Stat139_Final_Project

Brice Laurent, Linh Vu, Aissata Bah

2023-11-30

```
library(lme4)
```

```
## Loading required package: Matrix
```

```
# load data
data_clean <- read.csv("data/data_clean.csv")

# subset to 2021 data only
data_clean_2021 = data_clean[data_clean$year == 2021,]
df = data_clean_2021[2:138,]
```

```
# determine 1 representative variable from each of the 8 categories

# which(colnames(df) == "x1.1")
# which(colnames(df) == "x1.6")
cor(df[,40:45], use="na.or.complete")
```

```
##           x1.1      x1.2      x1.3      x1.4      x1.5      x1.6
## x1.1 1.0000000 0.8535752 0.7725164 0.6874921 0.8916864 0.8535429
## x1.2 0.8535752 1.0000000 0.7831357 0.8129867 0.8316357 0.8696162
## x1.3 0.7725164 0.7831357 1.0000000 0.8726919 0.7456900 0.8036759
## x1.4 0.6874921 0.8129867 0.8726919 1.0000000 0.6351786 0.7352018
## x1.5 0.8916864 0.8316357 0.7456900 0.6351786 1.0000000 0.9116563
## x1.6 0.8535429 0.8696162 0.8036759 0.7352018 0.9116563 1.0000000
```

```
# which(colnames(df) == "x2.1")
# which(colnames(df) == "x2.4")
cor(df[,47:50], use="na.or.complete")
```

```
##           x2.1      x2.2      x2.3      x2.4
## x2.1 1.0000000 0.8451849 0.8934019 0.8798892
## x2.2 0.8451849 1.0000000 0.9091039 0.6592868
## x2.3 0.8934019 0.9091039 1.0000000 0.7134683
## x2.4 0.8798892 0.6592868 0.7134683 1.0000000
```

```
# which(colnames(df) == "x3.1")
# which(colnames(df) == "x3.4")
cor(df[,52:55], use="na.or.complete")
```

```
##           x3.1      x3.2      x3.3      x3.4
## x3.1 1.0000000 0.7913838 0.6501569 0.7443906
## x3.2 0.7913838 1.0000000 0.7808461 0.8067962
## x3.3 0.6501569 0.7808461 1.0000000 0.8224281
## x3.4 0.7443906 0.8067962 0.8224281 1.0000000
```

```
# which(colnames(df) == "x4.1")
# which(colnames(df) == "x4.8")
cor(df[,57:64], use="na.or.complete")
```

```
##           x4.1      x4.2      x4.3      x4.4      x4.5      x4.6      x4.7
## x4.1 1.0000000 0.7403293 0.7950179 0.6012060 0.5975404 0.6795467 0.5884560
## x4.2 0.7403293 1.0000000 0.8939160 0.8361741 0.6590902 0.9360924 0.8053152
## x4.3 0.7950179 0.8939160 1.0000000 0.7294732 0.5877435 0.8731109 0.6906743
## x4.4 0.6012060 0.8361741 0.7294732 1.0000000 0.7583159 0.8640336 0.9631803
## x4.5 0.5975404 0.6590902 0.5877435 0.7583159 1.0000000 0.6519960 0.7979951
## x4.6 0.6795467 0.9360924 0.8731109 0.8640336 0.6519960 1.0000000 0.8303109
## x4.7 0.5884560 0.8053152 0.6906743 0.9631803 0.7979951 0.8303109 1.0000000
## x4.8 0.8045813 0.7836435 0.7703981 0.7475991 0.6666523 0.7661089 0.7528034
##           x4.8
## x4.1 0.8045813
## x4.2 0.7836435
## x4.3 0.7703981
## x4.4 0.7475991
## x4.5 0.6666523
## x4.6 0.7661089
## x4.7 0.7528034
## x4.8 1.0000000
```

```
# which(colnames(df) == "x5.1")
# which(colnames(df) == "x5.3")
cor(df[,66:68], use="na.or.complete")
```

```
##           x5.1      x5.2      x5.3
## x5.1 1.0000000 0.2311982 0.6927800
## x5.2 0.2311982 1.0000000 0.1809592
## x5.3 0.6927800 0.1809592 1.0000000
```

```
# which(colnames(df) == "x6.1")
# which(colnames(df) == "x6.5")
cor(df[,70:74], use="na.or.complete")
```

```
##           x6.1      x6.2      x6.3      x6.4      x6.5
## x6.1 1.0000000 0.8385560 0.7958826 0.7866191 0.7632958
## x6.2 0.8385560 1.0000000 0.7271352 0.8103286 0.8140194
## x6.3 0.7958826 0.7271352 1.0000000 0.7362246 0.6553697
## x6.4 0.7866191 0.8103286 0.7362246 1.0000000 0.8666779
## x6.5 0.7632958 0.8140194 0.6553697 0.8666779 1.0000000
```

```
# less correlation
# which(colnames(df) == "x7.1")
# which(colnames(df) == "x7.7")
cor(df[,76:82], use="na.or.complete")
```

```
##           x7.1      x7.2      x7.3      x7.4      x7.5      x7.6      x7.7
## x7.1 1.0000000 0.7674158 0.7203611 0.6554683 0.3707981 0.5668955 0.7346043
## x7.2 0.7674158 1.0000000 0.6913422 0.6759848 0.4405019 0.5951050 0.6719103
## x7.3 0.7203611 0.6913422 1.0000000 0.8775565 0.4062877 0.6457727 0.7828699
## x7.4 0.6554683 0.6759848 0.8775565 1.0000000 0.3674719 0.6362278 0.7588538
## x7.5 0.3707981 0.4405019 0.4062877 0.3674719 1.0000000 0.6911407 0.4360977
## x7.6 0.5668955 0.5951050 0.6457727 0.6362278 0.6911407 1.0000000 0.6547320
## x7.7 0.7346043 0.6719103 0.7828699 0.7588538 0.4360977 0.6547320 1.0000000
```

```
# which(colnames(df) == "x8.1")
# which(colnames(df) == "x8.6")
cor(df[,84:89], use="na.or.complete")
```

```
##           x8.1      x8.2      x8.3      x8.4      x8.5      x8.6
## x8.1 1.0000000 0.8162887 0.8056604 0.6161842 0.8067364 0.6323565
## x8.2 0.8162887 1.0000000 0.8458627 0.7035704 0.7662960 0.6258329
## x8.3 0.8056604 0.8458627 1.0000000 0.7157188 0.7794454 0.6409182
## x8.4 0.6161842 0.7035704 0.7157188 1.0000000 0.7033721 0.6646307
## x8.5 0.8067364 0.7662960 0.7794454 0.7033721 1.0000000 0.8087741
## x8.6 0.6323565 0.6258329 0.6409182 0.6646307 0.8087741 1.0000000
```

```
# subset data
cor(df[, c("x1.6", "x2.1", "x3.2", "x4.2", "x5.2", "x5.3", "x6.4", "x7.5", "x7.3", "x8.5")], use="na.or
```

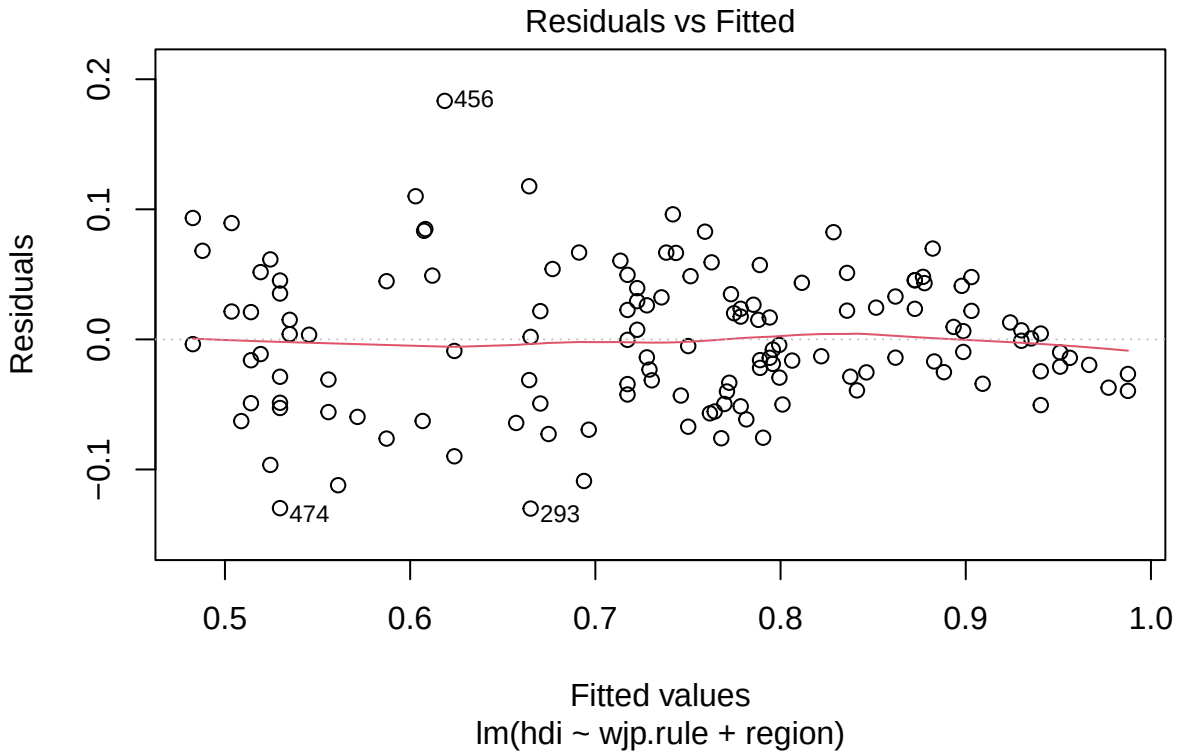
```
##           x1.6      x2.1      x3.2      x4.2      x5.2      x5.3      x6.4
## x1.6 1.0000000 0.7405120 0.7528504 0.8269984 0.2125287 0.4957072 0.7957523
## x2.1 0.7405120 1.0000000 0.7365611 0.7354224 0.2404047 0.7411529 0.8436650
## x3.2 0.7528504 0.7365611 1.0000000 0.7284214 0.2364262 0.5050670 0.7416231
## x4.2 0.8269984 0.7354224 0.7284214 1.0000000 0.4260745 0.5726462 0.7318598
## x5.2 0.2125287 0.2404047 0.2364262 0.4260745 1.0000000 0.1809592 0.2066998
## x5.3 0.4957072 0.7411529 0.5050670 0.5726462 0.1809592 1.0000000 0.6366518
## x6.4 0.7957523 0.8436650 0.7416231 0.7318598 0.2066998 0.6366518 1.0000000
## x7.5 0.2683288 0.5619566 0.3344774 0.3480453 0.1731546 0.6485681 0.5072420
## x7.3 0.7257268 0.8498694 0.6562820 0.7773149 0.2938412 0.6299382 0.7977201
## x8.5 0.7282903 0.9127606 0.6767124 0.7804367 0.2876731 0.6741872 0.7972382
##           x7.5      x7.3      x8.5
## x1.6 0.2683288 0.7257268 0.7282903
## x2.1 0.5619566 0.8498694 0.9127606
## x3.2 0.3344774 0.6562820 0.6767124
## x4.2 0.3480453 0.7773149 0.7804367
## x5.2 0.1731546 0.2938412 0.2876731
## x5.3 0.6485681 0.6299382 0.6741872
## x6.4 0.5072420 0.7977201 0.7972382
## x7.5 1.0000000 0.4062877 0.4474595
## x7.3 0.4062877 1.0000000 0.9464576
## x8.5 0.4474595 0.9464576 1.0000000
```

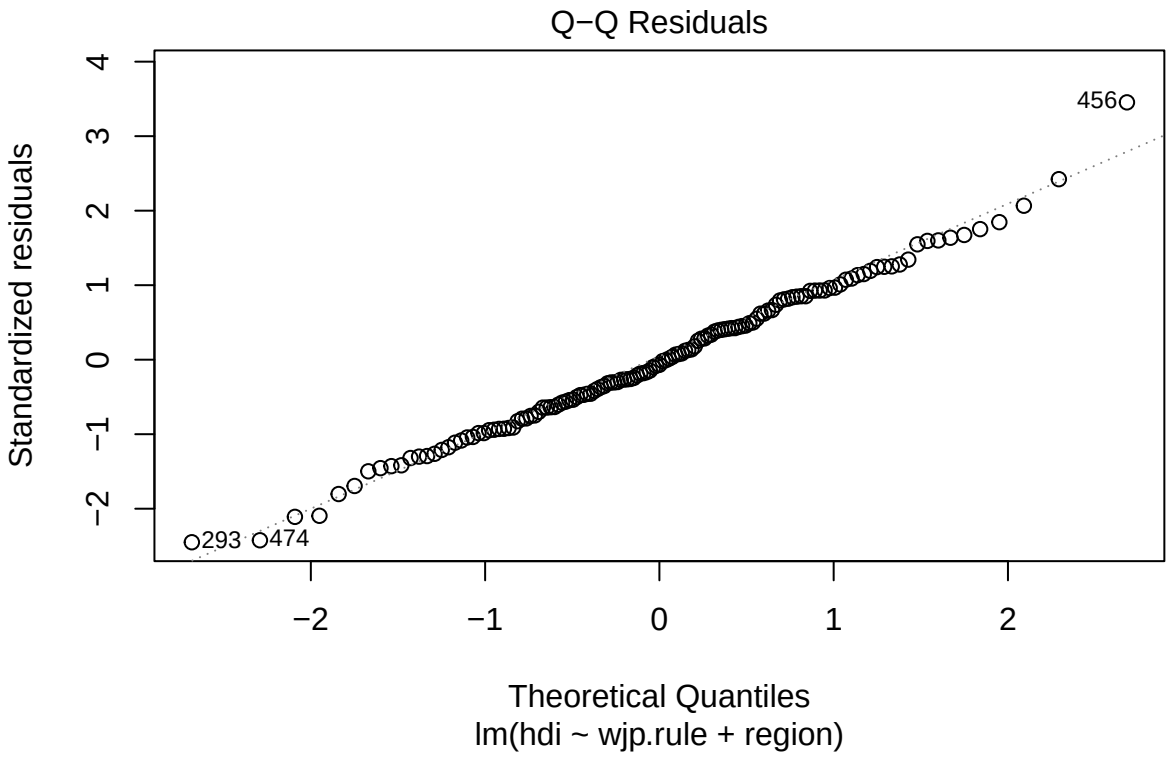
```
# chosen variables
cor(df[,c("x3.2", "x5.2", "x5.3", "x7.5", "x8.5")], use="na.or.complete")
```

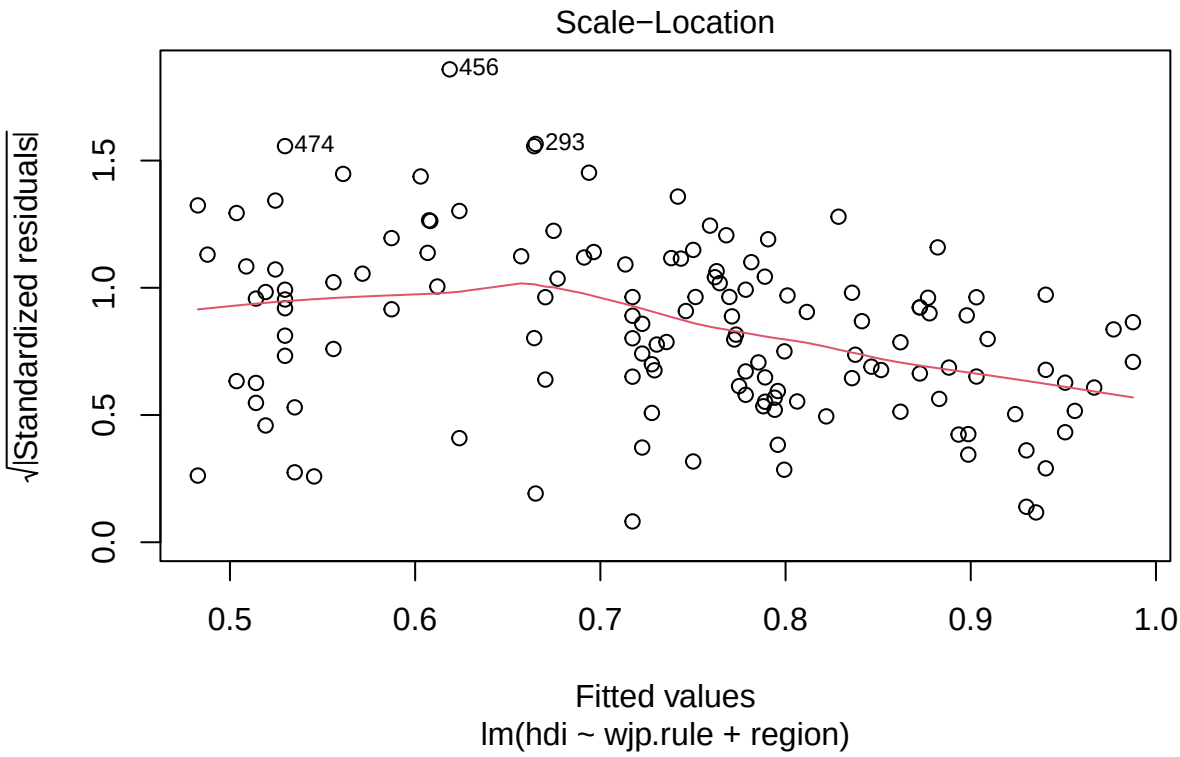
```
##           x3.2      x5.2      x5.3      x7.5      x8.5
## x3.2 1.0000000 0.2364262 0.5050670 0.3344774 0.6767124
```

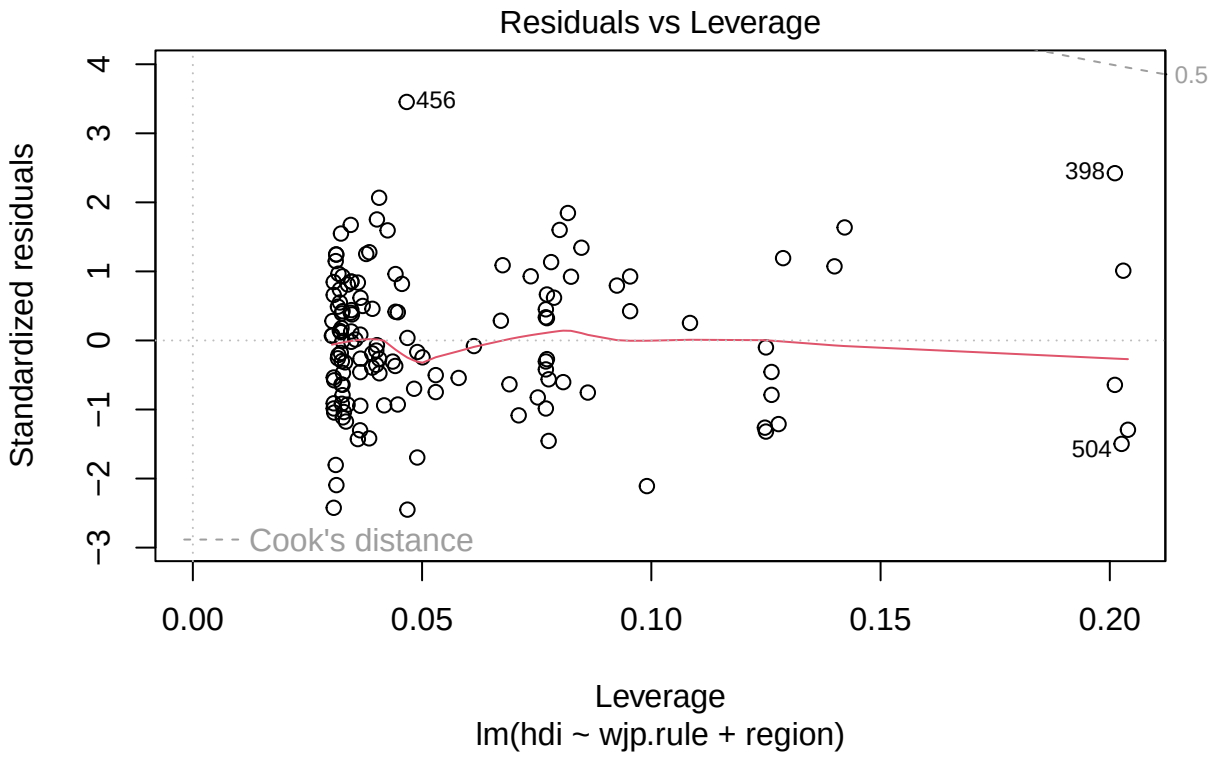
```
## x5.2 0.2364262 1.0000000 0.1809592 0.1731546 0.2876731
## x5.3 0.5050670 0.1809592 1.0000000 0.6485681 0.6741872
## x7.5 0.3344774 0.1731546 0.6485681 1.0000000 0.4474595
## x8.5 0.6767124 0.2876731 0.6741872 0.4474595 1.0000000
```

```r
#baseline using overall ROL as predictors
model0 = lm(hdi ~ wjp.rule + region, data = df)
plot(model0)
```

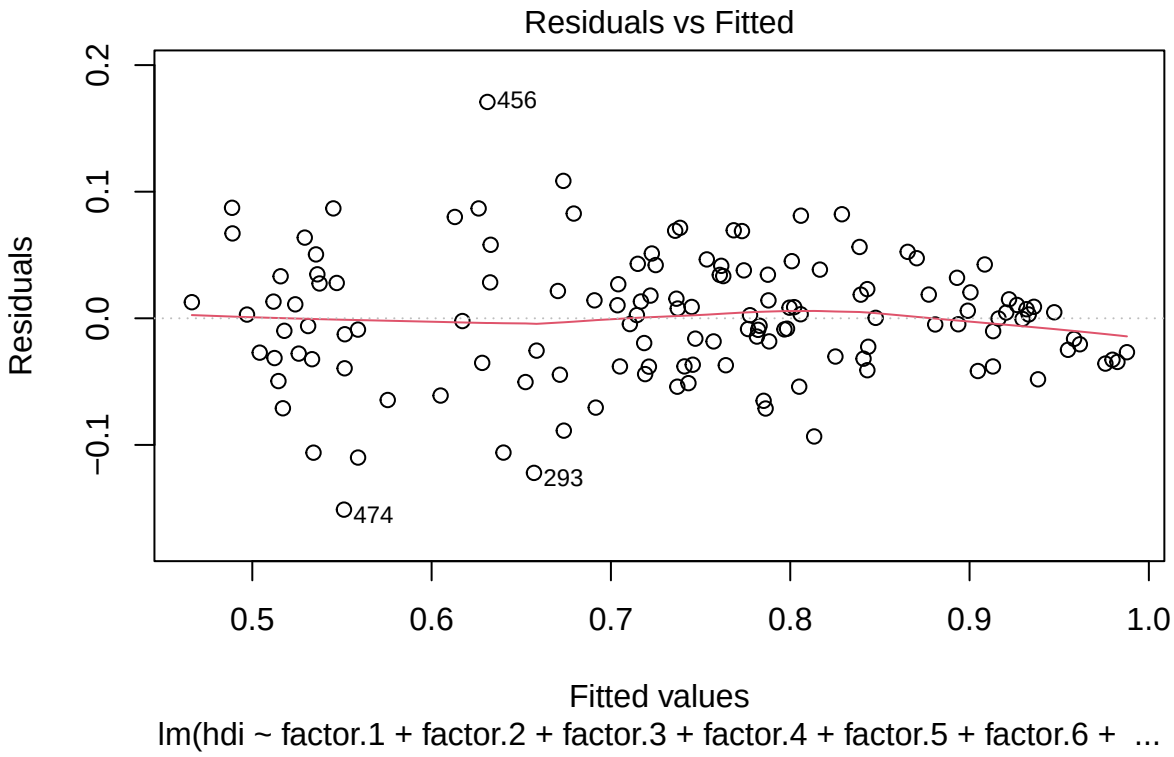


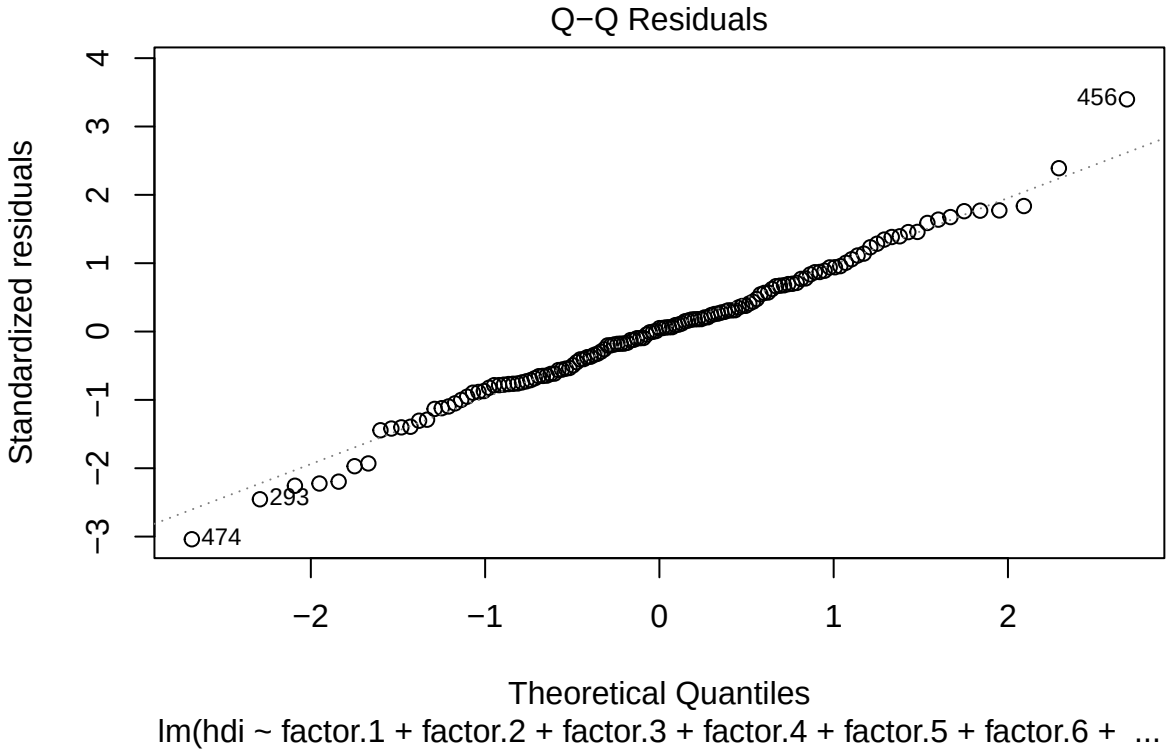

Q−Q Residuals
Standardized residuals
−2
−1
0
1
2
3
4
456
293
474
−2
−1
0
1
2
Theoretical Quantiles
lm(hdi ~ wjp.rule + region)

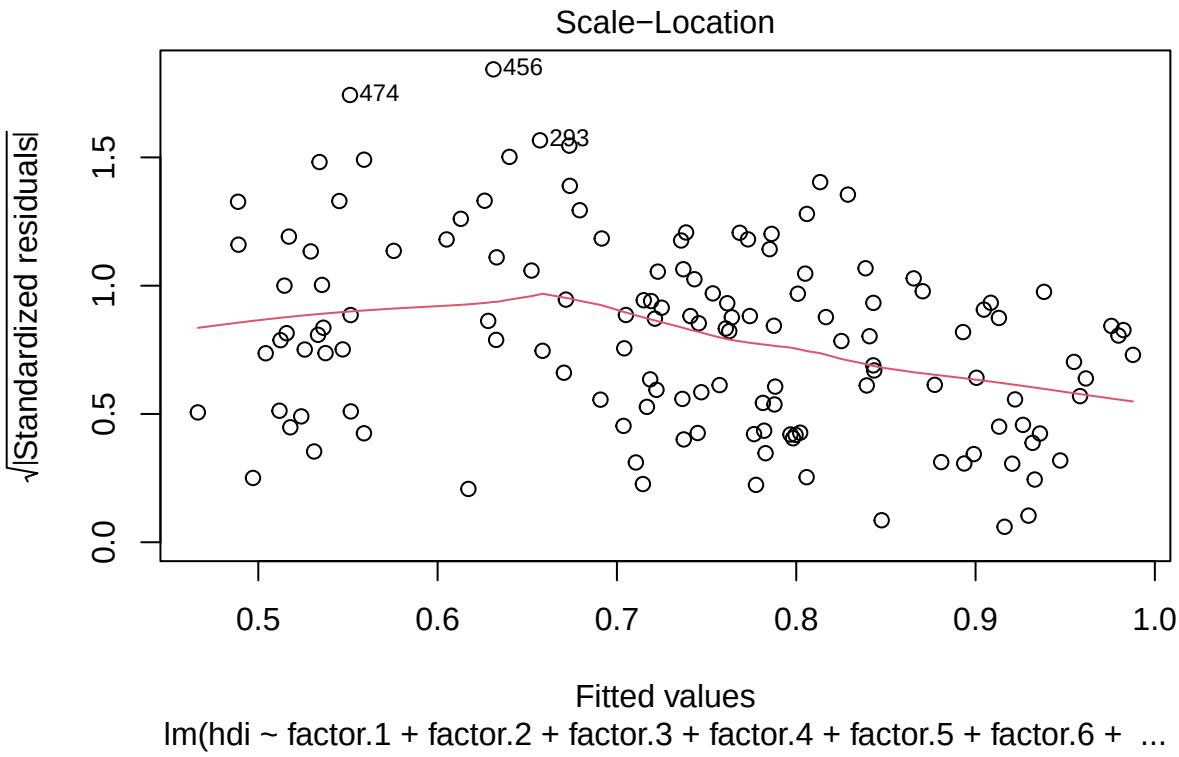

Scale−Location
456
474
293
√|Standardized residuals|
0.0
0.5
1.0
1.5
0.5
0.6
0.7
0.8
0.9
1.0
Fitted values
lm(hdi ~ wjp.rule + region)

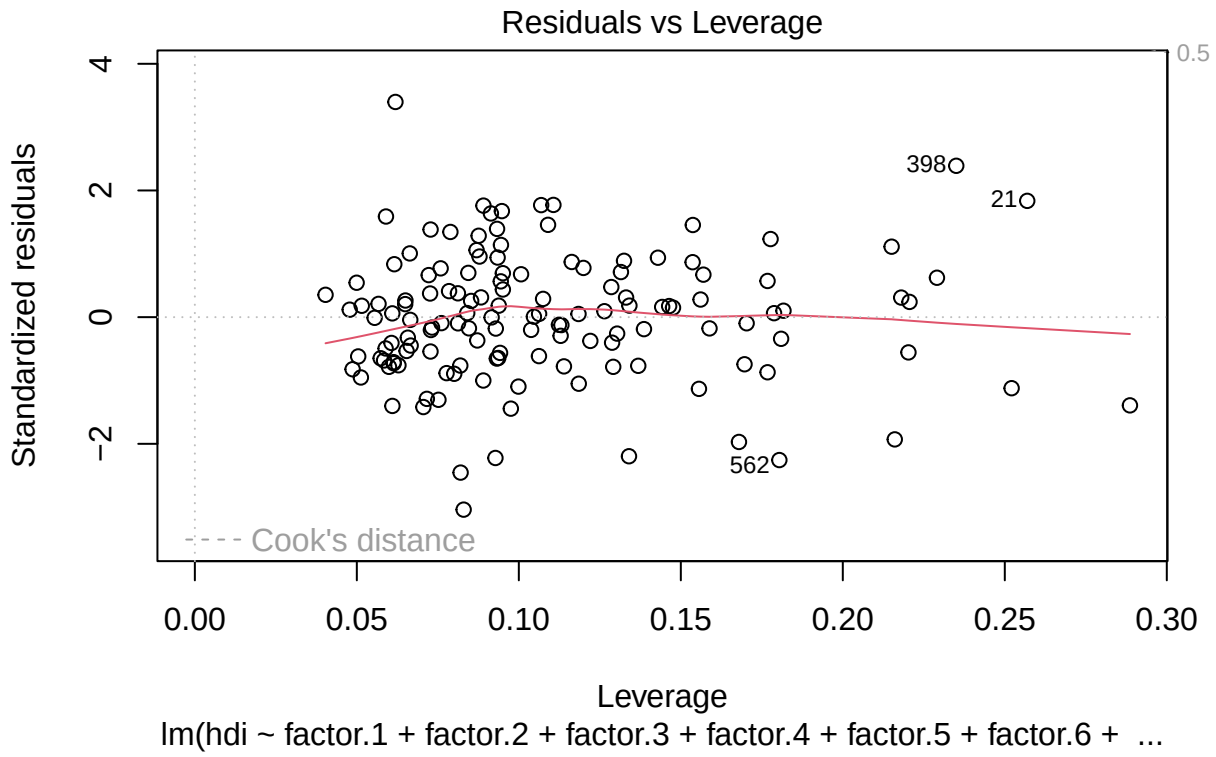



```
# using the individual categorical averages as predictors
model1 = lm(hdi~factor.1 +factor.2 +factor.3 +factor.4 +factor.5 +factor.6 +factor.7 + factor.8 + regio
plot(model1)
```


Residuals vs Fitted
Residuals
0.2
0.1
0.0
−0.1
456
293
474
0.5
0.6
0.7
0.8
0.9
1.0
Fitted values
lm(hdi ~ factor.1 + factor.2 + factor.3 + factor.4 + factor.5 + factor.6 + ...


Q−Q Residuals
Standardized residuals
Theoretical Quantiles
lm(hdi ~ factor.1 + factor.2 + factor.3 + factor.4 + factor.5 + factor.6 + ...
456
293
474
−3
−2
−1
0
1
2
3
4
−2
−1
0
1
2


Scale−Location
456
474
283
√|Standardized residuals|
0.0
0.5
1.0
1.5
0.5
0.6
0.7
0.8
0.9
1.0
Fitted values
lm(hdi ~ factor.1 + factor.2 + factor.3 + factor.4 + factor.5 + factor.6 + ...




```
# random intercept and slope model
model2 = lmer(hdi ~ wjp.rule + (1 + wjp.rule || region), data = df)

model3 = lmer(hdi~factor.1 +factor.2 +factor.3 +factor.4+factor.5 +factor.6 +factor.7 + factor.8 + (1 +
```

```
## boundary (singular) fit: see help('isSingular')
```

```
AIC(model0, model1, model2, model3)
```

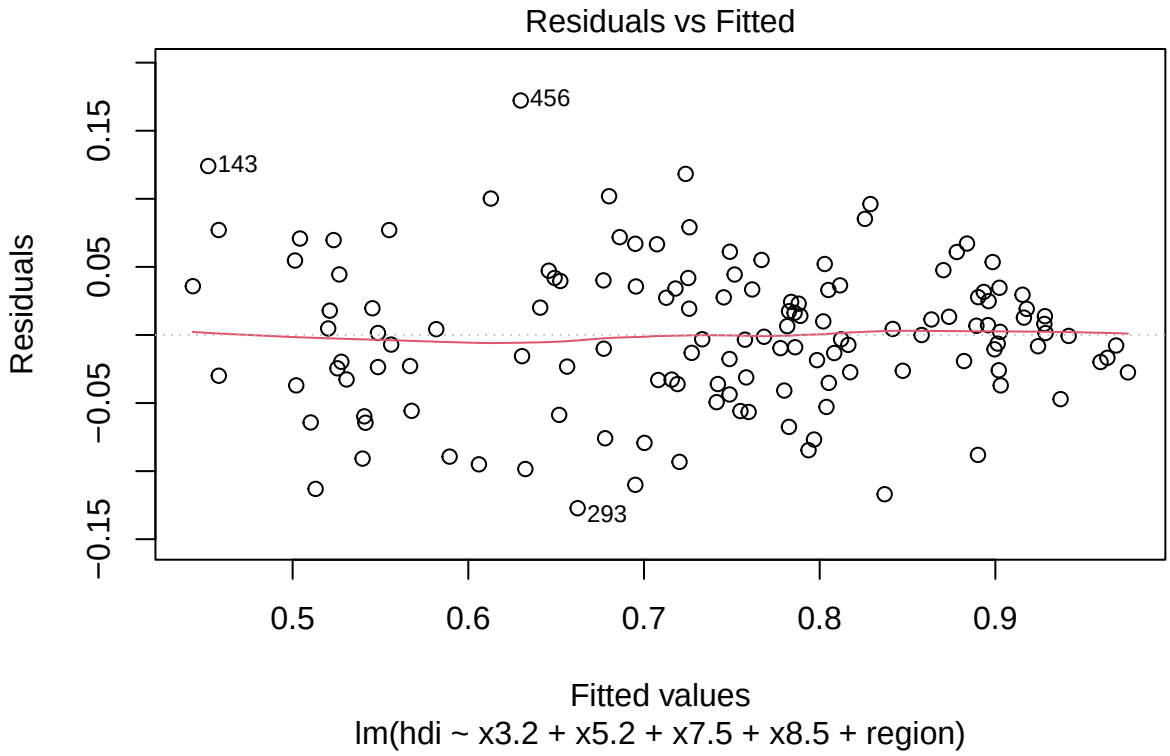
```
##        df       AIC
## model0  9 -399.3265
## model1 16 -405.5451
## model2  5 -368.2625
## model3 19 -338.0480
```

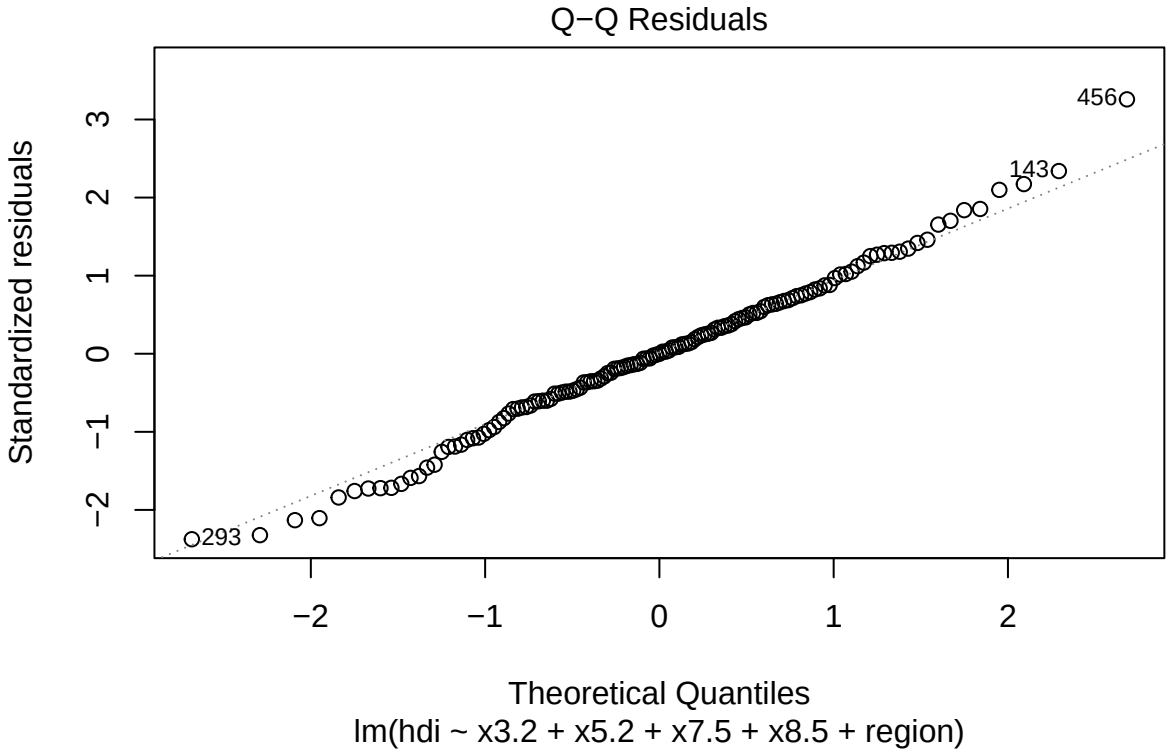
```
sub_df <- df[, c("country", "hdi", "region", "x3.2", "x5.2", "x7.5", "x8.5")]

summary(mod1 <- lm(hdi~x3.2 + x5.2 + x7.5 + x8.5 + region, data = sub_df))
```

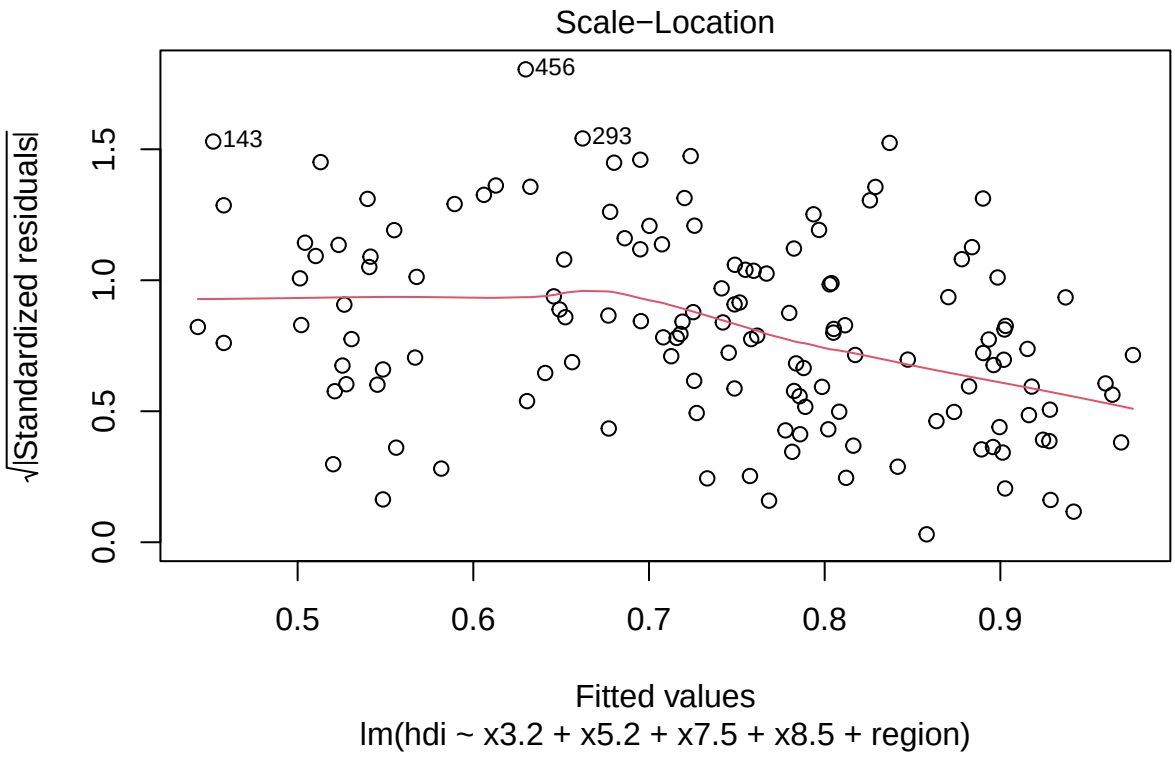
```
##
## Call:
## lm(formula = hdi ~ x3.2 + x5.2 + x7.5 + x8.5 + region, data = sub_df)
##
## Residuals:
```

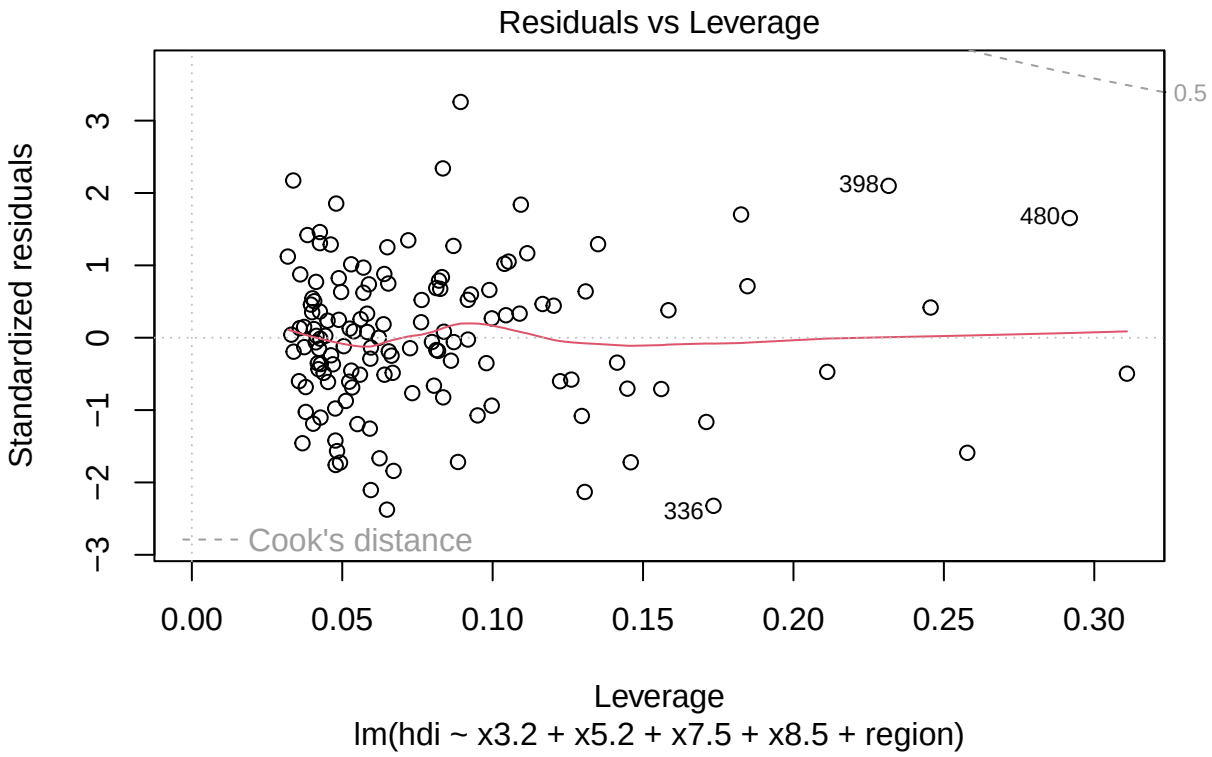
```
##       Min        1Q    Median        3Q       Max
## -0.127226 -0.032693 -0.000049  0.034111  0.172183
##
## Coefficients:
##                                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)                              0.506912   0.041299  12.274  < 2e-16 ***
## x3.2                                     0.106989   0.048354   2.213  0.02873 *
## x5.2                                     0.086219   0.036520   2.361  0.01976 *
## x7.5                                    -0.035743   0.035529  -1.006  0.31632
## x8.5                                     0.283789   0.037966   7.475 1.14e-11 ***
## regionEastern Europe & Central Asia      0.030991   0.021873   1.417  0.15899
## regionEU + EFTA + North America          0.040099   0.018989   2.112  0.03669 *
## regionLatin America & Caribbean         -0.044166   0.019879  -2.222  0.02808 *
## regionMiddle East & North Africa        -0.002925   0.025255  -0.116  0.90799
## regionSouth Asia                        -0.092780   0.031191  -2.975  0.00352 **
## regionSub-Saharan Africa                -0.186581   0.018840  -9.904  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05538 on 126 degrees of freedom
## Multiple R-squared:  0.8697, Adjusted R-squared:  0.8594
## F-statistic: 84.12 on 10 and 126 DF,  p-value: < 2.2e-16
```

```
plot(mod1)
```




Q−Q Residuals
Standardized residuals
−2
−1
0
1
2
3
456
143
293
−2
−1
0
1
2
Theoretical Quantiles
lm(hdi ~ x3.2 + x5.2 + x7.5 + x8.5 + region)


Scale−Location
456
143
293
√|Standardized residuals|
0.0
0.5
1.0
1.5
0.5
0.6
0.7
0.8
0.9
Fitted values
lm(hdi ~ x3.2 + x5.2 + x7.5 + x8.5 + region)


Residuals vs Leverage
Standardized residuals
Leverage
lm(hdi ~ x3.2 + x5.2 + x7.5 + x8.5 + region)
Cook's distance
0.5
398
480
336
−3
−2
−1
0
1
2
3
0.00
0.05
0.10
0.15
0.20
0.25
0.30