ML5 Risen & Gilovich Data Preparation

Contact: Maya Mathur (mmathur@stanford.edu)

June 4, 2017

Site-Level Data Preparation

Overview: A central script is called to prep each site's data automatically (merging various columns, producing standardized names, and excluding subjects per the a priori attention criterion) while outputting results of sanity checks and producing a within-site interaction plot. The script writes separate files with each site's prepped data. Lastly, all sites' prepped data are stitched into a single analysis dataset.

Plots show standard boxplots (quartiles) with lines overlaying group means.

```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr
```

```
## Conflicts with tidy packages ----------------------------------------------
```

```
## filter(): dplyr, stats
## lag():    dplyr, stats
```

Brigham Young

First manually add the `had.read` and `load` variables:

```
source("data_prep_functions.R")

d = read.csv("data/raw/Brigham Young/raw_byu.csv", header = TRUE)

# remove additional header rows
d = d[-c(1:2), ]

library(dplyr)

# variable names here are sometimes exactly the same rename
# them to avoid problems
names(d)[18:(length(names(d)) - 1)] = c("lkl1", "imp1", "bad1",
    "lkl2", "endnum1", "eff.split1", "count.hard1", "count.eff1",
    "imp2", "bad2", "lkl3", "endnum2", "eff.split2", "count.hard2",
    "count.eff2", "imp3", "bad3", "lkl4", "imp4", "bad4")

# make had.read variable
d$had.read = NA
d$had.read[!is.na(as.numeric(as.character(d$lkl1))) | !is.na(as.numeric(as.character(d$lkl3)))] = 0

d$had.read[!is.na(as.numeric(as.character(d$lkl2))) | !is.na(as.numeric(as.character(d$lkl4)))] = 1

# merge end-number columns warning about 'NAs introduced by
```

```
# coercion', but is correct
d$end.num = coalesce(as.numeric(as.character(d$endnum1)), as.numeric(as.character(d$endnum2)))

# make load variable based on end-number column
d$load = 0
d$load[!is.na(as.numeric(as.character(d$end.num)))] = 1


# merge effort-split columns warning about 'NAs introduced by
# coercion', but is correct
d$eff.split = coalesce(as.numeric(as.character(d$eff.split1)),
    as.numeric(as.character(d$eff.split2)))

# merge badness columns warning about 'NAs introduced by
# coercion', but is correct
d$badness = coalesce(as.numeric(as.character(d$bad1)), as.numeric(as.character(d$bad2)),
    as.numeric(as.character(d$bad3)), as.numeric(as.character(d$bad4)))

# merge importance columns warning about 'NAs introduced by
# coercion', but is correct
d$importance = coalesce(as.numeric(as.character(d$imp1)), as.numeric(as.character(d$imp2)),
    as.numeric(as.character(d$imp3)), as.numeric(as.character(d$imp4)))

# merge counting effort columns warning about 'NAs introduced
# by coercion', but is correct
d$count.eff = coalesce(as.numeric(as.character(d$count.eff1)),
    as.numeric(as.character(d$count.eff2)))

# merge counting effort columns warning about 'NAs introduced
# by coercion', but is correct
d$count.hard = coalesce(as.numeric(as.character(d$count.hard1)),
    as.numeric(as.character(d$count.hard2)))

write.csv(d, "data/raw/Brigham Young/manualprep_byu.csv")
```

Automatic data prep:

```
start.path = "data/raw/Brigham Young/manualprep_byu.csv"
end.path = "data/prepped"

prep_site_data(start.path = start.path, end.path = end.path,
    lkl.names = c("lkl1", "lkl2", "lkl3", "lkl4"), had.read.name = "had.read",
    load.name = "load", end.num.name = "end.num", eff.split.name = "eff.split",
    count.eff.name = "count.eff", count.hard.name = "count.hard",
    badness.name = "badness", importance.name = "importance",
    .site.name = "BYUI", .group = "c.dissimilar", .n.extra.header.rows = 0)
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   StartDate = col_character(),
##   EndDate = col_character(),
##   Status = col_character(),
```

```
##   IPAddress = col_character(),
##   Finished = col_logical(),
##   RecordedDate = col_character(),
##   ResponseId = col_character(),
##   RecipientLastName = col_character(),
##   RecipientFirstName = col_character(),
##   RecipientEmail = col_character(),
##   ExternalReference = col_character(),
##   LocationLatitude = col_double(),
##   LocationLongitude = col_double(),
##   DistributionChannel = col_character(),
##   Q12 = col_character(),
##   Q6...Topics = col_character()
## )
## See spec(...) for full column specifications.
##
##
## No extra header rows to delete.
##
## Rows in raw data = 90
##
## Head of skinny dataset before exclusions:
## # A tibble: 6 x 12
##      id .site.name       .group had.read  load   lkl eff.split count.eff
##   <int>      <chr>        <chr>    <dbl> <dbl> <dbl>     <dbl>     <dbl>
## 1     1       BYUI c.dissimilar        0     1     6         2         5
## 2     2       BYUI c.dissimilar        1     0     3        NA        NA
## 3     3       BYUI c.dissimilar        0     0     6        NA        NA
## 4     4       BYUI c.dissimilar        1     0     1        NA        NA
## 5     5       BYUI c.dissimilar        1     1     1         1        10
## 6     6       BYUI c.dissimilar        0     1     1         1         9
## # ... with 4 more variables: count.hard <dbl>, badness <dbl>,
## #   importance <dbl>, end.num <dbl>
##
##
## Subjects with missing had.read, load, or lkl:
##
## Subjects with load==1 but missing eff.split, count.eff, or count.hard:
##
## Bad subjects (failed to follow instructions): 11 17 19 35 81 85
##
## Final n = 84
##
## MARGINAL MEANS AND SDs FOR ANALYSIS AUDIT
##                      Overall
##   n                    84
##   load (mean (sd))  0.45 (0.50)
##   tempt (mean (sd)) 0.52 (0.50)
##   lkl (mean (sd))   3.27 (2.18)
```

Eotvos Lorand

First manually exclude 7 subjects who may have completed the experiment twice (note: these 7 are on top of any "bad subjects" excluded by the prep script):

```
d = read_csv("data/raw/Eotvos Lorand/raw_eotvos.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
d = d[!d$ResponseId %in% c("R_1H76w6p0V7mQW3T", "R_12mU6WnhCkSe5LH",
    "R_3ReInhemqnhITwd", "R_2tqY0auQ9EUmWCf", "R_Z4tlLy5HCyAXhRf",
    "R_28BCOFVHOcgVquF", "R_1LwH5vC3QBq4uFG"), ]

write_csv(d, "data/raw/Eotvos Lorand/manualprep_eotvos.csv")
```

Automatic data prep:

```
start.path = "data/raw/Eotvos Lorand/manualprep_eotvos.csv"
end.path = "data/prepped"

prep_site_data(start.path = start.path, end.path = end.path,
    lkl.names = c("L1.R1.scenario_1", "L1.R0.scenario_1", "L0.R1.text_1",
        "L0.R0.text_1"), had.read.name = "had.read", load.name = "load",
    end.num.name = "Q28", eff.split.name = "Q29_1", count.eff.name = "Q22_1",
    count.hard.name = "Q21_1", badness.name = "Q14_1", importance.name = "Q26_1",
    .site.name = "Eotvos", .group = "c.dissimilar", .n.extra.header.rows = 2)
```

```
## Parsed with column specification:
## cols(
##   .default = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
##
##
## Extra header rows to delete (first 3 cols):
## # A tibble: 2 x 3
##                        StartDate                       EndDate
##                            <chr>                         <chr>
## 1                     Start Date                      End Date
## 2 "{\"ImportId\":\"startDate\"}" "{\"ImportId\":\"endDate\"}"
## # ... with 1 more variables: Status <chr>
##
##
## First row of real data:
## # A tibble: 1 x 31
##             StartDate             EndDate Status      IPAddress Progress
##                 <chr>               <chr>  <chr>          <chr>    <chr>
## 1 2017-02-28 10:26:32 2017-02-28 10:27:56      0 157.181.60.140      100
## # ... with 26 more variables: `Duration (in seconds)` <chr>,
## #   Finished <chr>, RecordedDate <chr>, ResponseId <chr>,
## #   RecipientLastName <chr>, RecipientFirstName <chr>,
```

```
## #   RecipientEmail <chr>, ExternalReference <chr>, LocationLatitude <chr>,
## #   LocationLongitude <chr>, DistributionChannel <chr>, Q15 <chr>,
## #   Q16 <chr>, L1.R1.scenario_1 <chr>, L1.R0.scenario_1 <chr>, Q28 <chr>,
## #   Q29_1 <chr>, Q21_1 <chr>, Q22_1 <chr>, L0.R1.text_1 <chr>,
## #   L0.R0.text_1 <chr>, Q26_1 <chr>, Q14_1 <chr>, mTurkCode <chr>,
## #   load <chr>, had.read <chr>
##
## Rows in raw data = 291
##
## Head of skinny dataset before exclusions:
## # A tibble: 6 x 12
##      id .site.name       .group had.read  load   lkl eff.split count.eff
##   <int>      <chr>        <chr>    <dbl> <dbl> <dbl>     <dbl>     <dbl>
## 1     1     Eotvos c.dissimilar        1     0     1        NA        NA
## 2     2     Eotvos c.dissimilar        0     0     3        NA        NA
## 3     3     Eotvos c.dissimilar        1     0     5        NA        NA
## 4     4     Eotvos c.dissimilar        1     0     2        NA        NA
## 5     5     Eotvos c.dissimilar        0     0     4        NA        NA
## 6     6     Eotvos c.dissimilar        1     1     3         4         7
## # ... with 4 more variables: count.hard <dbl>, badness <dbl>,
## #   importance <dbl>, end.num <dbl>
##
##
## Subjects with missing had.read, load, or lkl:
##
## Subjects with load==1 but missing eff.split, count.eff, or count.hard: 170
##
## Bad subjects (failed to follow instructions): 11 28 84 140 169 193 266
##
## Final n = 284
##
## MARGINAL MEANS AND SDs FOR ANALYSIS AUDIT
##                      Overall
##   n                   284
##   load (mean (sd))  0.49 (0.50)
##   tempt (mean (sd)) 0.50 (0.50)
##   lkl (mean (sd))   3.58 (2.17)
```

KU Leuven

```
start.path = "data/raw/KU Leuven/raw_kul.csv"
end.path = "data/prepped"

prep_site_data(start.path = start.path, end.path = end.path,
    lkl.names = c("L1.R1.scenario_1", "L1.R0.scenario_1", "L0.R1.text_1",
        "L0.R0.text_1"), had.read.name = "had.read", load.name = "load",
    end.num.name = "Q28", eff.split.name = "Q29_1", count.eff.name = "Q22_1",
    count.hard.name = "Q21_1", badness.name = "Q14_1", importance.name = "Q26_1",
    .site.name = "KUL", .group = "c.dissimilar", .n.extra.header.rows = 1)
```

```
## Parsed with column specification:
## cols(
##   .default = col_character()
```

```
## )
## See spec(...) for full column specifications.
##
##
## Extra header rows to delete (first 3 cols):
## # A tibble: 1 x 3
##          V1          V2    V3
##       <chr>       <chr> <chr>
## 1 ResponseID ResponseSet  Name
##
##
## First row of real data:
## # A tibble: 1 x 27
##                  V1                   V2        V3    V4    V5
##               <chr>                <chr>     <chr> <chr> <chr>
## 1 R_1FPzjjMqHZwdcbk Default Response Set Anonymous  <NA>  <NA>
## # ... with 22 more variables: V6 <chr>, V7 <chr>, V8 <chr>, V9 <chr>,
## #   V10 <chr>, load <chr>, had.read <chr>, text <chr>,
## #   L1.R1.scenario_1 <chr>, L1.R0.scenario_1 <chr>, Q28 <chr>,
## #   Q29_1 <chr>, Q21_1 <chr>, Q22_1 <chr>, L0.R1.text_1 <chr>,
## #   L0.R0.text_1 <chr>, Q26_1 <chr>, Q14_1 <chr>, text.thanks <chr>,
## #   LocationLatitude <chr>, LocationLongitude <chr>,
## #   LocationAccuracy <chr>
##
## Rows in raw data = 127
##
## Head of skinny dataset before exclusions:
## # A tibble: 6 x 12
##      id .site.name       .group had.read  load   lkl eff.split count.eff
##   <int>      <chr>        <chr>    <dbl> <dbl> <dbl>     <dbl>     <dbl>
## 1     1        KUL c.dissimilar        1     0     2        NA        NA
## 2     2        KUL c.dissimilar        1     1     6         2         4
## 3     3        KUL c.dissimilar        0     1     2         4         9
## 4     4        KUL c.dissimilar        0     0     7        NA        NA
## 5     5        KUL c.dissimilar        1     1     2         3         9
## 6     6        KUL c.dissimilar        0     1     2         4         5
## # ... with 4 more variables: count.hard <dbl>, badness <dbl>,
## #   importance <dbl>, end.num <dbl>
##
##
## Subjects with missing had.read, load, or lkl:
##
## Subjects with load==1 but missing eff.split, count.eff, or count.hard:
##
## Bad subjects (failed to follow instructions): 2 30 34 65 71 85 89 102 106
##
## Final n = 118
##
## MARGINAL MEANS AND SDs FOR ANALYSIS AUDIT
##                     Overall
##   n                  118
##   load (mean (sd))  0.47 (0.50)
##   tempt (mean (sd)) 0.50 (0.50)
```

```
##   lkl (mean (sd))   2.11 (1.53)
```

University of Porto

```
start.path = "data/raw/PUC Rio/raw_puc.csv"
end.path = "data/prepped"

prep_site_data(start.path = start.path, end.path = end.path,
    lkl.names = c("L1.R1.scenario_1", "L1.R0.scenario_1", "L0.R1.text_1",
        "L0.R0.text_1"), had.read.name = "had.read", load.name = "load",
    end.num.name = "Q28", eff.split.name = "Q29_1", count.eff.name = "Q22_1",
    count.hard.name = "Q21_1", badness.name = "Q14_1", importance.name = "Q26_1",
    .site.name = "UP", .group = "c.dissimilar", .n.extra.header.rows = 1)
```

```
## Warning: Missing column names filled in: 'X31' [31]
```

```
## Parsed with column specification:
## cols(
##   .default = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
##
##
## Extra header rows to delete (first 3 cols):
## # A tibble: 1 x 3
##           V1          V2    V3
##        <chr>       <chr> <chr>
## 1 ResponseID ResponseSet  Name
##
##
## First row of real data:
## # A tibble: 1 x 31
##                  V1                  V2        V3    V4    V5
##               <chr>               <chr>     <chr> <chr> <chr>
## 1 R_2tnZTPfaEpn2cC2 Default Response Set Anonymous  <NA>  <NA>
## # ... with 26 more variables: V6 <chr>, V7 <chr>, V8 <chr>, V9 <chr>,
## #   V10 <chr>, mTurkCode <chr>, load <chr>, had.read <chr>, Q24 <chr>,
## #   Q13 <chr>, text <chr>, L1.R1.scenario_1 <chr>, L1.R0.scenario_1 <chr>,
## #   Q28 <chr>, Q29_1 <chr>, Q21_1 <chr>, Q22_1 <chr>, L0.R1.text_1 <chr>,
## #   L0.R0.text_1 <chr>, Q26_1 <chr>, Q14_1 <chr>, text.thanks <chr>,
## #   LocationLatitude <chr>, LocationLongitude <chr>,
## #   LocationAccuracy <chr>, X31 <chr>
##
## Rows in raw data = 106
##
## Head of skinny dataset before exclusions:
## # A tibble: 6 x 12
##      id .site.name       .group had.read  load   lkl eff.split count.eff
##   <int>      <chr>        <chr>    <dbl> <dbl> <dbl>     <dbl>     <dbl>
## 1     1         UP c.dissimilar        0     0    NA        NA        NA
## 2     2         UP c.dissimilar        0     0    NA        NA        NA
## 3     3         UP c.dissimilar        0     0     6        NA        NA
```

```
## 4     4         UP c.dissimilar        1     0     4        NA        NA
## 5     5         UP c.dissimilar        1     1     2         3         4
## 6     6         UP c.dissimilar        0     1     5         2         6
## # ... with 4 more variables: count.hard <dbl>, badness <dbl>,
## #   importance <dbl>, end.num <dbl>
##
##
## Subjects with missing had.read, load, or lkl: 1 2
##
## Subjects with load==1 but missing eff.split, count.eff, or count.hard: 15 106
##
## Bad subjects (failed to follow instructions): 13 14 25 34 35 41 52 54 58 69 85 89 102
##
## Final n = 91
##
## MARGINAL MEANS AND SDs FOR ANALYSIS AUDIT
##                      Overall
##   n                    91
##   load (mean (sd))  0.43 (0.50)
##   tempt (mean (sd)) 0.47 (0.50)
##   lkl (mean (sd))   4.11 (2.09)
```

Rose-Hulman IT

```
start.path = "data/raw/Rose-Hulman IT/raw_rose.csv"
end.path = "data/prepped"

d = prep_site_data(start.path = start.path, end.path = end.path,
    lkl.names = c("likelihood_1", "likelihood_1_1", "likelihood_1_2",
        "likelihood_1_3"), had.read.name = "had.read", load.name = "load",
    end.num.name = "end.num", eff.split.name = "effort.split_1",
    count.eff.name = "effort.count_1", count.hard.name = "difficulty_1",
    badness.name = "negativity_1", importance.name = "academic.pressure_1",
    .site.name = "RHIT", .group = "c.dissimilar", .n.extra.header.rows = 2)
```

```
## Warning: Duplicated column names deduplicated: 'likelihood_1' =>
## 'likelihood_1_1' [18], 'likelihood_1' => 'likelihood_1_2' [23],
## 'likelihood_1' => 'likelihood_1_3' [24]
```

```
## Parsed with column specification:
## cols(
##   .default = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
##
##
## Extra header rows to delete (first 3 cols):
## # A tibble: 2 x 3
##                       StartDate                     EndDate
##                           <chr>                       <chr>
## 1                    Start Date                    End Date
## 2 "{\"ImportId\":\"startDate\"}" "{\"ImportId\":\"endDate\"}"
```

```
## # ... with 1 more variables: Status <chr>
##
##
## First row of real data:
## # A tibble: 1 x 29
##             StartDate             EndDate     Status       IPAddress
##                 <chr>               <chr>      <chr>           <chr>
## 1 2017-01-05 18:49:40 2017-01-05 18:51:42 IP Address 137.112.236.167
## # ... with 25 more variables: Progress <chr>, `Duration (in
## #   seconds)` <chr>, Finished <chr>, RecordedDate <chr>, ResponseId <chr>,
## #   RecipientLastName <chr>, RecipientFirstName <chr>,
## #   RecipientEmail <chr>, ExternalReference <chr>, LocationLatitude <chr>,
## #   LocationLongitude <chr>, DistributionChannel <chr>,
## #   likelihood_1 <chr>, likelihood_1_1 <chr>, end.num <chr>,
## #   effort.split_1 <chr>, difficulty_1 <chr>, effort.count_1 <chr>,
## #   likelihood_1_2 <chr>, likelihood_1_3 <chr>, academic.pressure_1 <chr>,
## #   negativity_1 <chr>, mTurkCode <chr>, load <chr>, had.read <chr>
##
## Rows in raw data = 58
##
## Head of skinny dataset before exclusions:
## # A tibble: 6 x 12
##      id .site.name       .group had.read  load   lkl eff.split count.eff
##   <int>      <chr>        <chr>    <dbl> <dbl> <dbl>     <dbl>     <dbl>
## 1     1       RHIT c.dissimilar        0     1     8         6         1
## 2     2       RHIT c.dissimilar        0     0    10        NA        NA
## 3     3       RHIT c.dissimilar        1     0     1        NA        NA
## 4     4       RHIT c.dissimilar        1     1     3         3         6
## 5     5       RHIT c.dissimilar        0     0     6        NA        NA
## 6     6       RHIT c.dissimilar        0     1     3         3         5
## # ... with 4 more variables: count.hard <dbl>, badness <dbl>,
## #   importance <dbl>, end.num <dbl>
##
##
## Subjects with missing had.read, load, or lkl:
##
## Subjects with load==1 but missing eff.split, count.eff, or count.hard: 15
##
## Bad subjects (failed to follow instructions): 32 33
##
## Final n = 56
##
## MARGINAL MEANS AND SDs FOR ANALYSIS AUDIT
##                      Overall
##   n                    56
##   load (mean (sd))  0.48 (0.50)
##   tempt (mean (sd)) 0.54 (0.50)
##   lkl (mean (sd))   3.34 (2.33)
```

Stanford

```
start.path = "data/raw/Stanford/raw_stanford.csv"
end.path = "data/prepped"

prep_site_data(start.path = start.path, end.path = end.path,
    lkl.names = c("likelihood_1", "likelihood_1_1", "likelihood_1_2",
        "likelihood_1_3"), had.read.name = "had.read", load.name = "load",
    end.num.name = "end.num", eff.split.name = "effort.split_1",
    count.eff.name = "effort.count_1", count.hard.name = "difficulty_1",
    badness.name = "negativity_1", importance.name = "academic.pressure_1",
    .site.name = "Stanford", .group = "b.similar", .n.extra.header.rows = 2)
```

```
## Warning: Duplicated column names deduplicated: 'likelihood_1' =>
## 'likelihood_1_1' [18], 'likelihood_1' => 'likelihood_1_2' [23],
## 'likelihood_1' => 'likelihood_1_3' [24]
```

```
## Parsed with column specification:
## cols(
##   .default = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
##
##
## Extra header rows to delete (first 3 cols):
## # A tibble: 2 x 3
##                        StartDate                       EndDate
##                            <chr>                         <chr>
## 1                     Start Date                      End Date
## 2 "{\"ImportId\":\"startDate\"}" "{\"ImportId\":\"endDate\"}"
## # ... with 1 more variables: Status <chr>
##
##
## First row of real data:
## # A tibble: 1 x 29
##             StartDate             EndDate     Status     IPAddress
##                 <chr>               <chr>      <chr>         <chr>
## 1 2016-11-28 09:34:37 2016-11-28 09:36:33 IP Address 68.65.174.224
## # ... with 25 more variables: Progress <chr>, `Duration (in
## #   seconds)` <chr>, Finished <chr>, RecordedDate <chr>, ResponseId <chr>,
## #   RecipientLastName <chr>, RecipientFirstName <chr>,
## #   RecipientEmail <chr>, ExternalReference <chr>, LocationLatitude <chr>,
## #   LocationLongitude <chr>, DistributionChannel <chr>,
## #   likelihood_1 <chr>, likelihood_1_1 <chr>, end.num <chr>,
## #   effort.split_1 <chr>, difficulty_1 <chr>, effort.count_1 <chr>,
## #   likelihood_1_2 <chr>, likelihood_1_3 <chr>, academic.pressure_1 <chr>,
## #   negativity_1 <chr>, mTurkCode <chr>, load <chr>, had.read <chr>
##
## Rows in raw data = 74
##
## Head of skinny dataset before exclusions:
## # A tibble: 6 x 12
##      id .site.name    .group had.read  load   lkl eff.split count.eff
##   <int>      <chr>     <chr>    <dbl> <dbl> <dbl>     <dbl>     <dbl>
## 1     1   Stanford b.similar        0     0     1        NA        NA
```

```
## 2     2   Stanford b.similar        1     1     5         4         6
## 3     3   Stanford b.similar        0     0     7        NA        NA
## 4     4   Stanford b.similar        1     0     1        NA        NA
## 5     5   Stanford b.similar        0     0     3        NA        NA
## 6     6   Stanford b.similar        0     1     2         4         3
## # ... with 4 more variables: count.hard <dbl>, badness <dbl>,
## #   importance <dbl>, end.num <dbl>
##
##
## Subjects with missing had.read, load, or lkl: 10 21 31 37 38
##
## Subjects with load==1 but missing eff.split, count.eff, or count.hard: 10 21 22 31 37 38
##
## Bad subjects (failed to follow instructions): 18
##
## Final n = 68
##
## MARGINAL MEANS AND SDs FOR ANALYSIS AUDIT
##                      Overall
##   n                    68
##   load (mean (sd))  0.44 (0.50)
##   tempt (mean (sd)) 0.51 (0.50)
##   lkl (mean (sd))   2.50 (2.00)
```

University of Rhode Island

```
start.path = "data/raw/U Rhode Island/raw_uri.csv"
end.path = "data/prepped"

prep_site_data(start.path = start.path, end.path = end.path,
    lkl.names = c("L1.R1.scenario_1", "L1.R0.scenario_1", "L0.R1.text_1",
        "L0.R0.text_1"), had.read.name = "had.read", load.name = "load",
    end.num.name = "Q28", eff.split.name = "Q29_1", count.eff.name = "Q22_1",
    count.hard.name = "Q21_1", badness.name = "Q14_1", importance.name = "Q26_1",
    .site.name = "URI", .group = "c.dissimilar", .n.extra.header.rows = 2)
```

```
## Parsed with column specification:
## cols(
##   .default = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
##
##
## Extra header rows to delete (first 3 cols):
## # A tibble: 2 x 3
##                        StartDate                       EndDate
##                            <chr>                         <chr>
## 1                     Start Date                      End Date
## 2 "{\"ImportId\":\"startDate\"}" "{\"ImportId\":\"endDate\"}"
## # ... with 1 more variables: Progress <chr>
##
##
```

```
## First row of real data:
## # A tibble: 1 x 27
##       StartDate       EndDate Progress `Duration (in seconds)` Finished
##           <chr>         <chr>    <chr>                   <chr>    <chr>
## 1 2/7/2017 7:13 2/7/2017 7:22      100                     568     TRUE
## # ... with 22 more variables: RecordedDate <chr>, ResponseId <chr>,
## #   RecipientLastName <chr>, RecipientFirstName <chr>,
## #   RecipientEmail <chr>, ExternalReference <chr>, LocationLatitude <chr>,
## #   LocationLongitude <chr>, DistributionChannel <chr>,
## #   L1.R1.scenario_1 <chr>, L1.R0.scenario_1 <chr>, Q28 <chr>,
## #   Q29_1 <chr>, Q21_1 <chr>, Q22_1 <chr>, L0.R1.text_1 <chr>,
## #   L0.R0.text_1 <chr>, Q26_1 <chr>, Q14_1 <chr>, mTurkCode <chr>,
## #   load <chr>, had.read <chr>
##
## Rows in raw data = 90
##
## Head of skinny dataset before exclusions:
## # A tibble: 6 x 12
##      id .site.name       .group had.read  load   lkl eff.split count.eff
##   <int>      <chr>        <chr>    <dbl> <dbl> <dbl>     <dbl>     <dbl>
## 1     1        URI c.dissimilar        0     1     2         4         9
## 2     2        URI c.dissimilar        0     1     3         0         6
## 3     3        URI c.dissimilar        1     0     1        NA        NA
## 4     4        URI c.dissimilar        1     1     5         2         2
## 5     5        URI c.dissimilar        1     0     1        NA        NA
## 6     6        URI c.dissimilar        0     1     4         3         7
## # ... with 4 more variables: count.hard <dbl>, badness <dbl>,
## #   importance <dbl>, end.num <dbl>
##
##
## Subjects with missing had.read, load, or lkl:
##
## Subjects with load==1 but missing eff.split, count.eff, or count.hard: 30 33 44 49
##
## Bad subjects (failed to follow instructions): 2 9 14 17 44 48 49 60 61
##
## Final n = 81
##
## MARGINAL MEANS AND SDs FOR ANALYSIS AUDIT
##                      Overall
##   n                    81
##   load (mean (sd))  0.44 (0.50)
##   tempt (mean (sd)) 0.52 (0.50)
##   lkl (mean (sd))   3.21 (1.78)
```

UC Berkeley

This site used the RPP Qualtrics file instead of the updated ML5 one. The RPP file had exactly the same wording for the main questions but did not have the new "mechanistic" questions; hence all the missing data that the function complains about.

```
d = read_csv("data/raw/UC Berkeley/raw_ucb.csv")
```

```
## Warning: Missing column names filled in: 'X6' [6]
```

```
## Parsed with column specification:
## cols(
##   a = col_character(),
##   mTurkCode = col_character(),
##   V8 = col_character(),
##   V9 = col_character(),
##   V10 = col_character(),
##   X6 = col_character(),
##   `Cognitive load` = col_character(),
##   `Had read` = col_character(),
##   Q1 = col_character(),
##   Q2_1 = col_character(),
##   Q3 = col_character(),
##   Q4_1 = col_character(),
##   Q6 = col_character(),
##   Q7_1 = col_character(),
##   Q8 = col_character(),
##   Q9_1 = col_character(),
##   Q11_1 = col_character(),
##   Q14_1 = col_character(),
##   Q13 = col_character()
## )
```

```
# merge end-number columns warning about 'NAs introduced by
# coercion', but is correct
d$end.num = coalesce(as.numeric(as.character(d$Q3)), as.numeric(as.character(d$Q8)))
```

```
## Warning in eval_bare(dot$expr, dot$env): NAs introduced by coercion
```

```
## Warning in eval_bare(dot$expr, dot$env): NAs introduced by coercion
```

```
# merge effort-split columns warning about 'NAs introduced by
# coercion', but is correct
d$eff.split = coalesce(as.numeric(as.character(d$Q4_1)), as.numeric(as.character(d$Q9_1)))
```

```
## Warning in eval_bare(dot$expr, dot$env): NAs introduced by coercion
```

```
## Warning in eval_bare(dot$expr, dot$env): NAs introduced by coercion
```

```
# placeholders for vars not collected
d$badness = NA
d$importance = NA
d$count.eff = NA
d$count.hard = NA

write.csv(d, "data/raw/UC Berkeley/manualprep_ucb.csv")
```

Automatic prep:

```
start.path = "data/raw/UC Berkeley/manualprep_ucb.csv"
end.path = "data/prepped"

prep_site_data(start.path = start.path, end.path = end.path,
    lkl.names = c("Q2_1", "Q7_1", "Q11_1", "Q14_1"), had.read.name = "Had read",
    load.name = "Cognitive load", end.num.name = "end.num", eff.split.name = "eff.split",
    count.eff.name = "count.eff", count.hard.name = "count.hard",
    badness.name = "badness", importance.name = "badness", .site.name = "UCB",
```

```
    .group = "b.similar", .n.extra.header.rows = 1)
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   X1 = col_integer(),
##   end.num = col_integer(),
##   eff.split = col_integer()
## )
```

```
## See spec(...) for full column specifications.
```

```
##
##
## Extra header rows to delete (first 3 cols):
## # A tibble: 1 x 3
##      X1          a mTurkCode
##   <int>      <chr>     <chr>
## 1     1 ResponseID mTurkCode
##
##
## First row of real data:
## # A tibble: 1 x 26
##      X1                 a mTurkCode           V8           V9   V10    X6
##   <int>             <chr>     <chr>        <chr>        <chr> <chr> <chr>
## 1     2 R_1cSbvNAXaafqkde   4236033 3/1/17 15:43 3/1/17 16:13     1  <NA>
## # ... with 19 more variables: `Cognitive load` <chr>, `Had read` <chr>,
## #   Q1 <chr>, Q2_1 <chr>, Q3 <chr>, Q4_1 <chr>, Q6 <chr>, Q7_1 <chr>,
## #   Q8 <chr>, Q9_1 <chr>, Q11_1 <chr>, Q14_1 <chr>, Q13 <chr>,
## #   end.num <int>, eff.split <int>, badness <chr>, importance <chr>,
## #   count.eff <chr>, count.hard <chr>
##
## Rows in raw data = 224
##
## Head of skinny dataset before exclusions:
## # A tibble: 6 x 12
##      id .site.name    .group had.read  load   lkl eff.split count.eff
##   <int>      <chr>     <chr>    <dbl> <dbl> <dbl>     <dbl>     <dbl>
## 1     1        UCB b.similar        1     1     1         4        NA
## 2     2        UCB b.similar        0     1     7         1        NA
## 3     3        UCB b.similar        0     1     3         5        NA
## 4     4        UCB b.similar        1     0     2        NA        NA
## 5     5        UCB b.similar        1     0     6        NA        NA
## 6     6        UCB b.similar        1     0     1        NA        NA
## # ... with 4 more variables: count.hard <dbl>, badness <dbl>,
## #   importance <dbl>, end.num <dbl>
##
##
## Subjects with missing had.read, load, or lkl: 69
##
## Subjects with load==1 but missing eff.split, count.eff, or count.hard: 1 2 3 8 9 10 11 12 13 15 18 2
##
## Bad subjects (failed to follow instructions): 12 27 39 41 45 50 53 58 91 95 105 112 119 120 124 125
```

```
##
## Final n = 200
##
## MARGINAL MEANS AND SDs FOR ANALYSIS AUDIT
##                     Overall
##   n                  200
##   load (mean (sd))  0.43 (0.50)
##   tempt (mean (sd)) 0.51 (0.50)
##   lkl (mean (sd))   3.02 (2.27)
```

University of Pennsylvania

```
start.path = "data/raw/UPenn/raw_upenn.csv"
end.path = "data/prepped"

prep_site_data(start.path = start.path, end.path = end.path,
    lkl.names = c("L1.R1.scenario_1", "L1.R0.scenario_1", "L0.R1.text_1",
        "L0.R0.text_1"), had.read.name = "had.read", load.name = "load",
    end.num.name = "Q28", eff.split.name = "Q29_1", count.eff.name = "Q22_1",
    count.hard.name = "Q21_1", badness.name = "Q14_1", importance.name = "Q26_1",
    .site.name = "U Penn", .group = "b.similar", .n.extra.header.rows = 2)
```

```
## Parsed with column specification:
## cols(
##   .default = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
##
##
## Extra header rows to delete (first 3 cols):
## # A tibble: 2 x 3
##                         StartDate                       EndDate
##                             <chr>                         <chr>
## 1                      Start Date                      End Date
## 2 "{\"ImportId\":\"startDate\"}" "{\"ImportId\":\"endDate\"}"
## # ... with 1 more variables: Status <chr>
##
##
## First row of real data:
## # A tibble: 1 x 30
##         StartDate         EndDate     Status     IPAddress Progress
##             <chr>           <chr>      <chr>         <chr>    <chr>
## 1 3/20/2017 10:05 3/20/2017 10:06 IP Address 128.91.96.127      100
## # ... with 25 more variables: `Duration (in seconds)` <chr>,
## #   Finished <chr>, RecordedDate <chr>, ResponseId <chr>,
## #   RecipientLastName <chr>, RecipientFirstName <chr>,
## #   RecipientEmail <chr>, ExternalReference <chr>, LocationLatitude <chr>,
## #   LocationLongitude <chr>, DistributionChannel <chr>, Q18 <chr>,
## #   Q22 <chr>, L1.R1.scenario_1 <chr>, L1.R0.scenario_1 <chr>, Q28 <chr>,
## #   Q29_1 <chr>, Q21_1 <chr>, Q22_1 <chr>, L0.R1.text_1 <chr>,
## #   L0.R0.text_1 <chr>, Q26_1 <chr>, Q14_1 <chr>, load <chr>,
## #   had.read <chr>
```

```
##
## Rows in raw data = 359
##
## Head of skinny dataset before exclusions:
## # A tibble: 6 x 12
##      id .site.name    .group had.read  load   lkl eff.split count.eff
##   <int>      <chr>     <chr>    <dbl> <dbl> <dbl>     <dbl>     <dbl>
## 1     1     U Penn b.similar        0     0     3        NA        NA
## 2     2     U Penn b.similar        0     0     1        NA        NA
## 3     3     U Penn b.similar        0     0     3        NA        NA
## 4     4     U Penn b.similar        1     0     5        NA        NA
## 5     5     U Penn b.similar        1     0     0        NA        NA
## 6     6     U Penn b.similar        1     0     0        NA        NA
## # ... with 4 more variables: count.hard <dbl>, badness <dbl>,
## #   importance <dbl>, end.num <dbl>
##
##
## Subjects with missing had.read, load, or lkl:
##
## Subjects with load==1 but missing eff.split, count.eff, or count.hard: 85
##
## Bad subjects (failed to follow instructions): 33 45 46 48 66 100 118 120 125 135 146 169 171 193 211
##
## Final n = 335
##
## MARGINAL MEANS AND SDs FOR ANALYSIS AUDIT
##                      Overall
##   n                   335
##   load (mean (sd))  0.47 (0.50)
##   tempt (mean (sd)) 0.50 (0.50)
##   lkl (mean (sd))   2.85 (2.03)
```

Mechanical Turk

```
start.path = "data/raw/MTurk/raw_mturk.csv"
end.path = "data/prepped"

prep_site_data(start.path = start.path, end.path = end.path,
    lkl.names = c("L1.R1.scenario_1", "L1.R0.scenario_1", "L0.R1.text_1",
        "L0.R0.text_1"), had.read.name = "had.read", load.name = "load",
    end.num.name = "Q28", eff.split.name = "Q29_1", count.eff.name = "Q22_1",
    count.hard.name = "Q21_1", badness.name = "Q14_1", importance.name = "Q26_1",
    .site.name = "MTurk", .group = "a.mturk", .n.extra.header.rows = 2)
```

```
## Parsed with column specification:
## cols(
##   .default = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
##
##
## Extra header rows to delete (first 3 cols):
```

```
## # A tibble: 2 x 3
##                         StartDate                         EndDate
##                             <chr>                           <chr>
## 1                      Start Date                        End Date
## 2 "{\"ImportId\":\"startDate\"}" "{\"ImportId\":\"endDate\"}"
## # ... with 1 more variables: Status <chr>
##
##
## First row of real data:
## # A tibble: 1 x 33
##      StartDate      EndDate     Status   IPAddress Progress
##          <chr>        <chr>      <chr>       <chr>    <chr>
## 1 6/3/17 15:19 6/3/17 15:20 IP Address 73.0.20.244      100
## # ... with 28 more variables: `Duration (in seconds)` <chr>,
## #   Finished <chr>, RecordedDate <chr>, ResponseId <chr>,
## #   RecipientLastName <chr>, RecipientFirstName <chr>,
## #   RecipientEmail <chr>, ExternalReference <chr>, LocationLatitude <chr>,
## #   LocationLongitude <chr>, DistributionChannel <chr>,
## #   L1.R1.scenario_1 <chr>, L1.R0.scenario_1 <chr>, Q28 <chr>,
## #   Q29_1 <chr>, Q21_1 <chr>, Q22_1 <chr>, L0.R1.text_1 <chr>,
## #   L0.R0.text_1 <chr>, Q26_1 <chr>, Q14_1 <chr>, Q16 <chr>, Q18 <chr>,
## #   Q20 <chr>, Q22 <chr>, mTurkCode <chr>, load <chr>, had.read <chr>
##
## Rows in raw data = 3444
##
## Head of skinny dataset before exclusions:
## # A tibble: 6 x 12
##      id .site.name  .group had.read  load   lkl eff.split count.eff
##   <int>      <chr>   <chr>    <dbl> <dbl> <dbl>     <dbl>     <dbl>
## 1     1      MTurk a.mturk        0     1     5         2        10
## 2     2      MTurk a.mturk        0     0     1        NA        NA
## 3     3      MTurk a.mturk        1     0     1        NA        NA
## 4     4      MTurk a.mturk        1     1     2         5         6
## 5     5      MTurk a.mturk        1     0     2        NA        NA
## 6     6      MTurk a.mturk        0     0     1        NA        NA
## # ... with 4 more variables: count.hard <dbl>, badness <dbl>,
## #   importance <dbl>, end.num <dbl>
##
##
## Subjects with missing had.read, load, or lkl: 23 724 725 727 728 730 731 732 736 739 740 741 746 747
##
## Subjects with load==1 but missing eff.split, count.eff, or count.hard: 23 725 727 728 730 731 732 735
##
## Bad subjects (failed to follow instructions): 71 116 125 130 139 152 159 170 172 212 263 293 309 317
##
## Final n = 2973
##
## MARGINAL MEANS AND SDs FOR ANALYSIS AUDIT
##                     Overall
##   n                 2973
##   load (mean (sd))  0.44 (0.50)
##   tempt (mean (sd)) 0.50 (0.50)
##   lkl (mean (sd))   3.42 (2.39)
```

University of Virginia (UVA)

```
start.path = "data/raw/U Virginia/raw_uva.csv"
end.path = "data/prepped"

d = prep_site_data(start.path = start.path, end.path = end.path,
    lkl.names = c("likelihood_1", "likelihood_1_1", "likelihood_1_2",
        "likelihood_1_3"), had.read.name = "had.read", load.name = "load",
    end.num.name = "end.num", eff.split.name = "effort.split_1",
    count.eff.name = "effort.count_1", count.hard.name = "difficulty_1",
    badness.name = "negativity_1", importance.name = "academic.pressure_1",
    .site.name = "UVA", .group = "b.similar", .n.extra.header.rows = 2)
```

```
## Warning: Duplicated column names deduplicated: 'likelihood_1' =>
## 'likelihood_1_1' [18], 'likelihood_1' => 'likelihood_1_2' [23],
## 'likelihood_1' => 'likelihood_1_3' [24]
```

```
## Parsed with column specification:
## cols(
##   .default = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
##
##
## Extra header rows to delete (first 3 cols):
## # A tibble: 2 x 3
##                        StartDate                       EndDate
##                            <chr>                         <chr>
## 1                     Start Date                      End Date
## 2 "{\"ImportId\":\"startDate\"}" "{\"ImportId\":\"endDate\"}"
## # ... with 1 more variables: Status <chr>
##
##
## First row of real data:
## # A tibble: 1 x 29
##     StartDate      EndDate     Status      IPAddress Progress
##         <chr>        <chr>      <chr>          <chr>    <chr>
## 1 4/6/17 9:53 4/6/17 10:11 IP Address 128.143.174.55      100
## # ... with 24 more variables: `Duration (in seconds)` <chr>,
## #   Finished <chr>, RecordedDate <chr>, ResponseId <chr>,
## #   RecipientLastName <chr>, RecipientFirstName <chr>,
## #   RecipientEmail <chr>, ExternalReference <chr>, LocationLatitude <chr>,
## #   LocationLongitude <chr>, DistributionChannel <chr>,
## #   likelihood_1 <chr>, likelihood_1_1 <chr>, end.num <chr>,
## #   effort.split_1 <chr>, difficulty_1 <chr>, effort.count_1 <chr>,
## #   likelihood_1_2 <chr>, likelihood_1_3 <chr>, academic.pressure_1 <chr>,
## #   negativity_1 <chr>, mTurkCode <chr>, load <chr>, had.read <chr>
##
## Rows in raw data = 156
##
## Head of skinny dataset before exclusions:
## # A tibble: 6 x 12
##      id .site.name    .group had.read  load   lkl eff.split count.eff
```

```
##   <int>      <chr>     <chr>    <dbl> <dbl> <dbl>     <dbl>     <dbl>
## 1     1        UVA b.similar        1     0     1        NA        NA
## 2     2        UVA b.similar        0     1     4         2         8
## 3     3        UVA b.similar        1     0     0        NA        NA
## 4     4        UVA b.similar        0     0     5        NA        NA
## 5     5        UVA b.similar        0     1     6         2         8
## 6     6        UVA b.similar        1     1     1         5         9
## # ... with 4 more variables: count.hard <dbl>, badness <dbl>,
## #   importance <dbl>, end.num <dbl>
##
##
## Subjects with missing had.read, load, or lkl:
##
## Subjects with load==1 but missing eff.split, count.eff, or count.hard:
##
## Bad subjects (failed to follow instructions): 21 109 110 115 144
##
## Final n = 151
##
## MARGINAL MEANS AND SDs FOR ANALYSIS AUDIT
##                      Overall
##   n                   151
##   load (mean (sd))   0.49 (0.50)
##   tempt (mean (sd)) 0.49 (0.50)
##   lkl (mean (sd))    2.68 (2.15)
```

Aggregated Data Preparation

Stitch datasets:

```
# rbind all the datasets into one
b <- list.files(path = "data/prepped", pattern = "*.csv") %>%
    map_df(function(x) read_csv(paste0("data/prepped/", x))) %>%
    rename(site = .site.name, group = .group) %>% mutate(is.mturk = ifelse(group ==
    "a.mturk", 1, 0))
```

```
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   .site.name = col_character(),
##   .group = col_character(),
##   had.read = col_integer(),
##   load = col_integer(),
##   lkl = col_integer(),
##   eff.split = col_integer(),
##   count.eff = col_integer(),
##   count.hard = col_integer(),
##   badness = col_integer(),
##   importance = col_integer(),
##   end.num = col_integer(),
##   tempt = col_integer(),
##   excluded = col_integer(),
##   n.excl = col_integer()
```

```
## )
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   .site.name = col_character(),
##   .group = col_character(),
##   had.read = col_integer(),
##   load = col_integer(),
##   lkl = col_integer(),
##   eff.split = col_integer(),
##   count.eff = col_integer(),
##   count.hard = col_integer(),
##   badness = col_integer(),
##   importance = col_integer(),
##   end.num = col_integer(),
##   tempt = col_integer(),
##   excluded = col_integer(),
##   n.excl = col_integer()
## )
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   .site.name = col_character(),
##   .group = col_character(),
##   had.read = col_integer(),
##   load = col_integer(),
##   lkl = col_integer(),
##   eff.split = col_integer(),
##   count.eff = col_integer(),
##   count.hard = col_integer(),
##   badness = col_integer(),
##   importance = col_integer(),
##   end.num = col_integer(),
##   tempt = col_integer(),
##   excluded = col_integer(),
##   n.excl = col_integer()
## )
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   .site.name = col_character(),
##   .group = col_character(),
##   had.read = col_integer(),
##   load = col_integer(),
##   lkl = col_integer(),
##   eff.split = col_integer(),
##   count.eff = col_integer(),
##   count.hard = col_integer(),
##   badness = col_integer(),
##   importance = col_integer(),
##   end.num = col_integer(),
##   tempt = col_integer(),
##   excluded = col_integer(),
##   n.excl = col_integer()
```

```
## )
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   .site.name = col_character(),
##   .group = col_character(),
##   had.read = col_integer(),
##   load = col_integer(),
##   lkl = col_integer(),
##   eff.split = col_integer(),
##   count.eff = col_integer(),
##   count.hard = col_integer(),
##   badness = col_integer(),
##   importance = col_integer(),
##   end.num = col_integer(),
##   tempt = col_integer(),
##   excluded = col_integer(),
##   n.excl = col_integer()
## )
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   .site.name = col_character(),
##   .group = col_character(),
##   had.read = col_integer(),
##   load = col_integer(),
##   lkl = col_integer(),
##   eff.split = col_integer(),
##   count.eff = col_integer(),
##   count.hard = col_integer(),
##   badness = col_integer(),
##   importance = col_integer(),
##   end.num = col_integer(),
##   tempt = col_integer(),
##   excluded = col_integer(),
##   n.excl = col_integer()
## )
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   .site.name = col_character(),
##   .group = col_character(),
##   had.read = col_integer(),
##   load = col_integer(),
##   lkl = col_integer(),
##   eff.split = col_integer(),
##   count.eff = col_integer(),
##   count.hard = col_integer(),
##   badness = col_integer(),
##   importance = col_integer(),
##   end.num = col_integer(),
##   tempt = col_integer(),
##   excluded = col_integer(),
##   n.excl = col_integer()
```

```
## )
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   .site.name = col_character(),
##   .group = col_character(),
##   had.read = col_integer(),
##   load = col_integer(),
##   lkl = col_integer(),
##   eff.split = col_integer(),
##   count.eff = col_character(),
##   count.hard = col_character(),
##   badness = col_character(),
##   importance = col_character(),
##   end.num = col_integer(),
##   tempt = col_integer(),
##   excluded = col_integer(),
##   n.excl = col_integer()
## )
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   .site.name = col_character(),
##   .group = col_character(),
##   had.read = col_integer(),
##   load = col_integer(),
##   lkl = col_integer(),
##   eff.split = col_integer(),
##   count.eff = col_integer(),
##   count.hard = col_integer(),
##   badness = col_integer(),
##   importance = col_integer(),
##   end.num = col_integer(),
##   tempt = col_integer(),
##   excluded = col_integer(),
##   n.excl = col_integer()
## )
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   .site.name = col_character(),
##   .group = col_character(),
##   had.read = col_integer(),
##   load = col_integer(),
##   lkl = col_integer(),
##   eff.split = col_integer(),
##   count.eff = col_integer(),
##   count.hard = col_integer(),
##   badness = col_integer(),
##   importance = col_integer(),
##   end.num = col_integer(),
##   tempt = col_integer(),
##   excluded = col_integer(),
```

```
##   n.excl = col_integer()
## )
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   .site.name = col_character(),
##   .group = col_character(),
##   had.read = col_integer(),
##   load = col_integer(),
##   lkl = col_integer(),
##   eff.split = col_integer(),
##   count.eff = col_integer(),
##   count.hard = col_integer(),
##   badness = col_integer(),
##   importance = col_integer(),
##   end.num = col_integer(),
##   tempt = col_integer(),
##   excluded = col_integer(),
##   n.excl = col_integer()
## )
```

```
# add median SAT score for secondary analyses (estimated for
# 2018) per discussion with Dan Simons site of original study
# (Cornell): 2134 data from
# https://www.collegeraptor.com/college-rankings/details/MedianSAT
b$SAT[b$site == "Stanford"] = 2162
```

```
## Warning: Unknown or uninitialised column: 'SAT'.
```

```
b$SAT[b$site == "U Penn"] = 2178
b$SAT[b$site == "UCB"] = 2092
b$SAT[b$site == "UVA"] = 2032
b$SAT[b$site == "RHIT"] = 1951
b$SAT[b$site == "BYUI"] = 1943
b$SAT[b$site == "URI"] = 1182  # from their admissions website because not on College Raptor
b$SAT[b$site %in% c("MTurk", "Eotvos", "KUL", "UP")] = NA  # foreign or online sites

# write data
write_csv(b, "data/full_prepped_dataset.csv")
```

Session info for reproducibility.

```
devtools::session_info()
```

```
## Session info -------------------------------------------------------------
```

```
##  setting  value
##  version  R version 3.3.3 (2017-03-06)
##  system   x86_64, darwin13.4.0
##  ui       X11
##  language (EN)
##  collate  en_US.UTF-8
##  tz       America/New_York
##  date     2017-12-07
```

```
## Packages -----------------------------------------------------------------
```

```
##  package    * version date       source
```

```
##  assertthat   0.1     2013-12-06 CRAN (R 3.3.0)
##  backports    1.0.4   2016-10-24 CRAN (R 3.3.0)
##  base       * 3.3.3   2017-03-07 local
##  bindr        0.1     2016-11-13 CRAN (R 3.3.2)
##  bindrcpp   * 0.2     2017-06-17 CRAN (R 3.3.2)
##  broom        0.4.2   2017-02-13 CRAN (R 3.3.2)
##  cellranger   1.1.0   2016-07-27 cran (@1.1.0)
##  class        7.3-14  2015-08-30 CRAN (R 3.3.3)
##  colorspace   1.2-6   2015-03-11 CRAN (R 3.3.0)
##  datasets   * 3.3.3   2017-03-07 local
##  devtools   * 1.13.3  2017-08-02 CRAN (R 3.3.2)
##  digest       0.6.12  2017-01-27 cran (@0.6.12)
##  dplyr      * 0.7.4   2017-09-28 CRAN (R 3.3.2)
##  e1071        1.6-8   2017-02-02 CRAN (R 3.3.2)
##  evaluate     0.10.1  2017-06-24 cran (@0.10.1)
##  forcats      0.2.0   2017-01-23 CRAN (R 3.3.2)
##  formatR      1.4     2016-05-09 CRAN (R 3.3.0)
##  ggplot2    * 2.2.1   2016-12-30 CRAN (R 3.3.2)
##  glue         1.1.1   2017-06-21 CRAN (R 3.3.2)
##  graphics   * 3.3.3   2017-03-07 local
##  grDevices  * 3.3.3   2017-03-07 local
##  grid         3.3.3   2017-03-07 local
##  gtable       0.2.0   2016-02-26 CRAN (R 3.3.0)
##  haven        1.1.0   2017-07-09 cran (@1.1.0)
##  hms          0.3     2016-11-22 CRAN (R 3.3.2)
##  htmltools    0.3.6   2017-04-28 cran (@0.3.6)
##  httr         1.2.1   2016-07-03 CRAN (R 3.3.0)
##  jsonlite     1.5     2017-06-01 cran (@1.5)
##  knitr      * 1.17    2017-08-10 CRAN (R 3.3.2)
##  lattice      0.20-34 2016-09-06 CRAN (R 3.3.3)
##  lazyeval     0.2.0   2016-06-12 CRAN (R 3.3.0)
##  lubridate    1.6.0   2016-09-13 CRAN (R 3.3.0)
##  magrittr     1.5     2014-11-22 CRAN (R 3.3.0)
##  Matrix       1.2-8   2017-01-20 CRAN (R 3.3.2)
##  memoise      1.1.0   2017-04-21 cran (@1.1.0)
##  methods    * 3.3.3   2017-03-07 local
##  mnormt       1.5-4   2016-03-09 CRAN (R 3.3.0)
##  modelr       0.1.0   2016-08-31 CRAN (R 3.3.0)
##  munsell      0.4.3   2016-02-13 CRAN (R 3.3.0)
##  nlme         3.1-131 2017-02-06 CRAN (R 3.3.3)
##  parallel     3.3.3   2017-03-07 local
##  pkgconfig    2.0.1   2017-03-21 CRAN (R 3.3.2)
##  plyr         1.8.4   2016-06-08 CRAN (R 3.3.0)
##  psych        1.6.6   2016-06-28 CRAN (R 3.3.0)
##  purrr      * 0.2.2   2016-06-18 CRAN (R 3.3.0)
##  R6           2.2.2   2017-06-17 cran (@2.2.2)
##  Rcpp         0.12.13 2017-09-28 cran (@0.12.13)
##  readr      * 1.1.1   2017-05-16 cran (@1.1.1)
##  readxl       1.0.0   2017-04-18 cran (@1.0.0)
##  reshape2     1.4.2   2016-10-22 CRAN (R 3.3.0)
##  rlang        0.1.2   2017-08-09 CRAN (R 3.3.2)
##  rmarkdown    1.7     2017-11-10 cran (@1.7)
##  rprojroot    1.2     2017-01-16 CRAN (R 3.3.2)
##  rvest        0.3.2   2016-06-17 CRAN (R 3.3.0)
```

```
##  scales       0.4.1   2016-11-09 CRAN (R 3.3.2)
##  splines      3.3.3   2017-03-07 local
##  stats      * 3.3.3   2017-03-07 local
##  stringi      1.1.5   2017-04-07 cran (@1.1.5)
##  stringr      1.2.0   2017-02-18 cran (@1.2.0)
##  survey       3.31-5  2016-12-01 CRAN (R 3.3.2)
##  survival     2.40-1  2016-10-30 CRAN (R 3.3.0)
##  tableone   * 0.7.3   2015-11-11 CRAN (R 3.3.0)
##  tibble     * 1.3.3   2017-05-28 cran (@1.3.3)
##  tidyr      * 0.6.1   2017-01-10 CRAN (R 3.3.2)
##  tidyverse  * 1.1.1   2017-01-27 CRAN (R 3.3.2)
##  tools        3.3.3   2017-03-07 local
##  utils      * 3.3.3   2017-03-07 local
##  withr        1.0.2   2016-06-20 CRAN (R 3.3.0)
##  xml2         1.1.1   2017-01-24 CRAN (R 3.3.2)
##  yaml         2.1.14  2016-11-12 CRAN (R 3.3.2)
```