WEEK 8 HOMEWORK – SAMPLE SOLUTIONS

IMPORTANT NOTE

These homework solutions show multiple approaches and some optional extensions for most of the questions in the assignment. You don't need to submit all this in your assignments; they're included here just to help you learn more – because remember, the main goal of the homework assignments, and of the entire course, is to help you learn as much as you can, and develop your analytics skills as much as possible!

Question 11.1

Using the crime data set from Questions 8.2, 9.1, and 10.1, build a regression model using:

1. *Stepwise regression*
2. *Lasso*
3. *Elastic net*

For Parts 2 and 3, remember to scale the data first – otherwise, the regression coefficients will be on different scales and the constraint won't have the desired effect.

For Parts 2 and 3, use the `glmnet` *function in R.*

Notes on R:

- *For the elastic net model, what we called λ in the videos,* `glmnet` *calls "alpha"; you can get a range of results by varying alpha from 1 (lasso) to 0 (ridge regression) [and, of course, other values of alpha in between].*
- *In a function call like* `glmnet(x,y,family="mgaussian",alpha=1)` *the predictors* `x` *need to be in R's matrix format, rather than data frame format. You can convert a data frame to a matrix using* `as.matrix` *– for example,* `x <- as.matrix(data[,1:n-1])`
- *Rather than specifying a value of T,* `glmnet` *returns models for a variety of values of T.*

Here's one possible solution. Please note that a good solution doesn't have to try all of the possibilities in the code; they're shown to help you learn, but they're not necessary.

The file `solution 11.1.R` shows one way of answering this question. It runs stepwise regression, lasso, and elastic net on both the scaled raw data and principal components found using PCA. For each model, the R code does three things: (1) uses the method to identify a set of variables to use, (2) builds a regression model using those variables, and (3) removes variables that are insignificant in the regression and then builds a regression using the remaining variables. After building each model, the code reports the R-squared value on the training data, and then uses cross-validation to estimate the real R-squared

value of the model. For the elastic net models, we tested 11 different values of alpha, from 0.0 to 1.0 at intervals of 0.1. Note that you might see slightly different results depending on the random number generator.

The table below shows all of the R-squared values. [Note that other quality measures could be used too; we're just using R-squared to show how the comparison works.]

Model	***Variables***	***Adj-R^2 (training data)***	***R^2 (cross-validation)***
Stepwise regression on original data (all variables)	Ed, Ineq, M, M.F, Po1, Prob, U1, U2	0.74	0.67
Stepwise regression on original data (significant variables)	Ed, Ineq, M, Po2, Prob, U2	0.73	0.67
Lasso regression on original data (all variables)	Ed, Ineq, M, M.F, NW, Po1, Prob, So, U2	0.72	0.62
Lasso regression on original data (significant variables)	Ed, Ineq, M, Po2, Prob, U2	0.73	0.67
Elastic net (alpha=1) on original data (all variables)	Ed, Ineq, M, M.F, NW, Po1, Po2, Pop, Prob, So, U1, U2, Wealth	0.72	0.57
Elastic net (alpha=1) on original data (significant variables)	Ed, Ineq, M, Po2, Prob, U2	0.73	0.67
Stepwise regression on PCA data (all variables)	PC1, PC2, PC4, PC5, PC6, PC7, PC12, PC14, PC15	0.73	0.63
Stepwise regression on PCA data (significant variables)	PC1, PC2, PC4, PC5, PC7, PC12, PC14	0.71	0.63
Lasso regression on PCA data (all variables)	PC1, PC2, PC3, PC4, PC5, PC6, PC7, PC10, PC12, PC13, PC14, PC15	0.73	0.59
Lasso regression on PCA data (significant variables)	PC1, PC2, PC4, PC5, PC7, PC12, PC14	0.71	0.63
Elastic net (alpha=0.3) on PCA data (all variables)	PC1, PC2, PC3, PC4, PC5, PC6, PC7, PC12, PC14, PC15	0.73	0.63
Elastic net (alpha=0.3) on PCA data (significant variables)	PC1, PC2, PC4, PC5, PC7, PC12, PC14	0.71	0.63

There are a few interesting observations here.

First, notice that *all* of these models appear to be significantly better than what we found in the previous homework questions just using regression (either on the original variables or the PCs), regression trees, or random forests. So variable selection seems to make a big difference. Why? Part of the difference might be because of the small number of data points (only about three times the number of variables). So it's very easy for models to be overfit, and selecting a smaller subset of variables is important. For example, consider the full regression model on the original data: on training data its R^2 looked like 0.80, but cross-validation estimated an R^2 of 0.41. The variable-selection models did much better.

Second, notice (again) that it can be important to remove the variables that seem insignificant in a regression model. In several cases, doing that improved the model quality.

And third, in this case using PCA didn't seem to be beneficial – but in some cases, you'll probably find it to be very valuable.

More generally, by now you've used a lot of different modeling approaches on this one data set. Some approaches have worked better than others – but please don't think that what happened with this data set is going to happen with all others. There are some data sets where models that worked well here won't fare well, and other data sets where they will. It's often valuable to test a variety of approaches (and then use a validation data set and/or cross-validation to compare them), because it's often unclear up front which method will work best.