SRM Notes

dex

November 13, 2022

1 Introduction to Statistical Learning

1.1 ISLR Chapter 2

There is a prediction accuracy and model interpretability trade-off. Inflexible models are often more interpretable than flexible models. The most commonly used measure for quality of fit is the mean squared error

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

This is known as the training MSE and $\hat{f}$ is typically found by minimizing training MSE. However, the ideal $\hat{f}$ minimizes test MSE. The test MSE can be estimated by cross-validation. Usually the test MSE follows a U-shape where as the flexibility of the learning method increases, the test MSE decreases until it starts to increase again.

The expected test MSE can be decomposed into the following form:

$$E(y_0 - \hat{f}(x_0))^2 = \text{var}(\hat{f}(x_0)) + [\text{Bias}(\hat{f}(x_0))]^2 + \text{var}(\epsilon)$$

The ideal learning method achieves low variance and low bias. The final term $\text{var}(\epsilon)$ is said to be an irreducible error and this term can never be lowered and serves as a lower bound to the expected test MSE. Bias is introduced by approximating a complicated relationship by a simple model. Generally, as the flexibility of the model increases, the bias will decrease. Variance refers to the amount by which $\hat{f}$ changes if it is given a different training data set. In general, more flexible statistical methods have higher variance.

In the classification setting the training MSE is replaced with the training error rate defined as

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

Similarly to the regression setting, the ideal classifier minimizes test error rate. The expected test error rate is minimized by the Bayes classifier which assigns a test observation with predictor x_0 to the class j for which $P(Y = j \mid X = x_0)$ is largest. If there are only two classes, then the Bayes classifier corresponds to $P(Y = 1 \mid X = x_0) > 0.5$. The set for which the conditional probability is exactly 0.5 is called the Bayes decision boundary. The Bayes classifier produces the lowest possible test error rate called the Bayes error rate. The Bayes error rate is given by

$$1 - E\left(\max_j P(Y = j \mid X)\right)$$

Since for real data we do not know the conditional distribution, the actual Bayes classifier cannot be used but may be treated as the gold standard to compare other classification methods. We may instead try to classify observations with an estimated conditional distribution and one such method is the K-nearest neighbors (KNN) classifier.

$$\hat{P}(Y = j \mid X = x_0) = \frac{1}{K} \sum_{i \in \mathcal{N}_0} 1\{y_i = j\}$$

where $\mathcal{N}_0$ refers to the K points in the training data closest to x_0. KNN classifies the test observation x_0 to the class with the largest estimated conditional probability. The choice of K corresponds to the smoothness of the classifier, with higher K corresponding to lower bias and higher variance and lower K to higher bias but lower variance.

1.2 ISLR Chapter 3

1.2.1 Simple linear regression

Simple linear regression assumes the form

$$Y = \beta_0 + \beta_1 X + \epsilon$$

Estimates of β_i's are found by minimizing the residual sum of squares (RSS)

$$\text{RSS} = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

The minimizers are

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$
$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

where $\bar{y} = \frac{1}{n}\sum_{i=1}^{n}$, $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$.

$$\text{se}(\hat{\beta}_0)^2 = \sigma^2 \left[\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \right], \quad \text{se}(\hat{\beta}_1)^2 = \frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

where $\sigma^2 = \text{var}(\epsilon)$. These formulas are derived assuming the errors to be uncorrelated with common variance σ^2. Note that $\text{se}(\hat{\beta}_1)$ is smaller when the x_i are more spread out, which intuitively follows from there being more leverage to estimate the slope in this case. In general, σ^2 is not known but can be estimated from the data. This estimate of σ is known as the residual standard error given by the formula

$$\text{RSE} = \sqrt{\frac{\text{RSS}}{n-2}}$$

Standard errors can also be used to compute Wald confidence intervals. We can also perform hypothesis tests on the coefficients. To test the null hypothesis that there is no relationship between X and Y, we can test

$$H_0 : \beta_1 = 0, \quad H_1 = \beta_1 \neq 0$$

We compute a t-statistic given by

$$t = \frac{\hat{\beta}_1}{\text{SE}(\hat{\beta}_1)}$$

this follows a t_{n-2} distribution.

To quantify the extent to which the model fits the data, we can look at two related quantities: the RSE and the R^2 statistic. The RSE provides an absolute measure of lack of fit of the model to the data. The R^2 states the proportion of variance explained.

$$R^2 = \frac{\text{TSS} - \text{RSS}}{\text{TSS}} = 1 - \frac{\text{RSS}}{\text{TSS}}$$

where $\text{TSS} = \sum_{i=1}^{n}(y_i - \bar{y})^2$ is the total sum of squares. The statistic R^2 always lies between 0 and 1. It is also related to the correlation. If r is the sample correlation between X and Y, in the case of simple linear regression, $R^2 = r^2$.

1.2.2 Multiple Linear Regression

The multiple linear regression model takes the form

$$Y = \beta_0 + \left(\sum_{k=1}^{p} \beta_k X_k\right) + \epsilon$$

where X_j represents the jth predictor and β_j quantifies the association between that variable and the response. We interpret β_j as the average effect on Y of a unit increase in X_j holding all other predictors fixed.

Given estimators $\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_p$, predictions take the form

$$\hat{y} = \hat{\beta}_0 + \sum_{k=1}^{p} \hat{\beta}_k x_k$$

Multiple linear regression minimizes the sum of squared residuals

$$\text{RSS} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \sum_{k=1}^{p} \hat{\beta}_k x_{ik})^2$$

Given a full rank design matrix X, the OLS estimator is given by the equation

$$\hat{\beta}_{\text{OLS}} = (X^T X)^{-1} X^T Y$$

To answer if there is a relationship between the response and predictors we can take the hypothesis test with null hypothesis

$$H_0 : \beta_1 = \beta_2 = \cdots = \beta_p = 0$$

with the alternative hypothesis being that at least one β_k for $k > 0$ is not zero. For this purpose, we compute the F-statistic

$$F = \frac{(\text{TSS} - \text{RSS})/p}{\text{RSS}/(n - p - 1)}$$

where $\text{TSS} = \sum_{i=1}^{n}(y_i - \bar{y})^2$. If the model assumptions are correct, then

$$E[\text{RSS}/(n - p - 1)] = \sigma^2$$

and if H_0 holds, then

$$E[(\text{TSS} - \text{RSS})/p] = \sigma^2$$

Hence if there is no relationship between the response and predictors, we expect the F-statistic to be close to 1. If the alternative hypothesis holds, then $E[(\text{TSS} - \text{RSS})/p] > \sigma^2$ so we expect F to be greater than 1.

Sometimes we would like to instead test that a particular subset of q of the coefficients are zero. That is,

$$H_0 : \beta_{p-q+1} = \beta_{p-q+2} = \cdots = \beta_p = 0$$

Let RSS_0 denote the residual sum of squares that use all variables except the last q. Then we take the F-statistic

$$F = \frac{(\text{RSS}_0 - \text{RSS})/q}{\text{RSS}/(n-p-1)}$$

This F-statistic reports the partial effect of adding these q variables. In all of these cases, for a nested model with $p_1 < p_2$, we compare with the F distribution with $(p_2 - p_1, n - p_2 - 1)$ degrees of freedom.

Variable selection is the task of determining which predictors are associated with the response in order to fit a single model involving only those predictors. Various statistics can be used to jduge the quality of a model including Mallow's C_p, Akaike information criterion (AIC), Bayesian information criterion (BIC), and adjusted R^2. Since there are a total of 2^p models that contain subsets of p variables, it is infeasible to try every subset. More efficient approaches must be taken. There are three classical approaches:

1. *Forward selection.* Begin with the null model then fit p simple linear regressions and add to the null model the variable that results in the lowest RSS. We then add to that model the variable that results in the lowest RSS for the new two-variable model and continue until some stopping rule is satisfied.
2. *Backward selection.* We start with all variables in the model and remove the variable with the largest p-value. The new $(p-1)$-variable model is fit and the variable with the largest p-value is removed and we continue this procedure until a stopping rule is reached.
3. *Mixed selection.* This is a combination of forward and backward selection. Start with no variables, then add the variables that provide the best fit one-by-one. If at any point the p-value for one of the variables in the model rises above a threshold, we remove that variable from the model and we continue to perform forward and backward steps until all variables in the model have a sufficiently low p-value.

Backward selection cannot be used if $p > n$ and forward selection can always be used. Forward selection is a greedy approach and can include variables that later become redundant. Mixed selection can remedy this.

In multiple linear regression, $R^2 = \text{cor}(Y, \hat{Y})^2$. An R^2 value close to 1 indicates that the model explains a large portion of the variance in the response variable. The R^2 statistic will always increase when more variables are added to the model, even if those variables are only weakly associated with the response since adding another variable always results in a decrease in the residual sum of squares in the training data.

For multiple linear regression the RSE is defined as

$$\text{RSE} = \sqrt{\frac{\text{RSS}}{n-p-1}}$$

Thus, models with more variables can have higher RSE if the decrease in RSS is small relative to the increase in p.

The inaccuracy in the coefficient estimates is related to the reducible error and one can compute a confidence interval. Even if the true values for $\beta_0, \beta_1, \cdots, \beta_p$ are known, the response cannot be predicted perfectly. The error due to ϵ is called the irreducible error. Prediction errors are always wider than confidence intervals since they incorporate both the error in the estimate of the population and the randomness of the individual point.

1.2.3 Other considerations

Sometimes qualitative predictors known as factors are used. If a factor has two levels, we can create a dummy variable or an indicator. When a qualitative predictor has more than two levels, we can create additional dummy variables. There will always be one fewer dummy variable than the number of levels. The level with no dummy variable is known as the baseline.

The standard linear regression model has several highly restrictive assumptions that are often violated. It assumes that the relationship between the predictors and response are additive and linear. One way of relaxing the additive assumption is by introducing an interaction term. Take the following model for example:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \epsilon$$

The predictor $X_1 X_2$ is said to be an interaction term constructed by computing the product of X_1 and X_2. Note that the interpretation of parameters also changes with the inclusion of the interaction term. The hierarchical principle states that if we include an interaction in a model, we should also include the main effects, even if the p-values associated with their coefficients are not significant.

We can also extend the linear model by accommodating nonlinear relationships. For example, we can take polynomial regression. The following model is nonlinear in the predictors but linear in the coefficients:

$$\text{mpg} = \beta_0 + \beta_1 \times \text{horsepower} + \beta_2 \times \text{horsepower}^2 + \epsilon$$

A list of common problems associated with linear regression:

1. Non-linearity of the response-predictor relationships.
2. Correlation of error terms.
3. Non-constant variance of error terms.
4. Outliers.
5. High-leverage points.
6. Collinearity

If the true relationship is far from linear, then all conclusions drawn from the fit is suspect. Residual plots are a useful graphical tool to identify non-linearity. If the residual plot indicates there are non-linear associations, one approach is to use non-linear transformations of the predictors.

If the errors are correlated, the estimated standard errors tend to underestimate the true standard errors. Correlations frequently occur in time series data where individual data points are not independent. If error terms are positively correlated, there may be tracking in the residuals, that is, adjacent residuals have similar values.

We refer to a non-constant variance in the errors as heteroscedasticity. A nonlinear transformation may result in a reduction of heteroscedasticity. If we have a good idea of the variance of each response, then it is possible to fit with weighted least squares instead to accommodate for heteroscedasticity.

An outlier is a point for which y_i is far from the value predicted by the model. Removing an outlier may have little effect on the fit of the model but still have a high influence on the RSE. Residual plots

can be used to identify outliers. It can be difficult to decide how large a residual needs to be before it is considered an outlier. We can instead plot the studentized residuals, computed by dividing the residual e_i by its estimated standard error. Observations whose studentized residuals are greater than 3 in absolute value are possible outliers.

Observations with high leverage have an unusual value for x_i. High leverage observations have a substantial impact on the estimated regression fit. In order to quantify an observation's leverage, we compute the leverage statistic. For simple linear regression,

$$h_i = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum_{i'=1}^{n}(x_{i'} - \bar{x})^2}$$

For multiple linear regression, the leverage score of the ith observation is the ith diagonal entry of the hat-matrix $H = X(X^TX)^{-1}X^T$. That is,

$$h_{ii} = x_i(X^TX)^{-1}x_i^T$$

Leverage is between 0 and 1 and the sum of leverages is equal to the number of parameters. If $h_{ii} > \frac{2p}{n}$, it can be considered an outlier.

Collinearity refers to the situation in which two or more predictor variables are closely related to one another. It can be difficult to separate the individual effects of collinear variables on the response. Collinearity reduces the accuracy of the estimates of the regression coefficients, which causes the standard error of $\hat{\beta}_j$ to grow. This decreases the t-statistic, which in return reduces the power. One simple way to detect collinearity is in the correlation matrix of the predictors. An element that is large in absolute value indicates a pair of highly correlated variables. However, the correlation matrix does not detect multicollinearity, that is collinearity between three or more variables. One can instead look at the variance inflation factor (VIF). The VIF is the ratio of the variance of $\hat{\beta}_j$ when fitting the full model divided by the variance of $\hat{\beta}_j$ fit on its own. A VIF value that exceeds 5 or 10 indicates a problematic amount of multicollinearity.

$$\text{VIF}(\hat{\beta}_j) = \frac{1}{1 - R^2_{X_j|X_{-j}}}$$

where $R^2_{X_j|X_{-j}}$ is the R^2 from a regression of X_j onto all other predictors.

1.3 ISLR Chapter 4

1.3.1 Logistic Regression

Logistic regression models the probability that Y belongs to a particular category. Let $p(X) = P(Y = 1 \mid X)$. We take the model

$$\text{logit}(p(X)) = \log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \sum_{k=1}^{p}\beta_k X_k$$

To estimate the regression coefficients β_k's, we take the MLE.

If only a few predictors are used when other predictors are relevant, then there may be confounding.

We can extend the two-class logistic regression approach to $K > 2$ classes, known as multinomial logistic regression. We take the following model:

$$\log\left(\frac{P(Y = k \mid X = x)}{P(Y = K \mid X = x)}\right) = \sum_{j=0}^{p}\beta_{kj}x_j$$

The Kth class is known as the baseline and satisfies

$$P(Y = K \mid X = x) = \frac{1}{1 + \sum_{l=1}^{K-1} \exp(\sum_{j=0}^{p} \beta_{lj} x_j)}$$

and for $k \in \{1, 2, \cdots, K-1\}$,

$$P(Y = k \mid X = x) = \frac{\exp(\sum_{j=0}^{p} \beta_{kj} x_j)}{1 + \sum_{l=1}^{K-1} \exp(\sum_{j=0}^{p} \beta_{lj} x_j)}$$

An alternative coding for multinomial logistic regression is the softmax coding. Rather than selecting a baseline class, we treat all K classes symmetrically and assume for $k \in \{1, 2, \cdots, K\}$,

$$P(Y = k \mid X = x) = \frac{\exp(\sum_{j=0}^{p} \beta_{kj} x_j)}{\sum_{l=1}^{K} \exp(\sum_{j=0}^{p} \beta_{lj} x_j)}$$

1.3.2 Linear Discriminant Analysis

We now consider alternative approaches for classification. Let $f_k(X) = p(X|Y = k)$ be a conditional density. Let π_k represent the prior probability that a randomly chosen observation comes from the kth class. Bayes theorem states that

$$P(Y = k \mid X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$$

We use the abbreviation $p_k(x) = P(Y = k \mid X = x)$ to be the posterior probability that an observation $X = x$ belongs to the kth class. Generally estimating π_k is easy by just taking the fraction of the training sample that are in the kth class. Estimating $f_k(x)$ is more challenging.

Assume $p = 1$. Further assume $f_k(x)$ is normal.

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{1}{2\sigma_k^2}(x - \mu_k)^2\right)$$

Further assume that $\sigma_1^2 = \sigma_2^2 = \cdots = \sigma_K^2$. Then,

$$p_k(x) = \frac{\pi_k \exp(-\frac{1}{2\sigma^2}(x - \mu_k)^2)}{\sum_{l=1}^{K} \pi_l \exp(-\frac{1}{2\sigma^2}(x - \mu_l)^2)}$$

Thus, given this form we should assign observations to the class for which

$$\delta_k(x) = x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

is largest. The linear discriminant analysis (LDA) method approximates the Bayes classifier by plugging in estimates for π_k, μ_k and σ^2.

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i|y_i=k} x_i$$

$$\hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i|y_i=k} (x_i - \hat{\mu}_k)^2$$

where n is the total number of training observations, n_k is the number of training observations in the kth class. In the absence of any additional information, take

$$\hat{\pi}_k = \frac{n_k}{n}$$

The LDA classifier assigns observation $X = x$ to the class for which

$$\hat{\delta}_k(x) = x\frac{\hat{\mu}_k}{\hat{\sigma}^2} - \frac{\hat{\mu}_k^2}{2\hat{\sigma}^2} + \log(\hat{\pi}_k)$$

is largest. The name comes from the discriminant functions $\hat{\delta}_k(x)$ being linear functions of x.

We can extend the LDA classifier to multiple predictors by assuming X is drawn from a multivariate normal distribution with a class specific mean vector and common covariance matrix. We find that

$$\delta_k(x) = x^T\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k + \log \pi_k$$

Similar estimates for $\mu_1, \cdots, \mu_K$, $\pi_1, \cdots, \pi_K$ and Σ can be given. As before, $\hat{\delta}_k(x)$ is a linear function of x. For binary classification, a confusion matrix is a convenient way of displaying which types of errors are being made. A confusion matrix has entries true positive (TP), false negative (FN), false positive (FP), true negative (TN). Sensitivity, recall, or hit rate refers to the true positive rate (TPR). Specificity, selectivity refers to the true negative rate (TNR). The Bayes classifier assigns observations for the class for which the posterior probability $p_K(X)$ is greatest. For the two-class case, the threshold is 50%. However, if we are concerned about incorrect predictions for a particular class, we may change the threshold. The ROC curve displays the error rate for all possible thresholds. It plots the true positive rate (TPR) against the false positive rate (FPR). The overall performance of a classifier is summarized by the area under the curve (AUC) statistic. An ideal ROC curve is close to the top left corner.

1.3.3 Quadratic Discriminant Analysis

Quadratic discriminant analysis also assumes that the observations for each class are drawn from a normal distribution and plugs in estimators for the parameters into Bayes' theorem for prediction. Unlike LDA, QDA assumes that each class has its own covariance matrix. Then

$$\begin{aligned}\delta_k(x) &= -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) - \frac{1}{2}\log|\Sigma_k| + \log \pi_k \\ &= -\frac{1}{2}x^T\Sigma_k^{-1}x + x^T\Sigma_k^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma_k^{-1}\mu_k - \frac{1}{2}\log|\Sigma_k| + \log \pi_k\end{aligned}$$

QDA is more flexible than LDA and has lower bias but has greater variance. QDA is recommended if the training set isvery large or if the assumption of a common covariance matrix is untenable.

1.3.4 Naive Bayes

The Naive Bayes classifier assumes that within the kth class, the p predictors are independent. That is,

$$f_k(x) = \prod_{j=1}^{p} f_{kj}(x_j)$$

This assumption means the association between the predictors need not be estimated. The model assumes

$$P(Y = k \mid X = x) = \frac{\pi_k \prod_{j=1}^{p} f_{kj}(x_j)}{\sum_{l=1}^{K} \pi_l \prod_{j=1}^{p} f_{lj}(x_j)}$$

for $k \in \{1, 2, \cdots, K\}$. To estimate the one dimensional density functions f_{kj}, we can assume $X_j \mid Y = k \sim N(\mu_{jk}, \sigma_{jk}^2)$. This amounts to QDA with the additional assumption that class-specific covariance matrices are diagonal. Another option is to use a nonparametric estimate for f_{kj} such as a kernel density estimator. If X_j is a factor, then we can simply count the proportion of training observations for the jth predictor corresponding to each class.

It should be noted that the naive Bayes estimator is a generalized additive model and that LDA is a special case of Naive Bayes. Neither QDA nor naive Bayes is a special case of the other. Naive Bayes can be flexible but is limited to an additive fit whereas QDA contains quadratic terms.

We can compare the previous methods discussed with KNN. Since KNN is completely nonparametric, it should dominate LDA and logistic regression when the decision boundary is highly nonlinear and when n is large while p is small. KNN requires a lot of observations relative to the number of predictors. If n is modest or p is not very small, QDA may be preferred to KNN since it is nonlinear while still having a parametric form. KNN does not tell which predictors are important unlike logistic regression.

1.4 ISLR Chapter 5

Resampling methods involve repeatedly drawing samples from a training set and refitting a model of interest on each sample in order to obtain additional information about the fitted model. Resampling approaches are typically computationally expensive due to the need to fit the same statistical method multiple times. The most commonly used resampling methods are cross-validation and bootstrap.

Model assessment is the process of evaluating a model's performance and model selection is the process of selecting a proper level of flexibility for a model.

1.4.1 Cross Validation

In the absence of a designated test set, one can estimate the test error rate by holding out a subset of the training observations from the fitting process. The validation set error rate provides an estimate of the test error rate. This approach is conceptually simple and easy to implement but the validation estimate of the test error rate can be highly variable, and the validation set error rate tends to overestimate the test error rate for the model fit on the entire data set.

Leave-one-out-cross-validation (LOOCV) leaves a single observation out for the validation set and uses the remaining $n - 1$ training observations to fit the statistical learning method. A prediction is made for the excluded observation. This procedure is repeated for all other observations. The LOOCV estimate of the test MSE is the average of these n test error estimates.

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

The k-fold CV approach involves randomly dividing the set of observations into k groups, or folds, of approximately equal size. The first fold is used as a validation set and the method is fit on the remaining $k - 1$ folds. The mean squared error is computed on the observations in the held-out fold. This procedure is repeated by treating other folds as validation sets. The k-fold CV estimate averages these values:

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

In practice, one usually takes $k = 5$ or $k = 10$. The obvious advantage over LOOCV is the computation speed. It also gives more accurate estimates of the test error rate than does LOOCV due to a bias-variance trade-off. The outputs of the fitted models are correlated with each other, with the correlation

increasing with increased number of folds, and since the mean of highly correlated quantities has higher variance, the test error estimate of LOOCV tends to have higher variance than that of k-fold CV.

On classification problems, the LOOCV error rate is

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

where $\mathrm{Err}_i = 1\{y_i \neq \hat{y}_i\}$. Similar for cross-validation.

1.4.2 Bootstrap

The bootstrap emulates the process of obtaining new sample sets by repeatedly sampling observations from the original data set. This sampling is done with replacement. The bootstrap standard error of $\hat{\alpha}$ is given by

$$\mathrm{SE}_B(\hat{\alpha}) = \sqrt{\frac{1}{B-1}\sum_{r=1}^{B}\left(\hat{\alpha}^{*r} - \frac{1}{B}\sum_{r'=1}^{B}\hat{\alpha}^{*r'}\right)^2}$$

1.5 ISLR Chapter 6

1.5.1 Subset Selection

Best subset selection fits a separate least squares regression for every possible combination of the p predictors with the goal of identifying the one that is best. The algorithm is described in the following way:

1. Let $\mathcal{M}_0$ denote the null model, which contains no predictors.
2. For $k \in \{1, 2, \cdots, p\}$:
   - Fit all $\binom{p}{k}$ models that contain exactly k predictors.
   - Pick the best among these $\binom{p}{k}$ models, and call it $\mathcal{M}_k$. Best is defined as the one with the smallest RSS or largest R^2.
3. Select a single best model among $\mathcal{M}_0, \cdots, \mathcal{M}_p$ using cross-validated prediction error, C_p, AIC, BIC, or adjusted R^2.

In logistic regression, instead of ordering by RSS, we instead use deviance which is negative two times the maximized log-likelihood. The smaller the deviance, the better the fit. Best subset selection is infeasible for high values of p.

Forward stepwise selection is a computationally efficient alternative. Forward stepwise selection algorithm is as follows:

1. Let $\mathcal{M}_0$ denote the null model.
2. For $k \in \{0, 1, \cdots, p-1\}$,
   (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 $\mathcal{M}_{k+1}$. Best is defined as smallest RSS or deviance.
3. Select a single best model from among $\mathcal{M}_0, \cdots, \mathcal{M}_p$ using cross-validated prediction error, C_p, AIC, BIC, or adjusted R^2.

Backward stepwise selection is another alternative to best subset selection. The algorithm is as follows:

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

Forward and backward selection are not guaranteed to yield the best model. A hybrid approach where both backward and forward steps are available are possible.

Generally, training set MSE is an underestimate of the test MSE. But there are techniques for adjusting the training error for the model size. For a fitted least squares model containing d predictors, Mallow's C_p estimate of test MSE is given by

$$C_p = \frac{1}{n}(\text{RSS} + 2d\hat{\sigma}^2)$$

Typically $\hat{\sigma}^2$ is estimated using the full model containing all predictors. If $\hat{\sigma}^2$ is an unbiased estimator of σ^2, then C_p is an unbiased estimate of the test MSE.

The Aikake Information Criterion (AIC) is defined for a large class of model fit by maximum likelihood. AIC is given by

$$\text{AIC} = \frac{1}{n}(\text{RSS} + 2d\hat{\sigma}^2)$$

The Bayesian Information Criterion (BIC) takes the form

$$\text{BIC} = \frac{1}{n}(\text{RSS} + \log(n)d\hat{\sigma}^2)$$

The adjusted R^2 statistic is calculated as

$$\text{Adjusted } R^2 = 1 - \frac{\text{RSS}/(n-d-1)}{\text{TSS}/(n-1)}$$

Alternatively, we can directly estimate the test error using the validation set and cross-validation methods. The advantage is that it makes fewer assumptions about the underlying model. For cross-validation, we use the one-standard-error rule. Where we first calculate the standard error of the estimated test MSE for each model size, and then select the smallest model for which the estimated test error is within one standard error of the lowest point on the curve. The idea is that if a set of models appear to be more or less good, then we should choose the simplest model.

1.5.2 Shrinkage Methods

The OLS minimizes the RSS

$$\text{RSS} = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2$$

Ridge regression minimizes the RSS plus a shrinkage penalty.

$$\hat{\beta}_\lambda^R = \underset{\beta}{\text{argmin}}\, \text{RSS}(\beta) + \lambda \sum_{j=1}^{p} \beta_j^2$$

The tuning parameter λ serves to control the relative impact of the penalty. When $\lambda = 0$, there is no penalty and when $\lambda = \infty$, the regression coefficients equal zero. Ride regression will generate a different set of coefficients for each value of λ. Note that we do not shrink the intercept.

The standard least squares coefficients are scale equivariant. Multiplying the predictors by a constant c leads to a scaling of the coefficient estimates by c^{-1}. However, the ridge regression coefficient estimates are not scale equivariant so it is best to apply ridge regression after standardizing the predictors first.

Ridge regression is advantageous due to the bias-variance trade-off. As λ increases, the flexibility of the ridge regression fit decreases, leading to decreased variance but increased bias.

Ridge regression has the disadvantage of shrinking all coefficients towards zero but not setting any of them exactly to zero, leading to problems with model interpretation. The lasso is an alternative to ridge regression that sets some coefficients to zero. It has the form

$$\hat{\beta}_\lambda^R = \underset{\beta}{\operatorname{argmin}} \operatorname{RSS}(\beta) + \lambda \sum_{j=1}^{p} |\beta_j|$$

The ℓ_1 penalty forces some coefficient estimates to be exactly zero when the tuning parameter λ is sufficiently large. In this sense, the lasso performs variable selection. We say that the lasso yields sparse models.

Ridge regression and lasso are described as solutions to a convex optimization problem, which have the Lagrangian dual problems

$$\underset{\beta | \|\beta\|_1 \leq s}{\operatorname{argmin}} \operatorname{RSS}(\beta)$$

$$\underset{\beta | \|\beta\|_2 \leq s}{\operatorname{argmin}} \operatorname{RSS}(\beta)$$

For every λ, there is a corresponding s that gives the same coefficient estimates. Best subset selection may be thought as the solution to

$$\underset{\beta | \|\beta\|_0 \leq s}{\operatorname{argmin}} \operatorname{RSS}(\beta)$$

The ℓ_1 norm is a convex approximation to the ℓ_0 norm and hence the lasso may be thought of as a convex relaxation of the best subset selection problem.

Ridge regression and lasso requires a method to select the tuning parameter. We choose a grid of λ values and compute the cross-validation error for each value of λ then select the tuning parameter for which the cross-validation error is smallest. Then the model is refit with the selected value of the tuning parameter.

1.5.3 Dimension Reduction

Let $Z_1, Z_2, \cdots, Z_M$ represent $M < p$ linear combinations of the original p predictors.

$$Z_m = \sum_{j=1}^{p} \phi_{jm} X_j$$

for constants ϕ_{jm}, $j \in [p], m \in [M]$. Fit the linear regression model

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_{im} + \epsilon_i$$

with least squares. If the constants are chosen wisely then the dimension reduction should outperform least squares.

Principal components analysis (PCA) is a popular approach for deriving a low-dimensional set of features. PCA is performed by maximizing the Rayleigh quotient. The first vector satisfies

$$\phi_{\cdot 1} = \underset{w}{\text{argmax}}\{\frac{w^T X^T X w}{w^T w}\}$$

Given the first $k-1$ components, let $\hat{X}_k = X - \sum_{s=1}^{k-1} X\phi_{\cdot s}\phi_{\cdot s}^T$ and

$$\phi_{\cdot k} = \underset{w}{\text{argmax}}\{\frac{w^T \hat{X}_k^T \hat{X}_k w}{w^T x}\}$$

Note that even though PCR provides a simple way to perform regression using $M < p$ predictors, it is not a feature selection method. Generally, one should standardize the predictors before generating the principal components.

Partial least squares (PLS) is a supervised alternative to PCR. Even though the low rank approximation of X in PCR best accounts for X, it is not necessarily optimal for regression. The PLS1 algorithm is as follows.

1. Columnwise center X and y and set $X_0 = X$.
2. Repeat for $i = 1, \cdots, k$:
   (a) Set $w_i = \frac{X_{i-1}^T y}{\|X_{i-1}^T y\|}$
   (b) Set $t_i = \frac{X_{i-1} w_i}{\|X_{i-1} w_i\|}$
   (c) Set $v_i = X_{i-1}^T t_i$
   (d) Set $X_i = X_{i-1} - t_i v_i^T$
3. Collect the vectors into matrices W_k, T_k, V_k.

$$\hat{\beta}_{\text{PLSE}}^{(k)} = W_k (V_k^T W_k)^{-1} T_k^T y$$

For a given value of k, the PLSE has a better predictability than the corresponding PCR estimator. As with PCR, the number of dimensions is a tuning parameter typically chosen by cross-validation.

1.5.4 High dimensions

The low-dimensional setting is the case in which $n \gg p$. Data sets where $p > n$ are referred to as high-dimensional. Regularization plays a key role in high-dimensional problems and an appropriate tuning parameter selection is crucial for good predictive performance. The curse of dimensionality states that the test error tends to increase with dimension unless the additional features are truly associated with the response.

In the high-dimensional setting, the multicollinearity problem is extreme. Every variable in the model can be written as a linear combination of all other variables in the model. Thus we can never know exactly which variables are truly predictive of the outcome, and we can never identify the best coefficients for use in the regression. In the high-dimensional setting, if forward stepwise selection selects some predictors, it would be incorrect to conclude that these predictors predict more effectively than the other predictors not included. There are likely many sets of predictors that predict just as well as the selected model.

In the high-dimensional setting, it is easy to obtain a useless model with zero residuals. Thus one should not use traditional measures of model fit on training data as evidence of a good model fit in the high-dimensional setting. One must report results on an independent test set or cross-validation errors.

1.6 ISLR Chapter 7

1.6.1 Basis functions

Polynomial regression allows for nonlinearity by allowing for polynomials in the predictors. Using polynomial functions of the features imposes a global structure on the nonlinear function of X. One can instead use step functions. We can break the range of X into bins and fit a different constant in each bin. This converts a continuous variable into an ordered categorical variable. With indicators on the bins, this may be fit in the same way as linear regression.

Polynomial and piecewise constant regression models are special cases of a basis function approach. Instead of fitting a linear model in X, we can fit

$$y_i = \beta_0 + \sum_{j=1}^{K} \beta_j b_j(x_i) + \epsilon_i$$

One flexible class of basis functions are piecewise polynomials. The points where the coefficients change are called knots. We can also impose constraints on the piecewise polynomials by imposing continuity, continuity of the first derivative, and continuity of the second derivative. With third degree polynomials and the constraints, we get cubic splines. Cubic splines with K knots have $K+4$ degrees of freedom. One way to represent a cubic spline is with a truncated power basis. A truncated power basis function is defined as

$$h(x, \xi) = (x - \xi)_+^3$$

where ξ is the knot. Splines can have high variance at the outer range of the predictors. A natural spline is a regression spline with additional boundary constraints: the function is required to be linear at the boundary.

It is common to place the knots of a spline at uniform quantiles of the data. One can again use cross-validation to select for the degrees of freedom to be used.

We can also take a different approach to produce a spline. Note that if we wanted to find a function $g(x)$ that fits the observed data well, we can simply choose a g that interpolates all of the y_i. We can control this by requiring g to be smooth. We can find the function g that minimizes

$$\sum_{i=1}^{n} (y_i - g(x_i))^2 + \lambda \int g''(t)^2 dt$$

where λ is a nonnegative tuning parameter. The function g minimizing this objective is called a smoothing spline. A smoothing spline can be shown to be a piecewise cubic polynomial with knots at the unique values of $x_1, \cdots, x_n$, and have continuous first and second derivatives at each knot that is linear outside the extreme knots. The smoothing parameter λ controls the effective degrees of freedom. Define

$$\hat{g}_\lambda = S_\lambda y$$

where $\hat{g}_\lambda$ is the solution to the minimization problem for the smoothing spline. The effective degrees of freedom is defined as

$$\mathrm{df}_\lambda = \mathrm{tr}(S_\lambda)$$

The LOOCV can be computed efficiently for smoothing splines.

$$\mathrm{RSS}_{cv}(\lambda) = \sum_{i=1}^{n} \left[\frac{y_i - \hat{g}_\lambda(x_i)}{1 - \{S_\lambda\}_{ii}} \right]^2$$

1.6.2 Local Regression

Local regression computes the fit at a target point using only the nearby training observations. Local regression is sometimes referred to as a memory-based procedure because we need all the training data each time we wish to compute a prediction. The most important choice in a local regression is the span s, the proportion of points used to compute the local regression at x_0. The span controls the flexibility of the fit.

1. Gather the fraction $s = k/n$ of training points whose x_i are closest to x_0.
2. Assign a weight $K_{i0} = K(x_i, x_0)$ to each point in the neighbourhood so that the closer points have higher weight, where all but these k nearest neighbors have weight zeor.
3. Fit a weighted least squares regression on the y_i by finding a minimum of

$$\sum_{i=1}^{n} K_{i0}(y_i - \beta_0 - \beta_1 x_i)^2$$

4. The fitted value at x_0 is given by $\hat{f}(x_0) = \hat{\beta}_0 + \hat{\beta}_1 x_0$.

A varying coefficient model such as a local regression is a useful way of adapting a model to the most recently gathered data. However, local regression performs poorly if p is larger than 3 or 4 because there will generally be very few training observations close to x_0.

1.6.3 Generalized Additive Models

Generalized additive models (GAM) provide a general framework for extending a standard linear model by allowing nonlinear functions of each of the variables while maintaining additivity. Write the model as

$$y_i = \beta_0 + \sum_{j=1}^{p} f_j(x_{ij}) + \epsilon_i$$

GAMs allow us to fit a nonlinear f_j to each X_j so that we can automatically model non-linear relationships that standard linear regression will miss. The nonlinear fits can potentially make more accurate predictions for the response. Because the model is additive, we can examine the effect of each X_j on Y while holding all other variables fixed. The smoothness of f_j can be summarized by the degrees of freedom. The main limitation of GAMs is that the model must be additive. For this purpose, we can manually add interaction terms to GAMs or fit interaction terms using two-dimensional smoothers such as two-dimensional splines.

1.7 ISLR Chapter 8

This chapter discussed tree-based methods. To start, a regression tree involves dividing the predictor space into J distinct and non-overlapping regions $R_1, R_2, \cdots, R_J$. For every observation in region R_j we make the same prediction, the mean of the response variables for the training observations in R_j. While the regions can technically take any shape, it is often hyperrectangles or boxes. The ideal boxes $R_1, R_2, \cdots, R_J$ minimize the SSE

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{n_j})^2$$

To reduce the computational burden, a greedy approach called recursive binary splitting is used. This approach starts at the root (top) of the tree and splits the predictor space with two branches. It is

greedy because the best split is made at the particular step rather than looking ahead to make a better split in the future. Define the half-planes

$$R_1(j,s) = \{X \mid X_j < s\}, \quad R_2(j,s) \mid X_j \geq s\}$$

We seek to find the j and s that minimizes

$$\sum_{i|x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i|x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2$$

We can repeat this process to make further splits in the data until a stopping criterion such as continuing until no region contains more than five observations.

This process will likely overfit the data, leading to poor test set performance so in practice one grows a large tree T_0 and prunes it to obtain a subtree. Cost complexity pruning, also known as weakest link pruning, considers trees indexed by a nonnegative tuning parameter α. Let $|T|$ denote the number of terminal nodes of tree T. For each α, there is a subtree $T \subset T_0$ that minimizes

$$\sum_{m=1}^{|T|} \sum_{x_i \in R_m} (y_i - \hat{y}_{R_m})^2 + \alpha|T|$$

The tuning parameter α controls the subtree's complexity. It turns out that branches get pruned from the tree in a nested and predictable fashion. The tuning parameter α is selected via a validation set or cross-validation.

A classification tree is similar to a regression tree but is used for classification. For a classification tree, we predict that each observation belongs to the most commonly occurring class of training observations in the region to which it belongs. We are not only interested in the class prediction to a terminal node region but also the class proportions among the training observations that fall in the region. A different objective function must be used for classification. The Gini index is defined by

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$

A small value indicates that a node predominantly contains observations from a single class. An alternative to Gini index is entropy, given by

$$D = -\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}$$

The entropy will take on a value near zero if $\hat{p}_{mk}$ are all near zero or near one. Either the Gini index or the entropy are used to evaluate the quality of a particular split for pruning but the classification error rate is preferable if prediction accuracy of the final pruned tree is the goal. Sometimes splits will yield the same predicted value. The reason the split is made at all is because it increases node purity.

Advantages for decision trees:

- Trees are easily explained and easily interpreted by a non-expert.
- Trees can be displayed graphically.
- Trees more closely resembles human decision-making.
- Trees can handle qualitative predictors without needing to create dummy variables.

Disadvantages for decision trees:

- Trees do not have the same level of predictive accuracy as other regression and classification approaches.
- Trees are non-robust. Small changes in the data can lead to large changes in the final estimated tree.

An ensemble method combines many simple building block models to obtain a single more powerful model. The building block models are known as weak learners.

Decision trees suffer from high variance. Bootstrap aggregation, or bagging, is a general-procedure for reducing the variance of a statistical learning method, which can be used for decision trees.

$$\hat{f}_{\text{bag}}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^{*b}(x)$$

Construct B regression trees using B bootstrapped training sets, and average the resulting predictions. These trees are grown deep and are not pruned. Hence each individual tree has high variance but low bias and averaging these B trees reduces the variance. For classification, we take the majority rule: the overall prediction is the most commonly occurring class among the B predictions. Note here that the number of trees B is not a tuning parameter and a very large value of B will not lead to overfitting.

On average, each bagged tree makes use of around two-thirds of the observations. The remaining one third of the observations not used to fit a given bagged tree are referred to as the out-of-bag observations. We can predict the response for the ith observation using each of the trees in which that observation was OOB. This yields in about $B/3$ predictions for the ith observations. Average or take the majority vote to get a single OOB prediction for the ith observation. An overall OOB MSE or classification error can be computed this way and the resulting OOB error is a valid estimate of the test error for the bagged model. For large enough B, the OOB error is nearly equivalent to leave-one-out cross-validation error.

Bagging results in improved accuracy but it is hard to interpret the resulting model. One can obtain an overall summary of the importance of each predictor by averaging the amount of SSE decreased due to splits over a given predictor averaged over the B trees. A large value indicates an important predictor. For classification, averaging the amount of Gini index decreased by splits over a predictor averaged over the B trees.

Random forests improves on bagging by decorrelating the trees. As in bagging, one builds decision trees on bootstrapped training samples but also takes a random sample of m predictors as split candidates from the full set of p predictors. A fresh sample of m predictors is taken at each split and we typically choose $m \approx \sqrt{p}$. If $m = p$, this is simply bagging. A small value of m is helpful for a large number of correlated predictors.

Here is a boosting algorithm:

1. Set $\hat{f}(x) = 0$ and $r_i = y_i$ for all i in the training set.
2. For $b = 1, 2, \cdots, B$:
   (a) Fit a tree $\hat{f}^b$ with d splits to the training data (X, r).
   (b) Update $\hat{f}$ by adding a shrunken version of the new tree:

$$\hat{f}(x) \leftarrow \hat{f}(x) + \lambda\hat{f}^b(x)$$

(c) Update the residuals:

$$r_i \leftarrow r_i - \lambda \hat{f}^b(x_i)$$

3. Output the boosted model:

$$\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^b(x)$$

The tuning parameters are B, λ, d. Unlike bagging and random forests, a high B can actually lead to overfitting for boosting. Cross-validation is used to select B. The shrinkage parameter λ controls the rate at which boosting learns. Typically small positive numbers such as 0.01 and 0.001. Very small λ can require a large B. The number d of splits controls the complexity of the boosted ensemble. d is the interaction depth, controlling the interaction order of the boosted model since d splits can involve at most d variables. The choice $d = 1$ is referred to as a "stump" consisting of a single split which is equivalent to fitting an additive model. Smaller d have the benefit of being more interpretable.

Bayesian additive regression trees (BART) is another ensemble method that uses decision trees as building blocks. Each tree for BART is constructed in a random manner as in bagging and random forests, and each tree tries to capture signal not yet accounted for as in boosting. Let K denote the number of regression trees, B the number of iterations for which BART is run, $\hat{f}_k^b(x)$ represents the prediction at x for the kth regression tree used in the bth iteration. At the end of each iterations, we take $\hat{f}^b(x) = \sum_{k=1}^{K} \hat{f}_k^b(x)$ for all b.

In the first iteration of BART, all trees are initialized to have a single root node, with $\hat{f}_k^1(x) = \frac{1}{nK} \sum_{i=1}^{n} y_i$. Thus, $\hat{f}^1(x) = \sum_{k=1}^{K} \hat{f}_k^1(x) = \frac{1}{n} \sum_{i=1}^{n} y_i$. In subsequent iterations, BART updates each of the K trees one at a time. In the bth iteration, to update the kth tree, subtract from each response value the predictions from all but the kth tree to obtain a partial residual

$$r_i = y_i - \sum_{k'<k} \hat{f}_{k'}^b(x_i) - \sum_{k'>k} \hat{f}_{k'}^{b-1}(x_i)$$

for the ith observation. Rather than fitting a fresh tree to this partial residual, BART randomly chooses a perturbation to the tree from the previous iteration, favoring ones that improve the fit to the partial residual.

1. We may change the structure of the tree by adding or pruning branches.
2. WE may change the prediction in each terminal node of the tree.

The output of BART is a collection of prediction models,

$$\hat{f}^b(x) = \sum_{k=1}^{K} \hat{f}_k^b(x)$$

We typically throw away the first few of these prediction models, called the burn-in period. Let L denote the number of burn-in iterations. Then, we take the average after the burn-in iterations,

$$\hat{f}(x) = \frac{1}{B-L} \sum_{b=L+1}^{B} \hat{f}^b(x)$$

However, other quantities such as percentiles of $\hat{f}^{L+1}(x), \cdots, \hat{f}^B(x)$ can be taken. The BART method can be viewed as a Bayesian approach to fitting an ensemble of trees: randomly perturbing a tree to fit the residuals is drawing a new tree from a posterior distribution. Here is the complete algorithm for BART:

1. Let $\hat{f}_1^1(x) = \hat{f}_2^1(x) = \cdots = \hat{f}_K^1(x) = \frac{1}{nK}\sum_{i=1}^n y_i$.
2. Compute $\hat{f}^1(x) = \sum_{k=1}^K \hat{f}_k^1(x) = \frac{1}{n}\sum_{i=1}^n y_i$
3. For $b \in \{2, 3, \cdots, B\}$:
   (a) For $k \in [K], i \in [n]$, compute
   $$r_i = y_i - \sum_{k'<k} \hat{f}_{k'}^b(x_i) - \sum_{k'>k} \hat{f}_{k'}^{b-1}(x_i)$$
   (b) Fit a new tree, $\hat{f}_k^b(x)$ to r_i by randomly perturbing the kth tree from the previous iteration, $\hat{f}_k^{b-1}(x)$. Perturbations that improve the fit are preferred.
   (c) Compute $\hat{f}^b(x) = \sum_{k=1}^K \hat{f}_k^b(x)$
4. Compute the mean after L burn-in samples,
   $$\hat{f}(x) = \frac{1}{B-L}\sum_{b=L+1}^{B} \hat{f}^b(x)$$

Summary:

- In bagging, trees are grown from bootstrapped samples. The trees can be similar to each other and fail to thoroughly explore the model space.
- In random forests, the trees are grown from bootstrapped samples and each split is performed with a random subset of the features, decorrelating the trees relative to bagging.
- In boosting, there is no bootstrapping and the trees are grown successively with a slow learning rate.
- In BART, there is no bootstrapping and the trees are grown successively but the trees are perturbed for a more thorough exploration of the model space.

1.8 ISLR Chapter 12

Unsupervised learning is a set of statistical tools for the setting where only the features are available. The goal is to discover interesting things about the data and informative ways to visualize it. Unsupervised learning is often done as part of an exploratory data analysis.

1.8.1 Principal Component Analysis

Principal component analysis (PCA) finds a low-dimensional representation of the data set that contains as much of the variation as possible. We first center the data by subtracting the empirical mean. Then, for a design matrix X, the first principal component is given by

$$\phi_{1\cdot} = \underset{\|\phi\|=1}{\operatorname{argmax}} \|X\phi\|^2 = \underset{\|\phi\|=1}{\operatorname{argmax}} \phi^T X^T X \phi$$

Thus, $\phi_{1\cdot}$ is an eigenvector of $X^T X$ corresponding to the highest eigenvalue. The kth principal component can be found in this way:

$$\widehat{X}_k = X - \sum_{i=1}^{k-1} X\phi_{i\cdot}\phi_{i\cdot}^T$$

$$\phi_{k\cdot} = \underset{\|\phi|=1}{\operatorname{argmax}} \|\widehat{X}_k \phi\|^2$$

Indeed, $\phi_{i\cdot}$ correspond to the ith eigenvectors of $X^T X$ and $\phi_{i\cdot}^T X^T X \phi_{i\cdot}$ are eigenvalues. As normalized eigenvectors, the vectors $\{\phi_{i\cdot}\}$ form an orthonormal set. $\phi_{i\cdot}$ is called the ith principal component loading vector. In matrix form, we define

$$Z = X\Phi$$

X is the original data set, Φ contains the loading vectors, and Z contains the principal component scores.

Assuming that X is centered, the total variance of X is defined as

$$\frac{1}{n}\sum_{j=1}^{p}\sum_{i=1}^{n} x_{ij}^2$$

The variance explained by the mth principal component is

$$\frac{1}{n}\sum_{i=1}^{n} z_{im}^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{p}\phi_{jm}x_{ij}\right)^2$$

So the proportion of variance explained (PVE) of the mth principal component is

$$\frac{\sum_{i=1}^{n} z_{im}^2}{\sum_{j=1}^{p}\sum_{i=1}^{n} x_{ij}^2} = \frac{\sum_{i=1}^{n}(\sum_{j=1}^{p}\phi_{jm}x_{ij})^2}{\sum_{j=1}^{p}\sum_{i=1}^{n} x_{ij}^2}$$

In total, there are $\min\{n-1, p\}$ principal components and their PVEs sums to one.

The first M principal component loading and score vectors should be interpreted as the best M-dimensional approximation to the data in terms of residual sum of squares.

$$\widehat{\operatorname{Var}}(X) = \widehat{\operatorname{Var}}(Z) + \operatorname{MSE}(X - Z\Phi^T)$$

Moreover, the PVE is the R^2 of the approximation of X given by the first M principal components.

The results for PCA depend on whether the variables are individually scaled. If the features are on different units, then it may be desirable to first scale each variable to have standard deviation one before PCA. However, if the variables are measured in the same units, then one should not scale the variables.

To decide on the number of principal components to use, we can use a scree plot. A scree plot depicts the proportion of variance explained by each principal component. Cumulative proportion of variance plots are also used. Typically, the scree plot has a point where the proportion of variance explained by each subsequent principal component drops off referred to as the elbow of a scree plot. Visual inspection of the scree plot allows us to decide the number of principal components to use. It should be noted that this visual analysis is ad hoc but that there is no well-accepted objective way to decide the number of principal components.

1.8.2 Clustering

K-means clustering partitions the data into K distinct, non-overlapping clusters. The main idea of the K-means clustering is that a good clustering is one with low within-cluster variation. We would like to solve

$$\underset{C_1, C_2, \cdots, C_K}{\operatorname{argmin}} \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2$$

The K-means algorithm is as follows:

1. Randomly assign a number, from 1 to K to each of the observations.
2. Iterate until the cluster assignments do not change:
   (a) Compute the cluster centroid for each K clusters.
   (b) Assign each observation to the cluster whose centroid is closest.

This algorithm is guaranteed to decrease the value of the objective at each step. When the result no longer changes, a local optimum has been reached. The K-means algorithm depends on the initial cluster assignment of the observations so it is important to run the algorithm multiple times from different initial configurations then select the best solution.

One disadvantage of K-means clustering is the requirement to decide beforehand the number of clusters K. Hierarchical clustering is an alternative approach which does not require such a choice. Hierarchical clustering results in a tree-based representation of observations, called a dendrogram.

A bottom-up or agglomerative clustering refers to a dendrogram that is built starting from the leaves and combining up to the trunk. At the bottom of the dendrogram, each leaf represents one observations and as one moves up the tree, leaves fuse into branches and the branches fuse with leaves and other branches. The earlier the fusions occur, the more similar the group of observations are to each other. Observations that fuse later can be quite different. More precisely, the height of the fusion indicates how different two observations are. Note that we cannot draw conclusions about the similarity of two observations based on their proximity along the horizontal axis.

To identify clusters on the basis of a dendrogram, we make a horizontal cut across the dendrogram. The distinct sets of observations beneath the cut are the clusters. The height of the cut controls the number of clusters obtained and plays the same role as K in K-means clustering. This type of clustering is hierarchical in that clusters at a lower height are necessarily nested within the clusters at any greater height. Sometimes this assumption of hierarchical structure might be unrealistic.

Hierarchical clustering relies on a dissimilarity measure between each pair of observations, typically Euclidean distance. Observations with the smallest dissimilarity are fused first. The concept of dissimilarity between a pair of observations needs to be extended to pair of groups of observations to fuse clusters. Linkage defines the dissimilarity between two groups of observations. The most common types of linkage are complete, average, single, and centroid.

The hierarchical clustering algorithm is as follows:

1. Begin with n observations and treat each observation as its own cluster.
2. For $i = n, n-1, \cdots, 2$
   (a) Examine all pairwise inter-cluster dissimilarities among the i clusters and identify the pair of clusters that are least dissimilar. Fuse these two clusters. The dissimilarity between these two clusters indicates the height in the dendrogram at which the fusion should be placed.
   (b) Compute the new pairwise inter-cluster dissimilarities among the $i-1$ remaining clusters.

Linkage	Description
Complete	Largest pairwise dissimilarity.
Single	Smallest pairwise dissimilarity.
Average	Average pairwise dissimilarity.
Centroid	Dissimilarity between centroids of two clusters.

The most popular linkage are average, complete, and single linkage. A drawback of centroid linkage is that an inversion can occur where two clusters are fused at a height below the individual clusters. Average and complete linkage generally create more balanced dendrograms.

Other dissimilarity measures exist: for example, a correlation-based distance considers two observations to be similar if their features are highly correlated. In addition to selecting the dissimilarity measure, one should consider whether to scale the variables before the dissimilarity measure is computed.

Both K-means and hierarchical clustering will assign every observation to a cluster but sometimes there may be outliers that do not belong in any cluster. Outliers may greatly distort the resulting clusters. Mixture models are an approach that accommodates for outliers, resulting in a soft version of K-means clustering. Clustering methods are generally not robust to perturbations of the data. To get a sense of the robustness of the clusters, one can cluster subsets of the data. It is also important to remember that the results of clustering depend on the decisions such as whether the data is standardized, type of linkage, the height to cut the dendrogram, or the choice of K. Clustering results should not be taken as absolute truth but rather the starting point of a hypothesis and further study on an independent data set.

1.9 Exponential Families

Exponential families have densities of the form

$$f(y \mid \theta) = \exp[yb(\theta) + c(\theta) + d(y)]$$

The quantity $b(\theta)$ is called the natural parameter of the distribution. Parameters other than θ are called nuisance parameters. Here is a table of exponential families

Distribution	θ	$b(\theta)$	$c(\theta)$	$d(y)$
Binomial	p	$\log \frac{p}{1-p}$	$n \log(1-p)$	$\log \binom{n}{y}$
Negative Binomial	β	$\log \frac{\beta}{1+\beta}$	$-r \log(1+\beta)$	$\log \binom{r+y-1}{r-1}$
Poisson	λ	$\log \lambda$	$-\lambda$	$-\log y!$
Exponential	θ	$-\frac{1}{\theta}$	$-\alpha \log \theta$	0
Gamma	θ	$-\frac{1}{\theta}$	$-\alpha \log \theta$	$(\alpha-1)\log y - \log \Gamma(\alpha)$
Normal	μ	$\frac{\mu}{\sigma^2}$	$-\frac{\mu^2}{2\sigma^2}$	$-\frac{y^2}{2\sigma^2} - \frac{1}{2}\log 2\pi\sigma^2$
Inverse Gaussian	μ	$-\frac{\theta}{2\mu^2}$	$\frac{\theta}{\mu}$	$-\frac{\theta}{2y} + \frac{1}{2}\log \frac{\theta}{2\pi y^2}$

Tweedie distributions refer to a family of distributions where the variance is related to the mean through a power function

$$\text{var}(Y) = a[E(Y)]^p$$

Various exponential family distributions are tweedie distributions:

- Normal distribution, $p = 0$
- Poisson distribution, $p = 1$
- Compound Poisson-Gamma distribution, $1 < p < 2$
- Gamma distribution, $p = 2$
- Inverse Gaussian distribution, $p = 3$

In a GLM, the inverse of the link function transforms the linear formula for the mean to the original data set scale. The canonical link function is related to the function $b(\theta)$ mapping θ to the natural parameter.

Distribution	Canonical Link Function
Normal	Identity
Binomial	Logit
Poisson	Natural Logarithm
Gamma	Inverse

By differentiating $\int f(y \mid \theta)dy$ under the integral sign and rearranging, we get

$$E[Y] = -\frac{c'(\theta)}{b'(\theta)}$$

$$\text{var}[Y] = \frac{b''(\theta)c'(\theta) - c''(\theta)b'(\theta)}{b'(\theta)^3}$$

The score statistic takes the form

$$U = b'(\theta)Y + c'(\theta)$$

The score has mean zero and variance equal to the Fisher information.

$$I(\theta) = \frac{b''(\theta)c'(\theta)}{b'(\theta)} - c''(\theta)$$

2 Generalized Linear Model

2.0.1 Estimation

GLMs are estimated through maximum likelihood estimation. Consider independent random variables $Y_1, \cdots, Y_N$ in a GLM with $E(Y_i) = \mu_i$, $g(\mu_i) = x_i^T\beta$. We wish to estimate the parameters β. For each y_i, the log-likelihood is

$$l_i = y_i b(\theta_i) + c(\theta_i) + d(y_i)$$

Hence the score is

$$\frac{\partial l}{\partial \beta_j} = U_j = \sum_{i=1}^{N} \frac{\partial l_i}{\partial \theta_i}\frac{\partial \theta_i}{\partial \mu_i}\frac{\partial \mu_i}{\partial \beta_j}$$

Consider each multiplicand separately. First,

$$\frac{\partial l_i}{\partial \theta_i} = y_i b'(\theta_i) + c'(\theta_i) = b'(\theta_i)(y_i - \mu_i)$$

By differentiating $\mu_i = -\frac{c'(\theta_i)}{b'(\theta_i)}$ with respect to θ_i,

$$\frac{\partial \mu_i}{\partial \theta_i} = -\frac{c''(\theta_i)}{b'(\theta_i)} + \frac{c'(\theta_i)b''(\theta_i)}{[b'(\theta_i)^2} = b'(\theta_i)\text{var}(Y_i)$$

$$\frac{\partial \mu_i}{\partial \beta_j} = \frac{\partial \mu_i}{\partial \eta_i}x_{ij}$$

Therefore the score is

$$U_j = \sum_{i=1}^{N}\left[\frac{y_i - \mu_i}{\text{var}(Y_i)}x_{ij}\frac{\partial \mu_i}{\partial \eta_i}\right]$$

The information matrix is $I = \text{cov}(U)$, which can be simplified to

$$I_{jk} = \sum_{i=1}^{N}\frac{x_{ij}x_{ik}}{\text{var}(Y_i)}\left(\frac{\partial \mu_i}{\partial \eta_i}\right)^2$$

which can be rewritten as

$$I = X^T W X$$

where W is a diagonal matrix with

$$W_{ii} = \frac{1}{\text{var}(Y_i)} \left(\frac{\partial \mu_i}{\partial \eta_i} \right)^2$$

Fisher scoring generalizes to

$$b^{(m)} = b^{(m-1)} + [I^{(m-1)}]^{-1} U^{(m-1)}$$

where $b^{(m)}$ is the mth estimate of β and $I^{(m-1)}, U^{(m-1)}$ are I and U evaluated at $b^{(m)}$. The iterative equation can be expressed as

$$X^T W X b^{(m)} = X^T W z$$

where z has entries

$$z_i = \sum_{k=1}^{p} x_{ik} b_k^{(m-1)} + (y_i - \mu_i) \frac{\partial \eta_i}{\partial \mu_i}$$

with μ_i and $\frac{\partial \eta_i}{\partial \mu_i}$ evaluated at $b^{(m-1)}$. Note that this is the same form as the normal equations for a weighted least squares. Hence, the maximum likelihood estimators in a GLM are obtained by an iterative weighted least squares procedure.

2.0.2 Inference

A goodness of fit statistic is a summary statistic to describe how well the model fits the data. It may be based on a maximum value of the log-likelihood or the minimum value of the sum of squares criterion or a composite statistic based on the residuals.

1. Specify a model M_0 corresponding to H_0. Specify a more general model M_1.
2. Fit M_0 and calculate the goodness of fit statistic G_0. Fit M_1 and calculate the goodness of fit statistic G_1.
3. Calculate the improvement in fit, $G_1 - G_0$ or G_1/G_0.
4. Use the sampling distribution of $G_1 - G_0$ to test the null hypothesis $G_1 = G_0$.
5. If the null hypothesis is not rejected, then M_0 is the preferred model. If it is rejected, then M_1 is regarded as the better model.

The asymptotic sampling distribution for score statistics is

$$U^T I^{-1} U \xrightarrow{d} \chi_p^2$$

for a p-dimensional vector of parameters. We can take a Taylor expansion of the score statistic to get

$$U(\beta) = U(b) - I(b)(\beta - b)$$

Note that if b is the MLE, by definition $U(b) = 0$ so

$$(b - \beta) = I^{-1}(b) U$$

since $I = E(UU^T)$ and I is symmetric,

$$(b - \beta)^T I(b)(b - \beta) \xrightarrow{d} \chi_p^2$$

This is called the Wald statistic.
One way of assessing the adequacy of a model is to compare it with a more general model with the

maximum number of parameters that can be estimated, called a saturated model, also called a maximal or full model. Define the deviance or the log-likelihood ratio statistic as

$$D = 2[l(b_{\max}|y) - l(b|y)]$$

The deviance can be decomposed as

$$D = 2[l(b_{\max}|y) - l(\beta_{\max}|y)] - 2[l(b|y) - l(\beta|y)] + 2[l(\beta_{\max} \mid y) - l(\beta|y)]$$

The first term has distribution χ^2_n, the second term has distribution χ^2_m, and the third term, $v = 2[l(\beta_{\max}|y) - l(\beta|y)]$ is a positive constant that will be near zero if the model fits the data almost as well as the saturated model. Thus, the sampling distribution of the deviance is approximately

$$D \sim \chi^2(m - p, v)$$

where v is the non-centrality parameter.

Hypothesis tests on a p-dimensional parameter β can be done with a Wald statistic $(\hat{\beta} - \beta)^T I(\hat{\beta} - \beta) \sim \chi^2_p$ or the score statistic $U^T I^{-1} U \sim \chi^2_p$. Alternatively, one can compare the goodness of fit. Consider nested models where M_0 assumes a q-dimensional parametric model and M_1 is a more general p-dimensional parametric model containing M_0. Consider the null hypothesis $H_0 : \beta_{q+1} = \beta_{q+2} = \cdots = \beta_p = 0$ with $H_1 : \exists \beta_j \neq 0, j > q$. We can test H_0 against H_1 with the difference of the deviance statistics

$$\Delta D = D_0 - D_1 = 2[l(b_1|y) - l(b_0|y)]$$

Provided certain independence conditions hold, $\Delta D \sim \chi^2_{p-q}$.

2.0.3 Normal Linear Model

The MLE is

$$b = (X^T X)^{-1} X^T y$$

provided $X^T X$ is not singular. The estimator is unbiased with a covariance matrix of $\sigma^2 (X^T X)^{-1} = I^{-1}$. σ^2 is a nuisance parameter though

$$\hat{\sigma}^2 = \frac{1}{N - p}(y - Xb)^T (y - Xb)$$

is an unbiased estimator of σ^2 and can be used to estimate I and make inference about b. Note that the MLE and least squares estimators are the same.

The deviance is given by the formula

$$D = \frac{1}{\sigma^2}(y - Xb)^T (y - Xb)$$

The residual sum of squares is sometimes called the scaled deviance since

$$\text{RSS} = (y - Xb)^T (y - Xb) = \sigma^2 D$$

It can be expanded to the form

$$D = \frac{1}{\sigma^2}(y^T y - b^T X^T y)$$

To test H_0 against H_1 for nested normal linear models,

$$\Delta D = D_0 - D_1 = \frac{1}{\sigma^2}(b_1^T X_1^T y - b_0^T X_0^T y)$$

$$F = \frac{D_0 - D_1}{p-q} \div \frac{D_1}{N-p} = \frac{b_1 X_1^T y - b_0^T X_0^T y}{p-q} \div \frac{y^T y - b_1^T X_1^T y}{N-p}$$

$F \sim F(p-q, N-p)$ under H_0. Otherwise F has a non-central distribution. Let $S_0 = y^T y - b_0^T X_0^T y$ and $S_1 = y^T y - b_1^T X_1^T y$, then

$$F = \frac{S_0 - S_1}{p-q} \div \frac{S_1}{N-p}$$

Usually inferences on a parameter for an explanatory variable depends on other explanatory variables but an exception is made when the components are orthogonal. Suppose that

$$X = [X_1, \cdots, X_m]$$

where $X_j^T X_k = 0$, the zero matrix for $j \neq k$. Then X is said to be orthogonal and $X^T X$ is a block diagonal matrix and hence

$$b^T X^T y = \sum_{k=1}^{m} b_k^T X_k^T y$$

Consequently, the hypotheses

$$H_{01} : \beta_1 = 0, \cdots, H_{0m} : \beta_m = 0$$

can be tested independently. Except for some well-designed experiments, the design matrix is not orthogonal. Tests based on all other terms being included before $X_j \beta_j$ is added is called a Type III test. Tests that depend on the sequential order of fitting terms are called Type I.

Residuals are defined as

$$\hat{e}_i = y_i - x_i^T b = y_i - \hat{\mu}_i$$

The covariance matrix is

$$E[\hat{e}\hat{e}^T] = \sigma^2[I - X(X^T X)^{-1} X^T] = \sigma^2[I - H]$$

where H is the projection or hat matrix. The standardized residuals are

$$r_i = \frac{\hat{e}_i}{\hat{\sigma}\sqrt{1 - H_{ii}}}$$

where $\hat{\sigma}^2$ is an estimate of σ^2.
An outlier is an observation that is not well fitted by the model. An influential observation is one which has a large effect on inferences based on the model. H_{ii} the ith element on the diagonal of the hat matrix is called the leverage of the ith observation. As a rule of thumb, if h_{ii} is greater than $\frac{2p}{N}$, it may be a concern. Measures combining standardized residuals and leverage include

$$\text{DFFITS}_i = r_i \sqrt{\frac{H_{ii}}{1 - H_{ii}}}$$

$$D_i = \frac{1}{p} \left(\frac{H_{ii}}{1 - H_{ii}} \right) r_i^2$$

D_i is called Cook's distance. Cook's distance may also be obtained by fitting a model with and without each observation and seeing the difference this makes to estimates b.

$$D_i = \frac{1}{p}(b - b_{(i)})^T X^T X (b - b_{(i)})$$

The basic model of multiple linear regression includes an intercept term which is represented by a column of 1's in the design matrix.
Define $S, \hat{S}, \hat{S}_0$ to be

$$S = e^T e = (Y - X\beta)^T (Y - X\beta)$$

$$\hat{S} = (y - Xb)^T(y - Xb) = y^T y - b^T X^T y$$
$$\hat{S}_0 = y^T y - N\bar{y}^2$$

Then the coefficient of determination R^2 is defined as

$$R^2 = \frac{\hat{S}_0 - \hat{S}}{\hat{S}_0} = \frac{b^T X^T y - N\bar{y}^2}{y^T y - N\bar{y}^2}$$

The square root of R^2 is called the multiple correlation coefficient.

If some explanatory variables are highly correlated with one another, this is called collinearity or multicollinearity. Multicollinearity implies that the condition number of the design matrix X is large or that the estimating equation $(X^T X)b = X^T y$ is ill-conditioned. Collinearity may be detected by the variance inflation factor (VIF)

$$\text{VIF}_j = \frac{1}{1 - R^2_{(j)}}$$

where $R^2_{(j)}$ is the coefficient of determination obtained from regressing the jth explanatory variable against all other explanatory variables. If it is uncorrelated then VIF = 1. VIF increases as the correlation increases. One should be concerned if VIF > 5.

2.0.4 Analysis of Variance

Analysis of variance is used for comparing means of groups of continuous observations where the groups are defined by the levels of factors. If experimental units are randomly allocated to groups corresponding to J levels of a factor, this is a completely randomized experiment. Responses at the same level have the same expected value and so are called replicates.

There are three different specifications of a model to test the hypothesis that the response means differ among the factor levels.

1. $E(Y_{jk}) = \mu_j, j \in [K]$.
2. $E(Y_{jk}) = \mu + \alpha_j, j \in [J]$ with sum-to-zero constraint.
3. $E(Y_{jk}) = \mu + \alpha_j$ with $\alpha_1 = 0$. Corner point parameterization.

If there are two factors A and B that are crossed, then there are 4 models to consider.

1. Saturated model. $E(Y_{jkl}) = \mu + \alpha_j\beta_k + (\alpha\beta)_{jk}$
2. Additive model. $E(Y_{jkl}) = \mu + \alpha_j + \beta_k$
3. Only A. $E(Y_{jkl}) = \mu + \alpha_j$
4. Only B. $E(Y_{jkl}) = \mu + \beta_k$

Because these models are overspecified, we must add constraints. We can impose sum-to-zero constraints or corner point constraints.

If the data is balanced, that is, there are equal number of observations in each subgroup, then it is possible to specify the design matrix in such a way that it is orthogonal.

Analysis of covariance is used for models in which some of the explanatory variables are dummy variables representing factors and other are continuous measurements called covariates. We are interested in comparing means of subgroups by factor levels after adjustment for covariate effects. The saturated model is

$$E(Y_{jk}) = \mu_j + \gamma x_{jk}$$

The reduced model is

$$E(Y_{jk}) = \mu + \gamma x_{jk}$$

The term general linear model refers to Normal linear models with any combination of categorical and continuous explanatory variables.

2.0.5 Logistic Regression

Suppose that $Y_1, Y_2, \cdots, Y_N$ are independent random variables such that $Y_i \sim \text{Bin}(n_i, \pi_i)$. Suppose that

$$\text{logit}\pi_i = \log\left(\frac{\pi_i}{1-\pi_i}\right) = x_i^T\beta$$

The estimation process is the same whether the data are grouped as frequencies for each covariate pattern or each observation is coded 0 or 1 and its covariate pattern is listed separately. The deviance is

$$D = 2\sum_{i=1}^{N}\left[y_i \log\left(\frac{y_i}{\hat{y}_i}\right) + (n_i - y_i)\log\left(\frac{n_i - y_i}{n_i - \hat{y}_i}\right)\right]$$

This has the form

$$D = 2\sum o \log\frac{o}{e}$$

where o denotes observed successes and failures and e denotes fitted values. Since D does not involve nuisance parameters, hypotheses can be directly tested with the approximation

$$D \sim \chi^2_{N-p}$$

where p is the number of parameters estimated and N the number of covariate patterns.

For small studies or situations where there are few observations for each covariate pattern, asymptotic results are poor approximations. Software has been developed for exact methods.

Instead of MLE, we can estimate the parameters by minimizing the weighted sum of squares

$$S_w = \sum_{i=1}^{N}\frac{(y_i - n_i\pi_i)^2}{n_i\pi_i(1-\pi_i)}$$

This is equivalent to minimizing the Pearson chi-squared statistic

$$X^2 = \sum\frac{(o-e)^2}{e}$$

When evaluated at the estimated expected frequencies, X^2 is asymptotically equivalent by the delta method. By analogy with R^2, we have the

$$\text{pseudo}R^2 = \frac{l(\tilde{\pi}|y) - l(\hat{\pi}|y)}{l(\tilde{\pi}|y)}$$

where $\tilde{\pi} = \frac{\sum y_i}{\sum n_i}$ is from the minimal model. The pseudo-R^2 represents the proportional improvement in the log-likelihood due to the terms in the model of interest.

$$\text{AIC} = -2l(\hat{\pi}|y) + 2p$$

$$\text{BIC} = -2l(\hat{\pi}|y) + 2p\log n$$

The Pearson or chi-squared residual is

$$X_k = \frac{y_k - n_k\hat{\pi}_k}{\sqrt{n_k\hat{\pi}_k(1-\hat{\pi}_k)}}$$

The standardized Pearson residuals are

$$r_{P_k} = \frac{X_k}{\sqrt{1-h_k}}$$

where h_k is the leverage, obtained from the hat matrix. Deviance residuals are defined as

$$d_k = \text{sgn}(y_k - n_k\hat{\pi}_k)\left\{2\left[y_k \log\left(\frac{y_k}{n_k\hat{\pi}_k}\right) + (n_k - y_k)\log\left(\frac{n_k - y_k}{n_k - n_k\hat{\pi}_k}\right)\right]\right\}^{1/2}$$

Note that $\sum_{k=1}^{n} d_k^2 = D$, the deviance. Standardized deviance residuals are defined by

$$r_{D_k} = \frac{d_k}{\sqrt{1-h_k}}$$

Pearson and deviance residuals can be used for checking the adequacy of a model. They should be plotted against each continuous explanatory variable to check the assumption of linearity and against other explanatory variables not included in the model. They should be plotted in the order of measurements to check for serial correlation. Normal probability plots can also be used for large enough n since the standardized residuals should be approximately standard normal. If the data is binary or if n_k is small, then the plots may be uninformative and aggregated goodness of fit statistics as well as other diagnostics should be used.

For binary data, there may be overdispersion which may be due to inadequate specification of the model or more complex structure. One approach is to add an extra parameter ϕ so that $\text{var}(Y_i) = n_i\pi_i(1-\pi_i)\phi$, called the quasibinomial distribution.

Consider a random variable Y with J categories. Let $\pi_1, \pi_2, \cdots, \pi_J$ denote their respective probabilities. If there are n independent observations of Y resulting in $y_1, y_2, \cdots, y_J$ outcomes in each category, the multinomial distribution is

$$f(y \mid n) = \frac{n!}{y_1!y_2!\cdots y_J!}\pi_1^{y_1}\pi_2^{y_2}\cdots\pi_J^{y_J}$$

It can be shown that $E(Y_j) = n\pi_j$, $\text{var}(Y_j) = n\pi_j(1-\pi_j)$ and $\text{cov}(Y_j, Y_k) = -n\pi_j\pi_k$. Nominal logistic regression models are used when there is no natural order among the response categories. One category is arbitrarily chosen as the reference category. Then,

$$\text{logit}(\pi_j) = \log\left(\frac{\pi_j}{\pi_1}\right) = x_j^T\beta_j$$

The $(J-1)$ logit equations are used simultaneously to estimate the parameters β_j.

$$\hat{\pi}_j = \hat{\pi}_1 \exp(x_j^T b_j)$$

$$\hat{\pi}_1 = \frac{1}{1 + \sum_{j=2}^{J}\exp(x_j^T b_j)}$$

$$\hat{\pi}_j = \frac{\exp(x_j^T b_j)}{1 + \sum_{j=2}^{J}\exp(x_j^T b_j)}$$

The Pearson chi-squared residuals are

$$r_i = \frac{o_i - e_i}{\sqrt{e_i}}$$

Chi-squared statistic is

$$X^2 = \sum_{i=1}^{N} r_i^2$$

Deviance is $D = 2[l(b_{\max}) - l(b)]$ and likelihood ratio chi-squared statistic is

$$C = 2[l(b) - l(b_{\min})]$$

$$\text{Pseudo}R^2 = \frac{l(b_{\min}) - l(b)}{l(b_{\min})}$$

$$\text{AIC} = -2l(\hat{\pi} \mid y) + 2p$$

The odds ratio is

$$\text{OR}_j = \frac{\pi_{jp}}{\pi_{ja}} \div \frac{\pi_{1p}}{\pi_{1a}}$$

where π_{jp} and π_{ja} denote probabilities of response category according to whether exposure is present or absent, respectively.

In some situations, there may be a continuous variable z that is difficult to measure and may be assessed by cut points for the latent variable so that small values are classified as none, larger values as moderate, and high values as severe. The cutpoints $C_1, C_2, \cdots, C_J$ define J ordinal categories with associated probabilities.

The cumulative odds for the jth category are

$$\frac{P(z \leq C_j)}{P(z > C_j)} = \frac{\pi_1 + \pi_2 \cdots + \pi_j}{\pi_{j+1} + \cdots + \pi_J}$$

The cumulative logit model is

$$\log \frac{\pi_1 + \cdots + \pi_j}{\pi_{j+1} + \cdots + \pi_J} = x_j^T \beta_j$$

The proportional odds model is

$$\log \frac{\pi_1 + \cdots + \pi_j}{\pi_{j+1} + \cdots + \pi_J} = \beta_{0j} + \beta_1 x_1 + \cdots + \beta_{p-1} x_{p-1}$$

It is based on the assumption that the effects of the covariates $x_1, \cdots, x_{p-1}$ are the same for all categories on the logarithmic scale. If some of the categories are amalgamated, this does not change the parameter estimates $\beta_1, \cdots, \beta_{p-1}$ but the terms β_{0j} will be affected. This is the collapsibility property. The proportional odds model is also not affected by a reversal of labelling of the categories. The proportional odds model is the usual form of ordinal logistic regression.

2.0.6 Poisson Regression

Assume $Y_i \sim \text{Poisson}(\mu_i)$ with

$$\log \mu_i = \log n_i + x_i^T \beta$$

This differs from the usual specification of the linear component due to the $\log n_i$ term, called the offset. It is a known constant.

The rate ratio, RR for presence vs absence is

$$\text{RR} = \frac{E(Y_i \mid \text{present})}{E(Y_i \mid \text{absent})} = e^{\beta_j}$$

The Pearson residuals are

$$r_i = \frac{o_i - e_i}{\sqrt{e_i}}$$

The chi-squared goodness of fit statistic is related by

$$X^2 = \sum_i r_i^2 = \sum_i \frac{(o_i - e_i)^2}{e_i}$$

The deviance for a Poisson model is

$$D = 2 \sum_i [o_i \log(o_i/e_i) - (o_i - e_i)]$$

For most models $\sigma_i o_i = \sigma_i e_i$ so this simplifies to

$$D = 2 \sum_i [o_i \log(o_i/e_i)]$$

The deviance residuals are the components of D,

$$d_i = \text{sign}(o_i - e_i) \sqrt{2[o_i \log(o_i/e_i) - (o_i - e_i)]}$$

so that $D = \sum_i d_i^2$.

2.0.7 Q-Q and Box Plots

A five-number summary of the data consists of the minimum, the first quartile, the median, the third quartile, and the maximum.

The boxplot encloses the middle 50% of the data, with a line segment to indicate the median, and extending lines to the minimum and maximum. Box and whisker plots also have the middle box but extend to a lower fence and upper fence where

$$\text{LF} = Q_1 - 1.5(Q_3 - Q_1), \quad \text{UF} = Q_3 + 1.5(Q_3 - Q_1)$$

Points that lie outside the fences are called potential outliers. Points on the line are called adjacent points.

Q-Q plots plot sample quantiles against quantiles from a theoretical CDF. If the plot is mostly on the identity line $y = x$, then we see the two distributions as being similar. Otherwise, we suspect that the two distributions are not the same.

3 Time Series

3.1 Basics and Box-Jenkins

Consider a time series model that is stationary in the mean and variance. The model is second-order stationary if the correlation between variables only depends on the number of time steps separating them. The number of time steps between the variables is called the lag. A correlation of a variable

with itself at different times is called autocorrelation or serial correlation. If a time series model is second-order stationary, we define the autocovariance function (avcf), γ_k as

$$\gamma_k = E[(x_t - \mu)(x_{t+k} - \mu)]$$

The autocorrelation function (acf), ρ_k is

$$\rho_k = \frac{\gamma_k}{\sigma^2}$$

The sample acvf, c_k is calculated as

$$c_k = \frac{1}{n}\sum_{t=1}^{n-k}(x_t - \bar{x})(x_{t+k} - \bar{x})$$

The sample acf is defined as

$$r_k = \frac{c_k}{c_0}$$

The correlogram is a plot of r_k against k. If $\rho_k = 0$, the sampling distribution of r_k is approximately normal with a mean of $-n^{-1}$ and variance of n^{-1}. The dotted lines on a correlogram are drawn at

$$-\frac{1}{n} \pm \frac{1.96}{\sqrt{n}}$$

If r_k falls outside the lines, there is evidence against the null hypothesis $\rho_k = 0$, though we expect 5% of the estimates r_k to fall outside the lines even under the null hypothesis.

Suppose there are time series models for variables x and y that are stationary in the mean and variance. Their combined model is second-order stationary if correlations depend only on the lag, and we may define the cross-covariance function (ccvf) as

$$\gamma_k(x, y) = E[(x_{t+k} - \mu_x)(y_t - \mu_y)]$$

This is not a symmetric relationship, and x is lagging y by k.

$$\gamma_k(x, y) = \gamma_{-k}(y, x)$$

The lag k cross-correlation function ccf is defined by

$$\rho_k(x, y) = \frac{\gamma_k(x, y)}{\sigma_x \sigma_y}$$

The sample ccvf is defined as

$$c_k(x, y) = \frac{1}{n}\sum_{t=1}^{n-k}(x_{t+k} - \bar{x})(y_t - \bar{y})$$

The sample acf is defined as

$$r_k(x, y) = \frac{c_k(x, y)}{\sqrt{c_0(x, x)c_0(y, y)}}$$

A time series $\{w_t \mid t \in [n]\}$ is discrete white noise (DWN) if variables $w_1, w_2, \cdots, w_n$ are i.i.d. with a mean of zero. If in addition, the variables are normal, the series is called Gaussian white noise.

Let $\{x_t\}$ be a time series. It is a random walk if

$$x_t = x_{t-1} + w_t$$

where $\{w_t\}$ is a white noise series. Back substitution gives

$$x_t = \sum_{i=1}^{t} w_i$$

A random walk satisfies $\mu_x = 0, \gamma_k(t) = \text{cov}(x_t, x_{t+k}) = t\sigma^2$.
The backward shift operator is defined by

$$Bx_t = x_{t-1}$$

Also called the lag operator. The difference operator ∇ is defined by

$$\nabla x_t = x_t - x_{t-1}$$

Differencing can be a useful filtering procedure for non-stationary time series. The first-order differences of a random walk is a white noise series so the correlogram of the series of differences can be used to assess whether a given series is a random walk.

A random walk model with a drift parameter δ is

$$x_t = x_{t-1} + \delta + w_t$$

The series $\{x_t\}$ is an autoregressive process of order p, abbreviated AR(p) if

$$x_t = \alpha_1 x_{t-1} + \alpha_2 x_{t-2} + \cdots + \alpha_p x_{t-p} + w_t$$

where $\{w_t\}$ is white noise, α_i are model parameters with $\alpha_p \neq 0$ for an order p process. It can also be expressed as

$$\theta_p(B)x_t = (1 - \alpha_1 B - \alpha_2 B^2 - \cdots - \alpha_p B^p)x_t = w_t$$

The equation $\theta_p(B) = 0$ is called the characteristic equation. The roots of the characteristic equation must all exceed unity in absolute value for the process to be stationary. To derive the second-order properties for AR(1), note that

$$(1 - \alpha B)x_t = w_t \implies x_t = (1 - \alpha B)^{-1} w_t$$

$$x_t = \sum_{i=0}^{\infty} \alpha_i w_{t-i}$$

Hence $E(x_t) = \sum_{i=0}^{\infty} \alpha_i E(w_{t-i}) = 0$ and

$$\gamma_k = \text{cov}(x_t, x_{t+k}) = \frac{\alpha^k \sigma^2}{1 - \alpha^2}$$

The autocorrelation function follows as

$$\rho_k = \alpha^k$$

From this equation we see that autocorrelations are nonzero for all lags even though the underlying model x_t only depends on the previous value x_{t-1}. The partial autocorrelation at lag k is the correlation that results after removing the effect of any correlations due to the terms at shorter lags.

An AR(p) model can be fitted in `R` using the `ar` function. The method is generally fit with MLE and the order p of the process is chosen with AIC.

A time series model $\{x_t\}$ is linear if it can be expressed as

$$x_t = \alpha_0 + \alpha_1 u_{1,t} + \cdots + \alpha_m u_{m,t} + z_t$$

where $u_{i,t}$ is the value of the ith predictor at time t, z_t is the error at time t, and $\alpha_0, \alpha_1, \cdots, \alpha_m$ are model parameters, estimated by least squares.

Linear models for time series are non-stationary when they include functions of time. Differencing can often transform a non-stationary series with a deterministic trend to a stationary series. If the underlying trend is a polynomial of order m, then the mth order differencing is required to remove the trend.

If $\{x_t\}$ is a stationary time series with $Ex_t = \mu$, $\text{var}(x_t) = \sigma^2$, $\text{cor}(x_t, x_{t+k}) = \rho_k$, then the variance of the sample mean is

$$\text{var}(\bar{x}) = \frac{\sigma^2}{n}\left[1 + 2\sum_{k=1}^{n-1}(1 - \frac{k}{n})\rho_k\right]$$

Thus if $\rho_k > 0$, then the variance of the sample mean is greater than the independent case and if $\rho_k < 0$, the variance of the sample mean is smaller than the independent case. Since in time series regression, the residual series will be autocorrelated, we should use generalized least squares (GLS) instead for better estimates of the standard errors of the regression parameters.

Suppose a time series contains s seasons. A seasonal indicator model for a time series $\{x_t\}$ containing s seasons and a trend m_t is given by

$$x_t = m_t + s_t + z_t$$

where $s_t = \beta_i$ when t is in the ith season and $\{z_t\}$ is the residual error series, which may be autocorrelated. This may be also written as

$$x_t = m_t + \beta_{1+(t-1) \bmod s} + z_t$$

By treating the seasonal term s_t as a factor, the parameters for the model can be estimated by GLS.

If seasonal effects vary smoothly over the seasons, it may be more parameter efficient to use a smooth function instead of separate indices. The harmonic seasonal model is

$$x_t = m_t + \sum_{i=1}^{\lfloor s/2 \rfloor}[s_i \sin(2\pi it/s) + c_i \cos(2\pi it/s)] + z_t$$

where m_t is the trend with a constant term, s_i and c_i are unknown parameters.

The logarithm can be used to transform a multiplicative model into an additive model. If a time series is given by

$$x_t = m'_t s'_t z'_t$$

then applying the logarithm gives

$$y_t = \log x_t = \log m'_t + \log s'_t + \log z'_t = m_t + s_t + z_t$$

The process of using a transformation, fitting a model, then applying an inverse transformation introduces a bias in the forecasts. For the logarithm, an empirical correction factor may be applied for forecasting.

$$\hat{x}'_t = e^{\widehat{\log x_t}} \sum_{t=1}^{n} \frac{e^{z_i}}{n}$$

A time series model $\{x_t\}$ is said to be strictly stationary if the joint distribution of $\{x_{t_1}, \cdots, x_{t_n}\}$ is the same as the joint distribution of $\{x_{t_1+m}, \cdots, x_{t_n+m}\}$ for all $t_1, t_2, \cdots, t_n$ and m.

Regression can allow us to decompose a non-stationary series to a trend, seasonal components, and residual series. The residual series can be treated as a realization of a stationary error series.

A moving average (MA) process of order q is a linear combination of current white noise and q most recent past white noise terms.

$$x_t = w_t + \beta_1 w_{t-1} + \cdots + \beta_q w_{t-q}$$

where $\{w_t\}$ is white noise with variance σ_w^2.

$$x_t = (1 + \beta_1 B + \beta_2 B^2 + \cdots + \beta_q B^q) w_t = \phi_q(B) w_t$$

where ϕ_q is a polynomial of order q. The mean is clearly zero and the variance is $\sigma_w^2(1+\beta_1^2+\cdots+\beta_q^2)$. The autocorrelation function is

$$\rho_k = \frac{\sum_{i=0}^{q-k} \beta_i \beta_{i+k}}{\sum_{i=0}^{q} \beta_i^2}, k \in [q]$$

and $\rho_0 = 1$, $\rho_k = 0$ for $k > q$. The MA process is invertible if it can be expressed as

$$w_t = (1 - \beta B)^{-1} x_t = x_t + \beta x_{t-1} + \beta^2 x_{t-2} + \cdots$$

provided $|\beta| < 1$. An MA(q) process is invertible when all roots of $\phi_q(B)$ all exceed unity in absolute value.

When AR and MA terms are added together, it is an ARMA process. An ARMA(p, q) is

$$x_t = \alpha_1 x_{t-1} + \alpha_2 x_{t-2} + \cdots + \alpha_p x_{t-p} + w_t + \beta_1 w_{t-1} + \beta_2 w_{t-2} + \cdots + \beta_q w_{t-q}$$

where $\{w_t\}$ is white noise.

$$\theta_p(B) x_t = \phi_q(B) w_t$$

Some properties of an ARMA(p, q) process:

1. The process is stationary when all the roots of θ exceed unity in absolute value.
2. The process is invertible when all roots of ϕ exceed unity in absolute value.
3. The AR(p) model is the special case ARMA($p, 0$)
4. The MA(q) model is the special case ARMA($0, q$).
5. When fitting data, an ARMA model will often be more parameter efficient than a single MA or AR model.
6. When θ and ϕ share a common factor, a stationary model can be simplified.

A series $\{x_t\}$ is integrated of order d, denoted $I(d)$, if $\nabla^d x_t = w_t$. Since $\nabla = 1 - B$, $\{x_t\}$ is integrated of order d iff

$$(1 - B)^d x_t = w_t$$

A time series $\{x_t\}$ is an ARIMA(p, d, q) process if the dth differences of $\{x_t\}$ series are an ARMA(p, q) process.

$$\theta_p(B)(1 - B)^d x_t = \phi_q(B) w_t$$

A seasonal ARIMA model uses differencing at a lag equal to the number of seasons s to remove additional seasonal effects. The seasonal ARIMA$(p, d, q)(P, D, Q)_s$ model is expressed as

$$\Theta_P(B^s)\theta_p(B)(1 - B^s)^d(1 - B)^d x_t = \Phi_Q(B^s)\phi_q(B) w_t$$

3.2 Smoothing

The basic running average estimate is defined by

$$\hat{s}_t = \frac{1}{k}\sum_{i=1}^{k} y_{t-i+1}$$

where k is the running average length. The choice of k depends on the smoothing desired. The larger the k, the smoother is $\hat{s}_t$. To forecast the series,

$$\hat{s}_t = \hat{s}_{t-1} + \frac{y_t - y_{t-k}}{k}$$

If there are no trends in the data, then the second term is zero. For a series that can be expressed as

$$y_t = \beta_0 + \beta_1 t + \epsilon_t,$$

we can use a double smoothing procedure:

1. Create a smoothed series $\hat{s}_t^{(1)} = \frac{1}{k}\sum_{i=1}^{k} y_{t-i+1}$.
2. Create a doubly smoothed series $\hat{s}_t^{(2)} = \frac{1}{k}\sum_{i=1}^{k} \hat{s}_{t-i+1}^{(1)}$.

This procedure smooths out the effect of a linear trend in time. The estimate of the trend is

$$b_{1,T} = 2\frac{\hat{s}_T^{(1)} - \hat{s}_T^{(2)}}{k-1}$$

The resulting forecasts are

$$\hat{y}_{T+l} = \hat{s}_T + b_{1,T} l$$

Running averages can be expressed as weighted least squares estimates. By taking a weight of one for $t = T-k+1, \cdots, T$, the weighted least squares estimator returns the running average estimate. This model is called a locally constant mean model.

Exponential smoothing takes the form

$$\hat{s}_t = (1-w)\sum_{i=0}^{\infty} w^i y_{t-i}$$

Because observations are not available in the infinite past, we take the truncated version

$$\hat{s}_t = (1-w)\sum_{i=0}^{t} w^i y_{t-i}$$

This is the exponential smoothed estimate of the series. Like running average estimates, the smoothed estimates provide greater weights to more recent observations. Exponential smoothing has the recursive equation

$$\hat{s}_t = \hat{s}_{t-1} + (1-w)(y_t - \hat{s}_{t-1}) = (1-w)y_t + w\hat{s}_{t-1}$$

To forecast, we should use $\hat{y}_{T+l} = \hat{s}_T$. To decide w, we minimize the sum of squared one-step prediciton errors

$$\mathrm{SS}(w) = \sum_{t=1}^{T}(y_t - \hat{s}_{t-1})^2$$

For a linear trend in time, we can use a double exponential smoothing:

1. Take $\hat{s}_t^{(1)} = (1-w)y_t + w\hat{s}_{t-1}^{(1)}$.
2. Take $\hat{s}_t^{(2)} = (1-w)\hat{s}_t^{(1)} + w\hat{s}_{t-1}^{(2)}$.

The estimate of the trend is $b_{1,T} = \frac{1-w}{w}(\hat{s}_T^{(1)} - \hat{s}_T^{(2)})$. The forecasts are given by $\hat{y}_{T+l} = b_{0,T} + b_{1,T}l$, where the intercept is $b_{0,T} = 2\hat{s}_T^{(1)} - \hat{s}_T^{(2)}$.

Similar to running average smoothing, exponential smoothed estimates are WLS estimates with weights given by $w_t = w^{T-t}$. Exponential smoothing estimates are also called discounted least squares estimates.

Holt introduced the following generalization of the double exponential smoothing method for seasonal data. Let w_1 and w_2 be smoothing parameters and calculate recursively the estimates:

$$b_{0,t} = (1-w_1)y_t + w_1(b_{0,t-1} + b_{1,t-1})$$

$$b_{1,t} = (1-w_2)(b_{0,t} - b_{0,t-1}) + w_2 b_{1,t-1}$$

These estimates forecast the linear trend model, $y_t = \beta_0 + \beta_1 t + \epsilon_t$. This is a generalization since the same weight need not be used for both β_0 and β_1. Winters extended the Holt procedure to accommodate seasonal trends. The Holt-Winter seasonal additive model is

$$y_t = \beta_0 + \beta_1 t + S_t + \epsilon_t$$

where $S_t = S_{t-g}$, $\sum_{i=1}^{g} S_i = 0$. We employ three smoothing parameters. The parameter estimates for the model are determined recursively by

$$b_{0,t} = (1-w_1)(y_t - \hat{S}_{t-g}) + w_1(b_{0,t-1} + b_{1,t-1})$$

$$b_{1,t} = (1-w_2)(b_{0,t} - b_{0,t-1}) + w_2 b_{1,t-1}$$

$$\hat{S}_t = (1-w_3)(y_t - b_{0,t}) + w_3\hat{S}_{t-g}$$

Forecasts are determined using

$$\hat{y}_{T+l} = b_{0,T} + b_{1,T}l + \hat{S}_t(l)$$

where $\hat{S}_T(l) = \hat{S}_{T+l_1}$, $l \equiv l_1 (\text{mod } g)$. To compute the recursive estimates, we must decide on the initial starting values and a choice of smoothing parameters. To determine the initial starting values, it is recommended to fit a regression equation to the first portion of the data. The regression equation includes a linear trend in time and $g-1$ binary variables for seasonal variation. Only $g+1$ observations are required to determine the initial estimates $b_{0,0}, b_{1,0}, y_{1-g}, y_{2-g}, \cdots, y_0$. On the choice of smoothing parameters, analysts rely on rule of thumbs. Cryer and Miller recommend $w_1 = w_2 = 0.9$, $w_3 = 0.6$.

3.3 Unit Root Test

Consider the model

$$y_t = \mu_0 + \phi(y_{t-1} - \mu_0) + \mu_1(\phi + (1-\phi)t) + \epsilon_t$$

When $\phi = 1$, this is a random walk model with $y_t = \mu_1 + y_{t-1} + \epsilon_t$. When $\phi < 1$ and $\mu_1 = 0$, this is an AR(1) model. When $\phi = 0$, this is a linear trend in time model. It is customary to use least squares on the model

$$y_t - y_{t-1} = \beta_0 + (\phi - 1)y_{t-1} + \beta_1 t + \epsilon_t$$

where $\beta_0 = \mu_0(1-\phi) + \phi\mu_1$ and $\beta_1 = \mu_1(1-\phi)$. From this regression, let t_{DF} denote the t-statistic associated with the y_{t-1} variable. We wish to use the t-statistic to test the null hypothesis $H_0 : \phi = 1$ versus $H_1 : \phi < 1$. Because $\{y_{t-1}\}$ is a random walk under the null-hypothesis, the distribution of t_{DF}

is not the usual t-distribution. This test is called the Dickey-Fuller test. A criticism of the Dickey-Fuller test is that the disturbance term is assumed to be serially uncorrelated. The augmented Dickey-Fuller test statistic is the t-statistic associated with the y_{t-1} variable using OLS on the equation

$$y_t - y_{t-1} = \beta_0 + (\phi - 1)y_{t-1} + \beta_1 t + \sum_{j=1}^{p} \phi_j (y_{t-j} - y_{t-j-1}) + \epsilon_t$$

In this equation, the disturbance term is augmented by the autoregressive terms in the differences $y_{t-j} - y_{t-j-1}$. The idea is that these terms capture serial correlation in the disturbance term. Consensus is not reached on choosing the number of lags p. Analysts provide results of the test statistic for a number of choices of lags and hope that conclusions are similar.

3.4 ARCH/GARCH models

Let Ω_t denote the information set, the collection of knowledge about the process up to and including time t. We allow the variance to depend on time t by conditioning on the past,

$$\sigma_t^2 = \text{Var}(\epsilon_t \mid \Omega_{t-1})$$

We first present the autoregressive changing heteroscedasticity model of order q, ARCH(q). Assume ϵ_t given Ω_{t-1} is normally distributed with mean zero and variance σ_t^2. Further assume that the conditional variance is determined recursively by

$$\sigma_t^2 = w + \gamma_1 \epsilon_{t-1}^2 + \cdots + \gamma_q \epsilon_{t-q}^2 = w + \gamma(B)\epsilon_t^2$$

The term $w > 0$ is the long-run volatility parameter and $\gamma_1, \gamma_2, \cdots, \gamma_q$ are coefficients such that $\gamma_j \geq 0$ and $\gamma(1) = \sum_{j=1}^{q} \gamma_j < 1$. If $p = 1$, a large change to ϵ_{t-1}^2 can induce a large conditional variance σ_t^2. Higher orders of q capture longer-term effects. Despite having a changing conditional variance, the unconditional variance remains constant over time.

Next, we discuss the generalized ARCH model of order p, GARCH(p, q). As with the ARCH model, we assume the distribution of ϵ_t given Ω_{t-1} is normally distributed with mean zero and variance σ_t^2. It is determined recursively by

$$\sigma_t^2 - \delta_1 \sigma_{t-1}^2 - \cdots - \delta_p \sigma_{t-p}^2 = w + \gamma_1 \epsilon_{t-1}^2 + \cdots + \gamma_q \epsilon_{t-q}^2$$

or $\sigma_t^2 = w + \gamma(B)\epsilon_t^2 + \delta(B)\sigma_t^2$. In addition to ARCH($q$) requirements, we also need $\delta_j \geq 0$ and $\gamma(1) + \delta(1) < 1$. The GARCH($p, q$) model is also weakly stationary, with mean zero and variance $\text{Var}\epsilon_t = \frac{w}{1-\gamma(1)-\delta(1)}$.