Contact Experiment Data Analysis

First Step is to load the necessary package, If you dont have them just install them. For jjstatsplot you need to install it remotely. Just remove the dash and press enter. Then Press 3 (none package to be updated). ?group\_by()

#remotes::install\_github("sbalci/jjstatsplot") #Press 3 !!!!! i.e., installing/Updating none package!   
library(jmv)  
library(datasets)  
library(plyr)  
library(readr)  
library(dataframes2xls)  
library(data.table)  
library(plyr)  
library(ggstatsplot)

## Registered S3 method overwritten by 'broom.mixed':  
## method from   
## tidy.gamlss broom

## Registered S3 methods overwritten by 'lme4':  
## method from  
## cooks.distance.influence.merMod car   
## influence.merMod car   
## dfbeta.influence.merMod car   
## dfbetas.influence.merMod car

## In case you would like cite this package, cite it as:  
## Patil, I. (2018). ggstatsplot: "ggplot2" Based Plots with Statistical Details. CRAN.  
## Retrieved from https://cran.r-project.org/web/packages/ggstatsplot/index.html

library(jjstatsplot)  
library(lme4)

## Loading required package: Matrix

library(lmerTest)

##   
## Attaching package: 'lmerTest'

## The following object is masked from 'package:lme4':  
##   
## lmer

## The following object is masked from 'package:stats':  
##   
## step

library(ggplot2)  
library(rstatix)

##   
## Attaching package: 'rstatix'

## The following objects are masked from 'package:plyr':  
##   
## desc, mutate

## The following object is masked from 'package:stats':  
##   
## filter

library(coin)

## Loading required package: survival

##   
## Attaching package: 'coin'

## The following objects are masked from 'package:rstatix':  
##   
## chisq\_test, friedman\_test, kruskal\_test, sign\_test, wilcox\_test

library(ARTool)  
library(ggpubr)

##   
## Attaching package: 'ggpubr'

## The following object is masked from 'package:plyr':  
##   
## mutate

library(tidyverse)

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

## Found more than one class "atomicVector" in cache; using the first, from namespace 'Matrix'

## Also defined by 'Rmpfr'

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

## v tibble 3.0.4 v dplyr 1.0.2  
## v tidyr 1.1.2 v stringr 1.4.0  
## v purrr 0.3.4 v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::arrange() masks plyr::arrange()  
## x dplyr::between() masks data.table::between()  
## x purrr::compact() masks plyr::compact()  
## x dplyr::count() masks plyr::count()  
## x tidyr::expand() masks Matrix::expand()  
## x dplyr::failwith() masks plyr::failwith()  
## x dplyr::filter() masks rstatix::filter(), stats::filter()  
## x dplyr::first() masks data.table::first()  
## x dplyr::id() masks plyr::id()  
## x dplyr::lag() masks stats::lag()  
## x dplyr::last() masks data.table::last()  
## x dplyr::mutate() masks ggpubr::mutate(), rstatix::mutate(), plyr::mutate()  
## x tidyr::pack() masks Matrix::pack()  
## x dplyr::rename() masks plyr::rename()  
## x dplyr::summarise() masks plyr::summarise()  
## x dplyr::summarize() masks plyr::summarize()  
## x purrr::transpose() masks data.table::transpose()  
## x tidyr::unpack() masks Matrix::unpack()

library(dplyr)  
library("afex")

## \*\*\*\*\*\*\*\*\*\*\*\*  
## Welcome to afex. For support visit: http://afex.singmann.science/

## - Functions for ANOVAs: aov\_car(), aov\_ez(), and aov\_4()  
## - Methods for calculating p-values with mixed(): 'KR', 'S', 'LRT', and 'PB'  
## - 'afex\_aov' and 'mixed' objects can be passed to emmeans() for follow-up tests  
## - NEWS: library('emmeans') now needs to be called explicitly!  
## - Get and set global package options with: afex\_options()  
## - Set orthogonal sum-to-zero contrasts globally: set\_sum\_contrasts()  
## - For example analyses see: browseVignettes("afex")  
## \*\*\*\*\*\*\*\*\*\*\*\*

##   
## Attaching package: 'afex'

## The following object is masked from 'package:lme4':  
##   
## lmer

library("emmeans")   
library("multcomp")

## Loading required package: mvtnorm

## Loading required package: TH.data

## Loading required package: MASS

##   
## Attaching package: 'MASS'

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

## The following object is masked from 'package:rstatix':  
##   
## select

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

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

library(tinytex)

Import and merge all the csv files (x.csv where x = ID) of the folder. Note, that the ID is has already been added as column

Discard first X trials per interpenetration feedback condition and then create a summary table for each participant. You need to define **nrTrialsPerBlockToRemove**.

data$Part <- as.factor(data$Block < 4)  
levels(data$Part)

## [1] "FALSE" "TRUE"

data$Part <- factor(data$Part,levels = c("TRUE","FALSE"),  
 labels = c("Part 1","Part 2"))  
  
nrTrialsPerBlockToRemove <- 1  
#trialsToRemove <- seq(from = 1, to = nrTrialsPerBlockToRemove)  
  
 data <- data %>%  
 group\_by(ID, Part, InterpenetrationFeedback, FullyShaded) %>% # I have added here the fully shaded   
 slice(nrTrialsPerBlockToRemove+1:n())  
 # to double check we are discarding the right rows  
 #print(data[[i]]$Trial)  
  
 # This df will be used to create the subsets for 1st part and 2nd part of the experiment.   
   
data$InterpenetrationFeedback <- as.factor(data$InterpenetrationFeedback)  
data$FullyShaded <- as.factor(data$FullyShaded)

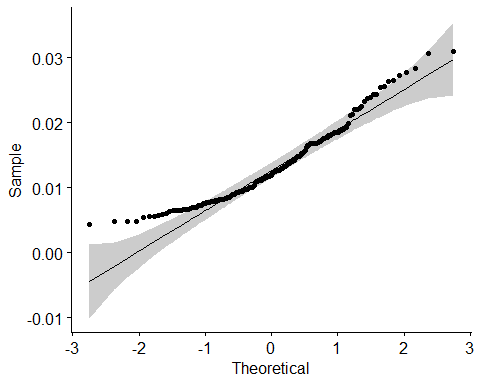
ParsiDF <- data  
  
  
# Exclude the IDs which produced the extreme values (i.e., = or > 3 coefficients from the mean)  
ParsiDF$ID[ParsiDF$ID == 9] <- NA   
ParsiDF$ID[ParsiDF$ID == 17] <- NA  
ParsiDF$ID[ParsiDF$ID == 20] <- NA  
  
ParsiDF <- na.omit(ParsiDF)  
  
ParsiDF <- aggregate(. ~ ID + Age + Gender + InterpenetrationFeedback + Part, ParsiDF, mean)  
  
#Before Conversion to logarithms (showing the abnormal distribution)  
shapiro\_test(ParsiDF$MaxInterpenetration)

## # A tibble: 1 x 3  
## variable statistic p.value  
## <chr> <dbl> <dbl>  
## 1 ParsiDF$MaxInterpenetration 0.937 0.000000946

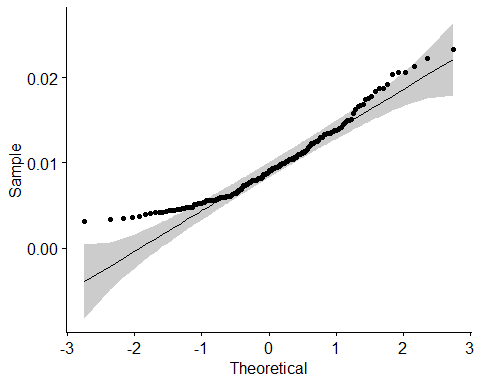
shapiro\_test(ParsiDF$AverageInterpenetration)

## # A tibble: 1 x 3  
## variable statistic p.value  
## <chr> <dbl> <dbl>  
## 1 ParsiDF$AverageInterpenetration 0.936 0.000000844

ggqqplot(ParsiDF$MaxInterpenetration)



ggqqplot(ParsiDF$AverageInterpenetration)



ParsiDF %>%  
 group\_by(InterpenetrationFeedback, Part) %>%  
 shapiro\_test(MaxInterpenetration)

## # A tibble: 8 x 5  
## InterpenetrationFeedback Part variable statistic p  
## <fct> <fct> <chr> <dbl> <dbl>  
## 1 Both Part 1 MaxInterpenetration 0.946 0.290   
## 2 Both Part 2 MaxInterpenetration 0.834 0.00225  
## 3 Electrotactile Part 1 MaxInterpenetration 0.967 0.677   
## 4 Electrotactile Part 2 MaxInterpenetration 0.859 0.00611  
## 5 NoFeedback Part 1 MaxInterpenetration 0.969 0.720   
## 6 NoFeedback Part 2 MaxInterpenetration 0.977 0.872   
## 7 Visual Part 1 MaxInterpenetration 0.860 0.00622  
## 8 Visual Part 2 MaxInterpenetration 0.898 0.0327

ParsiDF %>%  
 group\_by(InterpenetrationFeedback, Part) %>%  
 shapiro\_test(AverageInterpenetration)

## # A tibble: 8 x 5  
## InterpenetrationFeedback Part variable statistic p  
## <fct> <fct> <chr> <dbl> <dbl>  
## 1 Both Part 1 AverageInterpenetration 0.947 0.301   
## 2 Both Part 2 AverageInterpenetration 0.827 0.00175  
## 3 Electrotactile Part 1 AverageInterpenetration 0.964 0.592   
## 4 Electrotactile Part 2 AverageInterpenetration 0.877 0.0128   
## 5 NoFeedback Part 1 AverageInterpenetration 0.964 0.598   
## 6 NoFeedback Part 2 AverageInterpenetration 0.955 0.420   
## 7 Visual Part 1 AverageInterpenetration 0.851 0.00445  
## 8 Visual Part 2 AverageInterpenetration 0.914 0.0655

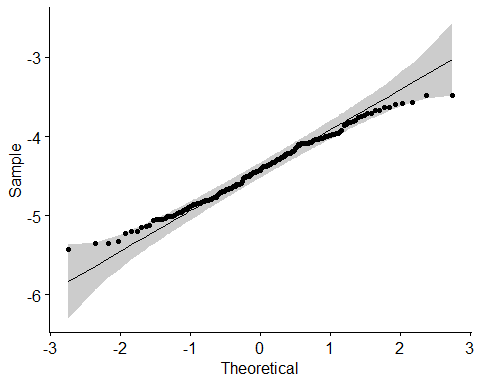
#After Conversion of the performance variables into logs (Normal Distribution)  
ParsiDF$MaxInterpenetration <- log(ParsiDF$MaxInterpenetration)  
  
ParsiDF$AverageInterpenetration <- log(ParsiDF$AverageInterpenetration)  
  
shapiro\_test(ParsiDF$MaxInterpenetration)

## # A tibble: 1 x 3  
## variable statistic p.value  
## <chr> <dbl> <dbl>  
## 1 ParsiDF$MaxInterpenetration 0.988 0.149

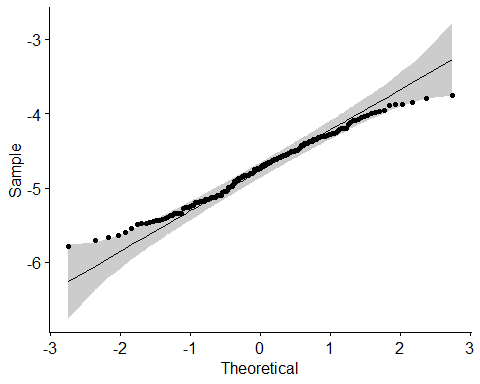
shapiro\_test(ParsiDF$AverageInterpenetration)

## # A tibble: 1 x 3  
## variable statistic p.value  
## <chr> <dbl> <dbl>  
## 1 ParsiDF$AverageInterpenetration 0.986 0.0873

ggqqplot(ParsiDF$MaxInterpenetration)



ggqqplot(ParsiDF$AverageInterpenetration)



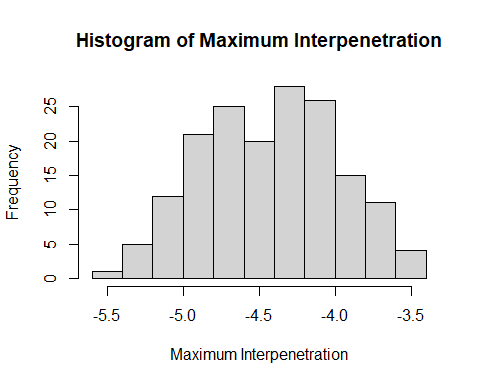
#Let's check the assumption for each interpenetration feedback and shade condition  
ParsiDF %>%  
 group\_by(InterpenetrationFeedback, Part) %>%  
 shapiro\_test(MaxInterpenetration)

## # A tibble: 8 x 5  
## InterpenetrationFeedback Part variable statistic p  
## <fct> <fct> <chr> <dbl> <dbl>  
## 1 Both Part 1 MaxInterpenetration 0.987 0.989  
## 2 Both Part 2 MaxInterpenetration 0.966 0.646  
## 3 Electrotactile Part 1 MaxInterpenetration 0.931 0.143  
## 4 Electrotactile Part 2 MaxInterpenetration 0.975 0.831  
## 5 NoFeedback Part 1 MaxInterpenetration 0.958 0.468  
## 6 NoFeedback Part 2 MaxInterpenetration 0.951 0.352  
## 7 Visual Part 1 MaxInterpenetration 0.963 0.582  
## 8 Visual Part 2 MaxInterpenetration 0.976 0.851

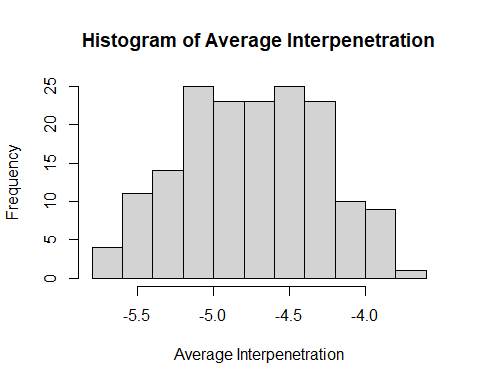
ParsiDF %>%  
 group\_by(InterpenetrationFeedback, Part) %>%  
 shapiro\_test(AverageInterpenetration)

## # A tibble: 8 x 5  
## InterpenetrationFeedback Part variable statistic p  
## <fct> <fct> <chr> <dbl> <dbl>  
## 1 Both Part 1 AverageInterpenetration 0.976 0.854  
## 2 Both Part 2 AverageInterpenetration 0.970 0.740  
## 3 Electrotactile Part 1 AverageInterpenetration 0.954 0.396  
## 4 Electrotactile Part 2 AverageInterpenetration 0.979 0.906  
## 5 NoFeedback Part 1 AverageInterpenetration 0.956 0.434  
## 6 NoFeedback Part 2 AverageInterpenetration 0.977 0.873  
## 7 Visual Part 1 AverageInterpenetration 0.955 0.428  
## 8 Visual Part 2 AverageInterpenetration 0.964 0.600

hist(ParsiDF$MaxInterpenetration,main = paste("Histogram of Maximum Interpenetration") , xlab = "Maximum Interpenetration")

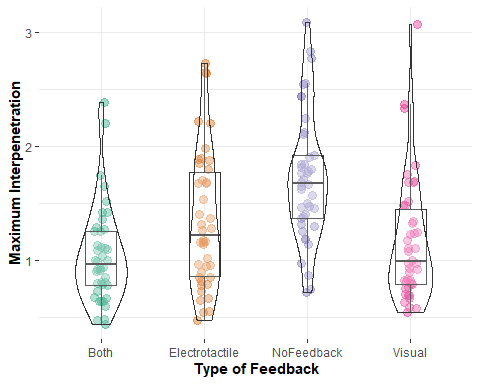


hist(ParsiDF$AverageInterpenetration, main = paste("Histogram of Average Interpenetration") , xlab = "Average Interpenetration")

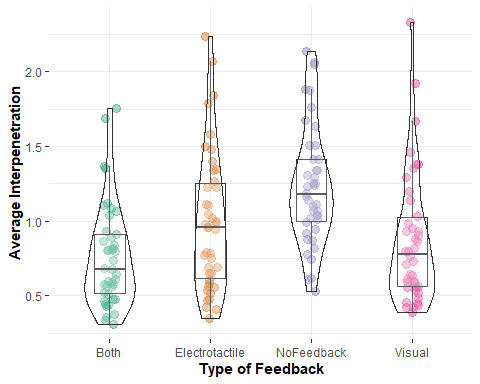


Let’s visualize the data per interpenetration feedback and/or part of the experiment (part 1 & part 2).

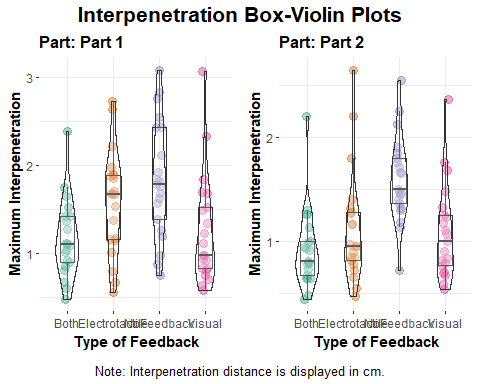
ParsiDFplots <- data # A dataframe just for the plots, so we show everything in real numbers and in centimeters!  
ParsiDFplots$ID[ParsiDFplots$ID == 9] <- NA  
ParsiDFplots$ID[ParsiDFplots$ID == 17] <- NA  
ParsiDFplots$ID[ParsiDFplots$ID == 20] <- NA  
ParsiDFplots <- na.omit(ParsiDFplots)  
ParsiDFplots <- aggregate(. ~ ID + Age + Gender + InterpenetrationFeedback + Part, ParsiDFplots, mean)  
  
ParsiDFplots$AverageInterpenetration <-100 \* ParsiDFplots$AverageInterpenetration #Converting meters to centimeters  
ParsiDFplots$MaxInterpenetration <- 100 \* ParsiDFplots$MaxInterpenetration #Converting meters to centimeters  
  
p1 <- ggstatsplot::ggbetweenstats(  
 data = ParsiDFplots,  
 x = "InterpenetrationFeedback", #Indepedent Variable  
 y = "MaxInterpenetration", # Depedent Variable  
 grouping.var = "Part", # 2nd IV   
 type = "p", # parametric test i.e., p values  
 pairwise.comparisons = FALSE, #compute pairwise comparisons  
 pairwise.display = "significant", # show only the significant ones  
 p.adjust.method = "bonferroni", # correction of p-value  
 effsize.type = "unbiased", # Calculates the Hedge's g for t tests and the partial Omega for ANOVA  
 results.subtitle = FALSE,  
 xlab = "Type of Feedback", #label of X axis  
 ylab = "Maximum Interpenetration", #label of y axis  
 sample.size.label = FALSE,  
 var.equal = TRUE, #Assuming Equal variances  
 mean.plotting = FALSE,  
 mean.ci = TRUE, #display the confidence interval of the mean  
 paired = TRUE, #indicating that we have a within subject design  
 title.text = "Interpenetration Box-Violin Plots",  
 caption.text = "Note: Interpenetration distance is displayed in cm.",  
 title.color = "black",  
 caption.color = "black"  
 )   
  
p2 <- ggstatsplot::ggbetweenstats(  
 data = ParsiDFplots,  
 x = "InterpenetrationFeedback",  
 y = "AverageInterpenetration",  
 grouping.var = "Part",  
 type = "p",  
 pairwise.comparisons = FALSE,  
 pairwise.display = "significant",  
 p.adjust.method = "bonferroni",  
 effsize.type = "unbiased",  
 results.subtitle = FALSE,  
 xlab = "Type of Feedback",  
 ylab = "Average Interpenetration",  
 sample.size.label = FALSE,  
 var.equal = TRUE,  
 mean.plotting = FALSE,  
 mean.ci = TRUE,  
 paired = TRUE,  
 title.text = "Interpenetration Box-Violin Plots",  
 caption.text = "Note: Interpenetration distance is displayed in cm.",  
 title.color = "black",  
 caption.color = "black"  
)  
  
  
# Replicating the above but this time we look on the effect of the type of feedback on the DVs in 1st and 2nd Part of the experiment individually  
p3 <- ggstatsplot::grouped\_ggbetweenstats(  
 data = ParsiDFplots,  
 x = "InterpenetrationFeedback",  
 y = "MaxInterpenetration",  
 grouping.var = "Part",  
 type = "p",  
 pairwise.comparisons = FALSE,  
 pairwise.display = "significant",  
 p.adjust.method = "bonferroni",  
 effsize.type = "unbiased",  
 results.subtitle = FALSE,  
 xlab = "Type of Feedback",  
 ylab = "Maximum Interpenetration",  
 sample.size.label = FALSE,  
 var.equal = TRUE,  
 mean.plotting = FALSE,  
 mean.ci = TRUE,  
 paired = TRUE,  
 title.text = "Interpenetration Box-Violin Plots",  
 caption.text = "Note: Interpenetration distance is displayed in cm.",  
 title.color = "black",  
 caption.color = "black"  
 )   
  
p4 <- ggstatsplot::grouped\_ggbetweenstats(  
 data = ParsiDFplots,  
 x = "InterpenetrationFeedback",  
 y = "AverageInterpenetration",  
 grouping.var = "Part",  
 type = "p",  
 pairwise.comparisons = FALSE,  
 pairwise.display = "significant",  
 p.adjust.method = "bonferroni",  
 effsize.type = "unbiased",  
 results.subtitle = FALSE,  
 xlab = "Type of Feedback",  
 ylab = "Average Interpenetration",  
 sample.size.label = FALSE,  
 var.equal = TRUE,  
 mean.plotting = FALSE,  
 mean.ci = TRUE,  
 paired = TRUE,  
 title.text = "Interpenetration Box-Violin Plots",  
 caption.text = "Note: Interpenetration distance is displayed in cm.",  
 title.color = "black",  
 caption.color = "black"  
)   
  
# Lets check the effect of shaded condition on DVs  
p5 <- ggstatsplot:: ggbetweenstats(  
 data = ParsiDFplots,  
 x = "Part",  
 y = "MaxInterpenetration",  
 grouping.var = "InterpenetrationFeedback",  
 type = "p",  
 pairwise.comparisons = FALSE,  
 pairwise.display = "significant",  
 p.adjust.method = "bonferroni",  
 effsize.type = "unbiased",  
 results.subtitle = FALSE,  
 xlab = "Order",  
 ylab = "Maximum Interpenetration",  
 sample.size.label = FALSE,  
 var.equal = TRUE,  
 mean.plotting = FALSE,  
 mean.ci = TRUE,  
 paired = TRUE,  
 title.text = "Interpenetration Box-Violin Plots",  
 caption.text = "Note: Interpenetration distance is displayed in cm.",  
 title.color = "black",  
 caption.color = "black"  
 )   
  
p6 <- ggstatsplot::ggbetweenstats(  
 data = ParsiDFplots,  
 x = "Part",  
 y = "AverageInterpenetration",  
 grouping.var = "InterpenetrationFeedback",  
 type = "p",  
 pairwise.comparisons = FALSE,  
 pairwise.display = "significant",  
 p.adjust.method = "bonferroni",  
 effsize.type = "unbiased",  
 results.subtitle = FALSE,  
 xlab = "Order",  
 ylab = "Average Interpenetration",  
 sample.size.label = FALSE,  
 var.equal = TRUE,  
 mean.plotting = FALSE,  
 mean.ci = TRUE,  
 paired = TRUE,  
 title.text = "Interpenetration Box-Violin Plots",  
 caption.text = "Note: Interpenetration distance is displayed in cm.",  
 title.color = "black",  
 caption.color = "black"  
)   
  
p7 <- ggstatsplot::grouped\_ggbetweenstats(  
 data = ParsiDFplots,  
 x = "Part",  
 y = "MaxInterpenetration",  
 grouping.var = "InterpenetrationFeedback",  
 type = "p",  
 pairwise.comparisons = FALSE,  
 pairwise.display = "significant",  
 p.adjust.method = "bonferroni",  
 effsize.type = "unbiased",  
 results.subtitle = FALSE,  
 xlab = "Order",  
 ylab = "Maximum Interpenetration",  
 sample.size.label = FALSE,  
 var.equal = TRUE,  
 mean.plotting = FALSE,  
 mean.ci = TRUE,  
 paired = TRUE,  
 title.text = "Interpenetration Box-Violin Plots",  
 caption.text = "Note: Interpenetration distance is displayed in cm.",  
 title.color = "black",  
 caption.color = "black")   
  
p8 <- ggstatsplot::grouped\_ggbetweenstats(  
 data = ParsiDFplots,  
 x = "Part",  
 y = "AverageInterpenetration",  
 grouping.var = "InterpenetrationFeedback",  
 type = "p",  
 pairwise.comparisons = FALSE,  
 pairwise.display = "significant",  
 p.adjust.method = "bonferroni",  
 effsize.type = "unbiased",  
 results.subtitle = FALSE,  
 xlab = "Order",  
 ylab = "Average Interpenetration",  
 sample.size.label = FALSE,  
 var.equal = TRUE,  
 mean.plotting = FALSE,  
 mean.ci = TRUE,  
 paired = TRUE,  
 title.text = "Interpenetration Box-Violin Plots",  
 caption.text = "Note: Interpenetration distance is displayed in cm.",  
 title.color = "black",  
 caption.color = "black"  
 )   
p1



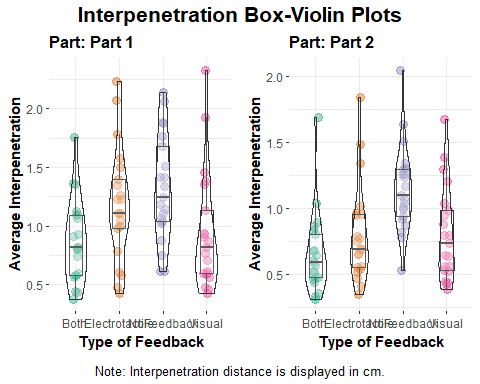
p2



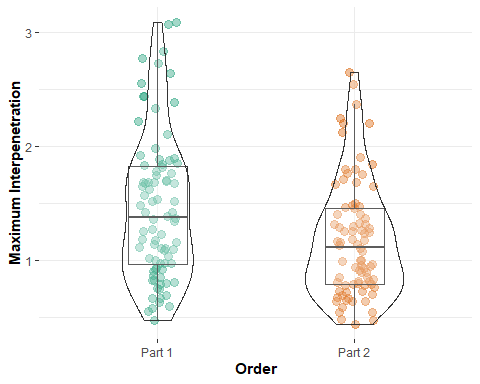
p3



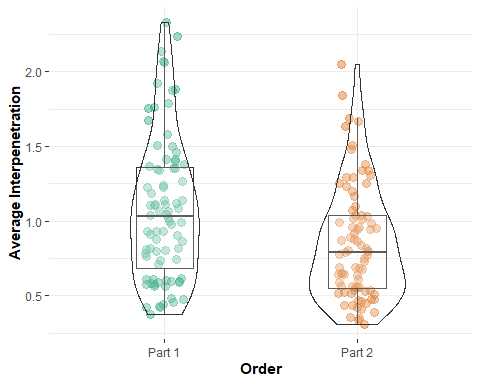
p4



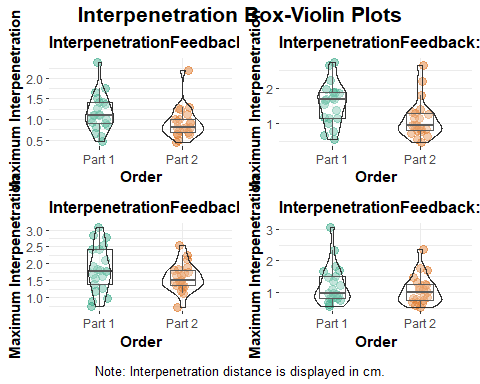
p5



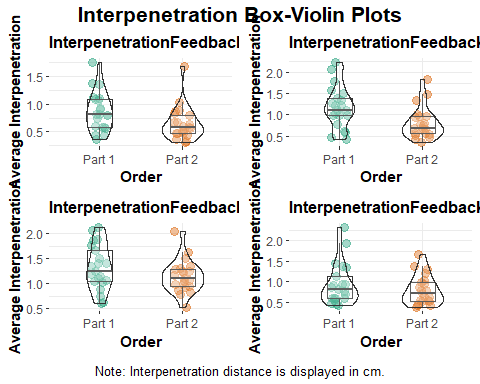
p6



p7



p8



Let’s check the Two Way Repeated Measures ANOVA

aMax <- aov\_ez("ID", "MaxInterpenetration", ParsiDF,  
 within = c("Part", "InterpenetrationFeedback"),  
 anova\_table = list(es = "pes"))  
  
knitr::kable(nice(aMax$anova\_table))

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Effect | df | MSE | F | pes | p.value |
| Part | 1, 20 | 0.06 | 29.43 \*\*\* | .595 | <.001 |
| InterpenetrationFeedback | 1.88, 37.52 | 0.12 | 28.36 \*\*\* | .586 | <.001 |
| Part:InterpenetrationFeedback | 2.06, 41.24 | 0.04 | 5.16 \*\* | .205 | .009 |

aAv <- aov\_ez("ID", "AverageInterpenetration", ParsiDF,  
 within = c("Part", "InterpenetrationFeedback"),  
 anova\_table = list(es = "pes"))  
  
knitr::kable(nice(aAv$anova\_table))

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Effect | df | MSE | F | pes | p.value |
| Part | 1, 20 | 0.07 | 31.74 \*\*\* | .613 | <.001 |
| InterpenetrationFeedback | 1.97, 39.31 | 0.12 | 25.89 \*\*\* | .564 | <.001 |
| Part:InterpenetrationFeedback | 2.11, 42.21 | 0.04 | 5.72 \*\* | .222 | .006 |

effectsize::omega\_squared(aMax, partial = TRUE, ci = 0.95)

## Parameter | Omega2 (partial) | 95% CI  
## ----------------------------------------------------------------  
## Part | 0.56 | [ 0.24, 0.74]  
## InterpenetrationFeedback | 0.56 | [ 0.38, 0.68]  
## Part:InterpenetrationFeedback | 0.16 | [-0.02, 0.32]

effectsize::omega\_squared(aAv, partial = TRUE, ci = 0.95)

## Parameter | Omega2 (partial) | 95% CI  
## ----------------------------------------------------------------  
## Part | 0.58 | [ 0.26, 0.75]  
## InterpenetrationFeedback | 0.54 | [ 0.35, 0.66]  
## Part:InterpenetrationFeedback | 0.18 | [-0.01, 0.34]

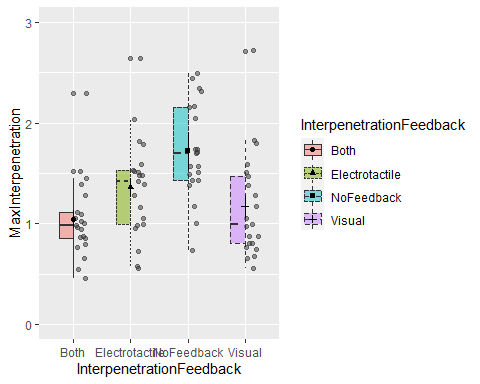
We can see that every type of feedback as well as the interrelationship with the part of the experiment have a large effect on DVs!!!!!!

Reference for interpreting Omega Squared Small effect: ω2 = 0.01; Medium effect: ω2 = 0.06; Large effect: ω2 = 0.14.

Let’s plot the main effects (Interpenetration Feedback OR Part of the experiment).

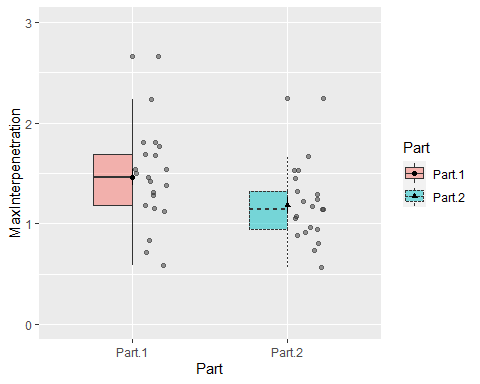
# ANOVAs just for the plots  
aMaxPlots <- aov\_ez("ID", "MaxInterpenetration", ParsiDFplots,  
 within = c("Part", "InterpenetrationFeedback"),  
 anova\_table = list(es = "pes"))  
  
aAvPlots <- aov\_ez("ID", "AverageInterpenetration", ParsiDFplots,  
 within = c("Part", "InterpenetrationFeedback"),  
 anova\_table = list(es = "pes"))  
  
#plots  
afex\_plot(aMaxPlots, x = "InterpenetrationFeedback", error = "within",   
 mapping = c("linetype", "shape", "fill"),  
 data\_geom = ggpol::geom\_boxjitter,   
 data\_arg = list(width = 0.5)) +  
 ylim(0, 3)

## NOTE: Results may be misleading due to involvement in interactions



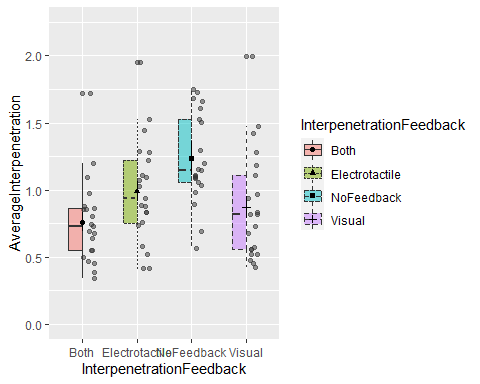
afex\_plot(aMaxPlots, x = "Part", error = "within",   
 mapping = c("linetype", "shape", "fill"),  
 data\_geom = ggpol::geom\_boxjitter,   
 data\_arg = list(width = 0.5)) +  
 ylim(0, 3)

## NOTE: Results may be misleading due to involvement in interactions



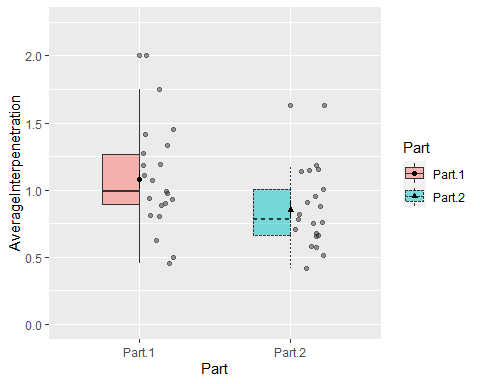
afex\_plot(aAvPlots, x = "InterpenetrationFeedback", error = "within",   
 mapping = c("linetype", "shape", "fill"),  
 data\_geom = ggpol::geom\_boxjitter,   
 data\_arg = list(width = 0.5)) +  
 ylim(0, 2.25)

## NOTE: Results may be misleading due to involvement in interactions



afex\_plot(aAvPlots, x = "Part", error = "within",   
 mapping = c("linetype", "shape", "fill"),  
 data\_geom = ggpol::geom\_boxjitter,   
 data\_arg = list(width = 0.5)) +  
 ylim(0, 2.25)

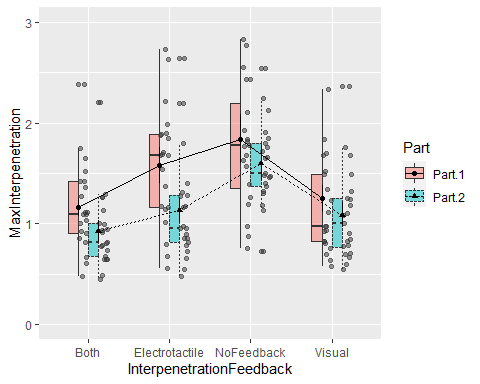
## NOTE: Results may be misleading due to involvement in interactions



Let’s plot the main interaction effects (Interpenetration Feedback AND Part of the experiment).

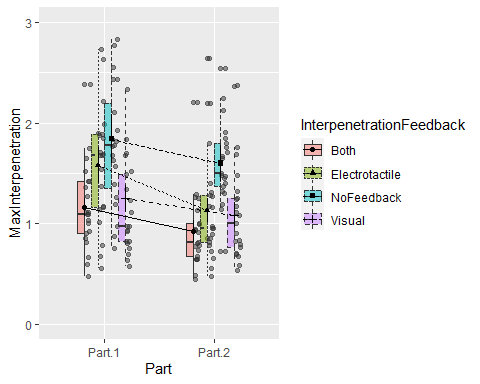
afex\_plot(aMaxPlots, x = "InterpenetrationFeedback", trace = "Part", error = "within",   
 mapping = c("linetype", "shape", "fill"),  
 data\_geom = ggpol::geom\_boxjitter,   
 data\_arg = list(width = 0.5)) +  
 ylim(0, 3)

## Warning: Removed 2 rows containing non-finite values (stat\_box\_jitter).



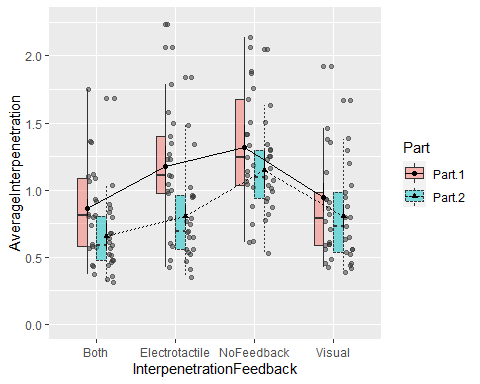
afex\_plot(aMaxPlots, x = "Part", trace = "InterpenetrationFeedback", error = "within",   
 mapping = c("linetype", "shape", "fill"),  
 data\_geom = ggpol::geom\_boxjitter,   
 data\_arg = list(width = 0.5)) +  
 ylim(0, 3)

## Warning: Removed 2 rows containing non-finite values (stat\_box\_jitter).



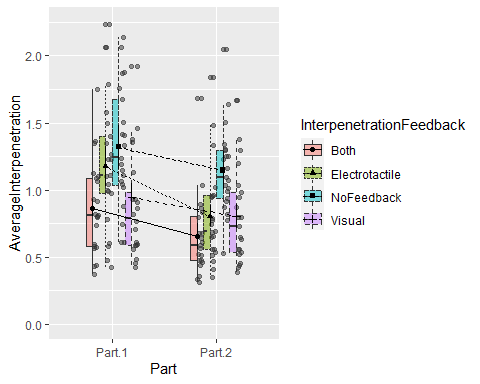
afex\_plot(aAvPlots, x = "InterpenetrationFeedback", trace = "Part", error = "within",   
 mapping = c("linetype", "shape", "fill"),  
 data\_geom = ggpol::geom\_boxjitter,   
 data\_arg = list(width = 0.5)) +  
 ylim(0, 2.25)

## Warning: Removed 1 rows containing non-finite values (stat\_box\_jitter).



afex\_plot(aAvPlots, x = "Part", trace = "InterpenetrationFeedback", error = "within",   
 mapping = c("linetype", "shape", "fill"),  
 data\_geom = ggpol:: geom\_boxjitter,   
 data\_arg = list(width = 0.5)) +  
 ylim(0, 2.25)

## Warning: Removed 1 rows containing non-finite values (stat\_box\_jitter).



Post-hoc Tests

"  
# Main Effects  
lsmMaxFeedback <- lsmeans(aMax,~ InterpenetrationFeedback)  
contrast(lsmMax, method=pairwise, interaction = TRUE)  
  
lsmMaxPart <- lsmeans(aMax,~ Part)  
contrast(lsmMax, method=pairwise, interaction = TRUE)  
  
lsmMaxIntEff <- lsmeans(aMax,~ InterpenetrationFeedback:Part)  
contrast(lsmMax, method=pairwise, interaction = TRUE)  
  
lsmAvFeedback <- lsmeans(aAv,~ InterpenetrationFeedback)  
contrast(lsmAvIntEff, method=pairwise, interaction = TRUE)  
  
lsmAvPart <- lsmeans(aAv,~ Part)  
contrast(lsmAvIntEff, method=pairwise, interaction = TRUE)  
  
lsmAvIntEff <- lsmeans(aAv,~ InterpenetrationFeedback:Part)  
contrast(lsmAvIntEff, method=pairwise, interaction = TRUE)  
"

## [1] "\n# Main Effects\nlsmMaxFeedback <- lsmeans(aMax,~ InterpenetrationFeedback)\ncontrast(lsmMax, method=pairwise, interaction = TRUE)\n\nlsmMaxPart <- lsmeans(aMax,~ Part)\ncontrast(lsmMax, method=pairwise, interaction = TRUE)\n\nlsmMaxIntEff <- lsmeans(aMax,~ InterpenetrationFeedback:Part)\ncontrast(lsmMax, method=pairwise, interaction = TRUE)\n\nlsmAvFeedback <- lsmeans(aAv,~ InterpenetrationFeedback)\ncontrast(lsmAvIntEff, method=pairwise, interaction = TRUE)\n\nlsmAvPart <- lsmeans(aAv,~ Part)\ncontrast(lsmAvIntEff, method=pairwise, interaction = TRUE)\n\nlsmAvIntEff <- lsmeans(aAv,~ InterpenetrationFeedback:Part)\ncontrast(lsmAvIntEff, method=pairwise, interaction = TRUE)\n"

#lsmMaxIntEff <- lsmeans(aMax,~ InterpenetrationFeedback:Part)  
#contrast(lsmMaxIntEff, method= "pairwise", interaction = TRUE)  
  
#lsmAvIntEff <- lsmeans(aAv,~ InterpenetrationFeedback:Part)  
#contrast(lsmAvIntEff, method= "pairwise", interaction = TRUE)  
  
#?lsmeans()  
# Only Interaction Effects  
  
contrast(emmeans(aMax,~ Part:InterpenetrationFeedback),   
 method="pairwise", interaction=TRUE)

## Part\_pairwise InterpenetrationFeedback\_pairwise estimate SE df t.ratio  
## Part.1 - Part.2 Both - Electrotactile -0.1201 0.0718 60 -1.672   
## Part.1 - Part.2 Both - NoFeedback 0.1262 0.0718 60 1.757   
## Part.1 - Part.2 Both - Visual 0.1153 0.0718 60 1.605   
## Part.1 - Part.2 Electrotactile - NoFeedback 0.2463 0.0718 60 3.429   
## Part.1 - Part.2 Electrotactile - Visual 0.2354 0.0718 60 3.277   
## Part.1 - Part.2 NoFeedback - Visual -0.0109 0.0718 60 -0.152   
## p.value  
## 0.0997   
## 0.0841   
## 0.1137   
## 0.0011   
## 0.0017   
## 0.8800

contrast(emmeans(aAv,~ Part:InterpenetrationFeedback),   
 method="pairwise", interaction=TRUE)

## Part\_pairwise InterpenetrationFeedback\_pairwise estimate SE df t.ratio  
## Part.1 - Part.2 Both - Electrotactile -0.0928 0.0738 60 -1.258   
## Part.1 - Part.2 Both - NoFeedback 0.1674 0.0738 60 2.268   
## Part.1 - Part.2 Both - Visual 0.1509 0.0738 60 2.046   
## Part.1 - Part.2 Electrotactile - NoFeedback 0.2602 0.0738 60 3.527   
## Part.1 - Part.2 Electrotactile - Visual 0.2437 0.0738 60 3.304   
## Part.1 - Part.2 NoFeedback - Visual -0.0164 0.0738 60 -0.223   
## p.value  
## 0.2132   
## 0.0269   
## 0.0452   
## 0.0008   
## 0.0016   
## 0.8243

The above compares the effect size of the chance of the types of interpenetration feedback between part 1 and part 2 of the experiment!

we can see that for MAX INTERPENETRATION the significant comparisons are: Estimate SE df t.ratio p.value 1) Part.1 - Part.2 Electrotactile - NoFeedback 0.2463 0.0718 60 3.429 0.0011

1. Part.1 - Part.2 Electrotactile - Visual 0.2354 0.0718 60 3.277 0.0017

This means that the improvement of the performance from part 1 to part 2 regarding the maximum interpenetration was significantly greater for the “Electrotactile Feedback” compared to the “No Feedback” and “Visual Feedback” respectively! The rest of the comparisons were insignificant!

For the AVERAGE INTERPENETRATION the significant comparisons are: Estimate SE df t.ratio p.value 1) Part.1 - Part.2 Both - NoFeedback 0.1674 0.0738 60 2.268 0.0269

1. Part.1 - Part.2 Both - Visual 0.1509 0.0738 60 2.046 0.0452
2. Part.1 - Part.2 Electrotactile - NoFeedback 0.2602 0.0738 60 3.527 0.0008
3. Part.1 - Part.2 Electrotactile - Visual 0.2437 0.0738 60 3.304 0.0016

This means that the improvement of the performance from part 1 to part 2 regarding the average interpenetration was significantly greater for the “Electrotactile Feedback” compared to the “No Feedback” and “Visual Feedback” respectively!

Also, that the improvement of the performance from part 1 to part 2 regarding the average interpenetration was significantly greater for the “Combined (i.e., Both) Feedback” compared to the “No Feedback” and “Visual Feedback” respectively! The rest of the comparisons were insignificant!