Lab05Solution

Frank Guo

2024-04-13

Task 1: Generate simulated data sets.

Step 1.1: Generate the data sets.

```r
my_glmnet_fit <- function(train_df, test_df, a=0){  # a=0 --> ridge; a=1 --> Lasso

  ## separate X and y
  train_x <- as.matrix(select(train_df, -Y))
  train_y <- train_df$Y

  test_x <- as.matrix(select(test_df, -Y))
  test_y <- test_df$Y

  ## run cv.glmnet
  cv.fit <- cv.glmnet(train_x, train_y, alpha = a, lambda = lambdas)
  plot(cv.fit)

  ## optimal lambda, and the corresponding glmnet to the FULL data
  opt.lambda <- cv.fit$lambda.1se   # lambda.min
  fit.glmnet <- cv.fit$glmnet.fit

  ## pick the coefficients
  betas <- as.matrix(coef(fit.glmnet, s=opt.lambda))
  n_nonzero_coef <- sum(betas!=0) # no of non-zero coef

  ## prediction and mSPE
  pred <- predict(fit.glmnet, s=opt.lambda, newx = test_x)
  mspe <- mean((test_y - pred)^2)

  ## collect results and returns
  return(list(alpha = a, MSPE = mspe,
              opt.lambda = opt.lambda,
              n_nonzero_coef = n_nonzero_coef))

}
```

Step 1.2: Understand the code

what is the similarity and difference between the two data sets, dat_A and dat_B?

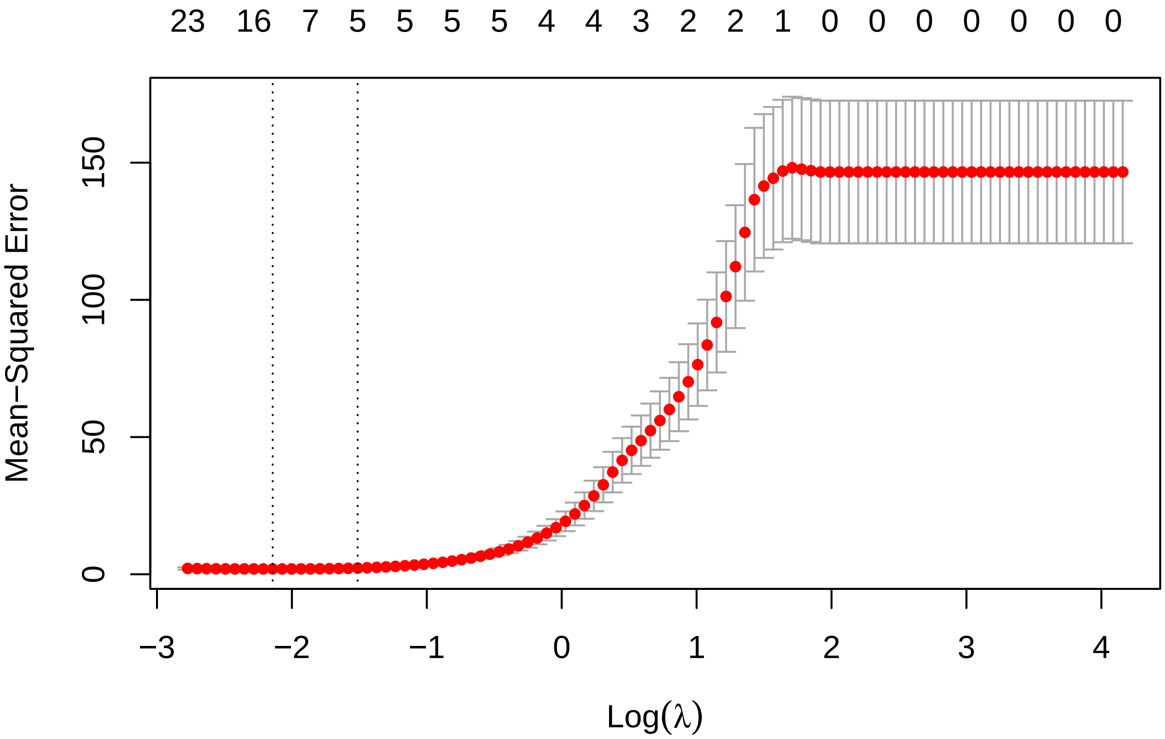
Answer:

dat_A and dat_B has the same predictors generated by mvrnorm with the mean=1 and highly correlated to each other.

But the dependent variable of dat_A heavy related to five of predictors. and dat_A equally related to all the predictors.

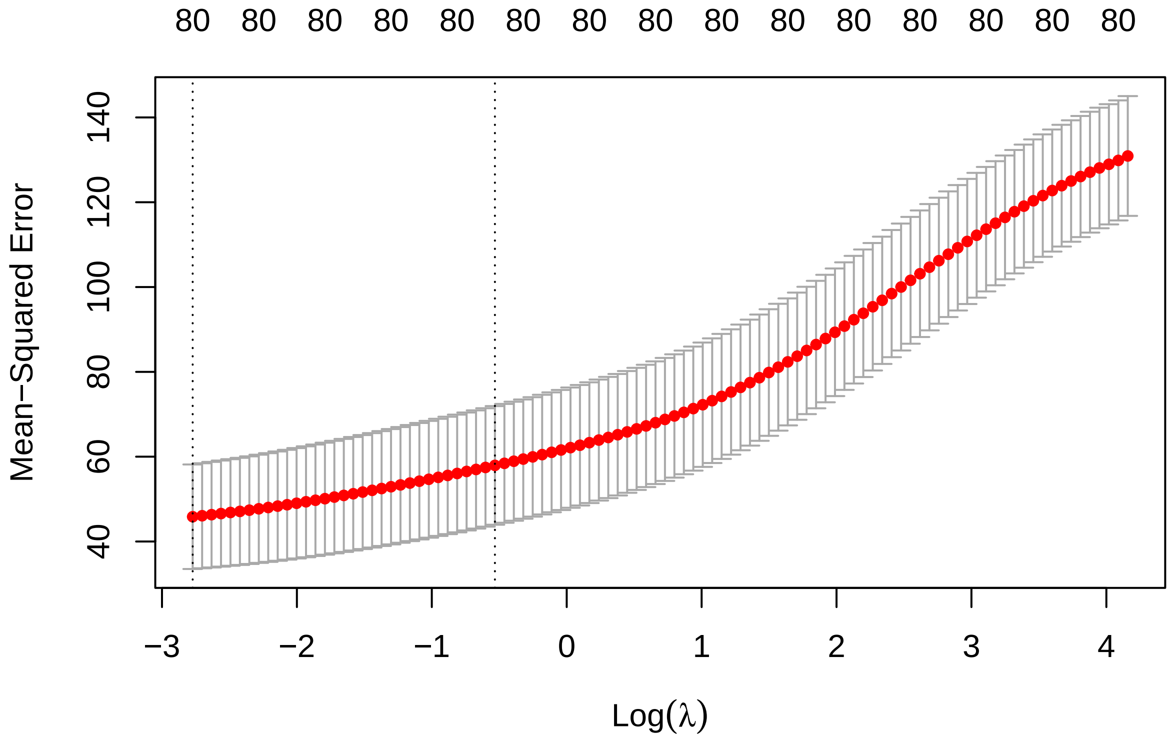
Step 1.3: Split into train and test.

```r
train_ind <- sample(1:n, ceiling(0.8*n))
test_ind <- (1:n)[-train_ind]

train_data_A <- dat_A[train_ind,]
test_data_A <- dat_A[test_ind,]

train_data_B <- dat_B[train_ind,]
test_data_B <- dat_B[test_ind,]

my_glmnet_fit(train_data_A,test_data_A,a=1)
```



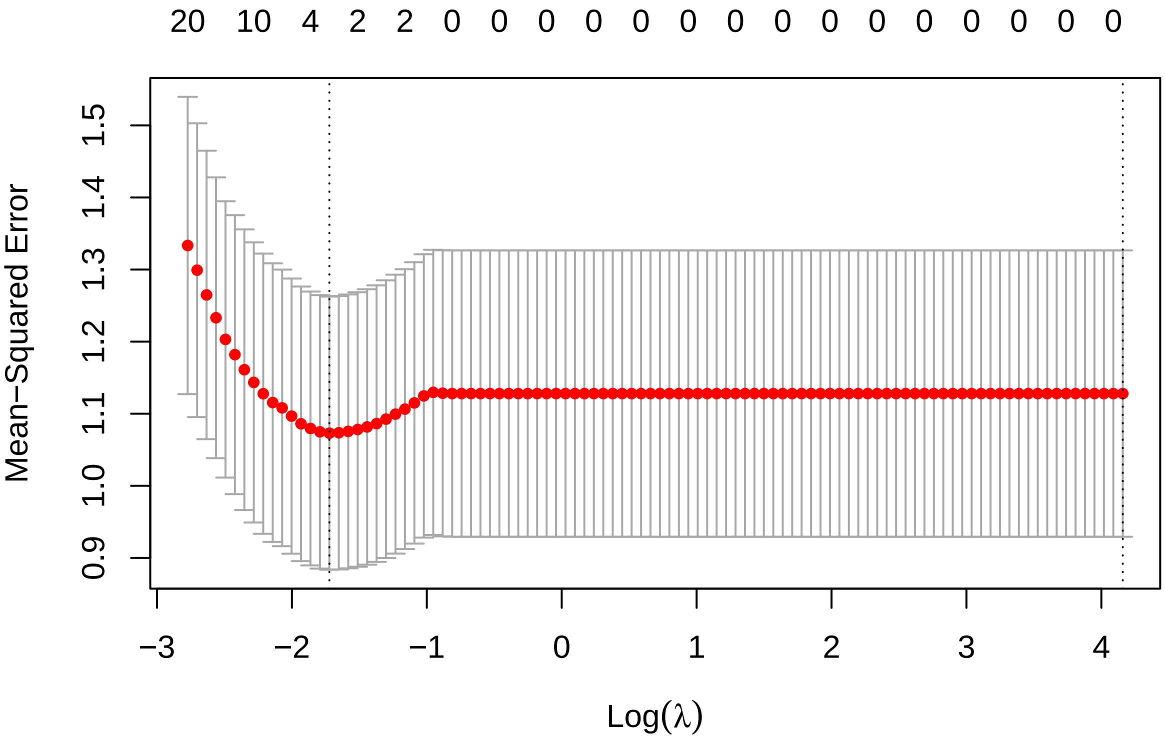
```
## $alpha
## [1] 1
##
## $MSPE
## [1] 2.899334
##
## $opt.lambda
## [1] 0.2203978
##
## $n_nonzero_coef
## [1] 6
```

```
my_glmnet_fit(train_data_A,test_data_A,a=0)
```



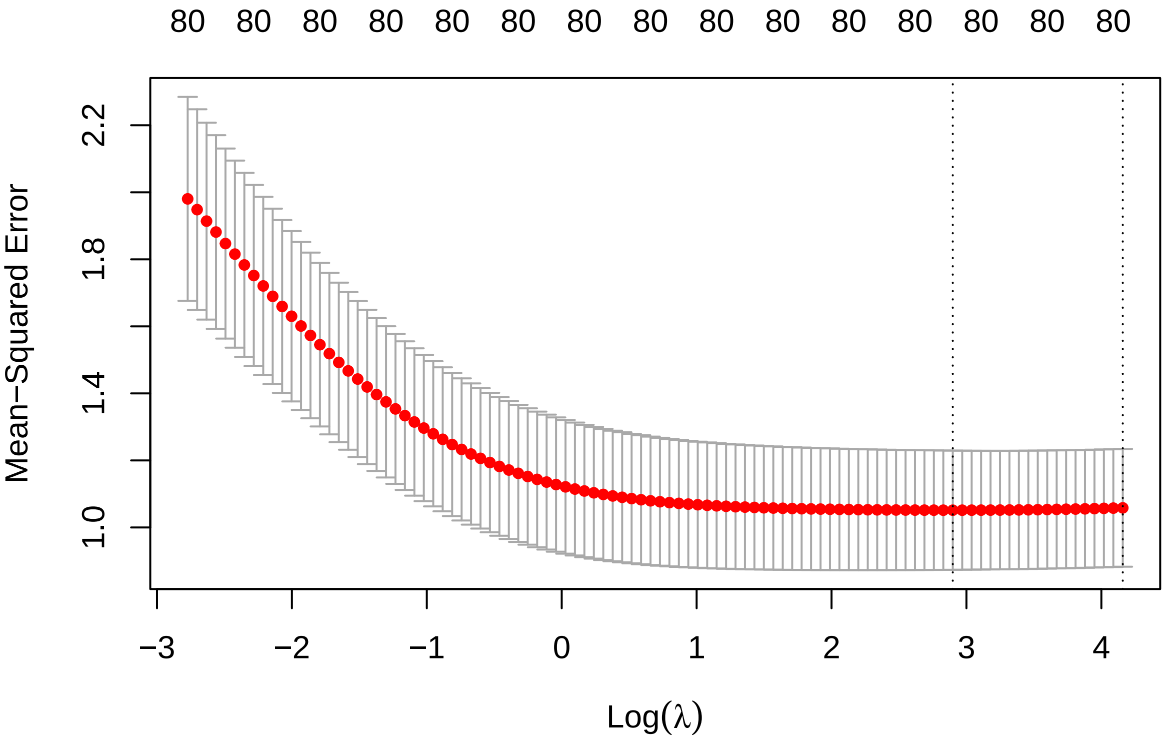
```
## $alpha
## [1] 0
##
## $MSPE
## [1] 51.87221
##
## $opt.lambda
## [1] 0.5873626
##
## $n_nonzero_coef
## [1] 81
```

```
my_glmnet_fit(train_data_B,test_data_B,a=1)
```



```
## $alpha
## [1] 1
##
## $MSPE
## [1] 1.025481
##
## $opt.lambda
## [1] 64
##
## $n_nonzero_coef
## [1] 1
```

```
my_glmnet_fit(train_data_B,test_data_B,a=0)
```



```
## $alpha
## [1] 0
##
## $MSPE
## [1] 1.019322
##
## $opt.lambda
## [1] 64
##
## $n_nonzero_coef
## [1] 81
```

Compare the models:

Lasso for dat_A recognises the five predictors well,and produce the good predict result. mspe 1.495747 is very good campares to the ridge 78.20096.

ridge for dat_b works well for high correlated predictors.lasso is not bad too and create simpler model. for mspe, ridge is a liitle bit better. I'd rather choose lasso in pratice.