K-FoldCV Test

John Francis Burkhart

10/6/2020

```
library(readr)
feature_matrix <- read_csv("feature_matrix.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   Primer_Name_L = col_character(),
##   Seq_L = col_character(),
##   exp = col_character(),
##   flank_L = col_logical(),
##   Primer_Name_R = col_character(),
##   Seq_R = col_character(),
##   flank_R = col_character(),
##   sample_id = col_character(),
##   sample_name = col_character(),
##   species = col_character(),
##   strain = col_logical()
## )
```

```
## See spec(...) for full column specifications.
```

```
View(feature_matrix)
data  <- feature_matrix
data <- data %>% select(is.numeric)
```

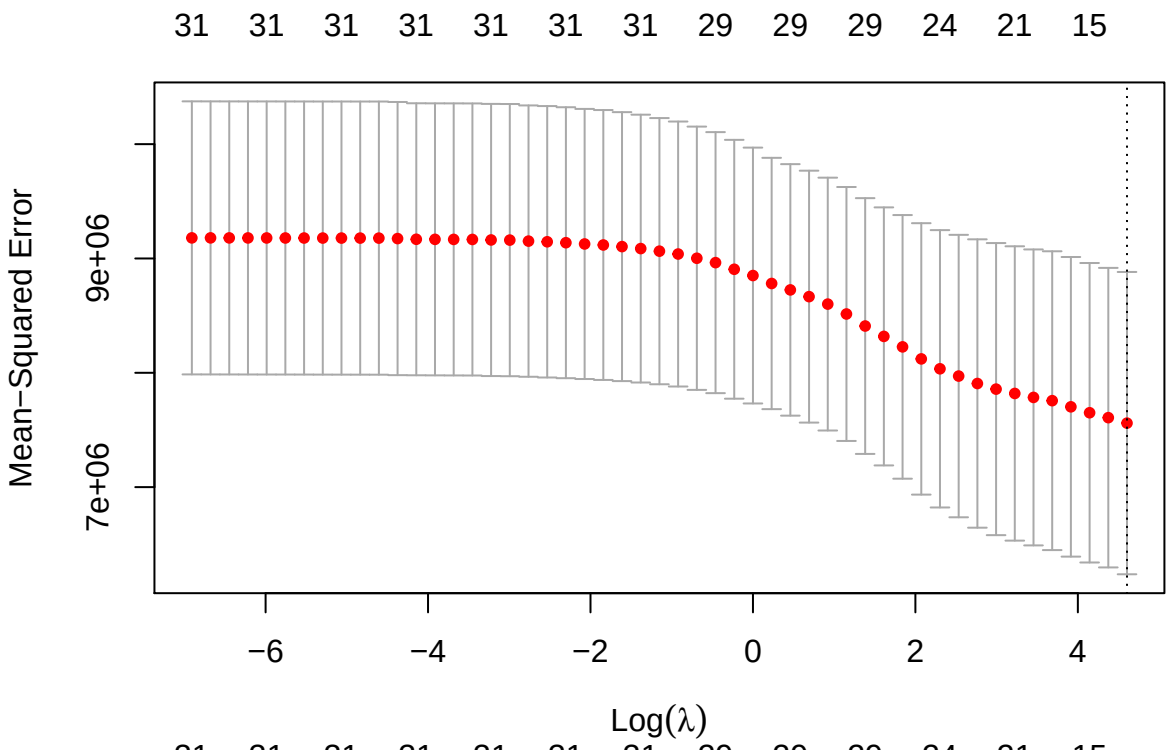
```
## Warning: Predicate functions must be wrapped in `where()`.
##
##   # Bad
##   data %>% select(is.numeric)
##
##   # Good
##   data %>% select(where(is.numeric))
##
## i Please update your code.
## This message is displayed once per session.
```

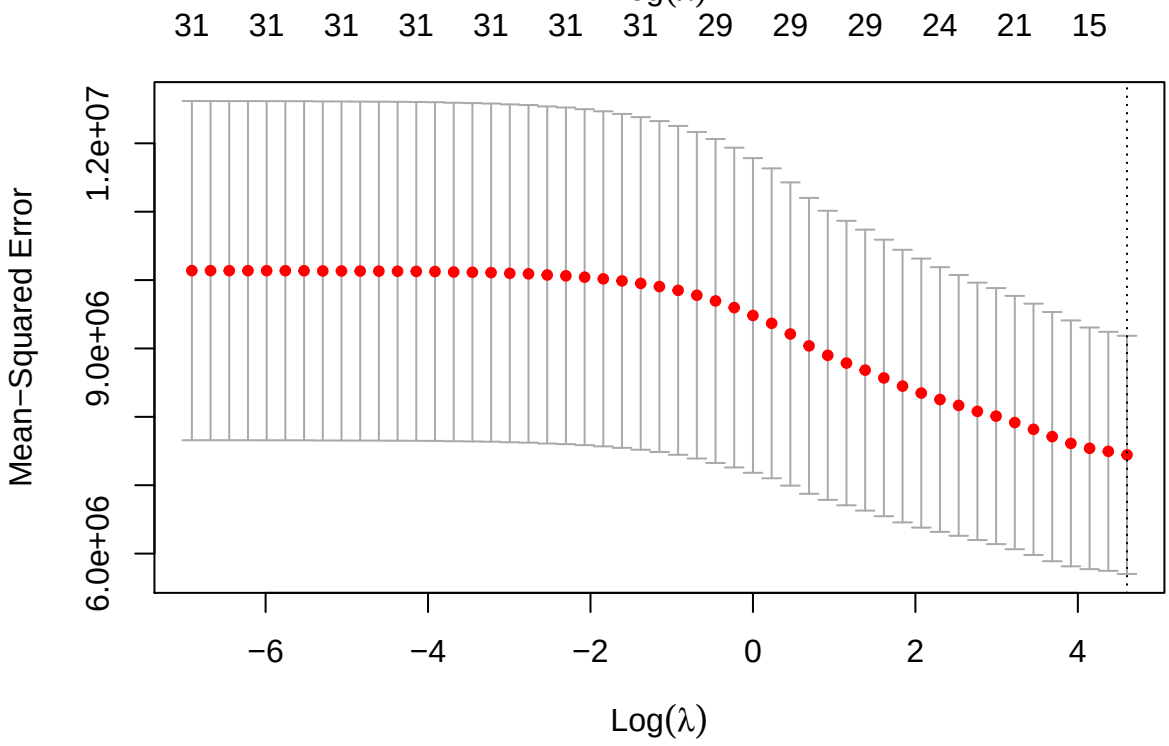
```
X <- select(data, -n_reads)
y <- data$n_reads

X <- as.matrix(X)
```

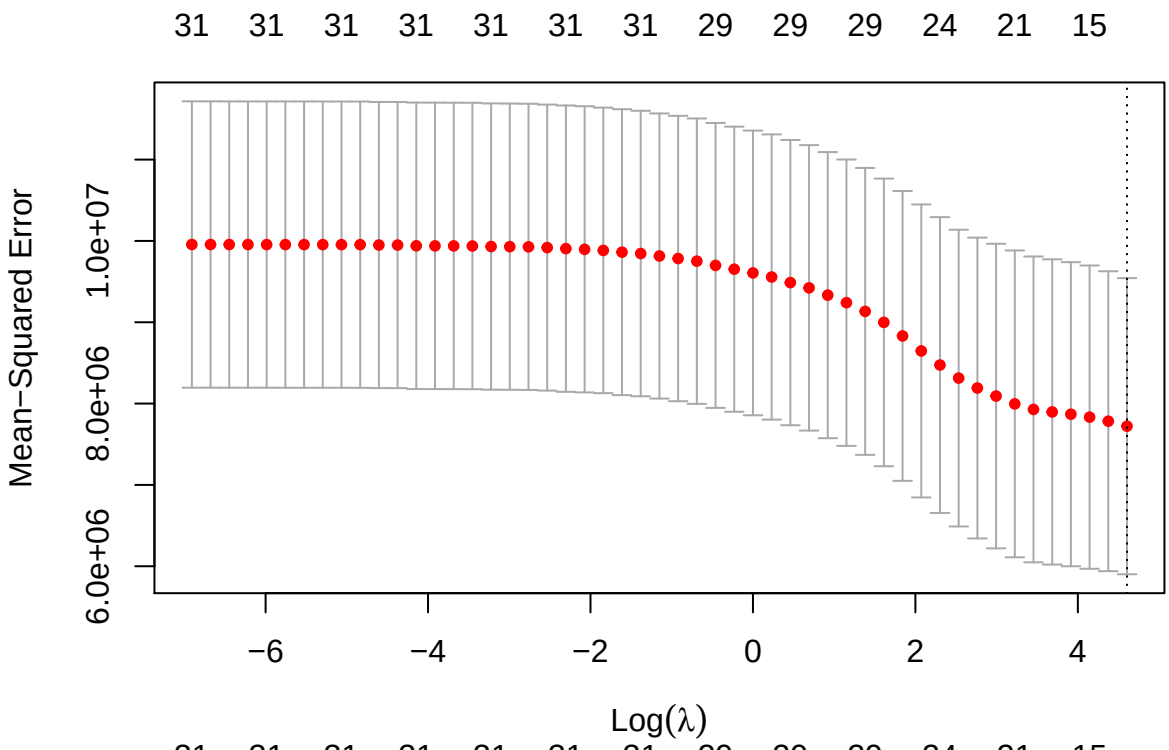
```r
X.scaled <- scale(X)

# K-Fold CV stuff

n.folds <- 5
set.seed(1)
fold.vec <- rep(sample(1:n.folds), l = nrow(X.scaled))
table(fold.vec)
```

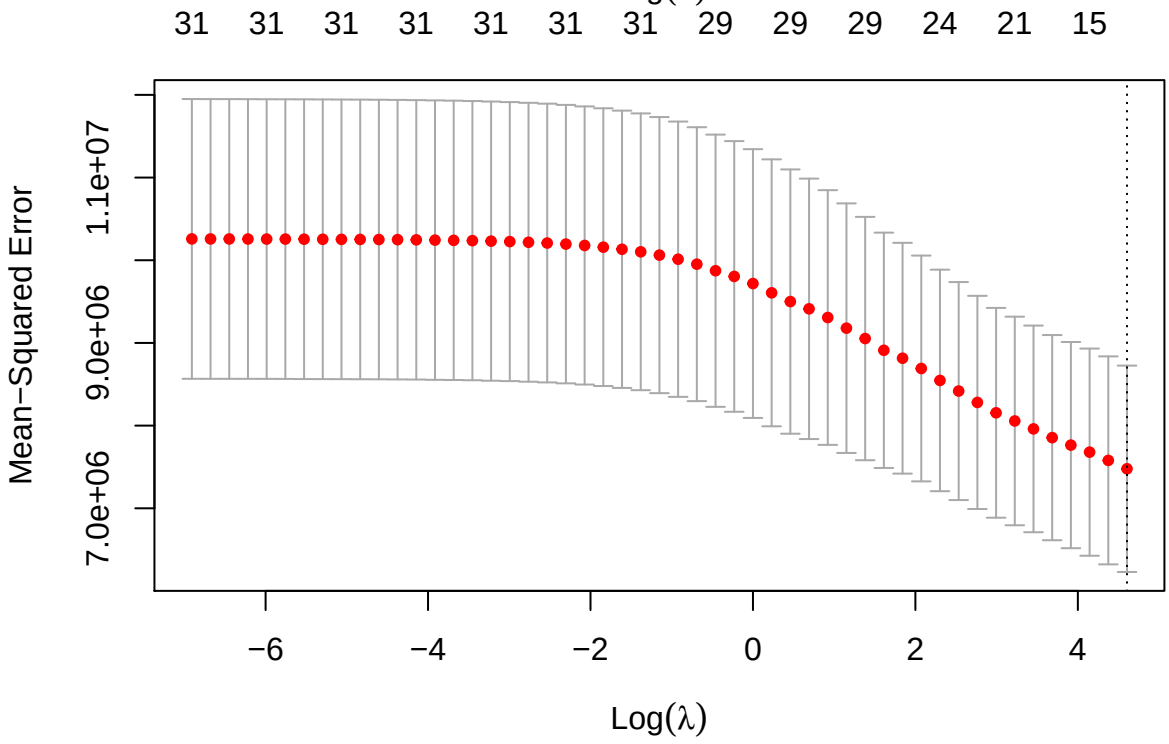
```
## fold.vec
##  1  2  3  4  5
## 40 39 39 39 39
```

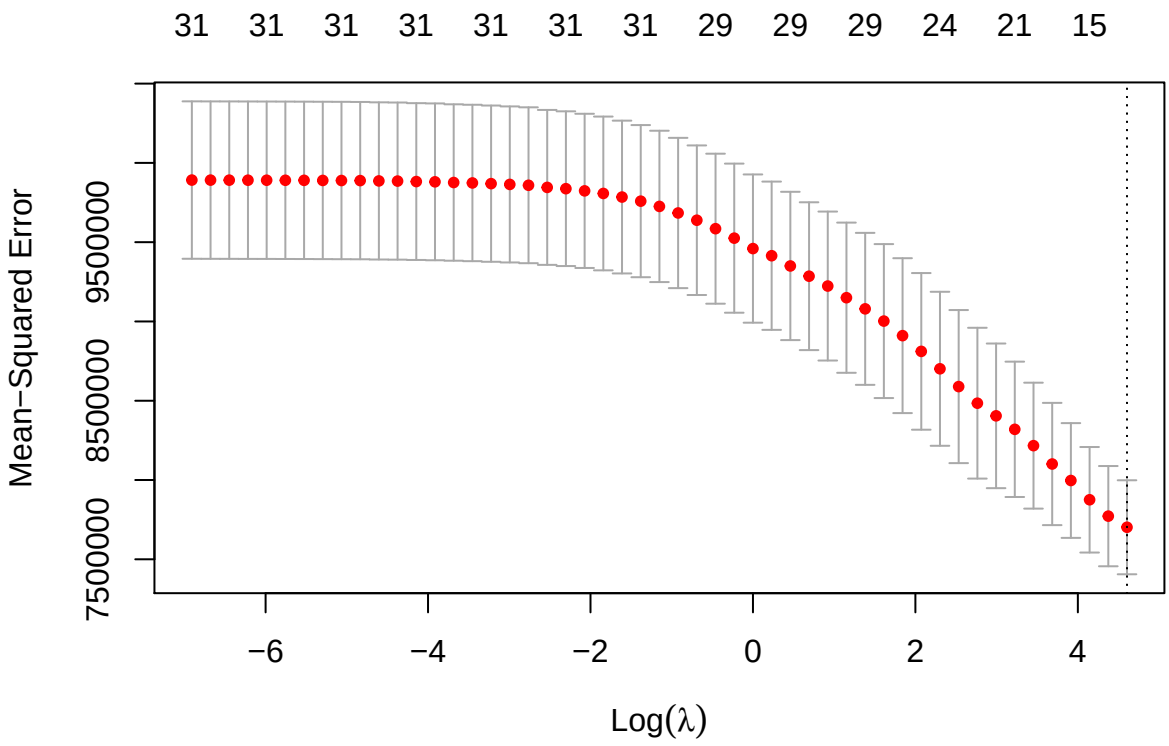
```r
for(test.fold in 1:n.folds){
  test.fold <- 1
  is.test <- fold.vec == test.fold
  is.train <- !is.test

  X.train = X.scaled[is.train, ]
  y.train = y[is.train]
  X.test <- X.scaled[is.test, ]
  y.test <- y[is.test]

  lambdas <- 10^seq(2, -3, by = -.1)

  # Setting alpha = 1 implements lasso regression
  lasso_reg <- cv.glmnet(X.train, y.train, alpha = 1, lambda = lambdas, standardize = TRUE, nfolds = 5)
  plot(lasso_reg)
  # Best
  lambda_best <- lasso_reg$lambda.min
  lambda_best

  lasso_model <- glmnet(X.train, y.train, alpha = 1, lambda = lambda_best, standardize = TRUE)
}
```


31 31 31 31 31 31 31 29 29 29 24 21 15
Mean−Squared Error
7e+06
9e+06
−6
−4
−2
0
2
4
Log(λ)


31 31 31 31 31 31 31 29 29 29 24 21 15
Mean−Squared Error
6.0e+06
9.0e+06
1.2e+07
−6
−4
−2
0
2
4
Log(λ)


31 31 31 31 31 31 31 29 29 29 24 21 15
Mean−Squared Error
6.0e+06
8.0e+06
1.0e+07
−6
−4
−2
0
2
4
Log(λ)


31 31 31 31 31 31 31 29 29 29 24 21 15
Mean−Squared Error
7.0e+06
9.0e+06
1.1e+07
−6
−4
−2
0
2
4
Log(λ)


31 31 31 31 31 31 31 29 29 29 24 21 15
Mean−Squared Error
7500000
8500000
9500000
−6
−4
−2
0
2
4
Log(λ)