TD8

Charles Vin

2022-04-04

```
library(car)
```

```
## Loading required package: carData
```

```
library(emmeans)
```

Question 1

```
df = read.csv("./MiniPONS.csv", sep=";")
head(df)
```

```
##     Group  Type Age Right_answers Audio_prosody Combined_channel Face_video
## 1 Bipolar  BD I  47            40             9               11          9
## 2 Bipolar  BD I  49            49            13               13         11
## 3 Bipolar  BD I  45            43             9               11         13
## 4 Bipolar  BD I  53            44            10               10         12
## 5 Bipolar BD II  50            50            14               13         11
## 6 Bipolar  BD I  31            54            13               14         14
##   Body_video Positive_valence Negative_valence Dominant Submissive
## 1         11               18               22       23         17
## 2         12               24               25       24         25
## 3         10               21               22       24         19
## 4         12               25               19       24         20
## 5         12               23               27       23         27
## 6         13               28               26       26         28
```

Question 2

```
is.factor(df$Group)
```

```
## [1] FALSE
```

```
is.factor(df$Type)
```

```
## [1] FALSE
```

```
df$Group = as.factor(df$Group)
df$Type = as.factor(df$Type)
levels(df$Type)
```

```
## [1] "BD I"    "BD II"   "Control" "UD"
```

```
levels(df$Group)
```

```
## [1] "Bipolar" "Control" "UD"
```

Question 3

```
by(df$Right_answers, df$Group, summary )
```

```
## df$Group: Bipolar
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   28.00   42.50   46.00   45.35   49.00   56.00
## ------------------------------------------------------------
## df$Group: Control
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   39.00   48.00   50.00   50.24   53.00   58.00
## ------------------------------------------------------------
## df$Group: UD
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   32.00   39.00   43.00   42.72   47.00   53.00
```

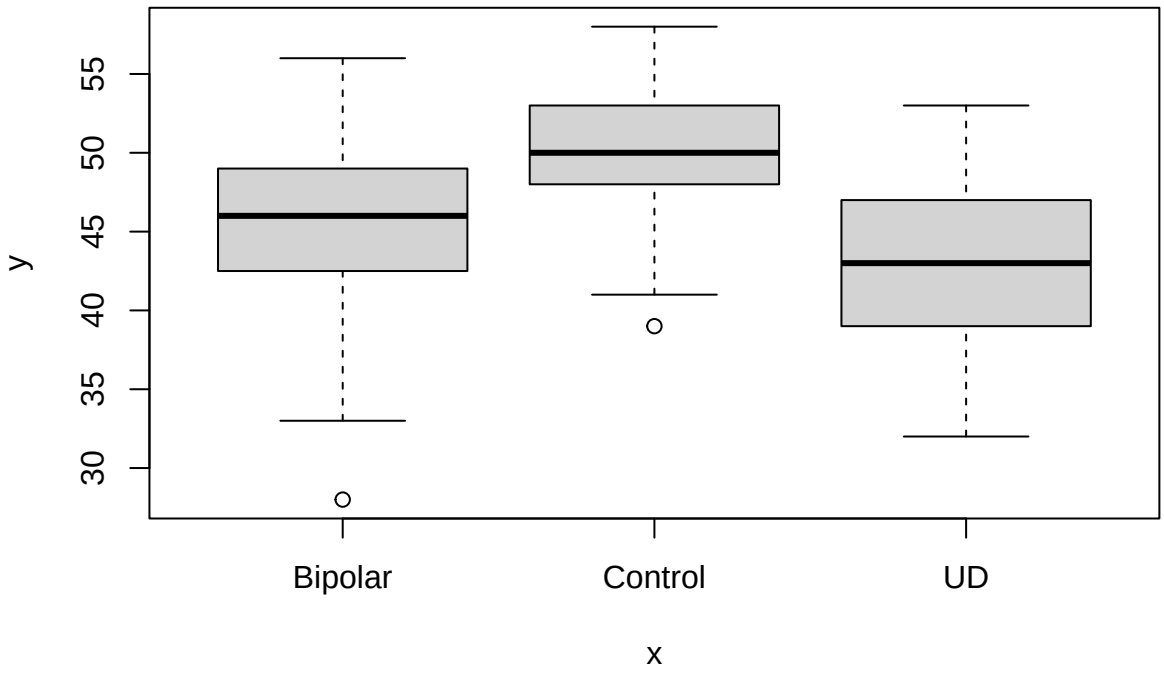
```
by(df$Right_answers, df$Group, sd )
```

```
## df$Group: Bipolar
## [1] 4.795104
## ------------------------------------------------------------
## df$Group: Control
## [1] 3.704569
## ------------------------------------------------------------
## df$Group: UD
## [1] 4.973343
```

On peut constater que les Bipolaire et les patients diagnostiqué UD ont en moyenne moins de bonne réponse que les sujets contrôles. De plus leurs écarts type est moins bon.

Question 4

```
plot(df$Group, df$Right_answers, type="b")
```



Question 5

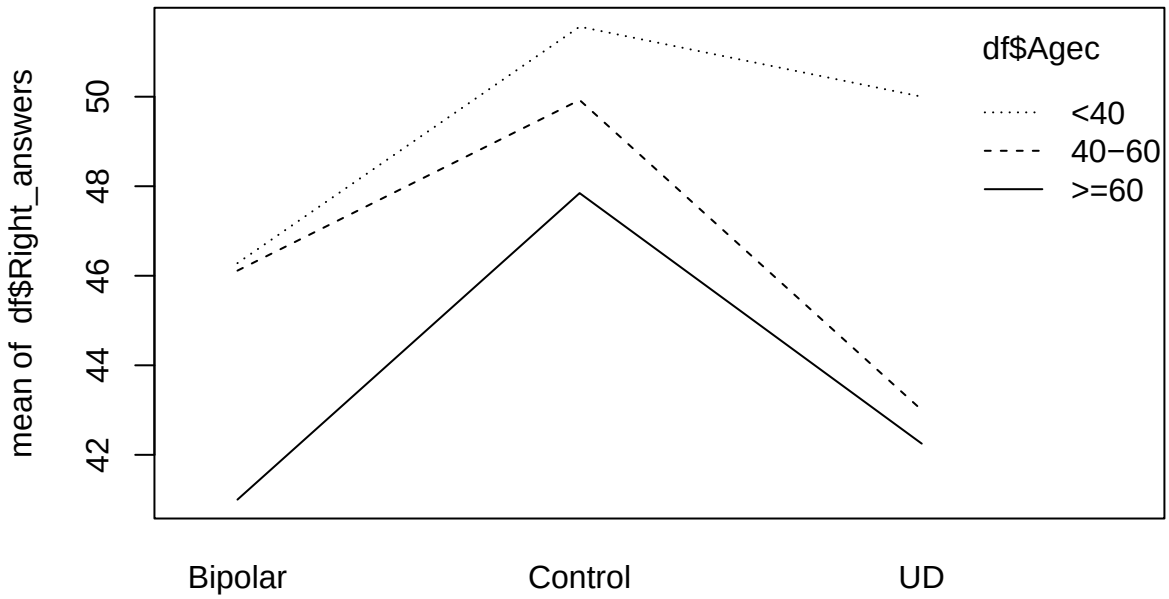
```
df$Agec=rep(2,length(df$Group))
df$Agec[which(df$Age<40)]=1
df$Agec[which(df$Age>=60)]=3
df$Agec=as.factor(df$Agec)
levels(df$Agec)=c("<40", "40-60", ">=60")
```

Question 6

```
table(df$Agec, df$Group)
```

```
##
##         Bipolar Control UD
##   <40        39      39  1
##   40-60      61      67 14
##   >=60       19      13 24
```

```
interaction.plot(df$Group, df$Agec, df$Right_answers)
```



Les effectifs ne sont pas vraiment équilibré dans avec le groupe UD.

Ils pourrait avoir de l'interaction. Considérons le plus grand des modèles possibles donc avec interaction. Puis on vas regarder si le les interraction valent le coup. On vas comparer le modèle avec interaction à celui sans interaction. Si le modèle sans intéraction est mieux, on regardera ensuite si chaque facteur vaut le coup.

Question 7

Modèle retenu

```
res = lm(df$Right_answers~df$Group*df$Agec)
res
```

```
##
## Call:
## lm(formula = df$Right_answers ~ df$Group * df$Agec)
##
## Coefficients:
##                   (Intercept)                df$GroupControl
##                       46.2821                        5.2821
##                    df$GroupUD                   df$Agec40-60
##                        3.7179                       -0.1673
##                   df$Agec>=60  df$GroupControl:df$Agec40-60
##                        -5.2821                       -1.4714
##      df$GroupUD:df$Agec40-60   df$GroupControl:df$Agec>=60
##                        -6.8327                        1.5641
##       df$GroupUD:df$Agec>=60
##                        -2.4679
```

```
summary(res)
```

```
##
## Call:
## lm(formula = df$Right_answers ~ df$Group * df$Agec)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.0000  -2.8462   0.0746   2.7179  10.0000
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    46.2821     0.6659  69.504  < 2e-16 ***
## df$GroupControl                 5.2821     0.9417   5.609 5.06e-08 ***
## df$GroupUD                      3.7179     4.2115   0.883    0.378
## df$Agec40-60                   -0.1673     0.8526  -0.196    0.845
## df$Agec>=60                    -5.2821     1.1634  -4.540 8.50e-06 ***
## df$GroupControl:df$Agec40-60   -1.4714     1.1952  -1.231    0.219
## df$GroupUD:df$Agec40-60        -6.8327     4.3881  -1.557    0.121
## df$GroupControl:df$Agec>=60     1.5641     1.7684   0.884    0.377
## df$GroupUD:df$Agec>=60         -2.4679     4.4008  -0.561    0.575
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.158 on 268 degrees of freedom
## Multiple R-squared:  0.3865, Adjusted R-squared:  0.3682
## F-statistic: 21.11 on 8 and 268 DF,  p-value: < 2.2e-16
```

Le summary n'est pas très interessant. Car c'est des comparaisons étrange et difficile à interpréter.

Le modèle :

$$y_{i,j,k} = \mu + \alpha_i + \beta_j + \gamma_{i,j} + \epsilon_{i,j,k}$$

Avec i= Groupe, j=groupe_age, k=individue. y = intercept + effet groupe + effet age + interaction + erreur.

Est-ce que tous les paramètres sont estimable ? Non le modèle est *****. Pour contrer ça, R ajoute des contraintes :

$\alpha_1 = 0, \alpha_{bipolar} = 0 \beta_1 = 0, \beta_{<40} = 0 \gamma_{1,1} = 0, \gamma_{bipolar,<40} = 0 \gamma_{1,2} = 0, \gamma_{bipolar,40-60} = 0 \gamma_{1,3} = 0, \gamma_{bipolar,>60} = 0 \gamma_{1,1} = 0, \gamma_{bipola}$

Eviter de s'aventurer dans l'interpretation car les contraintes dépende du logiciel. Elle a inssisté sur le pas interpreter ça.

Néanmoins, on peut voir que le modèle est utile (p value tout en bas)

```
anova(res)
```

```
## Analysis of Variance Table
##
## Response: df$Right_answers
##                    Df Sum Sq Mean Sq F value    Pr(>F)
## df$Group            2 2281.8 1140.88 65.9740 < 2.2e-16 ***
## df$Agec             2  491.1  245.56 14.1999 1.375e-06 ***
## df$Group:df$Agec    4  146.9   36.72  2.1232   0.07823 .
## Residuals         268 4634.5   17.29
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nous donne les tables des somme des effet de type 1 (car anova sans majuscule). On vas comparer les modèles dans l'ordre ou on a introduit les variables.

Ligne 4 : on test l'interaction : H_0 = sans interaction vs H_1 avec interaction. Dans notre cas avec erreur 5%, on choisis le modèle sans interaction

Ligne 3 : On test H_0 = modèle groupe VS H_1=sans interaction mais avec les deux facteur (additif) modèle age+groupe. On choisis l'age+groupe.

Ligne 2 : H_0 = Constrante VS H_0 = constante + groupe

```
Anova(res)
```

```
## Anova Table (Type II tests)
##
## Response: df$Right_answers
##                   Sum Sq  Df F value    Pr(>F)
## df$Group          1585.4   2 45.8382 < 2.2e-16 ***
## df$Agec            491.1   2 14.1999 1.375e-06 ***
## df$Group:df$Agec   146.9   4  2.1232   0.07823 .
## Residuals         4634.5 268
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Si on fait une Anova avec majuscucle

Ligne Groupe : H_0 Argg j'ai pas reussi

```
res2 = lm(df$Right_answers~df$Group+df$Agec)
res2
```

```
##
## Call:
## lm(formula = df$Right_answers ~ df$Group + df$Agec)
##
## Coefficients:
##     (Intercept)  df$GroupControl       df$GroupUD     df$Agec40-60
##         46.6137           4.7285          -0.8968          -1.1507
##     df$Agec>=60
##         -4.2021
```

```
summary(res2)
```

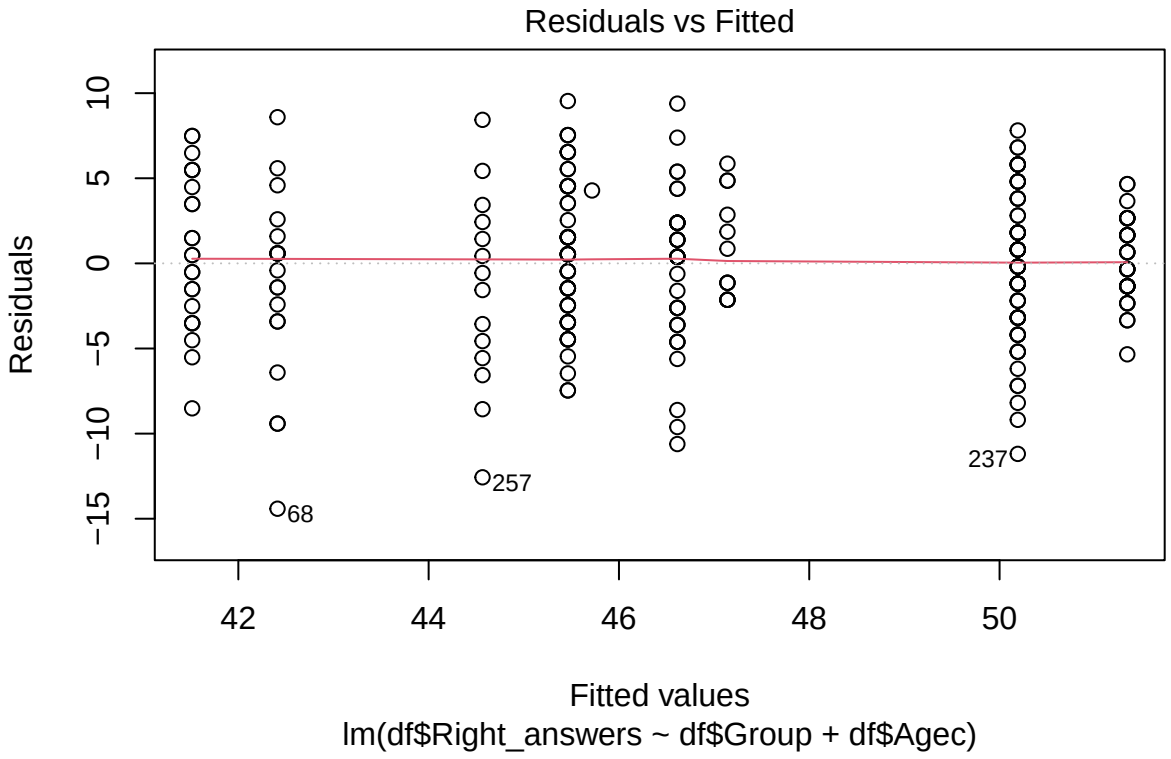
```
##
## Call:
## lm(formula = df$Right_answers ~ df$Group + df$Agec)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.4117  -2.4630   0.3863   2.6578   9.5370
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      46.6137     0.5447  85.582  < 2e-16 ***
## df$GroupControl   4.7285     0.5447   8.680 3.70e-16 ***
## df$GroupUD       -0.8968     0.8410  -1.066   0.2872
## df$Agec40-60     -1.1507     0.5926  -1.942   0.0532 .
## df$Agec>=60      -4.2021     0.8057  -5.216 3.63e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.193 on 272 degrees of freedom
## Multiple R-squared:  0.3671, Adjusted R-squared:  0.3578
## F-statistic: 39.44 on 4 and 272 DF,  p-value: < 2.2e-16
```

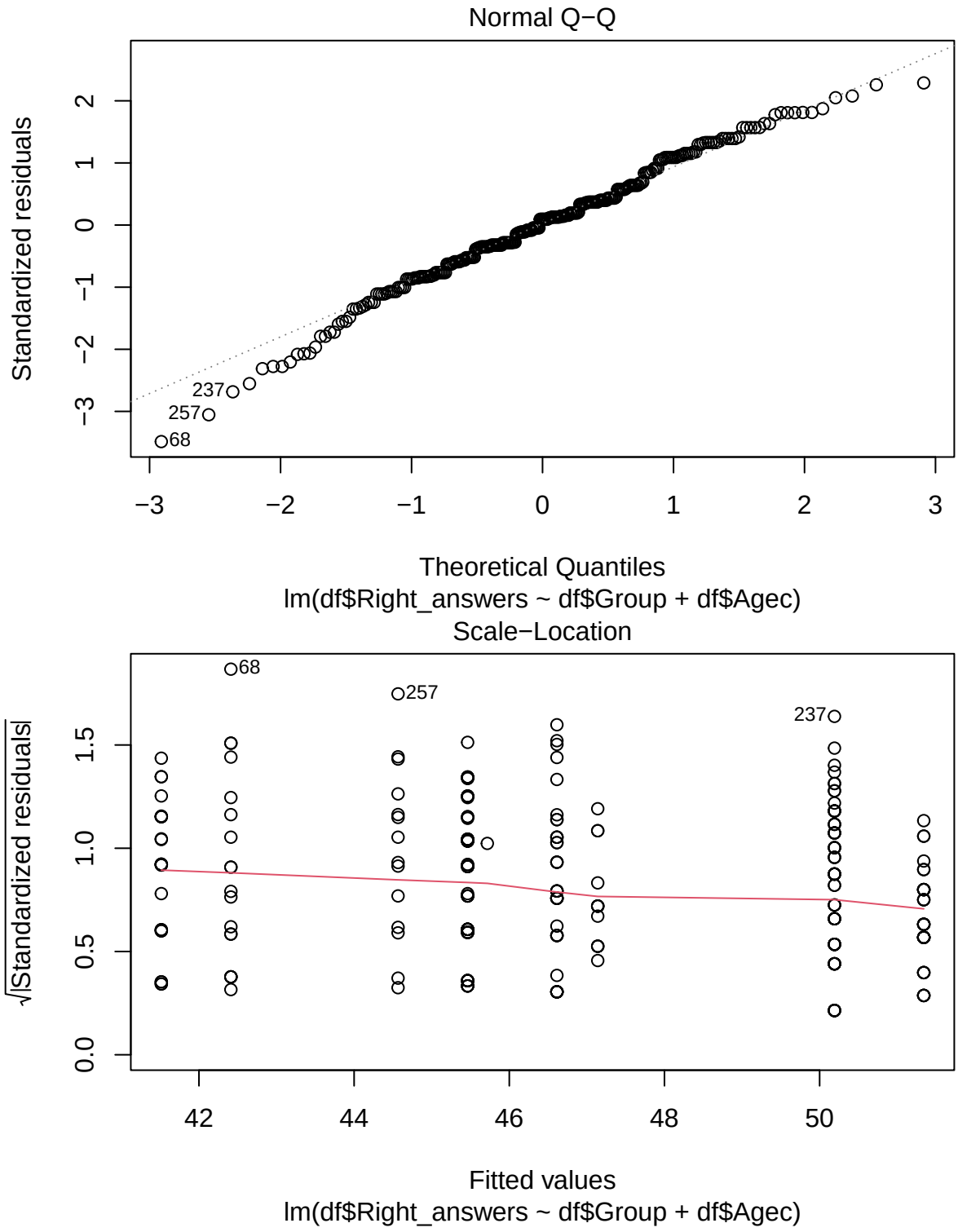
```
anova(res2)
```

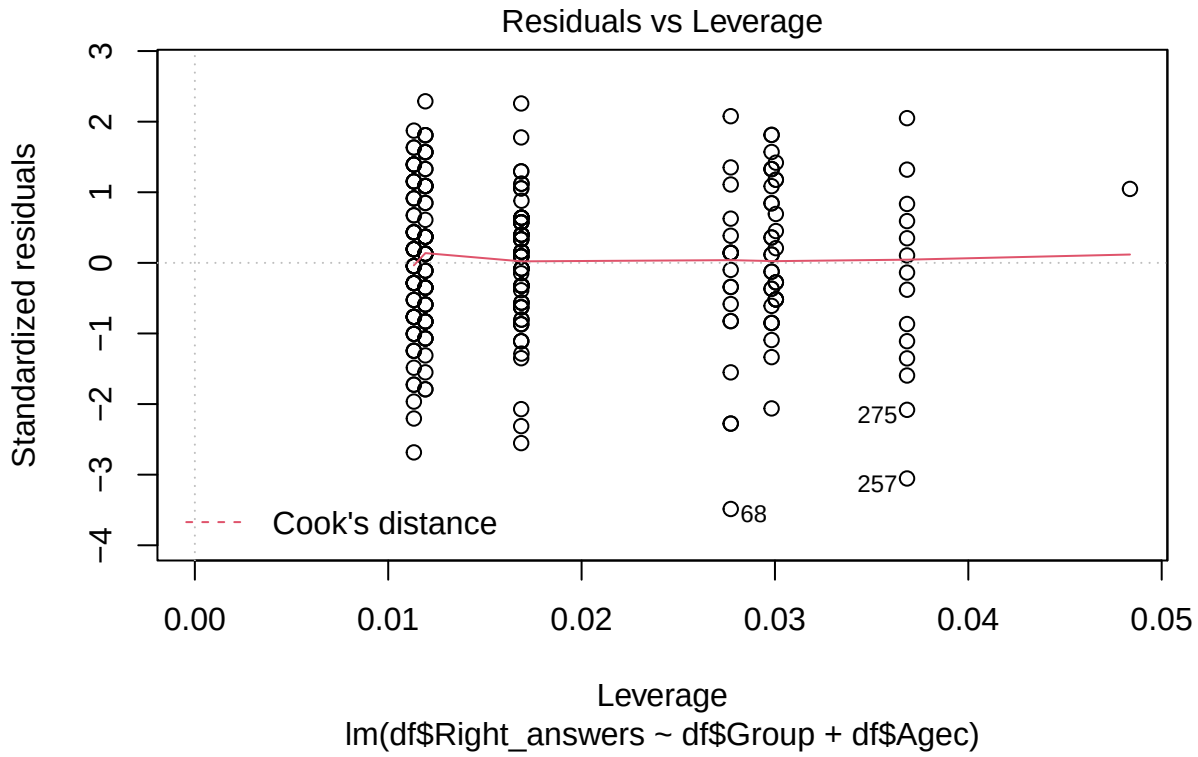
```
## Analysis of Variance Table
##
## Response: df$Right_answers
##            Df Sum Sq Mean Sq F value    Pr(>F)
## df$Group    2 2281.8 1140.88  64.902 < 2.2e-16 ***
## df$Agec     2  491.1  245.56  13.969 1.679e-06 ***
## Residuals 272 4781.4   17.58
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ici on voit un peu la même chose, surtout que le modèle est significatif

```
plot(res2)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(df$Right_answers ~ df$Group + df$Agec)
Scale−Location
√|Standardized residuals|
Fitted values
lm(df$Right_answers ~ df$Group + df$Agec)
68
257
237



Distance de cook parfaite sauf pour un point qui dépasse $ (p+1)/n = 2(i+j)/n = 0.043$. Ce points et le sujet qui était le seul dans son groupe, car il est seul il attire fortement les poids.

Q-plot ok décroche un peu sur les bord, variance ok

Question 8

```
?predict.lm
predict(res2, data.frame(Group="Control", Agec="<40"), interval="confidence") # IC à 95%
```

```
## Warning: 'newdata' had 1 row but variables found have 277 rows
```

```
##          fit      lwr      upr
## 1   45.46300 44.56182 46.36417
## 2   45.46300 44.56182 46.36417
## 3   45.46300 44.56182 46.36417
## 4   45.46300 44.56182 46.36417
## 5   45.46300 44.56182 46.36417
## 6   46.61374 45.54143 47.68604
## 7   45.46300 44.56182 46.36417
## 8   45.46300 44.56182 46.36417
## 9   45.46300 44.56182 46.36417
## 10  45.46300 44.56182 46.36417
## 11  45.46300 44.56182 46.36417
## 12  42.41166 41.03753 43.78579
## 13  45.46300 44.56182 46.36417
## 14  45.46300 44.56182 46.36417
## 15  45.46300 44.56182 46.36417
## 16  42.41166 41.03753 43.78579
## 17  45.46300 44.56182 46.36417
## 18  46.61374 45.54143 47.68604
```

```
## 19  46.61374 45.54143 47.68604
## 20  45.46300 44.56182 46.36417
## 21  46.61374 45.54143 47.68604
## 22  45.46300 44.56182 46.36417
## 23  45.46300 44.56182 46.36417
## 24  45.46300 44.56182 46.36417
## 25  45.46300 44.56182 46.36417
## 26  42.41166 41.03753 43.78579
## 27  45.46300 44.56182 46.36417
## 28  46.61374 45.54143 47.68604
## 29  45.46300 44.56182 46.36417
## 30  45.46300 44.56182 46.36417
## 31  42.41166 41.03753 43.78579
## 32  42.41166 41.03753 43.78579
## 33  45.46300 44.56182 46.36417
## 34  42.41166 41.03753 43.78579
## 35  45.46300 44.56182 46.36417
## 36  45.46300 44.56182 46.36417
## 37  45.46300 44.56182 46.36417
## 38  42.41166 41.03753 43.78579
## 39  45.46300 44.56182 46.36417
## 40  45.46300 44.56182 46.36417
## 41  45.46300 44.56182 46.36417
## 42  45.46300 44.56182 46.36417
## 43  45.46300 44.56182 46.36417
## 44  46.61374 45.54143 47.68604
## 45  45.46300 44.56182 46.36417
## 46  42.41166 41.03753 43.78579
## 47  46.61374 45.54143 47.68604
## 48  45.46300 44.56182 46.36417
## 49  45.46300 44.56182 46.36417
## 50  45.46300 44.56182 46.36417
## 51  42.41166 41.03753 43.78579
## 52  46.61374 45.54143 47.68604
## 53  46.61374 45.54143 47.68604
## 54  45.46300 44.56182 46.36417
## 55  46.61374 45.54143 47.68604
## 56  45.46300 44.56182 46.36417
## 57  45.46300 44.56182 46.36417
## 58  45.46300 44.56182 46.36417
## 59  45.46300 44.56182 46.36417
## 60  45.46300 44.56182 46.36417
## 61  42.41166 41.03753 43.78579
## 62  42.41166 41.03753 43.78579
## 63  45.46300 44.56182 46.36417
## 64  46.61374 45.54143 47.68604
## 65  45.46300 44.56182 46.36417
## 66  46.61374 45.54143 47.68604
## 67  45.46300 44.56182 46.36417
## 68  42.41166 41.03753 43.78579
## 69  46.61374 45.54143 47.68604
## 70  46.61374 45.54143 47.68604
## 71  45.46300 44.56182 46.36417
## 72  46.61374 45.54143 47.68604
```

```
## 73  46.61374 45.54143 47.68604
## 74  46.61374 45.54143 47.68604
## 75  45.46300 44.56182 46.36417
## 76  45.46300 44.56182 46.36417
## 77  45.46300 44.56182 46.36417
## 78  45.46300 44.56182 46.36417
## 79  45.46300 44.56182 46.36417
## 80  45.46300 44.56182 46.36417
## 81  42.41166 41.03753 43.78579
## 82  46.61374 45.54143 47.68604
## 83  46.61374 45.54143 47.68604
## 84  46.61374 45.54143 47.68604
## 85  45.46300 44.56182 46.36417
## 86  45.46300 44.56182 46.36417
## 87  45.46300 44.56182 46.36417
## 88  46.61374 45.54143 47.68604
## 89  46.61374 45.54143 47.68604
## 90  45.46300 44.56182 46.36417
## 91  46.61374 45.54143 47.68604
## 92  46.61374 45.54143 47.68604
## 93  42.41166 41.03753 43.78579
## 94  46.61374 45.54143 47.68604
## 95  46.61374 45.54143 47.68604
## 96  46.61374 45.54143 47.68604
## 97  42.41166 41.03753 43.78579
## 98  45.46300 44.56182 46.36417
## 99  42.41166 41.03753 43.78579
## 100 46.61374 45.54143 47.68604
## 101 45.46300 44.56182 46.36417
## 102 42.41166 41.03753 43.78579
## 103 46.61374 45.54143 47.68604
## 104 46.61374 45.54143 47.68604
## 105 46.61374 45.54143 47.68604
## 106 45.46300 44.56182 46.36417
## 107 46.61374 45.54143 47.68604
## 108 46.61374 45.54143 47.68604
## 109 46.61374 45.54143 47.68604
## 110 42.41166 41.03753 43.78579
## 111 45.46300 44.56182 46.36417
## 112 46.61374 45.54143 47.68604
## 113 46.61374 45.54143 47.68604
## 114 46.61374 45.54143 47.68604
## 115 46.61374 45.54143 47.68604
## 116 45.46300 44.56182 46.36417
## 117 45.46300 44.56182 46.36417
## 118 42.41166 41.03753 43.78579
## 119 46.61374 45.54143 47.68604
## 120 50.19150 49.31302 51.06999
## 121 50.19150 49.31302 51.06999
## 122 50.19150 49.31302 51.06999
## 123 50.19150 49.31302 51.06999
## 124 50.19150 49.31302 51.06999
## 125 51.34224 50.26945 52.41503
## 126 50.19150 49.31302 51.06999
```

```
## 127 50.19150 49.31302 51.06999
## 128 50.19150 49.31302 51.06999
## 129 50.19150 49.31302 51.06999
## 130 50.19150 49.31302 51.06999
## 131 47.14016 45.70937 48.57096
## 132 50.19150 49.31302 51.06999
## 133 50.19150 49.31302 51.06999
## 134 50.19150 49.31302 51.06999
## 135 47.14016 45.70937 48.57096
## 136 50.19150 49.31302 51.06999
## 137 51.34224 50.26945 52.41503
## 138 51.34224 50.26945 52.41503
## 139 50.19150 49.31302 51.06999
## 140 51.34224 50.26945 52.41503
## 141 50.19150 49.31302 51.06999
## 142 50.19150 49.31302 51.06999
## 143 50.19150 49.31302 51.06999
## 144 50.19150 49.31302 51.06999
## 145 47.14016 45.70937 48.57096
## 146 50.19150 49.31302 51.06999
## 147 51.34224 50.26945 52.41503
## 148 50.19150 49.31302 51.06999
## 149 50.19150 49.31302 51.06999
## 150 47.14016 45.70937 48.57096
## 151 47.14016 45.70937 48.57096
## 152 50.19150 49.31302 51.06999
## 153 47.14016 45.70937 48.57096
## 154 50.19150 49.31302 51.06999
## 155 50.19150 49.31302 51.06999
## 156 50.19150 49.31302 51.06999
## 157 50.19150 49.31302 51.06999
## 158 50.19150 49.31302 51.06999
## 159 50.19150 49.31302 51.06999
## 160 50.19150 49.31302 51.06999
## 161 50.19150 49.31302 51.06999
## 162 50.19150 49.31302 51.06999
## 163 51.34224 50.26945 52.41503
## 164 50.19150 49.31302 51.06999
## 165 47.14016 45.70937 48.57096
## 166 51.34224 50.26945 52.41503
## 167 50.19150 49.31302 51.06999
## 168 50.19150 49.31302 51.06999
## 169 50.19150 49.31302 51.06999
## 170 47.14016 45.70937 48.57096
## 171 51.34224 50.26945 52.41503
## 172 51.34224 50.26945 52.41503
## 173 50.19150 49.31302 51.06999
## 174 51.34224 50.26945 52.41503
## 175 50.19150 49.31302 51.06999
## 176 50.19150 49.31302 51.06999
## 177 50.19150 49.31302 51.06999
## 178 50.19150 49.31302 51.06999
## 179 50.19150 49.31302 51.06999
## 180 50.19150 49.31302 51.06999
```

```
## 181 50.19150 49.31302 51.06999
## 182 50.19150 49.31302 51.06999
## 183 51.34224 50.26945 52.41503
## 184 50.19150 49.31302 51.06999
## 185 51.34224 50.26945 52.41503
## 186 50.19150 49.31302 51.06999
## 187 47.14016 45.70937 48.57096
## 188 51.34224 50.26945 52.41503
## 189 51.34224 50.26945 52.41503
## 190 50.19150 49.31302 51.06999
## 191 51.34224 50.26945 52.41503
## 192 51.34224 50.26945 52.41503
## 193 51.34224 50.26945 52.41503
## 194 50.19150 49.31302 51.06999
## 195 50.19150 49.31302 51.06999
## 196 50.19150 49.31302 51.06999
## 197 50.19150 49.31302 51.06999
## 198 50.19150 49.31302 51.06999
## 199 50.19150 49.31302 51.06999
## 200 47.14016 45.70937 48.57096
## 201 51.34224 50.26945 52.41503
## 202 51.34224 50.26945 52.41503
## 203 51.34224 50.26945 52.41503
## 204 50.19150 49.31302 51.06999
## 205 50.19150 49.31302 51.06999
## 206 50.19150 49.31302 51.06999
## 207 51.34224 50.26945 52.41503
## 208 51.34224 50.26945 52.41503
## 209 50.19150 49.31302 51.06999
## 210 51.34224 50.26945 52.41503
## 211 51.34224 50.26945 52.41503
## 212 50.19150 49.31302 51.06999
## 213 51.34224 50.26945 52.41503
## 214 51.34224 50.26945 52.41503
## 215 51.34224 50.26945 52.41503
## 216 47.14016 45.70937 48.57096
## 217 50.19150 49.31302 51.06999
## 218 47.14016 45.70937 48.57096
## 219 51.34224 50.26945 52.41503
## 220 50.19150 49.31302 51.06999
## 221 47.14016 45.70937 48.57096
## 222 51.34224 50.26945 52.41503
## 223 51.34224 50.26945 52.41503
## 224 51.34224 50.26945 52.41503
## 225 50.19150 49.31302 51.06999
## 226 51.34224 50.26945 52.41503
## 227 51.34224 50.26945 52.41503
## 228 51.34224 50.26945 52.41503
## 229 50.19150 49.31302 51.06999
## 230 50.19150 49.31302 51.06999
## 231 51.34224 50.26945 52.41503
## 232 51.34224 50.26945 52.41503
## 233 51.34224 50.26945 52.41503
## 234 51.34224 50.26945 52.41503
```

```
## 235 50.19150 49.31302 51.06999
## 236 50.19150 49.31302 51.06999
## 237 50.19150 49.31302 51.06999
## 238 51.34224 50.26945 52.41503
## 239 41.51485 40.08931 42.94039
## 240 41.51485 40.08931 42.94039
## 241 41.51485 40.08931 42.94039
## 242 41.51485 40.08931 42.94039
## 243 44.56619 42.98209 46.15029
## 244 41.51485 40.08931 42.94039
## 245 44.56619 42.98209 46.15029
## 246 41.51485 40.08931 42.94039
## 247 41.51485 40.08931 42.94039
## 248 41.51485 40.08931 42.94039
## 249 41.51485 40.08931 42.94039
## 250 41.51485 40.08931 42.94039
## 251 41.51485 40.08931 42.94039
## 252 41.51485 40.08931 42.94039
## 253 41.51485 40.08931 42.94039
## 254 41.51485 40.08931 42.94039
## 255 41.51485 40.08931 42.94039
## 256 44.56619 42.98209 46.15029
## 257 44.56619 42.98209 46.15029
## 258 44.56619 42.98209 46.15029
## 259 41.51485 40.08931 42.94039
## 260 44.56619 42.98209 46.15029
## 261 41.51485 40.08931 42.94039
## 262 41.51485 40.08931 42.94039
## 263 41.51485 40.08931 42.94039
## 264 41.51485 40.08931 42.94039
## 265 44.56619 42.98209 46.15029
## 266 44.56619 42.98209 46.15029
## 267 44.56619 42.98209 46.15029
## 268 44.56619 42.98209 46.15029
## 269 41.51485 40.08931 42.94039
## 270 41.51485 40.08931 42.94039
## 271 41.51485 40.08931 42.94039
## 272 44.56619 42.98209 46.15029
## 273 44.56619 42.98209 46.15029
## 274 45.71693 43.90174 47.53212
## 275 44.56619 42.98209 46.15029
## 276 44.56619 42.98209 46.15029
## 277 41.51485 40.08931 42.94039
```

```
predict(res2, data.frame(Group="Control", Agec="<40"), interval="prediction") # IC à 95% + la variance
```

```
## Warning: 'newdata' had 1 row but variables found have 277 rows
```

```
##          fit      lwr      upr
## 1   45.46300 37.15972 53.76627
## 2   45.46300 37.15972 53.76627
## 3   45.46300 37.15972 53.76627
## 4   45.46300 37.15972 53.76627
## 5   45.46300 37.15972 53.76627
## 6   46.61374 38.29015 54.93732
```

```
## 7   45.46300 37.15972 53.76627
## 8   45.46300 37.15972 53.76627
## 9   45.46300 37.15972 53.76627
## 10  45.46300 37.15972 53.76627
## 11  45.46300 37.15972 53.76627
## 12  42.41166 34.04383 50.77948
## 13  45.46300 37.15972 53.76627
## 14  45.46300 37.15972 53.76627
## 15  45.46300 37.15972 53.76627
## 16  42.41166 34.04383 50.77948
## 17  45.46300 37.15972 53.76627
## 18  46.61374 38.29015 54.93732
## 19  46.61374 38.29015 54.93732
## 20  45.46300 37.15972 53.76627
## 21  46.61374 38.29015 54.93732
## 22  45.46300 37.15972 53.76627
## 23  45.46300 37.15972 53.76627
## 24  45.46300 37.15972 53.76627
## 25  45.46300 37.15972 53.76627
## 26  42.41166 34.04383 50.77948
## 27  45.46300 37.15972 53.76627
## 28  46.61374 38.29015 54.93732
## 29  45.46300 37.15972 53.76627
## 30  45.46300 37.15972 53.76627
## 31  42.41166 34.04383 50.77948
## 32  42.41166 34.04383 50.77948
## 33  45.46300 37.15972 53.76627
## 34  42.41166 34.04383 50.77948
## 35  45.46300 37.15972 53.76627
## 36  45.46300 37.15972 53.76627
## 37  45.46300 37.15972 53.76627
## 38  42.41166 34.04383 50.77948
## 39  45.46300 37.15972 53.76627
## 40  45.46300 37.15972 53.76627
## 41  45.46300 37.15972 53.76627
## 42  45.46300 37.15972 53.76627
## 43  45.46300 37.15972 53.76627
## 44  46.61374 38.29015 54.93732
## 45  45.46300 37.15972 53.76627
## 46  42.41166 34.04383 50.77948
## 47  46.61374 38.29015 54.93732
## 48  45.46300 37.15972 53.76627
## 49  45.46300 37.15972 53.76627
## 50  45.46300 37.15972 53.76627
## 51  42.41166 34.04383 50.77948
## 52  46.61374 38.29015 54.93732
## 53  46.61374 38.29015 54.93732
## 54  45.46300 37.15972 53.76627
## 55  46.61374 38.29015 54.93732
## 56  45.46300 37.15972 53.76627
## 57  45.46300 37.15972 53.76627
## 58  45.46300 37.15972 53.76627
## 59  45.46300 37.15972 53.76627
## 60  45.46300 37.15972 53.76627
```

```
## 61  42.41166 34.04383 50.77948
## 62  42.41166 34.04383 50.77948
## 63  45.46300 37.15972 53.76627
## 64  46.61374 38.29015 54.93732
## 65  45.46300 37.15972 53.76627
## 66  46.61374 38.29015 54.93732
## 67  45.46300 37.15972 53.76627
## 68  42.41166 34.04383 50.77948
## 69  46.61374 38.29015 54.93732
## 70  46.61374 38.29015 54.93732
## 71  45.46300 37.15972 53.76627
## 72  46.61374 38.29015 54.93732
## 73  46.61374 38.29015 54.93732
## 74  46.61374 38.29015 54.93732
## 75  45.46300 37.15972 53.76627
## 76  45.46300 37.15972 53.76627
## 77  45.46300 37.15972 53.76627
## 78  45.46300 37.15972 53.76627
## 79  45.46300 37.15972 53.76627
## 80  45.46300 37.15972 53.76627
## 81  42.41166 34.04383 50.77948
## 82  46.61374 38.29015 54.93732
## 83  46.61374 38.29015 54.93732
## 84  46.61374 38.29015 54.93732
## 85  45.46300 37.15972 53.76627
## 86  45.46300 37.15972 53.76627
## 87  45.46300 37.15972 53.76627
## 88  46.61374 38.29015 54.93732
## 89  46.61374 38.29015 54.93732
## 90  45.46300 37.15972 53.76627
## 91  46.61374 38.29015 54.93732
## 92  46.61374 38.29015 54.93732
## 93  42.41166 34.04383 50.77948
## 94  46.61374 38.29015 54.93732
## 95  46.61374 38.29015 54.93732
## 96  46.61374 38.29015 54.93732
## 97  42.41166 34.04383 50.77948
## 98  45.46300 37.15972 53.76627
## 99  42.41166 34.04383 50.77948
## 100 46.61374 38.29015 54.93732
## 101 45.46300 37.15972 53.76627
## 102 42.41166 34.04383 50.77948
## 103 46.61374 38.29015 54.93732
## 104 46.61374 38.29015 54.93732
## 105 46.61374 38.29015 54.93732
## 106 45.46300 37.15972 53.76627
## 107 46.61374 38.29015 54.93732
## 108 46.61374 38.29015 54.93732
## 109 46.61374 38.29015 54.93732
## 110 42.41166 34.04383 50.77948
## 111 45.46300 37.15972 53.76627
## 112 46.61374 38.29015 54.93732
## 113 46.61374 38.29015 54.93732
## 114 46.61374 38.29015 54.93732
```

```
## 115 46.61374 38.29015 54.93732
## 116 45.46300 37.15972 53.76627
## 117 45.46300 37.15972 53.76627
## 118 42.41166 34.04383 50.77948
## 119 46.61374 38.29015 54.93732
## 120 50.19150 41.89066 58.49235
## 121 50.19150 41.89066 58.49235
## 122 50.19150 41.89066 58.49235
## 123 50.19150 41.89066 58.49235
## 124 50.19150 41.89066 58.49235
## 125 51.34224 43.01859 59.66589
## 126 50.19150 41.89066 58.49235
## 127 50.19150 41.89066 58.49235
## 128 50.19150 41.89066 58.49235
## 129 50.19150 41.89066 58.49235
## 130 50.19150 41.89066 58.49235
## 131 47.14016 38.76284 55.51748
## 132 50.19150 41.89066 58.49235
## 133 50.19150 41.89066 58.49235
## 134 50.19150 41.89066 58.49235
## 135 47.14016 38.76284 55.51748
## 136 50.19150 41.89066 58.49235
## 137 51.34224 43.01859 59.66589
## 138 51.34224 43.01859 59.66589
## 139 50.19150 41.89066 58.49235
## 140 51.34224 43.01859 59.66589
## 141 50.19150 41.89066 58.49235
## 142 50.19150 41.89066 58.49235
## 143 50.19150 41.89066 58.49235
## 144 50.19150 41.89066 58.49235
## 145 47.14016 38.76284 55.51748
## 146 50.19150 41.89066 58.49235
## 147 51.34224 43.01859 59.66589
## 148 50.19150 41.89066 58.49235
## 149 50.19150 41.89066 58.49235
## 150 47.14016 38.76284 55.51748
## 151 47.14016 38.76284 55.51748
## 152 50.19150 41.89066 58.49235
## 153 47.14016 38.76284 55.51748
## 154 50.19150 41.89066 58.49235
## 155 50.19150 41.89066 58.49235
## 156 50.19150 41.89066 58.49235
## 157 50.19150 41.89066 58.49235
## 158 50.19150 41.89066 58.49235
## 159 50.19150 41.89066 58.49235
## 160 50.19150 41.89066 58.49235
## 161 50.19150 41.89066 58.49235
## 162 50.19150 41.89066 58.49235
## 163 51.34224 43.01859 59.66589
## 164 50.19150 41.89066 58.49235
## 165 47.14016 38.76284 55.51748
## 166 51.34224 43.01859 59.66589
## 167 50.19150 41.89066 58.49235
## 168 50.19150 41.89066 58.49235
```

```
## 169 50.19150 41.89066 58.49235
## 170 47.14016 38.76284 55.51748
## 171 51.34224 43.01859 59.66589
## 172 51.34224 43.01859 59.66589
## 173 50.19150 41.89066 58.49235
## 174 51.34224 43.01859 59.66589
## 175 50.19150 41.89066 58.49235
## 176 50.19150 41.89066 58.49235
## 177 50.19150 41.89066 58.49235
## 178 50.19150 41.89066 58.49235
## 179 50.19150 41.89066 58.49235
## 180 50.19150 41.89066 58.49235
## 181 50.19150 41.89066 58.49235
## 182 50.19150 41.89066 58.49235
## 183 51.34224 43.01859 59.66589
## 184 50.19150 41.89066 58.49235
## 185 51.34224 43.01859 59.66589
## 186 50.19150 41.89066 58.49235
## 187 47.14016 38.76284 55.51748
## 188 51.34224 43.01859 59.66589
## 189 51.34224 43.01859 59.66589
## 190 50.19150 41.89066 58.49235
## 191 51.34224 43.01859 59.66589
## 192 51.34224 43.01859 59.66589
## 193 51.34224 43.01859 59.66589
## 194 50.19150 41.89066 58.49235
## 195 50.19150 41.89066 58.49235
## 196 50.19150 41.89066 58.49235
## 197 50.19150 41.89066 58.49235
## 198 50.19150 41.89066 58.49235
## 199 50.19150 41.89066 58.49235
## 200 47.14016 38.76284 55.51748
## 201 51.34224 43.01859 59.66589
## 202 51.34224 43.01859 59.66589
## 203 51.34224 43.01859 59.66589
## 204 50.19150 41.89066 58.49235
## 205 50.19150 41.89066 58.49235
## 206 50.19150 41.89066 58.49235
## 207 51.34224 43.01859 59.66589
## 208 51.34224 43.01859 59.66589
## 209 50.19150 41.89066 58.49235
## 210 51.34224 43.01859 59.66589
## 211 51.34224 43.01859 59.66589
## 212 50.19150 41.89066 58.49235
## 213 51.34224 43.01859 59.66589
## 214 51.34224 43.01859 59.66589
## 215 51.34224 43.01859 59.66589
## 216 47.14016 38.76284 55.51748
## 217 50.19150 41.89066 58.49235
## 218 47.14016 38.76284 55.51748
## 219 51.34224 43.01859 59.66589
## 220 50.19150 41.89066 58.49235
## 221 47.14016 38.76284 55.51748
## 222 51.34224 43.01859 59.66589
```

```
## 223 51.34224 43.01859 59.66589
## 224 51.34224 43.01859 59.66589
## 225 50.19150 41.89066 58.49235
## 226 51.34224 43.01859 59.66589
## 227 51.34224 43.01859 59.66589
## 228 51.34224 43.01859 59.66589
## 229 50.19150 41.89066 58.49235
## 230 50.19150 41.89066 58.49235
## 231 51.34224 43.01859 59.66589
## 232 51.34224 43.01859 59.66589
## 233 51.34224 43.01859 59.66589
## 234 51.34224 43.01859 59.66589
## 235 50.19150 41.89066 58.49235
## 236 50.19150 41.89066 58.49235
## 237 50.19150 41.89066 58.49235
## 238 51.34224 43.01859 59.66589
## 239 41.51485 33.13843 49.89127
## 240 41.51485 33.13843 49.89127
## 241 41.51485 33.13843 49.89127
## 242 41.51485 33.13843 49.89127
## 243 44.56619 36.16133 52.97105
## 244 41.51485 33.13843 49.89127
## 245 44.56619 36.16133 52.97105
## 246 41.51485 33.13843 49.89127
## 247 41.51485 33.13843 49.89127
## 248 41.51485 33.13843 49.89127
## 249 41.51485 33.13843 49.89127
## 250 41.51485 33.13843 49.89127
## 251 41.51485 33.13843 49.89127
## 252 41.51485 33.13843 49.89127
## 253 41.51485 33.13843 49.89127
## 254 41.51485 33.13843 49.89127
## 255 41.51485 33.13843 49.89127
## 256 44.56619 36.16133 52.97105
## 257 44.56619 36.16133 52.97105
## 258 44.56619 36.16133 52.97105
## 259 41.51485 33.13843 49.89127
## 260 44.56619 36.16133 52.97105
## 261 41.51485 33.13843 49.89127
## 262 41.51485 33.13843 49.89127
## 263 41.51485 33.13843 49.89127
## 264 41.51485 33.13843 49.89127
## 265 44.56619 36.16133 52.97105
## 266 44.56619 36.16133 52.97105
## 267 44.56619 36.16133 52.97105
## 268 44.56619 36.16133 52.97105
## 269 41.51485 33.13843 49.89127
## 270 41.51485 33.13843 49.89127
## 271 41.51485 33.13843 49.89127
## 272 44.56619 36.16133 52.97105
## 273 44.56619 36.16133 52.97105
## 274 45.71693 37.26547 54.16839
## 275 44.56619 36.16133 52.97105
## 276 44.56619 36.16133 52.97105
```

```
## 277 41.51485 33.13843 49.89127
```

Question 9

```
library(emmeans)
emmeans(res2, ~Group)
```

```
##  Group   emmean    SE  df lower.CL upper.CL
##  Bipolar   44.8 0.404 272     44.0     45.6
##  Control   49.6 0.417 272     48.7     50.4
##  UD        43.9 0.711 272     42.5     45.3
##
## Results are averaged over the levels of: Agec
## Confidence level used: 0.95
```

Dans ce cas, ça utilise une moyenne marginale, on vas faire comme si on avait une répartition de sujet parfaite dans chaque classe. Ca se construit en pondérant les moyenne en m'étant le même coef pour chaque classe d'age.

La moyenne de UD change beaucoup car peu de jeune dans ce groupe, il faut compenser l'impact des score moins bon des vieux.

Ici on veut garder l'information de l'age dans la predition. Car sinon on pourait juste faire une anova à 1 facteur.

Question 11

On cherche à savoir si $\alpha_{BD} = \alpha_{UD}$. On vas tester $H_0 = \{\alpha_{BD} = \alpha_{UD}\}, H_1 = \{\alpha_{BD} \neq \alpha UD\}$. C'est un test de constraste. $1 * \alpha_{BD} + 0 * \alpha_{ctrl} - 1 * \alpha_{UD} = 0$

```
c1 = c(1,0,-1)
contrast(emmeans(res2, ~Group), list(Group=c1)) # Pas d'écart significatif
```

```
##  contrast estimate    SE  df t.ratio p.value
##  Group       0.897 0.841 272   1.066  0.2872
##
## Results are averaged over the levels of: Agec
```

Question 11

```
contrast(emmeans(res2, ~Group), "pairwise", adjust="Tukey")
```

```
##  contrast          estimate    SE  df t.ratio p.value
##  Bipolar - Control   -4.729 0.545 272  -8.680  <.0001
##  Bipolar - UD         0.897 0.841 272   1.066  0.5358
##  Control - UD         5.625 0.854 272   6.590  <.0001
##
## Results are averaged over the levels of: Agec
## P value adjustment: tukey method for comparing a family of 3 estimates
```

```
cat("\n\n\n")
```

```
contrast(emmeans(res2, ~Group), "pairwise", adjust="Bonferroni")
```

```
##  contrast          estimate    SE  df t.ratio p.value
```

```
##  Bipolar - Control   -4.729 0.545 272  -8.680  <.0001
##  Bipolar - UD         0.897 0.841 272   1.066  0.8616
##  Control - UD         5.625 0.854 272   6.590  <.0001
##
## Results are averaged over the levels of: Agec
## P value adjustment: bonferroni method for 3 tests
```

Ici ça fait plusieurs (3) test into une p-value je crois, ce qui explique la différence de p-value avec le test question d'avant.