For_Jim_toPDF

rwoo

19/08/2022

Data Cleaning and Formatting

Import data and clean from after intervention

```
setwd("C:/Users/rwoo/Documents/Thesis_Data_Analysis")
raw_post <- read.csv("Post.csv", stringsAsFactors = FALSE)
colnames(raw_post) <- c("Timestamp", "PID", "Desirability_List", "CC1_post", "CC2_post", "CC3_post", "CC
raw_post$PID = toupper(raw_post$PID) #Change all to uppercase
good_PID <- c("P21", "P3", "P4", "P64", "P9", "P68", "P66", "P60", "P70", "P81", "P69", "P72", "P71", "P
```

Import data and clean from before intervention. I make column names etc.

```
raw_pre <- read.csv("Pre.csv", stringsAsFactors = FALSE)
colnames(raw_pre) <- c("Timestamp", "PID", "Gender", "Sex", "Age", "Education", "College_Or_Uni", "Wate
#Only take good PID from pre
pre_Filtered <- raw_pre[which(raw_pre$PID %in% good_PID), ] #NOT WORK, only gives 16
```

Community Connectedness Analysis

CC1 You feel you are a part of the Waterloo community CC2 Participating in the Waterloo community is a positive thing for you. CC3 You feel a bond with the Waterloo community.
CC4 You are proud of the Waterloo community.
CC5 It is important for you to be aware of issues others face in your community
CC6 I feel aware of issues that others face in my community

Get change in community connectedness

```
#Pre
cc_pre <- pre_Filtered %>% select(PID, CC1_pre, CC2_pre, CC3_pre, CC4_pre, CC5_pre, CC6_pre)
#Post
cc_post <- raw_post %>% select(PID, CC1_post, CC2_post, CC3_post, CC4_post, CC5_post, CC6_post)
#Merge pre and post
cc <- merge(cc_pre, cc_post, by = "PID")
#Get delta cc NOTE: THIS METHOD SEEMS INEFFECIENT INVESTIGATE REGEX
cc <- mutate(cc, CC1_delta = CC1_post - CC1_pre)
cc <- mutate(cc, CC2_delta = CC2_post - CC2_pre)
cc <- mutate(cc, CC3_delta = CC3_post - CC3_pre)
cc <- mutate(cc, CC4_delta = CC4_post - CC4_pre)
cc <- mutate(cc, CC5_delta = CC5_post - CC5_pre)
cc <- mutate(cc, CC6_delta = CC6_post - CC6_pre)
```

```r
#Only take delta columns
cc_delta <- cc %>% select(CC1_delta, CC2_delta, CC3_delta, CC4_delta, CC5_delta, CC6_delta)
summary(cc_delta) #Probably of most note for meeting
```

```
##    CC1_delta          CC2_delta         CC3_delta         CC4_delta      
##  Min.   :-1.00000   Min.   :-2.0000   Min.   :-1.0000   Min.   :-1.0000  
##  1st Qu.: 0.00000   1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.0000  
##  Median : 0.00000   Median : 0.0000   Median : 0.0000   Median : 0.0000  
##  Mean   :-0.05882   Mean   :-0.1176   Mean   : 0.3529   Mean   :-0.1765  
##  3rd Qu.: 0.00000   3rd Qu.: 0.0000   3rd Qu.: 1.0000   3rd Qu.: 0.0000  
##  Max.   : 1.00000   Max.   : 2.0000   Max.   : 2.0000   Max.   : 1.0000  
##    CC5_delta         CC6_delta     
##  Min.   :-1.0000   Min.   :-1.0000  
##  1st Qu.: 0.0000   1st Qu.: 0.0000  
##  Median : 0.0000   Median : 0.0000  
##  Mean   : 0.5294   Mean   : 0.7647  
##  3rd Qu.: 1.0000   3rd Qu.: 2.0000  
##  Max.   : 3.0000   Max.   : 3.0000
```

```r
summary(cc_pre)
```

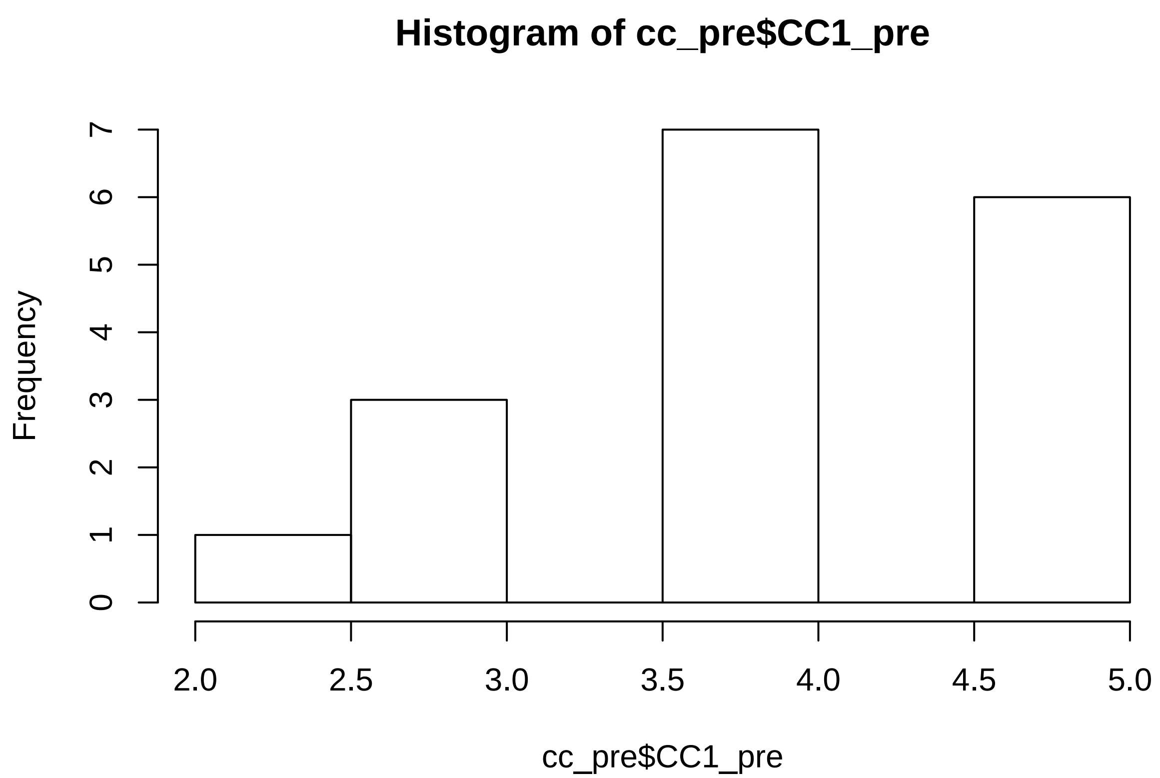
```
##      PID               CC1_pre         CC2_pre         CC3_pre     
##  Length:17          Min.   :2.000   Min.   :2.000   Min.   :1.000  
##  Class :character   1st Qu.:4.000   1st Qu.:4.000   1st Qu.:3.000  
##  Mode  :character   Median :4.000   Median :4.000   Median :4.000  
##                     Mean   :4.059   Mean   :4.294   Mean   :3.706  
##                     3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:4.000  
##                     Max.   :5.000   Max.   :5.000   Max.   :5.000  
##     CC4_pre         CC5_pre     CC6_pre 
##  Min.   :3.000   Min.   :2   Min.   :2  
##  1st Qu.:4.000   1st Qu.:3   1st Qu.:2  
##  Median :5.000   Median :4   Median :3  
##  Mean   :4.471   Mean   :4   Mean   :3  
##  3rd Qu.:5.000   3rd Qu.:5   3rd Qu.:3  
##  Max.   :5.000   Max.   :5   Max.   :5
```

```r
summary(cc_post)
```

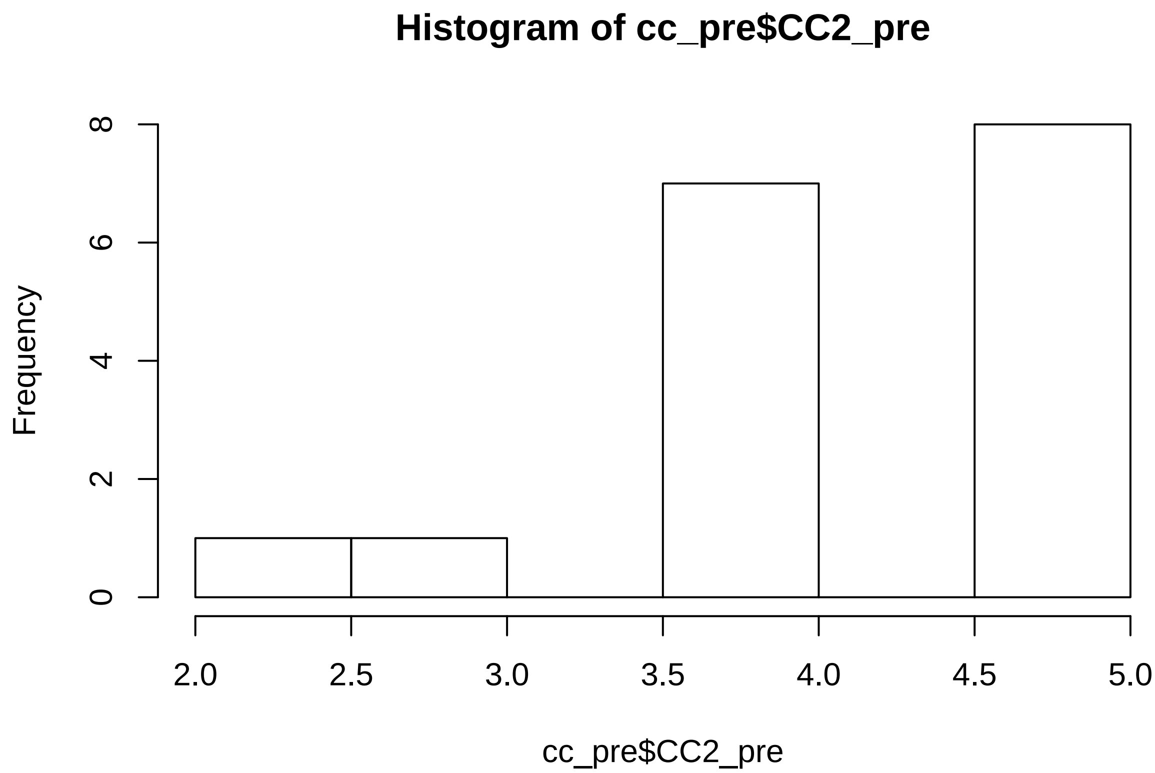
```
##      PID               CC1_post    CC2_post        CC3_post        CC4_post    
##  Length:17          Min.   :3   Min.   :2.000   Min.   :3.000   Min.   :3.000  
##  Class :character   1st Qu.:4   1st Qu.:4.000   1st Qu.:3.000   1st Qu.:4.000  
##  Mode  :character   Median :4   Median :4.000   Median :4.000   Median :4.000  
##                     Mean   :4   Mean   :4.176   Mean   :4.059   Mean   :4.294  
##                     3rd Qu.:4   3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:5.000  
##                     Max.   :5   Max.   :5.000   Max.   :5.000   Max.   :5.000  
##     CC5_post        CC6_post    
##  Min.   :4.000   Min.   :2.000  
##  1st Qu.:4.000   1st Qu.:3.000  
##  Median :5.000   Median :4.000  
##  Mean   :4.529   Mean   :3.765  
##  3rd Qu.:5.000   3rd Qu.:5.000  
##  Max.   :5.000   Max.   :5.000
```

Check assumptions for CC

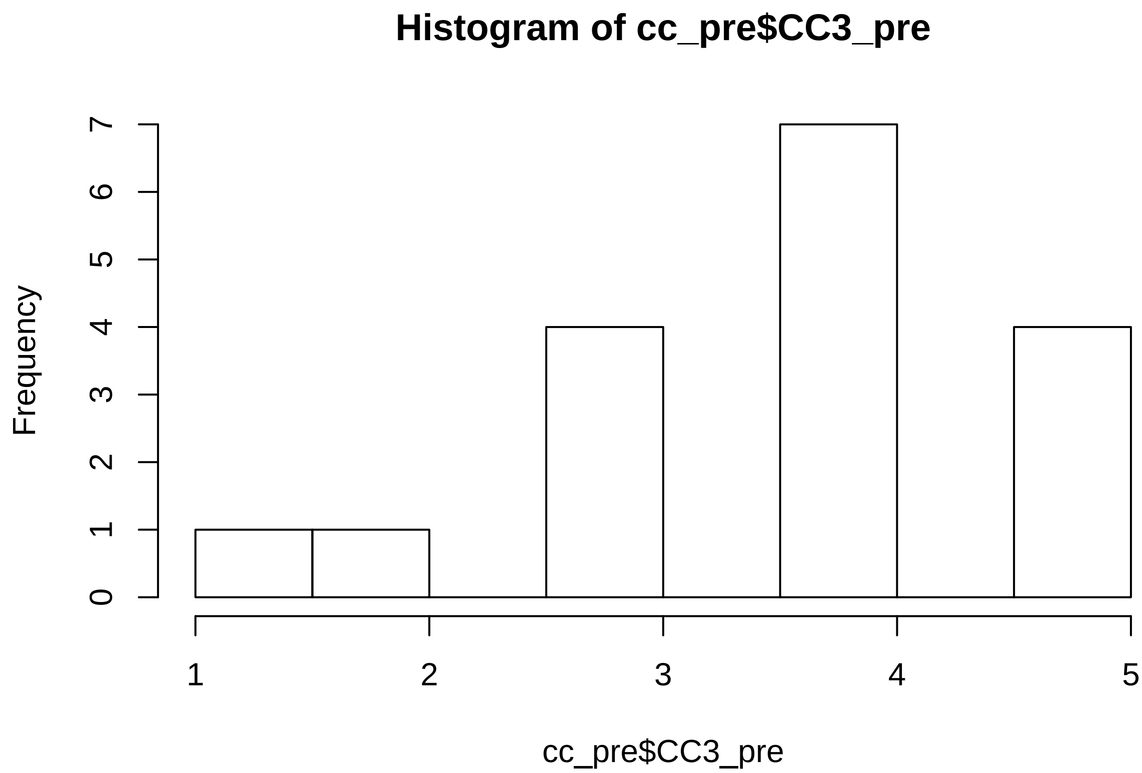
```
#Check shape before
hist(cc_pre$CC1_pre)
```



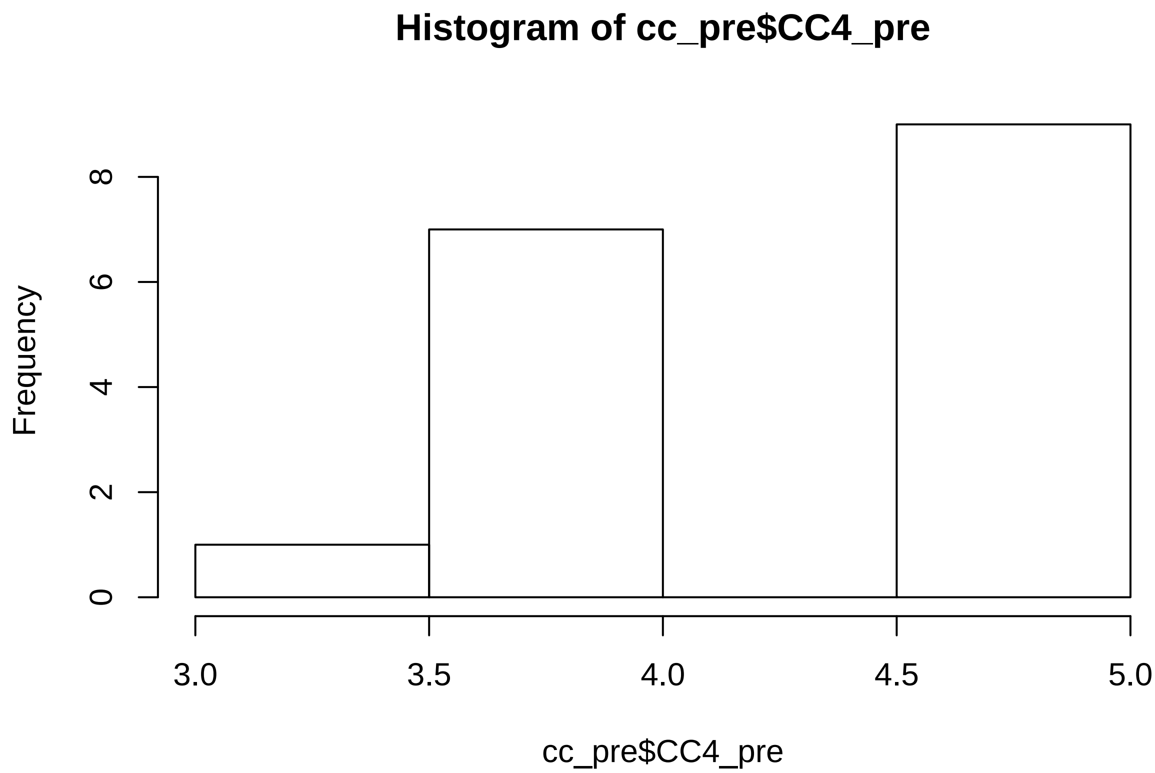

```
hist(cc_pre$CC2_pre)
```



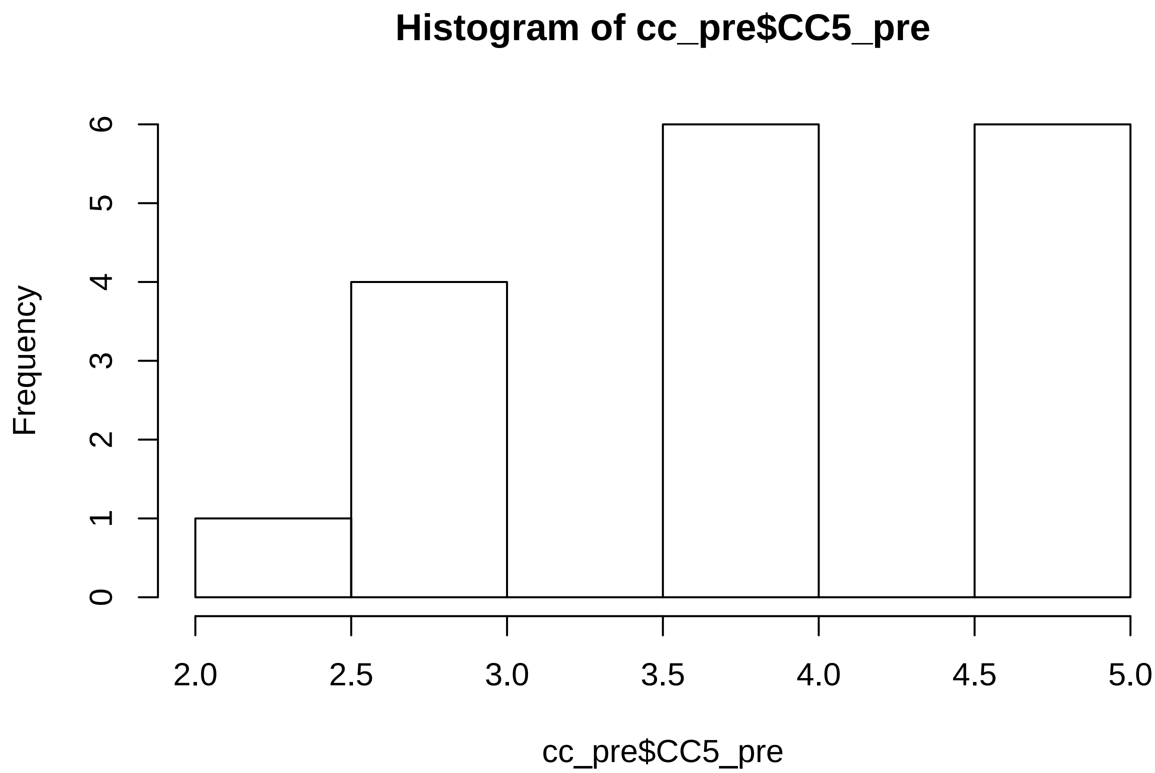

```
hist(cc_pre$CC3_pre)
```



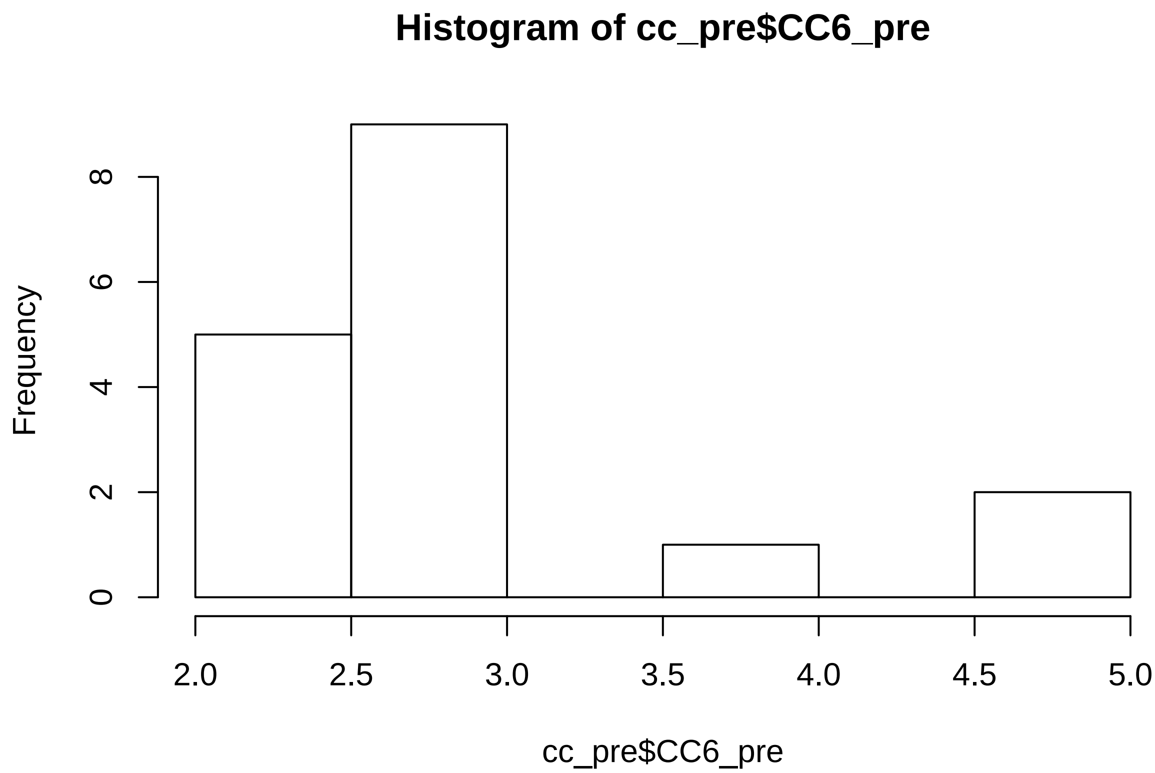

```
hist(cc_pre$CC4_pre)
```



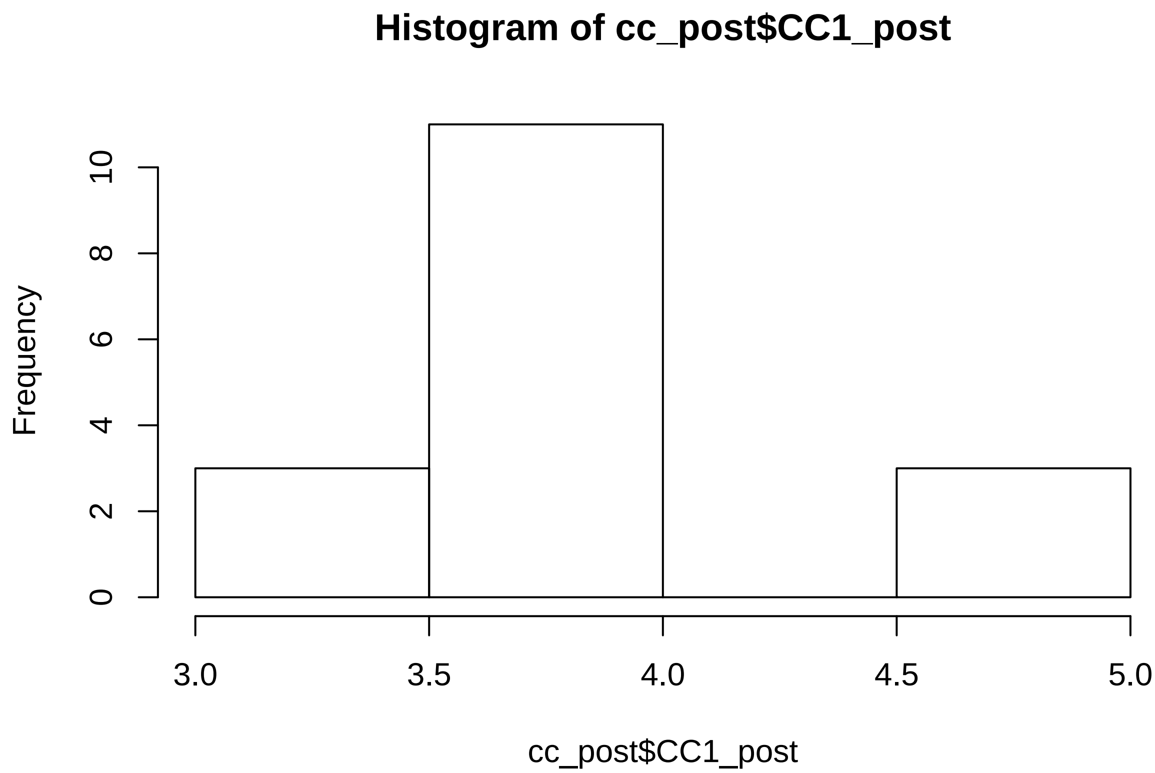

```
hist(cc_pre$CC5_pre)
```



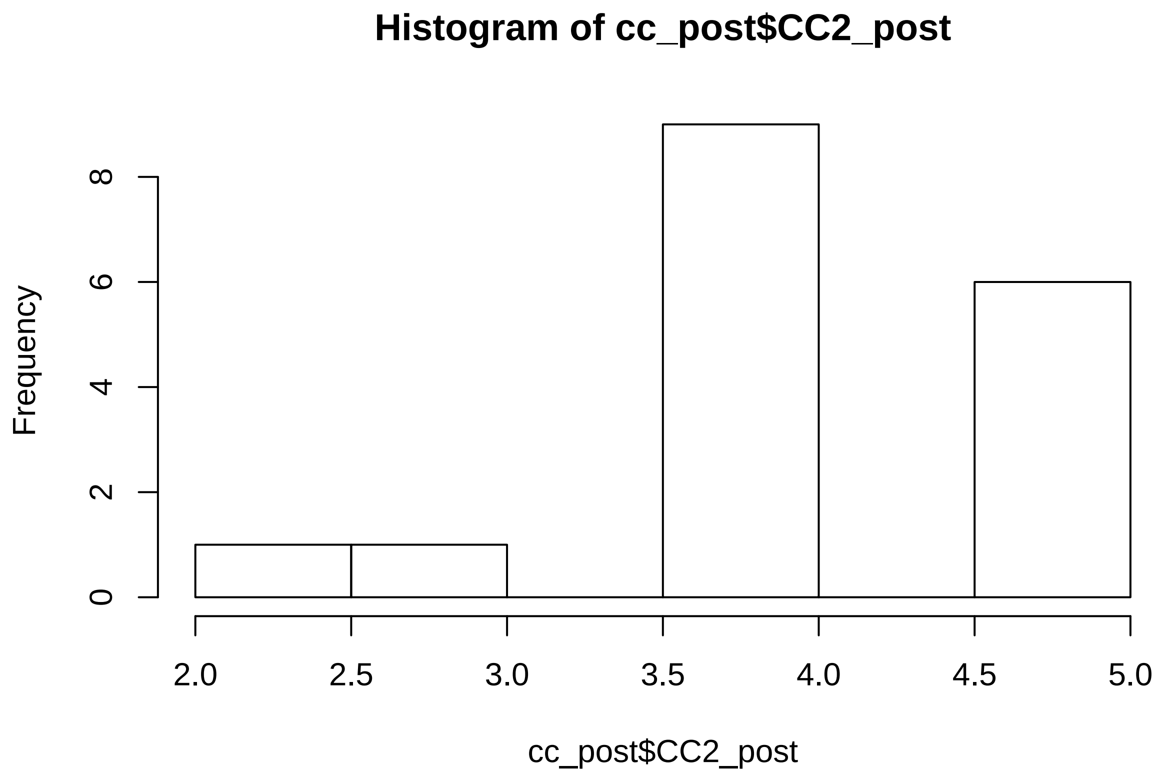

```
hist(cc_pre$CC6_pre)
```



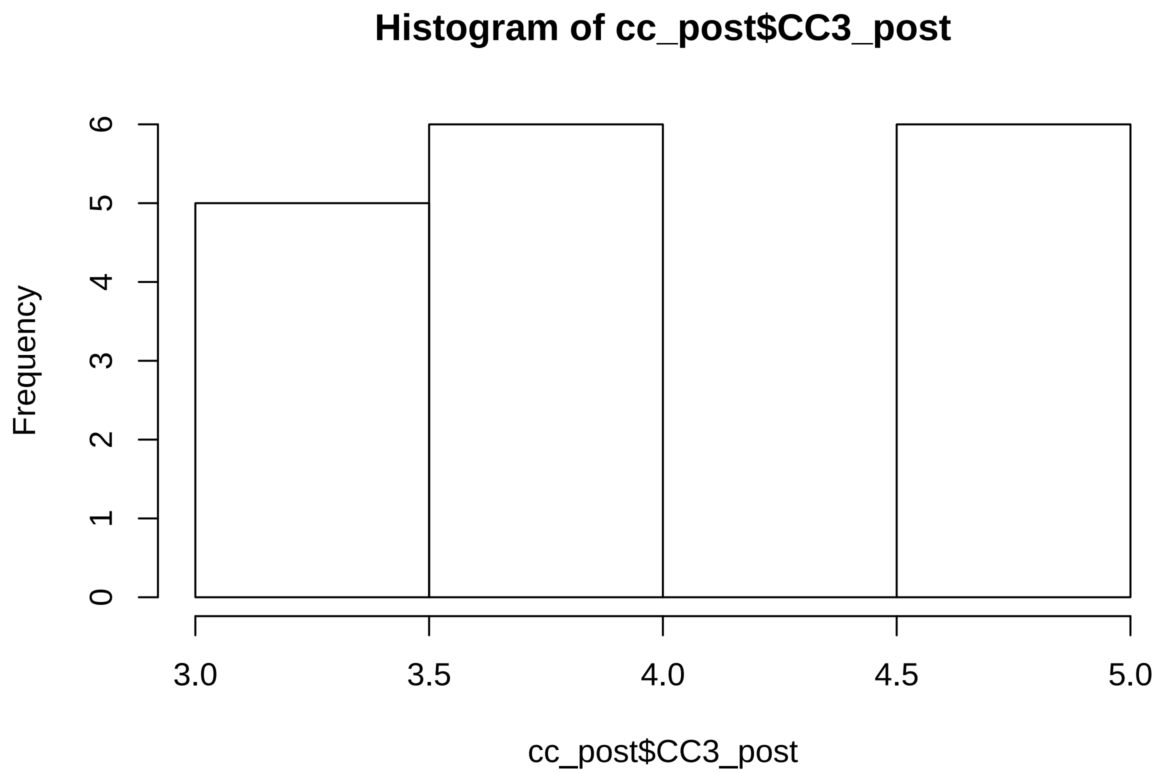

```
#Check shape after
hist(cc_post$CC1_post)
```



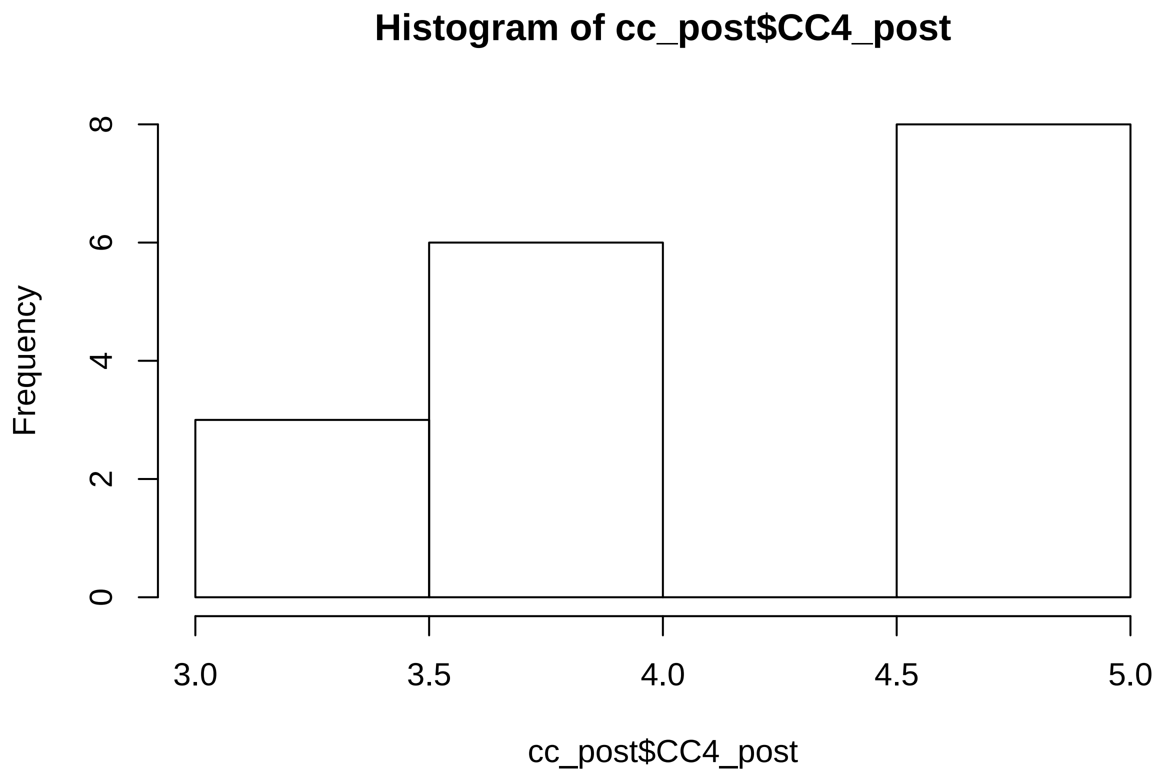

```
hist(cc_post$CC2_post)
```



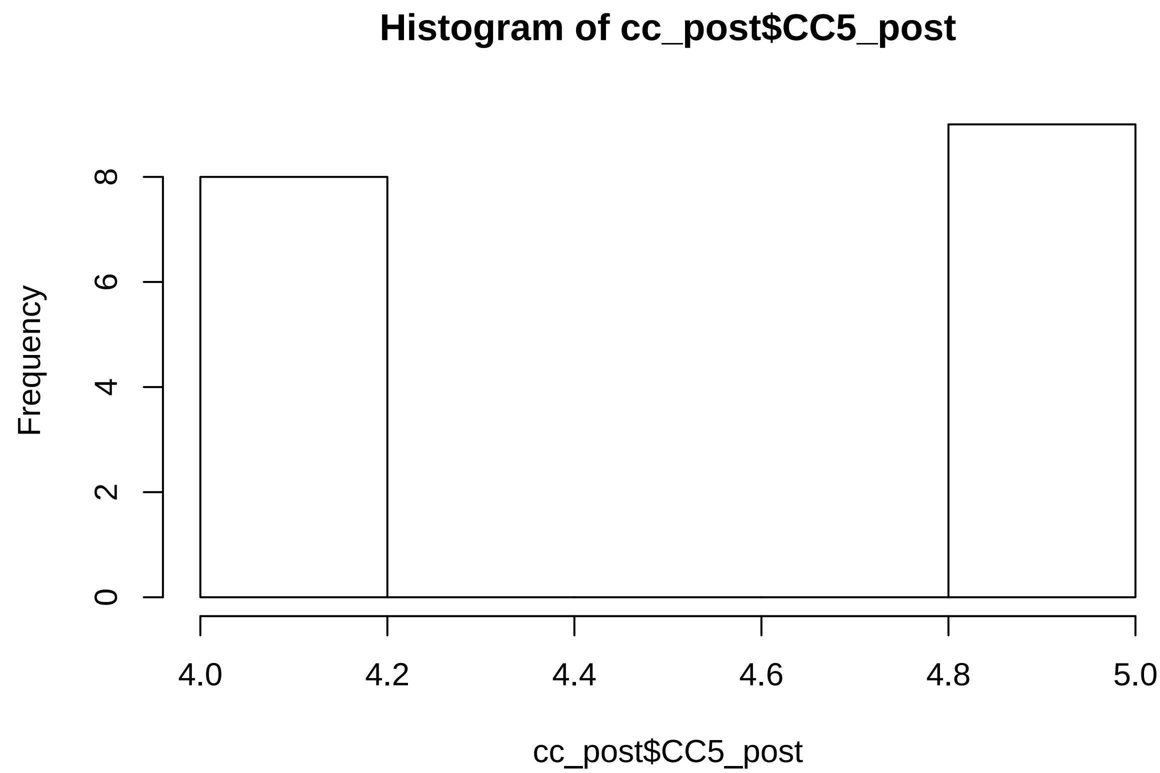

```
hist(cc_post$CC3_post)
```



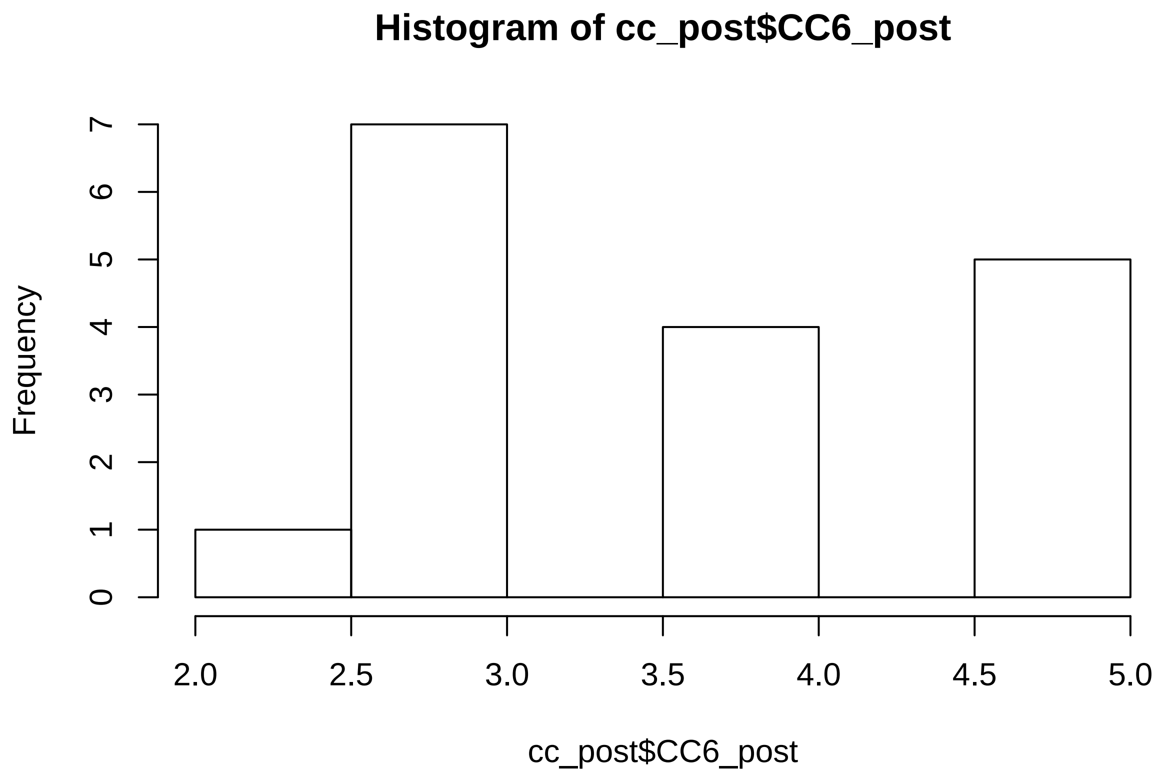

```
hist(cc_post$CC4_post)
```




```
hist(cc_post$CC5_post)
```




```
hist(cc_post$CC6_post)
```




```
#Inconclusive
```

Paired t-test Reading: https://www.jmp.com/en_ca/statistics-knowledge-portal/t-test/paired-t-test.html

```
t.test(cc_pre$CC1_pre, cc_post$CC1_post, paired = TRUE, alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  cc_pre$CC1_pre and cc_post$CC1_post
## t = 0.22291, df = 16, p-value = 0.8264
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5005932  0.6182403
## sample estimates:
## mean of the differences
##              0.05882353
```

```
t.test(cc_pre$CC2_pre, cc_post$CC2_post, paired = TRUE, alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  cc_pre$CC2_pre and cc_post$CC2_post
```

```
## t = 0.46035, df = 16, p-value = 0.6515
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.4241180  0.6594122
## sample estimates:
## mean of the differences
##               0.1176471
```

```r
t.test(cc_pre$CC3_pre, cc_post$CC3_post, paired = TRUE, alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  cc_pre$CC3_pre and cc_post$CC3_post
## t = -1.4606, df = 16, p-value = 0.1635
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.8652000  0.1593176
## sample estimates:
## mean of the differences
##              -0.3529412
```

```r
t.test(cc_pre$CC4_pre, cc_post$CC4_post, paired = TRUE, alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  cc_pre$CC4_pre and cc_post$CC4_post
## t = 0.82416, df = 16, p-value = 0.422
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2774454  0.6303866
## sample estimates:
## mean of the differences
##               0.1764706
```

```r
t.test(cc_pre$CC5_pre, cc_post$CC5_post, paired = TRUE, alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  cc_pre$CC5_pre and cc_post$CC5_post
## t = -2.4962, df = 16, p-value = 0.02386
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.9790251 -0.0797984
## sample estimates:
## mean of the differences
##              -0.5294118
```

```r
t.test(cc_pre$CC6_pre, cc_post$CC6_post, paired = TRUE, alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  cc_pre$CC6_pre and cc_post$CC6_post
## t = -2.6264, df = 16, p-value = 0.01833
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.3819409 -0.1474708
## sample estimates:
## mean of the differences
##              -0.7647059
```

Shapiro-Wilk normality test for the differences. IM CRYING, voilates assumption of normality (prob due to limited scope and small sample size)

```r
d1 <- with(cc, CC1_post - CC1_pre)
shapiro.test(d1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  d1
## W = 0.79848, p-value = 0.001934
```

```r
d2 <- with(cc, CC2_post - CC2_pre)
shapiro.test(d2)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  d2
## W = 0.85696, p-value = 0.01372
```

```r
d3 <- with(cc, CC3_post - CC3_pre)
shapiro.test(d3)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  d3
## W = 0.85594, p-value = 0.01323
```

```r
d4 <- with(cc, CC4_post - CC4_pre)
shapiro.test(d4)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  d4
## W = 0.70335, p-value = 0.0001262
```

```r
d5 <- with(cc, CC5_post - CC5_pre)
shapiro.test(d5)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  d5
## W = 0.88688, p-value = 0.04117
```

```r
d6 <- with(cc, CC6_post - CC6_pre)
shapiro.test(d6)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  d6
## W = 0.89158, p-value = 0.04921
```

Do Wilcoxon (paired) signed-rank test http://www.sthda.com/english/wiki/paired-samples-wilcoxon-test-in-r https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/assumptions-of-the-wilcox-sign-test/

```r
wilcox.test(cc_pre$CC1_pre, cc_post$CC1_post, paired = TRUE, alternative = "two.sided")
```

```
## Warning in wilcox.test.default(cc_pre$CC1_pre, cc_post$CC1_post, paired =
## TRUE, : cannot compute exact p-value with ties

## Warning in wilcox.test.default(cc_pre$CC1_pre, cc_post$CC1_post, paired =
## TRUE, : cannot compute exact p-value with zeroes

##
##  Wilcoxon signed rank test with continuity correction
##
## data:  cc_pre$CC1_pre and cc_post$CC1_post
## V = 30, p-value = 0.8319
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(cc_pre$CC2_pre, cc_post$CC2_post, paired = TRUE, alternative = "two.sided")
```

```
## Warning in wilcox.test.default(cc_pre$CC2_pre, cc_post$CC2_post, paired =
## TRUE, : cannot compute exact p-value with ties

## Warning in wilcox.test.default(cc_pre$CC2_pre, cc_post$CC2_post, paired =
## TRUE, : cannot compute exact p-value with zeroes

##
##  Wilcoxon signed rank test with continuity correction
##
## data:  cc_pre$CC2_pre and cc_post$CC2_post
## V = 26.5, p-value = 0.6675
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(cc_pre$CC3_pre, cc_post$CC3_post, paired = TRUE, alternative = "two.sided")
```

```
## Warning in wilcox.test.default(cc_pre$CC3_pre, cc_post$CC3_post, paired =
## TRUE, : cannot compute exact p-value with ties

## Warning in wilcox.test.default(cc_pre$CC3_pre, cc_post$CC3_post, paired =
## TRUE, : cannot compute exact p-value with zeroes

##
##  Wilcoxon signed rank test with continuity correction
##
## data:  cc_pre$CC3_pre and cc_post$CC3_post
## V = 22.5, p-value = 0.179
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(cc_pre$CC4_pre, cc_post$CC4_post, paired = TRUE, alternative = "two.sided")
```

```
## Warning in wilcox.test.default(cc_pre$CC4_pre, cc_post$CC4_post, paired =
## TRUE, : cannot compute exact p-value with ties

## Warning in wilcox.test.default(cc_pre$CC4_pre, cc_post$CC4_post, paired =
## TRUE, : cannot compute exact p-value with zeroes

##
##  Wilcoxon signed rank test with continuity correction
##
## data:  cc_pre$CC4_pre and cc_post$CC4_post
## V = 35, p-value = 0.4374
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(cc_pre$CC5_pre, cc_post$CC5_post, paired = TRUE, alternative = "two.sided")
```

```
## Warning in wilcox.test.default(cc_pre$CC5_pre, cc_post$CC5_post, paired =
## TRUE, : cannot compute exact p-value with ties

## Warning in wilcox.test.default(cc_pre$CC5_pre, cc_post$CC5_post, paired =
## TRUE, : cannot compute exact p-value with zeroes

##
##  Wilcoxon signed rank test with continuity correction
##
## data:  cc_pre$CC5_pre and cc_post$CC5_post
## V = 10, p-value = 0.03301
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(cc_pre$CC6_pre, cc_post$CC6_post, paired = TRUE, alternative = "two.sided")
```

```
## Warning in wilcox.test.default(cc_pre$CC6_pre, cc_post$CC6_post, paired =
## TRUE, : cannot compute exact p-value with ties
```

```
## Warning in wilcox.test.default(cc_pre$CC6_pre, cc_post$CC6_post, paired =
## TRUE, : cannot compute exact p-value with zeroes

##
##  Wilcoxon signed rank test with continuity correction
##
## data:  cc_pre$CC6_pre and cc_post$CC6_post
## V = 15, p-value = 0.03082
## alternative hypothesis: true location shift is not equal to 0
```

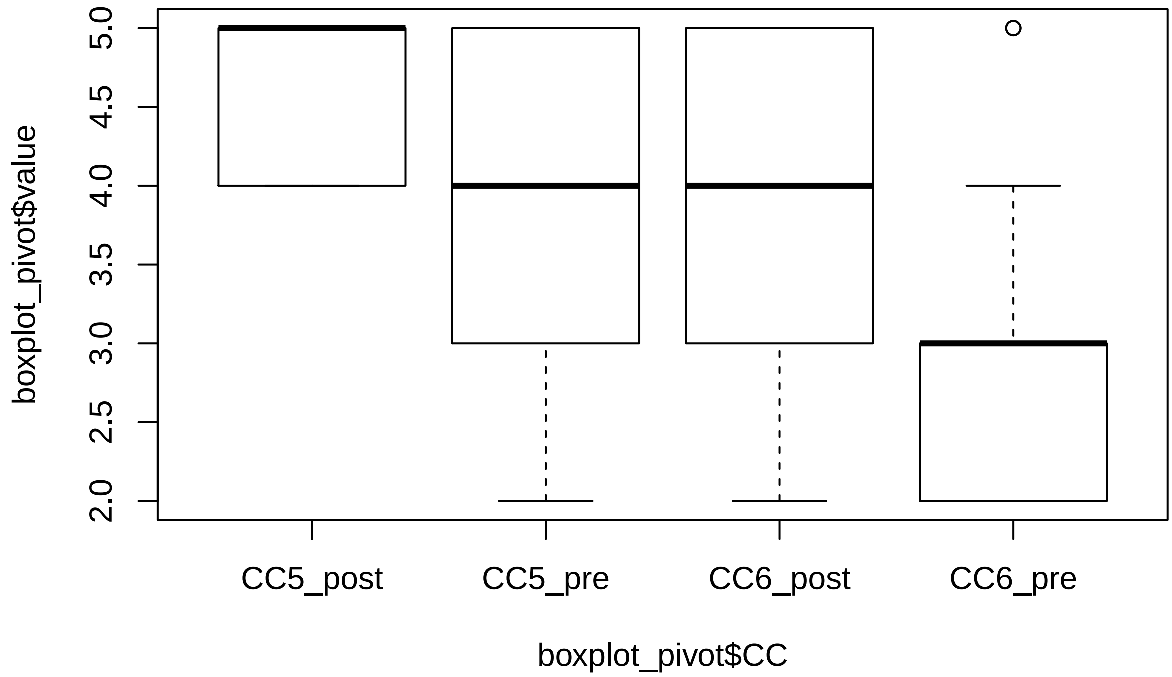
CC total

```
tot_pre <- cc_pre %>%
  mutate(CC_tot_pre = (CC1_pre + CC2_pre + CC3_pre + CC4_pre + CC5_pre + CC6_pre)/6)
tot_post <- cc_post %>%
  mutate(CC_tot_post = (CC1_post + CC2_post + CC3_post + CC4_post + CC5_post + CC6_post)/6)
wilcox.test(tot_pre$CC_tot_pre, tot_post$CC_tot_post, paired = TRUE, alternative = "two.sided")
```

```
## Warning in wilcox.test.default(tot_pre$CC_tot_pre, tot_post$CC_tot_post, :
## cannot compute exact p-value with ties

## Warning in wilcox.test.default(tot_pre$CC_tot_pre, tot_post$CC_tot_post, :
## cannot compute exact p-value with zeroes

##
##  Wilcoxon signed rank test with continuity correction
##
## data:  tot_pre$CC_tot_pre and tot_post$CC_tot_post
## V = 30, p-value = 0.09344
## alternative hypothesis: true location shift is not equal to 0
```

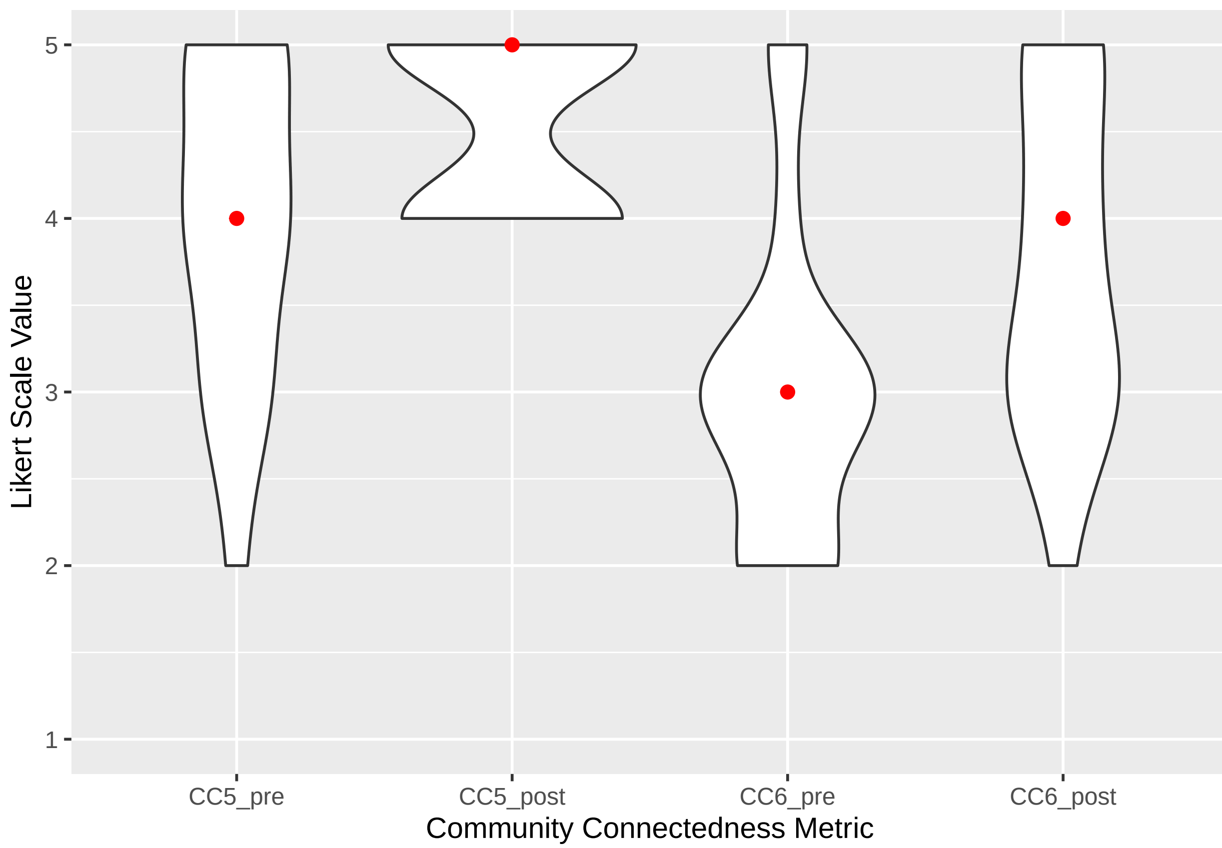
MAKE BOXPLOTS CC

```
cc_sig <- cc %>% select(CC5_pre, CC6_pre, CC5_post, CC6_post)
boxplot_pivot <- cc_sig %>%
  pivot_longer(
    cols = starts_with("CC"),
    names_to = "CC",
    values_to = "value",
    values_drop_na = TRUE
  )

boxplot(boxplot_pivot$value ~ boxplot_pivot$CC)
```



Format and make violin plot

```
#Change order
boxplot_pivot <- boxplot_pivot %>%
  mutate( CC=factor(CC,levels=c("CC5_pre", "CC5_post", "CC6_pre", "CC6_post")) )

# Basic violin plot
p <- ggplot(boxplot_pivot, aes(x=CC, y=value)) +
  geom_violin(position=position_dodge(1)) + coord_cartesian(ylim = c(1, 5)) + xlab("Community Connectedn
# violin plot with median points
p + stat_summary(fun=median, geom="point", size=2, color="red")
```



CC5 and CC6 are significant. Others are not

Relfection Inventory Analysis

RI1 The experience gives me ideas on how to overcome challenges
RI2 I learned from exploring the data
RI3 I enjoyed exploring the data
RI4 I reflected on my own experiences with mental health and accessing resources
RI5 The app would help me discuss mental health and resources with others

```
ri <- raw_post %>% select(PID, RI1, RI2, RI3, RI4, RI5)
summary(ri)
```

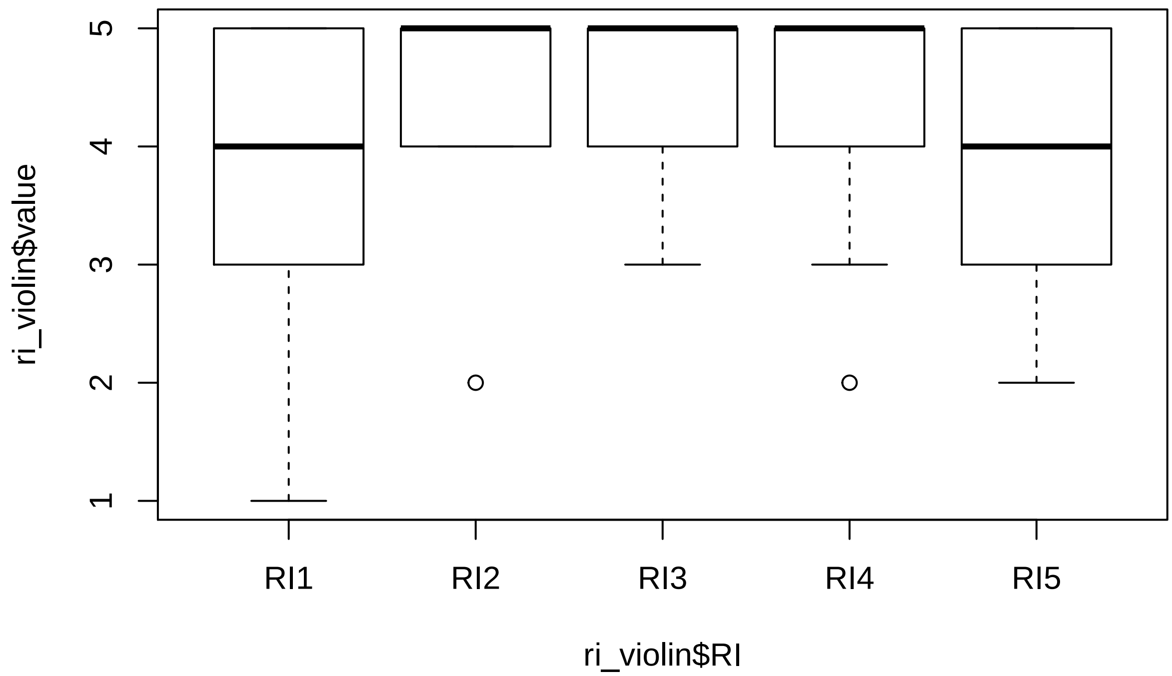
```
##      PID                 RI1             RI2             RI3
##  Length:17          Min.   :1.000   Min.   :2.000   Min.   :3.000
##  Class :character   1st Qu.:3.000   1st Qu.:4.000   1st Qu.:4.000
##  Mode  :character   Median :4.000   Median :5.000   Median :5.000
##                     Mean   :3.647   Mean   :4.471   Mean   :4.588
##                     3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:5.000
##                     Max.   :5.000   Max.   :5.000   Max.   :5.000
##       RI4             RI5
##  Min.   :2.000   Min.   :2.000
##  1st Qu.:4.000   1st Qu.:3.000
##  Median :5.000   Median :4.000
##  Mean   :4.235   Mean   :3.824
```

```
##   3rd Qu.:5.000   3rd Qu.:5.000
##   Max.   :5.000   Max.   :5.000
```

```r
ri
```

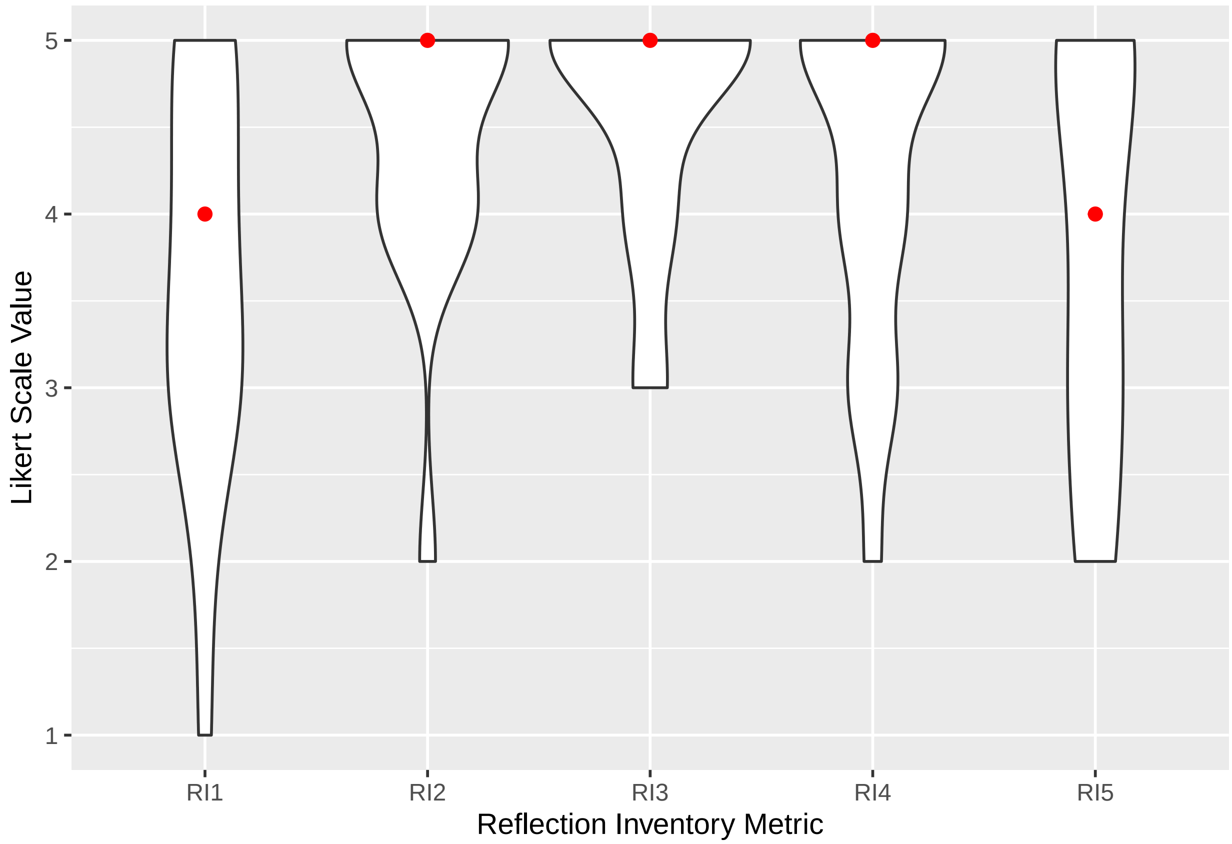
```
##    PID RI1 RI2 RI3 RI4 RI5
## 1   P4   3   4   5   3   4
## 2  P21   1   5   3   4   2
## 3   P3   2   4   4   2   2
## 4  P64   3   4   4   3   3
## 5   P9   5   5   5   5   5
## 6  P68   3   4   3   5   3
## 7  P66   5   5   5   5   5
## 8  P60   4   5   5   5   5
## 9  P70   5   5   5   5   5
## 10 P81   3   4   5   4   3
## 11 P69   4   5   5   4   4
## 12 P72   4   5   5   5   5
## 13 P71   5   5   5   5   5
## 14 P77   4   5   5   5   4
## 15 P75   3   2   4   3   2
## 16 P74   5   4   5   5   3
## 17 P82   3   5   5   4   5
```

```r
ri_sig <- ri %>% select(RI1, RI2, RI3, RI4, RI5)
ri_violin <- ri_sig %>%
  pivot_longer(
    cols = starts_with("RI"),
    names_to = "RI",
    values_to = "value",
    values_drop_na = TRUE
  )

boxplot(ri_violin$value ~ ri_violin$RI)
```



Violin Plot

```
p <- ggplot(ri_violin, aes(x=RI, y=value)) +
  geom_violin(position=position_dodge(1)) + coord_cartesian(ylim = c(1, 5)) + xlab("Reflection Inventory
# violin plot with median points
p + stat_summary(fun=median, geom="point", size=2, color="red")
```



Initial RI analysis shows that most reflection is around a 4 or 5 for both mean and median

Demographics

Parse and clean data

```
full_demo_phone <- pre_Filtered %>% select(PID, Gender, Sex, Age, Education, College_Or_Uni, Waterloo_R
```

In this section I just print all demographics summaries (Data exploration)

```
#Gender
full_demo_phone %>%
  group_by(Gender) %>%
  summarise(gender_count=n())
```

```
## # A tibble: 3 x 2
##   Gender      gender_count
## * <chr>              <int>
## 1 Man                    5
## 2 Non-binary             1
## 3 Woman                 11
```

```
#Sex
full_demo_phone %>%
  group_by(Sex) %>%
  summarise(sex_count=n())
```

```
## # A tibble: 3 x 2
##   Sex                     sex_count
## * <chr>                       <int>
## 1 Female                         11
## 2 Male                            5
## 3 Prefer not to disclose          1
```

```r
#Age
summary(full_demo_phone$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   20.00   23.00   25.00   25.53   26.00   44.00
```

```r
#Education
full_demo_phone %>%
  group_by(Education) %>%
  summarise(education_count=n())
```

```
## # A tibble: 4 x 2
##   Education                      education_count
## * <chr>                                    <int>
## 1 Bachelorâ€s degree                           9
## 2 Graduated high school or less                2
## 3 Post-graduate degree                         3
## 4 Some college, no degree                      3
```

```r
###NOTE: I STRONGLY SUSPECT SOME PEOPLE FILLED THIS IN WRONG
```

```r
#College_Or_Uni
full_demo_phone %>%
  group_by(College_Or_Uni) %>%
  summarise(college_or_uni_count=n())
```

```
## # A tibble: 1 x 2
##   College_Or_Uni college_or_uni_count
## * <chr>                         <int>
## 1 Yes                              17
```

```r
#Everyone went to college or uni which makes sense give the population. I will exclude this from the re
```

```r
#Waterloo_Relation
full_demo_phone %>%
  group_by(Waterloo_Relation) %>%
  summarise(waterloo_relation_count=n())
```

```
## # A tibble: 5 x 2
##   Waterloo_Relation                                 waterloo_relation_count
## * <chr>                                                               <int>
## 1 Former Undergraduate Student                                            1
## 2 Graduate Student                                                       10
## 3 Undergraduate Student                                                   4
## 4 Undergraduate Student, Graduate Student                                 1
## 5 Undergraduate Student, Graduate Student, Faculty                        1
```

```
#Majority grad students
```

```
#Community years
summary(as.integer(full_demo_phone$Community_Years))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.000   2.000   2.706   3.000   7.000
```

```
#Agerage is 2.7
```

```
#Occupation
full_demo_phone %>%
  group_by(Occupation) %>%
  summarise(occupation_count=n())
```

```
## # A tibble: 13 x 2
##    Occupation                                             occupation_count
##  * <chr>                                                             <int>
##  1 "Agile delivery manager"                                              1
##  2 "Engineer"                                                            1
##  3 "Game Developer / Temporary Associate"                                1
##  4 "grad student"                                                        1
##  5 "Graduate student"                                                    1
##  6 "Graduate Student"                                                    1
##  7 "PhD Student"                                                         1
##  8 "Registered nurse"                                                    1
##  9 "Software Developer"                                                  1
## 10 "student"                                                             1
## 11 "Student"                                                             4
## 12 "Student "                                                            2
## 13 "Supervisor at Peel Public Health for Vaccine Clinics"                1
```

```
#Mostly students, I need to just count manually because of case/whitespace/alternative words issues
```

```
#Headset_Expereince
full_demo_phone %>%
  group_by(Headset_Expereince) %>%
  summarise(Headset_Expereince_count=n())
```

```
## # A tibble: 2 x 2
##   Headset_Expereince Headset_Expereince_count
## * <chr>                                 <int>
## 1 No                                       10
## 2 Yes                                       7
```

```
# Around half in each category
```

```
#MobileAR_Expereince
full_demo_phone %>%
  group_by(MobileAR_Expereince) %>%
  summarise(MobileAR_Expereincee_count=n())
```

```
## # A tibble: 2 x 2
##   MobileAR_Expereince MobileAR_Expereincee_count
## * <chr>                                    <int>
## 1 No                                          10
## 2 Yes                                          7
```

```r
#Interestingly almost exactly the same as the headset! TODO: check if there is significant overlap betw
```

```r
#Phone
full_demo_phone %>%
  group_by(Phone) %>%
  summarise(phone_count=n())
```

```
## # A tibble: 15 x 2
##    Phone              phone_count
##  * <chr>                    <int>
##  1 Google Pixel 2               1
##  2 Huawei honor 8x              1
##  3 I phone 13 pro max           1
##  4 iphone                       1
##  5 iPhone 11                    2
##  6 Iphone 11                    1
##  7 Iphone 12 pro max            1
##  8 Iphone 13 pro                2
##  9 iPhone XR                    1
## 10 Iphone XR                    1
## 11 iPhone Xs                    1
## 12 Pixel 6                      1
## 13 Samsung 7                    1
## 14 Samsung Galaxy M21           1
## 15 Samsung Galaxy S9            1
```

```r
#Good mix, seems to be mostly iphones
```

Desirability Toolkit Analysis

I am cleaning the data because it is messy as it was people inputted

```r
#TODO: make desirability graph
Desirability_List <- raw_post$Desirability_List
Desirability_List <- gsub('[â€¢]', '', Desirability_List) #Use regex to remove special characters
Desirability_List
```

```
##  [1] " Compelling,  Creative,  Easy to use,  Engaging,  High quality,  Meaningful"
##  [2] " Annoying,  Not Valuable,  Overwhelming"
##  [3] " Creative,  High quality,  Innovative,  Useful"
##  [4] " Annoying,  Confusing,  Creative,  Cutting edge,  Engaging,  Hard to Use,  Innovative,  Meaning
##  [5] " Creative,  Easy to use,  Engaging,  Innovative,  Meaningful,  Overwhelming,  Useful"
##  [6] " Annoying,  Boring,  Slow,  Useful"
##  [7] " Confusing,  Creative,  Cutting edge,  Easy to use,  Empowering,  Engaging,  Innovative,  Meani
##  [8] " Creative,  Engaging,  High quality,  Irrelevant"
```

```
##  [9] " Creative,  Cutting edge,  Hard to Use,  High quality,  Innovative,  Meaningful,  Useful"
## [10] " Compelling,  Creative,  Cutting edge,  Easy to use,  Innovative,  Overwhelming,  Too Technical
## [11] " Confusing,  Engaging,  High quality,  Innovative,  Meaningful,  Relevant,  Useful"
## [12] " Creative,  Cutting edge,  Meaningful"
## [13] " Engaging,  High quality,  Innovative,  Meaningful,  Useful"
## [14] " Creative,  Easy to use,  Engaging,  Innovative,  Relevant"
## [15] " Compelling,  Creative,  Cutting edge,  Empowering,  Engaging,  Innovative,  Overwhelming,  Pe
## [16] " Creative,  Cutting edge,  Easy to use,  Engaging,  Innovative,  Meaningful,  Useful"
## [17] " Confusing,  Creative,  Engaging,  Hard to Use,  Innovative,  Meaningful"
```

Compiled words agnostic of participant

```
all_DT_words <- list()
for (participant in Desirability_List) {
  split <-  as.list(strsplit(participant, ","))
  all_DT_words <- c(all_DT_words, split[[1]])
}
all_DT_words2 <- list()
for (item in all_DT_words) {
  trimmed <- trimws(item)
  all_DT_words2 <- c(all_DT_words2, trimmed)
}
all_DT_words2
```

```
## [[1]]
## [1] "Compelling"
##
## [[2]]
## [1] "Creative"
##
## [[3]]
## [1] "Easy to use"
##
## [[4]]
## [1] "Engaging"
##
## [[5]]
## [1] "High quality"
##
## [[6]]
## [1] "Meaningful"
##
## [[7]]
## [1] "Annoying"
##
## [[8]]
## [1] "Not Valuable"
##
## [[9]]
## [1] "Overwhelming"
##
## [[10]]
## [1] "Creative"
```

```
##
## [[11]]
## [1] "High quality"
##
## [[12]]
## [1] "Innovative"
##
## [[13]]
## [1] "Useful"
##
## [[14]]
## [1] "Annoying"
##
## [[15]]
## [1] "Confusing"
##
## [[16]]
## [1] "Creative"
##
## [[17]]
## [1] "Cutting edge"
##
## [[18]]
## [1] "Engaging"
##
## [[19]]
## [1] "Hard to Use"
##
## [[20]]
## [1] "Innovative"
##
## [[21]]
## [1] "Meaningful"
##
## [[22]]
## [1] "Overwhelming"
##
## [[23]]
## [1] "Poor quality"
##
## [[24]]
## [1] "Creative"
##
## [[25]]
## [1] "Easy to use"
##
## [[26]]
## [1] "Engaging"
##
## [[27]]
## [1] "Innovative"
##
## [[28]]
## [1] "Meaningful"
```

```
##
## [[29]]
## [1] "Overwhelming"
##
## [[30]]
## [1] "Useful"
##
## [[31]]
## [1] "Annoying"
##
## [[32]]
## [1] "Boring"
##
## [[33]]
## [1] "Slow"
##
## [[34]]
## [1] "Useful"
##
## [[35]]
## [1] "Confusing"
##
## [[36]]
## [1] "Creative"
##
## [[37]]
## [1] "Cutting edge"
##
## [[38]]
## [1] "Easy to use"
##
## [[39]]
## [1] "Empowering"
##
## [[40]]
## [1] "Engaging"
##
## [[41]]
## [1] "Innovative"
##
## [[42]]
## [1] "Meaningful"
##
## [[43]]
## [1] "Relevant"
##
## [[44]]
## [1] "Creative"
##
## [[45]]
## [1] "Engaging"
##
## [[46]]
## [1] "High quality"
```

```
##
## [[47]]
## [1] "Irrelevant"
##
## [[48]]
## [1] "Creative"
##
## [[49]]
## [1] "Cutting edge"
##
## [[50]]
## [1] "Hard to Use"
##
## [[51]]
## [1] "High quality"
##
## [[52]]
## [1] "Innovative"
##
## [[53]]
## [1] "Meaningful"
##
## [[54]]
## [1] "Useful"
##
## [[55]]
## [1] "Compelling"
##
## [[56]]
## [1] "Creative"
##
## [[57]]
## [1] "Cutting edge"
##
## [[58]]
## [1] "Easy to use"
##
## [[59]]
## [1] "Innovative"
##
## [[60]]
## [1] "Overwhelming"
##
## [[61]]
## [1] "Too Technical"
##
## [[62]]
## [1] "Confusing"
##
## [[63]]
## [1] "Engaging"
##
## [[64]]
## [1] "High quality"
```

```
##
## [[65]]
## [1] "Innovative"
##
## [[66]]
## [1] "Meaningful"
##
## [[67]]
## [1] "Relevant"
##
## [[68]]
## [1] "Useful"
##
## [[69]]
## [1] "Creative"
##
## [[70]]
## [1] "Cutting edge"
##
## [[71]]
## [1] "Meaningful"
##
## [[72]]
## [1] "Engaging"
##
## [[73]]
## [1] "High quality"
##
## [[74]]
## [1] "Innovative"
##
## [[75]]
## [1] "Meaningful"
##
## [[76]]
## [1] "Useful"
##
## [[77]]
## [1] "Creative"
##
## [[78]]
## [1] "Easy to use"
##
## [[79]]
## [1] "Engaging"
##
## [[80]]
## [1] "Innovative"
##
## [[81]]
## [1] "Relevant"
##
## [[82]]
## [1] "Compelling"
```

```
##
## [[83]]
## [1] "Creative"
##
## [[84]]
## [1] "Cutting edge"
##
## [[85]]
## [1] "Empowering"
##
## [[86]]
## [1] "Engaging"
##
## [[87]]
## [1] "Innovative"
##
## [[88]]
## [1] "Overwhelming"
##
## [[89]]
## [1] "Personal"
##
## [[90]]
## [1] "Useful"
##
## [[91]]
## [1] "Creative"
##
## [[92]]
## [1] "Cutting edge"
##
## [[93]]
## [1] "Easy to use"
##
## [[94]]
## [1] "Engaging"
##
## [[95]]
## [1] "Innovative"
##
## [[96]]
## [1] "Meaningful"
##
## [[97]]
## [1] "Useful"
##
## [[98]]
## [1] "Confusing"
##
## [[99]]
## [1] "Creative"
##
## [[100]]
## [1] "Engaging"
```

```
##
## [[101]]
## [1] "Hard to Use"
##
## [[102]]
## [1] "Innovative"
##
## [[103]]
## [1] "Meaningful"
```

```
#Get frequencies of words
sort(table(unlist(all_DT_words2)), decreasing = T)
```

```
##
##      Creative    Innovative      Engaging    Meaningful        Useful
##            13            12            11            10             8
##  Cutting edge   Easy to use  High quality  Overwhelming     Confusing
##             7             6             6             5             4
##      Annoying    Compelling   Hard to Use      Relevant    Empowering
##             3             3             3             3             2
##        Boring    Irrelevant  Not Valuable      Personal  Poor quality
##             1             1             1             1             1
##          Slow Too Technical
##             1             1
```

Make graph object, this is all edges in

```
#Add verticies
toVert <- unique(all_DT_words2)
g <- make_empty_graph(directed = FALSE) %>%
  add_vertices(1, name="Compelling", color = "green") %>%
  add_vertices(1, name="Creative", color = "green") %>%
  add_vertices(1, name="Easy to use", color = "green") %>%
  add_vertices(1, name="Engaging", color = "green") %>%
  add_vertices(1, name="High quality", color = "green") %>%
  add_vertices(1, name="Meaningful", color = "green") %>%
  add_vertices(1, name="Innovative", color = "green") %>%
  add_vertices(1, name="Useful", color = "green") %>%
  add_vertices(1, name="Cutting edge", color = "green") %>%
  add_vertices(1, name="Empowering", color = "green") %>%
  add_vertices(1, name="Relevant", color = "green") %>%
  add_vertices(1, name="Personal", color = "green") %>%

  add_vertices(1, name="Annoying", color = "red") %>%
  add_vertices(1, name="Not Valuable", color = "red") %>%
  add_vertices(1, name="Overwhelming", color = "red") %>%
  add_vertices(1, name="Confusing", color = "red") %>%
  add_vertices(1, name="Hard to Use", color = "red") %>%
  add_vertices(1, name="Boring", color = "red") %>%
  add_vertices(1, name="Slow", color = "red") %>%
  add_vertices(1, name="Irrelevant", color = "red") %>%
  add_vertices(1, name="Too Technical", color = "red") %>%
  add_vertices(1, name="Poor quality", color = "red")
g
```

```
## IGRAPH 336e144 UN-- 22 0 --
## + attr: name (v/c), color (v/c)
## + edges from 336e144 (vertex names):
```

```
#### Add edges
#loop through participants
for (participant in Desirability_List) {

  #Split and trim
  split <-  as.list(strsplit(participant, ","))
  split_trimmed <- c()
  for (word in split){
    trimmed <- trimws(word)
    split_trimmed <- c(split_trimmed, trimmed)
  }

  #adding edges
  for(i1 in 1:length(split_trimmed)){
    for(i2 in 1:length(split_trimmed)){
      if (i1 < i2){ #then add edge

        #print("new edge")
        #print(are_adjacent(g, split_trimmed[i1], split_trimmed[i2]))
        #Check if there is an edge there already
        if (are_adjacent(g, split_trimmed[i1], split_trimmed[i2])) {
          #If there is an edge already increment
          ei <- get.edge.ids(g, c(split_trimmed[i1], split_trimmed[i2]))
          E(g)[ei]$weight = (E(g)[ei]$weight + 1)
        }
        else{
          g <- g + edge(split_trimmed[i1], split_trimmed[i2], color = "blue", weight=1)
        }
      }


    }
  }

}
```

Plot the graph

```
V(g)[[]]
```

```
## + 22/22 vertices, named, from 33a16e8:
##             name color
## 1     Compelling green
## 2       Creative green
## 3    Easy to use green
## 4       Engaging green
## 5   High quality green
## 6     Meaningful green
## 7     Innovative green
```

```
## 8         Useful green
## 9   Cutting edge green
## 10    Empowering green
## 11      Relevant green
## 12      Personal green
## 13      Annoying   red
## 14  Not Valuable   red
## 15  Overwhelming   red
## 16     Confusing   red
## 17   Hard to Use   red
## 18        Boring   red
## 19          Slow   red
## 20    Irrelevant   red
## 21 Too Technical   red
## 22  Poor quality   red
```

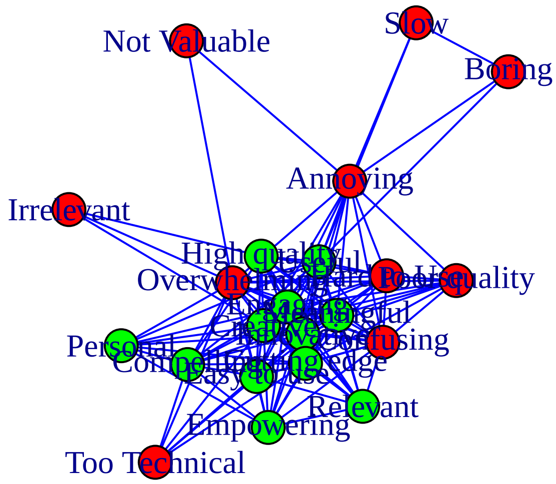
```
E(g)[[]]
```

```
## + 124/124 edges from 33a16e8 (vertex names):
##             tail          head tid hid color weight
## 1     Compelling      Creative   1   2  blue      3
## 2     Compelling   Easy to use   1   3  blue      2
## 3     Compelling      Engaging   1   4  blue      2
## 4     Compelling  High quality   1   5  blue      1
## 5     Compelling    Meaningful   1   6  blue      1
## 6       Creative   Easy to use   2   3  blue      6
## 7       Creative      Engaging   2   4  blue      9
## 8       Creative  High quality   2   5  blue      4
## 9       Creative    Meaningful   2   6  blue      8
## 10   Easy to use      Engaging   3   4  blue      5
## 11   Easy to use  High quality   3   5  blue      1
## 12   Easy to use    Meaningful   3   6  blue      4
## 13      Engaging  High quality   4   5  blue      4
## 14      Engaging    Meaningful   4   6  blue      8
## 15  High quality    Meaningful   5   6  blue      4
## 16      Annoying  Not Valuable  13  14  blue      1
## 17      Annoying  Overwhelming  13  15  blue      2
## 18  Not Valuable  Overwhelming  14  15  blue      1
## 19      Creative    Innovative   2   7  blue     10
## 20      Creative        Useful   2   8  blue      5
## 21  High quality    Innovative   5   7  blue      4
## 22  High quality        Useful   5   8  blue      4
## 23    Innovative        Useful   7   8  blue      7
## 24      Annoying     Confusing  13  16  blue      1
## 25      Creative      Annoying   2  13  blue      1
## 26  Cutting edge      Annoying   9  13  blue      1
## 27      Engaging      Annoying   4  13  blue      1
## 28      Annoying   Hard to Use  13  17  blue      1
## 29    Innovative      Annoying   7  13  blue      1
## 30    Meaningful      Annoying   6  13  blue      1
## 31      Annoying  Poor quality  13  22  blue      1
## 32      Creative     Confusing   2  16  blue      3
## 33  Cutting edge     Confusing   9  16  blue      2
## 34      Engaging     Confusing   4  16  blue      4
```

```
## 35     Confusing   Hard to Use  16  17  blue       2
## 36    Innovative     Confusing   7  16  blue       4
## 37    Meaningful     Confusing   6  16  blue       4
## 38  Overwhelming     Confusing  15  16  blue       1
## 39     Confusing  Poor quality  16  22  blue       1
## 40      Creative  Cutting edge   2   9  blue       7
## 41      Creative   Hard to Use   2  17  blue       3
## 42      Creative  Overwhelming   2  15  blue       4
## 43      Creative  Poor quality   2  22  blue       1
## 44      Engaging  Cutting edge   4   9  blue       4
## 45  Cutting edge   Hard to Use   9  17  blue       2
## 46    Innovative  Cutting edge   7   9  blue       6
## 47    Meaningful  Cutting edge   6   9  blue       5
## 48  Cutting edge  Overwhelming   9  15  blue       3
## 49  Cutting edge  Poor quality   9  22  blue       1
## 50      Engaging   Hard to Use   4  17  blue       2
## 51      Engaging    Innovative   4   7  blue       9
## 52      Engaging  Overwhelming   4  15  blue       3
## 53      Engaging  Poor quality   4  22  blue       1
## 54    Innovative   Hard to Use   7  17  blue       3
## 55    Meaningful   Hard to Use   6  17  blue       3
## 56  Overwhelming   Hard to Use  15  17  blue       1
## 57   Hard to Use  Poor quality  17  22  blue       1
## 58    Meaningful    Innovative   6   7  blue       8
## 59    Innovative  Overwhelming   7  15  blue       4
## 60    Innovative  Poor quality   7  22  blue       1
## 61    Meaningful  Overwhelming   6  15  blue       2
## 62    Meaningful  Poor quality   6  22  blue       1
## 63  Overwhelming  Poor quality  15  22  blue       1
## 64   Easy to use    Innovative   3   7  blue       5
## 65   Easy to use  Overwhelming   3  15  blue       2
## 66   Easy to use        Useful   3   8  blue       2
## 67      Engaging        Useful   4   8  blue       5
## 68    Meaningful        Useful   6   8  blue       5
## 69        Useful  Overwhelming   8  15  blue       2
## 70      Annoying        Boring  13  18  blue       1
## 71      Annoying          Slow  13  19  blue       1
## 72        Useful      Annoying   8  13  blue       1
## 73        Boring          Slow  18  19  blue       1
## 74        Useful        Boring   8  18  blue       1
## 75        Useful          Slow   8  19  blue       1
## 76   Easy to use     Confusing   3  16  blue       1
## 77    Empowering     Confusing  10  16  blue       1
## 78      Relevant     Confusing  11  16  blue       2
## 79      Creative    Empowering   2  10  blue       2
## 80      Creative      Relevant   2  11  blue       2
## 81   Easy to use  Cutting edge   3   9  blue       3
## 82  Cutting edge    Empowering   9  10  blue       2
## 83  Cutting edge      Relevant   9  11  blue       1
## 84   Easy to use    Empowering   3  10  blue       1
## 85   Easy to use      Relevant   3  11  blue       2
## 86      Engaging    Empowering   4  10  blue       2
## 87    Innovative    Empowering   7  10  blue       2
## 88    Meaningful    Empowering   6  10  blue       1
```

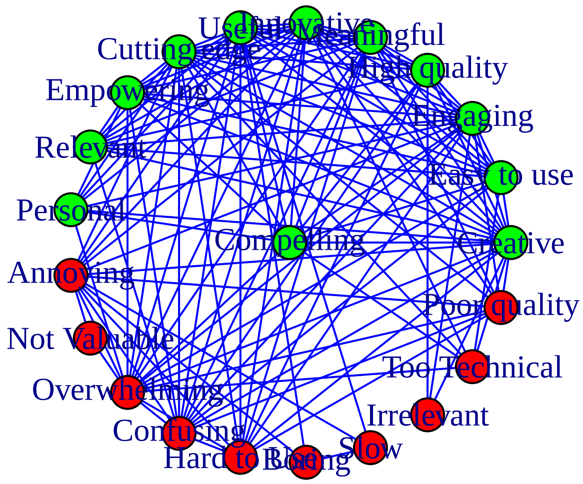
```
## 89    Empowering      Relevant  10  11  blue       1
## 90      Engaging      Relevant   4  11  blue       3
## 91    Innovative      Relevant   7  11  blue       3
## 92    Meaningful      Relevant   6  11  blue       2
## 93      Creative    Irrelevant   2  20  blue       1
## 94      Engaging    Irrelevant   4  20  blue       1
## 95  High quality    Irrelevant   5  20  blue       1
## 96  High quality  Cutting edge   5   9  blue       1
## 97        Useful  Cutting edge   8   9  blue       3
## 98  High quality   Hard to Use   5  17  blue       1
## 99        Useful   Hard to Use   8  17  blue       1
## 100   Compelling  Cutting edge   1   9  blue       2
## 101   Compelling    Innovative   1   7  blue       2
## 102   Compelling  Overwhelming   1  15  blue       2
## 103   Compelling Too Technical   1  21  blue       1
## 104     Creative Too Technical   2  21  blue       1
## 105 Cutting edge Too Technical   9  21  blue       1
## 106  Easy to use Too Technical   3  21  blue       1
## 107   Innovative Too Technical   7  21  blue       1
## 108 Overwhelming Too Technical  15  21  blue       1
## 109 High quality     Confusing   5  16  blue       1
## 110       Useful     Confusing   8  16  blue       1
## 111 High quality      Relevant   5  11  blue       1
## 112       Useful      Relevant   8  11  blue       1
## 113   Compelling    Empowering   1  10  blue       1
## 114   Compelling      Personal   1  12  blue       1
## 115   Compelling        Useful   1   8  blue       1
## 116     Creative      Personal   2  12  blue       1
## 117 Cutting edge      Personal   9  12  blue       1
## 118   Empowering  Overwhelming  10  15  blue       1
## 119   Empowering      Personal  10  12  blue       1
## 120       Useful    Empowering   8  10  blue       1
## 121     Engaging      Personal   4  12  blue       1
## 122   Innovative      Personal   7  12  blue       1
## 123     Personal  Overwhelming  12  15  blue       1
## 124       Useful      Personal   8  12  blue       1
```

```
plot(g)
```



Plot the graph using a force directed layout

```
coords <- layout_(g, as_star())
plot(g, layout = coords)
```



Make a cluter using cluster walktrap From: https://igraph.org/r/doc/cluster_walktrap.html

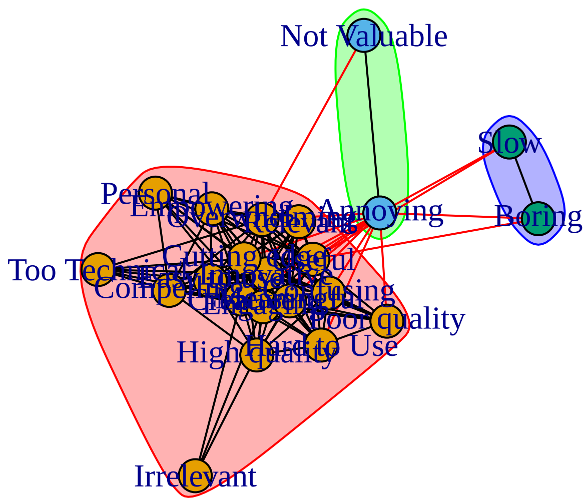
```
wc <- cluster_walktrap(g, steps=12)
modularity(wc)
```

```
## [1] 0.01805606
```

```
membership(wc)
```

```
##    Compelling      Creative   Easy to use      Engaging  High quality 
##             1             1             1             1             1 
##    Meaningful    Innovative        Useful  Cutting edge    Empowering 
##             1             1             1             1             1 
##      Relevant      Personal      Annoying  Not Valuable  Overwhelming 
##             1             1             2             2             1 
##     Confusing   Hard to Use        Boring          Slow    Irrelevant 
##             1             1             3             3             1 
## Too Technical  Poor quality 
##             1             1
```

```
plot(wc, g)
```



This is a new graph where I got rid of words that only happened once total.

```
once_list <- c("Boring", "Irrelevant", "Not Valuable", "Personal", "Poor quality", "Slow", "Too Technica
toVert <- unique(all_DT_words2)
g <- make_empty_graph(directed = FALSE) %>%
  add_vertices(1, name="Compelling", color = "green") %>%
  add_vertices(1, name="Creative", color = "green") %>%
  add_vertices(1, name="Easy to use", color = "green") %>%
  add_vertices(1, name="Engaging", color = "green") %>%
  add_vertices(1, name="High quality", color = "green") %>%
  add_vertices(1, name="Meaningful", color = "green") %>%
  add_vertices(1, name="Innovative", color = "green") %>%
  add_vertices(1, name="Useful", color = "green") %>%
  add_vertices(1, name="Cutting edge", color = "green") %>%
  add_vertices(1, name="Empowering", color = "green") %>%
  add_vertices(1, name="Relevant", color = "green") %>%

  add_vertices(1, name="Annoying", color = "red") %>%
  add_vertices(1, name="Overwhelming", color = "red") %>%
  add_vertices(1, name="Confusing", color = "red") %>%
  add_vertices(1, name="Hard to Use", color = "red")
g
```

```
## IGRAPH 33cba92 UN-- 15 0 --
## + attr: name (v/c), color (v/c)
## + edges from 33cba92 (vertex names):
```

```r
#### Add edges
#loop through participants
for (participant in Desirability_List) {

  #Split and trim
  split <-  as.list(strsplit(participant, ","))
  split_trimmed <- c()
  for (word in split){
    trimmed <- trimws(word)
    split_trimmed <- c(split_trimmed, trimmed)
  }

  #adding edges
  for(i1 in 1:length(split_trimmed)){
    for(i2 in 1:length(split_trimmed)){
      if (i1 < i2){ #then add edge

        #Check if in once list
        if ( !(split_trimmed[i1] %in% once_list | split_trimmed[i2] %in% once_list) ){

          #Check if there is an edge there already
          if (are_adjacent(g, split_trimmed[i1], split_trimmed[i2])) {
            #If there is an edge already increment
            ei <- get.edge.ids(g, c(split_trimmed[i1], split_trimmed[i2]))
            E(g)[ei]$weight = (E(g)[ei]$weight + 1)
          }
          else{
            g <- g + edge(split_trimmed[i1], split_trimmed[i2], color = "blue", weight=1)
          }
        }
      }



    }
  }

}
```

Plot paired down graph

```r
V(g)[[]]
```

```
## + 15/15 vertices, named, from 33fd8c8:
##           name color
## 1   Compelling green
## 2     Creative green
## 3  Easy to use green
## 4     Engaging green
## 5 High quality green
## 6   Meaningful green
## 7   Innovative green
## 8       Useful green
```

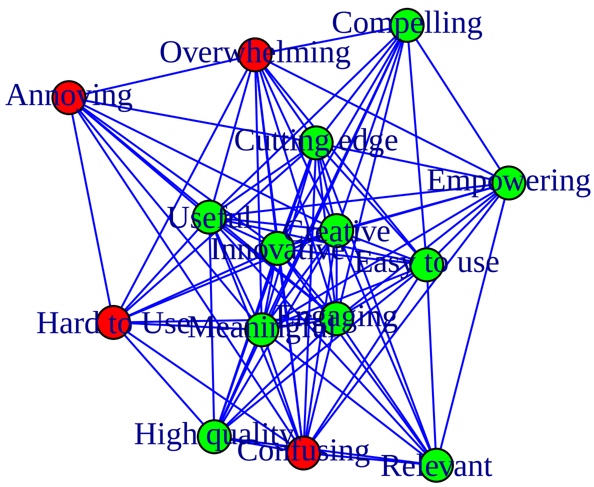
```
## 9  Cutting edge green
## 10   Empowering green
## 11     Relevant green
## 12     Annoying   red
## 13 Overwhelming   red
## 14    Confusing   red
## 15  Hard to Use   red
```

```
E(g)[[]]
```

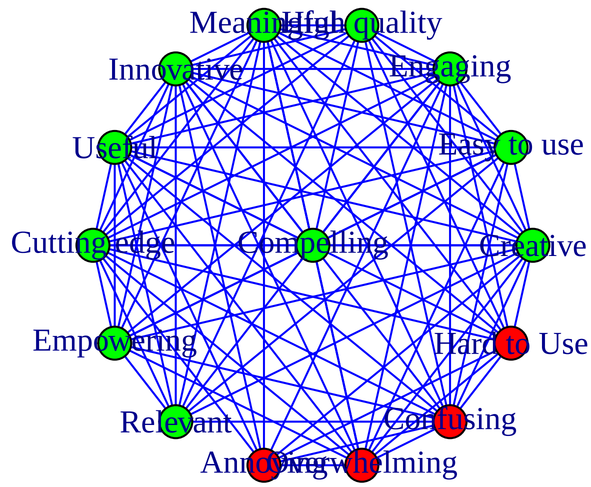
```
## + 91/91 edges from 33fd8c8 (vertex names):
##            tail         head tid hid color weight
## 1    Compelling     Creative   1   2  blue      3
## 2    Compelling  Easy to use   1   3  blue      2
## 3    Compelling     Engaging   1   4  blue      2
## 4    Compelling High quality   1   5  blue      1
## 5    Compelling   Meaningful   1   6  blue      1
## 6      Creative  Easy to use   2   3  blue      6
## 7      Creative     Engaging   2   4  blue      9
## 8      Creative High quality   2   5  blue      4
## 9      Creative   Meaningful   2   6  blue      8
## 10  Easy to use     Engaging   3   4  blue      5
## 11  Easy to use High quality   3   5  blue      1
## 12  Easy to use   Meaningful   3   6  blue      4
## 13     Engaging High quality   4   5  blue      4
## 14     Engaging   Meaningful   4   6  blue      8
## 15 High quality   Meaningful   5   6  blue      4
## 16     Annoying Overwhelming  12  13  blue      2
## 17     Creative   Innovative   2   7  blue     10
## 18     Creative       Useful   2   8  blue      5
## 19 High quality   Innovative   5   7  blue      4
## 20 High quality       Useful   5   8  blue      4
## 21   Innovative       Useful   7   8  blue      7
## 22     Annoying    Confusing  12  14  blue      1
## 23     Creative     Annoying   2  12  blue      1
## 24 Cutting edge     Annoying   9  12  blue      1
## 25     Engaging     Annoying   4  12  blue      1
## 26     Annoying  Hard to Use  12  15  blue      1
## 27   Innovative     Annoying   7  12  blue      1
## 28   Meaningful     Annoying   6  12  blue      1
## 29     Creative    Confusing   2  14  blue      3
## 30 Cutting edge    Confusing   9  14  blue      2
## 31     Engaging    Confusing   4  14  blue      4
## 32    Confusing  Hard to Use  14  15  blue      2
## 33   Innovative    Confusing   7  14  blue      4
## 34   Meaningful    Confusing   6  14  blue      4
## 35 Overwhelming    Confusing  13  14  blue      1
## 36     Creative Cutting edge   2   9  blue      7
## 37     Creative  Hard to Use   2  15  blue      3
## 38     Creative Overwhelming   2  13  blue      4
## 39     Engaging Cutting edge   4   9  blue      4
## 40 Cutting edge  Hard to Use   9  15  blue      2
## 41   Innovative Cutting edge   7   9  blue      6
## 42   Meaningful Cutting edge   6   9  blue      5
```

```
## 43 Cutting edge Overwhelming   9  13  blue      3
## 44     Engaging  Hard to Use   4  15  blue      2
## 45     Engaging   Innovative   4   7  blue      9
## 46     Engaging Overwhelming   4  13  blue      3
## 47   Innovative  Hard to Use   7  15  blue      3
## 48   Meaningful  Hard to Use   6  15  blue      3
## 49 Overwhelming  Hard to Use  13  15  blue      1
## 50   Meaningful   Innovative   6   7  blue      8
## 51   Innovative Overwhelming   7  13  blue      4
## 52   Meaningful Overwhelming   6  13  blue      2
## 53  Easy to use   Innovative   3   7  blue      5
## 54  Easy to use Overwhelming   3  13  blue      2
## 55  Easy to use       Useful   3   8  blue      2
## 56     Engaging       Useful   4   8  blue      5
## 57   Meaningful       Useful   6   8  blue      5
## 58       Useful Overwhelming   8  13  blue      2
## 59       Useful     Annoying   8  12  blue      1
## 60  Easy to use    Confusing   3  14  blue      1
## 61   Empowering    Confusing  10  14  blue      1
## 62     Relevant    Confusing  11  14  blue      2
## 63     Creative   Empowering   2  10  blue      2
## 64     Creative     Relevant   2  11  blue      2
## 65  Easy to use Cutting edge   3   9  blue      3
## 66 Cutting edge   Empowering   9  10  blue      2
## 67 Cutting edge     Relevant   9  11  blue      1
## 68  Easy to use   Empowering   3  10  blue      1
## 69  Easy to use     Relevant   3  11  blue      2
## 70     Engaging   Empowering   4  10  blue      2
## 71   Innovative   Empowering   7  10  blue      2
## 72   Meaningful   Empowering   6  10  blue      1
## 73   Empowering     Relevant  10  11  blue      1
## 74     Engaging     Relevant   4  11  blue      3
## 75   Innovative     Relevant   7  11  blue      3
## 76   Meaningful     Relevant   6  11  blue      2
## 77 High quality Cutting edge   5   9  blue      1
## 78       Useful Cutting edge   8   9  blue      3
## 79 High quality  Hard to Use   5  15  blue      1
## 80       Useful  Hard to Use   8  15  blue      1
## 81   Compelling Cutting edge   1   9  blue      2
## 82   Compelling   Innovative   1   7  blue      2
## 83   Compelling Overwhelming   1  13  blue      2
## 84 High quality    Confusing   5  14  blue      1
## 85       Useful    Confusing   8  14  blue      1
## 86 High quality     Relevant   5  11  blue      1
## 87       Useful     Relevant   8  11  blue      1
## 88   Compelling   Empowering   1  10  blue      1
## 89   Compelling       Useful   1   8  blue      1
## 90   Empowering Overwhelming  10  13  blue      1
## 91       Useful   Empowering   8  10  blue      1
```

```
plot(g)
```




```
coords <- layout_(g, as_star())
plot(g, layout = coords)
```



Plot graph clustered via cluster walktrap

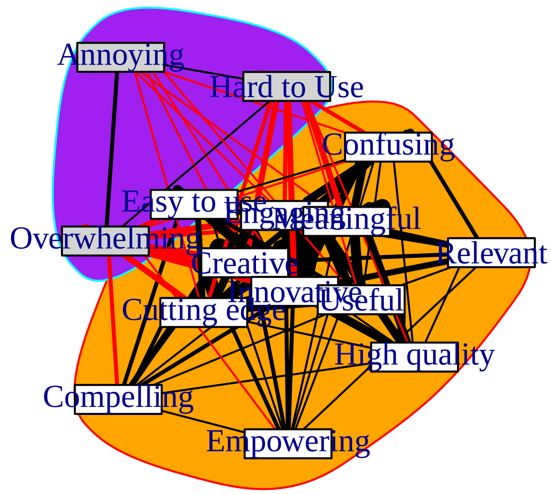
```
wc <- cluster_walktrap(g)
modularity(wc)
```

```
## [1] 0.00774914
```

```
membership(wc)
```

```
##   Compelling     Creative  Easy to use     Engaging High quality   Meaningful
##            1            1            1            1            1            1
##   Innovative       Useful Cutting edge   Empowering     Relevant     Annoying
##            1            1            1            1            1            2
## Overwhelming    Confusing  Hard to Use
##            2            1            2
```

```
new_cols <- c("white", "lightgray")[membership(wc)]
plot(wc, g, edge.width=E(g)$weight, vertex.shape="rectangle", vertex.size=45, col=new_cols, mark.col=c(
```


Annoying
Hard to Use
Confusing
Easy to use
Engaging
Meaningful
Overwhelming
Relevant
Creative
Innovative
Useful
Cutting edge
High quality
Compelling
Empowering