questionnaire

Andrew

01/26/2021

This file analyzes questionnaire ratings data.

1 Correctness and Confidence

```
df.data %>%
  select(subject_id, solver, correct_actual, q_confidence) %>%
  distinct() %>%
  group_by(solver) %>%
  summarize(n = n(),
            acc.mean = mean(correct_actual),
            acc.se = sd(correct_actual) / sqrt(n),
            conf.mean = mean(q_confidence),
            conf.se = sd(q_confidence) / sqrt(n)) %>%
  knitr::kable()
```

solver	n	acc.mean	acc.se	conf.mean	conf.se
Solver	84	0.952381	0.0233753	96.42857	1.398127
Non-Solver	84	0.500000	0.0548821	56.72619	2.709249

2 Forced-response questions

Forced-response summary statistics

```
df.questions_long %>%
  group_by(measure, solver) %>%
  summarize(mean = mean(value),
            se = sd(value) / sqrt(n())) %>%
  knitr::kable()
```

measure	solver	mean	se
Attention Check	Solver	0.9007937	0.0176555
Attention Check	Non-Solver	0.8253968	0.0236014
Solved Puzzle	Solver	0.9523810	0.0233753
Solved Puzzle	Non-Solver	0.5000000	0.0548821

measure	solver	mean	se
Puzzle Confidence	Solver	0.9642857	0.0139813
Puzzle Confidence	Non-Solver	0.5672619	0.0270925
Noticed	Solver	0.8095238	0.0431019
Noticed	Non-Solver	0.5833333	0.0541145
Checked Candidate	Solver	0.7500000	0.0529009
Checked Candidate	Non-Solver	0.4489796	0.0717921
Checked House	Solver	0.7254902	0.0631117
Checked House	Non-Solver	0.3636364	0.1049728

2.1 Did the raters correctly judge the correctness of subjects' responses?

```
df.data %>%
  mutate(correct_acc = correct_eval == correct_actual) %>%
  group_by(rater) %>%
  summarize(correct_acc = mean(correct_acc)) %>%
  knitr::kable()
```

rater	correct_acc
rater1	0.9702381
rater2	1.0000000

2.2 Did the raters agree on their decisions?

2.2.1 PD

```
df.data %>%
  select(subject_id, rater, pd) %>%
  pivot_wider(names_from = rater, values_from = pd) %>%
  mutate(agree = rater1 == rater2) %>%
  summarize(agree = mean(agree)) %>%
  knitr::kable()
```

agree
0.8928571

2.2.2 Awareness of Error

```
df.data %>%
  select(subject_id, rater, aoe) %>%
  drop_na(aoe) %>%
  pivot_wider(names_from = rater, values_from = aoe) %>%
  mutate(agree = rater1 == rater2) %>%
```

```
  summarize(agree = mean(agree)) %>%
  knitr::kable()
```

agree
1

2.2.3 Basis for Choice

Rule: Either both first bases match or a first basis match with second basis. Since our focus on this analysis is just first basis, if only second bases match, we should just count them as disagreements.

```
df.data %>%
  select(subject_id, rater, basis, basis2) %>%
  mutate(across(c(basis, basis2),
                .fns = as.character)) %>%
  pivot_wider(names_from = rater,
              values_from = c('basis', 'basis2'),
              names_glue = "{rater}_{.value}") %>%
  mutate(agree = rater1_basis == rater2_basis |
           rater1_basis == rater2_basis2 |
           rater1_basis2 == rater2_basis) %>%
  select(subject_id, agree, everything()) %>%
  summarize(agree = mean(agree)) %>%
  knitr::kable()
```

agree
0.75

2.3 Basis for Choice Group Differences

Chi-squared tests

```
df = df.data %>%
  filter(correct_actual)

chisq.test(df$solver, df$basis)
chisq.test(df$solver, df$valid_basis)
#>
#>  Pearson's Chi-squared test
#>
#> data:  df$solver and df$basis
#> X-squared = 125.86, df = 8, p-value < 2.2e-16
#>
#>
#>  Pearson's Chi-squared test with Yates' continuity correction
#>
#> data:  df$solver and df$valid_basis
#> X-squared = 99.116, df = 1, p-value < 2.2e-16
```

3 Education

```
#> Warning: Predicate functions must be wrapped in `where()`.
#>
#>   # Bad
#>   data %>% select(is_logical)
#>
#>   # Good
#>   data %>% select(where(is_logical))
#>
#> i Please update your code.
#> This message is displayed once per session.
```

3.1 Regressions

3.1.1 Education

```
lm.edu = lm(n_solved ~ education, df.edu)
lm.edu %>%
  summary()
#>
#> Call:
#> lm(formula = n_solved ~ education, data = df.edu)
#>
#> Residuals:
#>     Min      1Q  Median      3Q     Max
#> -54.468 -14.468  -5.468  19.532  35.266
#>
#> Coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)  36.5319     8.0334   4.548 8.22e-06 ***
#> education     1.4335     0.5283   2.713  0.00709 **
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 19.42 on 269 degrees of freedom
#> Multiple R-squared:  0.02664,    Adjusted R-squared:  0.02302
#> F-statistic: 7.362 on 1 and 269 DF,  p-value: 0.007093
```

3.1.2 Math

```
lm.math = lm(n_solved ~ alg + geom + trig + sv_calc +
               mv_calc + linalg + pr_stat + disc + logic,
             df.edu)
lm.math %>%
  summary()
#>
#> Call:
#> lm(formula = n_solved ~ alg + geom + trig + sv_calc + mv_calc +
#>     linalg + pr_stat + disc + logic, data = df.edu)
```

```
#>
#> Residuals:
#>     Min      1Q  Median      3Q     Max
#> -44.451 -13.674  -2.451  15.311  39.385
#>
#> Coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)  38.7799     3.0560  12.690  < 2e-16 ***
#> alg           9.6712     3.5019   2.762  0.00616 **
#> geom          9.8356     3.0426   3.233  0.00138 **
#> trig          3.1513     2.8563   1.103  0.27092
#> sv_calc       4.5978     3.4798   1.321  0.18756
#> mv_calc      -1.0651     3.9988  -0.266  0.79018
#> linalg        0.4345     3.2316   0.134  0.89314
#> pr_stat       2.6101     2.4217   1.078  0.28213
#> disc          6.4102     4.6641   1.374  0.17050
#> logic        -1.3061     3.4974  -0.373  0.70912
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 17.96 on 261 degrees of freedom
#> Multiple R-squared:  0.1922, Adjusted R-squared:  0.1643
#> F-statistic: 6.899 on 9 and 261 DF,  p-value: 6.499e-09
```

3.1.3 Algebra and Geometry

```
lm.ag = lm(n_solved ~ alg + geom, df.edu)
lm.ag %>%
  summary()
#>
#> Call:
#> lm(formula = n_solved ~ alg + geom, data = df.edu)
#>
#> Residuals:
#>     Min      1Q  Median      3Q     Max
#> -48.443 -15.221  -2.557  17.443  38.115
#>
#> Coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   40.771      2.948  13.832  < 2e-16 ***
#> alg            9.114      3.515   2.593     0.01 *
#> geom          12.672      2.934   4.318 2.22e-05 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 18.19 on 268 degrees of freedom
#> Multiple R-squared:  0.1491, Adjusted R-squared:  0.1427
#> F-statistic: 23.48 on 2 and 268 DF,  p-value: 4.03e-10
```

3.1.4 Education, algebra, and geometry

```
lm.eag = lm(n_solved ~ education + alg + geom, df.edu)
lm.eag %>%
  summary()
#>
#> Call:
#> lm(formula = n_solved ~ education + alg + geom, data = df.edu)
#>
#> Residuals:
#>     Min      1Q  Median      3Q     Max
#> -48.679 -14.311  -1.787  15.687  38.329
#>
#> Coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)  26.1242     7.8991   3.307 0.001072 **
#> education     1.0019     0.5017   1.997 0.046863 *
#> alg           9.6401     3.5055   2.750 0.006366 **
#> geom         11.5246     2.9743   3.875 0.000134 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 18.09 on 267 degrees of freedom
#> Multiple R-squared:  0.1616, Adjusted R-squared:  0.1522
#> F-statistic: 17.15 on 3 and 267 DF,  p-value: 3.234e-10
```

3.1.5 Education and math

```
lm.all = lm(n_solved ~ education + alg + geom + trig + sv_calc +
              mv_calc + linalg + pr_stat + disc + logic,
            df.edu)
lm.all %>%
  summary()
#>
#> Call:
#> lm(formula = n_solved ~ education + alg + geom + trig + sv_calc +
#>     mv_calc + linalg + pr_stat + disc + logic, data = df.edu)
#>
#> Residuals:
#>     Min      1Q  Median      3Q     Max
#> -43.342 -13.935  -1.897  15.224  41.030
#>
#> Coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)  30.5797     8.0532   3.797 0.000182 ***
#> education     0.5803     0.5273   1.101 0.272130
#> alg           9.9272     3.5082   2.830 0.005022 **
#> geom          9.4267     3.0640   3.077 0.002317 **
#> trig          2.6548     2.8906   0.918 0.359247
#> sv_calc       4.7275     3.4804   1.358 0.175533
#> mv_calc      -1.3774     4.0072  -0.344 0.731333
```

```
#> linalg        0.4284     3.2303   0.133 0.894590
#> pr_stat       2.1320     2.4594   0.867 0.386817
#> disc          6.2662     4.6640   1.344 0.180275
#> logic        -1.6276     3.5082  -0.464 0.643089
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 17.95 on 260 degrees of freedom
#> Multiple R-squared:  0.1959, Adjusted R-squared:  0.165
#> F-statistic: 6.335 on 10 and 260 DF,  p-value: 1.053e-08
```

3.2 Chi-sq Test

```
df.alggeom = df.edu %>%
  filter(in_qrate) %>%
  mutate(math = ifelse(alg + geom == 2, 'both', 'neither'),
         math = ifelse(alg + geom == 1, 'one', math)) %>%
  select(subject_id, solver, math)

df.alggeom %>%
  group_by(solver, math) %>%
  summarize(n = n()) %>%
  pivot_wider(names_from = math,
              values_from = n) %>%
  select(solver, both, one, neither) %>%
  replace_na(list(neither = 0)) %>%
  knitr::kable()
```

solver	both	one	neither
0	49	19	16
1	74	10	0

```
chisq.test(df.alggeom$solver, df.alggeom$math)
#>
#>  Pearson's Chi-squared test
#>
#> data:  df.alggeom$solver and df.alggeom$math
#> X-squared = 23.874, df = 2, p-value = 6.542e-06
```