MANOVA I

Mike Strube

November 19, 2018

1 Preliminaries

In this section, the RStudio workspace and console panes are cleared of old output, variables, and other miscellaneous debris. Packages are loaded and any required data files are retrieved.

```
options(replace.assign = TRUE, width = 65, digits = 4, scipen = 4, fig.width = 4,
    fig.height = 4)
# Clear the workspace and console.
rm(list = ls(all = TRUE))
cat("\f")
```

```
# Turn off showing of significance asterisks.
options(show.signif.stars = F)
# Set the contrast option; important for ANOVAs.
options(contrasts = c("contr.sum", "contr.poly"))
how_long <- Sys.time()
set.seed(123)
library(knitr)
```

```
library(psych)

## Warning:  package 'psych' was built under R version 3.5.1

library(ggplot2)

## Warning:  package 'ggplot2' was built under R version 3.5.1
##
## Attaching package:  'ggplot2'
## The following objects are masked from 'package:psych':
##
##    %+%, alpha

library(MASS)
library(sciplot)
library(dplyr)

## Warning:  package 'dplyr' was built under R version 3.5.1
##
## Attaching package:  'dplyr'
## The following object is masked from 'package:MASS':
##
##    select
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(aod)
library(MVN)

## sROC 0.1-2 loaded

library(boot)

##
## Attaching package: 'boot'
## The following object is masked from 'package:psych':
##
##     logit

library(car)

## Warning: package 'car' was built under R version 3.5.1
## Loading required package: carData
##
## Attaching package: 'car'
## The following object is masked from 'package:boot':
##
##     logit
## The following object is masked from 'package:dplyr':
##
##     recode
## The following object is masked from 'package:psych':
##
##     logit

library(LogisticDx)
library(biotools)

## Loading required package: rpanel
## Loading required package: tcltk
## Package `rpanel', version 1.1-4: type help(rpanel) for summary information
##
## Attaching package: 'rpanel'
## The following object is masked from 'package:boot':
##
##     poisons
## Loading required package: tkrplot
## Loading required package: lattice
##
## Attaching package: 'lattice'
## The following object is masked from 'package:boot':
##
##     melanoma
## Loading required package: SpatialEpi
## Loading required package: sp
```

```r
## ---
## biotools version 3.1
##
library(multcomp)
## Loading required package:  mvtnorm
## Loading required package:  survival
##
## Attaching package:  'survival'
## The following object is masked from 'package:boot':
##
##     aml
## The following object is masked from 'package:aod':
##
##     rats
## Loading required package:  TH.data
##
## Attaching package:  'TH.data'
## The following object is masked from 'package:MASS':
##
##     geyser
library(ez)
library(GGally)
##
## Attaching package:  'GGally'
## The following object is masked from 'package:dplyr':
##
##     nasa
library(qqplotr)
##
## Attaching package:  'qqplotr'
## The following objects are masked from 'package:ggplot2':
##
##     stat_qq_line, StatQqLine
library(gridExtra)
##
## Attaching package:  'gridExtra'
## The following object is masked from 'package:dplyr':
##
##     combine
library(reshape)
##
## Attaching package:  'reshape'
## The following object is masked from 'package:dplyr':
##
##     rename
library(emmeans)
```

```
## Warning:  package 'emmeans' was built under R version 3.5.1
## NOTE: As of emmeans versions > 1.2.3,
##       The 'cld' function will be deprecated in favor of 'CLD'.
##       You may use 'cld' only if you have package:multcomp attached.
##
## Attaching package:  'emmeans'
## The following object is masked from 'package:GGally':
##
##     pigs
## The following object is masked from 'package:multcomp':
##
##     cld
```

1.1 Data

```
setwd("C:\\Courses\\Psychology 516\\PowerPoint\\2018")

Skills <- read.table("manova.csv", sep = ",", header = TRUE)
Skills <- as.data.frame(Skills)
```

1.2 Data Modifications

Residualized versions of continuous predictors are created so that preliminary analyses are not contaminated by outcome differences. Labeled variables are created to assist in creation of some tables and graphs. Dummy codes and linear combinations are created for later specialized analyses.

```
# Residuals
Skills$P_Verbal_R <- lm(P_Verbal ~ as.factor(Group), data = Skills)$residuals
Skills$P_Quant_R <- lm(P_Quant ~ as.factor(Group), data = Skills)$residuals
Skills$C_Verbal_R <- lm(C_Verbal ~ as.factor(Group), data = Skills)$residuals
Skills$C_Quant_R <- lm(C_Quant ~ as.factor(Group), data = Skills)$residuals

# Labels
Skills$Tx_P2[Skills$Tx_P == "1"] <- "No Paper Tx"
Skills$Tx_P2[Skills$Tx_P == "2"] <- "Paper Tx"

Skills$Tx_C2[Skills$Tx_C == "1"] <- "No Computer Tx"
Skills$Tx_C2[Skills$Tx_C == "2"] <- "Computer Tx"

Skills$Group2[Skills$Group == "1"] <- "No Paper Tx and No Computer Tx"
Skills$Group2[Skills$Group == "2"] <- "Paper Tx and No Computer Tx"
Skills$Group2[Skills$Group == "3"] <- "No Paper Tx and Computer Tx"
Skills$Group2[Skills$Group == "4"] <- "Paper Tx and Computer Tx"

Skills$Group3[Skills$Group == "1"] <- "No P, No C"
Skills$Group3[Skills$Group == "2"] <- "P, No C"
Skills$Group3[Skills$Group == "3"] <- "No P, C"
Skills$Group3[Skills$Group == "4"] <- "P, C"

# Dummy variables to be used in between-groups analyses.
```

```r
Skills$D1[Skills$Group == 1] <- 1
Skills$D2[Skills$Group == 1] <- 0
Skills$D3[Skills$Group == 1] <- 0
Skills$D4[Skills$Group == 1] <- 0
Skills$D1[Skills$Group == 2] <- 0
Skills$D2[Skills$Group == 2] <- 1
Skills$D3[Skills$Group == 2] <- 0
Skills$D4[Skills$Group == 2] <- 0
Skills$D1[Skills$Group == 3] <- 0
Skills$D2[Skills$Group == 3] <- 0
Skills$D3[Skills$Group == 3] <- 1
Skills$D4[Skills$Group == 3] <- 0
Skills$D1[Skills$Group == 4] <- 0
Skills$D2[Skills$Group == 4] <- 0
Skills$D3[Skills$Group == 4] <- 0
Skills$D4[Skills$Group == 4] <- 1

# Outcome linear combinations to be used in repeated measures
# analyses.
Skills$Sum <- Skills$P_Verbal + Skills$P_Quant + Skills$C_Verbal +
    Skills$C_Quant
Skills$Domain <- Skills$P_Verbal - Skills$P_Quant + Skills$C_Verbal -
    Skills$C_Quant
Skills$Mode <- Skills$P_Verbal + Skills$P_Quant - Skills$C_Verbal -
    Skills$C_Quant
Skills$DxM <- Skills$P_Verbal - Skills$P_Quant - Skills$C_Verbal +
    Skills$C_Quant

# Create a non-factor version of the condition variables before
# converting them to factors.
Skills$Tx_P_NF <- Skills$Tx_P
Skills$Tx_C_NF <- Skills$Tx_C

# Convert to factors
Skills$Tx_P = factor(Skills$Tx_P, levels = c(1, 2), labels = c("No Tx(P)",
    "Tx(P)"))
Skills$Tx_C = factor(Skills$Tx_C, levels = c(1, 2), labels = c("No Tx(C)",
    "Tx(C)"))

# Sort file by Group
Skills <- Skills[order(Skills$Group), ]
```

2 Data Characteristics

These hypothetical data simulate a training study in which students are given training to take tests of verbal and quantitative ability. The training is conducted either with paper-and-pencil (standard) tests or with computer-administered tests (or both) and the tests are administered in both formats. The basic nature of these data is explored here.

2.1 Some Descriptive Statistics

Some basic descriptive statistics give an initial glimpse of the data.

```
describeBy(Skills[, 2:5], group = Skills$Group)

##
##  Descriptive statistics by group
## group: 1
##          vars  n  mean    sd median trimmed   mad   min   max
## P_Verbal    1 25 47.86 10.59  48.94   48.13  9.42 26.20 64.23
## P_Quant     2 25 47.52  9.99  46.27   47.74 10.40 23.98 66.84
## C_Verbal    3 25 45.72 10.84  45.45   45.65 10.75 21.06 71.22
## C_Quant     4 25 46.28 10.70  48.09   46.66 12.55 22.20 62.60
##          range  skew kurtosis   se
## P_Verbal 38.04 -0.13    -0.86 2.12
## P_Quant  42.85 -0.12    -0.29 2.00
## C_Verbal 50.16  0.05    -0.17 2.17
## C_Quant  40.41 -0.31    -0.87 2.14
## ------------------------------------------------
## group: 2
##          vars  n  mean    sd median trimmed   mad   min   max
## P_Verbal    1 25 61.86 12.84  59.35   61.91  7.72 27.31 87.95
## P_Quant     2 25 71.83 10.87  70.09   72.39 10.67 40.38 94.83
## C_Verbal    3 25 48.77 10.28  50.08   48.51  9.93 30.46 71.87
## C_Quant     4 25 49.65 10.97  47.15   49.77  9.42 23.54 70.41
##          range  skew kurtosis   se
## P_Verbal 60.64 -0.05     0.78 2.57
## P_Quant  54.46 -0.62     1.22 2.17
## C_Verbal 41.41  0.23    -0.64 2.06
## C_Quant  46.87 -0.01    -0.44 2.19
## ------------------------------------------------
## group: 3
##          vars  n  mean    sd median trimmed   mad   min   max
## P_Verbal    1 25 24.17 11.09  23.60   24.03 12.56  6.98 42.79
## P_Quant     2 25 32.78  9.35  31.30   32.33  9.65 15.05 56.80
## C_Verbal    3 25 53.36 10.30  55.94   53.78  6.77 33.88 70.76
## C_Quant     4 25 60.61  9.01  61.38   60.39  9.86 46.46 77.37
##          range  skew kurtosis   se
## P_Verbal 35.80  0.07    -1.28 2.22
## P_Quant  41.74  0.51     0.01 1.87
## C_Verbal 36.88 -0.59    -0.66 2.06
## C_Quant  30.91  0.14    -1.09 1.80
## ------------------------------------------------
## group: 4
##          vars  n  mean   sd median trimmed  mad   min    max
## P_Verbal    1 25 92.45 5.77  92.33   92.70 7.01 82.09 100.00
```

```
## P_Quant     2 25 81.93 8.76  82.15   82.41 7.82 62.46  97.12
## C_Verbal    3 25 82.43 8.78  82.06   82.81 9.74 65.30  95.87
## C_Quant     4 25 91.51 6.26  89.99   91.67 7.10 79.47 100.00
##          range  skew kurtosis   se
## P_Verbal 17.91 -0.09    -1.29 1.15
## P_Quant  34.66 -0.52    -0.42 1.75
## C_Verbal 30.57 -0.43    -0.70 1.76
## C_Quant  20.53 -0.04    -1.27 1.25
```

```r
with(Skills, tapply(P_Verbal, list(Tx_P, Tx_C), mean))
```

```
##          No Tx(C) Tx(C)
## No Tx(P)    47.86 24.17
## Tx(P)       61.86 92.45
```

```r
with(Skills, tapply(P_Quant, list(Tx_P, Tx_C), mean))
```

```
##          No Tx(C) Tx(C)
## No Tx(P)    47.52 32.78
## Tx(P)       71.83 81.93
```

```r
with(Skills, tapply(C_Verbal, list(Tx_P, Tx_C), mean))
```

```
##          No Tx(C) Tx(C)
## No Tx(P)    45.72 53.36
## Tx(P)       48.77 82.43
```

```r
with(Skills, tapply(C_Quant, list(Tx_P, Tx_C), mean))
```

```
##          No Tx(C) Tx(C)
## No Tx(P)    46.28 60.61
## Tx(P)       49.65 91.51
```

```r
with(Skills, tapply(P_Verbal, list(Tx_P, Tx_C), sd))
```

```
##          No Tx(C)  Tx(C)
## No Tx(P)    10.59 11.089
## Tx(P)       12.84  5.766
```

```r
with(Skills, tapply(P_Quant, list(Tx_P, Tx_C), sd))
```

```
##          No Tx(C) Tx(C)
## No Tx(P)    9.985 9.353
## Tx(P)      10.873 8.764
```

```r
with(Skills, tapply(C_Verbal, list(Tx_P, Tx_C), sd))
```

```
##          No Tx(C)  Tx(C)
## No Tx(P)    10.84 10.302
## Tx(P)       10.28  8.784
```

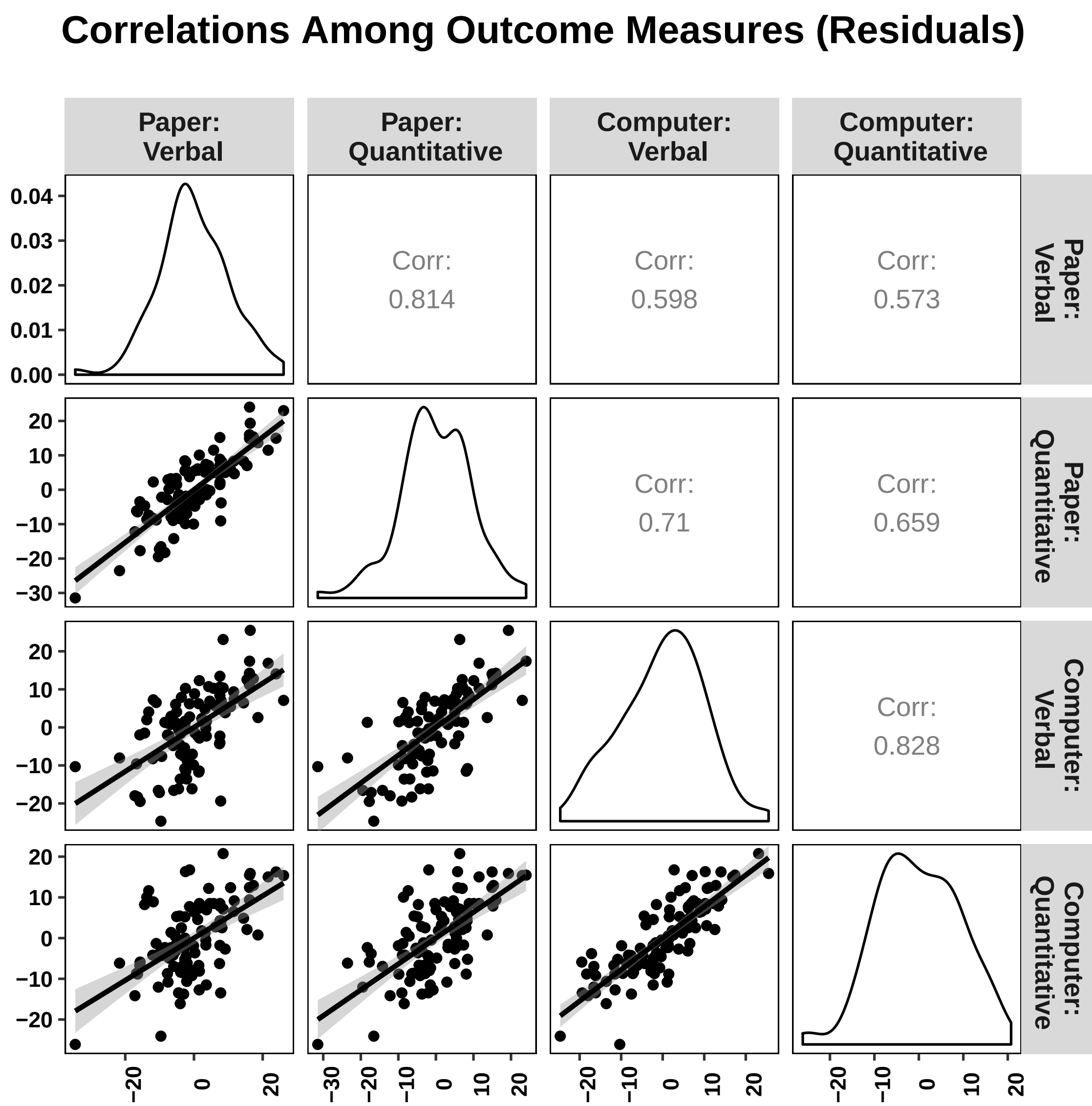
```r
with(Skills, tapply(C_Quant, list(Tx_P, Tx_C), sd))
```

```
##          No Tx(C) Tx(C)
## No Tx(P)    10.70 9.005
## Tx(P)       10.97 6.262
```

2.2 Basic Visualization

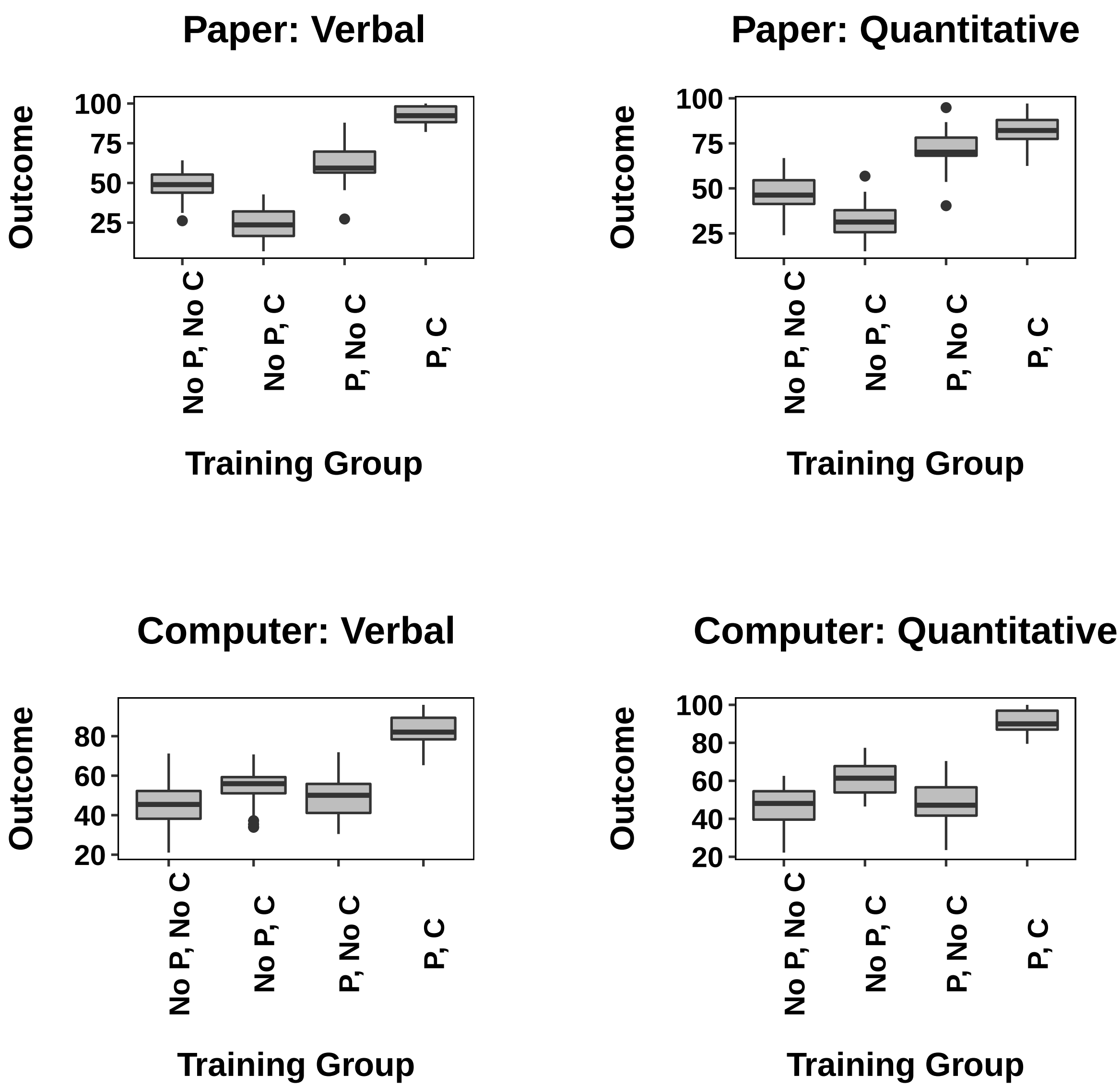
The basic nature of the data is easily viewed with some simple graphics.

```
ggpairs(Skills[9:12], lower = list(continuous = "smooth"), upper = list(continuous = "cor"),
    columnLabels = c("Paper:\n Verbal", "Paper:\n Quantitative", "Computer:\n Verbal",
        "Computer:\n Quantitative")) + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 9, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 9, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 16), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 16), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Correlations Among Outcome Measures (Residuals)")
```




```
Skills$Group4 <- factor(Skills$Group3, levels = c("No P, No C", "No P, C",
    "P, No C", "P, C"), labels = c("No P, No C", "No P, C", "P, No C",
    "P, C"))

p1 <- ggplot(Skills, aes(x = as.factor(Group4), y = P_Verbal)) + geom_boxplot(fill = "gray") +
    ylab("Outcome") + xlab("Training Group") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
```

```
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Paper: Verbal")

p2 <- ggplot(Skills, aes(x = as.factor(Group4), y = P_Quant)) + geom_boxplot(fill = "gray") +
    ylab("Outcome") + xlab("Training Group") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Paper: Quantitative")

p3 <- ggplot(Skills, aes(x = as.factor(Group4), y = C_Verbal)) + geom_boxplot(fill = "gray") +
    ylab("Outcome") + xlab("Training Group") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Computer: Verbal")

p4 <- ggplot(Skills, aes(x = as.factor(Group4), y = C_Quant)) + geom_boxplot(fill = "gray") +
    ylab("Outcome") + xlab("Training Group") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Computer: Quantitative")
grid.arrange(p1, p2, p3, p4, nrow = 2)
```



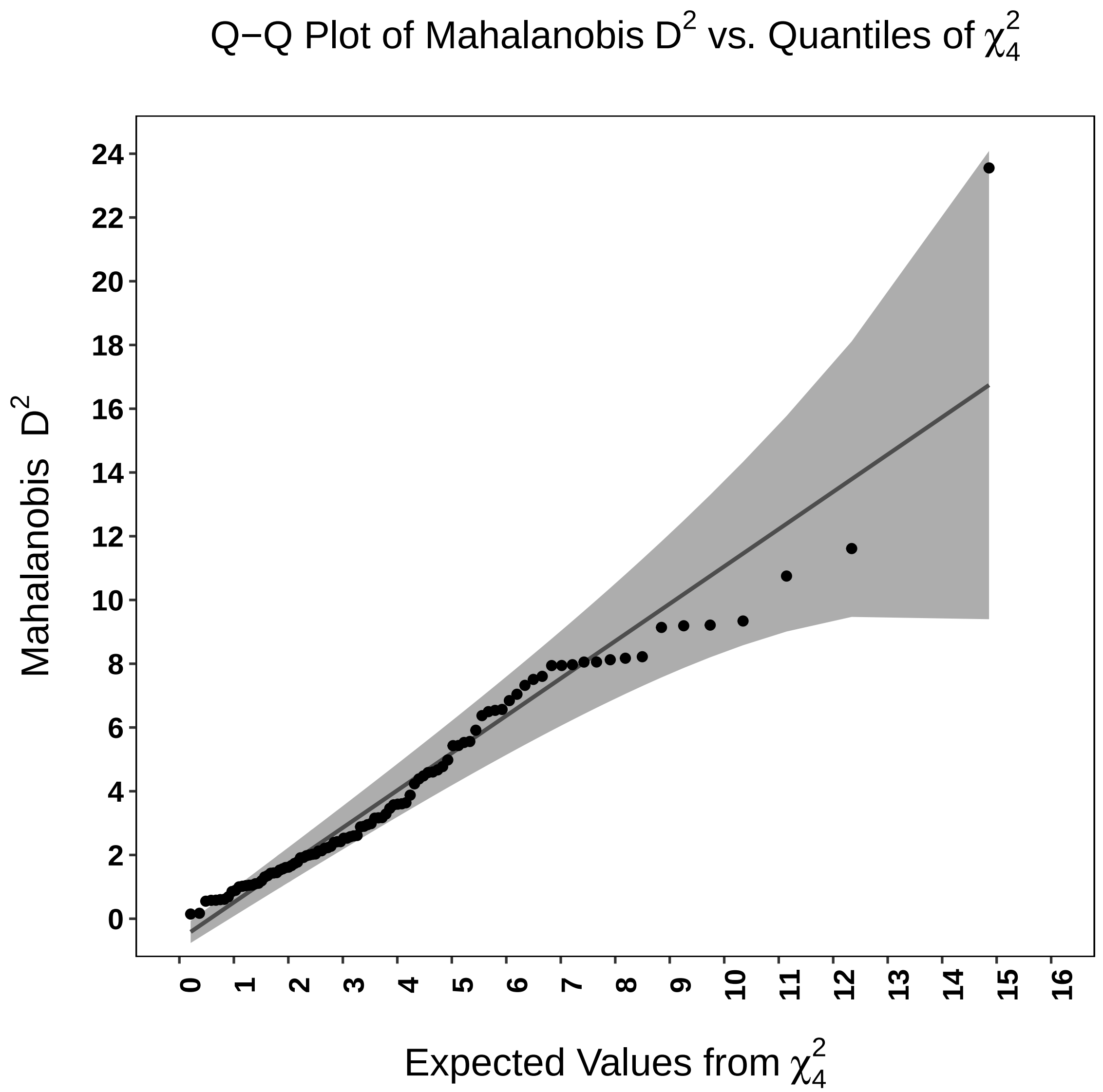
3 Multivariate Normality Assumption

The classification part of discriminant analysis (as well as any significance tests for the discriminant functions) rely on the multivariate normality assumption. Because MANOVA is inherently a discriminant analysis, we make the same assumption. The tests are performed on the residualized data so that group differences do not affect the results. Note that a violation of multivariate normality will also affect the test of homogeneity of covariance matrices.

3.1 Full Sample

```
mvn(Skills[, 9:12], mvnTest = "mardia")

## $multivariateNormality
##              Test        Statistic            p value Result
## 1 Mardia Skewness  33.714208460761 0.0281242141329968     NO
## 2 Mardia Kurtosis 2.67851447156136 0.0073949536550868     NO
## 3             MVN             <NA>               <NA>     NO
##
## $univariateNormality
##           Test   Variable Statistic   p value Normality
## 1 Shapiro-Wilk P_Verbal_R    0.9857    0.3545    YES
## 2 Shapiro-Wilk P_Quant_R     0.9843    0.2825    YES
## 3 Shapiro-Wilk C_Verbal_R    0.9881    0.5174    YES
## 4 Shapiro-Wilk C_Quant_R     0.9867    0.4203    YES
##
## $Descriptives
##              n       Mean Std.Dev  Median    Min   Max   25th
## P_Verbal_R 100 -2.821e-16  10.248 -1.2840 -34.55 26.09 -5.576
## P_Quant_R  100 -6.008e-17   9.626 -1.0052 -31.45 24.02 -5.440
## C_Verbal_R 100  2.696e-16   9.927  1.2883 -24.66 25.50 -7.084
## C_Quant_R  100 -1.776e-16   9.279 -0.8129 -26.11 20.75 -6.727
##             75th     Skew Kurtosis
## P_Verbal_R 7.550 -0.04563   0.5699
## P_Quant_R  6.135 -0.23775   0.6170
## C_Verbal_R 6.623 -0.15956  -0.2856
## C_Quant_R  7.167 -0.09505  -0.2942
```

```
CV <- cov(Skills[, 9:12])
D2_1 <- mahalanobis(Skills[, 9:12], center = colMeans(Skills[, 9:12]),
    cov = CV)
D2_1 <- as.data.frame(D2_1)
ggplot(D2_1, aes(sample = D2_1)) + stat_qq_band(distribution = "chisq",
    dparams = list(df = 4)) + stat_qq_line(distribution = "chisq",
    dparams = list(df = 4)) + stat_qq(distribution = "qchisq", dparams = list(df = 4)) +
    scale_y_continuous(breaks = seq(0, 24, 2)) + scale_x_continuous(breaks = seq(0,
    16, 1)) + coord_cartesian(xlim = c(0, 16), ylim = c(0, 24)) +
    xlab(expression("Expected Values from" * ~chi[4]^2)) + ylab(expression("Mahalanobis " *
    ~D^2)) + theme(text = element_text(size = 14, family = "sans",
    color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 16), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 16), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle(expression("Q-Q Plot of Mahalanobis" *
    ~D^2 * " vs. Quantiles of" * ~chi[4]^2))
```

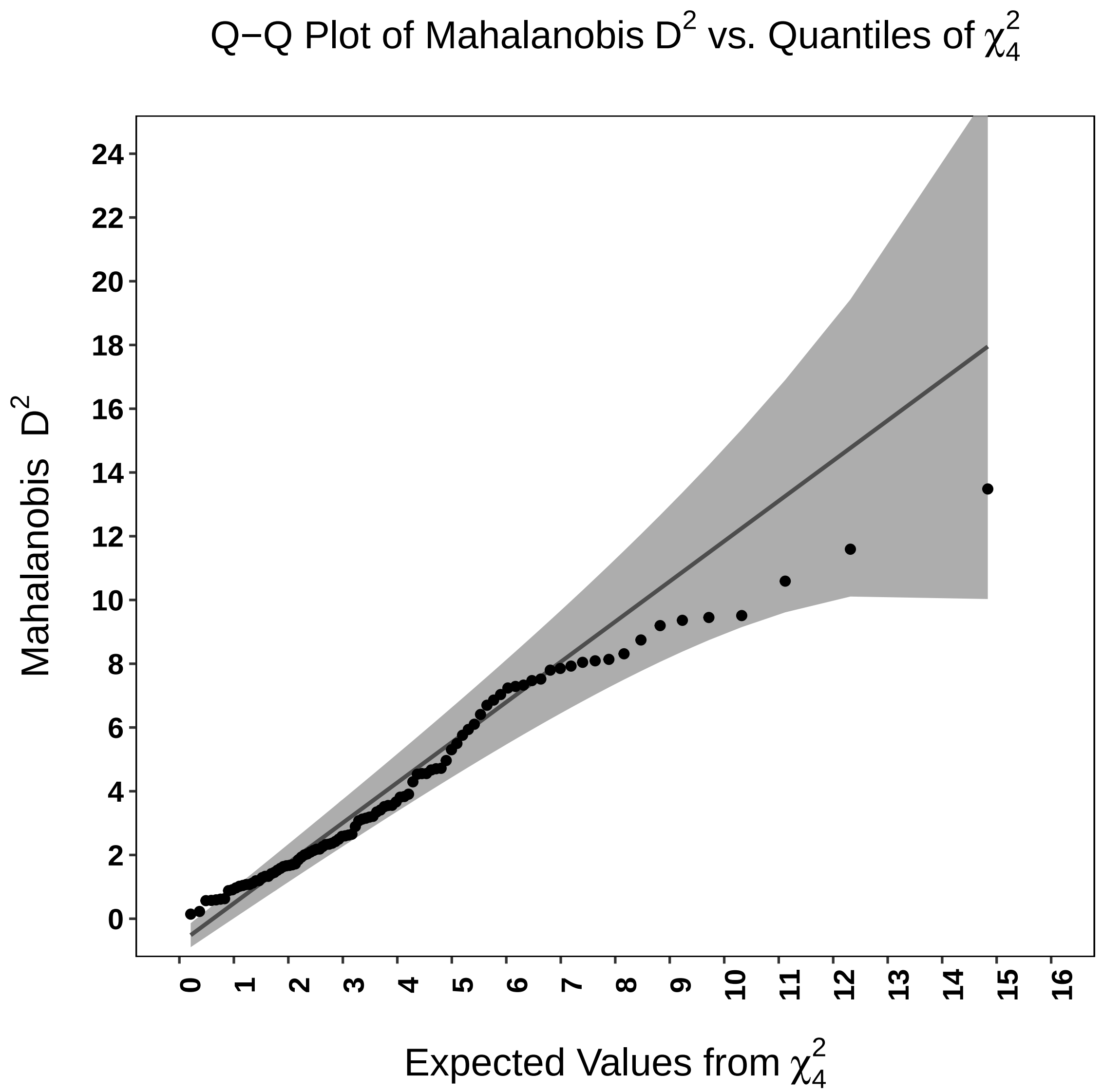


3.2 Outlier Excluded

```
Skills$D2_1 <- D2_1
Skills_Trimmed <- Skills[which(Skills$D2_1 != max(Skills$D2_1)), ]

mvn(Skills_Trimmed[, 9:12], mvnTest = "mardia")

## $multivariateNormality
##              Test          Statistic           p value Result
## 1 Mardia Skewness  20.2783280472259 0.440644455966184    YES
## 2 Mardia Kurtosis 0.600058653103908 0.548467146873458    YES
## 3             MVN              <NA>              <NA>    YES
##
```

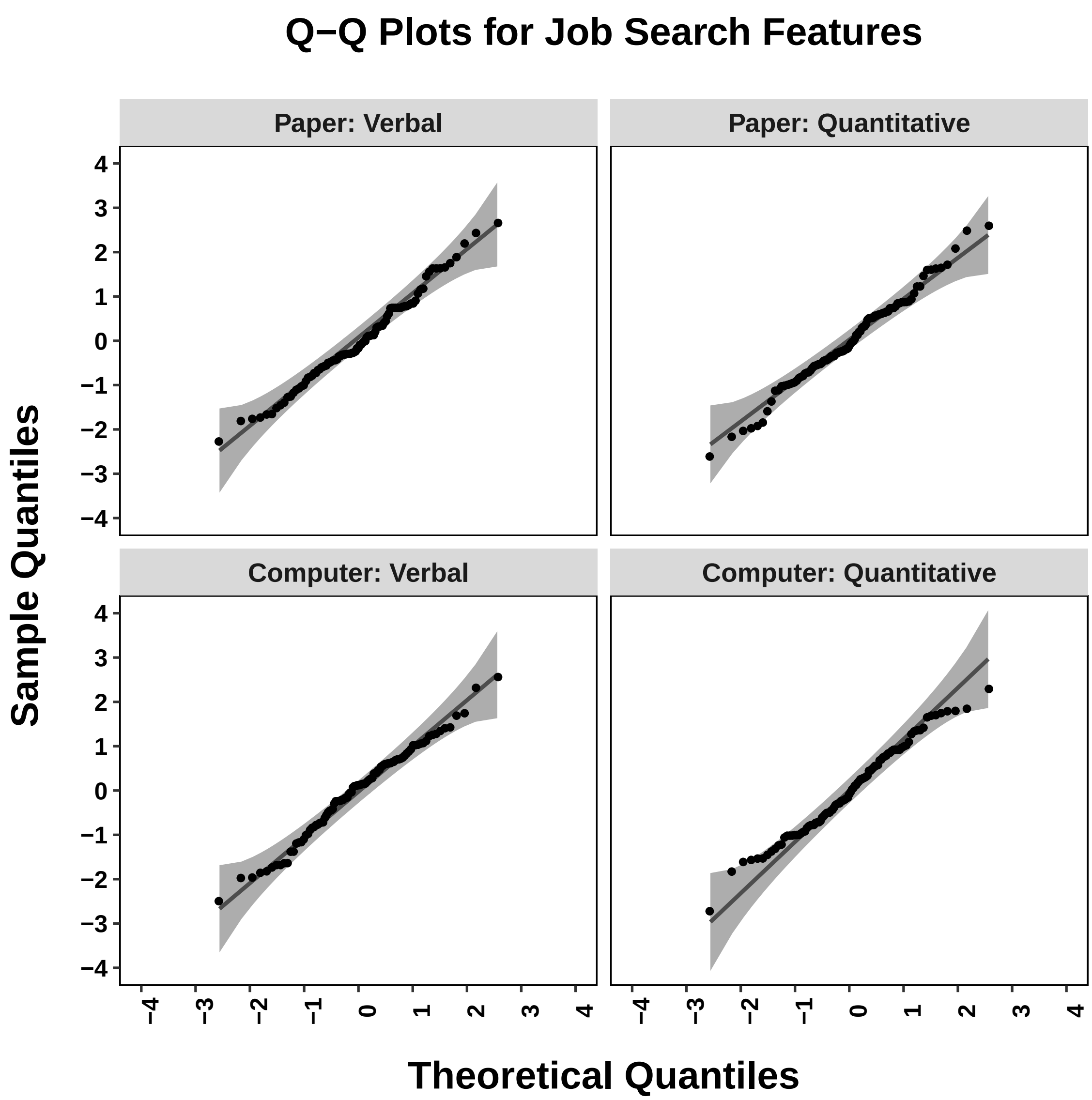
```
## $univariateNormality
##           Test   Variable Statistic   p value Normality
## 1 Shapiro-Wilk P_Verbal_R    0.9857    0.3630    YES
## 2 Shapiro-Wilk P_Quant_R     0.9889    0.5820    YES
## 3 Shapiro-Wilk C_Verbal_R    0.9872    0.4598    YES
## 4 Shapiro-Wilk C_Quant_R     0.9862    0.3923    YES
##
## $Descriptives
##             n   Mean Std.Dev  Median    Min   Max   25th  75th
## P_Verbal_R 99 0.3490   9.684 -1.2790 -21.66 26.09 -5.405 7.550
## P_Quant_R  99 0.3177   9.133 -0.7615 -23.53 24.02 -5.114 6.200
## C_Verbal_R 99 0.1044   9.922  1.3085 -24.66 25.50 -7.033 6.710
## C_Quant_R  99 0.2638   8.941 -0.4738 -24.09 20.75 -6.692 7.215
##                Skew Kurtosis
## P_Verbal_R  0.29712 -0.16599
## P_Quant_R   0.02811  0.07662
## C_Verbal_R -0.18123 -0.25780
## C_Quant_R   0.05667 -0.57682
```

```
CV <- cov(Skills_Trimmed[, 9:12])
D2_1 <- mahalanobis(Skills_Trimmed[, 9:12], center = colMeans(Skills_Trimmed[,
    9:12]), cov = CV)
D2_1 <- as.data.frame(D2_1)
ggplot(D2_1, aes(sample = D2_1)) + stat_qq_band(distribution = "chisq",
    dparams = list(df = 4)) + stat_qq_line(distribution = "chisq",
    dparams = list(df = 4)) + stat_qq(distribution = "qchisq", dparams = list(df = 4)) +
    scale_y_continuous(breaks = seq(0, 24, 2)) + scale_x_continuous(breaks = seq(0,
    16, 1)) + coord_cartesian(xlim = c(0, 16), ylim = c(0, 24)) +
    xlab(expression("Expected Values from" * ~chi[4]^2)) + ylab(expression("Mahalanobis " *
    ~D^2)) + theme(text = element_text(size = 14, family = "sans",
    color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 16), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 16), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle(expression("Q-Q Plot of Mahalanobis" *
    ~D^2 * " vs. Quantiles of" * ~chi[4]^2))
```




```
Skills_Trimmed_QQ <- scale(Skills_Trimmed[, 9:12])
Data_long <- melt(Skills_Trimmed_QQ)
Data_long <- as.data.frame(Data_long)
names(Data_long) <- c("Index", "feature", "value")
Data_long$feature_F <- factor(Data_long$feature, levels = c("P_Verbal_R",
    "P_Quant_R", "C_Verbal_R", "C_Quant_R"), labels = c("Paper: Verbal",
    "Paper: Quantitative", "Computer: Verbal", "Computer: Quantitative"))
p <- ggplot(Data_long, aes(sample = value)) + stat_qq_band() + stat_qq_line() +
    stat_qq(distribution = qnorm, size = 1) + scale_y_continuous(breaks = seq(-4,
    4, 1)) + scale_x_continuous(breaks = seq(-4, 4, 1)) + coord_cartesian(xlim = c(-4,
    4), ylim = c(-4, 4)) + xlab("Theoretical Quantiles") + ylab("Sample Quantiles") +
    theme(text = element_text(size = 14, family = "sans", color = "black",
        face = "bold"), axis.text.y = element_text(colour = "black",
        size = 10, face = "bold"), axis.text.x = element_text(colour = "black",
```

```
        size = 10, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
        0, 0, 0), size = 16), axis.title.y = element_text(margin = margin(0,
        15, 0, 0), size = 16), axis.line.x = element_blank(), axis.line.y = element_blank(),
        plot.title = element_text(size = 16, face = "bold", margin = margin(0,
            0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
            linetype = 1, color = "black"), panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
        plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
        legend.title = element_blank()) + ggtitle("Q-Q Plots for Job Search Features")
p + facet_wrap(~feature_F)
```



4 Homogeneity Assumption

We assume in discriminant analysis that the separate group variance-covariance matrices are homogeneous. Box's test can be used to test this assumption. Note, however, that it is also sensitive to violations of multivariate normality.

```
boxM(Skills[, 2:5], Skills$Group)

##
##  Box's M-test for Homogeneity of Covariance Matrices
##
## data:  Skills[, 2:5]
## Chi-Sq (approx.) = 93, df = 30, p-value = 2e-08

boxM(Skills[, 2:5], Skills$Group)$cov

## $`1`
##          P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal   112.10   94.54    67.22   46.25
## P_Quant     94.54   99.71    82.84   70.41
## C_Verbal    67.22   82.84   117.58  104.82
## C_Quant     46.25   70.41   104.82  114.46
##
## $`2`
##          P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal   164.89  121.83    90.29  121.84
## P_Quant    121.83  118.23    52.23   89.24
## C_Verbal    90.29   52.23   105.61   95.50
## C_Quant    121.84   89.24    95.50  120.37
##
## $`3`
##          P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal   122.97   77.14    60.02   34.16
## P_Quant     77.14   87.49    76.05   46.13
## C_Verbal    60.02   76.05   106.12   72.31
## C_Quant     34.16   46.13    72.31   81.09
##
## $`4`
##          P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal    33.24   37.85    33.51   22.40
## P_Quant     37.85   76.80    68.76   37.15
## C_Verbal    33.51   68.76    77.15   42.04
## C_Quant     22.40   37.15    42.04   39.21

boxM(Skills[, 2:5], Skills$Group)$pooled

##          P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal   108.30   82.84    62.76   56.16
## P_Quant     82.84   95.56    69.97   60.74
## C_Verbal    62.76   69.97   101.62   78.67
## C_Quant     56.16   60.74    78.67   88.79

boxM(Skills_Trimmed[, 2:5], Skills_Trimmed$Group)

##
##  Box's M-test for Homogeneity of Covariance Matrices
```

```
##
## data:  Skills_Trimmed[, 2:5]
## Chi-Sq (approx.) = 81, df = 30, p-value = 0.000002
```

```r
boxM(Skills_Trimmed[, 2:5], Skills_Trimmed$Group)$cov
```

```
## $`1`
##           P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal   112.10   94.54    67.22   46.25
## P_Quant     94.54   99.71    82.84   70.41
## C_Verbal    67.22   82.84   117.58  104.82
## C_Quant     46.25   70.41   104.82  114.46
##
## $`2`
##           P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal   117.98   77.91    78.04   86.27
## P_Quant     77.91   78.56    39.78   55.92
## C_Verbal    78.04   39.78   105.37   87.43
## C_Quant     86.27   55.92    87.43   94.73
##
## $`3`
##           P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal   122.97   77.14    60.02   34.16
## P_Quant     77.14   87.49    76.05   46.13
## C_Verbal    60.02   76.05   106.12   72.31
## C_Quant     34.16   46.13    72.31   81.09
##
## $`4`
##           P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal    33.24   37.85    33.51   22.40
## P_Quant     37.85   76.80    68.76   37.15
## C_Verbal    33.51   68.76    77.15   42.04
## C_Quant     22.40   37.15    42.04   39.21
```

```r
boxM(Skills_Trimmed[, 2:5], Skills_Trimmed$Group)$pooled
```

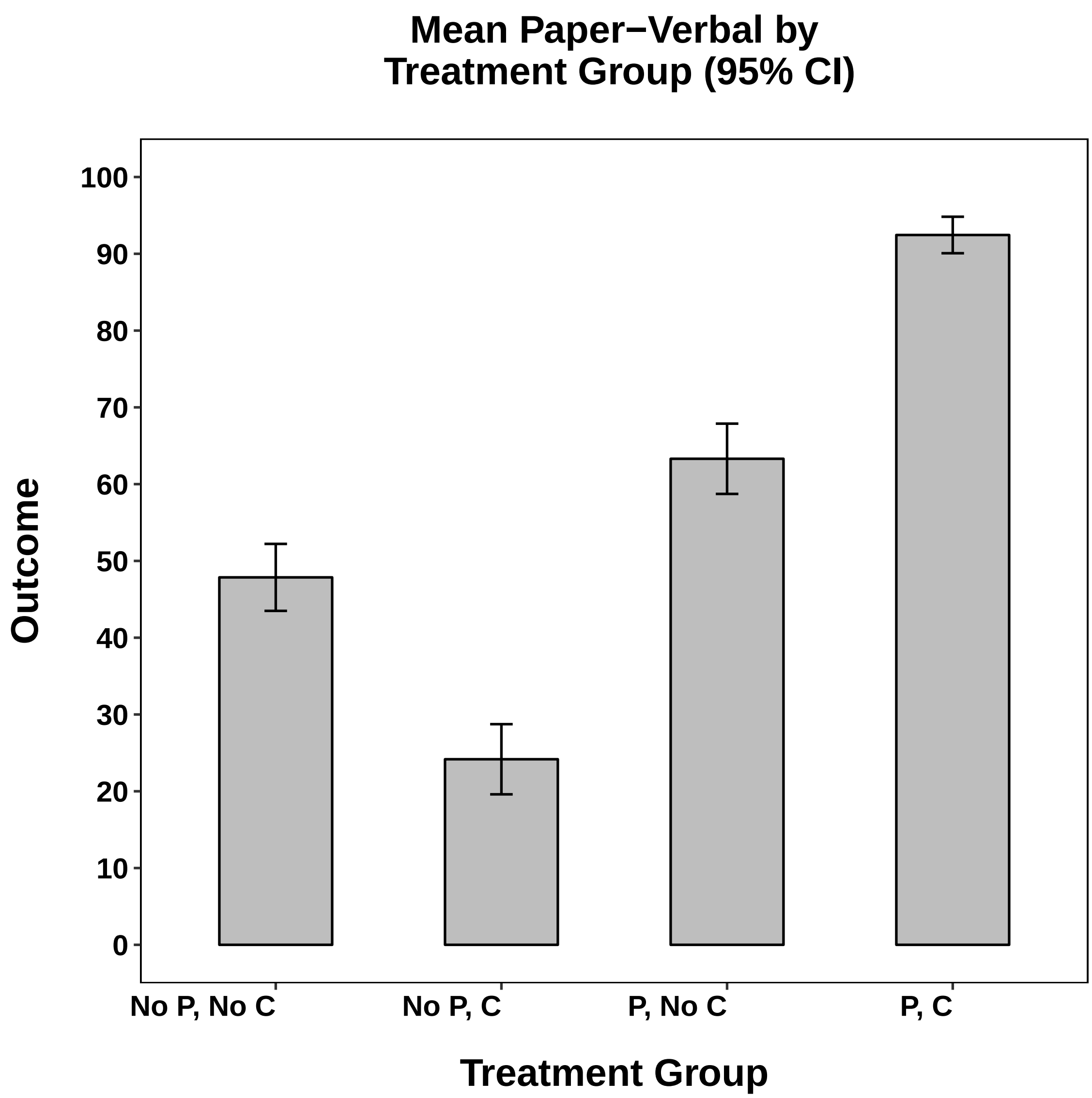
```
##           P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal    96.35   71.80    59.51   46.86
## P_Quant     71.80   85.71    67.14   52.37
## C_Verbal    59.51   67.14   101.52   76.54
## C_Quant     46.86   52.37    76.54   82.24
```

5 Means and Confidence Intervals

Displayed here are bar graphs of the condition means with 95% confidence intervals.

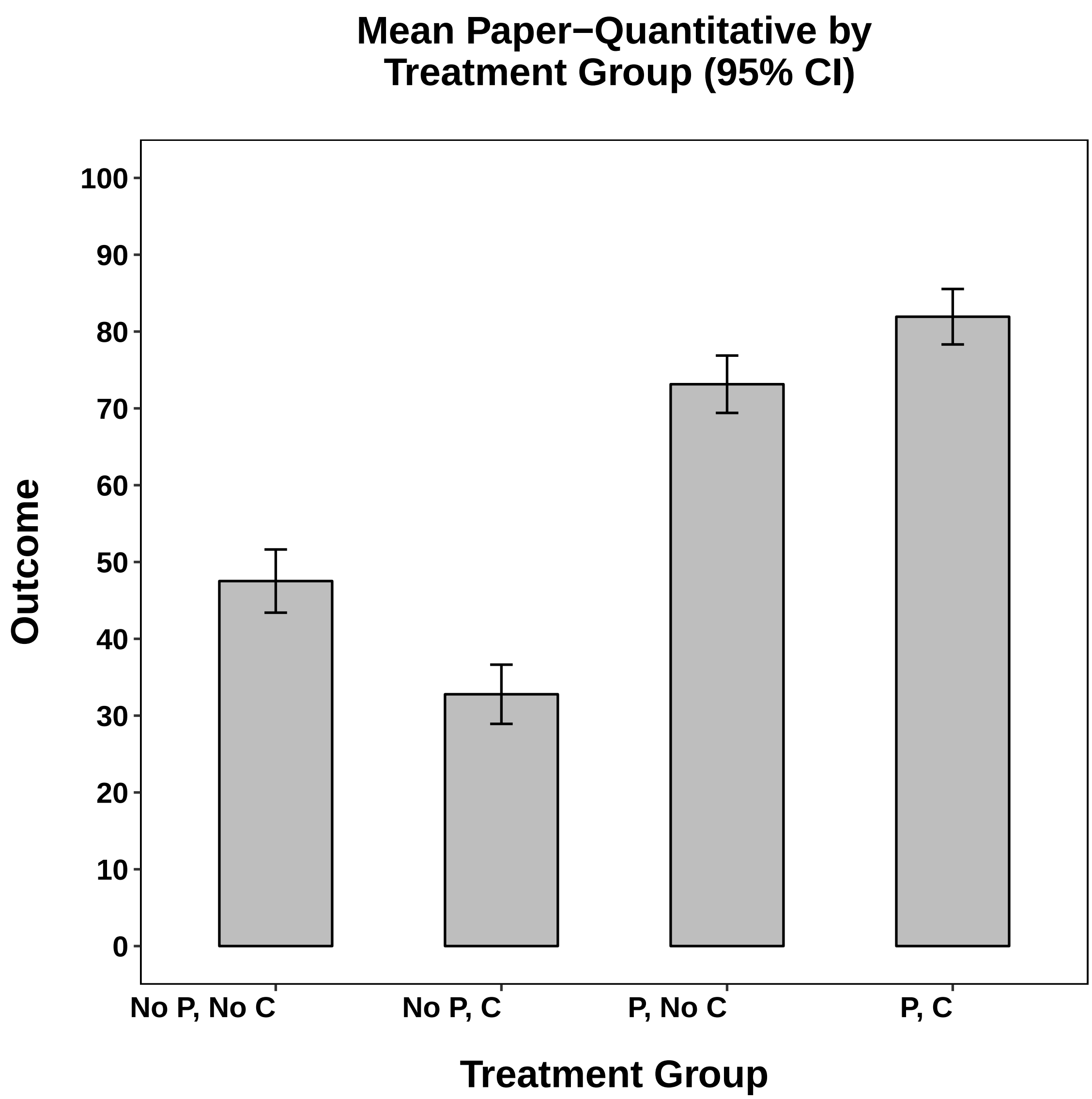
```
D <- describeBy(Skills_Trimmed[, 2:5], group = Skills_Trimmed$Group4)

plot_data <- matrix(NA, nrow = 4, ncol = 8)

for (i in 1:4) {
    for (j in 1:4) {
        plot_data[i, j] <- D[[i]]$mean[j]
        plot_data[i, j + 4] <- qt(0.975, D[[i]]$n[j]) * D[[i]]$sd[j]/sqrt(D[[i]]$n[j])
    }
}
plot_data <- as.data.frame(plot_data)
names(plot_data) <- c("PV_mean", "PQ_mean", "CV_mean", "CQ_mean",
    "PV_CI", "PQ_CI", "CV_CI", "CQ_CI")
plot_data$Group3 <- factor(c("No P, No C", "No P, C", "P, No C", "P, C"))
plot_data$Group4 <- factor(plot_data$Group3, levels = c("No P, No C",
    "No P, C", "P, No C", "P, C"), labels = c("No P, No C", "No P, C",
    "P, No C", "P, C"))
```

```
p1 <- ggplot(plot_data, aes(x = as.factor(Group4), y = PV_mean)) +
    geom_bar(position = position_dodge(), stat = "identity", color = "black",
        width = 0.5, fill = "grey") + geom_errorbar(aes(ymin = PV_mean -
    PV_CI, ymax = PV_mean + PV_CI), width = 0.1, position = position_dodge(0.5)) +
    scale_y_continuous(breaks = c(seq(0, 100, 10))) + coord_cartesian(ylim = c(0,
    100)) + xlab("Treatment Group") + ylab("Outcome") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 0, hjust = 1), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 16), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 16), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), panel.border = element_rect(fill = NA,
        size = 0.5), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Mean Paper-Verbal by\n Treatment Group (95% CI)")
print(p1)
```



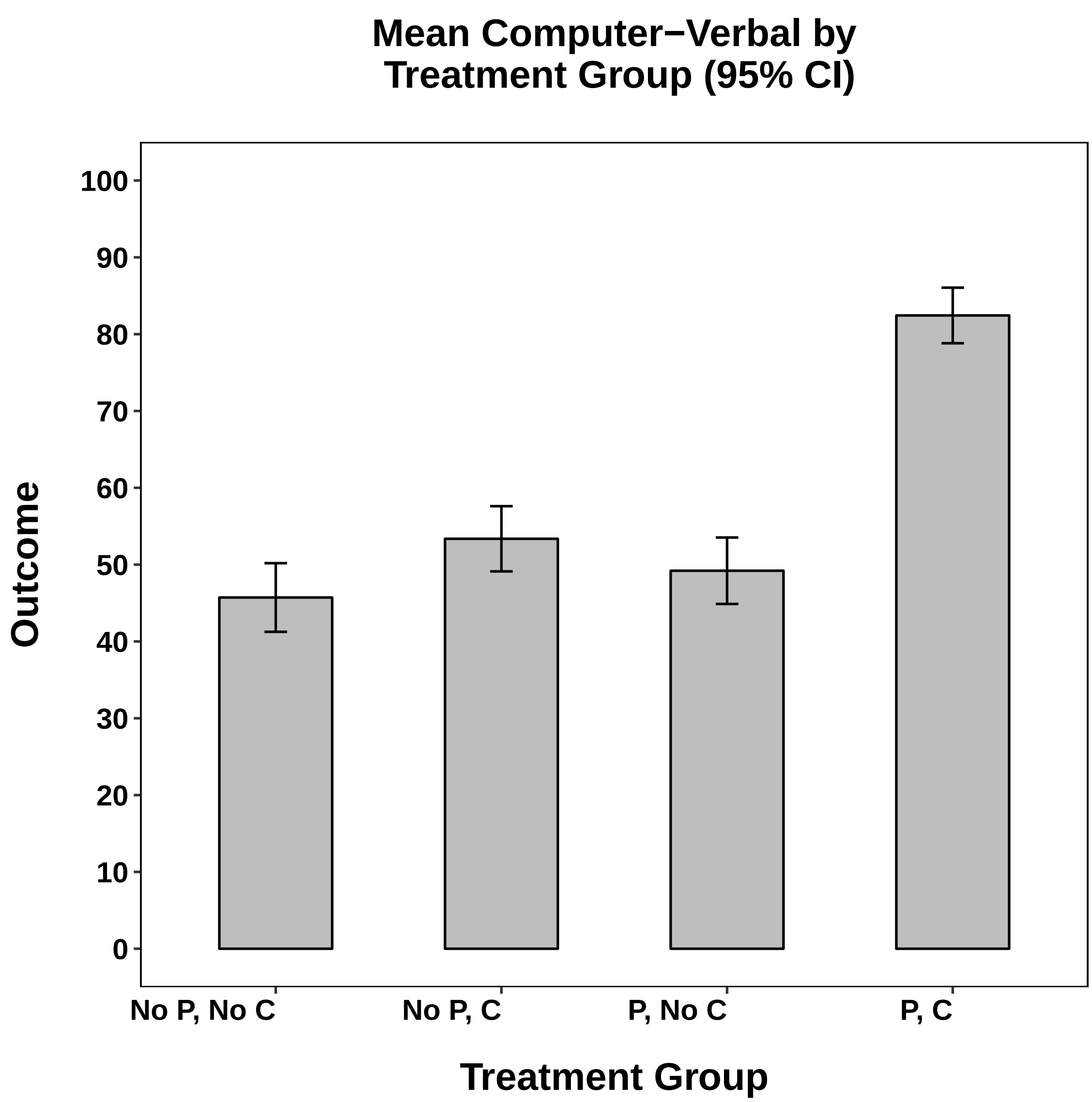

```
p2 <- ggplot(plot_data, aes(x = as.factor(Group4), y = PQ_mean)) +
    geom_bar(position = position_dodge(), stat = "identity", color = "black",
        width = 0.5, fill = "grey") + geom_errorbar(aes(ymin = PQ_mean -
    PQ_CI, ymax = PQ_mean + PQ_CI), width = 0.1, position = position_dodge(0.5)) +
    scale_y_continuous(breaks = c(seq(0, 100, 10))) + coord_cartesian(ylim = c(0,
    100)) + xlab("Treatment Group") + ylab("Outcome") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 0, hjust = 1), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 16), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 16), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
```

```r
    panel.grid.minor = element_blank(), panel.border = element_rect(fill = NA,
        size = 0.5), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Mean Paper-Quantitative by\n Treatment Group (95% CI)")
print(p2)
```

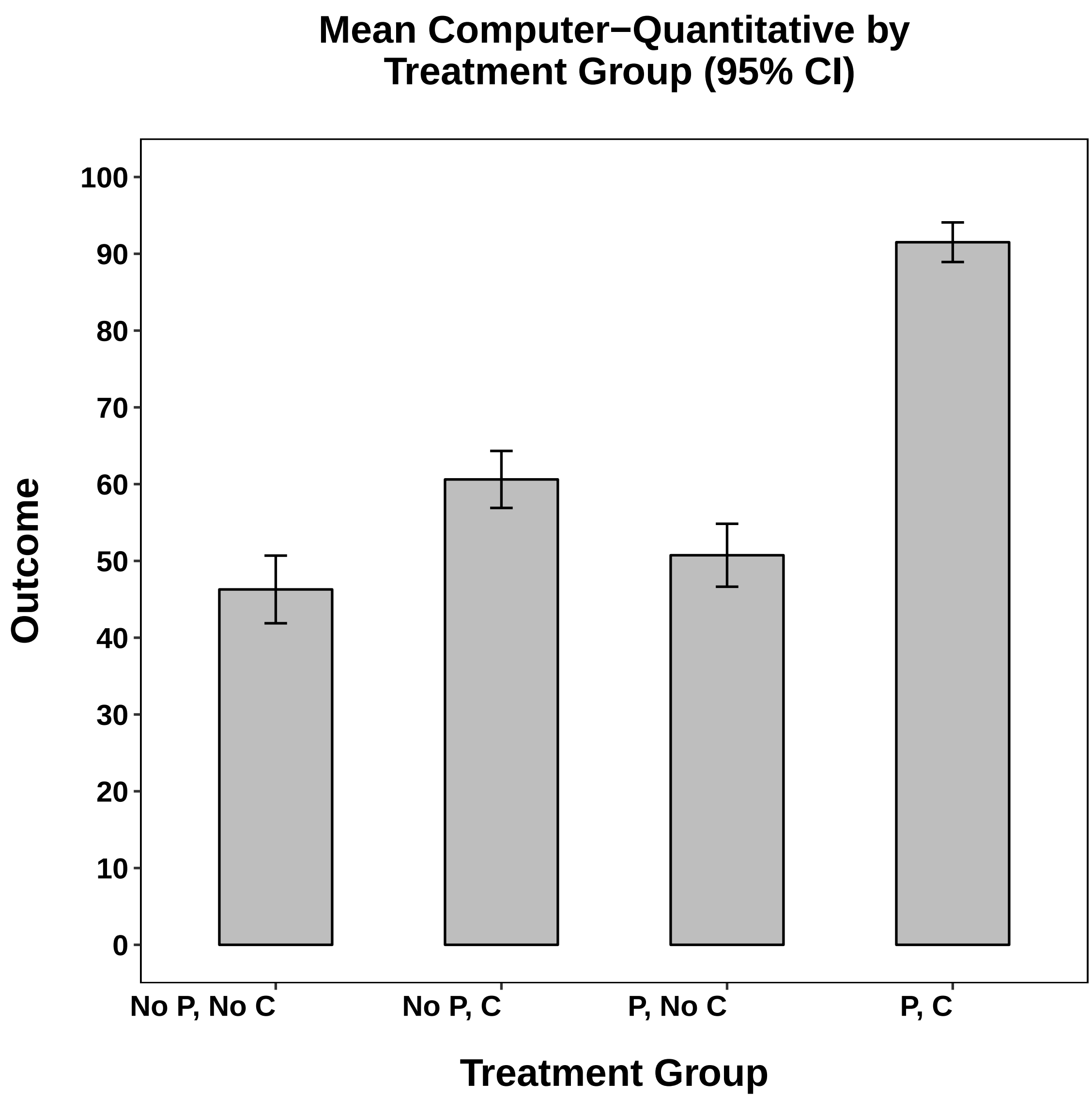



```r
p3 <- ggplot(plot_data, aes(x = as.factor(Group4), y = CV_mean)) +
    geom_bar(position = position_dodge(), stat = "identity", color = "black",
        width = 0.5, fill = "grey") + geom_errorbar(aes(ymin = CV_mean -
    CV_CI, ymax = CV_mean + CV_CI), width = 0.1, position = position_dodge(0.5)) +
    scale_y_continuous(breaks = c(seq(0, 100, 10))) + coord_cartesian(ylim = c(0,
    100)) + xlab("Treatment Group") + ylab("Outcome") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
```

```
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 0, hjust = 1), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 16), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 16), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), panel.border = element_rect(fill = NA,
        size = 0.5), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Mean Computer-Verbal by\n Treatment Group (95% CI)")
print(p3)
```



```r
p4 <- ggplot(plot_data, aes(x = as.factor(Group4), y = CQ_mean)) +
    geom_bar(position = position_dodge(), stat = "identity", color = "black",
        width = 0.5, fill = "grey") + geom_errorbar(aes(ymin = CQ_mean -
    CQ_CI, ymax = CQ_mean + CQ_CI), width = 0.1, position = position_dodge(0.5)) +
    scale_y_continuous(breaks = c(seq(0, 100, 10))) + coord_cartesian(ylim = c(0,
    100)) + xlab("Treatment Group") + ylab("Outcome") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 0, hjust = 1), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 16), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 16), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), panel.border = element_rect(fill = NA,
        size = 0.5), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Mean Computer-Quantitative by\n Treatment Group (95% CI)'
print(p4)
```




```
p1 <- ggplot(plot_data, aes(x = as.factor(Group4), y = PV_mean)) +
    geom_bar(position = position_dodge(), stat = "identity", color = "black",
        width = 0.5, fill = "grey") + geom_errorbar(aes(ymin = PV_mean -
    PV_CI, ymax = PV_mean + PV_CI), width = 0.1, position = position_dodge(0.5)) +
    scale_y_continuous(breaks = c(seq(0, 100, 20))) + coord_cartesian(ylim = c(0,
    100)) + xlab("Treatment Group") + ylab("Outcome") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 10, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 10, face = "bold", angle = 45, hjust = 1), axis.title.x = element_text(margin = margin(5,
    0, 0, 0), size = 12), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 12), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 14, face = "bold", margin = margin(0,
        0, 5, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
```

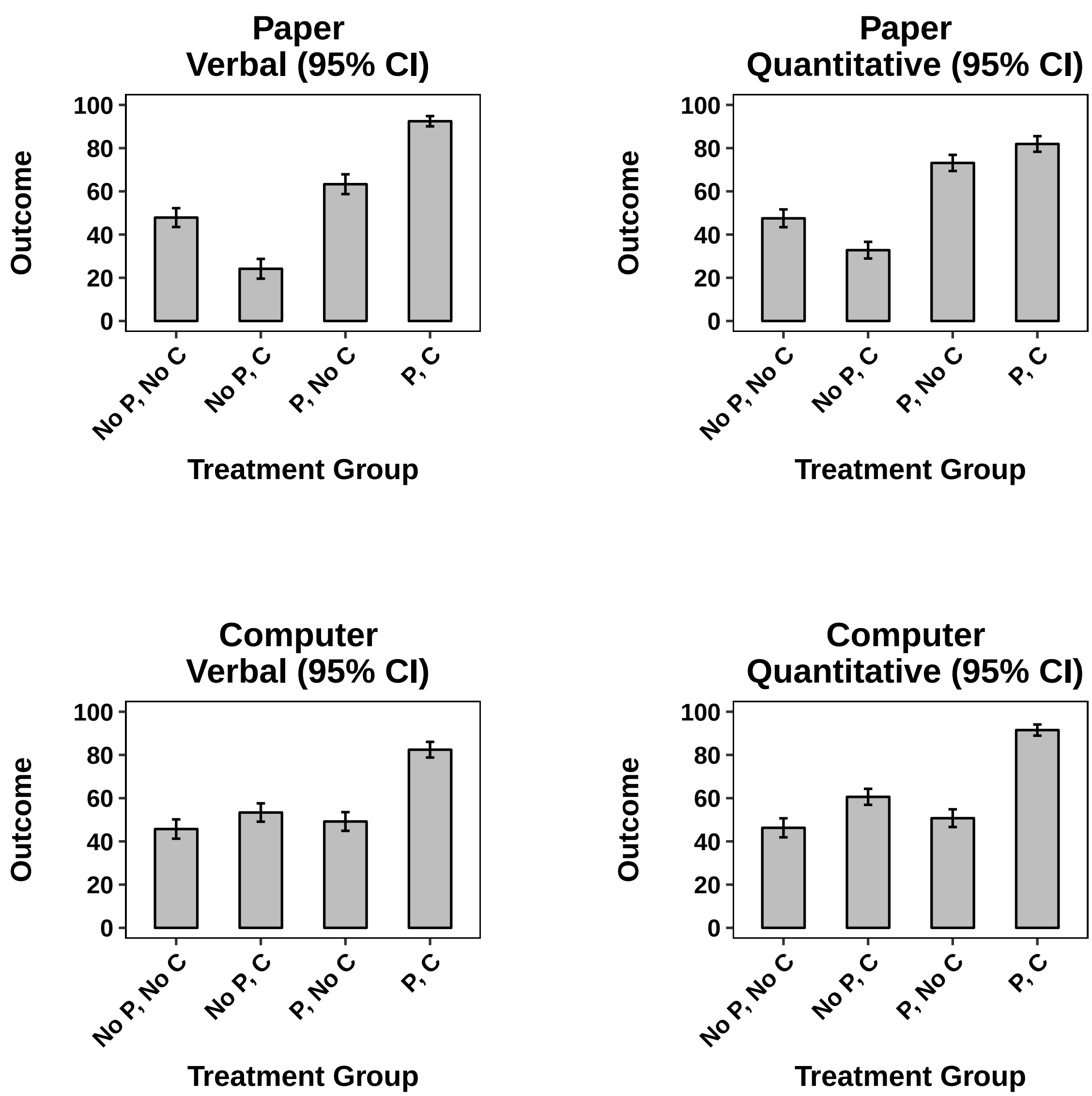
```
    panel.grid.minor = element_blank(), panel.border = element_rect(fill = NA,
        size = 0.5), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Paper \n Verbal (95% CI)")
```

```
p2 <- ggplot(plot_data, aes(x = as.factor(Group4), y = PQ_mean)) +
    geom_bar(position = position_dodge(), stat = "identity", color = "black",
        width = 0.5, fill = "grey") + geom_errorbar(aes(ymin = PQ_mean -
    PQ_CI, ymax = PQ_mean + PQ_CI), width = 0.1, position = position_dodge(0.5)) +
    scale_y_continuous(breaks = c(seq(0, 100, 20))) + coord_cartesian(ylim = c(0,
    100)) + xlab("Treatment Group") + ylab("Outcome") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 10, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 10, face = "bold", angle = 45, hjust = 1), axis.title.x = element_text(margin = margin(5,
    0, 0, 0), size = 12), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 12), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 14, face = "bold", margin = margin(0,
        0, 5, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), panel.border = element_rect(fill = NA,
        size = 0.5), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Paper \n Quantitative (95% CI)")
```

```
p3 <- ggplot(plot_data, aes(x = as.factor(Group4), y = CV_mean)) +
    geom_bar(position = position_dodge(), stat = "identity", color = "black",
        width = 0.5, fill = "grey") + geom_errorbar(aes(ymin = CV_mean -
    CV_CI, ymax = CV_mean + CV_CI), width = 0.1, position = position_dodge(0.5)) +
    scale_y_continuous(breaks = c(seq(0, 100, 20))) + coord_cartesian(ylim = c(0,
    100)) + xlab("Treatment Group") + ylab("Outcome") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 10, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 10, face = "bold", angle = 45, hjust = 1), axis.title.x = element_text(margin = margin(5,
    0, 0, 0), size = 12), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 12), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 14, face = "bold", margin = margin(0,
        0, 5, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), panel.border = element_rect(fill = NA,
        size = 0.5), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Computer \n Verbal (95% CI)")
```

```
p4 <- ggplot(plot_data, aes(x = as.factor(Group4), y = CQ_mean)) +
    geom_bar(position = position_dodge(), stat = "identity", color = "black",
        width = 0.5, fill = "grey") + geom_errorbar(aes(ymin = CQ_mean -
    CQ_CI, ymax = CQ_mean + CQ_CI), width = 0.1, position = position_dodge(0.5)) +
    scale_y_continuous(breaks = c(seq(0, 100, 20))) + coord_cartesian(ylim = c(0,
    100)) + xlab("Treatment Group") + ylab("Outcome") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 10, face = "bold"), axis.text.x = element_text(colour = "black",
```

```
    size = 10, face = "bold", angle = 45, hjust = 1), axis.title.x = element_text(margin = margin(5,
    0, 0, 0), size = 12), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 12), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 14, face = "bold", margin = margin(0,
        0, 5, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), panel.border = element_rect(fill = NA,
        size = 0.5), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Computer \n Quantitative (95% CI)")
grid.arrange(p1, p2, p3, p4, nrow = 2)
```



6 ANOVA of Each Outcome

Two forms of ANOVA are shown. In the first, no structure to the groups is assumed. In the second, the factorial is included in the design.

6.1 No Group Structure

```
AOV_1 <- aov(P_Verbal ~ as.factor(Group), data = Skills_Trimmed)
summary(AOV_1)

##                  Df Sum Sq Mean Sq F value Pr(>F)
## as.factor(Group)  3  61407   20469     212 <2e-16
## Residuals        95   9153      96

TukeyHSD(AOV_1)

##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = P_Verbal ~ as.factor(Group), data = Skills_Trimmed)
##
## $`as.factor(Group)`
##       diff     lwr    upr p adj
## 2-1  15.45   8.112  22.78     0
## 3-1 -23.69 -30.947 -16.43     0
## 4-1  44.59  37.334  51.85     0
## 3-2 -39.13 -46.469 -31.80     0
## 4-2  29.15  21.812  36.48     0
## 4-3  68.28  61.021  75.54     0

AOV_2 <- aov(P_Quant ~ as.factor(Group), data = Skills_Trimmed)
summary(AOV_2)

##                  Df Sum Sq Mean Sq F value Pr(>F)
## as.factor(Group)  3  38419   12806     149 <2e-16
## Residuals        95   8143      86

TukeyHSD(AOV_2)

##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = P_Quant ~ as.factor(Group), data = Skills_Trimmed)
##
## $`as.factor(Group)`
##        diff     lwr     upr  p adj
## 2-1  25.625  18.706  32.544 0.0000
## 3-1 -14.735 -21.583  -7.888 0.0000
## 4-1  34.415  27.567  41.263 0.0000
## 3-2 -40.361 -47.280 -33.442 0.0000
## 4-2   8.789   1.871  15.708 0.0068
## 4-3  49.150  42.302  55.998 0.0000

AOV_3 <- aov(C_Verbal ~ as.factor(Group), data = Skills_Trimmed)
summary(AOV_3)
```

```
##                   Df Sum Sq Mean Sq F value Pr(>F)
## as.factor(Group)  3  21084    7028    69.2 <2e-16
## Residuals        95   9644     102
```

```
TukeyHSD(AOV_3)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = C_Verbal ~ as.factor(Group), data = Skills_Trimmed)
##
## $`as.factor(Group)`
##       diff     lwr   upr  p adj
## 2-1  3.484 -4.0452 11.01 0.6220
## 3-1  7.643  0.1904 15.10 0.0422
## 4-1 36.714 29.2612 44.17 0.0000
## 3-2  4.158 -3.3712 11.69 0.4752
## 4-2 33.229 25.6995 40.76 0.0000
## 4-3 29.071 21.6183 36.52 0.0000
```

```
AOV_4 <- aov(C_Quant ~ as.factor(Group), data = Skills_Trimmed)
summary(AOV_4)
```

```
##                   Df Sum Sq Mean Sq F value Pr(>F)
## as.factor(Group)  3  31016   10339     126 <2e-16
## Residuals        95   7813      82
```

```
TukeyHSD(AOV_4)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = C_Quant ~ as.factor(Group), data = Skills_Trimmed)
##
## $`as.factor(Group)`
##       diff    lwr   upr  p adj
## 2-1  4.456 -2.322 11.23 0.3195
## 3-1 14.328  7.621 21.04 0.0000
## 4-1 45.223 38.515 51.93 0.0000
## 3-2  9.873  3.096 16.65 0.0014
## 4-2 40.767 33.990 47.54 0.0000
## 4-3 30.894 24.186 37.60 0.0000
```

6.2 Factorial Group Structure

```
AOV_5 <- aov(P_Verbal ~ Tx_P * Tx_C, data = Skills_Trimmed)
summary(AOV_5)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Tx_P         1  43991   43991  456.58 <2e-16
## Tx_C         1    150     150    1.55   0.22
## Tx_P:Tx_C    1  17267   17267  179.21 <2e-16
## Residuals   95   9153      96
```

```
AOV_6 <- aov(P_Quant ~ Tx_P * Tx_C, data = Skills_Trimmed)
summary(AOV_6)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## Tx_P         1  34759   34759  405.53 < 2e-16
## Tx_C         1    237     237    2.76     0.1
## Tx_P:Tx_C    1   3423    3423   39.94 8.4e-09
## Residuals   95   8143      86
```

```
AOV_7 <- aov(C_Verbal ~ Tx_P * Tx_C, data = Skills_Trimmed)
summary(AOV_7)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## Tx_P         1   6833    6833    67.3 1.1e-12
## Tx_C         1  10201   10201   100.5 < 2e-16
## Tx_P:Tx_C    1   4049    4049    39.9 8.6e-09
## Residuals   95   9644     102
```

```
AOV_8 <- aov(C_Quant ~ Tx_P * Tx_C, data = Skills_Trimmed)
summary(AOV_8)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## Tx_P         1   8099    8099    98.5 2.4e-16
## Tx_C         1  18593   18593   226.1 < 2e-16
## Tx_P:Tx_C    1   4324    4324    52.6 1.1e-10
## Residuals   95   7813      82
```

6.3 No-Intercept Model

A no-intercept approach has the advantage that any comparisons can be specified.

```
AOV_9 <- lm(P_Verbal ~ -1 + D1 + D2 + D3 + D4, data = Skills_Trimmed)
summary(AOV_9)
```

```
##
## Call:
## lm(formula = P_Verbal ~ -1 + D1 + D2 + D3 + D4, data = Skills_Trimmed)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -21.66  -6.20  -1.29   7.50  24.65
##
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)
## D1    47.86       1.96    24.4   <2e-16
## D2    63.30       2.00    31.6   <2e-16
## D3    24.17       1.96    12.3   <2e-16
## D4    92.45       1.96    47.1   <2e-16
##
## Residual standard error: 9.82 on 95 degrees of freedom
## Multiple R-squared:  0.977,Adjusted R-squared:  0.976
## F-statistic:  990 on 4 and 95 DF,  p-value: <2e-16
```

```
AOV_10 <- lm(P_Quant ~ -1 + D1 + D2 + D3 + D4, data = Skills_Trimmed)
summary(AOV_10)
```

```
##
## Call:
## lm(formula = P_Quant ~ -1 + D1 + D2 + D3 + D4, data = Skills_Trimmed)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -23.53  -5.42  -1.25   6.06  24.02
##
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)
## D1    47.52       1.85    25.7   <2e-16
## D2    73.14       1.89    38.7   <2e-16
## D3    32.78       1.85    17.7   <2e-16
## D4    81.93       1.85    44.2   <2e-16
##
## Residual standard error: 9.26 on 95 degrees of freedom
## Multiple R-squared:  0.979,Adjusted R-squared:  0.978
## F-statistic: 1.11e+03 on 4 and 95 DF,  p-value: <2e-16
```

```
AOV_11 <- lm(C_Verbal ~ -1 + D1 + D2 + D3 + D4, data = Skills_Trimmed)
summary(AOV_11)
```

```
##
## Call:
## lm(formula = C_Verbal ~ -1 + D1 + D2 + D3 + D4, data = Skills_Trimmed)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -24.66  -7.11   1.08   6.58  25.50
##
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)
## D1    45.72       2.02    22.7   <2e-16
## D2    49.20       2.06    23.9   <2e-16
## D3    53.36       2.02    26.5   <2e-16
## D4    82.43       2.02    40.9   <2e-16
##
## Residual standard error: 10.1 on 95 degrees of freedom
## Multiple R-squared:  0.973,Adjusted R-squared:  0.972
## F-statistic:  865 on 4 and 95 DF,  p-value: <2e-16
```

```
AOV_12 <- lm(C_Quant ~ -1 + D1 + D2 + D3 + D4, data = Skills_Trimmed)
summary(AOV_12)
```

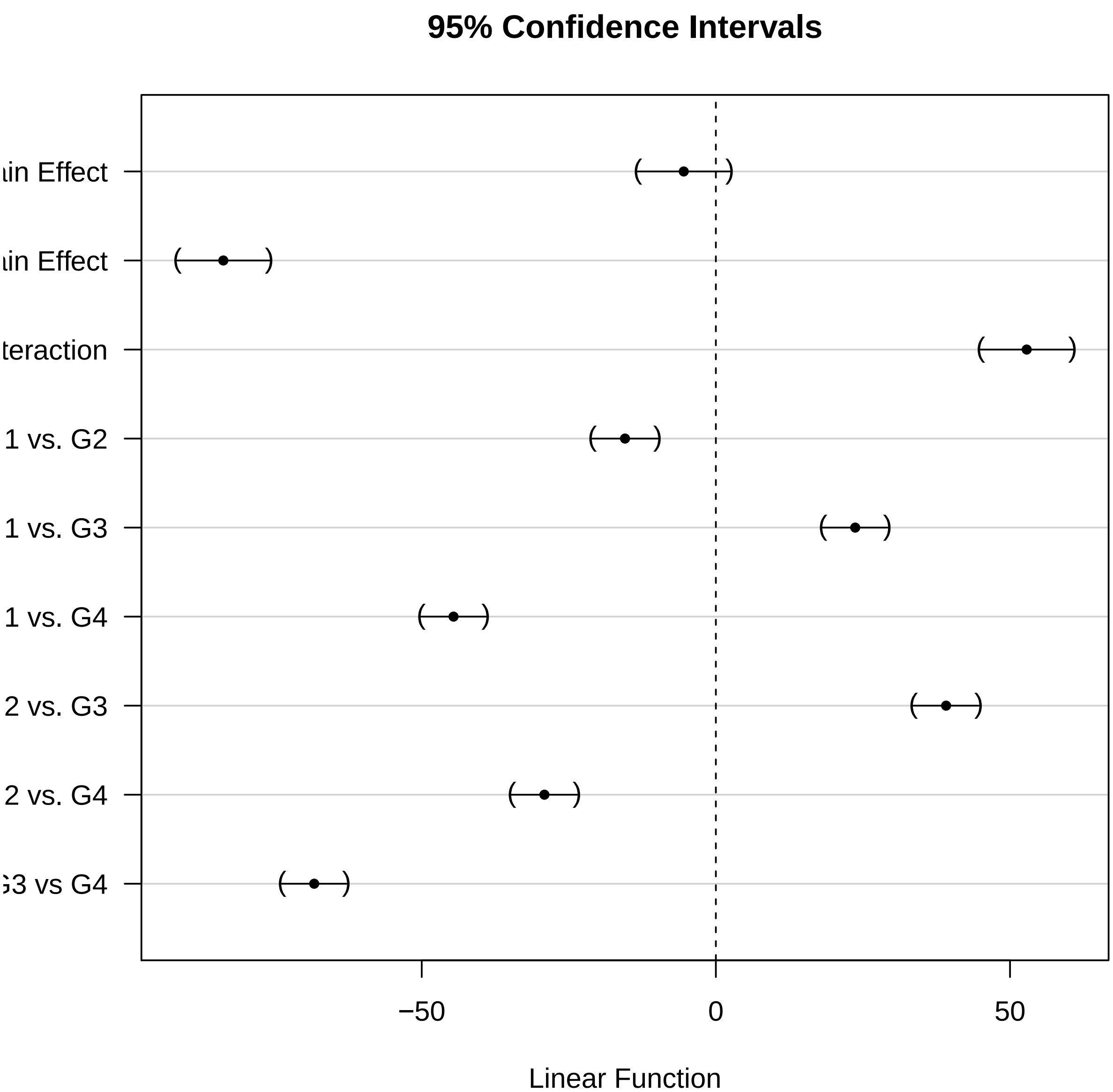
```
##
## Call:
## lm(formula = C_Quant ~ -1 + D1 + D2 + D3 + D4, data = Skills_Trimmed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.086  -6.736  -0.474   7.215  19.666
##
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)
## D1    46.28       1.81    25.5   <2e-16
```

```
## D2    50.74       1.85    27.4   <2e-16
## D3    60.61       1.81    33.4   <2e-16
## D4    91.51       1.81    50.5   <2e-16
##
## Residual standard error: 9.07 on 95 degrees of freedom
## Multiple R-squared:  0.982,Adjusted R-squared:  0.981
## F-statistic: 1.27e+03 on 4 and 95 DF,  p-value: <2e-16
```

```r
LM = matrix(c(1, 1, -1, -1, 1, -1, 1, -1, 1, -1, -1, 1, 1, -1, 0,
    0, 1, 0, -1, 0, 1, 0, 0, -1, 0, 1, -1, 0, 0, 1, 0, -1, 0, 0, 1,
    -1), nrow = 9, ncol = 4, byrow = TRUE)
rownames(LM) <- c("Mode Main Effect", "Domain Main Effect", "Interaction",
    "G1 vs. G2", "G1 vs. G3", "G1 vs. G4", "G2 vs. G3", "G2 vs. G4",
    "G3 vs G4")
LM
```

```
##                    [,1] [,2] [,3] [,4]
## Mode Main Effect      1    1   -1   -1
## Domain Main Effect    1   -1    1   -1
## Interaction           1   -1   -1    1
## G1 vs. G2             1   -1    0    0
## G1 vs. G3             1    0   -1    0
## G1 vs. G4             1    0    0   -1
## G2 vs. G3             0    1   -1    0
## G2 vs. G4             0    1    0   -1
## G3 vs G4              0    0    1   -1
```

```r
glht_LM_9 <- glht(AOV_9, linfct = LM, alternative = "two.sided", rhs = 0)
plot(confint(glht_LM_9, calpha = univariate_calpha()), main = "95% Confidence Intervals")
```




```
summary(glht_LM_9, adjusted("holm"))

##
##   Simultaneous Tests for General Linear Hypotheses
##
## Fit: lm(formula = P_Verbal ~ -1 + D1 + D2 + D3 + D4, data = Skills_Trimmed)
##
## Linear Hypotheses:
##                            Estimate Std. Error t value Pr(>|t|)
## Mode Main Effect == 0        -5.46       3.95   -1.38     0.17
## Domain Main Effect == 0     -83.73       3.95  -21.21  < 2e-16
## Interaction == 0             52.83       3.95   13.39  < 2e-16
## G1 vs. G2 == 0              -15.45       2.81   -5.51  6.2e-07
## G1 vs. G3 == 0               23.69       2.78    8.53  6.8e-13
```

```
## G1 vs. G4 == 0            -44.59       2.78  -16.06  < 2e-16
## G2 vs. G3 == 0             39.13       2.81   13.95  < 2e-16
## G2 vs. G4 == 0            -29.15       2.81  -10.39  < 2e-16
## G3 vs G4 == 0             -68.28       2.78  -24.59  < 2e-16
## (Adjusted p values reported -- holm method)
```

```
confint(glht_LM_9, p.adjust(method = "holm"))
```

```
##
##   Simultaneous Confidence Intervals
##
## Fit: lm(formula = P_Verbal ~ -1 + D1 + D2 + D3 + D4, data = Skills_Trimmed)
##
## Quantile = 2.686
## 95% family-wise confidence level
##
##
## Linear Hypotheses:
##                          Estimate lwr     upr
## Mode Main Effect == 0     -5.461  -16.063   5.142
## Domain Main Effect == 0 -83.728  -94.331 -73.126
## Interaction == 0          52.834   42.232  63.437
## G1 vs. G2 == 0           -15.447  -22.983  -7.911
## G1 vs. G3 == 0            23.687   16.228  31.145
## G1 vs. G4 == 0           -44.594  -52.053 -37.136
## G2 vs. G3 == 0            39.134   31.598  46.669
## G2 vs. G4 == 0           -29.147  -36.683 -21.612
## G3 vs G4 == 0            -68.281  -75.740 -60.823
```

6.4 Sums of Squares

Three different kinds of sums of squares can be used to partition the variance in ANOVA. Type I SS are sequential, allocating to an effect the variance it accounts for at its entry step. Type II SS represent the contribution an effect makes after controlling for any lower-order effects and any effects of the same order. In other words, in a three-factor design (A x B x C) the AB interaction would have all main effects as well as the AC and BC interactions controlled. Type III SS represent the unique contribution of an effect after all other effects have been controlled. This is the most conservative and the default in SPSS, for example. Different functions use different SS methods by default and the calculations can depend on whether the independent variables are defined as factors or not.

The SS type will matter if the independent variables are correlated. In a balanced design (equal n in a factorial structure), all effects are orthogonal and all SS methods will produce the same results. To make the consequences more apparent in the demonstration that follows, we'll create a big imbalance in the cell sample sizes.

A summary of the consequences for different approaches is given after the demonstrated methods.

```
# Create an imbalanced data set.
for (i in 1:length(Skills_Trimmed[, 1])) {
    Skills_Trimmed[i, "Index"] <- i
```

```
}
Skills_Sub <- Skills_Trimmed[Skills_Trimmed$Index < 10 | Skills_Trimmed$Index >
    30, ]

replications(P_Verbal ~ Tx_P * Tx_C, data = Skills_Sub)

## $Tx_P
## Tx_P
## No Tx(P)    Tx(P)
##       34       44
##
## $Tx_C
## Tx_C
## No Tx(C)    Tx(C)
##       28       50
##
## $`Tx_P:Tx_C`
##           Tx_C
## Tx_P       No Tx(C) Tx(C)
##   No Tx(P)        9    25
##   Tx(P)          19    25
```

6.4.1 The aov( ) Function

The aov( ) function will allocate effects according to order of entry, but the results will depend on whether the predictors are defined as factors. Order will be respected within levels of effects (i.e., within main effects, within two-way interactions, etc.).

```
AOV_5a <- aov(P_Verbal ~ Tx_P + Tx_C + Tx_P:Tx_C, data = Skills_Sub)
summary(AOV_5a)

##             Df Sum Sq Mean Sq F value  Pr(>F)
## Tx_P         1  43993   43993   524.1 < 2e-16
## Tx_C         1   1136    1136    13.5 0.00044
## Tx_P:Tx_C    1  13804   13804   164.4 < 2e-16
## Residuals   74   6211      84

AOV_5b <- aov(P_Verbal ~ Tx_C + Tx_P + Tx_P:Tx_C, data = Skills_Sub)
summary(AOV_5b)

##             Df Sum Sq Mean Sq F value Pr(>F)
## Tx_C         1      9       9    0.11   0.74
## Tx_P         1  45120   45120  537.54 <2e-16
## Tx_C:Tx_P    1  13804   13804  164.45 <2e-16
## Residuals   74   6211      84

AOV_5c <- aov(P_Verbal ~ Tx_C + Tx_P:Tx_C + Tx_P, data = Skills_Sub)
summary(AOV_5c)

##             Df Sum Sq Mean Sq F value Pr(>F)
## Tx_C         1      9       9    0.11   0.74
## Tx_P         1  45120   45120  537.54 <2e-16
## Tx_C:Tx_P    1  13804   13804  164.45 <2e-16
## Residuals   74   6211      84
```

To get complete control over order of entry, the predictors should not be defined as factors. But, note the last two models. Model 5f uses the I( ) function to calculate the product on the fly. This model respects the intended order. Model 5g uses the to produce the interaction. This model does not respect the intended order.

```
AOV_5d <- aov(P_Verbal ~ Tx_P_NF + Tx_C_NF + I(Tx_P_NF * Tx_C_NF),
    data = Skills_Sub)
summary(AOV_5d)

##                     Df Sum Sq Mean Sq F value  Pr(>F)
## Tx_P_NF              1  43993   43993   524.1 < 2e-16
## Tx_C_NF              1   1136    1136    13.5 0.00044
## I(Tx_P_NF * Tx_C_NF)  1  13804   13804   164.4 < 2e-16
## Residuals           74   6211      84

AOV_5e <- aov(P_Verbal ~ Tx_C_NF + Tx_P_NF + I(Tx_P_NF * Tx_C_NF),
    data = Skills_Sub)
summary(AOV_5e)

##                     Df Sum Sq Mean Sq F value Pr(>F)
## Tx_C_NF              1      9       9    0.11   0.74
## Tx_P_NF              1  45120   45120  537.54 <2e-16
## I(Tx_P_NF * Tx_C_NF)  1  13804   13804  164.45 <2e-16
## Residuals           74   6211      84

AOV_5f <- aov(P_Verbal ~ Tx_C_NF + I(Tx_P_NF * Tx_C_NF) + Tx_P_NF,
    data = Skills_Sub)
summary(AOV_5f)

##                     Df Sum Sq Mean Sq F value      Pr(>F)
## Tx_C_NF              1      9       9    0.11        0.74
## I(Tx_P_NF * Tx_C_NF)  1  55825   55825  665.06     < 2e-16
## Tx_P_NF              1   3099    3099   36.92 0.000000049
## Residuals           74   6211      84

AOV_5g <- aov(P_Verbal ~ Tx_C_NF + Tx_P_NF:Tx_C_NF + Tx_P_NF, data = Skills_Sub)
summary(AOV_5g)

##                 Df Sum Sq Mean Sq F value Pr(>F)
## Tx_C_NF          1      9       9    0.11   0.74
## Tx_P_NF          1  45120   45120  537.54 <2e-16
## Tx_C_NF:Tx_P_NF  1  13804   13804  164.45 <2e-16
## Residuals       74   6211      84
```

6.4.2 The lm( ) Function

The lm( ) function will use Type III SS if the predictors are not defined as factors. If they are defined as factors, then lm( ) appears to use Type II SS. Note that an attempt to use the I( ) to produce a product with variables defined as factors produces an error.

```
lm_5a <- lm(P_Verbal ~ Tx_P + Tx_C + Tx_P:Tx_C, data = Skills_Sub)
summary(lm_5a)

##
## Call:
```

```
## lm(formula = P_Verbal ~ Tx_P + Tx_C + Tx_P:Tx_C, data = Skills_Sub)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -17.18  -6.38  -1.28   7.55  23.43
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   57.750      1.131   51.07   <2e-16
## Tx_P1        -19.639      1.131  -17.37   <2e-16
## Tx_C1         -0.559      1.131   -0.49     0.62
## Tx_P1:Tx_C1   14.501      1.131   12.82   <2e-16
##
## Residual standard error: 9.16 on 74 degrees of freedom
## Multiple R-squared:  0.905,Adjusted R-squared:  0.901
## F-statistic:  234 on 3 and 74 DF,  p-value: <2e-16
```

```
lm_5b <- lm(P_Verbal ~ Tx_C + Tx_P + Tx_P:Tx_C, data = Skills_Sub)
summary(lm_5b)
```

```
##
## Call:
## lm(formula = P_Verbal ~ Tx_C + Tx_P + Tx_P:Tx_C, data = Skills_Sub)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -17.18  -6.38  -1.28   7.55  23.43
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   57.750      1.131   51.07   <2e-16
## Tx_C1         -0.559      1.131   -0.49     0.62
## Tx_P1        -19.639      1.131  -17.37   <2e-16
## Tx_C1:Tx_P1   14.501      1.131   12.82   <2e-16
##
## Residual standard error: 9.16 on 74 degrees of freedom
## Multiple R-squared:  0.905,Adjusted R-squared:  0.901
## F-statistic:  234 on 3 and 74 DF,  p-value: <2e-16
```

```
lm_5c <- lm(P_Verbal ~ Tx_C + Tx_P:Tx_C + Tx_P, data = Skills_Sub)
summary(lm_5c)
```

```
##
## Call:
## lm(formula = P_Verbal ~ Tx_C + Tx_P:Tx_C + Tx_P, data = Skills_Sub)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -17.18  -6.38  -1.28   7.55  23.43
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   57.750      1.131   51.07   <2e-16
## Tx_C1         -0.559      1.131   -0.49     0.62
## Tx_P1        -19.639      1.131  -17.37   <2e-16
```

```
## Tx_C1:Tx_P1   14.501      1.131   12.82   <2e-16
##
## Residual standard error: 9.16 on 74 degrees of freedom
## Multiple R-squared:  0.905,Adjusted R-squared:  0.901
## F-statistic:  234 on 3 and 74 DF,  p-value: <2e-16
```

The last two models here are also of note. Model lm_5f respects the order. Model lm_5g does not. It makes no difference in this case because a Type III partition is used.

```
lm_5d <- lm(P_Verbal ~ Tx_P_NF + Tx_C_NF + I(Tx_P_NF * Tx_C_NF), data = Skills_Sub)
summary(lm_5d)

##
## Call:
## lm(formula = P_Verbal ~ Tx_P_NF + Tx_C_NF + I(Tx_P_NF * Tx_C_NF),
##     data = Skills_Sub)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -17.18  -6.38  -1.28   7.55  23.43
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)            127.66      13.55    9.42  2.7e-14
## Tx_P_NF                -47.73       7.85   -6.08  4.9e-08
## Tx_C_NF                -85.89       7.65  -11.23  < 2e-16
## I(Tx_P_NF * Tx_C_NF)    58.00       4.52   12.82  < 2e-16
##
## Residual standard error: 9.16 on 74 degrees of freedom
## Multiple R-squared:  0.905,Adjusted R-squared:  0.901
## F-statistic:  234 on 3 and 74 DF,  p-value: <2e-16

lm_5e <- lm(P_Verbal ~ Tx_C_NF + Tx_P_NF + I(Tx_P_NF * Tx_C_NF), data = Skills_Sub)
summary(lm_5e)

##
## Call:
## lm(formula = P_Verbal ~ Tx_C_NF + Tx_P_NF + I(Tx_P_NF * Tx_C_NF),
##     data = Skills_Sub)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -17.18  -6.38  -1.28   7.55  23.43
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)            127.66      13.55    9.42  2.7e-14
## Tx_C_NF                -85.89       7.65  -11.23  < 2e-16
## Tx_P_NF                -47.73       7.85   -6.08  4.9e-08
## I(Tx_P_NF * Tx_C_NF)    58.00       4.52   12.82  < 2e-16
##
## Residual standard error: 9.16 on 74 degrees of freedom
## Multiple R-squared:  0.905,Adjusted R-squared:  0.901
## F-statistic:  234 on 3 and 74 DF,  p-value: <2e-16
```

```
lm_5f <- lm(P_Verbal ~ Tx_C_NF + I(Tx_P_NF * Tx_C_NF) + Tx_P_NF, data = Skills_Sub)
summary(lm_5f)

##
## Call:
## lm(formula = P_Verbal ~ Tx_C_NF + I(Tx_P_NF * Tx_C_NF) + Tx_P_NF,
##     data = Skills_Sub)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -17.18  -6.38  -1.28   7.55  23.43
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)              127.66      13.55    9.42  2.7e-14
## Tx_C_NF                  -85.89       7.65  -11.23  < 2e-16
## I(Tx_P_NF * Tx_C_NF)      58.00       4.52   12.82  < 2e-16
## Tx_P_NF                  -47.73       7.85   -6.08  4.9e-08
##
## Residual standard error: 9.16 on 74 degrees of freedom
## Multiple R-squared:  0.905,Adjusted R-squared:  0.901
## F-statistic:  234 on 3 and 74 DF,  p-value: <2e-16

lm_5g <- lm(P_Verbal ~ Tx_C_NF + Tx_P_NF:Tx_C_NF + Tx_P_NF, data = Skills_Sub)
summary(lm_5g)

##
## Call:
## lm(formula = P_Verbal ~ Tx_C_NF + Tx_P_NF:Tx_C_NF + Tx_P_NF,
##     data = Skills_Sub)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -17.18  -6.38  -1.28   7.55  23.43
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         127.66      13.55    9.42  2.7e-14
## Tx_C_NF             -85.89       7.65  -11.23  < 2e-16
## Tx_P_NF             -47.73       7.85   -6.08  4.9e-08
## Tx_C_NF:Tx_P_NF      58.00       4.52   12.82  < 2e-16
##
## Residual standard error: 9.16 on 74 degrees of freedom
## Multiple R-squared:  0.905,Adjusted R-squared:  0.901
## F-statistic:  234 on 3 and 74 DF,  p-value: <2e-16
```

Type III sums of squares allocates to each effect its unique variance accounted for in the outcome. One way to get Type III tests is to test the full model against models that exclude each effect in turn.

```
drop1(AOV_5a, ~., test = "F")

## Single term deletions
##
## Model:
```

```
## P_Verbal ~ Tx_P + Tx_C + Tx_P:Tx_C
##            Df Sum of Sq   RSS AIC F value Pr(>F)
## <none>                  6211 349
## Tx_P        1     25319 31530 474  301.64 <2e-16
## Tx_C        1        21  6232 348    0.24   0.62
## Tx_P:Tx_C   1     13804 20015 439  164.45 <2e-16
```

```
drop1(AOV_5d, ~., test = "F")
```

```
## Single term deletions
##
## Model:
## P_Verbal ~ Tx_P_NF + Tx_C_NF + I(Tx_P_NF * Tx_C_NF)
##                       Df Sum of Sq   RSS AIC F value      Pr(>F)
## <none>                            6211 349
## Tx_P_NF                1      3099  9311 379    36.9 0.000000049
## Tx_C_NF                1     10582 16794 425   126.1     < 2e-16
## I(Tx_P_NF * Tx_C_NF)  1     13804 20015 439   164.4     < 2e-16
```

Another approach is offered by the car package. The Anova( ) function (note the capitalization) allows Type II and Type III models. It is a bit more complex to use because it takes as input an object from a linear model fit.

```
lm <- lm(P_Verbal ~ Tx_P + Tx_C + Tx_P:Tx_C, data = Skills_Sub)
AOV_5g <- Anova(lm, type = "II")
AOV_5g
```

```
## Anova Table (Type II tests)
##
## Response: P_Verbal
##           Sum Sq Df F value  Pr(>F)
## Tx_P       45120  1   537.5 < 2e-16
## Tx_C        1136  1    13.5 0.00044
## Tx_P:Tx_C  13804  1   164.4 < 2e-16
## Residuals   6211 74
```

```
AOV_5h <- Anova(lm, type = "III")
AOV_5h
```

```
## Anova Table (Type III tests)
##
## Response: P_Verbal
##             Sum Sq Df F value Pr(>F)
## (Intercept) 218923  1 2608.13 <2e-16
## Tx_P         25319  1  301.64 <2e-16
## Tx_C            21  1    0.24   0.62
## Tx_P:Tx_C    13804  1  164.45 <2e-16
## Residuals     6211 74
```

```
lm <- lm(P_Verbal ~ Tx_P_NF + Tx_C_NF + Tx_P_NF:Tx_C_NF, data = Skills_Sub)
AOV_5i <- Anova(lm, type = "II")
AOV_5i
```

```
## Anova Table (Type II tests)
##
```

```
## Response: P_Verbal
##                 Sum Sq Df F value  Pr(>F)
## Tx_P_NF          45120  1   537.5 < 2e-16
## Tx_C_NF           1136  1    13.5 0.00044
## Tx_P_NF:Tx_C_NF  13804  1   164.4 < 2e-16
## Residuals         6211 74

AOV_5j <- Anova(lm, type = "III")
AOV_5j

## Anova Table (Type III tests)
##
## Response: P_Verbal
##                 Sum Sq Df F value  Pr(>F)
## (Intercept)       7448  1    88.7 2.7e-14
## Tx_P_NF           3099  1    36.9 4.9e-08
## Tx_C_NF          10582  1   126.1 < 2e-16
## Tx_P_NF:Tx_C_NF  13804  1   164.4 < 2e-16
## Residuals         6211 74

lm <- lm(P_Verbal ~ Tx_P_NF + Tx_C_NF + I(Tx_P_NF * Tx_C_NF), data = Skills_Sub)
AOV_5k <- Anova(lm, type = "II")
AOV_5k

## Anova Table (Type II tests)
##
## Response: P_Verbal
##                      Sum Sq Df F value      Pr(>F)
## Tx_P_NF                3099  1    36.9 0.000000049
## Tx_C_NF               10582  1   126.1     < 2e-16
## I(Tx_P_NF * Tx_C_NF)  13804  1   164.4     < 2e-16
## Residuals              6211 74

AOV_5l <- Anova(lm, type = "III")
AOV_5l

## Anova Table (Type III tests)
##
## Response: P_Verbal
##                      Sum Sq Df F value  Pr(>F)
## (Intercept)            7448  1    88.7 2.7e-14
## Tx_P_NF                3099  1    36.9 4.9e-08
## Tx_C_NF               10582  1   126.1 < 2e-16
## I(Tx_P_NF * Tx_C_NF)  13804  1   164.4 < 2e-16
## Residuals              6211 74
```

6.4.3 Summary

The aov( ) function uses Type I sums of squares, but will respect order only within levels of effects (main effects, interactions) if the predictors are defined as factors. If true order of entry is desired, then the predictors should not be defined as factors and the explicit model definition used to carefully control order.

Order does not matter with the lm( ) function, but predictor type does. If predictors

are not defined as factors, then true Type III sums of squares are given (unique effects). If predictors are defined as factors, then the results will be the same as drop1( ) with factors and Anova( ) with factors. Not clear what is going on here because they are not true Type III effects, but appear to be some hybrid of Type III and Type II.

The drop1( ) function when used with predictors in lm( ) that are not defined as factors will produce true Type III effects. When drop1( ) is used with predictors defined as factors, the previously alluded to hybrid effects are produced.

The Anova( ) function when used with predictors defined as factors in lm( ) will produce true Type II effects when that is requested. When Type III effects are requested, the hybrid results are produced. It makes sense that these are called Type III effects; they are the same as produced by lm( ) with factor predictors (because lm( ) is a Type III procedure). When the Anova( ) is used with predictors in lm( ) that are not defined as factors, true Type II and Type III effects are provided, depending on what is requested.

Bottom line. If true Type III are desired, don't use factors. If true Type I are desired, don't use factors. If true Type II are desired, use Anova and it doesn't matter if factors are used or not. The Anova( ) allows specification of Type II and Type III sums of squares.

*But . . . it seems to matter whether I(A*B) or A B is used when variables are not defined as factors (if they ARE factors, the I( ) cannot be used). More scrutiny is required. Stay tuned.*

7 Repeated Measures ANOVA

7.1 The aov( ) Function, Part I

The repeated measures also have a factorial structure which is incorporated into the analyses in a traditional ANOVA approach. In the following, the aov( ) function is used along with predictors not defined as factors. This will produce true Type I effects. Note that these analyses are possible because the data are in wide format and the repeated measures are incorporated into linear combinations that represent the sum, main effects, and interaction from the within-subjects part of the design. We request the intercepts for the within-subjects linear combinations because they represent the main effects.

```
# Sum over repeated measures.  This produces the between-subjects
# part of the design.
AOV_13 <- aov(I(P_Verbal + P_Quant + C_Verbal + C_Quant) ~ Tx_P_NF +
    Tx_C_NF + I(Tx_P_NF * Tx_C_NF), data = Skills_Trimmed)
summary(AOV_13, intercept = TRUE)

##                      Df  Sum Sq Mean Sq F value  Pr(>F)
## (Intercept)           1 5502086 5502086  4938.0 < 2e-16
## Tx_P_NF               1  323576  323576   290.4 < 2e-16
## Tx_C_NF               1   54846   54846    49.2 3.3e-10
## I(Tx_P_NF * Tx_C_NF)  1  101954  101954    91.5 1.4e-15
## Residuals            95  105852    1114

# Mode: The difference between paper and computer measures.
AOV_14 <- aov(I(P_Verbal + P_Quant - C_Verbal - C_Quant) ~ Tx_P_NF +
    Tx_C_NF + I(Tx_P_NF * Tx_C_NF), data = Skills_Trimmed)
summary(AOV_14, intercept = TRUE)

##                      Df Sum Sq Mean Sq F value   Pr(>F)
## (Intercept)           1   2086    2086     9.9   0.0022
## Tx_P_NF               1  49961   49961   237.1   < 2e-16
## Tx_C_NF               1  57851   57851   274.5   < 2e-16
## I(Tx_P_NF * Tx_C_NF)  1   3663    3663    17.4 0.000068
## Residuals            95  20020     211

# Domain: The difference between verbal and quantitative measures.
AOV_15 <- aov(I(P_Verbal - P_Quant + C_Verbal - C_Quant) ~ Tx_P_NF +
    Tx_C_NF + I(Tx_P_NF * Tx_C_NF), data = Skills_Trimmed)
summary(AOV_15, intercept = TRUE)

##                      Df Sum Sq Mean Sq F value  Pr(>F)
## (Intercept)           1   4125    4125   63.55 3.5e-12
## Tx_P_NF               1    255     255    3.93    0.05
## Tx_C_NF               1     60      60    0.92    0.34
## I(Tx_P_NF * Tx_C_NF)  1   5009    5009   77.17 6.5e-14
## Residuals            95   6166      65

# Mode x Domain interaction
AOV_16 <- aov(I(P_Verbal - P_Quant - C_Verbal + C_Quant) ~ Tx_P_NF +
    Tx_C_NF + I(Tx_P_NF * Tx_C_NF), data = Skills_Trimmed)
summary(AOV_16, intercept = TRUE)

##                      Df Sum Sq Mean Sq F value  Pr(>F)
## (Intercept)           1    786     786    10.7 0.00149
```

```
## Tx_P_NF              1    939     939    12.8 0.00055
## Tx_C_NF              1   3966    3966    54.0 6.9e-11
## I(Tx_P_NF * Tx_C_NF)  1   5627    5627    76.7 7.5e-14
## Residuals           95   6974      73
```

7.2 The aov( ) Function, Part II

The aov( ) function can also be used with a file in long format, requiring the specification of the within-subjects part of the design as part of the formula on the right-hand side. Note, however, that this function will test all effects against a common residual term, which is only appropriate if the pooled effects are homogeneous. This may be happening because the model is not balanced. The ezANOVA( ) function provides a better option that produces separate error terms for each within-subjects effect.

```
# Create a long form of the file.
Skills_Long <- matrix(NA, nrow = 4 * length(Skills_Trimmed[, 1]),
    ncol = 7)
Skills_Long <- as.data.frame(Skills_Long)
names(Skills_Long) <- c("ID", "Subject", "Outcome", "Mode", "Domain",
    "Tx_P", "Tx_C")

counter <- 0
for (i in 1:length(Skills_Trimmed[, 1])) {
    for (j in 1:4) {
        counter <- counter + 1
        Skills_Long[counter, "ID"] <- counter
        Skills_Long[counter, "Subject"] <- Skills_Trimmed[i, "ID"]
        Skills_Long[counter, "Tx_P"] <- Skills_Trimmed[i, "Tx_P"]
        Skills_Long[counter, "Tx_C"] <- Skills_Trimmed[i, "Tx_C"]
        if (j == 1) {
            Skills_Long[counter, "Outcome"] <- Skills_Trimmed[i, "P_Verbal"]
            Skills_Long[counter, "Mode"] <- "Paper"
            Skills_Long[counter, "Domain"] <- "Verbal"
        } else if (j == 2) {
            Skills_Long[counter, "Outcome"] <- Skills_Trimmed[i, "P_Quant"]
            Skills_Long[counter, "Mode"] <- "Paper"
            Skills_Long[counter, "Domain"] <- "Quant"
        } else if (j == 3) {
            Skills_Long[counter, "Outcome"] <- Skills_Trimmed[i, "C_Verbal"]
            Skills_Long[counter, "Mode"] <- "Computer"
            Skills_Long[counter, "Domain"] <- "Verbal"
        } else {
            Skills_Long[counter, "Outcome"] <- Skills_Trimmed[i, "C_Quant"]
            Skills_Long[counter, "Mode"] <- "Computer"
            Skills_Long[counter, "Domain"] <- "Quant"
        }
    }
}

Skills_Long <- as.data.frame(Skills_Long)
Skills_Long$Mode <- as.factor(Skills_Long$Mode)
Skills_Long$Domain <- as.factor(Skills_Long$Domain)
Skills_Long$Tx_P <- as.factor(Skills_Long$Tx_P)
```

```r
Skills_Long$Tx_C <- as.factor(Skills_Long$Tx_C)
Skills_Long$Subject <- as.factor(Skills_Long$Subject)

replications(Outcome ~ Tx_P * Tx_C, data = Skills_Long)
```

```
## $Tx_P
## Tx_P
##   1   2
## 200 196
##
## $Tx_C
## Tx_C
##   1   2
## 196 200
##
## $`Tx_P:Tx_C`
##     Tx_C
## Tx_P   1   2
##    1 100 100
##    2  96 100
```

```r
# Full repeated measures ANOVA using aov( ).
AOV_17 <- aov(Outcome ~ (Mode * Domain * Tx_P * Tx_C) + Error(Subject/(Mode *
    Domain)) + (Tx_P * Tx_C), data = Skills_Long)
summary(AOV_17)
```

```
##
## Error: Subject
##           Df Sum Sq Mean Sq F value  Pr(>F)
## Tx_P       1  80894   80894   290.4 < 2e-16
## Tx_C       1  13712   13712    49.2 3.3e-10
## Tx_P:Tx_C  1  25488   25488    91.5 1.4e-15
## Residuals 95  26463     279
##
## Error: Subject:Mode
##                Df Sum Sq Mean Sq F value   Pr(>F)
## Mode            1    522     522     9.9   0.0022
## Mode:Tx_P       1  12490   12490   237.1  < 2e-16
## Mode:Tx_C       1  14463   14463   274.5  < 2e-16
## Mode:Tx_P:Tx_C  1    916     916    17.4 0.000068
## Residuals      95   5005      53
##
## Error: Subject:Domain
##                  Df Sum Sq Mean Sq F value  Pr(>F)
## Domain            1   1031    1031   63.55 3.5e-12
## Domain:Tx_P       1     64      64    3.93    0.05
## Domain:Tx_C       1     15      15    0.92    0.34
## Domain:Tx_P:Tx_C  1   1252    1252   77.17 6.5e-14
## Residuals        95   1541      16
##
## Error: Subject:Mode:Domain
##                       Df Sum Sq Mean Sq F value  Pr(>F)
## Mode:Domain            1    197     197    10.7 0.00149
```

```
## Mode:Domain:Tx_P       1    235     235    12.8 0.00055
## Mode:Domain:Tx_C       1    991     991    54.0 6.9e-11
## Mode:Domain:Tx_P:Tx_C  1   1407    1407    76.7 7.5e-14
## Residuals             95   1743      18
```

7.3 The ezANOVA( ) Function

The ezANOVA( ) function produces separate error terms for each within-subjects effect, an effect size estimate, and the option to indicate the sums of squares type.

```
# Full repeated measures ANOVA using ezANOVA( ).
AOV_18a <- ezANOVA(dv = Outcome, wid = (Subject), within = .(Mode,
    Domain), between = .(Tx_P, Tx_C), data = Skills_Long, type = 2,
    detailed = TRUE)

## Warning: Data is unbalanced (unequal N per group). Make sure you specified a well-considered
value for the type argument to ezANOVA().

AOV_18a

## $ANOVA
##                   Effect DFn DFd        SSn   SSd        F
## 1            (Intercept)   1  95 1375521.56 26463 4937.988
## 2                   Tx_P   1  95   80207.43 26463  287.937
## 3                   Tx_C   1  95   13711.60 26463   49.223
## 5                   Mode   1  95     521.58  5005    9.900
## 9                 Domain   1  95    1031.15  1541   63.548
## 4              Tx_P:Tx_C   1  95   25488.38 26463   91.501
## 6              Tx_P:Mode   1  95   12764.70  5005  242.293
## 7              Tx_C:Mode   1  95   14462.82  5005  274.526
## 10           Tx_P:Domain   1  95      64.37  1541    3.967
## 11           Tx_C:Domain   1  95      14.96  1541    0.922
## 13           Mode:Domain   1  95     196.52  1743   10.708
## 8         Tx_P:Tx_C:Mode   1  95     915.68  5005   17.381
## 12      Tx_P:Tx_C:Domain   1  95    1252.25  1541   77.174
## 14      Tx_P:Mode:Domain   1  95     224.88  1743   12.253
## 15      Tx_C:Mode:Domain   1  95     991.45  1743   54.023
## 16 Tx_P:Tx_C:Mode:Domain   1  95    1406.80  1743   76.655
##             p p<.05       ges
## 1  1.050e-83     * 0.9753573
## 2  1.643e-30     * 0.6976962
## 3  3.348e-10     * 0.2829201
## 5  2.207e-03     * 0.0147862
## 9  3.453e-12     * 0.0288159
## 4  1.401e-15     * 0.4231044
## 6  6.975e-28     * 0.2686306
## 7  8.989e-30     * 0.2938654
## 10 4.927e-02     * 0.0018488
## 11 3.394e-01       0.0004303
## 13 1.488e-03     * 0.0056230
## 8  6.752e-05     * 0.0256719
## 12 6.519e-14     * 0.0347798
## 14 7.091e-04     * 0.0064292
## 15 6.871e-11     * 0.0277372
## 16 7.538e-14     * 0.0389051
```

```r
AOV_18b <- ezANOVA(dv = Outcome, wid = (Subject), within = .(Mode,
    Domain), between = .(Tx_P, Tx_C), data = Skills_Long, type = 3,
    detailed = TRUE)
```

```
## Warning: Data is unbalanced (unequal N per group). Make sure you specified a well-considered
value for the type argument to ezANOVA().
```

```r
AOV_18b
```

```
## $ANOVA
##                    Effect DFn DFd        SSn   SSd         F
## 1             (Intercept)   1  95 1375166.74 26463 4936.7141
## 2                    Tx_P   1  95   79269.40 26463  284.5694
## 3                    Tx_C   1  95   14098.29 26463   50.6115
## 5                    Mode   1  95     432.25  5005    8.2047
## 9                  Domain   1  95    1046.60  1541   64.5004
## 4               Tx_P:Tx_C   1  95   25488.38 26463   91.5008
## 6               Tx_P:Mode   1  95   12692.95  5005  240.9308
## 7               Tx_C:Mode   1  95   14386.35  5005  273.0740
## 10            Tx_P:Domain   1  95      58.64  1541    3.6141
## 11            Tx_C:Domain   1  95      12.27  1541    0.7562
## 13            Mode:Domain   1  95     181.26  1743    9.8764
## 8          Tx_P:Tx_C:Mode   1  95     915.68  5005   17.3810
## 12       Tx_P:Tx_C:Domain   1  95    1252.25  1541   77.1743
## 14       Tx_P:Mode:Domain   1  95     213.41  1743   11.6284
## 15       Tx_C:Mode:Domain   1  95    1015.84  1743   55.3524
## 16 Tx_P:Tx_C:Mode:Domain   1  95    1406.80  1743   76.6552
##            p p<.05       ges
## 1  1.063e-83     * 0.9753511
## 2  2.503e-30     * 0.6952092
## 3  2.106e-10     * 0.2885962
## 5  5.143e-03     * 0.0122849
## 9  2.587e-12     * 0.0292351
## 4  1.401e-15     * 0.4231044
## 6  8.460e-28     * 0.2675246
## 7  1.084e-29     * 0.2927665
## 10 6.032e-02       0.0016846
## 11 3.867e-01       0.0003529
## 13 2.233e-03     * 0.0051885
## 8  6.752e-05     * 0.0256719
## 12 6.519e-14     * 0.0347798
## 14 9.550e-04     * 0.0061032
## 15 4.474e-11     * 0.0284003
## 16 7.538e-14     * 0.0389051
```

7.4 The Anova( ) Function

The Anova( ) function produces multivariate tests, but not univariate tests, at least here for the first method shown. A separate effect sum of squares and cross-products matrix, and separate effect multivariate tests, are provided. Type II and Type III sums of squares can be requested.

```
# Full repeated measures ANOVA using Anova( ).
LM <- lm(cbind(P_Verbal, P_Quant, C_Verbal, C_Quant) ~ Tx_P * Tx_C,
    data = Skills_Trimmed)
AOV_19 <- Anova(LM, data = Skills_Trimmed, type = 2)
summary(AOV_19)

##
## Type II MANOVA Tests:
##
## Sum of squares and products for error:
##          P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal     9153    6821     5653    4452
## P_Quant      6821    8143     6379    4975
## C_Verbal     5653    6379     9644    7271
## C_Quant      4452    4975     7271    7813
##
## ------------------------------------------
##
## Term: Tx_P
##
## Sum of squares and products for the hypothesis:
##          P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal    43934   39109    17110   18571
## P_Quant     39109   34814    15231   16532
## C_Verbal    17110   15231     6663    7232
## C_Quant     18571   16532     7232    7850
##
## Multivariate Tests: Tx_P
##                  Df test stat approx F num Df den Df Pr(>F)
## Pillai            1     0.863    145.4      4     92 <2e-16
## Wilks             1     0.137    145.4      4     92 <2e-16
## Hotelling-Lawley  1     6.323    145.4      4     92 <2e-16
## Roy               1     6.323    145.4      4     92 <2e-16
##
## ------------------------------------------
##
## Term: Tx_C
##
## Sum of squares and products for the hypothesis:
##          P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal    149.5  -188.2     1235    1667
## P_Quant    -188.2   236.9    -1555   -2099
## C_Verbal   1235.0 -1554.6    10201   13772
## C_Quant    1667.3 -2098.8    13772   18593
##
## Multivariate Tests: Tx_C
##                  Df test stat approx F num Df den Df Pr(>F)
## Pillai            1     0.821    105.8      4     92 <2e-16
## Wilks             1     0.179    105.8      4     92 <2e-16
## Hotelling-Lawley  1     4.599    105.8      4     92 <2e-16
## Roy               1     4.599    105.8      4     92 <2e-16
##
## ------------------------------------------
##
```

```
## Term: Tx_P:Tx_C
##
## Sum of squares and products for the hypothesis:
##          P_Verbal P_Quant C_Verbal C_Quant
## P_Verbal    17267    7688     8362    8640
## P_Quant      7688    3423     3723    3847
## C_Verbal     8362    3723     4049    4184
## C_Quant      8640    3847     4184    4324
##
## Multivariate Tests: Tx_P:Tx_C
##                  Df test stat approx F num Df den Df Pr(>F)
## Pillai            1    0.7162    58.03      4     92 <2e-16
## Wilks             1    0.2838    58.03      4     92 <2e-16
## Hotelling-Lawley  1    2.5230    58.03      4     92 <2e-16
## Roy               1    2.5230    58.03      4     92 <2e-16
```

This method, however, does produce the univariate repeated measures F tests. They are based on separate error terms for each within-subjects effect. Note that a test of sphericity is not given because all within-subjects effects are 1 degree of freedom.

```
Mode <- factor(rep(c("Paper", "Computer"), c(2, 2)), levels = c("Paper",
    "Computer"))
Domain <- factor(rep(c("Verbal", "Quant"), 2), levels = c("Verbal",
    "Quant"))
idata <- data.frame(Mode, Domain)

LM_6 <- lm(cbind(P_Verbal, P_Quant, C_Verbal, C_Quant) ~ Tx_P * Tx_C,
    data = Skills_Trimmed)
LM_6

##
## Call:
## lm(formula = cbind(P_Verbal, P_Quant, C_Verbal, C_Quant) ~ Tx_P *
##     Tx_C, data = Skills_Trimmed)
##
## Coefficients:
##              P_Verbal  P_Quant  C_Verbal  C_Quant
## (Intercept)   56.94     58.84    57.68     62.29
## Tx_P1        -20.93    -18.69    -8.14     -8.84
## Tx_C1         -1.37      1.49   -10.22    -13.77
## Tx_P1:Tx_C1   13.21      5.88     6.40      6.61

ANOVA_1 <- Anova(LM_6, idata = idata, idesign = ~Mode * Domain, type = 2)
ANOVA_2 <- Anova(LM_6, idata = idata, idesign = ~Mode * Domain, type = 3)
summary(ANOVA_1, multivariate = FALSE)

##
## Univariate Type II Repeated-Measures ANOVA Assuming Sphericity
##
##                        Sum Sq num Df Error SS den Df F value
## (Intercept)           1375522      1    26463     95 4937.99
## Tx_P                    80207      1    26463     95  287.94
## Tx_C                    13712      1    26463     95   49.22
## Tx_P:Tx_C               25488      1    26463     95   91.50
## Mode                      522      1     5005     95    9.90
```

```
## Tx_P:Mode               12765       1     5005      95  242.29
## Tx_C:Mode               14463       1     5005      95  274.53
## Tx_P:Tx_C:Mode            916       1     5005      95   17.38
## Domain                   1031       1     1541      95   63.55
## Tx_P:Domain                64       1     1541      95    3.97
## Tx_C:Domain                15       1     1541      95    0.92
## Tx_P:Tx_C:Domain         1252       1     1541      95   77.17
## Mode:Domain               197       1     1743      95   10.71
## Tx_P:Mode:Domain          225       1     1743      95   12.25
## Tx_C:Mode:Domain          991       1     1743      95   54.02
## Tx_P:Tx_C:Mode:Domain    1407       1     1743      95   76.66
##                          Pr(>F)
## (Intercept)             < 2e-16
## Tx_P                    < 2e-16
## Tx_C                    3.3e-10
## Tx_P:Tx_C               1.4e-15
## Mode                    0.00221
## Tx_P:Mode               < 2e-16
## Tx_C:Mode               < 2e-16
## Tx_P:Tx_C:Mode          6.8e-05
## Domain                  3.5e-12
## Tx_P:Domain             0.04927
## Tx_C:Domain             0.33939
## Tx_P:Tx_C:Domain        6.5e-14
## Mode:Domain             0.00149
## Tx_P:Mode:Domain        0.00071
## Tx_C:Mode:Domain        6.9e-11
## Tx_P:Tx_C:Mode:Domain 7.5e-14

summary(ANOVA_2, multivariate = FALSE)

##
## Univariate Type III Repeated-Measures ANOVA Assuming Sphericity
##
##                          Sum Sq num Df Error SS den Df F value
## (Intercept)             1375167      1    26463     95 4936.71
## Tx_P                      79269      1    26463     95  284.57
## Tx_C                      14098      1    26463     95   50.61
## Tx_P:Tx_C                 25488      1    26463     95   91.50
## Mode                        432      1     5005     95    8.20
## Tx_P:Mode                 12693      1     5005     95  240.93
## Tx_C:Mode                 14386      1     5005     95  273.07
## Tx_P:Tx_C:Mode              916      1     5005     95   17.38
## Domain                     1047      1     1541     95   64.50
## Tx_P:Domain                  59      1     1541     95    3.61
## Tx_C:Domain                  12      1     1541     95    0.76
## Tx_P:Tx_C:Domain           1252      1     1541     95   77.17
## Mode:Domain                 181      1     1743     95    9.88
## Tx_P:Mode:Domain            213      1     1743     95   11.63
## Tx_C:Mode:Domain           1016      1     1743     95   55.35
## Tx_P:Tx_C:Mode:Domain      1407      1     1743     95   76.66
##                          Pr(>F)
## (Intercept)             < 2e-16
## Tx_P                    < 2e-16
```

```
## Tx_C                   2.1e-10
## Tx_P:Tx_C              1.4e-15
## Mode                   0.00514
## Tx_P:Mode              < 2e-16
## Tx_C:Mode              < 2e-16
## Tx_P:Tx_C:Mode         6.8e-05
## Domain                 2.6e-12
## Tx_P:Domain            0.06032
## Tx_C:Domain            0.38672
## Tx_P:Tx_C:Domain       6.5e-14
## Mode:Domain            0.00223
## Tx_P:Mode:Domain       0.00095
## Tx_C:Mode:Domain       4.5e-11
## Tx_P:Tx_C:Mode:Domain 7.5e-14
```

In this version, the factorial structure on the within-subjects side is ignored. Now the 3 degrees of freedom for the within-subjects effect require the sphericity assumption and that test is provided.

```
Measure <- factor(c("P_V", "P_Q", "C_V", "C_Q"), levels = c("P_V",
    "P_Q", "C_V", "C_Q"))
idata <- data.frame(Measure)

LM_7 <- lm(cbind(P_Verbal, P_Quant, C_Verbal, C_Quant) ~ Tx_P * Tx_C,
    data = Skills_Trimmed)
LM_7

##
## Call:
## lm(formula = cbind(P_Verbal, P_Quant, C_Verbal, C_Quant) ~ Tx_P *
##     Tx_C, data = Skills_Trimmed)
##
## Coefficients:
##              P_Verbal  P_Quant  C_Verbal  C_Quant
## (Intercept)   56.94     58.84    57.68     62.29
## Tx_P1        -20.93    -18.69    -8.14     -8.84
## Tx_C1         -1.37      1.49   -10.22    -13.77
## Tx_P1:Tx_C1   13.21      5.88     6.40      6.61

ANOVA_3 <- Anova(LM_7, idata = idata, idesign = ~Measure, type = 2)
ANOVA_4 <- Anova(LM_7, idata = idata, idesign = ~Measure, type = 3)
summary(ANOVA_3, multivariate = FALSE)

##
## Univariate Type II Repeated-Measures ANOVA Assuming Sphericity
##
##                     Sum Sq num Df Error SS den Df F value  Pr(>F)
## (Intercept)        1375522      1    26463     95  4938.0 < 2e-16
## Tx_P                 80207      1    26463     95   287.9 < 2e-16
## Tx_C                 13712      1    26463     95    49.2 3.3e-10
## Tx_P:Tx_C            25488      1    26463     95    91.5 1.4e-15
## Measure               1749      3     8290    285    20.1 8.1e-12
## Tx_P:Measure         13054      3     8290    285   149.6 < 2e-16
## Tx_C:Measure         15469      3     8290    285   177.3 < 2e-16
## Tx_P:Tx_C:Measure     3575      3     8290    285    41.0 < 2e-16
```

```
##
##
## Mauchly Tests for Sphericity
##
##                    Test statistic  p-value
## Measure                     0.577 6.59e-10
## Tx_P:Measure                0.577 6.59e-10
## Tx_C:Measure                0.577 6.59e-10
## Tx_P:Tx_C:Measure           0.577 6.59e-10
##
##
## Greenhouse-Geisser and Huynh-Feldt Corrections
##  for Departure from Sphericity
##
##                    GG eps Pr(>F[GG])
## Measure             0.717    4.2e-09
## Tx_P:Measure        0.717    < 2e-16
## Tx_C:Measure        0.717    < 2e-16
## Tx_P:Tx_C:Measure   0.717    < 2e-16
##
##                    HF eps Pr(>F[HF])
## Measure            0.7339  2.853e-09
## Tx_P:Measure       0.7339  1.779e-43
## Tx_C:Measure       0.7339  2.420e-48
## Tx_P:Tx_C:Measure 0.7339  7.766e-17
```

```
summary(ANOVA_4, multivariate = FALSE)
```

```
##
## Univariate Type III Repeated-Measures ANOVA Assuming Sphericity
##
##                     Sum Sq num Df Error SS den Df F value  Pr(>F)
## (Intercept)        1375167      1    26463     95  4936.7 < 2e-16
## Tx_P                 79269      1    26463     95   284.6 < 2e-16
## Tx_C                 14098      1    26463     95    50.6 2.1e-10
## Tx_P:Tx_C            25488      1    26463     95    91.5 1.4e-15
## Measure               1660      3     8290    285    19.0 2.8e-11
## Tx_P:Measure         12965      3     8290    285   148.6 < 2e-16
## Tx_C:Measure         15414      3     8290    285   176.7 < 2e-16
## Tx_P:Tx_C:Measure     3575      3     8290    285    41.0 < 2e-16
##
##
## Mauchly Tests for Sphericity
##
##                    Test statistic  p-value
## Measure                     0.577 6.59e-10
## Tx_P:Measure                0.577 6.59e-10
## Tx_C:Measure                0.577 6.59e-10
## Tx_P:Tx_C:Measure           0.577 6.59e-10
##
##
## Greenhouse-Geisser and Huynh-Feldt Corrections
##  for Departure from Sphericity
##
```

```
##                    GG eps Pr(>F[GG])
## Measure             0.717      1e-08
## Tx_P:Measure        0.717     <2e-16
## Tx_C:Measure        0.717     <2e-16
## Tx_P:Tx_C:Measure   0.717     <2e-16
##
##                    HF eps Pr(>F[HF])
## Measure            0.7339  7.222e-09
## Tx_P:Measure       0.7339  2.753e-43
## Tx_C:Measure       0.7339  3.081e-48
## Tx_P:Tx_C:Measure 0.7339  7.766e-17
```

8 Means, Standard Errors, Confidence Intervals, and Comparisons

The emmeans package provides considerable flexibility for obtaining marginal means, standard errors, confidence intervals, and comparisons. It is illustrated here for the full 2 x 2 x 2 x 2 design. A reference grid is defined for use in the follow-up functions from the emmeans package. Note that the emmeans package works with the aov() function, but not the ezANOVA() function.

```
model_rg <- ref_grid(AOV_17)
model_rg

## 'emmGrid' object with variables:
##     Mode = Computer, Paper
##     Domain = Quant, Verbal
##     Tx_P = 1, 2
##     Tx_C = 1, 2

summary(model_rg)

##  Mode     Domain Tx_P Tx_C prediction    SE    df
##  Computer Quant  1    1         46.28 1.916 156.7
##  Paper    Quant  1    1         47.51 1.916 156.7
##  Computer Verbal 1    1         45.72 1.916 156.7
##  Paper    Verbal 1    1         47.85 1.916 156.7
##  Computer Quant  2    1         50.74 1.941 158.0
##  Paper    Quant  2    1         73.14 1.941 158.0
##  Computer Verbal 2    1         49.20 1.941 158.0
##  Paper    Verbal 2    1         63.30 1.941 158.0
##  Computer Quant  1    2         60.61 1.916 156.7
##  Paper    Quant  1    2         32.78 1.916 156.7
##  Computer Verbal 1    2         53.36 1.916 156.7
##  Paper    Verbal 1    2         24.17 1.916 156.7
##  Computer Quant  2    2         91.51 1.916 156.7
##  Paper    Quant  2    2         81.93 1.916 156.7
##  Computer Verbal 2    2         82.43 1.916 156.7
##  Paper    Verbal 2    2         92.45 1.916 156.7
```

The function, emmeans( ), when used with the reference grid and specifications for particular effects, provides marginal and cell means, standard errors, and confidence limits.

```r
Tx_P_emm <- emmeans(model_rg, "Tx_P")

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

Tx_C_emm <- emmeans(model_rg, "Tx_C")

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

Mode_emm <- emmeans(model_rg, "Mode")

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

Domain_emm <- emmeans(model_rg, "Domain")

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

Tx_P_x_Tx_C_emm <- emmeans(model_rg, c("Tx_P", "Tx_C"))

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

Tx_P_x_Mode_emm <- emmeans(model_rg, c("Tx_P", "Mode"))

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

Tx_P_x_Domain_emm <- emmeans(model_rg, c("Tx_P", "Domain"))

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

Tx_C_x_Mode_emm <- emmeans(model_rg, c("Tx_C", "Mode"))

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

Tx_C_x_Domain_emm <- emmeans(model_rg, c("Tx_C", "Domain"))

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

Mode_x_Domain_emm <- emmeans(model_rg, c("Mode", "Domain"))

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

Tx_P_x_Tx_C_x_Mode_emm <- emmeans(model_rg, c("Tx_P", "Tx_C", "Mode"))

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

Tx_P_x_Tx_C_x_Domain_emm <- emmeans(model_rg, c("Tx_P", "Tx_C", "Domain"))

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

Tx_P_x_Mode_x_Domain_emm <- emmeans(model_rg, c("Tx_P", "Mode", "Domain"))

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

Tx_C_x_Mode_x_Domain_emm <- emmeans(model_rg, c("Tx_C", "Mode", "Domain"))

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

cell_means_emm <- emmeans(model_rg, c("Tx_P", "Tx_C", "Mode", "Domain"))

Tx_P_emm

##  Tx_P emmean    SE df lower.CL upper.CL
##  1     44.79 1.186 95    42.43    47.14
##  2     73.09 1.186 95    70.73    75.44
##
## Results are averaged over the levels of: Mode, Domain, Tx_C
## Confidence level used: 0.95
```

```
Tx_C_emm

##  Tx_C emmean    SE df lower.CL upper.CL
##  1     52.97 1.186 95    50.61    55.32
##  2     64.90 1.186 95    62.55    67.26
##
## Results are averaged over the levels of: Mode, Domain, Tx_P
## Confidence level used: 0.95

Mode_emm

##  Mode     emmean     SE    df lower.CL upper.CL
##  Computer  59.98 0.9146 129.7    58.17    61.79
##  Paper     57.89 0.9146 129.7    56.08    59.70
##
## Results are averaged over the levels of: Domain, Tx_P, Tx_C
## Confidence level used: 0.95

Domain_emm

##  Domain emmean     SE  df lower.CL upper.CL
##  Quant   60.56 0.8628 106    58.85    62.27
##  Verbal  57.31 0.8628 106    55.60    59.02
##
## Results are averaged over the levels of: Mode, Tx_P, Tx_C
## Confidence level used: 0.95

Tx_P_x_Tx_C_emm

##  Tx_P Tx_C emmean    SE df lower.CL upper.CL
##  1    1     46.84 1.673 95    43.52    50.16
##  2    1     59.10 1.691 95    55.74    62.45
##  1    2     42.73 1.673 95    39.41    46.05
##  2    2     87.08 1.673 95    83.76    90.40
##
## Results are averaged over the levels of: Mode, Domain
## Confidence level used: 0.95

Tx_P_x_Mode_emm

##  Tx_P Mode     emmean    SE    df lower.CL upper.CL
##  1    Computer  51.49 1.292 129.4    48.94    54.05
##  2    Computer  68.47 1.295 130.0    65.91    71.03
##  1    Paper     38.08 1.292 129.4    35.52    40.64
##  2    Paper     77.70 1.295 130.0    75.14    80.27
##
## Results are averaged over the levels of: Domain, Tx_C
## Confidence level used: 0.95

Tx_P_x_Domain_emm

##  Tx_P Domain emmean    SE    df lower.CL upper.CL
##  1    Quant   46.80 1.220 105.9    44.38    49.22
##  2    Quant   74.33 1.221 106.1    71.91    76.75
##  1    Verbal  42.78 1.220 105.9    40.36    45.19
##  2    Verbal  71.85 1.221 106.1    69.43    74.27
##
## Results are averaged over the levels of: Mode, Tx_C
## Confidence level used: 0.95
```

```
Tx_C_x_Mode_emm

##  Tx_C Mode     emmean    SE    df lower.CL upper.CL
##  1    Computer  47.99 1.295 130.0    45.42    50.55
##  2    Computer  71.98 1.292 129.4    69.42    74.53
##  1    Paper     57.95 1.295 130.0    55.39    60.51
##  2    Paper     57.83 1.292 129.4    55.27    60.39
##
## Results are averaged over the levels of: Domain, Tx_P
## Confidence level used: 0.95

Tx_C_x_Domain_emm

##  Tx_C Domain emmean    SE    df lower.CL upper.CL
##  1    Quant   54.42 1.221 106.1    52.00    56.84
##  2    Quant   66.71 1.220 105.9    64.29    69.13
##  1    Verbal  51.52 1.221 106.1    49.10    53.94
##  2    Verbal  63.10 1.220 105.9    60.68    65.52
##
## Results are averaged over the levels of: Mode, Tx_P
## Confidence level used: 0.95

Mode_x_Domain_emm

##  Mode     Domain emmean     SE  df lower.CL upper.CL
##  Computer Quant   62.28 0.9612 157    60.39    64.18
##  Paper    Quant   58.84 0.9612 157    56.94    60.74
##  Computer Verbal  57.68 0.9612 157    55.78    59.58
##  Paper    Verbal  56.94 0.9612 157    55.04    58.84
##
## Results are averaged over the levels of: Tx_P, Tx_C
## Confidence level used: 0.95

Tx_P_x_Tx_C_x_Mode_emm

##  Tx_P Tx_C Mode     emmean    SE    df lower.CL upper.CL
##  1    1    Computer  46.00 1.824 129.5    42.39    49.61
##  2    1    Computer  49.97 1.846 130.2    46.32    53.62
##  1    2    Computer  56.99 1.824 129.5    53.38    60.59
##  2    2    Computer  86.97 1.824 129.5    83.36    90.58
##  1    1    Paper     47.68 1.824 129.5    44.08    51.29
##  2    1    Paper     68.22 1.846 130.2    64.57    71.87
##  1    2    Paper     28.47 1.824 129.5    24.86    32.08
##  2    2    Paper     87.19 1.824 129.5    83.58    90.80
##
## Results are averaged over the levels of: Domain
## Confidence level used: 0.95

Tx_P_x_Tx_C_x_Domain_emm

##  Tx_P Tx_C Domain emmean    SE    df lower.CL upper.CL
##  1    1    Quant   46.90 1.721 106.0    43.49    50.31
##  2    1    Quant   61.94 1.740 106.2    58.49    65.39
##  1    2    Quant   46.70 1.721 106.0    43.28    50.11
##  2    2    Quant   86.72 1.721 106.0    83.31    90.13
##  1    1    Verbal  46.79 1.721 106.0    43.37    50.20
```

```
## 2    1    Verbal  56.25 1.740 106.2    52.80    59.70
## 1    2    Verbal  38.76 1.721 106.0    35.35    42.18
## 2    2    Verbal  87.44 1.721 106.0    84.03    90.85
##
## Results are averaged over the levels of: Mode
## Confidence level used: 0.95
```

```
Tx_P_x_Mode_x_Domain_emm
```

```
##  Tx_P Mode     Domain emmean    SE    df lower.CL upper.CL
##  1    Computer Quant   53.45 1.358 156.4    50.76    56.13
##  2    Computer Quant   71.12 1.361 157.7    68.43    73.81
##  1    Paper    Quant   40.15 1.358 156.4    37.47    42.83
##  2    Paper    Quant   77.53 1.361 157.7    74.85    80.22
##  1    Computer Verbal  49.54 1.358 156.4    46.86    52.22
##  2    Computer Verbal  65.82 1.361 157.7    63.13    68.51
##  1    Paper    Verbal  36.01 1.358 156.4    33.33    38.69
##  2    Paper    Verbal  77.87 1.361 157.7    75.19    80.56
##
## Results are averaged over the levels of: Tx_C
## Confidence level used: 0.95
```

```
Tx_C_x_Mode_x_Domain_emm
```

```
##  Tx_C Mode     Domain emmean    SE    df lower.CL upper.CL
##  1    Computer Quant   48.51 1.361 157.7    45.82    51.20
##  2    Computer Quant   76.06 1.358 156.4    73.38    78.74
##  1    Paper    Quant   60.33 1.361 157.7    57.64    63.02
##  2    Paper    Quant   57.35 1.358 156.4    54.67    60.04
##  1    Computer Verbal  47.46 1.361 157.7    44.77    50.15
##  2    Computer Verbal  67.90 1.358 156.4    65.22    70.58
##  1    Paper    Verbal  55.58 1.361 157.7    52.89    58.27
##  2    Paper    Verbal  58.31 1.358 156.4    55.63    60.99
##
## Results are averaged over the levels of: Tx_P
## Confidence level used: 0.95
```

```
cell_means_emm
```

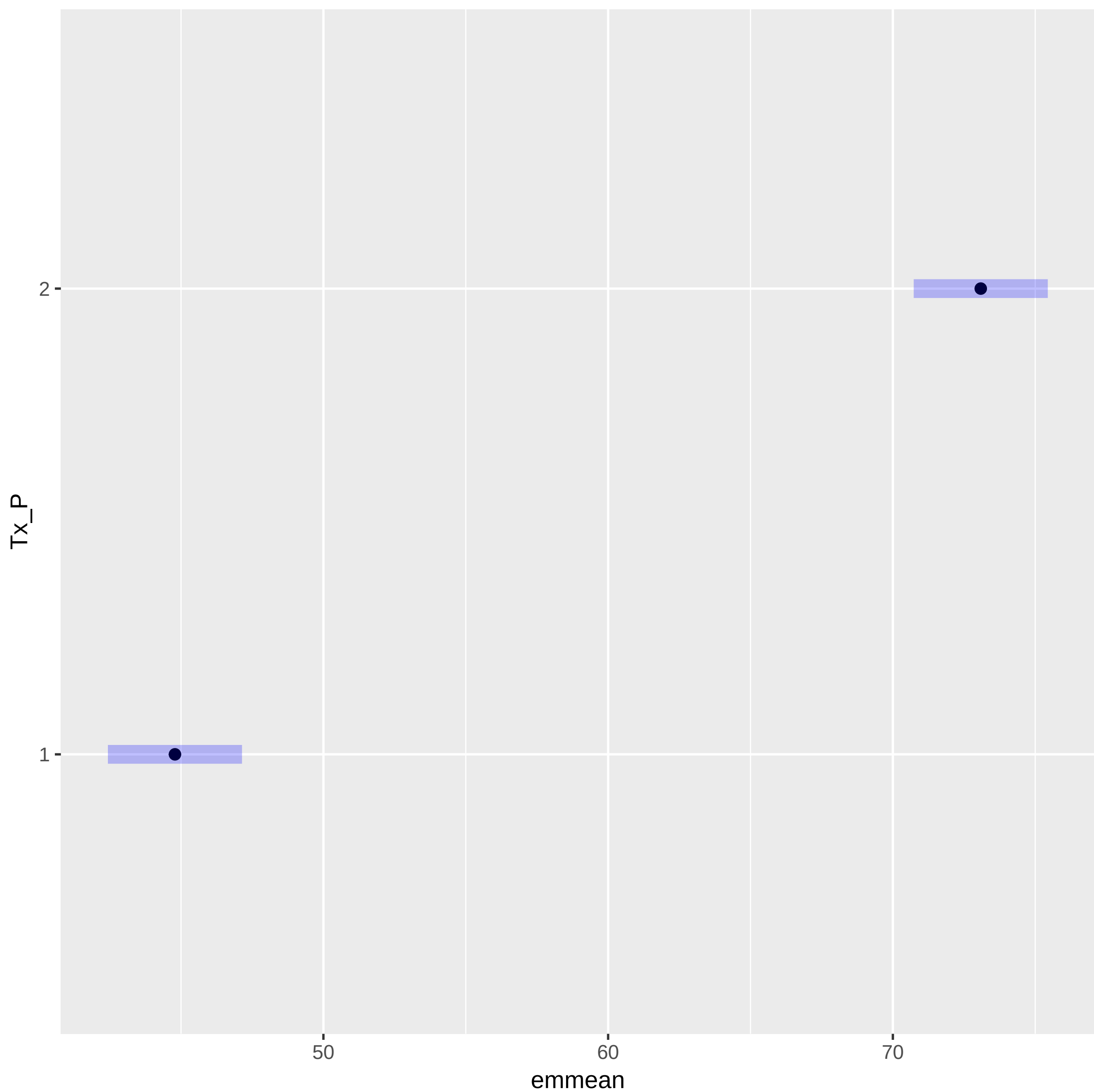
```
##  Tx_P Tx_C Mode     Domain emmean    SE    df lower.CL upper.CL
##  1    1    Computer Quant   46.28 1.916 156.7    42.50    50.07
##  2    1    Computer Quant   50.74 1.941 158.0    46.90    54.57
##  1    2    Computer Quant   60.61 1.916 156.7    56.83    64.40
##  2    2    Computer Quant   91.51 1.916 156.7    87.72    95.29
##  1    1    Paper    Quant   47.51 1.916 156.7    43.73    51.30
##  2    1    Paper    Quant   73.14 1.941 158.0    69.31    76.97
##  1    2    Paper    Quant   32.78 1.916 156.7    28.99    36.56
##  2    2    Paper    Quant   81.93 1.916 156.7    78.14    85.71
##  1    1    Computer Verbal  45.72 1.916 156.7    41.93    49.50
##  2    1    Computer Verbal  49.20 1.941 158.0    45.37    53.04
##  1    2    Computer Verbal  53.36 1.916 156.7    49.58    57.15
##  2    2    Computer Verbal  82.43 1.916 156.7    78.65    86.22
##  1    1    Paper    Verbal  47.85 1.916 156.7    44.07    51.64
##  2    1    Paper    Verbal  63.30 1.941 158.0    59.47    67.13
##  1    2    Paper    Verbal  24.17 1.916 156.7    20.38    27.95
```

```
##   2    2     Paper     Verbal  92.45 1.916 156.7     88.66     96.23
##
## Confidence level used: 0.95
```

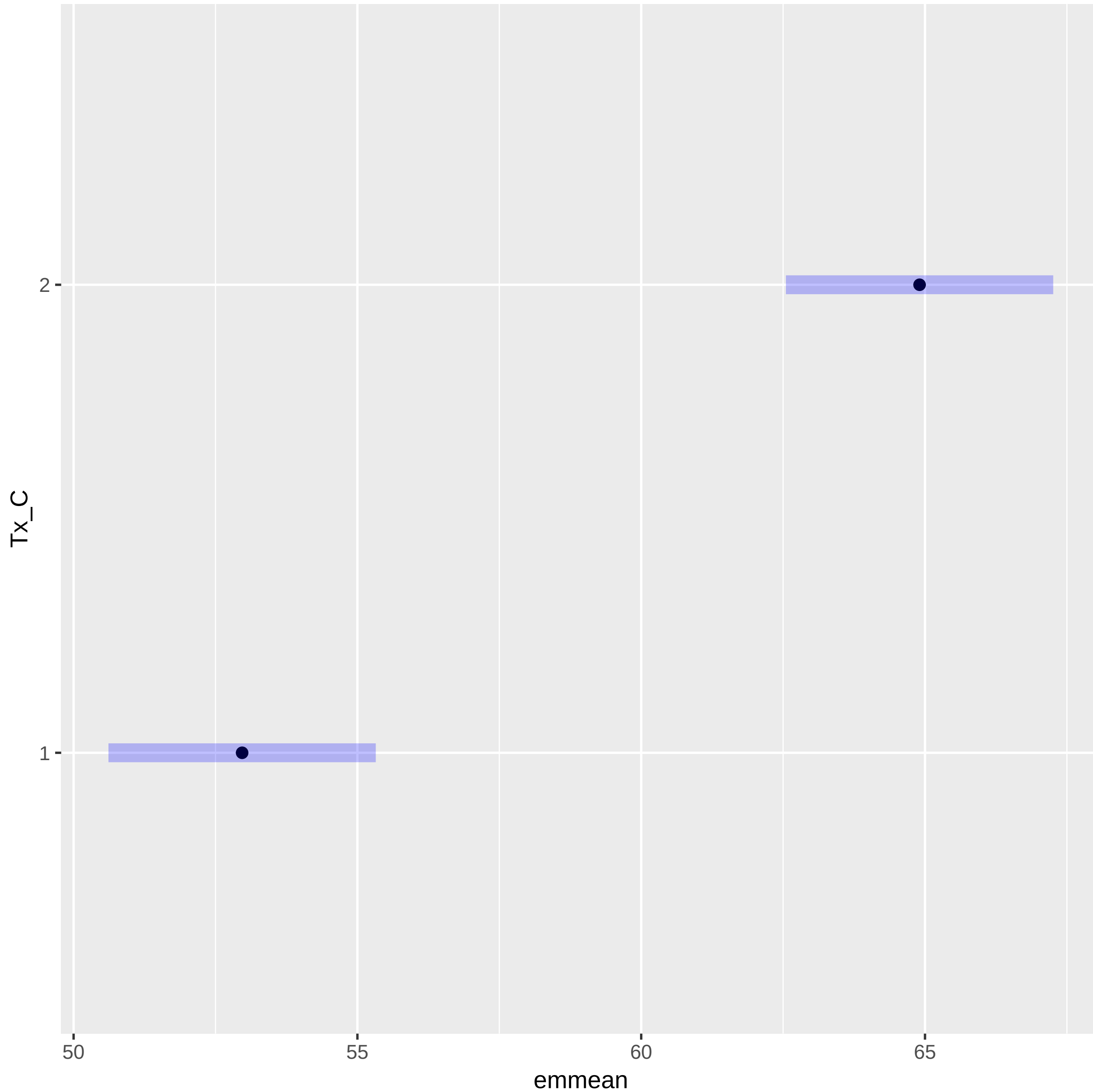
8.1 Plotting of Cell Means and Marginal Means

The means are plotted along with their confidence intervals. These plots are internal to the emmeans package. Nicer bar graphs can be created using ggplot2 with the information from the saved objects.

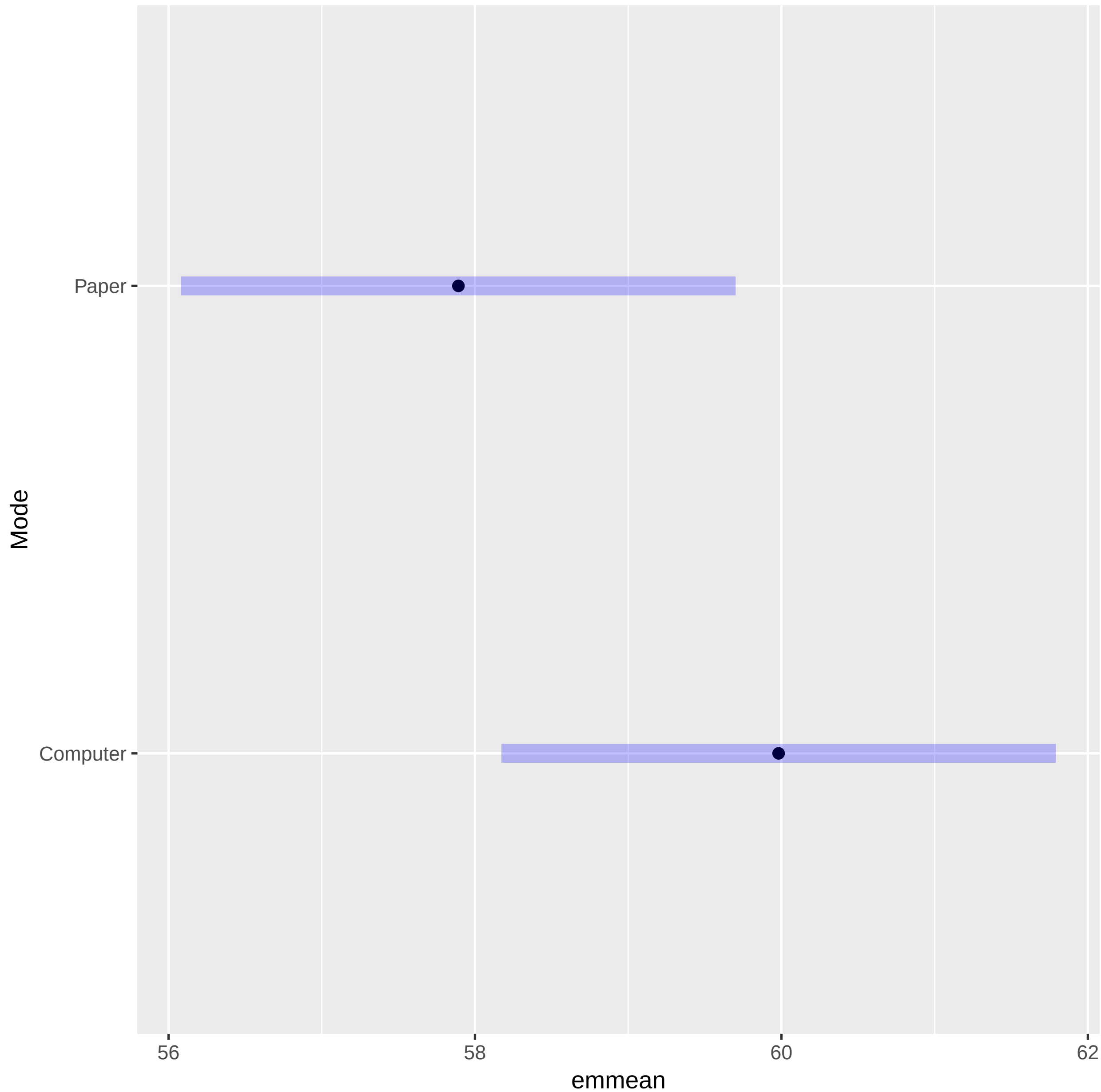
```
plot(Tx_P_emm, cex = 1.5, cex.axis = 2)
```



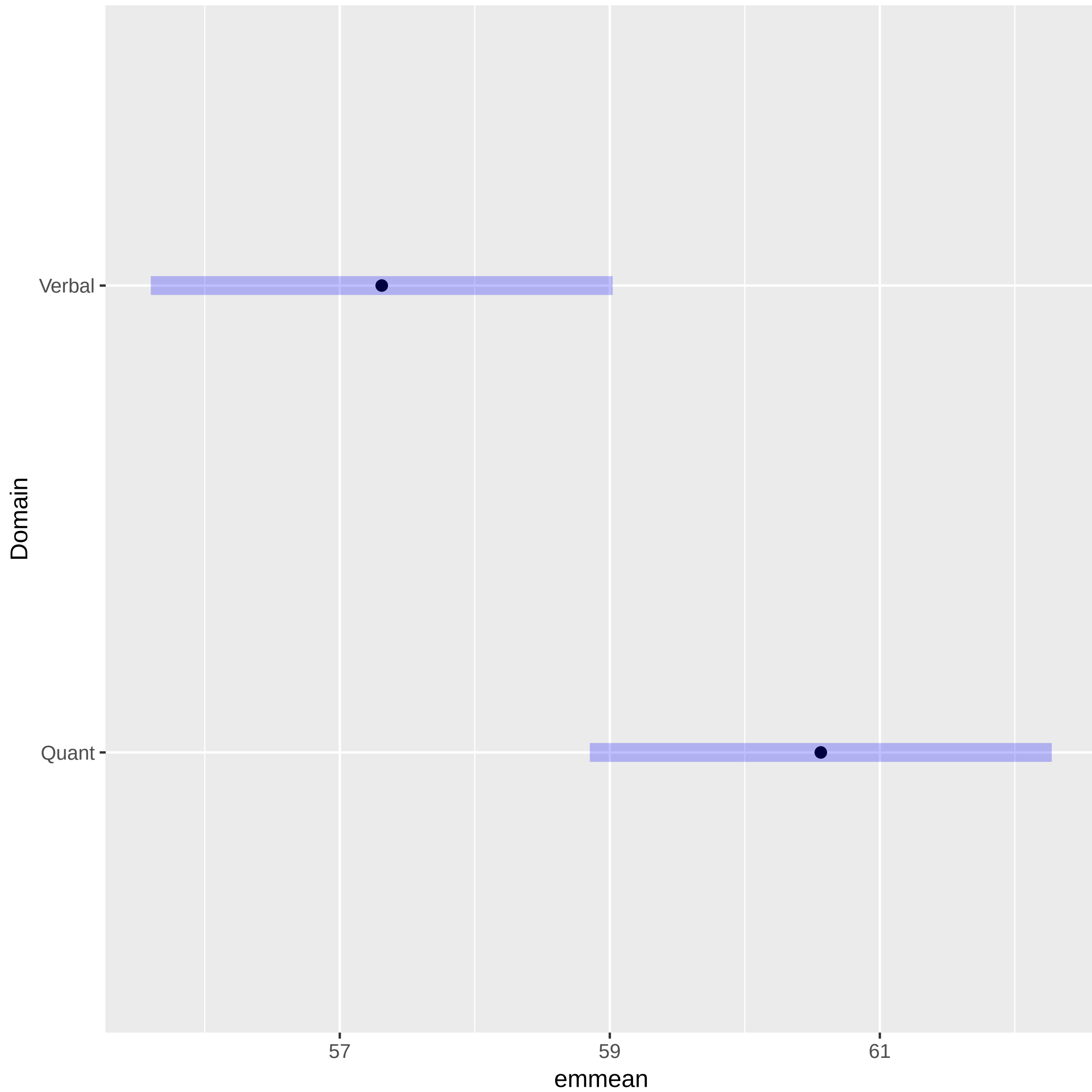
```r
plot(Tx_C_emm, cex = 1.5, cex.axis = 2)
```



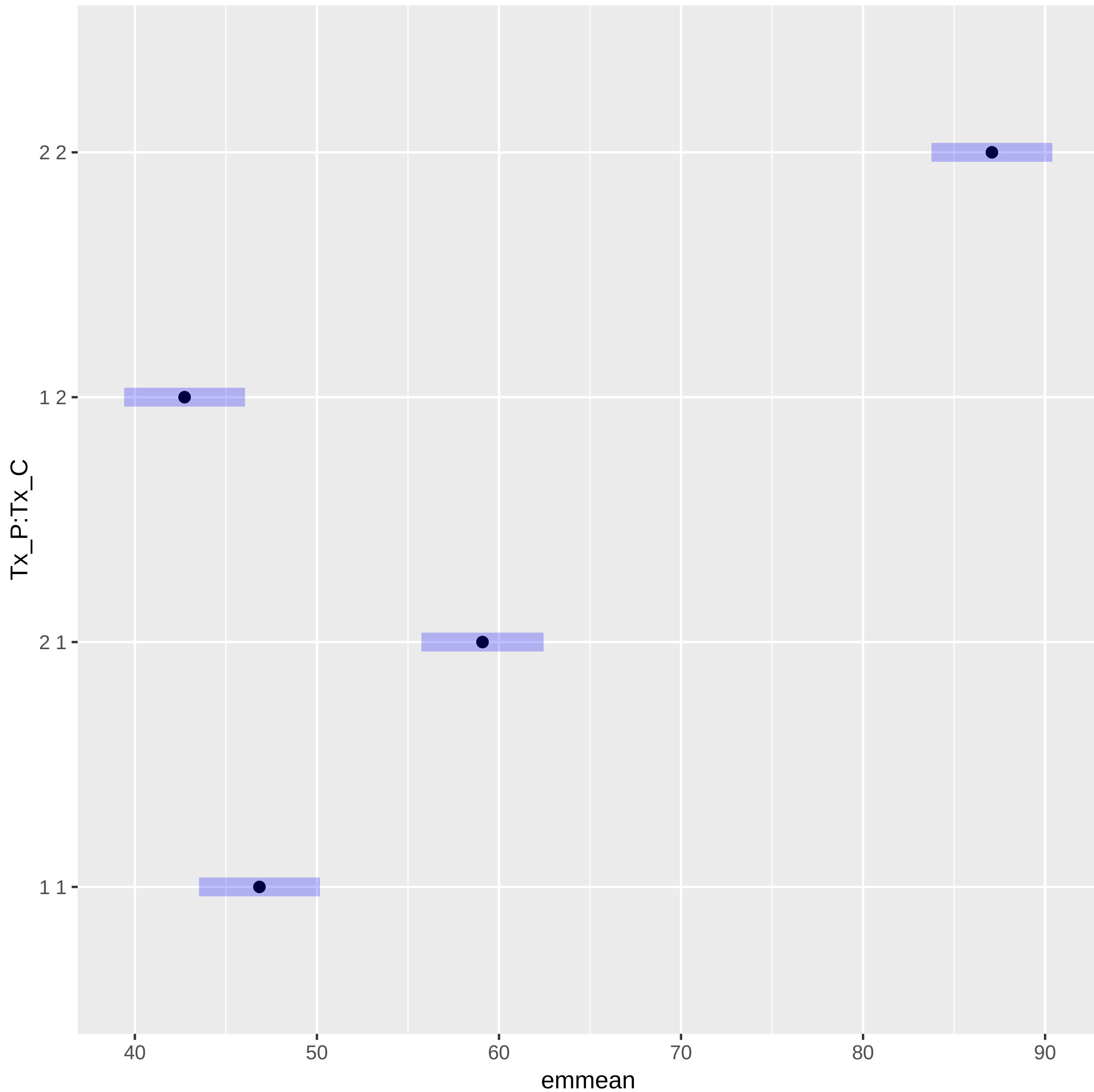

```r
plot(Mode_emm, cex = 1.5, cex.axis = 2)
```



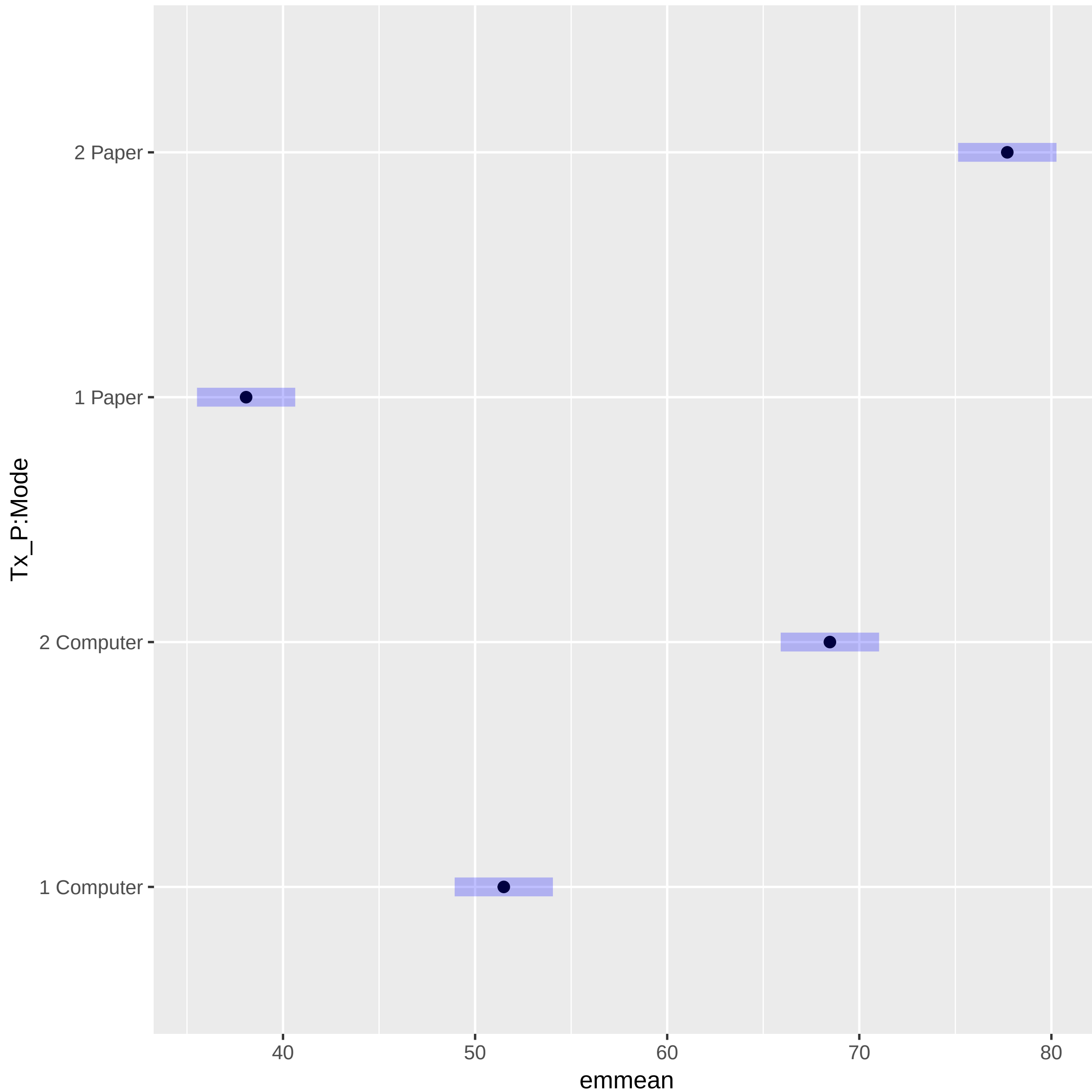

```
plot(Domain_emm, cex = 1.5, cex.axis = 2)
```



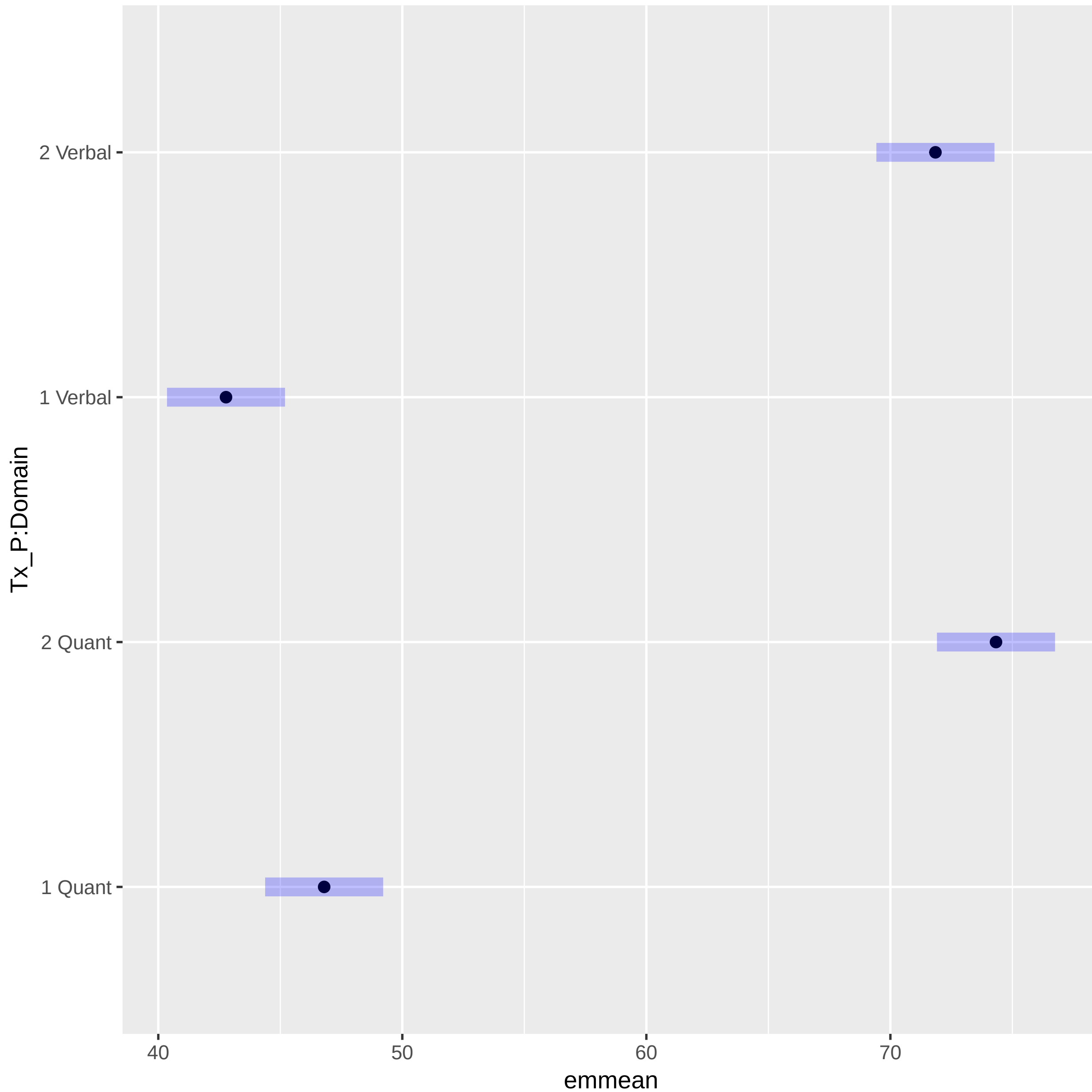

```
plot(Tx_P_x_Tx_C_emm, cex = 1.5, cex.axis = 2)
```



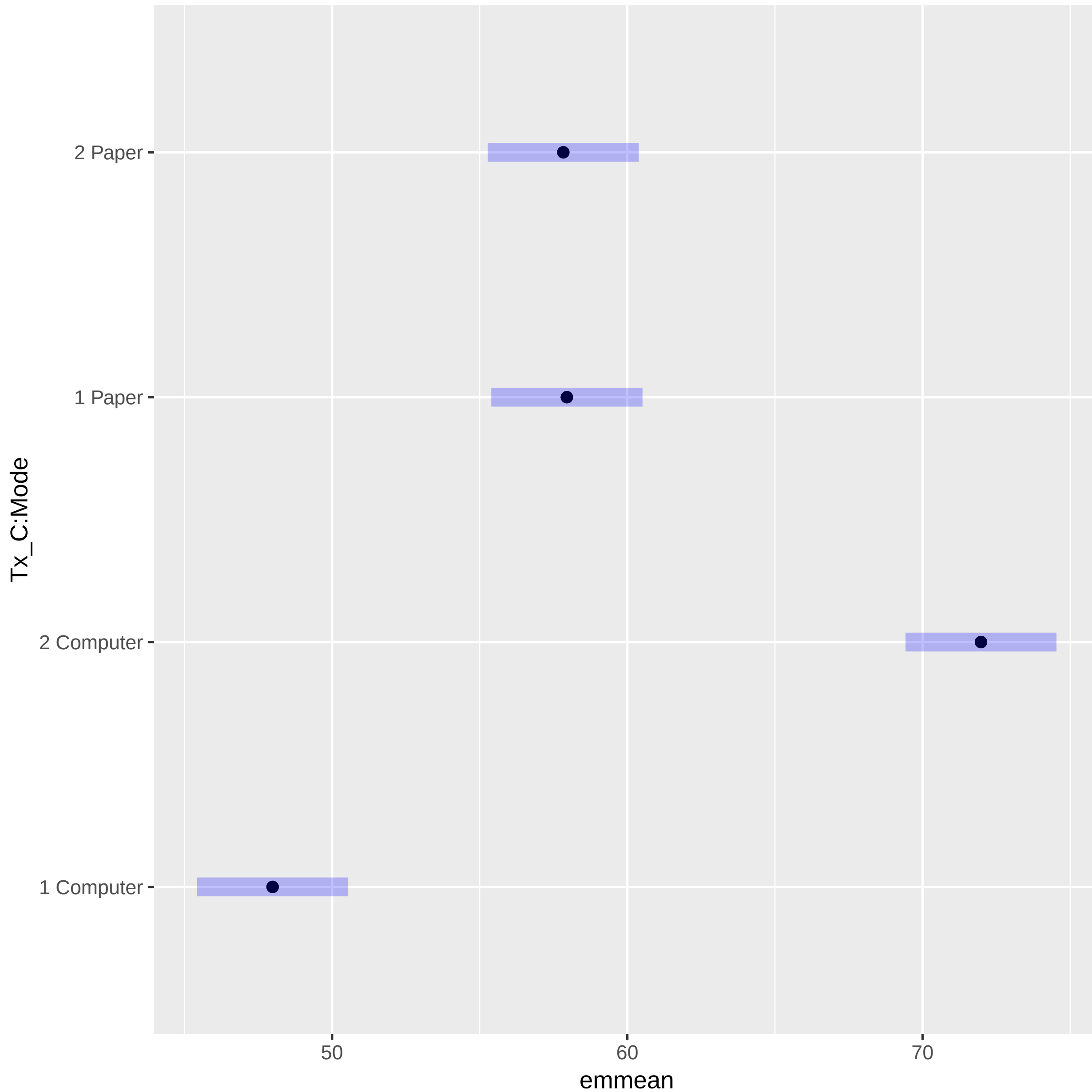

```
plot(Tx_P_x_Mode_emm, cex = 1.5, cex.axis = 2)
```



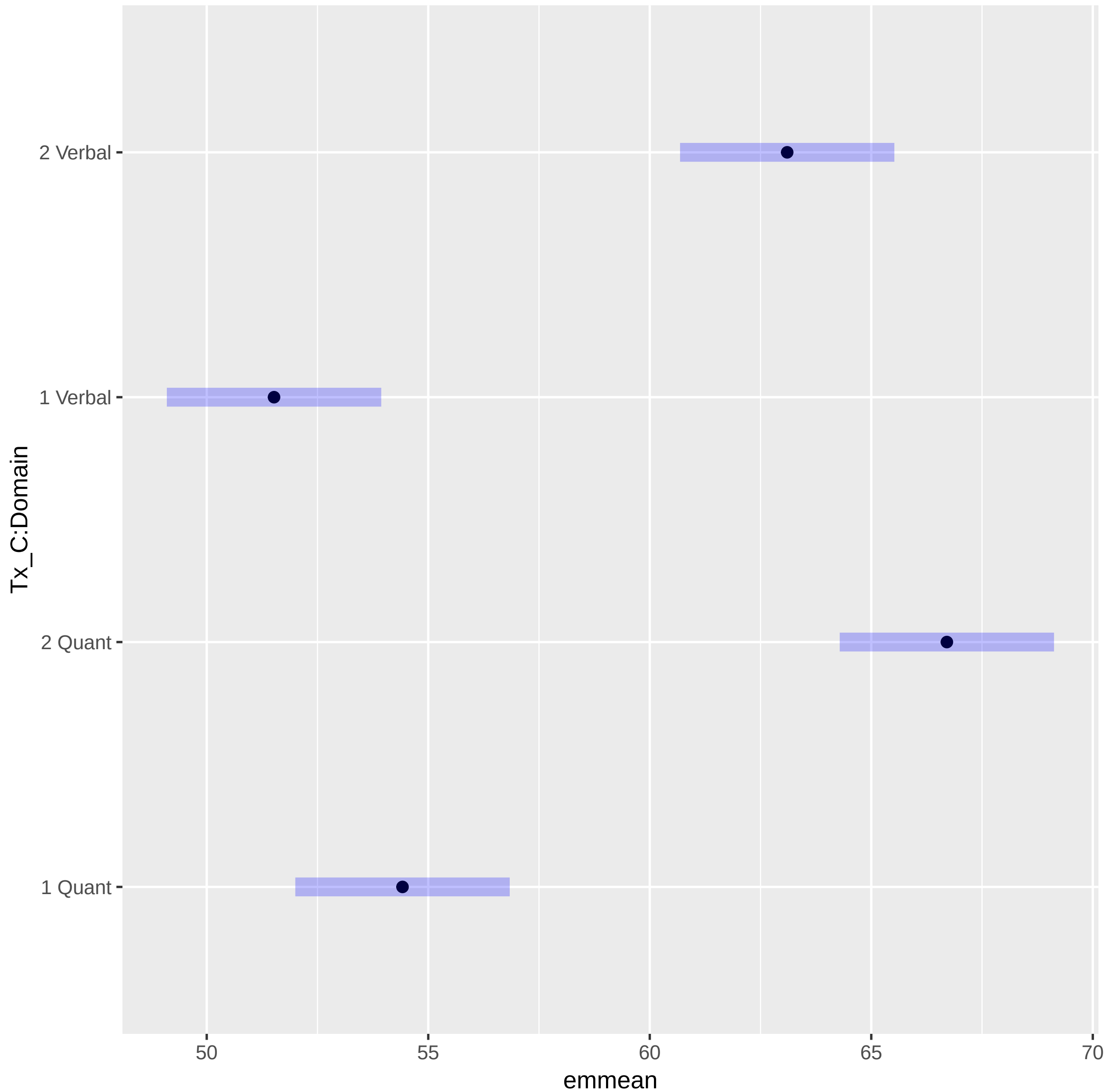

```r
plot(Tx_P_x_Domain_emm, cex = 1.5, cex.axis = 2)
```



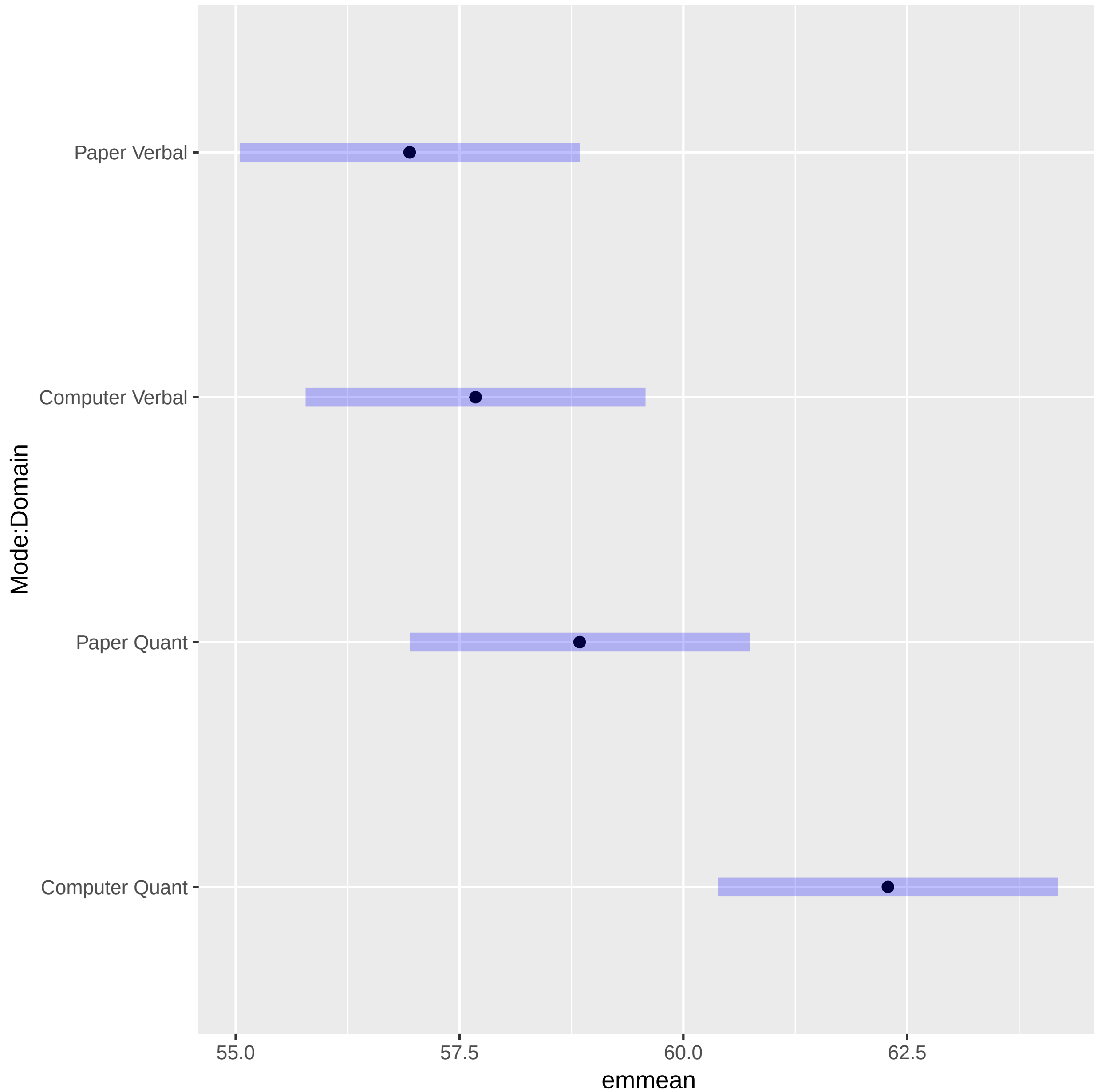

```
plot(Tx_C_x_Mode_emm, cex = 1.5, cex.axis = 2)
```



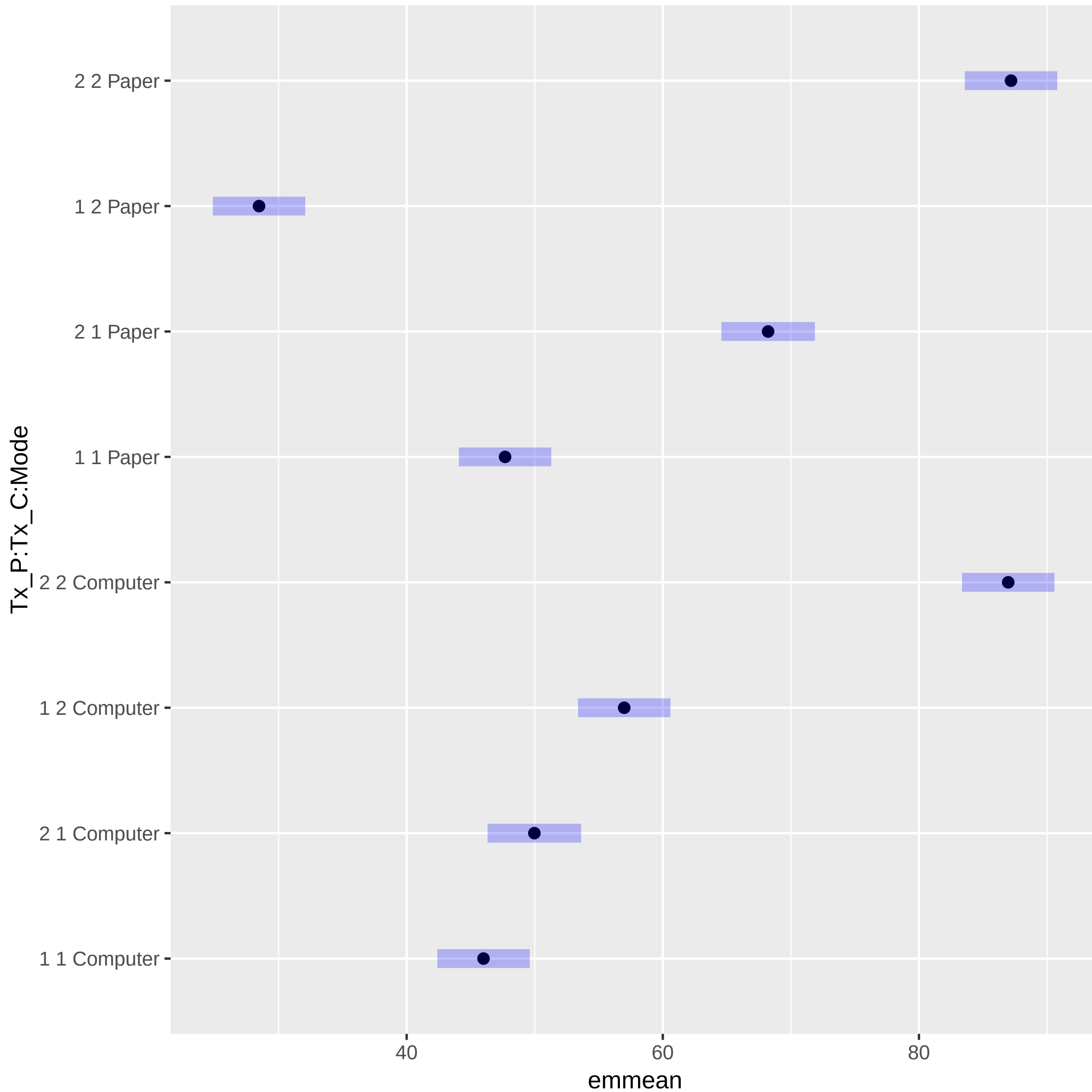

```r
plot(Tx_C_x_Domain_emm, cex = 1.5, cex.axis = 2)
```



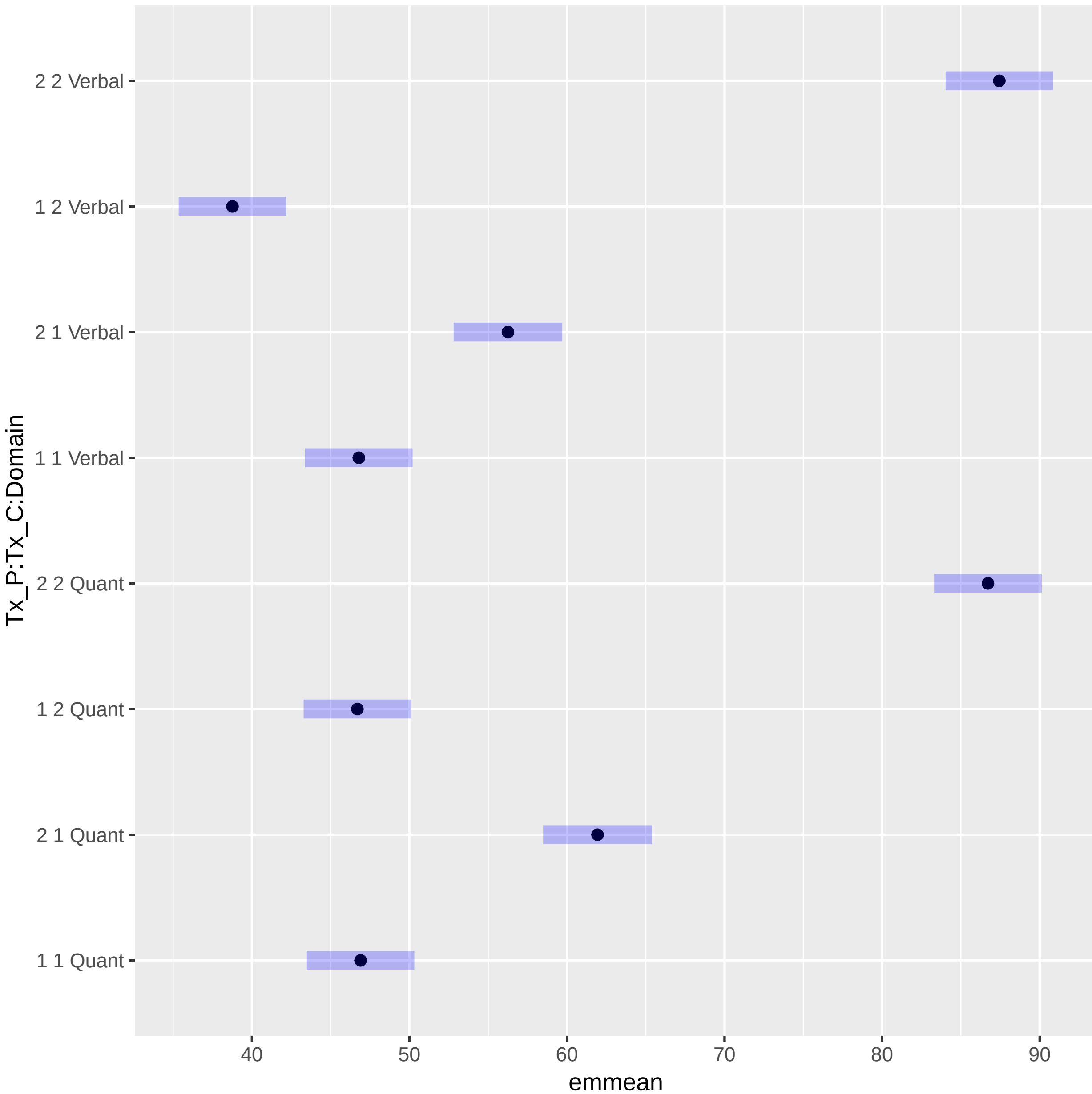

```
plot(Mode_x_Domain_emm, cex = 1.5, cex.axis = 2)
```



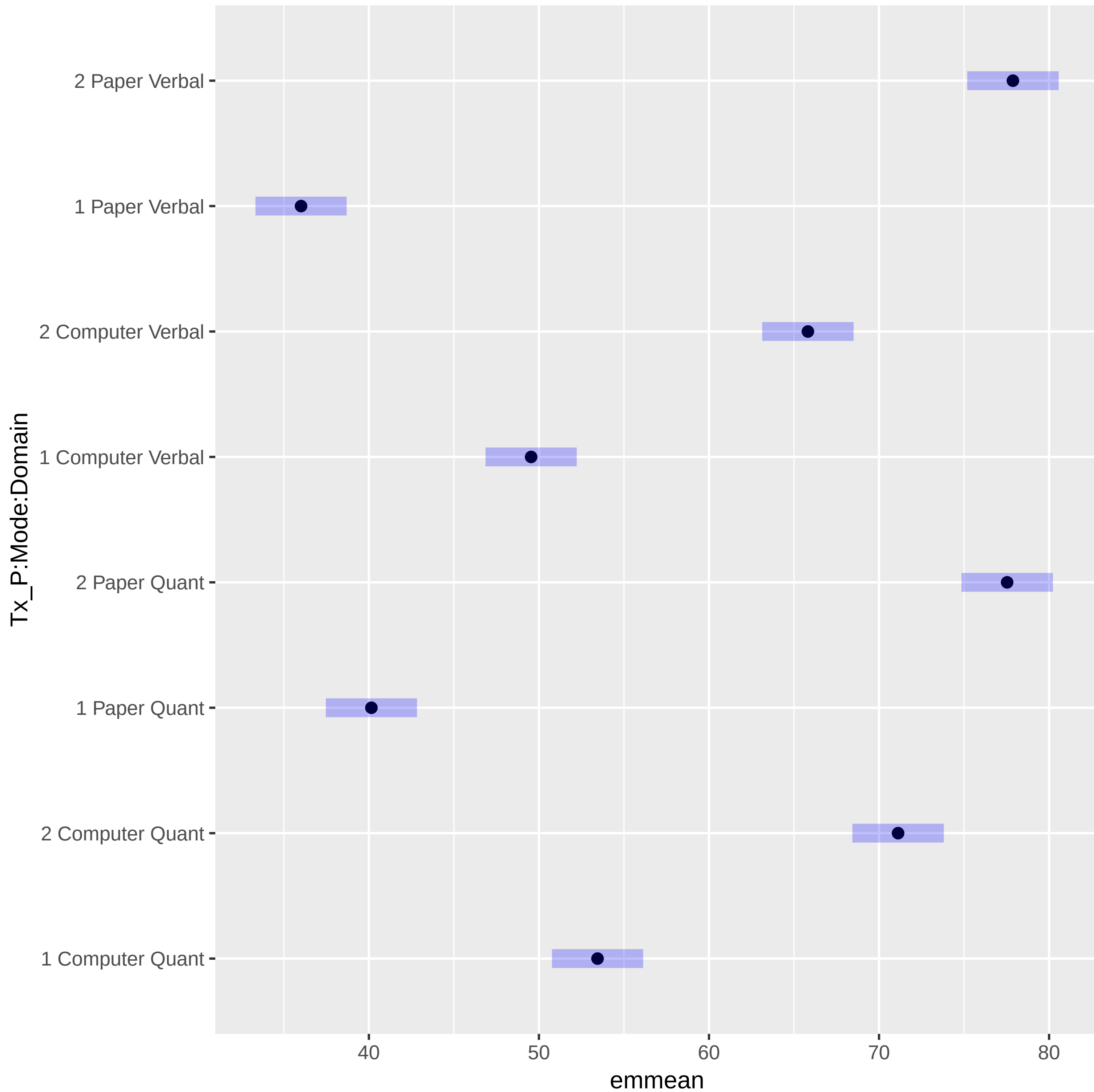

```
plot(Tx_P_x_Tx_C_x_Mode_emm, cex = 1.5, cex.axis = 2)
```



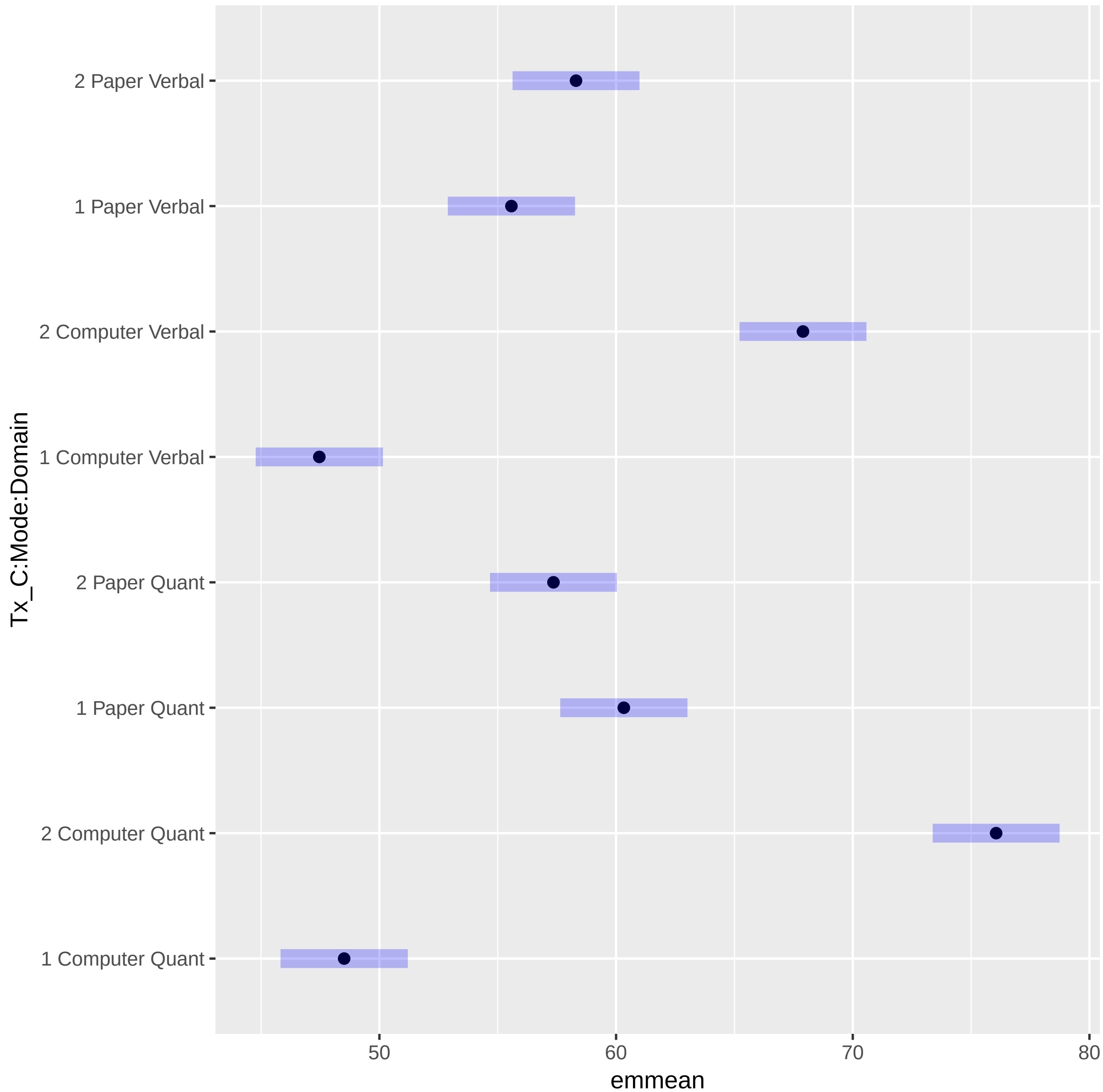

```
plot(Tx_P_x_Tx_C_x_Domain_emm, cex = 1.5, cex.axis = 2)
```



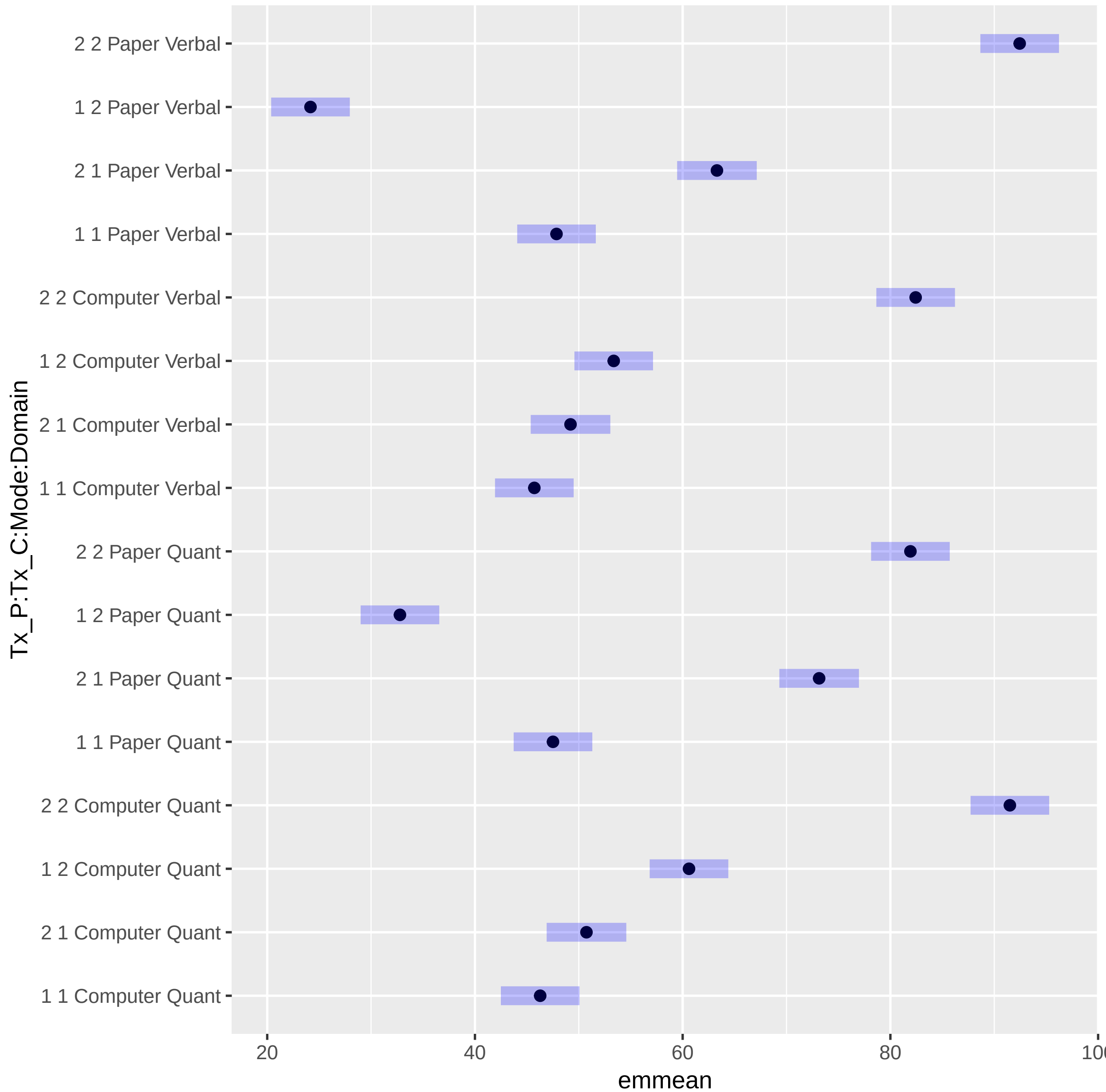

```r
plot(Tx_P_x_Mode_x_Domain_emm, cex = 1.5, cex.axis = 2)
```




```
plot(Tx_C_x_Mode_x_Domain_emm, cex = 1.5, cex.axis = 2)
```




```
plot(cell_means_emm, cex = 1.5, cex.axis = 2)
```



8.2 Comparisons Among Means

The CLD( ) function can be used to obtain pairwise comparisons. The "adjust" option allows specifying the particular form of Type I error control (Holm procedure is used here). The pairwise comparisons can be made at any level in the design. For interactions, the pairwise comparisons are conducted using a simple main effects approach. Pairs of means for one of the variables involved in the interaction are compared within each level of the other variable involved in the interaction (or each combination of levels if the interaction is three-way).

The output that has "group" as the right-most column provides a convenient summary display. Rows that do not share any numbers in the group column represent conditions that are significantly different.

There are many ways to do the comparisons, depending on the complexity of the effect. A few are illustrated below.

```
CLD(Tx_P_emm, alpha = 0.05, adjust = "holm", details = TRUE)

## $emmeans
##  Tx_P emmean    SE df lower.CL upper.CL .group
##  1     44.79 1.186 95    42.08    47.49  1
##  2     73.09 1.186 95    70.39    75.79   2
##
## Results are averaged over the levels of: Mode, Domain, Tx_C
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## significance level used: alpha = 0.05
##
## $comparisons
##  contrast estimate    SE df t.ratio p.value
##  2 - 1        28.3 1.678 95   16.87  <.0001
##
## Results are averaged over the levels of: Mode, Domain, Tx_C
```

```
CLD(Tx_C_emm, alpha = 0.05, adjust = "holm", details = TRUE)

## $emmeans
##  Tx_C emmean    SE df lower.CL upper.CL .group
##  1     52.97 1.186 95    50.27    55.67  1
##  2     64.90 1.186 95    62.20    67.61   2
##
## Results are averaged over the levels of: Mode, Domain, Tx_P
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## significance level used: alpha = 0.05
##
## $comparisons
##  contrast estimate    SE df t.ratio p.value
##  2 - 1       11.94 1.678 95   7.114  <.0001
##
## Results are averaged over the levels of: Mode, Domain, Tx_P
```

```
CLD(Mode_emm, alpha = 0.05, adjust = "holm", details = TRUE)

## $emmeans
##  Mode     emmean     SE    df lower.CL upper.CL .group
##  Paper     57.89 0.9146 129.7    55.82    59.97  1
##  Computer  59.98 0.9146 129.7    57.91    62.06   2
##
## Results are averaged over the levels of: Domain, Tx_P, Tx_C
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## significance level used: alpha = 0.05
##
## $comparisons
##  contrast         estimate     SE df t.ratio p.value
```

```
##  Computer - Paper     2.09 0.7296 95   2.864  0.0051
##
## Results are averaged over the levels of: Domain, Tx_P, Tx_C
```

```r
CLD(Domain_emm, alpha = 0.05, adjust = "holm", details = TRUE)

## $emmeans
##  Domain emmean     SE  df lower.CL upper.CL .group
##  Verbal  57.31 0.8628 106    55.35    59.27  1
##  Quant   60.56 0.8628 106    58.60    62.52   2
##
## Results are averaged over the levels of: Mode, Tx_P, Tx_C
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## significance level used: alpha = 0.05
##
## $comparisons
##  contrast       estimate     SE df t.ratio p.value
##  Quant - Verbal    3.252 0.4049 95   8.031  <.0001
##
## Results are averaged over the levels of: Mode, Tx_P, Tx_C
```

```r
CLD(Tx_P_x_Tx_C_emm, by = "Tx_P", alpha = 0.05, adjust = "holm", details = TRUE)

## $emmeans
## Tx_P = 1:
##  Tx_C emmean    SE df lower.CL upper.CL .group
##  2     42.73 1.673 95    38.47    46.99  1
##  1     46.84 1.673 95    42.58    51.10  1
##
## Tx_P = 2:
##  Tx_C emmean    SE df lower.CL upper.CL .group
##  1     59.10 1.691 95    54.79    63.40  1
##  2     87.08 1.673 95    82.82    91.34   2
##
## Results are averaged over the levels of: Mode, Domain
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_P = 1:
##  contrast estimate    SE df t.ratio p.value
##  1 - 2       4.113 2.360 95   1.742  0.0847
##
## Tx_P = 2:
##  contrast estimate    SE df t.ratio p.value
##  1 - 2      27.983 2.385 95  11.734  <.0001
##
## Results are averaged over the levels of: Mode, Domain

CLD(Tx_P_x_Tx_C_emm, by = "Tx_C", alpha = 0.05, adjust = "holm", details = TRUE)
```

```
## $emmeans
## Tx_C = 1:
##  Tx_P emmean    SE df lower.CL upper.CL .group
##  1     46.84 1.673 95    42.58    51.10  1
##  2     59.10 1.691 95    54.79    63.40   2
##
## Tx_C = 2:
##  Tx_P emmean    SE df lower.CL upper.CL .group
##  1     42.73 1.673 95    38.47    46.99  1
##  2     87.08 1.673 95    82.82    91.34   2
##
## Results are averaged over the levels of: Mode, Domain
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_C = 1:
##  contrast estimate    SE df t.ratio p.value
##  2 - 1       12.25 2.385 95   5.138  <.0001
##
## Tx_C = 2:
##  contrast estimate    SE df t.ratio p.value
##  2 - 1       44.35 2.360 95  18.789  <.0001
##
## Results are averaged over the levels of: Mode, Domain
```

```
CLD(Tx_P_x_Mode_emm, by = "Tx_P", alpha = 0.05, adjust = "holm", details = TRUE)
```

```
## $emmeans
## Tx_P = 1:
##  Mode     emmean    SE    df lower.CL upper.CL .group
##  Paper     38.08 1.292 129.4    34.80    41.35  1
##  Computer  51.49 1.292 129.4    48.22    54.77   2
##
## Tx_P = 2:
##  Mode     emmean    SE    df lower.CL upper.CL .group
##  Computer  68.47 1.295 130.0    65.19    71.75  1
##  Paper     77.70 1.295 130.0    74.43    80.98   2
##
## Results are averaged over the levels of: Domain, Tx_C
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_P = 1:
##  contrast         estimate    SE df t.ratio p.value
##  Computer - Paper   13.415 1.026 95  13.069  <.0001
##
## Tx_P = 2:
##  contrast         estimate    SE df t.ratio p.value
##  Computer - Paper    9.235 1.037 95   8.904  <.0001
```

```
##
## Results are averaged over the levels of: Domain, Tx_C
```

```r
CLD(Tx_P_x_Mode_emm, by = "Mode", alpha = 0.05, adjust = "holm", details = TRUE)
```

```
## $emmeans
## Mode = Computer:
##  Tx_P emmean    SE    df lower.CL upper.CL .group
##  1     51.49 1.292 129.4    48.22    54.77  1
##  2     68.47 1.295 130.0    65.19    71.75   2
##
## Mode = Paper:
##  Tx_P emmean    SE    df lower.CL upper.CL .group
##  1     38.08 1.292 129.4    34.80    41.35  1
##  2     77.70 1.295 130.0    74.43    80.98   2
##
## Results are averaged over the levels of: Domain, Tx_C
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Mode = Computer:
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1       16.98 1.829 129.7   9.279  <.0001
##
## Mode = Paper:
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1       39.63 1.829 129.7  21.660  <.0001
##
## Results are averaged over the levels of: Domain, Tx_C
```

```r
CLD(Tx_P_x_Domain_emm, by = "Tx_P", alpha = 0.05, adjust = "holm",
    details = TRUE)
```

```
## $emmeans
## Tx_P = 1:
##  Domain emmean    SE    df lower.CL upper.CL .group
##  Verbal  42.78 1.220 105.9    39.68    45.88  1
##  Quant   46.80 1.220 105.9    43.70    49.90   2
##
## Tx_P = 2:
##  Domain emmean    SE    df lower.CL upper.CL .group
##  Verbal  71.85 1.221 106.1    68.74    74.95  1
##  Quant   74.33 1.221 106.1    71.23    77.43   2
##
## Results are averaged over the levels of: Mode, Tx_C
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_P = 1:
##  contrast       estimate     SE df t.ratio p.value
```

```
##  Quant - Verbal    4.022 0.5697 95   7.060  <.0001
##
## Tx_P = 2:
##  contrast       estimate     SE df t.ratio p.value
##  Quant - Verbal    2.482 0.5756 95   4.312  <.0001
##
## Results are averaged over the levels of: Mode, Tx_C
```

```r
CLD(Tx_P_x_Domain_emm, by = "Domain", alpha = 0.05, adjust = "holm",
    details = TRUE)
```

```
## $emmeans
## Domain = Quant:
##  Tx_P emmean    SE    df lower.CL upper.CL .group
##  1     46.80 1.220 105.9    43.70    49.90  1
##  2     74.33 1.221 106.1    71.23    77.43   2
##
## Domain = Verbal:
##  Tx_P emmean    SE    df lower.CL upper.CL .group
##  1     42.78 1.220 105.9    39.68    45.88  1
##  2     71.85 1.221 106.1    68.74    74.95   2
##
## Results are averaged over the levels of: Mode, Tx_C
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Domain = Quant:
##  contrast estimate    SE  df t.ratio p.value
##  2 - 1       27.53 1.726 106   15.95  <.0001
##
## Domain = Verbal:
##  contrast estimate    SE  df t.ratio p.value
##  2 - 1       29.07 1.726 106   16.84  <.0001
##
## Results are averaged over the levels of: Mode, Tx_C
```

```r
CLD(Tx_C_x_Mode_emm, by = "Tx_C", alpha = 0.05, adjust = "holm", details = TRUE)
```

```
## $emmeans
## Tx_C = 1:
##  Mode     emmean    SE    df lower.CL upper.CL .group
##  Computer  47.99 1.295 130.0    44.71    51.26  1
##  Paper     57.95 1.295 130.0    54.67    61.23   2
##
## Tx_C = 2:
##  Mode     emmean    SE    df lower.CL upper.CL .group
##  Paper     57.83 1.292 129.4    54.56    61.11  1
##  Computer  71.98 1.292 129.4    68.70    75.25   2
##
## Results are averaged over the levels of: Domain, Tx_P
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
```

```
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_C = 1:
##  contrast         estimate    SE df t.ratio p.value
##  Paper - Computer    9.967 1.037 95    9.61  <.0001
##
## Tx_C = 2:
##  contrast         estimate    SE df t.ratio p.value
##  Paper - Computer   14.146 1.026 95   13.78  <.0001
##
## Results are averaged over the levels of: Domain, Tx_P
```

```
CLD(Tx_C_x_Mode_emm, by = "Mode", alpha = 0.05, adjust = "holm", details = TRUE)
```

```
## $emmeans
## Mode = Computer:
##  Tx_C emmean    SE    df lower.CL upper.CL .group
##  1     47.99 1.295 130.0    44.71    51.26  1
##  2     71.98 1.292 129.4    68.70    75.25   2
##
## Mode = Paper:
##  Tx_C emmean    SE    df lower.CL upper.CL .group
##  2     57.83 1.292 129.4    54.56    61.11  1
##  1     57.95 1.295 130.0    54.67    61.23  1
##
## Results are averaged over the levels of: Domain, Tx_P
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Mode = Computer:
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1     23.9919 1.829 129.7  13.114  <.0001
##
## Mode = Paper:
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1      0.1213 1.829 129.7   0.066  0.9472
##
## Results are averaged over the levels of: Domain, Tx_P
```

```
CLD(Tx_C_x_Domain_emm, by = "Tx_C", alpha = 0.05, adjust = "holm",
    details = TRUE)
```

```
## $emmeans
## Tx_C = 1:
##  Domain emmean    SE    df lower.CL upper.CL .group
##  Verbal  51.52 1.221 106.1    48.42    54.62  1
##  Quant   54.42 1.221 106.1    51.32    57.52   2
##
## Tx_C = 2:
##  Domain emmean    SE    df lower.CL upper.CL .group
##  Verbal  63.10 1.220 105.9    60.00    66.20  1
```

```
##  Quant   66.71 1.220 105.9    63.61    69.81   2
##
## Results are averaged over the levels of: Mode, Tx_P
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_C = 1:
##  contrast       estimate     SE df t.ratio p.value
##  Quant - Verbal    2.900 0.5756 95   5.038  <.0001
##
## Tx_C = 2:
##  contrast       estimate     SE df t.ratio p.value
##  Quant - Verbal    3.604 0.5697 95   6.327  <.0001
##
## Results are averaged over the levels of: Mode, Tx_P
```

```
CLD(Tx_C_x_Domain_emm, by = "Domain", alpha = 0.05, adjust = "holm",
    details = TRUE)
```

```
## $emmeans
## Domain = Quant:
##  Tx_C emmean    SE    df lower.CL upper.CL .group
##  1     54.42 1.221 106.1    51.32    57.52  1
##  2     66.71 1.220 105.9    63.61    69.81   2
##
## Domain = Verbal:
##  Tx_C emmean    SE    df lower.CL upper.CL .group
##  1     51.52 1.221 106.1    48.42    54.62  1
##  2     63.10 1.220 105.9    60.00    66.20   2
##
## Results are averaged over the levels of: Mode, Tx_P
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Domain = Quant:
##  contrast estimate    SE  df t.ratio p.value
##  2 - 1       12.29 1.726 106   7.120  <.0001
##
## Domain = Verbal:
##  contrast estimate    SE  df t.ratio p.value
##  2 - 1       11.58 1.726 106   6.712  <.0001
##
## Results are averaged over the levels of: Mode, Tx_P
```

```
CLD(Mode_x_Domain_emm, by = "Mode", alpha = 0.05, adjust = "holm",
    details = TRUE)
```

```
## $emmeans
## Mode = Computer:
##  Domain emmean     SE  df lower.CL upper.CL .group
```

```
##  Verbal  57.68 0.9612 157    55.25    60.11  1
##  Quant   62.28 0.9612 157    59.86    64.71   2
##
## Mode = Paper:
##  Domain emmean     SE  df lower.CL upper.CL .group
##  Verbal  56.94 0.9612 157    54.51    59.37  1
##  Quant   58.84 0.9612 157    56.41    61.27   2
##
## Results are averaged over the levels of: Tx_P, Tx_C
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Mode = Computer:
##  contrast       estimate     SE    df t.ratio p.value
##  Quant - Verbal    4.605 0.5911 189.3   7.791  <.0001
##
## Mode = Paper:
##  contrast       estimate     SE    df t.ratio p.value
##  Quant - Verbal    1.899 0.5911 189.3   3.212  0.0015
##
## Results are averaged over the levels of: Tx_P, Tx_C
```

```
CLD(Mode_x_Domain_emm, by = "Domain", alpha = 0.05, adjust = "holm",
    details = TRUE)
```

```
## $emmeans
## Domain = Quant:
##  Mode     emmean     SE  df lower.CL upper.CL .group
##  Paper     58.84 0.9612 157    56.41    61.27  1
##  Computer  62.28 0.9612 157    59.86    64.71   2
##
## Domain = Verbal:
##  Mode     emmean     SE  df lower.CL upper.CL .group
##  Paper     56.94 0.9612 157    54.51    59.37  1
##  Computer  57.68 0.9612 157    55.25    60.11  1
##
## Results are averaged over the levels of: Tx_P, Tx_C
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Domain = Quant:
##  contrast         estimate     SE  df t.ratio p.value
##  Computer - Paper   3.4432 0.8472 154   4.064  0.0001
##
## Domain = Verbal:
##  contrast         estimate     SE  df t.ratio p.value
##  Computer - Paper   0.7366 0.8472 154   0.869  0.3860
##
## Results are averaged over the levels of: Tx_P, Tx_C
```

```r
CLD(Tx_P_x_Tx_C_x_Mode_emm, by = "Tx_P", alpha = 0.05, adjust = "holm",
    details = TRUE)
```

```
## $emmeans
## Tx_P = 1:
##  Tx_C Mode     emmean    SE    df lower.CL upper.CL .group
##  2    Paper     28.47 1.824 129.5    23.40    33.54  1
##  1    Computer  46.00 1.824 129.5    40.93    51.07   2
##  1    Paper     47.68 1.824 129.5    42.61    52.75   2
##  2    Computer  56.99 1.824 129.5    51.92    62.06    3
##
## Tx_P = 2:
##  Tx_C Mode     emmean    SE    df lower.CL upper.CL .group
##  1    Computer  49.97 1.846 130.2    44.84    55.10  1
##  1    Paper     68.22 1.846 130.2    63.09    73.35   2
##  2    Computer  86.97 1.824 129.5    81.90    92.04    3
##  2    Paper     87.19 1.824 129.5    82.12    92.26    3
##
## Results are averaged over the levels of: Domain
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 8 estimates
## P value adjustment: holm method for 6 tests
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_P = 1:
##  contrast                estimate    SE    df t.ratio p.value
##  1,Computer - 2,Paper     17.5274 2.574 129.7   6.810  <.0001
##  1,Paper - 2,Paper        19.2111 2.574 129.7   7.464  <.0001
##  1,Paper - 1,Computer      1.6837 1.452  95.0   1.160  0.2490
##  2,Computer - 2,Paper     28.5130 1.452  95.0  19.642  <.0001
##  2,Computer - 1,Computer  10.9857 2.574 129.7   4.268  0.0001
##  2,Computer - 1,Paper      9.3020 2.574 129.7   3.614  0.0009
##
## Tx_P = 2:
##  contrast                estimate    SE    df t.ratio p.value
##  1,Computer - 2,Paper     18.2498 1.482  95.0  12.318  <.0001
##  1,Paper - 2,Paper        36.9982 2.601 129.7  14.227  <.0001
##  1,Paper - 1,Computer     18.7483 2.601 129.7   7.209  <.0001
##  2,Computer - 2,Paper     37.2183 2.601 129.7  14.312  <.0001
##  2,Computer - 1,Computer  18.9684 2.601 129.7   7.294  <.0001
##  2,Computer - 1,Paper      0.2201 1.452  95.0   0.152  0.8798
##
## Results are averaged over the levels of: Domain
## P value adjustment: holm method for 6 tests
```

```r
CLD(Tx_P_x_Tx_C_x_Mode_emm, by = "Tx_C", alpha = 0.05, adjust = "holm",
    details = TRUE)
```

```
## $emmeans
## Tx_C = 1:
##  Tx_P Mode     emmean    SE    df lower.CL upper.CL .group
##  1    Computer  46.00 1.824 129.5    40.93    51.07  1
##  1    Paper     47.68 1.824 129.5    42.61    52.75  1
##  2    Computer  49.97 1.846 130.2    44.84    55.10  1
```

```
##  2    Paper     68.22 1.846 130.2    63.09    73.35   2
##
## Tx_C = 2:
##  Tx_P Mode     emmean    SE    df lower.CL upper.CL .group
##  1    Paper     28.47 1.824 129.5    23.40    33.54  1
##  1    Computer  56.99 1.824 129.5    51.92    62.06   2
##  2    Computer  86.97 1.824 129.5    81.90    92.04    3
##  2    Paper     87.19 1.824 129.5    82.12    92.26    3
##
## Results are averaged over the levels of: Domain
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 8 estimates
## P value adjustment: holm method for 6 tests
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_C = 1:
##  contrast                estimate    SE    df t.ratio p.value
##  1,Paper - 1,Computer      1.6837 1.452  95.0   1.160  0.4980
##  2,Computer - 1,Computer   3.9700 2.601 129.7   1.527  0.3879
##  2,Computer - 1,Paper      2.2863 2.601 129.7   0.879  0.4980
##  2,Paper - 1,Computer     22.2198 2.601 129.7   8.544  <.0001
##  2,Paper - 1,Paper        20.5361 2.601 129.7   7.897  <.0001
##  2,Paper - 2,Computer     18.2498 1.482  95.0  12.318  <.0001
##
## Tx_C = 2:
##  contrast                estimate    SE    df t.ratio p.value
##  1,Paper - 1,Computer     28.5130 1.452  95.0  19.642  <.0001
##  2,Computer - 1,Computer  58.4956 2.574 129.7  22.727  <.0001
##  2,Computer - 1,Paper     29.9825 2.574 129.7  11.649  <.0001
##  2,Paper - 1,Computer     58.7157 2.574 129.7  22.812  <.0001
##  2,Paper - 1,Paper        30.2026 2.574 129.7  11.734  <.0001
##  2,Paper - 2,Computer      0.2201 1.452  95.0   0.152  0.8798
##
## Results are averaged over the levels of: Domain
## P value adjustment: holm method for 6 tests
```

```
CLD(Tx_P_x_Tx_C_x_Mode_emm, by = "Mode", alpha = 0.05, adjust = "holm",
    details = TRUE)
```

```
## $emmeans
## Mode = Computer:
##  Tx_P Tx_C emmean    SE    df lower.CL upper.CL .group
##  1    1     46.00 1.824 129.5    40.93    51.07  1
##  2    1     49.97 1.846 130.2    44.84    55.10  1
##  1    2     56.99 1.824 129.5    51.92    62.06   2
##  2    2     86.97 1.824 129.5    81.90    92.04    3
##
## Mode = Paper:
##  Tx_P Tx_C emmean    SE    df lower.CL upper.CL .group
##  1    2     28.47 1.824 129.5    23.40    33.54  1
##  1    1     47.68 1.824 129.5    42.61    52.75   2
##  2    1     68.22 1.846 130.2    63.09    73.35    3
##  2    2     87.19 1.824 129.5    82.12    92.26     4
```

```
##
## Results are averaged over the levels of: Domain
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 8 estimates
## P value adjustment: holm method for 6 tests
## significance level used: alpha = 0.05
##
## $comparisons
## Mode = Computer:
##  contrast  estimate    SE    df t.ratio p.value
##  2,1 - 1,1    3.970 2.601 129.7   1.527  0.1293
##  1,2 - 1,1   10.986 2.574 129.7   4.268  0.0001
##  1,2 - 2,1    7.016 2.601 129.7   2.698  0.0158
##  2,2 - 1,1   40.968 2.574 129.7  15.917  <.0001
##  2,2 - 2,1   36.998 2.601 129.7  14.227  <.0001
##  2,2 - 1,2   29.983 2.574 129.7  11.649  <.0001
##
## Mode = Paper:
##  contrast  estimate    SE    df t.ratio p.value
##  2,1 - 1,1   19.211 2.574 129.7   7.464  <.0001
##  1,2 - 1,1   39.747 2.601 129.7  15.284  <.0001
##  1,2 - 2,1   20.536 2.601 129.7   7.897  <.0001
##  2,2 - 1,1   58.716 2.574 129.7  22.812  <.0001
##  2,2 - 2,1   39.505 2.574 129.7  15.348  <.0001
##  2,2 - 1,2   18.968 2.601 129.7   7.294  <.0001
##
## Results are averaged over the levels of: Domain
## P value adjustment: holm method for 6 tests
```

```r
CLD(Tx_P_x_Tx_C_x_Mode_emm, by = c("Tx_P", "Tx_C"), alpha = 0.05,
    adjust = "holm", details = TRUE)
```

```
## $emmeans
## Tx_P = 1, Tx_C = 1:
##  Mode     emmean    SE    df lower.CL upper.CL .group
##  Computer  46.00 1.824 129.5    40.93    51.07  1
##  Paper     47.68 1.824 129.5    42.61    52.75  1
##
## Tx_P = 1, Tx_C = 2:
##  Mode     emmean    SE    df lower.CL upper.CL .group
##  Paper     28.47 1.824 129.5    23.40    33.54  1
##  Computer  56.99 1.824 129.5    51.92    62.06   2
##
## Tx_P = 2, Tx_C = 1:
##  Mode     emmean    SE    df lower.CL upper.CL .group
##  Computer  49.97 1.846 130.2    44.84    55.10  1
##  Paper     68.22 1.846 130.2    63.09    73.35   2
##
## Tx_P = 2, Tx_C = 2:
##  Mode     emmean    SE    df lower.CL upper.CL .group
##  Computer  86.97 1.824 129.5    81.90    92.04  1
##  Paper     87.19 1.824 129.5    82.12    92.26  1
##
## Results are averaged over the levels of: Domain
```

```
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 8 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_P = 1, Tx_C = 1:
##  contrast         estimate    SE df t.ratio p.value
##  Paper - Computer   1.6837 1.452 95   1.160  0.2490
##
## Tx_P = 1, Tx_C = 2:
##  contrast         estimate    SE df t.ratio p.value
##  Paper - Computer  28.5130 1.452 95  19.642  <.0001
##
## Tx_P = 2, Tx_C = 1:
##  contrast         estimate    SE df t.ratio p.value
##  Paper - Computer  18.2498 1.482 95  12.318  <.0001
##
## Tx_P = 2, Tx_C = 2:
##  contrast         estimate    SE df t.ratio p.value
##  Paper - Computer   0.2201 1.452 95   0.152  0.8798
##
## Results are averaged over the levels of: Domain
```

```
CLD(Tx_P_x_Tx_C_x_Mode_emm, by = c("Tx_P", "Mode"), alpha = 0.05,
    adjust = "holm", details = TRUE)
```

```
## $emmeans
## Tx_P = 1, Mode = Computer:
##  Tx_C emmean    SE    df lower.CL upper.CL .group
##  1     46.00 1.824 129.5    40.93    51.07  1
##  2     56.99 1.824 129.5    51.92    62.06   2
##
## Tx_P = 1, Mode = Paper:
##  Tx_C emmean    SE    df lower.CL upper.CL .group
##  2     28.47 1.824 129.5    23.40    33.54  1
##  1     47.68 1.824 129.5    42.61    52.75   2
##
## Tx_P = 2, Mode = Computer:
##  Tx_C emmean    SE    df lower.CL upper.CL .group
##  1     49.97 1.846 130.2    44.84    55.10  1
##  2     86.97 1.824 129.5    81.90    92.04   2
##
## Tx_P = 2, Mode = Paper:
##  Tx_C emmean    SE    df lower.CL upper.CL .group
##  1     68.22 1.846 130.2    63.09    73.35  1
##  2     87.19 1.824 129.5    82.12    92.26   2
##
## Results are averaged over the levels of: Domain
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 8 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_P = 1, Mode = Computer:
```

```
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1       10.99 2.574 129.7   4.268  <.0001
##
## Tx_P = 1, Mode = Paper:
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1       19.21 2.574 129.7   7.464  <.0001
##
## Tx_P = 2, Mode = Computer:
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1       37.00 2.601 129.7  14.227  <.0001
##
## Tx_P = 2, Mode = Paper:
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1       18.97 2.601 129.7   7.294  <.0001
##
## Results are averaged over the levels of: Domain
```

```
CLD(Tx_P_x_Tx_C_x_Mode_emm, by = c("Tx_C", "Mode"), alpha = 0.05,
    adjust = "holm", details = TRUE)
```

```
## $emmeans
## Tx_C = 1, Mode = Computer:
##  Tx_P emmean    SE    df lower.CL upper.CL .group
##  1     46.00 1.824 129.5    40.93    51.07  1
##  2     49.97 1.846 130.2    44.84    55.10  1
##
## Tx_C = 1, Mode = Paper:
##  Tx_P emmean    SE    df lower.CL upper.CL .group
##  1     47.68 1.824 129.5    42.61    52.75  1
##  2     68.22 1.846 130.2    63.09    73.35   2
##
## Tx_C = 2, Mode = Computer:
##  Tx_P emmean    SE    df lower.CL upper.CL .group
##  1     56.99 1.824 129.5    51.92    62.06  1
##  2     86.97 1.824 129.5    81.90    92.04   2
##
## Tx_C = 2, Mode = Paper:
##  Tx_P emmean    SE    df lower.CL upper.CL .group
##  1     28.47 1.824 129.5    23.40    33.54  1
##  2     87.19 1.824 129.5    82.12    92.26   2
##
## Results are averaged over the levels of: Domain
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 8 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_C = 1, Mode = Computer:
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1        3.97 2.601 129.7   1.527  0.1293
##
## Tx_C = 1, Mode = Paper:
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1       20.54 2.601 129.7   7.897  <.0001
```

```
##
## Tx_C = 2, Mode = Computer:
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1       29.98 2.574 129.7  11.649  <.0001
##
## Tx_C = 2, Mode = Paper:
##  contrast estimate    SE    df t.ratio p.value
##  2 - 1       58.72 2.574 129.7  22.812  <.0001
##
## Results are averaged over the levels of: Domain
```

```
CLD(cell_means_emm, by = c("Tx_P"), alpha = 0.05, adjust = "holm",
    details = TRUE)
```

```
## $emmeans
## Tx_P = 1:
##  Tx_C Mode     Domain emmean    SE    df lower.CL upper.CL
##  2    Paper    Verbal  24.17 1.916 156.7    18.41    29.92
##  2    Paper    Quant   32.78 1.916 156.7    27.03    38.53
##  1    Computer Verbal  45.72 1.916 156.7    39.97    51.47
##  1    Computer Quant   46.28 1.916 156.7    40.53    52.04
##  1    Paper    Quant   47.51 1.916 156.7    41.76    53.27
##  1    Paper    Verbal  47.85 1.916 156.7    42.10    53.61
##  2    Computer Verbal  53.36 1.916 156.7    47.61    59.11
##  2    Computer Quant   60.61 1.916 156.7    54.86    66.36
##  .group
##   1
##    2
##     3
##     3
##     3
##     3
##     3
##      4
##
## Tx_P = 2:
##  Tx_C Mode     Domain emmean    SE    df lower.CL upper.CL
##  1    Computer Verbal  49.20 1.941 158.0    43.38    55.03
##  1    Computer Quant   50.74 1.941 158.0    44.91    56.56
##  1    Paper    Verbal  63.30 1.941 158.0    57.48    69.13
##  1    Paper    Quant   73.14 1.941 158.0    67.32    78.97
##  2    Paper    Quant   81.93 1.916 156.7    76.18    87.68
##  2    Computer Verbal  82.43 1.916 156.7    76.68    88.18
##  2    Computer Quant   91.51 1.916 156.7    85.75    97.26
##  2    Paper    Verbal  92.45 1.916 156.7    86.70    98.20
##  .group
##   1
##   1
##    2
##     3
##      4
##      4
##       5
```

```
##       5
##
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 16 estimates
## P value adjustment: holm method for 28 tests
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_P = 1:
##  contrast                               estimate    SE    df
##  2,Paper,Quant - 2,Paper,Verbal           8.6125 1.176 189.3
##  1,Computer,Verbal - 2,Paper,Verbal      21.5517 2.705 157.0
##  1,Computer,Verbal - 2,Paper,Quant       12.9393 2.705 157.0
##  1,Computer,Quant - 2,Paper,Verbal       22.1155 2.705 157.0
##  1,Computer,Quant - 2,Paper,Quant        13.5031 2.705 157.0
##  1,Computer,Quant - 1,Computer,Verbal     0.5638 1.176 189.3
##  1,Paper,Quant - 2,Paper,Verbal          23.3479 2.705 157.0
##  1,Paper,Quant - 2,Paper,Quant           14.7354 2.705 157.0
##  1,Paper,Quant - 1,Computer,Verbal        1.7962 1.660 148.4
##  1,Paper,Quant - 1,Computer,Quant         1.2324 1.686 154.0
##  1,Paper,Verbal - 2,Paper,Verbal         23.6867 2.705 157.0
##  1,Paper,Verbal - 2,Paper,Quant          15.0742 2.705 157.0
##  1,Paper,Verbal - 1,Computer,Verbal       2.1350 1.686 154.0
##  1,Paper,Verbal - 1,Computer,Quant        1.5712 1.660 148.4
##  1,Paper,Verbal - 1,Paper,Quant           0.3388 1.176 189.3
##  2,Computer,Verbal - 2,Paper,Verbal      29.1946 1.686 154.0
##  2,Computer,Verbal - 2,Paper,Quant       20.5821 1.660 148.4
##  2,Computer,Verbal - 1,Computer,Verbal    7.6429 2.705 157.0
##  2,Computer,Verbal - 1,Computer,Quant     7.0791 2.705 157.0
##  2,Computer,Verbal - 1,Paper,Quant        5.8467 2.705 157.0
##  2,Computer,Verbal - 1,Paper,Verbal       5.5079 2.705 157.0
##  2,Computer,Quant - 2,Paper,Verbal       36.4440 1.660 148.4
##  2,Computer,Quant - 2,Paper,Quant        27.8315 1.686 154.0
##  2,Computer,Quant - 1,Computer,Verbal    14.8922 2.705 157.0
##  2,Computer,Quant - 1,Computer,Quant     14.3284 2.705 157.0
##  2,Computer,Quant - 1,Paper,Quant        13.0960 2.705 157.0
##  2,Computer,Quant - 1,Paper,Verbal       12.7572 2.705 157.0
##  2,Computer,Quant - 2,Computer,Verbal     7.2493 1.176 189.3
##  t.ratio p.value
##    7.323  <.0001
##    7.968  <.0001
##    4.784  <.0001
##    8.176  <.0001
##    4.992  <.0001
##    0.479  1.0000
##    8.632  <.0001
##    5.448  <.0001
##    1.082  1.0000
##    0.731  1.0000
##    8.757  <.0001
##    5.573  <.0001
##    1.267  1.0000
##    0.946  1.0000
##    0.288  1.0000
```

```
##   17.320  <.0001
##   12.397  <.0001
##    2.826  0.0533
##    2.617  0.0876
##    2.162  0.2574
##    2.036  0.3038
##   21.951  <.0001
##   16.511  <.0001
##    5.506  <.0001
##    5.297  <.0001
##    4.842  <.0001
##    4.716  0.0001
##    6.164  <.0001
##
## Tx_P = 2:
##  contrast                               estimate    SE    df
##  2,Paper,Quant - 2,Paper,Verbal           1.5349 1.200 189.3
##  1,Computer,Verbal - 2,Paper,Verbal      14.0976 1.720 154.0
##  1,Computer,Verbal - 2,Paper,Quant       12.5627 1.694 148.4
##  1,Computer,Quant - 2,Paper,Verbal       23.9370 1.694 148.4
##  1,Computer,Quant - 2,Paper,Quant        22.4021 1.720 154.0
##  1,Computer,Quant - 1,Computer,Verbal     9.8394 1.200 189.3
##  1,Paper,Quant - 2,Paper,Verbal          32.7264 2.733 157.0
##  1,Paper,Quant - 2,Paper,Quant           31.1915 2.733 157.0
##  1,Paper,Quant - 1,Computer,Verbal       18.6289 2.733 157.0
##  1,Paper,Quant - 1,Computer,Quant         8.7894 2.733 157.0
##  1,Paper,Verbal - 2,Paper,Verbal         33.2292 2.733 157.0
##  1,Paper,Verbal - 2,Paper,Quant          31.6943 2.733 157.0
##  1,Paper,Verbal - 1,Computer,Verbal      19.1316 2.733 157.0
##  1,Paper,Verbal - 1,Computer,Quant        9.2922 2.733 157.0
##  1,Paper,Verbal - 1,Paper,Quant           0.5027 1.660 148.4
##  2,Computer,Verbal - 2,Paper,Verbal      42.3021 2.733 157.0
##  2,Computer,Verbal - 2,Paper,Quant       40.7672 2.733 157.0
##  2,Computer,Verbal - 1,Computer,Verbal  28.2045 2.733 157.0
##  2,Computer,Verbal - 1,Computer,Quant   18.3651 2.733 157.0
##  2,Computer,Verbal - 1,Paper,Quant       9.5757 1.686 154.0
##  2,Computer,Verbal - 1,Paper,Verbal      9.0729 1.176 189.3
##  2,Computer,Quant - 2,Paper,Verbal      43.2450 2.733 157.0
##  2,Computer,Quant - 2,Paper,Quant       41.7101 2.733 157.0
##  2,Computer,Quant - 1,Computer,Verbal   29.1475 2.733 157.0
##  2,Computer,Quant - 1,Computer,Quant    19.3080 2.733 157.0
##  2,Computer,Quant - 1,Paper,Quant       10.5186 1.176 189.3
##  2,Computer,Quant - 1,Paper,Verbal      10.0158 1.686 154.0
##  2,Computer,Quant - 2,Computer,Verbal    0.9429 1.660 148.4
##  t.ratio p.value
##    1.279  0.6077
##    8.194  <.0001
##    7.414  <.0001
##   14.127  <.0001
##   13.021  <.0001
##    8.197  <.0001
##   11.975  <.0001
##   11.413  <.0001
##    6.816  <.0001
```

```
##    3.216  0.0063
##   12.159  <.0001
##   11.597  <.0001
##    7.000  <.0001
##    3.400  0.0043
##    0.303  1.0000
##   15.479  <.0001
##   14.917  <.0001
##   10.320  <.0001
##    6.720  <.0001
##    5.681  <.0001
##    7.715  <.0001
##   15.824  <.0001
##   15.262  <.0001
##   10.665  <.0001
##    7.065  <.0001
##    8.944  <.0001
##    5.942  <.0001
##    0.568  1.0000
##
## P value adjustment: holm method for 28 tests
```

```
CLD(cell_means_emm, by = c("Tx_P", "Tx_C"), alpha = 0.05, adjust = "holm",
    details = TRUE)
```

```
## $emmeans
## Tx_P = 1, Tx_C = 1:
##  Mode     Domain emmean    SE    df lower.CL upper.CL .group
##  Computer Verbal  45.72 1.916 156.7    39.97    51.47  1
##  Computer Quant   46.28 1.916 156.7    40.53    52.04  1
##  Paper    Quant   47.51 1.916 156.7    41.76    53.27  1
##  Paper    Verbal  47.85 1.916 156.7    42.10    53.61  1
##
## Tx_P = 1, Tx_C = 2:
##  Mode     Domain emmean    SE    df lower.CL upper.CL .group
##  Paper    Verbal  24.17 1.916 156.7    18.41    29.92  1
##  Paper    Quant   32.78 1.916 156.7    27.03    38.53   2
##  Computer Verbal  53.36 1.916 156.7    47.61    59.11    3
##  Computer Quant   60.61 1.916 156.7    54.86    66.36     4
##
## Tx_P = 2, Tx_C = 1:
##  Mode     Domain emmean    SE    df lower.CL upper.CL .group
##  Computer Verbal  49.20 1.941 158.0    43.38    55.03  1
##  Computer Quant   50.74 1.941 158.0    44.91    56.56  1
##  Paper    Verbal  63.30 1.941 158.0    57.48    69.13   2
##  Paper    Quant   73.14 1.941 158.0    67.32    78.97    3
##
## Tx_P = 2, Tx_C = 2:
##  Mode     Domain emmean    SE    df lower.CL upper.CL .group
##  Paper    Quant   81.93 1.916 156.7    76.18    87.68  1
##  Computer Verbal  82.43 1.916 156.7    76.68    88.18  1
##  Computer Quant   91.51 1.916 156.7    85.75    97.26   2
##  Paper    Verbal  92.45 1.916 156.7    86.70    98.20   2
##
```

```
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 16 estimates
## P value adjustment: holm method for 6 tests
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_P = 1, Tx_C = 1:
##  contrast                          estimate    SE    df t.ratio
##  Computer,Quant - Computer,Verbal    0.5638 1.176 189.3   0.479
##  Paper,Quant - Computer,Verbal       1.7962 1.660 148.4   1.082
##  Paper,Quant - Computer,Quant        1.2324 1.686 154.0   0.731
##  Paper,Verbal - Computer,Verbal      2.1350 1.686 154.0   1.267
##  Paper,Verbal - Computer,Quant       1.5712 1.660 148.4   0.946
##  Paper,Verbal - Paper,Quant          0.3388 1.176 189.3   0.288
##  p.value
##   1.0000
##   1.0000
##   1.0000
##   1.0000
##   1.0000
##   1.0000
##
## Tx_P = 1, Tx_C = 2:
##  contrast                          estimate    SE    df t.ratio
##  Computer,Quant - Computer,Verbal    8.6125 1.176 189.3   7.323
##  Paper,Quant - Computer,Verbal      29.1946 1.686 154.0  17.320
##  Paper,Quant - Computer,Quant       20.5821 1.660 148.4  12.397
##  Paper,Verbal - Computer,Verbal     36.4440 1.660 148.4  21.951
##  Paper,Verbal - Computer,Quant      27.8315 1.686 154.0  16.511
##  Paper,Verbal - Paper,Quant          7.2493 1.176 189.3   6.164
##  p.value
##   <.0001
##   <.0001
##   <.0001
##   <.0001
##   <.0001
##   <.0001
##
## Tx_P = 2, Tx_C = 1:
##  contrast                          estimate    SE    df t.ratio
##  Computer,Quant - Computer,Verbal    1.5349 1.200 189.3   1.279
##  Paper,Quant - Computer,Verbal      14.0976 1.720 154.0   8.194
##  Paper,Quant - Computer,Quant       12.5627 1.694 148.4   7.414
##  Paper,Verbal - Computer,Verbal     23.9370 1.694 148.4  14.127
##  Paper,Verbal - Computer,Quant      22.4021 1.720 154.0  13.021
##  Paper,Verbal - Paper,Quant          9.8394 1.200 189.3   8.197
##  p.value
##   0.2026
##   <.0001
##   <.0001
##   <.0001
##   <.0001
##   <.0001
##
```

```
## Tx_P = 2, Tx_C = 2:
##  contrast                          estimate    SE    df t.ratio
##  Computer,Quant - Computer,Verbal   0.5027 1.660 148.4   0.303
##  Paper,Quant - Computer,Verbal      9.5757 1.686 154.0   5.681
##  Paper,Quant - Computer,Quant       9.0729 1.176 189.3   7.715
##  Paper,Verbal - Computer,Verbal    10.5186 1.176 189.3   8.944
##  Paper,Verbal - Computer,Quant     10.0158 1.686 154.0   5.942
##  Paper,Verbal - Paper,Quant         0.9429 1.660 148.4   0.568
##  p.value
##   1.0000
##   <.0001
##   <.0001
##   <.0001
##   <.0001
##   1.0000
##
## P value adjustment: holm method for 6 tests
```

```
CLD(cell_means_emm, by = c("Tx_P", "Tx_C", "Mode"), alpha = 0.05,
    adjust = "holm", details = TRUE)
```

```
## $emmeans
## Tx_P = 1, Tx_C = 1, Mode = Computer:
##  Domain emmean    SE    df lower.CL upper.CL .group
##  Verbal  45.72 1.916 156.7    39.97    51.47  1
##  Quant   46.28 1.916 156.7    40.53    52.04  1
##
## Tx_P = 1, Tx_C = 1, Mode = Paper:
##  Domain emmean    SE    df lower.CL upper.CL .group
##  Quant   47.51 1.916 156.7    41.76    53.27  1
##  Verbal  47.85 1.916 156.7    42.10    53.61  1
##
## Tx_P = 1, Tx_C = 2, Mode = Computer:
##  Domain emmean    SE    df lower.CL upper.CL .group
##  Verbal  53.36 1.916 156.7    47.61    59.11  1
##  Quant   60.61 1.916 156.7    54.86    66.36   2
##
## Tx_P = 1, Tx_C = 2, Mode = Paper:
##  Domain emmean    SE    df lower.CL upper.CL .group
##  Verbal  24.17 1.916 156.7    18.41    29.92  1
##  Quant   32.78 1.916 156.7    27.03    38.53   2
##
## Tx_P = 2, Tx_C = 1, Mode = Computer:
##  Domain emmean    SE    df lower.CL upper.CL .group
##  Verbal  49.20 1.941 158.0    43.38    55.03  1
##  Quant   50.74 1.941 158.0    44.91    56.56  1
##
## Tx_P = 2, Tx_C = 1, Mode = Paper:
##  Domain emmean    SE    df lower.CL upper.CL .group
##  Verbal  63.30 1.941 158.0    57.48    69.13  1
##  Quant   73.14 1.941 158.0    67.32    78.97   2
##
## Tx_P = 2, Tx_C = 2, Mode = Computer:
##  Domain emmean    SE    df lower.CL upper.CL .group
```

```
##  Verbal  82.43 1.916 156.7    76.68    88.18  1
##  Quant   91.51 1.916 156.7    85.75    97.26   2
##
## Tx_P = 2, Tx_C = 2, Mode = Paper:
##  Domain emmean    SE    df lower.CL upper.CL .group
##  Quant   81.93 1.916 156.7    76.18    87.68  1
##  Verbal  92.45 1.916 156.7    86.70    98.20   2
##
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 16 estimates
## significance level used: alpha = 0.05
##
## $comparisons
## Tx_P = 1, Tx_C = 1, Mode = Computer:
##  contrast       estimate    SE    df t.ratio p.value
##  Quant - Verbal   0.5638 1.176 189.3   0.479  0.6322
##
## Tx_P = 1, Tx_C = 1, Mode = Paper:
##  contrast       estimate    SE    df t.ratio p.value
##  Quant - Verbal   0.3388 1.176 189.3   0.288  0.7736
##
## Tx_P = 1, Tx_C = 2, Mode = Computer:
##  contrast       estimate    SE    df t.ratio p.value
##  Quant - Verbal   7.2493 1.176 189.3   6.164  <.0001
##
## Tx_P = 1, Tx_C = 2, Mode = Paper:
##  contrast       estimate    SE    df t.ratio p.value
##  Quant - Verbal   8.6125 1.176 189.3   7.323  <.0001
##
## Tx_P = 2, Tx_C = 1, Mode = Computer:
##  contrast       estimate    SE    df t.ratio p.value
##  Quant - Verbal   1.5349 1.200 189.3   1.279  0.2026
##
## Tx_P = 2, Tx_C = 1, Mode = Paper:
##  contrast       estimate    SE    df t.ratio p.value
##  Quant - Verbal   9.8394 1.200 189.3   8.197  <.0001
##
## Tx_P = 2, Tx_C = 2, Mode = Computer:
##  contrast       estimate    SE    df t.ratio p.value
##  Quant - Verbal   9.0729 1.176 189.3   7.715  <.0001
##
## Tx_P = 2, Tx_C = 2, Mode = Paper:
##  contrast       estimate    SE    df t.ratio p.value
##  Quant - Verbal  10.5186 1.176 189.3   8.944  <.0001
```

```r
Sys.time() - how_long
```

```
## Time difference of 27.9 secs
```