Lab 5: Using Amelia to analyze missing data

This lab will cover the analysis of missing data using the ncbirths dataset in the openintro package. This dataset contains data on 1000 births in the state of North Carolina in 2004. Data on 13 variables is available, including the weight of the baby at birth, and various information about the baby and the parents.

Exercise 1 Load (install if necessary) the following packages.

```r
library(Amelia)
library(Zelig)
library(openintro)
library(ggplot2)
```

You can find out more information about the ncbirths dataset by typing **?ncbirths**.

Exercise 2 Begin by loading the data and looking at patterns of missingness. How many variables have missing data? Which have the most missing data?

```r
data(ncbirths)
missmap(ncbirths)
```

Exercise 3 Do some exploratory plots of the data to see what relationships might be interesting to include in a model. Do other features of the data stand out to you?

The Amelia R package runs a form of the bootstrap-based EM algorithm (a.k.a. the "EMB" algorithm) to perform multiple imputation (for more see Honaker and King (2010).)

Exercise 4 Without doing any modeling on the data yet, let's begin by running a multiple imputation on the ncbirths dataset and plotting the results. Notice that here we specify that seven of the variables are "nominal" variables (a.k.a. unordered factors). We specify 10 replicates, but you may choose more or less. In the resulting figure, the red lines indicate the smoothed density estimate of the average of each imputed observation and the black lines show the smoothed density estimate of the observed data. Do the imputations seem on average to be reasonable?

```r
imp_ncb <- amelia(x = ncbirths, m = 10,
                  noms = c("mature", "premie", "lowbirthweight",
                           "marital", "gender", "habit", "whitemom"))
plot(imp_ncb)
```

Exercise 5 Now, choose one of the m imputed datasets, and plot the imputed observations for the 'gained' variable on top of the density of observed data. What does the 'gained' variable represent? Do negative values make sense for this variable? Try inputting several different values for i. Do you see any negative imputed values? [Note: there are probably nicer or simpler ways of making the plots than the code below, but this does work... Feel free to write up some nicer code.]

```
i=1 ## choose the i^th dataset
one_imp <- imp_ncb$imputations[[i]]$gained # stores the imputed values
obs_data <- ncbirths$gained # stores the observed data

hist(obs_data, prob=TRUE, breaks=20, xlim=c(-20, 90))
lines(density(obs_data[!is.na(obs_data)]))
rug(one_imp[is.na(obs_data)], col="red", ticksize=.2)
```

Exercise 6 Regardless of whether you saw any negative imputations, you could argue that since there are no negative observed values for 'gained' and since a pregnant woman should not lose weight overall, we should not allow for the possibilitiy for there to be imputed negative values. We can modify our call to Amelia to correct this by adding the "logs" argument to the function call. This means that the imputation for the "gained" variable is done on the log scale. Run the individual imputation plots from the previous exercise. Do you see any negative imputed values of "gained"?

```
imp_ncb1 <- amelia(x = ncbirths, m = 10,
                   noms = c("mature", "premie", "lowbirthweight",
                            "marital", "gender", "habit", "whitemom"),
                   logs="gained")
plot(imp_ncb1)
```

A nice feature of the Amelia package is that it has implemented a technique to show you how well your dataset can be used to impute different variables. The technique is explained in more detail in the Amelia vignette, but the main jist is that you run a small simulation where you go through your dataset and for each observed datapoint, you pretend to leave it out and then attempt to impute it repeatedly. Since you know the true value, you can test how well you can impute that value. This routine is called using the `overimpute()` function. For the graphs generated, if the points lie near and the corresponding confidence intervals (generated using the multiple imputations for each missing datapoint) cross the the $y = x$ line, then the imputation model is able to predict the hypothetical missing values well. If the CIs do not cross the $y = x$ line then the imputation may not be providing that much additional information for the variable chosen.

Exercise 7 Run the `overimpute` routine for the gained, fage, and visits variables. Does the imputation perform well for each of the variables? For ones where it seems to perform better, other variables in the dataset must be informing Amelia's guesses of the missing values. For these variables, which other variables in the dataset do you think are informing the imputation process?

```
overimpute(imp_ncb, var = "gained")
overimpute(imp_ncb, var = "fage")
overimpute(imp_ncb, var = "visits")
```

Even if we are not making terribly good imputations on some variables, it still allows us to keep that observation in the dataset for a given model. Note that we can use the `Zelig` package to fit a regression model to the imputed data. For a regression model, this automatically does the multiple imputation estimation.

Exercise 8 Run both a linear regression on the dataset with missing values and then on the multiply imputed datasets. Summarize the results in a table, compare the coefficient estimates. Are there any large differences between the two analyses?

```r
m1 <- lm(weight ~ fage + mage + weeks + visits + marital + gained +
             gender + habit + whitemom, data = ncbirths)
m2 <- zelig(weight ~ fage + mage + weeks + visits + marital +
                gained + gender + habit + whitemom,
            data = imp_ncb, model = "ls")
```

Exercise 9 Interpret the results of the regression. Also, check the usual model diagnostics. Does the model appear to meet the typical regression assumptions?

Exercise 10 **Extra Credit:** Plot a histogram of the `weight` variable. Does it appear to follow a normal distribution? Do the residuals from the models seem to follow the assumptions of linear regression? Recall that if the outcome does not appear to be normal, you can use the Box-Cox method to identify an appropriate transformation of the response variable (more details in Faraway, Chapter 8.1). What are two reasons to be cautious about transforming the response variable? Use the `boxcox()` function to determine an appropriate transformation of the response variable. Re-run your analysis above with the transformed response. Do your results change in a substantive way?

```r
boxcox(m1, plotit=T)
bc <- boxcox(m1, plotit=T, lambda=seq(1.1, 1.6, by=.01))
lam <- with(bc, x[which(y==max(y))])
```

For this lab, you should submit a single PDF file that summarizes your findings and answers to these questions.