Introduction to Machine Learning - Lab 2

Gustav Sternelöv

Thursday, October 29, 2015

Assignment 1

1.1

The function *ridgereg_nfoldCV* which does ridge regression by using n-fold cross validation is implemented at the first step of assignment 1. The R-code used to create the function can be seen below.

```
ridgereg_nfoldCV <- function(x, y, lambda, nfolds){
  # Create the folds and initialize CV vector
  n <- length(y)
  seques <- floor(n/nfolds)
  reps <- n%%nfolds
  groups <- rep(seq(1,nfolds, 1), seques)
  end_values <- rep(nfolds, reps)
  folds <- c(groups, end_values)
  folds <- sort(folds)

  x <- cbind(rep(1, nrow(x)), x)
  CV <- 0
  for (i in 1:nfolds){
    testIndexes <- which(folds==i,arr.ind=TRUE)
    testData_x <- x[testIndexes, ]
    testData_y <- y[testIndexes]
    trainData_x <- x[-testIndexes, ]
    trainData_y <- y[-testIndexes]

    # Perform ridge regression on train data
    x_t <- t(trainData_x)
    I <- diag(ncol(trainData_x))
    BetaRidge <- solve(x_t %*% trainData_x + lambda * I) %*% x_t %*% trainData_y
    #y_hat <- trainData_x %*% BetaRidge
    # Test regression on test data and compare with true values for test data
    y_hat_test <- testData_x %*% BetaRidge
    # Calculates CV
    CV[i] <- sum((testData_y - y_hat_test)^2)
  }
  CV_score <- (1/nfolds * sum(CV))
  return(CV_score)
}
```

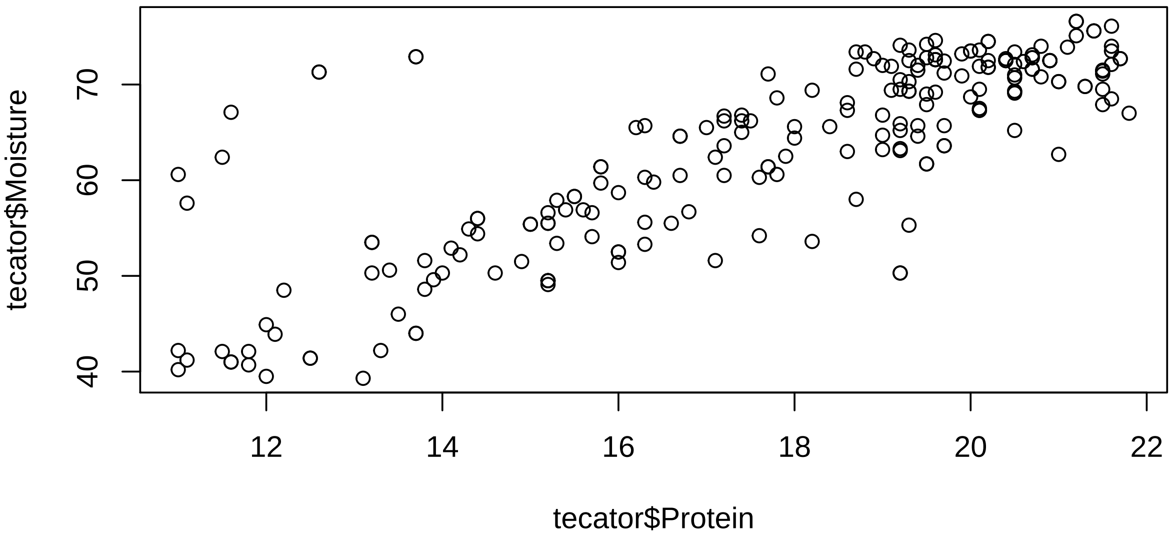
1.2

The function implemented in 1.2 is tested with the longley data set. Number of folds is set to 10 and the value for lambda goes from 1 to 7, by 1. The given values for the CV score is shown in the following table and it can be seen that the CV score decreases for higher values of lambda.

```
## [1] 2.408752
## [1] 2.234662
## [1] 2.082532
## [1] 1.961026
## [1] 1.862823
## [1] 1.781982
## [1] 1.714302
```

Assignment 2

2.1



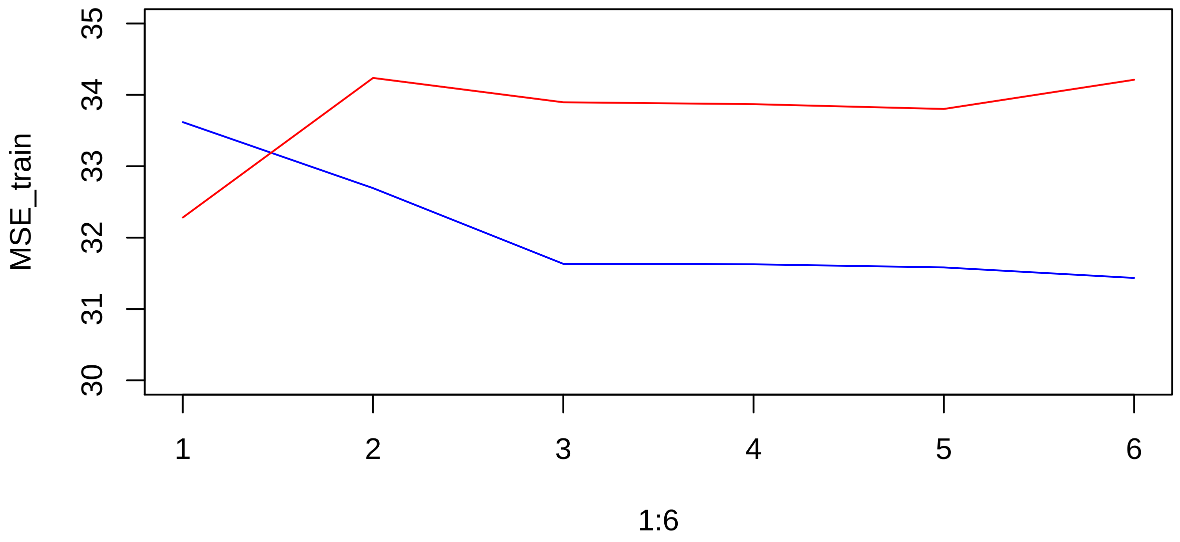
To describe moisture with protein data with a linear model may be an appopriate approach by the look of the plot.

2.2

A Probabilistic model…

2.3

The data set is divided into a training set and a validation set where each set contains 50 percent of the observations.



2.4

Now the whole data set is used to construct the same models. To AIC values is computed to evaluate the models and decide which model that is the best one.

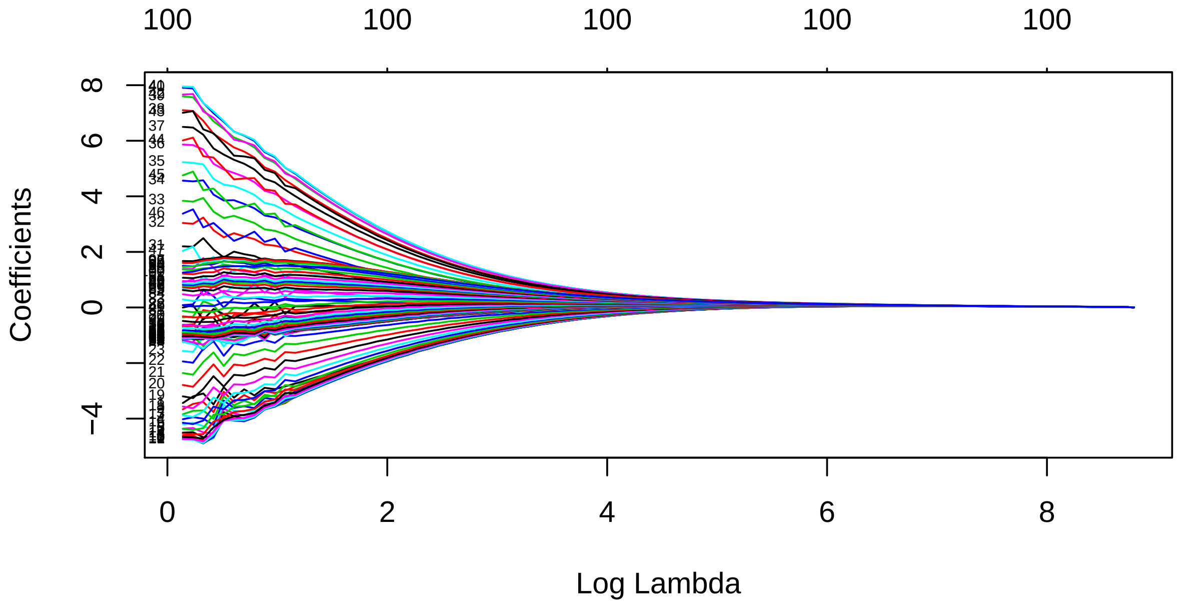
```
## [1] 754.7710 756.4501 753.6657 755.3104 757.1727 759.1538
```

2.5

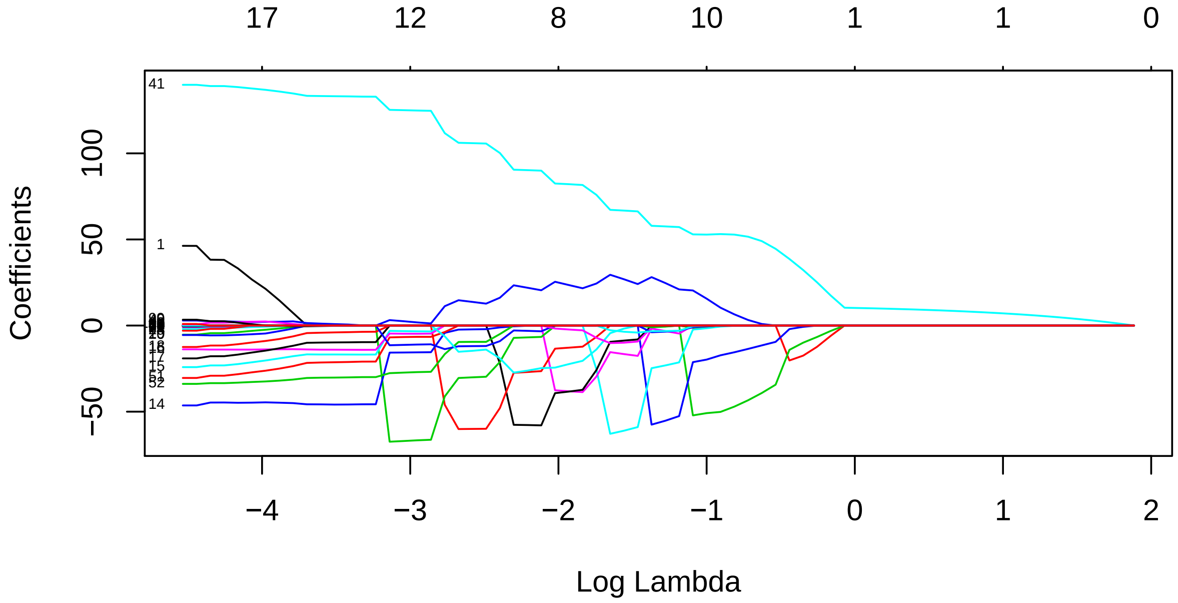
Another area where the AIC criterion can be useful is in the variable selection. Here the function *stepAIC* starts with a linear model that has *fat* as response variable and *channel1-channel100* as predictors. it is concluded that 63 of the predictors are selected.

2.6 - 2.7

With the same response and predictor variables as in 2.5 a ridge regression model is fitted. The difference between a ridge regression model and a linear model is the use of a penalty factor λ in the former model. The value of λ affects the coefficients in such a way that they shrinks. How the values of the coefficients in the ridge regression model depends on the log of λ is illustrated by plotting these values.

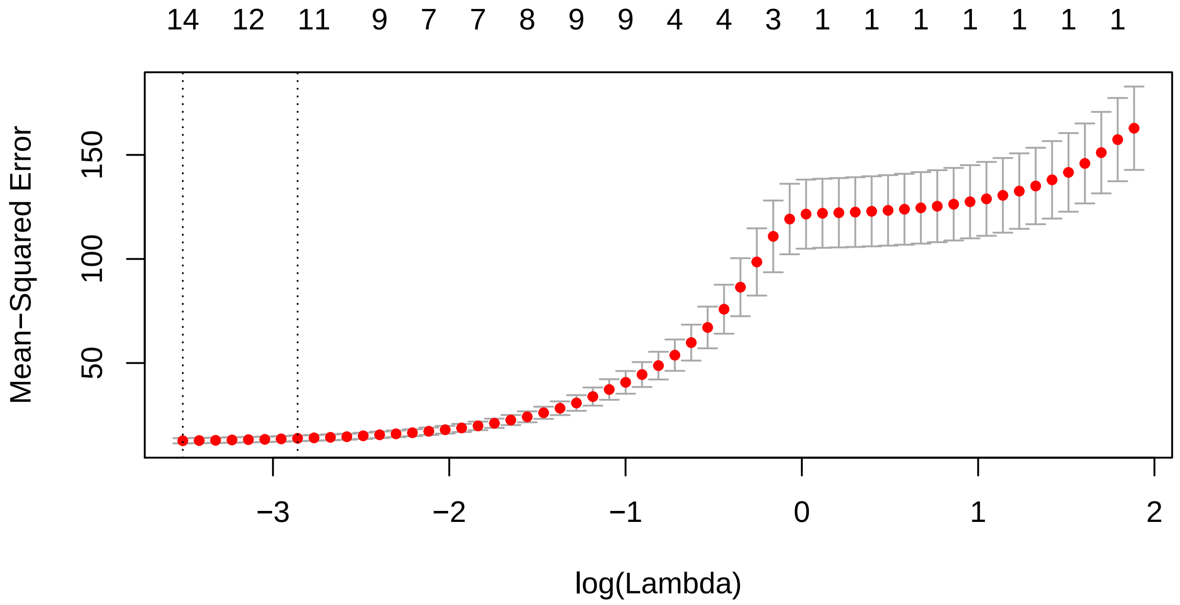


When fitting a LASSO model instead the dependence between the coefficients and log of λ changes.



2.8

To find a optimal LASSO model a possible approach is to use cross-validation



2.9

The variable selection done with *stepAIC* and the selection done with the cross-validated LASSO model choose a significantly different amount of variables. With *stepAIC* a model with 63 variables were obtained and from the cross-validated LASSO computations a model with 14 variables.