Logistic regression with R

Fani Apostolidou Kiouti

23/01/2022

Contents

Packages in use

```
library(ggplot2)
library(dplyr);library(tidyr);library(magrittr)
library(epiDisplay)
library(gtsummary);library(gt)
```

1 Explore data and convert data types

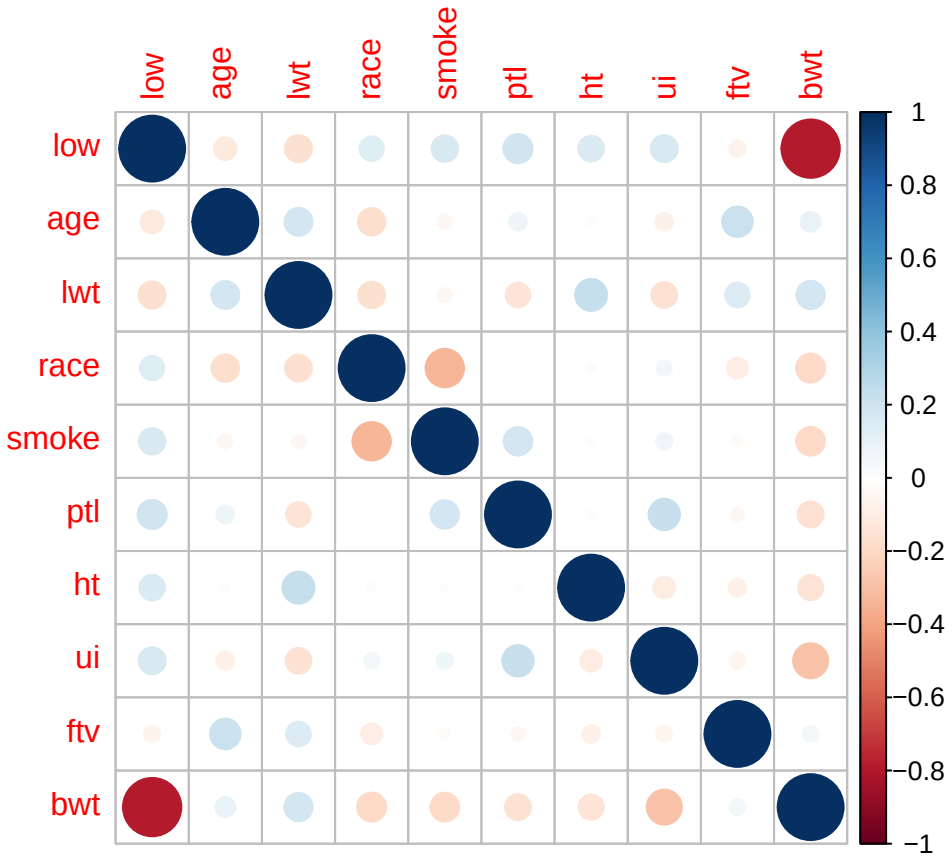
The following example is based on the `birthwt` data frame from `{MASS}`. It consists of 189 observations and 10 columns. Data were collected at Baystate Medical Center, Springfield, Mass during 1986 and was originally used in Hosmer and Lemeshow's book 'Applied Logistic Regression'. The following variables are recorded:

Variable name	Information
low	indicator of birth weight less than 2.5 kg
age	mother's age in years
lwt	mother's weight in pounds at last menstrual period
race	mother's race (1 = white, 2 = black, 3 = other)
smoke	smoking status during pregnancy
ptl	number of previous premature labours
ht	history of hypertension
ui	presence of uterine irritability
ftv	number of physician visits during the first trimester
bwt	birth weight in grams

```
# load dataset
glimpse(MASS::birthwt)
```

```
## Rows: 189
## Columns: 10
## $ low   <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
## $ age   <int> 19, 33, 20, 21, 18, 21, 22, 17, 29, 26, 19, 19, 22, 30, 18, 18, ~
## $ lwt   <int> 182, 155, 105, 108, 107, 124, 118, 103, 123, 113, 95, 150, 95, 1~
## $ race  <int> 2, 3, 1, 1, 1, 3, 1, 3, 1, 1, 3, 3, 3, 3, 1, 1, 2, 1, 3, 1, 3, 1~
## $ smoke <int> 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0~
## $ ptl   <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0~
## $ ht    <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0~
## $ ui    <int> 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1~
## $ ftv   <int> 0, 3, 1, 2, 0, 0, 1, 1, 1, 0, 0, 1, 0, 2, 0, 0, 0, 3, 0, 1, 2, 3~
## $ bwt   <int> 2523, 2551, 2557, 2594, 2600, 2622, 2637, 2637, 2663, 2665, 2722~
```

```
# create a correlation plot: note that low and bwt are essentially the same variable
corrplot::corrplot(cor(MASS::birthwt))
```




```
# concerting into factors
lbw <- MASS::birthwt %>%
  mutate(low.b = factor(low, levels = c(0,1), labels = c("Normal", "Underweight"))) %>%
  mutate(race = factor(race, levels = c(1,2,3), labels = c("White", "Black", "Other"))) %>%
  mutate(smoke = factor(smoke, levels = c(0,1), labels = c("No", "Yes"))) %>%
  mutate(ht = factor(ht, levels = c(0,1), labels = c("Normal", "Hypertension"))) %>%
  mutate(ui = factor(ui, levels = c(0,1), labels = c("Normal", "Urinary irritability")))
glimpse(lbw)
```

```
## Rows: 189
## Columns: 11
## $ low   <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
## $ age   <int> 19, 33, 20, 21, 18, 21, 22, 17, 29, 26, 19, 19, 22, 30, 18, 18, ~
## $ lwt   <int> 182, 155, 105, 108, 107, 124, 118, 103, 123, 113, 95, 150, 95, 1~
## $ race  <fct> Black, Other, White, White, White, Other, White, Other, White, W~
## $ smoke <fct> No, No, Yes, Yes, Yes, No, No, No, Yes, Yes, No, No, No, No, Yes~
## $ ptl   <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0~
## $ ht    <fct> Normal, Normal, Normal, Normal, Normal, Normal, Normal, Normal, ~
## $ ui    <fct> Urinary irritability, Normal, Normal, Urinary irritability, Urin~
## $ ftv   <int> 0, 3, 1, 2, 0, 0, 1, 1, 1, 0, 0, 1, 0, 2, 0, 0, 0, 3, 0, 1, 2, 3~
## $ bwt   <int> 2523, 2551, 2557, 2594, 2600, 2622, 2637, 2637, 2663, 2665, 2722~
## $ low.b <fct> Normal, Normal, Normal, Normal, Normal, Normal, Normal, Normal, ~
```

To fit a logistic regression model in R the `glm()` function is used with the argument `family` set to `binomial()`:

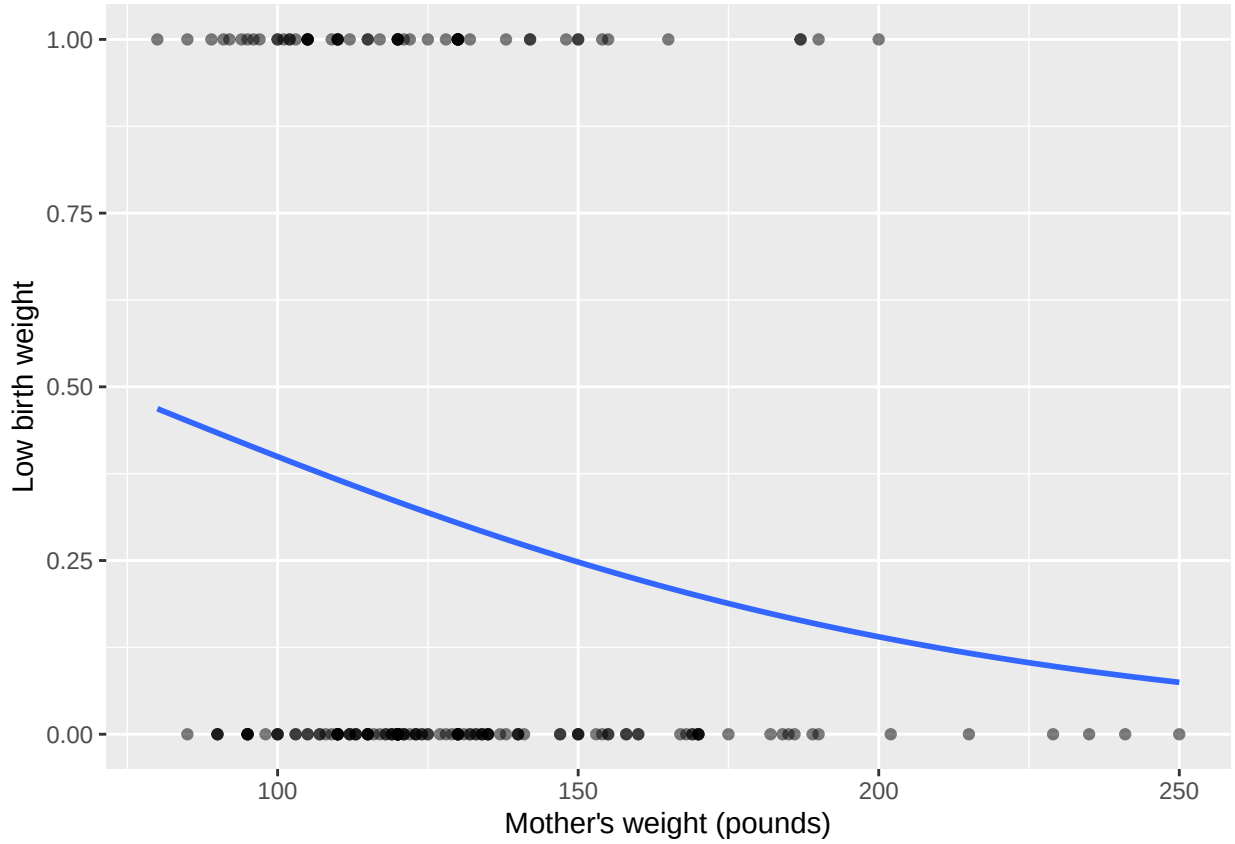
```
model <- glm(dependent ~ independent, data = data.name, family = binomial())
```

2 Define model

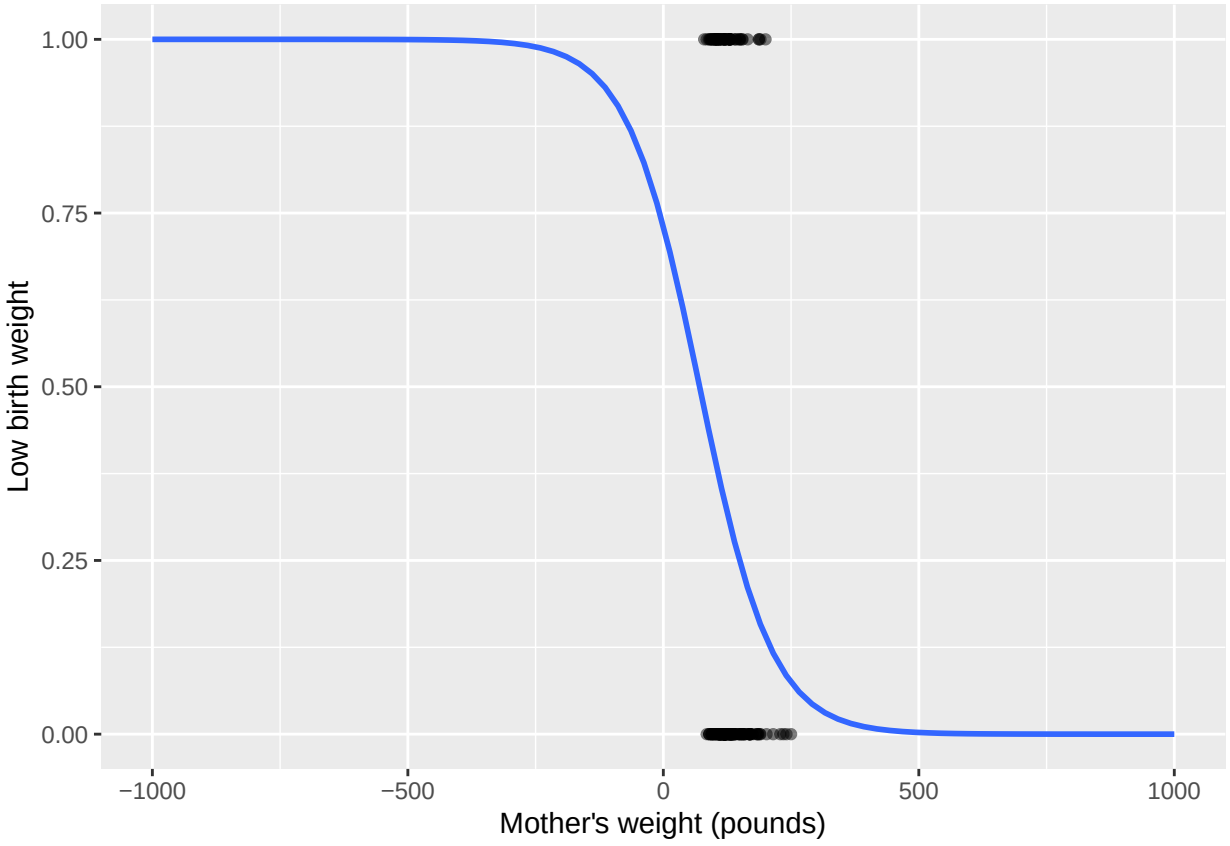
To proceed to assessment of model prerequisites, first define the model. In this example, fit `low` for `lwt`:

```
mod_lwt <- glm(low.b ~ lwt, data = lbw, family = binomial)

  ### draw a sigmoid curve for the model
    # probability is modeled as a fraction that ranges from 0 to 1;
    # subtract 1 from the converted values as follows
ggplot(lbw, aes(x = lwt, y = as.numeric(low.b) - 1)) +
  geom_point(alpha = .5) +
  stat_smooth(method = "glm", se = FALSE, fullrange = TRUE,
              method.args = list(family = binomial)) +
  ylab("Low birth weight") + xlab("Mother's weight (pounds)")
```




```
# A full sigmoid curve for the logistic model is obtained by increasing range
ggplot(lbw, aes(x = lwt, y = as.numeric(low.b) - 1)) +
  geom_point(alpha = .5) +
  stat_smooth(method = "glm", se = FALSE, fullrange = TRUE,
              method.args = list(family = binomial)) +
  ylab("Low birth weight") + xlab("Mother's weight (pounds)") + xlim(-1000, 1000)
```



3 Assess model assumptions

Logistic regression assumptions:

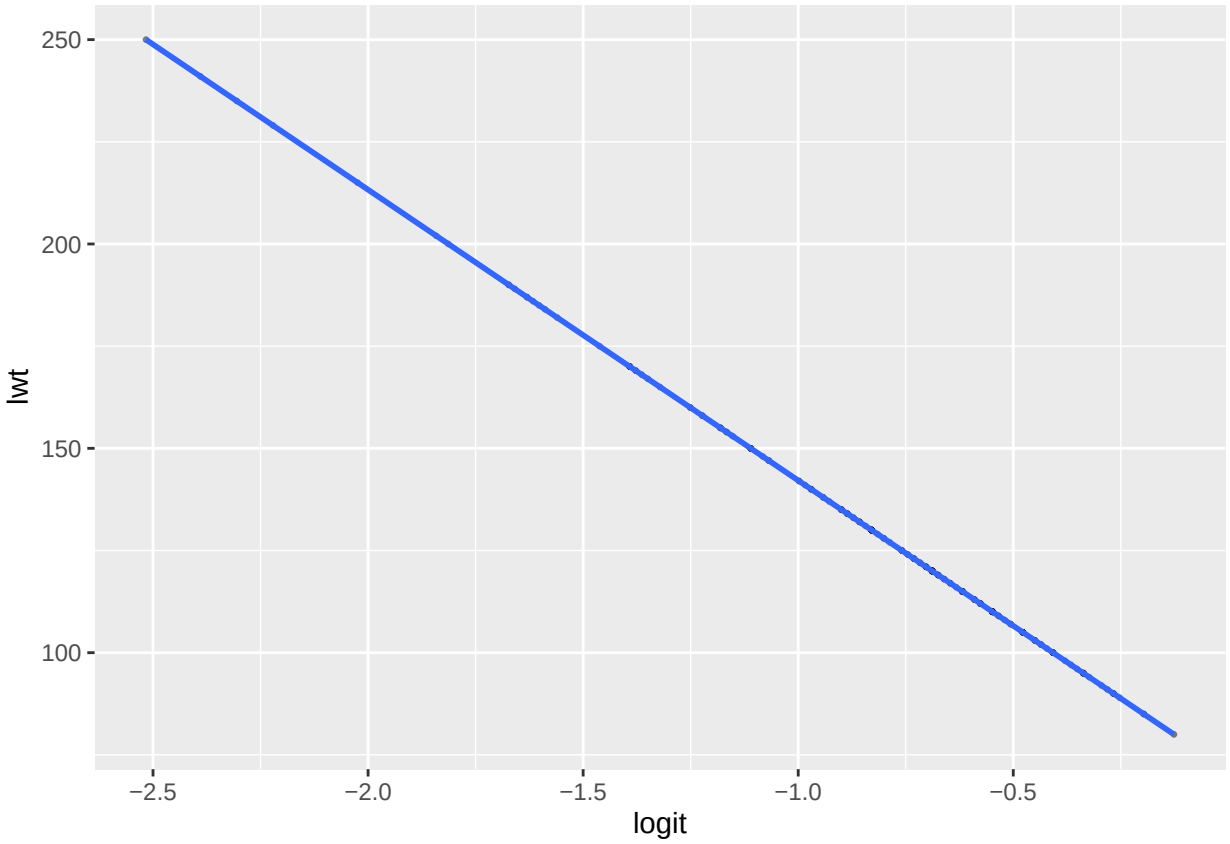
- Independence (only relevant for repeated measurements)
- Linearity in the logit for continuous variables
- Lack of strongly influential outliers
- Absence of multicollinearity (multivariate models)

3.1 Linearity (continuous predictors)

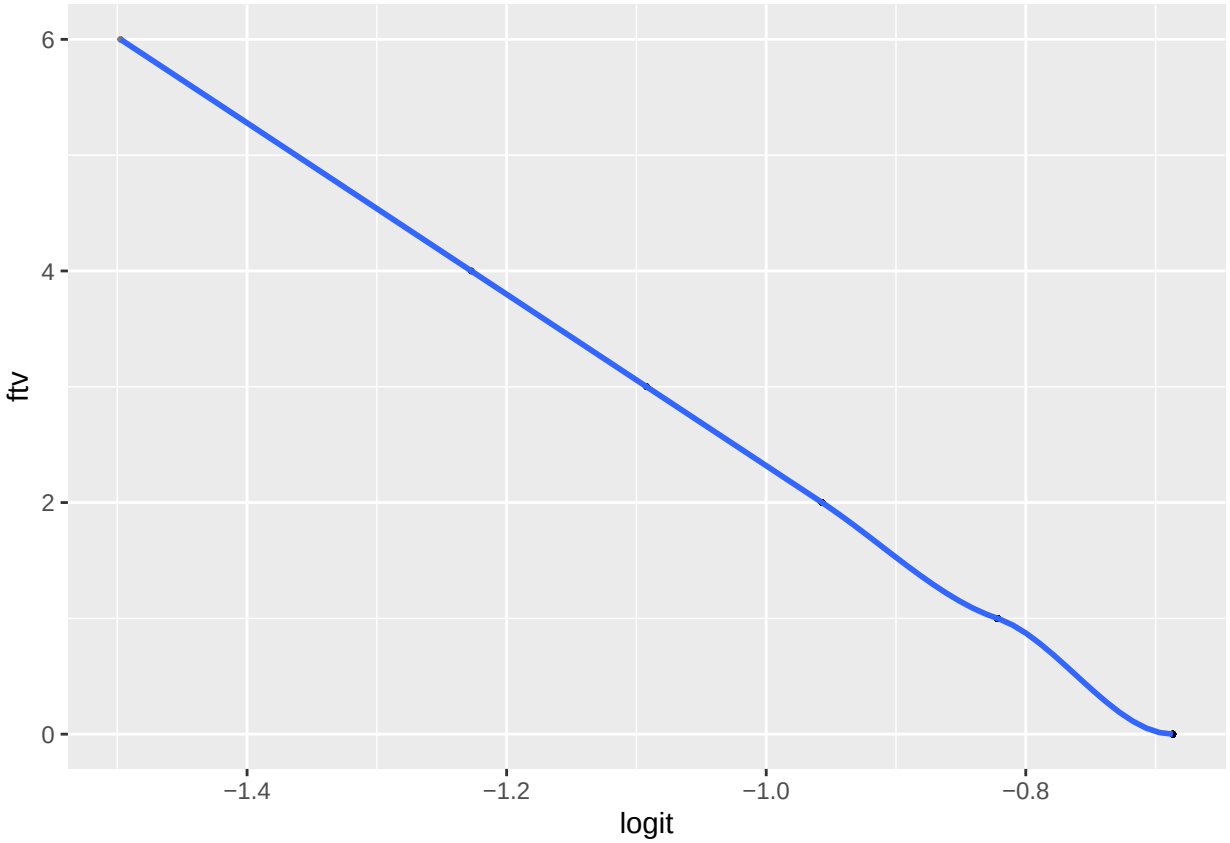
The assumption of linearity refers to the relationship between the log odds of the response variable and the each continuous predictor. The logarithm of the odds for a positive outcome are used. There are two ways to assess it: a visual inspection of their scatterplot and a formal test, the Box-Tidwell.

```
  ## Calculate predicted values for the model
prob <- predict(mod_lwt, type = "response")
  ### isolate continuous predictors
dt_cont <- lbw %>%
  select(lwt)
  ### bind logit and probabilities
dt_cont %<>%
  mutate(logit = log(prob/(1-prob))) # log odds
```

```
  ### scatter plots
ggplot(dt_cont, aes(logit, lwt)) +
  geom_point(size = 0.5, alpha = 0.5) +
  geom_smooth(method = "loess")
```



The relationship is linear beyond doubt. Another example with the use of the number of physician visits shows a very small deviation; it is still considered linear:



The Box-Tidwell test adds a log-transformed interaction term for the continuous independent variable into the model. The significance of this interaction term is assessed with its p-value. The `{car}` package includes a function to run it; be cautious to use the integer version of the response variable, not the factorized one (in our example, use `low` instead of `low.b`).

```
  ### Box-Tidwell test
car::boxTidwell(low ~ lwt, data = lbw, family = binomial)
```

```
##  MLE of lambda Score Statistic (z) Pr(>|z|)
##        -3.9735              1.1146    0.265
##
## iterations =  5
```

The p-value is > 0.05 indicating presence of linearity.

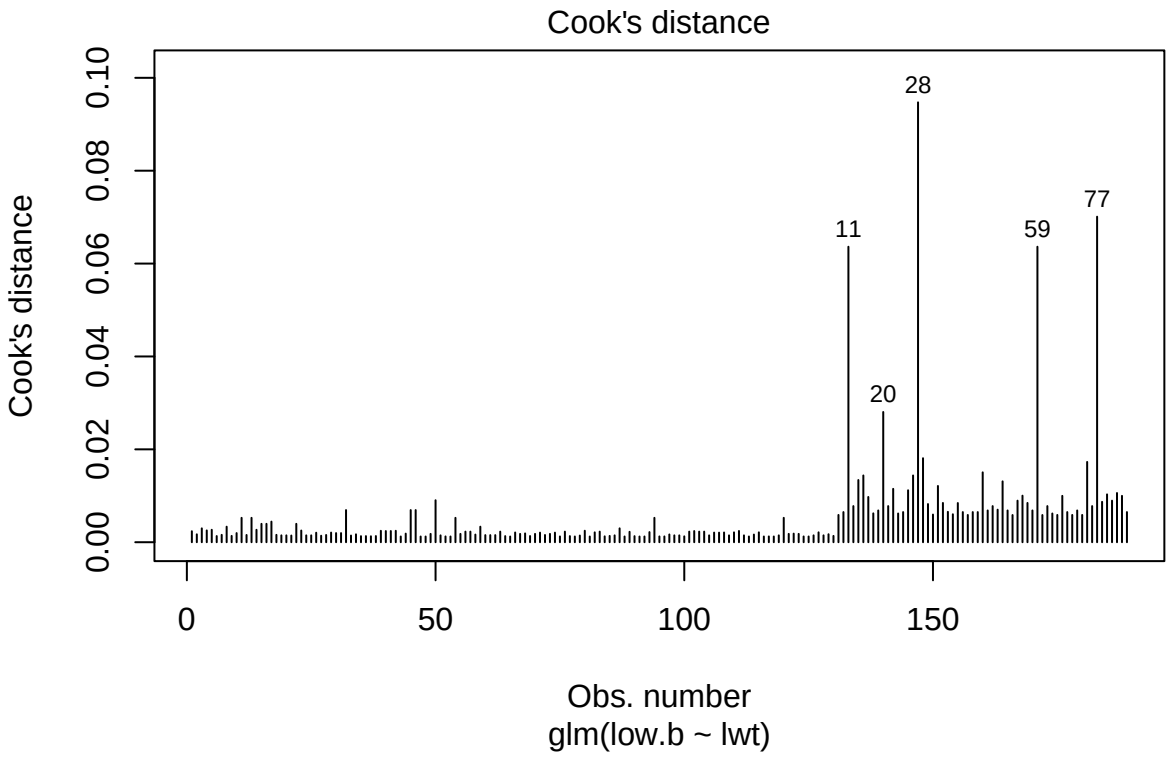
3.2 Identify influential values

To fit a logistic regression model there must be no influential values. Outliers are potential influential values, so we assess influence with Cook's distance and isolate outliers using the standardized residuals.

Inference on Cook's distance has various thresholds: - Observation with Cook's Distance > 3*mean(Cook's Di) - Observation with Cook's Distance > 4/N(observations) - Observation with percentile > 50 using the F-distribution

To facilitate inference, a operational guideline of Cook's Di > 1 has been suggested; however it is advised to look at values above 0.5.

```
  ### Visualize Cook's distance from base R
plot(mod_lwt, which = 4, id.n = 5)
```




```
  ### Calculate Cook's distance
mod <- broom::augment(mod_lwt) %>%
  mutate(id = 1:n()) # useful for the plot of standardized residuals
  ### Cook's Distance > 3*mean(Cook's Di)
which(mod$.cooksd > 3*mean(mod$.cooksd))
```

```
## [1] 133 140 147 148 171 181 183
```

```
  ### Cook's Distance > 4/N(observations)
which(mod$.cooksd > 4/nrow(mod))
```

```
## [1] 133 140 147 171 183
```

```
  ### F-distribution's 50th percentile function for this dataset
qf(.5, 10, 179) # ncol, nrow-ncol
```

```
## [1] 0.9376918
```

```
which(mod$.cooksd > qf(.5, 10, 179))
```

```
## integer(0)
```

```
  ### Rule of thumb
which(mod$.cooksd > 1)
```

```
## integer(0)
```

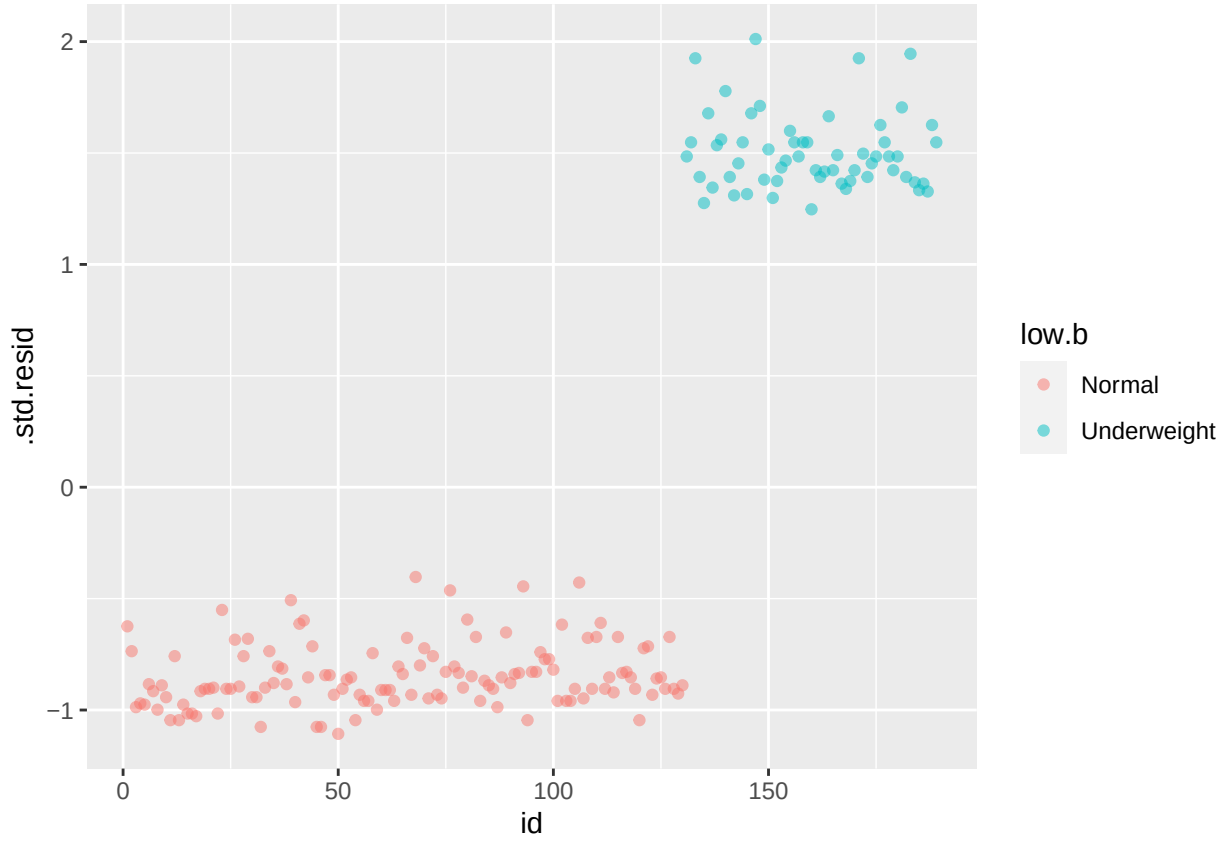
```
which(mod$.cooksd > .5)
```

```
## integer(0)
```

There are some observations that might require attention but the last three criteria do not raise concerns for influence.

The {performance} function check_outliers() facilitates the above calculations and hosts similar assessment methods for most generalized linear models.

```
### Calculate the standardized residuals (>3 implies an outlier)
  ### Plot standardized residuals
ggplot(mod, aes(id, .std.resid)) +
  geom_point(aes(color = low.b), alpha = 0.5)
```




```
  ### Filter data points with absolute standardized residuals > 3
mod %>%
  filter(abs(.std.resid) > 3)
```

```
## # A tibble: 0 x 10
## # ... with 10 variables: .rownames <chr>, low.b <fct>, lwt <int>,
## #   .fitted <dbl>, .resid <dbl>, .std.resid <dbl>, .hat <dbl>, .sigma <dbl>,
## #   .cooksd <dbl>, id <int>
```

There are no outliers, neither in the visual inspection nor satisfying the criterion

4 Assess model fit

4.1 Log-likelihood ratio test

Having assured that assumptions are met, assess model fit with the **log-likelihood** method. The larger the statistic the poorer the fit (compared to the log-likelihood of the null model).

```
mod_lwt$null.deviance
```

```
## [1] 234.672
```

```
mod_lwt$deviance
```

```
## [1] 228.6907
```

```
  ### Calculate the difference in degrees of freedom
mod_lwt$df.null - mod_lwt$df.residual
```

```
## [1] 1
```

```
  ### Retrieve the model Chi-square
mod_lwt$null.deviance - mod_lwt$deviance
```

```
## [1] 5.981327
```

```
  ### compute the p-value
1 - pchisq(5.98, 1)
```

```
## [1] 0.014469
```

```
  ### alternatively, use lrtest with a null model for comparison
lrtest(mod_lwt, glm(low.b ~ 1, lbw, family = binomial))
```

```
## Likelihood ratio test for MLE method
## Chi-squared 1 d.f. =  5.981327 , P value =  0.01445812
```

Since the deviance in the lwt model is smaller than the null and the p-value is smaller than 0.05, the model fit is better with the addition of the lwt as an explanatory variable.

4.2 Nagelkerke's R-squared

Nagelkerke's R-2 is a pseudo R value to assess goodness of fit. The R2 value is the variability explained by the model, in this example 4.4%.

```
rms::lrm(low.b ~ lwt, lbw)
```

```
## Registered S3 method overwritten by 'rms':
##   method       from
##   print.lrtest epiDisplay
```

```
## Logistic Regression Model
##
##  rms::lrm(formula = low.b ~ lwt, data = lbw)
##
##                        Model Likelihood    Discrimination    Rank Discrim.
##                              Ratio Test           Indexes          Indexes
##  Obs           189    LR chi2      5.98    R2       0.044    C       0.613
##   Normal       130    d.f.            1    g        0.452    Dxy     0.226
##   Underweight   59    Pr(> chi2) 0.0145    gr       1.571    gamma   0.232
##  max |deriv| 5e-08                         gp       0.088    tau-a   0.098
##                                            Brier    0.208
##
##            Coef    S.E.   Wald Z Pr(>|Z|)
##  Intercept  0.9983 0.7853  1.27  0.2036
##  lwt       -0.0141 0.0062 -2.28  0.0227
##
```

5 Retrieve model summaries

There are many ways to display results, either with explicit summaries or tables for print.

```
  ## base R
summary(mod_lwt)
```

```
##
## Call:
## glm(formula = low.b ~ lwt, family = binomial, data = lbw)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0951  -0.9022  -0.8018   1.3609   1.9821
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.99831    0.78529   1.271   0.2036
## lwt         -0.01406    0.00617  -2.279   0.0227 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 234.67  on 188  degrees of freedom
## Residual deviance: 228.69  on 187  degrees of freedom
## AIC: 232.69
##
## Number of Fisher Scoring iterations: 4
```

```
cbind(exp(coef(mod_lwt)), exp(confint(mod_lwt)))
```

```
##                            2.5 %     97.5 %
## (Intercept) 2.7137035 0.6180617 13.6228447
## lwt         0.9860401 0.9733982  0.9973535
```

```
  ## epiDisplay
logistic.display(mod_lwt)
```

```
##
## Logistic regression predicting low.b : Underweight vs Normal
##
##                   OR(95%CI)      P(Wald's test) P(LR-test)
## lwt (cont. var.) 0.99 (0.97,1)  0.023          0.014
##
## Log-likelihood = -114.3453
## No. of observations = 189
## AIC value = 232.6907
```

```
  ## gtsummary
tbl_regression(mod_lwt, exponentiate = T,
               label = lwt ~ "Mother's weight (pounds) at last menstrual period")
```

Characteristic	OR	95% CI	p-value
Mother's weight (pounds) at last menstrual period	0.99	0.97, 1.00	0.023

6 Data and session handling

6.1 Using relevel() to change the reference level

```
  ### Initial coding for race uses as reference "White"
modelRC <- glm(low.b ~ race, data = lbw, family = binomial)
tbl_regression(modelRC, exponentiate = T, label = race ~ "Mother's race")
```

Characteristic	OR	95% CI	p-value
Mother's race			
White	—	—	
Black	2.33	0.93, 5.77	0.068
Other	1.89	0.96, 3.76	0.067

```
  ### Changing to "Other"
lbw %<>%
  mutate(race = relevel(race, ref = 3))
modelRC <- glm(low.b ~ race, data = lbw, family = binomial)
tbl_regression(modelRC, exponentiate = T, label = race ~ "Mother's race")
```

Characteristic	OR	95% CI	p-value
Mother's race			
Other	—	—	
White	0.53	0.27, 1.05	0.067
Black	1.23	0.48, 3.09	0.7

6.2 Fitting multiple models

Automating the process of fitting univariate logistic regression models for each independent variable can be accomplished either with a `tidy` approach or functional programming. Note that the resulting objects are lists, to isolate a model use list indexing, i.e. `mods[["age"]]`.

```
mods <- lbw %>%
  select(-low) %>%
  purrr::map(~glm(lbw$low.b ~.x, data = lbw, family = binomial))
  ## Alternative
#mod_f <- function(x){
#  glm(lbw$low.b ~ x, family = binomial)
# }
#mods <- lapply(lbw[,-1], mod_f)
```

7 Multiple logistic regression

Univariate models

Characteristic	OR[1]	95% CI[1]	p-value
Mother's age	0.95	0.89, 1.01	0.10
Mother's weight	0.99	0.97, 1.00	0.023
Mother's race			
Other	—	—	
White	0.53	0.27, 1.05	0.067
Black	1.23	0.48, 3.09	0.7
Smoking status			
No	—	—	
Yes	2.02	1.08, 3.80	0.028
Previous preterm labours	2.23	1.22, 4.28	0.011
Hypertension			
Normal	—	—	
Hypertension	3.37	1.03, 11.8	0.046
Uterine irritability			
Normal	—	—	
Urinary irritability	2.58	1.13, 5.88	0.023
Physician visits (1st trimester	0.87	0.63, 1.17	0.4

[1]OR = Odds Ratio, CI = Confidence Interval

Taking into account the results of the fitted univariate models ($p < 0.2$), we can fit a multivariate model.

```
multilog <- glm(low.b ~ lwt + age + smoke + race + ptl + ht + ui,
                data = lbw,
                family = binomial)
summary(multilog)
```

```
##
## Call:
## glm(formula = low.b ~ lwt + age + smoke + race + ptl + ht + ui,
##     family = binomial, data = lbw)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9120  -0.8174  -0.5224   0.9714   2.1807
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)              1.326038   1.104915   1.200  0.23009
## lwt                     -0.015183   0.006928  -2.192  0.02841 *
## age                     -0.027070   0.036452  -0.743  0.45772
## smokeYes                 0.923349   0.400853   2.303  0.02125 *
## raceWhite               -0.861635   0.439191  -1.962  0.04978 *
## raceBlack                0.401584   0.538868   0.745  0.45613
## ptl                      0.541755   0.346264   1.565  0.11768
## htHypertension           1.833696   0.691765   2.651  0.00803 **
## uiUrinary irritability   0.758597   0.459389   1.651  0.09867 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 234.67  on 188  degrees of freedom
## Residual deviance: 201.43  on 180  degrees of freedom
## AIC: 219.43
##
## Number of Fisher Scoring iterations: 4
```

```
tbl_regression(multilog, exponentiate = T, label = list(age ~ "Mother's age",
                                                         lwt ~ "Mother's weight",
                                                         race  ~ "Mother's race",
                                                         smoke ~ "Smoking status",
                                                         ptl ~ "Previous preterm labours",
                                                         ht ~ "Hypertension",
                                                         ui ~ "Uterine irritability"))
```

Characteristic	**OR**	**95% CI**	**p-value**
Mother's weight	0.98	0.97, 1.00	0.028
Mother's age	0.97	0.90, 1.04	0.5
Smoking status			

Characteristic	OR	95% CI	p-value
No			
Yes	2.52	1.16, 5.65	0.021
Mother's race			
Other			
White	0.42	0.17, 0.99	0.050
Black	1.49	0.52, 4.33	0.5
Previous preterm labours	1.72	0.88, 3.48	0.12
Hypertension			
Normal			
Hypertension	6.26	1.67, 26.5	0.008
Uterine irritability			
Normal			
Urinary irritability	2.14	0.86, 5.27	0.10

Seeing that PTL does not contribute significantly, one may wish to recode it, into a two-level factor to explore its potential as an explanatory variable:

```
table(lbw$ptl)
```

```
##
##   0   1   2   3
## 159  24   5   1
```

```
  ### reduce the integer into a two-level factor variable with levels:
  ### None (no previous preterm labours), Any (any number)
lbw %<>%
  mutate(ptl.b = factor(ptl, levels = c("0", "1", "1", "1"),
                        labels = c("None", "Any", "Any", "Any")))
```

Having the recoded PTL, run the multiple model again:

```
multilog.b <- glm(low.b ~ lwt + age + smoke + race + ptl.b + ht + ui,
                  data = lbw,
                  family = binomial)
summary(multilog.b)
```

```
##
## Call:
## glm(formula = low.b ~ lwt + age + smoke + race + ptl.b + ht +
##     ui, family = binomial, data = lbw)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6437  -0.7762  -0.5031   0.9221   2.2013
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)             1.378485   1.140560   1.209  0.22682
## lwt                    -0.014203   0.007133  -1.991  0.04647 *
## age                    -0.041885   0.038641  -1.084  0.27839
```

```
## smokeYes                0.912355   0.422148   2.161  0.03068 *
## raceWhite              -0.743183   0.470220  -1.581  0.11399
## raceBlack               0.371300   0.557688   0.666  0.50555
## ptl.bAny                1.573454   0.524231   3.001  0.00269 **
## htHypertension          1.797062   0.705569   2.547  0.01087 *
## uiUrinary irritability  0.842106   0.489255   1.721  0.08521 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 227.02  on 182  degrees of freedom
## Residual deviance: 187.09  on 174  degrees of freedom
##   (6 observations deleted due to missingness)
## AIC: 205.09
##
## Number of Fisher Scoring iterations: 4
```

```
tbl_regression(multilog.b, exponentiate = T, label = list(age ~ "Mother's age",
                                                          lwt ~ "Mother's weight",
                                                          race  ~ "Mother's race",
                                                          smoke ~ "Smoking status",
                                                          ptl.b ~ "Previous preterm labours",
                                                          ht ~ "Hypertension",
                                                          ui ~ "Uterine irritability"))
```

Characteristic	**OR**	**95% CI**	**p-value**
Mother's weight	0.99	0.97, 1.00	0.046
Mother's age	0.96	0.89, 1.03	0.3
Smoking status			
No			
Yes	2.49	1.10, 5.83	0.031
Mother's race			
Other			
White	0.48	0.19, 1.19	0.11
Black	1.45	0.48, 4.36	0.5
Previous preterm labours			
None			
Any	4.82	1.77, 14.1	0.003
Hypertension			
Normal			
Hypertension	6.03	1.56, 26.2	0.011
Uterine irritability			
Normal			
Urinary irritability	2.32	0.88, 6.10	0.085

Note the difference in all p-values: it is a new covariate we are adding, consequently the entire model is affected.

7.1 Multicollinearity assumption diagnostics

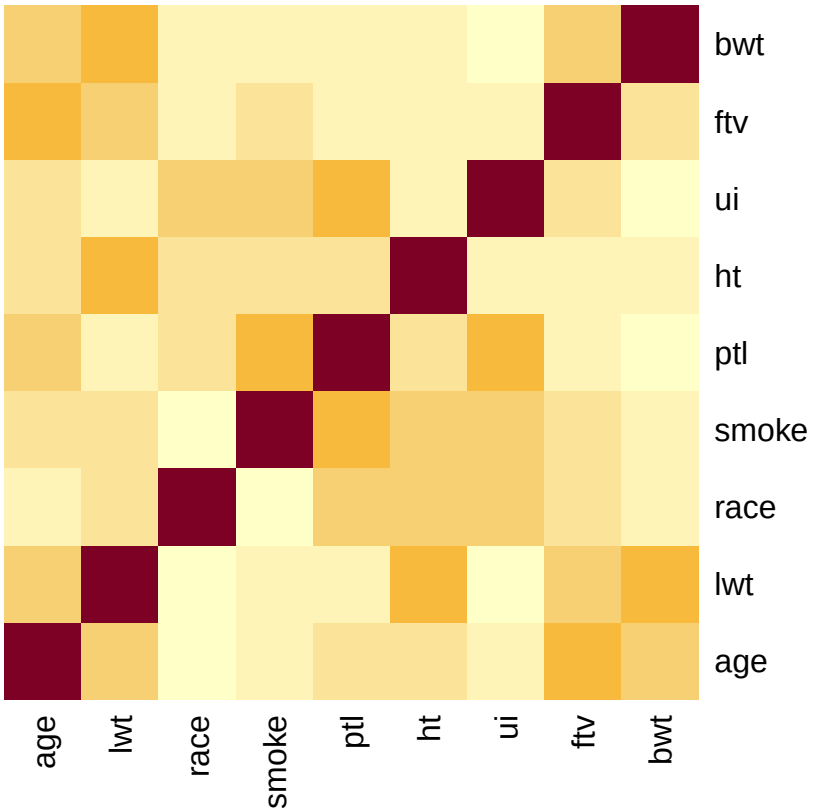
An additional assumption for multivariate models is the absence of multicollinearity: the absence of highly correlated independent variables. A VIF exceeding **5** is evidence of multicollinearity:

```
car::vif(glm(low.b ~ lwt + age + smoke + race + ptl + ht + ui,
        data = lbw,
        family = binomial))
```

```
##           GVIF Df GVIF^(1/(2*Df))
## lwt   1.296289  1        1.138547
## age   1.063637  1        1.031328
## smoke 1.339541  1        1.157386
## race  1.500277  2        1.106733
## ptl   1.087525  1        1.042845
## ht    1.155113  1        1.074762
## ui    1.059519  1        1.029330
```

A correlation matrix heatmap may also be useful to identify correlated independent variables, such as `bwt` and `lwt` (which is to be expected as `bwt` derives the response variable in this model.)

```
cm <- cor(MASS::birthwt[,-1], method = "pearson")
heatmap(cm, Rowv = NA, Colv = NA)
```



8 Comparing models

The analysis of deviance table and Akaike's Information Criterion (AIC) are used to compare models. Here we fit a reduced model (including only those variables with a p-value < 0.05 in the univariate analysis) to compare with the full multivariate model.

```
multilog.red <- glm(low.b ~ lwt + smoke + race + ht,
                    data = lbw,
                    family = binomial)
```

The analysis of deviance table uses the Chi-squared test to compare models. The derived p-value implies that the reduced model is not an improvement of the full model, despite the smaller deviance.

```
anova(multilog, multilog.red, test = "Chisq")
```

```
## # A tibble: 2 x 5
##   `Resid. Df` `Resid. Dev`    Df Deviance `Pr(>Chi)`
##         <dbl>        <dbl> <dbl>    <dbl>      <dbl>
## 1         180         201.    NA    NA       NA
## 2         183         208.    -3    -6.82     0.0778
```

8.1 AIC

Obtaining the AIC is straightforward in R: either use the `summary()` or directly use `AIC()`.

```
AIC(multilog)
```

```
## [1] 219.427
```

```
AIC(multilog.red)
```

```
## [1] 220.2474
```

The smaller the AIC, the better the fit. Here, the initial full model fits the data better than the reduced.

9 Stepwise model selection

The same principle in stepwise model selection is used as in linear regression. The bakcward elimination model excludes the `age` variable while forward selection adds all variables to reach the best model.

```
# Backward elimination
step(multilog.b, direction = "backward")
```

```
## Start:  AIC=205.09
## low.b ~ lwt + age + smoke + race + ptl.b + ht + ui
##
##         Df Deviance    AIC
## - age    1   188.30 204.30
```

```
## <none>       187.09 205.09
## - ui     1   190.01 206.01
## - race   2   192.20 206.20
## - lwt    1   191.49 207.49
## - smoke  1   191.90 207.90
## - ht     1   193.87 209.87
## - ptl.b  1   196.72 212.72
##
## Step:  AIC=204.3
## low.b ~ lwt + smoke + race + ptl.b + ht + ui
##
##         Df Deviance    AIC
## <none>       188.30 204.30
## - ui     1   191.47 205.47
## - race   2   194.75 206.75
## - smoke  1   193.56 207.56
## - lwt    1   193.68 207.68
## - ht     1   195.24 209.24
## - ptl.b  1   196.95 210.95

##
## Call:  glm(formula = low.b ~ lwt + smoke + race + ptl.b + ht + ui, family = binomial,
##     data = lbw)
##
## Coefficients:
##            (Intercept)                    lwt                smokeYes
##                0.59600               -0.01532                0.94226
##              raceWhite              raceBlack               ptl.bAny
##               -0.81481                0.40844                1.44842
##         htHypertension  uiUrinary irritability
##                1.82969                0.86886
##
## Degrees of Freedom: 182 Total (i.e. Null);  175 Residual
##   (6 observations deleted due to missingness)
## Null Deviance:       227
## Residual Deviance: 188.3     AIC: 204.3

# Forward selection
lbwn <- na.omit(lbw)
nullmod <- glm(low.b ~ 1, data = lbwn, family = binomial)
step(nullmod, scope = list(lower = formula(nullmod), upper = formula(multilog.b)),
     direction = "forward", na.rm = T)

## Start:  AIC=229.02
## low.b ~ 1
##
##         Df Deviance    AIC
## + ptl.b  1   212.07 216.07
## + smoke  1   221.51 225.51
## + ui     1   221.53 225.53
## + lwt    1   222.01 226.01
## + ht     1   222.97 226.97
## + race   2   222.25 228.25
```

```
## + age    1   224.41 228.41
## <none>       227.02 229.02
##
## Step:  AIC=216.07
## low.b ~ ptl.b
##
##         Df Deviance    AIC
## + age    1   207.11 213.11
## + lwt    1   207.94 213.94
## + ht     1   208.09 214.09
## + ui     1   208.29 214.29
## + smoke  1   208.82 214.82
## + race   2   208.01 216.01
## <none>       212.07 216.07
##
## Step:  AIC=213.11
## low.b ~ ptl.b + age
##
##         Df Deviance    AIC
## + ht     1   203.23 211.23
## + ui     1   203.88 211.88
## + smoke  1   204.20 212.20
## + lwt    1   204.40 212.40
## <none>       207.11 213.11
## + race   2   204.67 214.67
##
## Step:  AIC=211.23
## low.b ~ ptl.b + age + ht
##
##         Df Deviance    AIC
## + lwt    1   198.01 208.01
## + ui     1   199.05 209.05
## + smoke  1   200.40 210.40
## <none>       203.23 211.23
## + race   2   200.97 212.97
##
## Step:  AIC=208.01
## low.b ~ ptl.b + age + ht + lwt
##
##         Df Deviance    AIC
## + ui     1   194.70 206.70
## + smoke  1   195.35 207.35
## <none>       198.01 208.01
## + race   2   195.25 209.25
##
## Step:  AIC=206.7
## low.b ~ ptl.b + age + ht + lwt + ui
##
##         Df Deviance   AIC
## + smoke  1    192.2 206.2
## <none>        194.7 206.7
## + race   2    191.9 207.9
##
## Step:  AIC=206.2
```

```
## low.b ~ ptl.b + age + ht + lwt + ui + smoke
##
##        Df Deviance    AIC
## + race  2   187.09 205.09
## <none>      192.20 206.20
##
## Step:  AIC=205.09
## low.b ~ ptl.b + age + ht + lwt + ui + smoke + race

##
## Call:  glm(formula = low.b ~ ptl.b + age + ht + lwt + ui + smoke + race,
##     family = binomial, data = lbwn)
##
## Coefficients:
##            (Intercept)                ptl.bAny                     age
##                1.37849                 1.57345                -0.04188
##         htHypertension                     lwt  uiUrinary irritability
##                1.79706                -0.01420                 0.84211
##               smokeYes               raceWhite               raceBlack
##                0.91236                -0.74318                 0.37130
##
## Degrees of Freedom: 182 Total (i.e. Null);  174 Residual
## Null Deviance:       227
## Residual Deviance: 187.1     AIC: 205.1
```

10 Sum up

Model fitting consists of four processes:

1. Data exploration
2. Assess model assumptions
3. Assess model fit
4. Reduce model variables to the best fitting model