

Acquisizione e pre-processing del dataset

Leggo il dataset delle frequenze cardiache rilevate ai tempi 0..5 e Dimissione di 88 pazienti:

```
> dati = read.csv2("https://www.crescenziogallo.it/pub/Dataset_ANOVA_FC.csv")
```

Esamino la struttura del dataframe (616 righe = 88 pazienti x 7 tempi):

```
> str(dati)
```

```
'data.frame':      616 obs. of  2 variables:
 $ Tempo      : chr  "FC_0" "FC_0" "FC_0" "FC_0" ...
 $ Frequenza: num  92 94 80 74 74 65 85 117 95 86 ...
```

Converto la variabile Tempo in categorica e rivedo la struttura:

```
> dati$Tempo = factor(dati$Tempo)
```

```
> str(dati)
```

```
'data.frame':      616 obs. of  2 variables:
 $ Tempo      : Factor w/ 7 levels "FC_0","FC_1",...: 1 1 1 1 1 1 1 1 1 1 ...
 $ Frequenza: num  92 94 80 74 74 65 85 117 95 86 ...
```

Esamino la parte iniziale e finale del dataset e lo registro:

```
> head(dati)
```

	Tempo	Frequenza
1	FC_0	92
2	FC_0	94
3	FC_0	80
4	FC_0	74
5	FC_0	74
6	FC_0	65

```
> tail(dati)
```

	Tempo	Frequenza
611	FC_Dim	60
612	FC_Dim	80
613	FC_Dim	71
614	FC_Dim	73
615	FC_Dim	55
616	FC_Dim	55

```
> save(dati, file = "Dataset_ANOVA_FC.RData")
```

Analisi statistiche preliminari

Imposto il dataframe "dati" nella search path di R:

```
> attach(dati)
```

Imposto una sequenza di colori da utilizzare nel grafico successivo:

```
> colori = c("azure", "beige", "bisque", "bisque1", "bisque2", "bisque3", "bisque4")
```

Traccio il grafico dei boxplot comparati per Tempo:

```
> boxplot(Frequenza ~ Tempo, col = colori)
```

Estraggo ora i soli tempi FC_0 ed FC_1:

```
> df2 = subset(dati, Tempo=="FC_0" | Tempo=="FC_1")
```

```
> df2$Tempo = factor(df2$Tempo)
```

```
> str(df2)
```

```
'data.frame':    176 obs. of  2 variables:
 $ Tempo      : Factor w/ 2 levels "FC_0","FC_1": 1 1 1 1 1 1 1 1 1 1 ...
 $ Frequenza: num  92 94 80 74 74 65 85 117 95 86 ...
```

Controllo la normalità delle distribuzioni delle frequenze (FC_0 quasi, FC_1 no):

```
> library(nortest)
```

```
> shapiro.test(df2$Frequenza[Tempo=="FC_0"])
```

```
Shapiro-Wilk normality test
```

```
data:  df2$Frequenza[Tempo == "FC_0"]
```

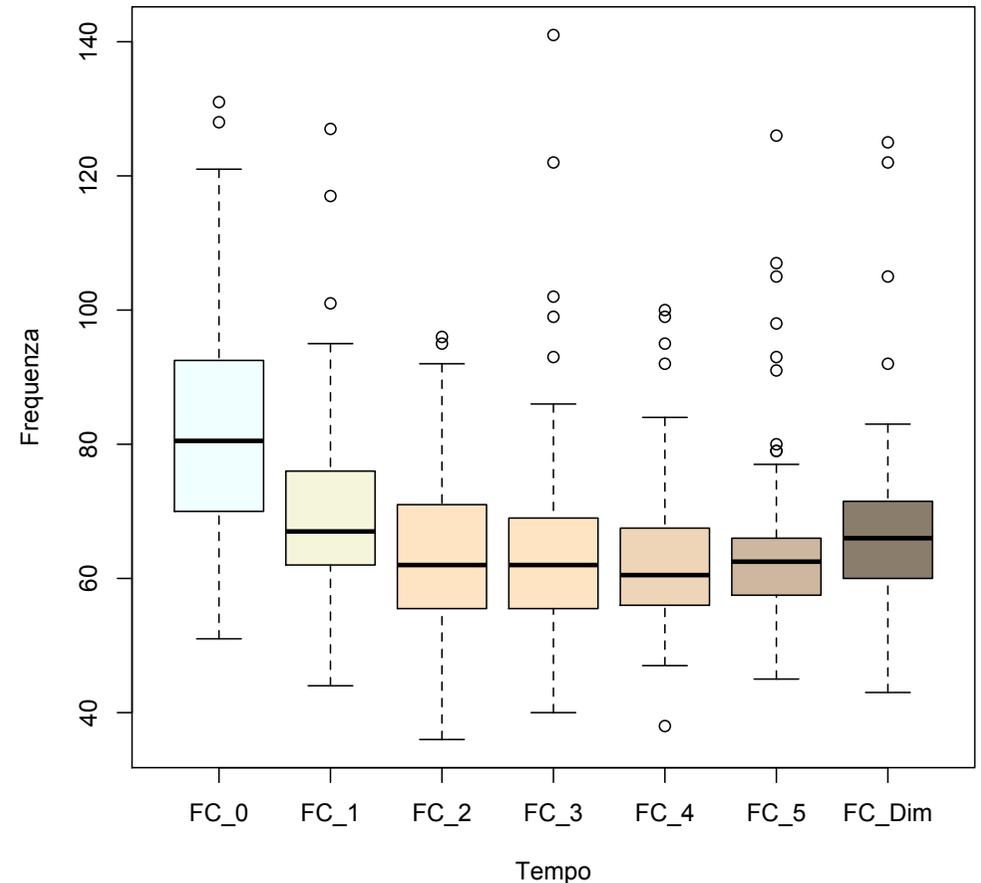
```
W = 0.97167, p-value = 0.05071
```

```
> shapiro.test(df2$Frequenza[Tempo=="FC_1"])
```

```
Shapiro-Wilk normality test
```

```
data:  df2$Frequenza[Tempo == "FC_1"]
```

```
W = 0.88847, p-value = 1.608e-06
```



Analisi dei gruppi (t-test, Wilcoxon, ANOVA)

Eseguo il t-test appaiato FC_0 vs FC_1 (parametrico, Welch):

```
> t.test(df2$Frequenza ~ df2$Tempo)
Welch Two Sample t-test
data: df2$Frequenza by df2$Tempo
t = 5.0517, df = 167.16, p-value = 1.138e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 7.289509 16.642309
sample estimates:
mean in group FC_0 mean in group FC_1
      82.21591      70.25000
```

Eseguo il test appaiato signed-rank di Wilcoxon FC_0 vs FC_1 (non parametrico, più attendibile), nel quale viene riportato il p-value (calcolato rispetto alla variabile aleatoria t) e le medie della Frequenza nei sette gruppi:

```
> wilcox.test(df2$Frequenza ~ df2$Tempo, paired = TRUE, alternative = "two.sided")
Wilcoxon signed rank test with continuity correction
data: df2$Frequenza by df2$Tempo
V = 2573, p-value = 1.353e-09
alternative hypothesis: true location shift is not equal to 0
```

Ora esamino TUTTI i sette gruppi FC_0-FC_5 e FC_Dim. Controllo prima la normalità delle distribuzioni dei sette tempi (risulta "quasi" normale solo FC_0):

```
> shapiro.test(Frequenza[Tempo=="FC_0"])
Shapiro-Wilk normality test
data: Frequenza[Tempo == "FC_0"]
W = 0.97167, p-value = 0.05071
> shapiro.test(Frequenza[Tempo=="FC_1"])
Shapiro-Wilk normality test
data: Frequenza[Tempo == "FC_1"]
W = 0.88847, p-value = 1.608e-06
```

```
> shapiro.test(Frequenza[Tempo=="FC_2"])
```

```
Shapiro-Wilk normality test
```

```
data: Frequenza[Tempo == "FC_2"]
```

```
W = 0.96054, p-value = 0.008947
```

```
> shapiro.test(Frequenza[Tempo=="FC_3"])
```

```
Shapiro-Wilk normality test
```

```
data: Frequenza[Tempo == "FC_3"]
```

```
W = 0.81674, p-value = 4.449e-09
```

```
> shapiro.test(Frequenza[Tempo=="FC_4"])
```

```
Shapiro-Wilk normality test
```

```
data: Frequenza[Tempo == "FC_4"]
```

```
W = 0.8986, p-value = 4.357e-06
```

```
> shapiro.test(Frequenza[Tempo=="FC_5"])
```

```
Shapiro-Wilk normality test
```

```
data: Frequenza[Tempo == "FC_5"]
```

```
W = 0.78175, p-value = 4.248e-10
```

```
> shapiro.test(Frequenza[Tempo=="FC_Dim"])
```

```
Shapiro-Wilk normality test
```

```
data: Frequenza[Tempo == "FC_Dim"]
```

```
W = 0.84594, p-value = 3.977e-08
```

Quindi eseguo l'ANOVA tra i sette tempi (anche se non dovrei, data la non normalità dei tempi):

```
> summary(aov(Frequenza ~ Tempo))
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Tempo	6	23938	3990	20.82	<2e-16 ***
Residuals	609	116715	192		

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Eseguo il confronto tra i sette tempi con un modello lineare di regressione multipla (anche se non dovrei, data la non normalità dei tempi), nel quale viene riportato lo stesso p-value (<2e-16) dato da aov, ma ci vengono fornite molte altre informazioni, utili a giudicare in prima battuta la 'bontà' del modello:

```
> summary(lm(Frequenza ~ Tempo))
```

```
Call:
```

```
lm(formula = Frequenza ~ Tempo)
```

```
Residuals:
```

```
      Min       1Q   Median       3Q      Max
-31.216 -8.250 -2.114   5.114  76.080
```

```
Coefficients:
```

```
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    82.216      1.476   55.711 < 2e-16 ***
TempoFC_1     -11.966      2.087   -5.733 1.55e-08 ***
TempoFC_2     -18.295      2.087   -8.766 < 2e-16 ***
TempoFC_3     -17.295      2.087   -8.287 7.42e-16 ***
TempoFC_4     -19.250      2.087   -9.224 < 2e-16 ***
TempoFC_5     -17.705      2.087   -8.483 < 2e-16 ***
TempoFC_Dim   -15.330      2.087   -7.345 6.63e-13 ***
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 13.84 on 609 degrees of freedom
```

```
Multiple R-squared:  0.1702,    Adjusted R-squared:  0.162
```

```
F-statistic: 20.82 on 6 and 609 DF,  p-value: < 2.2e-16
```

Si osservino le stime dei 'coefficienti del modello' ottenute con lm: si intuisce che queste sono le medie di Frequenza nei sette gruppi FC_0 (che qui viene denominato con la dicitura (Intercept)) e dei gruppi successivi, che si ottengono sommando ad (Intercept) la differenza indicata da TempoFC_..., ossia: $82.216 - 11.966 = 70.25$ e così via.

In realtà, però, per applicare l'ANOVA (e il modello lineare) occorre assicurarsi che: 1) i dati della variabile dipendente (Frequenza, nel nostro caso) siano distribuiti normalmente in tutti i (sette) gruppi: questa è l'ipotesi di normalità; 2) la loro dispersione, in termini di deviazione standard, sia la medesima (cioè $\sigma_1 = \sigma_2 = \dots$): questa è l'ipotesi di omoschedasticità.

A rigore, mancando anche una sola di queste ipotesi, non è possibile applicare l'ANOVA: quello che si otterrebbe in tal caso sarebbe una stima del p-value non attendibile.

Cosa si fa se i dati non sono normali? Si può ricorrere alle varianti non parametriche (test di Kruskal-Wallis, `kruskal.test`, per n campioni indipendenti e test di Wilcoxon/Mann-Whitney, `wilcox.test`, nel confronto tra due gruppi), possiamo tentare di trasformare la risposta con il metodo di Box e Cox, come implementato nel pacchetto MASS:

```
> library(MASS)
```

```
> modello = aov(Frequenza ~ Tempo)
```

> boxcox(modello)

Il grafico della log-verosimiglianza del modello suggerisce di adottare un esponente λ approssimativamente uguale a -0.8 per ottimizzare l'analisi.

Per verificare invece l'ipotesi di omoschedasticità $\sigma_1 = \sigma_2 = \dots$, possiamo ricorrere al test di Bartlett:

> bartlett.test(Frequenza ~ Tempo)

Bartlett test of homogeneity of variances

data: Frequenza by Tempo

Bartlett's K-squared = 23.164, df = 6, p-value = 0.0007433

La brutta notizia qui è rappresentata dal p-value, 0.0007433. Questo significa che dovremmo prendere la decisione che i dati nei sette gruppi sono eteroschedastici (ed in effetti il boxplot iniziale non lascia spazio a dubbi).

Fortunatamente, abbiamo una via di uscita elegante, e la discuteremo nella prossima sezione dedicata ai confronti multipli. Per intanto, ricordiamo che alcuni autori suggeriscono di evitare di utilizzare il test di Bartlett, in quanto sensibile alla mancanza di normalità, e di preferire invece il test robusto di Levene effettuato sulle mediane:

> library(car)

Carico il pacchetto richiesto: carData

> leveneTest(Frequenza, Tempo)

Levene's Test for Homogeneity of Variance (center = median)

	Df	F	value	Pr(>F)
--	----	---	-------	--------

group	6	3.4279	0.002455	**
-------	---	--------	----------	----

609

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

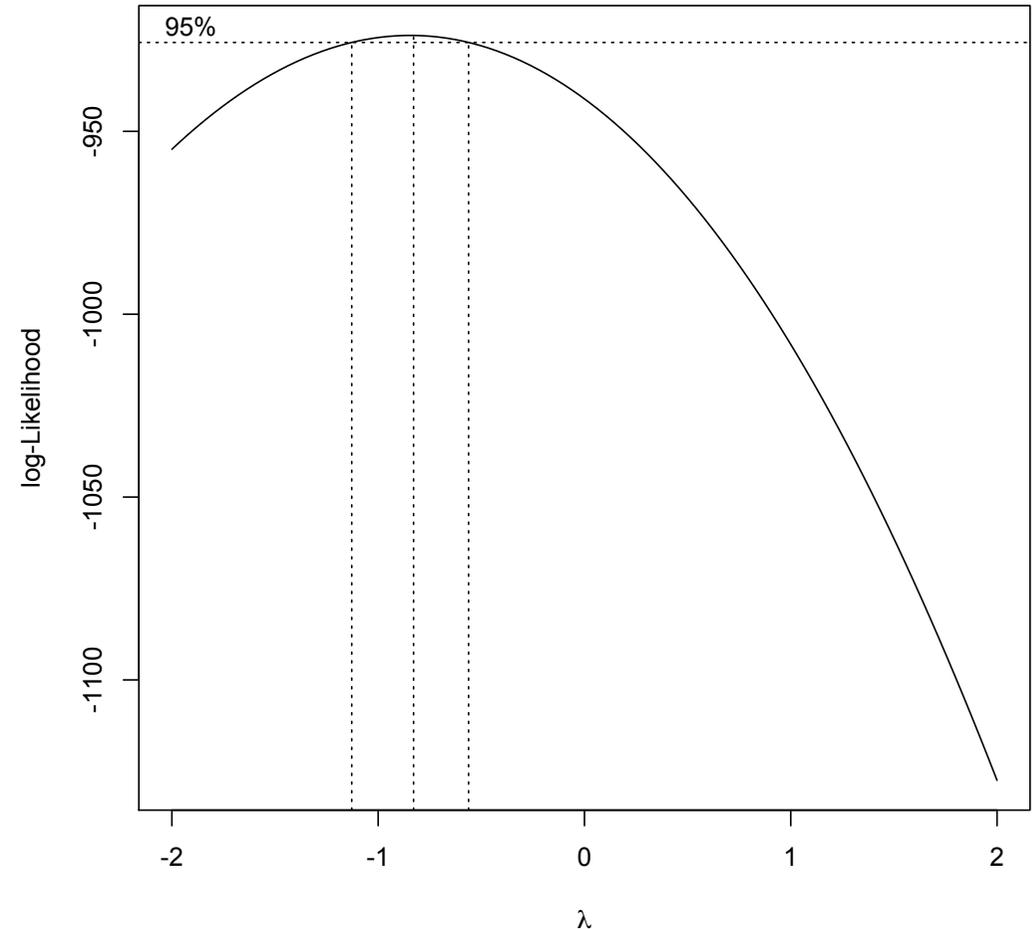
Nell'ipotesi di applicabilità (ricordiamo qui non verificata; poi eseguiremo un test non parametrico) a puro titolo di esempio eseguiamo l'analisi della varianza della Frequenza rispetto ai sette livelli di Tempo (utilizziamo il comando `summary.lm` per ottenere un output che metta in luce anche i cosiddetti effect size del modello):

> modello = aov(Frequenza ~ Tempo)

> summary.lm(modello)

Call:

`aov(formula = Frequenza ~ Tempo)`



Residuals:

Min	1Q	Median	3Q	Max
-31.216	-8.250	-2.114	5.114	76.080

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	82.216	1.476	55.711	< 2e-16 ***
TempoFC_1	-11.966	2.087	-5.733	1.55e-08 ***
TempoFC_2	-18.295	2.087	-8.766	< 2e-16 ***
TempoFC_3	-17.295	2.087	-8.287	7.42e-16 ***
TempoFC_4	-19.250	2.087	-9.224	< 2e-16 ***
TempoFC_5	-17.705	2.087	-8.483	< 2e-16 ***
TempoFC_Dim	-15.330	2.087	-7.345	6.63e-13 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.84 on 609 degrees of freedom

Multiple R-squared: 0.1702, Adjusted R-squared: 0.162

F-statistic: 20.82 on 6 and 609 DF, p-value: < 2.2e-16

Discutiamo punto per punto l'output ottenuto.

Estimate. Vengono calcolate le medie della Frequenza nei sette gruppi:

- FC_0, ossia (Intercept), con una media di 82.216
- FC_1, con una media di $82.216 - 11.966 = 70.25$
- FC_2, con una media di $82.216 - 18.295 = 63.921$
- FC_3, con una media di $82.216 - 17.295 = 64.921$
- FC_4, con una media di $82.216 - 19.250 = 62.966$
- FC_5, con una media di $82.216 - 17.705 = 64.511$
- FC_Dim, con una media di $82.216 - 15.330 = 66.886$

Std. Error. L'errore standard (standard error of the mean), è una misura di inaffidabilità della stima della media.

t value e Pr(> |t|). Il valore t del test si ottiene semplicemente dividendo la stima dei coefficienti rispetto al loro standard error. Ad esempio, per (Intercept), si ha: $82.216/1.476 \approx 55.711$. I p-value che vengono riportati sono valori marginali (ciascuno non tiene in considerazione le correlazioni con gli altri due) e sono relativi alla variabile aleatoria t calcolata, con gli opportuni gradi di libertà, nei sette valori t .

Residual standard error. L'equazione matematica della Anova, per come viene fornito l'output di R, è la seguente: $y_{ij} = \mu_1 + \delta_i + \varepsilon_{ij}$, dove $\mu_1 = 82.216$ è la media del livello FC_0, primo in ordine alfabetico; l'indice i varia da 1 a 7 ed indica i sette livelli del fattore Tempo; i δ_i valgono rispettivamente 0, -11.966, -18.295, -17.295, -19.250, -17.705 e -15.330; ε_{ij} infine rappresentano i residui di ciascun dato $j = 1 \dots 616$, che devono essere distribuiti normalmente, con una media nulla e con una deviazione standard costante. Tale deviazione standard stimata vale 13.84, ed è appunto il residual standard error.

Multiple R-squared. Si tratta del coefficiente di determinazione (il quadrato del coefficiente di correlazione ρ di Pearson).

F-statistic. Si tratta del consuntivo della ANOVA, che fornisce un p-value $< 2.2e-16 \approx 0$. In questo esempio non dobbiamo fare affidamento su questo risultato proprio perché stiamo operando in condizioni di eteroschedasticità (il test di Levene lo ha dimostrato); nel seguito saneremo questa imprecisione.

Diagnostica del modello

Per sapere se la ANOVA è stata eseguita in maniera appropriata è opportuno controllare alcuni dettagli, in maniera visuale.

Con i seguenti comandi otteniamo i quattro grafici mostrati nella pagina:

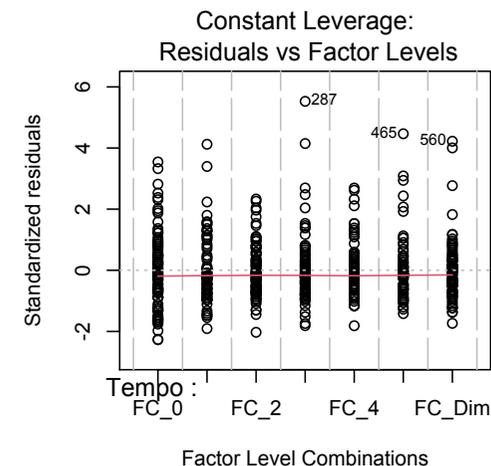
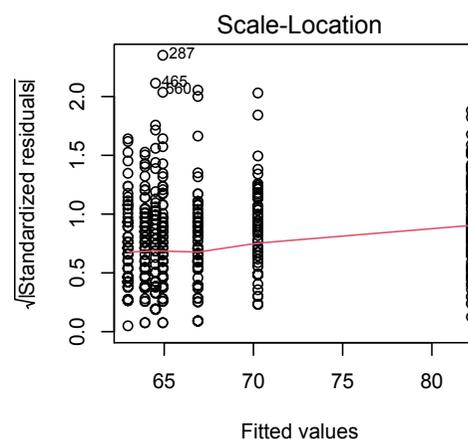
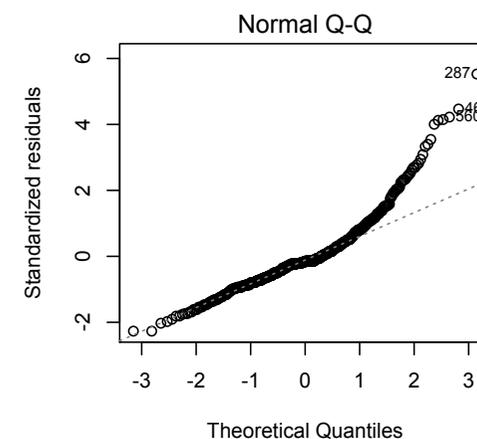
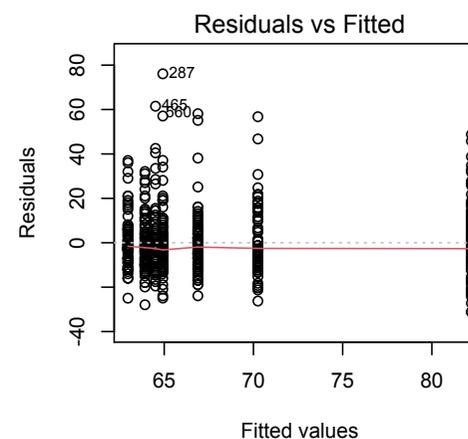
```
> par(mfrow = c(2,2))
```

```
> plot(modello)
```

Nel primo grafico, **Residual vs. Fitted**, osserviamo sette 'cluster' di punti, corrispondenti ai sette livelli di Tempo. Quello che si vuole è che la riga rossa sia allineata allo 0 (media nulla dei residui) e che sia pressoché orizzontale (assenza di drift dei dati).

Si vuole inoltre che i 'cluster' siano approssimativamente della medesima ampiezza: questo non succede, ed è un cattivo segno di eteroschedasticità: abbiamo già detto che nel seguito rimedieremo a questo.

Nel secondo pannello un plot quantile quantile (**Normal Q-Q**) ci dice se, complessivamente, c'è normalità dei residui: abbiamo già accertato all'inizio con il test di Shapiro-Wilk che questo requisito non è soddisfatto. Il terzo pannello **Scale-Location** essenzialmente rispecchia il primo, mentre nel quarto pannello **Residuals vs Leverage** vengono messi in evidenza eventuali punti influenti, o outlier, del modello. Se ve ne fossero, la curva rossa non sarebbe orizzontale, e vi sarebbe qualche punto molto distante dagli altri esternamente ad una curva di tipo iperbolico, che qui non vediamo.



Confronti multipli (post-hoc)

I confronti multipli vengono effettuati per "consolidare" il risultato dell'ANOVA tra i gruppi, indagando su quali coppie di gruppi la differenza è statisticamente significativa.

Abbiamo un'esigenza importante di cui tener conto nella questione dei confronti multipli: non vogliamo correre il rischio di amplificare il 'tasso di errore di primo tipo', ossia quello di rigettare non correttamente l'ipotesi nulla del test, ad un prespecificato livello di significatività α (di solito 0.05).

Spieghiamoci con l'esempio: siccome in Tempo ci sono sette gruppi, si devono fare 21 confronti (FC_0 vs. FC_1; ...; FC_0 vs. FC_Dim; FC_1 vs. FC_2; ...; FC_2 vs. FC_Dim; ...; FC_5 vs. FC_Dim). Questo comporta che, se si sceglie un livello α del 5%, la probabilità di commettere un errore di primo tipo è di circa il 30%:

$$1 - (1 - 5/100) \cdot (1 - 5/100) = 1 - (1 - 5/100)^7 = 0.3016627$$

Le cose peggiorano sensibilmente all'aumentare dei gruppi; ecco spiegata la ragione per la quale è necessario 'aggiustare' le stime dei p-value che si ottengono dai comandi aov e lm.

L'approccio metodologicamente corretto consiste nel considerare il 'test delle differenze significative oneste' di Tukey, implementato nel pacchetto multcomp, avendo cura di utilizzare uno stimatore 'sandwich' della matrice di varianza-covarianza per compensare il problema dell'eteroschedasticità dei dati (vedi documentazione pacchetto sandwich).

Eseguiamo perciò i comandi seguenti:

```
> library(multcomp)
> library(sandwich)
> posthoc = glht(modello, linfct = mcp(Tempo = "Tukey"), vcov = sandwich)
> summary(posthoc)
```

Simultaneous Tests for General Linear Hypotheses					FC_Dim - FC_1 == 0				
Multiple Comparisons of Means: Tukey Contrasts					FC_3 - FC_2 == 0	1.0000	2.0470	0.489	0.99897
Fit: aov(formula = Frequenza ~ Tempo)					FC_4 - FC_2 == 0	-0.9545	1.7258	-0.553	0.99793
Linear Hypotheses:					FC_5 - FC_2 == 0	0.5909	1.8572	0.318	0.99991
	Estimate	Std. Error	t value	Pr(> t)	FC_Dim - FC_2 == 0	2.9659	1.8700	1.586	0.68828
FC_1 - FC_0 == 0	-11.9659	2.3552	-5.081	< 0.001 ***	FC_4 - FC_3 == 0	-1.9545	2.0270	-0.964	0.96074
FC_2 - FC_0 == 0	-18.2955	2.2056	-8.295	< 0.001 ***	FC_5 - FC_3 == 0	-0.4091	2.1400	-0.191	1.00000
FC_3 - FC_0 == 0	-17.2955	2.4484	-7.064	< 0.001 ***	FC_Dim - FC_3 == 0	1.9659	2.1510	0.914	0.96983
FC_4 - FC_0 == 0	-19.2500	2.1870	-8.802	< 0.001 ***	FC_5 - FC_4 == 0	1.5455	1.8351	0.842	0.98004
FC_5 - FC_0 == 0	-17.7045	2.2921	-7.724	< 0.001 ***	FC_Dim - FC_4 == 0	3.9205	1.8480	2.121	0.33776
FC_Dim - FC_0 == 0	-15.3295	2.3025	-6.658	< 0.001 ***	FC_Dim - FC_5 == 0	2.3750	1.9713	1.205	0.89073
FC_2 - FC_1 == 0	-6.3295	1.9345	-3.272	0.01922 *	---				
FC_3 - FC_1 == 0	-5.3295	2.2074	-2.414	0.19170	Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
FC_4 - FC_1 == 0	-7.2841	1.9133	-3.807	0.00282 **	(Adjusted p values reported -- single-step method)				
FC_5 - FC_1 == 0	-5.7386	2.0327	-2.823	0.07154 .					

Con `glht` si effettua un test di ipotesi lineare generale, rispetto ad una funzione lineare di interesse `linfct` che si ottiene dalla matrice dei contrasti parametrici `mcp` calcolata sul fattore Tempo con il metodo di Tukey (vedi testo: Bretz F., Hothorn T., Westfall P. (2010): *Multiple Comparisons Using R*, CRC Press).

Vogliamo solo qui ricordare che se non avessimo avuto il problema dell'eteroschedasticità, il comando da dare sarebbe stato:

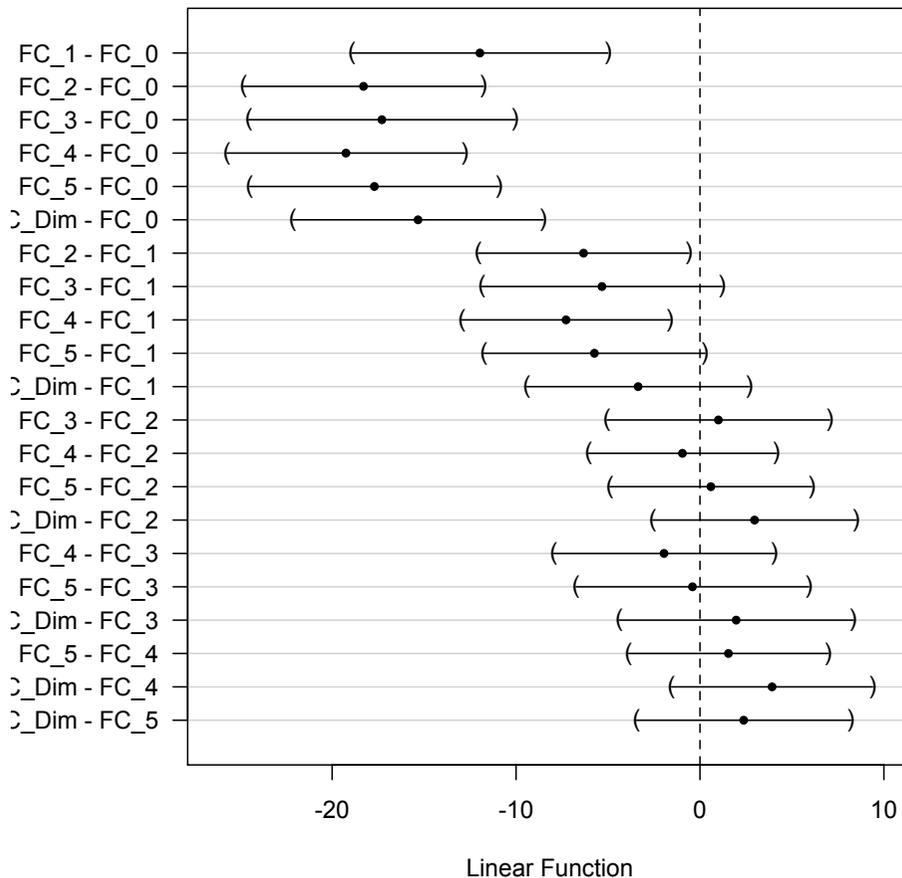
```
posthoc2 = glht(modello, linfct = mcp(Tempo = "Tukey"))
```

e che i confronti multipli con il metodo di Bonferroni si sarebbero ottenuti con il comando:

```
summary(posthoc2, test = adjusted(type = "bonferroni"))
```

Possiamo anche rappresentare, con il comando `plot(posthoc)`, gli intervalli di fiducia relativi ai tre confronti multipli:

95% family-wise confidence level



Analisi della varianza non parametrica

Abbiamo capito che Tempo è un predittore significativo del modello ANOVA, ed abbiamo visto che il livello FC_0 differisce in senso statistico dagli altri livelli.

Eseguiamo ora l'analisi con il metodo corretto Kruskal-Wallis essendo i tempi distribuiti in modo non normale:

```
> kruskal.test(Frequenza ~ Tempo)
```

```
Kruskal-Wallis rank sum test
```

```
data: Frequenza by Tempo
```

```
Kruskal-Wallis chi-squared = 101.008, df = 6, p-value < 2.2e-16
```

Come si vede il p-value risultante è identico al test ANOVA, confermando la varianza tra i gruppi.

Per concludere effettuiamo i test post-hoc con la funzione `dunn.test()`:

```
> dunn.test(Frequenza, Tempo)
```

```
Kruskal-Wallis rank sum test
```

```
data: Frequenza and Tempo
```

```
Kruskal-Wallis chi-squared = 101.0754, df = 6, p-value = 0
```

Comparison of Frequenza by Tempo
(No adjustment)

Col	Mean-I						
Row	Mean	FC_0	FC_1	FC_2	FC_3	FC_4	FC_5
FC_1		4.178665					
		0.0000*					
FC_2		7.472526	3.293861				
		0.0000*	0.0005*				
FC_3		7.502825	3.324159	0.030298			
		0.0000*	0.0004*	0.4879			
FC_4		8.292495	4.113830	0.819969	0.789670		
		0.0000*	0.0000*	0.2061	0.2149		
FC_5		7.336076	3.157411	-0.136449	-0.166748	-0.956418	
		0.0000*	0.0008*	0.4457	0.4338	0.1694	
FC_Dim		5.791907	1.613241	-1.680619	-1.710917	-2.500588	-1.544169
		0.0000*	0.0533	0.0464	0.0435	0.0062*	0.0613

alpha = 0.05

Reject Ho if $p \leq \alpha/2$

Possiamo effettuare anche i test post-hoc con la correzione di Bonferroni, che in questo caso elimina la coppia FC_4-FC_dim:

```
> dunn.test(Frequenza, Tempo, method = c("bonferroni"))
```

Kruskal-Wallis rank sum test

data: Frequenza and Tempo

Kruskal-Wallis chi-squared = 101.0754, df = 6, p-value = 0

Comparison of Frequenza by Tempo
(Bonferroni)

Col Mean-I	FC_0	FC_1	FC_2	FC_3	FC_4	FC_5
FC_1	4.178665 0.0003*					
FC_2	7.472526 0.0000*	3.293861 0.0104*				
FC_3	7.502825 0.0000*	3.324159 0.0093*	0.030298 1.0000			
FC_4	8.292495 0.0000*	4.113830 0.0004*	0.819969 1.0000	0.789670 1.0000		
FC_5	7.336076 0.0000*	3.157411 0.0167*	-0.136449 1.0000	-0.166748 1.0000	-0.956418 1.0000	
FC_Dim	5.791907 0.0000*	1.613241 1.0000	-1.680619 0.9748	-1.710917 0.9145	-2.500588 0.1302	-1.544169 1.0000

alpha = 0.05

Reject Ho if p <= alpha/2