Resampling Methods

Chapter 5

March 20, 2018

1. 5.1. Validation set approach

2. 5.2. Cross validation

3. 5.3. Bootstrap

About Resampling

- An important statistical tool.
- Pretending the data as population and repeatedly draw sample from the data.
- Main task: assess the validity/accuracy of statistical methods and models.
    - **Cross-validation**: estimate the *test error* of models
    - **Bootstrap**: quantify the *uncertainty* of estimators

Validation and Cross validation

- Validation set approach.
- LOOCV (Leave-one-out cross valiation)
- K-fold cross validation.

Bootstrap

- Sampling with replacement (typically) n times, n is the sample size of data.
- Especially useful in statistical inference to quantify the uncertainty of estimates (can be even more accurate than normal approximation).
- All purpose resampling procedure.
- Used in ensemble methods of machine learning, for example,
  - **bagging**,
  - **random forest**.

Training error is not sufficient enough

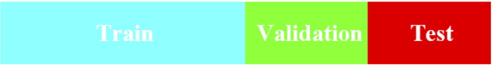
- training error easily computable with training data.
- because of possibility of over-fit, it cannot be used to properly assess test error.
- It is possible to "estimate" the test error, by, for example, making adjustments of the training error.
- The adjusted R-squared, AIC, BIC, etc serve this purpose.
- These methods rely on certain assumptions and are not for general purpose.

Test error: cross-validation

- Test error would be also easily computable, if test data are well designated.
- Normally we are just given ... data.
- Shall have to create “test data” for the purpose of computing test error.
- Artificially separate data into “training data” and “test data” for validation purpose is called cross-validation.
- The “test data” here should be more accurately called *validation data* or hold out data, meaning that they not used in training.
- Model fitting only uses the training data.

Ideal scenario for performance assessment

- In a “data-rich” scenario, we can afford to separate the data into three parts:
    - training data: used to train various models.
    - validation data: used to assess the models and identify the best.
    - test data: test the results of the best model.
- Usually, people also call validation data or hold-out data as test data.



Train Validation Test

Validation set approach

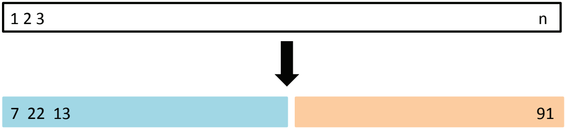


Figure: 5.1. A schematic display of the validation set approach. A set of n observations are randomly split into a training set (shown in blue, containing observations 7, 22, and 13, among others) and a validation set (shown in beige, and containing observation 91, among others). The statistical learning method is fit on the training set, and its performance is evaluated on the validation set.

Example: Auto Data

- A non-linear relationship between mpg and horsepower
- mpg $\sim$ horsepower + horsepower^2 is better than mpg $\sim$ horsepower.
- Should we add higher terms into the model?, like as cubic or even higher?
- One can check the p-values of regression coeffeicients to answer the question.
- In fact, a model selection problem, and we can use validation set approach.

Example: Auto Data

- randomly split the 392 observations into two sets:
- a training set containing 196 of the data points, and a validation set containing the remaining 196 observations.
- fit various regression models on the training sample
- The validation set error rates result from evaluating their performance on the validation sample.
- Here we MSE as a measure of validation set error, are shown in the left-hand panel of Figure 5.2.

Validation set approach

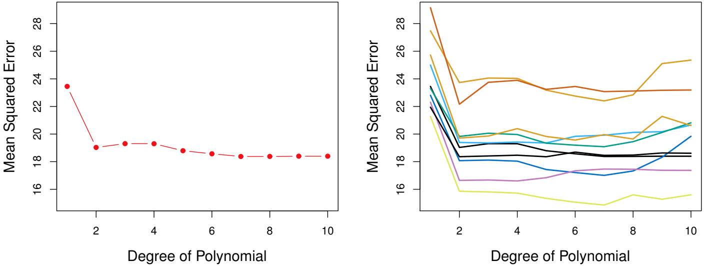


Figure: 5.2. The validation set approach was used on the Auto data set in order to estimate the test error that results from predicting mpg using polynomial functions of horsepower. Left: Validation error estimates for a single split into training and validation data sets. Right: The validation method was repeated ten times, each time using a different random split of the observations into a training set and a validation set. This illustrates the variability in the estimated test MSE that results from this approach.

Example: Auto Data

- The validation set MSE for the quadratic fit is considerably smaller than for the linear fit.
- validation set MSE for the cubic fit is actually slightly larger than for the quadratic fit.
- This implies that including a cubic term in the regression does NOT lead to better prediction than simply using a quadratic term.
- Repeat the process of randomly splitting the sample set into two parts, we will get a somewhat different estimate for the test MSE.

Example: Auto Data

- A quadratic term has a dramatically smaller validation set MSE than the model with only a linear term.
- Not much benefit in including cubic or higher-order polynomial terms in the model.
- Each of the ten curves results in a different test MSE estimate for each of the ten regression models considered.
- No consensus among the curves as to which model results in the smallest validation set MSE.
- Based on the variability among these curves, all that we can conclude with any confidence is that the linear fit is not adequate for this data.
- The validation set approach is conceptually simple and is easy to i

A summary

- The validation estimate of the test error rate can be highly variable, depending on the random split.
- Only a subset of the observations–the training set are used to fit the model.
- Statistical methods tend to perform worse when trained on fewer observations.
- The validation set error rate may tend to overestimate the test error rate for the model fit on the entire data set.

Cross validation: overcome the drawback of validation set approach

- Our ultimate goal is to produce the best model with best prediction accuracy.
- Validation set approach has a drawback of using ONLY training data to fit model.
- The validation data do not participate in model building but only model assessment.
- A “waste” of data.
- We need more data to participate in model building.

Another drawback of validation set approach

- It may over-estimate the test error for the model with all data used to fit.
- Statistical methods tend to perform worse when trained on fewer observations.
- The validation set error rate may tend to overestimate the test error rate for the model fit on the entire data set.
- The method of cross validation could overcome these drawbacks, by effectively using EVERY data point in model building!

The leave-one-out cross-validation

- Suppose the data contain n data points.
- First, pick data point 1 as validation set, the rest as training set. fit the model on the training set, evaluate the test error, on the validation set, denoted as say MSE_1.
- Second, pick data point 2 as validation set, the rest as training set. fit the model on the training set, evaluate the test error on the validation set, denoted as say MSE_2.
- ..... (repeat the procedure for all data point.)
- Obtain an estimate of the test error by combining the MSE_i, $i =, ..., n$:

$$\mathrm{CV}_{(n)} = \frac{1}{n} \sum_{i=1}^{n} \mathrm{MSE}_i$$

LOOCV

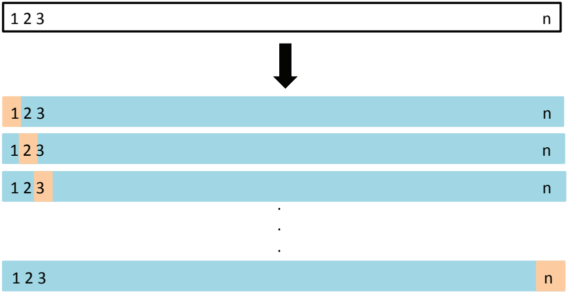


Figure: 5.3. A schematic display of LOOCV. A set of n data points is repeatedly split into a training set (shown in blue) containing all but one observation, and a validation set that contains only that observation (shown in beige). The test error is then estimated by averaging the n resulting MSE's. The first training set contains all but observation 1, the second training set contains all but observation 2, and so forth.

Advantages of LOOCV

- Far less bias, since the training data size $(n - 1)$ is close to the entire data size (n).
- One single test error estimate (thanks to the averaging), without the variablity validation set approach.
- A disadvantage: could be computationally expensive since the model need to be fit n times.
- The MSE_i may be too much correlated.

LOOCV applied to Auto data:

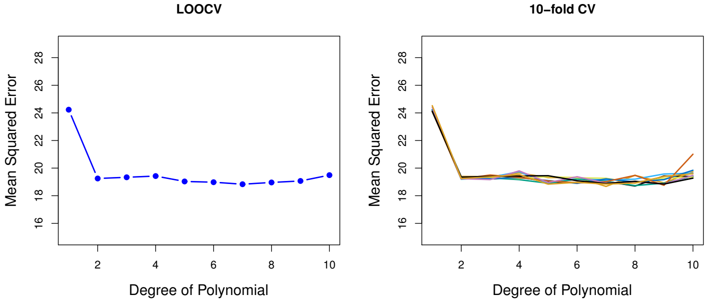


Figure: 5.4. Cross-validation was used on the Auto data set in order to estimate the test error that results from predicting mpg using polynomial functions of horsepower. Left: The LOOCV error curve. Right: 10-fold CV was run nine separate times, each with a different random split of the data into ten parts. The figure shows the nine slightly different CV error curves.

Complexity of LOOCV in linear model?

- Consinder linear model:

$$y_i = \mathbf{x}_i^T \beta + \epsilon_i, \qquad i = 1, .., n$$

  and the fitted values $y_i = \mathbf{x}_i^T \hat{\beta}$, where $\hat{\beta}$ is the least squares estimate of β based on all data $(\mathbf{x}_i, y_i), i = 1, ..., n$.

- Using LOOCV, the

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

  where $\hat{y}_i^{(i)} = \mathbf{x}_i^T \hat{\beta}^{(i)}$ is the model predictor of y_i based on the linear model fitted by all data except $(\mathbf{x}_i, y_i)$ (delete one), i.e., $\hat{\beta}^{(i)}$ is the least squares estimate of β based on all data but $(\mathbf{x}_i, y_i)$.

Simple Formula of LOOCV in linear model

- Looks to be complicated to compute least squares estimate n times.
- Easy formula:

$$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\Big(\frac{y_i - \hat{y}_i}{1 - h_i}\Big)^2$$

  where $\hat{y}_i$ is the fitted values of least squares method based on all data. h_i is the leverage.

Recall: Leverage

- Recall the hat matrix

$$\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T \quad \text{as } \hat{y} = \mathbf{H}y.$$

  Let $h_{ij} = \mathbf{x}_i^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_j$ be the (i, j) elements of $\mathbf{H}$.
- The leverage of the i-th observation is just the i-th diagonal element of $\mathbf{H}$, denoted as h_{ii}.
- A high leverage implies that observation is quite influential. Note that the average of h_{ii} is $(p+1)/n$.
- E.g., if h_{ii} is greater than $2(p+1)/n$, twice of the average, is generally considered large.

Fast computation of cross-validation I

- The leave-one-out cross-validation statistic is given by
$$CV = \frac{1}{N}\sum_{i=1}^{N} e_{[i]}^2,$$
where $e_{[i]} = y_i - \hat{y}_{[i]}$, the observations are given by $y_1, \ldots, y_N$, and $\hat{y}_{[i]}$ is the predicted value obtained when the model is estimated with the ith case deleted.
- Suppose we have a linear regression model $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{e}$. The $\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$ and $\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ is the hat matrix. It has this name because it is used to compute $\hat{\mathbf{Y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{H}\mathbf{Y}$. If the diagonal values of $\mathbf{H}$ are denoted by $h_1, \ldots, h_N$, then the leave-one-oout cross-validation statistic can be computed using
$$CV = \frac{1}{N}\sum_{i=1}^{N} [e_i/(1-h_i)]^2,$$
where $e_i = y_i - \hat{y}_i$ is predicted value obtained when the model is estimated with all data included.

Fast computation of cross-validation II

Proof

- Let $\mathbf{X}_{[i]}$ and $\mathbf{Y}_{[i]}$ be similar to $\mathbf{X}$ and $\mathbf{Y}$ but with the ith row deleted in each case. Let $\mathbf{x}_i^T$ be the ith row of $\mathbf{X}$ and let

$$\hat{\boldsymbol{\beta}}_{[i]} = (\mathbf{X}_{[i]}^T \mathbf{X}_{[i]})^{-1} \mathbf{X}_{[i]}^T \mathbf{Y}_{[i]}$$

be the estimate of $\boldsymbol{\beta}$ without the ith case. Then $e_{[i]} = y_i - \mathbf{x}_i^T \hat{\boldsymbol{\beta}}_{[i]}$.

- Now $\mathbf{X}_{[i]}^T \mathbf{X}_{[i]} = (\mathbf{X}^T\mathbf{X} - \mathbf{x}_i\mathbf{x}_i^T)$ and $\mathbf{x}_i^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_i = h_i$. So by the Sherman-Morrison-Woodbury formula,

$$(\mathbf{X}_{[i]}^T \mathbf{X}_{[i]})^{-1} = (\mathbf{X}^T\mathbf{X})^{-1} + \frac{(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_i\mathbf{x}_i^T(\mathbf{X}^T\mathbf{X})^{-1}}{1 - h_i}.$$

Fast computation of cross-validation III

Proof

- Also note that $\mathbf{X}_{[i]}^T \mathbf{Y}_{[i]} = \mathbf{X}^T\mathbf{Y} - \mathbf{x}y_i$. Therefore

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_{[i]} &= \left[(\mathbf{X}^T\mathbf{X})^{-1} + \frac{(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_i\mathbf{x}_i^T(\mathbf{X}^T\mathbf{X})^{-1}}{1-h_i}\right](\mathbf{X}^T\mathbf{Y} - \mathbf{x}_i y_i) \\
&= \hat{\boldsymbol{\beta}} - \left[\frac{(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_i}{1-h_i}\right][y_i(1-h_i) - \mathbf{x}_i^T\hat{\boldsymbol{\beta}} + h_i y_i] \\
&= \hat{\boldsymbol{\beta}} - (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_i e_i/(1-h_i)
\end{aligned}$$

- Thus

$$\begin{aligned}
e_{[i]} &= y_i - \mathbf{x}_i^T\hat{\boldsymbol{\beta}}_{[i]} \\
&= y_i - \mathbf{x}_i^T\left[\hat{\boldsymbol{\beta}} - (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_i e_i/(1-h_i)\right] \\
&= e_i + h_i e_i/(1-h_i) = e_i/(1-h_i)
\end{aligned}$$

Simplicity of LOOCV in linear model

- One fit (with all data) does it all!
- The prediction error rate (in terms of MSE) is just weighted average of the least squares fit residuals.
- High leverage point gets more weight in prediction error estimation.

K-fold cross validation

- Divide the data into K subsets, usually of equal or similar sizes (n/K).
- Treat one subset as validation set, the rest together as a training set. Run the model fitting on training set. Calculate the test error estimate on the validation set, denoted as MSE_i, say.
- Repeat the procedures over every subset.
- Average over the above K estimates of the test errors, and obtain

$$\mathrm{CV}_{(K)} = \frac{1}{K}\sum_{i=1}^{K}\mathrm{MSE}_i$$

- LOOCV is a special case of K-fold cross validation, actually n-fold cross validation.

K-fold cross validation

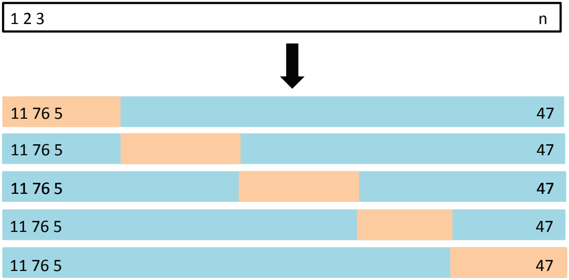


Figure: 5.5. A schematic display of 5-fold CV. A set of n observations is randomly split into five non-overlapping groups. Each of these fifths acts as a validation set (shown in beige), and the remainder as a training set (shown in blue). The test error is estimated by averaging the five resulting MSE estimates.

K-fold cross validation

- Common choices of K: $K = 5$ or $K = 10$.
- Advantage over LOOCV: 1. computationally lighter, espeically for complex model with large data. 2. Likely less variance (to be addressed later)
- Advantage over validation set approach: Less variability resulting from the data-split, thanks to the averaging.

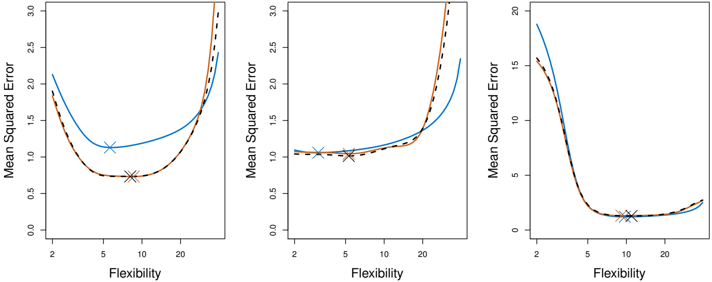


Figure: 5.6. True and estimated test MSE for the simulated data sets in Figures 2.9 (left), 2.10 (center), and 2.11 (right). The true test MSE is shown in blue, the LOOCV estimate is shown as a black dashed line, and the 10-fold CV estimate is shown in orange. The crosses indicate the minimum of each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

Figure 2.9

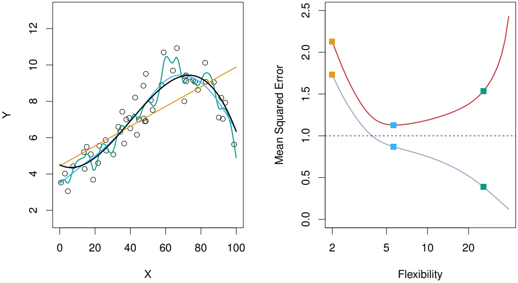


Figure: 2.9. Left: Data simulated from f, shown in black. Three estimates of f are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves). Right: Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line).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

Figure 2.10

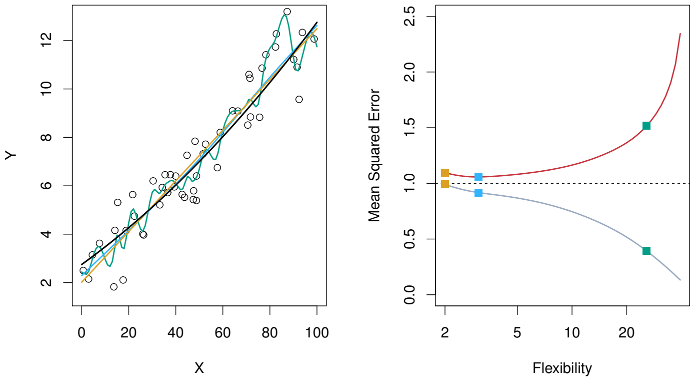


Figure: 2.10. Details are as in Figure 2.9, using a different true f that is much closer to linear. In this setting, linear regression provides a very good fit to the data.

Figure 2.11

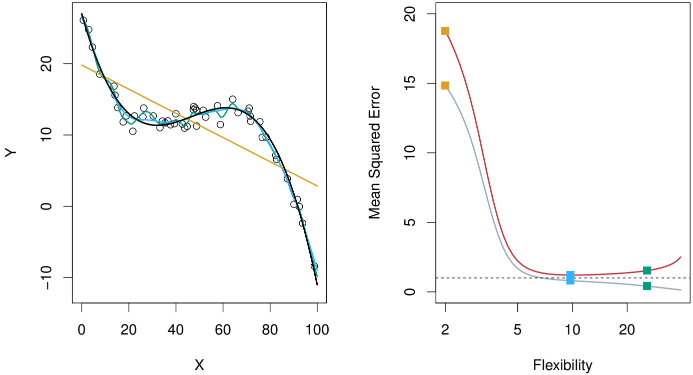


Figure: 2.11. Details are as in Figure 2.9, using a different f that is far from linear. In this setting, linear regression provides a very poor fit to the data.

Special interests in the complexity parameter at miminum test error

- A family of models indexed by a parameter, usually represeting flexibity or complexity of models.
- Such parameter is often called tuning parameter; it could even be a number of variables.
- Example: Order of polynomials of horsepower in the Auto data example.
- Example: Penalization parameters in ridge, lasso, etc (to be addressed in the next chapter.)
- Intend to find the best model within this family, i.e., to find the value of this tuning parameter.
- Care less of the actual value of the test error.
- In the above simulated data, all of the CV curves come close to identifying the correct level of flexibility.

Bias variance trade-off

- In terms of bias of estimation of test error: validation set appraoch has more bias due to smaller size of training data; LOOCV is nearly unbiased; K-fold (e.g, K=5 or 10) has intermediate bias.
- In view of bias, LOOCV is most preferred; and K-fold cross validation next.
- But, K-fold cross validation has smaller variance than that of LOOCV.
- The n traing sets LOOCV are too similar to each other. As a result, the trained models are too postively correlated.
- The K training sets of K-fold cross validation are much less similar to each other.
- As a result, the K-fold cross validation generally has less variance than LOOCV.

Cross validation for classification

- MSE is a popular criterio to measure predition/estimation accuracy for regression.
- There are other criteria.
- For classification with qualitative response, a natural choice is: 1 for incorrect classification and 0 for correct classification.
- For LOOCV, this leads to $\mathrm{Err}_i = I(y_i \neq \hat{y}_i^{(i)})$, where $y_i^{(i)}$ is the classification of i-th observation based on model fitted not using i-th observation.
- Then,

$$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n} Err_i$$

which is just the average number of incorrect classifaciton..

Example

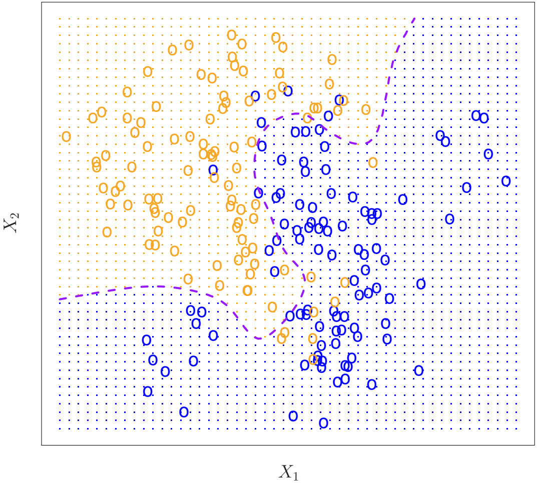


Figure 2.13. A simulated data set consisting of 100 observations in each of two groups, indicated in blue and in orange. The purple dashed line represents the Bayes decision boundary. The orange background grid indicates the region in which a test observation will be assigned to the orange class, and the blue background grid indicates the region in which a test observation will be assigned to the blue class.

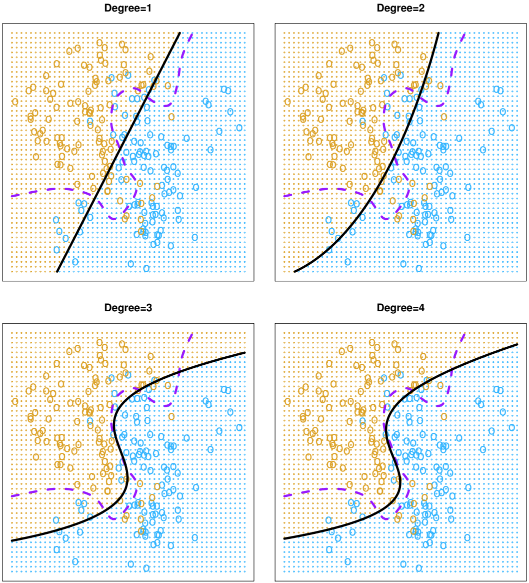

Degree=1
Degree=2
Degree=3
Degree=4

FIGURE 5.7. Logistic regression fits on the two-dimensional classification data displayed in Figure 2.13. The Bayes decision boundary is represented using a purple dashed line. Estimated decision boundaries from linear, quadratic, cubic and quartic (degrees 1 to 4) logistic regressions are displayed in black. The (TRUE) test error rates for the four logistic regression fits are respectively 0.201, 0.197, 0.160, and 0.162, while the Bayes error rate is 0.133.

Remark about the simulated example.

- The previous example is simulated.
- The true population distribution is known.
- The figures 0.201, 0.197, 0.160, and 0.162, and 0.133 (for Bayes error rate) are the true test error, computed based on true population distribution.
- In practice the true population distribution is unknown. Thus the true test error cannot be computed.
- We use cross validation to solve the problem.

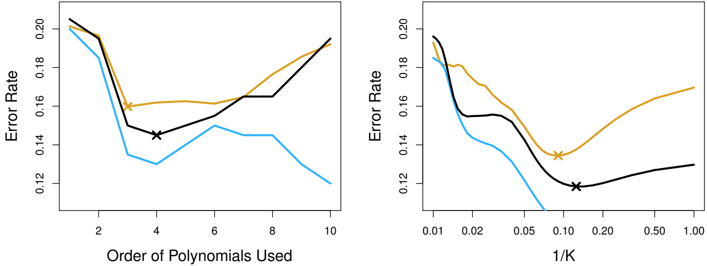


Figure: 5.8. Test error (brown), training error (blue), and 10-fold CV error (black) on the two-dimensional classification data displayed in Figure 5.7. Left: Logistic regression using polynomial functions of the predictors. The order of the polynomials used is displayed on the x-axis. Right: The KNN classifier with different values of K, the number of neighbors used in the KNN classifier.

- Training error declines in general when model complexity increases.
- Some times even reaches 0.
- Test error general declines first and then increases.
- 10-fold cross validation provides reasonable estimate of the test error, with slight under-estimation.

Boostrap as a resampling procedure.

- Suppose we have data $x_1, ..., x_n$, representing the ages of n randomly selected people in HK.
- Use sample mean $\bar{x}$ to estimate the population mean μ, the avearge age of all residents of HK.
- How to justify the estimation error $\bar{x} - \mu$? Usually by t-confidence interval, test of hypothesis.
- They rely on normality assumption or central limit theorm.
- Is there another reliable way?
- Just bootstrap:

Boostrap as a resampling procedure.

- Take n random sample (with replacement) from $x_1, ..., x_n$.
- calculate the sample mean of the “re-sample”, denoted as $\bar{x}_1^*$.
- Repeat the above a large number M times. We have $\bar{x}_1^*, \bar{x}_2^*, ..., \bar{x}_M^*$.
- Use the distribution of $\bar{x}_1^* - \bar{x}, ..., \bar{x}_M^* - \bar{x}$ to approximate that of $\bar{x} - \mu$.

- Essential idea: Treat the data distribtion (more professionally called empirical distributoin) as a proxy of the population distribution.
- Mimic the data generation from the true population, by trying resampling from the empirical distribution.
- Mimic your statistical procedure (such as computing an estimate $\bar{x}$) on data, by doing the same on the resampled data.
- Evalute your statistical procedure (which may be difficult because it involves randomness and the unknown population distribution) by evaluting your analogue procudres on the re-samples.

Example

- X and Y are two random variables. Then minimizer of $\mathrm{var}(\alpha X + (1-\alpha)Y))$ is

$$\alpha = \frac{\sigma_Y^2 - \sigma_{XY}}{\sigma_X^2 + \sigma_Y^2 - 2\sigma_{XY}}$$

- Data: $(X_1, Y_1), ..., (X_n, Y_n)$.
- We can compute sample variances and covariances.
- Estimate α by

$$\hat{\alpha} = \frac{\hat{\sigma}_Y^2 - \hat{\sigma}_{XY}}{\hat{\sigma}_X^2 + \hat{\sigma}_Y^2 - 2\hat{\sigma}_{XY}}$$

- How to evalute $\hat{\alpha} - \alpha$, (remember $\hat{\alpha}$ is random and α is unknown).
- Use Bootstrap

Example

- Sample n resamples from $(X_1, Y_1), ..., (X_n, Y_n)$, and compute the sample the sample variance and covariances for this resample. And then compute
$$\hat{\alpha}^* = \frac{(\hat{\sigma}_Y^*)^2 - \hat{\sigma}_{XY}^*}{(\hat{\sigma}_X^*)^2 + (\hat{\sigma}_Y^*)^2 - 2\hat{\sigma}_{XY}^*}$$
- Repeat this procedure, and we have $\hat{\alpha}_1^*, ..., \hat{\alpha}_M^*$ for a large M.
- Use the distribution of $\hat{\alpha}_1^* - \hat{\alpha}, ..., \hat{\alpha}_M^* - \hat{\alpha}$ to approximate the distribution of $\hat{\alpha} - \alpha$.
- For example, we can use
$$\frac{1}{M}\sum_{j=1}^{M}(\hat{\alpha}_j^* - \hat{\alpha})^2$$
to estimate $E(\hat{\alpha} - \alpha)^2$.
- Use Bootstrap

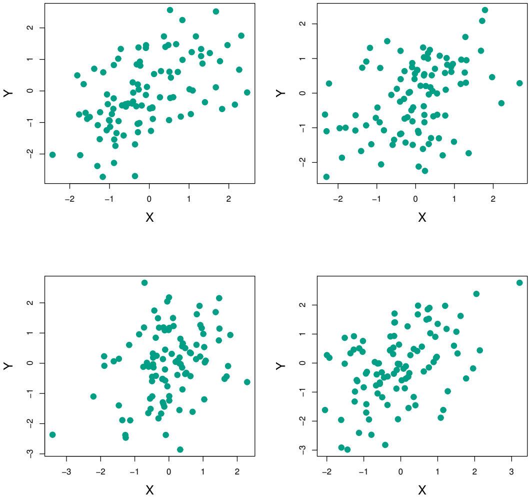


Figure: 5.9. Each panel displays 100 simulated returns for investments X and Y. From left to right and top to bottom, the resulting estimates for α are 0.576, 0.532, 0.657, and 0.651.

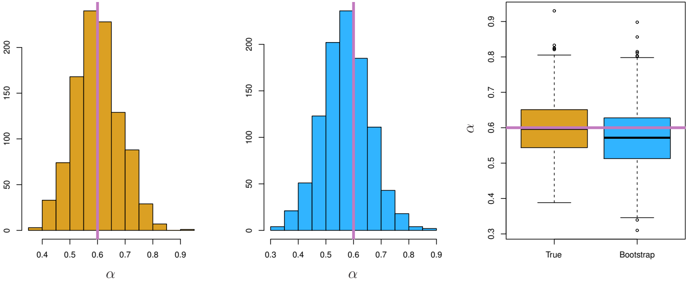
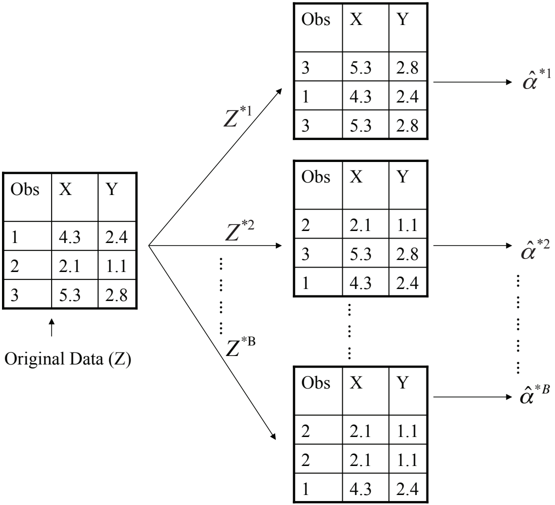


Figure: 5.10. Left: A histogram of the estimates of α obtained by generating 1,000 simulated data sets from the true population. Center: A histogram of the estimates of α obtained from 1,000 bootstrap samples from a single data set. Right: The estimates of α displayed in the left and center panels are shown as boxplots. In each panel, the pink line indicates the true value of α.



Obs	X	Y
1	4.3	2.4
2	2.1	1.1
3	5.3	2.8

Obs	X	Y
3	5.3	2.8
1	4.3	2.4
3	5.3	2.8

Obs	X	Y
2	2.1	1.1
3	5.3	2.8
1	4.3	2.4

Obs	X	Y
2	2.1	1.1
2	2.1	1.1
1	4.3	2.4

Figure 5.11. A graphical illustration of the bootstrap approach on a small sample containing $n = 3$ observations. Each bootstrap data set contains n observations, sampled with replacement from the original data set. Each bootstrap data set is used to obtain an estimate of α.

Exercises

Exercise 5.4 of ISLR (print 7): Exercises 1, 2, 5, 6, and 8* (* is a Bonus question, optional).

End of Chapter 5.