Linear Model Selection and Regularization

Recall the standard linear model

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \epsilon$$

As we have seen before, one typically fits this model using least squares.

Here, we will discuss some ways in which the simple linear model can be improved, by replacing plain least squares fitting with some alternative fitting procedures.

Why might we want to use another fitting procedure instead of least squares?

- *Prediction Accuracy*: especially when $p > n$, to control the variance.
- *Model Interpretability*: By removing irrelevant features — that is, by setting the corresponding coefficient estimates to zero — we can obtain a model that is more easily interpreted.

Here, we will discuss three important classes of methods.

- *Subset Selection*. This approach involves identifying a subset of the p predictors that we believe to be related to the response. We then fit a model using least squares on the reduced set of variables.
- *Shrinkage*. This approach involves fitting a model involving all p predictors. However, the estimated coefficients are shrunken towards zero relative to the least squares estimates. This shrinkage (also known as 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. Hence, shrinkage methods can also perform variable selection.
- *Dimension Reduction*. This approach involves projecting the p predictors into an M-dimensional subspace, where $M < p$. This is achieved by

computing M different linear combinations, or projections, of the variables. Then these M projections are used as predictors to fit a linear regression model by least squares.

Subset Selection

Here we consider some methods for selecting subsets of predictors. These include best subset and stepwise model selection procedures.

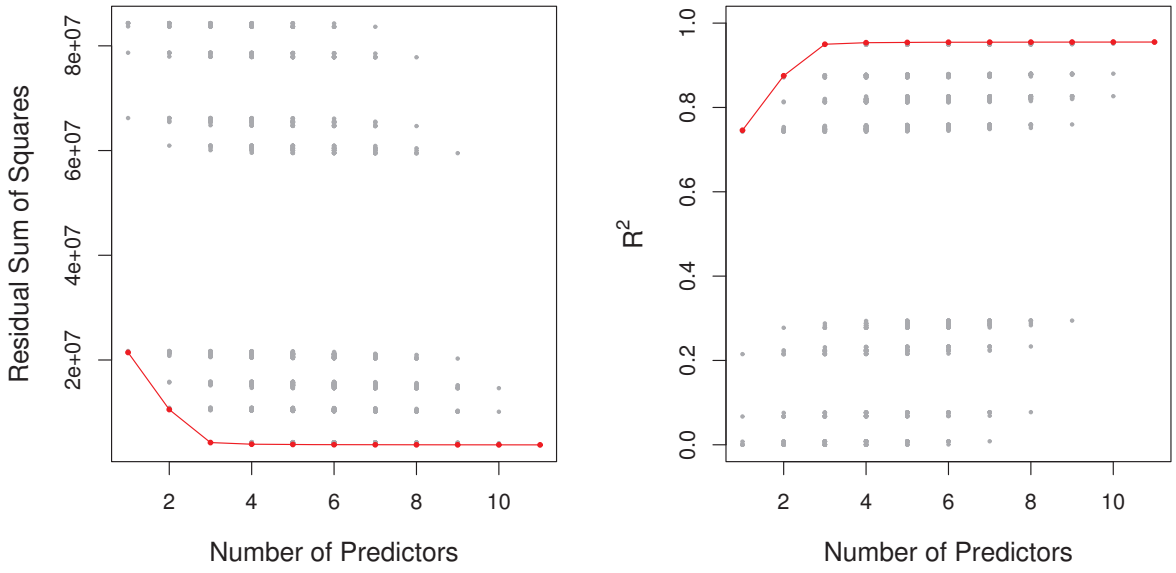
Best Subset Selection

Best Subset Selection Procedure:

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

In total, there are 2^p possible models:

$$\binom{p}{0} + \binom{p}{1} + \binom{p}{2} + \cdots + \binom{p}{p} = 2^p$$



For each possible model containing a subset of the ten predictors in the Credit data set, the RSS and R^2 are displayed. The red frontier tracks the best model for a given number of predictors, according to RSS and R^2. Though the data set contains only ten predictors, the x-axis ranges from 1 to 11, since one of the variables is categorical and takes on three values, leading to the creation of two dummy variables.

Notes:

- Although we have presented best subset selection here for least squares regression, the same ideas apply to other types of models, such as logistic regression.
- In the case of logistic regression, instead of ordering models by RSS, we instead use the deviance, a measure that plays the role of RSS for a broader class of models. The deviance is negative two times the maximized log-likelihood; the smaller the deviance, the better the fit.

Stepwise Selection

- For computational reasons, best subset selection cannot be applied with very large p.

In general, there are 2^p models that involve subsets of p predictors.

If $p=10$, $2^p \approx 1000$ possible models

If $p=20$, $2^p \approx 1,000,000$ possible models

- Best subset selection may also suffer from statistical problems when p is large. The larger the search space, the higher the chance of finding models that look good on the training data, even though they might not have any predictive power on future data.
- Thus an enormous search space can lead to overfitting and high variance of the coefficient estimates.
- For both of these reasons, stepwise methods, which explore a far more restricted set of models, are attractive alternatives to best subset selection.

Forward Stepwise Selection

Forward stepwise selection 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. In particular, at each step the variable that gives the greatest additional improvement to the fit is added to the model.

Forward Stepwise Selection Procedure:

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

Unlike best subset selection, which involved fitting 2^p models, forward stepwise selection involves fitting one null model, along with $p-k$ models in the kth iteration, for $k = 0, ..., p-1$.

This amounts to a total of

$$1 + \sum_{k=0}^{p-1} (p-k) = 1 + \frac{p(p+1)}{2}$$

models.

when $p = 20$, $2^p \approx 1{,}000{,}000$ but $1 + \frac{p(p+1)}{2} = 211$

Note: Although forward stepwise tends to do well in practice, it is not guaranteed to find the best possible model out of all 2^p models containing subsets of the p predictors.

For instance, suppose that in a given data set with $p = 3$ predictors, the best possible one-variable model contains X_1, and the best possible two-variable model instead contains X_2 and X_3. Then forward stepwise selection will fail to select the best possible two-variable model, because $\mathcal{M}_1$ will contain X_1, so $\mathcal{M}_2$ must also contain X_1 together with one additional variable.

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

The first four selected models for best subset selection and forward stepwise selection on the Credit data set. The first three models are identical but the fourth models differ.

Backward Stepwise Selection

Like forward stepwise selection, backward stepwise selection provides an efficient alternative to best subset selection. However, unlike forward stepwise selection, it begins with the full least squares model containing all p predictors, and then iteratively removes the least useful predictor, one-at-a-time.

Backward Stepwise Selection Procedure:

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

Notes:

- Like forward stepwise selection, the backward selection approach searches through only $1 + p(p + 1)/2$ models, and so can be applied in settings where p is too large to apply best subset selection.
- Also like forward stepwise selection, backward stepwise selection is not guaranteed to yield the best model containing a subset of the p predictors.
- 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). In contrast, forward stepwise can be used even when $n < p$, and so is the only viable subset method when p is very large.

Choosing the Optimal Model

The model containing all of the predictors will always have the smallest RSS and the largest R^2, since these quantities are related to the training error. Instead, we wish to choose a model with a low test error.

The training error can be a poor estimate of the test error. Therefore, RSS and R^2 are not suitable for selecting the best model among a collection of models with different numbers of predictors.

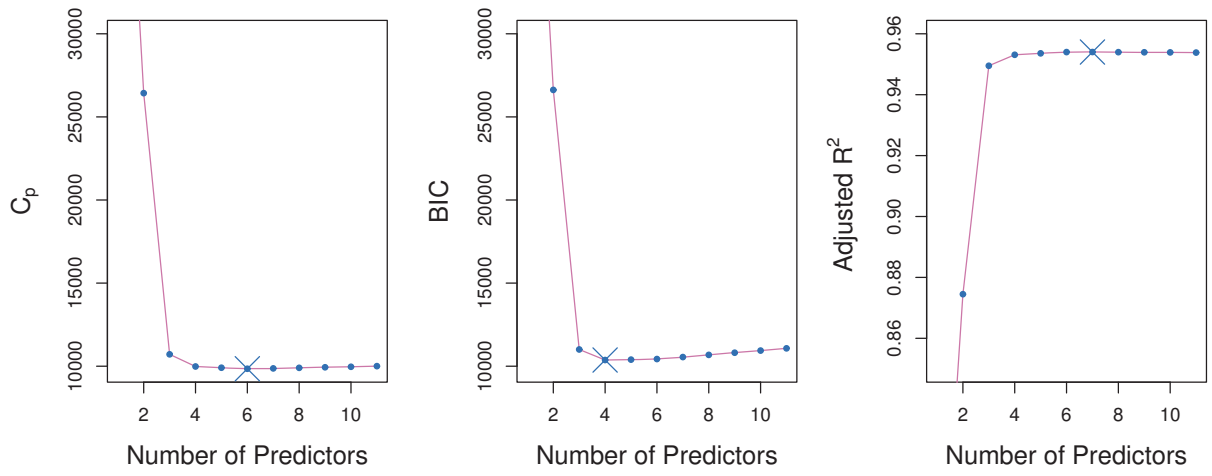
In order to select the best model with respect to test error, we need to estimate this test error. There are two common approaches:

1. We can indirectly estimate test error by making an adjustment to the training error to account for the bias due to overfitting.
2. We can directly estimate the test error, using either a validation set approach or a cross-validation approach.

C_p, AIC, BIC, and Adjusted R^2

There are a number of techniques for adjusting the training error for the model size which can be used to select among a set of models with different numbers of variables.

We consider four such approaches: C_p, Akaike information criterion (AIC), Bayesian information criterion (BIC), and adjusted R^2.



C_p, BIC, and adjusted R^2 are shown for the best models of each size for the Credit data set. C_p and BIC are estimates of test MSE. In the middle plot we see that the BIC estimate of test error shows an increase after four variables are selected. The other two plots are rather flat after four variables are included.

Mallow's C_p:

For a fitted least squares model containing d predictors, the C_p estimate of test MSE is computed using the equation

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

$MSE = \frac{RSS}{n}$

where $\hat{\sigma}^2$ is an estimate of the variance of the error ϵ associated with each response measurement.

- The C_p statistic adds a penalty of $2d\hat{\sigma}^2$ to the training RSS in order to adjust for the fact that the training error tends to underestimate the test error.
- Clearly, the penalty increases as the number of predictors in the model increases; this is intended to adjust for the corresponding decrease in training RSS.
- The C_p statistic tends to take on a small value for models with a low test error, so when determining which of a set of models is best, we choose the model with the lowest C_p value.

AIC:

The AIC criterion is defined for a large class of models fit by maximum likelihood:

$$\text{AIC} = -2\log L + 2d$$

where L is the maximized value of the likelihood function for the estimated model.

In the case of the linear model with Gaussian errors, maximum likelihood and least squares are the same thing, and C_p and AIC are equivalent.

BIC:

For the least squares model with d predictors, the BIC is, up to irrelevant constants, given by

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

- Like C_p, the BIC will tend to take on a small value for a model with a low test error, and so generally we select the model that has the lowest BIC value.

- Notice that BIC replaces the $2d\hat{\sigma}^2$ used by C_p with a $\log(n)d\hat{\sigma}^2$ term, where n is the number of observations.
- Since $\log n > 2$ for any $n > 7$, the BIC statistic generally places a heavier penalty on models with many variables, and hence results in the selection of smaller models than C_p.

Adjusted R^2:

For a least squares model with d variables, the adjusted R^2 statistic is calculated as

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

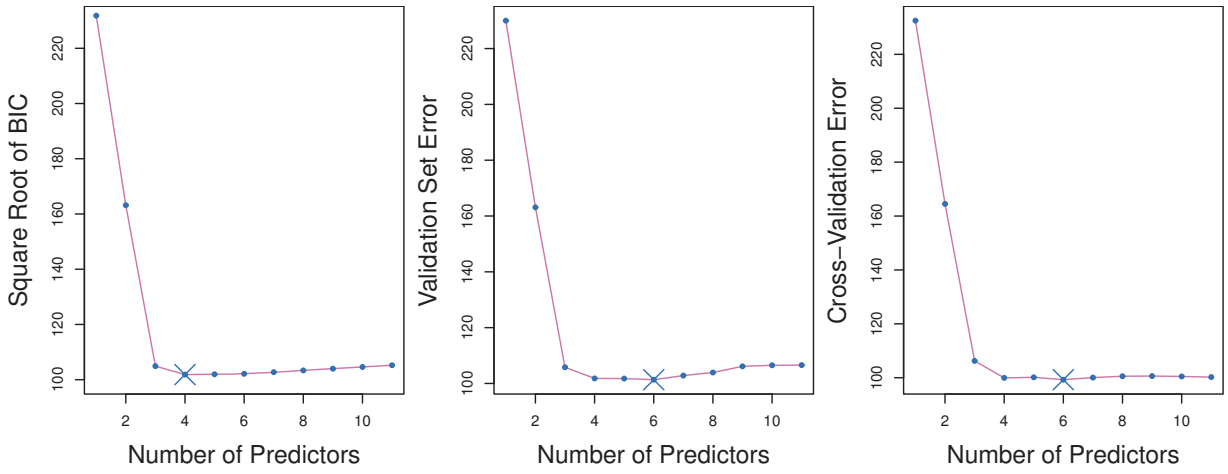
Recall that

$R^2 = 1 - \frac{RSS}{TSS}$ where $TSS = \sum_{i=1}^{n} (y_i - \bar{y})^2$

- Unlike C_p, AIC, and BIC, for which a small value indicates a model with a low test error, a large value of adjusted R^2 indicates a model with a small test error.
- Maximizing the adjusted R^2 is equivalent to minimizing $\frac{\text{RSS}}{n-d-1}$. While RSS always decreases as the number of variables in the model increases, $\frac{\text{RSS}}{n-d-1}$ may increase or decrease, due to the presence of d in the denominator.
- Unlike the R^2 statistic, the adjusted R^2 statistic pays a price for the inclusion of unnecessary variables in the model.

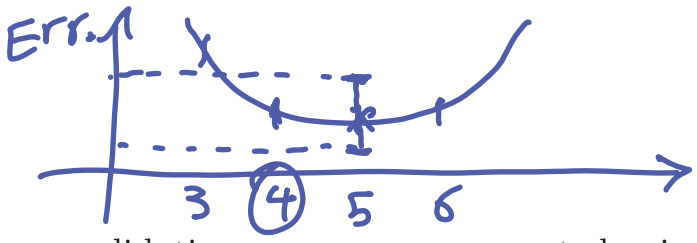
Validation and Cross-Validation

- We can compute the validation set error or the cross-validation error for each model under consideration, and then select the model for which the resulting estimated test error is smallest.
- This procedure has an advantage relative to AIC, BIC, C_p, and adjusted R^2, in that it provides a direct estimate of the test error, and makes fewer assumptions about the true underlying model.



For the Credit data set, three quantities are displayed for the best model containing d predictors, for d ranging from 1 to 11. The overall best model, based on each of these quantities, is shown as a blue cross. Left: Square root of BIC. Center: Validation set errors. Right: Cross-validation errors.

- The figure displays, as a function of d, the BIC, validation set errors, and cross-validation errors on the Credit data, for the best d-variable model.
- The validation errors were calculated by randomly selecting three-quarters of the observations as the training set, and the remainder as the validation set.



- The cross-validation errors were computed using $k = 10$ folds. In this case, the validation and cross-validation methods both result in a six-variable model.
- However, all three approaches suggest that the four-, five-, and six-variable models are roughly equivalent in terms of their test errors.
- In this setting, we can select a model using the one-standard-error rule. 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.

Performing Subset Selection Methods in R

Here we apply the best subset selection approach to the Credit data in R. We wish to predict an individual's balance (average credit card debt for each individual) on the basis of various predictors.

```
> library(ISLR2)
> names(Credit)
 [1] "Income"    "Limit"     "Rating"    "Cards"     "Age"
 [6] "Education" "Own"       "Student"   "Married"   "Region"
[11] "Balance"
> dim(Credit)
[1] 400  11
```

The regsubsets() function (part of the leaps library) performs best subset selection by identifying the best model that contains a given number of predictors, where best is quantified using RSS. The syntax is the same as for lm(). The summary() command outputs the best set of variables for each model size.

```
> library(leaps)
> reg_fit_Credit_full <- regsubsets(Balance ~ ., Credit)
> summary(reg_fit_Credit_full)
```

```
Subset selection object
Call: regsubsets.formula(Balance ~ ., Credit)
11 Variables  (and intercept)
            Forced in Forced out
Income          FALSE      FALSE
Limit           FALSE      FALSE
Rating          FALSE      FALSE
Cards           FALSE      FALSE
Age             FALSE      FALSE
Education       FALSE      FALSE
OwnYes          FALSE      FALSE
StudentYes      FALSE      FALSE
MarriedYes      FALSE      FALSE
RegionSouth     FALSE      FALSE
RegionWest      FALSE      FALSE
1 subsets of each size up to 8
Selection Algorithm: exhaustive
         Income Limit Rating Cards Age Education OwnYes StudentYes
1  ( 1 ) " "    " "   "*"    " "   " " " "       " "    " "
2  ( 1 ) "*"    " "   "*"    " "   " " " "       " "    " "
3  ( 1 ) "*"    " "   "*"    " "   " " " "       " "    "*"
4  ( 1 ) "*"    "*"   " "    "*"   " " " "       " "    "*"
5  ( 1 ) "*"    "*"   "*"    "*"   " " " "       " "    "*"
6  ( 1 ) "*"    "*"   "*"    "*"   "*" " "       " "    "*"
7  ( 1 ) "*"    "*"   "*"    "*"   "*" " "       "*"    "*"
8  ( 1 ) "*"    "*"   "*"    "*"   "*" " "       "*"    "*"
         MarriedYes RegionSouth RegionWest
1  ( 1 ) " "        " "          " "
2  ( 1 ) " "        " "          " "
3  ( 1 ) " "        " "          " "
4  ( 1 ) " "        " "          " "
5  ( 1 ) " "        " "          " "
6  ( 1 ) " "        " "          " "
7  ( 1 ) " "        " "          " "
8  ( 1 ) " "        " "          "*"
```

An asterisk indicates that a given variable is included in the corresponding model. For instance, this output indicates that the best two-variable model contains only Income and Rating. By default, regsubsets() only reports results up to the best eight-variable model. But the nvmax option can be used

in order to return as many variables as are desired. Here we fit up to an 11-variable model.

```
> reg_fit_Credit_full <- regsubsets(Balance ~ ., Credit, nvmax = 11)
> reg_summary <- summary(reg_fit_Credit_full)
> reg_summary
Subset selection object
Call: regsubsets.formula(Balance ~ ., Credit, nvmax = 11)
11 Variables  (and intercept)
            Forced in Forced out
Income          FALSE      FALSE
Limit           FALSE      FALSE
Rating          FALSE      FALSE
Cards           FALSE      FALSE
Age             FALSE      FALSE
Education       FALSE      FALSE
OwnYes          FALSE      FALSE
StudentYes      FALSE      FALSE
MarriedYes      FALSE      FALSE
RegionSouth     FALSE      FALSE
RegionWest      FALSE      FALSE
1 subsets of each size up to 11
Selection Algorithm: exhaustive
          Income Limit Rating Cards Age Education OwnYes StudentYes
1  ( 1 )  " "    " "   "*"    " "   " " " "       " "    " "
2  ( 1 )  "*"    " "   "*"    " "   " " " "       " "    " "
3  ( 1 )  "*"    " "   "*"    " "   " " " "       " "    "*"
4  ( 1 )  "*"    "*"   " "    "*"   " " " "       " "    "*"
5  ( 1 )  "*"    "*"   "*"    "*"   " " " "       " "    "*"
6  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       " "    "*"
7  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       "*"    "*"
8  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       "*"    "*"
9  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       "*"    "*"
10  ( 1 ) "*"    "*"   "*"    "*"   "*" " "       "*"    "*"
11  ( 1 ) "*"    "*"   "*"    "*"   "*" "*"       "*"    "*"
          MarriedYes RegionSouth RegionWest
1  ( 1 )  " "        " "         " "
2  ( 1 )  " "        " "         " "
3  ( 1 )  " "        " "         " "
4  ( 1 )  " "        " "         " "
5  ( 1 )  " "        " "         " "
```

```
6  ( 1 )  " "         " "          " "
7  ( 1 )  " "         " "          " "
8  ( 1 )  " "         " "          "*"
9  ( 1 )  "*"         " "          "*"
10  ( 1 ) "*"         "*"          "*"
11  ( 1 ) "*"         "*"          "*"
```

The summary() function also returns R^2, RSS, adjusted R^2, C_p, and BIC. We can examine these to try to select the best overall model.

```
> names(reg_summary)
[1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat"
[8] "obj"
> reg_summary$rsq
 [1] 0.7458484 0.8751179 0.9498788 0.9535800 0.9541606 0.9546879
 [7] 0.9548167 0.9548880 0.9549636 0.9550468 0.9551016
```

Plotting RSS, adjusted R^2, C_p, and BIC for all of the models at once will help us decide which model to select. Note the type = "l" option tells R to connect the plotted points with lines.

```
> par(mfrow = c(2, 2))
> plot(reg_summary$rss, xlab = "Number of Variables", ylab = "RSS",
   type = "l")
> plot(reg_summary$adjr2, xlab = "Number of Variables", ylab = "Adjusted RSq",
   type = "l")
> which.max(reg_summary$adjr2)
[1] 7
> points(which.max(reg_summary$adjr2),
   reg_summary$adjr2[which.max(reg_summary$adjr2)],
   col = "red", cex = 2, pch = 20)
> plot(reg_summary$cp, xlab = "Number of Variables", ylab = "Cp", type = "l")
> which.min(reg_summary$cp)
[1] 6
> points(which.min(reg_summary$cp), reg_summary$cp[which.min(reg_summary$cp)],
   col = "red", cex = 2, pch = 20)
> plot(reg_summary$bic, xlab = "Number of Variables", ylab = "BIC",
```