Econ 5253 - Spring 2020 **Problem Set 9** **Due: Mar. 31 beginning of class**

This problem set will give you practice in using cross-validation to tune linear regression prediction models via LASSO, ridge regression, and elastic net.

The objective function of these models is:

$$\min_{\beta,\lambda,\alpha} \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 + \lambda\left(\alpha\sum_k |\beta_k| + (1-\alpha)\sum_k \beta_k^2\right)$$

where $\hat{y}_i = x_i'\beta$ and where λ and α are parameters that must be *tuned* using cross-validation techniques.

When $\alpha = 0$ we have the **ridge regression model**. When $\alpha = 1$ we have the **LASSO model**. When $\alpha \in (0,1)$ we have the **elastic net model**. Importantly, α must be in the closed interval $[0,1]$.

As with the previous problem sets, you will submit this problem set by pushing the document to *your* (private) fork of the class repository. You will put this and all other problem sets in the path `/DScourseS20/ProblemSets/PS9/` and name the file `PS9_LastName.*`. Your OSCER home directory and GitHub repository should be perfectly in sync, such that I should be able to find these materials by looking in either place. Your directory should contain at least three files:

- `PS9_LastName.R` (you can also do this in Python or Julia if you prefer, but I think it will be much more difficult to use either of those alternative for this problem set)
- `PS9_LastName.tex`
- `PS9_LastName.pdf`

1. Type `git pull origin master` from your OSCER `DScourseS20` folder to make sure your OSCER folder is synchronized with your GitHub repository.
2. Synchronize your fork with the class repository by doing a `git fetch upstream` and then merging the resulting branch. (`git merge upstream/master -m "commit message"`)
3. Install the following machine learning packages if you haven't already:
   - `mlr`
   - `glmnet`
4. Load the housing data from UCI, following the example in the lecture notes.
5. Add new features to the data set by creating 6th degree polynomials of each of the features, as well as 3rd degree interactions of each. To do so, add the following code to your script.[1] What is the dimension of your training data (`housing.train`)?

[1] Credit to Adam Kapelner in this GitHub issue for providing the following code.

```
housing$lmedv    <- log(housing$medv)
housing$medv     <- NULL # drop median value
formula    <- as.formula(lmedv ~ .^3 +
                         poly(crim, 6) +
                         poly(zn, 6) +
                         poly(indus, 6) +
                         poly(nox, 6) +
                         poly(rm, 6) +
                         poly(age, 6) +
                         poly(dis, 6) +
                         poly(rad, 6) +
                         poly(tax, 6) +
                         poly(ptratio, 6) +
                         poly(b, 6) +
                         poly(lstat, 6))
mod_matrix <- data.frame(model.matrix(formula, housing))
#now replace the intercept column by the response since MLR will do
#"y ~ ." and get the intercept by default
mod_matrix[, 1] = housing$lmedv
colnames(mod_matrix)[1] = "lmedv" #make sure to rename it otherwise MLR
    won't find it
head(mod_matrix) #just make sure everything is hunky-dory

# Break up the data:
n <- nrow(mod_matrix)
train <- sample(n, size = .8*n)
test  <- setdiff(1:n, train)
housing.train <- mod_matrix[train,]
housing.test  <- mod_matrix[test, ]
```

6. Following the example from the lecture notes, estimate a LASSO model to predict log median house value, where the penalty parameter λ is tuned by 6-fold cross validation. What is the optimal value of λ? What is the in-sample RMSE? What is the out-of-sample RMSE (i.e. the RMSE in the test data)?

7. Repeat the previous question, but now estimate a ridge regression model where again the penalty parameter λ is tuned by 6-fold CV. What is the optimal value of λ now? What is the in-sample RMSE? What is the out-of-sample RMSE (i.e. the RMSE in the test data)?

8. Repeat the previous question, but now estimate the elastic net model. In this case, you will need to use cross validation to tune the optimal λ and α (the relative weight on LASSO and ridge). What are the optimal values of λ and α after doing 6-fold cross

validation? What is the in-sample RMSE? What is the out-of-sample RMSE? Does the optimal value of α lead you to believe that you should use LASSO or ridge regression for this prediction task?

9. In your .tex file, answer the questions posed in the preceding four questions. Explain why you would not be able to estimate a simple linear regression model on the `housing.train` dataframe. Using the RMSE values of each of the tuned models in the previous three questions, comment on where your model stands in terms of the bias-variance tradeoff.

10. Compile your .tex file, download the PDF and .tex file, and transfer it to your cloned repository on OSCER using your SFTP client of choice (or via `scp` from your laptop terminal). You may also copy and paste your .tex file from your browser directly into your terminal via `nano` if you prefer, but you will need to use SFTP or `scp` to transer the PDF.[2]

11. You should turn in the following files: .tex, .pdf, and any additional scripts (e.g. .R, .py, or .jl) required to reproduce your work. Make sure that these files each have the correct naming convention (see top of this problem set for directions) and are located in the correct directory (i.e. `~/DScourseS20/ProblemSets/PS9`).

12. Synchronize your local git repository (in your OSCER home directory) with your GitHub fork by using the commands in Problem Set 2 (i.e. `git add`, `git commit -m "message"`, and `git push origin master`). Once you have done this, issue a `git pull` from the location of your other local git repository (e.g. on your personal computer). Verify that the PS9 files appear in the appropriate place in your other local repository.

[2] If you want to try out something different, you can compile your .tex file on OSCER by typing `pdflatex myfile.tex` at the command prompt of the appropriate directory. This will create the PDF directly on OSCER, removing the requirement to use SFTP or `scp` to move the file over.