

CGM Analyses

Libraries and data

Libraries

```
library(tidyverse) # general functionality beyond base R
library(qualtrics) # for importing Qualtrics data sets
library(car) # for Anova(), i.e., Type III ANOVA, contrast testing
library(emmeans) # for simple effects
```

Load data

Qualtrics data files:

```
dfE1 <- read_survey("CGM-E1-data.csv")
dfE2 <- read_survey("CGM-E2-data.csv")
dfE3 <- read_survey("CGM-E3-data.csv")
```

Filter out incomplete entries

Incomplete entries are marked in Qualtrics as FALSE in the Finished column.

```
dfE1 <- dfE1[dfE1$Finished==TRUE,]
dfE2 <- dfE2[dfE2$Finished==TRUE,]
dfE3 <- dfE3[dfE3$Finished==TRUE,]
```

Filter out test data

In the `DistributionChannel` column, values completed by participants are labeled `anonymous`. Values generated during research team testing are labeled `preview`.

```
dfE1 <- dfE1[dfE1$DistributionChannel=="anonymous",]  
dfE2 <- dfE2[dfE2$DistributionChannel=="anonymous",]  
dfE3 <- dfE3[dfE3$DistributionChannel=="anonymous",]
```

Filter out nonprofessional investor participants

We include a question in our post-experiment questionnaire about investment experience as an added check that our participants are from the intended population (experienced investors). In all experiments, we exclude from our analysis any participants who respond inconsistently between the screening questions and the demographic questions (*i.e.*, who report never having made investments despite having previously self-reported to Prolific that they have both investment experience and experience analyzing financial statements when making investments). This inconsistency suggests either (a) they did not carefully attend to our survey questions, or (b) they intentionally misrepresented themselves to Prolific to qualify for our study. In either case, their responses are likely unreliable and may not reflect our intended population (investors with at least some experience).

```
dfE1 <- dfE1[dfE1$investExperience=="Yes",]  
dfE2 <- dfE2[dfE2$investExperience=="Yes",]  
dfE3 <- dfE3[dfE3$InvExp=="Yes",]
```

Dependent measure cleanup

Experiments 1 and 2: Aesthetic changes to dependent measure column names.

```
dfE1 <- dfE1 %>%  
  rename(  
    dvRisk = dvRisk_1,  
    dvInvestment = dvInvestment_1  
  )
```

```
dfE2 <- dfE2 %>%  
  rename(  
    dvRisk = dvRisk_1  
  )
```

Experiment 3: Because we counterbalanced order in which we collected dependent measures in Experiment 3, we need to combine the counterbalanced data into a single column (one column for each measure).

```
dfE3$dvRisk <- case_when(  
  dfE3$Order == 0 ~ dfE3$DvRisk1_1,  
  dfE3$Order == 1 ~ dfE3$DVRisk0_1  
)
```

```
dfE3$dvAR <- case_when(  
  dfE3$Order == 0 ~ dfE3$AR0_1,  
  dfE3$Order == 1 ~ dfE3$AR1_1  
)
```

Independent measure cleanup

Cleaning up condition labels (aesthetic/readability reasons)

```
dfE1 <- dfE1 %>%  
  rename(  
    unlinkedStrBSlabel = tradStrTradLabel,  
    unlinkedStrISlabel = tradStrNewLabel,  
    linkedStrBSlabel = newStrTradLabel,  
    linkedStrISlabel = newStrNewLabel  
  )
```

```
dfE2 <- dfE2 %>%  
  rename(  
    indirectLowerRisk = IMG,  
    indirectHigherRisk = IMB,  
    theoreticalLowerRisk = NMG,  
    theoreticalHigherRisk = NMB  
  )
```

Creating individual labels for each ANOVA factor

```
dfE1$Linking <- case_when(  
  dfE1$unlinkedStrISlabel == 1 | dfE1$unlinkedStrBSlabel == 1 ~  
  ↪ "unlinked",
```

```

        dfE1$linkedStrISlabel == 1 | dfE1$linkedStrBSlabel == 1 ~ "linked",
        dfE1$directMethod == 1 ~ "directMethod"
    )

dfE1$Linking <- factor(dfE1$Linking, levels = c("unlinked", "linked",
↪ "directMethod"))

```

```

dfE1$Labeling <- case_when(
    dfE1$unlinkedStrBSlabel == 1 | dfE1$linkedStrBSlabel == 1 ~
↪ "balanceSheet",
    dfE1$unlinkedStrISlabel == 1 | dfE1$linkedStrISlabel == 1 ~
↪ "incomeStmt",
    dfE1$directMethod == 1 ~ "directMethod"
)

dfE1$Labeling <- factor(dfE1$Labeling, levels = c("balanceSheet",
↪ "incomeStmt", "directMethod"))

```

```

dfE2$Presentation <- case_when(
    dfE2$indirectHigherRisk == 1 | dfE2$indirectLowerRisk == 1 ~
↪ "indirectMethod",
    dfE2$theoreticalHigherRisk == 1 | dfE2$theoreticalLowerRisk == 1 ~
↪ "theoreticalPresentation"
)

dfE2$Presentation <- factor(dfE2$Presentation, levels = c("indirectMethod",
↪ "theoreticalPresentation"))

```

```

dfE2$liquidityRisk <- case_when(
    dfE2$indirectLowerRisk == 1 | dfE2$theoreticalLowerRisk == 1 ~
↪ "lowerRisk",
    dfE2$indirectHigherRisk == 1 | dfE2$theoreticalHigherRisk == 1 ~
↪ "higherRisk"
)

dfE2$liquidityRisk <- factor(dfE2$liquidityRisk, levels = c("lowerRisk",
↪ "higherRisk"))

```

```

dfE3$Presentation <- case_when(
    dfE3$traditional == 1 ~ "indirectMethod",
    dfE3$new == 1 ~ "theoreticalPresentation"
)

```

```
)

dfE3$Presentation <- factor(dfE3$Presentation, levels = c("indirectMethod",
↪ "theoreticalPresentation"))
```

```
dfE3$Order <- case_when(
  dfE3$Order == 0 ~ "arFirst",
  dfE3$Order == 1 ~ "riskFirst"
)

dfE3$Order <- factor(dfE3$Order, levels = c("arFirst", "riskFirst"))
```

Add interaction term (for contrast coding)

```
dfE1$interaction <- interaction(dfE1$Linking, dfE1$Labeling)
dfE2$interaction <- interaction(dfE2$Presentation, dfE2$liquidityRisk)
```

Experiment 1: Creating separate dataframe without direct method

For Experiment 1, we filter out Direct Method data because our contrast test and ANOVA compares only the four cells of our 2×2 interaction.

```
dfE1noDM <- subset(dfE1, Linking != "directMethod")
# Reset factor levels from main data frame
# to drop "Direct Method" factor level
dfE1noDM <- droplevels(dfE1noDM)
# Restate factors for the "no DM" data frame
dfE1noDM$Linking <- factor(dfE1noDM$Linking, levels = c("unlinked",
↪ "linked"))
dfE1noDM$Labeling <- factor(dfE1noDM$Labeling, levels = c("balanceSheet",
↪ "incomeStmt"))
```

Statistical tests performed in R

Table 1

Descriptive statistics

```
as.data.frame(  
  dfE1 %>%  
  group_by(Linking, Labeling) %>%  
  summarize(  
    mean = mean(dvRisk),  
    sd = sd(dvRisk),  
    n = n(),  
    .groups = "drop"  
  )  
)
```

	Linking	Labeling	mean	sd	n
1	unlinked	balanceSheet	48.87234	21.70115	47
2	unlinked	incomeStmt	48.38095	24.26777	42
3	linked	balanceSheet	44.30435	23.47942	46
4	linked	incomeStmt	55.72500	24.76141	40
5	directMethod	directMethod	48.73810	25.59159	42

ANOVA

Again, for Experiment 1, we filter out Direct Method data in our main ANOVA because this test compares only the four cells of our 2×2 interaction.

```
anovaElrisk <- lm(dvRisk ~ Linking * Labeling, data = subset(dfE1, Linking !=  
  ↪ "directMethod"), contrasts = list(Linking = contr.sum, Labeling =  
  ↪ contr.sum))  
aovElrisk <- as.data.frame(Anova(anovaElrisk, type = "III"))  
aovElrisk
```

	Sum Sq	Df	F value	Pr(>F)
(Intercept)	423853.35515	1	766.7386357	4.326522e-65
Linking	83.92552	1	0.1518189	6.972881e-01
Labeling	1300.82754	1	2.3531599	1.268774e-01

Linking:Labeling	1545.29067	1	2.7953868	9.636478e-02
Residuals	94528.85293	171	NA	NA

Contrast

The following is a manual replication of the analysis that the authors performed in CompTrasts (<https://choicelab.shinyapps.io/ContrastAppShinyIO/>).

Planned contrast variance table

Fit model and run contrast test

```
# Planned contrast
model <- lm(dvRisk ~ 0 + interaction, data = dfE1noDM)
means <- emmeans(model, ~ interaction)
weights <- list(c(-1,-1,-1,3))
contrastTest <- summary(contrast(means,weights))

# Residual interaction
residualAnova <- anova(lm(dvRisk ~ interaction, data = dfE1noDM))
```

Extract residual ANOVA values

```
# Extract table values from residual interaction ANOVA
ssFactor <- residualAnova["interaction", "Sum Sq"]
ssError <- residualAnova$`Sum Sq`[2]
dfError <- residualAnova$Df[2]
msError <- residualAnova$`Mean Sq`[2]
```

Extract and compute planned contrast components

```
# Extract and compute contrast table values
ssContrast <- contrastTest$t.ratio**2 * msError
dfContrast <- 1
msContrast <- ssContrast / dfContrast
fContrast <- contrastTest$t.ratio**2
pContrast <- contrastTest$p.value
```

Compute residual components

```
ssResidBtwn <- ssFactor - ssContrast
dfResidBtwn <- nlevels(dfE1noDM$interaction) - 2
msResidBtwn <- ssResidBtwn / dfResidBtwn
fStatBtwn <- msResidBtwn / msError
pValBtwn <- pf(fStatBtwn, dfResidBtwn, dfError, lower.tail = FALSE)
```

Assemble summary table

```
contrastSummaryTable <- tibble(
  term = c("Planned Contrast", "Residual Between", "Error"),
  sumSq = c(ssContrast, ssResidBtwn, ssError),
  df = c(dfContrast, dfResidBtwn, dfError),
  meanSq = c(msContrast, msResidBtwn, msError),
  fStatistic = c(fContrast, fStatBtwn, NA),
  pValue = c(pContrast, pValBtwn, NA)
)
```

```
as.data.frame(contrastSummaryTable)
```

	term	sumSq	df	meanSq	fStatistic	pValue
1	Planned Contrast	2248.7742	1	2248.7742	4.067968	0.04526818
2	Residual Between	588.8528	2	294.4264	0.532609	0.58804197
3	Error	94528.8529	171	552.8003	NA	NA

Planned contrast q-squared

Appendix A of Guggenmos, Piercey, and Agolia (2018) provides a formula for r when cell sizes are unequal:

$$r = \sqrt{\frac{F_{\text{unequal } n_i}}{F_{\text{unequal } n_i} + F_{BCRV} \times df}}$$

where $F_{\text{unequal } n_i} = F_{\text{contrast}} \cdot \frac{\sum_i w_i^2}{\sum_i \frac{w_i^2}{n_i}} \cdot \frac{1}{n_i}$ and df = the number of levels in the contrast interaction minus 2.

Because our cell sizes are unequal, we use this formula to compute r . From there, we compute q^2 as $1 - r^2$.

```

weights <- unlist(weights)

numerator <- sum(weights**2)

cellSizes <- aggregate(dvRisk ~ Linking *
  ↳ Labeling, data=dfE1noDM, FUN=length)$dvRisk

denominator <- sum(weights**2/cellSizes)

oneOverAvgN <- 1/mean(cellSizes)

fUnequal <- fContrast * numerator/denominator * oneOverAvgN

rVal <- sqrt(fUnequal / (fUnequal + fStatBtwn * dfResidBtwn))

rVal

```

```
[1] 0.8843674
```

```
rVal**2
```

```
[1] 0.7821056
```

```
1 - rVal**2
```

```
[1] 0.2178944
```

Simple effects

```
emmeans(anovaE1risk, pairwise ~ Labeling | Linking)
```

```
$emmeans
```

```
Linking = unlinked:
```

Labeling	emmean	SE	df	lower.CL	upper.CL
balanceSheet	48.9	3.43	171	42.1	55.6
incomeStmnt	48.4	3.63	171	41.2	55.5

Linking = linked:

Labeling	emmean	SE	df	lower.CL	upper.CL
balanceSheet	44.3	3.47	171	37.5	51.1
incomeStmt	55.7	3.72	171	48.4	63.1

Confidence level used: 0.95

\$contrasts

Linking = unlinked:

contrast	estimate	SE	df	t.ratio	p.value
balanceSheet - incomeStmt	0.491	4.99	171	0.098	0.9217

Linking = linked:

contrast	estimate	SE	df	t.ratio	p.value
balanceSheet - incomeStmt	-11.421	5.08	171	-2.247	0.0259

```
emmeans(anovaElrisk, pairwise ~ Linking | Labeling)
```

\$emmeans

Labeling = balanceSheet:

Linking	emmean	SE	df	lower.CL	upper.CL
unlinked	48.9	3.43	171	42.1	55.6
linked	44.3	3.47	171	37.5	51.1

Labeling = incomeStmt:

Linking	emmean	SE	df	lower.CL	upper.CL
unlinked	48.4	3.63	171	41.2	55.5
linked	55.7	3.72	171	48.4	63.1

Confidence level used: 0.95

\$contrasts

Labeling = balanceSheet:

contrast	estimate	SE	df	t.ratio	p.value
unlinked - linked	4.57	4.88	171	0.937	0.3502

Labeling = incomeStmt:

contrast	estimate	SE	df	t.ratio	p.value
unlinked - linked	-7.34	5.19	171	-1.414	0.1592

Hypothesis 2

This compares conditions 4 and 5 (theoretically-motivated Presentation and direct method) using Mean Square Error (MSE) as the denominator (*i.e.*, the pooled within-group variance across all conditions). This is appropriate because using the full error term produces a more accurate estimate of population variance.

```
anovaE1riskDM <- lm(dvRisk ~ interaction, data = dfE1)
emMeansE1riskDM <- emmeans(anovaE1riskDM, ~ interaction)
customContrastE1riskDM <- list(
  levels = c(0,0,0,1,-1)
)
contrast(emMeansE1riskDM, customContrastE1riskDM)
```

contrast	estimate	SE	df	t.ratio	p.value
levels	6.99	5.29	212	1.322	0.1877

Table 2

For the statistical analysis of the binary (0/1) `Understand` variable, we used SAS's `proc catmod` function. Please refer to authors' SAS files for documentation of this analysis.

Table 3

Descriptive statistics

```
as.data.frame(
  dfE1 %>%
  group_by(Linking, Labeling) %>%
  summarize(
    mean = mean(dvInvestment),
    sd = sd(dvInvestment),
    n = n(),
    .groups = "drop"
  )
)
```

	Linking	Labeling	mean	sd	n
1	unlinked	balanceSheet	47.76596	22.71009	47
2	unlinked	incomeStmt	45.97619	24.49240	42
3	linked	balanceSheet	50.58696	26.36882	46
4	linked	incomeStmt	37.25000	22.37644	40
5	directMethod	directMethod	41.42857	26.42444	42

ANOVA

```
anovaE1invest <- lm(dvInvestment ~ Linking * Labeling, data = subset(dfE1,
  ↳ Linking != "directMethod"), contrasts = list(Linking = contr.sum,
  ↳ Labeling = contr.sum))
aovE1invest <- as.data.frame(Anova(anovaE1invest, type = "III"))
aovE1invest
```

	Sum Sq	Df	F value	Pr(>F)
(Intercept)	359062.144	1	619.3470920	9.880957e-59
Linking	379.757	1	0.6550438	4.194399e-01
Labeling	2491.883	1	4.2982542	3.965106e-02
Linking:Labeling	1452.080	1	2.5046956	1.153552e-01
Residuals	99136.054	171	NA	NA

Contrast

The following is a manual replication of the analysis that the authors performed in CompTrasts (<https://choicelab.shinyapps.io/ContrastAppShinyIO/>).

Planned contrast variance table

Fit model and run contrast test

```
# Planned contrast
model <- lm(dvInvestment ~ 0 + interaction, data = dfE1noDM)
means <- emmeans(model, ~ interaction)
weights <- list(c(1,1,1,-3))
contrastTest <- summary(contrast(means,weights))

# Residual interaction
residualAnova <- anova(lm(dvInvestment ~ interaction, data = dfE1noDM))
```

Extract residual ANOVA values

```
# Extract table values from residual interaction ANOVA
ssFactor <- residualAnova["interaction", "Sum Sq"]
ssError <- residualAnova$`Sum Sq`[2]
dfError <- residualAnova$Df[2]
msError <- residualAnova$`Mean Sq`[2]
```

Extract and compute planned contrast components

```
# Compute contrast table values
ssContrast <- contrastTest$t.ratio**2 * msError
dfContrast <- 1
msContrast <- ssContrast / dfContrast
fContrast <- contrastTest$t.ratio**2
pContrast <- contrastTest$p.value
```

Compute residual components

```
ssResidBtwn <- ssFactor - ssContrast
dfResidBtwn <- nlevels(dfE1noDM$interaction) - 2
msResidBtwn <- ssResidBtwn / dfResidBtwn
fStatBtwn <- msResidBtwn / msError
pValBtwn <- pf(fStatBtwn, dfResidBtwn, dfError, lower.tail = FALSE)
```

Assemble summary table

```
contrastSummaryTable <- tibble(
  term = c("Planned Contrast", "Residual Between", "Error"),
  sumSq = c(ssContrast, ssResidBtwn, ssError),
  df = c(dfContrast, dfResidBtwn, dfError),
  meanSq = c(msContrast, msResidBtwn, msError),
  fStatistic = c(fContrast, fStatBtwn, NA),
  pValue = c(pContrast, pValBtwn, NA)
)
```

```
as.data.frame(contrastSummaryTable)
```

	term	sumSq	df	meanSq	fStatistic	pValue
1	Planned Contrast	3637.1015	1	3637.1015	6.2736444	0.0131906
2	Residual Between	521.2789	2	260.6395	0.4495776	0.6386494
3	Error	99136.0539	171	579.7430	NA	NA

Planned contrast q-squared

Appendix A of Guggenmos, Piercey, and Agoglia (2018) provides a formula for r when cell sizes are unequal:

$$r = \sqrt{\frac{F_{\text{unequal } n_i}}{F_{\text{unequal } n_i} + F_{BCRV} \times df}}$$

where $F_{\text{unequal } n_i} = F_{\text{contrast}} \cdot \frac{\sum_i w_i^2}{\sum_i \frac{w_i^2}{n_i}} \cdot \frac{1}{n_i}$ and df = the number of levels in the contrast interaction minus 2.

Because our cell sizes are unequal, we use this formula to compute r . From there, we compute q^2 as $1 - r^2$.

```
weights <- unlist(weights)

numerator <- sum(weights**2)

cellSizes <- aggregate(dvInvestment ~ Linking *
  ↪ Labeling, data=dfE1noDM, FUN=length)$dvInvestment

denominator <- sum(weights**2/cellSizes)

oneOverAvgN <- 1/mean(cellSizes)

fUnequal <- fContrast * numerator/denominator * oneOverAvgN

rVal <- sqrt(fUnequal / (fUnequal + fStatBtwn * dfResidBtwn))

rVal
```

```
[1] 0.9314979
```

```
rVal**2
```

```
[1] 0.8676883
```

```
1 - rVal**2
```

```
[1] 0.1323117
```

Simple effects

```
emmeans(anovaE1invest, pairwise ~ Labeling | Linking)
```

```
$emmeans
```

```
Linking = unlinked:
```

Labeling	emmean	SE	df	lower.CL	upper.CL
balanceSheet	47.8	3.51	171	40.8	54.7
incomeStmt	46.0	3.72	171	38.6	53.3

```
Linking = linked:
```

Labeling	emmean	SE	df	lower.CL	upper.CL
balanceSheet	50.6	3.55	171	43.6	57.6
incomeStmt	37.2	3.81	171	29.7	44.8

```
Confidence level used: 0.95
```

```
$contrasts
```

```
Linking = unlinked:
```

contrast	estimate	SE	df	t.ratio	p.value
balanceSheet - incomeStmt	1.79	5.11	171	0.350	0.7267

```
Linking = linked:
```

contrast	estimate	SE	df	t.ratio	p.value
balanceSheet - incomeStmt	13.34	5.21	171	2.562	0.0113

```
emmeans(anovaE1invest, pairwise ~ Linking | Labeling)
```

```
$emmeans
```

```
Labeling = balanceSheet:
```

Linking	emmean	SE	df	lower.CL	upper.CL
unlinked	47.8	3.51	171	40.8	54.7
linked	50.6	3.55	171	43.6	57.6

```
Labeling = incomeStmt:
  Linking  emmean  SE  df lower.CL upper.CL
unlinked  46.0  3.72 171    38.6    53.3
linked    37.2  3.81 171    29.7    44.8
```

Confidence level used: 0.95

\$contrasts

```
Labeling = balanceSheet:
```

```
contrast      estimate    SE  df t.ratio p.value
unlinked - linked    -2.82 4.99 171  -0.565  0.5729
```

```
Labeling = incomeStmt:
```

```
contrast      estimate    SE  df t.ratio p.value
unlinked - linked     8.73 5.32 171   1.640  0.1028
```

Hypothesis 2

This compares conditions 4 and 5 (theoretically-motivated Presentation and direct method) using Mean Square Error (MSE) as the denominator (*i.e.*, the pooled within-group variance across all conditions). This is appropriate because using the full error term produces a more accurate estimate of population variance.

```
anovaE1investDM <- lm(dvInvestment ~ interaction, data = dfE1)
emMeansE1investDM <- emmeans(anovaE1investDM, ~ interaction)
customContrastE1investDM <- list(
  levels = c(0,0,0,-1,1)
)
contrast(emMeansE1investDM, customContrastE1investDM)
```

```
contrast estimate    SE  df t.ratio p.value
levels      4.18 5.42 212   0.770  0.4419
```

Table 4

Descriptive statistics

```
as.data.frame(
  dfE2 %>%
  group_by(Presentation, liquidityRisk) %>%
  summarize(
    mean = mean(dvRisk),
    sd = sd(dvRisk),
    n = n(),
    .groups = "drop"
  )
)
```

	Presentation	liquidityRisk	mean	sd	n
1	indirectMethod	lowerRisk	35.66667	21.94782	51
2	indirectMethod	higherRisk	41.53659	24.89588	41
3	theoreticalPresentation	lowerRisk	43.23913	25.00150	46
4	theoreticalPresentation	higherRisk	57.83333	26.77906	42

ANOVA

```
anovaE2 <- lm(dvRisk ~ Presentation * liquidityRisk, data = dfE2, contrasts =
  ↪ list(Presentation = contr.sum, liquidityRisk = contr.sum))
aovE2 <- as.data.frame(Anova(anovaE2, type = "III"))
aovE2
```

	Sum Sq	Df	F value	Pr(>F)
(Intercept)	354923.3661	1	587.048626	5.960701e-58
Presentation	6362.4791	1	10.523637	1.410387e-03
liquidityRisk	4676.6668	1	7.735278	6.005282e-03
Presentation:liquidityRisk	849.9822	1	1.405883	2.373393e-01
Residuals	106407.7314	176	NA	NA

Contrast

The following is a manual replication of the analysis that the authors performed in CompTrasts (<https://choicelab.shinyapps.io/ContrastAppShinyIO/>).

Planned contrast variance table

Fit model and run contrast test

```
# Planned contrast
model <- lm(dvRisk ~ 0 + interaction, data = dfE2)
means <- emmeans(model, ~ interaction)
weights <- list(c(-1,-1,-1,3))
contrastTest <- summary(contrast(means,weights))

# Residual interaction
residualAnova <- anova(lm(dvRisk ~ interaction, data = dfE2))
```

Extract residual ANOVA values

```
# Extract table values from residual interaction ANOVA
ssFactor <- residualAnova["interaction", "Sum Sq"]
ssError <- residualAnova$`Sum Sq`[2]
dfError <- residualAnova$Df[2]
msError <- residualAnova$`Mean Sq`[2]
```

Extract and compute planned contrast components

```
# Compute contrast table values
ssContrast <- contrastTest$t.ratio**2 * msError
dfContrast <- 1
msContrast <- ssContrast / dfContrast
fContrast <- contrastTest$t.ratio**2
pContrast <- contrastTest$p.value
```

Compute residual components

```
ssResidBtwn <- ssFactor - ssContrast
dfResidBtwn <- nlevels(dfE1noDM$interaction) - 2
msResidBtwn <- ssResidBtwn / dfResidBtwn
fStatBtwn <- msResidBtwn / msError
pValBtwn <- pf(fStatBtwn, dfResidBtwn, dfError, lower.tail = FALSE)
```

Assemble summary table

```
contrastSummaryTable <- tibble(
  term = c("Planned Contrast", "Residual Between", "Error"),
  sumSq = c(ssContrast, ssResidBtwn, ssError),
  df = c(dfContrast, dfResidBtwn, dfError),
  meanSq = c(msContrast, msResidBtwn, msError),
  fStatistic = c(fContrast, fStatBtwn, NA),
  pValue = c(pContrast, pValBtwn, NA)
)
```

```
as.data.frame(contrastSummaryTable)
```

	term	sumSq	df	meanSq	fStatistic	pValue
1	Planned Contrast	10053.144	1	10053.1437	16.628052	6.878274e-05
2	Residual Between	1798.903	2	899.4514	1.487706	2.287168e-01
3	Error	106407.731	176	604.5894	NA	NA

Planned contrast q-squared

Appendix A of Guggenmos, Piercey, and Agoglia (2018) provides a formula for r when cell sizes are unequal:

$$r = \sqrt{\frac{F_{\text{unequal } n_i}}{F_{\text{unequal } n_i} + F_{BCRV} \times df}}$$

where $F_{\text{unequal } n_i} = F_{\text{contrast}} \cdot \frac{\sum_i w_i^2}{\sum_i \frac{w_i^2}{n_i}} \cdot \frac{1}{n_i}$ and df = the number of levels in the contrast interaction minus 2.

Because our cell sizes are unequal, we use this formula to compute r . From there, we compute q^2 as $1 - r^2$.

```
weights <- unlist(weights)

numerator <- sum(weights**2)

cellSizes <- aggregate(dvRisk ~ Presentation *
  ↪ liquidityRisk, data=dfE2, FUN=length)$dvRisk

denominator <- sum(weights**2/cellSizes)

oneOverAvgN <- 1/mean(cellSizes)
```

```
fUnequal <- fContrast * numerator/denominator * oneOverAvgN
rVal <- sqrt(fUnequal / (fUnequal + fStatBtwn * dfResidBtwn))
rVal
```

```
[1] 0.9175078
```

```
rVal**2
```

```
[1] 0.8418205
```

```
1 - rVal**2
```

```
[1] 0.1581795
```

Simple effects

```
emmeans(anovaE2, pairwise ~ Presentation | liquidityRisk)
```

```
$emmeans
```

```
liquidityRisk = lowerRisk:
```

Presentation	emmean	SE	df	lower.CL	upper.CL
indirectMethod	35.7	3.44	176	28.9	42.5
theoreticalPresentation	43.2	3.63	176	36.1	50.4

```
liquidityRisk = higherRisk:
```

Presentation	emmean	SE	df	lower.CL	upper.CL
indirectMethod	41.5	3.84	176	34.0	49.1
theoreticalPresentation	57.8	3.79	176	50.3	65.3

```
Confidence level used: 0.95
```

```
$contrasts
```

```
liquidityRisk = lowerRisk:
```

contrast	estimate	SE	df	t.ratio	p.value
indirectMethod - theoreticalPresentation	-7.57	5.0	176	-1.515	0.1317

```
liquidityRisk = higherRisk:
  contrast                estimate SE df t.ratio p.value
indirectMethod - theoreticalPresentation -16.30 5.4 176 -3.019 0.0029
```

```
emmeans(anovaE2, pairwise ~ liquidityRisk | Presentation)
```

```
$emmeans
```

```
Presentation = indirectMethod:
```

liquidityRisk	emmean	SE	df	lower.CL	upper.CL
lowerRisk	35.7	3.44	176	28.9	42.5
higherRisk	41.5	3.84	176	34.0	49.1

```
Presentation = theoreticalPresentation:
```

liquidityRisk	emmean	SE	df	lower.CL	upper.CL
lowerRisk	43.2	3.63	176	36.1	50.4
higherRisk	57.8	3.79	176	50.3	65.3

Confidence level used: 0.95

```
$contrasts
```

```
Presentation = indirectMethod:
```

contrast	estimate	SE	df	t.ratio	p.value
lowerRisk - higherRisk	-5.87	5.16	176	-1.138	0.2566

```
Presentation = theoreticalPresentation:
```

contrast	estimate	SE	df	t.ratio	p.value
lowerRisk - higherRisk	-14.59	5.25	176	-2.781	0.0060

Table 5

Descriptive statistics

```
as.data.frame(
  dfE3 %>%
  group_by(Presentation, Order) %>%
  summarize(
    mean = mean(dvRisk),
    sd = sd(dvRisk),
    n = n(),
```

```

    .groups = "drop"
  )
)

```

	Presentation	Order	mean	sd	n
1	indirectMethod	arFirst	57.13043	26.18345	23
2	indirectMethod	riskFirst	33.27273	12.03638	22
3	theoreticalPresentation	arFirst	69.91667	19.99330	24
4	theoreticalPresentation	riskFirst	66.75000	19.17029	24

ANOVA

```

anovaE3 <- lm(dvRisk ~ Presentation * Order, data = dfE3, contrasts =
  ↪ list(Presentation = contr.sum, Order = contr.sum))
aovE3 <- as.data.frame(Anova(anovaE3, type = "III"))
aovE3

```

	Sum Sq	Df	F value	Pr(>F)
(Intercept)	299308.426	1	744.687660	5.166262e-45
Presentation	12424.449	1	30.912373	2.792451e-07
Order	4239.468	1	10.547914	1.641225e-03
Presentation:Order	2485.220	1	6.183295	1.476195e-02
Residuals	35771.306	89	NA	NA

Simple effects

```

emmeans(anovaE3, pairwise ~ Presentation | Order)

```

\$emmeans

Order = arFirst:

Presentation	emmean	SE	df	lower.CL	upper.CL
indirectMethod	57.1	4.18	89	48.8	65.4
theoreticalPresentation	69.9	4.09	89	61.8	78.0

Order = riskFirst:

Presentation	emmean	SE	df	lower.CL	upper.CL
--------------	--------	----	----	----------	----------

indirectMethod	33.3	4.27	89	24.8	41.8
theoreticalPresentation	66.8	4.09	89	58.6	74.9

Confidence level used: 0.95

\$contrasts

Order = arFirst:

contrast	estimate	SE	df	t.ratio	p.value
indirectMethod - theoreticalPresentation	-12.8	5.85	89	-2.186	0.0315

Order = riskFirst:

contrast	estimate	SE	df	t.ratio	p.value
indirectMethod - theoreticalPresentation	-33.5	5.92	89	-5.657	<0.0001

```
emmeans(anovaE3, pairwise ~ Order | Presentation)
```

\$emmeans

Presentation = indirectMethod:

Order	emmean	SE	df	lower.CL	upper.CL
arFirst	57.1	4.18	89	48.8	65.4
riskFirst	33.3	4.27	89	24.8	41.8

Presentation = theoreticalPresentation:

Order	emmean	SE	df	lower.CL	upper.CL
arFirst	69.9	4.09	89	61.8	78.0
riskFirst	66.8	4.09	89	58.6	74.9

Confidence level used: 0.95

\$contrasts

Presentation = indirectMethod:

contrast	estimate	SE	df	t.ratio	p.value
arFirst - riskFirst	23.86	5.98	89	3.990	0.0001

Presentation = theoreticalPresentation:

contrast	estimate	SE	df	t.ratio	p.value
arFirst - riskFirst	3.17	5.79	89	0.547	0.5856