Stat331 - Project

Humble Masood and Karim Reimoo

December 04, 2018

Summary

The objective of the report is to analyze the relation between healthy male single-fetus birth weight and some explanatory variables. We started the report with pre-fitting data diagnostic. We identified that the dataset had many NA's based on the dataset summary. We decided to remove father's height and father's weight because they present data suggested no association between them and the male single-fetus birth weight based on their pair plots. We applied mice imputing strategy to fill in the remaining missing values. After that we converted all the categorical variables in our data set into factors and figured out a way to deal with cateogries of covariates with low observations. We then created a pair plot and showed the associations between covariates and our response. We have also dealt with any data inconsistency and then we have applied two automated slection strategies (forward and step-wise) to narrow down our models. Based on the lowest RMSPE (generated by cross validation of each model on all five imputed datasets), we decided to select our final two models on which we did further analysis. We did an indepth analysis by creating residual vs predicted plots, qqplots and again used cross validation from before to retain one final model. And finally, we have added our recommendation and listed any drawbacks with this model. Our final model narrowed down the covariates to number, mht, meth, gestation:number, parity, gestation, income, gestation:income, med:mwt, smoke, med, mwt, time, gestation:med and meth:income.

1 Pre-fitting data diagnostic

1.1 Original data-set summary

A preliminary assessment of the data is to look at the **summary** of the initial data set in table 1.1:

NA Count	Covariate	Covariate reperesentation
499	fwt	father's weight
492	fht	father's height
124	income	family's yearly income in $2500 increments
36	mwt	mother's weight
31	feth	father's ethinicity
22	mht	mother's height
21	number	# of cigarettes smoked per day by mother when she was smoking
13	gestation	length of gestation period
13	fed	father's education
10	smoke	Does mother smoke?
10	time	Time since mother quit smoking before pregnancy
7	fage	father's age
2	mage	mother's age
1	meth	mother's ethinicity
1	med	mother's education

1.2 Treatment of erroneous data

By looking at table 1.1, we observe that two observations have value 0 for **marital** covariate which does not correspond to any defined marital status. It is possible that whoever was conducting the survey might have recorded incorrect values. Thus, we are dropping these two observations from our data set.

1.3 Treatment of NAs

- We have decided to drop **fwt** and **fht** covariates from our data set because most of the data is missing and the pair plots in figure 1.1 suggests that there is no strong association between male single-fetus birth weight and the known observations of "fwt" and "fht". It is possible that an association emerges if the missing observations were known.
- We have decided to impute the missing data. We could use single or multiple imputation strategies. However, single imputation results in an underestimation of standard errors. Therefore, we have decided to use multiple imputation by chained equation to fill all the remaining missing data.

1.4 Converting categorical variables into factors

The following table shows the assigned factor levels for each categorical variables:

Categorical Variable	Factor Levels
smoke	"never", "until_pregnancy", "used_to_not_anymore", "smokes_now"
med & fed	"elementary", "middle", "high", "high+trade", "high+college", "college_grad", "trade", "high_unclear"
marital	married","legally_separated","divorced","widowed","never_marr
time	"never_smoked", "<1year", "1-2years", "2-3years", "3-4years", "5-9years", ">10years", "during_pregnancy", "still_smokes", "quit_but_dont_know_when"
number	"never_smoked", "1-4", "5-9", "10-14", "15-19", "20-29", "30-39", "40-60", ">60", "smoked_but_dont_know_how_much"
income	"<2500", "2500-4999", "5000-7499", "7500-9999", "10000-12499", "12500-14999", "15000-17499", "17500-19999", "20000-22499", ">=22500"
meth & feth	"Caucasian", "Mexican", "African-American", "Asian", "Mixed", "Other"

Interpretation of each factor level is the same as defined in the project description.

1.5 Handling covariate categories with few observations

The summary in table 1.1 also suggests important information about the categorical covariates which are as follows:

- **marital**: We can see that the marital covariate has high observations of "married"" mothers and very few observations of each of the other categories. Therefore, we have decided to convert marital into a binary categorical covariate with two categories i.e "married" and "not married". We have lumped together all observations with category other than "married" into "not married" category. After this transformation, we have 1095 observations for married and 20 for not married. Even after this, "not married" mothers are based on a relatively small sample size, so more data is required to support any statement that our final model will make about them.
- **meth & feth**: We also notice that both meth has relatively low observations in each category other than "Caucasian" and "African-American" mothers. Therefore, we can convert meth into a categorical covariate with three levels. "Caucasian", "African-American" and "Other" are the three levels that we can divide our data set into. We will lump together all observations with category other than "Caucasian" and "African-American" into "Other". This new "Other" is different from the original "Other" in the data set and it now represents ethinicities of mothers who are neither caucasian nor african american. After this transformation we have 733 Caucasian mothers, 189 African-American mothers and 82 "others".
- **med & fed**: We have dropped "high school unclear" & "trade" because there was not enough data to make any strong statements involving it. Bundled "elementary", "middle school" and "high school" categories into a new category called "upto high school" so that we have few regression coeffecients to fit and we have more data for each category which will give more weight to any statement that we make. All other factor level remain unchanged.
- **income**: We have changed income to three factor levels "0-4999", "5000-14999" and ">=15000". The reason for the change is so we have few regression coeffecients to fit and we have more data for each category which will give more weight to any statement that we make.
- **time**: We have dropped "quit_but_dont_know_when" and "still_smokes" because there was not enough data to make any strong statements involving it. We have changed time to four factor levels "never_smoked", "during_pregnancy", "0-2years", ">2years". The reason for the change is so we have few regression coeffecients to fit and we have more data for each category which will give more weight to any statement that we make.
- **number**: We have dropped "smoked_but_dont_know_how_much" because there was not enough data to make any strong statements involving it. We have changed number to three factor levels "never_smoked", "1-9" and ">=10". The reason for the change is so we have few regression coeffecients to fit and we have more data for each category which will give more weight to any statement that we make.

1.6 Pairs-plot for all the continuous covariates

The pair-plots of all the continous covariates suggests that:

- The strongest association is between wt and gestation which suggests that the longer the gestation period will be, the higher the weight of a male single-fetus birth.
- There is also strong association between fage and mage which suggests that couples having a baby together are usually of the same age.

- Parity and mht should be considered as categorical covariates. We know that parity and mht are continuous covariates but by looking at the pair plot we can observe that parity contains 13 categories and mht contains 19 categories. This is most likely because mothers were asked about their previous pregnancy and their height on a discrete scale. So we have decided to treat parity and mht as categorical variables basend on the given data with the following factor levels:
- parity: Factor levels "no_prev_preg", "one_prev_preg", "two_prev_preg", "at_least_three_prev_preg".
- mht: "at_most_60", "61-63", "64-66", "at_least_67" which represents mother's height in inches.

1.7 Summary after dealing with erroneous data, NAs and covariate categories

After cleaning the data, we get another representation of the data by looking at the summary given in the five tables from Table 1.2.1 to Table 1.2.5.

2 Model selection

2.1 Detecting and Removing Data Inconsistency

After running automated model selection on our imputed data sets, we decided to run a table command to see if there are any data inconsistency. We found that there are row observations in 'smoke' with level "smokes_now" and 'number' with level "never_smoked" which does not make sense. Similarly, there are also observations in 'smoke' with level "until_pregnancy" and 'number' with level "never_smoked". Therefore, we have decided to remove the rows with these observations.

2.2 Automated model selection

After removing data incosistencies, we have applied forward selection and step-wise. We have selected forward selection because it is time-effecient and selected step-wise because it is a compromise between the two approaches.

2.3 Treatment of NAs in Beta_hat

- We noticed that time and number are encoding the same information about the mother (whether or not mother smoked and if she did smoke then how much which means there are linearly dependent columns in our model matrix). Forward and step-wise selection are reporting NAs for the beta hats of covariate 'time' (for both still_smokes and never_smoked levels) and we have decided not to do anything about it because the information of interest is present in the subset of beta hats corresponding to the covariate 'number' (for all its levels).

2.5 Selecting Best Forward & Stepwise Selection Models

After checking the formula for the five forward selection models, we notice that model 1, 3 and 4 are similar so we have three different models to consider. (We will be removing model#4 from our analysis). We can apply cross-validation technique to narrow these three models to one.

After checking the formula for the five stepwise selection models, we notice that model 1, 4 and 5 are similar so we have three different models to consider. (We will be removing model#4 from our analysis). We can apply cross-validation technique to narrow these three models to one.

Here are the Root Mean Squre Prediction Error (RMSPE) for each of the models on the five imputed datasets:

Stepwise selection:

	Mstep1	Mstep2	Mstep3	Mstep5
chds1	234.72	234.77	236.13	236.13
chds2	237.65	237.14	238.23	238.23
chds3	235.72	235.24	236	236
chds4	233.51	233.44	234.86	234.86
chds5	237.64	236.93	236.09	236.09

Forward selection:

	Mfwd1	Mfwd2	Mfwd3	Mfwd5
chds1	237.67	238.98	237.67	238.66
chds2	240.18	240.30	240.18	239.81
chds3	237.81	238.58	237.81	237.86
chds4	236.74	237.80	236.74	237.50
chds5	240.58	239.96	240.58	239.09

We are cross validating each model against the other three models on each of the five imputed data sets. Since **Mstep2** had the lowest RMSPE on three of the five datasets, we selected it from among the stepwise selection models. Since **Mfwd1** had the lowest RMSPE on four of the five datasets, we selected it from among the forward selection models.

2.6 Pooling Best Forward & Stepwise Selection models

Now, we will pool the best forward selected & stepwise selected models onto our five imputed data sets. A summary for both models are shown in Figure 2.6.1 and 2.6.2

3 Model Diagnostics

3.1 QQ And Residual vs Predicted plots

We have plotted studentized residuals on the data scale against the predictions for both Mfwd1 and Mstep2. Based on this plot we do not see any systematic change in the variance as a function of the predicted values. Hence, it is safe to conclude that the homoskedastic assumption is not violated. Moreover, based on this plot we have no reason to believe that the mean weight is not a linear function of the model (Mfwd1 and Mstep2) covariates.

We used the standardized residuals to display the qq-plot for Mfwd1 and Mstep2. Based on the plot we can see that the residuals are in fact normally distributed and so our final assumption is also not violated.

3.2 Influence Leverage measures

We can see that there are many flagged points (influential and leverage points) have

Weight against each continous covariate figure makes predictions with all observations and with one observation ommited: either the most influention one or the one with highest leverage. We can see that the difference between predictions with and without the ommitted observations is same when either the influential observation

is removed or the leverage observation is removed. Thus we can not conclude any statements about safely removing any observations from our data set.

Our Final model

Referring to our table of RMSPEs, we notice that RMSPE values for Mstep2 are lower than Mfwd1 across all five data sets. Therefore, we have decided to select Mstep2 as our final model.

	est	se	t	df	p-value
(Intercept)	20.03679477	24.29257214	0.8248116	497.4304	4.098739e-01
gestation	0.18663010	0.07065211	2.6415360	936.1199	8.390710e-03
parity	0.75662737	0.24354616	3.1067103	1178.8230	1.937022e-03
meth	-1.08470929	0.27828124	-3.8978886	978.6146	1.036457e-04
med	-9.00372346	5.83252831	-1.5437085	819.5355	1.230450e-01
mht	0.86549993	0.21324808	4.0586529	408.5263	5.915430e-05
mwt	-0.02786400	0.04649179	-0.5993315	865.7375	5.491086e-01
income	-10.65283848	4.01643533	-2.6523117	151.5827	8.845084e-03
smoke	2.62457538	1.25385290	2.0932084	1139.3309	3.655125e-02
time	-0.78429345	0.72252111	-1.0854955	1146.3678	2.779305e-01
number	-16.69027381	3.95339179	-4.2217606	337.5619	3.120349e-05
gestation:income	0.03703425	0.01418726	2.6103874	156.2921	9.923749e-03
gestation:number	0.05231351	0.01416649	3.6927653	328.4358	2.596979e-04
gestation:med	0.01517251	0.01978612	0.7668256	662.8842	4.434582e-01
med:mwt	20.03747103	0.01494988	2.5064444	953.4797	1.236074e-02
meth:income	0.08425812	0.06538605	1.2886253	475.9368	1.981544e-01

Discussion

What are the most important factors associated with high/influencing birth weight?

We are going to consider any covariate in the final model with a p-value of less than 0.05 as significant. Based on the model (Mstep2) p-values in table above, we have that the following covariates are significant (ranked in decreasing order of p-values):

1. number: p-value = 3.12e-5
2. mht: p-value = 5.92e-5
3. meth: p-value = 1.04e-4
4. gestation:number: p-value = 2.60e-4
5. parity: p-value = 1.94e-3
6. gestation: p-value = 8.39e-3
7. income: p-value = 8.85e-3
8. gestation:income: p-value = 9.92e-3
9. med:mwt: p-value = 1.24e-2
10. smoke: p-value = 3.66e-2

In our final model (Mstep2) any coefficient with p-value greater than 0.05 is defined as having a high p-value. Based on this definition, the coefficients (excluding intercept) with high p-value are: 1. med: p-value = 1.23e-1 2. mwt: p-value = 5.49e-1 3. time: p-value = 2.78e-1 4. gestation:med: p-value = 4.43e-1 5. meth:income: p-value = 1.98e-1

Recommendations:

1. It is preferable that the mother should not be a smoker. Mother's who do not smoke are less likely to have babies with low birth weight than mothers who do
2. The mother, if she happens to be a smoker, should try to cut down on the number of cigarettes smoked daily. The number of cigarettes smoked impacts the birth weight. Therefore, the fewer the cigarette a mother smokes daily the more likely she is to avoid having a baby with low birth weight.
3. The longer the mother can cut down on her cigarette usage during her pregnancy the more likely it is that the baby's birth weight does not end up being low.

4. The parents should adequately educates themselves about family planning before deciding on how many children to have. The baby's weight is impactedby the number of previous pregnancies (including fetal deaths and still births) therefore, the fewer the pregnancies the more likely it is that the baby's weight is not low.

5. The parents should ensure that they have sufficient income to support a healthy lifestyle for themselves and any kids that they have or that the plan to have. Higher income results in a better lifestyle which impacts the mother's health and the baby's health therefore, it is less likely that a newborn baby ends up having a low weight.

6. The parents should strive to ensure the that the length of pregnancy (gestation) is as normal as it can possibly be. To achieve this, they should follow their doctor's recommendation about the kind of lifestyle that the mother should have during her pregnancy. The mother should make a conscious effort to avoid anything negative that might affect the length of pregnancy such as stressing too much or engaging in manual labour. This will ensure that the baby is less likely to have a low weight.

Why coefficients with high p-values are there?

1. Common sense tells us that mwt should be a good predictor of the male single-fetus birth weight because the more healthy the mother is (as measured by her weight) the more healthy should the baby be at time of birth. But mht and mwt are known to be strongly correlated therefore, it is possible that mwt becomes stastically less significant in mht's presence accounting for its low p-value in the final model.

2. Smoking or not smoking should have a statistically significant impact on the birth weight (as is confirmed by the presence of covariates number, time and smoke in our final/best model). However, number of cigarettes smoked (while the mother was/is smoking) tends to have a larger impact than the time the mother quit smoking (if ever). Hence, while time may be a good predictor of the birth weight on its own right, it may not be as good a predictor in the presence of number as individually resulting, in a high p-value for its coefficient.

3. gestation:med and meth:income lose some of their significance in the presence of gestation, meth and income as individual predictors. Therefore, the two have low p-values

4. med: Our common sense also tells us that although mother's education might have some impact on birth weight but it may not be always true. The education can help mother make better decisions regarding her lifestyles which will impact her baby's birth weight. However, there are many cases where a low educated mother might make better decisions as well. Therefore, our model predicts that med is not a significant predictor and it aligns with our common sense.

Outlying observations that might be safe to remove We can not conclude any statements about safely removing any observations from our data set because our plots of weight against each continous covariate in section 3.2 does not show us any signs of any differences.

Violating assumptions and limitations of our model

Our final model does not violate any assumptions. However, there can be some possible defeciences about correctly performing leverage and influential measures. We have detected one outlier in each model but we have not properly considered the effect of outliers on our data set which may affect the recommendations that we provided.

STAT331 Final Project

Humble Masood & Karim Reimoo

1 Pre-fitting data diagnostic

1.1 Original dataset summary

```
chds <- read.csv("chds_births.csv", sep=",")
summary(chds)
```

```
##        wt          gestation         parity            meth
##  Min.   : 55.0   Min.   :148.0   Min.   : 0.000   Min.   : 0.000
##  1st Qu.:108.8   1st Qu.:272.0   1st Qu.: 0.000   1st Qu.: 0.000
##  Median :120.0   Median :280.0   Median : 1.000   Median : 3.000
##  Mean   :119.6   Mean   :279.3   Mean   : 1.932   Mean   : 3.129
##  3rd Qu.:131.0   3rd Qu.:288.0   3rd Qu.: 3.000   3rd Qu.: 7.000
##  Max.   :176.0   Max.   :353.0   Max.   :13.000   Max.   :10.000
##                  NA's   :13                       NA's   :1
##       mage            med             mht             mwt
##  Min.   :15.00   Min.   :0.000   Min.   :53.00   Min.   : 87.0
##  1st Qu.:23.00   1st Qu.:2.000   1st Qu.:62.00   1st Qu.:114.8
##  Median :26.00   Median :2.000   Median :64.00   Median :125.0
##  Mean   :27.26   Mean   :2.917   Mean   :64.05   Mean   :128.6
##  3rd Qu.:31.00   3rd Qu.:4.000   3rd Qu.:66.00   3rd Qu.:139.0
##  Max.   :45.00   Max.   :7.000   Max.   :72.00   Max.   :250.0
##  NA's   :2       NA's   :1       NA's   :22      NA's   :36
##       feth            fage            fed             fht
##  Min.   : 0.000   Min.   :18.00   Min.   :0.000   Min.   :60.0
##  1st Qu.: 0.000   1st Qu.:25.00   1st Qu.:2.000   1st Qu.:68.0
##  Median : 3.000   Median :29.00   Median :4.000   Median :71.0
##  Mean   : 3.154   Mean   :30.35   Mean   :3.127   Mean   :70.2
##  3rd Qu.: 7.000   3rd Qu.:34.00   3rd Qu.:5.000   3rd Qu.:72.0
##  Max.   :10.000   Max.   :62.00   Max.   :7.000   Max.   :78.0
##  NA's   :31       NA's   :7       NA's   :13      NA's   :492
##       fwt             marital         income          smoke
##  Min.   :110.0   Min.   :0.000   Min.   :0.000   Min.   :0.0000
##  1st Qu.:155.0   1st Qu.:1.000   1st Qu.:2.000   1st Qu.:0.0000
##  Median :170.0   Median :1.000   Median :3.000   Median :1.0000
##  Mean   :171.2   Mean   :1.038   Mean   :3.701   Mean   :0.8018
##  3rd Qu.:185.0   3rd Qu.:1.000   3rd Qu.:5.000   3rd Qu.:1.0000
##  Max.   :260.0   Max.   :5.000   Max.   :9.000   Max.   :3.0000
##  NA's   :499                     NA's   :124     NA's   :10
##       time            number
##  Min.   :0.0000   Min.   :0.00
##  1st Qu.:0.0000   1st Qu.:0.00
##  Median :1.0000   Median :1.00
##  Mean   :0.9625   Mean   :1.76
##  3rd Qu.:1.0000   3rd Qu.:3.00
##  Max.   :9.0000   Max.   :8.00
##  NA's   :10       NA's   :21
```

Table 1.1

1.2 Treatment of erroneous data

```
# drop observations which have value 0 for covariate 'marital'
chds <- chds[chds$marital != 0, ]
```

1.3 Treatment of NAs

```
pairs(~ wt + fwt + fht, data=chds)
```

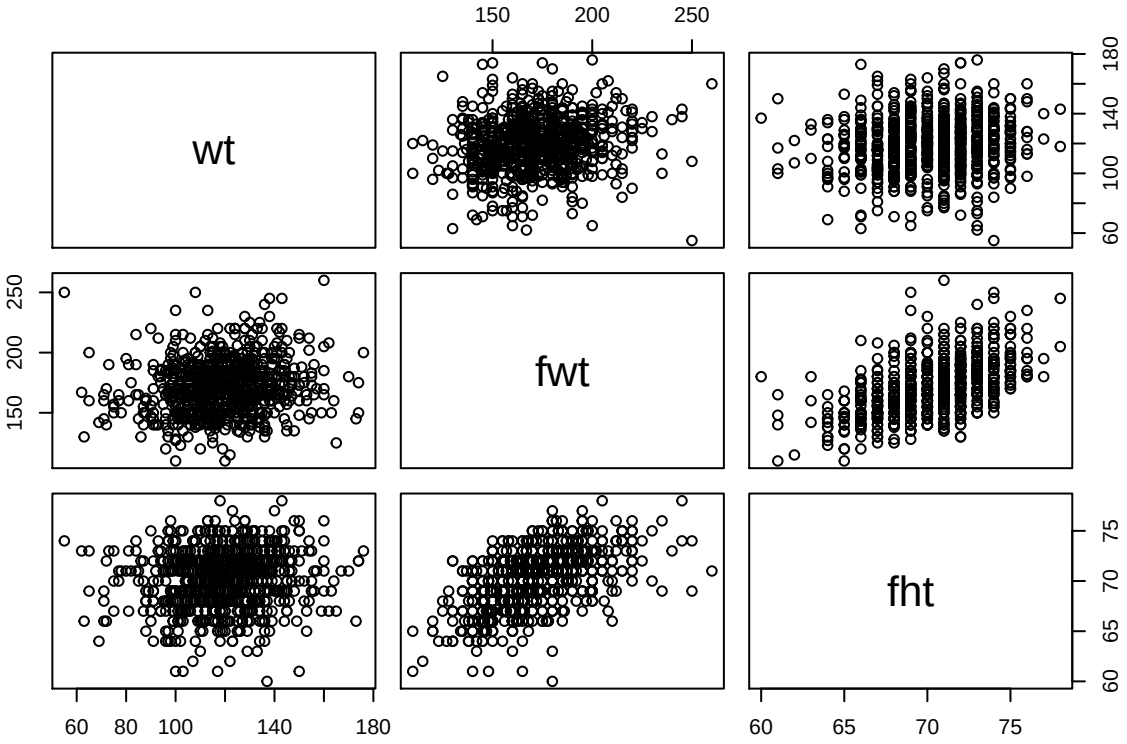


Figure 1.1

```
library(mice)
```

```
## Loading required package: lattice
```

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

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

```
# removing father's weight and height from our dataset
chds$fwt <- NULL
chds$fht <- NULL

# imputing missing data
system.time({
  chds_imp = mice(chds, m=5, printFlag=FALSE, maxit=50, seed=331)
})
```

```
##    user  system elapsed
##  31.298   1.736  33.348
```

```r
# extracting five imputed datasets
chds1 <- complete(chds_imp, 1)
chds2 <- complete(chds_imp, 2)
chds3 <- complete(chds_imp, 3)
chds4 <- complete(chds_imp, 4)
chds5 <- complete(chds_imp, 5)
```

1.4 Converting categorical covariates into factors

```
# constructing smoke level factors
smoke_levels <- c("never", "until_pregnancy", "used_to_not_anymore", "smokes_now")
encode_smoke <- function(chds)
{
  smoke2 <- smoke_levels[chds$smoke+1]
  smoke2 <- factor(smoke2, levels=smoke_levels)
  chds$smoke <- smoke2

  return(chds)
}

chds1 <- encode_smoke(chds1)
chds2 <- encode_smoke(chds2)
chds3 <- encode_smoke(chds3)
chds4 <- encode_smoke(chds4)
chds5 <- encode_smoke(chds5)

# constructing med factors
med_levels <- c("elementary", "middle", "high", "high+trade", "high+college", "college_grad",
                "trade", "high_unclear")
encode_med <- function(chds)
{
  med2 <- med_levels[chds$med+1]
  med2 <- factor(med2, levels=med_levels)
  chds$med <- med2
  return(chds)
}

chds1 <- encode_med(chds1)
chds2 <- encode_med(chds2)
chds3 <- encode_med(chds3)
chds4 <- encode_med(chds4)
chds5 <- encode_med(chds5)

fed_levels <- c("elementary", "middle", "high", "high+trade", "high+college", "college_grad",
                "trade", "high_unclear")
# constructing fed factors
encode_fed <- function(chds)
{
  fed2 <- fed_levels[chds$fed+1]
  fed2 <- factor(fed2, levels=fed_levels)
  chds$fed <- fed2
  return(chds)
}

chds1 <- encode_fed(chds1)
chds2 <- encode_fed(chds2)
chds3 <- encode_fed(chds3)
chds4 <- encode_fed(chds4)
chds5 <- encode_fed(chds5)
```

```
# constructiong marital factors
marital_levels <- c("married", "legally_separated", "divorced", "widowed", "never_married",
encode_marital <- function(chds)
{
  marital2 <- marital_levels[chds$marital]
  marital2 <- factor(marital2, levels=marital_levels)
  chds$marital <- marital2
  return(chds)
}

chds1 <- encode_marital(chds1)
chds2 <- encode_marital(chds2)
chds3 <- encode_marital(chds3)
chds4 <- encode_marital(chds4)
chds5 <- encode_marital(chds5)

# constructing time factors
time_levels <- c("never_smoked", "<1year", "1-2years", "2-3years", "3-4years", "5-9years", ">10years",
encode_time <- function(chds)
{
  time2 <- time_levels[chds$time+1]
  time2 <- factor(time2, levels=time_levels)
  chds$time <- time2
  return(chds)
}

chds1 <- encode_time(chds1)
chds2 <- encode_time(chds2)
chds3 <- encode_time(chds3)
chds4 <- encode_time(chds4)
chds5 <- encode_time(chds5)

# constructing number factors
number_levels <- c("never_smoked", "1-4", "5-9", "10-14", "15-19", "20-29", "30-39", "40-60", ">60",
encode_number <- function(chds)
{
  number2 <- number_levels[chds$number+1]
  number2 <- factor(number2, levels=number_levels)
  chds$number <- number2
  return(chds)
}

chds1 <- encode_number(chds1)
chds2 <- encode_number(chds2)
chds3 <- encode_number(chds3)
chds4 <- encode_number(chds4)
chds5 <- encode_number(chds5)

# constructing income factors
income_levels <- c("<2500", "2500-4999", "5000-7499", "7500-9999", "10000-12499", "12500-14999",
                   "15000-17499", "17500-19999", "20000-22499", ">=22500")
encode_income <- function(chds)
{
```

```
  income2 <- income_levels[chds$income+1]
  income2 <- factor(income2, levels=income_levels)
  chds$income <- income2
  return(chds)
}

chds1 <- encode_income(chds1)
chds2 <- encode_income(chds2)
chds3 <- encode_income(chds3)
chds4 <- encode_income(chds4)
chds5 <- encode_income(chds5)

# constructing meth and feth factors
eth_levels <- c("Caucasian", "Mexican", "African-American", "Asian", "Mixed", "Other")
encode_eth <- function(chds)
{
  max_eth <- max(chds$meth) + 1
  eth_threshold <- c(0, 6, 7, 8, 9, 10, 11)
  meth <- cut(chds$meth, breaks=eth_threshold, labels= eth_levels, right=FALSE)
  feth <- cut(chds$feth, breaks=eth_threshold, labels= eth_levels, right=FALSE)
  chds$meth <- meth
  chds$feth <- feth
  chds$meth <- factor(chds$meth, levels=eth_levels)
  chds$feth <- factor(chds$feth, levels=eth_levels)
  return(chds)
}

chds1 <- encode_eth(chds1)
chds2 <- encode_eth(chds2)
chds3 <- encode_eth(chds3)
chds4 <- encode_eth(chds4)
chds5 <- encode_eth(chds5)
```

1.5 Handling covariate categories with few observations

```
# Changing factor levels for marital
change_marital_factors <- function(chds)
{
  chds$marital[chds$marital != "married"] <- "not_married"
  chds$marital <- droplevels(chds$marital)
  return(chds)
}

chds1 <- change_marital_factors(chds1)
chds2 <- change_marital_factors(chds2)
chds3 <- change_marital_factors(chds3)
chds4 <- change_marital_factors(chds4)
chds5 <- change_marital_factors(chds5)

# Changing factor levels for meth
change_meth_factor_levels <- function(chds)
{
  chds$meth[chds$meth == "Mexican"] <- "Other"
  chds$meth[chds$meth == "Asian"] <- "Other"
  chds$meth[chds$meth == "Mixed"] <- "Other"
  chds$meth <- droplevels(chds$meth)
  return(chds)
}

chds1 <- change_meth_factor_levels(chds1)
chds2 <- change_meth_factor_levels(chds2)
chds3 <- change_meth_factor_levels(chds3)
chds4 <- change_meth_factor_levels(chds4)
chds5 <- change_meth_factor_levels(chds5)

# Changing factor levels for med
change_med_levels <- function(chds)
{
  # Adding new category "upto_medical_school" in categories for med
  med_levels <- c("upto_high_school", med_levels)
  chds$med <- factor(chds$med, levels=med_levels)

  # Shift observations in category "elementary" to category "upto_high_school"
  chds$med[chds$med == "elementary"] <- "upto_high_school"

  # Shift observations in category "middle" to category "upto_high_school"
  chds$med[chds$med == "middle"] <- "upto_high_school"

  # Shift observations in category "high" to category "upto_high_school"
  chds$med[chds$med == "high"] <- "upto_high_school"

  # Removing all observations with level "high_unclear"
  chds <- chds[chds$med != "high_unclear", ]

  # Drop all categories which have no observation and "high_unclear" category
  chds$med <- droplevels(chds$med)
```

```
  return(chds)
}

chds1 <- change_med_levels(chds1)
chds2 <- change_med_levels(chds2)
chds3 <- change_med_levels(chds3)
chds4 <- change_med_levels(chds4)
chds5 <- change_med_levels(chds5)

# Changing factor levels for feth
change_feth_levels <- function(chds)
{
  # Converting feth into a categorical covariate with categories "Caucasian",
  # "African-American", "Other"
  chds$feth[chds$feth == "Mexican"] <- "Other"
  chds$feth[chds$feth == "Asian"] <- "Other"
  chds$feth[chds$feth == "Mixed"] <- "Other"
  chds$feth <- droplevels(chds$feth)

  return(chds)
}

chds1 <- change_feth_levels(chds1)
chds2 <- change_feth_levels(chds2)
chds3 <- change_feth_levels(chds3)
chds4 <- change_feth_levels(chds4)
chds5 <- change_feth_levels(chds5)

# Changing factor levels for fed
change_fed_levels <- function(chds)
{
  # Handling fed
  # adding new category "upto_medical_school" in categories for fed
  fed_levels <- c("upto_high_school", fed_levels)
  chds$fed <- factor(chds$fed, levels=fed_levels)

  # shift observations in category "elementary" to category "upto_high_school"
  chds$fed[chds$fed == "elementary"] <- "upto_high_school"

  # shift observations in category "middle" to category "upto_high_school"
  chds$fed[chds$fed == "middle"] <- "upto_high_school"

  # shift observations in category "high" to category "upto_high_school"
  chds$fed[chds$fed == "high"] <- "upto_high_school"

  # Removing all observations with level "trade"
  chds <- chds[chds$fed != "trade",]

  # Removing all observations with level "high_school_unclear"
  chds <- chds[chds$fed != "high_unclear", ]

  # drop all categories which have no observation and "high_unclear" category
  chds$fed <- droplevels(chds$fed)
```

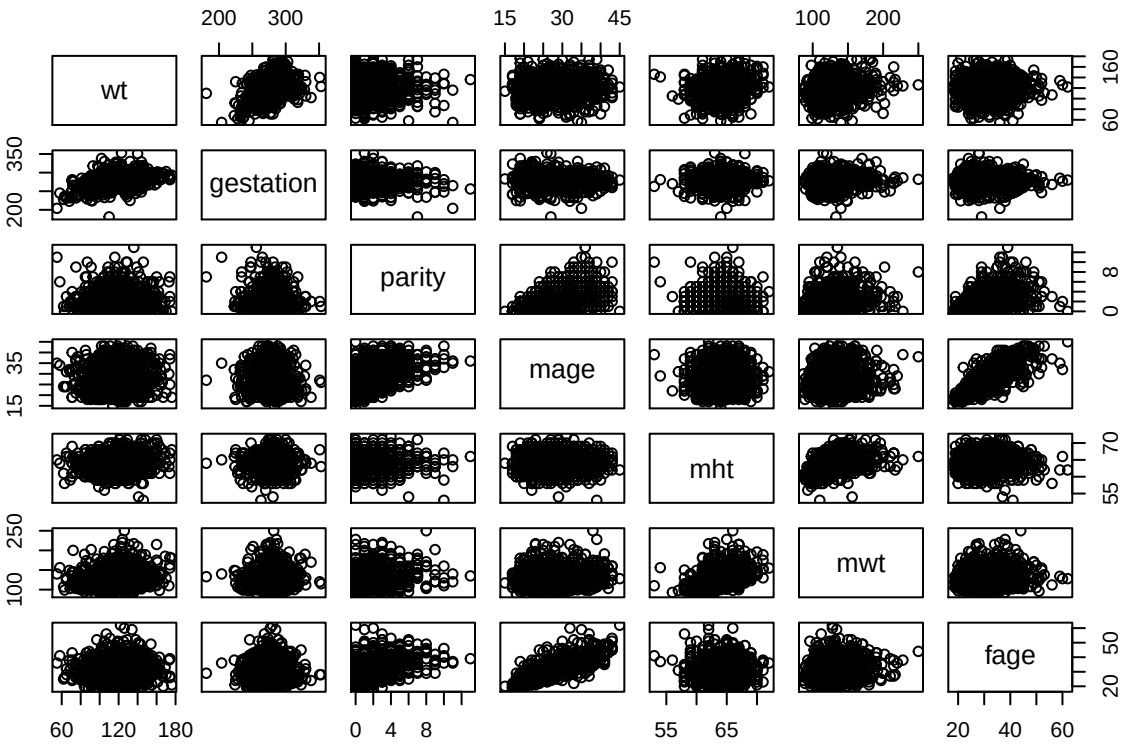
```
  return(chds)
}

chds1 <- change_fed_levels(chds1)
chds2 <- change_fed_levels(chds2)
chds3 <- change_fed_levels(chds3)
chds4 <- change_fed_levels(chds4)
chds5 <- change_fed_levels(chds5)

# Changing factor levels for income
change_income_levels <- function(chds)
{
  # Adding new categories "0-4999" and ">=20000" to income
  income_levels <- c("0-4999", income_levels, "5000-14999", ">=15000")

  chds$income <- factor(chds$income, levels=income_levels)
  chds$income[chds$income == "<2500"] <- "0-4999"
  chds$income[chds$income == "2500-4999"] <- "0-4999"

  chds$income[chds$income == "5000-7499"] <- "5000-14999"
  chds$income[chds$income == "7500-9999"] <- "5000-14999"
  chds$income[chds$income == "10000-12499"] <- "5000-14999"
  chds$income[chds$income == "12500-14999"] <- "5000-14999"

  chds$income[chds$income == "15000-17499"] <- ">=15000"
  chds$income[chds$income == "17500-19999"] <- ">=15000"
  chds$income[chds$income == "20000-22499"] <- ">=15000"
  chds$income[chds$income == ">=22500"] <- ">=15000"

  # drop all income categories which have no observation
  chds$income <- droplevels(chds$income)

  return(chds)
}

chds1 <- change_income_levels(chds1)
chds2 <- change_income_levels(chds2)
chds3 <- change_income_levels(chds3)
chds4 <- change_income_levels(chds4)
chds5 <- change_income_levels(chds5)

# Changing factor levels for time
change_time_levels <- function(chds)
{
  # Adding new categories "0-2years" ">2years" to time
  time_levels <- c("0-2years", ">2years", time_levels)

  chds$time <- factor(chds$time, levels=time_levels)
  chds$time[chds$time == "<1year"] <- "0-2years"
  chds$time[chds$time == "1-2years"] <- "0-2years"

  chds$time[chds$time == "2-3years"] <- ">2years"
  chds$time[chds$time == "3-4years"] <- ">2years"
```

```
  chds$time[chds$time == "5-9years"] <- ">2years"
  chds$time[chds$time == ">10years"] <- ">2years"

  # Removing all observations with level "quit_but_dont_know_when"
  chds <- chds[chds$time != "quit_but_dont_know_when",]

  # Removing all observations with level "still_smokes"
  chds <- chds[chds$time != "still_smokes",]

  # Drop all time categories which have no observation
  chds$time <- droplevels(chds$time)

  return(chds)
}

chds1 <- change_time_levels(chds1)
chds2 <- change_time_levels(chds2)
chds3 <- change_time_levels(chds3)
chds4 <- change_time_levels(chds4)
chds5 <- change_time_levels(chds5)


# Changing factor levels for time
change_number_levels <- function(chds)
{
  #Adding new categories "1-9" ">=10" to number
  number_levels <- c(number_levels, "1-9", ">=10")

  chds$number <- factor(chds$number, levels=number_levels)
  chds$number[chds$number == "1-4"] <- "1-9"
  chds$number[chds$number == "5-9"] <- "1-9"

  chds$number[chds$number == "10-14"] <- ">=10"
  chds$number[chds$number == "10-14"] <- ">=10"
  chds$number[chds$number == "15-19"] <- ">=10"
  chds$number[chds$number == "20-29"] <- ">=10"
  chds$number[chds$number == "30-39"] <- ">=10"
  chds$number[chds$number == "40-60"] <- ">=10"
  chds$number[chds$number == ">60"] <- ">=10"

  # Removing all observations with level "smoked_but_dont_know_how_much"
  chds <- chds[chds$number != "smoked_but_dont_know_how_much",]

  # Drop all number categories which have no observation
  chds$number <- droplevels(chds$number)

  return(chds)
}

chds1 <- change_number_levels(chds1)
chds2 <- change_number_levels(chds2)
chds3 <- change_number_levels(chds3)
chds4 <- change_number_levels(chds4)
```

```r
chds5 <- change_number_levels(chds5)
```

1.6 pairs plot for all the continuous covariates

```
pairs(~ wt + gestation + parity + mage + mht + mwt + fage, data=chds1)
```




```
library(plyr)
# counting distinct values in covariates "parity" and "mht"
count(chds1, vars="parity")
```

```
##    parity freq
## 1       0  311
## 2       1  302
## 3       2  236
## 4       3  165
## 5       4   80
## 6       5   51
## 7       6   30
## 8       7   14
## 9       8    8
## 10      9    6
## 11     10    4
## 12     11    2
## 13     13    1
```

```
count(chds1, vars="mht")
```

```
##    mht freq
## 1   53    1
## 2   54    1
## 3   56    1
## 4   57    1
## 5   58   10
```

```
## 6   59   28
## 7   60   53
## 8   61  103
## 9   62  131
## 10  63  167
## 11  64  179
## 12  65  182
## 13  66  152
## 14  67  108
## 15  68   54
## 16  69   20
## 17  70   13
## 18  71    5
## 19  72    1
```

```r
# encode parity
parity_levels <- c("0", "1", "2", ">=3")
parity_levels <- c("no_prev_preg", "one_prev_preg", "two_prev_preg", "at_least_three_prev_preg")
encode_parity <- function(chds)
{
  max_parity <- max(chds$parity) + 1
  parity_threshold <- c(0, 1, 2, 3, max_parity)
  parity2 <- cut(chds$parity, breaks=parity_threshold, labels=parity_levels, right=FALSE)
  chds$parity <- parity2
  chds$parity <- factor(chds$parity, levels=parity_levels)

  return(chds)
}

chds1 <- encode_parity(chds1)
chds2 <- encode_parity(chds2)
chds3 <- encode_parity(chds3)
chds4 <- encode_parity(chds4)
chds5 <- encode_parity(chds5)

# encode mht
mht_levels <- c("at_most_60", "61-63", "64-66", "at_least_67")
encode_mht <- function(chds)
{
  max_mht <- max(chds$mht)
  mht_threshold <- c(0, 60, 63, 66, max_mht)
  mht2 <- cut(chds$mht, breaks=mht_threshold, labels= mht_levels, right=TRUE)
  chds$mht <- mht2
  chds$mht <- factor(chds$mht, levels=mht_levels)

  return(chds)
}

chds1 <- encode_mht(chds1)
chds2 <- encode_mht(chds2)
chds3 <- encode_mht(chds3)
chds4 <- encode_mht(chds4)
chds5 <- encode_mht(chds5)
```

```r
# print comprehensive summary of factor encoded dataset
summary(chds1)
```

```
##        wt          gestation                            parity
##  Min.   : 55.0   Min.   :181.0   no_prev_preg              :311
##  1st Qu.:109.0   1st Qu.:272.0   one_prev_preg             :302
##  Median :120.0   Median :280.0   two_prev_preg             :236
##  Mean   :119.7   Mean   :279.4   at_least_three_prev_preg:361
##  3rd Qu.:131.0   3rd Qu.:288.0
##  Max.   :176.0   Max.   :353.0
##                meth          mage                      med
##  Caucasian       :852   Min.   :15.00   upto_high_school:637
##  African-American:241   1st Qu.:23.00   high+trade      : 64
##  Other           :117   Median :26.00   high+college    :293
##                         Mean   :27.17   college_grad    :216
##                         3rd Qu.:31.00
##                         Max.   :45.00
##          mht           mwt                       feth          fage
##  at_most_60 : 95   Min.   : 87.0   Caucasian       :855   Min.   :18.00
##  61-63      :401   1st Qu.:114.0   African-American:262   1st Qu.:25.00
##  64-66      :513   Median :125.0   Other           : 93   Median :29.00
##  at_least_67:201   Mean   :128.6                          Mean   :30.29
##                    3rd Qu.:139.0                          3rd Qu.:34.00
##                    Max.   :250.0                          Max.   :62.00
##               fed             marital            income
##  upto_high_school:567   married    :1184   0-4999    :254
##  high+trade      : 35   not_married:  26   5000-14999:682
##  high+college    :264                      >=15000   :274
##  college_grad    :344
##
##
##                  smoke                   time              number
##  never              :539   0-2years        :580   never_smoked:543
##  until_pregnancy    :486   >2years         : 76   1-9         :319
##  used_to_not_anymore: 94   never_smoked    :539   >=10        :348
##  smokes_now         : 91   during_pregnancy: 15
##
##
```

Table 1.2

2 Model selection

2.1 Detecting and Removing Data Inconsistency

```
# Removing inconsistencies between number and smoke
chds1 <- chds1[!(chds1$smoke == "smokes_now" & chds1$number == "never_smoked"), ]
chds1 <- chds1[!(chds1$smoke == "until_pregnancy" & chds1$number == "never_smoked"), ]

chds2 <- chds2[!(chds2$smoke == "until_pregnancy" & chds2$number == "never_smoked"), ]

chds4 <- chds4[!(chds4$smoke == "smokes_now" & chds4$number == "never_smoked"), ]
chds4 <- chds1[!(chds4$smoke == "until_pregnancy" & chds4$number == "never_smoked"), ]


table(chds1[c("number", "smoke")])
```

```
##               smoke
## number         never until_pregnancy used_to_not_anymore smokes_now
##   never_smoked   539               0                   0          0
##   1-9              0             198                  63         58
##   >=10             0             286                  31         31
```

```
table(chds2[c("number", "smoke")])
```

```
##               smoke
## number         never until_pregnancy used_to_not_anymore smokes_now
##   never_smoked   539               0                   0          0
##   1-9              0             198                  64         60
##   >=10             0             285                  31         32
```

```
table(chds3[c("number", "smoke")])
```

```
##               smoke
## number         never until_pregnancy used_to_not_anymore smokes_now
##   never_smoked   536               1                   0          0
##   1-9              0             197                  63         59
##   >=10             0             291                  31         32
```

```
table(chds4[c("number", "smoke")])
```

```
##               smoke
## number         never until_pregnancy used_to_not_anymore smokes_now
##   never_smoked   539               0                   0          0
##   1-9              0             198                  63         58
##   >=10             0             286                  31         31
```

```
table(chds5[c("number", "smoke")])
```

```
##               smoke
## number         never until_pregnancy used_to_not_anymore smokes_now
##   never_smoked   537               0                   0          0
##   1-9              0             200                  63         61
##   >=10             0             284                  32         33
```

2.2 Automated model selection

```
forward_selection <- function(chds)
{
  # minimal model: intercept only
  Mminimal <- lm(wt ~ 1, data=chds)
  # all main effects and interaction
  Mmaximal <- lm(wt ~ (.)^2, data=chds)

  system.time({
    Mfwd <- step(object=Mminimal,
                 scope=list(lower=Mminimal, upper=Mmaximal),
                 direction="forward",
                 trace=FALSE)
  })

  return(Mfwd)
}

Mfwd1 <- forward_selection(chds1)
Mfwd2 <- forward_selection(chds2)
Mfwd3 <- forward_selection(chds3)
Mfwd4 <- forward_selection(chds4)
Mfwd5 <- forward_selection(chds5)

stepwise_selection <- function(chds)
{
# minimal model: intercept only
  Mminimal <- lm(wt ~ 1, data=chds)
  # all main effects and interaction
  Mmaximal <- lm(wt ~ (.)^2, data=chds)
  # starting model for stepwise selection
  Mstart <- lm(wt ~ ., data=chds)

  system.time({
    Mstep <- step(object=Mstart,
                 scope=list(lower=Mminimal, upper=Mmaximal),
                 direction="both",
                 trace=FALSE)
  })

  return(Mstep)
}

Mstep1 <- stepwise_selection(chds1)
Mstep2 <- stepwise_selection(chds2)
Mstep3 <- stepwise_selection(chds3)
Mstep4 <- stepwise_selection(chds4)
Mstep5 <- stepwise_selection(chds5)
```

2.3 Treatment of NAs in Beta_hat

```
# Extracting regression coeffecients
Beta_hat_Mfwd1 <- coef(Mfwd1)
Beta_hat_Mfwd2 <- coef(Mfwd2)
Beta_hat_Mfwd3 <- coef(Mfwd3)
Beta_hat_Mfwd4 <- coef(Mfwd4)
Beta_hat_Mfwd5 <- coef(Mfwd5)

Beta_hat_Mstep1 <- coef(Mstep1)
Beta_hat_Mstep2 <- coef(Mstep2)
Beta_hat_Mstep3 <- coef(Mstep3)
Beta_hat_Mstep4 <- coef(Mstep4)
Beta_hat_Mstep5 <- coef(Mstep5)

# Checking for any NAs in regression coeffcients
names(Beta_hat_Mfwd1)[is.na(Beta_hat_Mfwd1)]
```

```
## [1] "number>=10"                 "timenever_smoked"
## [3] "timeduring_pregnancy"       "gestation:timenever_smoked"
```

```
names(Beta_hat_Mfwd2)[is.na(Beta_hat_Mfwd2)]
```

```
## [1] "number>=10"                 "timenever_smoked"
## [3] "timeduring_pregnancy"       "gestation:timenever_smoked"
```

```
names(Beta_hat_Mfwd3)[is.na(Beta_hat_Mfwd3)]
```

```
## [1] "timenever_smoked"           "timeduring_pregnancy"
## [3] "gestation:timenever_smoked"
```

```
names(Beta_hat_Mfwd4)[is.na(Beta_hat_Mfwd4)]
```

```
## [1] "number>=10"                 "timenever_smoked"
## [3] "timeduring_pregnancy"       "gestation:timenever_smoked"
```

```
names(Beta_hat_Mfwd5)[is.na(Beta_hat_Mfwd5)]
```

```
## [1] "number>=10"           "timenever_smoked"     "timeduring_pregnancy"
```

```
names(Beta_hat_Mstep1)[is.na(Beta_hat_Mstep1)]
```

```
## [1] "timenever_smoked"     "timeduring_pregnancy" "number>=10"
```

```
names(Beta_hat_Mstep2)[is.na(Beta_hat_Mstep2)]
```

```
## [1] "timenever_smoked"     "timeduring_pregnancy" "number>=10"
```

```
names(Beta_hat_Mstep3)[is.na(Beta_hat_Mstep3)]
```

```
## [1] "timenever_smoked"     "timeduring_pregnancy"
```

```
names(Beta_hat_Mstep4)[is.na(Beta_hat_Mstep4)]
```

```
## [1] "timenever_smoked"     "timeduring_pregnancy" "number>=10"
```

```
names(Beta_hat_Mstep5)[is.na(Beta_hat_Mstep5)]
```

```
## [1] "timenever_smoked"     "timeduring_pregnancy" "number>=10"
```

2.4 Examining regression coeffecients in our fitted models

```
formula(Mfwd1)
```

```
## wt ~ gestation + smoke + mht + meth + parity + number + mwt +
##     time + income + gestation:number + gestation:mwt + meth:mwt +
##     gestation:mht + gestation:time + gestation:income + meth:income
## <environment: 0x7fc96251cb58>
```

```
formula(Mfwd2)
```

```
## wt ~ gestation + smoke + mht + meth + parity + number + mwt +
##     time + gestation:number + meth:mwt + gestation:mwt + gestation:mht +
##     gestation:time
## <environment: 0x7fc95f4800b0>
```

```
formula(Mfwd3)
```

```
## wt ~ gestation + smoke + mht + meth + parity + number + mwt +
##     time + gestation:number + meth:mwt + gestation:mwt + gestation:mht +
##     gestation:time
## <environment: 0x7fc95f656c38>
```

```
formula(Mfwd4)
```

```
## wt ~ gestation + smoke + mht + meth + parity + number + mwt +
##     time + income + gestation:number + gestation:mwt + meth:mwt +
##     gestation:mht + gestation:time + gestation:income + meth:income
## <environment: 0x7fc96040ce68>
```

```
formula(Mfwd5)
```

```
## wt ~ gestation + smoke + mht + meth + parity + number + mwt +
##     time + gestation:number + meth:mwt + gestation:mwt + gestation:mht
## <environment: 0x7fc95de89600>
```

```
formula(Mstep1)
```

```
## wt ~ gestation + parity + meth + mage + med + mht + mwt + income +
##     smoke + time + number + gestation:time + gestation:mage +
##     meth:income + gestation:med + med:mwt + gestation:meth +
##     gestation:mht + mage:med + meth:mwt + gestation:mwt + mage:mwt
## <environment: 0x7fc960413060>
```

```
formula(Mstep2)
```

```
## wt ~ gestation + parity + meth + mht + mwt + fage + income +
##     smoke + time + number + gestation:income + meth:mwt + gestation:mwt +
##     gestation:mht + meth:fage + gestation:fage + meth:income +
##     gestation:time
## <environment: 0x7fc95f64d350>
```

```
formula(Mstep3)
```

```
## wt ~ gestation + parity + meth + med + mht + mwt + income + smoke +
##     time + number + gestation:income + gestation:number + gestation:med +
##     med:mwt + meth:mwt + meth:income
## <environment: 0x7fc9603f5550>
```

```r
formula(Mstep4)
```

```
## wt ~ gestation + parity + meth + mage + med + mht + mwt + income +
##     smoke + time + number + gestation:time + gestation:mage +
##     meth:income + gestation:med + med:mwt + gestation:meth +
##     gestation:mht + mage:med + meth:mwt + gestation:mwt + mage:mwt
## <environment: 0x7fc95f60e828>
```

```r
formula(Mstep5)
```

```
## wt ~ gestation + parity + meth + mage + mht + mwt + fage + income +
##     smoke + time + number + gestation:income + gestation:number +
##     mwt:income + gestation:mage + gestation:mwt + meth:mwt +
##     meth:fage + mage:number + gestation:mht
## <environment: 0x7fc95efa89f0>
```

2.5 Selecting Best Forward and Stepwise Selection Models

```r
# We have four different models from forward and stepwise selection each
# therefore, the following code is written
cv <- function(chds, split, n_iter, M1, M2, M3, M4, names, set_name)
{
  n <- nrow(chds)

  mspe1 <- rep(NA, n_iter)
  mspe2 <- rep(NA, n_iter)
  mspe3 <- rep(NA, n_iter)
  mspe4 <- rep(NA, n_iter)

  for (i in 1:n_iter) {

    # shuffle dataset
    shuffled <- chds[sample(n), ]

    # extract training set
    train_ind <- 1:round(split * n)
    train <- shuffled[train_ind, ]

    # extract test set
    test_ind <- (round(split * n) + 1):n
    test <- shuffled[test_ind, ]

    # refit the models on the subset of training data
    M1_cv <- lm(formula(M1), data=train)
    M2_cv <- lm(formula(M2), data=train)
    M3_cv <- lm(formula(M3), data=train)
    M4_cv <- lm(formula(M4), data=train)

    # out-of-sample residuals for all five models
    # that is, testing data - predictions with training parameters
    M1_res <- test$wt - predict(M1_cv, newdata=test)
    M2_res <- test$wt - predict(M2_cv, newdata=test)
    M3_res <- test$wt - predict(M3_cv, newdata=test)
```

```
    M4_res <- test$wt - predict(M4_cv, newdata=test)

    # mean-square prediction errors
    mspe1[i] <- mean(M1_res^2)
    mspe2[i] <- mean(M2_res^2)
    mspe3[i] <- mean(M3_res^2)
    mspe4[i] <- mean(M4_res^2)
  }

  # drop any NA's in our computed mspe
  mspe1 <- mspe1[!is.na(mspe1)]
  mspe2 <- mspe2[!is.na(mspe2)]
  mspe3 <- mspe3[!is.na(mspe3)]
  mspe4 <- mspe4[!is.na(mspe4)]

  # box-plot and histogram
  # plot rMSPE and out-of-sample log(Lambda)
  par(mfrow = c(1,2))
  par(mar = c(4.5, 4.5, .1, .1))
  boxplot(x = list(sqrt(mspe1), sqrt(mspe2), sqrt(mspe3), sqrt(mspe4)), names = names, cex = .7,
          ylab = expression(sqrt(MSPE)), col = c("yellow", "orange", "red", "blue"), main=set_name)

  return(c(mean(mspe1), mean(mspe2), mean(mspe3), mean(mspe4)))
}

# Stepwise Selection Final Model Competition
n_iter <- 2000
split <- 0.80

snames=expression(M[step1], M[step2], M[step3], M[step5])
fnames <- expression(M[fwd1], M[fwd2], M[fwd3], M[fwd5])
```

```r
# Cross validation of unique step-wise selection models on chds1
cv(chds1, split, n_iter, Mstep1, Mstep2, Mstep3, Mstep5, snames, "chds1")
```

```
## [1] 237.7413 236.2284 235.8677 238.1386
```

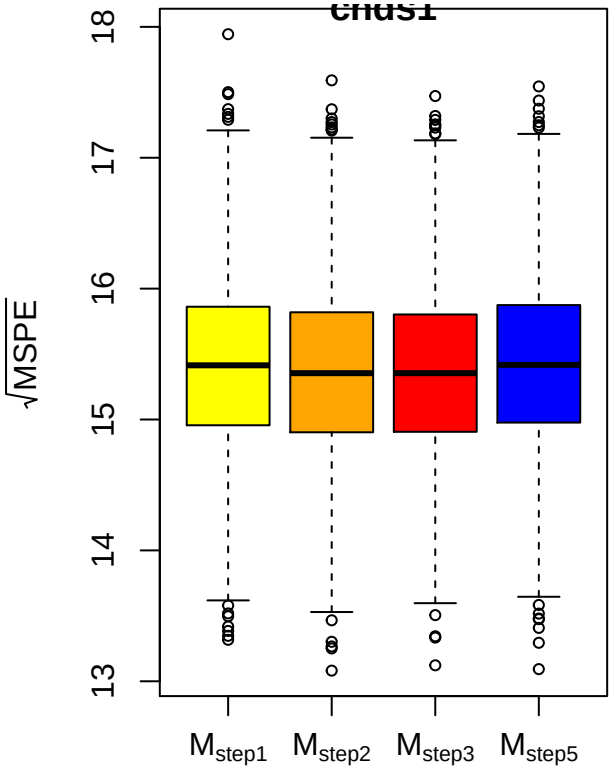


Figure 2.5.1

```r
# Cross validation of unique step-wise selection models on chds2
cv(chds2, split, n_iter, Mstep1, Mstep2, Mstep3, Mstep5, snames, "chds2")
```

```
## [1] 236.8506 233.8629 233.7576 235.0977
```

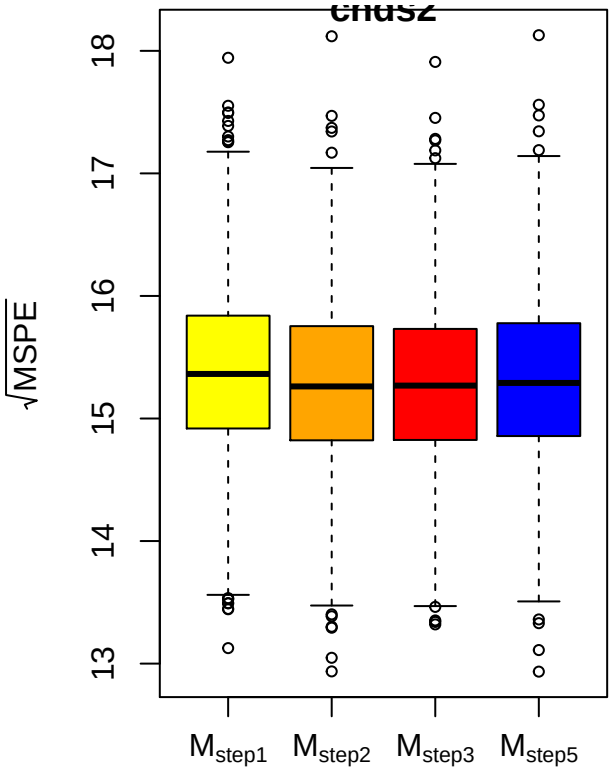


Figure 2.5.2

```r
# Cross validation of unique step-wise selection models on chds3
cv(chds3, split, n_iter, Mstep1, Mstep2, Mstep3, Mstep5, snames, "chds3")
```

```
## [1] 238.7438 235.9643 241.6236 242.3106
```

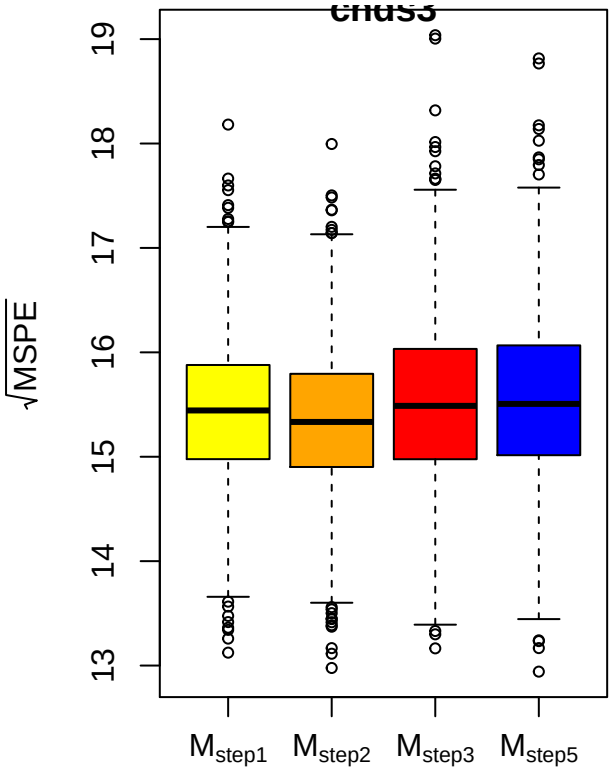


Figure 2.5.3

```r
# Cross validation of unique step-wise selection models on chds4
cv(chds4, split, n_iter, Mstep1, Mstep2, Mstep3, Mstep5, snames, "chds4")
```

```
## [1] 237.2352 236.3139 235.4707 237.9770
```

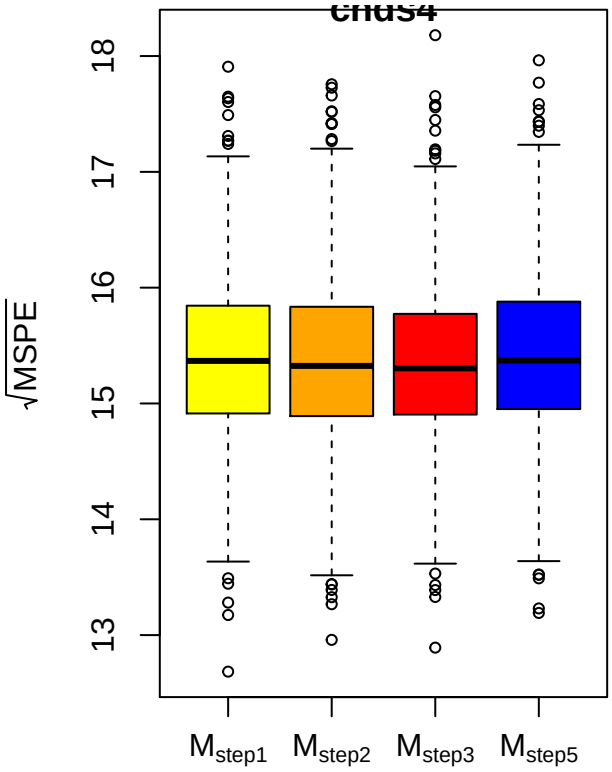


Figure 2.5.4

```
# Cross validation of unique step-wise selection models on chds5
cv(chds5, split, n_iter, Mstep1, Mstep2, Mstep3, Mstep5, snames, "chds5")
```

```
## [1] 236.0238 235.2492 234.0324 234.6332
```

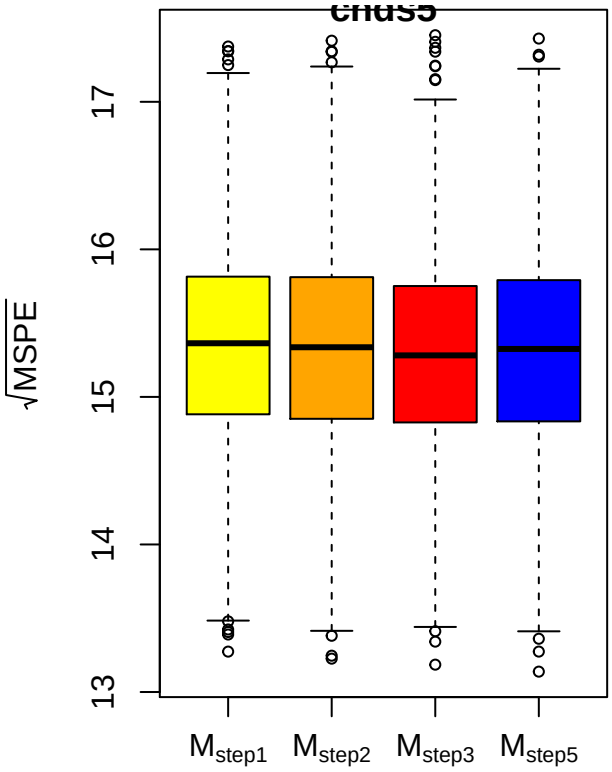


Figure 2.5.5

```
# Cross validation of unique forward selection models on chds1
cv(chds1, split, n_iter, Mfwd1, Mfwd2, Mfwd3, Mfwd5, fnames, "chds1")
```

```
## [1] 237.0056 237.7224 237.7224 237.7557
```

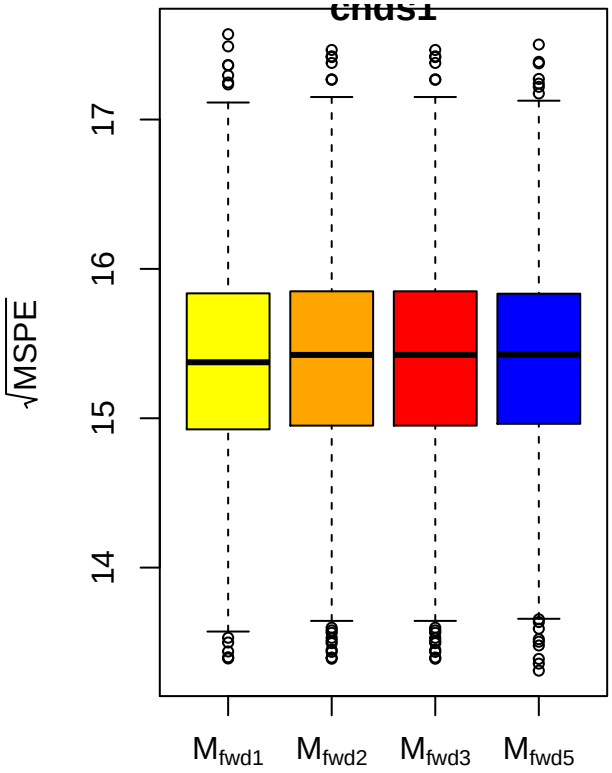


Figure 2.5.6

```r
# Cross validation of unique forward selection models on chds2
cv(chds2, split, n_iter, Mfwd1, Mfwd2, Mfwd3, Mfwd5, fnames, "chds2")
```

```
## [1] 233.9227 236.0313 236.0313 235.4989
```

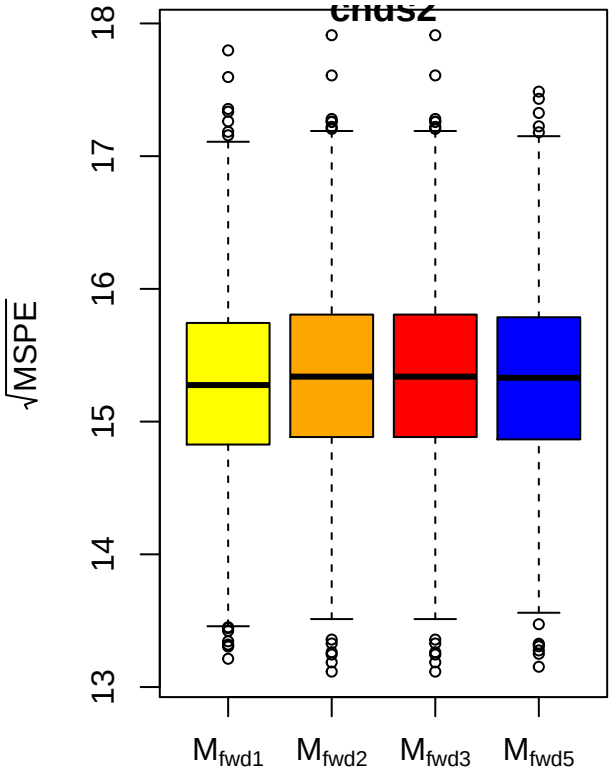


Figure 2.5.7

```
# Cross validation of unique forward selection models on chds3
cv(chds3, split, n_iter, Mfwd1, Mfwd2, Mfwd3, Mfwd5, fnames, "chds3")
```

```
## [1] 244.6117 246.9143 246.9143 245.7439
```

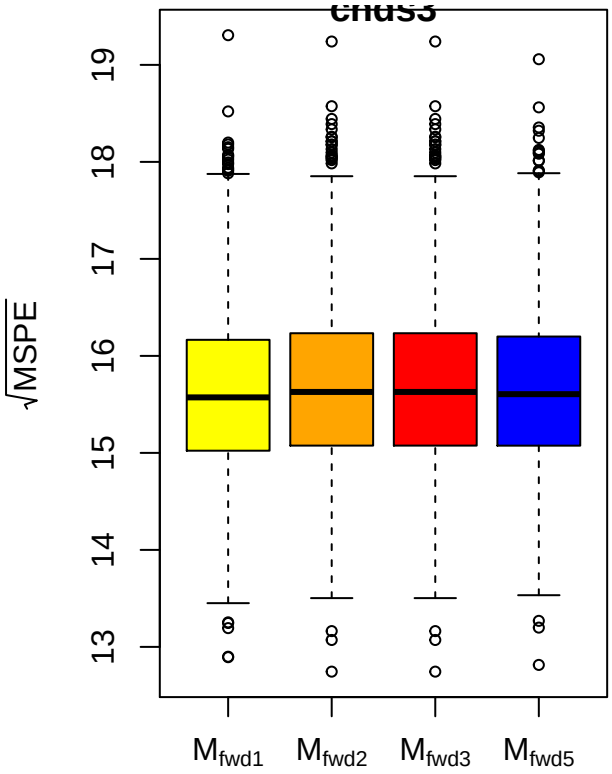


Figure 2.5.8

```
# Cross validation of unique forward selection models on chds4
cv(chds4, split, n_iter, Mfwd1, Mfwd2, Mfwd3, Mfwd5, fnames, "chds4")
```

```
## [1] 237.2637 237.7336 237.7336 237.5722
```

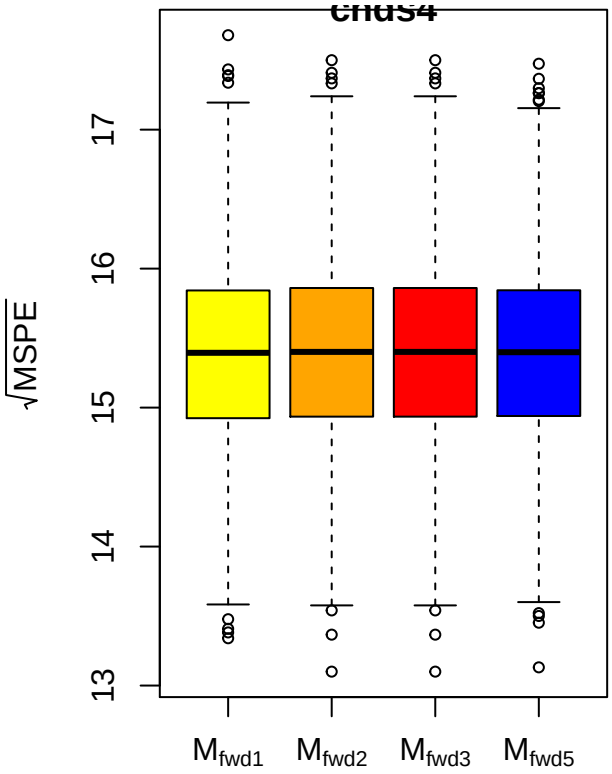


Figure 2.5.9

```
# Cross validation of unique forward selection models on chds5
cv(chds5, split, n_iter, Mfwd1, Mfwd2, Mfwd3, Mfwd5, fnames, "chds5")
```

```
## [1] 236.4525 237.1031 237.1031 236.4362
```

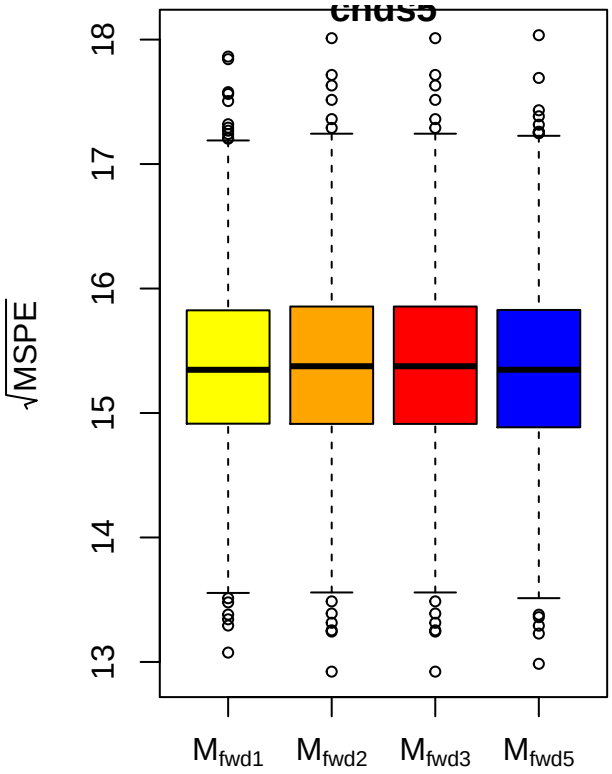


Figure 2.5.10

2.6 Pooling Best Forward and Stepwise Selection Models

```
# pooling Mfwd1
Mfwd_best <- with(chds_imp,
                  lm(wt ~ gestation + smoke + mht + meth + parity + number + mwt + time + meth:mwt
                         + gestation:number + gestation:mwt + gestation:mht + gestation:time
                    )
                  )
summary(pool(Mfwd_best))
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.4
```

```
##                        estimate    std.error  statistic        df
## (Intercept)      286.635798112 2.121318e+02  1.3512155 1184.9647
## gestation         -0.854048943 7.592879e-01 -1.1248025 1186.5255
## smoke              2.951990243 1.259158e+00  2.3444168 1104.1435
## mht               -6.515368979 3.524773e+00 -1.8484509 1168.6582
## meth               1.186008557 8.444018e-01  1.4045547 1190.0021
## parity             0.802309017 2.400369e-01  3.3424398 1151.5579
## number           -17.013281891 4.168066e+00 -4.0818166  426.2992
## mwt                1.213877230 4.329522e-01  2.8037211  506.1643
## time              -4.145132550 5.996015e+00 -0.6913146  308.2650
## meth:mwt          -0.015679485 6.479987e-03 -2.4196786 1193.8084
## gestation:number   0.053209113 1.497169e-02  3.5539825  392.1629
## gestation:mwt     -0.003892398 1.534250e-03 -2.5370035  476.8059
## gestation:mht      0.026612463 1.262313e-02  2.1082302 1163.6409
## gestation:time     0.010812865 2.132096e-02  0.5071472  298.5339
##                       p.value
## (Intercept)      1.768824e-01
## gestation        2.608988e-01
## smoke            1.922007e-02
## mht              6.478430e-02
## meth             1.604139e-01
## parity           8.561844e-04
## number           4.765685e-05
## mwt              5.133479e-03
## time             4.895023e-01
## meth:mwt         1.568284e-02
## gestation:number 3.942639e-04
## gestation:mwt    1.130704e-02
## gestation:mht    3.521904e-02
## gestation:time   6.121452e-01
```

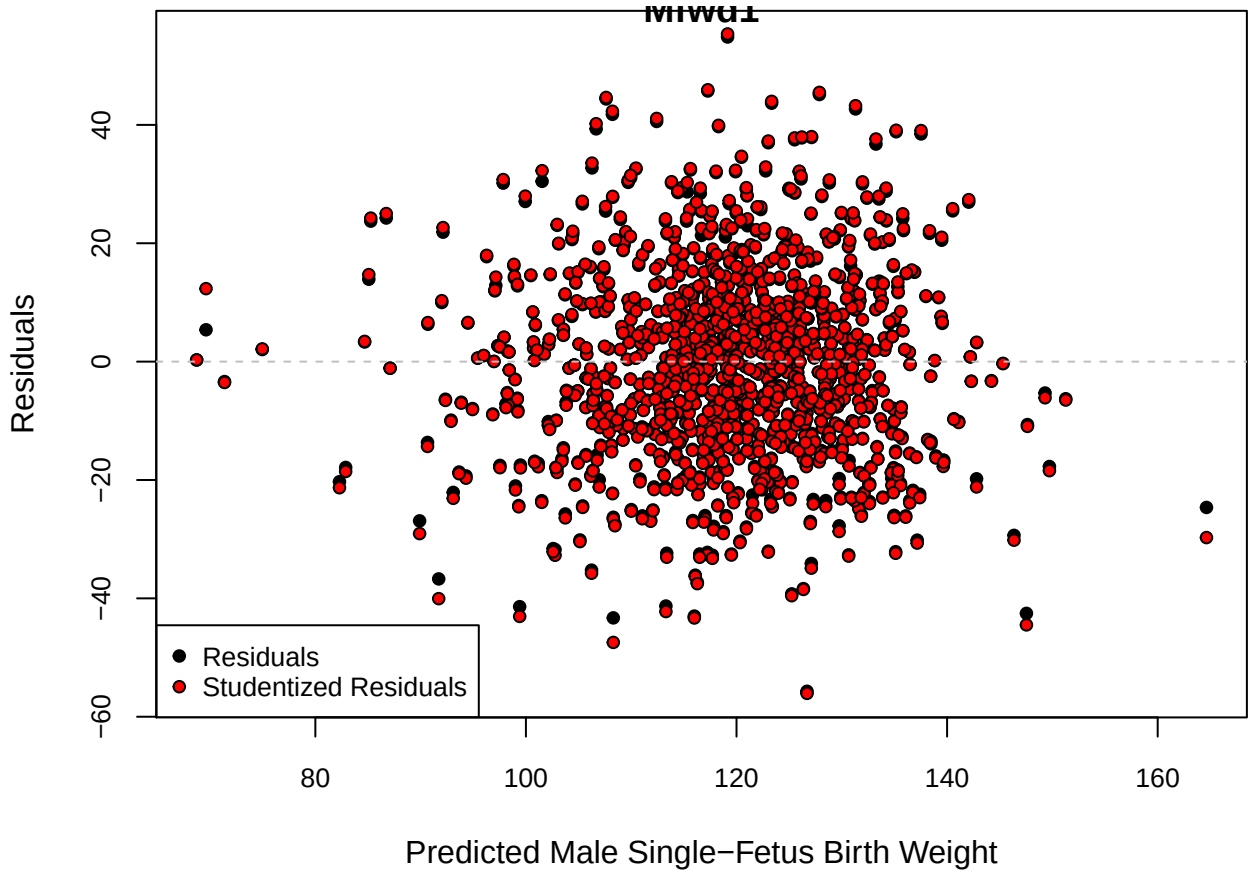
```r
# pooling Mstep2
Mstep_best <- with(chds_imp,
                   lm(wt ~ gestation + parity + meth + med + mht + mwt + income + smoke + time +
                          number + gestation:income + gestation:number + gestation:med + med:mwt
                          + meth:income
                     )
                   )
summary(pool(Mstep_best))
```

```
##                        estimate   std.error  statistic         df
## (Intercept)       13.535418277 24.21924750  0.5588703  480.95321
## gestation          0.197997205  0.07429570  2.6649888  219.47155
## parity             0.788104833  0.24509169  3.2155510 1007.00066
## meth              -1.130197706  0.28487153 -3.9673943  483.15072
## med               -7.360880275  5.76729828 -1.2763134  826.21145
## mht                0.922168277  0.20991939  4.3929637  647.51353
## mwt               -0.029351869  0.04786674 -0.6131997  347.46861
## income           -11.329145913  4.51150197 -2.5111694   39.34306
## smoke              2.865076492  1.25989261  2.2740640 1048.70223
## time              -0.953065475  0.72110793 -1.3216683 1197.81696
## number           -16.344808640  3.83793758 -4.2587479  879.32330
## gestation:income   0.039498709  0.01622960  2.4337452   34.43462
## gestation:number   0.050884371  0.01377624  3.6936330  822.91544
## gestation:med      0.009491779  0.01941864  0.4887972 1108.84936
## med:mwt            0.036627650  0.01531363  2.3918338  507.17364
## meth:income        0.098196350  0.06592166  1.4895916  367.27768
##                       p.value
## (Intercept)      5.763547e-01
## gestation        7.802565e-03
## parity           1.336732e-03
## meth             7.697497e-05
## med              2.020921e-01
## mht              1.217051e-05
## mwt              5.398607e-01
## income           1.216400e-02
## smoke            2.313842e-02
## time             1.865310e-01
## number           2.216618e-05
## gestation:income 1.508893e-02
## gestation:number 2.310376e-04
## gestation:med    6.250748e-01
## med:mwt          1.691791e-02
## meth:income      1.365949e-01
```

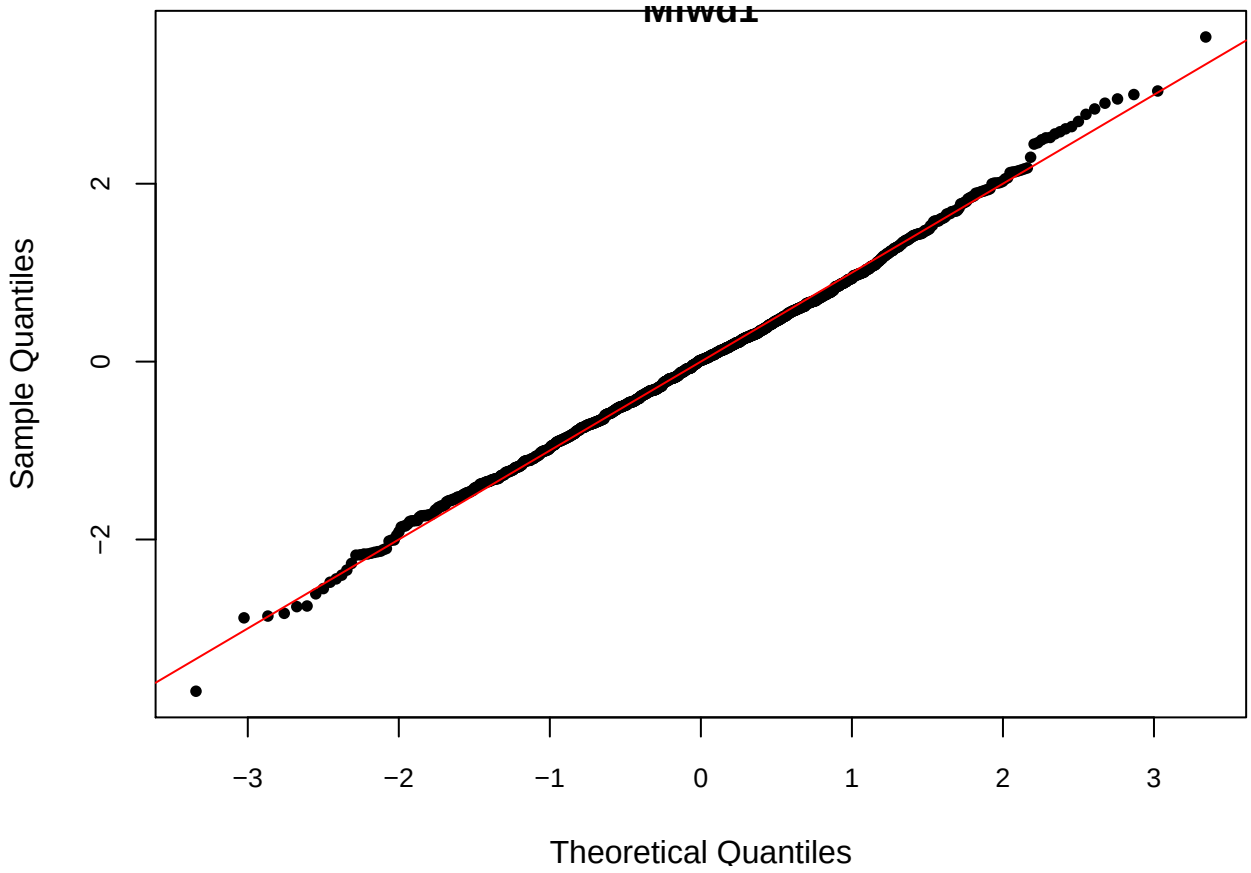
3 Model Diagnostics

3.1 QQ & Residual Plots

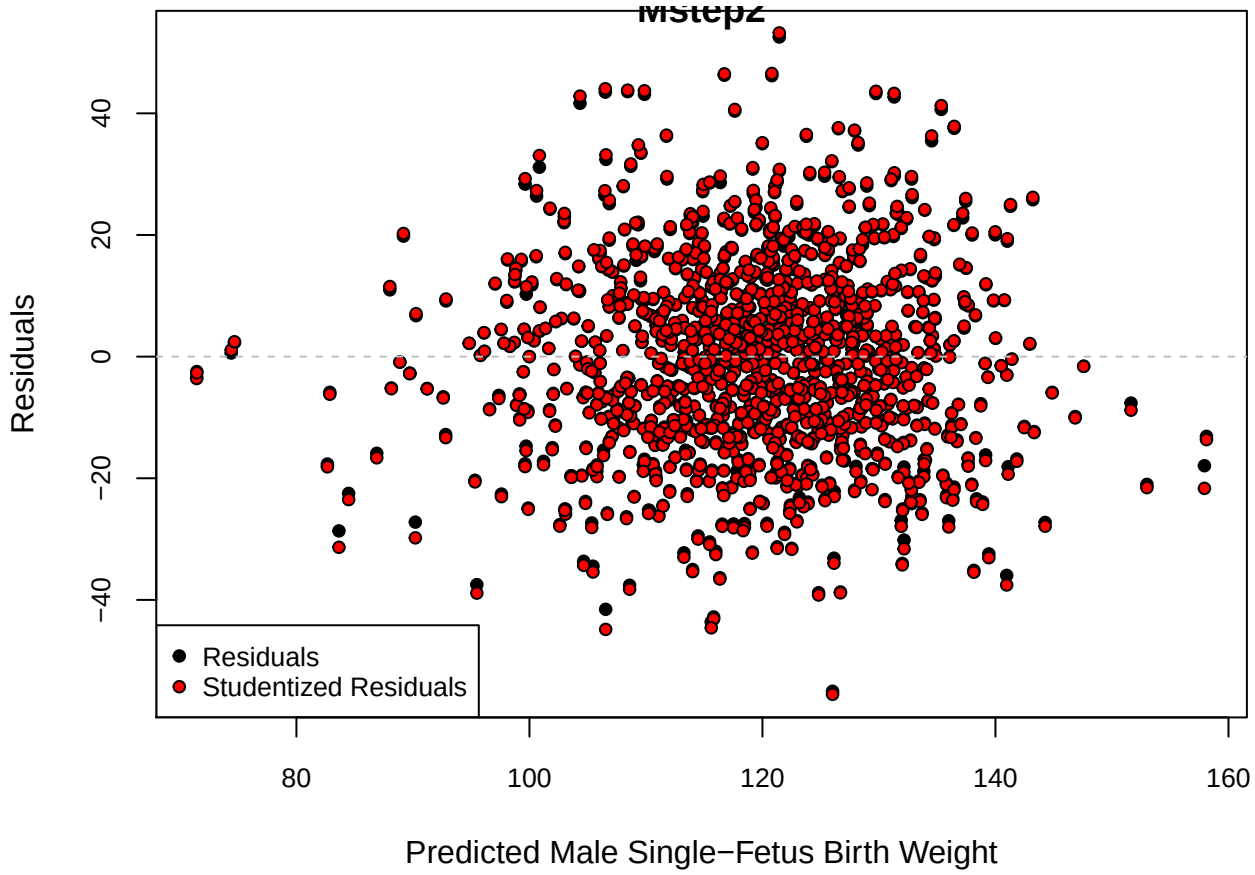
```
qq_plot <- function(M, main)
{
  sigma_hat <- sqrt(sum(resid(M)^2) / M$df.residual)
  sigma_hat <- round(sigma_hat, 2) # round to 2 decimal places

  cex <- 0.8
  qqnorm(resid(M) / sigma_hat, pch = 16, cex = cex, cex.axis = cex, main=main)
  abline(a = 0, b = 1, col = "red") # add 45 degree line
}

residual_vs_predicted_plot <- function(M, main)
{
  res <- resid(M)
  sigma_hat <- sqrt(sum(res^2) / M$df.residual)
  sigma_hat <- round(sigma_hat, 2) # round to 2 decimal places
  h <- hatvalues(M)
  pred <- predict(M)
  res_stu <- resid(M) / sqrt(1-h) # studentized residuals, but on the data scale

  cex <- .8 # controls the size of the points and labels
  par(mar = c(4,4,.1,.1))
  plot(pred, resid(M), pch = 21, bg = "black", cex = cex,
       cex.axis = cex, xlab = "Predicted Male Single-Fetus Birth Weight",
       ylab = "Residuals", main=main)
  points(pred, res_stu, pch = 21, bg = "red", cex = cex)
  legend(x = "bottomleft", c("Residuals", "Studentized Residuals"),
         pch = 21, pt.bg = c("black", "red"), pt.cex = cex, cex = cex)
  abline(h=0, lty=2, col="grey")
}

residual_vs_predicted_plot(Mfwd1, "Mfwd1")
```



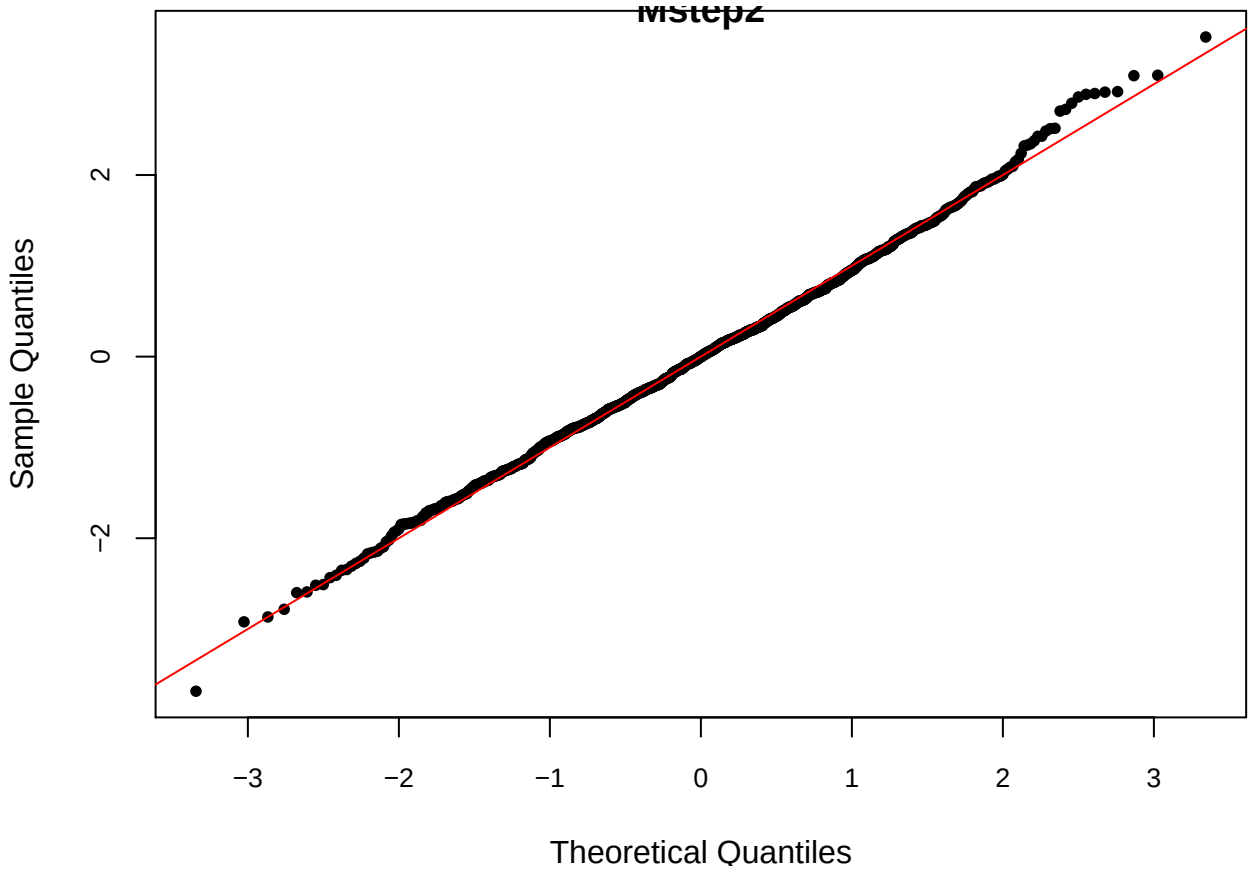

```
qq_plot(Mfwd1, "Mfwd1")
```




```
residual_vs_predicted_plot(Mstep2, "Mstep2")
```



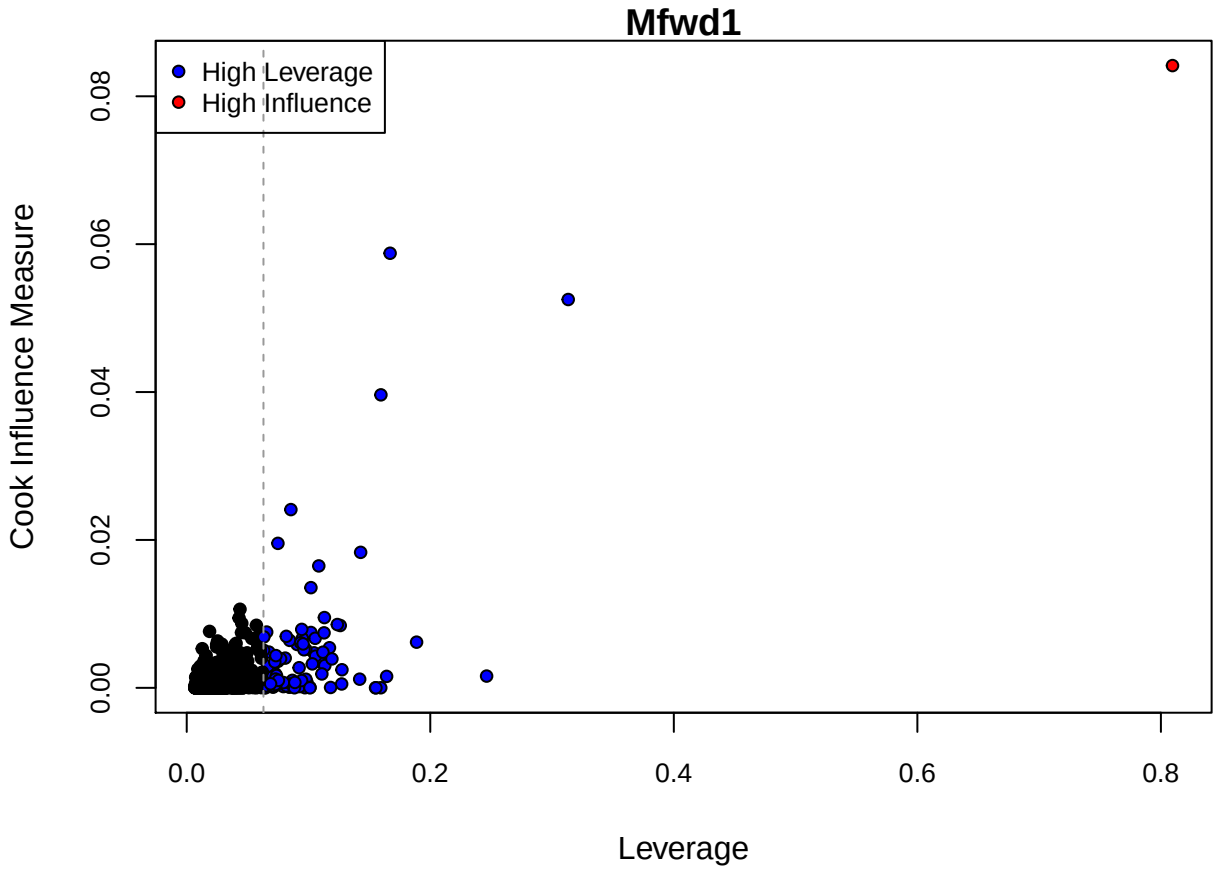

```
qq_plot(Mstep2, "Mstep2")
```



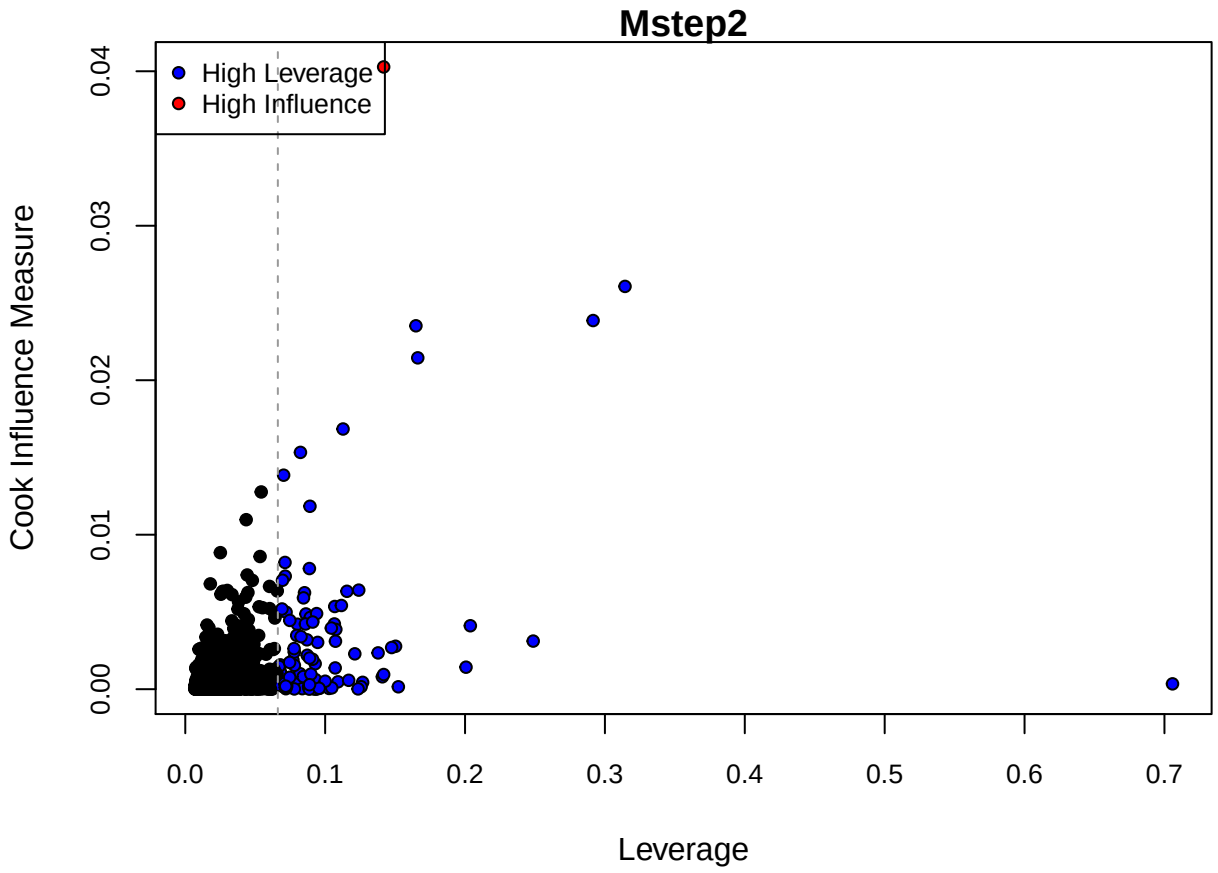
3.2 Leverage And Influence Analysis

```
cooks_distance_vs_leverage_plot <- function(M, data, main)
{
  p <- length(coef(M))
  n <- nrow(data)
  h <- hatvalues(M)
  res <- resid(M)
  hbar <- p / n

  # cook's distance vs. leverage
  D <- cooks.distance(M)

  # flag some of the points
  infl_ind <- which.max(D) # top influence point
  lev_ind <- h > 2*hbar # leverage more than 2x the average
  clrs <- rep("black", len = n)

  clrs[lev_ind] <- "blue"
  clrs[infl_ind] <- "red"
  par(mfrow = c(1,1), mar = c(4,4,1,1))

  cex <- .8
  plot(h, D, xlab = "Leverage", ylab = "Cook Influence Measure",
       pch = 21, bg = clrs, cex = cex, cex.axis = cex, main=main)
```

```
  abline(v = 2*hbar, col = "grey60", lty = 2) # 2x average leverage
  legend("topleft", legend = c("High Leverage", "High Influence"), pch = 21,
         pt.bg = c("blue", "red"), cex = cex, pt.cex = cex)

}

cooks_distance_vs_leverage_plot(Mfwd1, chds1, "Mfwd1")
```




```
cooks_distance_vs_leverage_plot(Mstep2, chds2, "Mstep2")
```

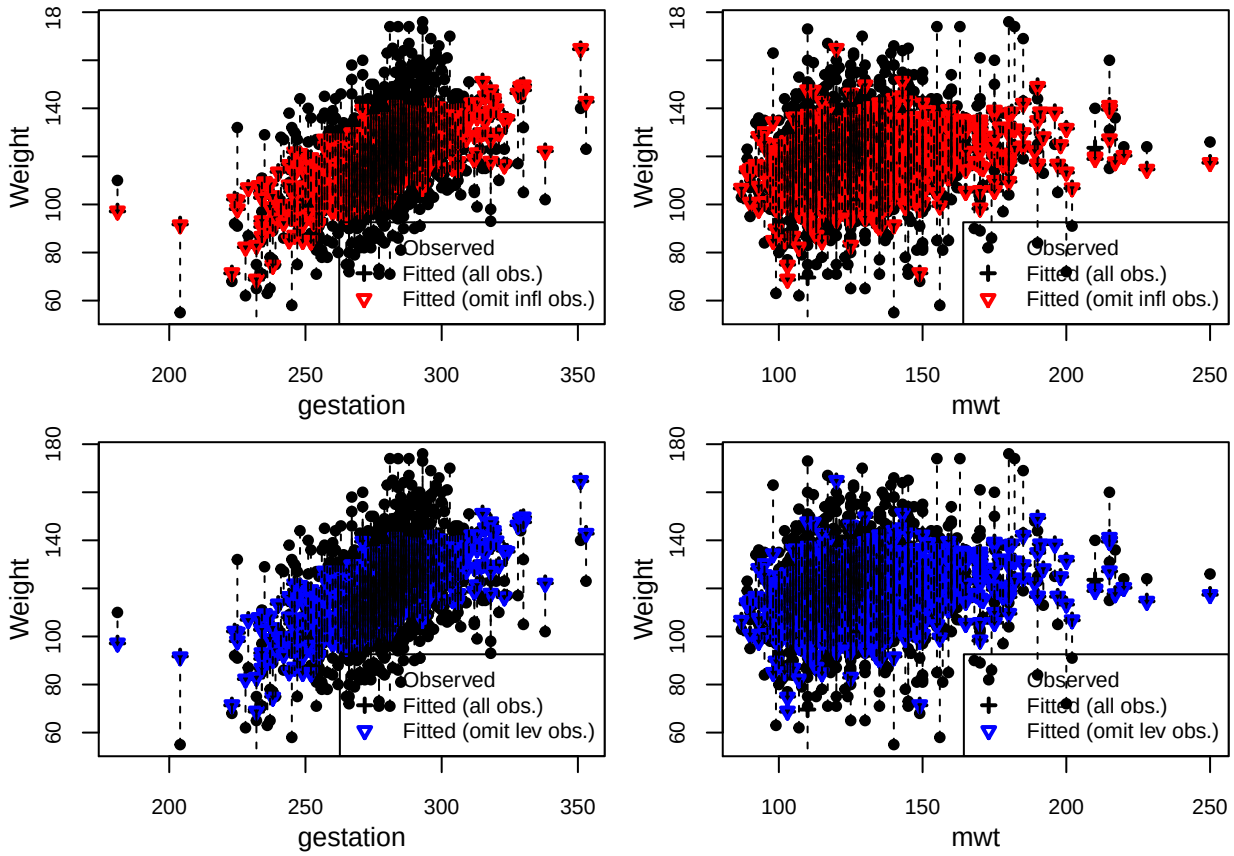


3.3 Cook's High Influence vs High Leverage Plots

High Influence vs High Leverage Mfwd1 Plot

```
#cooks_highinf_vs_highlev Mfwd1
p <- length(coef(Mfwd1))
n <- nrow(chds1)
h <- hatvalues(Mfwd1)
res <- resid(Mfwd1)
hbar <- p / n

# cook's distance vs. leverage
D <- cooks.distance(Mfwd1)

# flag some of the points
infl_ind <- which.max(D) # top influence point
lev_ind <- h > 2*hbar # leverage more than 2x the average
clrs <- rep("black", len = n)

omit_ind <- c(infl_ind, # most influential
              which.max(h)) # highest leverage
names(omit_ind) <- c("infl", "lev")

yobs <- chds1[,"wt"] # observed values
```

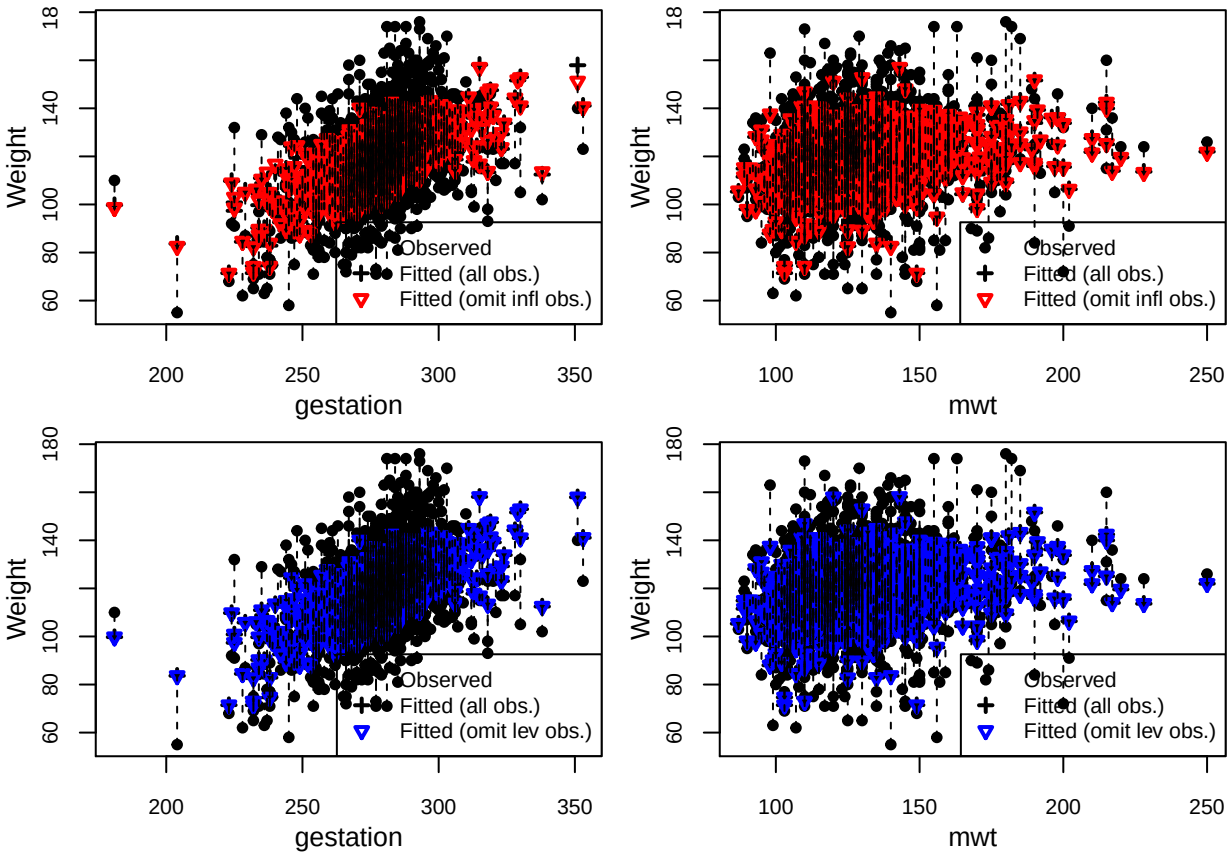
```r
# all observations
Mf <- lm(formula(Mfwd1), data=chds1)
yfitf <- predict(Mf) # fitted values
par(mfrow = c(2,2))
par(mar = c(3.5,3.5,0,0)+.1)
cex <- .8
clrs2 <- c("red", "blue")

for(jj in 1:length(omit_ind)) {
  Mo <- lm(formula(Mfwd1), data=chds1, subset = -omit_ind[jj])
  yfito <- predict(Mo, newdata = chds1) # fitted values at all observations
  for (ii in c("gestation", "mwt")) {
    # response vs. each covariate
    xobs <- chds1[,ii] # covariate
    ylim <- range(yobs, yfitf, yfito) # y-range of plot
    # observed
    plot(xobs, yobs, pch = 21, bg = clrs, cex = cex, cex.axis = cex, xlab = "", ylab = "")

    title(xlab = ii, ylab = "Weight", line = 2)
    # fitted, all observations
    points(xobs, yfitf, pch = 3, lwd = 2, cex = cex)
    # fitted, one omitted observation
    points(xobs, yfito, col = clrs2[jj], pch = 6, lwd = 2, cex = cex)
    # connect with lines
    segments(x0 = xobs, y0 = pmin(yobs, yfitf, yfito),
             y1 = pmax(yobs, yfitf, yfito), lty = 2) # connect them

    legend("bottomright", legend = c("Observed", "Fitted (all obs.)",
                                     paste0("Fitted (omit ", names(omit_ind)[jj], " obs.)")),
           pch = c(21,3,6), lty = c(NA, NA, NA),
           lwd = c(NA, 2, 2), pt.bg = "black",
           col = c("black", "black", clrs2[jj]),
           cex = cex, pt.cex = cex)
  }
}
```



High Influence vs High Leverage Mstep2 Plot

```
p <- length(coef(Mstep2))
n <- nrow(chds2)
h <- hatvalues(Mstep2)
res <- resid(Mstep2)
hbar <- p / n

# cook's distance vs. leverage
D <- cooks.distance(Mstep2)

# flag some of the points
infl_ind <- which.max(D) # top influence point
lev_ind <- h > 2*hbar # leverage more than 2x the average
clrs <- rep("black", len = n)

omit_ind <- c(infl_ind, # most influential
              which.max(h)) # highest leverage
names(omit_ind) <- c("infl", "lev")

yobs <- chds2[,"wt"] # observed values
Mf <- lm(formula(Mstep2), data=chds2)
yfitf <- predict(Mf) # fitted values
par(mfrow = c(2,2))
par(mar = c(3.5,3.5,0,0)+.1)
```

```
cex <- .8
clrs2 <- c("red", "blue")

for(jj in 1:length(omit_ind)) {
  # model with omitted observation
  Mo <- lm(formula(Mstep2), data=chds2, subset = -omit_ind[jj])
  yfito <- predict(Mo, newdata = chds2) # fitted values at all observations
  for (ii in c("gestation", "mwt")) {
    # response vs. each covariate
    xobs <- chds2[,ii] # covariate
    ylim <- range(yobs, yfitf, yfito) # y-range of plot
    # observed
    plot(xobs, yobs, pch = 21, bg = clrs, cex = cex, cex.axis = cex, xlab = "", ylab = "")

    title(xlab = ii, ylab = "Weight", line = 2)
    # fitted, all observations
    points(xobs, yfitf, pch = 3, lwd = 2, cex = cex)
    # fitted, one omitted observation
    points(xobs, yfito, col = clrs2[jj], pch = 6, lwd = 2, cex = cex)
    # connect with lines
    segments(x0 = xobs, y0 = pmin(yobs, yfitf, yfito),
             y1 = pmax(yobs, yfitf, yfito), lty = 2) # connect them

    legend("bottomright", legend = c("Observed", "Fitted (all obs.)",
                                     paste0("Fitted (omit ", names(omit_ind)[jj], " obs.)")),
           pch = c(21,3,6), lty = c(NA, NA, NA),
           lwd = c(NA, 2, 2), pt.bg = "black",
           col = c("black", "black", clrs2[jj]),
           cex = cex, pt.cex = cex)
  }
}
```


Weight
gestation
mwt
Observed
Fitted (all obs.)
Fitted (omit infl obs.)
Fitted (omit lev obs.)