IST772– Problem Set 9

Sathish Kumar Rajendiran

Attribution statement: 1. I did this homework by myself, with help from the book and the professor.

```
# import libraries
#create a function to ensure the libraries are imported
EnsurePackage <- function(x){
  x <- as.character(x)
    if (!require(x,character.only = TRUE)){
      install.packages(pkgs=x, repos = "http://cran.us.r-project.org")
      require(x, character.only = TRUE)
    }
  }
```

Chapter 10, Exercise 1

The data sets package in R contains a small data set called swiss that contains n = 47 observations of socio-economic indicators for each of 47 French-speaking provinces of Switzerland at about 1888. Use "?swiss" to display help about the data set. All the data in this data set are metric, but one, Catholic, shows a very bimodal distribution. We can dichotomize this variable to create binary variable as follows:

```
?swiss
# Swiss Fertility and Socioeconomic Indicators (1888) Data
# Standardized fertility measure and socio-economic indicators for each of 47 French-
speaking provinces of Switzerland at about 1888.

# A data frame with 47 observations on 6 variables, each of which is in percent, i.e.
, in [0, 100].
# [,1]  Fertility   Ig, 'common standardized fertility measure'
# [,2]  Agriculture % of males involved in agriculture as occupation
# [,3]  Examination % draftees receiving highest mark on army examination
# [,4]  Education   % education beyond primary school for draftees.
# [,5]  Catholic    % 'catholic' (as opposed to 'protestant').
# [,6]  Infant.Mortality    live births who live less than 1 year.
# All variables but 'Fertility' give proportions of the population.

summary(swiss)
```

```
##    Fertility      Agriculture     Examination      Education    
##  Min.   :35.00   Min.   : 1.20   Min.   : 3.00   Min.   : 1.00  
##  1st Qu.:64.70   1st Qu.:35.90   1st Qu.:12.00   1st Qu.: 6.00  
##  Median :70.40   Median :54.10   Median :16.00   Median : 8.00  
##  Mean   :70.14   Mean   :50.66   Mean   :16.49   Mean   :10.98  
##  3rd Qu.:78.45   3rd Qu.:67.65   3rd Qu.:22.00   3rd Qu.:12.00  
##  Max.   :92.50   Max.   :89.70   Max.   :37.00   Max.   :53.00  
##     Catholic       Infant.Mortality
##  Min.   :  2.150   Min.   :10.80   
##  1st Qu.:  5.195   1st Qu.:18.15   
##  Median : 15.140   Median :20.00   
##  Mean   : 41.144   Mean   :19.94   
##  3rd Qu.: 93.125   3rd Qu.:21.70   
##  Max.   :100.000   Max.   :26.60
```

```
dim(swiss)
```

```
## [1] 47  6
```

```
# View(swiss)

str(swiss)
```

```
## 'data.frame':    47 obs. of  6 variables:
##  $ Fertility       : num  80.2 83.1 92.5 85.8 76.9 76.1 83.8 92.4 82.4 82.9 ...
##  $ Agriculture     : num  17 45.1 39.7 36.5 43.5 35.3 70.2 67.8 53.3 45.2 ...
##  $ Examination     : int  15 6 5 12 17 9 16 14 12 16 ...
##  $ Education       : int  12 9 5 7 15 7 7 8 7 13 ...
##  $ Catholic        : num  9.96 84.84 93.4 33.77 5.16 ...
##  $ Infant.Mortality: num  22.2 22.2 20.2 20.3 20.6 26.6 23.6 24.9 21 24.4 ...
```

```
# define variables
Fertility <- swiss$Fertility
Education <- swiss$Education
Examination <- swiss$Examination
Catholic <- swiss$Catholic
Infant <- swiss$Infant
Agriculture <- swiss$Agriculture

colnames(swiss)
```

```
## [1] "Fertility"        "Agriculture"      "Examination"      "Education"       
## [5] "Catholic"         "Infant.Mortality"
```

```r
# load the necessary library for further processing...
EnsurePackage("tidyverse")
```

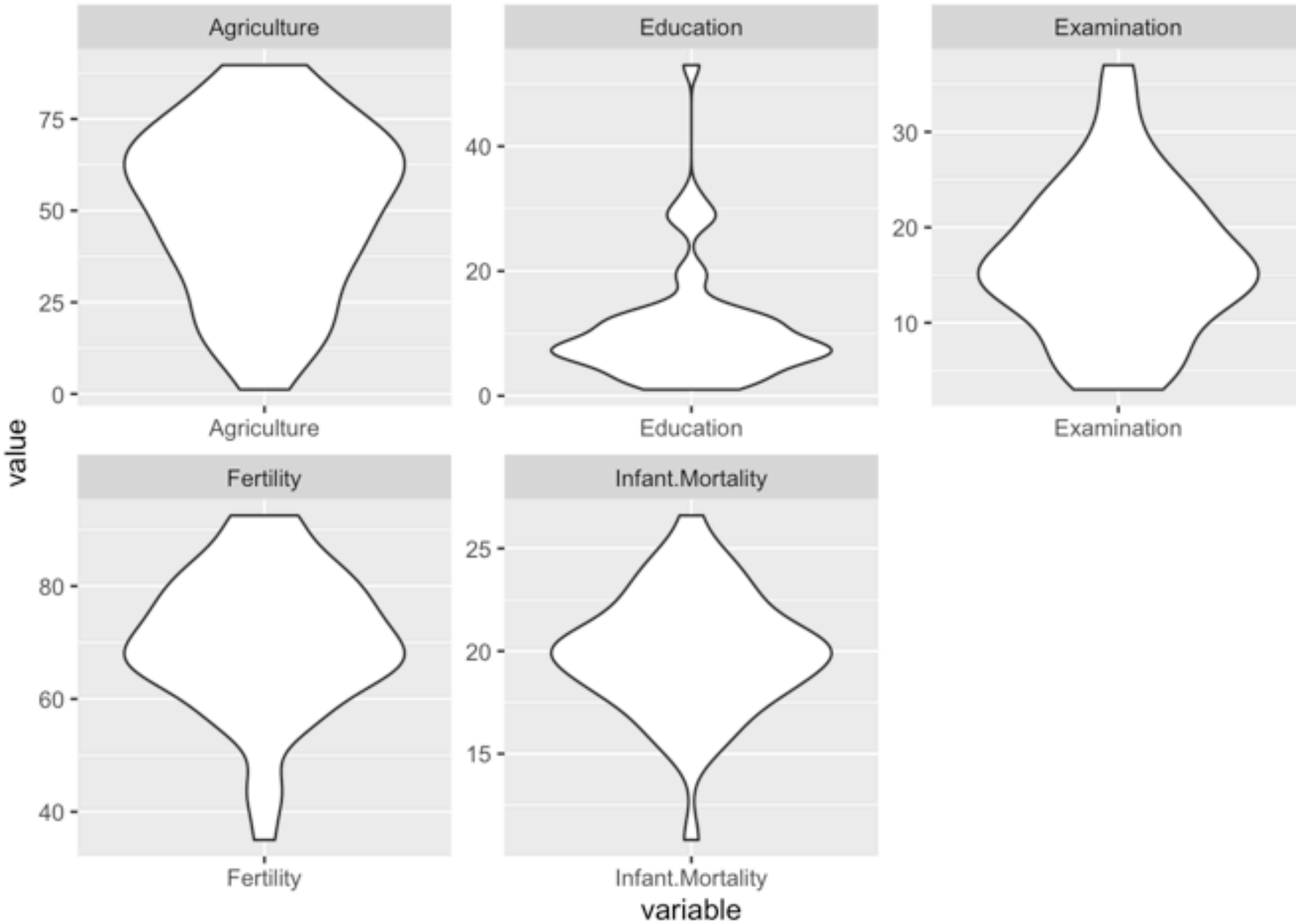
```
## Loading required package: tidyverse
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.0 ──
```

```
## ✓ ggplot2 3.3.2     ✓ purrr   0.3.4
## ✓ tibble  3.0.1     ✓ dplyr   1.0.0
## ✓ tidyr   1.0.2     ✓ stringr 1.4.0
## ✓ readr   1.3.1     ✓ forcats 0.5.0
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

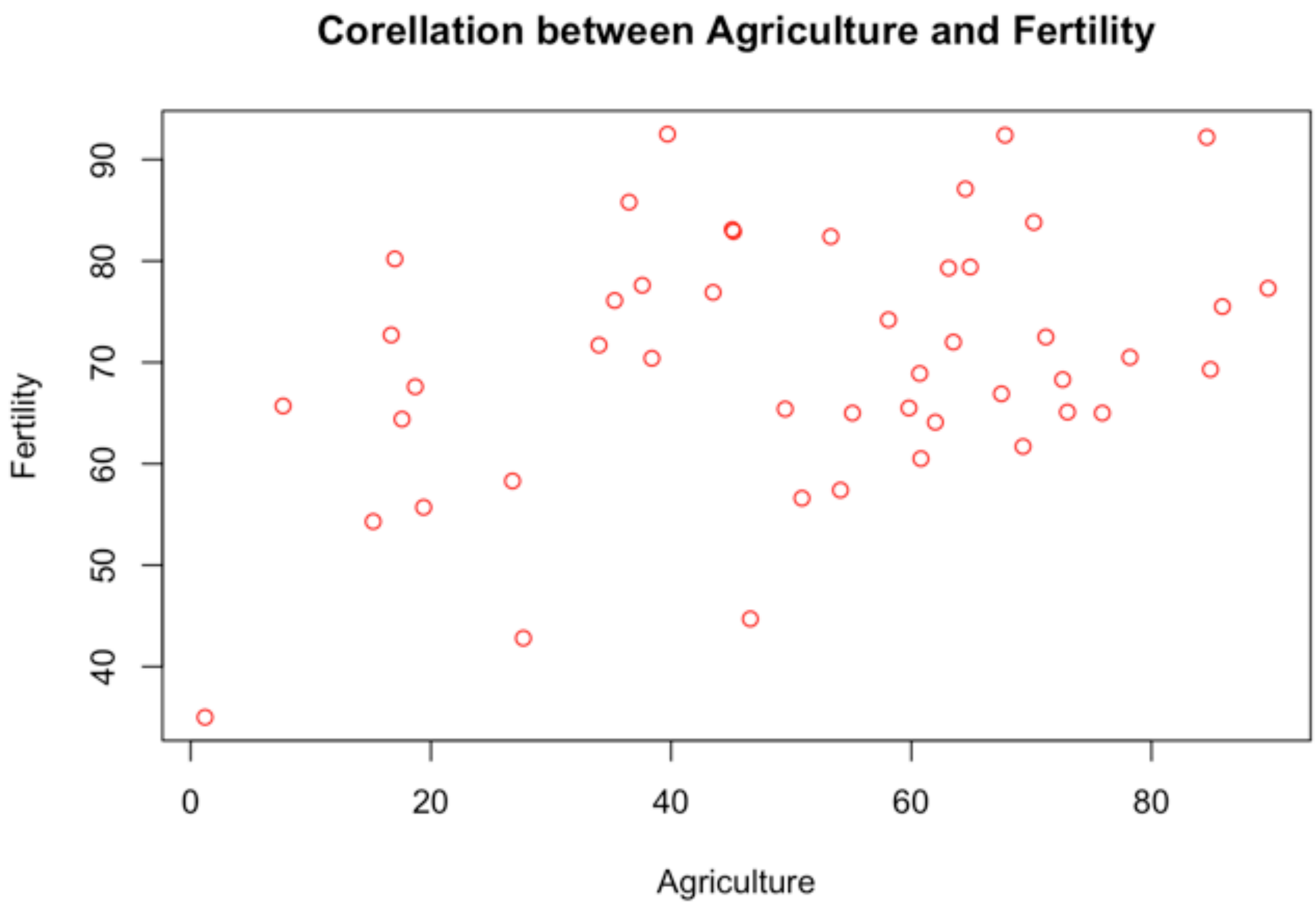
```r
swiss %>%
  pivot_longer(cols= -Catholic, names_to="variable", values_to="value", values_drop_n
a = TRUE) %>%
  ggplot(aes(x=variable, y=value)) + geom_violin() + facet_wrap( ~ variable, scales="
free")
```




```
swiss$Catholic.b <- as.integer(swiss$Catholic > 60) # 60 looks like a gap in the hist
ogram
table(swiss$Catholic.b)
```

```
##
##  0  1
## 31 16
```

```
# Corellation between Agriculture and Fertility
plot(Agriculture, Fertility
     ,main="Corellation between Agriculture and Fertility"
     ,col="red")
```



Use logistic regression to predict Catholic.b, using two metric variables in the data set, Fertility and Agriculture (1 pt). Run any necessary diagnostics. (1 pt) Interpret the resulting null hypothesis significance tests. (1 pt)

1) Question 1: Logistic Regression Analysis on Swiss data

- glm() | Dependent variable (Catholic.b) & Fertility and Agriculture as Predictors/Independent variables with binomial(link = "logit") link function
    - glm(Catholic.b ~ Fertility + Agriculture,data=swiss,family = binomial(link = "logit")) is stored on a variable "swiss.glm"
    - above code runs a logistic regression analysis on Catholic.b as the dependent variable and Fertility and Agriculture as the predictors from "swiss" dataset.
- Diagnostics:
    - From the glm(Catholic.b ~ Fertility + Agriculture,data=swiss,family = binomial(link = "logit")) ; this formula predicts "Catholic.b" values from Fertility and Agriculture combined from the dataset "swiss"
    - Procedure glm() is similar to lm() procedure and expects the dependent variable (variable, in question for prediction) first. Followed by, independent/prdictor variable and a link function. It can have number of predictor variables follows after "~" symbol. "." after "~" means - it expects to include all the remaining variables from the dataset. In this case, we are only trying to predict

“fertility” rate based on “Education” & “Agriculture” variable.
  - In addition, “data=swiss” implies what is the sample/population the prediction is run against from the observations it contains.
  - Successul execution of the glm() procedure provides results as shown below.
    - call
    - Deviance Residuals
    - Coefficients
    - Significant codes
    - Null and Residual deviance with degrees of freedom
    - Number of Fisher Scoring iterations
  - Chi-Square analysis on logistic regression
    - anova(swiss.glm, test=“Chisq”) -performs chi-square analysis on the glm() output
  - Convert the log odds for the coefficient on the predictor into regular odds
    - exp(coef(glmOut)) - converts log odds into regular odds
  - Null hypothesis
- Results:
  - Once the glm() procedure executes successfuly, it returns various data points as an outcome. The first two lines defines the model, we wanted.
    - glm(formula = Catholic.b ~ Fertility + Agriculture, family = binomial(link = “logit”), data = swiss)
    - By specifying binomial() - it invokes the inverse logit or logistic function as the basis for fitting the X variables to the Y variable.
  - Next, Summary of residuals that gives an overview of errors of prediction.
    - With min as -1.50316 and max of 2.39654 shows, distribution of residuals between -ve to positive almost spreading equally on both sides, with Median almost 0 (-0.01404).
    - It seems the residuals are symmetrically distributed.
    - hist(residuals(swiss.glm)) suggests the same.
  - Coefficients shows the key results.
    - Intercept is at -34.07275.
    - Slopes for Education variable is 0.35010 and Agriculture is 0.13078 are way off from the intercept or B-weights. These coefficients define the logrithm of the odds of the Y variable.
    - Std.Errors around the estimates of slope and intercept shows the estimated sampling distribution around these point estimates.
    - z-value shows the student’s t-test of the null hypothesis test that each estimated coefficients is equal to zero.
    - two asterik (*) indicates the significance level of alpha at p < 0.01.
    - one asterik (*) indicates that the significance level of alpha level is 0.05.
    - With above p-value; * Fertility has the strong coefficient value as 0.00293 far less than the test of significance (p < 0.01); * Agriculture has p-value as 0.01384 far less than the test of significance (p < 0.05) This shows that , both “Agriculture” and “Fertility” are both statistically significant.
    - Null Hypothesis * Null hypothesis is that the log odds of catholic.b is equal to 0 in the population. Since the log odds of Fertility and Education both are statistically significant and less than thier respective alpha levels - we can reject the null hypothesis.

    - In addition, the conversion from log odds to regular output (exp(coef(glmOut))), (Intercept)

Fertility Agriculture 0.0000000000000001593646 1.419211326551196750145 1.139719619964769448117 From above its is infered that 1.419:1 on fertility and 1.139:1 on Agriculture to likely to claim catholic.b

- The first Chi-Square model compares three nested models.
    - anova(swiss.glm, test="Chisq") - includes both predictors and tests the level of significance on these predictors. It confirms that both predictors with 0.0000009474 and 0.0001207 is far less the the p value (0.001) and they are statistically significant.
    - 24.032 is the chi-square value (residual deviance from top line - residual deviance from 2nd line) 60.284 - 36.252 = 24.032 is tested for significance on one degree of freedom.

Please find more details from the R code below,

```
options(scipen=999)  # turn-off scientific notation like 1e+48
swiss.glm <- glm(Catholic.b ~ Fertility + Agriculture,data=swiss,family = binomial(li
nk = "logit"))
summary(swiss.glm)
```

```
##
## Call:
## glm(formula = Catholic.b ~ Fertility + Agriculture, family = binomial(link = "logi
t"),
##     data = swiss)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.50316  -0.24820  -0.01404   0.14290   2.39654
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -34.07275   11.31906  -3.010  0.00261 **
## Fertility     0.35010    0.11768   2.975  0.00293 **
## Agriculture   0.13078    0.05314   2.461  0.01384 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 60.284  on 46  degrees of freedom
## Residual deviance: 21.470  on 44  degrees of freedom
## AIC: 27.47
##
## Number of Fisher Scoring iterations: 8
```

```
#Chi-Square analysis on logistic regression
anova(swiss.glm, test="Chisq")
```

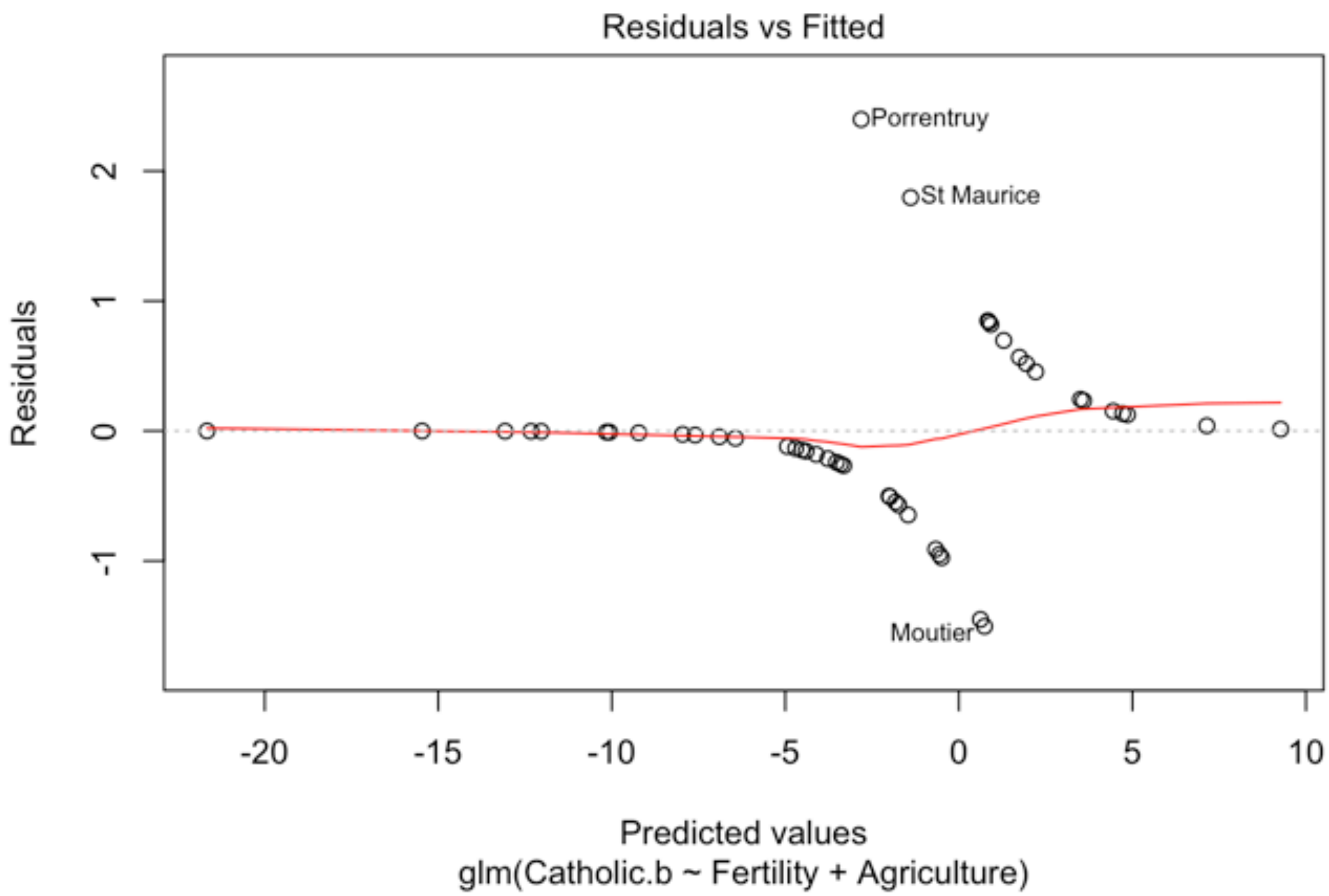
	Df <int>	Deviance <dbl>	Resid. Df <int>	Resid. Dev <dbl>	Pr(>Chi) <dbl>
NULL	NA	NA	46	60.28383	NA
Fertility	1	24.03211	45	36.25172	0.0000009474259
Agriculture	1	14.78148	44	21.47024	0.0001207152326

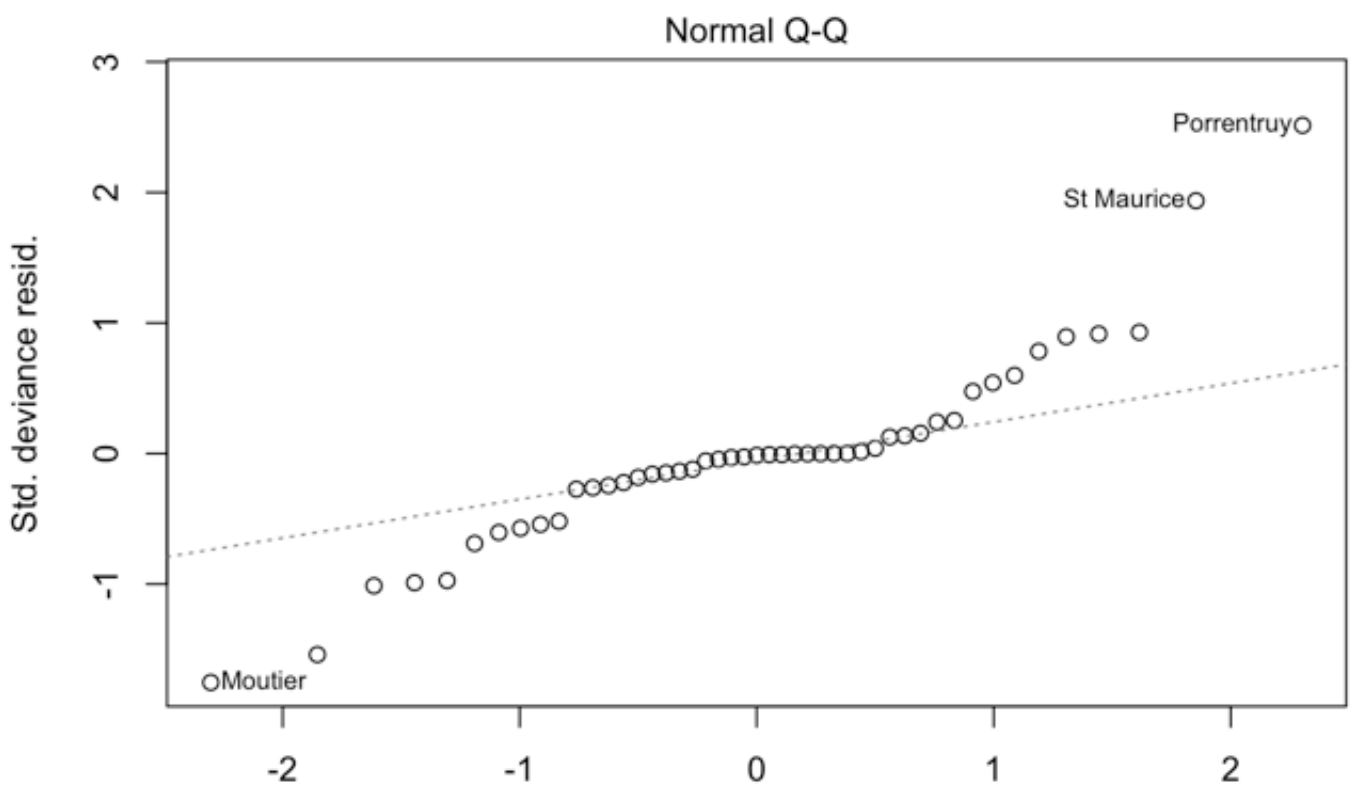
3 rows

```
# Convert the log odds for the coefficient on the predictor into regular odds
exp(coef(swiss.glm))
```

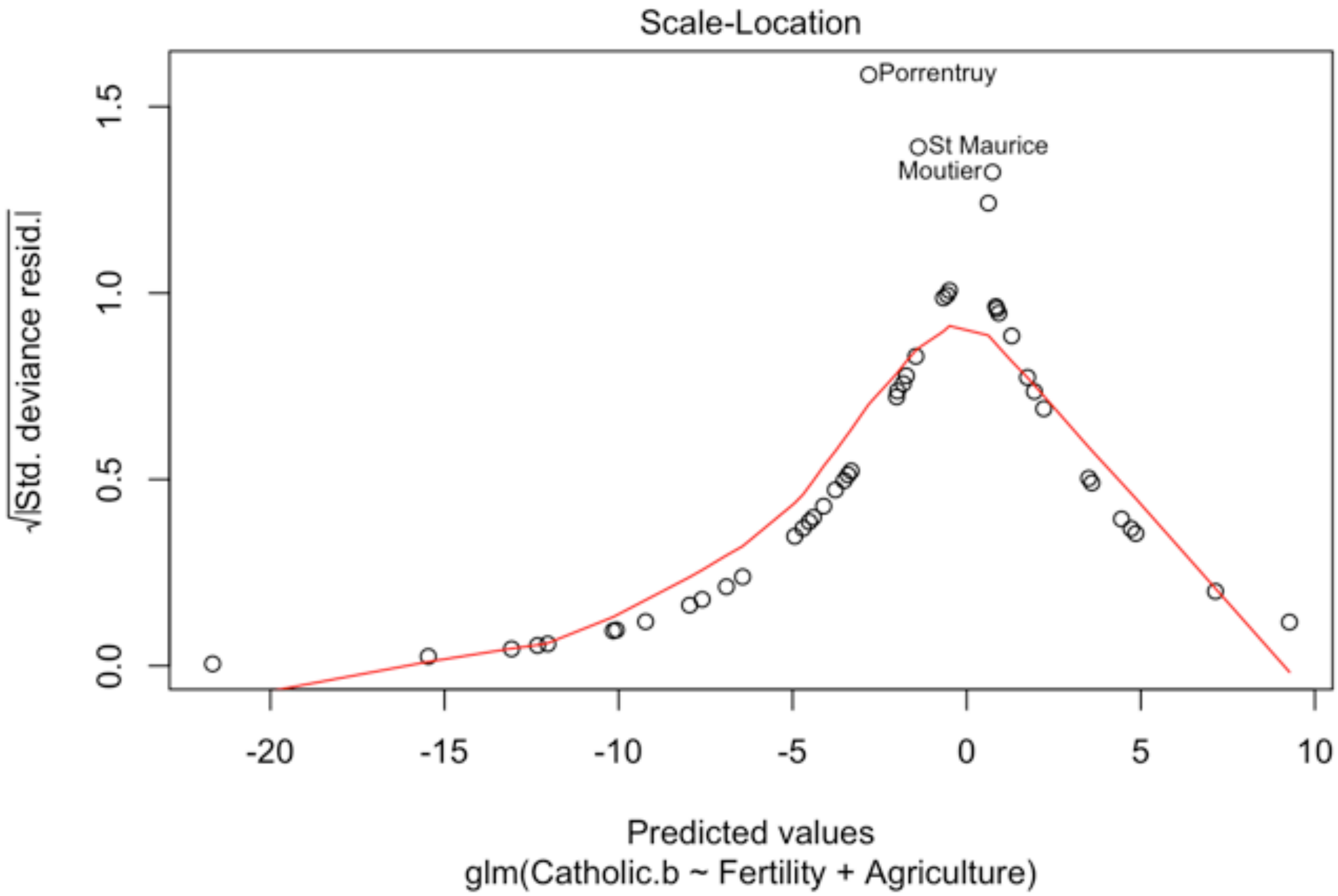
```
##             (Intercept)               Fertility             Agriculture 
## 0.000000000000001593646 1.419211326551196750145 1.139719619964769448117
```

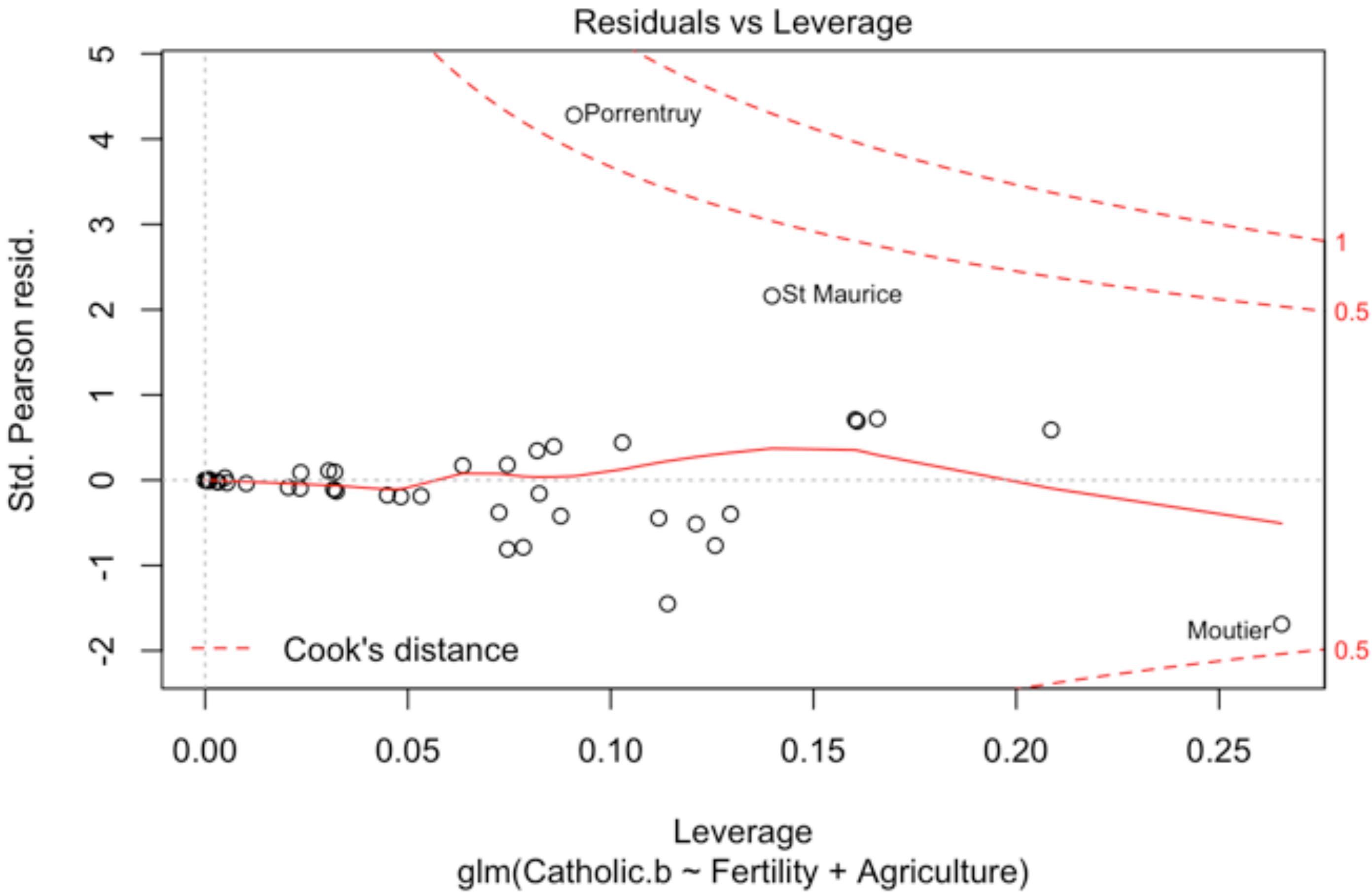
```
plot(swiss.glm)
```


Residuals vs Fitted
Porrentruy
St Maurice
Moutier
Residuals
Predicted values
glm(Catholic.b ~ Fertility + Agriculture)

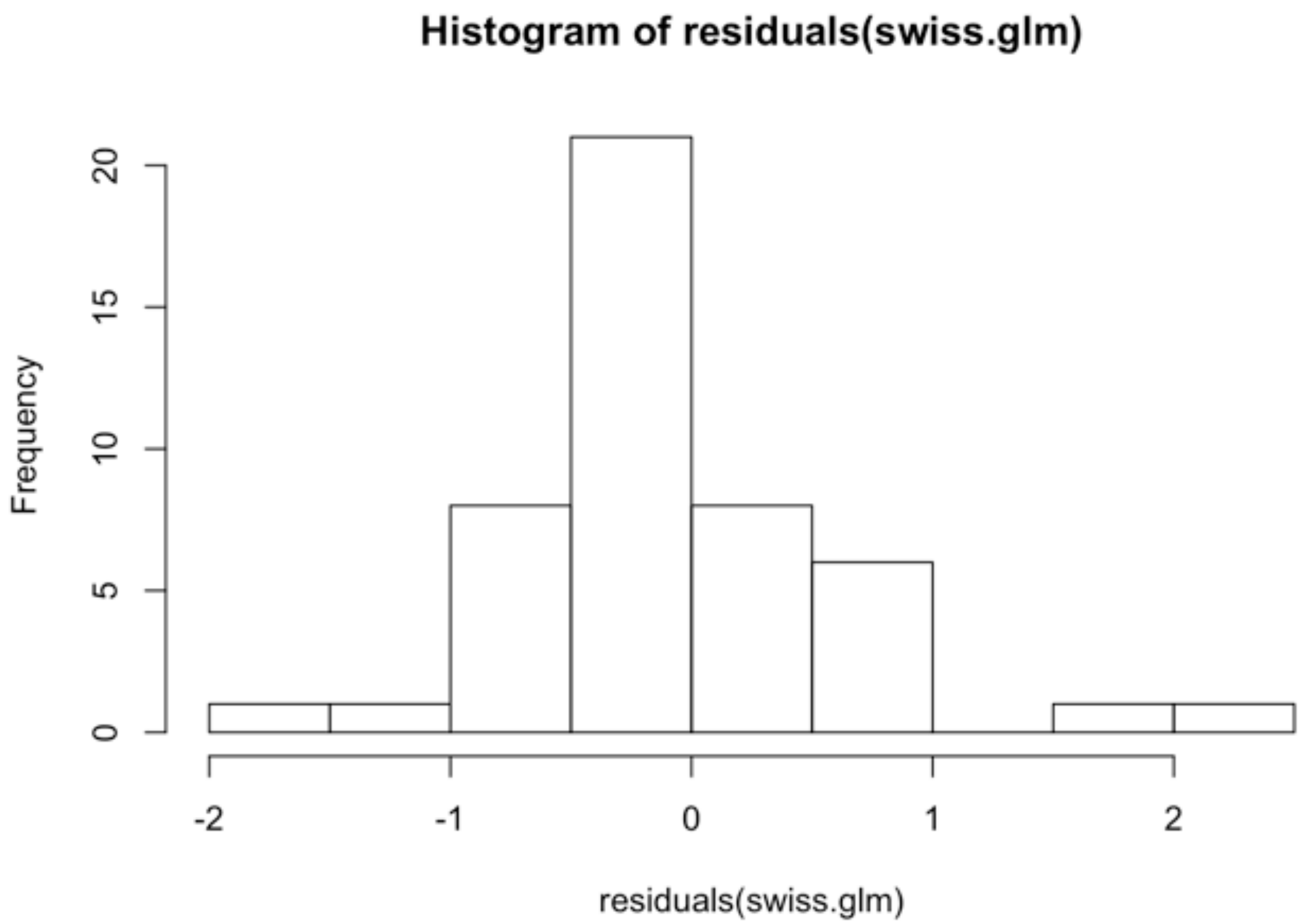

Normal Q-Q
Porrentruy
St Maurice
Moutier
Std. deviance resid.

Theoretical Quantiles
glm(Catholic.b ~ Fertility + Agriculture)






```
hist(residuals(swiss.glm))
```



Chapter 10, Exercise 5

As noted in the chapter, the BaylorEdPsych add-in package contains a procedure for generating pseudo-R-squared values from the output of the glm() procedure. Use the results of Exercise 1 to generate, report, and interpret a Nagelkerke pseudo-R-squared value. You might also examine the confusion matrix. (1 pt)

2) Question 5: Pseudo-R-squared values

- Pseudo-R-squared value is calculated as below,
  - PseudoR2(swiss.glm) - this formula outputs various coeffients including the Nagelkerke in scope.
- Nagelkerke - Interpretation
  - The Nagelkerke comes significanttly larger that than few the other values at 0.7778181
    - McFadden 0.6438474
    - Adj.McFadden 0.5111418
    - Cox.Snell 0.5621247
    - McKelvey.Zavoina 0.9169675
    - Effron 0.6902251
  - Nagelkerke R-squared value is interpreted as the propotion of variance in the outcome variable from earlier exercise “Catholic.b” accounted for the predictor variables “Agriculture” and “Fertility”.

    - As only had a sample size of 47 observations, we were able to detect the much smaller effect on these predictor variables.
- In addtion, to conclude - lets explore on the confusion matrix as well
    - Its a contingency table that compares the observed outcome vs the predicted results
    - table(round(predict(swiss.glm,type="response")),Catholic.b) - formula produces confusion matrix on the glm model output.
    - Outcome is dichotimized to 0 and 1 signifying result towards Catholic.b or not.
    - 0 means Catholic.b as Yes and 1 as Catholic.b as No
    - Overall accuracy is calculated as (29+14)/47 = 0.9148936 (~91.5% accuracy with these predictor variables)
    - 91.5% accuracy further suggesting the significance of the predictor variables.

Please find more details from the R code below,

```
# load the necessary library for further processing...
EnsurePackage("BaylorEdPsych")
```

```
## Loading required package: BaylorEdPsych
```

```
PseudoR2(swiss.glm) # generate pseudo-R-squared values
```

```
##         McFadden     Adj.McFadden        Cox.Snell       Nagelkerke
##        0.6438474        0.5111418        0.5621247        0.7778181
## McKelvey.Zavoina           Effron            Count        Adj.Count
##        0.9169675        0.6902251        0.9148936        0.7500000
##              AIC    Corrected.AIC
##       27.4702414       28.0283810
```

```
cat("\nconfusion matrix:\n")
```

```
##
## confusion matrix:
```

```
#confusion matrix
table(round(predict(swiss.glm,type="response")),swiss$Catholic.b)
```

```
##
##       0  1
##    0 29  2
##    1  2 14
```

```
# (29+14)/47
```

Chapter 10, Exercise 6

Continue the analysis of the Chile data set described in this chapter. The data set is in the "car" package, so you will have to install.packages() and library() that package first, and then use the data(Chile) command to get access to the data set and "? Chile" to see the documentation. Pay close attention to the transformations needed to isolate cases with the Yes and No votes as shown in this chapter. Add a new predictor, statusquo, into the model and remove the income variable. Your new model specification should be vote ~ age + statusquo. The statusquo variable is a rating that each respondent gave indicating whether they preferred change or maintaining the status quo. Conduct general linear model (1 pt + 1 pt) and Bayesian analysis on this model (1 pt) and report and interpret all relevant results (1 pt). Compare the AIC from this model to the AIC from the model that was developed in the chapter (using income and age as predictors).

3) Question 6: Logistic Regression Analysis on Chile data

- Data Preprocessing:
    - Chile data is imped by enabling "car" package"
    - ChileYN dataframe is created by having the values Y and N split from the observations with Chile$vote==Y and N respectively
    - removed "income" from this newly created dataframe
    - missing values are removed
    - Variable ChileYN$vote is adjusted to Factor and the outcome is changed to numeric further to simply keep the values as 0 and 1; 0 = Vote and 1 = No vote (No)
- glm() on ChileYN dataset
    - glm(vote ~ age + statusquo,data=ChileYN,family = binomial(link = "logit")) is stored on a variable "chile.glm"
    - above code runs a logistic regression analysis on vote as the dependent variable and age and statusquo as the predictors from "ChileYN" dataset.
    - "binomial" link function changes the output of the glm model to inverse logit or logistic regression
- Diagnostics:
    - From the glm(vote ~ age + statusquo,data=ChileYN,family = binomial(link = "logit")) ; this formula predicts "vote" values from age and statusquo combined from the dataset "ChileYN"
    - Procedure glm() is similar to lm() procedure and expects the dependent variable (variable, in question for prediction) first. Followed by, independent/prdictor variable and a link function. It can have number of predictor variables follows after "~" symbol. "." after "~" means - it expects to include all the remaining variables from the dataset. In this case, we are only trying to predict "vote" rate based on "Age" & "Statusquo" variables."binomial" is the link function changes the output of the glm model to inverse logit or logistic regression
    - In addition, "data=ChileYN" implies what is the sample/population the prediction is run against from the observations it contains.
    - Successul execution of the glm() procedure provides results as shown below.

        - call
        - Deviance Residuals
        - Coefficients
        - Significant codes
        - Null and Residual deviance with degrees of freedom
        - Number of Fisher Scoring iterations
    - Chi-Square analysis on logistic regression
        - anova(chile.glm, test=“Chisq”) -performs chi-square analysis on the glm() output
    - Convert the log odds for the coefficient on the predictor into regular odds
        - exp(coef(chile.glm)) - converts log odds into regular odds
    - Confusion matrix
        - table(round(predict(chile.glm,type=“response”)),ChileYN$vote) - this formula creates confusion marix
    - Null hypothesis
    - Run bayesian analysis on top the glm output
        - bayesLogitOut <- MCMClogit(formula = vote ~ age + statusquo, data = ChileYN) formula generates BayesLogit output
    - Confirm alternate hypothesis and its test of significance on the dependent variables.
    - Run AIC procedure
        - stepAIC(chile.glm) formula generates AIC output
        - Compare it against old AIC on age+income predictors
- Results:
    - Once the glm() procedure executes successfuly, it returns various data points as an outcome. The first two lines defines the model, we wanted.
        - glm(formula = vote ~ age + statusquo, family = binomial(link = “logit”), data = ChileYN)
        - By specifying binomial() - it invokes the inverse logit or logistic function as the basis for fitting the X variables to the Y variable.
    - Next, Summary of residuals that gives an overview of errors of prediction.
        - With min as -3.2095 and max of 2.8789 shows, distribution of residuals between -ve to positive almost spreading equally on both sides, with Median almost 0 (-0.1840).
        - It seems the residuals are symmetrically distributed.
        - hist(residuals(chile.glm)) suggests the same.
    - Coefficients shows the key results.
        - Intercept is at -0.193759.
        - Slopes for age variable is 0.011322 and statusquo is 3.174487 are way off from the intercept or B-weights. These coefficients define the logrithm of the odds of the Y variable.
        - Std.Errors around the estimates of slope and intercept shows the estimated sampling distribution around these point estimates.
        - z-value shows the student’s t-test of the null hypothesis test that each estimated coefficients is equal to zero.
        - With above p-value; * age has the weak coefficient value as 0.011322 higher than the test of significance ($p < 0.01$) at 0.0972 * statusquo has p-value as 0.0000000000000002 far less than the test of significance ($p < 0.001$) This shows that , “statusquo” as statistically significant where age as not significant.
        - Null Hypothesis * Null hypothesis is that the log odds of vote is equal to 0 in the

population. Since the log odds of statusquo is statistically significant and less than thier respective alpha level - we can reject the null hypothesis. * However, age not having the stronger significance , we fail to reject null hypothesis with Age as predictor for Vote.
    - In addition, the conversion from log odds to regular output (exp(coef(chile.glm))), (Intercept) age statusquo 0.8238564 1.0113863 23.9145451 From above its is infered that 1.0113863:1 on age and 23.9145451:1 on statusquo to likely to claim vote. Showing stronger odds on statusquo to predict vote
    - The first Chi-Square model compares three nested models.
        - anova(chile.glm, test="Chisq") - includes both predictors and tests the level of significance on these predictors.
        - It confirms that both predictors with 0.000000004964 (age) and 0.0000000000000000022(statusquo) is far less the the p value (0.001) and they are statistically significant.
        - 34.2 is the chi-square value (residual deviance from top line - residual deviance from 2nd line) 2360.29 - 2326.09 = 34.2 is tested for significance on one degree of freedom.
- Bayesian Logit * MCMCPack (MCMClogit) does the the bayesian estimation of logistic regression * bayesLogitOut <- MCMClogit(formula = vote ~ age + statusquo, data = ChileYN) formula is used to simulate Bayesian estimation * Wih sample size of 10,000 observations - population mean of the standard errors * With earlier pre-processing the depedent variable is converted into No=0 and Yes = 1. * Output contains the posterior distribution of parameters representing both intercept and cofficients on age and statusquo calibrated as log-odds. * Point extimates for the intercept and the coefficients are similar to outputs from the logistic regression. * 95% HDI interval shows no overlap on age and statusquo variables. * 95% HDI shows age as significant as it overlaps with 0. * more detailed analysis on the HDI is shown on the trace chart below and it captures extensive information about the alternative hypothesis for each of the coefficients being estimated. * density curve on age is spread across 0 and HDI lower bound (2.5%) at -0.002005 and uppoer bound (97.5%) at 0.02499. * density curve on statusquo shows HDI well over 0 with HDI lower bound (2.5%) at 2.914442 and Upper bound (97.5%) at 3.48698. favoring alternative hypothesis.
- AIC is calculated at 740.5207 on age + statusquo Step Df Deviance Resid. Df Resid. Dev AIC 1 1700 734.5207 740.5207
- AIC is at 2330.1 on age + income; with two step process Step Df Deviance Resid. Df Resid. Dev AIC 1 1700 2326.029 2332.029 2 - income 1 0.06212372 1701 2326.091 2330.091

Please find more details from the R code below,

```
options(scipen=999)  # turn-off scientific notation like 1e+48

# load the necessary library for further processing...
EnsurePackage("car") # Regression helper package: Chile data
```

```
## Loading required package: car
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```
EnsurePackage("MCMCpack") # Download MCMCpack package
```

```
## Loading required package: MCMCpack
```

```
## Loading required package: coda
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
## ##
## ## Markov Chain Monte Carlo Package (MCMCpack)
```

```
## ## Copyright (C) 2003-2020 Andrew D. Martin, Kevin M. Quinn, and Jong Hee Park
```

```
## ##
## ## Support provided by the U.S. National Science Foundation
```

```
## ## (Grants SES-0350646 and SES-0350613)
## ##
```

```
EnsurePackage("dlookr") # outlier analysis
```

```
## Loading required package: dlookr
```

```
## Loading required package: mice
```

```
##
## Attaching package: 'mice'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
##
## Attaching package: 'dlookr'
```

```
## The following object is masked from 'package:base':
##
##     transform
```

```
EnsurePackage("mice") # missing data
EnsurePackage("visdat") # missing data
```

```
## Loading required package: visdat
```

```
cat("All Packages are available")
```

```
## All Packages are available
```

```
#import Chile dataset
data(Chile)

#structure of Chile dataset
str(Chile)
```

```
## 'data.frame':    2700 obs. of  8 variables:
##  $ region    : Factor w/ 5 levels "C","M","N","S",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ population: int  175000 175000 175000 175000 175000 175000 175000 175000 175000 
175000 ...
##  $ sex       : Factor w/ 2 levels "F","M": 2 2 1 1 1 1 2 1 1 2 ...
##  $ age       : int  65 29 38 49 23 28 26 24 41 41 ...
##  $ education : Factor w/ 3 levels "P","PS","S": 1 2 1 1 3 1 2 3 1 1 ...
##  $ income    : int  35000 7500 15000 35000 35000 7500 35000 15000 15000 15000 ...
##  $ statusquo : num  1.01 -1.3 1.23 -1.03 -1.1 ...
##  $ vote      : Factor w/ 4 levels "A","N","U","Y": 4 2 4 2 2 2 2 2 3 2 ...
```

```
# Summary of Chile dataset
summary(Chile)
```

```
##  region     population      sex           age          education  
##  C :600   Min.   :  3750   F:1379   Min.   :18.00   P   :1107  
##  M :100   1st Qu.: 25000   M:1321   1st Qu.:26.00   PS  : 462  
##  N :322   Median :175000            Median :36.00   S   :1120  
##  S :718   Mean   :152222            Mean   :38.55   NA's:  11  
##  SA:960   3rd Qu.:250000            3rd Qu.:49.00              
##           Max.   :250000            Max.   :70.00              
##                                     NA's   :1                  
##      income         statusquo          vote    
##  Min.   :  2500   Min.   :-1.80301   A   :187  
##  1st Qu.:  7500   1st Qu.:-1.00223   N   :889  
##  Median : 15000   Median :-0.04558   U   :588  
##  Mean   : 33876   Mean   : 0.00000   Y   :868  
##  3rd Qu.: 35000   3rd Qu.: 0.96857   NA's:168  
##  Max.   :200000   Max.   : 2.04859             
##  NA's   :98       NA's   :17
```

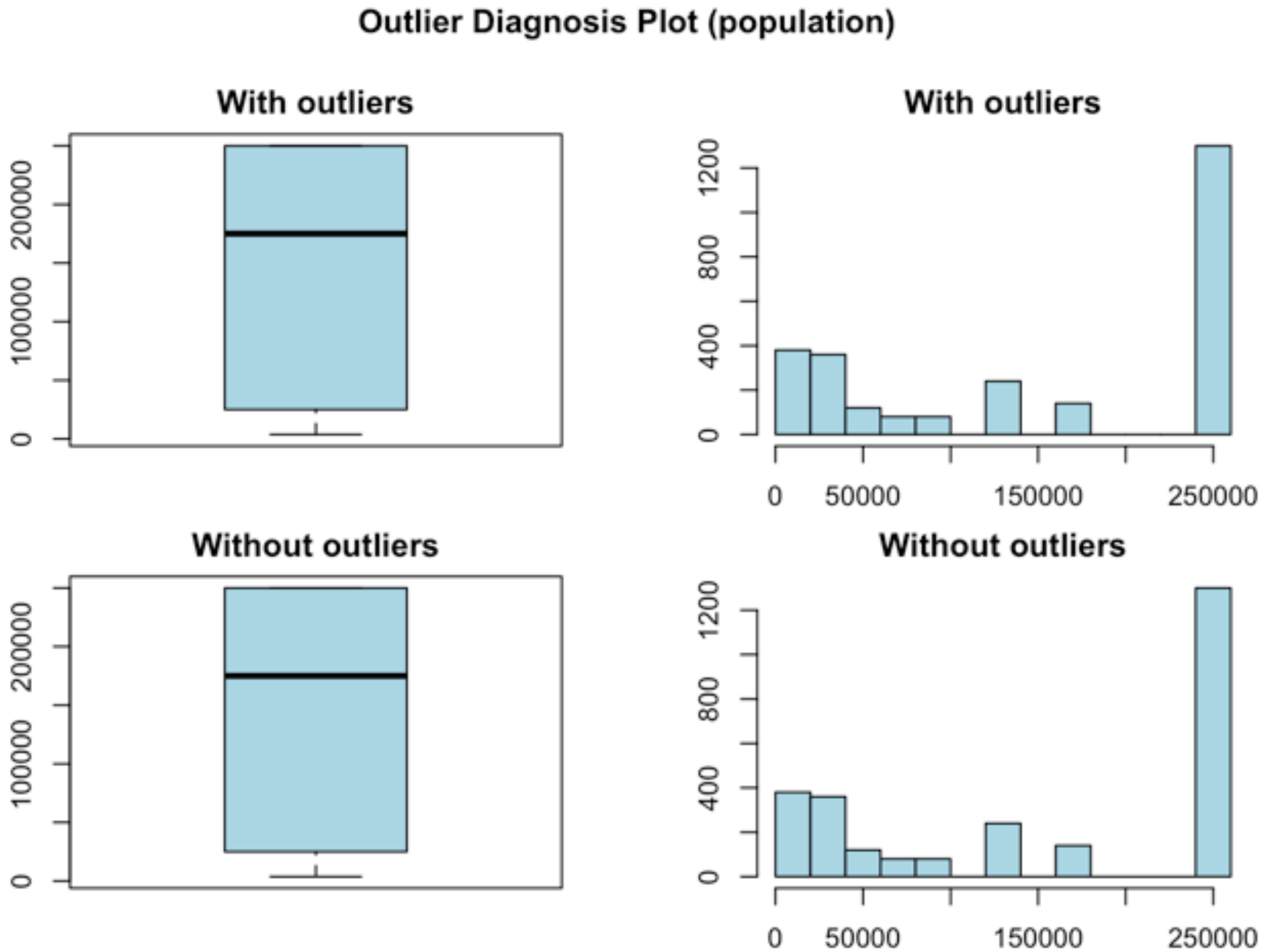
```
#outlier analysis
diagnose_outlier(Chile)
```

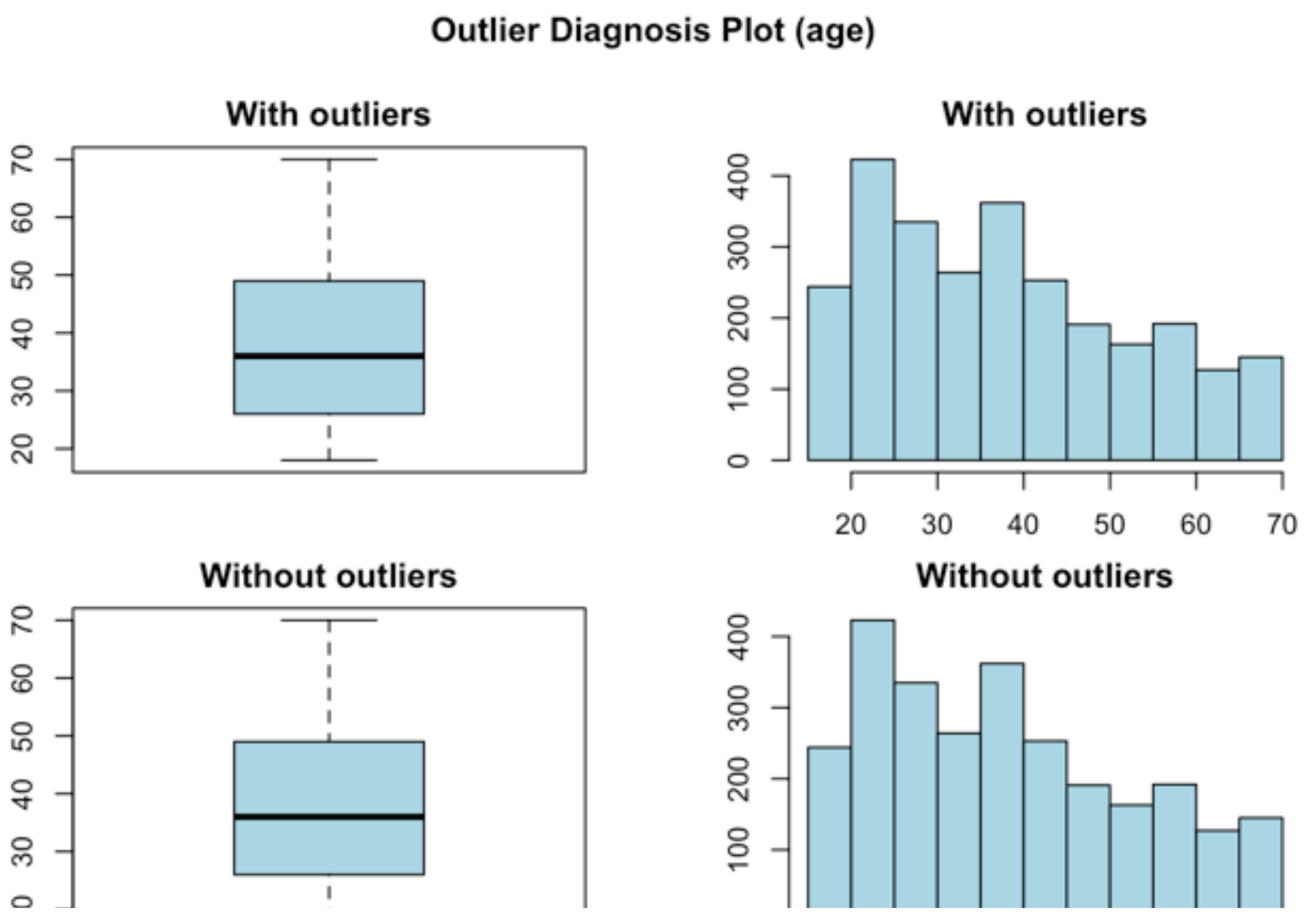
variables <chr>	outliers_cnt <int>	outliers_ratio <dbl>	outliers_mean <dbl>	with_mean <dbl>	
population	0	0.000000	NaN	152222.22222222221899	1522
age	0	0.000000	NaN	38.54872174879585	

income	164	6.074074	159756.1	33875.86471944658115	25
statusquo	0	0.000000	NaN	-0.00000001118151	

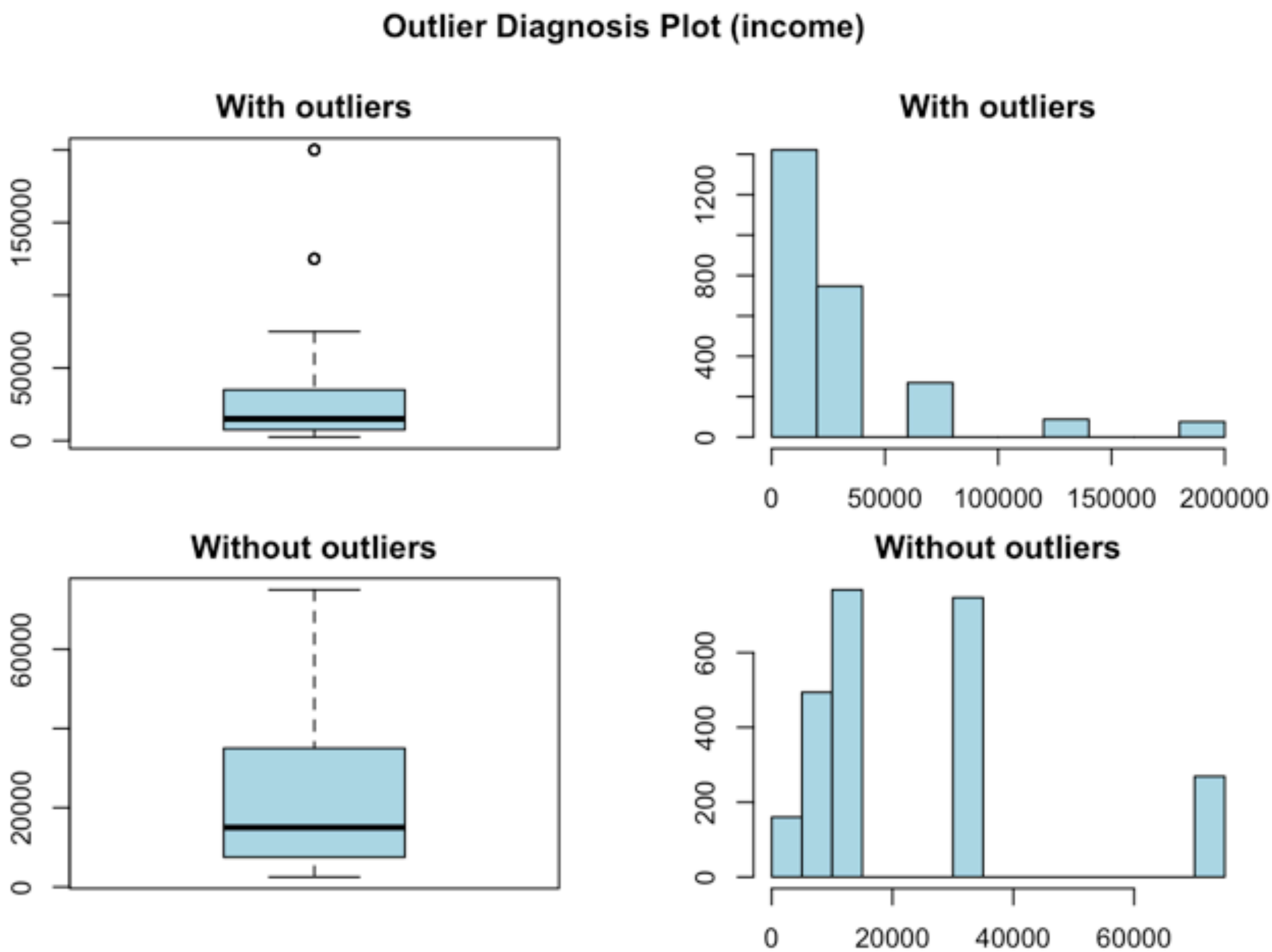
4 rows

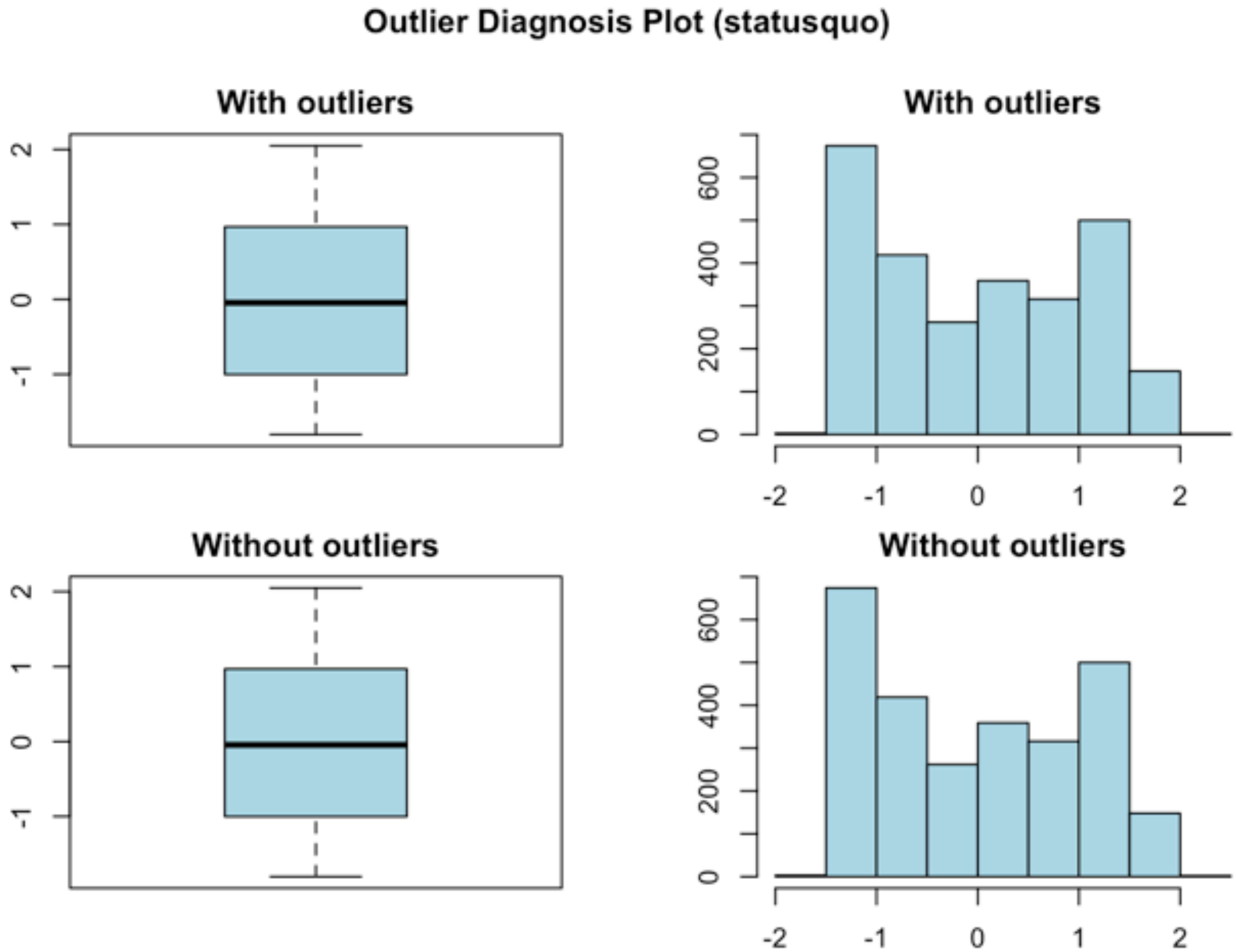
```
plot_outlier(Chile)
```


Outlier Diagnosis Plot (population)
With outliers
With outliers
Without outliers
Without outliers
0
100000
200000
0
400
800
1200
0
50000
150000
250000

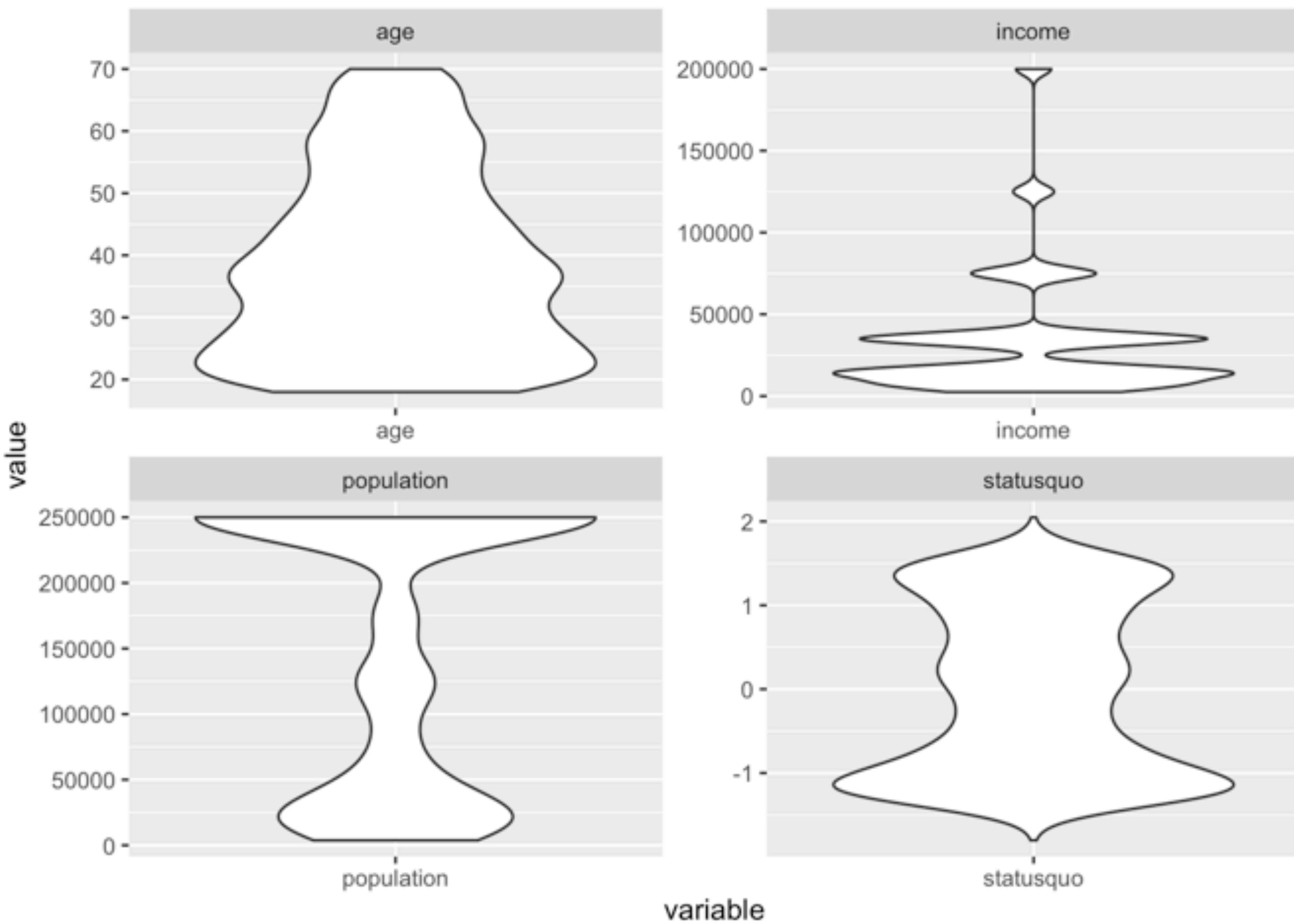

Outlier Diagnosis Plot (age)
With outliers
With outliers
Without outliers
Without outliers
20
30
40
50
60
70
0
100
200
300
400


2
0
20
30
40
50
60
70


Outlier Diagnosis Plot (income)
With outliers
150000
50000
0
With outliers
1200
800
400
0
0
50000
100000
150000
200000
Without outliers
60000
20000
0
Without outliers
600
400
200
0
0
20000
40000
60000



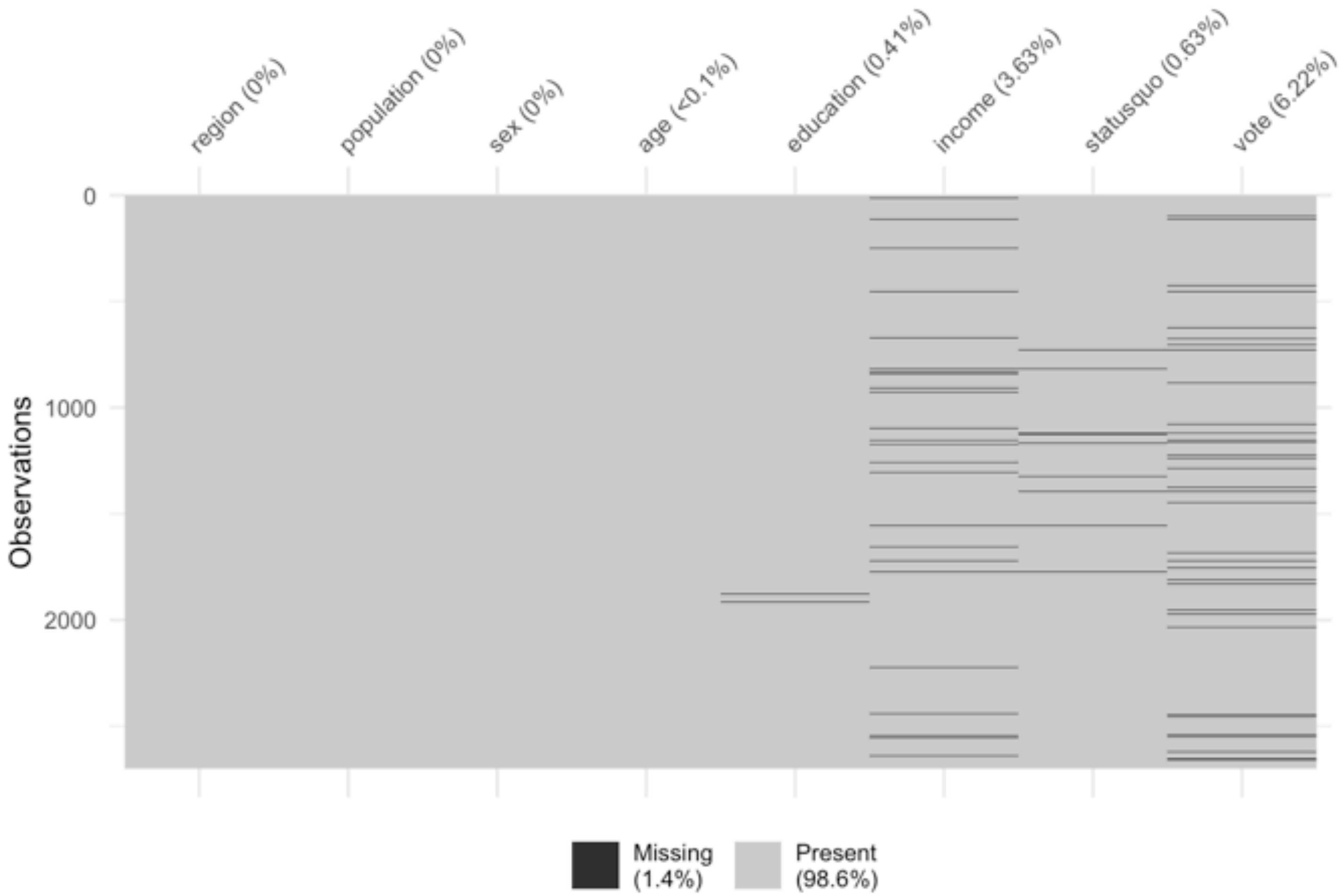

```
#data distribution
Chile %>%
  pivot_longer(cols=-c(region,sex,vote,education), names_to="variable", values_to="va
lue", values_drop_na = TRUE) %>%
  ggplot(aes(x=variable, y=value)) + geom_violin() + facet_wrap( ~ variable, scales="
free")
```




```
# missing data
md.pattern(Chile, plot=FALSE)
```

```
##      region population sex age education statusquo income vote
## 2431      1          1   1   1         1         1      1    1   0
## 150       1          1   1   1         1         1      1    0   1
## 77        1          1   1   1         1         1      0    1   1
## 14        1          1   1   1         1         1      0    0   2
## 8         1          1   1   1         1         0      1    1   1
## 3         1          1   1   1         1         0      1    0   2
## 5         1          1   1   1         1         0      0    1   2
## 9         1          1   1   1         0         1      1    1   1
## 1         1          1   1   1         0         1      0    1   2
## 1         1          1   1   1         0         0      0    0   4
## 1         1          1   1   0         1         1      1    1   1
##           0          0   0   1        11        17     98  168 295
```

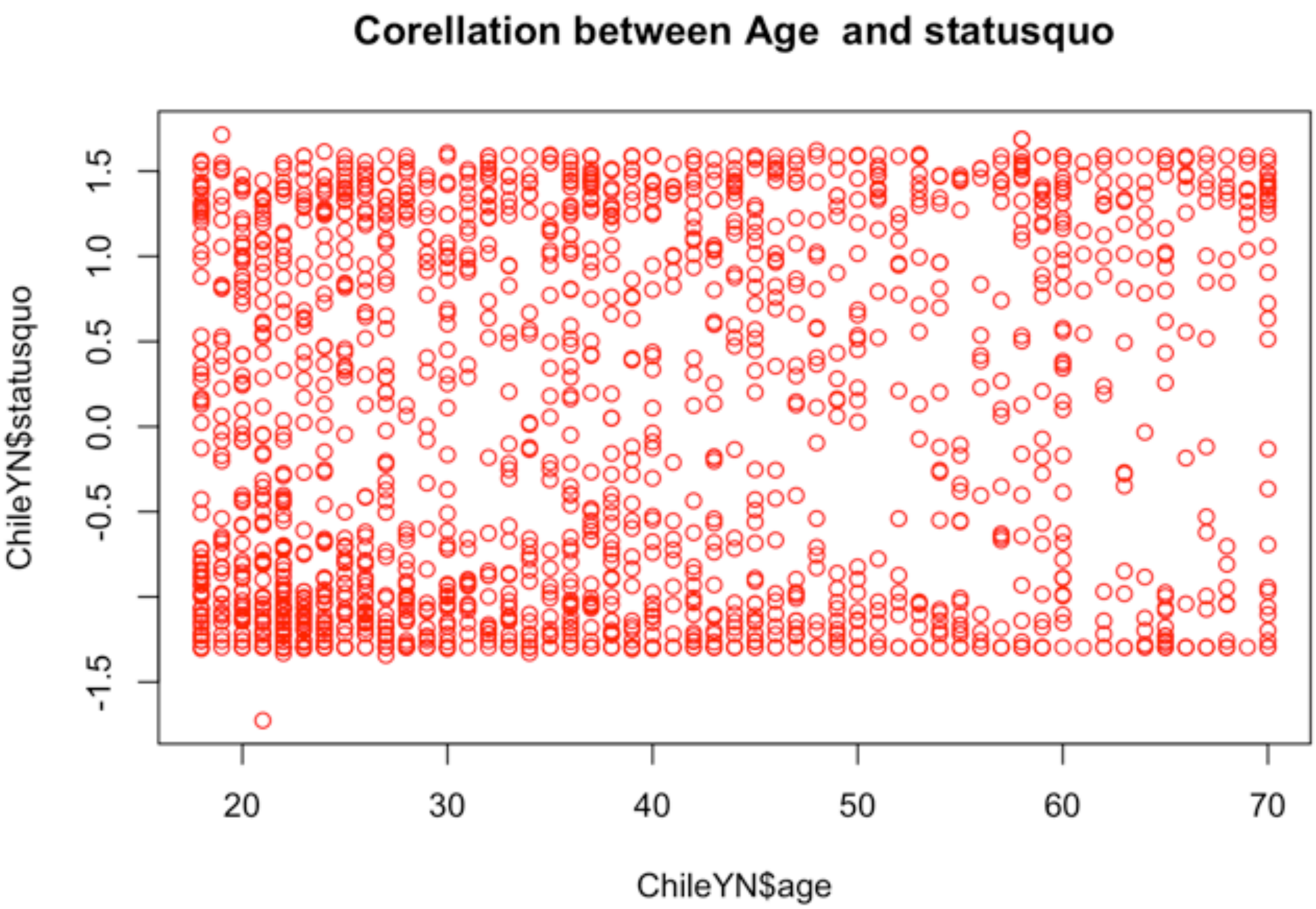
```
# missing data visualization
vis_miss(Chile)
```




```
options(scipen=999)  # turn-off scientific notation like 1e+48

ChileY <- Chile[Chile$vote == "Y",] # Grab the Yes votes
ChileN <- Chile[Chile$vote == "N",] # Grab the No votes
ChileYN <- rbind(ChileY,ChileN) # Make a new dataset with those
ChileYN <- ChileYN[complete.cases(ChileYN),] # Get rid of missing
ChileYN$vote <- factor(ChileYN$vote,levels=c("N","Y")) # Fix the factor

# Corellation between Agriculture and Fertility
plot(ChileYN$age, ChileYN$statusquo
     ,main="Corellation between Age  and statusquo"
     ,col="red")
```




```
str(ChileYN)
```

```
## 'data.frame':    1703 obs. of  8 variables:
##  $ region    : Factor w/ 5 levels "C","M","N","S",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ population: int  175000 175000 175000 175000 175000 175000 175000 175000 175000
175000 ...
##  $ sex       : Factor w/ 2 levels "F","M": 2 1 2 1 2 1 1 2 1 2 ...
##  $ age       : int  65 38 64 46 67 38 55 18 24 58 ...
##  $ education : Factor w/ 3 levels "P","PS","S": 1 1 1 3 1 3 2 3 2 1 ...
##  $ income    : int  35000 15000 15000 75000 75000 35000 35000 75000 35000 35000 ..
.
##  $ statusquo : num  1.01 1.23 1.37 1.51 1.32 ...
##  $ vote      : Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 2 ...
```

```
# remove income column
ChileYN <- ChileYN[,!grepl("income",colnames(ChileYN))]
ChileYN$vote <- as.numeric(ChileYN$vote) - 1 # Adjust the outcome
#table(ChileYN$vote)

str(ChileYN)
```

```
## 'data.frame':    1703 obs. of  7 variables:
##  $ region    : Factor w/ 5 levels "C","M","N","S",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ population: int  175000 175000 175000 175000 175000 175000 175000 175000 175000
175000 ...
##  $ sex       : Factor w/ 2 levels "F","M": 2 1 2 1 2 1 1 2 1 2 ...
##  $ age       : int  65 38 64 46 67 38 55 18 24 58 ...
##  $ education : Factor w/ 3 levels "P","PS","S": 1 1 1 3 1 3 2 3 2 1 ...
##  $ statusquo : num  1.01 1.23 1.37 1.51 1.32 ...
##  $ vote      : num  1 1 1 1 1 1 1 1 1 1 ...
```

```
summary(ChileYN$statusquo)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -1.72594 -1.09671 -0.18511 -0.00467  1.16602  1.71355
```

```
summary(ChileYN)
```

```
##  region     population     sex          age        education   statusquo
##  C :374   Min.   :  3750   F:814   Min.   :18.00   P :671    Min.   :-1.72594
##  M : 54   1st Qu.: 25000   M:889   1st Qu.:25.00   PS:343    1st Qu.:-1.09671
##  N :230   Median :175000           Median :36.00   S :689    Median :-0.18511
##  S :476   Mean   :150716           Mean   :38.06             Mean   :-0.00467
##  SA:569   3rd Qu.:250000           3rd Qu.:49.00             3rd Qu.: 1.16602
##           Max.   :250000           Max.   :70.00             Max.   : 1.71355
##       vote
##  Min.   :0.0000
##  1st Qu.:0.0000
##  Median :0.0000
##  Mean   :0.4909
##  3rd Qu.:1.0000
##  Max.   :1.0000
```

```
# missing data
md.pattern(ChileYN, plot=FALSE)
```

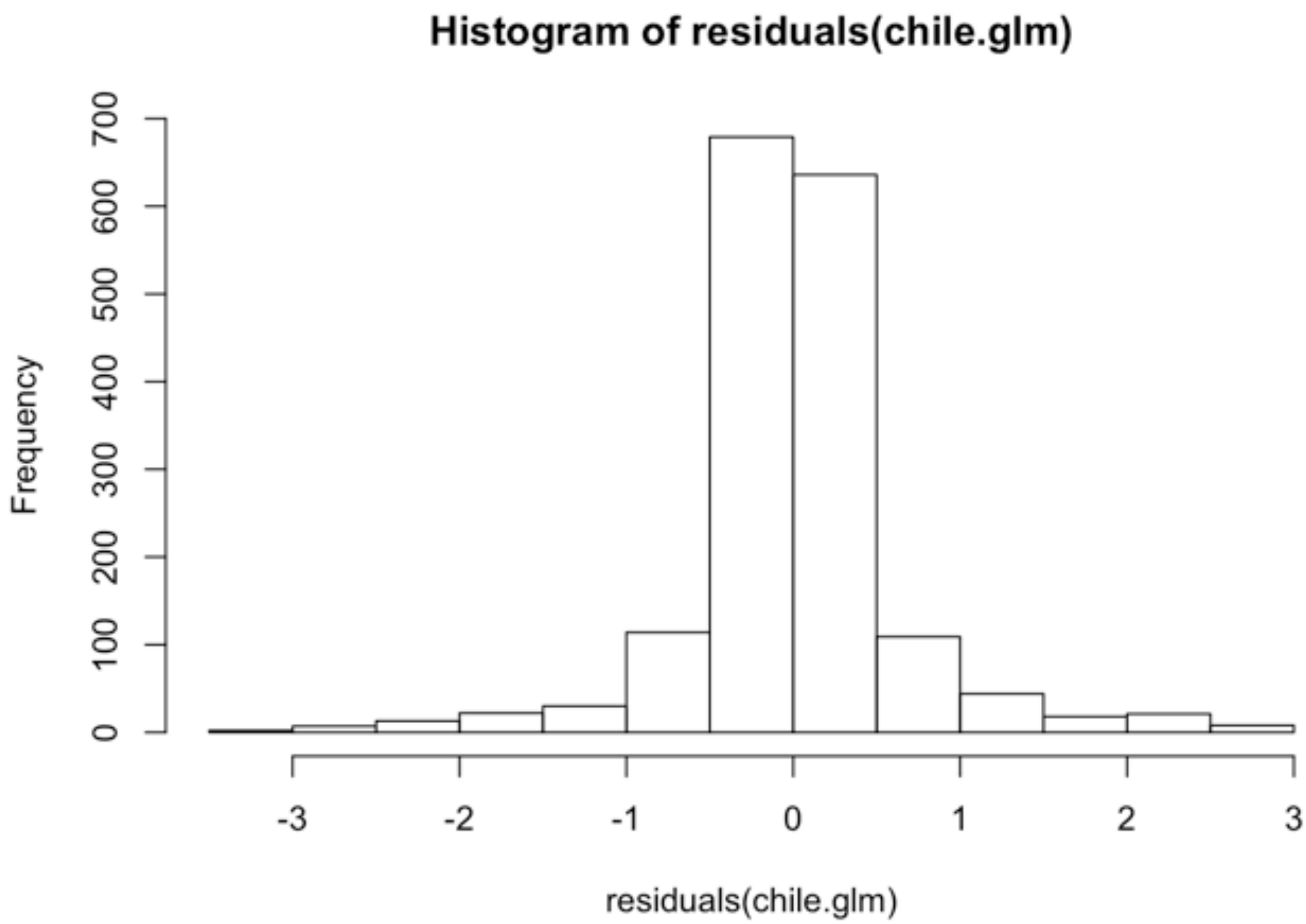
```
##  /\     /\
## {  `---'  }
## {  O   O  }
## ==>  V <==  No need for mice. This data set is completely observed.
##  \  \|/  /
##   `-----'
```

```
##      region population sex age education statusquo vote
## 1703      1          1   1   1         1         1    1 0
##           0          0   0   0         0         0    0 0
```

```
# Crete glm model
chile.glm <- glm(vote ~ age + statusquo,data=ChileYN,family = binomial(link = "logit"
))
summary(chile.glm)
```

```
##
## Call:
## glm(formula = vote ~ age + statusquo, family = binomial(link = "logit"),
##     data = ChileYN)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.2095  -0.2830  -0.1840   0.1889   2.8789
##
## Coefficients:
##              Estimate Std. Error z value            Pr(>|z|)
## (Intercept) -0.193759   0.270708  -0.716              0.4741
## age          0.011322   0.006826   1.659              0.0972 .
## statusquo    3.174487   0.143921  22.057 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2360.29  on 1702  degrees of freedom
## Residual deviance:  734.52  on 1700  degrees of freedom
## AIC: 740.52
##
## Number of Fisher Scoring iterations: 6
```

```
#histogram on residuals chile.glm
hist(residuals(chile.glm))
```




```
#Chi-Square analysis on logistic regression
anova(chile.glm, test="Chisq")
```

	Df <int>	Deviance <dbl>	Resid. Df <int>	Resid. Dev <dbl>	Pr(>Chi) <dbl>
NULL	NA	NA	1702	2360.2950	NA
age	1	34.20349	1701	2326.0915	0.000000004963989
statusquo	1	1591.57079	1700	734.5207	0.000000000000000

3 rows

```
# Convert the log odds for the coefficient on the predictor into regular odds
exp(coef(chile.glm))
```

```
## (Intercept)         age   statusquo
##   0.8238564   1.0113863  23.9145451
```

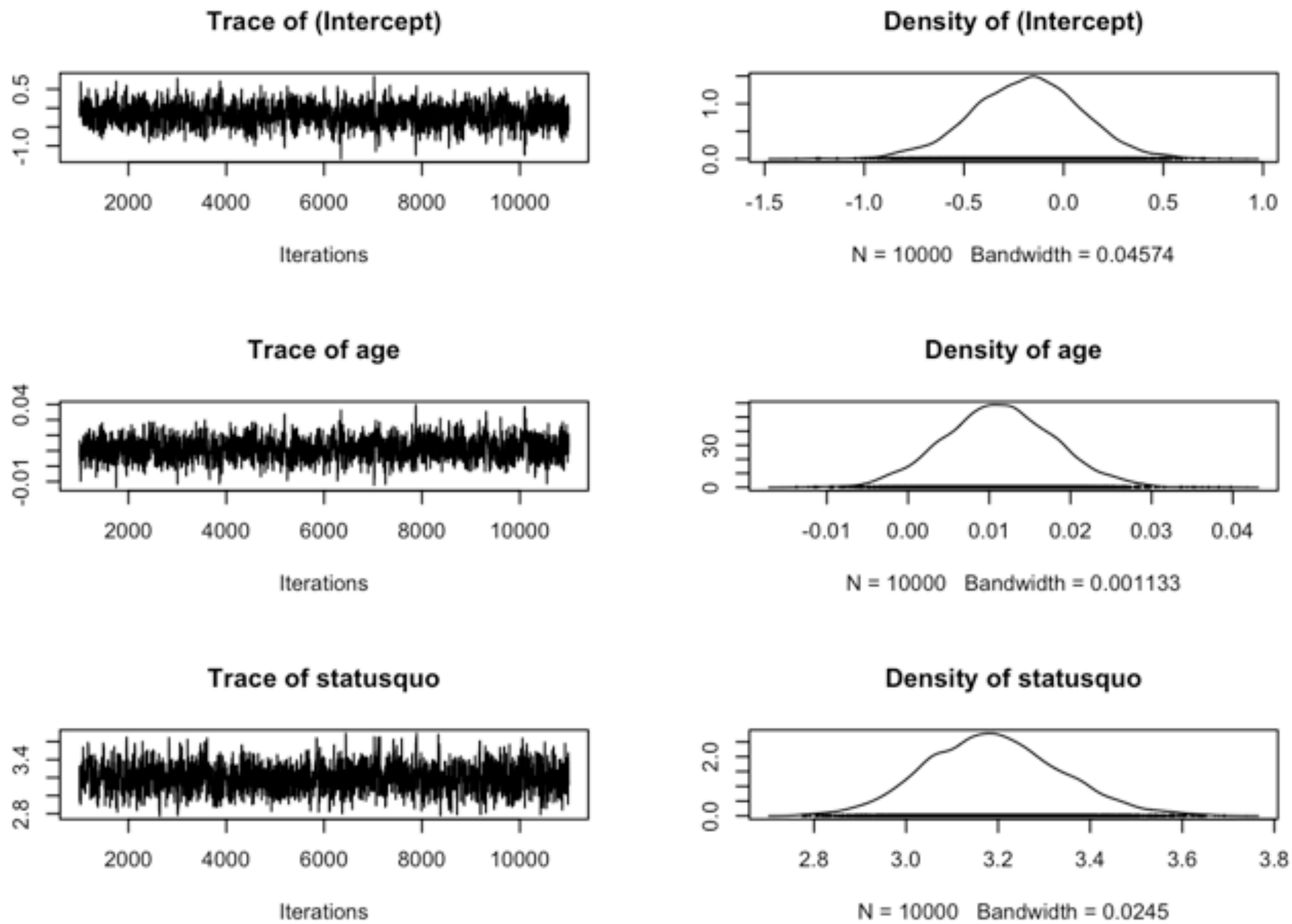
```r
#confusion matrix
table(round(predict(chile.glm,type="response")),ChileYN$vote)
```

```
##
##       0   1
##   0 810  74
##   1  57 762
```

```r
set.seed(271) # Control randomization
#bayesian estimation of logistic regression
bayesLogitOut <- MCMClogit(formula = vote ~ age + statusquo, data = ChileYN)
summary(bayesLogitOut) # Summarize the results
```

```
##
## Iterations = 1001:11000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 10000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##                 Mean       SD   Naive SE Time-series SE
## (Intercept) -0.18272 0.272640 0.00272640       0.008938
## age          0.01123 0.006817 0.00006817       0.000223
## statusquo    3.19061 0.145853 0.00145853       0.004993
##
## 2. Quantiles for each variable:
##
##                  2.5%       25%      50%        75%   97.5%
## (Intercept) -0.742761 -0.365241 -0.17552 -0.0003872 0.34439
## age         -0.002005  0.006733  0.01121  0.0157683 0.02499
## statusquo    2.914442  3.087259  3.18546  3.2847388 3.48698
```

```r
plot(bayesLogitOut)
```




```
EnsurePackage("MASS") # AIC

stepOut <- stepAIC(chile.glm)
```

```
## Start:  AIC=740.52
## vote ~ age + statusquo
##
##              Df Deviance     AIC
## <none>            734.52  740.52
## - age         1   737.27  741.27
## - statusquo   1  2326.09 2330.09
```

```
stepOut$anova
```

Step	Df	Deviance	Resid. Df	Resid. Dev	AIC
<fctr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
	NA	NA	1700	734.5207	740.5207

1 row

```
# stepOutOLD <- stepAIC(chout)
# stepOutOLD$anova
```

Chapter 10, Exercise 7

Bonus R code question: Develop your own custom function that will take the posterior distribution of a coefficient from the output object from an MCMClogit() analysis and automatically create a histogram of the posterior distributions of the coefficient in terms of regular odds (instead of log-odds). Make sure to mark vertical lines on the histogram indicating the boundaries of the 95% HDI. (1 pt) Run the function on your regression results. (1 pt)

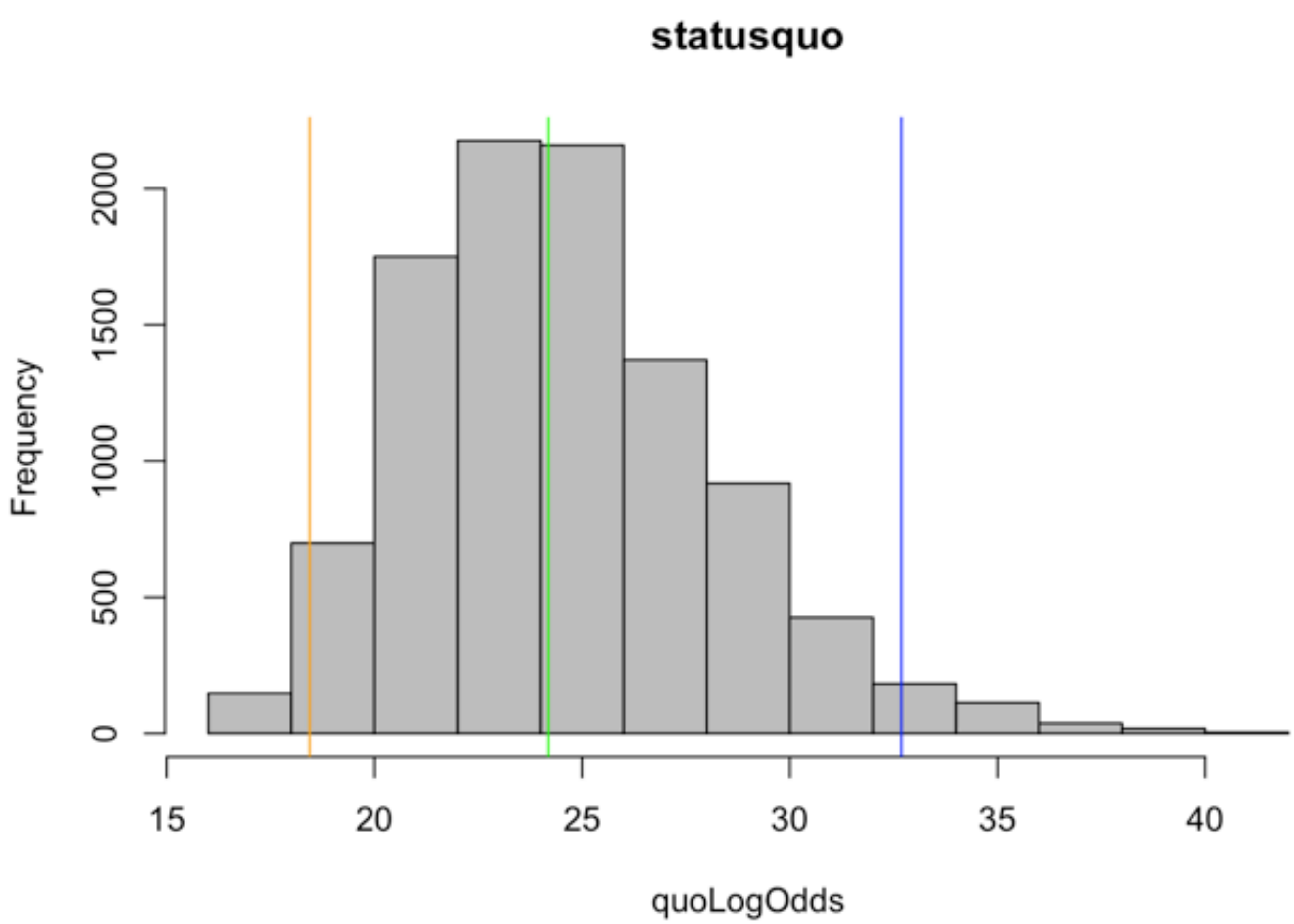
- Custom Function:
    - histquoLogOdds takes one of the predictor variable output (log odds) as input
    - converts the log-odds into regular odds with exponential function as exp(quoLogOdds)
    - hist(quoLogOdds, main=prdictor,col="grey") - procedure creates histogram with HDI lines
    - HDI lines
        - 2.5% or HDI lower bound | abline(v=quantile(quoLogOdds,c(0.025)),col="orange")
        - 97.5% or HDI upper bound | abline(v=quantile(quoLogOdds,c(0.975)),col="blue")
        - median or 50% percent quantile | abline(v=quantile(quoLogOdds,c(0.50)),col="green")
- Output of the function produces histograms
    - histquoLogOdds("statusquo") - histogram on statusquo regular odds
        - Status histograms has values between~15 to ~34 as lower and upper bounds
    - histquoLogOdds("age") - histogram on age regular odds
        - Age histograms has more consistent values between just below 1 and over 1.

Please find more details from the R code below,

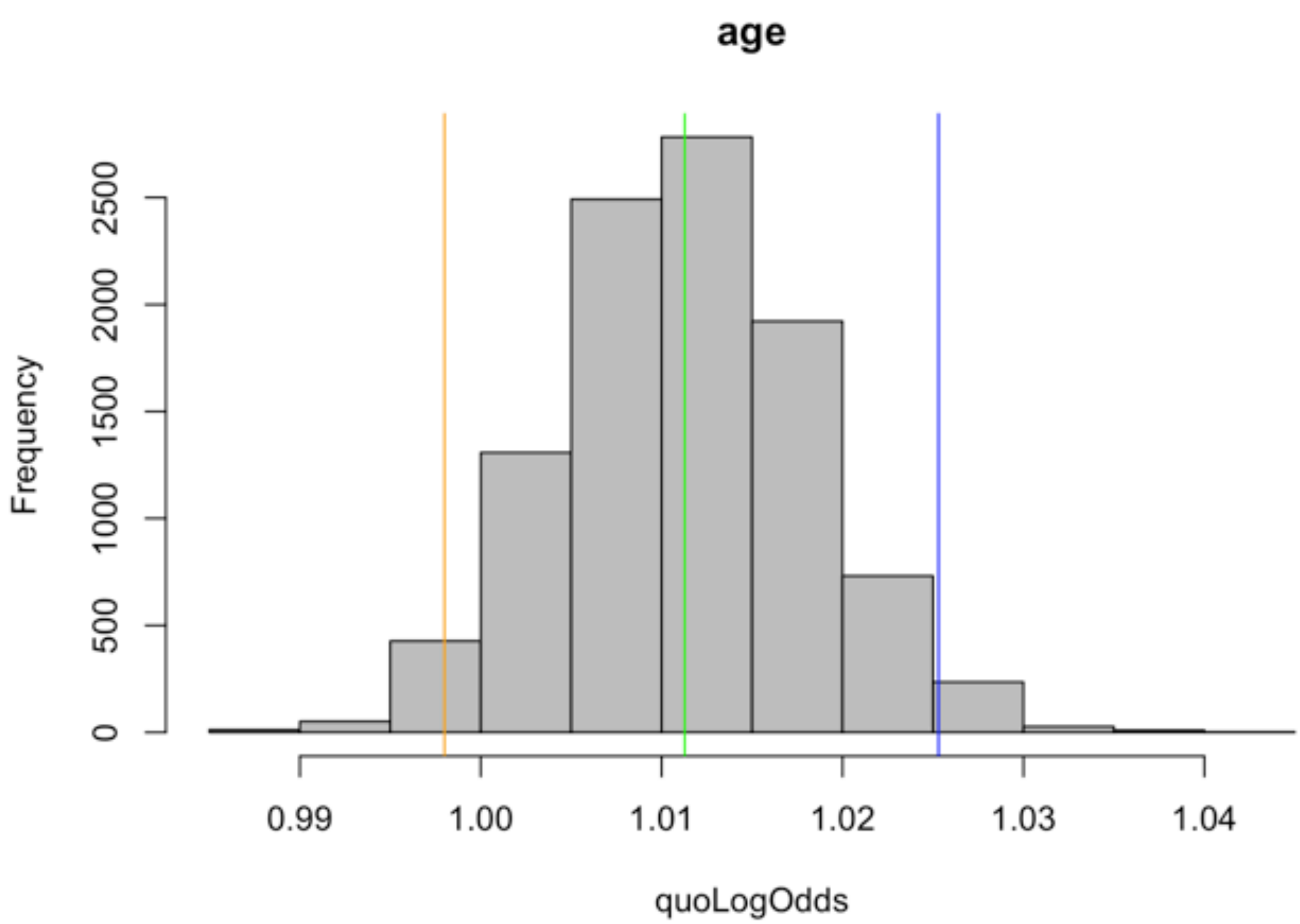
```
# bayesLogitOut hist

# Custom function to output histograms with HDI vertical lines
# prdictor bayesLogitOut - output variable coeffients converted to regular odds (inst
ead of log-odds)
histquoLogOdds <- function(prdictor)
  {
  quoLogOdds <- as.matrix(bayesLogitOut[,prdictor])
  quoLogOdds <- exp(quoLogOdds)  # regular odds (instead of log-odds)
  hist(quoLogOdds, main=prdictor,col="grey") # hist
  abline(v=quantile(quoLogOdds,c(0.025)),col="orange")
  abline(v=quantile(quoLogOdds,c(0.975)),col="blue")
  abline(v=quantile(quoLogOdds,c(0.50)),col="green")
  }

#Plot histograms
histquoLogOdds("statusquo")
```




```
histquoLogOdds("age")
```


age
Frequency
0
500
1000
1500
2000
2500
0.99
1.00
1.01
1.02
1.03
1.04
quoLogOdds