Chs. 4 Extra Exercises: Part 2

February 25, 2019

1. On pg. 90 of CH 4 in *Statistical Rethinking*, the book uses a function `cov2cor()` to convert a covariance matrix into a correlation matrix. Write your own function which does the same and write out the computations using matrix algebra notation.

Solution:

```
my_cov2cor <- function(cov_mat){
  diag <- diag(cov_mat)
  diag_inv_sqrt <- (diag)^(-1/2)
  corr_mat <- diag_inv_sqrt * cov_mat * rep(diag_inv_sqrt, each = dim(cov_mat)[1L])

  return(corr_mat)
}
```

Now let's compare the output of this function to the built-in `cov2cor()` function:

```r
#Test run using data from Ch. 4 pg. 87-90
data(Howell1)
d <- Howell1
d2 <- d[d$age >= 18,]

flist <- alist(
  height ~ dnorm(mu, sigma),
  mu ~ dnorm(178, 20),
  sigma ~ dunif(0, 50)
)

m4.1 <- map(flist, data=d2)

cov2cor(vcov(m4.1))

##                mu       sigma
## mu    1.000000000 0.001816966
## sigma 0.001816966 1.000000000

my_cov2cor(vcov(m4.1))

##                mu       sigma
## mu    1.000000000 0.001816966
## sigma 0.001816966 1.000000000
```

2. On pg. 92 of CH 4 in *Statistical Rethinking*, the author talks about regression to the mean and shrinkage. Read more about shrinkage estimation (including those articles I gave you last week), to explain more about the purpose and benefit of using shrinkage estimation. Include citations for any references you use.

Solution:

Shrinkage estimation generallly reduces the variance in parameter estimation. This may serve to improve out-of-sample prediction by reducing the variance of the model while increasing the bias. If the "correct" parameters for a particular shrinkage method are chosen, then the trade-off between variance and bias should result in a model with better predictive capability when compared with an OLS model.

Ridge regression, one such shrinkage method, is very similar to OLS regression. It includes a penalty term with the RSS calculation which penalizes the model for having large β values. The parameter $\lambda \geq 0$ controls the size of the penalty, with $\lambda = 0$ simplifying the model to OLS regression. The penalty term shrinks the β terms towards 0 (but never reaching 0).

Lasso regression is another shrinkage method which decreases the variance of the β values. Lasso minimizes the RSS, but includes the constraint that $\sum_{j=1}^{p} |\beta_j| \leq t$. If t is chosen sufficiently small, then some of the β values will be reduced all the way to 0. In this manner, lasso regression is able to perform a sort of continuous subset selection.

Cite later:

- Introduction to Statistical Learning
- Elements of Statistical Learning
- Regression Modeling Strategies
- Brown (2007)
- Brown (2008)
- Copas (1983)

3. On pg. 99 of CH 4 in *Statistical Rethinking*, the author writes, "But in more complex models, strong [parameter] correlations like this can make it difficult to fit the model to the data."
   (a) Explain why correlations between pairs of parameters is a problem.
   (b) Why does the author only center the x-variable, `weight`, and not the y-variable `height`?

Solution:

(a)

4. Use the `d2` data from the chapter to answer the following questions, which are extensions of **4H2**:
   (a) Fit a frequentist simple linear regression with `weight` as the explanatory variable and `height` as the response variable. Make a scatterplot of the data and plot both the frequentist and Bayesian lines. Is there much of a difference?
   (b) Check the regression assumptions on your frequentist model. Include any relevant tables/graphs with your assumption checks.
   (c) Are correlations between pairs of parameters a problem in frequentist regression? Calculate thevariance-covariance matrix for $\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \hat{\beta}_1)$ using the formulas in the appendix to CH 3 of *Regression Analysis by Example* (your regression textbook).

(d) Does centering `weight` make a difference in the correlations between parameters in frequentist regression? Does it make a difference in the parameter estimates themselves? Try using the function `scale()` to center `weight` in your code.

(e) Use the function `confint()` to create a 95% confidence interval of the slope for your frequentist regression. Calculate a 95% posterior probability interval for the slope for your Bayesian regression. Interpret both and explain the difference between the two.

(f) Create a figure with 4 scatterplots, each with `weight` on the x-axis and `height` on the y-axis. On the first plot, add the Bayesian fitted model and add the 95% HPDI intervals (like Fig. 4.8). On the second plot, add the Bayesian fitted model and add the 95% PI intervals (also like Fig. 4.8). On the third plot, add the frequentist fitted model and the 95% confidence intervals for prediction (like slide 35 in Lecture 3 of the regression class). Finally, on the fourth plot, add the frequentist fitted model and the 95% prediction intervals for prediction (also like slide 35 in Lecture 3 of the regression class). Interpret all four plots and explain the connections between them.