# Coding

Zhendong Zhang

2020/10/13

```
gss <- read.csv("gss.csv")
str(gss)
```

```
## 'data.frame':    20602 obs. of  5 variables:
##  $ caseid           : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ age              : num  52.7 51.1 63.6 80 28 63 58.8 80 63.8 25.2 ...
##  $ income_respondent: int  37500 12500 37500 62500 12500 12500 12500 12500 12500 12500 ...
##  $ feelings_life    : int  8 10 8 10 8 9 4 10 8 5 ...
##  $ total_children   : int  1 5 5 1 0 2 2 7 0 1 ...
```

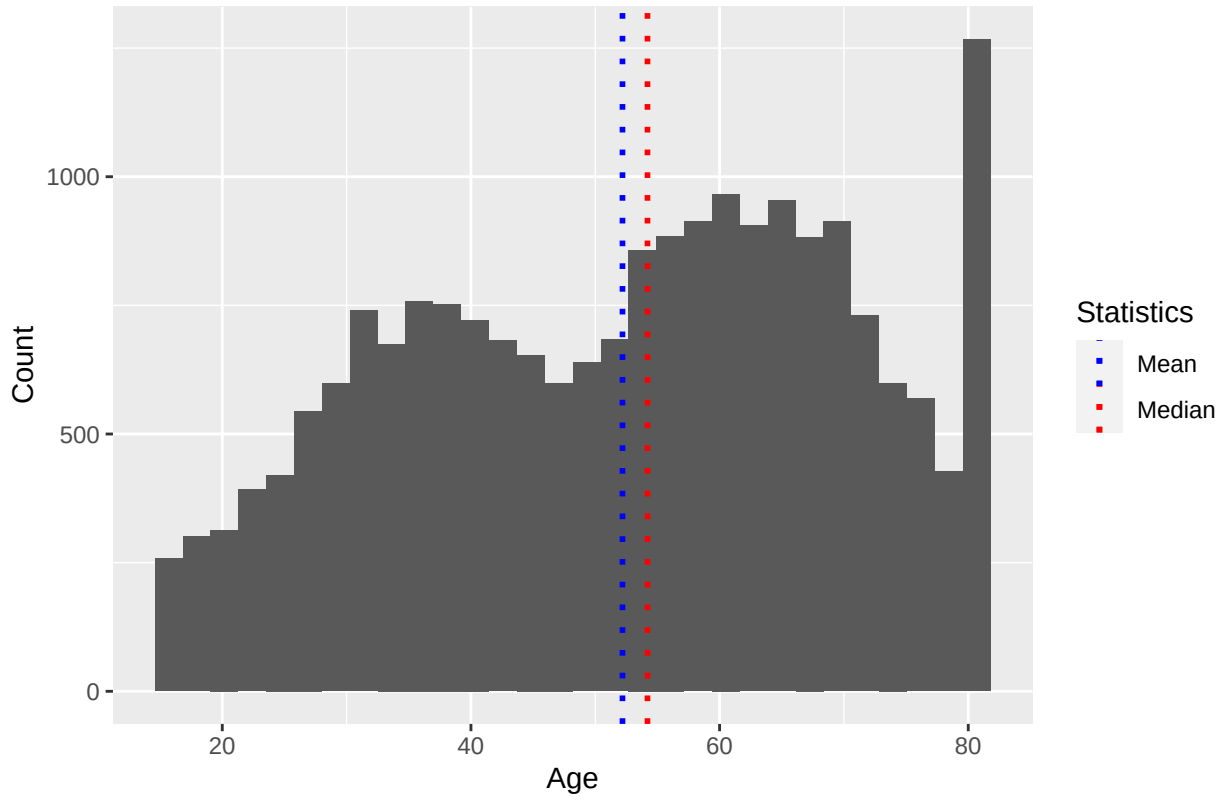
```
stat <- data.frame(row.names = paste(c("Mean", "Median", "Minimum", "Maximun")))
stat[,1] <- c(round(mean(gss$age), digits = 1),
              median(gss$age),
              min(gss$age),
              max(gss$age))
stat[,2] <- c(round(mean(gss$income_respondent), digits = 2),
              median(gss$income_respondent),
              min(gss$income_respondent),
              max(gss$income_respondent))
stat[,3] <- c(round(mean(na.omit(gss$feelings_life)), digits = 1),
              median(na.omit(gss$feelings_life)),
              min(na.omit(gss$feelings_life)),
              max(na.omit(gss$feelings_life)))
stat[,4] <- c(round(mean(na.omit(gss$total_children)), digits = 2),
              median(na.omit(gss$total_children)),
              min(na.omit(gss$total_children)),
              max(na.omit(gss$total_children)))
colnames(stat) <- c("Age", "Income", "Feelings of Life", "Number of Children")

grid.table(stat)
```

| | Age | Income | Feelings of Life | Number of Children |
|---|---|---|---|---|
| *Mean* | 52.2 | 45775.29 | 8.1 | 1.68 |
| *Median* | 54.2 | 37500 | 8 | 2 |
| *Minimum* | 15 | 12500 | 0 | 0 |
| *Maximun* | 80 | 125000 | 10 | 7 |

```
gss %>%
  ggplot(aes(x=age)) + geom_histogram() + labs(x = "Age", y = "Count", title = "Figure1.1: Age Distribu
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

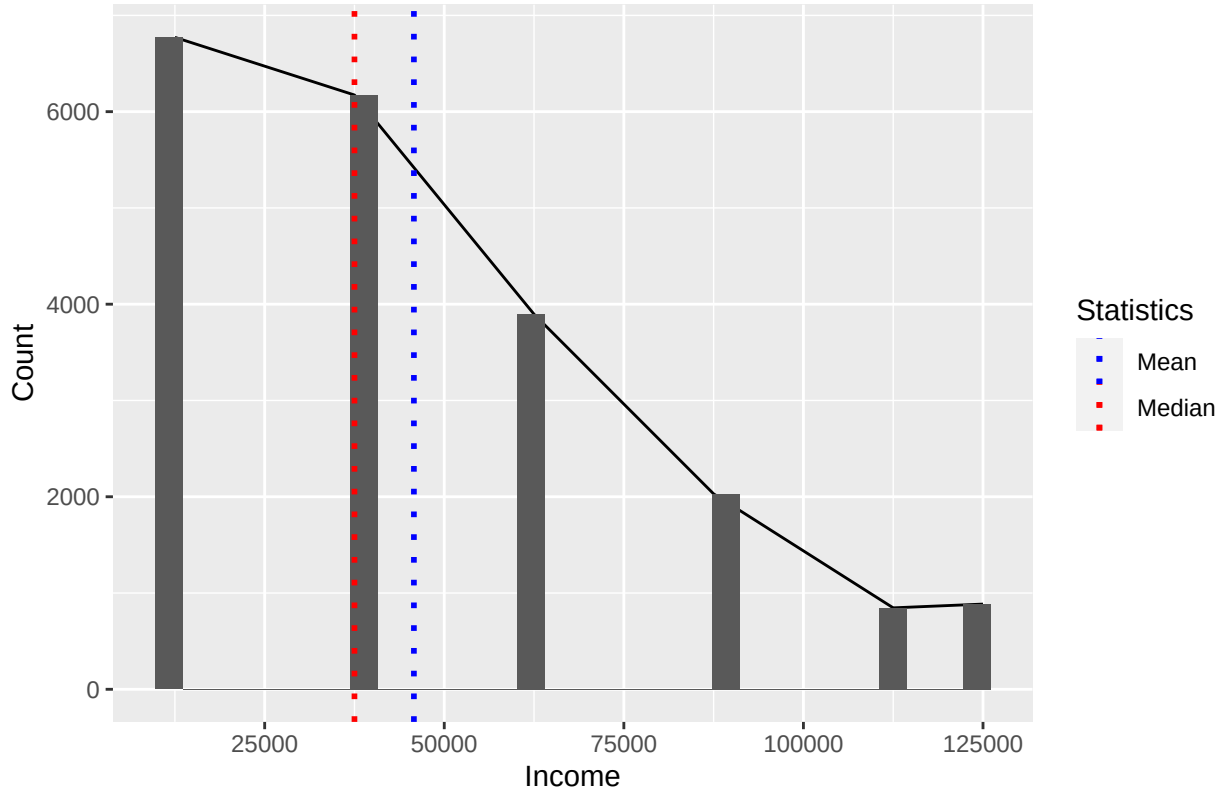
Figure1.1: Age Distribution




```
gss %>%
  ggplot(aes(x=income_respondent)) + geom_line(stat = "count") + geom_histogram() + labs(x = "Income", y
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

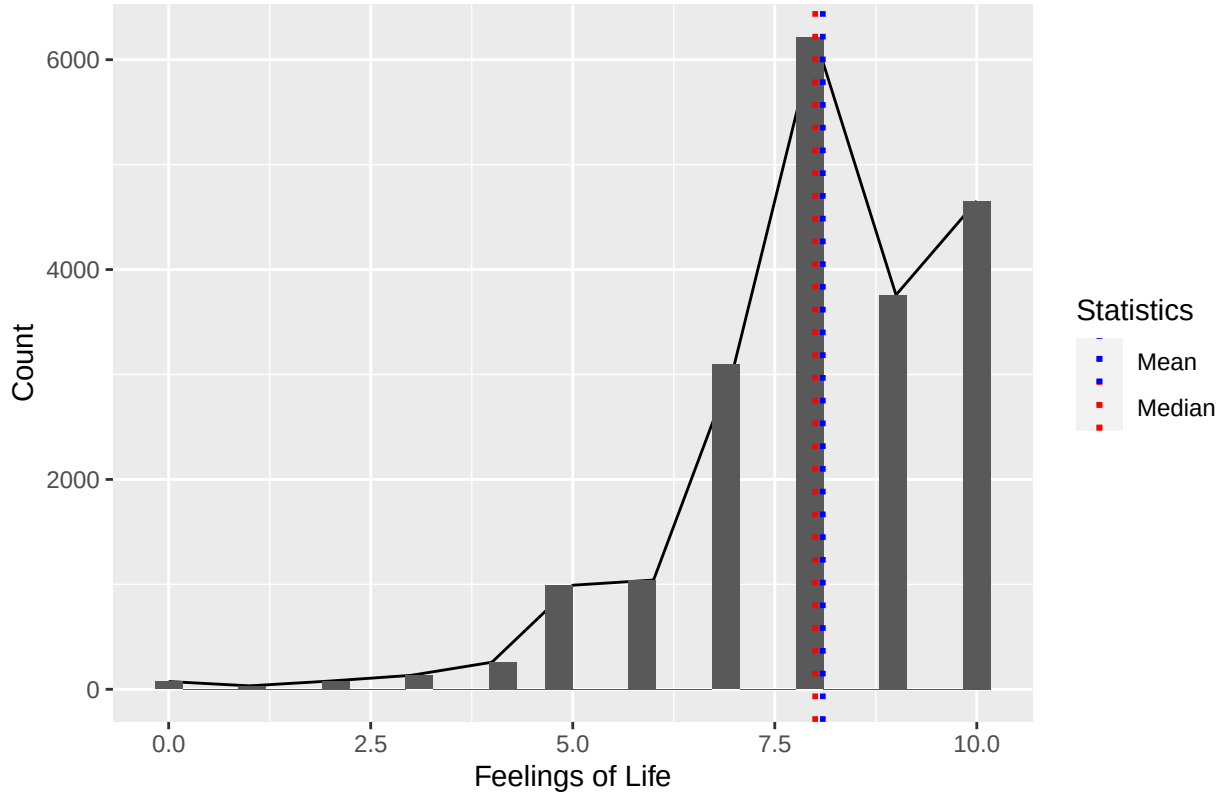
Figure1.2: Income Distribution




```
gss %>%
  ggplot(aes(x=feelings_life)) + geom_line(stat = "count") + geom_histogram() + labs(x = "Feelings of Li
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

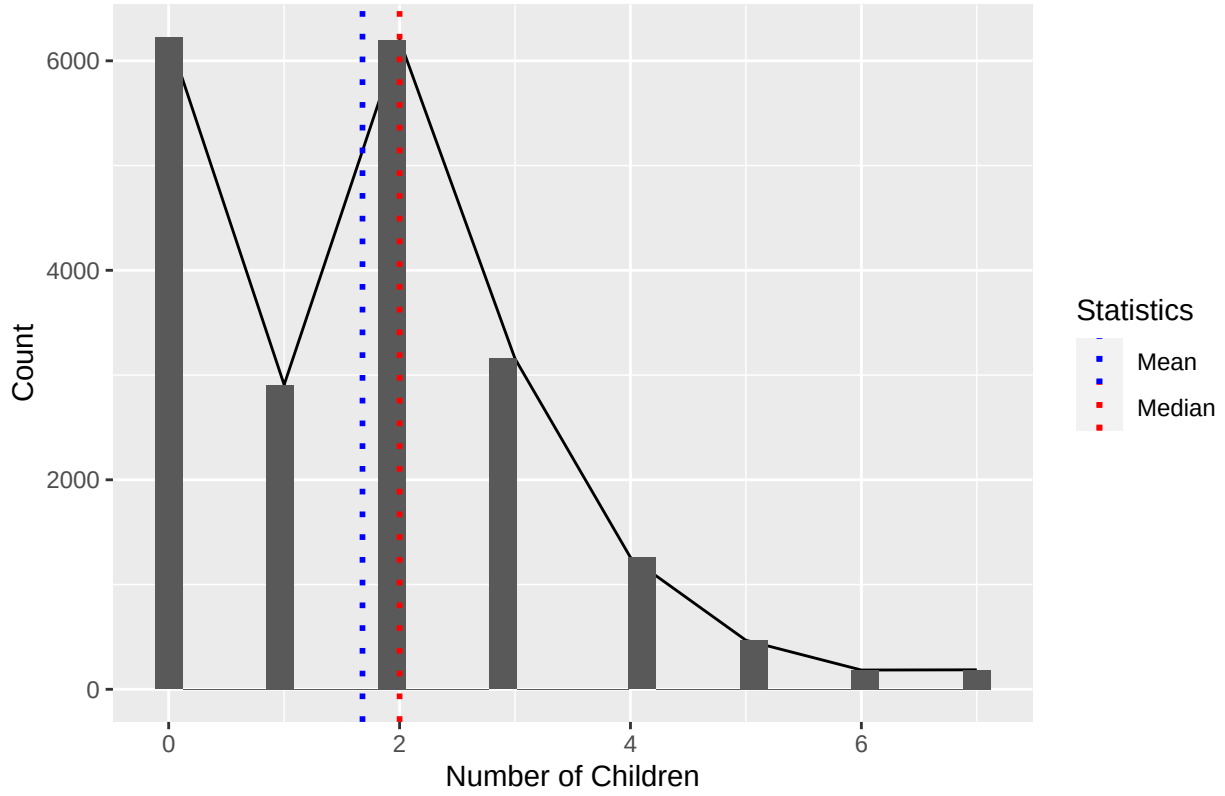
Figure1.3: Feelings of Life Distribution




```
gss %>%
  ggplot(aes(x=total_children)) + geom_line(stat = "count") + geom_histogram() + labs(x = "Number of Chi
```

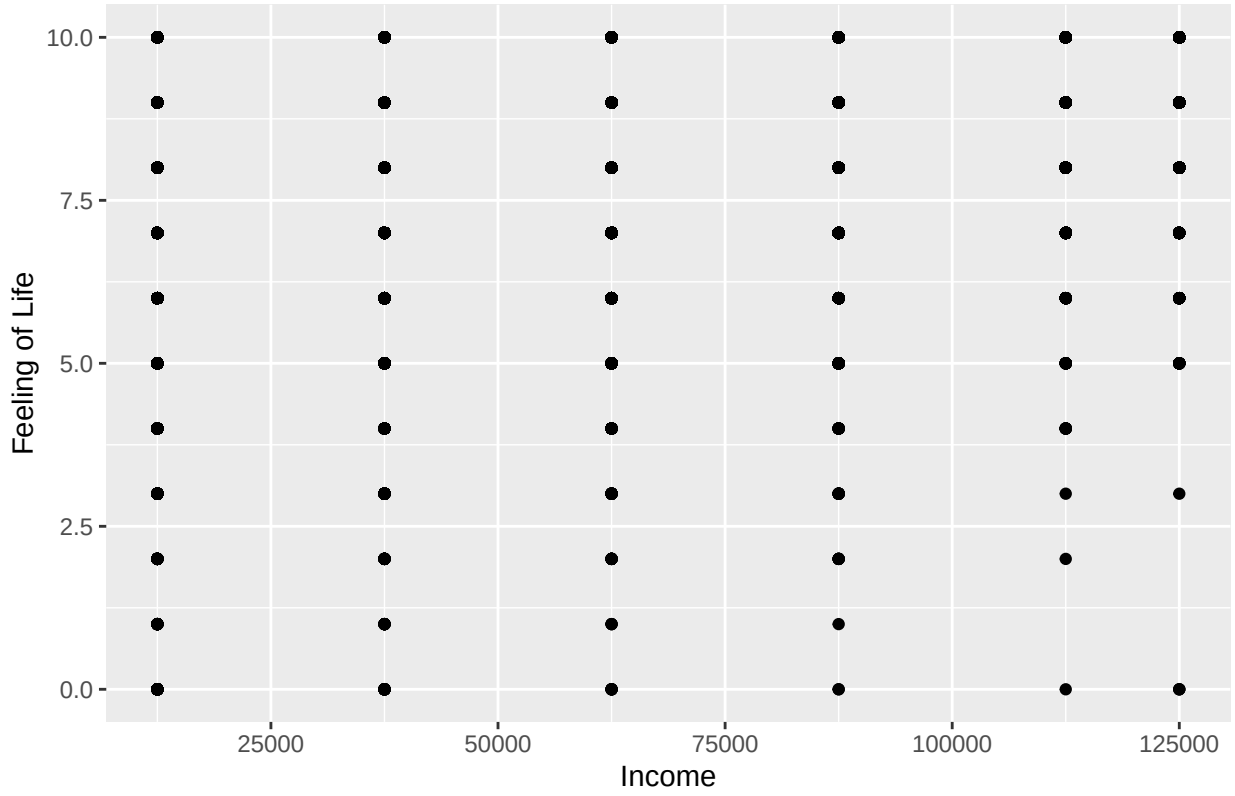
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Figure1.4: Total Number of Children



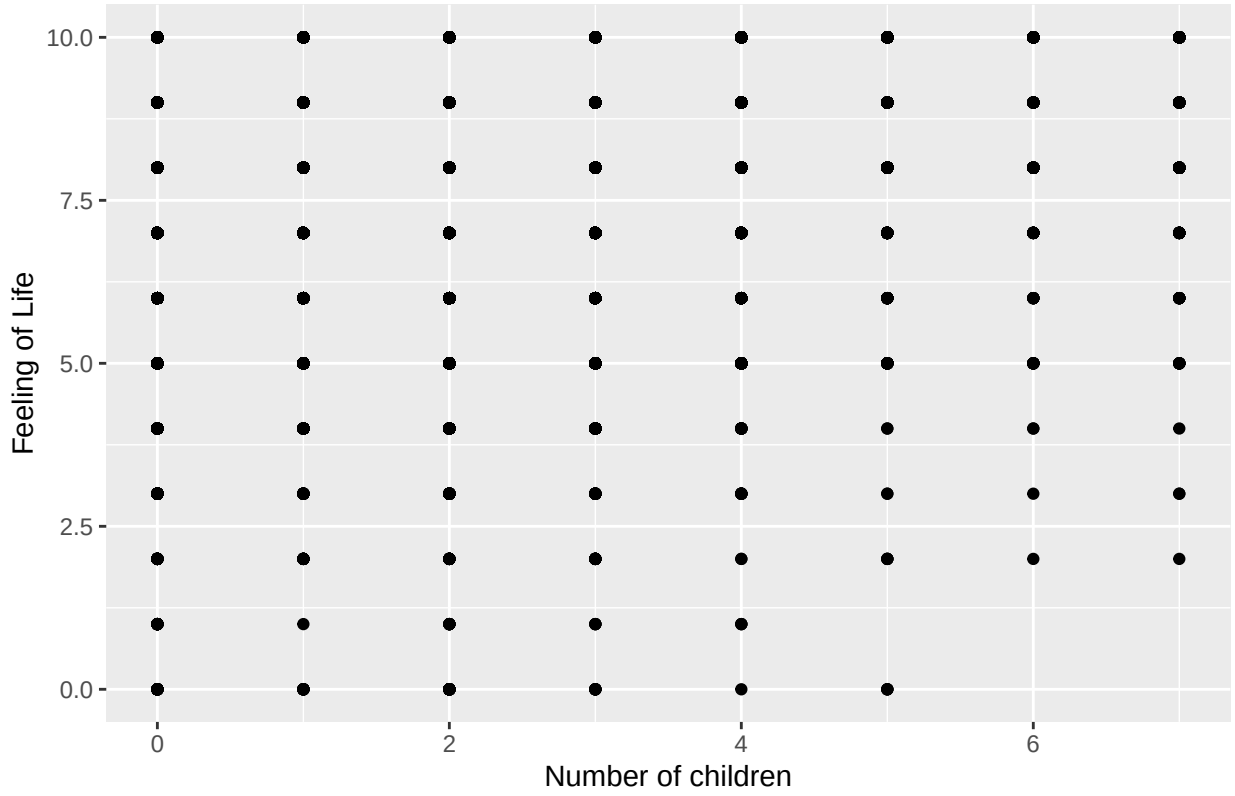

```
gss %>%
  ggplot(aes(y = feelings_life, x = income_respondent)) + geom_point() + labs(x = "Income", y = "Feelin
```

Figure: Income vs. Feeling of Life




```
gss %>%
  ggplot(aes(y = feelings_life, x = total_children)) + geom_point() + labs(x = "Number of children", y =
```

Figure: Number of Children vs. Feeling of Life

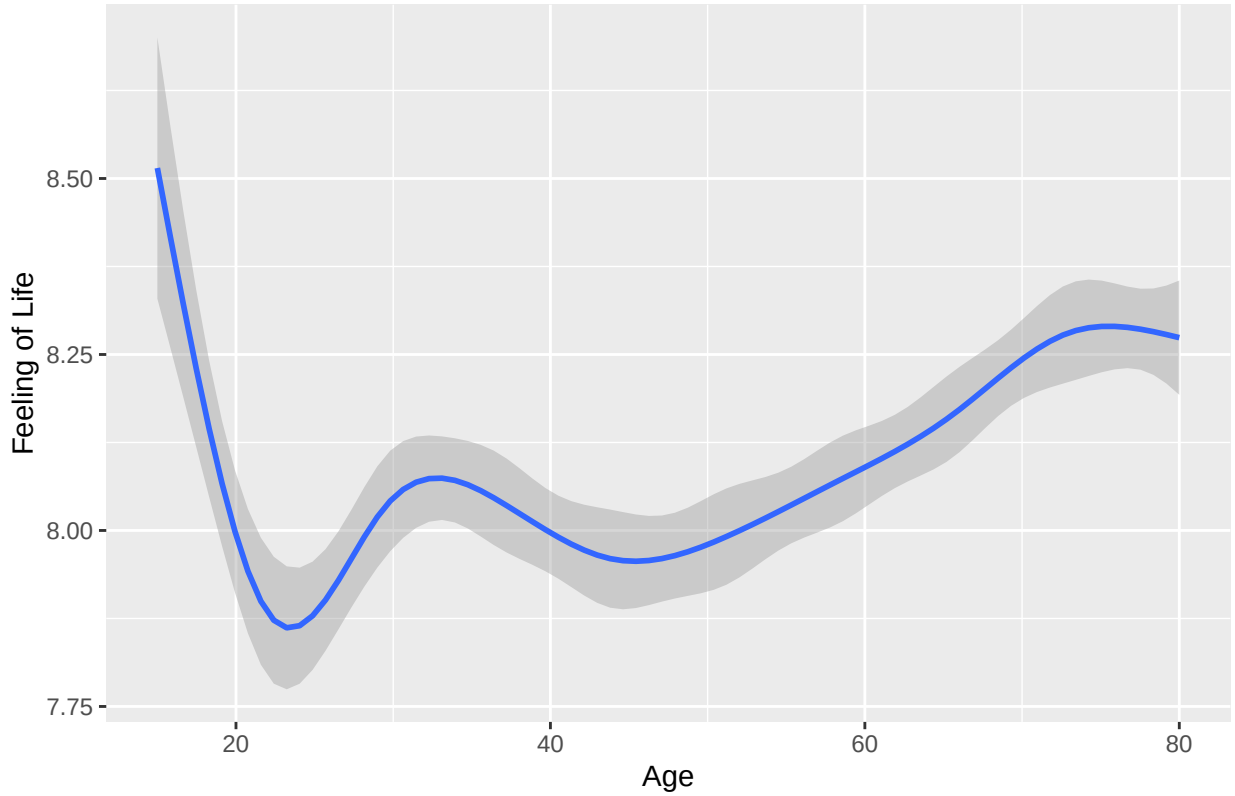



```
# These scatter plots are not useful, might be deleted.
```

```
gss_income <- gss %>%
  group_by(income_respondent) %>%
  summarise(mean_feel = mean(feelings_life, na.rm=T), .groups = 'drop')

gss_child <- gss %>%
  group_by(total_children) %>%
  summarise(mean_feel = mean(feelings_life, na.rm=T), .groups = 'drop')


gss %>%
  ggplot(aes(y = feelings_life, x = age)) + geom_smooth() + labs(x = "Age", y = "Feeling of Life", titl
```

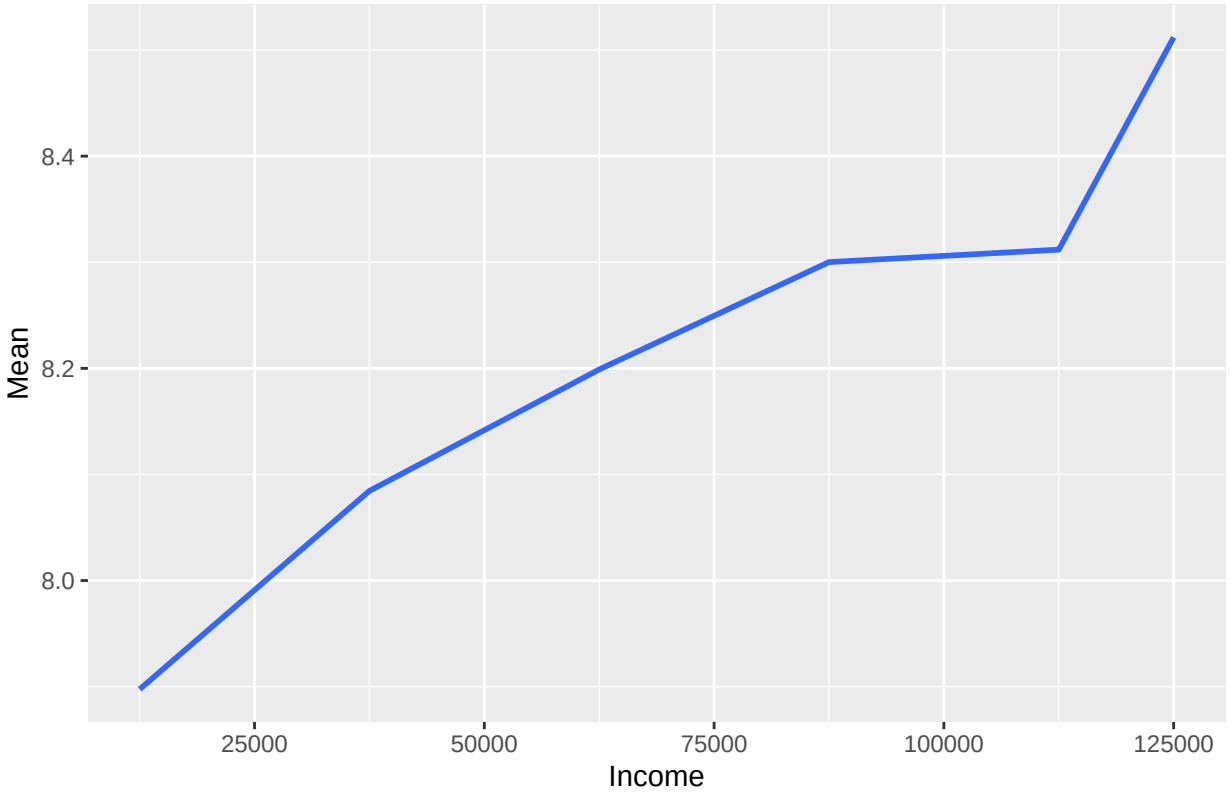
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

Figure2.1: Age vs. Feeling of Life



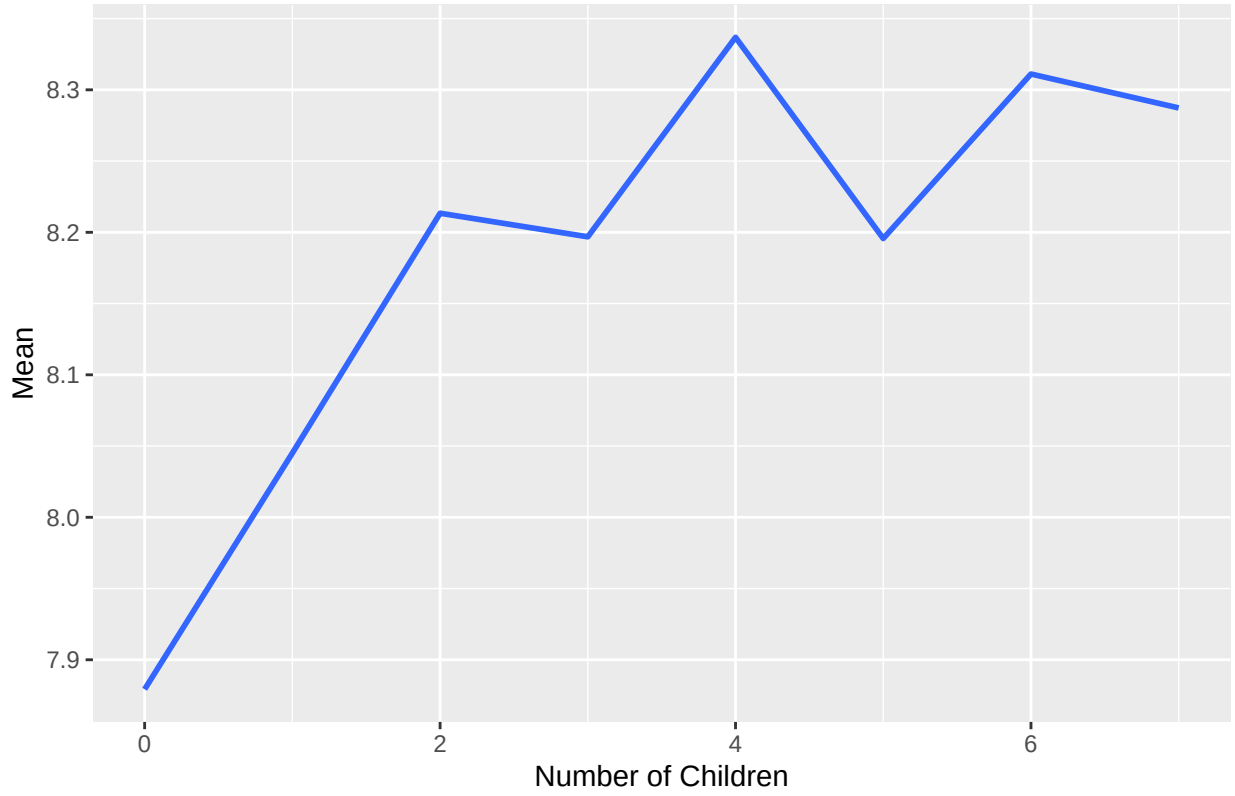

```
#Age can be plotted without mean, because ages are more consistent. Other variables would produce error


#I have plotted these two plot by using mean for each group, might be useful.

gss_income %>%
  ggplot(aes(y = mean_feel, x = income_respondent)) + geom_smooth(stat='identity') + labs(x = "Income",
```

Figure2.2: Income vs. Feeling of Life




```
gss_child %>%
  ggplot(aes(y = mean_feel, x = total_children)) + geom_smooth(stat='identity') + labs(x = "Number of Ch
```

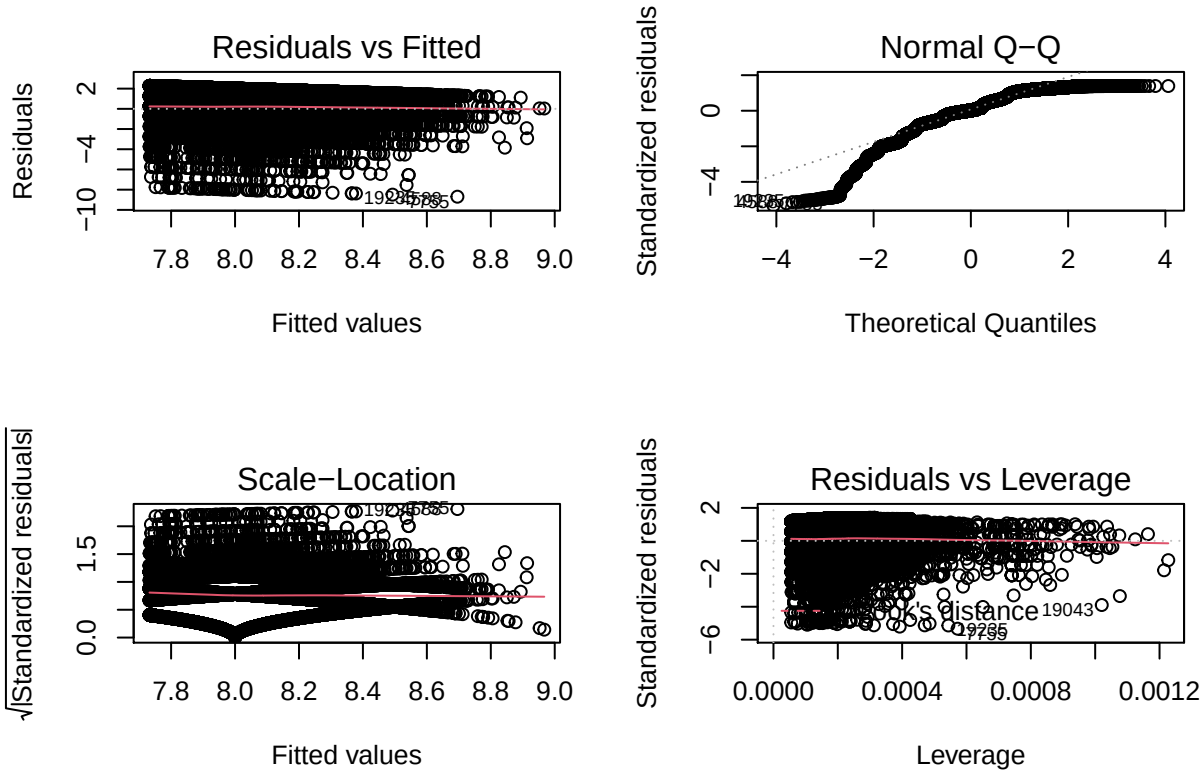
Figure2.3: Number of Children vs. Feeling of Life




```
mod <- lm(feelings_life ~ age + income_respondent + total_children, gss)
summary(mod)
```

```
##
## Call:
## lm(formula = feelings_life ~ age + income_respondent + total_children,
##     data = gss)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6956 -0.8790  0.0333  1.1291  2.2677
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       7.647e+00  3.899e-02 196.123   <2e-16 ***
## age               1.559e-03  7.209e-04   2.162   0.0306 *
## income_respondent 4.987e-06  3.511e-07  14.202   <2e-16 ***
## total_children    8.223e-02  8.601e-03   9.561   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.631 on 20308 degrees of freedom
##   (290 observations deleted due to missingness)
## Multiple R-squared:  0.01705,    Adjusted R-squared:  0.01691
## F-statistic: 117.4 on 3 and 20308 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(mod)
```




References: - Baptiste Auguie (2017). gridExtra: Miscellaneous Functions for "Grid" Graphics. R package version 2.3. https://CRAN.R-project.org/package=gridExtra

- Wickham et al., (2019). Welcome to the tidyverse. Journal of Open Source Software, 4(43), 1686, https://doi.org/10.21105/joss.01686