Problem Set 2b

2023-11-29

Question 1: Load, prepare, and summarize the data.

```r
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-8
```

```r
library(corrplot)
```

```
## corrplot 0.92 loaded
```

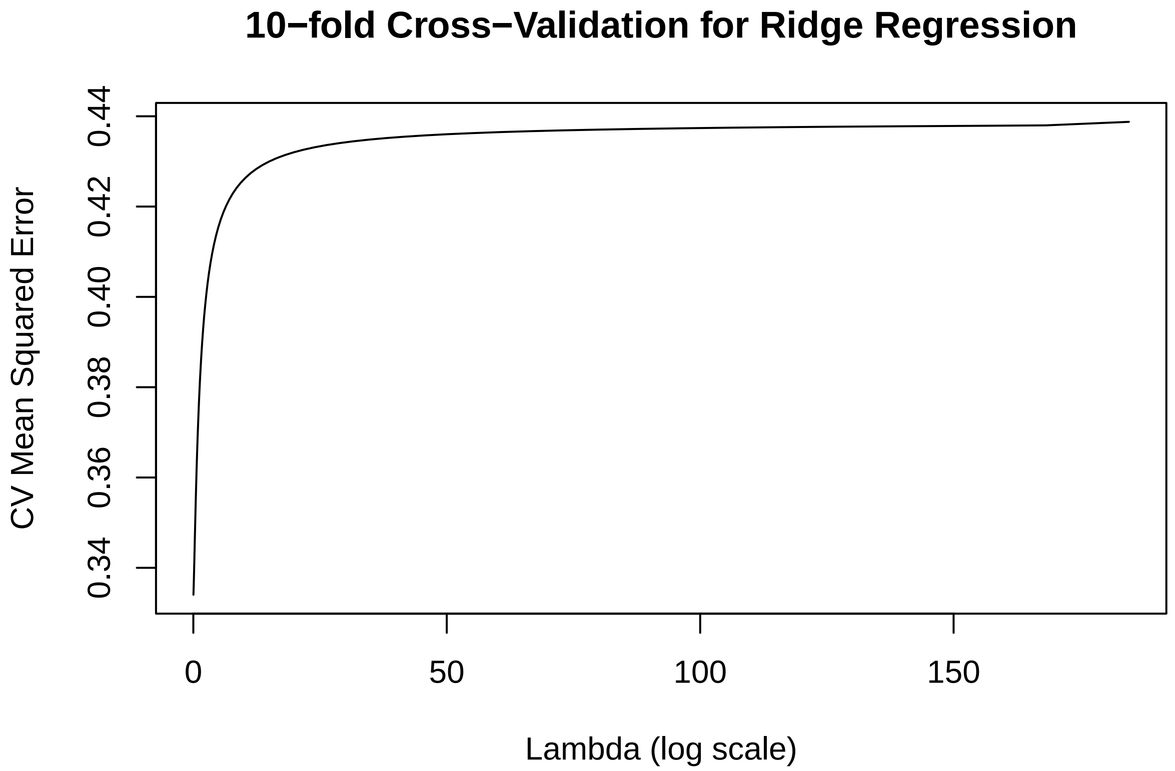
```r
library(hdm)
# This command loads data contained in a R-package.
data(cps2012)
?cps2012
# Construct a regressor matrix for use in the different models.
x <- model.matrix( ~ -1 + female + widowed + divorced + separated + nevermarried +
hsd08+hsd911+ hsg+cg+ad+mw+so+we+exp1+exp2+exp3, data=cps2012)
dim(x)
```

```
## [1] 29217    16
```

```r
y <- cps2012$lnw
```

Question 2: Apply Ridge Regression with CV

```r
ridge_fit <- rlasso(x, y, method = "ridge")
# 10 fold cross validation
cv_fit <- cv.glmnet(x, y, alpha = 0, nfolds = 10)
# Plot it
plot(cv_fit$lambda, cv_fit$cvm, type = "l", xlab = "Lambda (log scale)",
     ylab = "CV Mean Squared Error", main = "10-fold Cross-Validation for Ridge Regression")
```



```
optimal_lambda <- cv_fit$lambda.min
```

```
ridge_optimal_fit <- rlasso(x, y, method = "ridge", lambda = optimal_lambda)
num_variables_used <- sum(ridge_optimal_fit$coef != 0)
cat("Number of variables used in Ridge regression fit:", num_variables_used, "\n")
```

```
## Number of variables used in Ridge regression fit: 14
```

The reason the test MSE for a Ridge regression is often smaller than for OLS when lambda is not zero is due to various factors such as shrinkage of coefficients, bias-variance tradeoff, multicollinearity reduction, and improved stability.

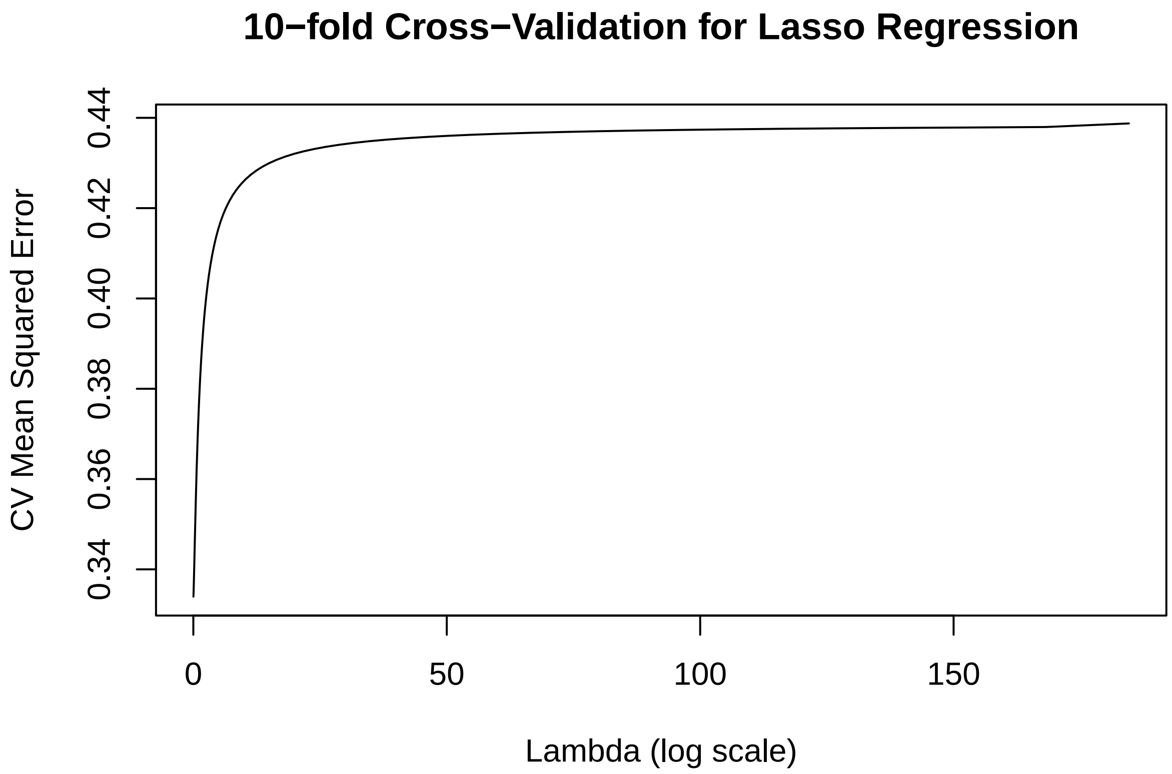
```
cat("The optimal value for lambda in our Ridge Regression problem is: ", optimal_lambda)
```

```
## The optimal value for lambda in our Ridge Regression problem is:  0.01845918
```

Whether unrestricted OLS is optimal here or Ridge Regression is optimal here depends on various factors such as the presence of multicollinearity and the relationship between predictors and the response variable. So based on that and our problem we need to decide which is better.

Question 3: Apply Lasso Regression with CV

```
cv_fit_lasso <- cv.glmnet(x, y, alpha = 1, nfolds = 10)
# 10 fold cross validation
cv_fit <- cv.glmnet(x, y, alpha = 0, nfolds = 10)
# Plot of the MSE for all the lambda values used
plot(cv_fit$lambda, cv_fit$cvm, type = "l", xlab = "Lambda (log scale)",
     ylab = "CV Mean Squared Error", main = "10-fold Cross-Validation for Lasso Regression")
```



```
optimal_lambda_lasso <- cv_fit_lasso$lambda.min
cat("Optimal Lambda selected by cross-validation (Lasso):", optimal_lambda_lasso, "\n")
```

```
## Optimal Lambda selected by cross-validation (Lasso): 7.452004e-05
```

```
lasso_model_final <- glmnet(x, y, alpha = 1, lambda = optimal_lambda_lasso)
num_variables_used_lasso <- sum(coef(lasso_model_final) != 0)
cat("Number of variables used in the optimal Lasso fit:", num_variables_used_lasso, "\n")
```

```
## Number of variables used in the optimal Lasso fit: 17
```

```
coefficients_lasso <- coef(lasso_model_final)
cat("Coefficients of the optimal Lasso model:\n")
```

```
## Coefficients of the optimal Lasso model:
```

```
print(coefficients_lasso)
```

```
## 17 x 1 sparse Matrix of class "dgCMatrix"
##                      s0
## (Intercept)   2.47732797
## female       -0.27953288
## widowed      -0.14203369
## divorced     -0.07796664
## separated    -0.10818413
## nevermarried -0.13131962
## hsd08          -0.61290619
## hsd911         -0.39086942
## hsg            -0.17319759
## cg              0.35189072
## ad              0.59926504
## mw             -0.10546213
## so             -0.05371613
## we             -0.01092662
## exp1            0.04141845
## exp2           -0.12645027
## exp3            0.01342990
```

Judging by the two graphs, there doesn't seem to be too much of a difference between the two models. But I would still choose Lasso because it essentially removes the variables that are not really contributing to the prediction. This way we are keeping the variables that are mostly likely significant to the problem.

It seems to be that gender is an important variable in that it has an inversely proportional relationship with the dependent variable due it being a negative number.

Question 4: Using a more flexible model

```
X <- model.matrix( ~ -1 +female+
female:(widowed+divorced+separated+nevermarried+
hsd08+hsd911+ hsg+cg+ad+mw+so+we+exp1+exp2+exp3) +
+ (widowed + divorced + separated + nevermarried +
hsd08+hsd911+ hsg+cg+ad+mw+so+we+exp1+exp2+exp3)^2,
data=cps2012)
dim(X)
```

```
## [1] 29217   136
```

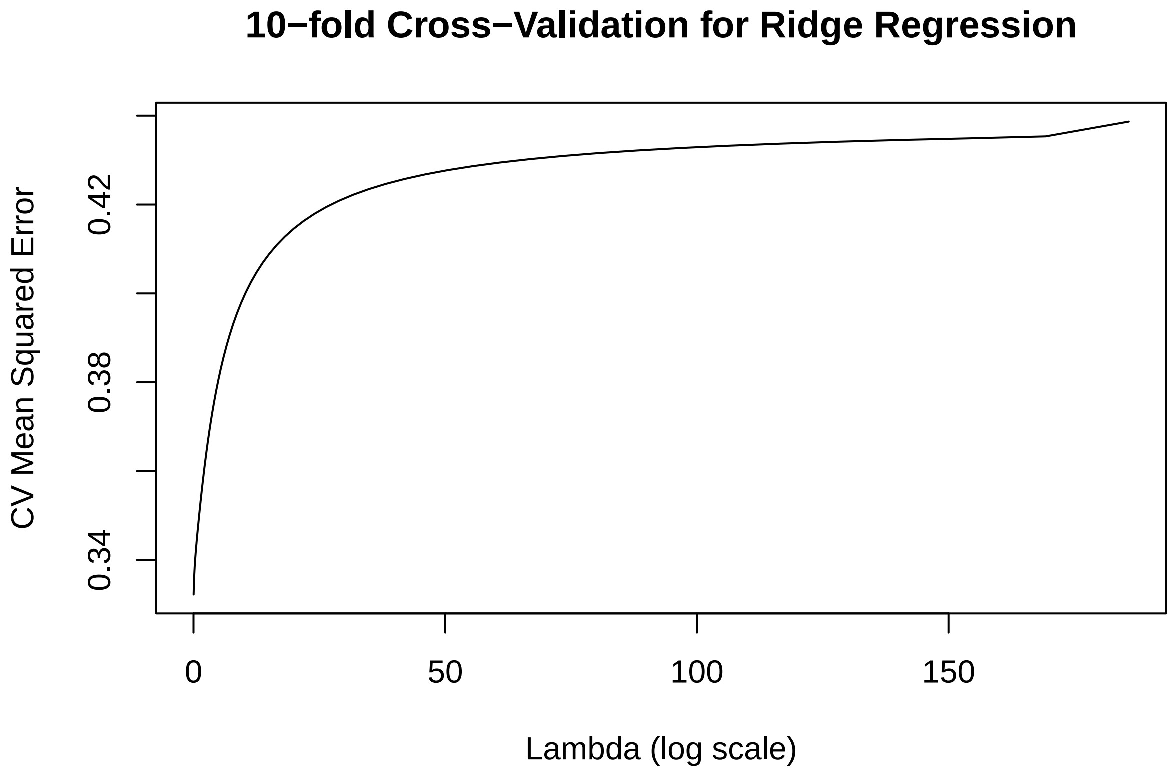
```
# Safety check: Exclude all constant variables.
X <- X[,which(apply(X, 2, var)!=0)]
dim(X)
```

```
## [1] 29217   116
```

```
index.gender <- grep("female", colnames(X))
```

Ridge

```
ridge_fit <- rlasso(X, y, method = "ridge")
# 10 fold cross validation
cv_fit <- cv.glmnet(X, y, alpha = 0, nfolds = 10)
# Plot it
plot(cv_fit$lambda, cv_fit$cvm, type = "l", xlab = "Lambda (log scale)",
     ylab = "CV Mean Squared Error", main = "10-fold Cross-Validation for Ridge Regression")
```

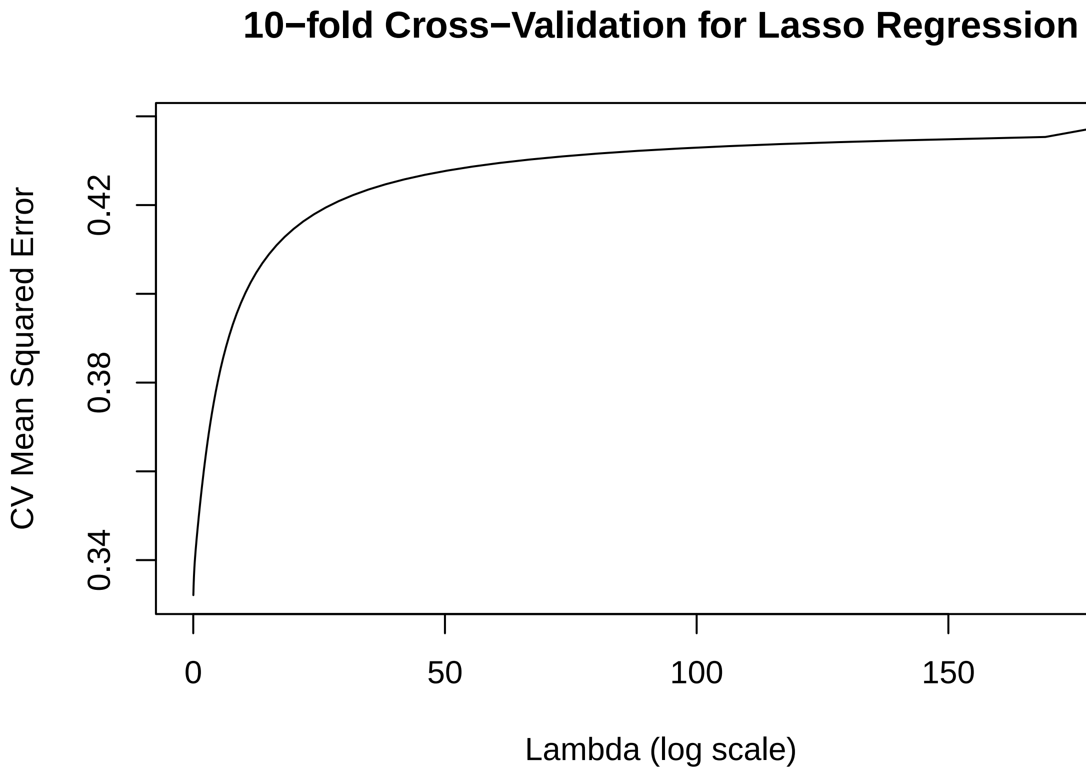


```
optimal_lambda <- cv_fit$lambda.min
ridge_optimal_fit <- rlasso(X, y, method = "ridge", lambda = optimal_lambda)
num_variables_used <- sum(ridge_optimal_fit$coef != 0)
cat("Number of variables used in Ridge regression fit:", num_variables_used, "\n")
```

```
## Number of variables used in Ridge regression fit: 27
```

Lasso

```
cv_fit_lasso <- cv.glmnet(X, y, alpha = 1, nfolds = 10)
# 10 fold cross validation
cv_fit <- cv.glmnet(X, y, alpha = 0, nfolds = 10)
# Plot of the MSE for all the lambda values used
plot(cv_fit$lambda, cv_fit$cvm, type = "l", xlab = "Lambda (log scale)",
     ylab = "CV Mean Squared Error", main = "10-fold Cross-Validation for Lasso Regression")
```



```
optimal_lambda_lasso <- cv_fit_lasso$lambda.min
cat("Optimal Lambda selected by cross-validation (Lasso):", optimal_lambda_lasso, "\n")
```

```
## Optimal Lambda selected by cross-validation (Lasso): 0.0006372962
```

```
lasso_model_final <- glmnet(X, y, alpha = 1, lambda = optimal_lambda_lasso)
num_variables_used_lasso <- sum(coef(lasso_model_final) != 0)
cat("Number of variables used in the optimal Lasso fit:", num_variables_used_lasso, "\n")
```

```
## Number of variables used in the optimal Lasso fit: 80
```

```
coefficients_lasso <- coef(lasso_model_final)
cat("Coefficients of the optimal Lasso model:\n")
```

```
## Coefficients of the optimal Lasso model:
```

```
print(coefficients_lasso)
```

```
## 117 x 1 sparse Matrix of class "dgCMatrix"
##                                s0
## (Intercept)          2.596233e+00
## female              -2.207278e-01
## widowed             -2.307525e-02
## divorced            -1.794405e-01
## separated           -4.543581e-02
## nevermarried        -2.050378e-01
## hsd08               -4.116636e-01
## hsd911              -3.534483e-01
## hsg                 -1.625663e-01
## cg                    1.971261e-01
## ad                    4.810745e-01
## mw                   -7.114277e-02
## so                   -3.556343e-02
## we                    .
## exp1                  1.869156e-02
## exp2                 -1.013134e-02
## exp3                 -2.834932e-03
## female:widowed        7.689439e-02
## female:divorced       1.222220e-01
## female:separated      6.664423e-03
## female:nevermarried   1.632487e-01
## female:hsd08         -5.967830e-02
## female:hsd911        -1.442357e-01
## female:hsg           -2.032816e-02
## female:cg             6.036889e-03
## female:ad            -1.103626e-02
## female:mw            -6.735017e-04
## female:so            -5.929632e-03
## female:we             .
## female:exp1          -5.554351e-03
## female:exp2           .
## female:exp3           1.453081e-03
## widowed:hsd911        1.126175e-01
## widowed:hsg           .
## widowed:cg           -2.569666e-01
## widowed:ad           -8.497548e-02
## widowed:mw           -5.098113e-02
## widowed:so           -6.511919e-02
## widowed:we            4.803940e-02
## widowed:exp1         -2.974963e-03
## widowed:exp2          .
## widowed:exp3          .
## divorced:hsd08        9.444911e-02
## divorced:hsd911       .
## divorced:hsg          8.283089e-03
## divorced:cg          -4.618702e-04
## divorced:ad           .
## divorced:mw           4.451576e-02
## divorced:so           .
## divorced:we           4.048800e-02
## divorced:exp1         .
## divorced:exp2         .
## divorced:exp3         1.229606e-03
## separated:hsd08       1.211885e-01
## separated:hsd911     -2.130296e-01
## separated:hsg         1.535430e-03
## separated:cg          9.229505e-03
## separated:ad         -5.327447e-02
## separated:mw         -1.323531e-01
## separated:so         -5.822794e-02
## separated:we         -9.296857e-03
## separated:exp1        .
## separated:exp2        .
## separated:exp3       .
## nevermarried:hsd08   6.578534e-03
## nevermarried:hsd911  4.009765e-02
## nevermarried:hsg     .
## nevermarried:cg      3.303676e-02
## nevermarried:ad      .
## nevermarried:mw     -2.775103e-02
## nevermarried:so      5.012219e-02
## nevermarried:we      .
## nevermarried:exp1    .
## nevermarried:exp2    .
## nevermarried:exp3   -2.194143e-03
## hsd08:mw            -4.877256e-01
## hsd08:so            -2.488618e-01
## hsd08:we             2.318765e-01
## hsd08:exp1           .
## hsd08:exp2           .
## hsd08:exp3           2.102025e-03
## hsd911:mw           -5.061813e-02
## hsd911:so           -7.174314e-02
## hsd911:we            3.155136e-02
## hsd911:exp1          .
## hsd911:exp2          .
## hsd911:exp3          2.650893e-03
## hsg:mw               1.350882e-02
## hsg:so               .
## hsg:we               6.984277e-03
## hsg:exp1             .
## hsg:exp2             .
## hsg:exp3             .
## cg:mw               -1.712891e-02
## cg:so                .
## cg:we               -2.402728e-02
## cg:exp1              1.237619e-02
## cg:exp2              .
## cg:exp3             -7.304647e-03
## ad:mw               -4.426540e-02
## ad:so                .
## ad:we               -6.840779e-02
## ad:exp1              1.392988e-02
## ad:exp2              .
## ad:exp3             -1.266302e-02
## mw:exp1              .
## mw:exp2              .
## mw:exp3             -1.554129e-03
## so:exp1             -1.283287e-04
## so:exp2             -2.889172e-03
## so:exp3             -1.865648e-05
## we:exp1              .
## we:exp2              .
## we:exp3              1.806044e-04
## exp1:exp2           -6.031410e-05
## exp1:exp3            .
## exp2:exp3            .
```

Question 5: What is the most preferred prediction model of all

I would prefer Lasso still simply because of the fact that it removes variables that are not essential to the problem. As for whether the interaction between gender and education is important for predicting wages, I would say it is. Although most of the female and education indicators interaction terms are negative except the cg indicator because it's positive, they all share an inversely proportional relationship with our dependent variable. That is why Lasso also selects this variable. If it was not of any value, then Lasso should have removed it.