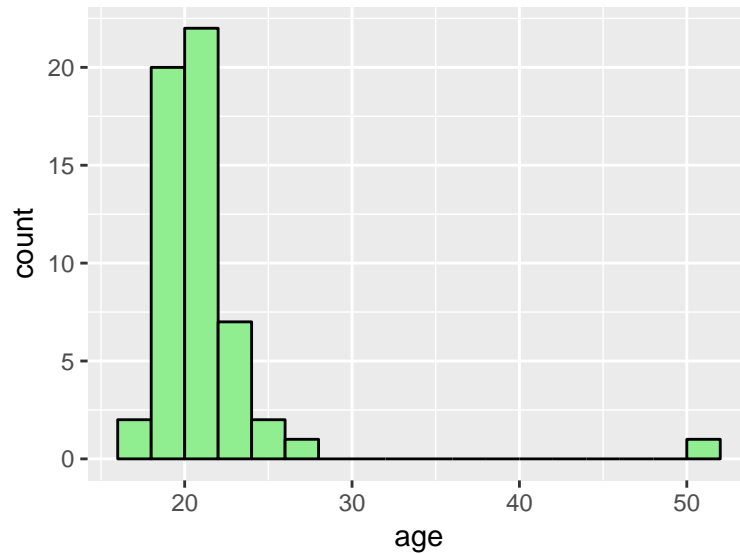


# Information about the participants in the cartogram experiment

## Demographics

### Age distribution



### Gender

Gender	Frequency
Female	37
Male	18

### Highest level of education

Education	Frequency
High school	19
Some college	22
Bachelor's	1
Bachelor's	12
Master's	1

## Information about prior knowledge

### Familiarity with interactive computer graphics

Familiarity	Frequency
Very unfamiliar	0
Unfamiliar	12
Somewhat familiar	29
Familiar	14
Very familiar	0

### Familiarity with cartograms

Familiarity	Frequency
Very unfamiliar	12
Unfamiliar	0
Somewhat familiar	17
Familiar	4
Very familiar	0
No response	22

### Map affinity

On a world map, would you be able to point at the location where you were born?

Answer	Frequency
Yes, with confidence	39
Yes, with a little effort	10
Yes, with much effort	2
Possibly, but I'm uncertain	3
No	1

When you encounter the names of unfamiliar locations (e.g. countries, islands, lakes), how frequently do you immediately look them up on a map to find out where they are?

Answer	Frequency
Never	2
Rarely	15
Sometimes	19
Generally	16
Always	3

## Colour vision

We showed six Ishihara plates to each participant. Participants 28, 37, 42, 50 responded differently from the majority in exactly one case each. Participant 2 had two, participant 34 even three abnormal responses. We should investigate whether participants 2 and 34 showed unusual responses to the remaining questions too.

## Questions

### Taxon (Task Taxonomy According to Nusrat & Kobourov 2015)

- CL - cluster: find the region that has the most similar data to a given region.

Sample Question: On the other monitor, you can see a conventional map of Brazil (left) and a cattle population cartogram (right). Out of the states listed below, which has a cattle population most similar to Mato Grosso do Sul (MS)? A. Rondonia (RO), B. Para (PA), C. Mato Grosso (MT), D. San Paulo (SP)

- CO - compare: compare data between two regions.

Sample Question: On the other monitor, you can see a conventional map of Germany (left) and a GDP cartogram (right). Is the GDP of Mecklenburg-Vorpommern (MV) higher than, approximately equal to or lower than the GDP of Rheinland-Pfalz (RP)? A. Higher, B. Approximately equal, C. Lower

- DC - detect change: compare areas of a given region in two cartograms.

Sample Question: On the other monitor, you can see a conventional map of Brazil (left), a cattle population cartogram and a human population (right). You may have to use the cartogram selector tab to view both cartograms. Is the area of Amazonas (AM) in the cattle population cartogram higher than, approximately equal to or lower than its area in the human population cartogram? A. Higher, B. Approximately equal, C. Lower

- FA - find adjacency: find all neighbours of a given region.

Sample Question: On the other monitor, you can see a conventional map of Brazil (left) and a population cartogram (right). Which states are neighbours of the state highlighted in red in the population cartogram below? There may be more than one correct answer. A. Bahia (BA), B. Minas Gerais (MG), C. Maranhao (MA), D. Sergipe (SE)

- FL - filter: find all regions whose data surpasses that of a given region.

Sample Question: On the other monitor, you can see a conventional map of Germany (left) and a population cartogram (right). Out of the states listed below, which one(s) have a population higher than Baden-Wuerttemberg (BW)? There may be more than one correct answer. A. Nordrhein-Westfalen (NW), B. Hessen (HE), C. Niedersachsen (NI), D. Bayern (BY).

- FT - find top: find the region that has the highest data across the country.

Sample Question: On the other monitor, you can see a conventional map of the USA (left) and a crop production cartogram (right). Which state of the USA has the highest crop production? A. California (CA), B. Montana (MT), C. Maine (ME), D. Iowa (IA).

- RE - recognize: identify the name of a given region highlighted in the cartogram.

Sample Question: On the other monitor, you can see a conventional map of the India (left) and a population cartogram (right). Which is the name of the state highlighted in the population cartogram? A. Haryana (HR), B. Punjab (PB), C. Rajasthan (RJ), D. Himachal Pradesh (HP).

- SU - summarize: compare data of three different regions in two cartograms.

Sample Question: On the other monitor, you can see a conventional map of Germany (left) and population cartograms for the years 1985 and 2015 (right). You may have to use the cartogram selector tab to view both cartograms. Three different regions are highlighted in yellow, purple and pink. What can you say about the change in population for each region from 1985 to 2015? For all three regions, A. Growth, B. Approximately No Change, C. Shrinking.

## **Treatment Groups**

A - all (i.e. highlighting, infotip and switch button),

H - highlighting,

I - infotip,

N - none,

S - switch button.

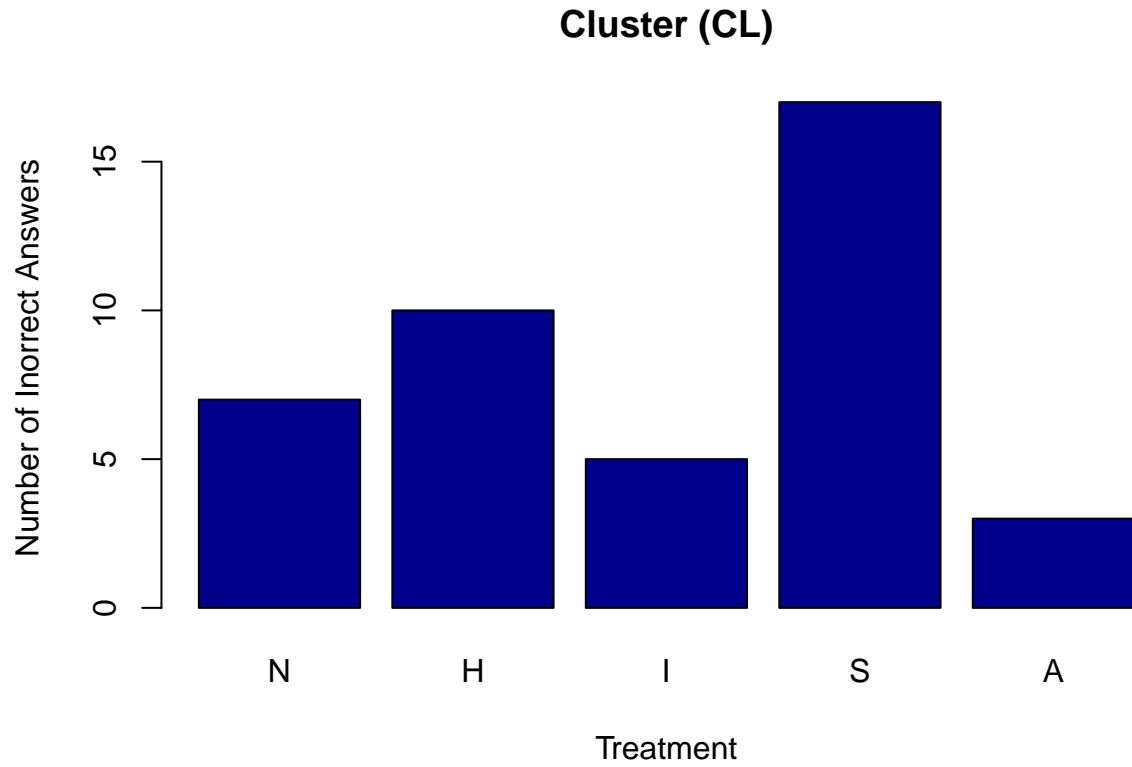
## **Correctness**

A question is considered to be answered correctly if and only if all correct answers and no incorrect answers are selected. A question is considered to be answered incorrectly otherwise. There is no partial credit given for any question.

## Results

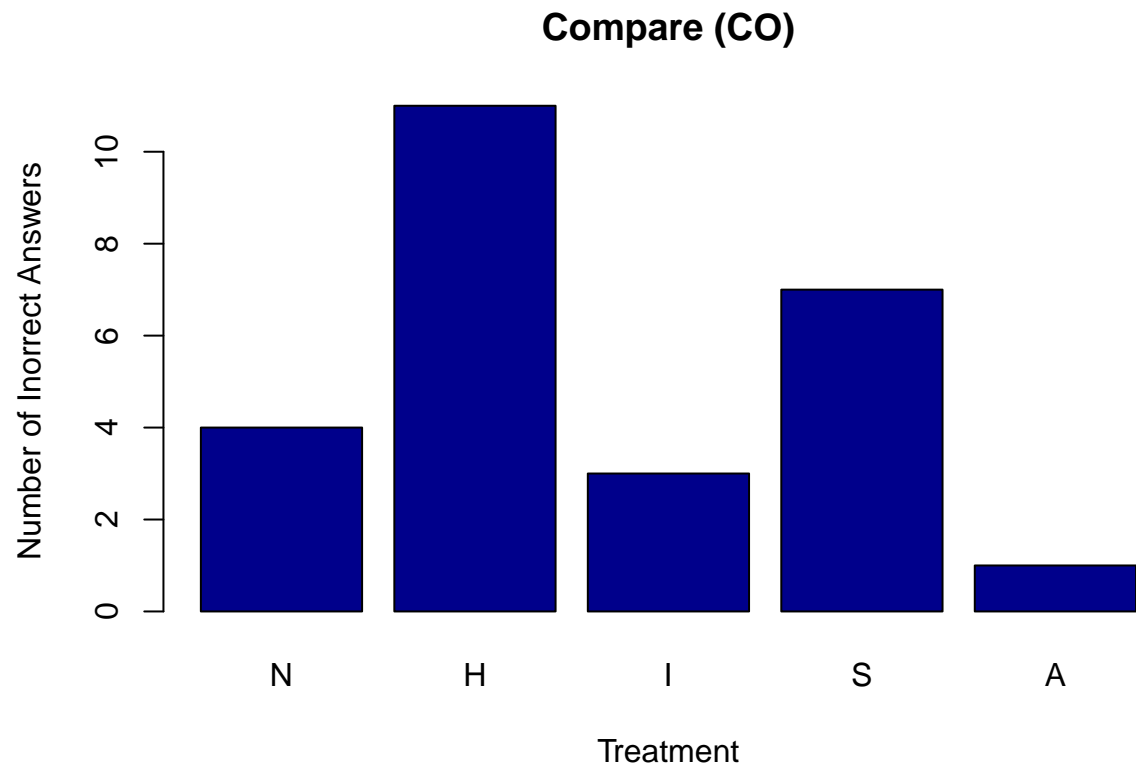
### Histograms of the number of Incorrect Answers

CL



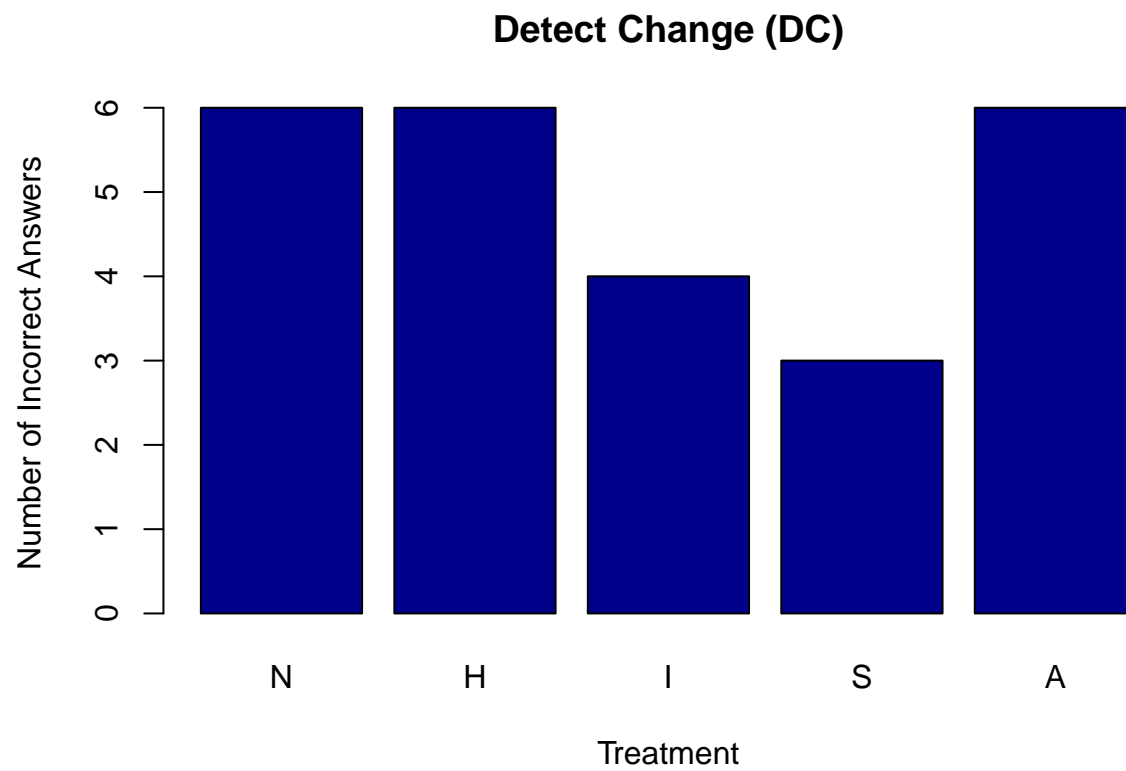
6 out of the 15 participants (one data point missing) who are provided with switch actually used switch.

CO



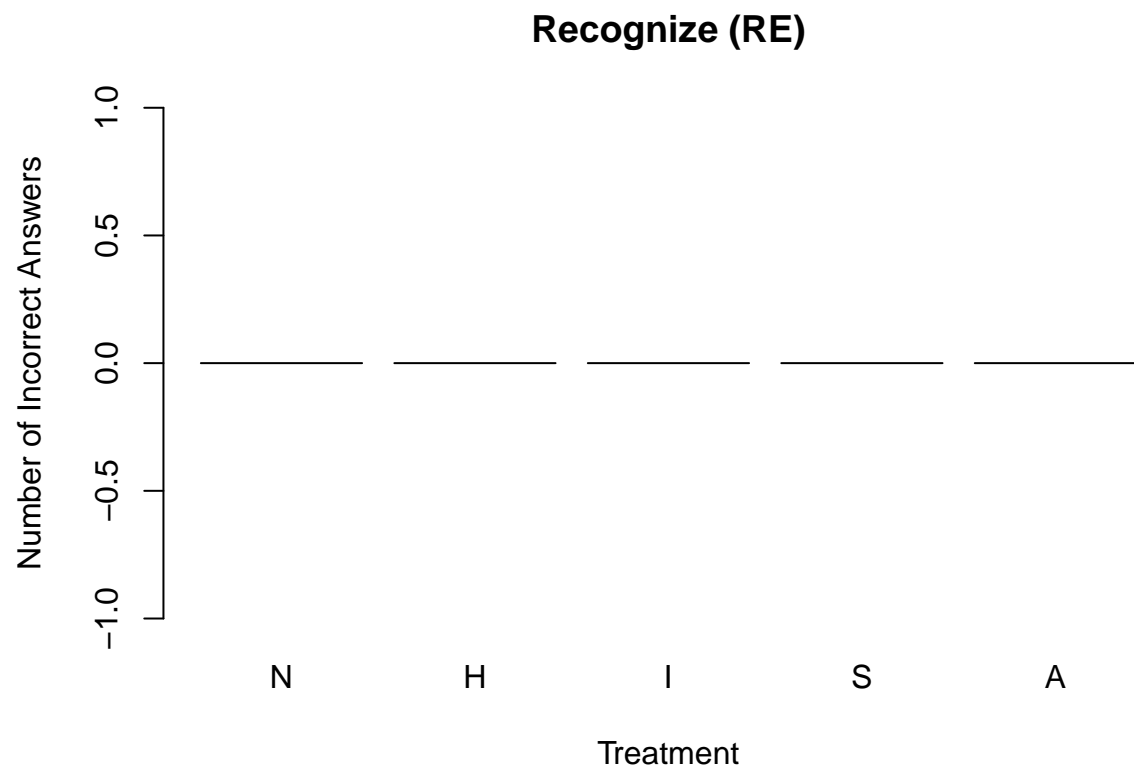
no significant effects

DC



no significant effects

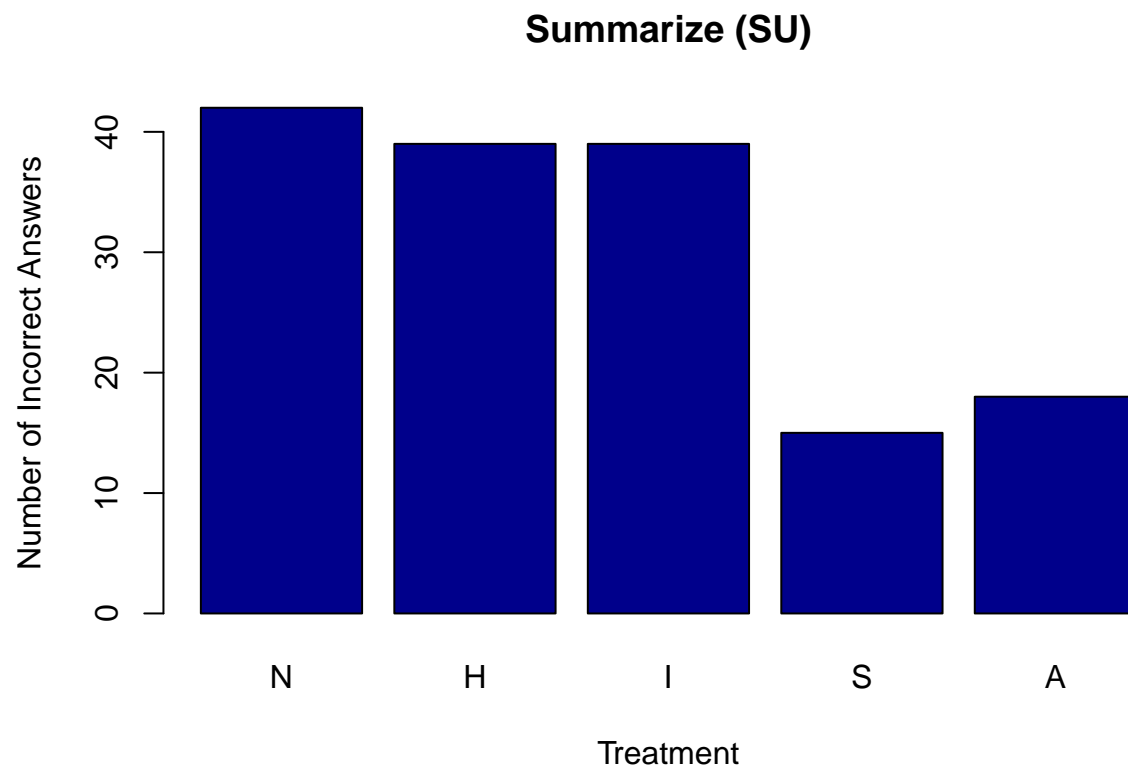
RE



no significant effects

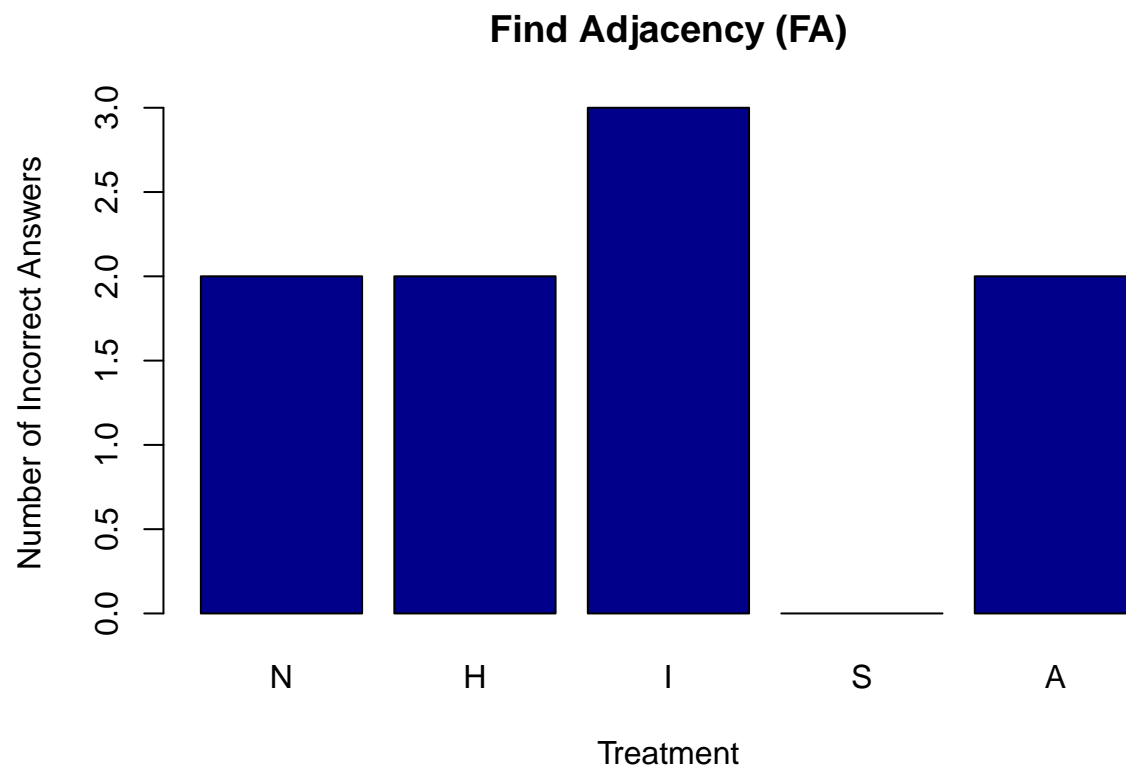


SU



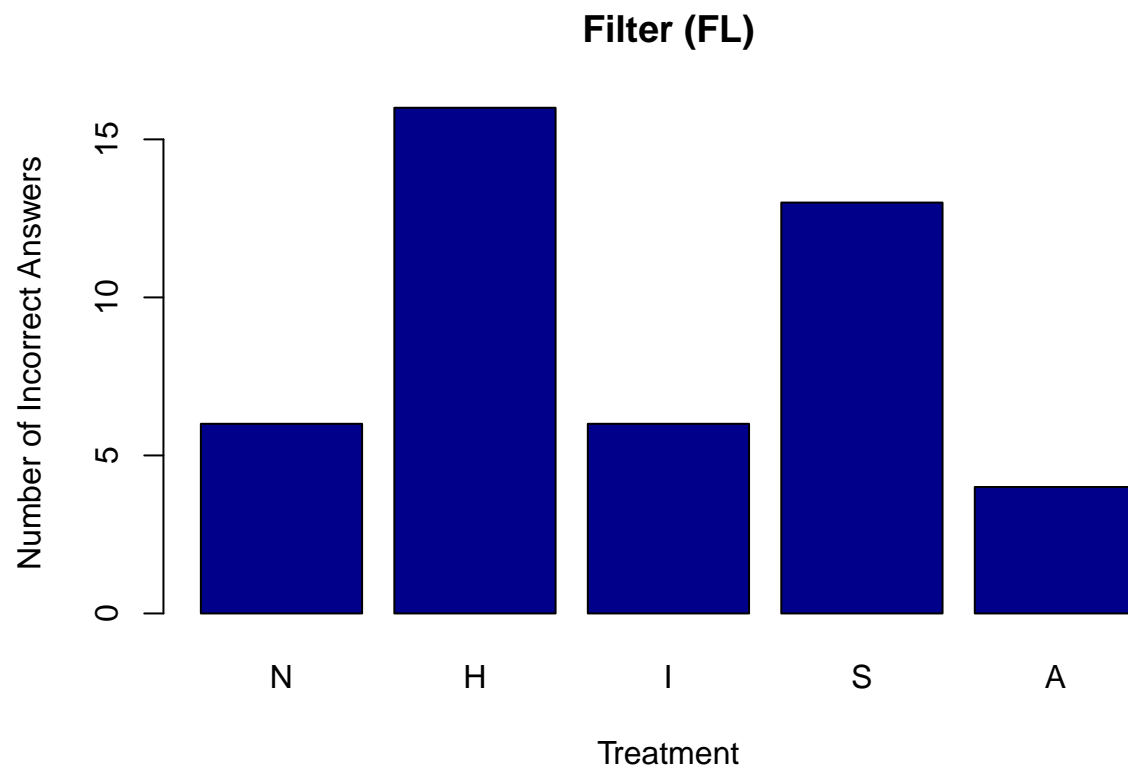
18 out of 18 participants who are provided with switch actually used switch.

FA



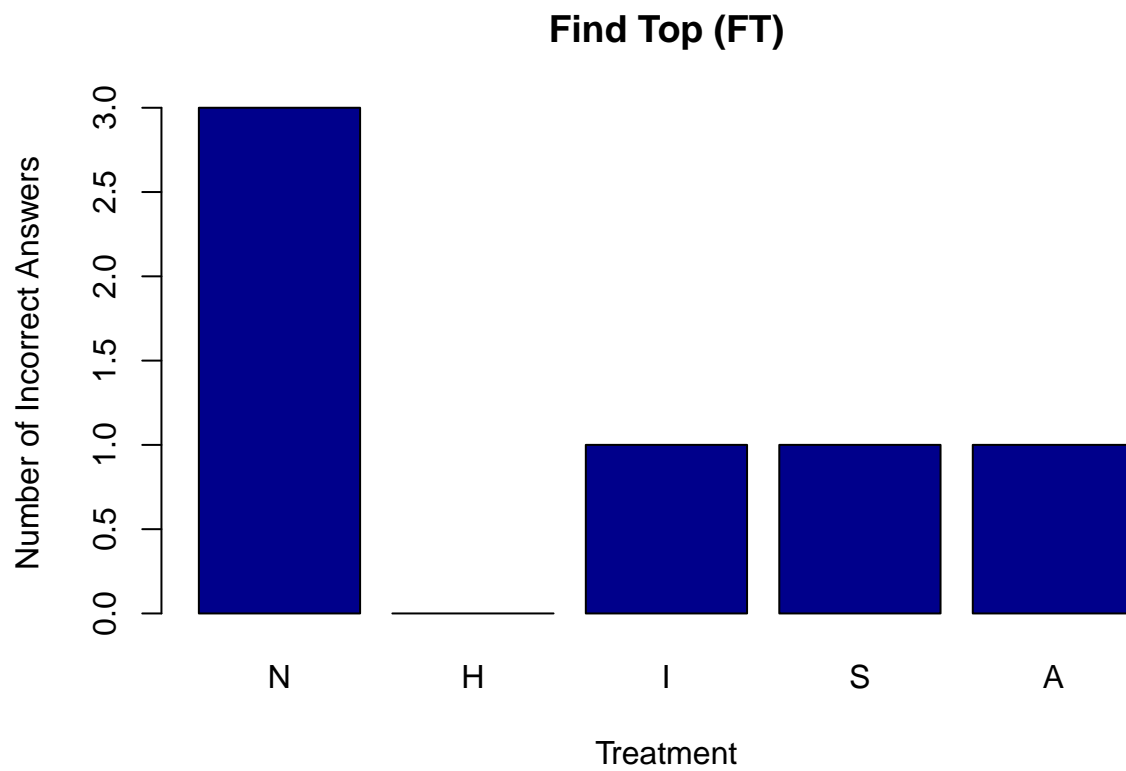
no significant effects

FL



no significant effects

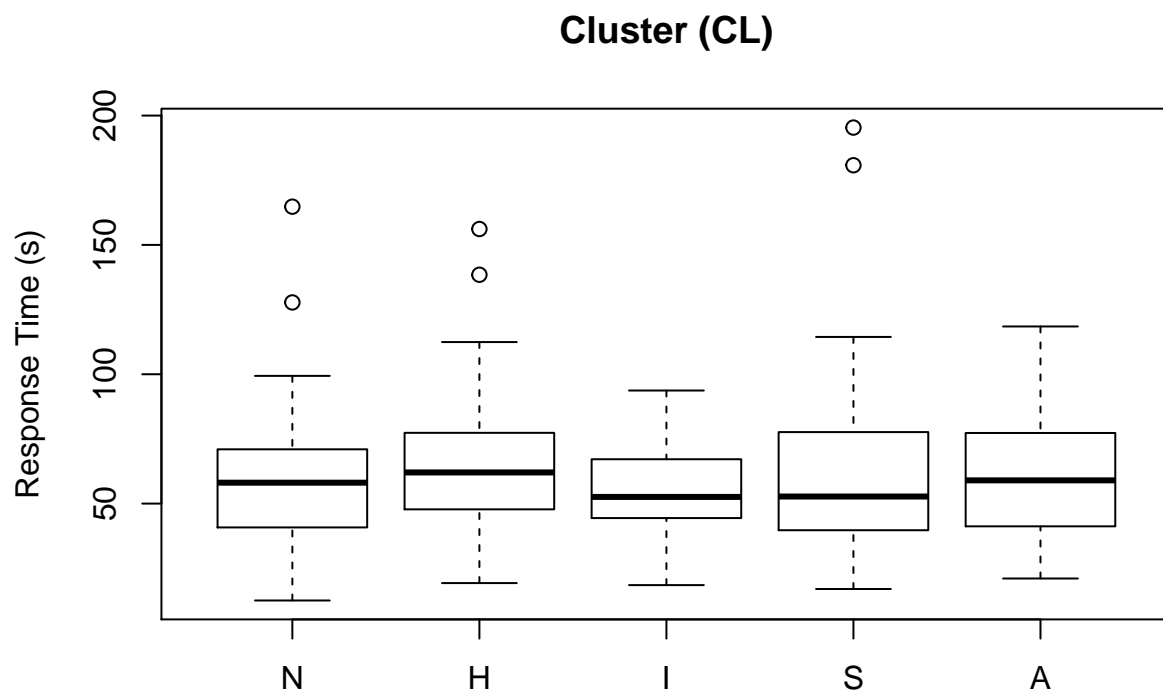
FT



no significant effects

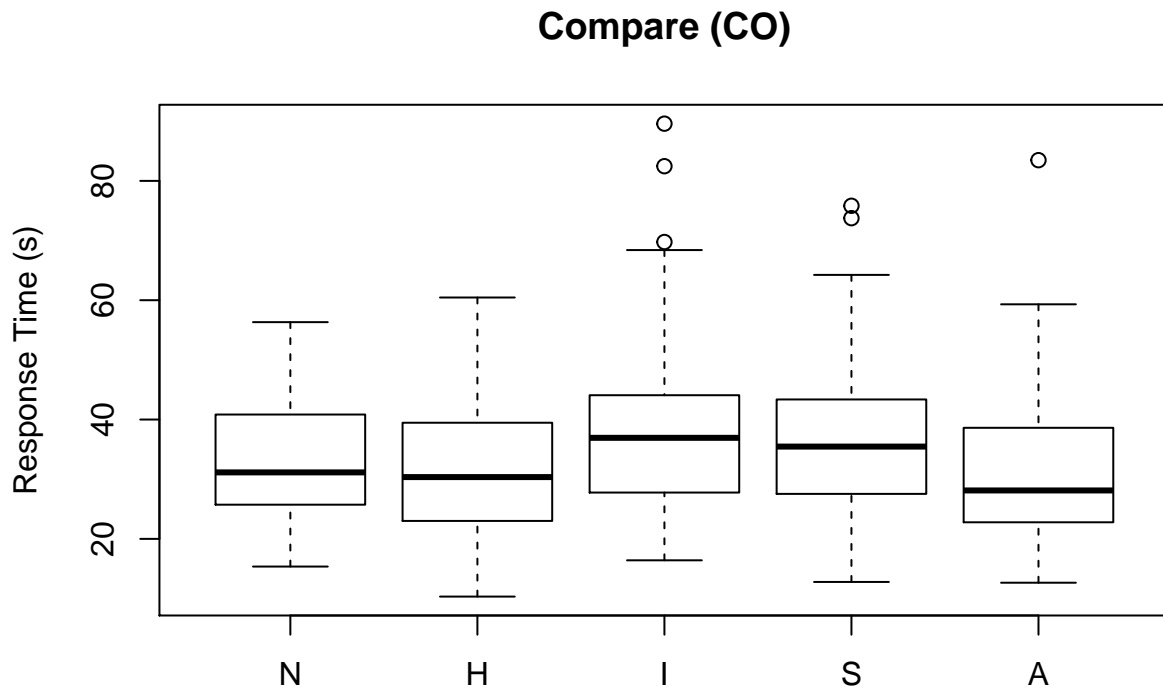
### Boxplots of Response Time Conditioned on Correctness

CL



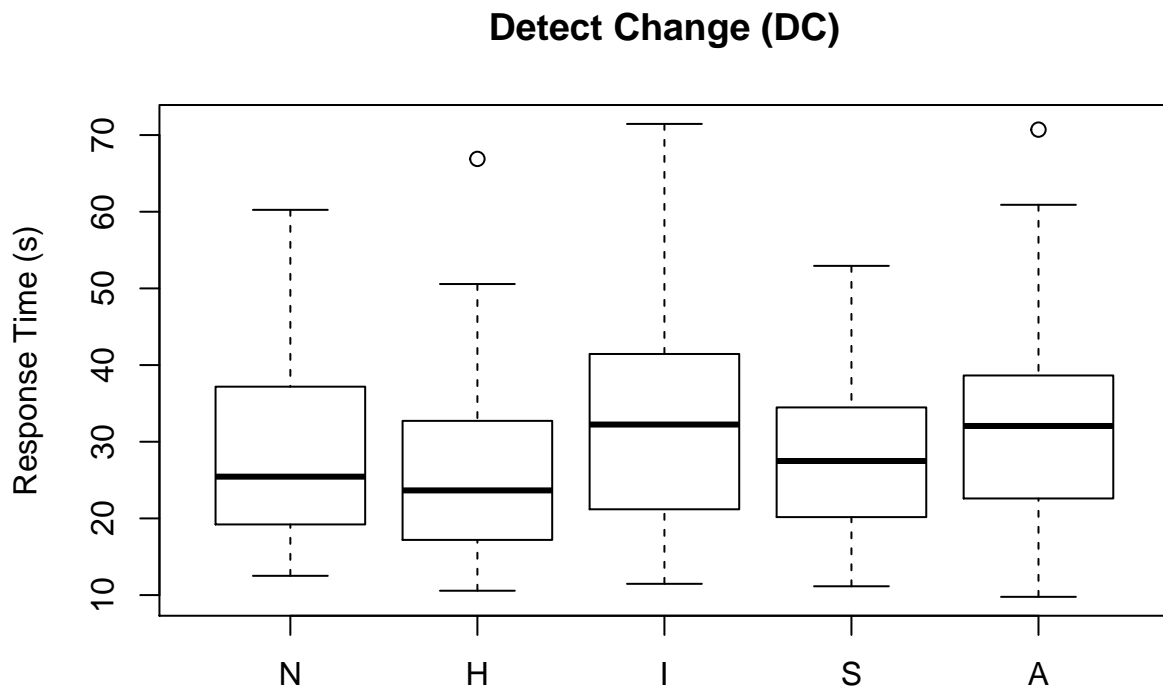
nothing significant

CO



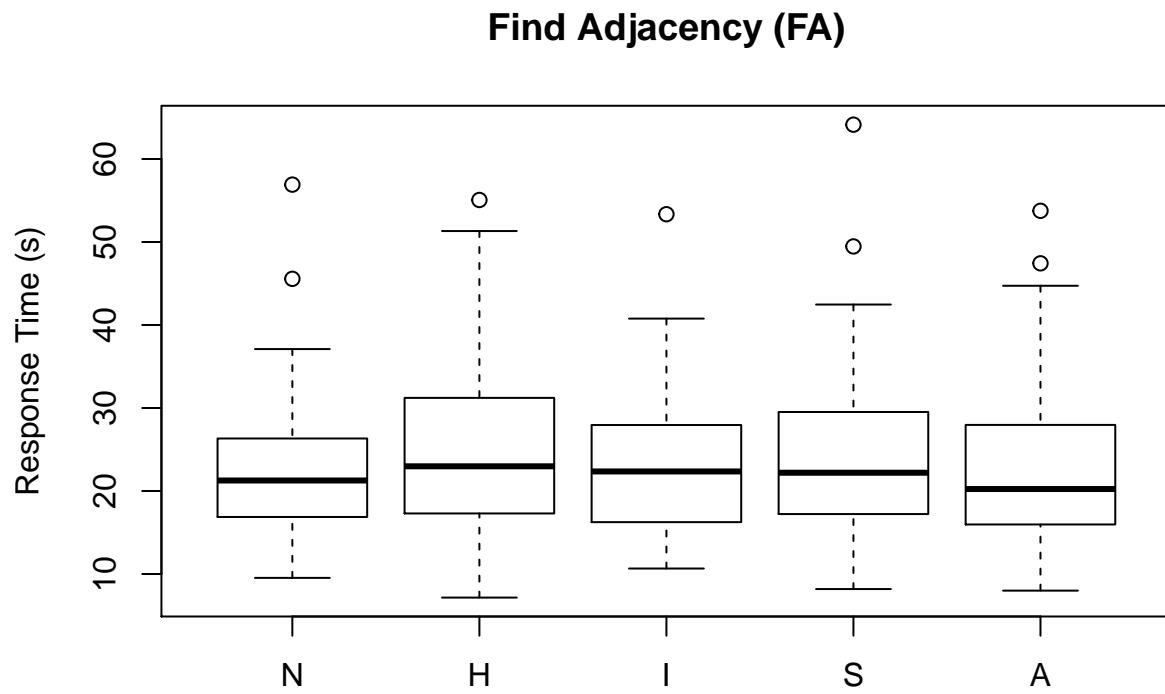
H does not help answer correctly, but the correct answer is quicker.

DC



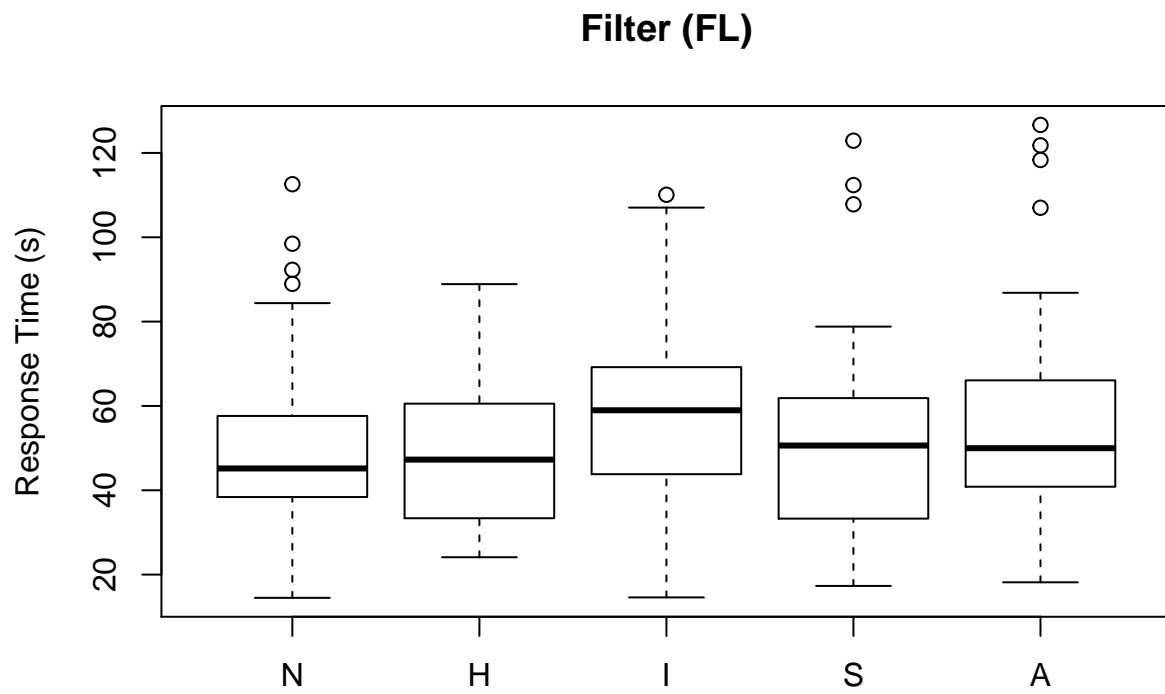
most participants use switch several times back and forth before choosing their answers

FA



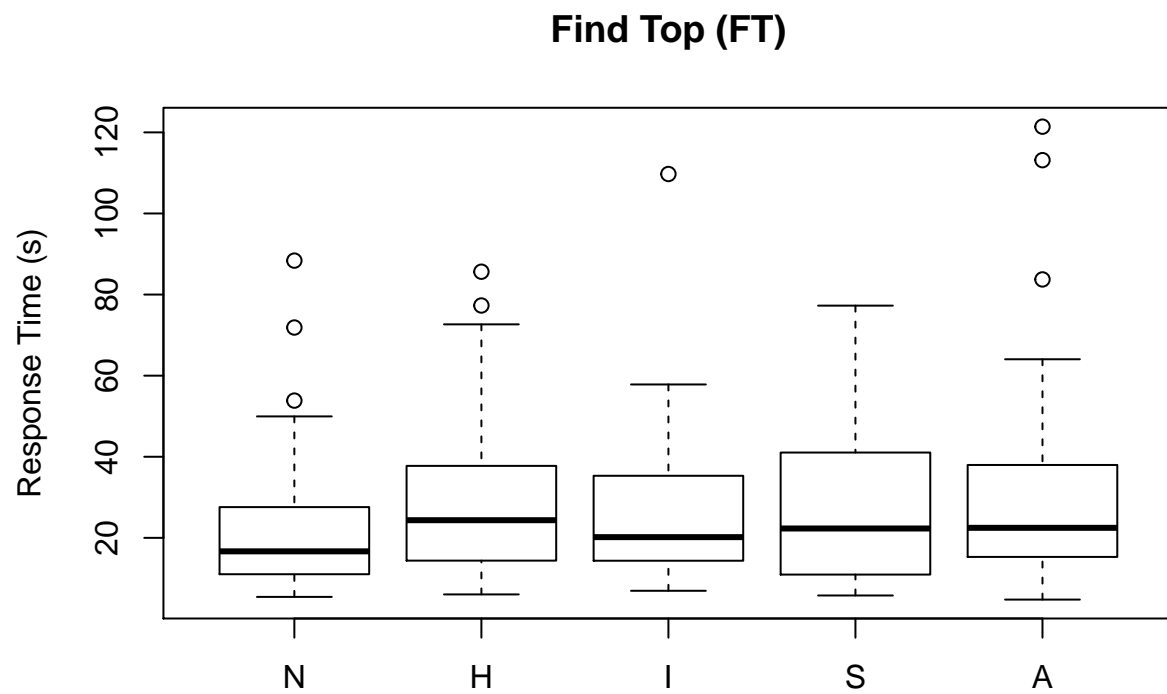
nothing significant

FL



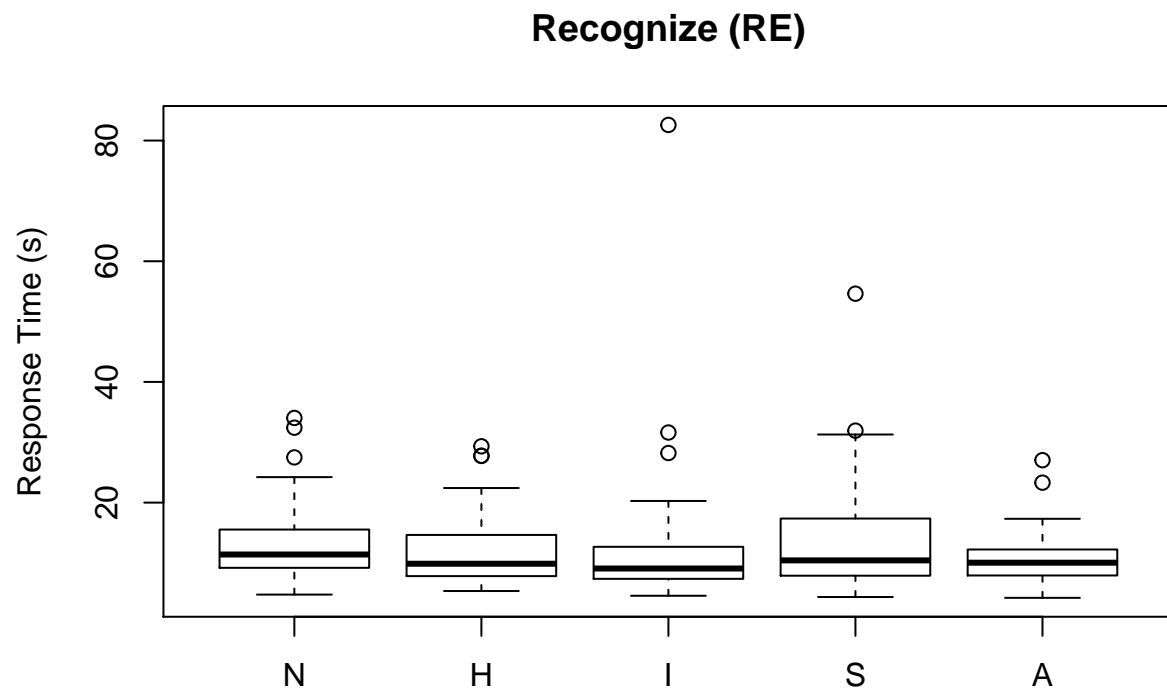
nothing significant

FT



nothing significant

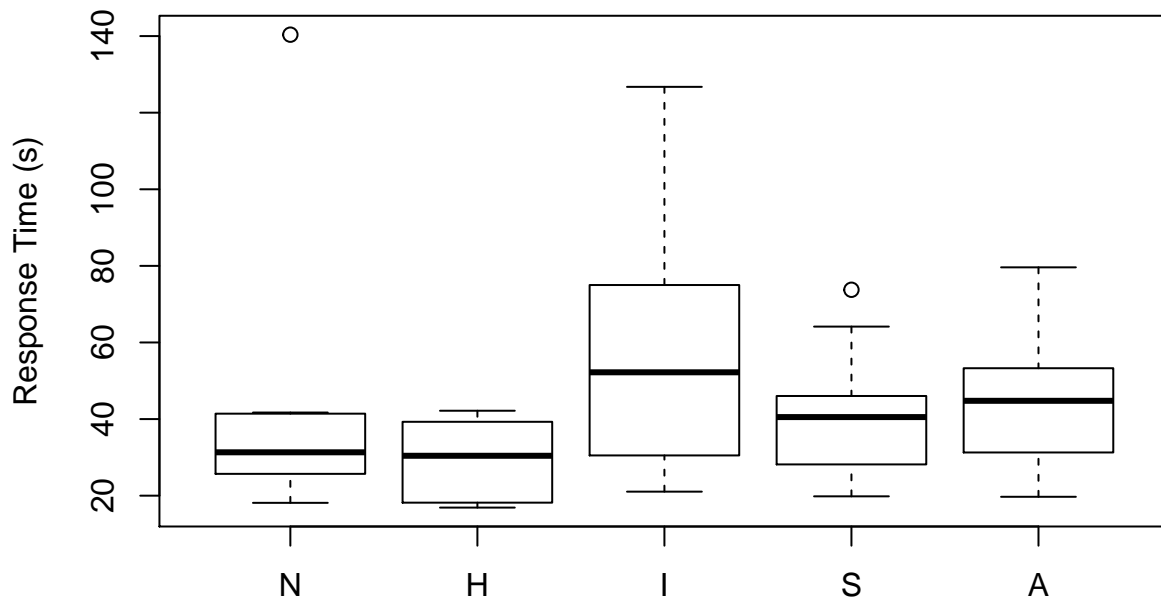
RE



nothing significant

SU

## Summarize (SU)



the spread is significantly higher for I because of two outliers (>100s).

## Experiments

### Pairwise Prop Test

```
taxonomy <- rep(c("CL", "CO", "DC", "RE", "SU", "FA", "FL", "FT"))

for (i in 1:8) {
  print(taxonomy[i])
  ppt_matrix = response %>% select(treatment, taxon, is_correct)
  ppt_matrix_incorrect = filter(ppt_matrix, taxon == taxonomy[i], is_correct == FALSE)
  ppt_matrix_incorrect = select(ppt_matrix_incorrect, treatment)
  print(pairwise.prop.test(
    c(nrow(filter(ppt_matrix_incorrect, treatment == "N")),
      nrow(filter(ppt_matrix_incorrect, treatment == "H")),
      nrow(filter(ppt_matrix_incorrect, treatment == "I")),
      nrow(filter(ppt_matrix_incorrect, treatment == "S")),
      nrow(filter(ppt_matrix_incorrect, treatment == "A"))),
    rep(55, 5)))
}
```

## [1] "CL"

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared  
## approximation may be incorrect

##

## Pairwise comparisons using Pairwise comparison of proportions



```

##
## data:  c(nrow(filter(ppt_matrix_incorrect, treatment == "N")), nrow(filter(ppt_matrix_incorrect, ou
##
##      1      2      3      4
## 2 1.000 -      -      -
## 3 1.000 1.000 -      -
## 4 0.302 1.000 0.079 -
## 5 1.000 0.535 1.000 0.013
##
## P value adjustment method: holm
## [1] "C0"

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

##
## Pairwise comparisons using Pairwise comparison of proportions
##
## data:  c(nrow(filter(ppt_matrix_incorrect, treatment == "N")), nrow(filter(ppt_matrix_incorrect, ou
##
##      1      2      3      4
## 2 0.669 -      -      -
## 3 1.000 0.407 -      -
## 4 1.000 1.000 1.000 -
## 5 1.000 0.059 1.000 0.531
##
## P value adjustment method: holm
## [1] "DC"

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

##
## Pairwise comparisons using Pairwise comparison of proportions
##
## data:  c(nrow(filter(ppt_matrix_incorrect, treatment == "N")), nrow(filter(ppt_matrix_incorrect, ou
##
##      1 2 3 4

```

```

## 2 1 - - -
## 3 1 1 - -
## 4 1 1 1 -
## 5 1 1 1 1
##
## P value adjustment method: holm
## [1] "RE"

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

##
## Pairwise comparisons using Pairwise comparison of proportions
##
## data:  c(nrow(filter(ppt_matrix_incorrect, treatment == "N")), nrow(filter(ppt_matrix_incorrect, ou
##
##    1 2 3 4
## 2 - - - -
## 3 - - - -
## 4 - - - -
## 5 - - - -
##
## P value adjustment method: holm
## [1] "SU"
##
## Pairwise comparisons using Pairwise comparison of proportions
##
## data:  c(nrow(filter(ppt_matrix_incorrect, treatment == "N")), nrow(filter(ppt_matrix_incorrect, ou

```

```

##
##      1      2      3      4
## 2 1.00000 -      -      -
## 3 1.00000 1.00000 -      -
## 4 7.0e-06 9.6e-05 9.6e-05 -
## 5 9.6e-05 0.00081 0.00081 1.00000
##
## P value adjustment method: holm
## [1] "FA"

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

##
## Pairwise comparisons using Pairwise comparison of proportions
##
## data:  c(nrow(filter(ppt_matrix_incorrect, treatment == "N")), nrow(filter(ppt_matrix_incorrect, ou
##
##      1 2 3 4
## 2 1 - - -
## 3 1 1 - -
## 4 1 1 1 -
## 5 1 1 1 1
##
## P value adjustment method: holm
## [1] "FL"
##
## Pairwise comparisons using Pairwise comparison of proportions

```

```

##
## data:  c(nrow(filter(ppt_matrix_incorrect, treatment == "N")), nrow(filter(ppt_matrix_incorrect, ou
##
##      1      2      3      4
## 2 0.287 -      -      -
## 3 1.000 0.287 -      -
## 4 0.781 1.000 0.781 -
## 5 1.000 0.065 1.000 0.287
##
## P value adjustment method: holm
## [1] "FT"

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

## Warning in prop.test(x[c(i, j)], n[c(i, j)], ...): Chi-squared
## approximation may be incorrect

##
## Pairwise comparisons using Pairwise comparison of proportions
##
## data:  c(nrow(filter(ppt_matrix_incorrect, treatment == "N")), nrow(filter(ppt_matrix_incorrect, ou
##
##      1 2 3 4
## 2 1 - - -
## 3 1 1 - -
## 4 1 1 1 -
## 5 1 1 1 1
##
## P value adjustment method: holm

```

## ANOVA

```
taxonomy <- rep(c("CL", "CO", "DC", "RE", "SU", "FA", "FL", "FT"))

for (i in 1:8) {
  print(taxonomy[i])
  anova_matrix = response %>% select(response_time, treatment, taxon, is_correct)
  anova_matrix_correct = filter(anova_matrix, taxon == taxonomy[i], is_correct == TRUE)
  anova_matrix_correct = select(anova_matrix_correct, response_time, treatment)
  print(summary(aov(response_time ~ treatment, anova_matrix_correct)))
}

## [1] "CL"
##               Df Sum Sq Mean Sq F value Pr(>F)
## treatment      4    3289    822.3   1.044  0.385
## Residuals    228  179505    787.3
## [1] "CO"
##               Df Sum Sq Mean Sq F value  Pr(>F)
## treatment      4    2558    639.5   3.975 0.00383 **
## Residuals    244   39259    160.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## [1] "DC"
##               Df Sum Sq Mean Sq F value  Pr(>F)
## treatment      4    2102    525.4   3.572 0.00748 **
## Residuals    245   36041    147.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## [1] "RE"
##               Df Sum Sq Mean Sq F value Pr(>F)
## treatment      4     260    65.03   1.128  0.344
## Residuals    270   15563    57.64
## [1] "SU"
##               Df Sum Sq Mean Sq F value Pr(>F)
## treatment      4    1652    412.9   1.024  0.398
## Residuals    117   47159    403.1
## [1] "FA"
##               Df Sum Sq Mean Sq F value Pr(>F)
## treatment      4     309    77.17   0.812  0.518
## Residuals    261   24806    95.04
## [1] "FL"
##               Df Sum Sq Mean Sq F value Pr(>F)
## treatment      4    2418    604.4   1.247  0.292
## Residuals    225  109083    484.8
## [1] "FT"
##               Df Sum Sq Mean Sq F value Pr(>F)
## treatment      4    1838    459.5   1.217  0.304
## Residuals    264   99677    377.6
```