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) %>%
```

```r
  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.

```r
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

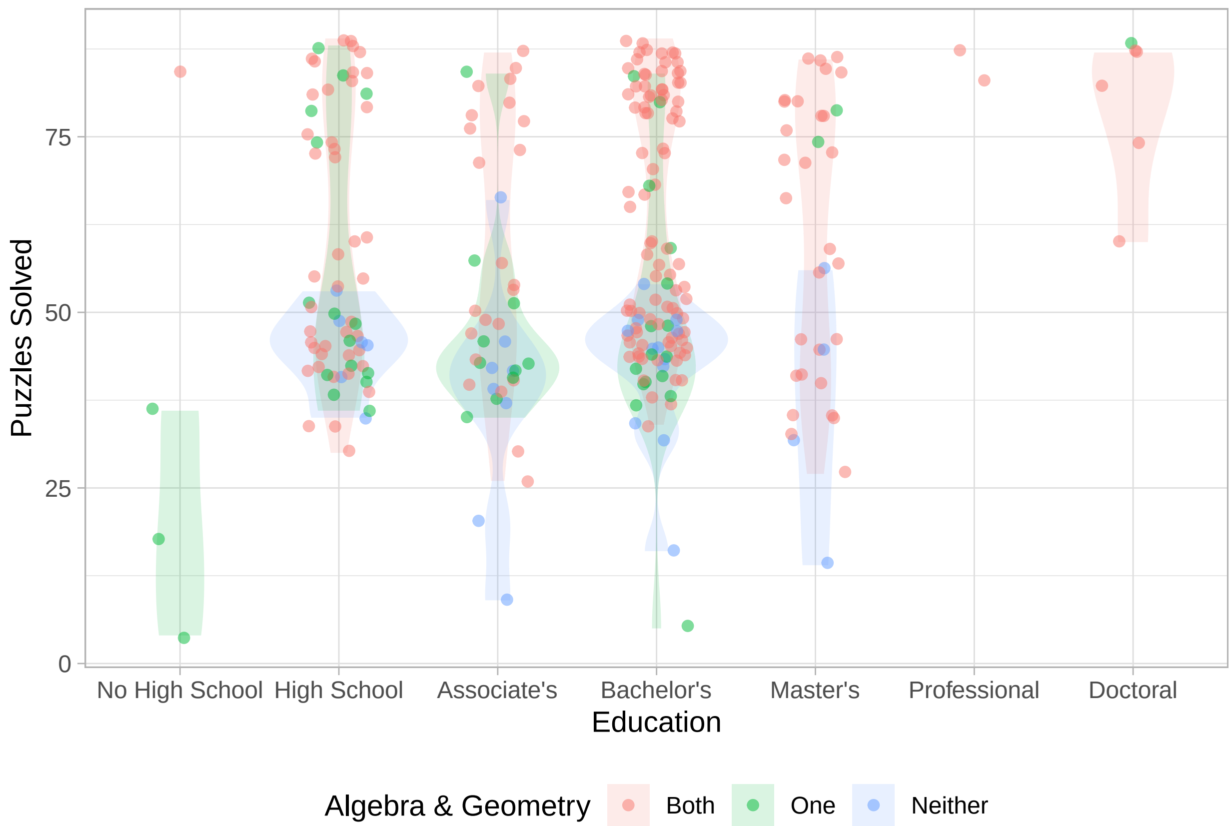
```r
df = df.data %>%
  filter(correct_actual)

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

3.1 Plot

```
df.edu %>%
  ggplot(aes(x = edu_level, y = n_solved, color = alg_geom, fill = alg_geom)) +
  geom_violin(position = 'identity',
              color = NA,
              alpha = .15) +
  geom_point(position = position_jitter(width = .2),
             alpha = .5) +
  labs(x = "Education",
       y = "Puzzles Solved",
       color = "Algebra & Geometry",
       fill = "Algebra & Geometry") +
  theme(legend.position = "bottom")
```



3.2 Regressions

3.2.1 Education

```r
lm.edu = brm(n_solved ~ education,
             data = df.edu,
             save_pars = save_pars(all=TRUE),
             seed = 0,
             cores = 4,
             refresh = 0,
             file = 'cache/education/edu')

lm.edu %>%
  summary()

bayes_R2(lm.edu)
#>  Family: gaussian
#>   Links: mu = identity; sigma = identity
#> Formula: n_solved ~ education
#>    Data: df.edu (Number of observations: 271)
#> Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
#>          total post-warmup samples = 4000
#>
#> Population-Level Effects:
#>           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> Intercept    36.13      7.92    20.51    51.17 1.00     4422     3049
#> education     1.46      0.52     0.45     2.45 1.00     4429     2885
#>
#> Family Specific Parameters:
#>       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> sigma    19.43      0.84    17.91    21.17 1.00     4257     2844
#>
#> Samples were drawn using sampling(NUTS). For each parameter, Bulk_ESS
#> and Tail_ESS are effective sample size measures, and Rhat is the potential
#> scale reduction factor on split chains (at convergence, Rhat = 1).
#>      Estimate Est.Error        Q2.5      Q97.5
#> R2 0.03175403 0.0198214 0.002755936 0.07670891
```

3.2.2 Math

```r
lm.math = brm(n_solved ~ alg + geom + trig + sv_calc +
               mv_calc + linalg + pr_stat + disc + logic,
             data = df.edu,
             save_pars = save_pars(all=TRUE),
             seed = 0,
             cores = 4,
             refresh = 0,
             file = 'cache/education/math')

lm.math %>%
  summary()

bayes_R2(lm.math)
#>  Family: gaussian
#>   Links: mu = identity; sigma = identity
```

```
#> Formula: n_solved ~ alg + geom + trig + sv_calc + mv_calc + linalg + pr_stat + disc + logic
#>    Data: df.edu (Number of observations: 271)
#> Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
#>          total post-warmup samples = 4000
#>
#> Population-Level Effects:
#>           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> Intercept    38.76      3.20    32.51    44.88 1.00     7108     3390
#> alg           9.68      3.56     2.87    16.78 1.00     4740     3315
#> geom          9.81      3.16     3.78    16.03 1.00     4332     2962
#> trig          3.14      2.82    -2.37     8.72 1.00     5065     3234
#> sv_calc       4.62      3.54    -2.16    11.54 1.00     4282     3112
#> mv_calc      -1.01      3.96    -8.98     6.61 1.00     4129     3066
#> linalg        0.39      3.14    -5.58     6.54 1.00     5433     3168
#> pr_stat       2.62      2.42    -2.27     7.24 1.00     5178     2976
#> disc          6.36      4.70    -3.14    15.40 1.00     5258     2824
#> logic        -1.28      3.54    -8.29     5.73 1.00     5415     2857
#>
#> Family Specific Parameters:
#>       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> sigma    18.03      0.82    16.50    19.78 1.00     5666     3061
#>
#> Samples were drawn using sampling(NUTS). For each parameter, Bulk_ESS
#> and Tail_ESS are effective sample size measures, and Rhat is the potential
#> scale reduction factor on split chains (at convergence, Rhat = 1).
#>     Estimate  Est.Error      Q2.5     Q97.5
#> R2 0.2065324 0.03845325 0.1317709 0.2792023
```

3.2.3 Algebra and Geometry

```
lm.ag = brm(n_solved ~ alg + geom,
            data = df.edu,
            save_pars = save_pars(all=TRUE),
            seed = 0,
            cores = 4,
            refresh = 0,
            file = 'cache/education/ag')

lm.ag %>%
  summary()

bayes_R2(lm.ag)
#>  Family: gaussian
#>   Links: mu = identity; sigma = identity
#> Formula: n_solved ~ alg + geom
#>    Data: df.edu (Number of observations: 271)
#> Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
#>          total post-warmup samples = 4000
#>
#> Population-Level Effects:
#>           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> Intercept    40.72      2.91    34.97    46.40 1.00     5095     3593
```

```
#> alg           9.13      3.50     2.15    16.07 1.00     3437     3022
#> geom         12.70      2.93     7.07    18.69 1.00     3046     2869
#>
#> Family Specific Parameters:
#>       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> sigma    18.24      0.79    16.79    19.86 1.00     3705     2974
#>
#> Samples were drawn using sampling(NUTS). For each parameter, Bulk_ESS
#> and Tail_ESS are effective sample size measures, and Rhat is the potential
#> scale reduction factor on split chains (at convergence, Rhat = 1).
#>     Estimate  Est.Error       Q2.5     Q97.5
#> R2 0.1524159 0.03567152 0.08557971 0.2258986
```

3.2.4 Education, algebra, and geometry

```
lm.eag = brm(n_solved ~ education + alg + geom,
            data = df.edu,
            save_pars = save_pars(all=TRUE),
            seed = 0,
            cores = 4,
            refresh = 0,
            file = 'cache/education/eag')

lm.eag %>%
  summary()

bayes_R2(lm.eag)
#>  Family: gaussian
#>   Links: mu = identity; sigma = identity
#> Formula: n_solved ~ education + alg + geom
#>    Data: df.edu (Number of observations: 271)
#> Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
#>          total post-warmup samples = 4000
#>
#> Population-Level Effects:
#>           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> Intercept    25.48      7.61    10.40    39.89 1.00     4329     3363
#> education     1.05      0.48     0.10     1.99 1.00     4081     3192
#> alg           9.59      3.53     2.57    16.40 1.00     3726     3048
#> geom         11.48      2.99     5.61    17.33 1.00     3602     3234
#>
#> Family Specific Parameters:
#>       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> sigma    18.11      0.77    16.67    19.69 1.00     4533     2836
#>
#> Samples were drawn using sampling(NUTS). For each parameter, Bulk_ESS
#> and Tail_ESS are effective sample size measures, and Rhat is the potential
#> scale reduction factor on split chains (at convergence, Rhat = 1).
#>     Estimate  Est.Error       Q2.5     Q97.5
#> R2 0.1675445 0.03656471 0.09834484 0.2396844
```

3.2.5 Education and math

```
lm.all = brm(n_solved ~ education + alg + geom + trig + sv_calc +
               mv_calc + linalg + pr_stat + disc + logic,
            data = df.edu,
            save_pars = save_pars(all=TRUE),
            seed = 0,
            cores = 4,
            refresh = 0,
            file = 'cache/education/all')

lm.all %>%
  summary()

bayes_R2(lm.all)
#>  Family: gaussian
#>   Links: mu = identity; sigma = identity
#> Formula: n_solved ~ education + alg + geom + trig + sv_calc + mv_calc + linalg + pr_stat + disc + lo
#>    Data: df.edu (Number of observations: 271)
#> Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
#>          total post-warmup samples = 4000
#>
#> Population-Level Effects:
#>           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> Intercept    30.14      7.68    15.36    45.15 1.00     5366     3101
#> education     0.61      0.50    -0.39     1.56 1.00     4793     3236
#> alg           9.93      3.53     2.98    16.81 1.00     4347     3416
#> geom          9.40      3.09     3.39    15.53 1.00     4275     3276
#> trig          2.70      2.99    -3.22     8.55 1.00     4809     3165
#> sv_calc       4.64      3.46    -2.18    11.55 1.00     3892     3363
#> mv_calc      -1.28      4.00    -8.87     6.83 1.00     3886     3460
#> linalg        0.40      3.24    -6.02     6.69 1.00     4376     2902
#> pr_stat       2.10      2.50    -2.73     6.96 1.00     4232     3181
#> disc          6.20      4.62    -3.09    15.17 1.00     4742     3062
#> logic        -1.62      3.56    -8.52     5.51 1.00     5005     2933
#>
#> Family Specific Parameters:
#>       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> sigma    18.02      0.81    16.50    19.70 1.00     5442     2959
#>
#> Samples were drawn using sampling(NUTS). For each parameter, Bulk_ESS
#> and Tail_ESS are effective sample size measures, and Rhat is the potential
#> scale reduction factor on split chains (at convergence, Rhat = 1).
#>     Estimate  Est.Error      Q2.5    Q97.5
#> R2 0.2126737 0.03768455 0.1393256 0.284608
```

3.2.6 Compare models

Education does not substantially improve the models when all the math courses are accounted for. It is more substantial (moderately strong evidence) when only considering algebra and geometry.

```r
bayes_factor(lm.all, lm.math, silent=T)
bayes_factor(lm.eag, lm.ag, silent=T)
#> Estimated Bayes factor in favor of lm.all over lm.math: 2.62438
#> Estimated Bayes factor in favor of lm.eag over lm.ag: 11.51723
```

Conversely, considering algebra and geometry is extremely helpful compared to just years of education.

```r
bayes_factor(lm.eag, lm.edu, silent=T)
#> Estimated Bayes factor in favor of lm.eag over lm.edu: 28063832862.90940
```

The BF shows extremely high favor towards including other math courses. It certainly adds predictive power collectively, but the significance of individual terms is unclear.

```r
bayes_factor(lm.all, lm.eag, silent=T)
bayes_factor(lm.math, lm.ag, silent=T)
#> Estimated Bayes factor in favor of lm.all over lm.eag: 452583435.98694
#> Estimated Bayes factor in favor of lm.math over lm.ag: 1990704576.17459
```

3.3 Chi-sq Test

```r
df.edu %>%
  filter(in_qrate) %>%
  group_by(solver, alg_geom) %>%
  summarize(n = n()) %>%
  pivot_wider(names_from = alg_geom,
              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

```r
df = df.edu %>%
  filter(in_qrate)
chisq.test(df$solver, df$alg_geom)
#>
#>  Pearson's Chi-squared test
#>
#> data:  df$solver and df$alg_geom
#> X-squared = 23.874, df = 2, p-value = 6.542e-06
```