Chapter 6 Linear Model Selection and Regularization

Alternative fitting procedures can yield better prediction accuracy and model interpretability.

1. Prediction accuracy:

Bias and variance in least squares approach:

1. Provided that the true relationship between the response and the predictors are approximately linear, least squares estimates will have low bias.
2. If 🡪the number of observations() is much larger than the number of variables()🡪The least squares estimates tend to have a low variance, will perform well on test observations
3. If 🡪 then there can be a lot of variability in the least squares fit, resulting in overfitting and consequently poor predictions
4. If 🡪There is no longer a unique least squares coefficient estimate: the variance is infinite so the method cannot be used at all
   1. By constraining or shrinking the estimated coefficients, we can often substantially reduce the variance at the cost of negligible increase in bias
5. Model Interpretability:
6. Irrelevant variables: Variables in multiple linear regression not associated with the response🡪lead to unnecessary complexity in the resulting model.
   1. Feature selection/variable selection: Excluding irrelevant variables from the multiple regression model

Alternatives to least squares fit:

1. Subset Selection: Involves identifying a subset of the p predictors that we believe to be related to the response
2. Shrinkage: Involves fitting a model involving all *p*🡪 Estimated coefficients are shrunken towards zero relative to the least squares estimates🡪Shrinkage(regularization) has the effect of reducing variance🡪 Depending on what type of shrinkage is performed, some of the coefficients may be estimated to be exactly zero
3. Dimension Reduction: Involves projecting *p* predictors into *M-*dimensional subspace, where *M<p*.(Achieved by computing *M* different *linear combinations*, or *projections*, of the variables)

6.1 Subset Selection

6.1.1 Best Subset Selection

Fit a separate least squares regression for each possible combination of the *p* predictors.

A screenshot of a cell phone

Description automatically generated

Step 3 A low RSS or high indicates a model with a low training error, whereas we wish to choose a model that has a low test error.

1. Because the RSS of these models decreases monotonically, and the increases monotonically, as the number of features included in the models increases.
2. Use cross-validated prediction error, , in order to select among ,…,
3. Logistics regression best selection: Use deviance🡪The smaller the deviance, the better the fit.

Computation Limitation for Best Subset Selection:

1. Number of possible models that must be considered grows rapidly as *p* increases🡪In general, there are models that involve subsets of predictors.

6.1.2 Stepwise Selection

An enormous search space can lead to overfitting and high variance of the coefficient estimates.

Forward Stepwise Selection

1. Begins with a model containing no predictors, and then adds predictors to the model, one-at-a-time, until all of the predictors are in the model.
2. At each step the variable that gives the greatest additional improvement to the fit is added to the model.

A screenshot of a cell phone

Description automatically generated

1. Forward stepwise selection involves fitting one null model, along with models in the iteration, for .

1. In step 2, we can choose the model with the lowest RSS or the highest
2. In step 3, we must identify the best model among a set of models with different numbers of variables.

Forward stepwise selection can be applied even in the high-dimensional setting where :

* 1. Can construct ,…, models only, each submodel is fit using least squares, which will not yield a unique solution if

Backward Stepwise Selection:

1. Begins with the full least squares model containing all predictors, and then iteratively removes the least useful predictor one-at-a-time

A screenshot of a cell phone

Description automatically generated

1. Searches through selection approaches through

models, and so can applied in settings where is too large to apply best subset selection.

1. Backward stepwise selection is not guaranteed to yield the best model containing a subset of the *p* predictors.
   1. Backward selection requires that the number of samples *n* is larger than the number of variables *p*(so that the full model can be fit)

Hybrid Approaches

1. Variables are added to the model sequentially, in analogy to forward selection.
2. Adding each new variable, the method may also remove any variables that no longer provide an improvement to the model fit.

6.1.3 Choosing the Optimal Model

1. RSS and are not suitable for selecting the best model among a collection of models with different number of predictors

a. We can directly estimate test error by making an adjustment to the training error to account for the bias due to overfitting.

b. We can directly estimate the test error, using either a validation set approach or a cross-validation approach.

:

1. Training set MSE is generally an underestimate of the test MSE
   * 1. We specifically estimate the regression coefficients such that the training RSS is as small as possible
     2. Training error will decrease as more variables are included in the model(test error may not)

Techniques for *adjusting* the training error for the model size available:

:

For a fitted least squares model containing *d* predictors,



1. is an estimate of the variance of the error associated with each response measurement.
2. statistic adds a penalty of to the training RSS in order to adjust the fact that training error tends to underestimate the test error🡪Penaluty increases as the number of predictors increase
3. tends to take on a small value for models with a low test error

:

1. Defined for a large class of models fit by maximum likelihood
2. Formula of AIC:

A picture containing drawing

Description automatically generated

\*For least squares models, and are proportional to each other

:

1. Formula of BIC:

A close up of a logo

Description automatically generated

1. BIC will tend to take on a small value for a model with a low test error
2. The BIC statistics generally places a heavier penalty on models with many variables, and hence results in the selection of smaller models than

:

For a least squares model with d variables: the adjusted statistic is calculated as

A picture containing object, clock

Description automatically generated

1. A large value of adjusted indicates a model with low test error
2. Once al of the correct variables have been included in the mode, adding additional noise variables will lead to only a very small decrease in RSS.
3. Adjusted will have only correct variables and no noise variables🡪 Pays a price for the inclusion of unnecessary variables in the model

Validation and Cross-Validation

1. Compute the validation set error or the cross-validation error for each model under consideration🡪Select the model for which resulting estimated test error is smallest
2. Provides a direct estimate of the test error
3. Can be used in a wider range of model selection tasks, even in cases where it is hard to pinpoint the model degrees of freedom or hard to estimate the error variance
4. Can be computationally intensive
5. One standard error rule:
   1. First calculate the standard error of the estimates test MSE for each model size
   2. Select the smallest model for which the estimated test error is within one standard error of the lowest point on the curve

6.2 Shrinkage Methods

Fit a model containing all *p* predictors using a technique that constrains or regularizes the coefficient estimates, or equivalently, that shrinks the coefficient estimates toward zero.

\*Shrinking estimates significantly reduce their variance

6.2.1 Ridge Regression

The ridge regression coefficient estimates are the values that minimize:

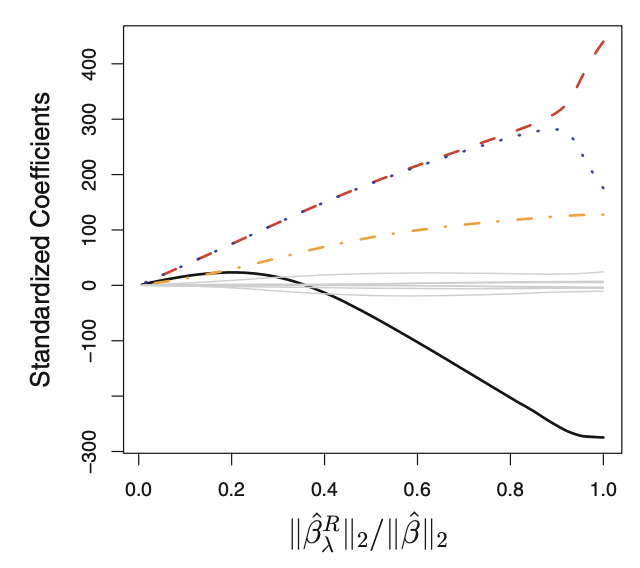
A close up of a clock

Description automatically generated

* + 1. Shrinkage penalty: Small when ,…,🡪Has the effect of shrinking the estimates toward zero.
    2. Tuning parameter  serves to control the relative impact of these two terms
    3. As , the impact of shrinkage penalty grows, and the ridge regression coefficient estimates will approach 0
    4. The shrinking penalty is not applied to the intercept🡪Intercept is simply a measure of the mean value of the response when

An application to the Credit Data

1. Null model: As is extremely large, then all of the ridge coefficient estimates are basically 0.
2. Individual coefficients may occasionally increase as increase



:

1. is the of the vector, defined as

A picture containing object, clock, drawing

Description automatically generated

1. As increases, the of will always decrease and so will
2. Standard least squares coefficient estimates are scale equivalent:
   1. Regardless of how the predictor is scaled, will remain the same.
3. The ridge regression coefficients can change substantially when multiplying a given predictor by a constant
   1. Ridge regression can only be applied after standardizing the predictors, using:

A picture containing object, clock

Description automatically generated

\*Denominator is the estimated standard deviation of the *j*th predictor🡪 All of the standardized predictors will have a standard deviation of one

Why Does Ridge Regression Improve Over Least Squares?

Bias Variance Trade-off:

1. As increases, the flexibility of the ridge regression fit decreases, leading to decreased variance but increased bias.
2. As increase., the shrinkage of the ridge coefficient estimates leads to a substantial reduction in the variance of the predictions, at the expense of a slight increase in bias.
3. Beyond the critical point, the decrease in variance due to the increase in slows, and the shrinkage on the coefficient causes them to be significantly underestimated, resulting in a large increase in bias.

Least Squares vs. Ridge:

1. When the relationship between response and predictors is close to linear, the least squares estimates will have low bias but may have high variance
   1. A small change in the training data can cause a large change in the least squares coefficient estimates
2. When the number of variables *p* is almost as large as the number of observations *n*🡪Least squares estimates will be extremely variable
3. If *p>n*, the least squares will not have a unique solution, but ridge regression can still perform well.

Ridge Regression vs Best Subset Selection:

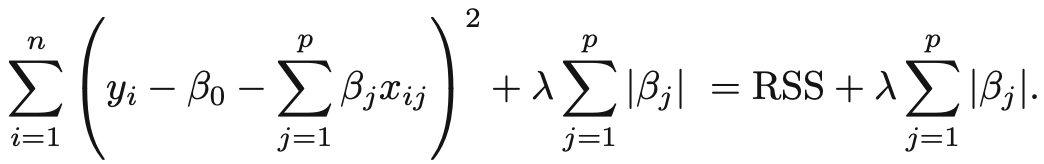
1. Ridge regression is less computationally intense, only fits a single model

6.2.2. The Lasso

Disadvantage of Ridge Regression:

1. Include all *p* predictors in the final model🡪Penalty term only shrinks the coefficients towards zero, but it will not set any of them to exactly zero

Lasso coefficients, , minimize the quantity:



* Lasso uses an penalty instead of an penalty
* The norm of a coefficient vector is given by

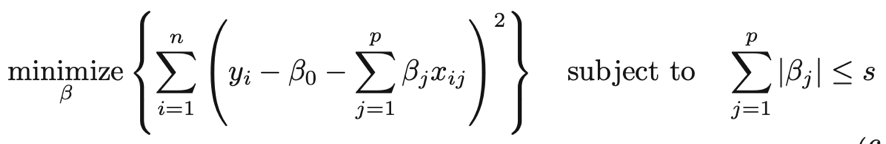
A picture containing table, stool, drawing

Description automatically generated

* penalty term has the effect of forcing some of the coefficient estimates to be exactly equal to zero when the tunning parameter is sufficiently large🡪Lasso performs variable selection
* Lasso yields sparse models—models that involve only a subset of the variables
  1. When becomes sufficiently large, the lasso gives the null model in which all coefficient estimates equal to zero
  2. Depending on the value of , the lasso can produce a model involving any number of variables

Another Formulation for Ridge Regression and the Lasso:

Lasso:



For every value of , there is some *s* such that above equation and the lasso equation will give the same estimates

1. We perform lasso to find the lowest RSS, subject to the constraint that there is a *budget s* for how large can be
2. The larger the budget s, the closer lasso to the least squares solution

Ridge Regression:

A picture containing object, clock

Description automatically generated

For every value of , there is some *s* such that above equation and the ridge equation will give the same estimates

1. We perform ridge to find the lowest RSS, subject to the constraint that there is a *budget s* for how large can be
2. The larger the budget s, the closer ridge to the least squares solution

Best Subset Selection:

A picture containing table, clock

Description automatically generated

Finding a set of coefficient estimates such that RSS is as small as possible, subject to the constraint that no more than *S* coefficients can be nonzero

1. We can interpret ridge regression and lasso as computationally feasible alternatives to best subset selection

The Variable Selection Property of the Lasso

Lasso and Ridge

A close up of a logo

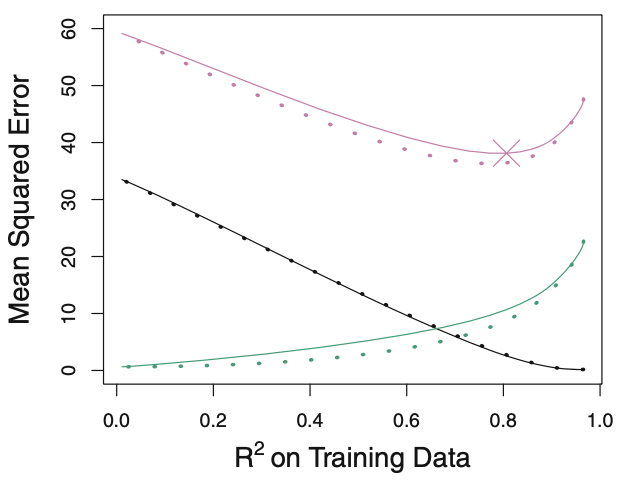
Description automatically generatedA close up of a logo

Description automatically generated

1. Blue area is the constraint region:
   1. Lasso has corners at each of the axes:
   2. Ridge is circular,
2. The ellipses that are centered around represent regions of constant RSS, all of the points on a given ellipse share a common value of RSS
3. Lasso and ridge regression coefficient estimates are given by the first point at which an ellipse contacts the constraint region.
   1. Ridge regression has a circular constraint with no sharp points, this intersection will generally nor occur on an axis🡪Ridge regression coefficient estimates will be exclusively non-zero
   2. Lasso constraint has corners at each of the axes, so the ellipse will often intersect the constraint region at an axis🡪One of the coefficient will equal to zero

Comparing the Lasso and Ridge Regression

R squared vs. MSE: Useful for comparing models with different types of regularization



\*Lasso implicitly assumes that a number of the coefficients truly equal zero.

Neither lasso or ridge regression will universally dominate the other:

1. Both lasso and ridge can yield a reduction in variance with a small increase in bias🡺 Can generate more accurate predictions
2. Lasso performs variable selection, and hence results in models that are easier to interpret.

A Simple Special Case for Ridge Regression and the Lasso

1. Ridge regression more or less shrinks every dimension of the data by the same proportion
2. Lasso more or less shrinks all coefficients toward zero by a similar amount, and sufficiently small coefficients are shrunken all the way to zero

Bayesian Interpretation for Ridge Regression and the Lasso

1. Lasso prior is steeply peeked peaked at zero🡪Lasso expects a priori that many of the coefficients are exactly zero
2. The ridge prior is flatter and flatter at zero🡪 ridge assumes the coefficients are randomly distributed about zero
3. Ridge regression and lasso regression follow directly from assuming the usual linear model with normal errors, together with a simple prior distribution for

6.2.3 Selecting the Tunning Parameter

Cross Validation:

1. Choose a grid of values, and compute the cross-validation error for each value of
2. Select the tunning parameter value for which the cross-validation error is smallest
3. The model is re-fit using all of the available observations and the selected value of the tunning parameter

6.3 Dimension Reduction Methods

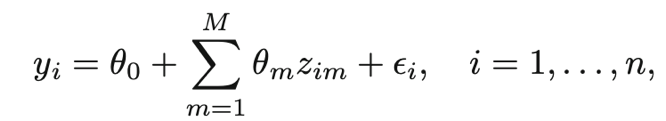
Transform the predictors and then fit a least squares model using the transformed variables

Let represent linear combinations of the original *p* predictors

A picture containing object, clock

Description automatically generated

For some constants ... We can fit the linear regression model using least squares:

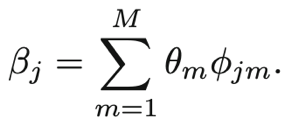


Dimension reduction:

1. This approach reduces the problem of estimating *(p+1)* coefficients to the simpler problem of estimating *M+1* coefficients where *M<p🡪* Dimension has been reduced to *M<p*

A picture containing object, clock

Description automatically generated



1. Dimension reduction serves to constrain the estimated coefficients🡪Has the potential to bias the coefficient estimates
2. In situations where *p* is large relative to *n*, selecting a value of can significantly reduce the variance of the fitted coefficient.
3. All dimension reduction work in two steps
   1. Step 1: All transformed predictors are obtained
   2. The model is fitted using these M predictors

6.3.1 Principal Components Regression

An Overview of Principal Component Analysis

1. First principal component direction of the data is that along which the observations vary the most
   1. Projecting onto the first principal component direction yields the largest variance
   2. First component vector defines the line that is as close as possible to the data

Example:

A close up of a map

Description automatically generated

Principal Component Scores:



First Principal Component:



1. Principle component is a single summary of the joint pop and ad budgets for each location.(Dimension Reduction)

Given that ,

this is the linear combination for which is maximized

We can generally construct up to *p* distinct principal components.

Second Principal Component is a linear combination of the variables that is uncorrelated with , and has the largest variance subject to this constraints

* Zero-correlation Constraint: The direction must be perpendicular, or orthogonal to the first principal component direction.
* By construction, the first component will contain the most information.

The Principal Components Regression Approach

Assumptions of PCR:

The principal components regression(PCR) approach involves constructing the first *M* principal components, ,…,, and then using these components as the predictors in a linear regression model that is fir using least squares.

1. Only a small components suffice to explain most of the variability in the data, as the relationship with the response

If assumption of PCR holds:

Fitting a least squares model using PCR will lead to better results than fitting a least squares model

1. Most information are contained in principal components🡪Avoids overfitting
2. PCR will tend to do well in cases when the first few principal components are sufficient to capture most of the variation in the predictors
3. PCR is not a feature selection method!
   1. PCR does not result in the development of a model that relies upon a small set of the original features
   2. Ridge regression is a continuous version of PCR
4. The number of principal components, *M*, is typically chosen by cross-validation.
5. We should standardizing each predictor prior to generating principal components
   1. Ensures all variables on the same scale
   2. Or high variance variables will play a larger role in the principal components obtained

6.3.2 Partial Least Squares

Drawbacks of PCR:

Unsupervised Learning:

1. The linear combinations identified by PCR are unsupervised, since the response is not used to determine the principal component directions🡪The response does not supervise the identification of the principal components
2. There is no guarantee that the directions that best explain the predictors will also be the best directions to use for predicting the response

Supervised Learning PLS:

1. PLS identifies new features in a supervised way—it makes use of the response Y in order to identify new features that not only approximate the old features well, but also that are related to the response.
2. After standardizing the predictors, PLS computes equal to the coefficient from the simple linear regression of onto .
   1. The coefficient is proportional to the correlation between and
   2. PLS places the highest weight on the variables that are most strongly related to the response
3. PLS direction does not fit the predictors as closely as does PCA, but it does a better job of explaining the response
   1. To identify the second PLS, we first adjust each of the variables for , by regressing each variable on and taking residuals🡪
      1. Orthogonalized data: Residuals can be interpreted as the remaining information that has not been explained by the first PLS direction
   2. Fit using the same approach as
   3. Number *M* of partial least squares directions used in PLS is a tunning parameter that is typically chosen by cross-validation
4. While the supervised dimension reduction of PLS can reduce bias, it also has the potential to increase variance.

6.4 Considerations in High Dimensions

6.4..1 High-Dimensional Data

Dimension: The size of the number of predictors, *p*.

Low-dimensional setting: The number of observations, *n*, is much greater than *p*, the number of features,.

High-dimensional setting: Data sets containing more features than observations,

1. Least squares linear regression are not appropriate
2. Bias-Variance trade-off and the danger of overfitting

6.4.2 What Goes Wrong in High Dimensions?

Unsuitable Techniques: Linear regression, logistics regression, Linear Discriminant analysis.

When :

Regardless of whether or not there is a relationship between the features and the response, least squares will yield a set of coefficient estimates that result in a perfect fit to the data.

1. Perfect fit will almost certainly lead to overfitting of the data(model is too flexible)🡪 Perform poorly on an independent dataset

6.4.3 Regression in High Dimensions

Stepwise selection, ridge regression, the lasso, and principal components regression are particularly useful for performing regression in the high-dimensional settings

1. Regularization or shrinkage plays a key role in high-dimensional problems
2. Appropriate tunning parameter selection is crucial for good predictive performance
3. The test error tends to increase as the dimensionality of the problem increases, unless additional features are truly associated with the response🡪Curse of dimensionality

6.4.4 Interpreting Results in High Dimensions

Multi-collinearity in high-dimensional settings:

1. Any variable in the model can be written as a linear combination of all other variables in the model.
2. We can never know which variables truly are predictable of the outcome
3. It is important to report errors on an independent test set, or cross-validation errors.

6.5 Lab1: Subset Selection Methods

6.5.1 Best Subset Selection

Is.na(): used to identify missing observations, returns a vector the same length as the input vector, with a TRUE for any elements that are missing, and a FALSE for non-missing elements.

Sum(is.na()): count all missing values

A screenshot of a cell phone

Description automatically generated

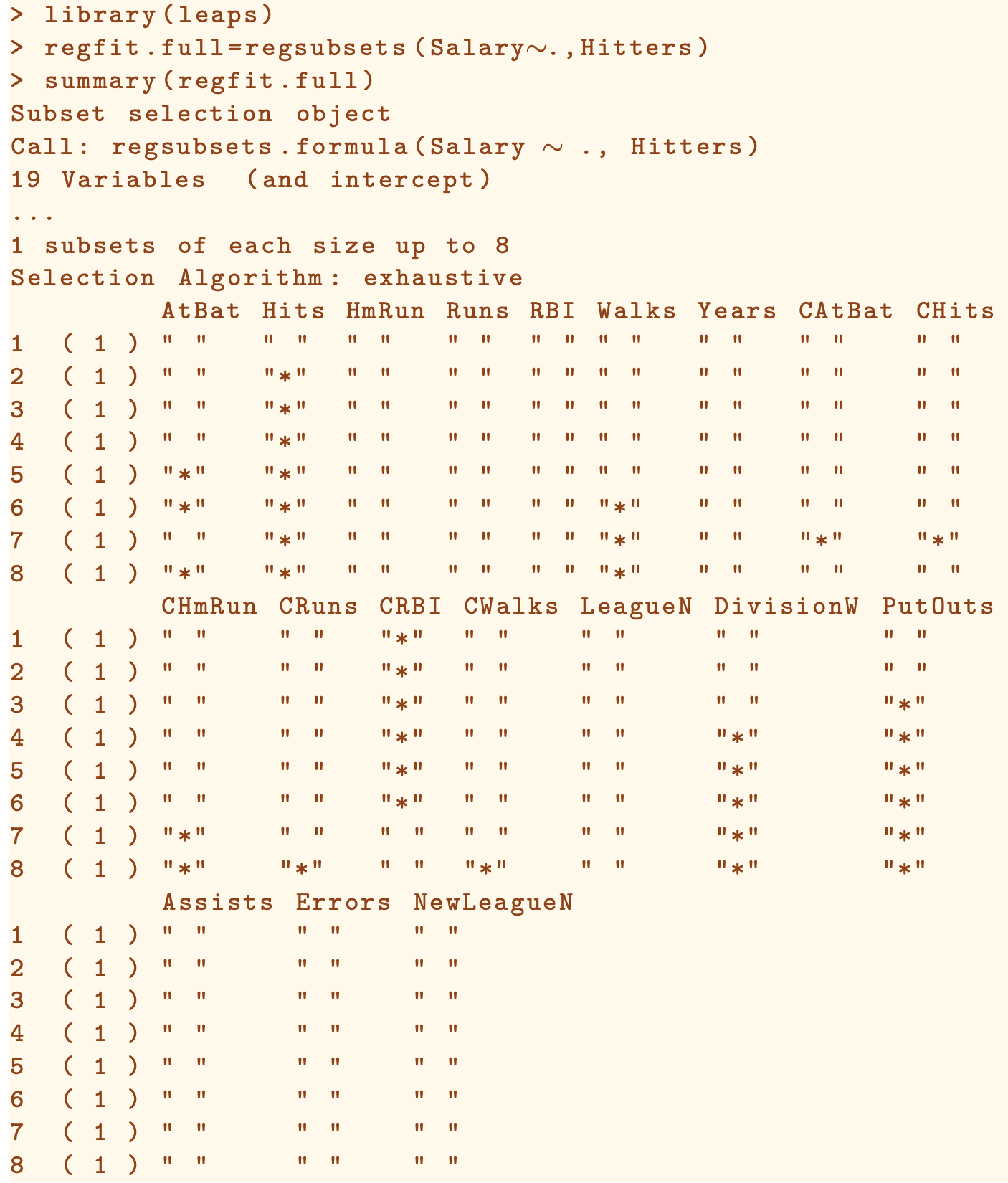
Na.omit(): removes all of the rows that have missing values in any variable

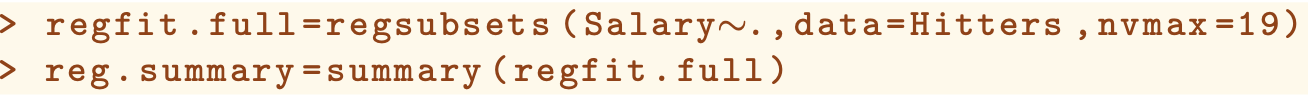
A screenshot of a cell phone

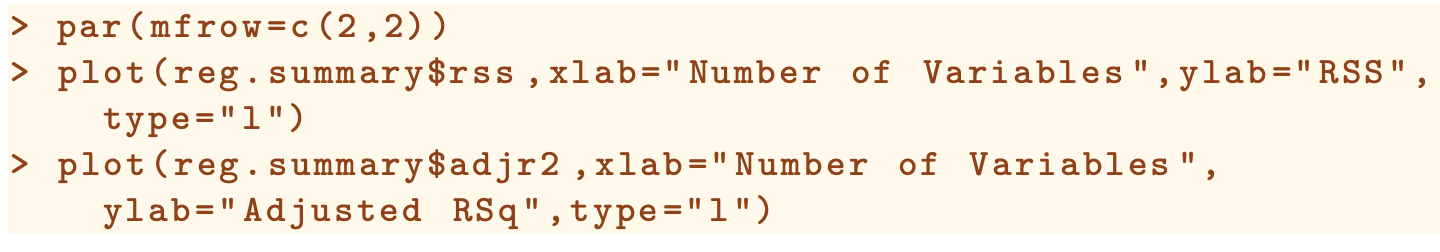
Description automatically generated

Regsubsets(): Part of the leaps library. performs best subset selection by identifying the best model that contains a given number of predictors, where best is quantifies using RSS(syntax the same as lm())

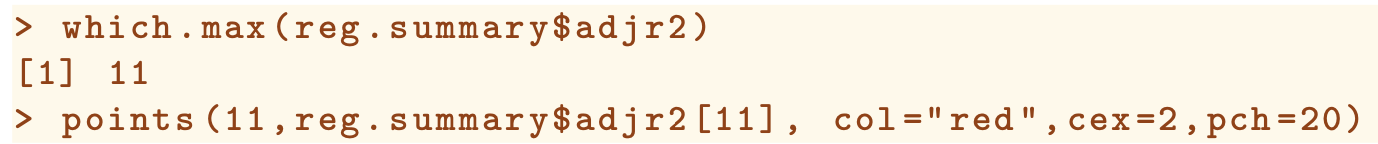
Summary() outputs the best set of variables



1. An asterisk indicates that given variable is included in the corresponding mode
2. Regsubsets() only reports up to the best eight-variable model
3. Nvmax option can be used in order to return as many variables as de
4. Summary can also return r squared RSS, adjusted R squared, Cp, and BIC
5. Plot() can be used to plot various metrics against the type of variables. Type=”1” tells R to connect the plotted points with lines



Points(): work likes plot(), expect that it puts points on a plot that has already been created, instead of creating a new plot



1. Regsubsets has a built in plot() which can be used to display the selected variables for the best model with
2. Regsubsetes has a built in plot() command which can be used to display the selected variables for the best model with a given number of predictors

A screenshot of a cell phone

Description automatically generated

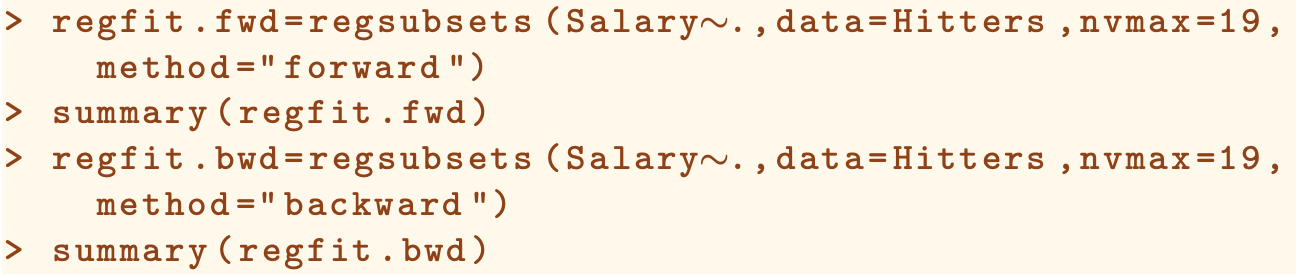
1. Top row of each plot contains a black square for each variable selected according to the optimal model associated with the statistic
   1. Use coef() to see the coefficient estimates of the model

A picture containing knife

Description automatically generated

6.5.2 Forward and Backward Stepwise Slection

Rgesubsets() function to perform forward and backward selection, using the argument method=”forward” or method=”backward”



6.5.3 Choosing Among Models Using the Validation Set Approach and Cross Validation

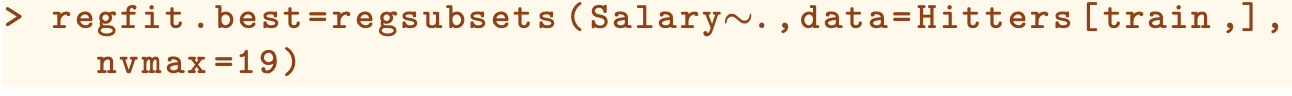
Validation Set Approach:

1. Only the training error is used to determine which model of a given size is best.

Step 1: Create train/test split



Step 2: Apply best subset selection to the training data



Step 3. Build an “X” matrix from the data



Step 4. Extract the coefficients from the fitted model given the variable size, multiply them into the appropriate column of the test model matrix to form the predictions, and test MSE

A picture containing table

Description automatically generated

A picture containing table, knife

Description automatically generated

A screenshot of a cell phone

Description automatically generated

\*Select the lowest test error model

Step 5. Use best subset selection on the full data set, and select the best 10-variable model

\*We must use the beset subset selection on the full data set and select the best ten-variable model

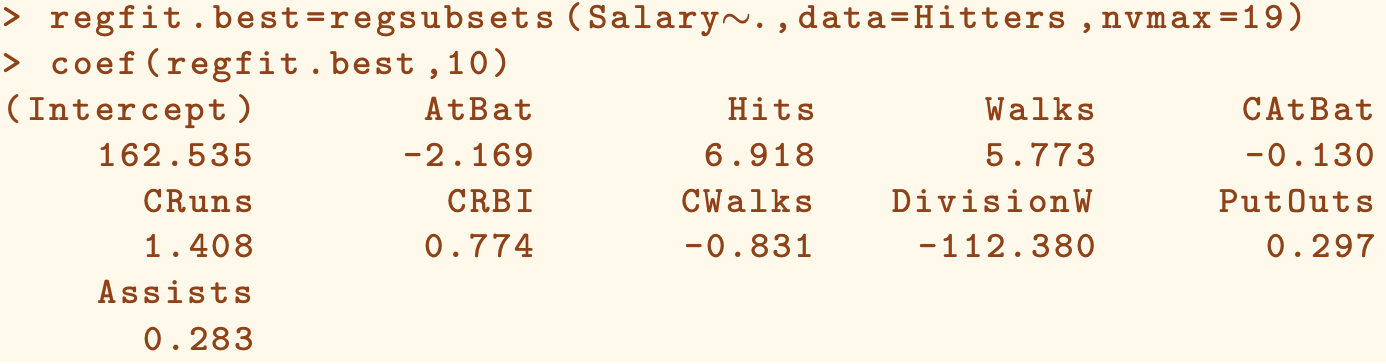
A screenshot of a cell phone

Description automatically generated

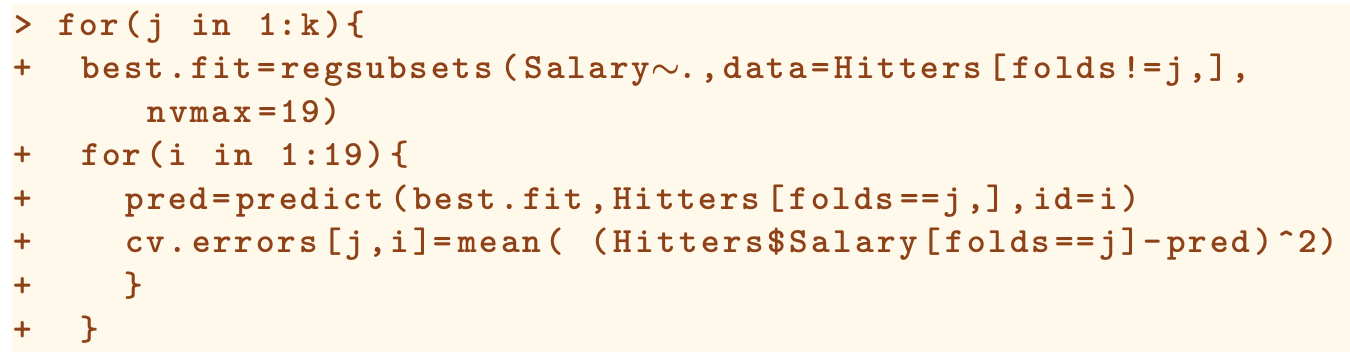
Cross Validation:

We must perform best subset selection within each of the k training sets

Step 1: Create a vector that allocates each observation to one of the k folds, and create a matrix in which we store the results



Step 2: Write a loop that performs cross-validation. In the jth fold, the elements of folds that equal j are in the test set, and the remainder are in the training set. Make predictions for each model size, and compute the test errors on the appropriate subset, and store them in the appropriate slot in the matrix cv.errors🡪Returns a matrix of which (I,j)th elemtn correspond to the test MSE for the ith cross-validation fold for the best j-variable model

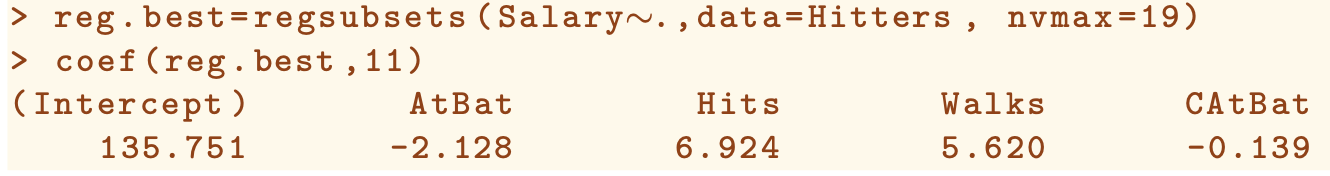


Use apply() to average over the columns of the matrix in order to obtain a vector for which the jth element is the cross validation error for the j-variable mode.

A screenshot of a cell phone

Description automatically generated

Perform best subset selection on the full data in order to obtain the 11-variable model



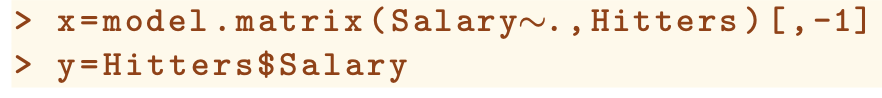
6.6 Lab2: Ridge Regression and the Lasso

Use glmnet package to perform ridge and lasso:

Main function in the package is glmnet(): can be used to fit ridge regression models and lasso models.

1. We must pass in an x matrix as well as a y vector
2. We do not use the y~x syntax

Model.matrix(): useful for creating the x matrix, producing a matrix corresponding to the and automatically transforms any qualitative variables into dummy variables



6.6.1 Ridge Regression

Glmnet():

1. Alpha=0: ridge regression(default0
2. Alpha=1: lasso model
3. Lambda: Needs to be selected(default automatically select a range of lambda values)
4. By default, glmnet() function standardizes the variables so that they are on the same scale(to turn off, use standardize=False)

A close up of a logo

Description automatically generated

1. Associated with each value of lamda is a vector of ridge regression coefficients, stored in a matrix that can e accessed by coef(), returns a matrix of n rows(one for

Each predictor plus and intercept) and m columns(one for each lamnda)

A close up of a logo

Description automatically generated

1. To access the fit of a certain lamda(access the columns)

A screenshot of a cell phone

Description automatically generated

1. Predict(): can be used to obtain the ridge regression for a given value of lambda,

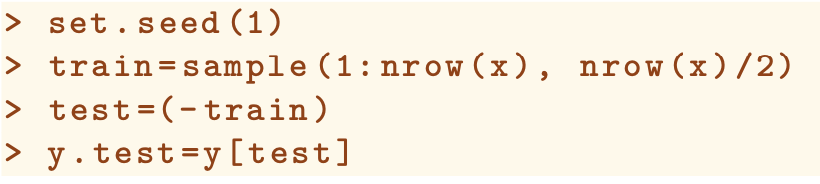
Type=”coefficients”

A screenshot of a cell phone

Description automatically generated

Train-test split algorithm:

1. Produce a random vector of True, False elements and select the observations corresponding to True for training data
2. Randomly choose a subset of numbers between 1 and n; these can then be used as the indices for the training observations



1. Fit a ridge regression model on the training set
2. Use predict() function to get the response on the test set, replacing “type=”coefficients”” with newx argument

A picture containing knife, table

Description automatically generated

\*Least squares model is simply a ridge regression with lambda=0

Cv.glmnet(): by default, the function performs ten-fold cross-validation, though this can be changed using the argument nfolds.

cv.out$lamda.min: returns the lambda that results in the lowest cross validation error

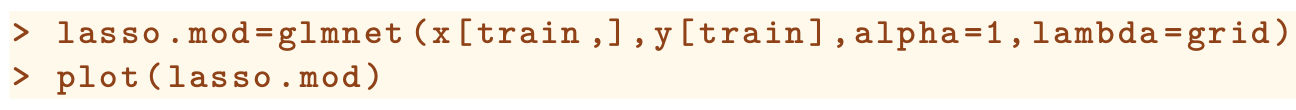
A picture containing bird, flower

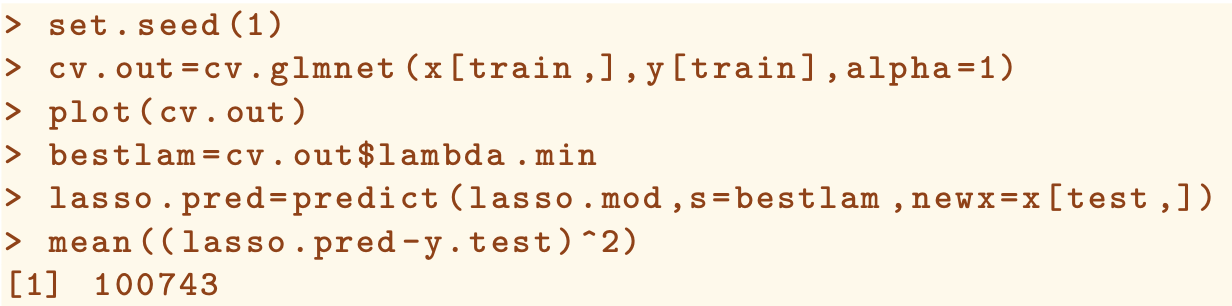
Description automatically generatedA screenshot of a cell phone

Description automatically generated

6.6.2 The Lasso

The same as ridge regression, simply change alpha to 1





A screenshot of a cell phone

Description automatically generated

6.7 Lab 3: PCR and PLS Regression

Principal components regression(PCR) can be performed using the pcr() function, which is part of the pls library

Pcr(): similar to the syntax of lm() with additional options.

1. Scale=TRUE: has the effect of standardizing each predictor
2. Validation=”CV” causes pcr() to compute the ten-fold cross-validation error for each possible value of M, the number of principle components used
3. Summary() returns the fitting results
4. %of variance: the amount of information about the predictors or the response that is captured using M principal components

A screenshot of a cell phone

Description automatically generated

Validationplot(): used to plot the cross-validation scores function.

1. Val.type=”MSEP” will cause the cross-validation MSE to be plotted

Training-testing the PCR model:

1. Step 1: train the model, identify the M for lowest cross-validation error

A picture containing knife, table

Description automatically generated

1. Step2: Compute the test MSE(the result is more difficult to interpret because it doe not perform any kind variable selection or even directly produce coefficient estimates)

A picture containing table

Description automatically generated

1. Fit the PCR on the full data set, using M=7, the number of components identified by cross-validationA screenshot of a cell phone

   Description automatically generated

6.7.2 Partial Least Squares

Plsr() function: implements the partial least squares(PLS). Syntax is the same as pcr()

A screenshot of a cell phone

Description automatically generated

Perform PLS on the full data set, using M=2, the number of components identified by crosss-validation

\*PCR only attempts to maximize the amount of variance explained in the predictors, while PLS searches for directions that explain variance in to in both the predictors and response.