Model Assessment and Selection

Statistical Learning

Master in Big Data. University of Santiago de Compostela

Manuel Mucientes

Introduction

- We observe a quantitative response Y and p predictors $X=(X_1, \ldots, X_p)$: $Y = f(X) + \varepsilon$
  - ε: error term, and has zero mean

- Prediction: $\hat{Y} = \hat{f}(X)$

- Expected value of the squared difference between the predicted and actual value:

$$
\begin{aligned}
E(Y - \hat{Y})^2 &= E[f(X) + \epsilon - \hat{f}(X)]^2 \\
&= \underbrace{[f(X) - \hat{f}(X)]^2}_{\text{Reducible}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{Irreducible}}
\end{aligned}
$$

Introduction (ii)

- Model assessment: given a model, estimate its prediction error (generalization error) on new data
- Model selection: estimate the performance of different models in order to choose the best one
- No free lunch theorem: no one method dominates all others over all possible data sets
  - Select on a dataset the method that produces the best result

Measuring the Quality of Fit

- Quality of fit for regression:
  - Mean Squared Error (MSE): $MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$
  - Residual Sum of Squares (RSS): $\mathrm{RSS} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
  - R^2 statistic: $R^2 = \frac{\mathrm{TSS} - \mathrm{RSS}}{\mathrm{TSS}} = 1 - \frac{\mathrm{RSS}}{\mathrm{TSS}}$ $\mathrm{TSS} = \sum(y_i - \bar{y})^2$
- Training MSE vs. test MSE
- Choose the method with lowest test MSE: $\mathrm{Ave}(\hat{f}(x_0) - y_0)^2$
- There is no guarantee that the method with the lowest training MSE will also have the lowest test MSE

Measuring the Quality of Fit (ii)

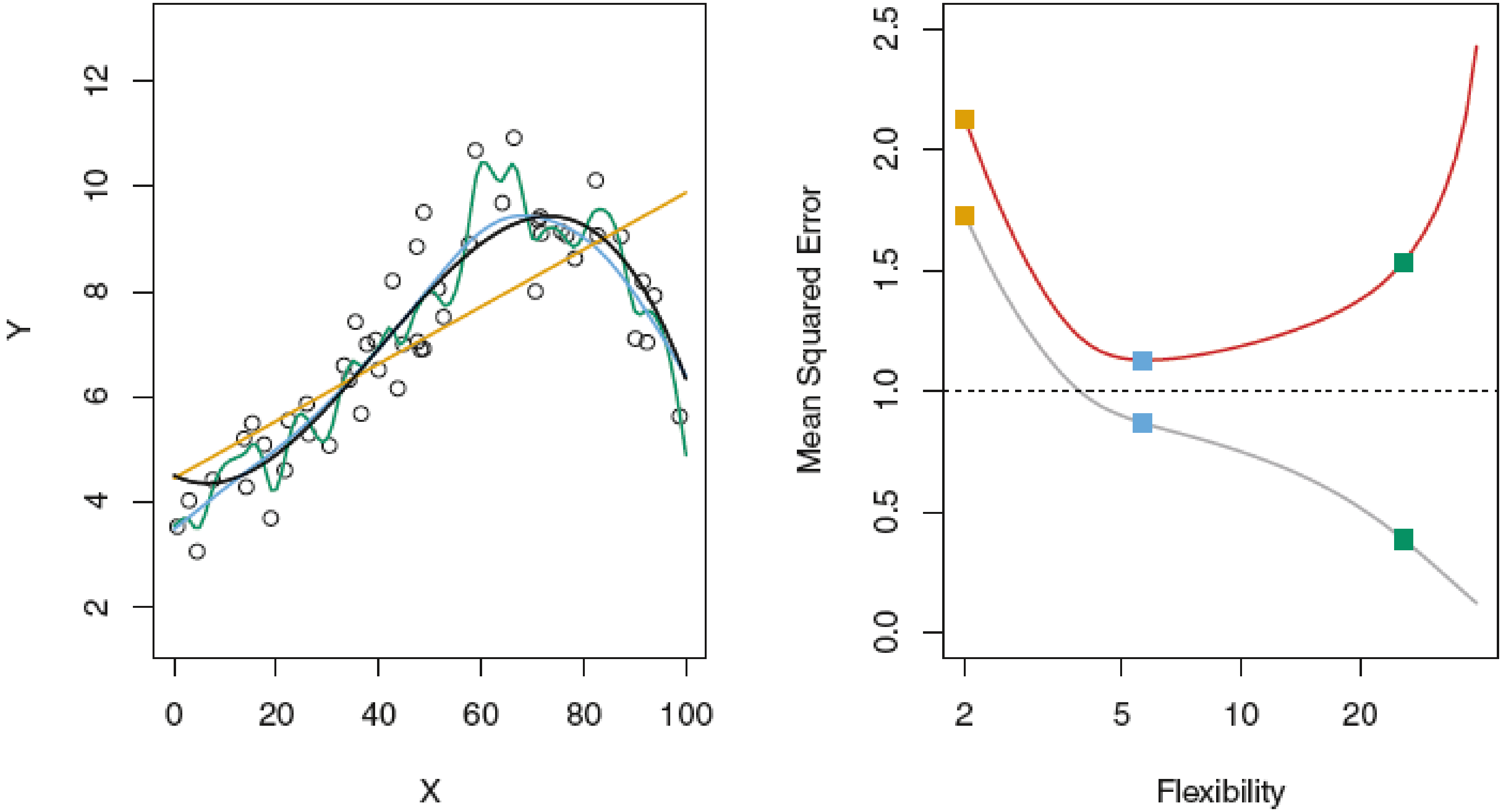


FIGURE 2.9. Left: *Data simulated from f, shown in black. Three estimates of f are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves).* Right: *Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line). Squares represent the training and test MSEs for the three fits shown in the left-hand panel.*

Measuring the Quality of Fit (iii)

- As model flexibility (complexity) increases, training MSE decreases, but test MSE may not
- Always expect training MSE to be smaller than test MSE
- Overfitting the data: small training MSE, but large test MSE
  - A less flexible model would have yielded a smaller test MSE

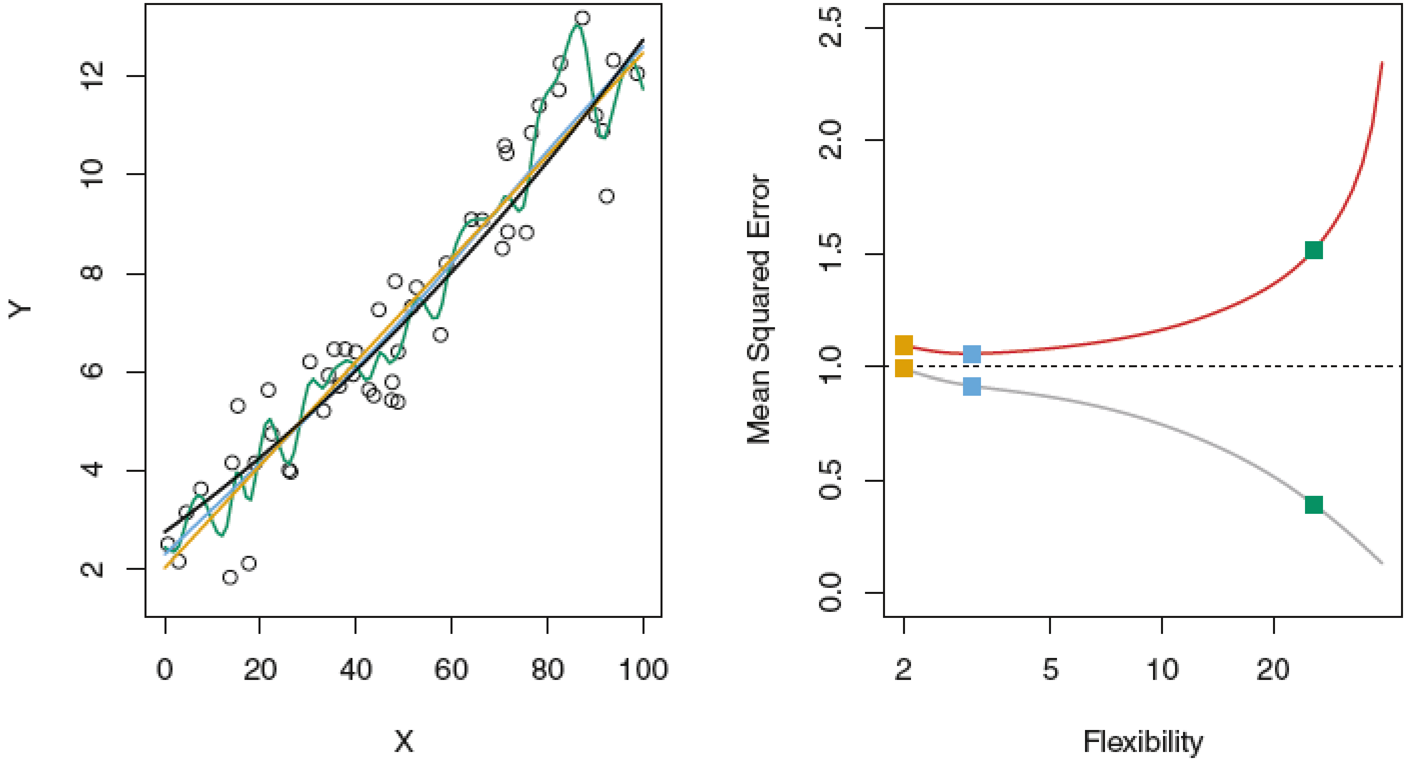


Fig. 2.10

Measuring the Quality of Fit (iv)

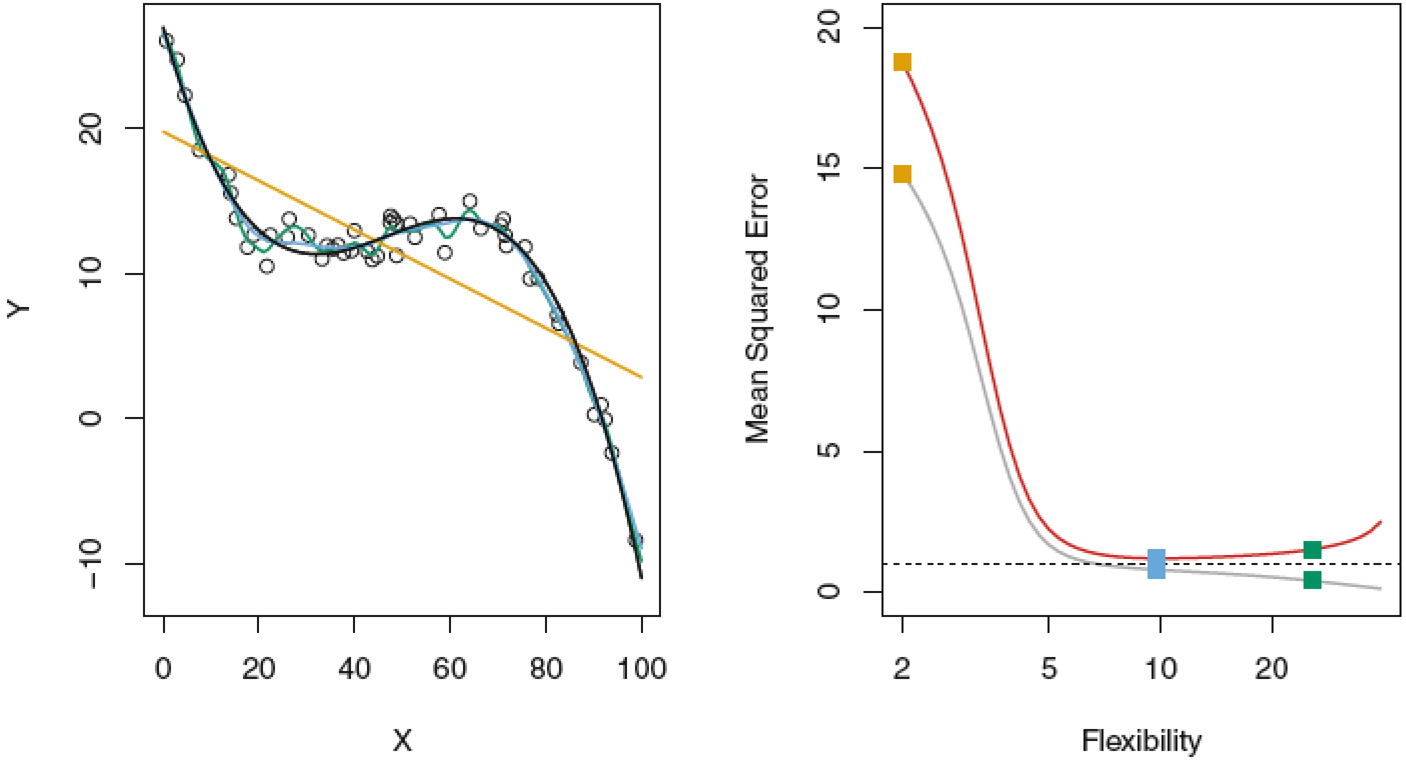


Fig. 2.11

Bias-Variance Trade-Off

- Expected test MSE:

$$E\left(y_0 - \hat{f}(x_0)\right)^2 = \mathrm{Var}(\hat{f}(x_0)) + [\mathrm{Bias}(\hat{f}(x_0))]^2 + \mathrm{Var}(\epsilon)$$

  - Repeatedly estimate *f* from a large number of training sets, and test each at x_0
  - Overall expected test MSE: average the expected test MSE over all x_0 in the test set

- Var(ε): irreducible error, test MSE can never lie below

- Variance: amount by which $\hat{f}$ would change if it is estimated with a different training set
  - More flexible methods (more complexity) have (generally) higher variance
    - Example: moving a single observation in Fig. 2.9

Bias-Variance Trade-Off (ii)

- Bias: error introduced by approximating a real-life problem with a much simpler model
  - Example: linear model has a high bias in Fig. 2.11
  - Example: linear model has a low bias in Fig. 2.10
  - Generally, more flexible methods result in less bias

- General rule: as we use more flexible methods, the variance will increase and the bias will decrease

- For a class of methods, as we increase the flexibility:
  - Initially, bias tends to decrease faster than the variance increases
  - At some point, increasing the flexibility has little impact on bias, but significantly increases variance

Bias-Variance Trade-Off (iii)

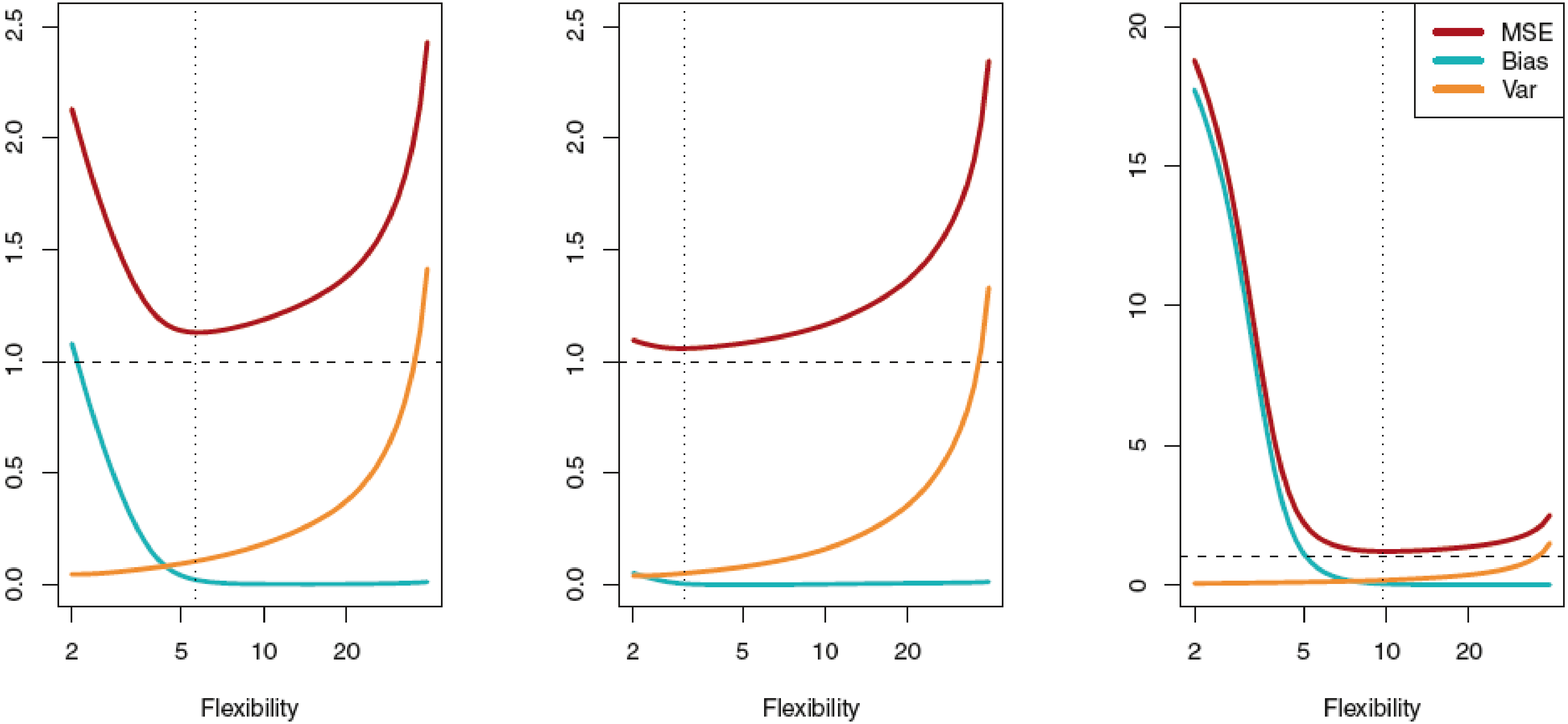


FIGURE 2.12. *Squared bias (blue curve), variance (orange curve), Var(ϵ) (dashed line), and test MSE (red curve) for the three data sets in Figures 2.9–2.11. The vertical dotted line indicates the flexibility level corresponding to the smallest test MSE.*

Classification Setting

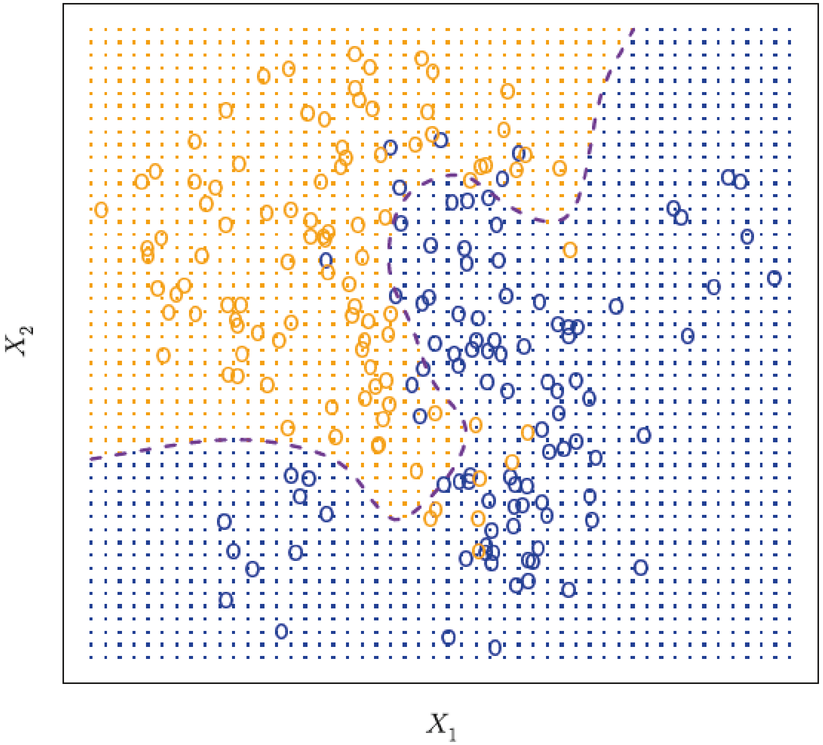
- Most of the concepts for the regression setting are valid for classification
- Quality of a fit (accuracy)
  - Training error rate: $\frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$
  - Test error rate: $\text{Ave}\left(I(y_0 \neq \hat{y}_0)\right)$
- Test error rate is minimized by the Bayes classifier
- Bayes classifier:
  - Assign each observation to the most likely class given its predictor values
  - I.e., assign to the class that maximizes $\Pr(Y = j|X = x_0)$

Classification Setting (ii)

- Example: simulated data
  - 100 observations of two classes
  - Purple dashed line: Bayes decision boundary
- Lowest test error rate: Bayes error rate

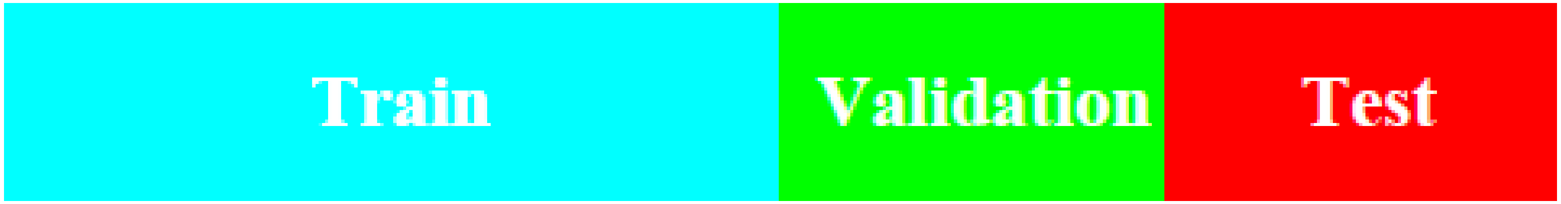
$$1 - E\left(\max_j \Pr(Y = j|X)\right)$$

- Example: classes overlap
  - Bayes error rate is 0.1304
- Bayes error rate is analogous to the irreducible error
- Real data: conditional distribution of *Y* given *X* is unknown
  - Computing the Bayes classifier is impossible



Data for assessment and selection

- Data-rich situation:



Train

Validation

Test

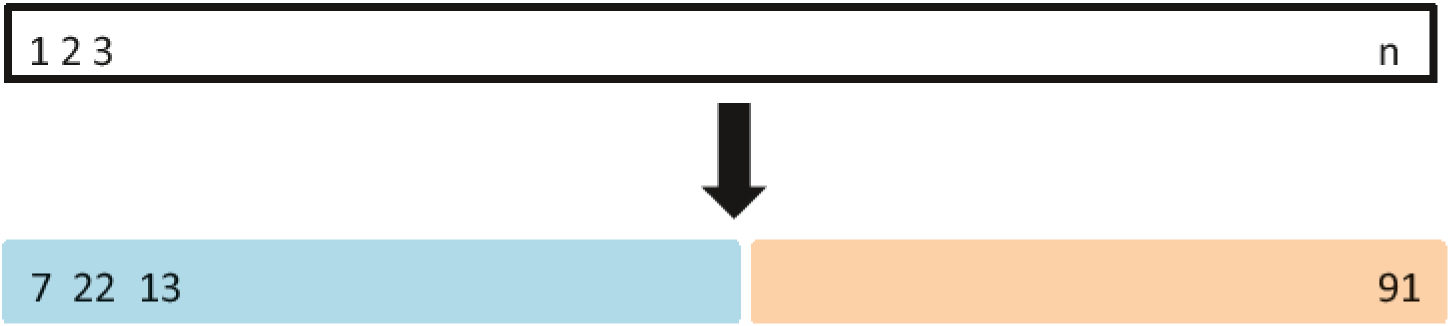
  - Training set: fit the model
  - Validation set: estimate the test MSE for model selection
  - Test set: assess the generalization error of the final chosen model
    - Must be kept in a “vault”: **DO NOT USE THE TEST SET TO SELECT THE VALUE OF ANY PARAMETER OR HYPER-PARAMETER**

Data for assessment and selection (ii)

- Resampling methods:
  - Estimate test MSE using the training data
  - Repeatedly draw samples from a training set and refit a model on each sample
    - Obtain additional information of a fitted model: variability
  - **Cross-validation**:
    - In general, best method both for model selection and assessment
    - Can be applied to any learning method
  - Bootstrap

Validation Set Approach

- Divide the available observations in training set and validation set or hold-out set

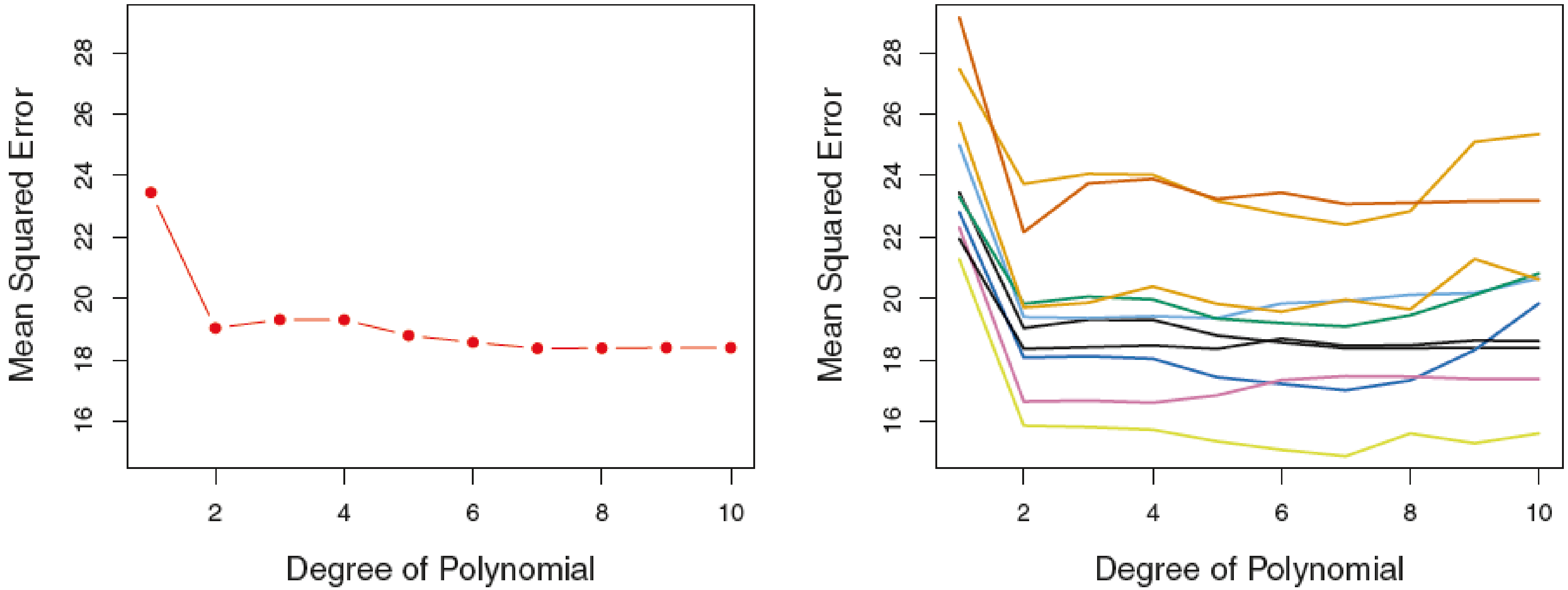


  - Validation set error provides an estimate of the test error

- Example: Auto data set
  - Predict *mpg* using *horsepower*, *horsepower*^{2}, etc.
  - 392 observations: training (196) and validation (196) sets

Validation Set Approach (ii)

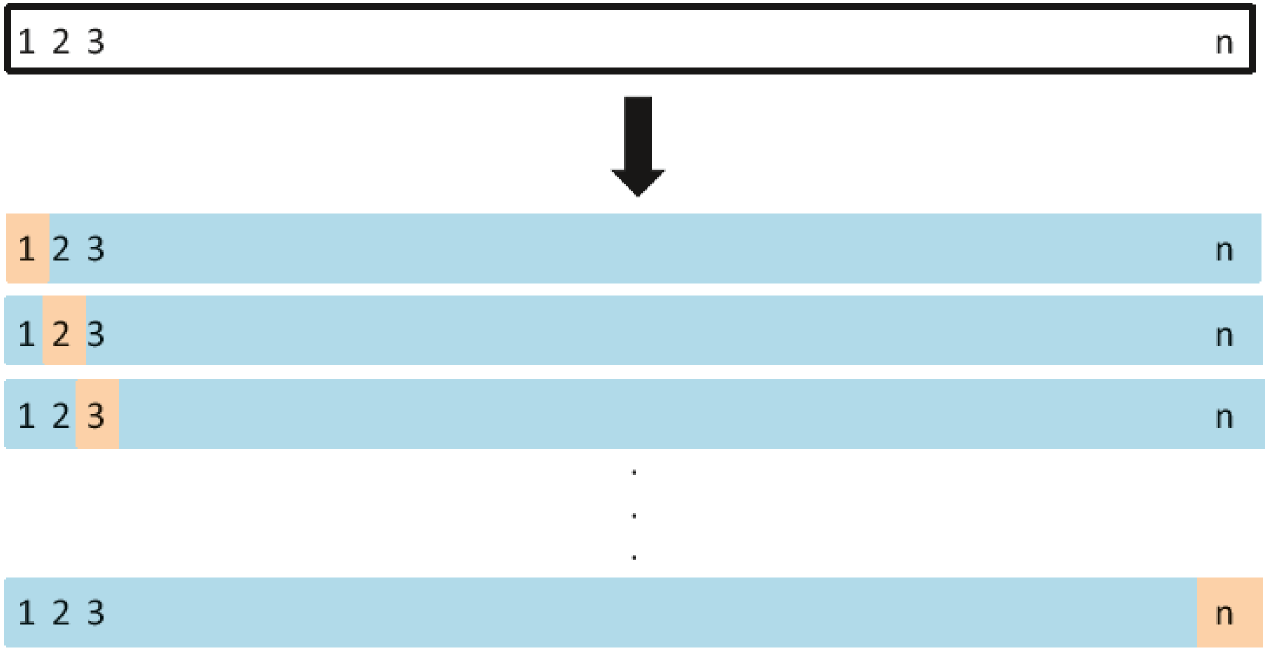
- Example: Auto data set



- Drawbacks of the validation set approach:
  - High variability of the test error estimation, depending on training/validation set split
  - Only a subset of the observations are included in the training set: worse performance of the trained model
    - Validation set error overestimates the test error of the model fit on the entire data set

Leave-One-Out Cross Validation (LOOCV)

- A single observation is used for the validation set:

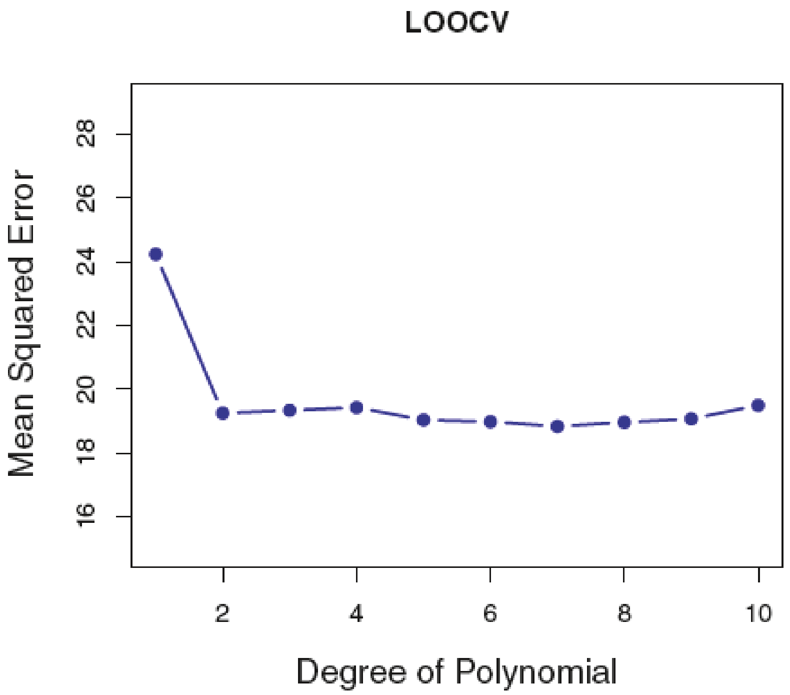


- LOOCV estimate of the test error: $$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\mathrm{MSE}_i$$

© Manuel Mucientes

LOOCV (ii)

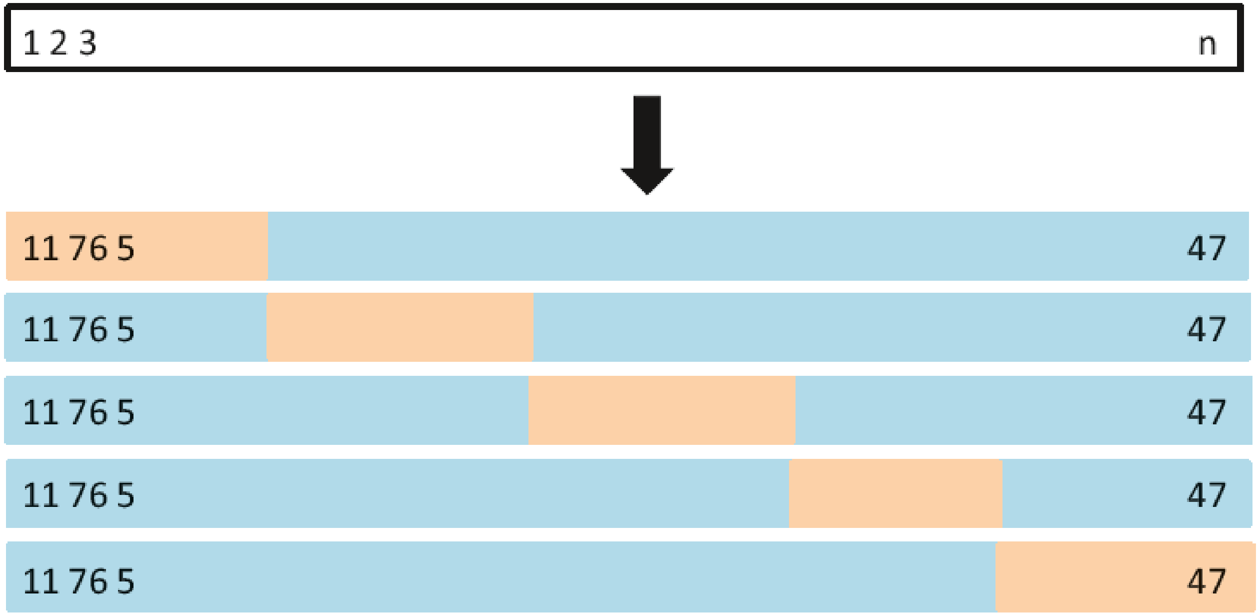
- Advantages of LOOCV over the validation set approach:
  - Far less bias: training with *n*-1 observations
    - LOOCV has a lower overestimation of the test error
  - No randomness in the training/validation set splits
    - Always the same results



- Example: Auto data set
- Disadvantage of LOOCV: fit the model *n* times
  - For least squares or polynomial regression, there is a shortcut (James et al., 2013, pp. 98)

k-Fold Cross-Validation

- Divide the observations into *k* folds or groups of approximately equal size

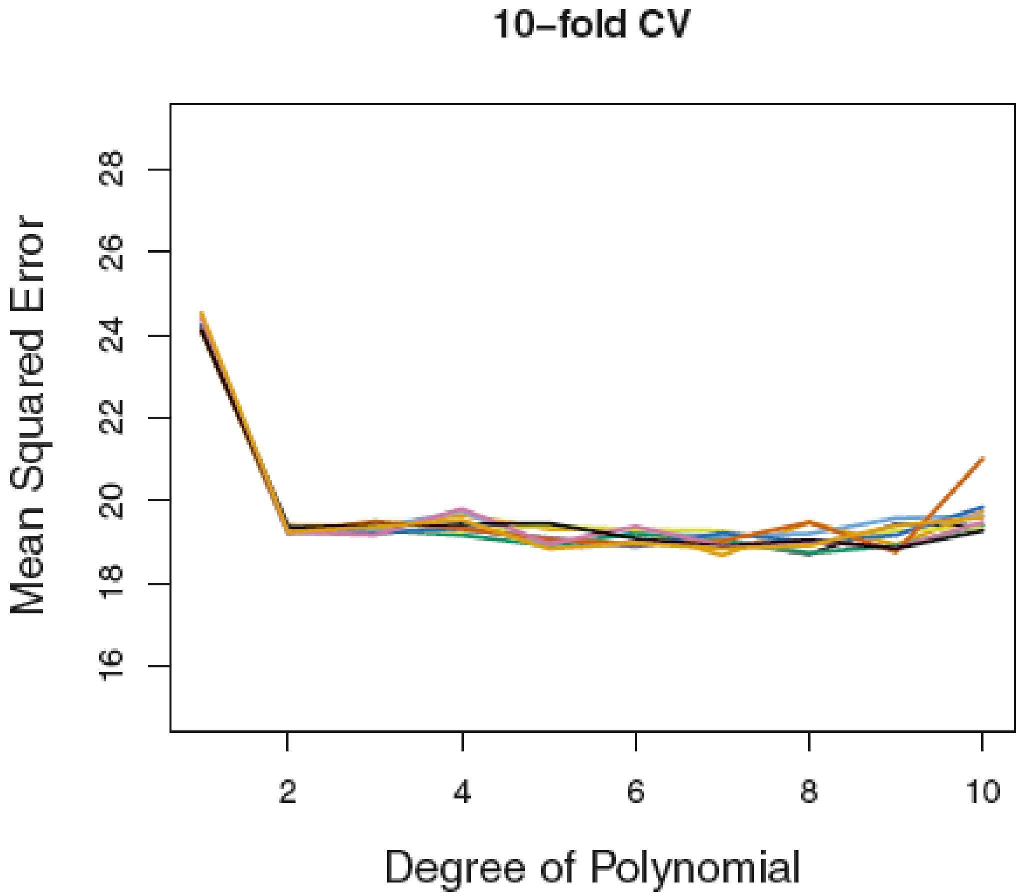


- *k*-fold CV test error estimate: $\mathrm{CV}_{(k)} = \frac{1}{k}\sum_{i=1}^{k} \mathrm{MSE}_i$

- Typical values for *k*: 5, 10
  - Computational advantage over LOOCV

k-Fold Cross-Validation (ii)

- Example: Auto data set
  - Some variability in the CV estimates, but much lower than with the validation set approach



k-Fold Cross-Validation (iii)

- Example: simulated data
  - Actual estimate of the test MSE
  - Location of the minimum point in the estimated test MSE curve

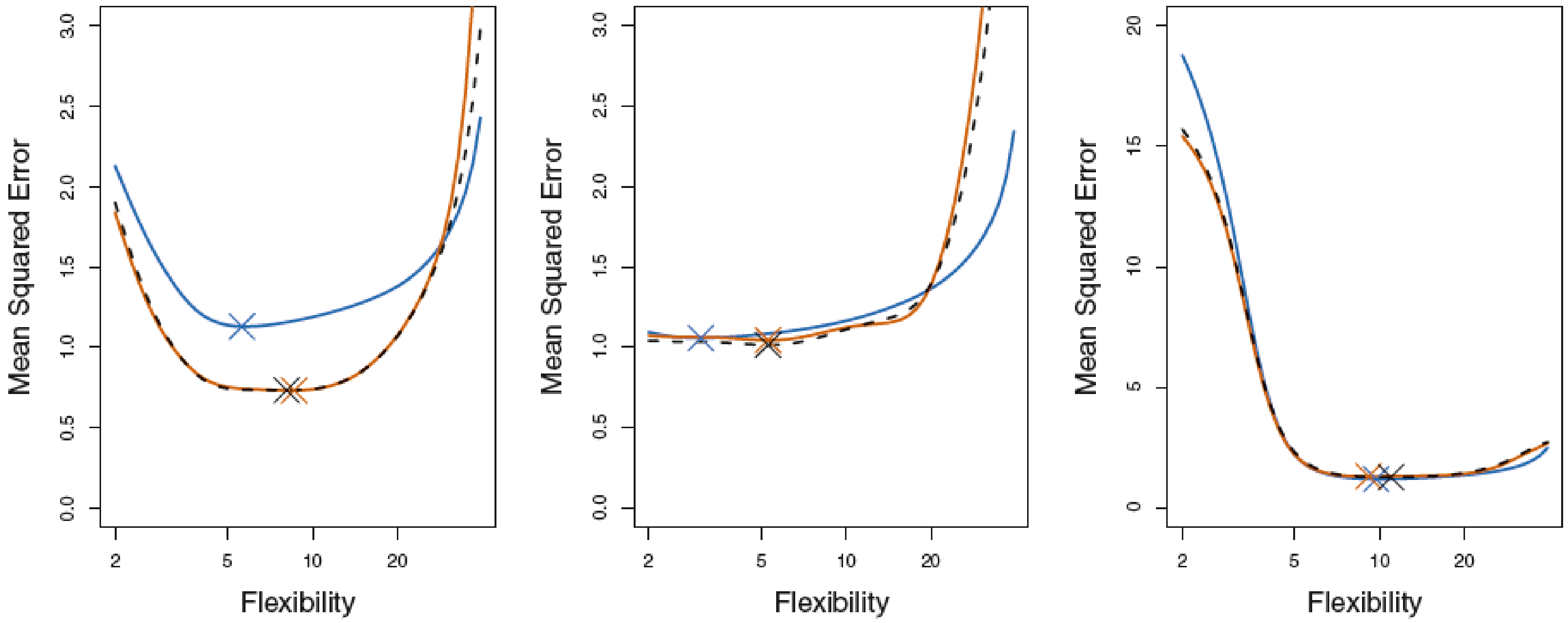


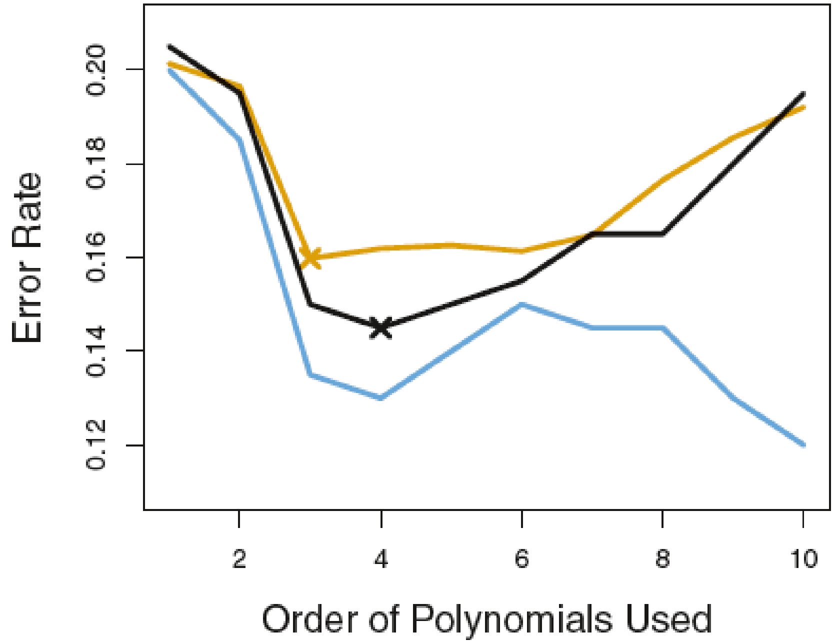
FIGURE 5.6. *True and estimated test MSE for the simulated data sets in Figures 2.9* (left)*, 2.10* (center)*, and 2.11* (right)*. The true test MSE is shown in blue, the LOOCV estimate is shown as a black dashed line, and the* 10*-fold CV estimate is shown in orange. The crosses indicate the minimum of each of the MSE curves.*

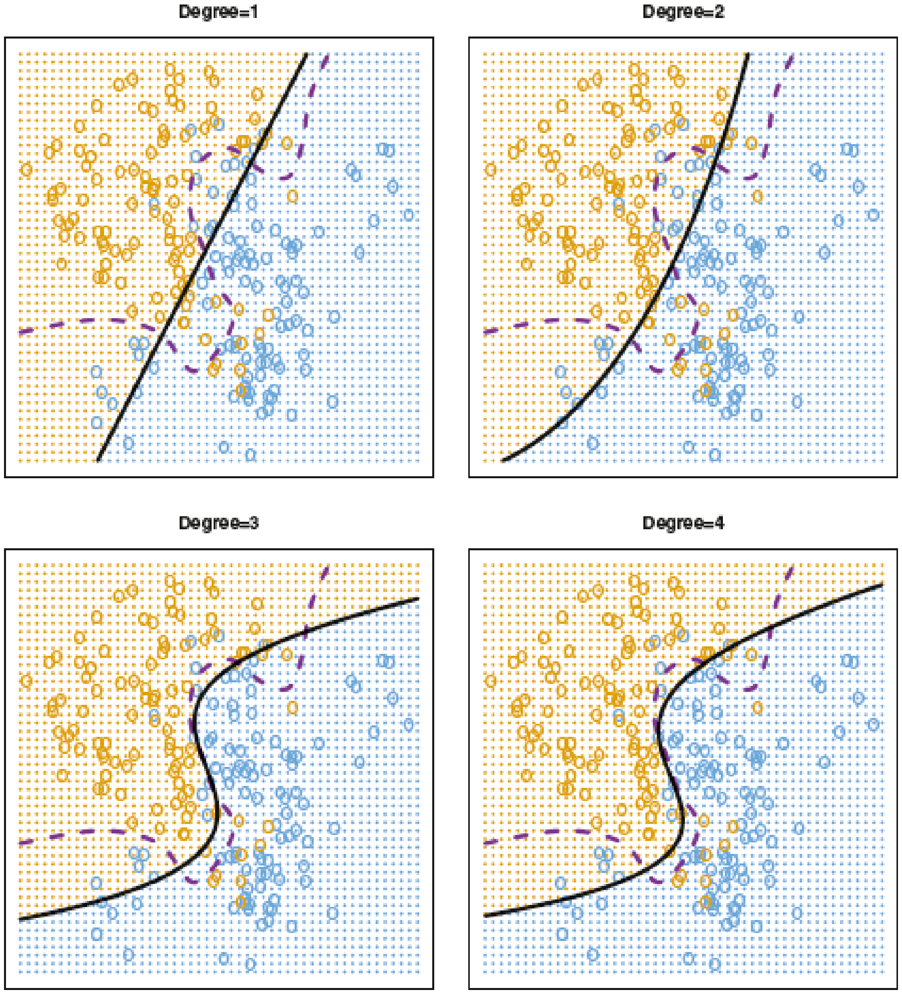
k-Fold Cross-Validation (iv)

- Bias:
  - Validation set approach: overestimates the test error, high bias
  - LOOCV: unbiased estimate of the test error
  - *k*-fold CV: intermediate level of bias
- Variance: LOOCV has a higher variance than *k*-fold CV
  - LOOCV:
    - high overlapping among training sets
    - We average the outputs, which are highly (positively) correlated
  - *k*-fold CV: models less correlated with each other
  - Mean of many highly correlated quantities has a higher variance
- Bias-variance trade-off: *k*-fold CV is better for $k=5$ or $k=10$
  - Empirically has shown test error estimates that suffer neither from high bias nor from very high variance

k-Fold Cross-Validation (v)

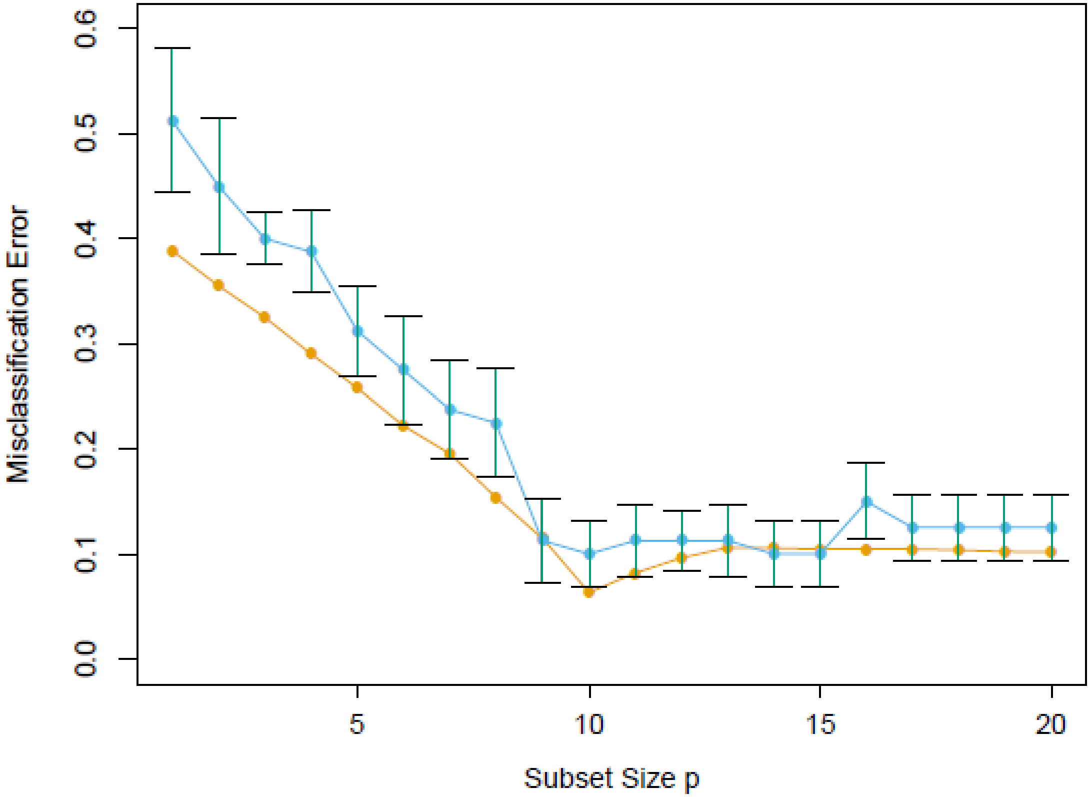
- Example: simulated data, two classes, two input variables
  - Bayer error rate: 0.133
  - Fit logistic regression with polynomials of different degrees
  - Training (blue), test (brown) and 10-fold CV (black) errors





k-Fold Cross-Validation (vi)

- One-standard-error rule:
  - Calculate the standard error of the estimated test MSE for each model
  - Select the simplest model for which the estimated test error is within one standard deviation of the lowest point in the curve
- Example:
  - Test error (orange)
  - CV error (blue)



Right Way to Do Cross-Validation

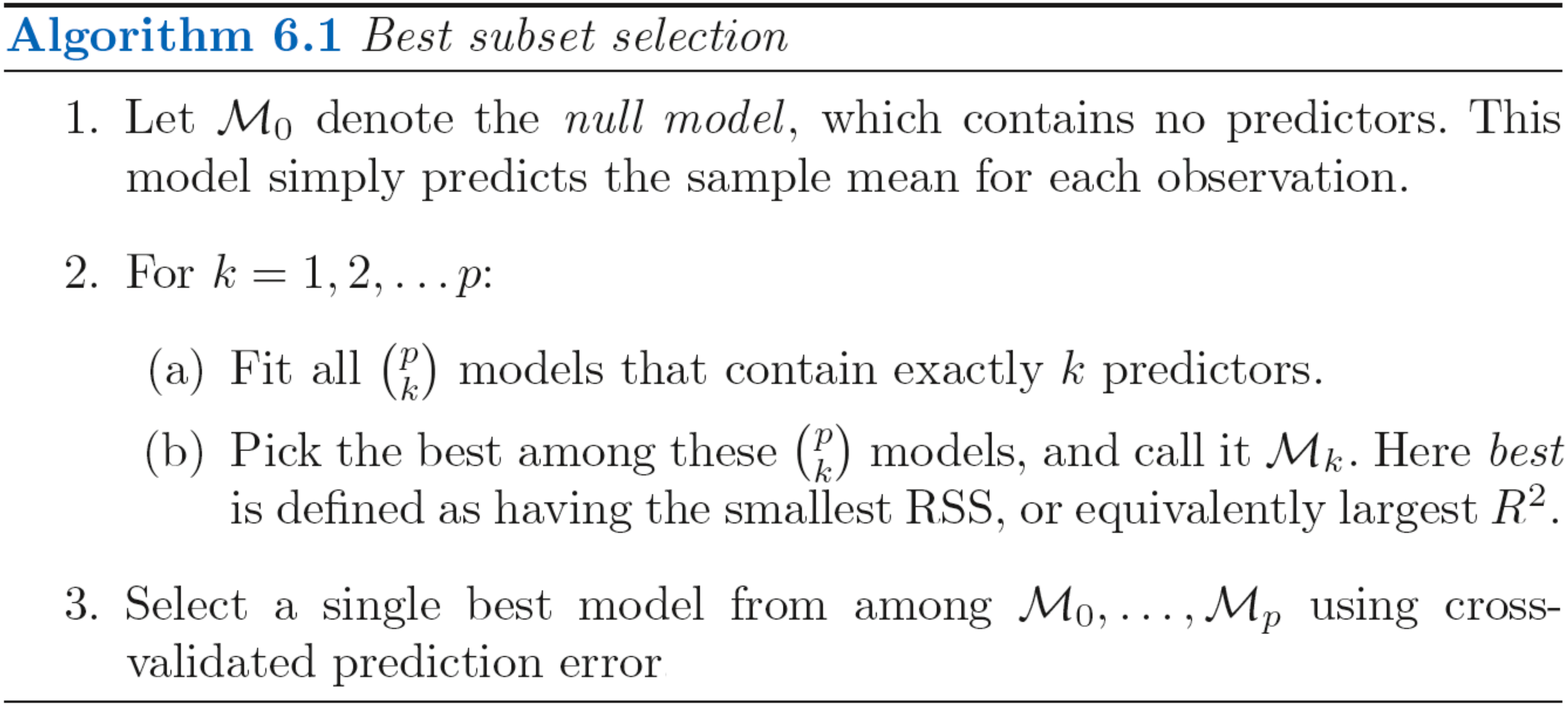
- If you do not do model selection:
  - Divide the samples into *K* folds
  - For each fold *k*:
    - Train the model with all the folds but *k*
    - Test the model with fold *k*
  - Calculate the mean and standard deviation of the estimated test error

Right Way to Do Cross-Validation (ii)

- If you do model selection: hyper-parameters have to be estimated (validation set)
  - Divide the data into training and test sets
  - Divide the training samples into *K* folds
  - For each combination of values of the hyper-parameters:
    - For each fold *k*:
      - Train the model with all the folds but *k*
      - Test the model with fold *k*
    - Calculate the mean and standard deviation of the validation error
  - Select the values of the hyper-parameters with the one-standard-error rule
  - Train the model with the selected hyper-parameters and using **all the samples of the training set**
  - Estimate the test error of the model with the test set

Subset Selection

- Best subset selection:

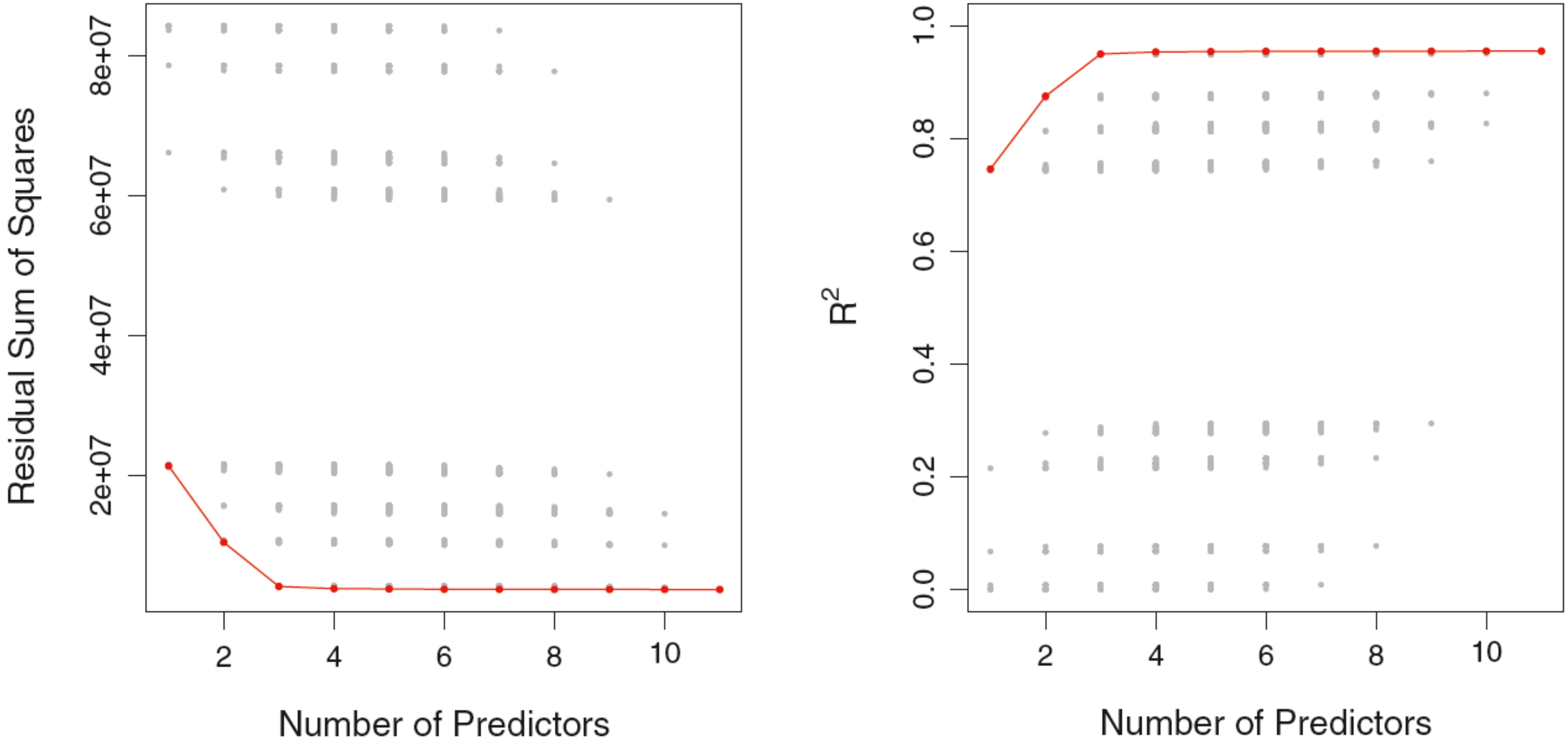


Algorithm 6.1 *Best subset selection*

1. Let $\mathcal{M}_0$ denote the *null model*, which contains no predictors. This model simply predicts the sample mean for each observation.
2. For $k = 1, 2, \ldots p$:
   (a) Fit all $\binom{p}{k}$ models that contain exactly k predictors.
   (b) Pick the best among these $\binom{p}{k}$ models, and call it $\mathcal{M}_k$. Here *best* is defined as having the smallest RSS, or equivalently largest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error

Subset Selection (ii)

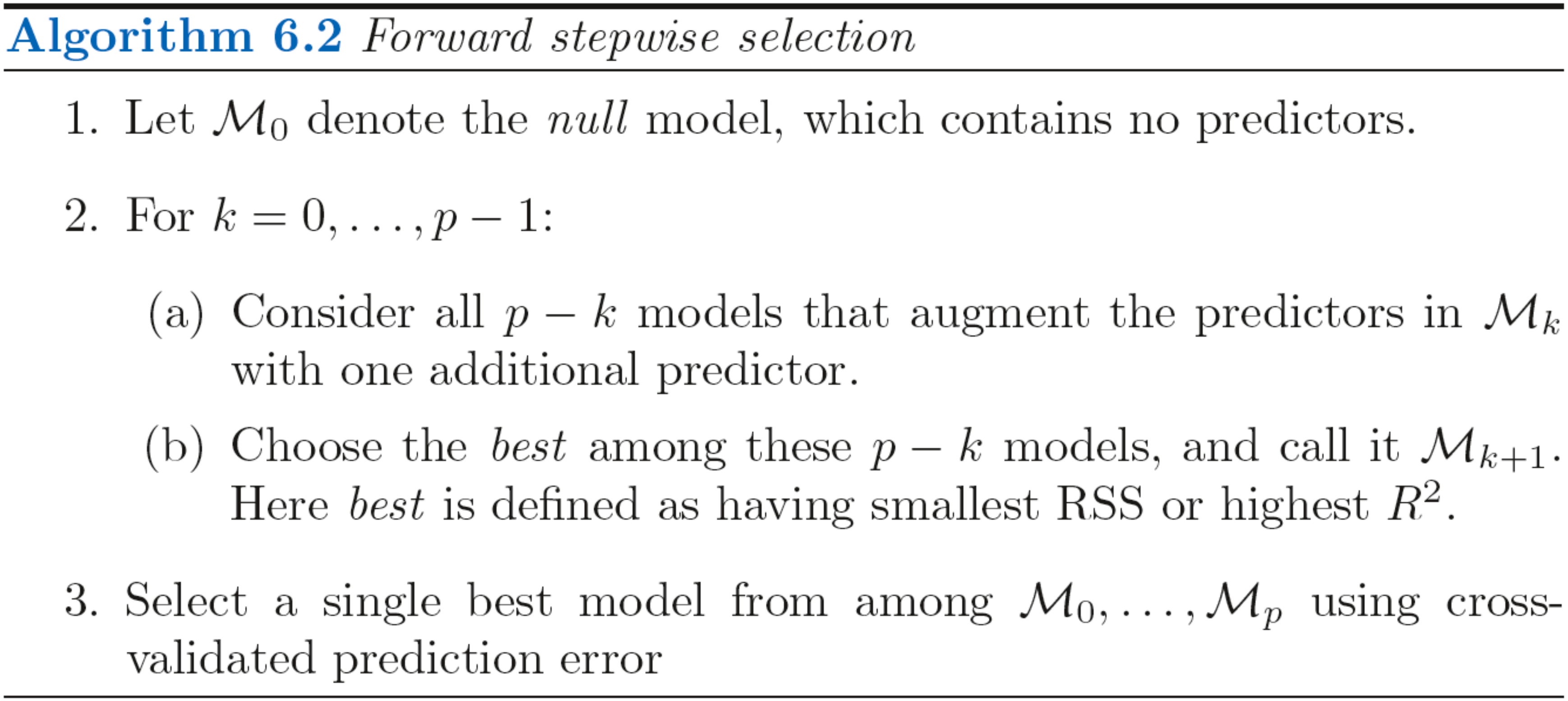
- Example: Credit dataset



- Best subset selection:
  - Computationally infeasible for large *p*: involves fitting 2^p models
  - Statistical problems for large *p*: overfitting and high variance

Subset Selection (iii)

- Forward stepwise selection



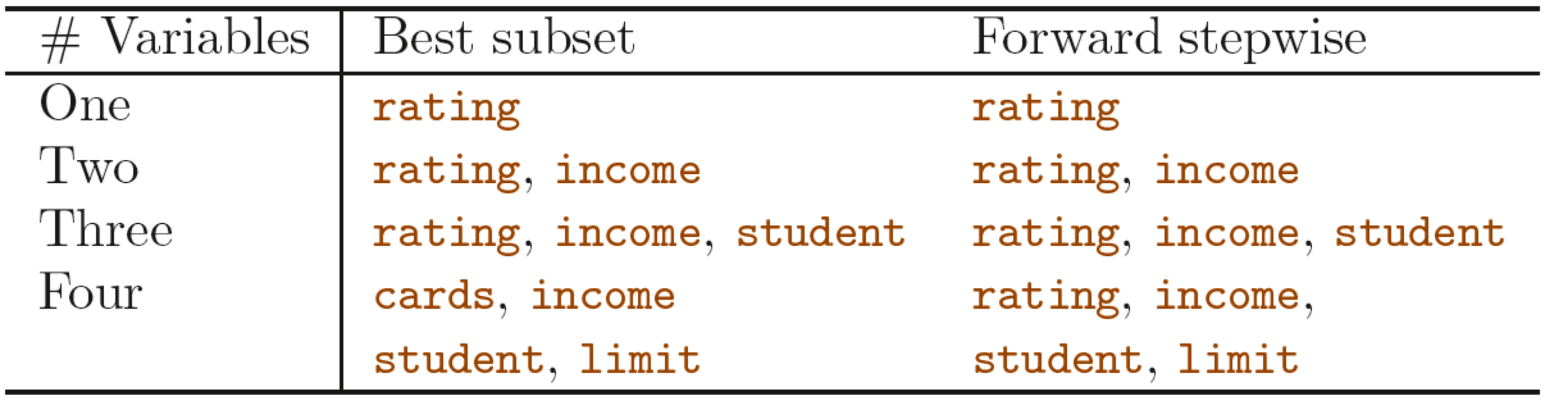
Algorithm 6.2 *Forward stepwise selection*

```
1. Let $\mathcal{M}_0$ denote the *null* model, which contains no predictors.
2. For $k = 0, \ldots, p-1$:
    (a) Consider all $p - k$ models that augment the predictors in $\mathcal{M}_k$
        with one additional predictor.
    (b) Choose the *best* among these $p - k$ models, and call it $\mathcal{M}_{k+1}$.
        Here *best* is defined as having smallest RSS or highest $R^2$.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error
```

  - Involves fitting 1+$\sum_{k=0}^{p-1}(p-k) = 1+p(p+1)/2$ models
  - No guarantee to yield the best model

Subset Selection (iv)

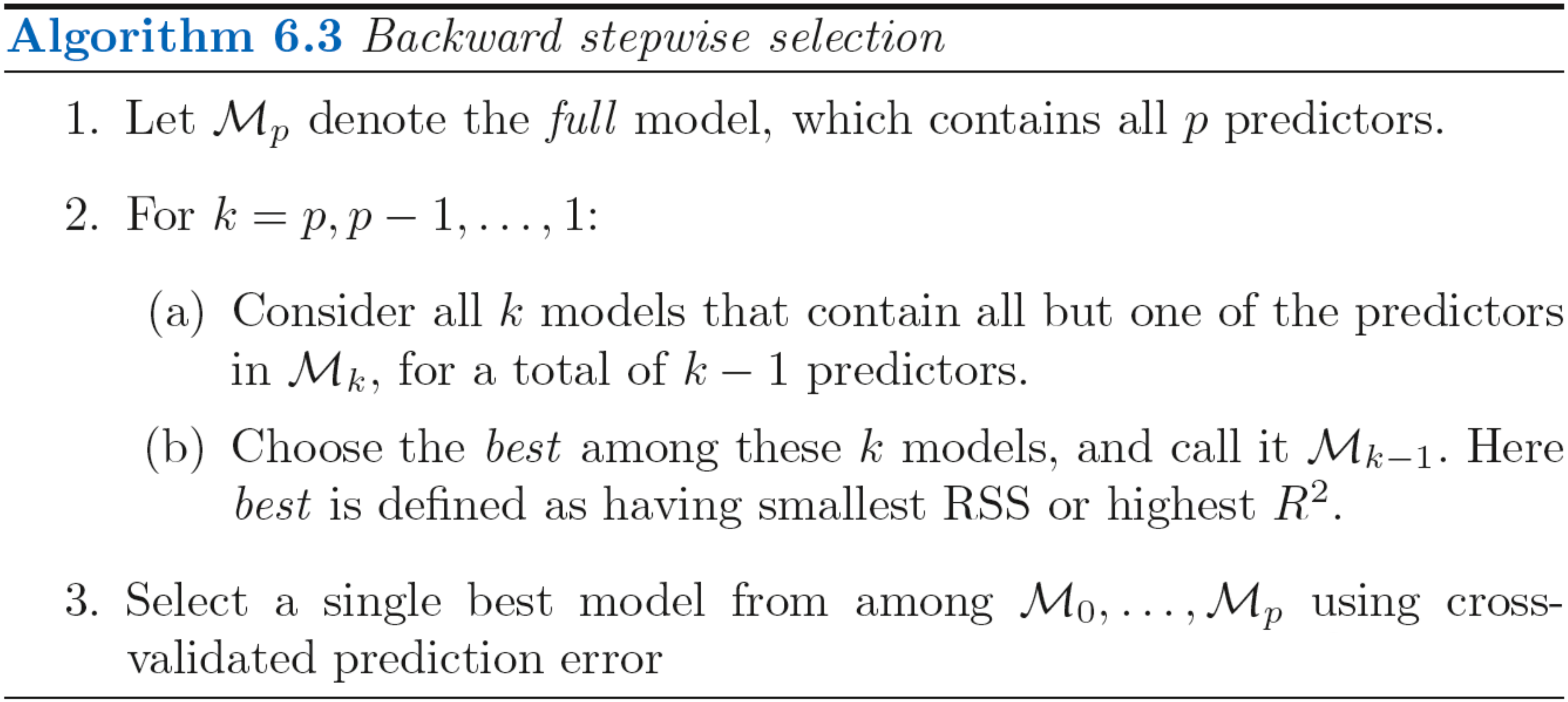
- Example: Credit dataset



# Variables	Best subset	Forward stepwise
One	rating	rating
Two	rating, income	rating, income
Three	rating, income, student	rating, income, student
Four	cards, income student, limit	rating, income, student, limit

Subset Selection (v)

- Backward stepwise selection



Algorithm 6.3 *Backward stepwise selection*

```
1. Let $\mathcal{M}_p$ denote the full model, which contains all $p$ predictors.
2. For $k = p, p-1, \ldots, 1$:
    (a) Consider all $k$ models that contain all but one of the predictors
        in $\mathcal{M}_k$, for a total of $k-1$ predictors.
    (b) Choose the best among these $k$ models, and call it $\mathcal{M}_{k-1}$. Here
        best is defined as having smallest RSS or highest $R^2$.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-
   validated prediction error
```

- Hybrid approaches
  - Add variables sequentially (forward), remove variables that do not improve the model (backward)

Reducing the error

- What can you do if your model makes large errors on the test set?

- High bias = underfit: high training error and high test error

- High variance = overfit: low training error and much higher test error

- High bias:
  - Adding new samples will not help
  - Add new meaningful features
  - Decrease regularization

Reducing the error (ii)

- High variance and low bias:
  - Add new samples
  - Reduce the number of features (features selection/extraction)
    - Helps if there are irrelevant features
  - Increase regularization

Bibliography

- G. James, D. Witten, T. Hastie, y R. Tibshirani, An Introduction to Statistical Learning with Applications in R. Springer, 2013.
  - Chapters 2, 5, 6 (sec. 6.1)
- T. Hastie, R. Tibshirani, y J. Friedman, The elements of statistical learning. Springer, 2009.
  - Chapters 7, 8 (sec. 8.2)