Machine Learning | Homework 3

Liana Harutyunyan

March 13, 2019

Libraries that are going to be used:

```
library(ggplot2)
library(dplyr)
library(glmnet)
```

```
## Warning: package 'glmnet' was built under R version 3.5.2
```

```
## Warning: package 'foreach' was built under R version 3.5.2
```

1. Use the rnorm() function to generate a predictor X of length n = 100, as well as a noise vector e of length n = 100 (score = 5).

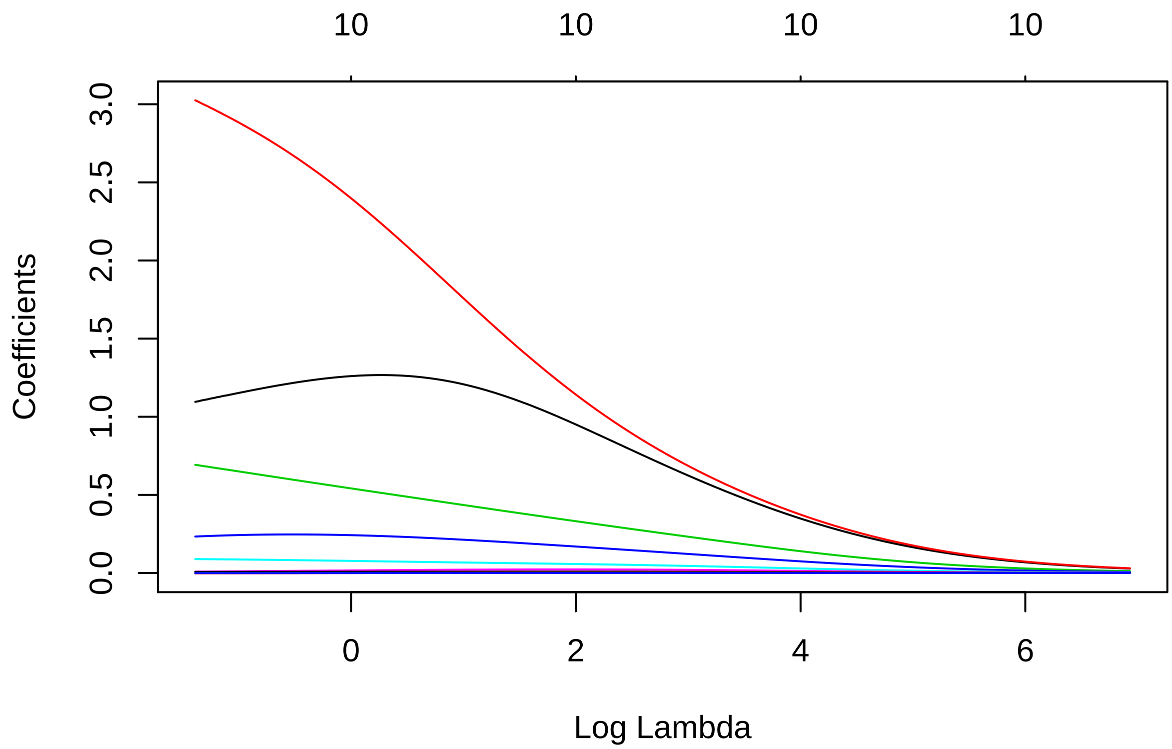
```
set.seed(1)

X <- rnorm(100)
e <- rnorm(100)
```

Generate a response vector Y of length n = 100 according to the model $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + e$**, where** β_0**,** β_1**,** β_2**,** β_3 **are constants of your choice (score = 5).**

```
b0 <- 2
b1 <- 0.3
b2 <- 4.1
b3 <- 1.45

Y <- b0 + b1 * X + b2 * X^2 + b3 * X^3 + e

data.full <- data.frame(Y = Y, X = X)
xmat <- model.matrix(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(

set.seed(1)
train_indices <- sample(1:length(Y), length(Y) / 2)
test_indices <- (-train_indices)
y_train <- Y[train_indices]
y_test <- Y[test_indices]
x_train <- xmat[train_indices, ]
x_test <- xmat[test_indices, ]
```

2. Fit the Ridge model to the simulated data, using X**,** X^2**, ...,** X^10 **as predictors (score = 10).**

```
grid <- 2^seq (10 , -2 , length = 300)
mod_ridge <- glmnet(xmat, Y, alpha = 0, lambda = grid)

plot(mod_ridge, "lambda")
```



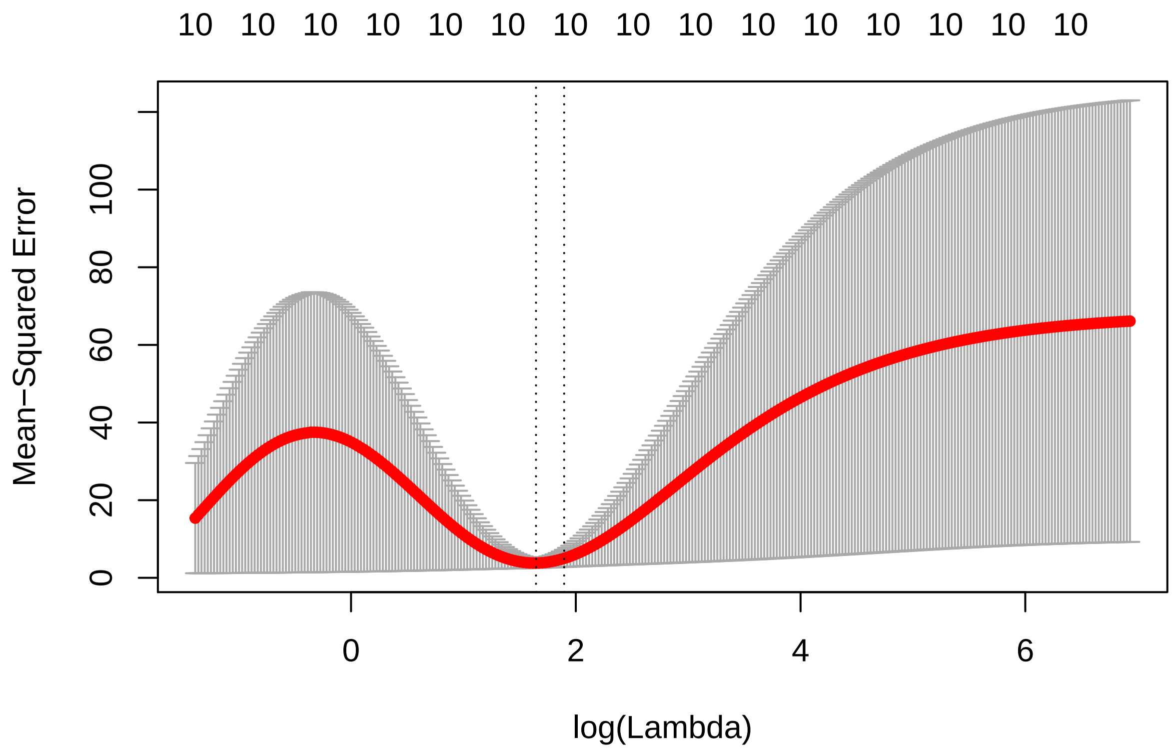
Use cross-validation (10-fold) to select the optimal value of λ (score =10).

```
cv_out <- cv.glmnet(x_train, y_train, alpha = 0, lambda = grid)
bestlam_ridge <- cv_out$lambda.min
bestlam_ridge
```

```
## [1] 5.185855
```

Create plots of the cross-validation error as a function of λ (score = 10).

```
plot(cv_out)
```



Report the resulting coefficient estimates and discuss the results (score = 10).

```
mod_ridge_best <- glmnet(xmat, Y, alpha = 0)
predict ( mod_ridge_best , type ="coefficients", s = bestlam_ridge )
```

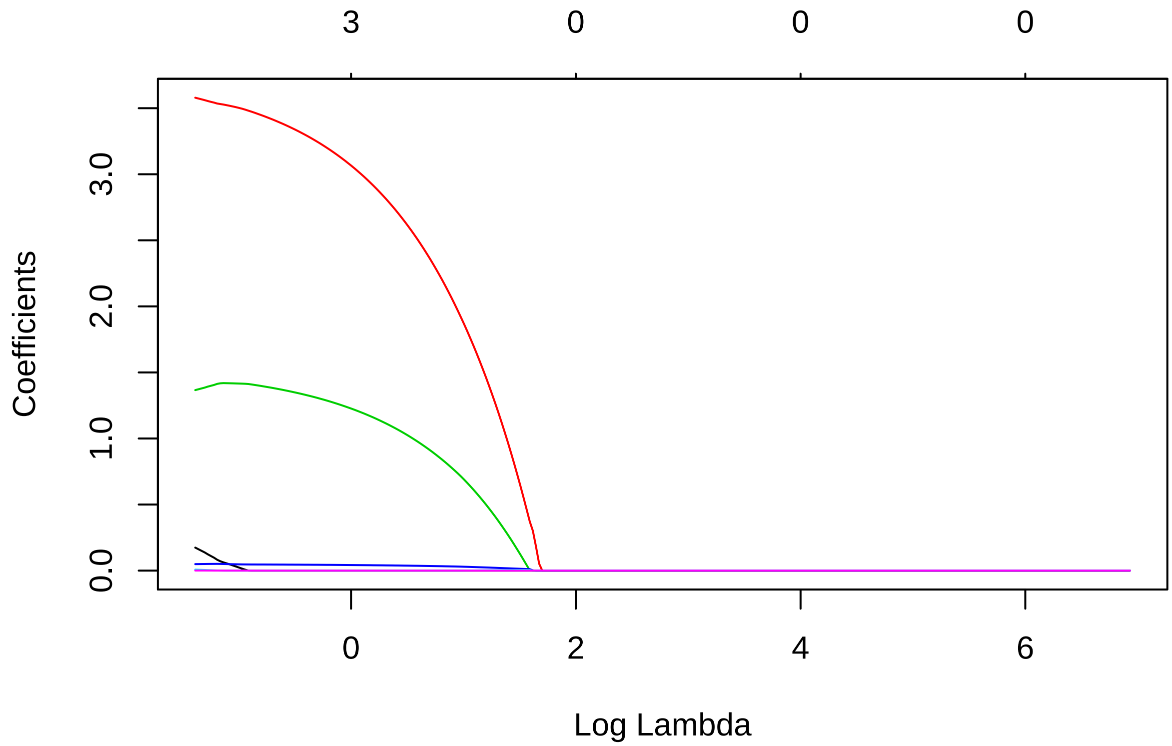
```
## 11 x 1 sparse Matrix of class "dgCMatrix"
##                         1
## (Intercept) 3.5576309352
## X           1.0583564741
## I(X^2)      1.3450117394
## I(X^3)      0.3677716157
## I(X^4)      0.1859531553
## I(X^5)      0.0612914817
## I(X^6)      0.0231318165
## I(X^7)      0.0090105750
## I(X^8)      0.0028582247
## I(X^9)      0.0012842853
## I(X^10)     0.0003540504
```

As we should expect, all the predicted coefficients are non-zero as ridge does not do variable selection.

3. Fit the Lasso model to the simulated data, using X, X^2, . . . , X^10 as predictors (score = 10).

```
mod_lasso <- glmnet(xmat, Y, alpha = 1, lambda = grid)
```

```r
plot(mod_lasso, "lambda")
```



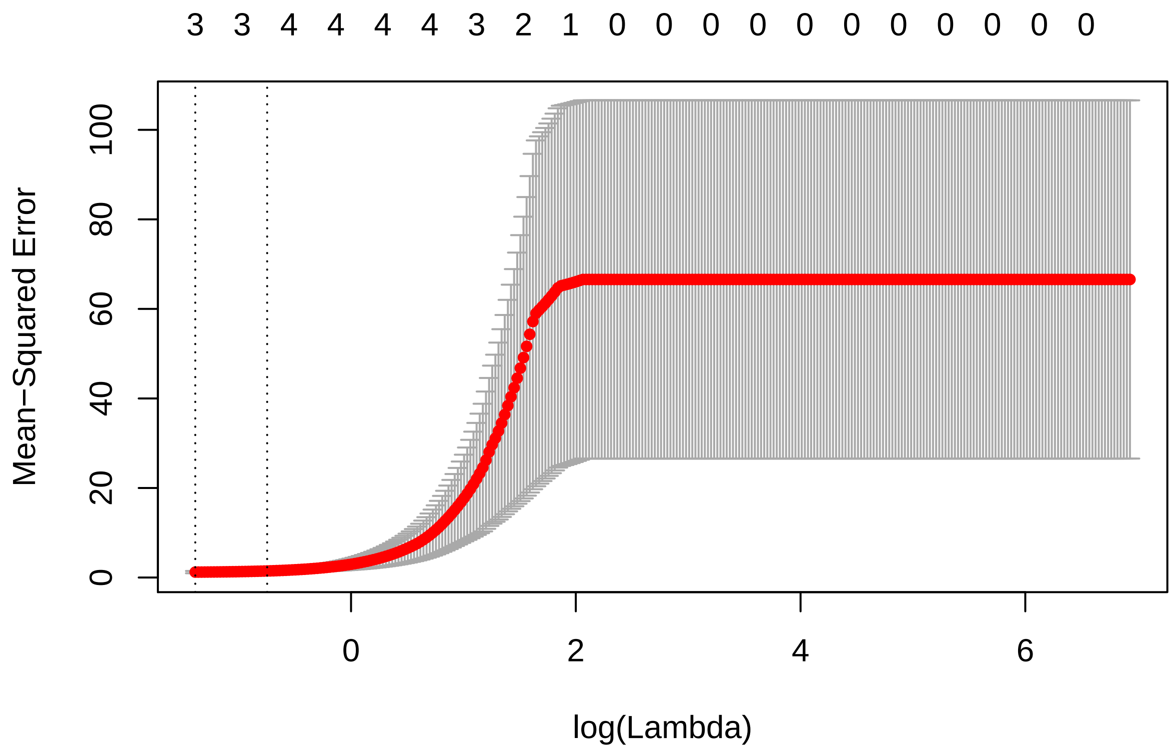
Use cross-validation (10-fold) to select the optimal value of λ (score =10).

```r
cv_out <- cv.glmnet(x_train, y_train, alpha = 1, lambda = grid)
bestlam_lasso <- cv_out$lambda.min
bestlam_lasso
```

```
## [1] 0.25
```

Create plots of the cross-validation error as a function of λ (score = 10).

```r
plot(cv_out)
```



Report the resulting coefficient estimates and discuss the results (score = 10).

```
mod_lasso_best <- glmnet(xmat, Y, alpha = 1)
predict ( mod_lasso_best , type ="coefficients", s = bestlam_lasso )
```

```
## 11 x 1 sparse Matrix of class "dgCMatrix"
##                        1
## (Intercept) 2.3109306983
## X           0.1768158286
## I(X^2)      3.5788434093
## I(X^3)      1.3631351768
## I(X^4)      0.0492303511
## I(X^5)      0.0084942397
## I(X^6)      .
## I(X^7)      0.0005412495
## I(X^8)      .
## I(X^9)      .
## I(X^10)     .
```

As we can see here we have a complete different picture, almost half of coefficients are zero (X^6, X^8 ...). The others are the selected, more important variables, as Lasso does variable selection.

4. Compare the results (score = 10).

As we can see below Test MSE for Ridge Regression was 4.742906, while one for Lasso Regression was 0.858333. As we can see Lasso regression gave better results with its best lambda than the Ridge.

Moreover, as stated before, Lasso has a significant advantage, which is doing variable selection. Here we see that a part of its estimated coefficients are zero, which makes the number of variables being used in Lasso model 5 out of 10, while the number of variables in Ridge is 10 out of 10.

```
ridge_pred <- predict(mod_ridge,s = bestlam_ridge, newx = x_test)
mean((ridge_pred-y_test)^2)
```

```
## [1] 4.742906
```

```
lasso_pred <- predict(mod_lasso,s = bestlam_lasso, newx = x_test)
mean((lasso_pred-y_test)^2)
```

```
## [1] 0.858333
```