

**School of Mathematics and Statistics**

**MSc Data-Intensive Analysis**

**MSc Applied Statistics and Datamining**

**MT5762 INTRODUCTORY DATA ANALYSIS**

GROUP: DRUNKEN MASTER 2

ID Numbers: 180012191; 180025784; 110013122; 180029941, 180024795

**What relationships are there between the measured variables and the birth weight of babies?**

*Producing a model that describes potential drivers of low birth-weight babies.*

Tutor: Dr. Carl Donovan

**Executive Summary**

What is the aim?

The present report is a data analysis project, which focusses mainly on fitting linear models. It intends to determine what relationships are there between the measured variables and the birth weight of babies[[1]](#footnote-1).

The data used in this report is part of a larger group of studies from the Child Health and Development Studies (CHDS).

It is known that there are many potential drivers of Low-Birth Weight (LBW) babies and our aim is to determine how these drivers correlate and influence the outcome of our final model.

How are we going to do it?

In order to do so it is required to produce a model that describes potential drivers of low birth-weight babies.

What are the results?

What conclusions can be drawn / Recommendations?

Although 'smoke' had no correlation with birth weight, we are compelled by common sense to infer that there would be an effect of smoking in baby-weight at birth.

**INDEX**

**Content**

[1 INTRODUCTION 1](#_Toc528944261)

[1.1 Variables in Data File Babies 3](#_Toc528944262)

[2 METHODS 4](#_Toc528944263)

[2.1 Data Cleaning 4](#_Toc528944264)

[2.2 Data Exploration 5](#_Toc528944265)

[2.2.1 Baby Birth Date Weight 5](#_Toc528944266)

[2.2.2 Gestation Period vs. Birth Weight 5](#_Toc528944267)

[2.2.3 Scatterplot of Mother’s Weight vs. Baby´s Weight 6](#_Toc528944268)

[2.2.4 Analysis of wt.1 (mother's weight) 7](#_Toc528944269)

[2.2.5 Scatterplot of father's weight against baby's weight 7](#_Toc528944270)

[2.2.6 Mother´s Smoking Habits 8](#_Toc528944271)

[2.3 Model Fitting 8](#_Toc528944272)

[2.3.1 Data Model - Fitting a Model Using Provided Variables 8](#_Toc528944273)

[2.3.2 Fitting a First Order Interaction Model 10](#_Toc528944274)

[2.4 Other Tested Models 13](#_Toc528944275)

[3 FIVE-FOLD CROSS VALIDATION 16](#_Toc528944276)

[3.1.1 Five-Fold Cross-Validation 16](#_Toc528944277)

[3.1.2 Mean Square Error (MSE) 17](#_Toc528944278)

[3.2 Predictions 17](#_Toc528944279)

[4 DISCUSSION 20](#_Toc528944280)

[4.1 Discussion 1 20](#_Toc528944281)

[5 CONCLUSIONS AND RECOMMENDATIONS/DISCUSSION SUMMARY 21](#_Toc528944282)

[6 BIBLIOGRAPHY 22](#_Toc528944283)

# INTRODUCTION

As anyone ever said “life is a gamble” to you? Such a statement reflects the feeling that our lives are surrounded by unpredictable, or “random”, events (Wild & Seber, 2000, p.1).

The present report analyses and discuss some results that can answer the question “what relationships are there between the measured variables and the birth weight of babies?”

The data used in this report is part of a larger group of studies from the Child Health and Development Studies (CHDS), which *“are prospective longitudinal studies on medical and social aspects of pregnancies and on the health and development of children”[[2]](#footnote-2)*.

Previous studies indicate that there are many potential drivers of low birth-weight (LBW) babies. According to Kramer (1987), “*factors with well-established direct causal impacts on intrauterine growth*” and consequently LBW, “*include infant sex, racial/ethnic origin, maternal height, pre-pregnancy weight, paternal weight and height, maternal birth weight, parity, history of prior low-birth-weight infants, gestational weight gain and caloric intake, general morbidity and episodic illness, malaria, cigarette smoking, alcohol consumption, and tobacco chewing*”[[3]](#footnote-3).

The data set we are analysing in this report contains most of the variables mentioned by Kramer above and will be discussed later.

“*Of the 127 million infants born in the world in 1982, 20 million (16%) were estimated to weigh less than 2500g., and over 90% of these infants were born in developing countries, a function not only of the higher birth rate in these countries but also of their LBW[[4]](#footnote-4)*” (Kramer, M, 1987, p.664).

Data cleaning, analysis and plotting were produced in the R programming language using the software R-Studio version 3.5.1 (R Core Team, 2018).

# METHODS

## Data Cleaning

The data were cleaned to remove unknown values that were being presented as numerical within the data set. All variables had been classified as integers within the programming software so the numerical ones were changed to numerical to allow analyses to be performed on them.

## Data Exploration

Exploratory analyses were performed on the data to investigate the potential for the existence of relationships between the variables and birth weight. Correlation values were obtained and used to select which variables to explore. These variables were visualised with scatterplots, giving an indication of the strength of the relationship. The categorical variable of mother’s smoking habits was plotted as a boxplot.

## Model Fitting

### Fitting a Model Using Provided Variables

We use all the variable provided in the data except id and data in our analysis to find a linear model that have the best fit. We choose AIC backwards selection method by using step function. After we get the lowest AIC scores (3359.82), we get the model with following variables (table 1).

|  |
| --- |
| Table 1: Summary |

We also use Anova function to check whether the variable have contributed to the predictive ability of the model, all the p value is less than 0.05, which means all the variables that we selection contribute to the model’s predictive ability. We trace down the extreme residuals in the model. Then we want to check the assumptions about the model. For normality, we use Shapiro Wilk normality test which null hypothesis is the population is normally distributed which our p value is 0.09, we fail to reject the hypothesis. From QQ plot of residuals of the model below and Shapiro Wilk normality test, we could conclude that the model fit normality.

|  |
| --- |
| Figure 9: Histogram of Residuals |

|  |
| --- |
| Figure 10: Normal Q-Q |

For linearity, we have plotted a residual against fitted values graph, even though our graph is not perfect, it is a well-behaved graph that show the linearity of the model. For heteroscedasticity, we use Breusche-Pagan test (ncvTest) which null hypothesis is constant error variance, p value is 0.05 which reject the null hypothesis. So heteroscedasticity do exist and we could also see from the graph of residual against fitted data. For autocorrelation, we use Durbin Watson statistic which null hypothesis is linear regression residuals are uncorrelated, p value is 0.54 which means that we fail to reject the null hypothesis. For collinearity, we use variance inflation factors to find collinearity in our model. Since all of our variance inflation factor is less than 10, we don’t need to worry about collinearity in our model. (The name of the model is dataModel in r file, you could find all the r file in appendix modelselection-Su.r)

|  |
| --- |
| Figure 11: Residuals vs Fitted |

### Fitting a First Order Interaction Model

In this model. We only looked at first order interactions between two variables. There is a possible that second order interaction between three or more variables maybe useful to analysis. We use all the variables and first order interaction between every two variables in the data except **id** and **date**, so we have roughly 200 to 300 variables to start with. Then we will proceed, choose AIC backwards selection approaches using step function. After this stage, we eliminated a high number of variables, being now reduced to some 50 – 60 variables. Then we examine the collinearity of the first order Model. It is observed that there are a considerable number of variable which GVIF number is larger than 10, which prompt us to the following step.

1. We find the maximum number of GVIF; if it is larger than 10 then we remove it;

2. Repeat checking collinearity using vif function get the maximum - repeat the step 1;

After we remove all the collinearity that variance inflation number is larger than 10, we use AIC backwards selection again since we already form a new model by deleting a lot of collinear variables. After model selection, we are left with 12 variables, the AIC scores of the model is 3358.58. We check the collinearity again and the variance inflation factor, now all VIF are less than 10. After we get the model below, we check for the assumption.

|  |
| --- |
| Table 2: Final Models Coefficients |

For normality, we use Shapiro Wilk normality test which null hypothesis is the population is normally distributed which our p value is 0.22, we fail to reject the hypothesis.

|  |
| --- |
| Figure 12: Histogram of Residuals |

|  |
| --- |
| Figure 13: Normal Q-Q |

|  |
| --- |
| Figure 14: Residuals vs Fitted |

From QQ plot of residuals of the model below and Shapiro Wilk normality test, we could conclude that the model fit normality. For linearity, we have plotted a residual against fitted values graph, even though our graph is not perfect, it is a well-behaved graph that show the linearity of the model. For heteroscedasticity, we use Breusche-Pagan test (ncvTest) which null hypothesis is constant error variance, p value is 0.16 which fail to reject the null hypothesis. So heteroscedasticity do not exist and we could also see from the graph of residual against fitted data. For autocorrelation, we use Durbin Watson statistic which null hypothesis is linear regression residuals are uncorrelated, p value is 0.08 which means that we fail to reject the null hypothesis. Now this model pass all the assumption. (The name of the model is finalModel, you could find all the r file in appendix 3 modelselection-Su.r)

## Other Tested Models

Apart from data model, we tried fitting other models based on other criteria too. Using logic as our, we tried to fit certain other interaction-effect models to observe the effect variables had on baby weight.

The other interaction - effect models tested were:

* **Parity** and **mother’s weight** against **baby weight**
* **Mother’s weight and income** against **baby weight**
* **Smoke and mother’s weight** against baby weight

These models were fitted and their AIC scores were for each were really high as compared to dataModel, hence they were not chosen for the final model selection.

AIC scores for the fitted models were:

* **Parity** and **mother’s weight** against **baby weight** = 10547.07
* **Mother’s weight and income** against **baby weight** = 9494.029
* **Smoke and mother’s weight** against **baby weight =** 10457.45

Although, on further testing we found some interesting results. Despite these AIC scores the models passed the model diagnostic tests.

|  |
| --- |
| Figure 15: Normal Q-Q plot |

Normality test for the model – Smoke and mother’s weight against baby weight, passes the test.

|  |
| --- |
| Figure 16: Normsl Q-Q Plot |

Normality test for the model – Income and mother’s weight against baby weight, passes the test.

|  |
| --- |
| Figure 17: Normal Q-Q Plot |

Normality test for the model – Parity and mother’s weight against baby weight, passes the test.

As far as the Durbin-Watson tests are concerned, the p-values for each model were:

* **Parity** and **mother’s weight** against **baby weight** = 0.0468
* **Mother’s weight and income** against **baby weight** = 0.046
* **Smoke and mother’s weight** against **baby weight =** 0.674

Clearly the autocorrelation in these models are either very insignificant or not present at all.

Even for the ncv test the p-values for the models were as follows:

* **Parity** and **mother’s weight** against **baby weight** = 0.03291
* **Mother’s weight and income** against **baby weight** = 0.11154
* **Smoke and mother’s weight** against **baby weight =** 0.43456

The tests show that there is heteroscedasticity in two models but one model does not have it present.

# FIVE-FOLD CROSS VALIDATION

### Five-Fold Cross-Validation

We have, so far, looked upon various models in order to find a better performing one. However, it can be difficult to determine if these improvements in scores result from the captures of better relationships within our model or if we are just overfitting the model. In order to clarify this aspect we use validation techniques such as *k*-Fold Cross Validation (James, Witten, Hastie, & Tibshirani, 2014).

In the cross-validation, the training set is divided into sub-samples, and each single sub-sample will be saved as the data for the verification of model while the other *k-1* groups of sample will be used for training. Cross-validation is repeated *k* times, of which each sub-sample is verified once. The average number of results or other combinations are used, and a single estimate is finally obtained. The advantage of this method is that it repeatedly uses randomly generated sub-samples for training and validation. Actually, 10-Fold Cross Validation is the most commonly used[[5]](#footnote-5).

In our experiment, *k* has a specific value, 5, the reference to the model with be 5-Fold Cross-Validation.

### Mean Square Error (MSE)

Mean Square Error (MSE) is used to evaluate the quality of an [estimator](https://en.wikipedia.org/wiki/Estimator) (parameter) or a predictor (some [random variable](https://en.wikipedia.org/wiki/Random_variable)), in other words,is the average of the square of the errors. MSE satisfies the equation as below:

MSE (*T*) = var (*T*) + (bias (*T*))2

where bias(*T*) = E(*T*) - **

Usually, if the MSE of one model is larger, the error of this model will be larger.

## Predictions

Cross-validation for linear regression can be easily achieved in R Studio. The function ‘cv.lm’ can predict the accuracy for multiple linear regression. By using the function ‘cv.lm’, the output will be displayed, including the analysis of variance table and the observations in five test sets. The whole output can be found in Appendix 2. The output gives the plots of the cross-validation predicted value of two models as below in Figure 15 and 16. It is quite hard to say whether **dataModel** or **finalModel** is the better because the five regression lines all seems parallel in both plots.

Thus, the focus of the output should be the comparison of overall ms (mean square) of both finalModel and dataModel. From the output, the overall ms of finalModel is 268 whilst which of dataModel is 255. It can be predicted that dataModel is a bit more suitable for this case than finalModel.

|  |
| --- |
| Preditedcrossvalidation-final.png  Figure 18: Cross-Validation predicted values for FinalModel |

|  |
| --- |
| Predictedobserveddata.png  Figure 19: Cross Validation Predicted Values for DataModel |

Also, by using get\_mse function, the output turns out that the MSE value of finalModel is 258 while the MSE value of dataModel is 248. Although the discrepancy between these two model is not so large, it is clear that dataModel is better than finalModel in this case.

# DISCUSSION

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut varius laoreet consectetur. Nulla in vulputate lacus. In et volutpat ante. In ultrices turpis neque, id dictum erat auctor ac. Pellentesque mattis, magna rutrum rutrum placerat, orci augue molestie nunc, vel porttitor risus tellus non dui. Vivamus id convallis odio. Duis ut nulla id nulla gravida vulputate a condimentum tellus. Ut purus justo, tempus sed iaculis at, accumsan id sapien. Etiam eu massa vehicula, accumsan urna sed, varius magna. Maecenas dapibus arcu leo, et tempus leo tincidunt id.

## Discussion 1

# CONCLUSIONS AND RECOMMENDATIONS/DISCUSSION SUMMARY

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut varius laoreet consectetur. Nulla in vulputate lacus. In et volutpat ante. In ultrices turpis neque, id dictum erat auctor ac. Pellentesque mattis, magna rutrum rutrum placerat, orci augue molestie nunc, vel porttitor risus tellus non dui. Vivamus id convallis odio. Duis ut nulla id nulla gravida vulputate a condimentum tellus. Ut purus justo, tempus sed iaculis at, accumsan id sapien. Etiam eu massa vehicula, accumsan urna sed, varius magna. Maecenas dapibus arcu leo, et tempus leo tincidunt id.

# BIBLIOGRAPHY

Brownlee, J. (2018). A gentle introduction to k-fold cross-validation, Statistical Methods, 23 May. Obtido de https://machinelearningmastery.com/k-fold-cross-validation/

Crawley, M. J. (2015). *Statistics - An Introduction Using R*. ( lda John Wiley & Sons, Ed.) (Second Edi). Sussex.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2014). *An Introduction to Statistical Learning with Applications*. (S. Publishing, Ed.) (7th ed.). New York.

Kramer, M, S. (1987). Determinants of low birth weight: methodological assessment and meta-analysis. *Bulletin of the World Health Organization, 65 (5): 663-737 (1987)*. Obtido de https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2491072/?page=1

Van den Broeck, J., Argeseanu Cunningham, S., Eeckles, R., & Herbst, K. (2005). Data Cleaning: Detecting, Diagnosing, and Editing Data Abnormalities. Obtido de https://doi.org/10.1371/journal.pmed.0020267

Wild, C. J., & Seber, G. A. F. (2000). *Chance Encounters - A first Course in Data Analysis and Inference*. (J. W. & Son, Ed.) (1st ed.). USA: John Wiley & Son, Inc.

**APPENDIX**

Appendix 1 – Abbreviations

**Abbreviations**

**A**

AIC = Akaike´s Information Criterion

**D**

drace = father’s race, coding same as mother´s race

dage = father´s age, coding same as mothers age

ded = father´s education, coding same as mother´s education

dht = father´s height, coding same as mothers height

dwt = father´s weight, coding same as mothers weight

**E**

ed = mother´s education

**G**

GVIF – Variance Inflation Factor

**H**

ht = mother´s height in inches to the last completed inch

**I**

id = identification number

inc = family yearly income in $2500 increments

**L**

LBW = Low Birth Weigh

**N**

number = number of cigarettes smoked per day for past and current smokers

**W**

wt = birth weight in ounces

Appendix 2 Output







Appendix 3 - modelselection-Su.r

  #setwd("~/Masters/")

library(tidyverse)

library(ggplot2)

library(car)

library(GGally)

library(effects)

#setwd("~/Masters/")

babies.data <- read.table("babies23.data", header = TRUE)

#since we are working in our directory, I change the directory that I think that

#people use this project can run it.

#observations from data set:

# pluralty is always 5

# outcome is always 1

# there are values of 999 for gestation but readme doc does not clarify if

# these are unknown - CLEANED ANYWAY

# all subjects are male

# for race, I'm unsure why white is assigned six values (0-5) - one unknown

# two unknown ages (mother) - CLEANED

# one unknown education (mother) - CLEANED

# many unknown heights (mother) - CLEANED

# many unkown weights (mother) - CLEANED

# five unknown fathers' races as well as values of 10? - 99s CLEANED -

# many unknown fathers' ages - CLEANED

# many unknown fathers' educations - CLEANED

# many unknown fathers' heights - CLEANED

# many unknown fathers' weights - CLEANED

# no explanation of 0 in marital status - assume unknown?

# many unknown incomes - CLEANED

# ten unknown smokers - CLEANED

# nine unknown quitting times, one not asked - CLEANED

# ten unknown number of cigarettes smoked - CLEANED

##### cleaning the data as per unknown values above #####

clean.data <- babies.data

clean.data$gestation[clean.data$gestation == "999"] <- NA

clean.data$age[clean.data$age == "99"] <- NA

clean.data$ed[clean.data$ed == "9"] <- NA

clean.data$ht[clean.data$ht == "99"] <- NA

clean.data$wt[clean.data$wt == "99"] <- NA

clean.data$drace[clean.data$drace == "99"] <- NA

clean.data$dage[clean.data$dage == "99"] <- NA

clean.data$ded[clean.data$ded == "9"] <- NA

clean.data$dht[clean.data$dht == "99"] <- NA

clean.data$dwt[clean.data$dwt == "999"] <- NA

clean.data$inc[clean.data$inc == "98"] <- NA

clean.data$smoke[clean.data$smoke == "9"] <- NA

clean.data$time[clean.data$time == "99"] <- NA

clean.data$time[clean.data$time == "98"] <- NA

clean.data$number[clean.data$number == "98"] <- NA

clean.data$wt.1[clean.data$wt.1 == "999"] <- NA

#make some factors numeric

clean.data <- clean.data %>% mutate\_each(funs(as.numeric), 5)

clean.data <- clean.data %>% mutate\_each(funs(as.numeric), 7)

clean.data <- clean.data %>% mutate\_each(funs(as.numeric), 10)

clean.data <- clean.data %>% mutate\_each(funs(as.numeric), 12:13)

clean.data <- clean.data %>% mutate\_each(funs(as.numeric), 15)

clean.data <- clean.data %>% mutate\_each(funs(as.numeric), 17:18)

####### Exploration of the birthweight data #######

#normally distributed

hist(clean.data$wt)

summary(clean.data$wt)

##################################

clean.data.naomit <- na.omit(clean.data)

# select data that does not contain id and data of birth

# consider this two factor does not have effect on baby birth weight

# on the real life

clean.data.naomit <- clean.data.naomit %>% dplyr::select(-id, -date)

#factor(clean.data.naomit$id)

dataModel <- lm(wt ~., data = clean.data.naomit)

summary(dataModel)

#try to use Anova

Anova(dataModel)

#model selection use AIC

dataModel <- step(dataModel)

Anova(dataModel)

#check about normality of dataModel's residual

qqnorm(resid(dataModel))

qqline(resid(dataModel))

#the qq plot looks great but the shapiro test, p value is large than 0.05,

# so the residual of the data Model is normal

shapiro.test(resid(dataModel))

hist(resid(dataModel))

# we track down the extreme residuals

bigResid <- which(abs(resid(dataModel))>5)

clean.data.naomit[bigResid,]

#plot residuals against fitted values

dataResid <- resid(dataModel)

plot(fitted(dataModel),dataResid, ylab= "Residuals", xlab = "Fitted Values")

#it looks good

#https://onlinecourses.science.psu.edu/stat501/node/277/

# do Breusche-Pagan test with respect to fitted model

ncvTest(dataModel)

# null hypothesis: constant error variance. "If we have constant error variance

#then the variation in the residuals should be unrelated to any coveriant."

# null hypothesis is rejected since the p value is less than 0.05

#MT5761 notes page 22

# need to write durbinWatsonTest on model

durbinWatsonTest(dataModel)

#null hypothesis: error are uncorrelated, fail to reject the null hypothesis

plot(dataModel, which = 1:2)

#collinearity

numericOnly <- clean.data.naomit %>% select\_if(is.numeric)

#use with caution, picture is sooo huge and difficult to generate

# and do harm to my computer and not useful because we have sooo many variabales

#ggpairs(numericOnly)

vif(dataModel)

# all number is less than 10, do not have to delete any variable

#calculate confidence interval of the model

confint(dataModel)

#add more effect plot if you want and select variable that you

# think is interested

#plot(effect(term="gestation", mod = dataModel))

#plot(effect(term="smoke", mod = dataModel))

#plot(effect(term="number", mod = dataModel))

cols\_to\_change = c(1, 2, 3, 4,6, 8, 9, 11, 14, 16, 19, 20:23)

for(i in cols\_to\_change){

class(clean.data[, i]) = "factor"

}

cols\_to\_change

#create a first order iteraction for every variable

firstorderModel <- lm(wt ~.\*., data = numericOnly)

summary(firstorderModel)

#model selection use AIC

firstorderModel <- step(firstorderModel)

summary(firstorderModel)

Anova(firstorderModel)

qqnorm(resid(firstorderModel))

qqline(resid(firstorderModel))

shapiro.test(resid(firstorderModel))

hist(resid(firstorderModel))

firstorderResid <- resid(firstorderModel)

plot(fitted(firstorderModel),firstorderResid, ylab= "Residuals", xlab = "Fitted Values")

ncvTest(firstorderModel)

durbinWatsonTest(firstorderModel)

plot(firstorderModel, which = 1:2)

# we exam the collinearity of the firstorderModel we find that there are a lot of

# variable that its GVIF number is larger than 10, so in the following step.

# 1. we find the maximum number of GVIF, if it is larger than 10,remove it

# 2. do the vif function again to check the collinearity and get the maximum repeat the step 1

# we do the above two steps until all the variable's collinearity GVIF is less than 10

# or we do not have a collinearity problem anymore

# following just the process of removing every variable that is collinear

k<-vif(firstorderModel)

k[which.max(k)]

alteredModel <-update(firstorderModel,.~.-ht:marital )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-race )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-smoke )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-dht:race)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-dage)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-age:marital)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-drace)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-dht:inc)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-gestation:number)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-wt.1)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-ht:smoke)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-marital:dage )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-ed )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-parity )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-age:dwt )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-marital:race )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-age:race )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-dwt:wt.1 )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-gestation:drace )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-ded:dwt )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-dwt:dage )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-gestation:smoke )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-ded:time )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-marital:ed )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-dage:race )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-dwt:ed )

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-gestation:parity)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-ed:smoke)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-age:drace)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-dwt:race)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-dwt:smoke)

p<-vif(alteredModel)

p[which.max(p)]

alteredModel <-update(alteredModel,.~.-inc:ed)

p<-vif(alteredModel)

p[which.max(p)]

#finally, we finish deleting collinear variable and we do a AIC do a backward

#model selection and get the finalModel

finalModel <- step(alteredModel)

#check final model colinearity and all of them are less than 10, it works.

vif(finalModel)

#get summary of finalModel

summary(finalModel)

#use qq plot and Shapiro-Wilk normality test to test the normality

# because the p value in Shapiro-Wilk normality test is larger than 0.05,

# the data is normal, the QQ plot show the same result

qqnorm(resid(finalModel))

qqline(resid(finalModel))

shapiro.test(resid(finalModel))

hist(resid(finalModel))

plot(finalModel, which = 1:2)

# do Breusche-Pagan test with respect to fitted model

ncvTest(finalModel)

# null hypothesis: constant error variance. "If we have constant error variance

#then the variation in the residuals should be unrelated to any coveriant."

# null hypothesis is rejected since the p value is less than 0.05

# need to write durbinWatsonTest on model

durbinWatsonTest(finalModel)

#null hypothesis: error variances are uncorrelated, fail to reject the null hypothesis

#MT5761 notes page 22

Anova(finalModel)

#get the confidence interval

confint(finalModel)

1. <https://moody.st-andrews.ac.uk/moodle/course/view.php?id=8191> [↑](#footnote-ref-1)
2. <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-3016.1988.tb00218.x> [↑](#footnote-ref-2)
3. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2491072/?page=1> [↑](#footnote-ref-3)
4. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2491072/?page=2> [↑](#footnote-ref-4)
5. <https://machinelearningmastery.com/k-fold-cross-validation/> [↑](#footnote-ref-5)