Exampl_t_ANOVA.Rmd

F.A. Barrios

10/10/2020

Examples Chap03

The examples for chapter 3 using data from the heart and estrogen/progestin study (HERS), a clinical trial of hormone therapy (HT) for prevention of recurent heart attacks and death among 2,763 post-menopausal women with existing voronary heart disease (CHD)

Introduction

t-Test example presented in Tabel 3.1 of the t-Test of difference in average glucose by exercise for the women that are not diabetic. These eamples are to revisit som t-test R estimations

```
# setwd("~/Dropbox/Fdo/ClaseStats/RegresionClass/RegresionR_code")
# To set the working directory at the user dir
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(multcomp)
```

```
## Loading required package: mvtnorm
```

```
## Loading required package: survival
```

```
## Loading required package: TH.data
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
##
## Attaching package: 'TH.data'
```

```
## The following object is masked from 'package:MASS':
##
##     geyser
```

```
library(car)
```

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

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```
library(emmeans)

hers <- read_csv("~/Dropbox/Fdo/ClaseStats/RegressionClass/RegressionR_code/DataRegressBook/Chap3/hersda
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   HT = col_character(),
##   raceth = col_character(),
##   nonwhite = col_character(),
##   smoking = col_character(),
##   drinkany = col_character(),
##   exercise = col_character(),
##   physact = col_character(),
##   globrat = col_character(),
##   poorfair = col_character(),
##   htnmeds = col_character(),
##   statins = col_character(),
##   diabetes = col_character(),
##   dmpills = col_character(),
##   insulin = col_character()
## )
## i Use `spec()` for the full column specifications.
```

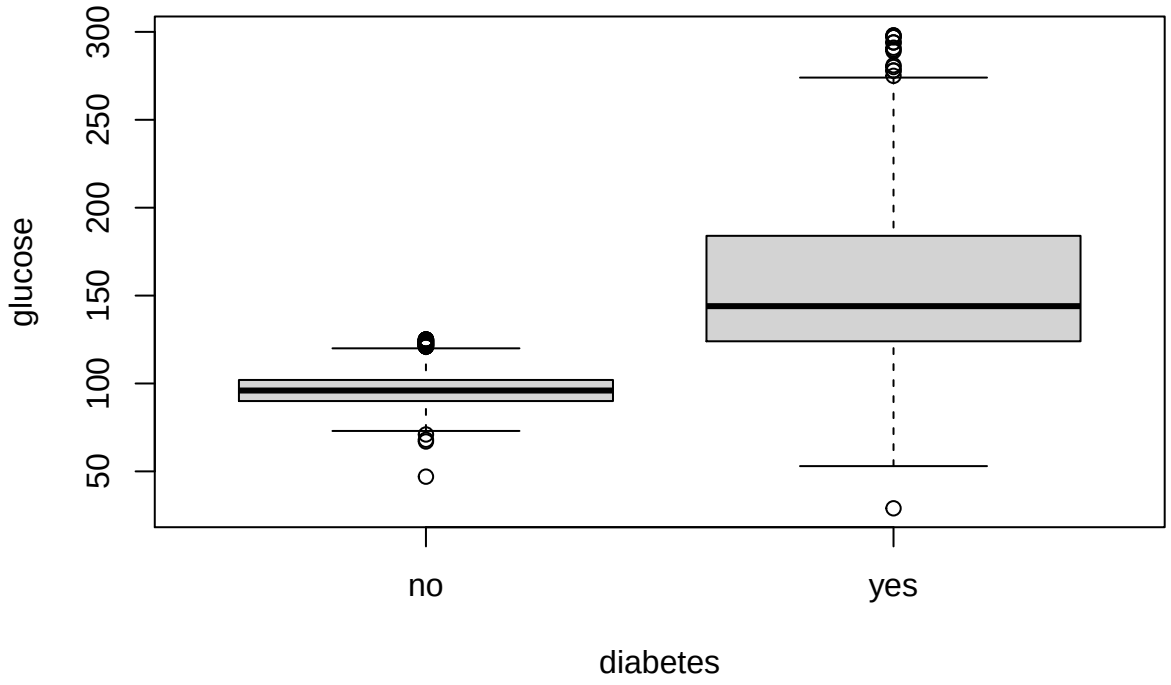
```
# Loading the HERS database in hers varible
summary(hers)
```

```
##       HT                  age           raceth            nonwhite
##  Length:2763        Min.   :44.00   Length:2763        Length:2763
##  Class :character   1st Qu.:62.00   Class :character   Class :character
##  Mode  :character   Median :67.00   Mode  :character   Mode  :character
##                     Mean   :66.65
##                     3rd Qu.:72.00
##                     Max.   :79.00
##
##    smoking            drinkany           exercise           physact
##  Length:2763        Length:2763        Length:2763        Length:2763
##  Class :character   Class :character   Class :character   Class :character
```

```
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    globrat            poorfair            medcond         htnmeds
##  Length:2763        Length:2763        Min.   :0.0000   Length:2763
##  Class :character   Class :character   1st Qu.:0.0000   Class :character
##  Mode  :character   Mode  :character   Median :0.0000   Mode  :character
##                                        Mean   :0.3721
##                                        3rd Qu.:1.0000
##                                        Max.   :1.0000
##
##    statins            diabetes           dmpills            insulin
##  Length:2763        Length:2763        Length:2763        Length:2763
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##      weight            BMI            waist             WHR
##  Min.   : 37.50   Min.   :15.21   Min.   : 56.90   Min.   :0.624
##  1st Qu.: 62.20   1st Qu.:24.64   1st Qu.: 82.00   1st Qu.:0.811
##  Median : 71.00   Median :27.75   Median : 90.50   Median :0.867
##  Mean   : 72.73   Mean   :28.58   Mean   : 91.74   Mean   :0.870
##  3rd Qu.: 81.40   3rd Qu.:31.73   3rd Qu.:100.30   3rd Qu.:0.923
##  Max.   :132.00   Max.   :54.13   Max.   :170.00   Max.   :1.218
##  NA's   :2        NA's   :5       NA's   :2        NA's   :3
##     glucose         weight1           BMI1           waist1
##  Min.   : 29.0   Min.   : 37.70   Min.   :14.73   Min.   : 59.00
##  1st Qu.: 91.0   1st Qu.: 61.20   1st Qu.:24.34   1st Qu.: 81.30
##  Median : 99.0   Median : 70.40   Median :27.54   Median : 90.00
##  Mean   :112.2   Mean   : 72.04   Mean   :28.36   Mean   : 91.12
##  3rd Qu.:114.0   3rd Qu.: 80.90   3rd Qu.:31.54   3rd Qu.:100.00
##  Max.   :298.0   Max.   :142.00   Max.   :54.04   Max.   :142.00
##                  NA's   :150      NA's   :153     NA's   :151
##       WHR1           glucose1          tchol            LDL
##  Min.   :0.6060   Min.   : 42.0   Min.   :110.0   Min.   : 36.8
##  1st Qu.:0.8100   1st Qu.: 91.0   1st Qu.:201.0   1st Qu.:119.6
##  Median :0.8630   Median :100.0   Median :224.0   Median :141.0
##  Mean   :0.8668   Mean   :114.5   Mean   :228.6   Mean   :145.0
##  3rd Qu.:0.9200   3rd Qu.:116.0   3rd Qu.:252.0   3rd Qu.:166.0
##  Max.   :1.1500   Max.   :440.0   Max.   :465.0   Max.   :393.4
##  NA's   :151      NA's   :150     NA's   :4       NA's   :11
##       HDL               TG            tchol1           LDL1
##  Min.   : 14.00   Min.   : 31.0   Min.   : 92.0   Min.   :-20.0
##  1st Qu.: 41.00   1st Qu.:116.0   1st Qu.:193.0   1st Qu.:106.6
##  Median : 49.00   Median :157.0   Median :214.0   Median :128.8
##  Mean   : 50.26   Mean   :166.1   Mean   :219.2   Mean   :132.4
##  3rd Qu.: 57.00   3rd Qu.:208.0   3rd Qu.:242.0   3rd Qu.:154.1
##  Max.   :130.00   Max.   :476.0   Max.   :535.0   Max.   :450.2
##  NA's   :11       NA's   :4       NA's   :150     NA's   :155
##       HDL1              TG1              SBP             DBP
```

```
##  Min.   : 14.00   Min.   :  31.0   Min.   : 83.0   Min.   : 45.00
##  1st Qu.: 42.00   1st Qu.: 119.0   1st Qu.:122.0   1st Qu.: 67.00
##  Median : 50.00   Median : 157.0   Median :134.0   Median : 72.00
##  Mean   : 51.78   Mean   : 175.8   Mean   :135.1   Mean   : 73.15
##  3rd Qu.: 59.00   3rd Qu.: 214.0   3rd Qu.:147.0   3rd Qu.: 80.00
##  Max.   :124.00   Max.   :1016.0   Max.   :224.0   Max.   :102.00
##  NA's   :155      NA's   :150                      NA's   :1
##      age10
##  Min.   :4.400
##  1st Qu.:6.200
##  Median :6.700
##  Mean   :6.665
##  3rd Qu.:7.200
##  Max.   :7.900
##
```

```
boxplot(glucose ~ diabetes, data=hers)
```




```
# For the
boxplot(glucose[hers$diabetes == "no"] ~ exercise[hers$diabetes == "no"], alternative="two.sided", data=
```

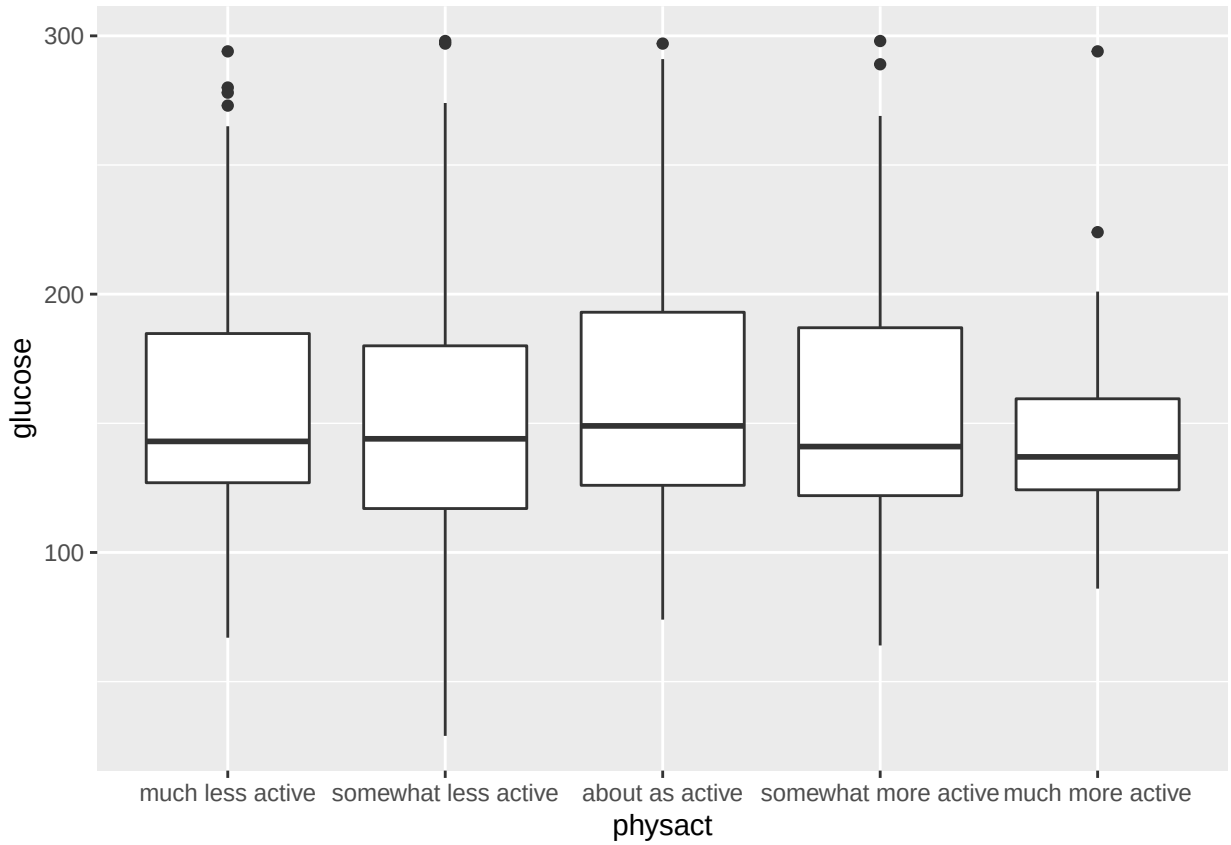

```
t.test(glucose[hers$diabetes == "no"] ~ exercise[hers$diabetes == "no"], data=hers, alternative="two.si
```

```
##
##  Two Sample t-test
##
## data:  glucose[hers$diabetes == "no"] by exercise[hers$diabetes == "no"]
## t = 3.8685, df = 2030, p-value = 0.000113
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.8346242 2.5509539
## sample estimates:
##  mean in group no mean in group yes
##          97.36104          95.66825
```

How it looks using other code

```
#  Example of the HERS data for diabetic participants
hers_yesdi <- filter(hers, diabetes == "yes")
hers_yesdi <- mutate(hers_yesdi, physact = factor(physact, levels=c("much less active","somewhat less a

#  Example of ANOVA with HERS data for diabetic participants
#
ggplot(data = hers_yesdi, mapping = aes(x = physact, y = glucose)) + geom_boxplot(na.rm = TRUE)
```




```
glucose_yesdi_act <- lm(glucose ~ physact, data = hers_yesdi)
Anova(glucose_yesdi_act, type="II")
```

```
## Anova Table (Type II tests)
##
## Response: glucose
##             Sum Sq  Df F value Pr(>F)
## physact      17992   4   1.925 0.1044
## Residuals 1696313 726
```

```
#
S(glucose_yesdi_act)
```

```
## Call: lm(formula = glucose ~ physact, data = hers_yesdi)
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   155.789      5.095  30.575   <2e-16 ***
## physactsomewhat less active    -4.590      6.235  -0.736    0.462
## physactabout as active          5.191      5.958   0.871    0.384
## physactsomewhat more active    -1.398      6.362  -0.220    0.826
## physactmuch more active       -11.789      8.320  -1.417    0.157
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard deviation: 48.34 on 726 degrees of freedom
## Multiple R-squared: 0.0105
## F-statistic: 1.925 on 4 and 726 DF,  p-value: 0.1044
```

```
##     AIC     BIC
## 7751.41 7778.98
```

```
glucose_emmeans <- emmeans(glucose_yesdi_act, "physact")
contrast(glucose_emmeans, adjust="sidak")
```

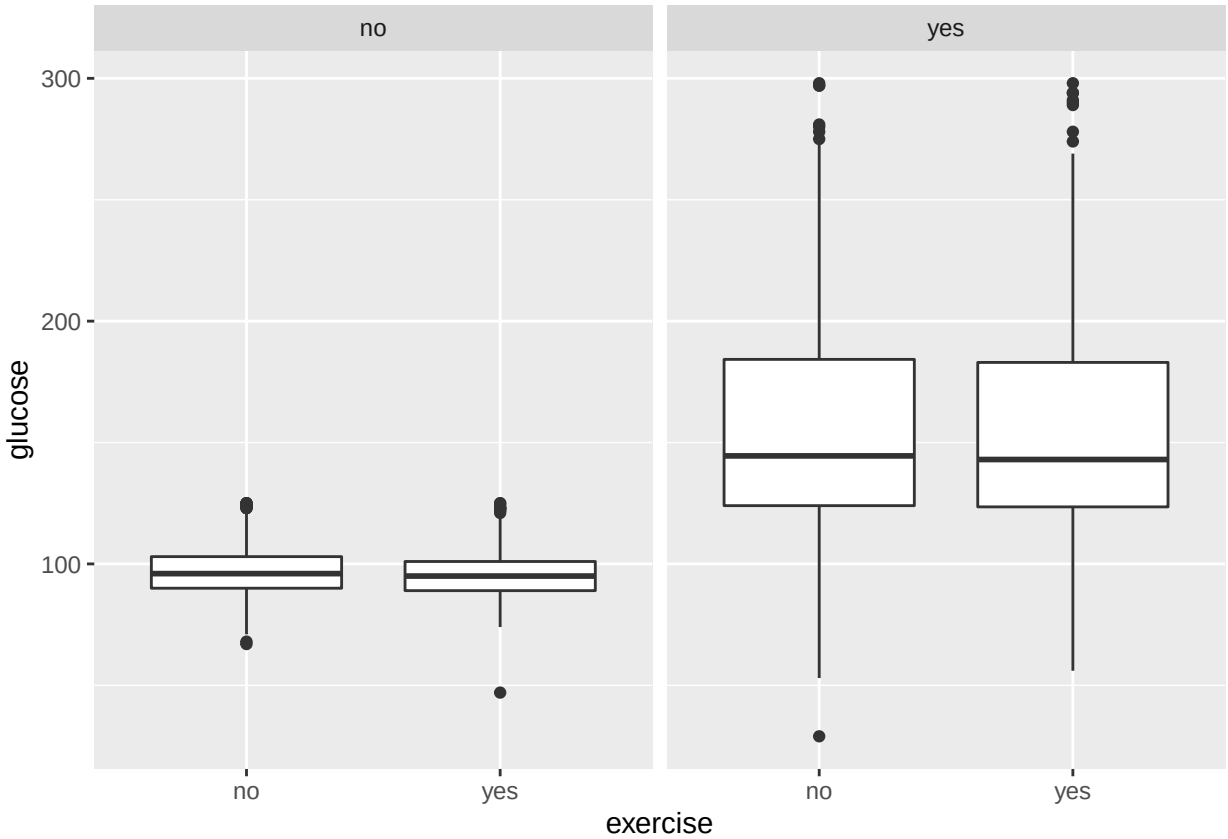
```
##  contrast                    estimate   SE  df t.ratio p.value
##  much less active effect         2.52 4.45 726  0.565  0.9856
##  somewhat less active effect    -2.07 3.46 726 -0.599  0.9815
##  about as active effect          7.71 3.16 726  2.441  0.0722
##  somewhat more active effect     1.12 3.60 726  0.311  0.9991
##  much more active effect        -9.27 5.50 726 -1.687  0.3830
##
## P value adjustment: sidak method for 5 tests
```

Including Plots

```
summary(hers$diabetes)
```

```
##    Length     Class      Mode
##      2763 character character
```

```
ggplot(data = hers, mapping = aes(x = exercise, y = glucose)) + geom_boxplot() + facet_grid( . ~ diabet
```



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

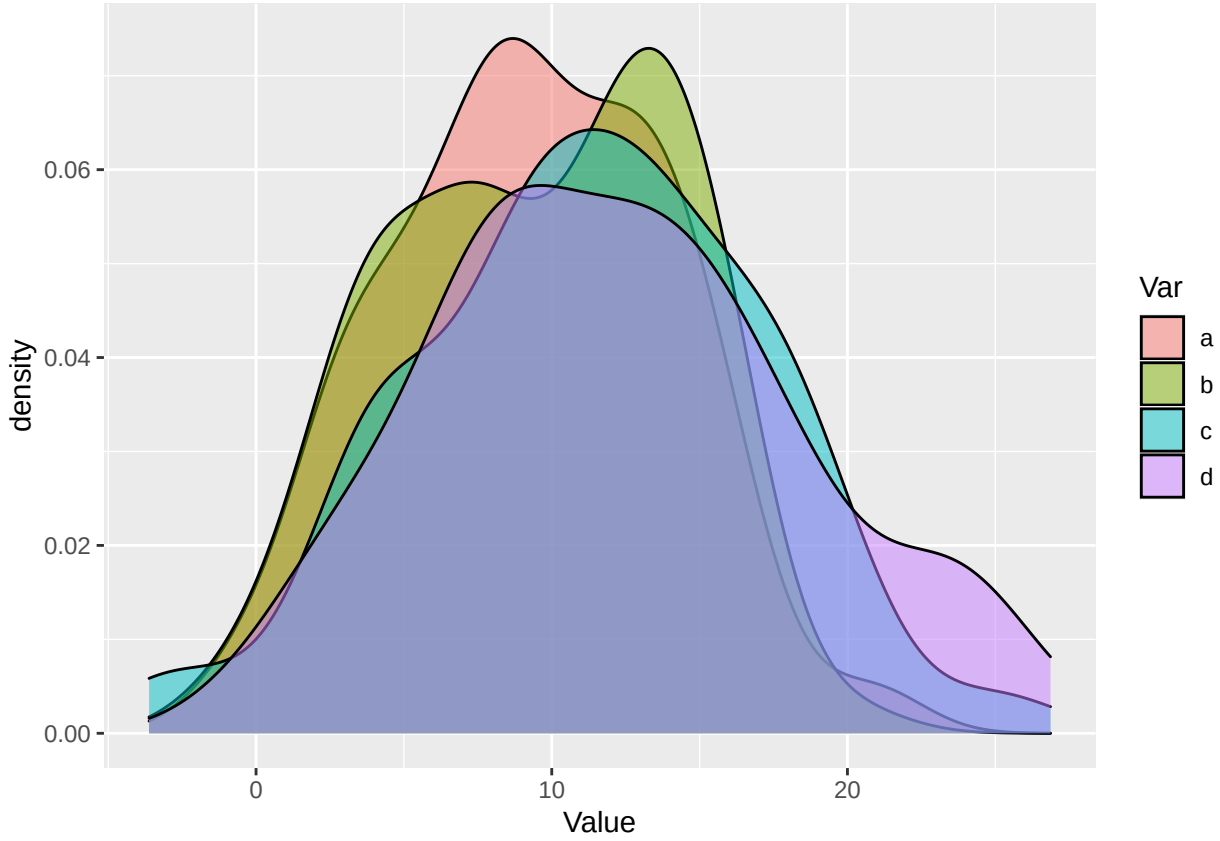
Example from R-bloggers

First we build four random variables with two different distributions.

```
# Create the four groups
set.seed(10)
df1 <- data.frame(Var="a", Value=rnorm(100,10,5))
df2 <- data.frame(Var="b", Value=rnorm(100,10,5))
df3 <- data.frame(Var="c", Value=rnorm(100,11,6))
df4 <- data.frame(Var="d", Value=rnorm(100,11,6))

# merge them in one data frame
df<-rbind(df1,df2,df3,df4)

# convert Var to a factor
df$Var<-as.factor(df$Var)
df%>%ggplot(aes(x=Value, fill=Var))+geom_density(alpha=0.5)
```



The ANOVA (taken from R-bloggers) ANOVA (ANalysis Of VAriance) is a statistical test used to compare two or more groups to see if they are significantly different. The ANOVA model and some examples. The null hypothesis in ANOVA is that there is no difference between means and the alternative is that the means are not all equal. This means that when we are dealing with many groups, we cannot compare them pairwise. We can simply answer if the means between groups can be considered as equal or not.

```
# ANOVA
model1<-lm(Value~Var, data=df)
anova(model1)
```

```
## Analysis of Variance Table
##
```

```
## Response: Value
##             Df  Sum Sq Mean Sq F value    Pr(>F)
## Var          3   565.7 188.565   6.351 0.0003257 ***
## Residuals 396 11757.5  29.691
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Tukey multiple comparisons

What about if we want to compare all the groups pairwise? In this case, we can apply the Tukey's HSD which is a single-step multiple comparison procedure and statistical test, Tukey's Honest Significant Difference (Tukey's HSD). It can be used to find means that are significantly different from each other.

```
summary(glht(model1, mcp(Var="Tukey")))
```

```
##
##   Simultaneous Tests for General Linear Hypotheses
##
## Multiple Comparisons of Means: Tukey Contrasts
##
##
## Fit: lm(formula = Value ~ Var, data = df)
##
## Linear Hypotheses:
##             Estimate Std. Error t value Pr(>|t|)
## b - a == 0   0.2079     0.7706   0.270  0.99312
## c - a == 0   1.8553     0.7706   2.408  0.07727 .
## d - a == 0   2.8758     0.7706   3.732  0.00129 **
## c - b == 0   1.6473     0.7706   2.138  0.14298
## d - b == 0   2.6678     0.7706   3.462  0.00329 **
## d - c == 0   1.0205     0.7706   1.324  0.54795
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- single-step method)
```