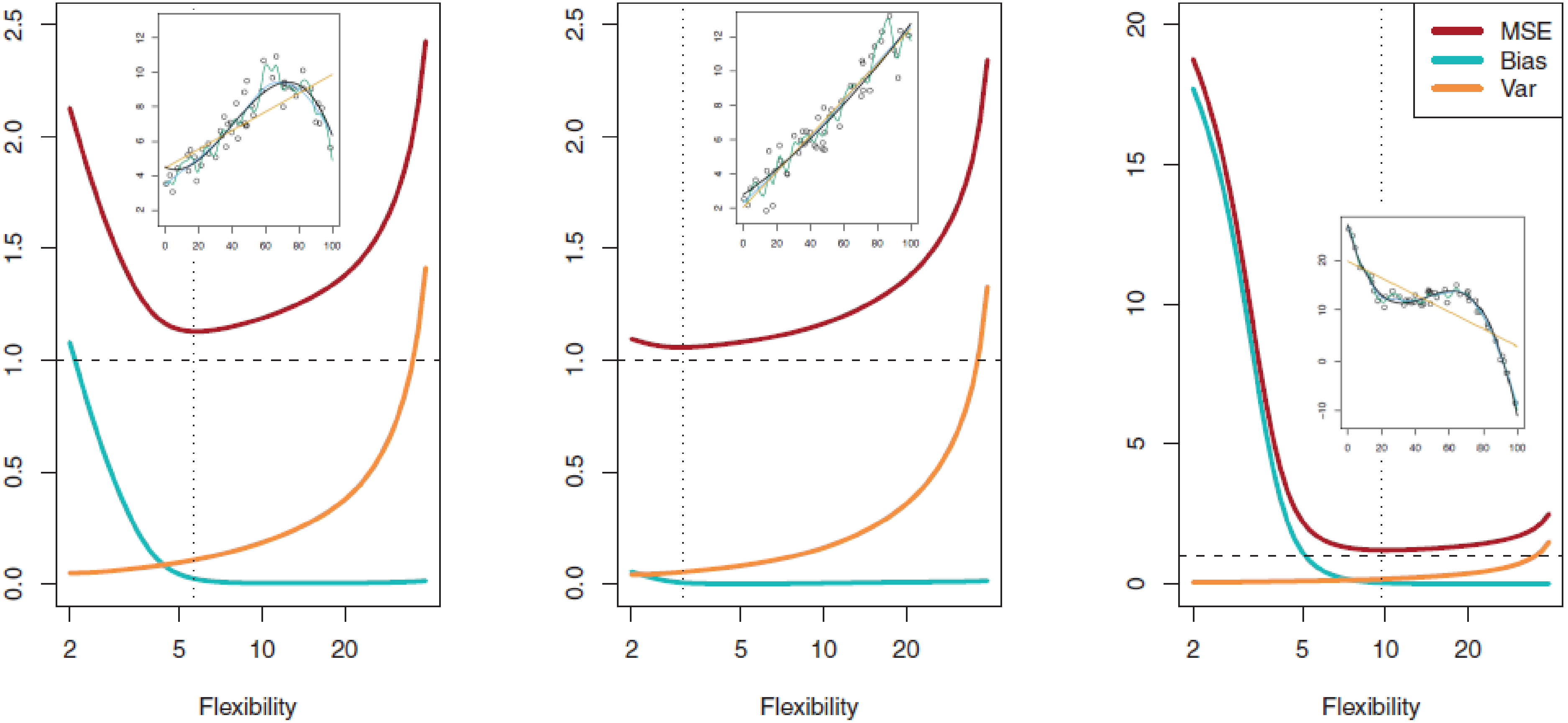


FIGURE 2.12. *Squared bias (blue curve), variance (orange curve), Var(ϵ) (dashed line), and test MSE (red curve) for the three data sets in Figures 2.9–2.11. The vertical dotted line indicates the flexibility level corresponding to the smallest test MSE.*

What about Classification?

- For a classification problem we can use the error rate i.e.

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

  - Where $I(y_i \neq \hat{y}_i)$ is an indicator function, which will give 1 if the condition $(y_i \neq \hat{y}_i)$ is correct, otherwise it gives a 0.
  - The error rate represents the fraction of incorrect classifications, or misclassifications

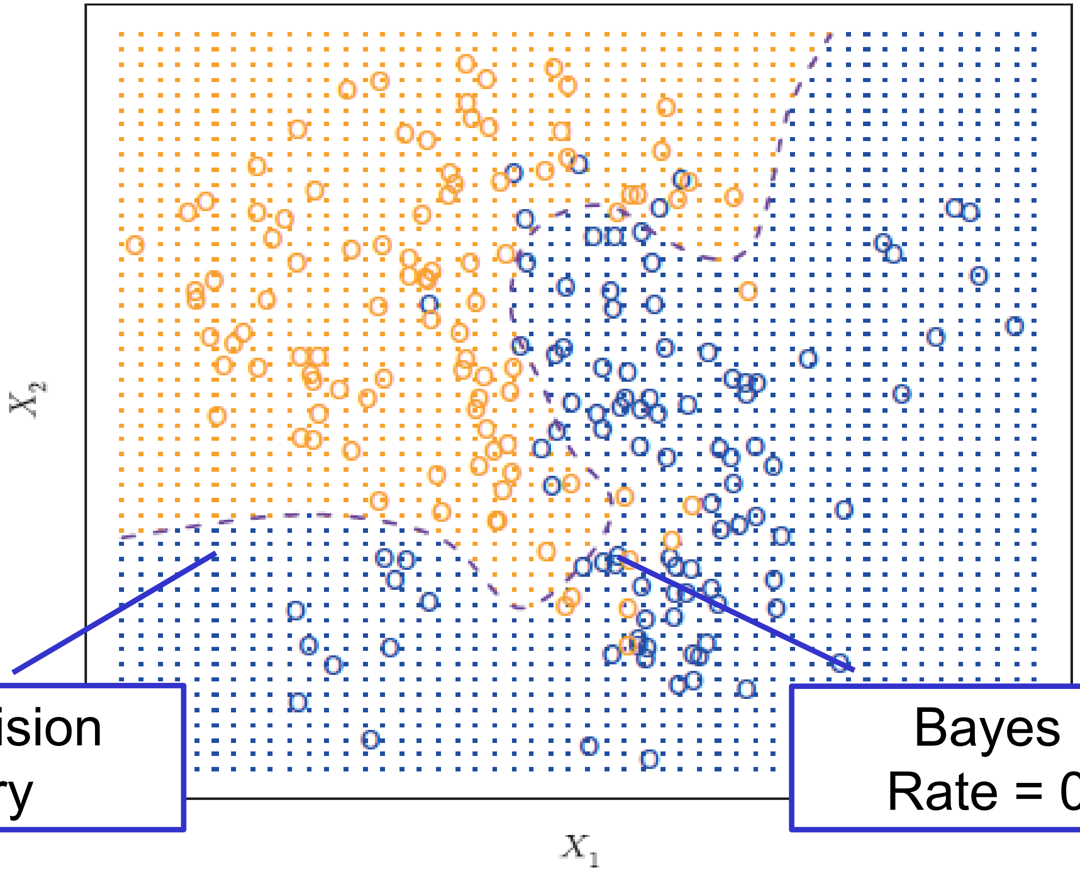
- The Bayes Classifier minimizes the Average Test Error Rate

$$\max_j P(Y = j \mid X = x_0)$$

- The **Bayes error rate** refers to the lowest possible Error Rate achievable knowing the “true” distribution of the data

$$1 - E\left(\max_j \Pr(Y = j|X)\right)$$

Bayes Classifier



Bayes Decision Boundary

Bayes Error Rate = 0.1304

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. 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.*

K-Nearest Neighbors (KNN)

- The k Nearest Neighbors method is a non parametric model often used to estimate the Bayes Classifier
  - For any given X we find the k closest neighbors to X in the training data, and examine their corresponding Y
  - If the majority of the Y's are orange we predict orange otherwise guess blue.

- Some notes about such a simple classifier …
  - The smaller the k, the more flexible the method will be
  - KNN has "zero" training time, some cost at runtime to find the k closest neighbors reduced by indexing
  - KNN has problems in higher dimensional spaces which require approximate methods

KNN Example with k = 3

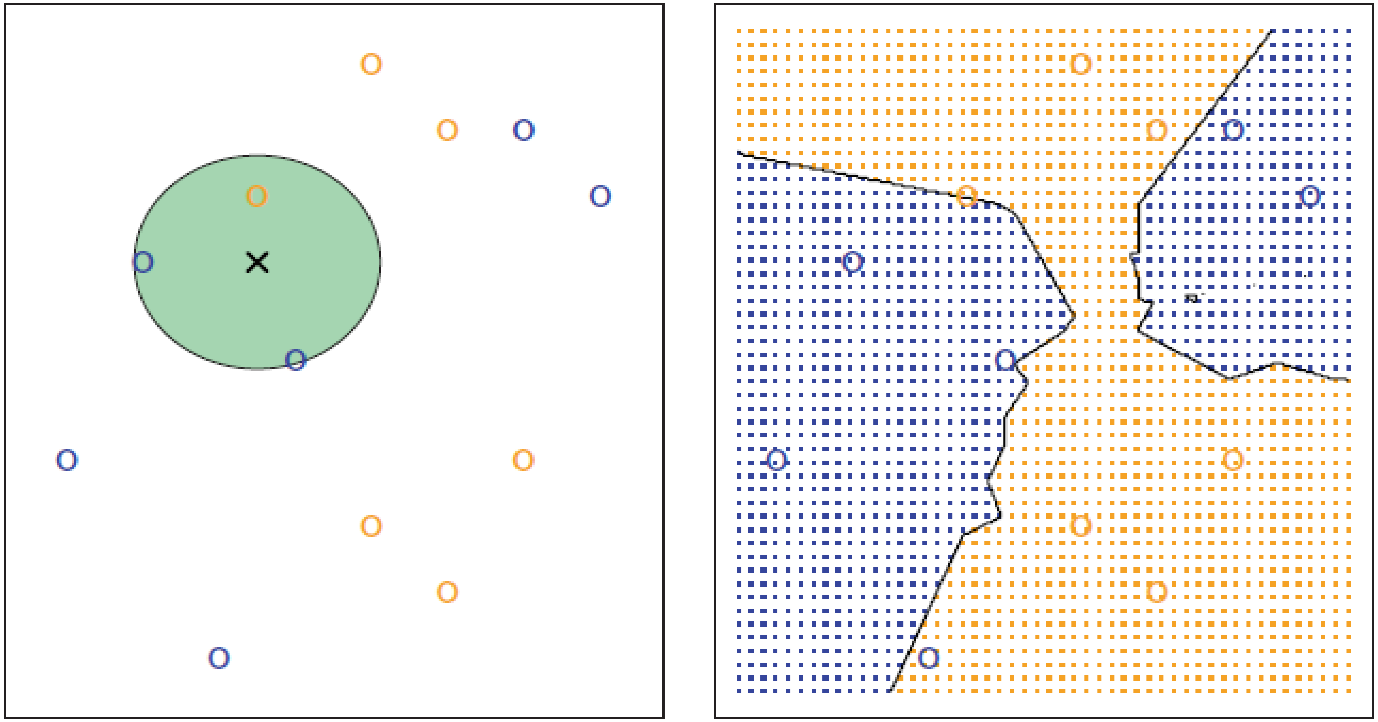
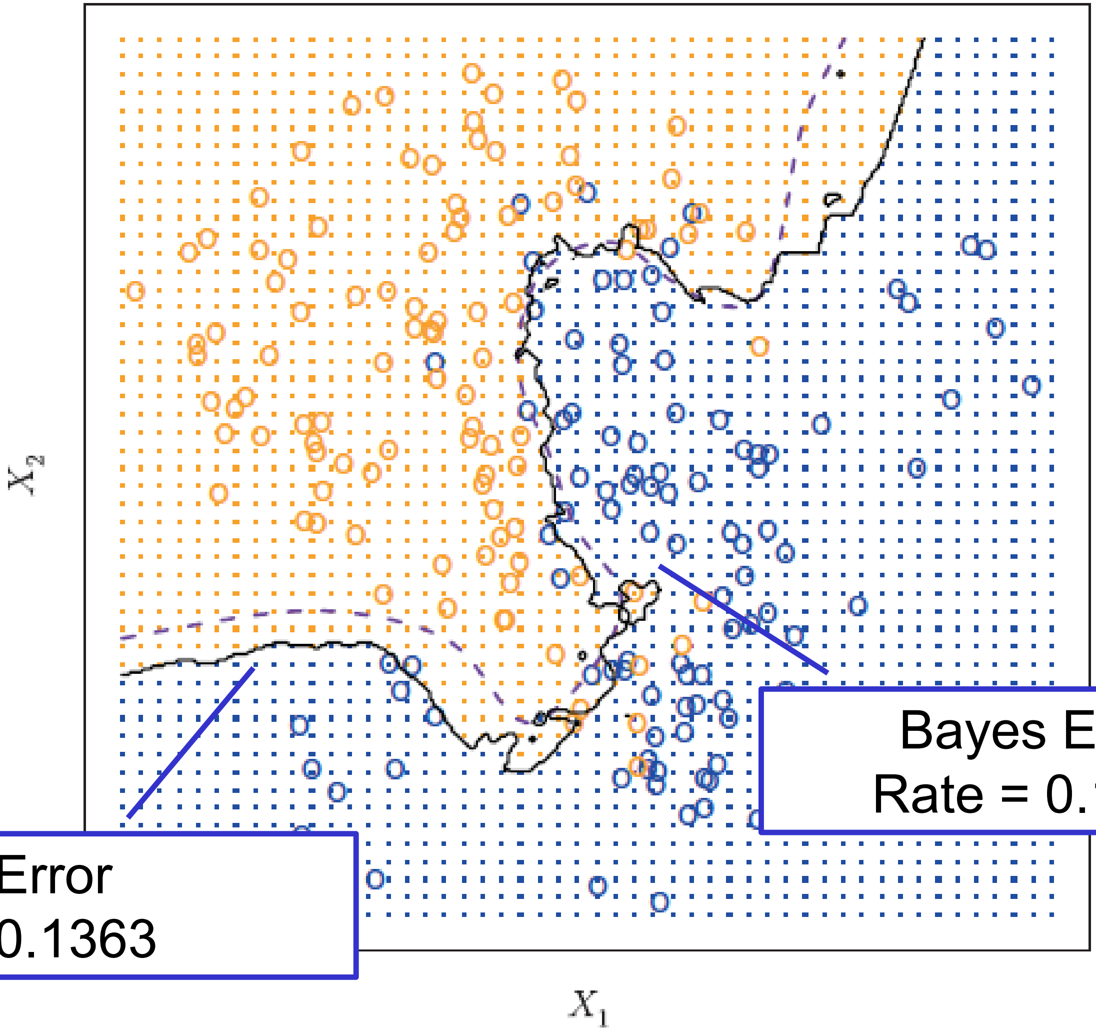


FIGURE 2.14. *The KNN approach, using* $K = 3$, *is illustrated in a simple situation with six blue observations and six orange observations.* Left: *a test observation at which a predicted class label is desired is shown as a black cross. The three closest points to the test observation are identified, and it is predicted that the test observation belongs to the most commonly-occurring class, in this case blue.* Right: *The KNN decision boundary for this example is shown in black. The blue grid indicates the region in which a test observation will be assigned to the blue class, and the orange grid indicates the region in which it will be assigned to the orange class.*

Simulated Data: K = 10



KNN Error
Rate = 0.1363

Bayes Error
Rate = 0.1304

FIGURE 2.15. *The black curve indicates the KNN decision boundary on the data from Figure 2.13, using $K = 10$. The Bayes decision boundary is shown as a purple dashed line. The KNN and Bayes decision boundaries are very similar.*

K = 1 and K = 100

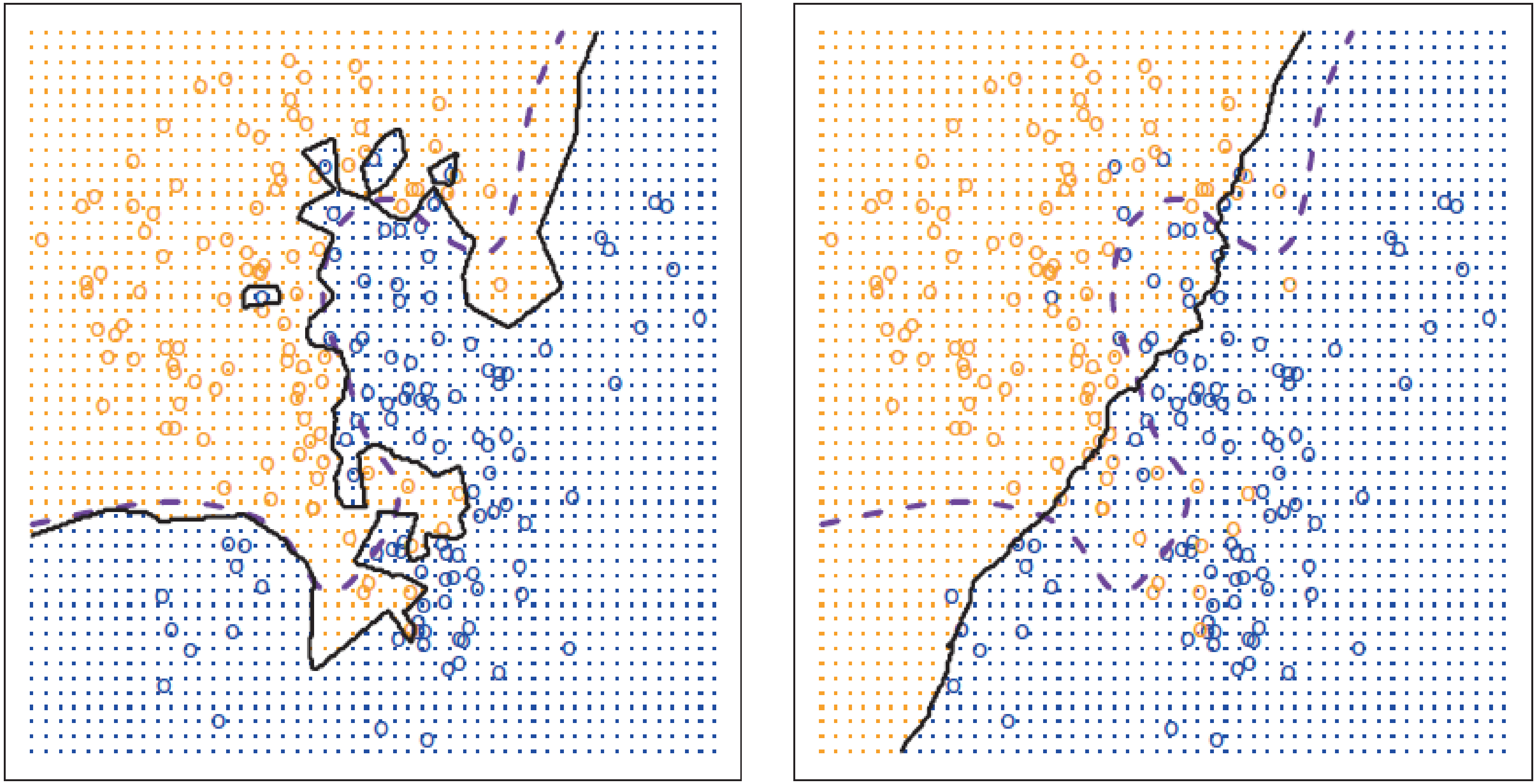


FIGURE 2.16. *A comparison of the KNN decision boundaries (solid black curves) obtained using $K = 1$ and $K = 100$ on the data from Figure 2.13. With $K = 1$, the decision boundary is overly flexible, while with $K = 100$ it is not sufficiently flexible. The Bayes decision boundary is shown as a purple dashed line.*

Training vs. Test Error Rates

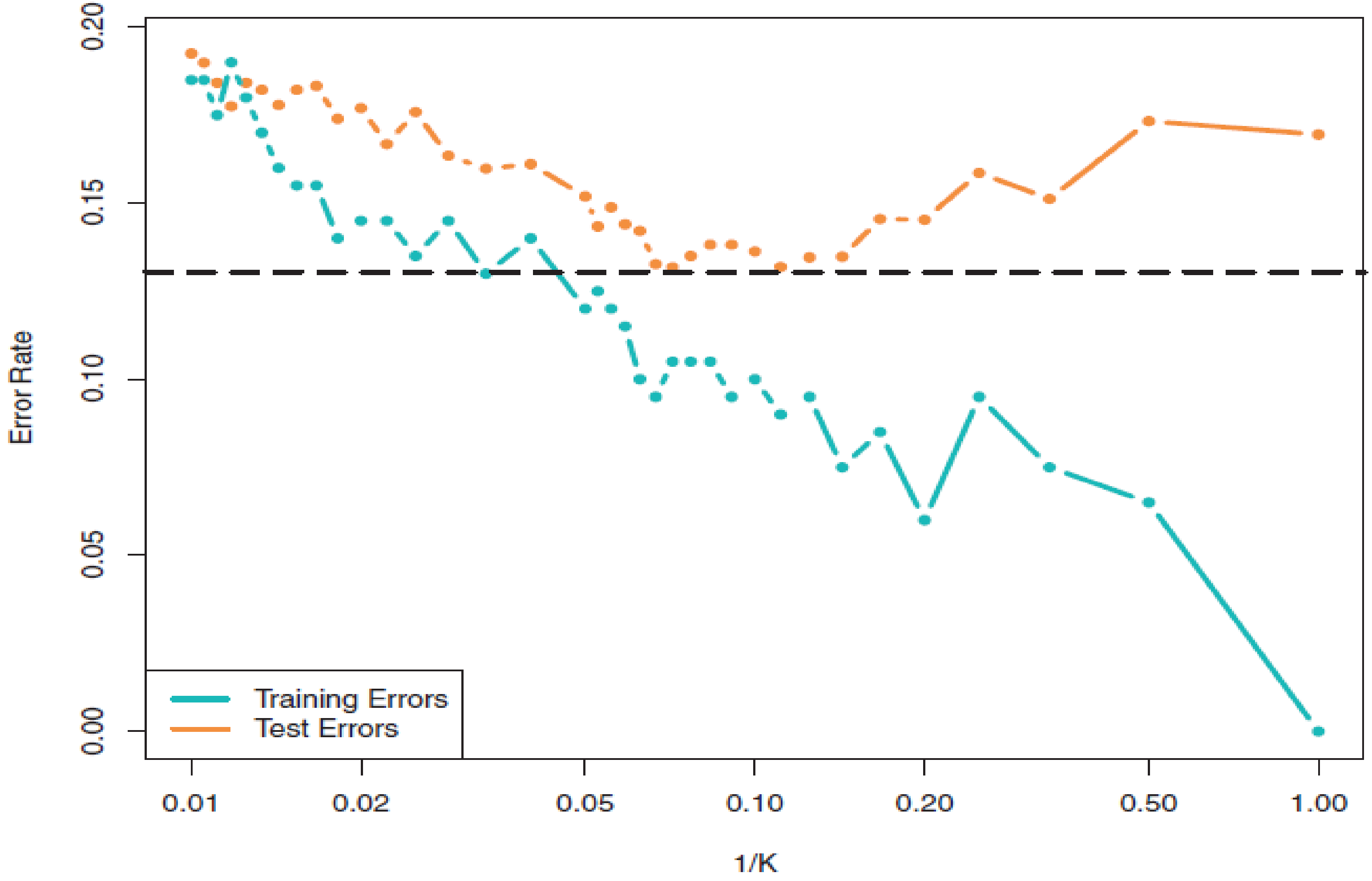
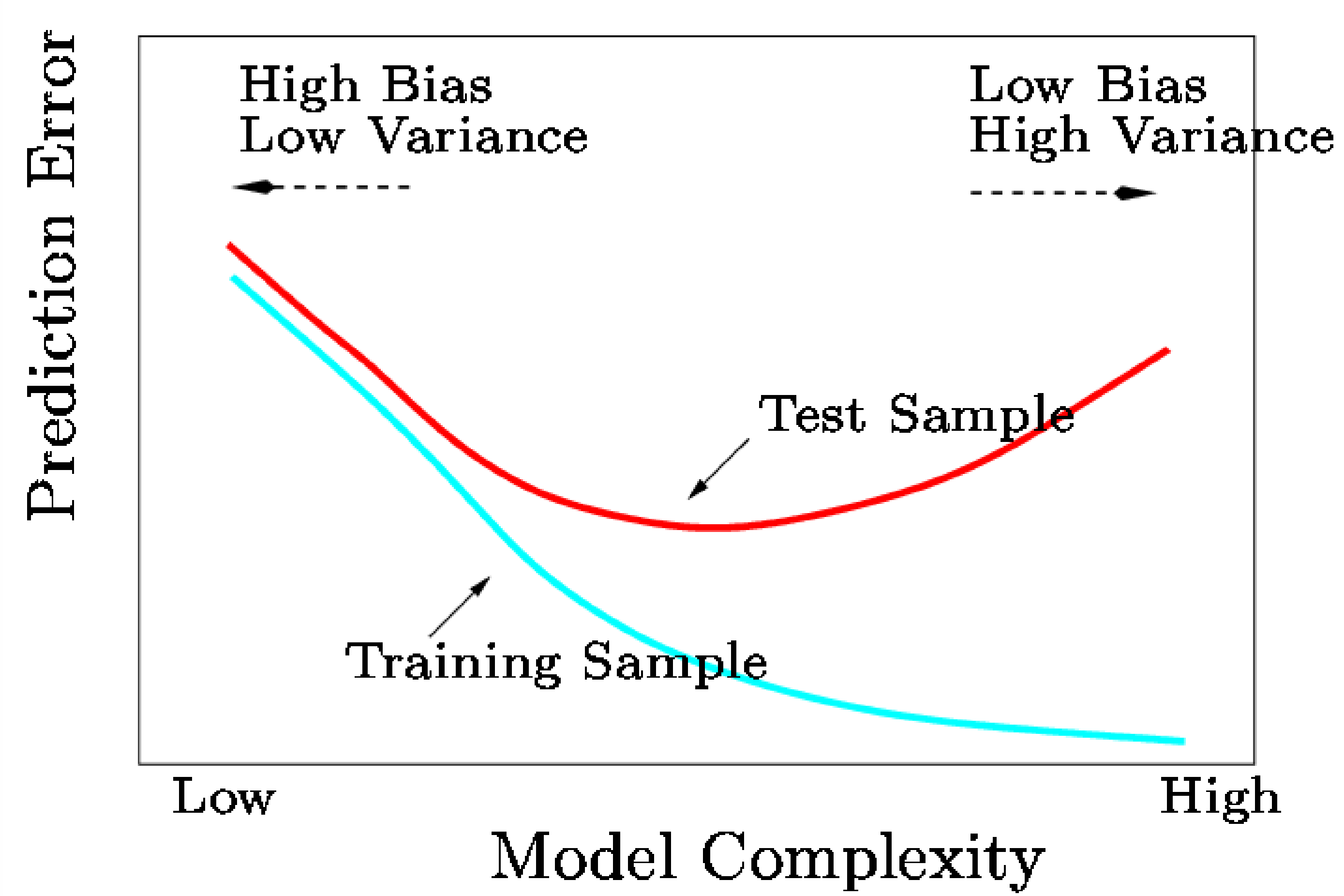


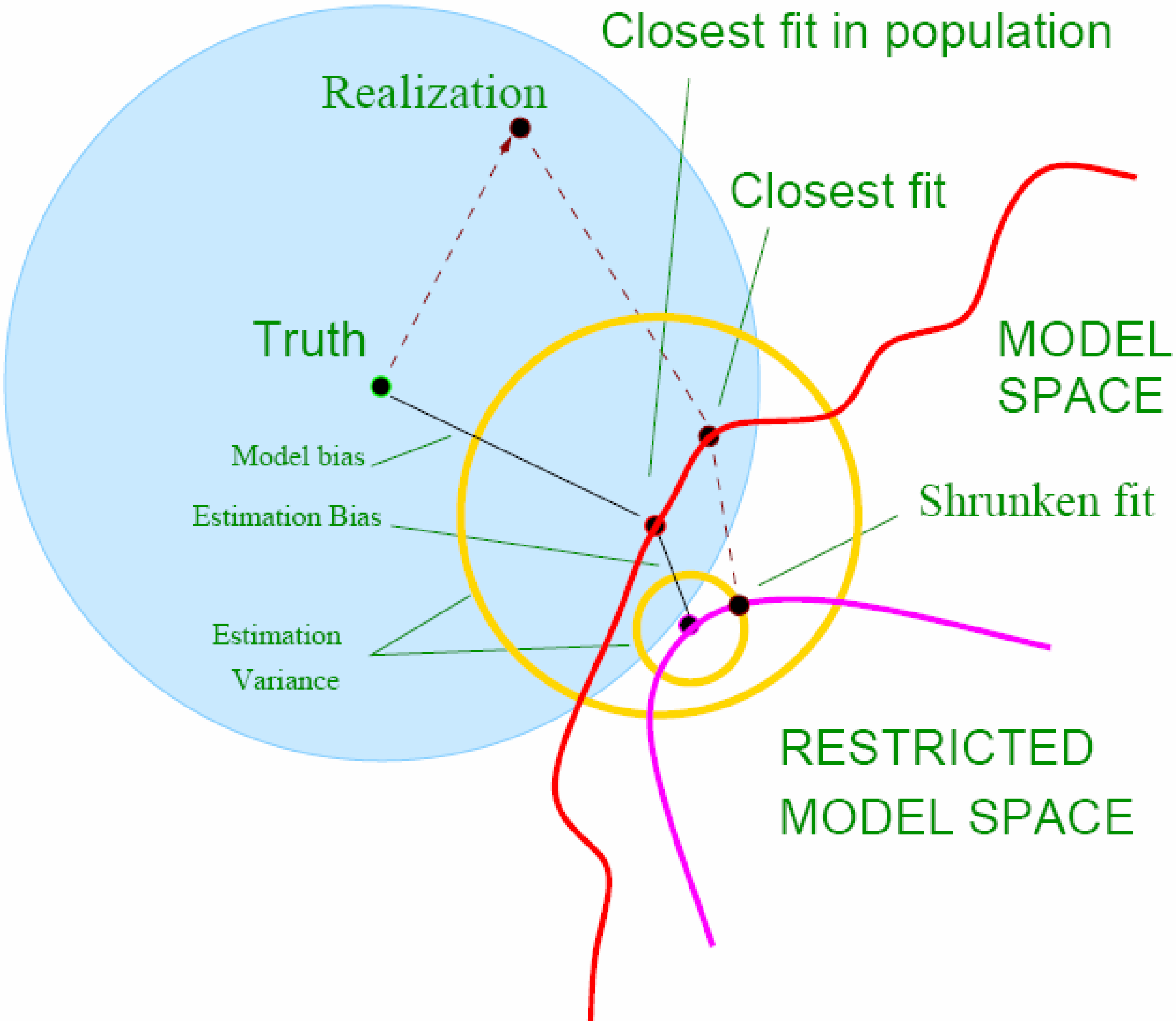
FIGURE 2.17. *The KNN training error rate (blue, 200 observations) and test error rate (orange, 5,000 observations) on the data from Figure 2.13, as the level of flexibility (assessed using 1/K) increases, or equivalently as the number of neighbors K decreases. The black dashed line indicates the Bayes error rate. The jumpiness of the curves is due to the small size of the training data set.*

A Fundamental Picture

- Training errors will always decline while test errors will decline at first (as reductions in bias dominate) but will then start to increase again (as increases in variance dominate).



A More Fundamental Picture



Closest fit in population
Realization
Closest fit
Truth
MODEL SPACE
Model bias
Estimation Bias
Shrunken fit
Estimation Variance
RESTRICTED MODEL SPACE