Mixed Model TTP

Data

```
Datos %>%
  ungroup() %>%
  dplyr::select(Subject,Level,TTP) %>%
  group_by(Subject,Level) %>%
  mutate(mid = 1:n()) %>%
  pivot_wider(names_from=mid,values_from=TTP) %>%
  arrange(Subject,Level) %>%
  kable("latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down"))
```

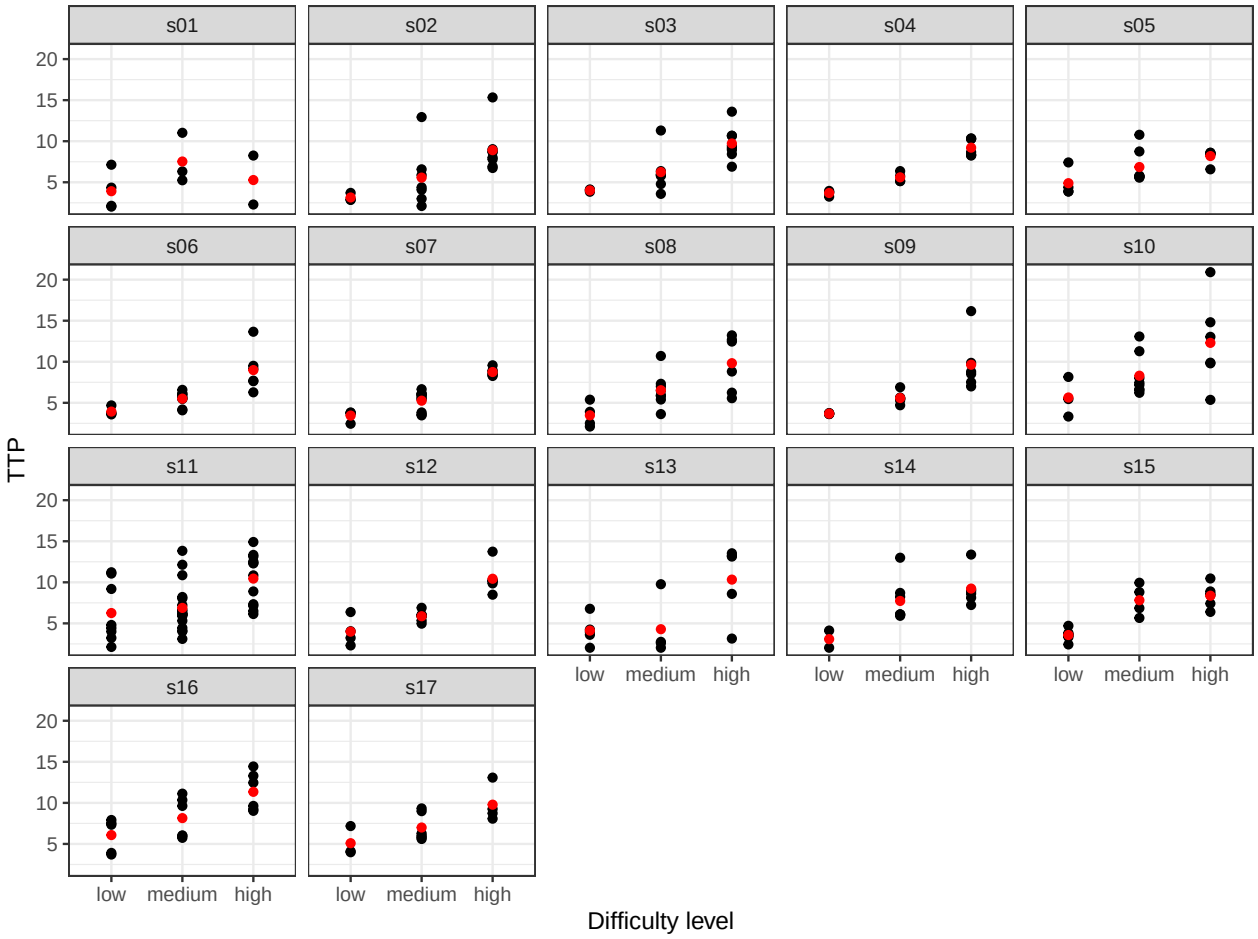
Subject	Level	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
s01	low	4.346	2.133	2.033	7.137															
s01	medium	6.319	5.243	11.024																
s01	high	8.253	2.292																	
s02	low	2.851	3.708	2.890																
s02	medium	6.579	4.386	2.113	12.938	5.881	4.087	2.990												
s02	high	8.014	8.771	15.330	6.937	9.030	6.738	8.652	7.795											
s03	low	4.127	4.047	3.847																
s03	medium	5.960	5.781	5.901	11.303	3.588	4.784	6.359												
s03	high	8.432	6.898	9.170	10.665	9.429	8.951	10.665	13.596											
s04	low	3.768	3.249	3.808	3.947															
s04	medium	6.359	5.761	5.143	5.143															
s04	high	8.253	8.353	8.672	10.287	8.313	10.287	10.346												
s05	low	4.386	3.867	7.416	3.887															
s05	medium	5.801	5.562	5.641	5.741	8.752	10.785	5.721												
s05	high	8.413	8.592	6.579	8.472	8.592	8.492													
s06	low	3.768	3.688	3.588	4.685															
s06	medium	5.881	5.642	5.482	6.140	4.186	6.579	5.502	6.080	4.087										
s06	high	6.279	7.635	7.695	9.509	13.655	9.150													
s07	low	2.452	3.828	3.728	3.768															
s07	medium	5.601	5.981	6.140	6.658	5.442	3.588	6.060	5.841	3.828	3.469									
s07	high	8.532	8.273	8.951	8.393	8.791	9.569													
s08	low	2.113	3.907	2.512	5.383															
s08	medium	5.402	5.841	5.901	7.316	3.628	10.705	6.997	6.499											
s08	high	8.811	13.217	5.562	12.679	6.239	12.459													
s09	low	3.728	3.748	3.628	3.668	3.648														
s09	medium	5.642	5.323	5.502	5.482	4.705	5.542	6.897												
s09	high	9.868	7.516	6.997	8.512	8.791	16.167													
s10	low	5.462	8.153	3.329																
s10	medium	8.014	11.283	13.077	6.220	6.539	7.456	7.177	6.658											
s10	high	9.888	14.812	13.037	20.912	9.808	5.362													
s11	low	4.007	9.190	4.804	11.064	3.230	2.133	4.426	11.223											
s11	medium	6.160	7.217	6.140	4.446	5.322	8.213	6.638	13.835	6.040	10.864	6.060	6.12	4.087	8.034	3.11	4.027	6.459	5.901	12.14
s11	high	7.336	10.845	13.336	14.912	13.137	6.499	12.360	7.137	6.140	12.300	8.891	12.54							
s12	low	4.047	3.229	2.312	6.379															
s12	medium	5.941	5.981	5.343	6.897	4.964	5.861	6.060	5.921											
s12	high	10.107	9.868	10.227	8.492	13.735	10.187													
s13	low	2.034	6.778	3.608	4.246	3.947														
s13	medium	9.768	2.751	2.612	2.033															
s13	high	13.137	13.237	8.592	13.536	3.150														
s14	low	2.033	4.127																	
s14	medium	6.100	12.998	5.921	6.080	8.233	6.080	8.712												
s14	high	8.632	8.133	8.931	7.237	9.110	13.377													
s15	low	3.688	4.705	2.432	3.588	3.329	3.828	3.509												
s15	medium	6.858	8.811	5.662	9.948															
s15	high	8.592	8.433	10.466	7.416	6.399	8.891													
s16	low	7.536	7.336	7.914	3.708	3.907														
s16	medium	5.741	5.980	6.040	11.124	9.629	10.346													
s16	high	9.230	9.629	13.297	12.459	9.031	14.433													
s17	low	3.987	4.107	7.177																
s17	medium	6.280	8.991	5.801	5.981	9.330	5.602													
s17	high	8.732	8.074	9.250	13.077															

Level	n	MD	SD
low	72	4.358	1.937
medium	124	6.507	2.316
high	106	9.650	2.796

Summary by group

```
Datos %>%
  group_by(Level) %>%
  summarise(n=n(),MD=mean(TTP),SD=sd(TTP)) %>%
  kable() %>%
  kable_styling(latex_options = c("striped"))
```

```
(q <-Datos %>% ggplot(aes(x=Level,y=TTP)) +
    geom_point() + facet_wrap(~ Subject)+
    labs(x="Difficulty level")+theme_bw()+
    stat_summary(fun="mean", geom="point",color="red"))
```



Random Intercept and Slope Model

The following model is used to investigate whether there are significant differences between the study variables:

$$y_{ij} = \mu + l_k + s_j + (sl)_{jk} + \epsilon_{ij}, \tag{1}$$

where y_{ij} is the response variable (TTP) for the i-th observation from the j-th subject, μ is the intercept, l_k is the k-th difficulty level, s_j is the jth subject effect, $(sl)_{jk}$ is the subject-level effect, i.e., the k-th level effect at the j-th subject, ϵ_{ij} is the error term (residual) for the ith observation from the jth subject.

We called *level* l a fixed effect, and ϵ is our *error term* that represent deviations from our predictions due to *random* factors that we cannot control experimentally. However, several measurements were taken for each subject at each difficulty level and that violates the assumption of independence of a linear model. On the other hand, each individual has a different cognitive load capacity, and this will be a characteristic factor that will affect all the responses of the same subject, which will make these responses interdependent instead of independent, see figure **??**. The way we approaches this situation is adding a random effect to the subject and to the subject-level interaction. This allows us to solve this lack of independence by assuming a different intercept and slope for each subject. And finally, we assume that the residual, subject and subject-level effects are all relations of separate distributions, all with zero means:

$$\begin{aligned} \epsilon_{ij} &\sim N(0, \sigma^2), \\ s_j &\sim N(0, \sigma_s^2), \\ (sl)_{jk} &\sim N(0, \sigma_{sl}^2). \end{aligned}$$

Hence, s_j and $(sl)_{jk}$ are now random effects, and μ and l_k are fixed effects.

Using the **R** notation the model is

$$TTP = (b_0 + u_{Subject}) + b_{Level} Level + \epsilon$$

In order to evaluate if there is an effect due to the difficulty level we will use the likelihood ratio test of the model with the *Level* effect against the model without the *Level* effect.

```
TTP_mixed_reducido <- lme4::lmer(TTP ~ 1 + (1+Level|Subject),data=Datos,REML=F)

TTP_mixed_lme4 <- lme4::lmer(TTP ~ Level + (1+Level|Subject),data=Datos,REML=F)

anova(TTP_mixed_reducido,TTP_mixed_lme4)
```

```
## Data: Datos
## Models:
## TTP_mixed_reducido: TTP ~ 1 + (1 + Level | Subject)
## TTP_mixed_lme4: TTP ~ Level + (1 + Level | Subject)
##                    npar  AIC  BIC logLik deviance Chisq Df    Pr(>Chisq)
## TTP_mixed_reducido    8 1445 1475   -715     1429
## TTP_mixed_lme4       10 1403 1440   -692     1383  46.5  2 0.00000000008 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p-value of the ratio test is significant at a level of 0.001.

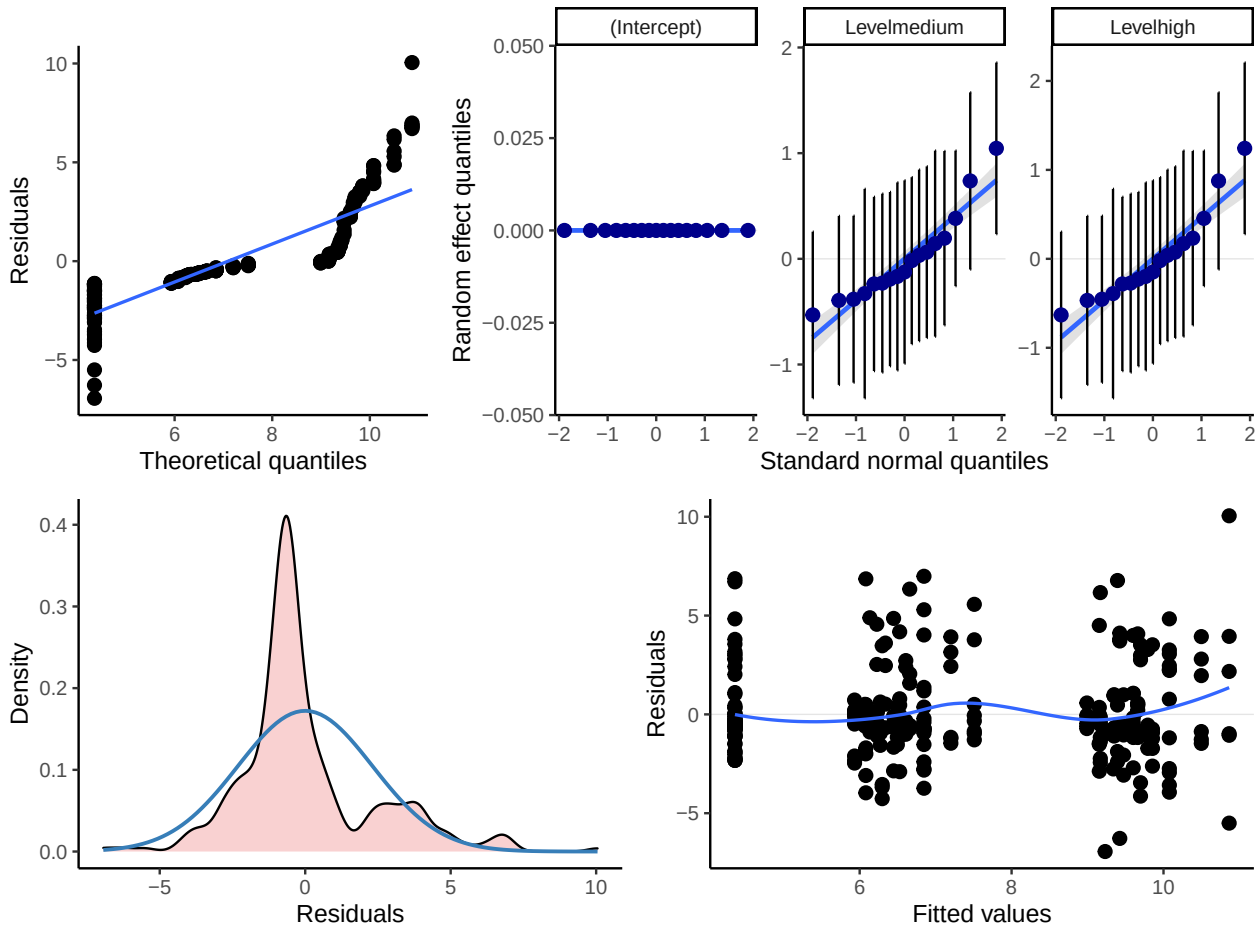
```
TTP_mixed_lme4 <- lme4::lmer(TTP ~ Level + (1+Level|Subject),data=Datos)
summary(TTP_mixed_lme4)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: TTP ~ Level + (1 + Level | Subject)
##    Data: Datos
```

```
##
## REML criterion at convergence: 1385
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -2.946 -0.463 -0.239  0.222  4.266
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  Subject  (Intercept) 0.000    0.000
##           Levelmedium 0.349    0.591     NaN
##           Levelhigh   0.493    0.702     NaN 1.00
##  Residual             5.546    2.355
## Number of obs: 302, groups:  Subject, 17
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)    4.358      0.278   15.70
## Levelmedium    2.105      0.380    5.54
## Levelhigh      5.264      0.400   13.17
##
## Correlation of Fixed Effects:
##             (Intr) Lvlmdm
## Levelmedium -0.731
## Levelhigh   -0.694  0.676
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

```
p<-plot_model(TTP_mixed_lme4, type = "diag")

({p[[1]]+theme(plot.title=element_blank(),plot.subtitle=element_blank())+scale_x_continuous(name="Theor
```




```
contr <- glht(TTP_mixed_lme4,linfct=mcp(Level="Tukey"))
summary(contr, test = adjusted("holm"))
```

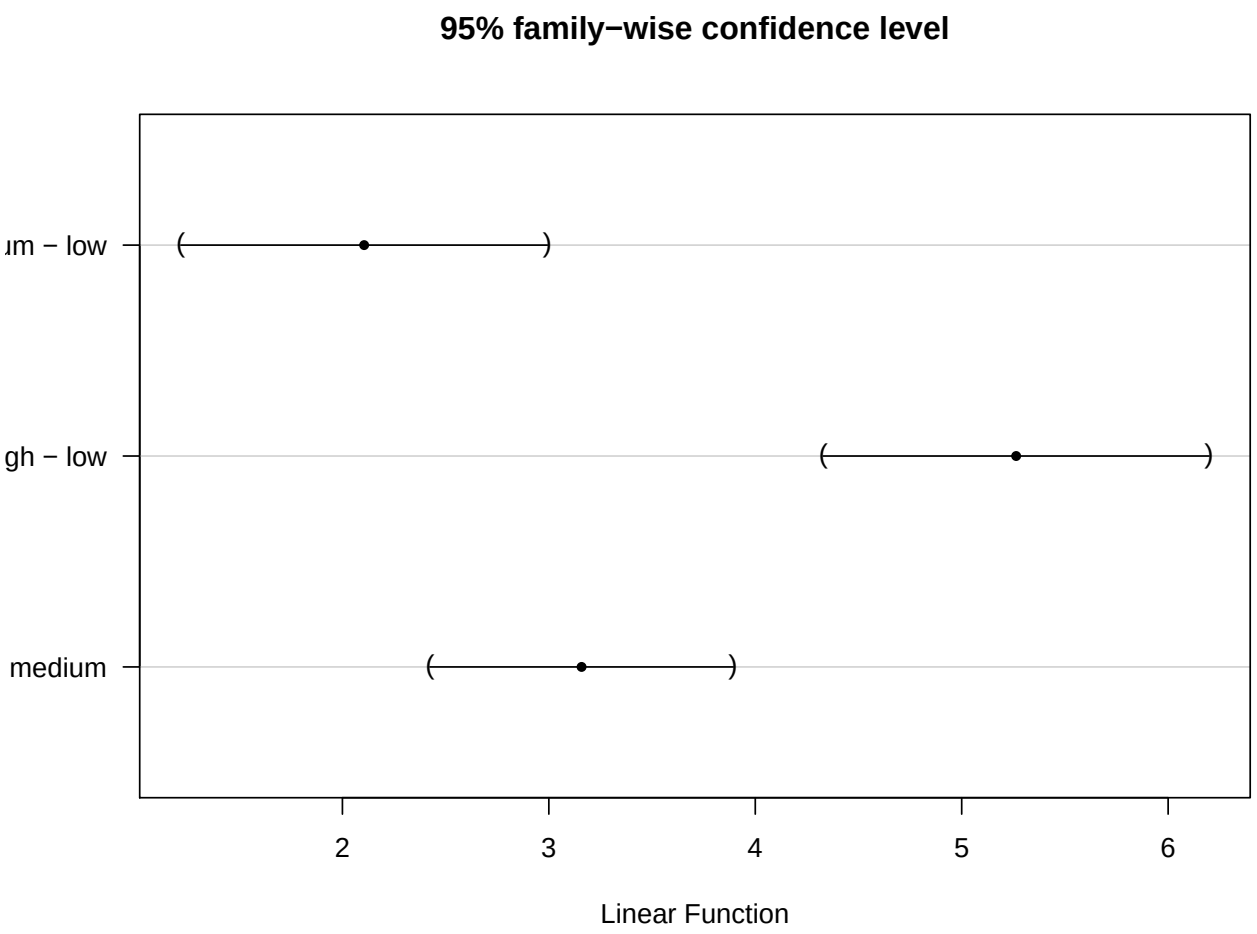
```
##
##   Simultaneous Tests for General Linear Hypotheses
##
## Multiple Comparisons of Means: Tukey Contrasts
##
##
## Fit: lme4::lmer(formula = TTP ~ Level + (1 + Level | Subject), data = Datos)
##
## Linear Hypotheses:
##                     Estimate Std. Error z value             Pr(>|z|)
## medium - low == 0     2.105      0.380    5.54           0.00000003 ***
## high - low == 0       5.264      0.400   13.17 < 0.0000000000000002 ***
## high - medium == 0    3.159      0.314   10.05 < 0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- holm method)
```

```
confint(contr)
```

```
##
##   Simultaneous Confidence Intervals
##
```

```
## Multiple Comparisons of Means: Tukey Contrasts
##
##
## Fit: lme4::lmer(formula = TTP ~ Level + (1 + Level | Subject), data = Datos)
##
## Quantile = 2.338
## 95% family-wise confidence level
##
##
## Linear Hypotheses:
##                     Estimate lwr   upr
## medium - low == 0  2.105    1.217 2.993
## high - low == 0    5.264    4.329 6.198
## high - medium == 0 3.159    2.424 3.893
```

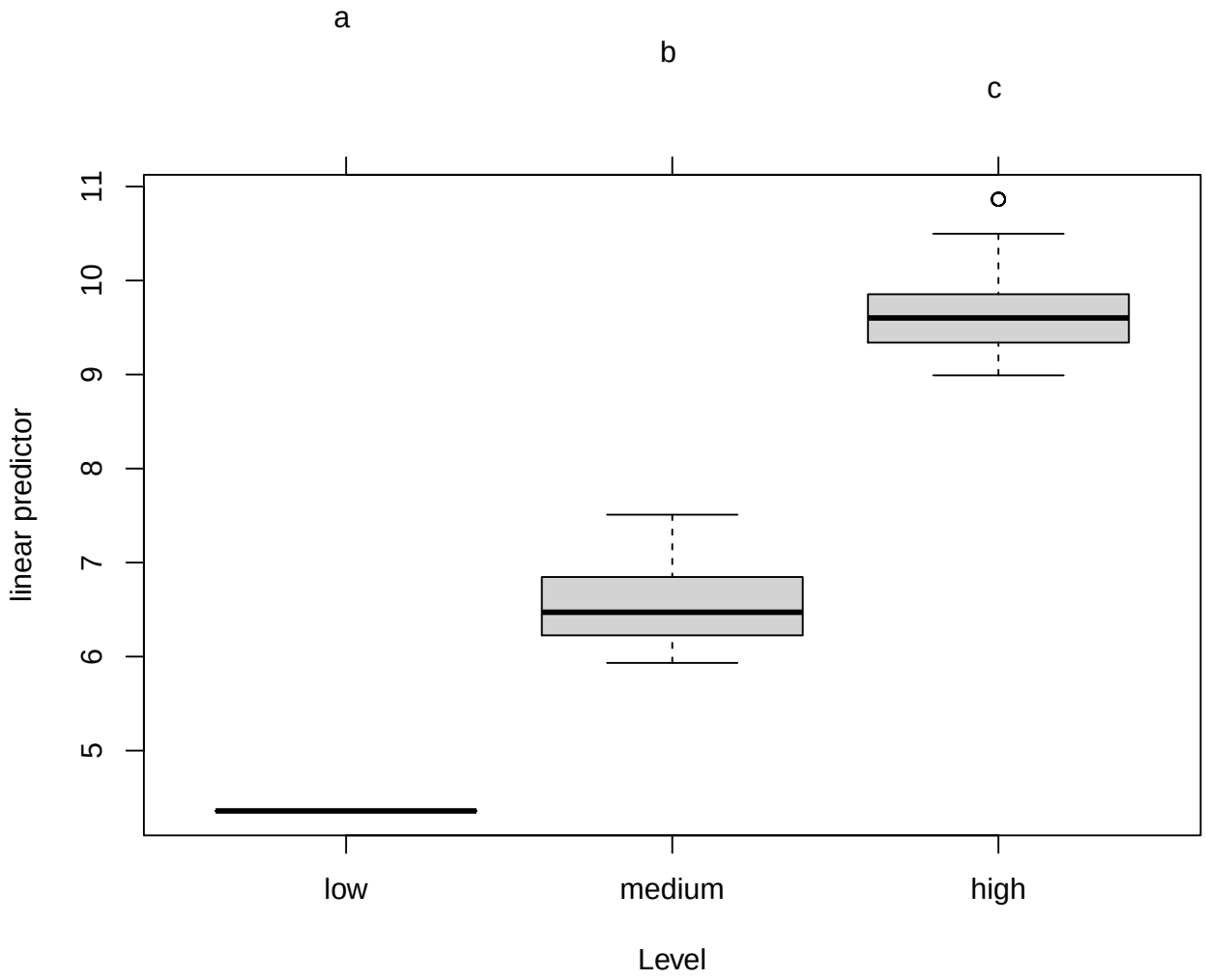
```
plot(confint(contr))
```




```
contr.cld <- cld(contr)
old.par <- par(mai=c(1,1,1.25,1), no.readonly = TRUE)
plot(contr.cld)
```

Subject	Name	Training	Nivel	BLPS	MPDC	APCPS	PD	Entropy	TTP	PDS	SequenceMemory	SMN	id	Level	res	fit
s10	SequenceMemory_r18	FALSE	6	4.260	0.0646	0.0152	0.3601	-2.115	20.91	0	r18	18	11	high	10.047	10.864
s11	SequenceMemory_r11	FALSE	3	3.495	0.1883	0.0539	0.4335	-1.881	13.83	0	r11	11	14	medium	6.988	6.846
s11	SequenceMemory_r31	FALSE	1	3.512	0.2471	0.0704	0.6464	-1.914	11.22	0	r31	31		low	6.865	4.358
s02	SequenceMemory_r20	FALSE	3	3.963	0.0280	0.0071	0.2739	-1.998	12.94	0	r20	20	11	medium	6.856	6.082
s09	SequenceMemory_r32	FALSE	6	3.512	0.3679	0.1048	0.6011	-1.957	16.17	0	r32	32	18	high	6.773	9.394
s11	SequenceMemory_r07	FALSE	1	3.806	-0.0707	-0.0186	0.0779	-1.902	11.06	0	r07	7	8	low	6.706	4.358

Subject	Name	Training	Nivel	BLPS	MPDC	APCPS	PD	Entropy	TTP	PDS	SequenceMemory	SMN	id	Level	res	fit
s02	SequenceMemory_r13	FALSE	3	4.256	-0.2476	-0.0582	0.0684	-2.005	2.113	0	r13	13	7	medium	-3.968	6.082
s08	SequenceMemory_r12	FALSE	6	3.474	0.1036	0.0298	0.2330	-1.839	5.562	0	r12	12	8	high	-4.135	9.697
s13	SequenceMemory_r15	FALSE	3	4.255	-0.3314	-0.0779	-0.0439	-1.973	2.033	0	r15	15	8	medium	-4.264	6.297
s10	SequenceMemory_r32	FALSE	6	4.077	0.2042	0.0501	0.4647	-2.099	5.362	0	r32	32	17	high	-5.502	10.864
s13	SequenceMemory_r22	FALSE	6	3.995	-0.1146	-0.0287	0.1799	-1.957	3.150	0	r22	22	11	high	-6.275	9.424
s01	SequenceMemory_r17	FALSE	6	4.023	-0.2674	-0.0665	0.1099	-1.911	2.292	0	r17	17	7	high	-6.939	9.231




```
par(old.par)
```

```
Datos2 = Datos
Datos2$res = residuals(TTP_mixed_lme4,type="pearson")
Datos2$fit = fitted(TTP_mixed_lme4,type="pearson")

Datos2 %>% arrange(desc(res)) %>% head() %>% kable() %>%
  kable_styling(latex_options = c("striped", "scale_down"))
```

```
Datos2 %>% arrange(desc(res)) %>% tail() %>% kable() %>%
  kable_styling(latex_options = c("striped", "scale_down"))
```

```r
shapiro.test(Datos2$res)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Datos2$res
## W = 0.91, p-value = 0.000000000001
```

```r
goftest::ad.test(Datos2$res,null="pnorm",mean=mean(Datos2$res), sd=sd(Datos2$res), estimated=TRUE)
```

```
##
##  Anderson-Darling test of goodness-of-fit
##  Braun's adjustment using 17 groups
##  Null hypothesis: Normal distribution
##  with parameters mean = 0.000000000000001351039405442555, sd =
##  2.31632730908459
##  Parameters assumed to have been estimated from data
##
## data:  Datos2$res
## Anmax = 2.8, p-value = 0.4
```

```r
rstatix::levene_test(data=ungroup(Datos2),res~Level)
```

```
## # A tibble: 1 x 4
##     df1   df2 statistic      p
##   <int> <int>     <dbl>  <dbl>
## 1     2   299      3.73 0.0250
```

The same model without the outliers

We repeat the analysis without the outlier

```r
# we exclude the outlier
Datos <- Datos %>% filter(!(Subject=="s10"&SMN==18))
```

```r
TTP_mixed_lme4 <- lme4::lmer(TTP ~ Level + (1+Level|Subject),data=Datos)
summary(TTP_mixed_lme4)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: TTP ~ Level + (1 + Level | Subject)
##    Data: Datos
##
## REML criterion at convergence: 1361
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -3.129 -0.469 -0.243  0.265  3.046
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  Subject  (Intercept) 0.000    0.000
##           Levelmedium 0.360    0.600     NaN
##           Levelhigh   0.144    0.379     NaN 1.00
##  Residual             5.246    2.290
## Number of obs: 301, groups:  Subject, 17
```

```
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)    4.358      0.270   16.15
## Levelmedium    2.117      0.372    5.68
## Levelhigh      5.178      0.363   14.27
##
## Correlation of Fixed Effects:
##             (Intr) Lvlmdm
## Levelmedium -0.725
## Levelhigh   -0.744  0.643
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

```
anova(TTP_mixed_lme4)
```

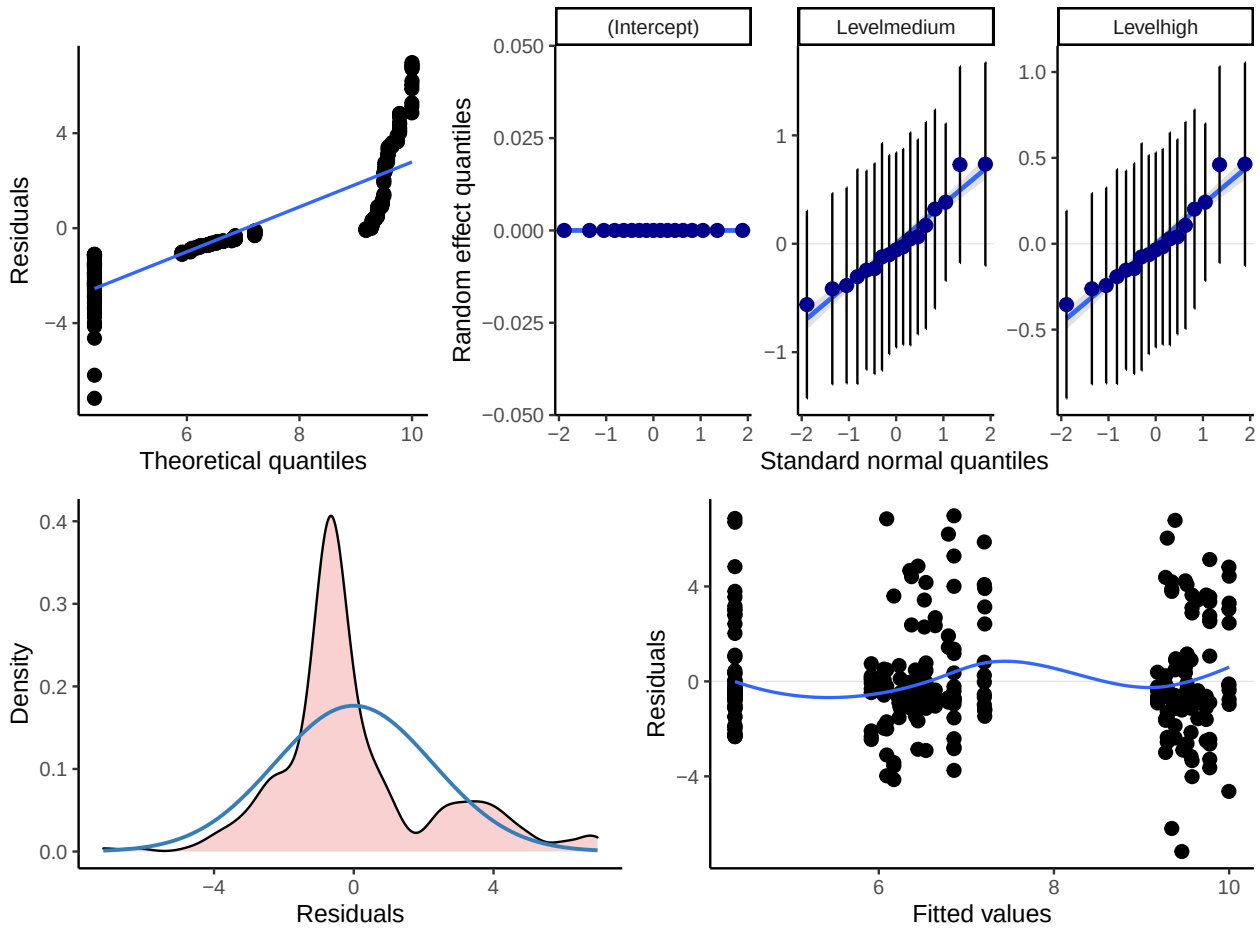
```
## Analysis of Variance Table
##       npar Sum Sq Mean Sq F value
## Level    2   1177     589     112
```

```
coef(TTP_mixed_lme4)
```

```
## $Subject
##     (Intercept) Levelmedium Levelhigh
## s01       4.358       1.996     5.102
## s02       4.358       1.732     4.935
## s03       4.358       2.089     5.161
## s04       4.358       1.887     5.033
## s05       4.358       2.016     5.114
## s06       4.358       1.702     4.916
## s07       4.358       1.556     4.824
## s08       4.358       2.181     5.219
## s09       4.358       1.873     5.025
## s10       4.358       2.847     5.639
## s11       4.358       2.501     5.421
## s12       4.358       2.060     5.142
## s13       4.358       1.814     4.987
## s14       4.358       2.437     5.380
## s15       4.358       2.162     5.207
## s16       4.358       2.852     5.642
## s17       4.358       2.287     5.285
##
## attr(,"class")
## [1] "coef.mer"
```

```
p<-plot_model(TTP_mixed_lme4, type = "diag")

(q<-{p[[1]]+theme(plot.title=element_blank(),plot.subtitle=element_blank())+scale_x_continuous(name="Th
```




```
# muy importante Tukey para lme4.
contr <- glht(TTP_mixed_lme4,linfct=mcp(Level="Tukey"))
summary(contr, test = adjusted("holm"))
```

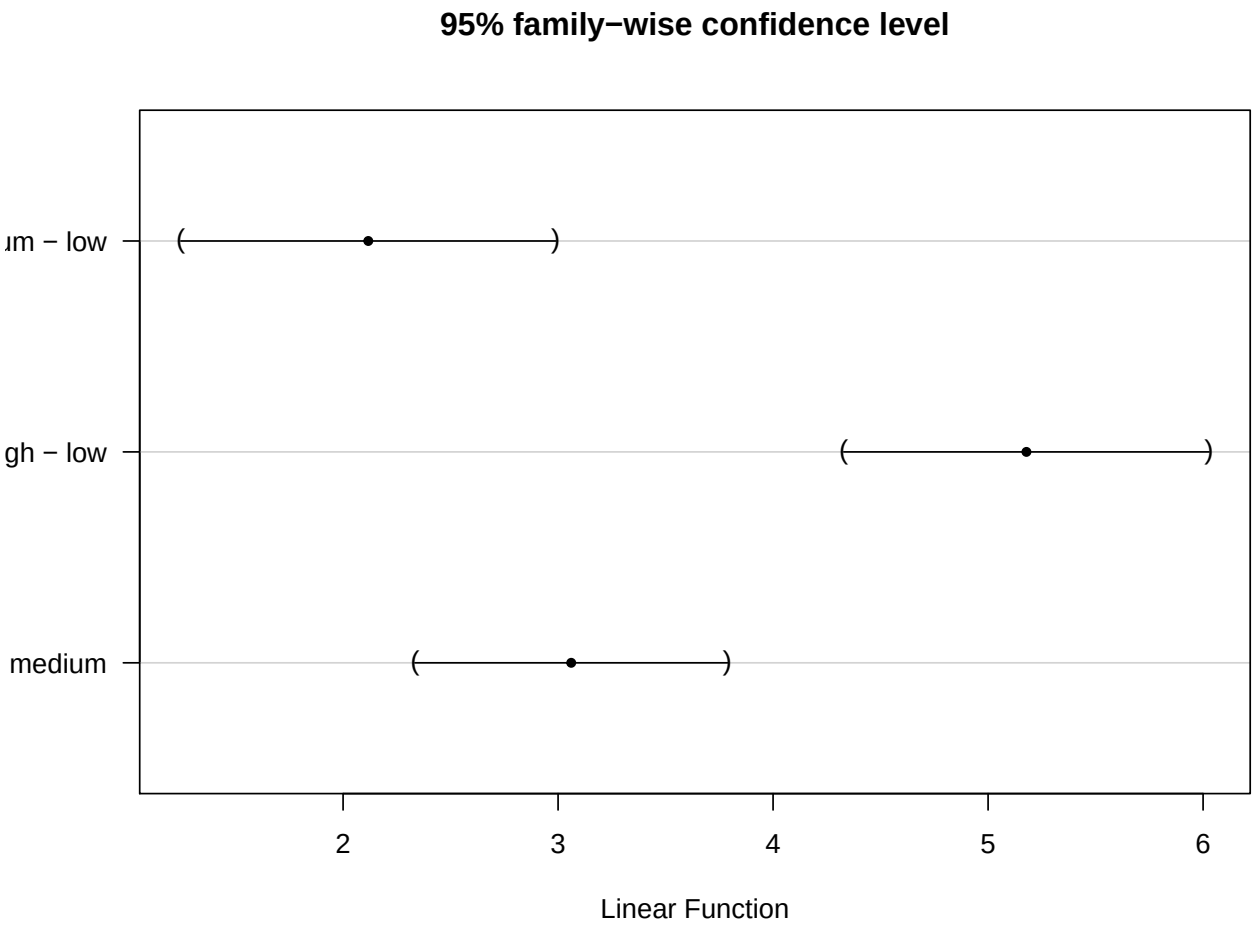
```
##
##   Simultaneous Tests for General Linear Hypotheses
##
## Multiple Comparisons of Means: Tukey Contrasts
##
##
## Fit: lme4::lmer(formula = TTP ~ Level + (1 + Level | Subject), data = Datos)
##
## Linear Hypotheses:
##                     Estimate Std. Error z value             Pr(>|z|)
## medium - low == 0      2.117      0.372    5.68          0.000000013 ***
## high - low == 0        5.178      0.363   14.27 < 0.0000000000000002 ***
## high - medium == 0     3.061      0.311    9.85 < 0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- holm method)
```

```
confint(contr)
```

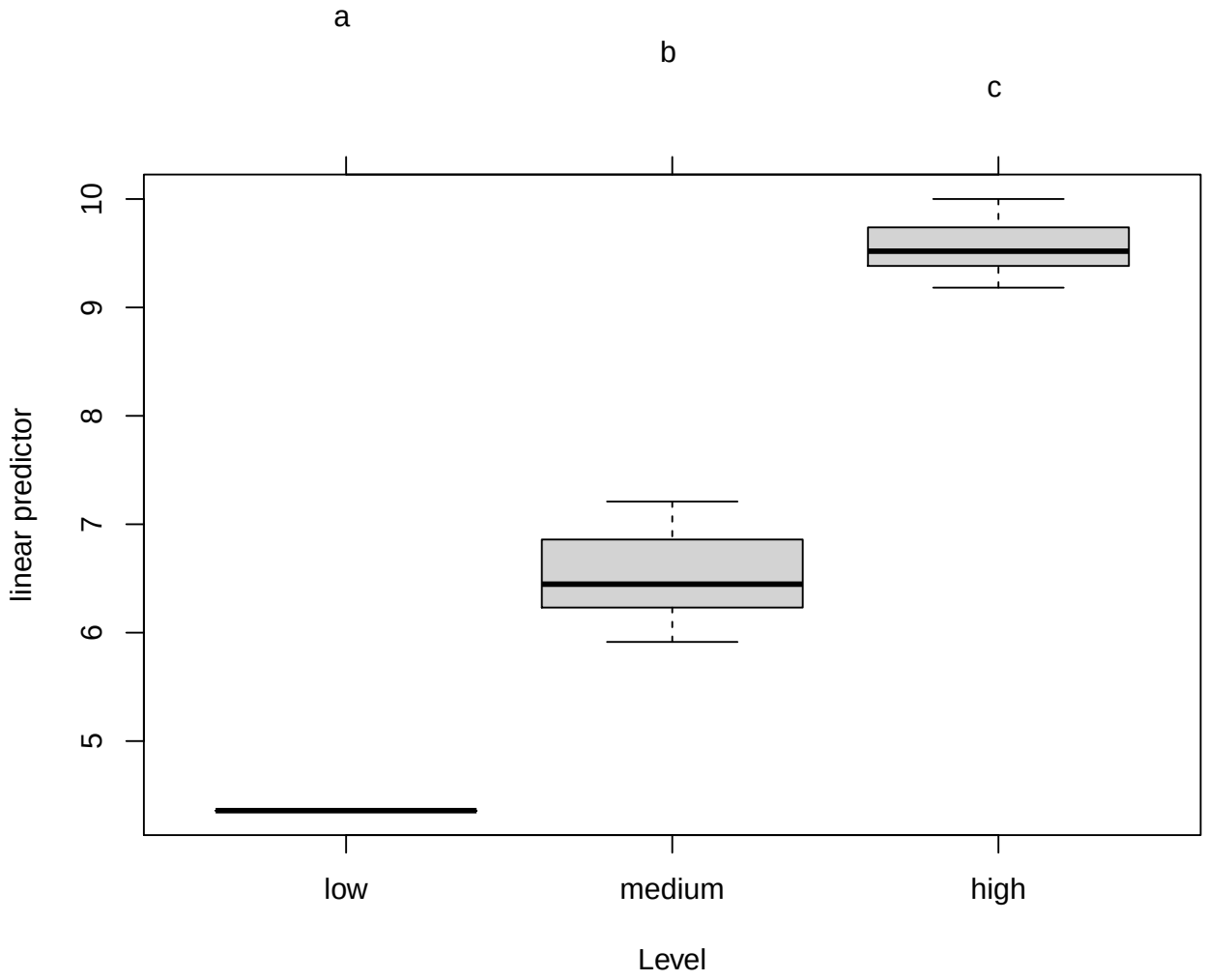
```
##
##   Simultaneous Confidence Intervals
```

```
##
## Multiple Comparisons of Means: Tukey Contrasts
##
##
## Fit: lme4::lmer(formula = TTP ~ Level + (1 + Level | Subject), data = Datos)
##
## Quantile = 2.341
## 95% family-wise confidence level
##
##
## Linear Hypotheses:
##                      Estimate lwr   upr
## medium - low == 0  2.117    1.245 2.989
## high - low == 0    5.178    4.329 6.028
## high - medium == 0 3.061    2.334 3.788
```

```
plot(confint(contr))
```




```
contr.cld <- cld(contr)
### use sufficiently large upper margin
old.par <- par(mai=c(1,1,1.25,1), no.readonly = TRUE)
### plot
plot(contr.cld)
```




```
par(old.par)
```

```
Datos2=Datos
Datos2$res = residuals(TTP_mixed_lme4,type="pearson")
Datos2$fit = fitted(TTP_mixed_lme4,type="pearson")
shapiro.test(Datos2$res)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Datos2$res
## W = 0.91, p-value = 0.000000000002
```

```
goftest::ad.test(Datos2$res,null="pnorm",mean=mean(Datos2$res), sd=sd(Datos2$res), estimated=TRUE)
```

```
##
##  Anderson-Darling test of goodness-of-fit
##  Braun's adjustment using 17 groups
##  Null hypothesis: Normal distribution
##  with parameters mean = 0.00000000000000143493040908279, sd =
##  2.25897724385221
##  Parameters assumed to have been estimated from data
##
```

```
## data:  Datos2$res
## Anmax = 3.5, p-value = 0.2
```

```
rstatix::levene_test(data=ungroup(Datos2),res~Level)
```

```
## # A tibble: 1 x 4
##     df1   df2 statistic      p
##   <int> <int>     <dbl>  <dbl>
## 1     2   298      3.42 0.0340
```

Non parametric tests

```
kruskal.test(TTP ~ Level, data=Datos)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  TTP by Level
## Kruskal-Wallis chi-squared = 143, df = 2, p-value <0.0000000000000002
```

```
PMCMR::posthoc.kruskal.nemenyi.test(data=Datos,TTP~Level, dist="Tukey")
```

```
##
##  Pairwise comparisons using Tukey and Kramer (Nemenyi) test
##                    with Tukey-Dist approximation for independent samples
##
## data:  TTP by Level
##
##        low                  medium
## medium 0.00000024888115     -
## high   < 0.0000000000000002 0.00000000000014
##
## P value adjustment method: none
```

```
PMCMRplus::tukeyTest(data=Datos,TTP~Level)
```

```
##        low              medium
## medium 0.00000000502121 -
## high   0.00000000000081 0.00000000000084
```