HW6

Your Name Goes Here

Details

Due Date

Please commit and push your submission for this assignment to GitHub by 5:00 PM Wednesday Oct 30.

Grading

20% of your grade on this assignment is for completion. A quick pass will be made to ensure that you've made a reasonable attempt at all problems.

Some of the problems will be graded more carefully for correctness. In grading these problems, an emphasis will be placed on full explanations of your thought process. You usually won't need to write more than a few sentences for any given problem, but you should write complete sentences! Understanding and explaining the reasons behind your decisions is more important than making the "correct" decision.

Solutions to all problems will be provided.

Collaboration

You are allowed to work with others on this assignment, but you must complete and submit your own write up. You should not copy large blocks of code or written text from another student.

Sources

You may refer to class notes, our textbook, Wikipedia, etc.. All sources you refer to must be cited in the space I have provided at the end of this problem set.

In particular, you may find the following resources to be valuable:

- Courses assigned on DataCamp
- Example R code from class
- Cheat sheets and resources linked from [http://www.evanlray.com/stat340_f2019/resources.html]

Load Packages

The following R code loads packages needed in this assignment.

```r
library(readr)
library(dplyr)
library(ggplot2)
library(caret)
```

Conceptual Problems

Problem 1: Bias/variance trade offs

For parts (a) through (c), indicate which of i. through iv. is correct. Justify your answer.

(a) Relative to least squares, the lasso is:

i. More flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

ii. More flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias

iii. Less flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

iv. Less flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias

Use the space below to indicate your choice and justify your answer:

iii. LASSO regression imposes a penalty on the size of the coefficients in the linear regression model, which prevents the coefficient estimates from becoming too large. This is a guard against overfitting the training data, and it affects both the bias and the variance of the model.

Often, a linear regression model that has overfit the data will have coefficient estimates that are too large. For example, if I have many possible explanatory variables but they are not all relevant to predicting the response, the optimal coefficient estimates for any irrelevant covariates would be 0. A model estimate that overfits the training data might estimate those coefficients to be something other than 0. LASSO shrinks those coefficient estimates towards 0, thereby reducing the tendency to overfit the training data. In turn this means that the variance of the predicted values is reduced relative to least squares. Another way of framing this is that the model is less flexible when estimated with LASSO because it is less likely to overfit the training data.

The trade off for this is that LASSO introduces bias there there was no bias in least squares estimation. To see this, suppose that there is a variable in our model that is actually relevant to predicting the response. On average across training sets, least squares will get you the right coefficient estimate for that variable, and the resulting predictions are unbiased. However, LASSO will shrink your coefficient estimates towards 0, introducing bias in the predicted values as well.

Overall, LASSO is less flexible than least squares and will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

(b) Relative to least squares, ridge regression is:

i. More flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

ii. More flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias

iii. Less flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

iv. Less flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias

Use the space below to indicate your choice and justify your answer:

iii. The explanation/justification here is exactly the same as for LASSO in part (a). Ridge regression and LASSO both basically work by penalizing the magnitude of coefficient estimates, so the bias/variance trade off they make relative to least squares estimation is the same.

(c) Relative to least squares, non-linear regression methods like regression trees and KNN regression are:

i. More flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

ii. More flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias

iii. Less flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

iv. Less flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias

Use the space below to indicate your choice and justify your answer:

ii. KNN regression is much more flexible than a linear regression method. To see this, recall our example of regression using KNN where the function we were estimating was linear, a polynomial, or a trigonometric function (discussed on Monday, Sep 30). We saw that the same KNN approach with the same value of K was able to successfully estimate all three of those different forms for the relationship between an explanatory and response variable. To achieve good fits with a regression model, we would have to worry much more about specifying exactly the right functional form for that relationship. Because the KNN regression model is more flexible than least squares linear regression, it has lower bias. It will successfully estimate the true function in settings where a linear model would be inappropriate. However, a more flexible model is also more likely to overfit the training data, so KNN has higher variance.

Problem 2: More bias/variance trade offs

Describe how the bias and variance of predictions from LASSO regression change as the penalty parameter λ increases.

As the penalty parameter λ increases, the predictions from LASSO regression have higher bias but lower variance.

Applied Problems

Problem 3: Predicting house sale prices

We have information on the sale price and characteristics of 2,930 houses sold in the city of Ames, IA between 2006 and 2010. Below I read the data in and split the data evenly into training and test sets.

```
houses <- read_csv("http://www.evanlray.com/data/AmesHousing/AmesHousing.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   Lot_Frontage = col_double(),
##   Lot_Area = col_double(),
##   Year_Built = col_double(),
##   Year_Remod_Add = col_double(),
##   Mas_Vnr_Area = col_double(),
##   BsmtFin_SF_1 = col_double(),
##   BsmtFin_SF_2 = col_double(),
##   Bsmt_Unf_SF = col_double(),
##   Total_Bsmt_SF = col_double(),
##   First_Flr_SF = col_double(),
##   Second_Flr_SF = col_double(),
##   Low_Qual_Fin_SF = col_double(),
##   Gr_Liv_Area = col_double(),
##   Bsmt_Full_Bath = col_double(),
##   Bsmt_Half_Bath = col_double(),
##   Full_Bath = col_double(),
##   Half_Bath = col_double(),
##   Bedroom_AbvGr = col_double(),
##   Kitchen_AbvGr = col_double(),
##   TotRms_AbvGrd = col_double()
##   # ... with 15 more columns
## )

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

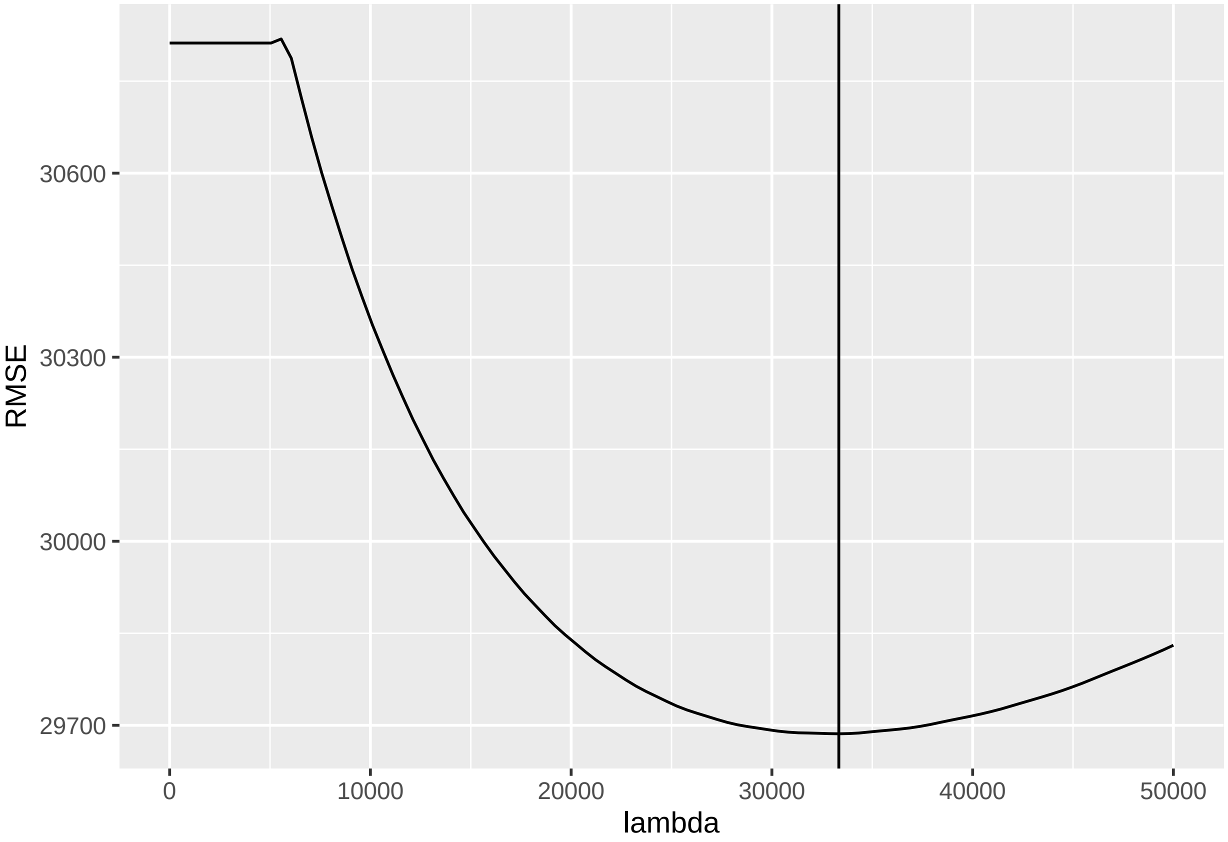
```r
char_cols <- houses %>% summarize_all(is.character) %>% as.matrix()
houses <- houses %>%
  mutate_at(which(char_cols[1, , drop = TRUE]), factor)

set.seed(38355)
train_inds <- caret::createDataPartition(houses$Sale_Price, p = 0.5)
houses_train <- houses %>% slice(train_inds[[1]])
houses_test <- houses %>% slice(-train_inds[[1]])
```

(a) Fit a linear regression model to the data using sale price as the response and all other covariates as predictors.

```r
lm_fit <- train(
  form = Sale_Price ~ .,
  data = houses_train,
  method = "lm",
  trControl = trainControl(method = "none")
)
```

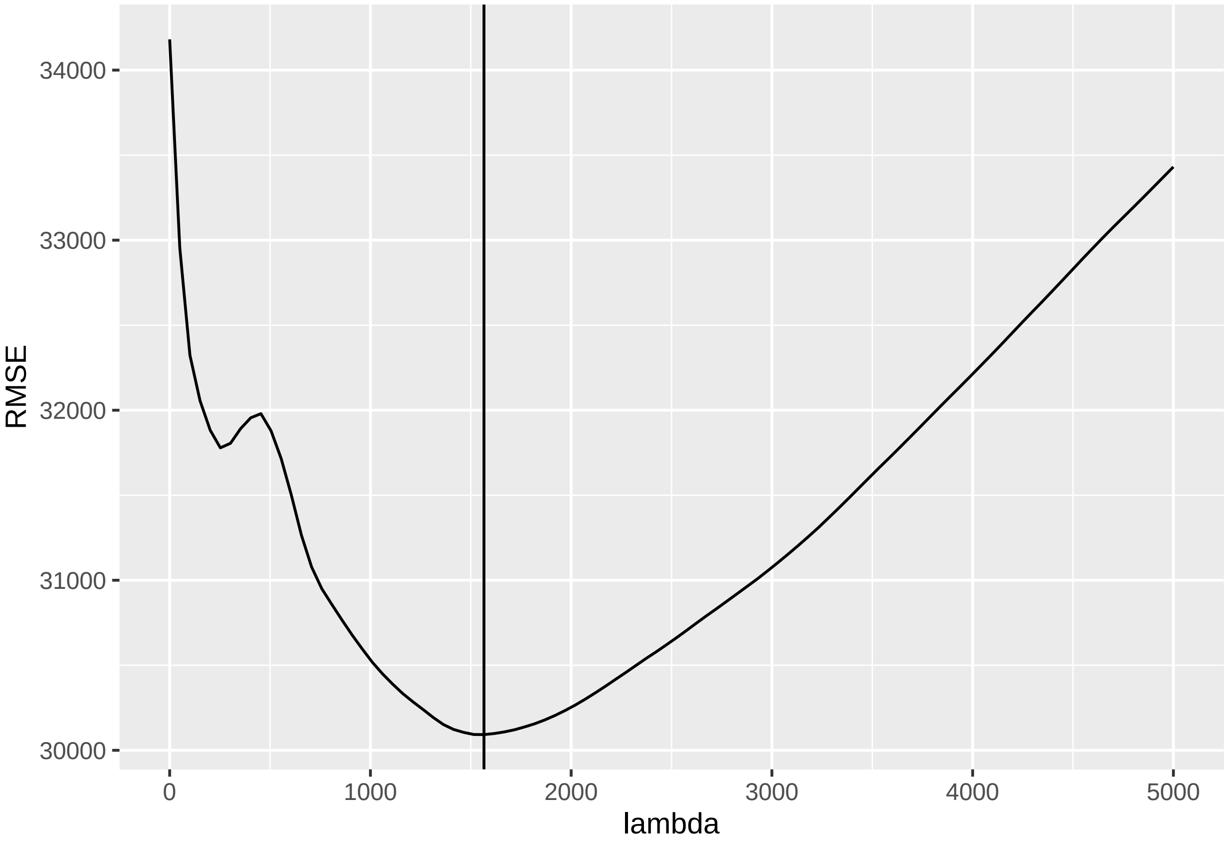
(b) Use ridge regression to fit a linear model using sale price as the response and all other covariates as predictors. You should do cross-validation to select the value of the penalty parameter lambda. Make a plot of the results of cross-validation. Make sure you explored a broad enough range of values for lambda in cross-validation to see a U-shape in the cross-validated RMSE. You don't need to implement cross-validation manually; you can have the train function perform cross-validation for you.

```r
ridge_fit <- train(
  form = Sale_Price ~ .,
  data = houses_train,
  method = "glmnet",
  trControl = trainControl(method = "cv"),
  tuneGrid = data.frame(
    alpha = 0,
    lambda = seq(from = 0, to = 50000, length = 100)
  )
)

ggplot(data = ridge_fit$results, mapping = aes(x = lambda, y = RMSE)) +
  geom_line() +
  geom_vline(xintercept = ridge_fit$bestTune$lambda)
```



(c) Use LASSO regression to fit a linear model using sale price as the response and all other covariates as predictors. You should do cross-validation to select the value of the penalty parameter lambda. Make a plot of the results of cross-validation. Make sure you explored a broad enough range of values for lambda in cross-validation to see a U-shape in the cross-validated RMSE. You don't need to implement cross-validation manually; you can have the train function perform cross-validation for you.

```
lasso_fit <- train(
  form = Sale_Price ~ .,
  data = houses_train,
  method = "glmnet", # method for fit
  trControl = trainControl(method = "cv"),
  tuneGrid = data.frame(
    alpha = 1,
    lambda = seq(from = 0, to = 5000, length = 100)
  )
)

ggplot(data = lasso_fit$results, mapping = aes(x = lambda, y = RMSE)) +
  geom_line() +
  geom_vline(xintercept = lasso_fit$bestTune$lambda)
```

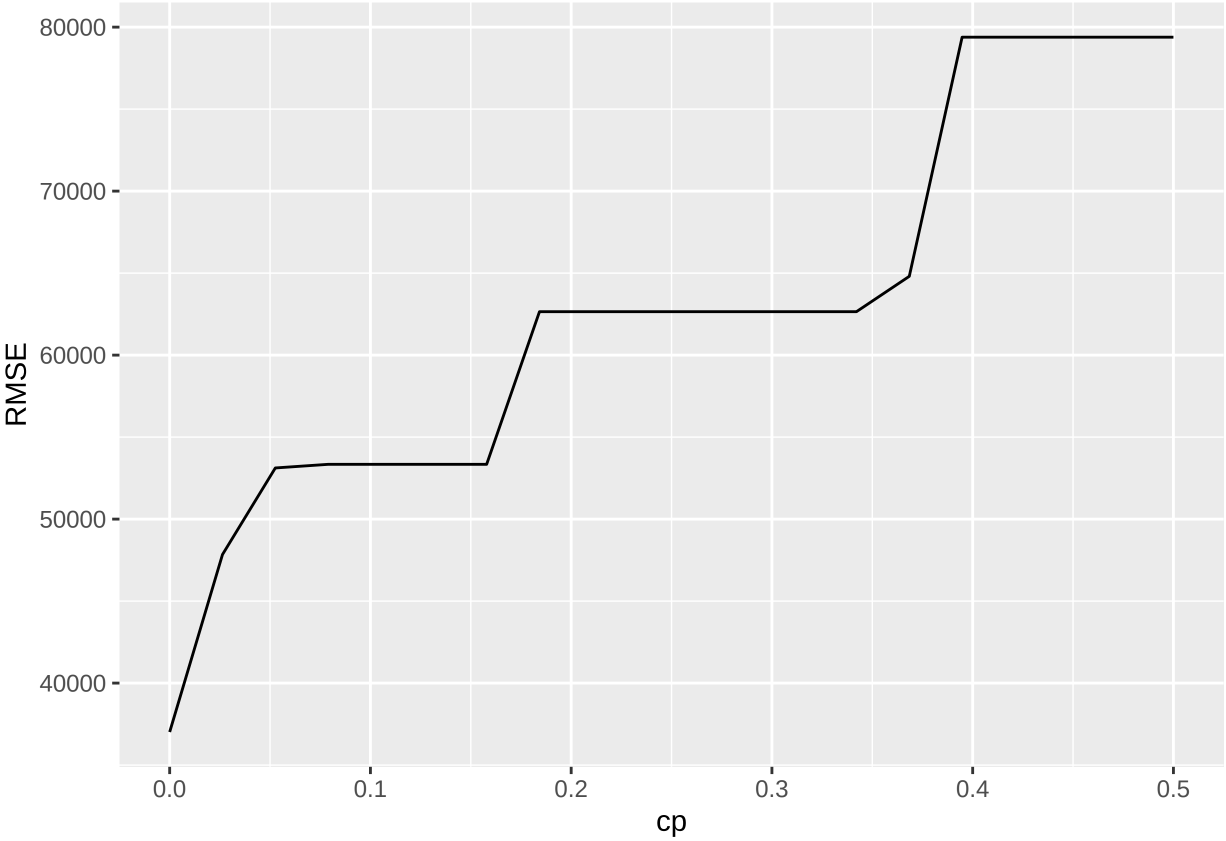


(d) Fit a regression tree using sale price as the response and all other covariates as predictors. You should do cross-validation to select the value of the penalty parameter cp. Make a plot of the results of cross-validation. Make sure you explored a broad enough range of values for cp in cross-validation to be confident you found a value of cp that yields the smallest possible cross-validated RMSE. You don't need to implement cross-validation manually; you can have the train function perform cross-validation for you.

```
tree_fit <- train(
  form = Sale_Price ~ .,
  data = houses_train,
  method = "rpart",
  trControl = trainControl(method = "cv"),
  tuneGrid = data.frame(
    cp = seq(from = 0, to = 0.5, length = 20)
  )
)
```

```
## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info =
## trainInfo, : There were missing values in resampled performance measures.
```

```
ggplot(data = tree_fit$results, mapping = aes(x = cp, y = RMSE)) +
  geom_line() +
  geom_vline(xintercept = tree_fit$bestTune$lambda)
```



(e) Find the test set MSE for each of the four models you fit in parts (a) through (d). Which model has best performance? Which has worst performance?

```
mean((houses_test$Sale_Price - predict(lm_fit, newdata = houses_test))^2)
```

```
## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient
## fit may be misleading

## [1] 1546143194
```

```
mean((houses_test$Sale_Price - predict(ridge_fit, newdata = houses_test))^2)
```

```
## [1] 870769775
```

```
mean((houses_test$Sale_Price - predict(lasso_fit, newdata = houses_test))^2)
```

```
## [1] 882483163
```

```
mean((houses_test$Sale_Price - predict(tree_fit, newdata = houses_test))^2)
```

```
## [1] 1428766807
```

The ridge regression approach had the best test set MSE, with LASSO close behind. The linear model estimated by least squares had the worst test set MSE.

Collaboration and Sources

If you worked with any other students on this assignment, please list their names here.

If you referred to any sources (including our text book), please list them here. No need to get into formal citation formats, just list the name of the book(s) you used or provide a link to any online resources you used.