# DS-6030 Homework Module 5

Tom Lever

06/24/2023

**DS 6030 | Spring 2023 | University of Virginia**

8. In this exercise, we will generate simulated data, and will then use this data to perform best subset selection.

   (a) Use the `rnorm()` function to generate a predictor $X$ of length $n = 100$, as well as a noise vector $\epsilon$ of length $n = 100$.

```
set.seed(2)
X <- rnorm(n = 100, mean = 0, sd = 1)
epsilon <- rnorm(n = 100, mean = 0, sd = 1*10^{-6})
X[1:3]
```

```
# [1] -0.8969145  0.1848492  1.5878453
```

```
epsilon[1:3]
```

```
# [1]  1.074459e-06  2.605978e-07 -3.142720e-07
```

   (b) Generate a response vector $Y$ of length $n = 100$ according to the model $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \epsilon$, where $\beta_0$, $\beta_1$, $\beta_2$, $\beta_3$ are constants of your choice.

```
beta_0 <- 1
beta_1 <- 2
beta_2 <- 3
beta_3 <- 4
Y <- beta_0 + beta_1 * X + beta_2 * I(X^2) + beta_3 * I(X^3) + epsilon
Y[1:3]
```

```
# [1] -1.266573  1.497471 27.752887
```

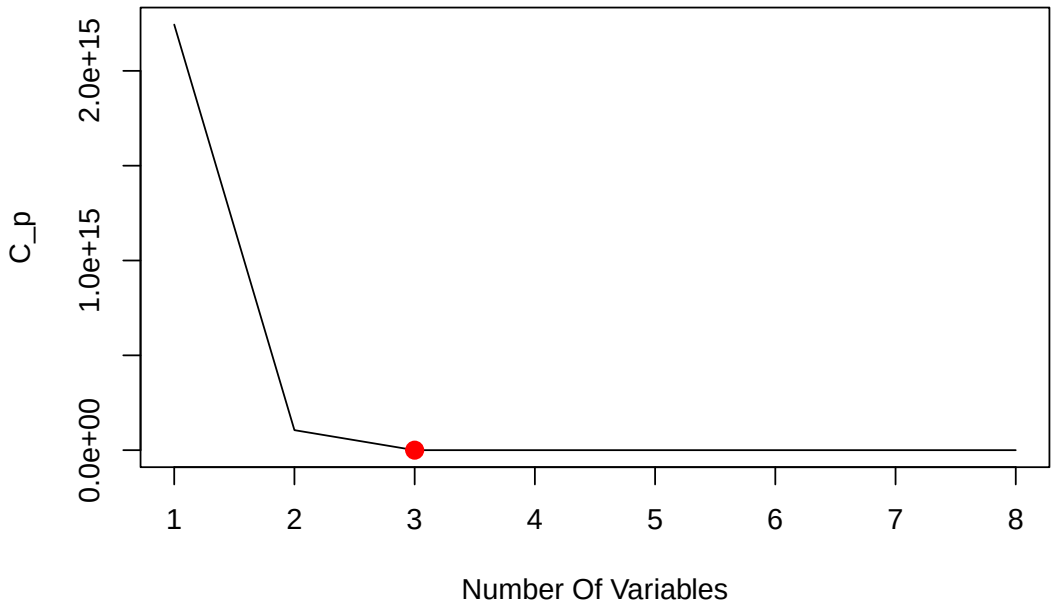
   (c) Use the `regsubsets()` function to perform best subset selection in order to choose the best model containing the predictors $X, X^2 \ldots, X^{10}$. What is the best model obtained according to $Cp$, BIC, and adjusted $R^2$? Show some plots to provide evidence for your answer, and report the coefficients of the best model obtained. Note you will need to use the `data.frame()` function to create a single data set containing both $X$ and $Y$.

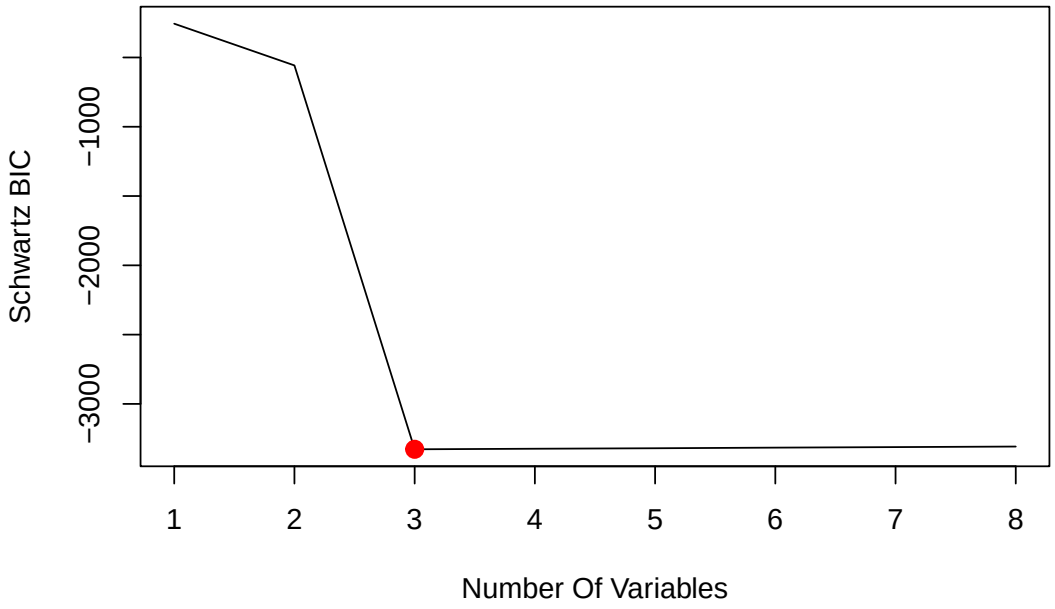
```
data_frame <- data.frame(X = X, Y = Y)
head(data_frame, n = 3)
```

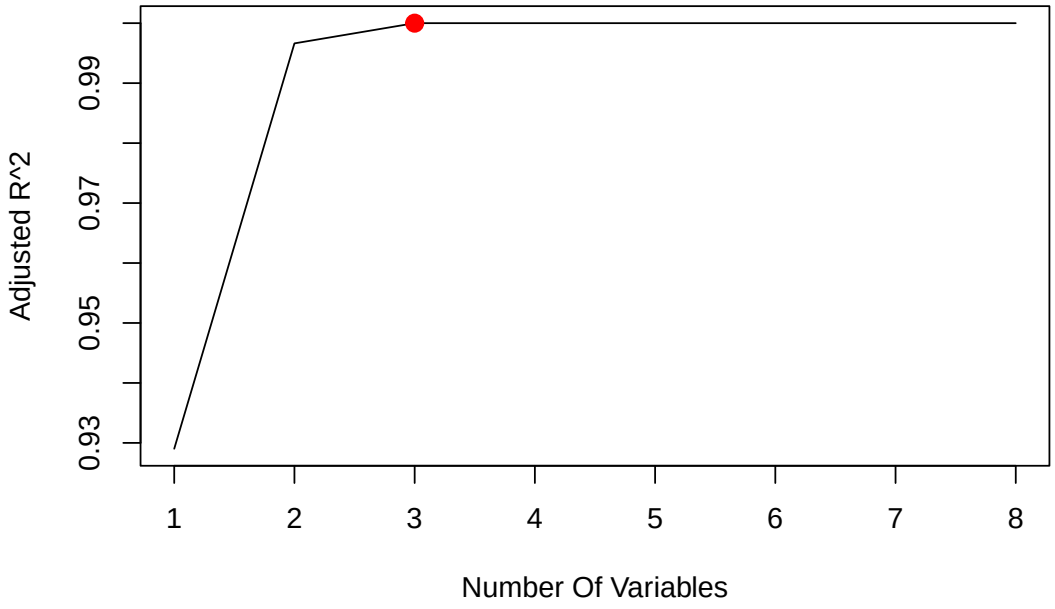
```
#           X           Y
# 1 -0.8969145 -1.26657....
# 2  0.1848492 1.497470....
# 3  1.5878453 27.75288....
```

```r
library(leaps)
formula <- Y ~ X + I(X^2) + I(X^3) + I(X^4) + I(X^5) + I(X^6) + I(X^7) + I(X^8) + I(X^9) + I(X
subset_selection_object <- regsubsets(
    x = formula,
    data = data_frame,
    method = "exhaustive"
)
analyze_subset_selection_object <- function(subset_selection_object) {
 summary_for_subset_selection_object <- summary(object = subset_selection_object)
 Mallows_Cp <- summary_for_subset_selection_object$cp
 index_of_model_with_minimum_Mallows_Cp <- which.min(Mallows_Cp)
 coefficients_by_Mallows_Cp <- coef(
     subset_selection_object, index_of_model_with_minimum_Mallows_Cp
 )
 Schwartz_BIC <- summary_for_subset_selection_object$bic
 index_of_model_with_minimum_Schwartz_BIC <- which.min(Schwartz_BIC)
 coefficients_by_Schwartz_BIC <- coef(
     subset_selection_object, index_of_model_with_minimum_Schwartz_BIC
 )
 adjusted_R2 <- summary_for_subset_selection_object$adjr2
 index_of_model_with_maximum_adjusted_R2 <- which.max(adjusted_R2)
 coefficients_by_adjusted_R2 <- coef(
     subset_selection_object, index_of_model_with_maximum_adjusted_R2
 )
 plot(Mallows_Cp, xlab = "Number Of Variables", ylab = "C_p", type = "l")
 index_of_minimum_Mallows_Cp <- which.min(Mallows_Cp)
 minimum_Mallows_Cp <- Mallows_Cp[index_of_minimum_Mallows_Cp]
 points(
     index_of_minimum_Mallows_Cp,
     minimum_Mallows_Cp,
     col = "red",
     cex = 2,
     pch = 20
 )
 plot(Schwartz_BIC, xlab = "Number Of Variables", ylab = "Schwartz BIC", type = "l")
 index_of_minimum_Schwartz_BIC <- which.min(Schwartz_BIC)
 minimum_Schwartz_BIC <- Schwartz_BIC[index_of_minimum_Schwartz_BIC]
 points(
     index_of_minimum_Schwartz_BIC,
     minimum_Schwartz_BIC,
     col = "red",
     cex = 2,
     pch = 20
 )
 plot(adjusted_R2, xlab = "Number Of Variables", ylab = "Adjusted R^2", type = "l")
 index_of_maximum_adjusted_R2 <- which.max(adjusted_R2)
 maximum_adjusted_R2 <- adjusted_R2[index_of_maximum_adjusted_R2]
 points(
     index_of_maximum_adjusted_R2,
     maximum_adjusted_R2,
     col = "red",
     cex = 2,
     pch = 20
```

```
 )
 coefficients <- list(
     coefficients_by_Mallows_Cp = coefficients_by_Mallows_Cp,
     coefficients_by_Schwartz_BIC = coefficients_by_Schwartz_BIC,
     coefficients_by_adjusted_R2 = coefficients_by_adjusted_R2
 )
 return(coefficients)
}
analyze_subset_selection_object(subset_selection_object)
```






```
# $coefficients_by_Mallows_Cp
# (Intercept)           X      I(X^2)      I(X^3)
#           1           2           3           4
#
```

```
# $coefficients_by_Schwartz_BIC
# (Intercept)           X      I(X^2)      I(X^3)
#           1           2           3           4
#
# $coefficients_by_adjusted_R2
# (Intercept)           X      I(X^2)      I(X^3)
#           1           2           3           4
```

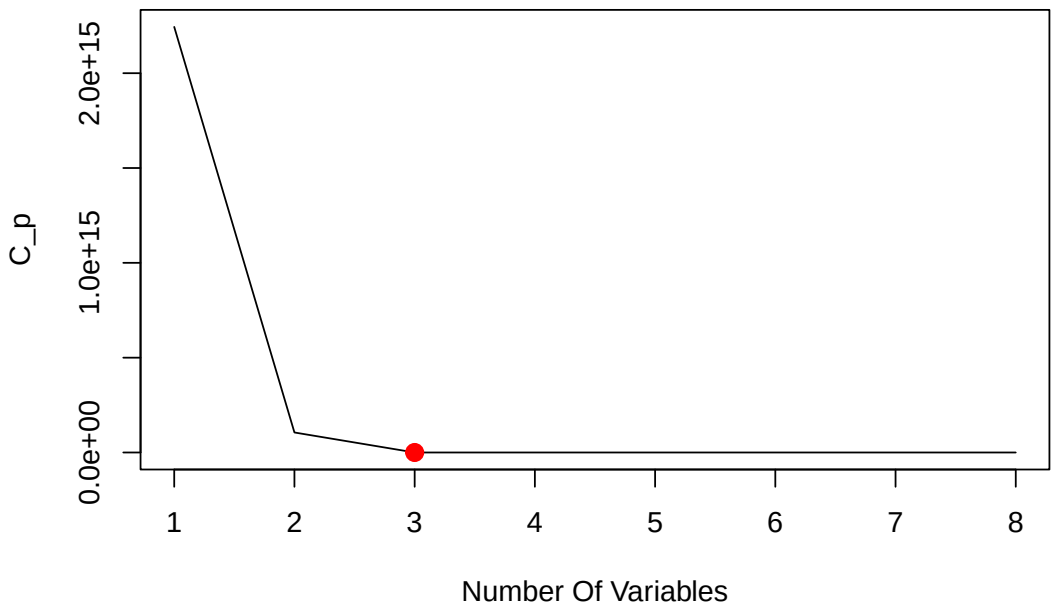
The best model obtained according to Mallow's $C_p$, the Schwartz Bayesian Information Criterion, and adjusted $R^2$ is

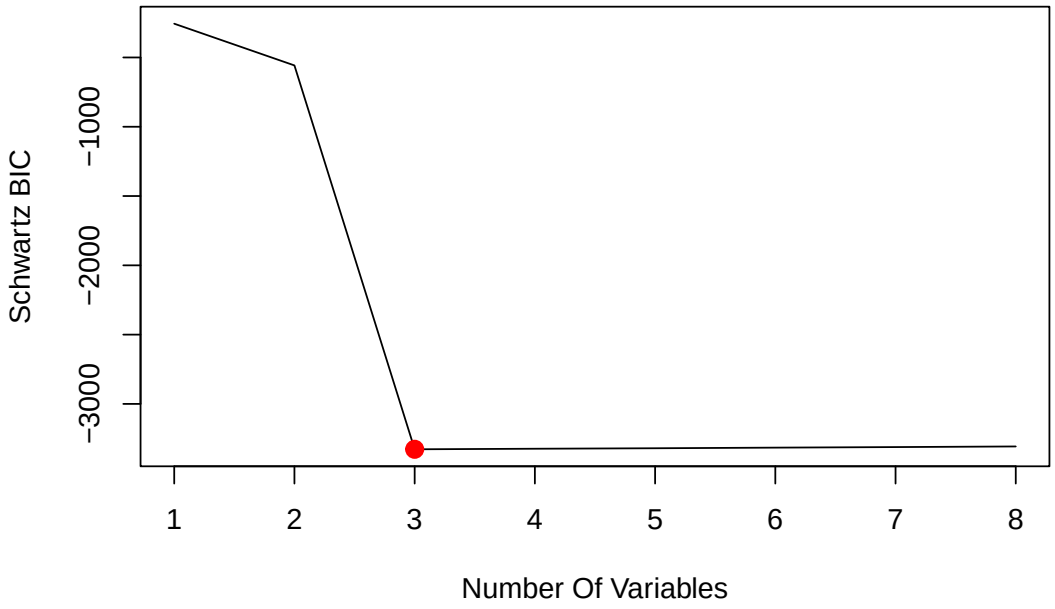
$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 = 1 + 2X + 3X^2 + 4X^3$$

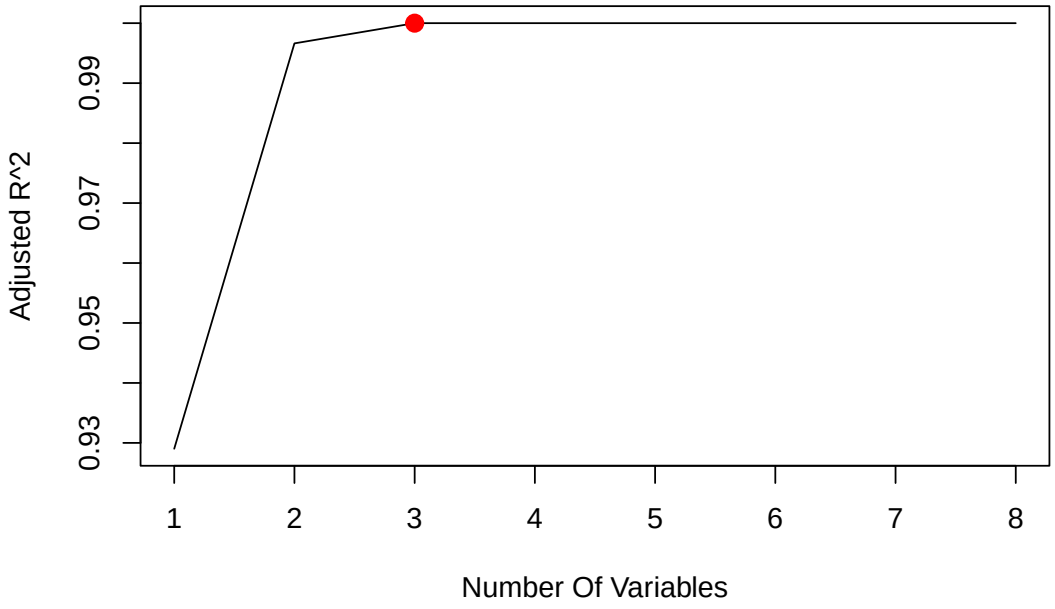
(d) Repeat (c), using forward stepwise selection and also using backwards stepwise selection. How does your answer compare to the results in (c)?

The models chosen by forward and backward selection involve the true predictive terms and their coefficients.

```
subset_selection_object <- regsubsets(
    x = formula,
    data = data_frame,
    method = "forward"
)
analyze_subset_selection_object(subset_selection_object)
```



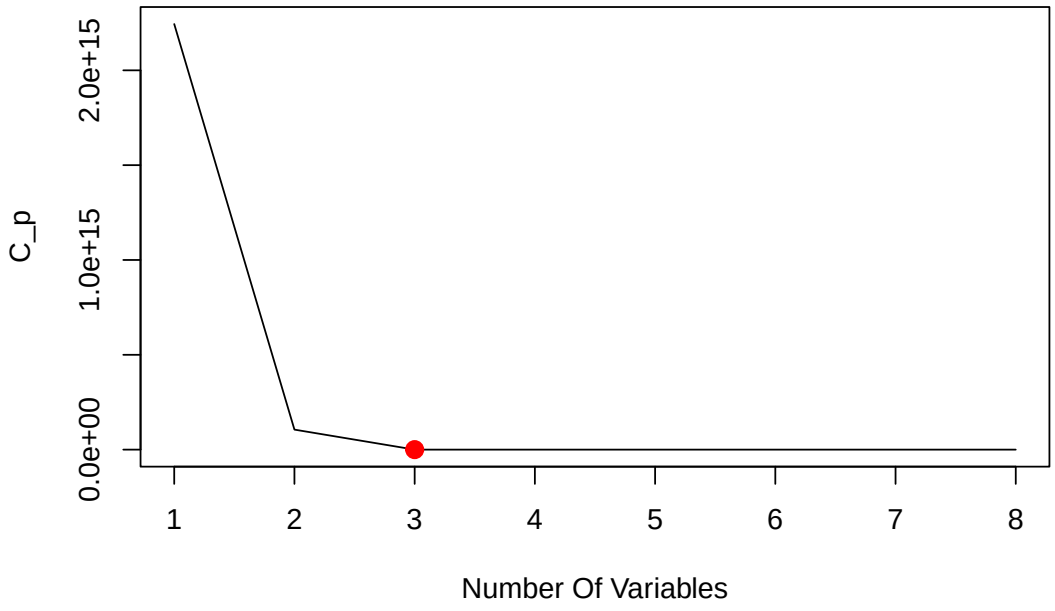


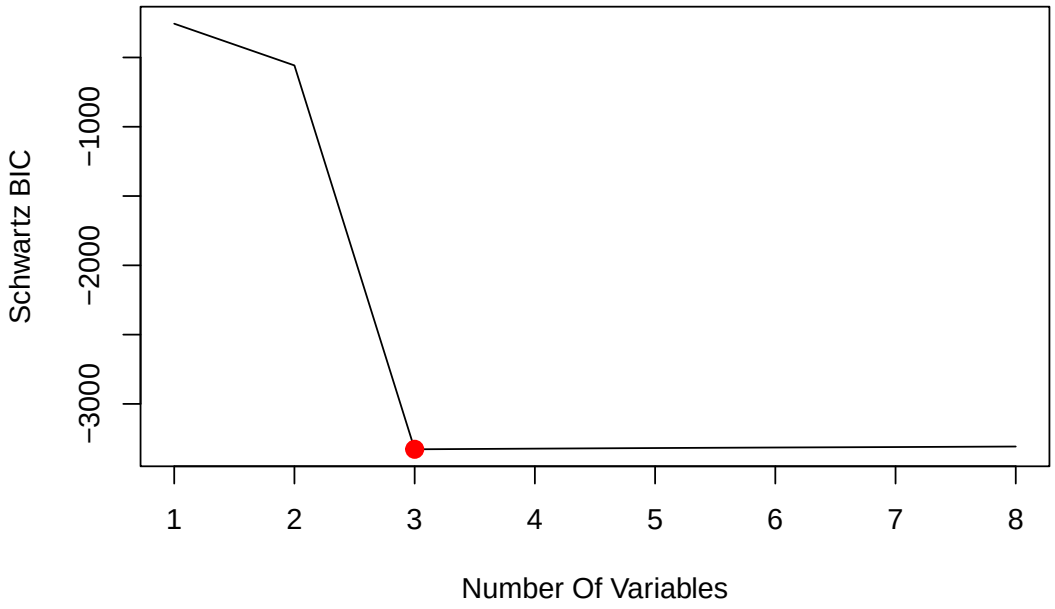



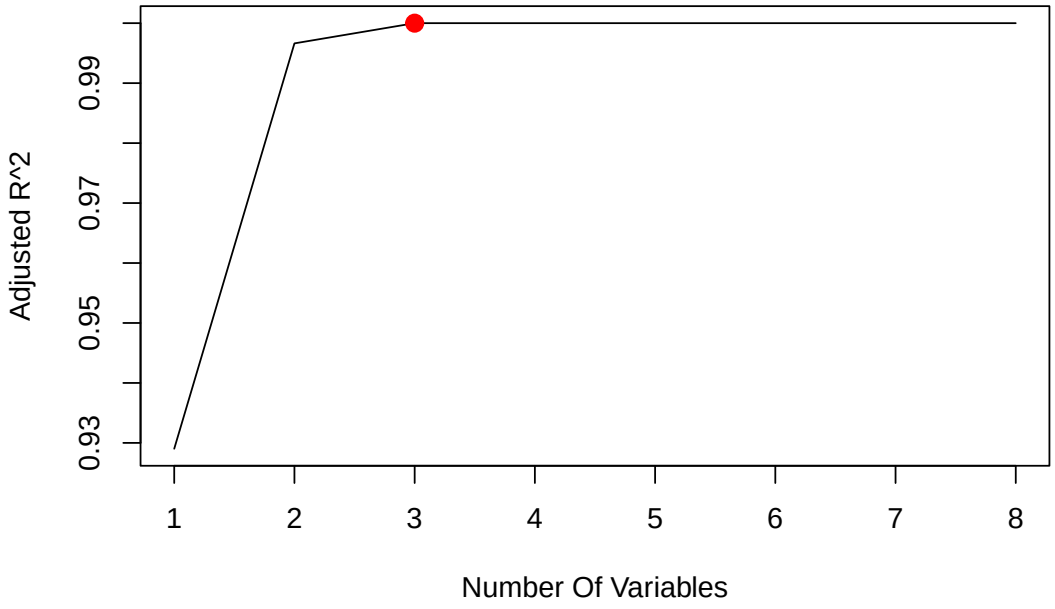

```
# $coefficients_by_Mallows_Cp
# (Intercept)           X      I(X^2)      I(X^3)
#           1           2           3           4
#
```

```
# $coefficients_by_Schwartz_BIC
# (Intercept)           X      I(X^2)      I(X^3)
#           1           2           3           4
#
# $coefficients_by_adjusted_R2
# (Intercept)           X      I(X^2)      I(X^3)
#           1           2           3           4
```

```
subset_selection_object <- regsubsets(
    x = formula,
    data = data_frame,
    method = "backward"
)
analyze_subset_selection_object(subset_selection_object)
```





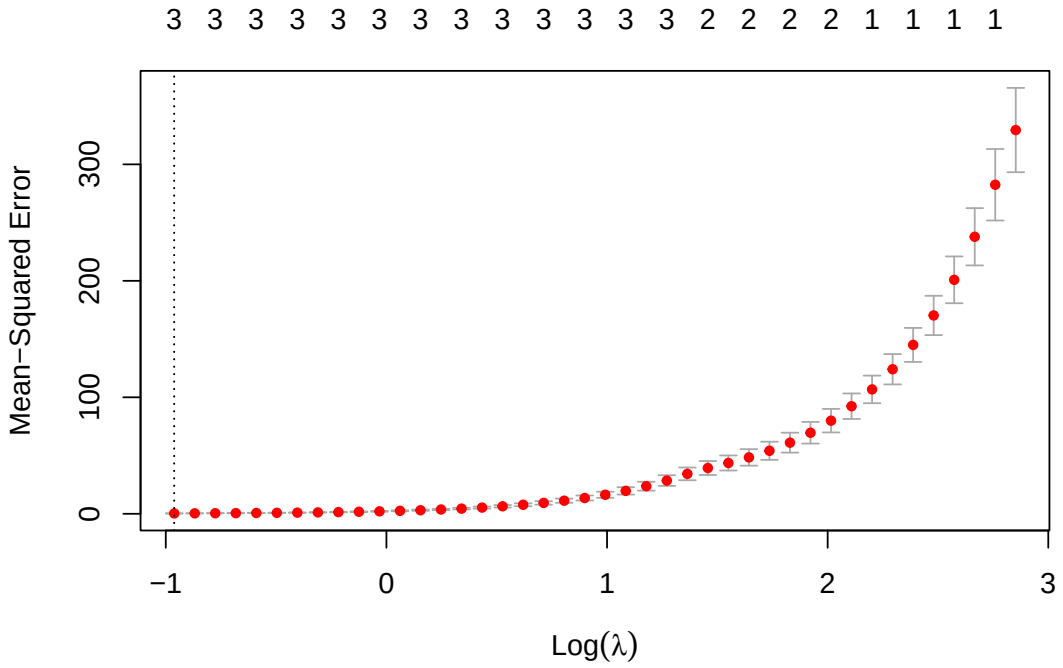




```
# $coefficients_by_Mallows_Cp
# (Intercept)           X      I(X^2)      I(X^3)
#           1           2           3           4
#
```

```
# $coefficients_by_Schwartz_BIC
# (Intercept)           X      I(X^2)      I(X^3)
#           1           2           3           4
#
# $coefficients_by_adjusted_R2
# (Intercept)           X      I(X^2)      I(X^3)
#           1           2           3           4
```

(e) Now fit a lasso model to the simulated data, again using $X, X^2 \ldots, X^{10}$ as predictors. Use cross-validation to select the optimal value of $\lambda$. Create plots of the cross-validation error as a function of $\lambda$. Report the resulting coefficient estimates, and discuss the results obtained.

```
full_model_matrix <- model.matrix(object = formula, data = data_frame)[, -1]
the_cv.glmnet <- glmnet::cv.glmnet(x = full_model_matrix, y = Y, alpha = 1)
plot(the_cv.glmnet)
```




```
optimal_value_of_lambda <- the_cv.glmnet$lambda.min
optimal_value_of_lambda
```

```
# [1] 0.3822143
```

```
polynomial_lasso_regression_model <- glmnet::glmnet(full_model_matrix, y = Y, alpha = 1)
predict(object = polynomial_lasso_regression_model, s = optimal_value_of_lambda, type = "coeff
```

```
# 11 x 1 sparse Matrix of class "dgCMatrix"
#                     s1
# (Intercept) 1.364119
# X           1.990943
# I(X^2)      2.730444
# I(X^3)      3.899032
```

```
# I(X^4)      .
# I(X^5)      .
# I(X^6)      .
# I(X^7)      .
# I(X^8)      .
# I(X^9)      .
# I(X^10)     .
```

The models chosen by polynomial lasso regression involve the true predictive terms and rough approximations of their coefficients. $\beta_0 = 1.364$, $\beta_1 = 1.991$, $\beta_2 = 2.730$, and $\beta_3 = 3.899$.

(f) Now generate a response vector $Y$ according to the model $Y = \beta_0 + \beta_7 X^7 + \epsilon$, and perform best subset selection and the lasso. Discuss the results obtained.

## 9. In this exercise, we will predict the number of applications received using the other variables in the College data set.

(a) Split the data set into a training set and a test set.

(b) Fit a linear model using least squares on the training set, and report the test error obtained.

(c) Fit a ridge regression model on the training set, with $\lambda$ chosen by cross-validation. Report the test error obtained.

(d) Fit a lasso model on the training set, with $\lambda$ chosen by cross-validation. Report the test error obtained, along with the number of non-zero coefficient estimates.

(e) Fit a PCR model on the training set, with $M$ chosen by cross-validation. Report the test error obtained, along with the value of $M$ selected by cross-validation.

(f) Fit a PLS model on the training set, with $M$ chosen by cross-validation. Report the test error obtained, along with the value of $M$ selected by cross-validation.

(g) Comment on the results obtained. How accurately can we predict the number of college applications received? Is there much difference among the test errors resulting from these five approaches?