One-way ANOVA
=============

This R markdown document provides an example of performing a regression using the lm() function in R and compares the output with the jmv::ANOVA() function in the jmv (Jamovi) package.

Package management in R

```r
# keep a list of the packages used in this script
packages <- c("tidyverse","rio","jmv")
```

This next code block has eval=FALSE because you don’t want to run it when knitting the file. Installing packages when knitting an R notebook can be problematic.

```r
# check each of the packages in the list and install them if they're not installed already
for (i in packages){
  if(! i %in% installed.packages()){
    install.packages(i,dependencies = TRUE)
  }
  # show each package that is checked
  print(i)
}
```

```r
# load each package into memory so it can be used in the script
for (i in packages){
  library(i,character.only=TRUE)
  # show each package that is loaded
  print(i)
}
```

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

    ## v ggplot2 3.3.3     v purrr   0.3.4
    ## v tibble  3.0.6     v dplyr   1.0.4
    ## v tidyr   1.1.2     v stringr 1.4.0
    ## v readr   1.4.0     v forcats 0.5.1

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

    ## [1] "tidyverse"
    ## [1] "rio"
    ## [1] "jmv"
```

ANOVA is a linear model

The ANOVA is a type of linear model. We're going to compare the output from the lm() function in R with ANOVA output.To use a categorical variable in a linear model it needs to be dummy coded. One group needs to be coded as 0 and the other group needs to be coded as 1.If you compare the values for F from lm() and t from the t-test you'll see that t^2 = F. You should also notice that the associated p values are equal.

Open data file

The rio package works for importing several different types of data files. We're going to use it in this class. There are other packages which can be used to open datasets in R. You can see several options by clicking on the Import Dataset menu under the Environment tab in RStudio. (For a csv file like we have this week we'd use either From Text(base) or From Text (readr). Try it out to see the menu dialog.)

```r
# Using the file.choose() command allows you to select a file to import from another folder.
dataset <- rio::import(file.choose())
# This command will allow us to import a file included in our project folder.
# dataset <- rio::import("Album Sales.sav")
```

Get R code from Jamovi output

You can get the R code for most of the analyses you do in Jamovi.

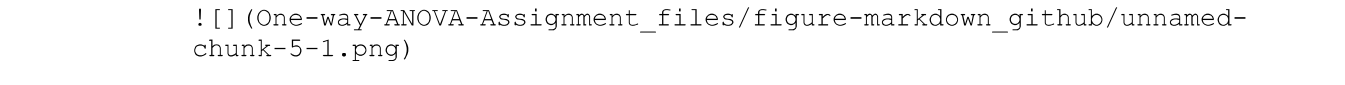
1. Click on the three vertical dots at the top right of the Jamovi window.
2. Click on the Syndax mode check box at the bottom of the Results section.
3. Close the Settings window by clicking on the Hide Settings arrow at the top right of the settings menu.
4. you should now see the R code for each of the analyses you just ran.

lm() function in R

Many linear models are calculated in R using the lm() function. We'll look at how to perform a regression using the lm() function since it's so common.

Visualization

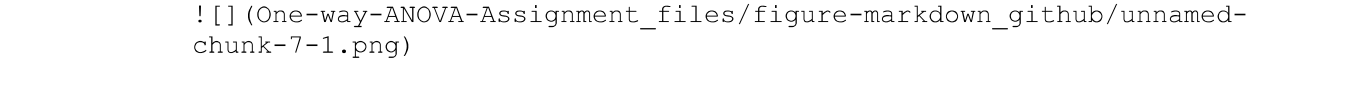
```r
ggplot(dataset, aes(x = Happiness))+
  geom_histogram(binwidth = 1, color = "black", fill = "white")+
  facet_grid(Dose ~ .)
```



```r
# Make a factor for the box plot
dataset <- dataset %>% mutate(Dose_f = as.factor(Dose))
levels(dataset$Dose_f)
```

```
## [1] "1" "2" "3"
```

```r
ggplot(dataset, aes(x = Dose_f, y = Happiness)) +
  geom_boxplot()
```



Dummy codes

To use a categorical variable in a regression model, the categorical variables need to be dummy coded. Here’s a nice post describing some different methods of creating dummy code variables.
https://www.marsja.se/create-dummy-variables-in-r/

You basically need 1 fewer dummy code variables than the number of categories in the original variable. If your original variable has 2 categories (like male and female in sex), then you need 1 dummy code variable. The dummy code variables need to be coded using 0 and 1. You need to pick once of the categories to be the reference category. That will get coded with a 0.

Since our example variable Dose has 3 categories, we’ll create 2 variables as dummy variables. We’ll use control as the reference category. We’ll create a dummy variable for the 15 minute group and a dummy variable for the 30 minute group where individuals in those groups will get a 1 if they’re in that group and a 0 if they’re not.

```r
dataset$d15min <- ifelse(dataset$Dose == 2, 1, 0)
dataset$d30min <- ifelse(dataset$Dose == 3, 1, 0)
```

Computation

```r
model <- lm(formula = Happiness ~ d15min + d30min, data = dataset)
model
```

```
##
## Call:
## lm(formula = Happiness ~ d15min + d30min, data = dataset)
```

```
##
## Coefficients:
## (Intercept)       d15min       d30min
##         2.2          1.0          2.8
```

Model assessment

``` r
summary(model)
```

```
##
## Call:
## lm(formula = Happiness ~ d15min + d30min, data = dataset)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##   -2.0   -1.2   -0.2    0.9    2.0
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.2000     0.6272   3.508  0.00432 **
## d15min        1.0000     0.8869   1.127  0.28158
## d30min        2.8000     0.8869   3.157  0.00827 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.402 on 12 degrees of freedom
## Multiple R-squared:  0.4604, Adjusted R-squared:  0.3704
## F-statistic: 5.119 on 2 and 12 DF,  p-value: 0.02469
```

Standardized residuals from lm()

You might notice lm() does not provide the standardized residuals. Those must me calculated separately.

``` r
standardized = lm(scale(Happiness) ~ scale(d15min) + scale(d30min),
data=dataset)
summary(standardized)
```

```
##
## Call:
## lm(formula = scale(Happiness) ~ scale(d15min) + scale(d30min),
##     data = dataset)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1316 -0.6790 -0.1132  0.5092  1.1316
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -1.541e-16  2.049e-01   0.000  1.00000
## scale(d15min)  2.761e-01  2.449e-01   1.127  0.28158
## scale(d30min)  7.730e-01  2.449e-01   3.157  0.00827 **
```

```
    ## ---
    ## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
    ##
    ## Residual standard error: 0.7935 on 12 degrees of freedom
    ## Multiple R-squared:  0.4604, Adjusted R-squared:  0.3704
    ## F-statistic: 5.119 on 2 and 12 DF,  p-value: 0.02469
```

function in Jamovi

Compare the output from the lm() function with the output from the function in the jmv package.

```r
jmv::ANOVA(
  formula = Happiness ~ Dose,
  data = dataset,
  effectSize = c("partEta", "omega"),
  homo = TRUE,
  norm = TRUE,
  qq = TRUE,
  postHoc = ~ Dose,
  postHocCorr = c("tukey", "holm"),
  postHocES = "d")
```

```
    ##
    ##  ANOVA
    ##
    ##  ANOVA - Happiness
    ##
-------------------------------------------------------------------------------------------------
    ##                 Sum of Squares    df    Mean Square    F           p
<U+03B7>²p          <U+03C9>²
    ##
-------------------------------------------------------------------------------------------------
    ##    Dose               20.13333     2      10.066667    5.118644
0.0246943    0.4603659    0.3544858
    ##    Residuals          23.60000    12       1.966667
    ##
-------------------------------------------------------------------------------------------------
    ##
    ##
    ##  ASSUMPTION CHECKS
    ##
    ##  Homogeneity of Variances Test (Levene's)
    ##  ----------------------------------------
    ##    F             df1    df2    p
    ##  ----------------------------------------
    ##    0.09169054      2     12    0.9130204
    ##  ----------------------------------------
    ##
    ##
    ##  Normality Test (Shapiro-Wilk)
    ##  -----------------------------
    ##    Statistic    p
```

```
    ##  ------------------------------
    ##    0.9166916    0.1714696
    ##  ------------------------------
    ##
    ##
    ##  POST HOC TESTS
    ##
    ##  Post Hoc Comparisons - Dose
    ##
-------------------------------------------------------------------------------------------------
    ##    Dose          Dose    Mean Difference    SE           df          t
p-tukey      p-holm       Cohen's d
    ##
-------------------------------------------------------------------------------------------------
    ##    1       -     2             -1.000000    0.8869423    12.00000
-1.127469    0.5162761    0.2815839    0.2911113
    ##            -     3             -2.800000    0.8869423    12.00000
-3.156913    0.0209244    0.0248043    0.8151115
    ##    2       -     3             -1.800000    0.8869423    12.00000
-2.029444    0.1474576    0.1303844    0.5240003
    ##
-------------------------------------------------------------------------------------------------
```