PS3

Huanye Liu

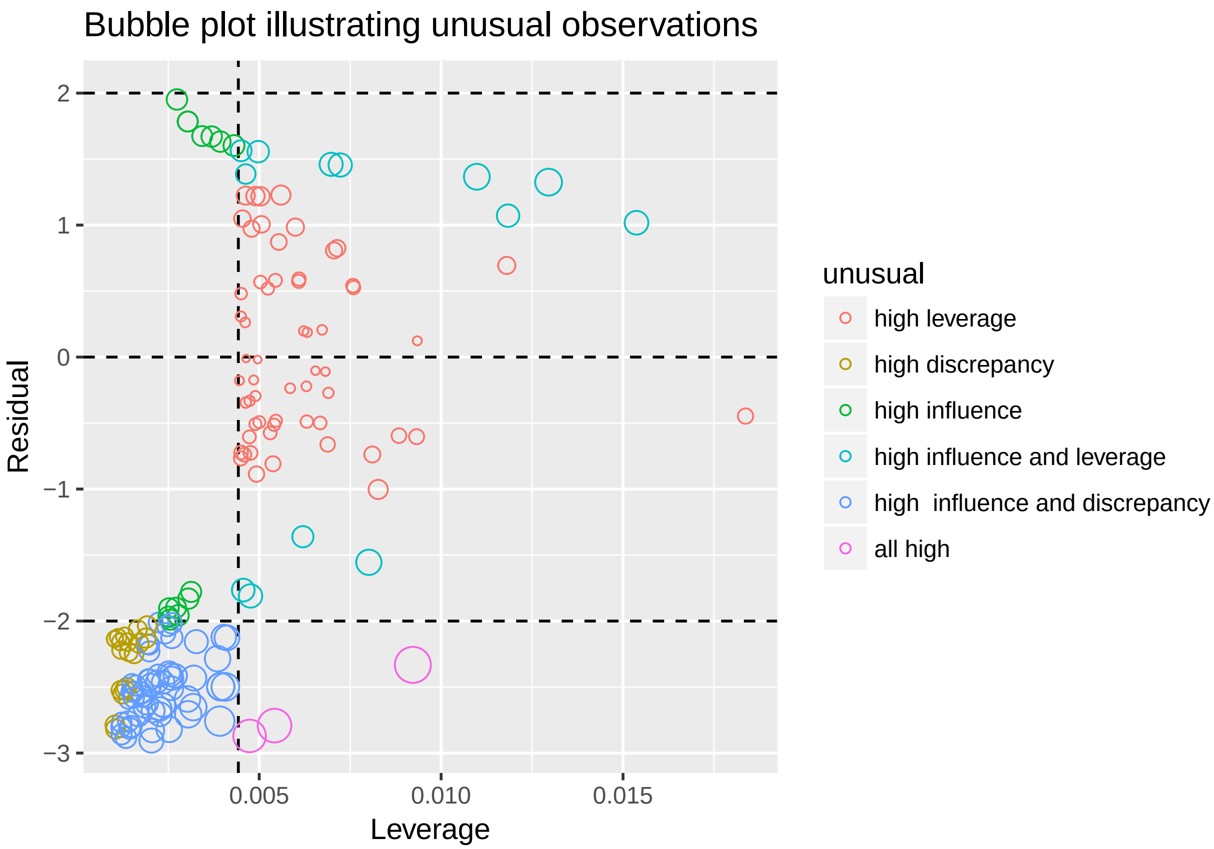
```
bd=lm(biden~age+female+educ,data=biden)
bd$coefficients
```

```
## (Intercept)         age      female        educ
## 68.62101396  0.04187919  6.19606946 -0.88871263
```

Regression diagnostics

1 We use a colored bubble plot to illustrate the leverage and discrepancy for each observation as shown below. From the bubble plot we can see among all 167 either high leverage or high discrepancy observations, 90 observations have high influence, most of which are due to high discrepancy.

```
biden_<- biden %>%
  mutate(hat = hatvalues(bd),
         student = rstudent(bd),
         cooksd = cooks.distance(bd)) %>%
  mutate(lev = ifelse(hat > 2 * mean(hat), 2, 1),
         discre = ifelse(abs(student) > 2, 20, 10),
         influ = ifelse(cooksd > 4/(nrow(.) - (length(coef(bd)) - 1) - 1), 200, 100))

b_estimate <- mean(biden_$hat)

biden_ %>%
  dplyr::filter(lev == 2 | discre == 20 | influ == 200) %>%
  mutate(unusual = lev + discre + influ) %>%
  mutate(unusual = factor(unusual, levels = c(112, 121, 211, 212, 221,222), labels = c("high leverage",
  {.} -> biden_e



ggplot(biden_e, aes(hat, student)) +
  geom_hline(yintercept = 0, linetype = 2) +
  geom_hline(yintercept = 2, linetype = 2) +
  geom_hline(yintercept = -2, linetype = 2) +
  geom_vline(xintercept = 2*b_estimate, linetype = 2) +
  geom_point(aes(size = cooksd, color = unusual), shape = 1) +
  labs(title = "Bubble plot illustrating unusual observations",
       x = "Leverage",
       y = "Residual") +
  scale_size(guide = "none")
```

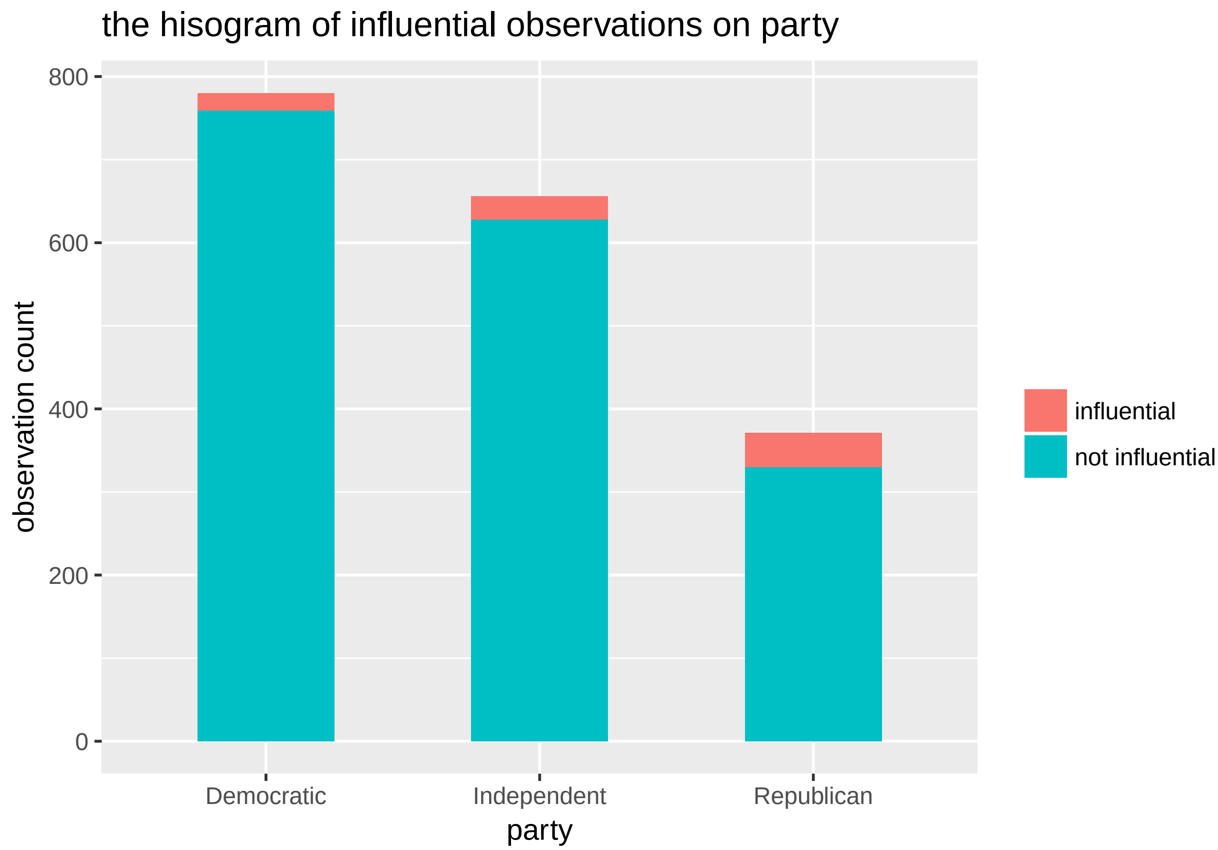



```
all = nrow(biden_e)
n_influence = nrow(filter(biden_e, influ==200))
```

To decide how to deal with these unusual observations, we need to further look at the histogram of influential observations based on participant's party affiliation, which shows that the party affiliation may be an important determinant to the influential observations. Therefore, we could respecify the model by adding the attributes rep and dem to control for the influential effect.

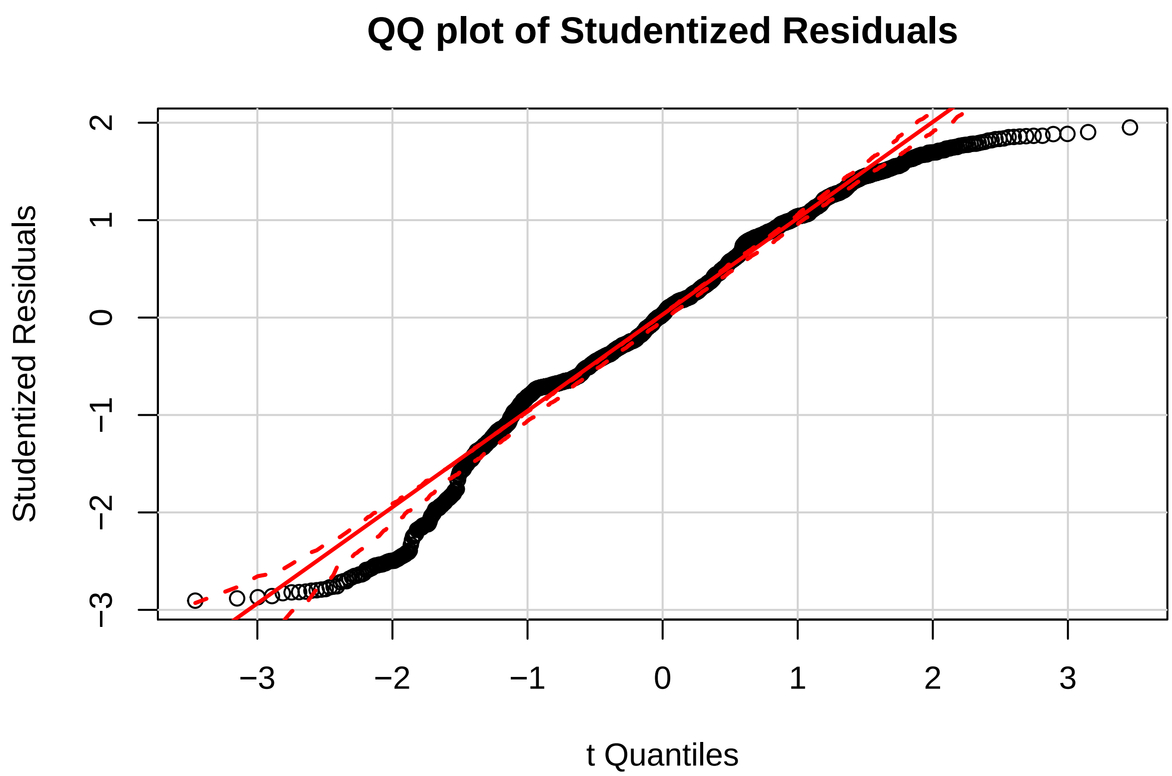
```
biden_ %>%
  mutate(influential = factor(ifelse(influ == 200, "influential", "not influential"))) %>%
  mutate(party = ifelse(dem==1, "Democratic", ifelse(rep==1, "Republican", "Independent"))) %>%
  {.} -> biden_e

ggplot(biden_e, mapping = aes(x = party)) +
  geom_histogram(mapping = aes(fill = influential), width = 0.5, stat="count") +
  labs(title = "the hisogram of influential observations on party ",
        x = "party",
        y = "observation count") +
  guides(fill = guide_legend(title = ''))
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

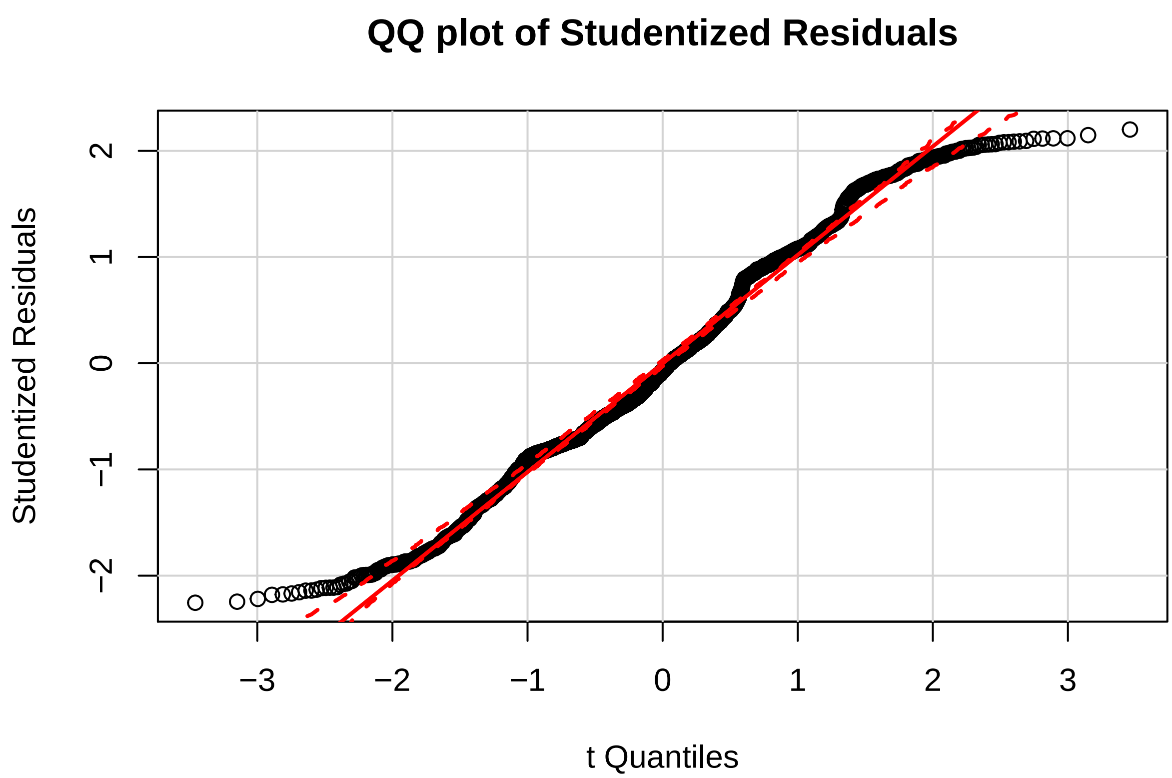


2 The plot below shows the non-normally distributed errors because the dot plot deviates from the straight line to a relatively large extent. We could fix this problem by power-transforming the outcome or predictors, and we choose to exponentiate the outcome variable to 1.5 in this case.

```
car::qqPlot(bd, main = "QQ plot of Studentized Residuals",
            ylab = "Studentized Residuals")
```




```
bd1 = lm(biden^1.5~age+female+educ,data=biden)
car::qqPlot(bd1, main = "QQ plot of Studentized Residuals",
            ylab = "Studentized Residuals")
```



So we can see from the plot above that the the dot plot line are more straight that the original one after the transformation.

3 Using the Breusch-Pagan test, we do find significant heteroskedasticity in the margin errors for our model, which means the estimated standard errors of predictor coeffecients are biased estimates.

```
bptest(bd)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  bd
## BP = 22.559, df = 3, p-value = 4.989e-05
```

4 Using the vif command, we can check the multicollinearity problem, and the result shows that no multicollinearality between predictors exists in the model.

```
car::vif(bd)
```

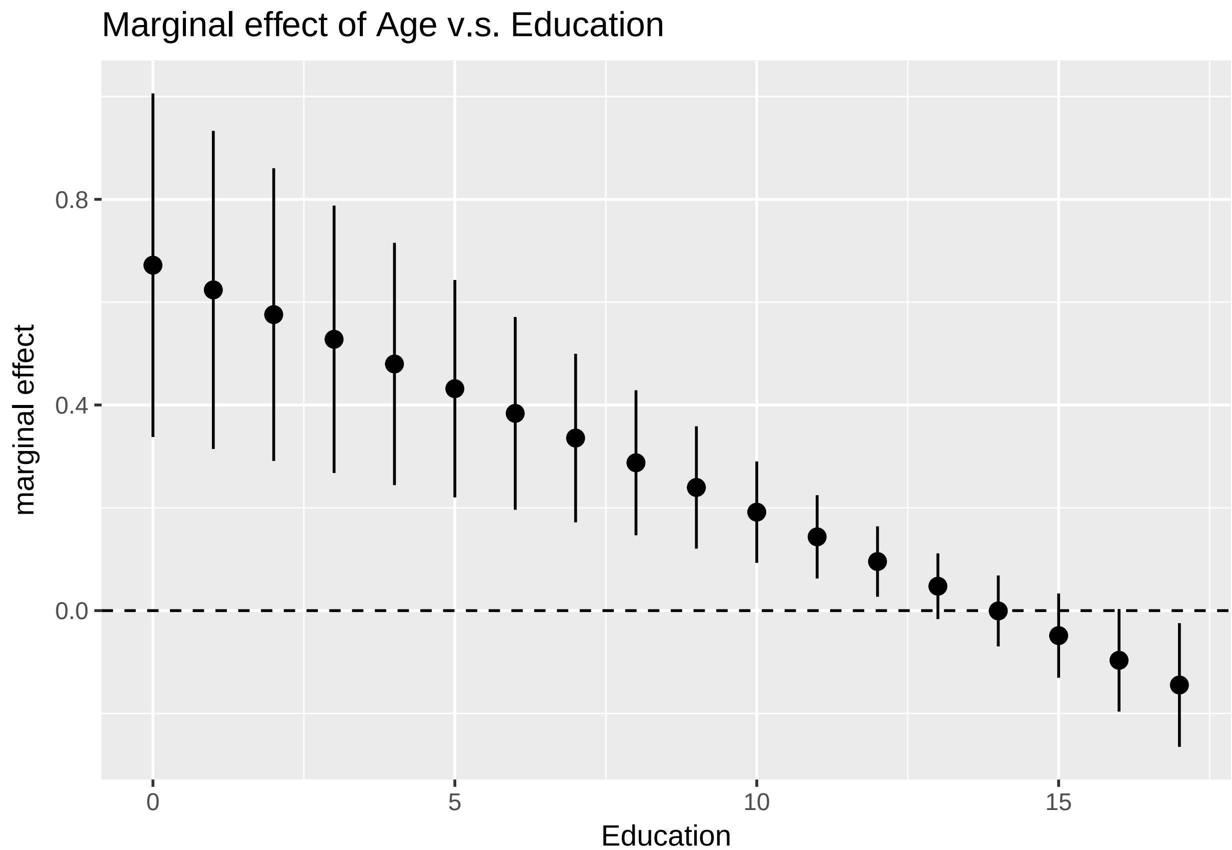
```
##      age   female     educ
## 1.013369 1.001676 1.012275
```

Interaction Terms

```
bd_inter <- lm(biden ~ age + educ + age*educ, data = biden)
```

1 Running the code below, we can see that marginal effect of age is significant, and as the years of education increase, the marginal effect decreases.

```
effect <- function(model, mod_var){
  int.name <- names(model$coefficients)[[which(str_detect(names(model$coefficients), ":"))]]
  marg_var <- str_split(int.name, ":")[[1]][[which(str_split(int.name, ":")[[1]] != mod_var)]]
  beta.hat <- coef(model)
  cov <- vcov(model)
  if(class(model)[[1]] == "lm"){
    z <- seq(min(model$model[[mod_var]]), max(model$model[[mod_var]]))
  } else {
    z <- seq(min(model$data[[mod_var]]), max(model$data[[mod_var]]))
  }
  dy.dx <- beta.hat[[marg_var]] + beta.hat[[int.name]] * z
  se.dy.dx <- sqrt(cov[marg_var, marg_var] +
                     z^2 * cov[int.name, int.name] +
                     2 * z * cov[marg_var, int.name])
  data_frame(z = z,
             dy.dx = dy.dx,
             se = se.dy.dx)
}

effect(bd_inter, "educ") %>%
  ggplot(aes(z, dy.dx,
             ymin = dy.dx - 1.96 * se,
             ymax = dy.dx + 1.96 * se)) +
  geom_pointrange() +
  geom_hline(yintercept = 0, linetype = 2) +
  labs(title = "Marginal effect of Age v.s. Education",
       x = "Education",
       y = "marginal effect")
```

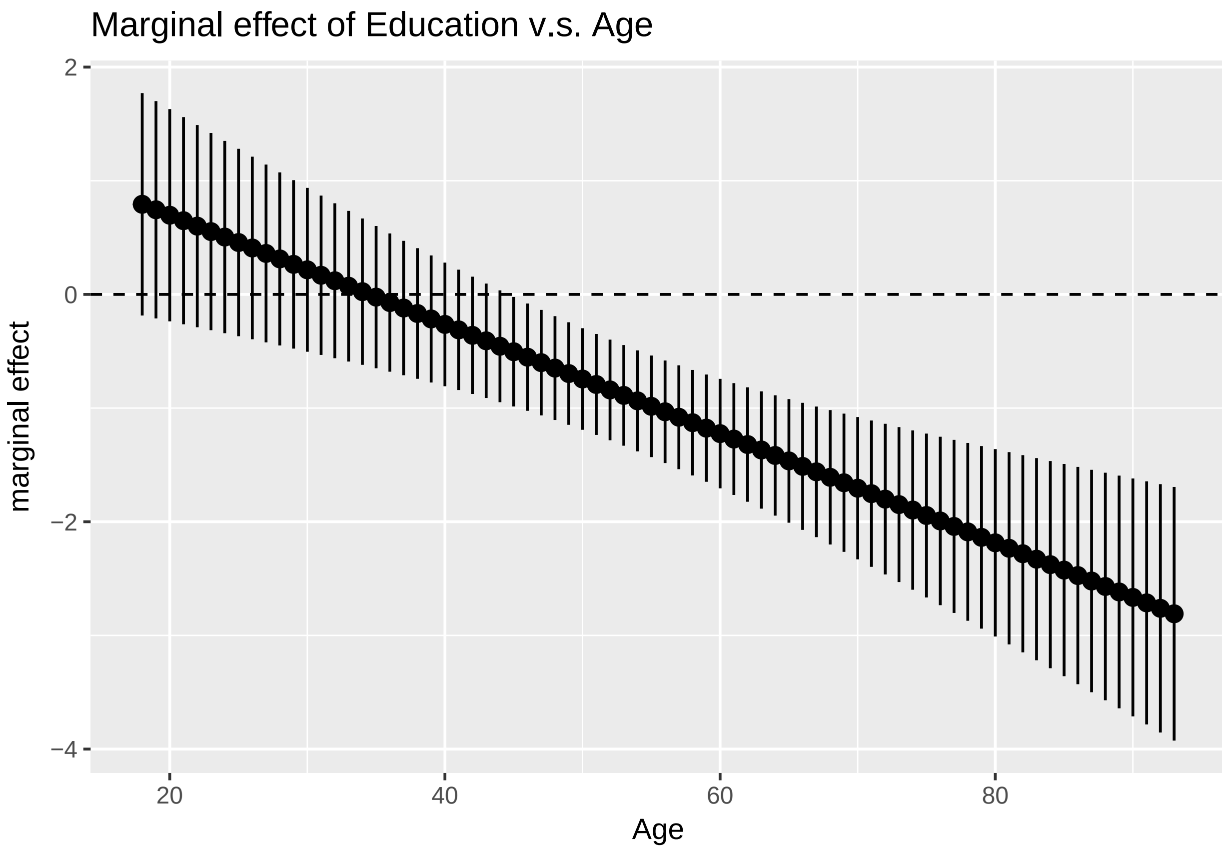



```r
linearHypothesis(bd_inter, "age + age:educ")
```

```
## Linear hypothesis test
##
## Hypothesis:
## age  + age:educ = 0
##
## Model 1: restricted model
## Model 2: biden ~ age + educ + age * educ
##
##   Res.Df    RSS Df Sum of Sq     F    Pr(>F)
## 1   1804 985149
## 2   1803 976688  1    8461.2 15.62 8.043e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2 Similarly, we can see from the graph below that marginal effect of education is also significant, and as age increases, the marginal effect decreases.

```r
effect(bd_inter, "age") %>%
  ggplot(aes(z, dy.dx,
             ymin = dy.dx - 1.96 * se,
             ymax = dy.dx + 1.96 * se)) +
  geom_pointrange() +
  geom_hline(yintercept = 0, linetype = 2) +
  labs(title = "Marginal effect of Education v.s. Age",
       x = "Age",
       y = "marginal effect")
```




```
linearHypothesis(bd_inter, "educ + age:educ")
```

```
## Linear hypothesis test
##
## Hypothesis:
## educ  + age:educ = 0
##
## Model 1: restricted model
## Model 2: biden ~ age + educ + age * educ
##
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1   1804 979537
## 2   1803 976688  1    2849.1 5.2595 0.02194 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

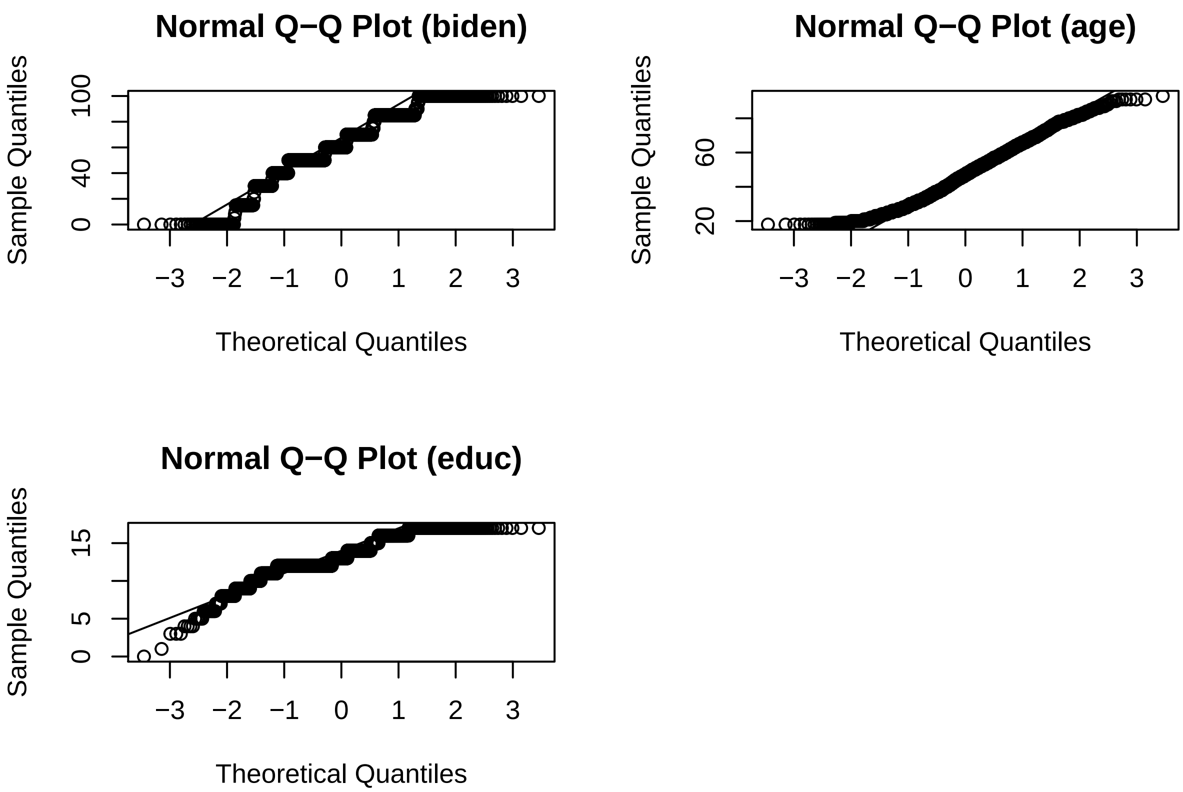
Missing data

First we test the multivariate normality. As the graph below shows, the dataset is not multivariate normal and we could transform the predictor age and the predictor education by squaring both.

```
biden_ <- biden %>%
  select(-female, -rep, -dem)
uniPlot(biden_, type = "qqplot")
mardiaTest(biden_, qqplot = FALSE)
```

```
##    Mardia's Multivariate Normality Test
```

```
## ---------------------------------------
##    data : biden_
##
##    g1p            : 1.026978
##    chi.skew       : 309.2915
##    p.value.skew   : 1.685187e-60
##
##    g2p            : 16.028
##    z.kurtosis     : 3.989148
##    p.value.kurt   : 6.631109e-05
##
##    chi.small.skew : 310.0622
##    p.value.small  : 1.157661e-60
##
##    Result          : Data are not multivariate normal.
## ---------------------------------------
```

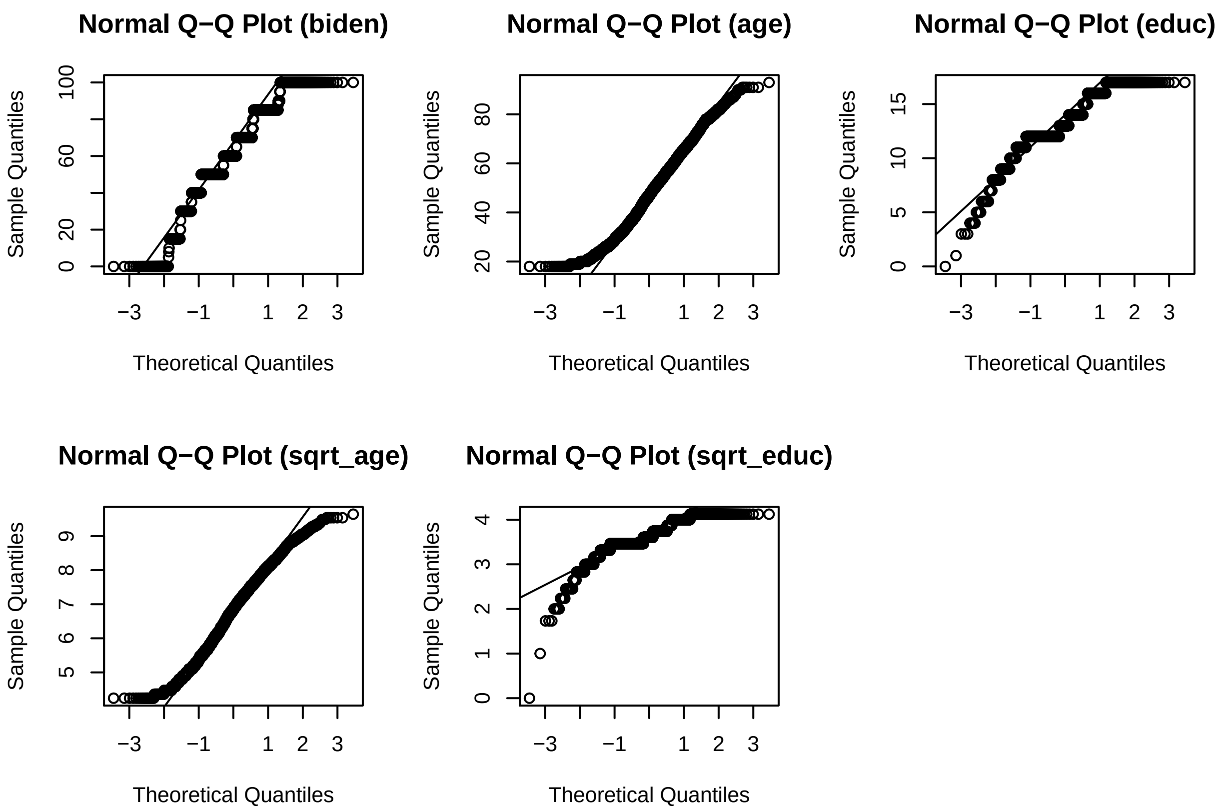


Below shows the QQ plot after the transforming:

```r
biden_trans <- biden_ %>%
  mutate(sqrt_age = sqrt(age),
         sqrt_educ = sqrt(educ))

uniPlot(biden_trans, type = "qqplot")
mardiaTest(biden_trans%>% select(sqrt_educ, sqrt_age), qqplot = FALSE)
```

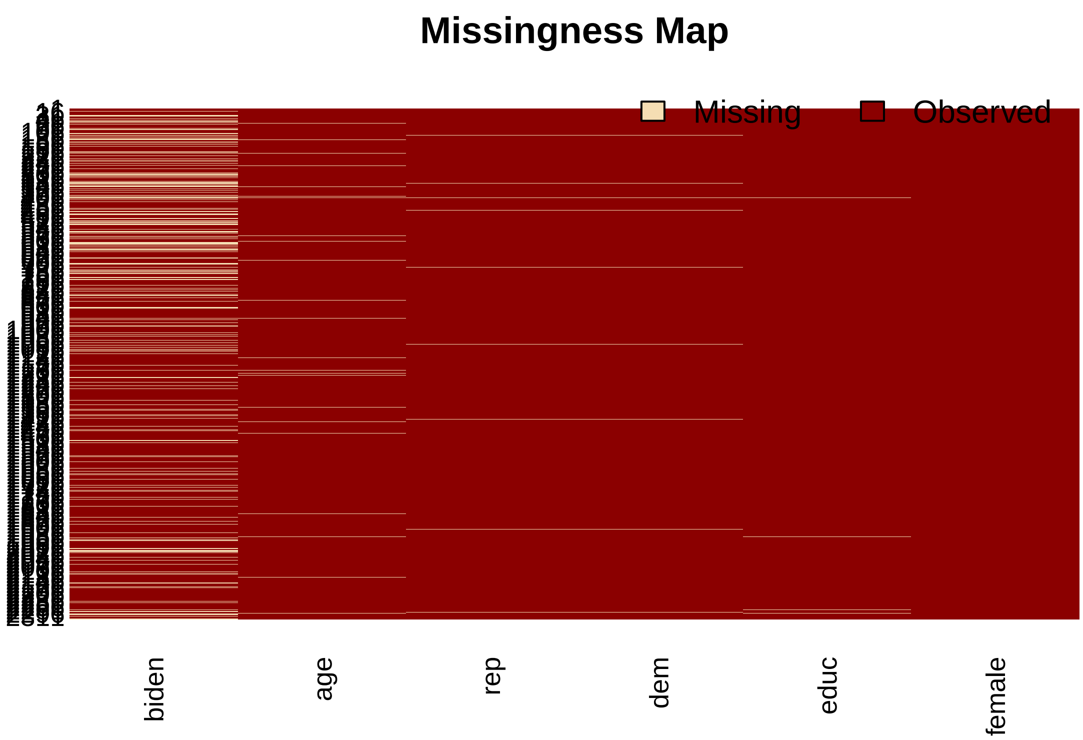
```
##    Mardia's Multivariate Normality Test
## ---------------------------------------
##    data : biden_trans %>% select(sqrt_educ, sqrt_age)
##
```

```
##    g1p            : 2.899308
##    chi.skew       : 873.1748
##    p.value.skew   : 1.080348e-187
##
##    g2p            : 16.64292
##    z.kurtosis     : 45.92505
##    p.value.kurt   : 0
##
##    chi.small.skew : 875.593
##    p.value.small  : 3.233329e-188
##
##    Result          : Data are not multivariate normal.
## ---------------------------------------
```



Now for the missingness in the data, we can use the missmap function as below:

```
biden.out <- biden_raw %>%
  mutate(dem = as.numeric(dem),
         rep = as.numeric(rep)) %>%
  amelia(., m=5, sqrts = c("age", "educ"),
         noms = c("female", "dem", "rep"), p2s = 0)
missmap(biden.out)
```



For comparision with the original non-imputed model, runing the following code, we can see from the table that there is no significant difference between models before and after the mutliple imputation procedure because of the relatively small number of missing values and the failing to meet the multivariate normality of the imputed model.

```
models_imp <- data_frame(data = biden.out$imputations) %>%
  mutate(model = map(data, ~ lm(biden ~ age + female + educ,
                                data = .x)),
         coef = map(model, tidy)) %>%
  unnest(coef, .id = "id")
models_imp
```

```
## # A tibble: 20 <U+00D7> 6
##       id        term    estimate  std.error statistic      p.value
##    <chr>       <chr>       <dbl>      <dbl>     <dbl>        <dbl>
## 1   imp1 (Intercept) 65.47948694 2.98740603 21.918509 6.189838e-97
## 2   imp1         age  0.04623947 0.02764602  1.672554 9.454993e-02
## 3   imp1      female  4.69304892 0.96227576  4.877031 1.149959e-06
## 4   imp1        educ -0.61131493 0.18534875 -3.298188 9.878144e-04
## 5   imp2 (Intercept) 64.71126294 2.94625506 21.963904 2.708561e-97
## 6   imp2         age  0.06991513 0.02727674  2.563177 1.043448e-02
## 7   imp2      female  5.02055147 0.95137893  5.277131 1.434113e-07
## 8   imp2        educ -0.67247514 0.18326041 -3.669506 2.484999e-04
## 9   imp3 (Intercept) 64.96594420 3.02391461 21.484054 1.593238e-93
## 10  imp3         age  0.04246986 0.02808372  1.512259 1.306043e-01
## 11  imp3      female  6.23829209 0.97484213  6.399284 1.882721e-10
## 12  imp3        educ -0.61219534 0.18781528 -3.259561 1.131985e-03
## 13  imp4 (Intercept) 66.69889786 3.01359894 22.132639 1.242494e-98
## 14  imp4         age  0.06000351 0.02797523  2.144880 3.206604e-02
## 15  imp4      female  4.83459067 0.97250680  4.971267 7.138255e-07
## 16  imp4        educ -0.76984007 0.18733987 -4.109323 4.106049e-05
```

```
## 17  imp5 (Intercept) 66.84449561 2.96776708 22.523498 9.302396e-102
## 18  imp5         age  0.07508669 0.02764793  2.715816  6.660305e-03
## 19  imp5      female  5.70179897 0.96225702  5.925443  3.579021e-09
## 20  imp5        educ -0.86956258 0.18484690 -4.704231  2.698067e-06
```

```
mi.meld.plus <- function(df_tidy){

  coef.out <- df_tidy %>%
    select(id:estimate) %>%
    spread(term, estimate) %>%
    select(-id)

  se.out <- df_tidy %>%
    select(id, term, std.error) %>%
    spread(term, std.error) %>%
    select(-id)

  combined.results <- mi.meld(q = coef.out, se = se.out)

  data_frame(term = colnames(combined.results$q.mi),
             estimate.mi = combined.results$q.mi[1, ],
             std.error.mi = combined.results$se.mi[1, ])
}


tidy(bd) %>%
  left_join(mi.meld.plus(models_imp)) %>%
  select(-statistic, -p.value)
```

```
## Joining, by = "term"
```

```
##          term    estimate  std.error estimate.mi std.error.mi
## 1 (Intercept) 68.62101396 3.59600465 65.74001751   3.17602206
## 2         age  0.04187919 0.03248579  0.05874293   0.03183097
## 3      female  6.19606946 1.09669702  5.29765642   1.20087157
## 4        educ -0.88871263 0.22469183 -0.70707761   0.22228097
```