Midterm Project

Matthew Tichenor

October 27, 2015

This paper is divided into 5 main sections. First is an introduction which describes the dataset and chooses a response variable and full model. Second is the complete-case analysis, which analyzes only the observations which do not have missing values. The third section focuses on imputation, specifically multiple-imputation through the MICE algorithm. Since the MICE algorithm wasn't introduced in class, this section has several subsections. Following this the results from the imputed data are discussed. Lastly, the R code used for this project is displayed before a list of references.

1 Introduction

I picked the Mammals dataset, which was used in the article "Sleep in Mammals: Ecological and Constitutional Correlations"[2]. The dataset includes 62 observations from mammals. The variables are species of animal, bodyweight in kg, brain weight in g, slow wave, i.e. non-dreaming, sleep in hrs per day, paradoxical, i.e. dreaming, sleep in hrs per day, total sleep in hrs per day, life span in yrs, gestation time in days, and three categorical variables: predation index, sleep exposure index and overall danger index.

Using this dataset, one possible model has total sleep as the response variable, and the rest of the variables, excluding species, as an explanatory variable. To make the model simpler, the 4 observations where total sleep had missing values as well as the categorical variables were removed from the dataset.

Before moving forward, it is important to see whether or not a linear model is sufficient. In other words, we need to check whether or not there is a linear relationship between the response variable, total sleep, and the explanatory variables, or a linear relationship between the response variable and a transformation of the explanatory variables.

Figure 1 shows scatter plots for the response variable vs. (a transformation of) explanatory variables. With the exception of a few outlying observations, there is a linear relationship between the response variable and (a transformation of) the explanatory variable for all the variables of interest.

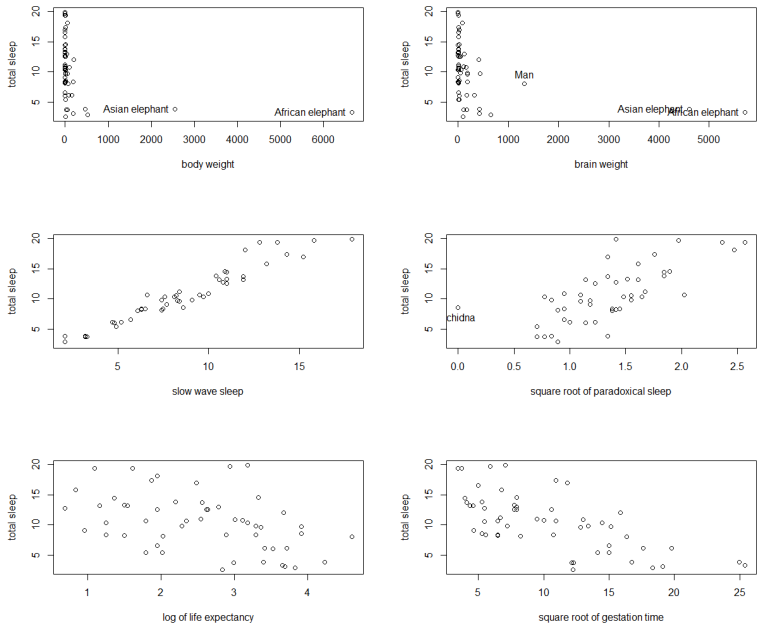

Figure 1: Scatter Plots of Response Var vs Explanatory Var

Using the transformations which give a linear relationship in the scatter plots, the model of interest is

$$totsleep_i = \beta_0 + \beta_1 bodywt_i + \beta_2 brainwt_i + \beta_3 slwave_i + \beta_4 \sqrt{pdsleep_i} + \beta_5 log(life_i) + \beta_6 \sqrt{gest_i} + \varepsilon_i$$

for $i = 1, ..., 58$, where the explanatory variables are taken as constants, ε_i is normally distributed with mean 0 and variance σ^2.

Although removing the categorical variables makes the model simpler, the large number of missing values makes up for this lack of complexity. The dataset has 20 observations with missing values (including the 4 that were removed). In the next section, all 20 missing observations are removed from the dataset, and the analysis continues; this is called the complete case analysis.

2 Complete Case Analysis

While 20 observations were removed from the dataset, this alone does not guarantee that there are no outliers remaining in the dataset; and it certainly does not guarantee that the other assumptions, which are needed for multiple linear regression to make sense, are satisfied.

Figure 2 displays the diagnostic plots needed to check that the linear regression assumptions have not been violated. In the top left pane, the residuals are plotted against the fitted values. The dashed line is the line $y = 0$, and the solid

red line is a smoothed line that goes through the center of the points. Besides observations 16, 39 and 61, these residuals are symmetric about 0. There also doesn't appear to be any pattern in this plot that would suggest a non-constant variance.

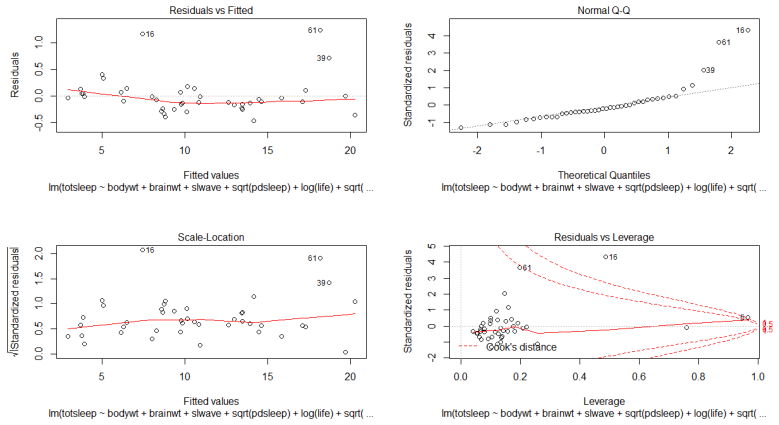

Figure 2: Diagnostic Plots

In the top right pane of Figure 2 is a Normal qq plot of the residuals. This plot is used to check if the residuals are not normally distributed. Again, observations 16, 39 and 61 are causing some problems with these assumptions. If these observations were to be removed, then ans assumption of normality would be easier to believe. The Normality assumption is needed for the hypothesis tests in later analysis.

In the bottom left pane of Figure 2 the square root of the standardized residuals are plotted against the fitted values. Just as in the top left pane, the red line is a smoothed line going through the center of the data points. The standardized residuals should have a variance of 1. Note that this plot eliminates the sign of the residual, i.e. negative values appear as positive values. Yet again, the observations 16, 39 and 61 are causing problems with a diagnostic plot.

In the bottom right pain of Figure 2 standardized residuals are plotted against leverage. Like the other two plots, the red line is a smoothed line that passes through the center of the data points, but there is an additional feature of this plot. The dotted red lines are contours representing the Cook's distance. The standardized residuals are the same as before, and the leverage is a measure of how much influence an observation has on the regression line. Points that have a high leverage or points that lie near those contours are likely outliers. Observations 16 and 61 are shown to be outliers, but this time a new observation, 5, is shown to be a likely outlier.

To satisfy assumptions and continue the analysis using multiple linear regression, observations 5, 16, 39 and 61 were removed from the dataset.

After fitting a model again using multiple linear regression, we find there is one

more outlier in the dataset, by examining the diagnostic plots one more time. This outlier is revealed by the relatively large cook's distance in Figure 3.

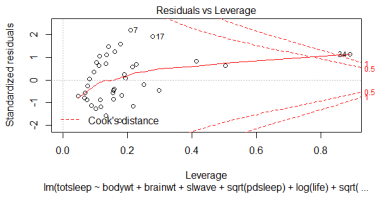

Figure 3: Outlier Found Using Cook's Distance

After removing this observation, the diagnostic plots make our assumptions needed more believable. These plots, shown in Figure 4, are the same plots as in Figure 1, except that any observation, that was suspected of being an outlier, had been removed.

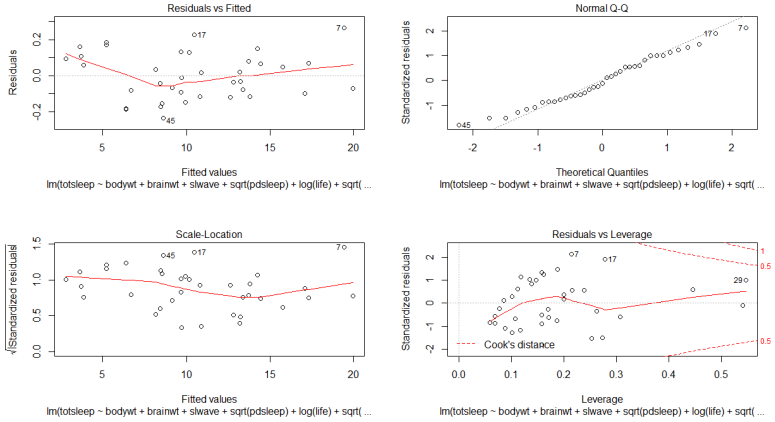

Figure 4: Diagnostic Plots Without Outliers

Comfortable with the assumptions needed to fit the model and perform hypothesis tests, I performed multiple linear regression using the model: $totsleep_i = \beta_0 + \beta_1 bodywt_i + \beta_2 brainwt_i + \beta_3 slwave_i + \beta_4\sqrt{pdsleep_i} + \beta_5 log(life_i) + \beta_6\sqrt{gest_i} + \varepsilon_i$, for $i = 1, \ldots, n$. Table 1 summarizes the results of 7 t-tests. It appears that the variables $bodywt$, $log(life)$ and $sqrt(gest)$ are not significant. The variable $brainwt$ is not significant at the 5% significance level, but with a p-value of 0.06, this variable could be included in a model.

It may seem reasonable to preform further tests for model selection, however a large portion of the data was removed. The original dataset had 62 observations. After removing outliers and observations with missing data, the dataset had only 37 variables. Approximately 40% of the data was removed, so it is unlikely that any model selected, using 37 observations would reasonably predict the total sleep for a given mammal.

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-1.7478	0.1663	-10.51	0.0000
bodywt	0.0002	0.0004	0.64	0.5271
brainwt	-0.0007	0.0004	-1.92	0.0639
slwave	0.9814	0.0088	111.90	0.0000
sqrt(pdsleep)	2.7586	0.0860	32.08	0.0000
log(life)	0.0638	0.0402	1.58	0.1236
sqrt(gest)	0.0068	0.0091	0.75	0.4610

Table 1: Regression Summary

To fix this problem, the missing values of the data can be filled with reasonable approximations. This is called imputation, which is discussed in the next section.

3 Imputation

There are several methods for handling missing values. In the last section, the observations containing missing values were deleted; this procedure is known as complete case analysis. When missing values are filled in only one dataset, the procedure is a single imputation. One simple, single imputation would be filling in a variable's missing values with the average. This imputation has an advantage that the variable's mean will be unchanged by the imputation, but a disadvantage is that this method reduces any correlation between variables[10]. A more compicated method is to form another regression model between the variable with missing values and the other values, then fill in the missing values with the fitted values from the regression. The problem with this method is that the fitted value ignores the error term. This means the missing value is filled in with a likely value, but we can say nothing about our uncertainty about this value.

Multiple imputation addresses the problem of increased uncertainty due to imputation. Figure 5 is a schematic of multiple imputation. The first step in multiple imputation is to fill in the missing values, i.e. impute the dataset, m times. In Figure 5, the dataset is imputed 3 times. In the second step, parameters are estimated for each of the m datasets. The third and final step of multiple imputation pools the parameter estimates from the imputed datasets.

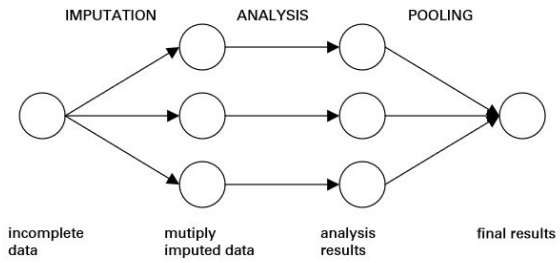

Figure 5: Multiple Imputation Diagram

Common to multiple imputed methods is an assumption regarding the type of missingness. It must be assumed that the missing values are missing at random (MAR).

3.1 MAR Assumption

The MAR assumption states that the missingness of a variable does not depend on the variable. This definition allows the missingness of variable to be conditioned on other variables. As an example suppose that we have observations from two random variables Z and X, where Z has some missing values and X has no missing values. Define R_Z to be an indicator vector, where $R_Z = 1$ if we have a missing value and 0 otherwise. The MAR assumption is really saying that $P(R_Z = 1|Z, X) = P(R_Z = 1|X)$[1]. When there are more variables with missing values, the previous statement changes, but the basic idea remains the same. If the data is MAR, the missingness of a variable does not depend on the variable itself.

Testing for MAR is difficult if not impossible[11]. I assume that the data is MAR, to use an imputation method and pool the results.

The method of imputation used in this analysis is Multivariate Imputation by Chained Equations (MICE). MICE is one multiple imputation method which follows Rubin's Rules of multiple imputation. The R package is maintained and was developed by Stef Van Buuren[1].

3.2 The MICE Algorithm

The MICE method follows the steps[7] in Figure 5, where the most complicated part is the first step, generating m imputed datasets. The default method of imputing numeric data, is fully conditional specification.

In fully conditional specification (FCS), the data is imputed with some intial values. For each variable with missing values, the sample mean and standard deviation are calculated using the observed values, then draws from a normal

[1]For more information on the package, see the user manual[8]

distribution with mean and variance equal to the sample mean and sample variance are used to fill in the value. Also, for each variable with missing values, the draws are limited to the minimum and maximum of the observed values. FCS only begins at this step.

FCS is an iterative Monte-Carlo Markov Chain (MCMC) process. Let $X = (X_1, \ldots, X_p)$ denote a set of p random variables. Each variable $X_j = (X_j^{mis}, X_j^{obs})$ maybe partially observed. For each iteration, $i = 1, \ldots, k$ draw imputation X_1^{i+1} from $P(X_1|X_2^i, X_3^i, \ldots, X_p^i)$, then draw imputation X_2^{i+1} from $P(X_2|X_1^{i+1}, X_3^i, \ldots, X_p^i)$, and so forth drawing the last imputation X_p^{i+1} from $P(X_p|X_1^{i+1}, \ldots, X_{p-1}^{i+1})$[3]. This process is then repeated for each of the m imputed datasets. The default number of iterations is 5[2], which was used in this analysis.

A Gibbs Sampler generates the observations from the conditional distributions, where the observations are approximations[9]. The Gibbs Sampler does not aim for convergence to a particular value, but rather a convergence to a distribution[7]. The MICE package[8] includes a plot function to monitor convergence.

3.3 Checking Convergence

For each iteration and each imputation, missing values are imputed, but the sample mean and standard deviation of the imputed missing values are also calculated. Let $Y_j^{mis(m,l)}$ denote the vector of missing variables for variable Y_j at iteration l for dataset m. The values $m_j^{(m,l)} = mean(Y_j^{mis(m,l)})$ and $s_j^{(m,l)} = \sqrt{Var(Y_j^{mis(m,l)})}$ are the mean and variance of the vector of missing values[3], and it is these values that are used to monitor convergence in distribution.

[2]Five iterations may seem like a small number, but a simulation study has shown that five is adequate[7]

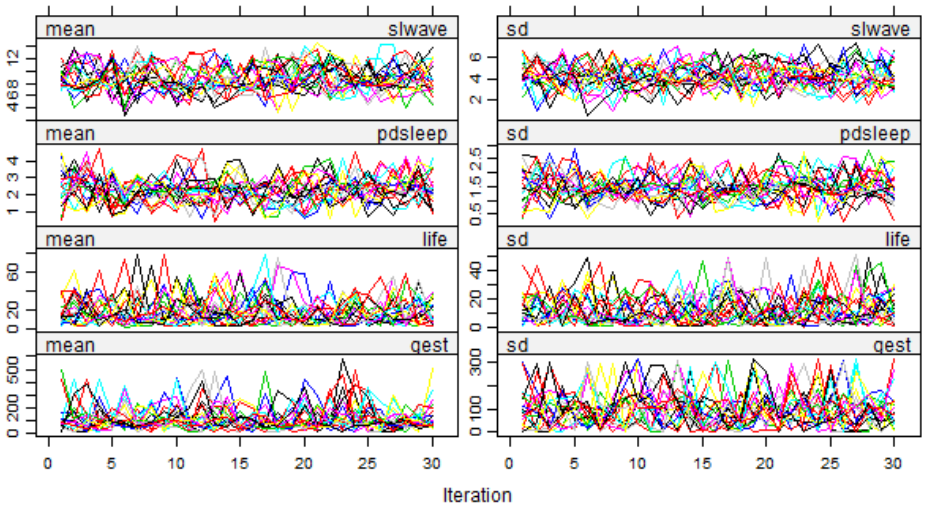

Figure 6: Checking for Convergence in Distribution

Figure 6 displays plots of $m_j^{(m,l)}$ vs. number of iterations and $s_j^{(m,l)}$ vs. number of iterations, for each imputed dataset and each variable. To ensure valid estimates, the data was imputed for 30 iterations for 20 imputed datasets. The top left figure shows $m^{(m,l)}$ for the variable *slwave*. Each colored line represents a sequence of $m^{(m,l)}$ for one particular imputed dataset. There are 20 imputed datasets, hence 20 different sequences plotted. Convergence is diagnosed when the variance between different sequences is no larger than the variance with each individual sequence. The sequences should overlap and be free of trend. [7]. It appears that this is the case for this plot as well as all the other plots in Figure 6. The top right pane plots the sequences of $s^{(m,l)}$ for the 20 imputed datasets for the variable *slwave*. All left panes plot means, and all right panes plot standard deviations. In order from top to bottom the variables are $slwave, pdsleep, life$ and $gest$.

3.4 Pooling the Estimates

With the convergence checked, the remaining tasks are to fit each imputed dataset with a linear model, then combine the results a vector of estimates $\hat{\beta}$. The model $totsleep_i = \beta_0 + \beta_1 bodywt_i + \beta_2 brainwt_i + \beta_3 slwave_i + \beta_4\sqrt{pdsleep_i} + \beta_5 log(life_i) + \beta_6\sqrt{gest_i} + \varepsilon_i$, for $i = 1, \ldots, n$ was used in this step.

Since there are 20 imputed data sets, we have 20 $\hat{\beta}^{(i)}$'s, where each $\hat{\beta}^{(i)} = (\hat{\beta}_1^{(i)}, \ldots, \hat{\beta}_6^{(i)})$. Also there are 20 $\hat{V}^{(i)}$'s, where $\hat{V}^{(i)}$ is the estimated variance-covariance matrix for $\hat{\beta}^{(i)}$. These vector and matrix estimates are combined using Rubin's Rules. Let m be the number of imputed datasets, let $\bar{\beta} = \frac{1}{m}\sum \hat{\beta}^{(i)}$, let $B = \frac{1}{m-1}\sum(\hat{\beta}^{(i)} - \bar{\beta})(\hat{\beta}^{(i)} - \bar{\beta})^T$, let $\bar{V} = \frac{1}{m}\sum \hat{V}^{(i)}$ and let $T = \bar{V} + (1 + \frac{1}{m})B$. In T the terms B and $\bar{V}$ represent the between imputation and average within

variances respectively. By Rubin's Rules $\bar{\beta}$ is our estimate of β and T is the variance of our estimate[4].

The Mice package[8] has functions to fit a model for each dataset as well as combine these estimates following Rubin's Rules.

The t-test used following Rubin's Rules is not very different from the t-test used in the complete-case analysis. Let $se_j = \sqrt{T_j j}$ be the standard error for our estimate of a particular β_j, $\bar{\beta}_j$. The corresponding t-test statistic is $\frac{\bar{\beta}_j}{se_j} \sim t_{v_j}$.

3.5 Results

The results of the pooling and t-tests are displayed in Table 2. The meaning behind the column names aren't particular hard to figure out, except for fmi and $lambda$. fmi is the fraction of missing information for a variable. $lambda$ is the proportion of the variation attributable to the missing data. Particularly important is the column $Pr(>|t|)$, where each row is the p-value for the t-test statistic for a variable. At the 5% siginificance level, the only significant variables are $slwave$ and $\sqrt{pdsleep}$.

	est	se	t	df	Pr(>|t|)	lo 95	hi 95	nmis	fmi	lambda
(Intercept)	0.85	2.59	0.33	9.39	0.75	-4.97	6.68		0.78	0.74
bodywt	-0.00	0.00	-0.69	12.10	0.50	-0.00	0.00	0.00	0.70	0.66
brainwt	0.00	0.00	0.43	25.02	0.67	-0.00	0.00	0.00	0.42	0.37
slwave	0.79	0.25	3.22	3.31	0.04	0.05	1.53	10.00	0.95	0.92
sqrt(pdsleep)	2.99	1.23	2.43	6.75	0.05	0.06	5.92		0.86	0.82
log(life)	0.11	0.48	0.24	14.28	0.82	-0.92	1.14		0.65	0.60
sqrt(gest)	-0.13	0.12	-1.09	10.72	0.30	-0.41	0.14		0.74	0.70

Table 2: Pooling Summary

It appears that a better model may be $totsleep_i \sim \beta_0 + \beta_1 slwave_i + \beta_2 \sqrt{pdsleep} + \epsilon_i$ for $i = 1, \ldots, n$. In model selection this new model would be called a nested model, since every term in the new model appears in the old model. Fortunately there is a test to compare linear models. Meng and Rubin[6] proposed a Wald-method test in comparing different models. The null hypothesis for the test is that the nested model is the correct one[5]. The MICE package[8] contains a function for this test, and the p-value is shown below.

```
$pvalue
          [,1]
[1,] 0.4730809
```

Since the p-value is reasonably high, the null hypothesis should not be rejected, thus the nested model is the correct one.

4 Discussion

We already have doubts about the complete-case analysis, and with the imputed data, it seems that the correct model is $totsleep_i \sim slwave_i + \sqrt{pdsleep}_i + \varepsilon_i$ for $i = 1, \ldots, n$. The natural question is, does this model tell us anything particularly useful? The answer is no. If we refer to the paper associated with the dataset[2], we would see that $totlseep = slwave + pdsleep$, i.e. total sleep is the sum of slow wave sleep and paradoxical sleep[3]. I feel that I have used very complicated methods to verify a result that should be common sense.

5 R Code

The R code for this assignment is presented below:

```
df <- read.csv("data/sleep.csv")
df
df <- df[-c(21,31,41,62),]
df <- df[,-c(9,10,11)]
names(df)
plot(df$bodywt, df$totsleep, xlab="body weight", ylab="total sleep")
identify(df$bodywt, df$totsleep, labels=df$species, pos=T)
plot(df$brainwt, df$totsleep, xlab="brain weight", ylab="total sleep")
identify(df$brainwt, df$totsleep, labels=df$species, pos=T)
plot(df$slwave, df$totsleep, xlab="slow wave sleep", ylab="total sleep")
plot(sqrt(df$pdsleep), df$totsleep, xlab="square root of paradoxical sleep", ylab="total s
identify(sqrt(df$pdsleep), df$totsleep, labels=df$species, pos=T)
plot(log(df$life), df$totsleep, xlab="log of life expectancy", ylab="total sleep)
plot(sqrt(df$gest), df$totsleep, xlab="square root of gestation time", ylab="total sleep)
mm <- lm(totsleep~bodywt+brainwt+slwave+sqrt(pdsleep)+log(life)+sqrt(gest),data=df)
plot(mm)
origdata <- read.csv("data/sleep.csv")
df2 <- origdata[-c(5,16,21,31,39,41,61,62),-c(9,10,11)]
mm2 <- lm(totsleep~bodywt+brainwt+slwave+sqrt(pdsleep)+log(life)+sqrt(gest),data=df2)
plot(mm2)
df3 <- origdata[-c(5,16,21,31,34,41,61,62),-c(9,10,11)]
mm3 <- lm(totsleep~bodywt+brainwt+slwave+sqrt(pdsleep)+log(life)+sqrt(gest),data=df3)
plot(mm3)
summary(mm3)
library(xtable)
xtable(summary(mm3))
library(mice)
imp20 <- mice(df,m=20,maxit=30)
plot(imp20, layout=c(2,4))
fit20 <- lm.mids(totsleep ~ slwave + sqrt(pdsleep) + bodywt + brainwt + log(life) + sqrt(g
end <- pool(fit20)
```

[3] While it is not true that your total amount of sleep is the sum of slow wave and paradoxical sleep[12], the authors of the dataset defined *total sleep* to be the sum of those two variables.

```
summary(end)
xtable(summary(end))
altfit20 <- lm.mids(totsleep ~ slwave + sqrt(pdsleep), imp20)
pool.compare <- (fit20, altfit20, method="Wald")
```

References

[1] Paul D. Allison. Design and inference. http://statisticalhorizons.com/wp-content/uploads/2012/01/Milsap-Allison.pdf, 2012.

[2] Truett Allison and Domenic V. Cicchetti. Sleep in mammals: Ecological and constitutional correlates. *Science*, (4266):732, 1976.

[3] IBM. Ibm knowledge center: Multiple imputation algorithms. http://www-01.ibm.com/support/knowledgecenter/SSLVMB_21.0.0/com.ibm.spss.statistics.help/alg_multiple_imputation_multivariate_fcs.htm?lang=en.

[4] IBM. Ibm knowledge center: Rubin's rules. http://www-01.ibm.com/support/knowledgecenter/SSLVMB_21.0.0/com.ibm.spss.statistics.help/alg_mi-pooling_rubin.htm?lang=en.

[5] idre UCLA. Faq: How are the likelihood ratio, wald, and lagrange multiplier (score) tests different and/or similar? http://www.ats.ucla.edu/stat/mult_pkg/faq/general/nested_tests.htm, 2015.

[6] Xiao-Li Meng and Donald B. Rubin. Performing likelihood ratio tests with multiply-imputed data sets. *Biometrika*, (1):103, 1992.

[7] year = 1999 Stef van Buuren, Karin Oudshoorn . Flexible multivariate imputation by mice. http://www.stefvanbuuren.nl/publications/Flexible

[8] Stef van Buuren. Package 'mice'. https://cran.r-project.org/web/packages/mice/mice.pdf.

[9] Wikipedia. Gibbs sampling. https://en.wikipedia.org/wiki/Gibbs_sampling, 2015.

[10] Wikipedia. Imputation (statistics). https://en.wikipedia.org/wiki/Imputation_%28statistics%29, 2015.

[11] Wikipedia. Missing data. https://en.wikipedia.org/wiki/Missing_data#Missing_at_random, 2015.

[12] Wikipedia. Sleep. https://en.wikipedia.org/wiki/Sleep, 2015.