# Homework5

*Echo Liu*

*November 12, 2018*

## Question 1: Missing Data Mechanics

#### a) Create a dataset with 30% of the age values missing completely at random

```
# Create a dataset with 30% pf the age values missing completely at random
prop.m = 0.3  # 30% missingness
set.seed(2018)
mcar   = runif(nrow(treeage), min=0, max=1)
treeage$age.mcar = ifelse(mcar<prop.m, NA, treeage$age)
treeage$age <- NULL
treeage
```

```
##    number diameter age.mcar
## 1       1     12.0      125
## 2       2     11.4      119
## 3       3      7.9       NA
## 4       4      9.0       NA
## 5       5     10.5       99
## 6       6      7.9      117
## 7       7      7.3       69
## 8       8     10.2       NA
## 9       9     11.7      154
## 10     10     11.3      168
## 11     11      5.7       61
## 12     12      8.0       80
## 13     13     10.3      114
## 14     14     12.0      147
## 15     15      9.2      122
## 16     16      8.5      106
## 17     17      7.0       NA
## 18     18     10.7       88
## 19     19      9.3       97
## 20     20      8.2       99
```

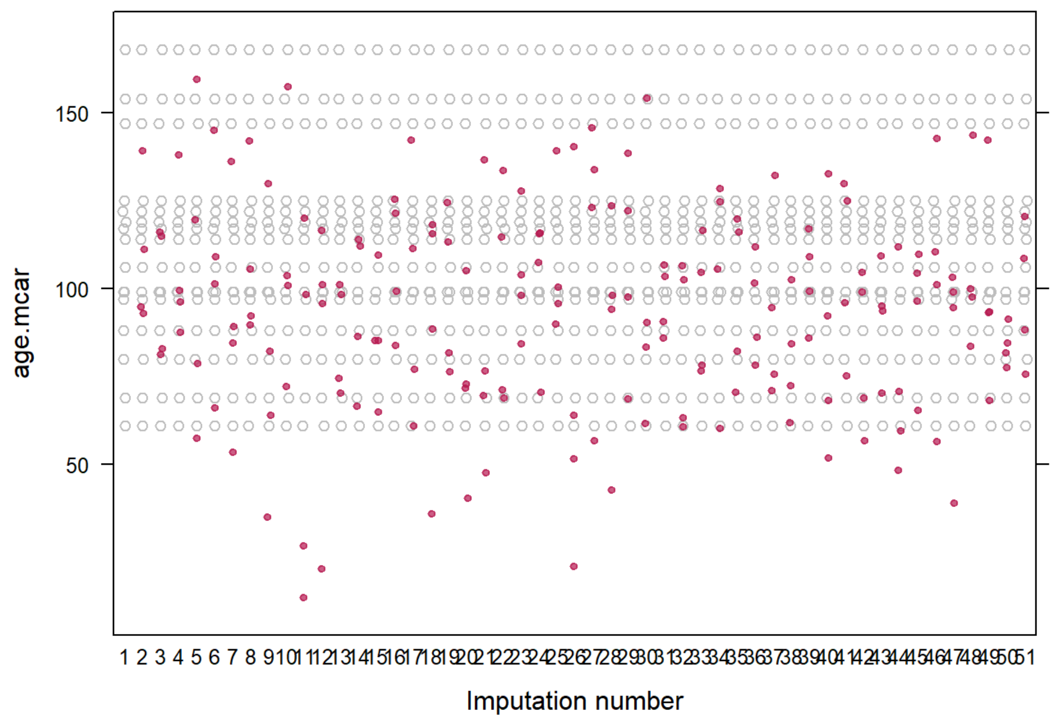
#### b) Use a multiple imputation approach to fill in missing ages

```
#Multiple Imputation Method
treeage.mi50 <- mice(treeage, m = 50, defaultMethod = c("norm", "logreg", "polyreg", "polr"),set.seed(2018), print = FA
LSE)

#Look at the first couple of completed dataset
d1 <- complete(treeage.mi50, 1)
d1
```

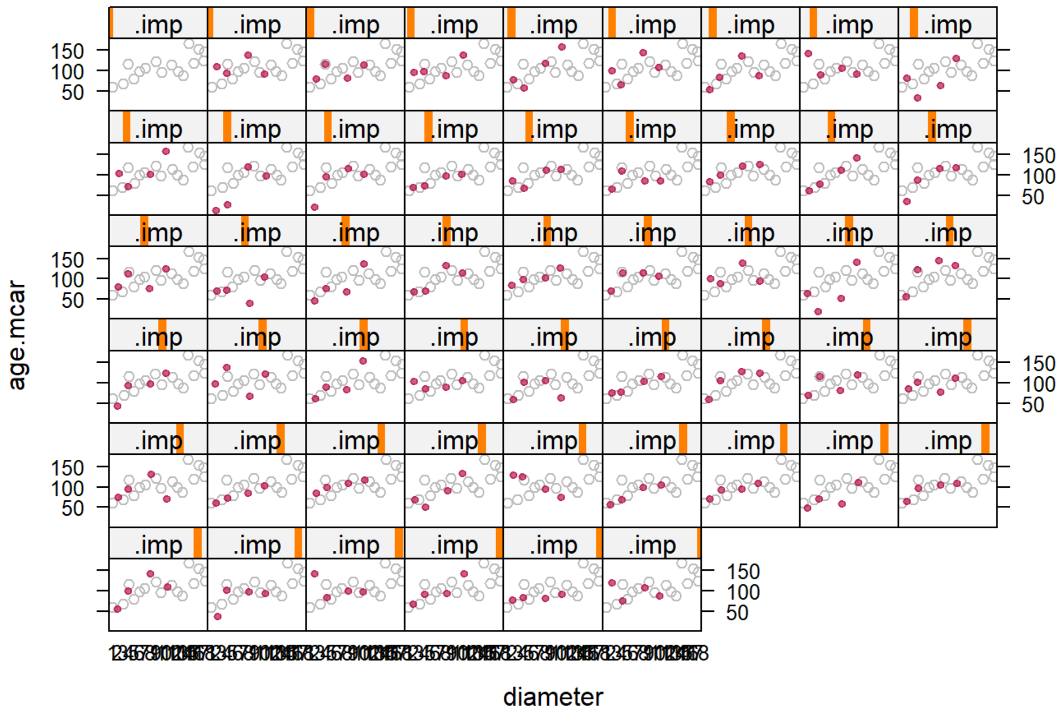
```
##    number diameter  age.mcar
## 1       1     12.0 125.00000
## 2       2     11.4 119.00000
## 3       3      7.9  94.94214
## 4       4      9.0 139.29265
## 5       5     10.5  99.00000
## 6       6      7.9 117.00000
## 7       7      7.3  69.00000
## 8       8     10.2  92.97899
## 9       9     11.7 154.00000
## 10     10     11.3 168.00000
## 11     11      5.7  61.00000
## 12     12      8.0  80.00000
## 13     13     10.3 114.00000
## 14     14     12.0 147.00000
## 15     15      9.2 122.00000
## 16     16      8.5 106.00000
## 17     17      7.0 111.21932
## 18     18     10.7  88.00000
## 19     19      9.3  97.00000
## 20     20      8.2  99.00000
```

**Multiple Imputation Diagnostics**

```
#Plot marginal distribution of age
stripplot(treeage.mi50, age.mcar~.imp, col = c("grey", mdc(2)), pch = c(1, 20))
```



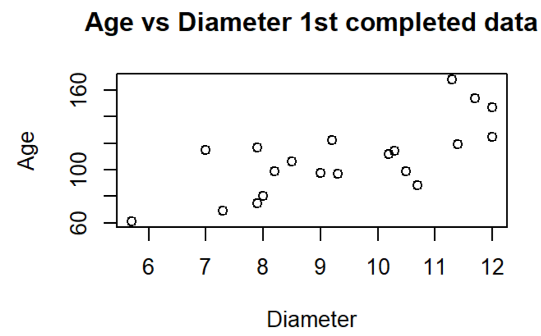

```
#no obvious problems with the imputations from this plot

#Plot scatter plot of age versus diameter
stripplot(treeage.mi50, age.mcar~diameter|.imp, col = c("grey", mdc(2)), pch = c(1, 20))
```



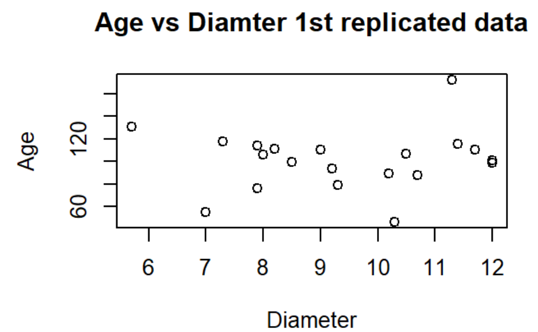

**Posterior Predictive Checks (run the diagnostics on at least two of the completed datasets)**

```
#let's append the data and make replicates
treeage.ppcheck <- rbind(treeage, treeage)

#now blank every value in age variable with missing value
treeage.ppcheck[21:40, 3] = NA

#run the MI software on the completed data
treeage.ppcheck.mi = mice(treeage.ppcheck, m = 50, defaultMethod = c("norm", "logreg", "polyreg", "polr"), print = FALS
E)

#get the completed datasets
d1ppcheck <- complete(treeage.ppcheck.mi, 1)
d2ppcheck <- complete(treeage.ppcheck.mi, 2)
```

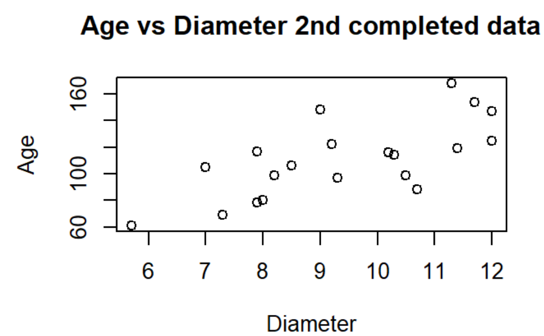
**Graphs for the first dataset**

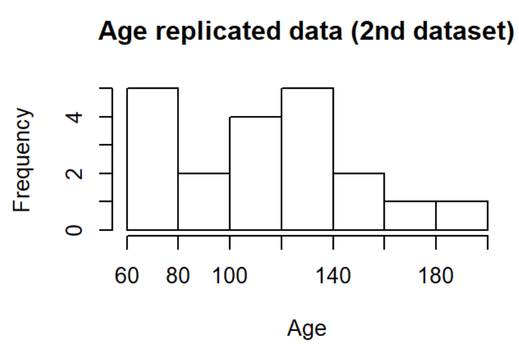



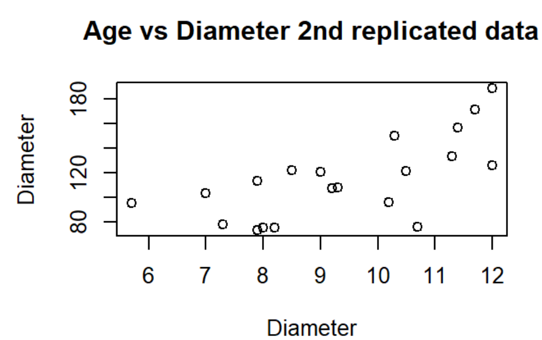





**Graphs for the second dataset**

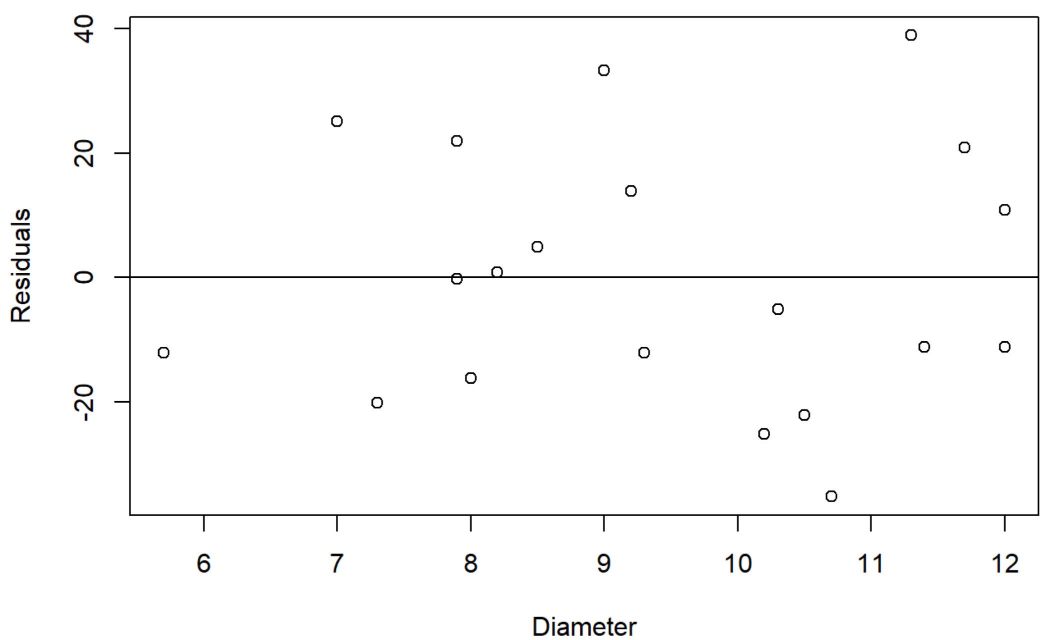

Histograms and boxplots look fine. Marginal distribution of age in replicated data may not completely match the that in completed data, but this is due to this relatively small sample size. We may need to collect more data to have a better imputation quality. But overall, I'm pretty with satisfied with this model. Therefore, I'll skip c).

**d) Estimate a regression of age on diameter directly**

```
#Use one of the completed dataset
agereg1 = lm (age.mcar~diameter, data = d1)
summary(agereg1)
```

```
##
## Call:
## lm(formula = age.mcar ~ diameter, data = d1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.140 -13.115  -2.646  15.622  38.851
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   15.986     25.440   0.628  0.53764
## diameter      10.014      2.658   3.768  0.00141 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.19 on 18 degrees of freedom
## Multiple R-squared:  0.441,  Adjusted R-squared:  0.4099
## F-statistic:  14.2 on 1 and 18 DF,  p-value: 0.001408
```

```
#To check residuals
par(mfrow = c(1,1))
plot(agereg1$residuals, x = d1$diameter, xlab = "Diameter", ylab = "Residuals")
abline(0,0)
```




```
#The residual plot satisfies the constant variance assumption of linear regression. So no transformation is needed.
```

```
#Multiple Imputation inferences on all m=50 data sets
ageregMI50 <- with(data = treeage.mi50, lm (age.mcar ~ diameter))
agereg <- pool(ageregMI50)
summary(agereg, conf.int = T)
```

```
##                estimate std.error  statistic       df      p.value     2.5 %
## (Intercept) -6.227396 28.601759 -0.2177277 11.85952 0.831134775 -68.62725
## diameter    12.051563  2.907879  4.1444509 12.57347 0.001234808   5.74774
##                97.5 %
## (Intercept) 56.17246
## diameter    18.35539
```

The linear regression model shows that if diameter of a tree is 0, then its age is -6.227 year, which makes no sense because tree always has tree trunk with some diameter.

Slope of the diameter means that if diameter of a tree increases by 1 meter, then tree's age is expected to be 12.0515 older.

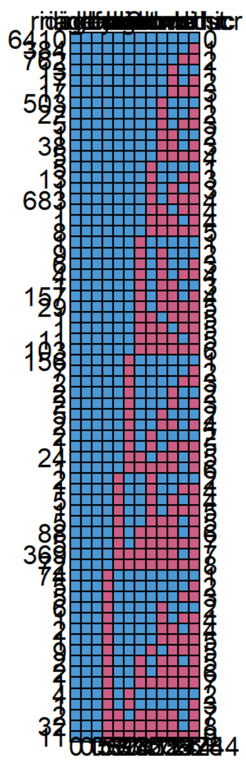
We're 95% confident that if diameter of a tree increases by 1 meter, its age will be $(5.74774, 18.35539)$ older.

# Question 2: Multiple imputation in NHANES data

## a) Use a multiple imputation approach to fill in missing values

**Step 1: Data cleaning**

```
#data cleaning
nhanes$wtmec2yr <- NULL
nhanes$sdmvstra <- NULL
nhanes$sdmvpsu <- NULL
md.pattern(nhanes)
```



```
#make factor variables
nhanes$riagendr <- as.factor(nhanes$riagendr)
nhanes$ridreth2 <- as.factor(nhanes$ridreth2)
nhanes$dmdeduc <- as.factor(nhanes$dmdeduc)
nhanes$indfminc <- as.factor(nhanes$indfminc)

#look at the correlations between each variables
cor(nhanes[,c(1:2,7:12)], use = "complete.obs")
```

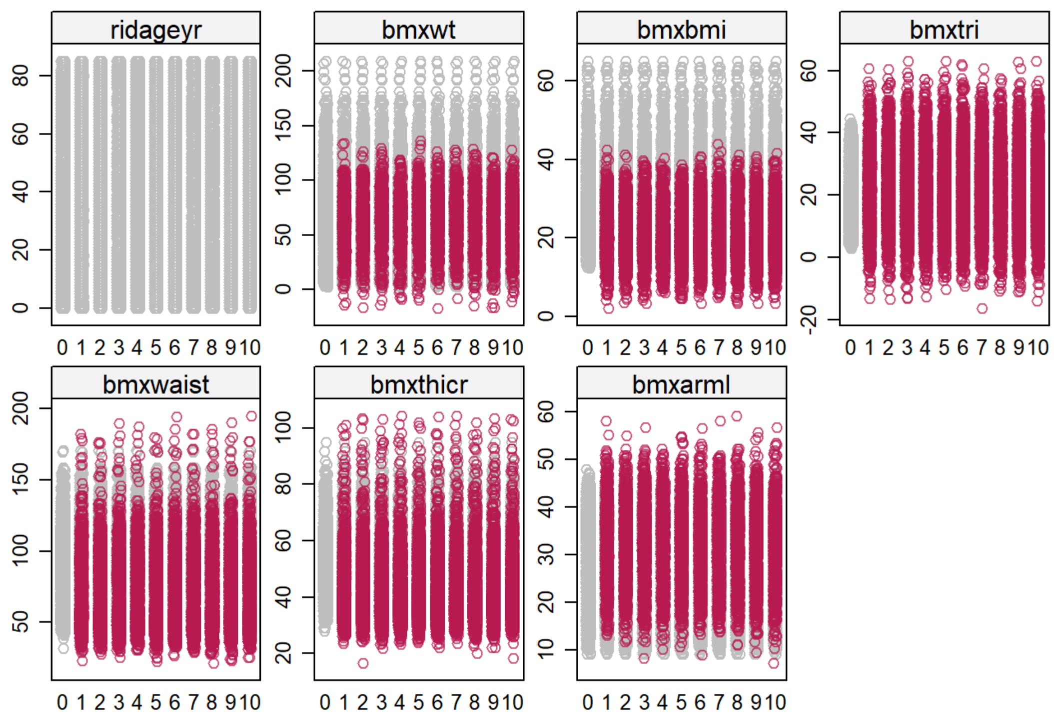
```
##                age  ridageyr     bmxwt    bmxbmi     bmxtri  bmxwaist
## age      1.0000000 0.9999152 0.3851653 0.3864768 0.16136102 0.5390222
## ridageyr 0.9999152 1.0000000 0.3851088 0.3865109 0.16133723 0.5389748
## bmxwt    0.3851653 0.3851088 1.0000000 0.8791986 0.40952411 0.8981476
## bmxbmi   0.3864768 0.3865109 0.8791986 1.0000000 0.63796690 0.9161481
## bmxtri   0.1613610 0.1613372 0.4095241 0.6379669 1.00000000 0.5272132
## bmxwaist 0.5390222 0.5389748 0.8981476 0.9161481 0.52721319 1.0000000
## bmxthicr 0.1050753 0.1049976 0.8530036 0.8368241 0.53379885 0.7172995
## bmxarml  0.3483631 0.3482102 0.7668720 0.4662624 0.04997022 0.5801877
##           bmxthicr    bmxarml
## age      0.1050753 0.34836313
## ridageyr 0.1049976 0.34821024
## bmxwt    0.8530036 0.76687198
## bmxbmi   0.8368241 0.46626241
## bmxtri   0.5337989 0.04997022
## bmxwaist 0.7172995 0.58018775
## bmxthicr 1.0000000 0.57235866
## bmxarml  0.5723587 1.00000000
```

```
#we find that the correlaion between variable age and ridageyr is 0.999917, which suggests extremely high collinearity.
 Therefore, we drop age variable since it has many missing valus but ridageyr doesn't.
nhanes$age <- NULL
```

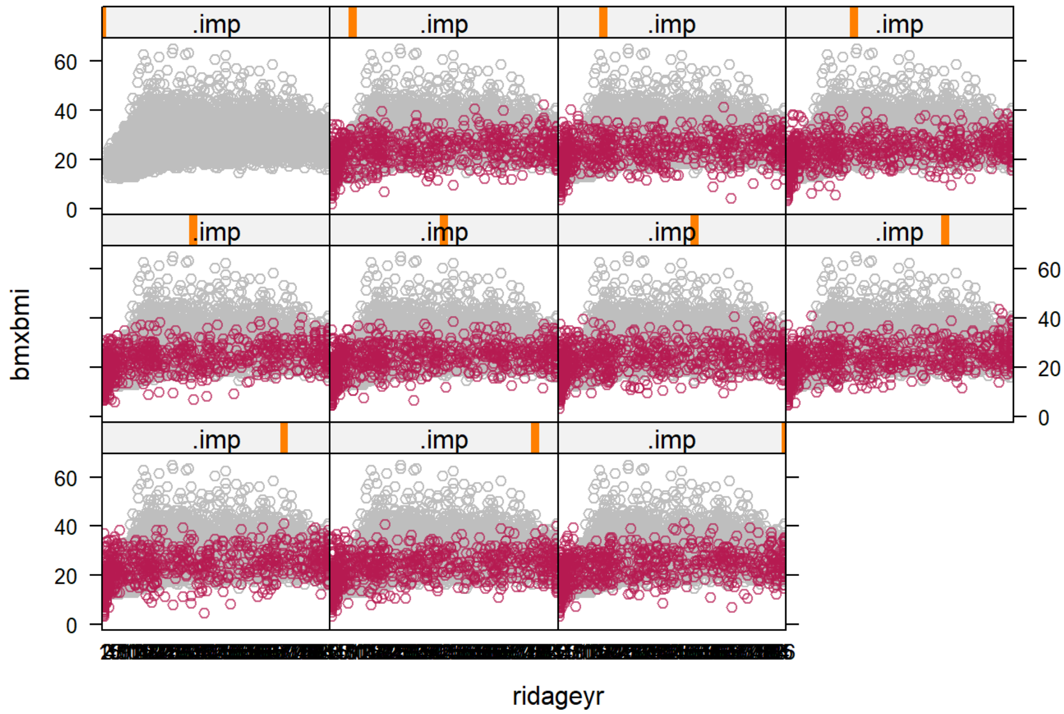
```
#let's create 10 multiple imputations for the missing values
nhanes.mi10 <- mice(nhanes, m = 10, defaultMethod = c("norm", "logreg", "polyreg", "polr"), set.seed(2018), print = FAL
SE)

#We check quality of imputation on two completed datasets and both look reasonable.
ds1 <- complete(nhanes.mi10,1)
ds2<- complete(nhanes.mi10,2)
```

### Multiple Imputation Diagnostics

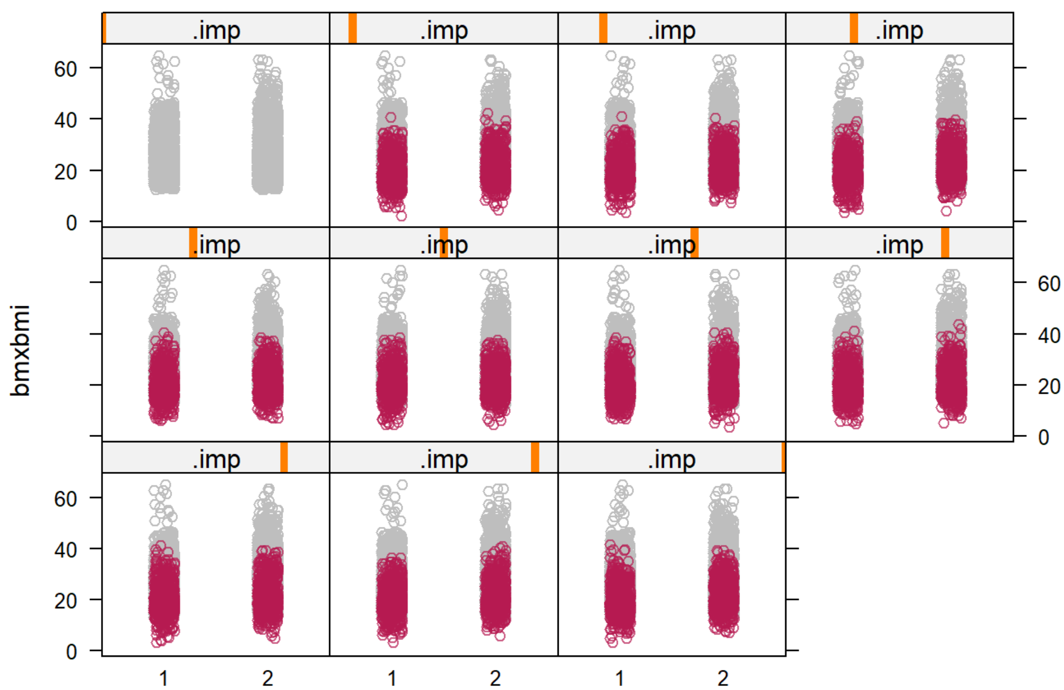
```
#continuous variable
stripplot(nhanes.mi10, col = c("grey", mdc(2), pch = c(1, 20)))
```



```
#categorical variable (bmxbmi by age and bmxbmi by gender)
stripplot(nhanes.mi10, bmxbmi~ridageyr|.imp, col = c("grey", mdc(2), pch = c(1, 20)))
```



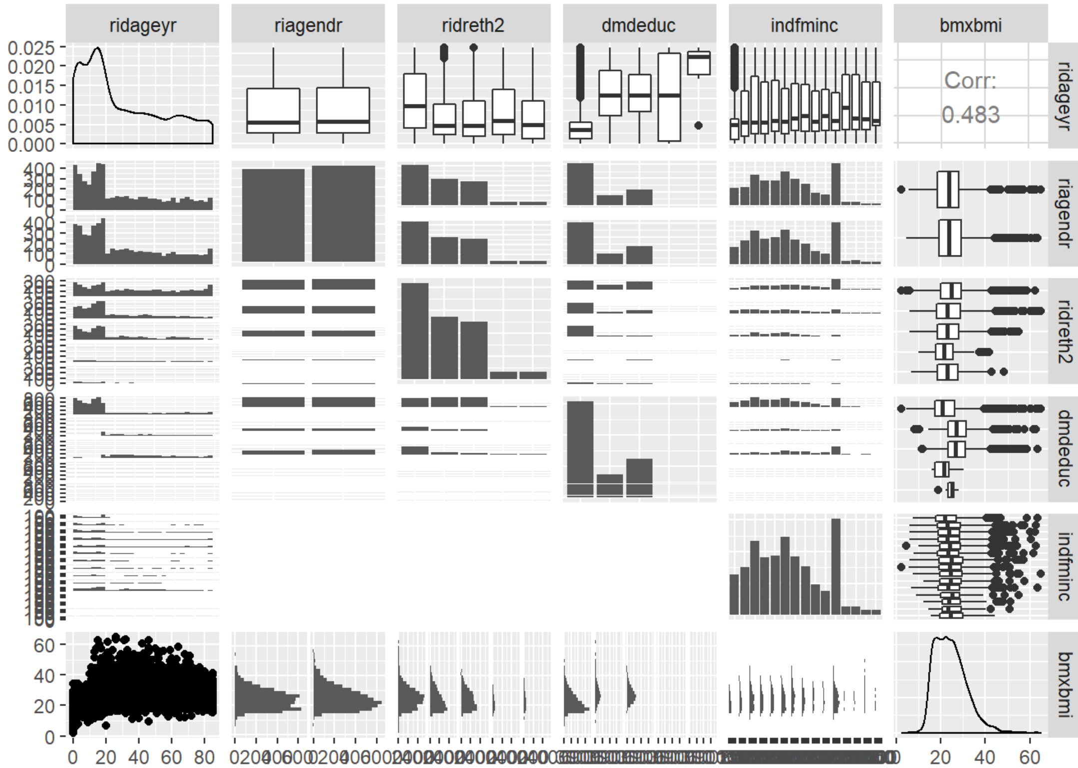

```
stripplot(nhanes.mi10, bmxbmi~riagendr|.imp, col = c("grey", mdc(2), pch = c(1, 20)))
```




No evidence that imputation models are poorly specified for what we want to do.

b) Run a model that predicts BMI from some subsets of age, gender, race, education and income.

**Exploratory data analysis**




It seems that the size of boxes for highest level of education is very uneqal, let's check the reason.

```
##
##    1    2    3    7    9
## 6158 1494 2451    8   11
```

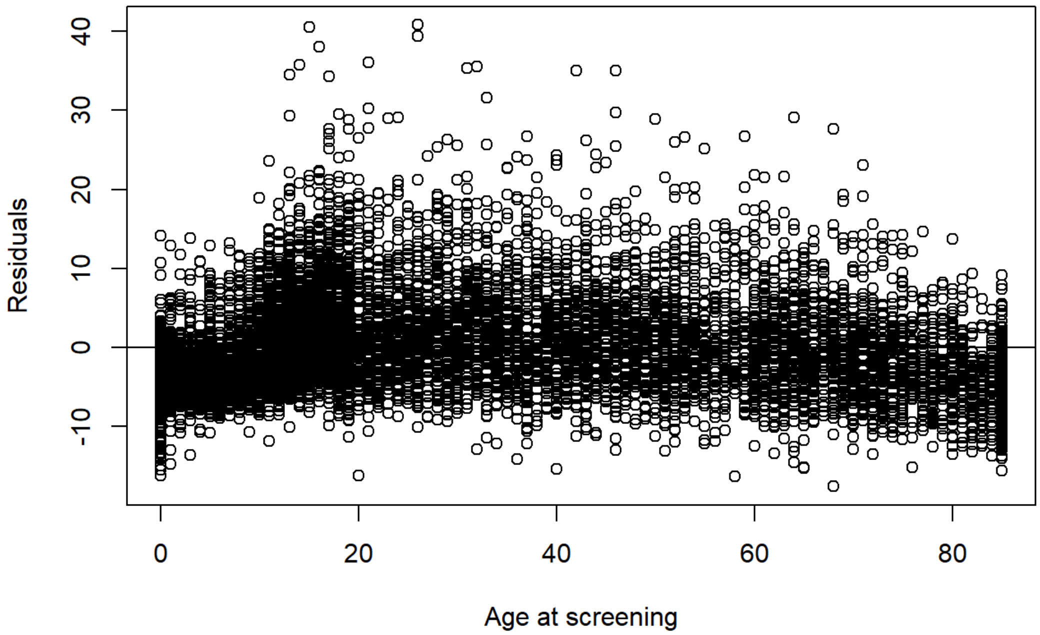
Category 7 and 9 have really small sample sizes compared to category 1-3, which makes sense because they correspond to those who refuse to answer and those who don't know their education level, which are rare. Otherwise, no real patterns show up in the scatter plots and constant variance assumptions seem to be satisfied.

**Modelling: Plain Vanilla**

```
##
## Call:
## lm(formula = bmxbmi ~ ridageyr + as.factor(riagendr) + as.factor(ridreth2) +
##     as.factor(dmdeduc) + as.factor(indfminc), data = ds1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.576  -3.993  -1.144   2.791  40.814
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           18.316277   0.277578  65.986  < 2e-16 ***
## ridageyr               0.116552   0.002808  41.512  < 2e-16 ***
## as.factor(riagendr)2   0.699071   0.120969   5.779 7.74e-09 ***
## as.factor(ridreth2)2   1.990101   0.157673  12.622  < 2e-16 ***
## as.factor(ridreth2)3   1.693607   0.164488  10.296  < 2e-16 ***
## as.factor(ridreth2)4  -1.264646   0.346985  -3.645 0.000269 ***
## as.factor(ridreth2)5   0.874163   0.345867   2.527 0.011504 *
## as.factor(dmdeduc)2    3.207676   0.191188  16.778  < 2e-16 ***
## as.factor(dmdeduc)3    3.024092   0.169839  17.806  < 2e-16 ***
## as.factor(dmdeduc)7   -3.027840   2.153454  -1.406 0.159743
## as.factor(dmdeduc)9   -3.144081   1.839180  -1.710 0.087389 .
## as.factor(indfminc)2   0.314844   0.328213   0.959 0.337448
## as.factor(indfminc)3  -0.102011   0.301194  -0.339 0.734852
## as.factor(indfminc)4  -0.341522   0.315767  -1.082 0.279472
## as.factor(indfminc)5   0.142237   0.313793   0.453 0.650354
## as.factor(indfminc)6  -0.432007   0.297462  -1.452 0.146446
## as.factor(indfminc)7   0.396576   0.314387   1.261 0.207185
## as.factor(indfminc)8  -0.045135   0.327423  -0.138 0.890363
## as.factor(indfminc)9   0.818858   0.369708   2.215 0.026791 *
## as.factor(indfminc)10  0.326981   0.395271   0.827 0.408125
## as.factor(indfminc)11  0.068496   0.291704   0.235 0.814358
## as.factor(indfminc)12  0.051646   0.588995   0.088 0.930129
## as.factor(indfminc)13  0.685325   0.573634   1.195 0.232229
## as.factor(indfminc)77 -0.278565   0.706814  -0.394 0.693506
## as.factor(indfminc)99  0.142902   0.714263   0.200 0.841430
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.073 on 10097 degrees of freedom
## Multiple R-squared:  0.2805, Adjusted R-squared:  0.2788
## F-statistic:   164 on 24 and 10097 DF,  p-value: < 2.2e-16
```

#### Residual Plots

```
plot(bmireg1$residuals, x = ds1$ridageyr, xlab = "Age at screening", ylab = "Residuals")
abline(0,0)
```

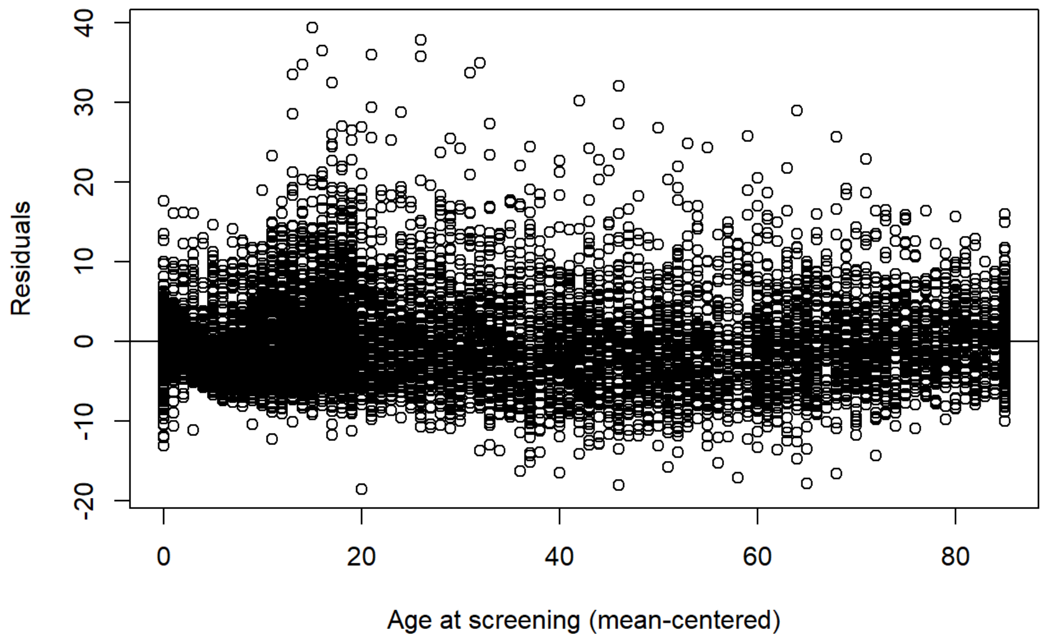



There seems to be a quadratic trend in the residual plot of age. Therefore, we create a quadratic term for ridageyr.

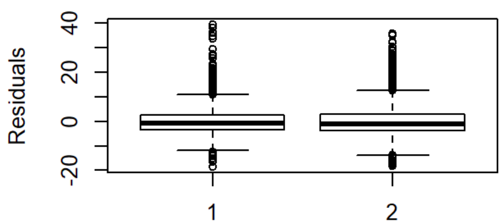
**Modelling: Square term**

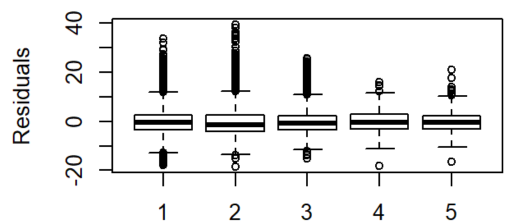
```
##
## Call:
## lm(formula = bmxbmi ~ ridageyr + ridageyr2 + as.factor(riagendr) +
##     as.factor(ridreth2) + as.factor(dmdeduc) + as.factor(indfminc),
##     data = ds1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.527  -3.581  -0.847   2.567  39.333
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           15.5726750  0.2621772  59.398  < 2e-16 ***
## ridageyr               0.5202864  0.0095893  54.257  < 2e-16 ***
## ridageyr2             -0.0049113  0.0001124 -43.707  < 2e-16 ***
## as.factor(riagendr)2   0.7302948  0.1109361   6.583 4.84e-11 ***
## as.factor(ridreth2)2   1.1682023  0.1458114   8.012 1.26e-15 ***
## as.factor(ridreth2)3   0.7826088  0.1522766   5.139 2.81e-07 ***
## as.factor(ridreth2)4  -1.7529815  0.3183974  -5.506 3.77e-08 ***
## as.factor(ridreth2)5   0.1068483  0.3176613   0.336 0.736607
## as.factor(dmdeduc)2    0.7062061  0.1844330   3.829 0.000129 ***
## as.factor(dmdeduc)3    0.1330526  0.1692138   0.786 0.431711
## as.factor(dmdeduc)7   -1.8966785  1.9749850  -0.960 0.336902
## as.factor(dmdeduc)9   -2.3024946  1.6867220  -1.365 0.172261
## as.factor(indfminc)2   0.2285475  0.3009928   0.759 0.447684
## as.factor(indfminc)3  -0.0987652  0.2762085  -0.358 0.720669
## as.factor(indfminc)4  -0.2993557  0.2895741  -1.034 0.301264
## as.factor(indfminc)5  -0.0434399  0.2877935  -0.151 0.880025
## as.factor(indfminc)6  -0.5507073  0.2727995  -2.019 0.043542 *
## as.factor(indfminc)7   0.2204093  0.2883350   0.764 0.444634
## as.factor(indfminc)8  -0.4991348  0.3004410  -1.661 0.096676 .
## as.factor(indfminc)9   0.1421204  0.3393920   0.419 0.675408
## as.factor(indfminc)10 -0.1875881  0.3626724  -0.517 0.605001
## as.factor(indfminc)11 -0.7046548  0.2680899  -2.628 0.008591 **
## as.factor(indfminc)12 -0.1645642  0.5401579  -0.305 0.760632
## as.factor(indfminc)13  0.4096446  0.5260859   0.779 0.436195
## as.factor(indfminc)77 -0.3210934  0.6481817  -0.495 0.620346
## as.factor(indfminc)99 -0.0500173  0.6550261  -0.076 0.939135
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.569 on 10096 degrees of freedom
## Multiple R-squared:  0.3949, Adjusted R-squared:  0.3934
## F-statistic: 263.6 on 25 and 10096 DF,  p-value: < 2.2e-16
```

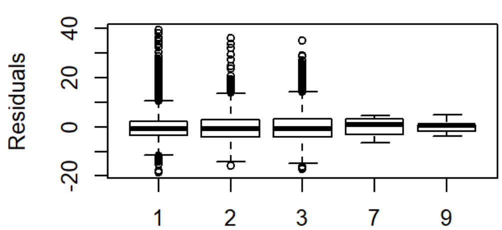
**Residual Plots**

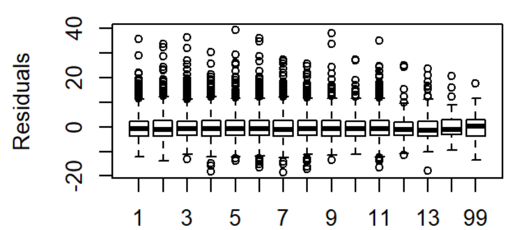



R-sqaure improves quite significantly when we try square of ridageyr, at the same time points are more evenly distributed along x axis in residual plot for ridageyr. The hook pattern disappears.




All plots look good (satisfied linearity and constant-variance assumption). This is out final model. We could apply multiple imputation combining rule to obtain point estimates and confidence interval.

**Multiple Imputation inferences on all m=10 data sets**

```
#make transformed ridageryr since it has no missing values.
nhanes$ridageyr2 <- nhanes$ridageyr^2
nhanes2.mi10 <- mice(nhanes, m = 10, defaultMethod = c("norm", "logreg", "polyreg", "polr"), set.seed(2018), print = FA
LSE)
bmiregMI10 <- with(data = nhanes2.mi10, lm (bmxbmi~ ridageyr + ridageyr2 + as.factor(riagendr) + as.factor(ridreth2) +
 as.factor(dmdeduc) + as.factor(indfminc)))
bmiregin <- pool(bmiregMI10)
summary(bmiregin, conf.int = T)
```

```
##                          estimate    std.error    statistic        df
## (Intercept)          15.616935489 0.2746022873  56.87110490  746.0691
## ridageyr              0.525299299 0.0102922265  51.03845125  538.4723
## ridageyr2            -0.004965364 0.0001203143 -41.26993446  520.3285
## as.factor(riagendr)2   0.712893495 0.1137682549   6.26618995 1725.0230
## as.factor(ridreth2)2   1.122875772 0.1482260792   7.57542653 2646.4382
## as.factor(ridreth2)3   0.688358858 0.1573488258   4.37473146 1248.1222
## as.factor(ridreth2)4  -1.593877977 0.3325876384  -4.79235484  838.8206
## as.factor(ridreth2)5   0.134571038 0.3200646441   0.42044956 4879.5211
## as.factor(dmdeduc)2    0.602245812 0.1943116042   3.09938161 1031.9893
## as.factor(dmdeduc)3    0.087026990 0.1834640891   0.47435436  424.6384
## as.factor(dmdeduc)7   -2.427354805 2.4446103487  -0.99294139  170.7487
## as.factor(dmdeduc)9   -1.980058289 1.8007340204  -1.09958398  119.1043
## as.factor(indfminc)2   0.046458442 0.3071658808   0.15124870 2508.9927
## as.factor(indfminc)3  -0.126408793 0.2827473479  -0.44707331 1876.8351
## as.factor(indfminc)4  -0.331396071 0.2930366804  -1.13090304 3411.9180
## as.factor(indfminc)5   0.017204102 0.2970199475   0.05792238 1344.3925
## as.factor(indfminc)6  -0.495116172 0.2875877897  -1.72161750  622.4756
## as.factor(indfminc)7   0.197255086 0.3033309908   0.65029651  662.3085
## as.factor(indfminc)8  -0.495675970 0.3118591539  -1.58942254 1088.7367
## as.factor(indfminc)9   0.200637315 0.3480139317   0.57652093 1683.0070
## as.factor(indfminc)10 -0.281639120 0.3774345109  -0.74619334  878.1597
## as.factor(indfminc)11 -0.690020893 0.2760657064  -2.49948066 1340.5861
## as.factor(indfminc)12 -0.428098021 0.5776071211  -0.74115780  356.2812
## as.factor(indfminc)13  0.310226438 0.5627388999   0.55127953  336.5218
## as.factor(indfminc)77 -0.449210127 0.6565008169  -0.68424915 1552.8862
## as.factor(indfminc)99 -0.275843577 0.6614426089  -0.41703327 2677.0899
##                           p.value        2.5 %       97.5 %
## (Intercept)          0.000000e+00 15.077850351 16.156020628
## ridageyr             0.000000e+00  0.505081462  0.545517135
## ridageyr2            0.000000e+00 -0.005201725 -0.004729002
## as.factor(riagendr)2  4.018215e-10  0.489755249  0.936031740
## as.factor(ridreth2)2  4.263256e-14  0.832225065  1.413526478
## as.factor(ridreth2)3  1.241174e-05  0.379661473  0.997056244
## as.factor(ridreth2)4  1.697277e-06 -2.246679698 -0.941076256
## as.factor(ridreth2)5  6.741756e-01 -0.492899780  0.762041857
## as.factor(dmdeduc)2   1.950293e-03  0.220954880  0.983536745
## as.factor(dmdeduc)3   6.352685e-01 -0.273583830  0.447637810
## as.factor(dmdeduc)7   3.207878e-01 -7.252904800  2.398195189
## as.factor(dmdeduc)9   2.715677e-01 -5.545659437  1.585542858
## as.factor(indfminc)2  8.797858e-01 -0.555866187  0.648783071
## as.factor(indfminc)3  6.548420e-01 -0.680941024  0.428123438
## as.factor(indfminc)4  2.581515e-01 -0.905941228  0.243149085
## as.factor(indfminc)5  9.538128e-01 -0.565468873  0.599877078
## as.factor(indfminc)6  8.520221e-02 -1.059875985  0.069643641
## as.factor(indfminc)7  5.155313e-01 -0.398351163  0.792861336
## as.factor(indfminc)8  1.120298e-01 -1.107588939  0.116237000
## as.factor(indfminc)9  5.642897e-01 -0.481948345  0.883222975
## as.factor(indfminc)10 4.555866e-01 -1.022418157  0.459139917
## as.factor(indfminc)11 1.247014e-02 -1.231588688 -0.148453098
## as.factor(indfminc)12 4.586334e-01 -1.564046004  0.707849963
## as.factor(indfminc)13 5.814673e-01 -0.796702563  1.417155440
## as.factor(indfminc)77 4.938503e-01 -1.736931758  0.838511504
## as.factor(indfminc)99 6.766724e-01 -1.572833658  1.021146504
```