Manipulate survey data

Scott Moore

October 4, 2024

Table of contents

This file imports, transforms, and then exports the data that I created for the *Assessment Institute* meeting in Indianapolis in October 2024.

0.1 Setup & import libraries

Standard library import steps. Enables all that is to come.

```
library(tidyverse)
library(skimr)
```

0.2 Import data

Data in CSV format with headers.

```
survey <- read_csv("data/retention_survey_history.csv")
```

0.3 Examine data

Show us a bit of information about the imported data.

```
glimpse(survey)
```

```
Rows: 33,524
Columns: 24
$ Year         <dbl> 2012, 2012, 2012, 2012, 2012, 2012, 2012, 2012, 2012, 201~
$ ID           <chr> "xuojqdfdozvu", "vvwkinqvnibo", "ibyjcmiopiqk", "lsqamawy~
$ NPS          <dbl> 8, 8, 8, 8, 8, 8, 8, 4, 4, 7, 6, 6, 6, 4, 7, 4, 6, 7, 4, ~
$ Field        <chr> "Undecl", "SocSci", "CompSci", "Other", "SocSci", "Undecl~
$ ClassLevel   <chr> "Sr", "Sr", "Sr", "Sr", "Sr", "Sr", "Sr", "Fresh", "Fresh~
$ Status       <chr> "Full-time", "Full-time", "Full-time", "Full-time", "Full~
$ Gender       <chr> "Male", "Female", "Male", "Female", "Other", "Female", "M~
$ BirthYear    <dbl> 1990, 1999, 1999, 1989, 1993, 1988, 1996, 1991, 1990, 198~
$ FinPL        <chr> "No", "No", "No", "No", "Yes", "Yes", "No", "No", "No", "~
$ FinSch       <chr> "No", "No", "No", "Yes", "Yes", "Yes", "Yes", "No", "No",~
$ FinGov       <chr> "No", "Yes", "Yes", NA, "No", "No", "No", "No", "No", "No~
$ FinSelf      <chr> "No", "Yes", "No", "No", "Yes", "Yes", "No", "Yes", "No",~
$ FinPar       <chr> "Yes", "No", NA, "Yes", "No", "No", "No", "No", "No", "Ye~
$ FinOther     <chr> "No", "No", "No", "No", "No", "No", "No", "No", "No", "No~
$ TooDifficult <chr> "Disagree", "Strongly Disagree", "Disagree", "Agree", NA,~
$ NotRelevant  <chr> "Strongly Disagree", "Disagree", NA, "Strongly Disagree",~
$ PoorTeaching <chr> "Agree", "Agree", "Agree", "Disagree", "Agree", "Strongly~
$ UnsuppFac    <chr> NA, "Neutral", "Neutral", "Neutral", NA, "Neutral", "Stro~
$ Grades       <chr> NA, "Disagree", "Strongly Disagree", NA, NA, "Agree", NA,~
$ Sched        <chr> "Strongly Agree", "Strongly Disagree", "Strongly Disagree~
$ ClassTooBig  <chr> "Neutral", NA, "Strongly Disagree", "Strongly Disagree", ~
$ BadAdvising  <chr> "Disagree", "Disagree", NA, "Disagree", "Strongly Disagre~
$ FinAid       <chr> "Strongly Agree", "Strongly Agree", "Strongly Agree", "Ag~
$ OverallValue <chr> "Strongly Agree", "Strongly Agree", "Neutral", "Strongly ~
```

```
names(survey)
```

```
 [1] "Year"          "ID"            "NPS"           "Field"         "ClassLevel"  
 [6] "Status"        "Gender"        "BirthYear"     "FinPL"         "FinSch"      
[11] "FinGov"        "FinSelf"       "FinPar"        "FinOther"      "TooDifficult"
[16] "NotRelevant"   "PoorTeaching"  "UnsuppFac"     "Grades"        "Sched"       
[21] "ClassTooBig"   "BadAdvising"   "FinAid"        "OverallValue"
```

```
skim(survey)
```

Table 1: Data summary

Name	survey
Number of rows	33524
Number of columns	24
Column type frequency:	
character	21
numeric	3
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
ID	0	1.00	12	12	0	33524	0
Field	12	1.00	2	9	0	15	0
ClassLevel	0	1.00	2	5	0	5	0
Status	627	0.98	5	9	0	3	0
Gender	875	0.97	4	6	0	3	0
FinPL	1674	0.95	2	3	0	2	0
FinSch	1615	0.95	2	3	0	2	0
FinGov	1586	0.95	2	3	0	2	0
FinSelf	1622	0.95	2	3	0	2	0
FinPar	1699	0.95	2	3	0	2	0
FinOther	1699	0.95	2	3	0	2	0
TooDifficult	6645	0.80	5	17	0	5	0
NotRelevant	6622	0.80	5	17	0	5	0
PoorTeaching	6819	0.80	5	17	0	5	0
UnsuppFac	6676	0.80	5	17	0	5	0
Grades	6637	0.80	5	17	0	5	0
Sched	6684	0.80	5	17	0	5	0
ClassTooBig	6634	0.80	5	17	0	5	0
BadAdvising	6735	0.80	5	17	0	5	0
FinAid	6676	0.80	5	17	0	5	0
OverallValue	6631	0.80	5	17	0	5	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Year	0	1	2014.50	1.71	2012	2013	2014	2016	2017	
NPS	0	1	5.95	1.52	4	4	6	7	8	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
BirthYear	0	1	1994.01	3.76	1988	1991	1994	1997	2000	

0.4 Fix NA values

Just in case any columns represent `NA` values in a variety of ways, we can standardize the representation before we continue to simplify any logic later in the process.

0.4.1 Create standardized representation of NA

Actually do the transformation.

```
survey <-
  survey |>
    mutate(across(c(Field:Gender,
                    FinPL:OverallValue),
                  ~ case_when(is.na(.x) ~ NA,
                              .x == "" ~ NA,
                              .x == "NA" ~ NA,
                              .x == "--" ~ NA,
                              .default = .x)))
```

0.4.2 Ensure that it works

Just what it says — ensure that it works. This counts the number of appearances in every single column of the *non-standard* ways in which `NA` values might have been represented. We should see nothing but zero (`0`) values in the table below.

```
survey |>
  summarize(across(everything(),
                   ~sum(.x %in% c("NA", "--", "")))) |>
  print(width = Inf)
```

```
# A tibble: 1 x 24
   Year    ID   NPS Field ClassLevel Status Gender BirthYear FinPL FinSch FinGov
  <int> <int> <int> <int>      <int>  <int>  <int>     <int> <int>  <int>  <int>
1     0     0     0     0          0      0      0         0     0      0      0
  FinSelf FinPar FinOther TooDifficult NotRelevant PoorTeaching UnsuppFac Grades
    <int>  <int>    <int>        <int>       <int>        <int>     <int>  <int>
1       0      0        0            0           0            0         0      0
  Sched ClassTooBig BadAdvising FinAid OverallValue
  <int>       <int>       <int>  <int>        <int>
1     0           0           0      0            0
```

0.5 Factors

0.5.1 Define factors

Define factors for as many fields as we can. Specify its order when we can so that later analyses are better organized.

```
fin_cols <- c("FinPL", "FinSch", "FinGov",
              "FinSelf", "FinPar", "FinOther")
exp_cols <- c("TooDifficult", "NotRelevant", "PoorTeaching",
              "UnsuppFac", "Grades", "Sched", "ClassTooBig",
              "BadAdvising", "FinAid", "OverallValue")
fin_levels <- c("Yes", "No")
exp_levels <- c("Strongly Disagree", "Disagree", "Neutral", "Agree",
                "Strongly Agree")
class_levels <- c("Fresh", "Soph", "Jr", "Sr")

survey <-
  survey |>
    mutate(across(all_of(exp_cols),
                  ~factor(.x,
                          levels = exp_levels,
                          ordered = TRUE)))
survey <-
  survey |>
    mutate(across(all_of(fin_cols),
                  ~factor(.x,
                          levels = fin_levels,
                          ordered = TRUE)))
survey <-
  survey |>
    mutate(ClassLevel = factor(ClassLevel,
                               levels = class_levels,
                               ordered = TRUE))
survey <-
  survey |>
    mutate(Status = factor(Status,
                           levels = c("Full-time", "Part-time",
                           "Other")))
survey <-
  survey |>
    mutate(Gender = factor(Gender,
                           levels = c("Female", "Male", "Other")))
```

We want to handle `Field` of study somewhat differently. It has `NA` values in the column, but we already have `Undecl` and `Other` as values. We would rather include the `NA` values in the analysis, but we want to be able to see if the results for that value differ from the results of these other two values. Let's make `Unknown` an accepted value for `Field`.

```
poss_fields <- c("LifeSci", "PhysSci", "PubHealth", "Nurs", "OthHealth",
                 "PubAdm", "SocSci", "ArtsHum",
                 "CompSci", "Eng", "Bus", "Ed", "ArchUP",
                 "Other", "Undecl", "Unknown")
survey$Field <-
  survey$Field |>
  replace_na("Unknown")
survey <-
  survey |>
    mutate(Field = factor(Field,
                          levels = poss_fields))
```

0.5.2 Validate the factor creation process

Let's show the `structure` of the table. This will allow us to validate that all of our `factor` declarations above worked.

```
str(survey)
```

```
tibble [33,524 x 24] (S3: tbl_df/tbl/data.frame)
 $ Year        : num [1:33524] 2012 2012 2012 2012 2012 ...
 $ ID          : chr [1:33524] "xuojqdfdozvu" "vvwkinqvnibo" "ibyjcmiopiqk" "lsqamawyancj" ...
 $ NPS         : num [1:33524] 8 8 8 8 8 8 8 4 4 7 ...
 $ Field       : Factor w/ 16 levels "LifeSci","PhysSci",..: 15 7 9 14 7 15 15 10 7 7 ...
 $ ClassLevel  : Ord.factor w/ 4 levels "Fresh"<"Soph"<..: 4 4 4 4 4 4 4 1 1 3 ...
 $ Status      : Factor w/ 3 levels "Full-time","Part-time",..: 1 1 1 1 1 2 2 1 1 1 ...
 $ Gender      : Factor w/ 3 levels "Female","Male",..: 2 1 2 1 3 1 2 1 1 2 ...
 $ BirthYear   : num [1:33524] 1990 1999 1999 1989 1993 ...
 $ FinPL       : Ord.factor w/ 2 levels "Yes"<"No": 2 2 2 2 1 1 2 2 2 2 ...
 $ FinSch      : Ord.factor w/ 2 levels "Yes"<"No": 2 2 2 1 1 1 1 2 2 1 ...
 $ FinGov      : Ord.factor w/ 2 levels "Yes"<"No": 2 1 1 NA 2 2 2 2 2 2 ...
 $ FinSelf     : Ord.factor w/ 2 levels "Yes"<"No": 2 1 2 2 1 1 2 1 2 1 ...
 $ FinPar      : Ord.factor w/ 2 levels "Yes"<"No": 1 2 NA 1 2 2 2 2 2 1 ...
 $ FinOther    : Ord.factor w/ 2 levels "Yes"<"No": 2 2 2 2 2 2 2 2 2 2 ...
 $ TooDifficult: Ord.factor w/ 5 levels "Strongly Disagree"<..: 2 1 2 4 NA NA 3 1 5 1 ...
 $ NotRelevant : Ord.factor w/ 5 levels "Strongly Disagree"<..: 1 2 NA 1 3 NA 5 NA NA 4 ...
 $ PoorTeaching: Ord.factor w/ 5 levels "Strongly Disagree"<..: 4 4 4 2 4 1 4 1 2 5 ...
 $ UnsuppFac   : Ord.factor w/ 5 levels "Strongly Disagree"<..: NA 3 3 3 NA 3 1 5 3 NA ...
 $ Grades      : Ord.factor w/ 5 levels "Strongly Disagree"<..: NA 2 1 NA NA 4 NA 2 NA 2 ...
 $ Sched       : Ord.factor w/ 5 levels "Strongly Disagree"<..: 5 1 1 4 1 5 4 4 4 5 ...
 $ ClassTooBig : Ord.factor w/ 5 levels "Strongly Disagree"<..: 3 NA 1 1 5 NA 5 4 NA 2 ...
 $ BadAdvising : Ord.factor w/ 5 levels "Strongly Disagree"<..: 2 2 NA 2 1 2 1 NA 2 1 ...
 $ FinAid      : Ord.factor w/ 5 levels "Strongly Disagree"<..: 5 5 5 4 NA NA 5 5 1 NA ...
 $ OverallValue: Ord.factor w/ 5 levels "Strongly Disagree"<..: 5 5 3 5 2 5 5 4 NA 2 ...
```

This is another way to validate that it worked. This will show 6 distinct values for each column in the table. It will pad with `NA` values if it has less than six. The ordered values should be listed in order; the others are essentially random and nothing should be read into it.

```
get_distinct_and_pad <- function(column, numvals = 6) {
  distinct_values <- tibble(value = column) |>
                     distinct(value) |>
                     arrange(value) |>
                     slice_head(n = numvals) |>
                     pull(value)

  # Pad with NA if there are less than n values
  length(distinct_values) <- numvals
  return(distinct_values)
}

survey |>
  map(get_distinct_and_pad) |>
  as.data.frame()
```

```
  Year           ID NPS     Field ClassLevel    Status Gender BirthYear FinPL
1 2012 aaapjamvmsgd   4   LifeSci      Fresh Full-time Female      1988   Yes
2 2013 aacgcpuqbhue   6   PhysSci       Soph Part-time   Male      1989    No
3 2014 aacgctmukgti   7 PubHealth         Jr     Other  Other      1990  <NA>
4 2015 aacukkmcqpdr   8      Nurs         Sr      <NA>   <NA>      1991  <NA>
5 2016 aacuutwvloft  NA OthHealth       <NA>      <NA>   <NA>      1992  <NA>
6 2017 aadtesmehzsd  NA    PubAdm       <NA>      <NA>   <NA>      1993  <NA>
  FinSch FinGov FinSelf FinPar FinOther      TooDifficult       NotRelevant
1    Yes    Yes     Yes    Yes      Yes Strongly Disagree Strongly Disagree
2     No     No      No     No       No          Disagree          Disagree
3   <NA>   <NA>    <NA>   <NA>     <NA>           Neutral           Neutral
4   <NA>   <NA>    <NA>   <NA>     <NA>             Agree             Agree
5   <NA>   <NA>    <NA>   <NA>     <NA>    Strongly Agree    Strongly Agree
6   <NA>   <NA>    <NA>   <NA>     <NA>              <NA>              <NA>
       PoorTeaching         UnsuppFac            Grades             Sched
1 Strongly Disagree Strongly Disagree Strongly Disagree Strongly Disagree
2          Disagree          Disagree          Disagree          Disagree
3           Neutral           Neutral           Neutral           Neutral
4             Agree             Agree             Agree             Agree
5    Strongly Agree    Strongly Agree    Strongly Agree    Strongly Agree
6              <NA>              <NA>              <NA>              <NA>
        ClassTooBig       BadAdvising            FinAid      OverallValue
```

```
1 Strongly Disagree Strongly Disagree Strongly Disagree Strongly Disagree
2          Disagree          Disagree          Disagree          Disagree
3           Neutral           Neutral           Neutral           Neutral
4             Agree             Agree             Agree             Agree
5    Strongly Agree    Strongly Agree    Strongly Agree    Strongly Agree
6              <NA>              <NA>              <NA>              <NA>
```

0.6 Convert "wide" to "long" data

R has to have data in "long" format for analysis. The pivot_longer function is the tool that allows us to make the transformation from "wide" to "long".

```
survey <-
  survey |>
    pivot_longer(
      names_to = "Question",
      cols = c(TooDifficult:OverallValue),
      values_to = "Response"
    ) |>
    arrange(Year, ID, Question)
```

0.7 Prepare numerical data

When analyzing survey data, it can be helpful to have both a string reprentation and an integer representation of response data. This stsatement creates the new NumResp column that will hold the numeric response data while retaining the string Response column.

```
survey <-
  survey |>
    mutate(NumResp = case_when(
      Response == "Strongly Disagree" ~ 1,
      Response == "Disagree" ~ 2,
      Response == "Neutral" ~ 3,
      Response == "Agree" ~ 4,
      Response == "Strongly Agree" ~ 5,
      .default = NA))
```

0.8 Look at data again

Now that we have made all of these transformations, let's simply print out information about the survey table in its new form for documentation.

```
glimpse(survey)
```

```
Rows: 335,240
Columns: 17
$ Year       <dbl> 2012, 2012, 2012, 2012, 2012, 2012, 2012, 2012, 2012, 2012,~
$ ID         <chr> "aacuutwvloft", "aacuutwvloft", "aacuutwvloft", "aacuutwvlo~
$ NPS        <dbl> 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7,~
$ Field      <fct> Undecl, Undecl, Undecl, Undecl, Undecl, Undecl, Undecl, Und~
$ ClassLevel <ord> Jr, Jr, Jr, Jr, Jr, Jr, Jr, Jr, Jr, Jr, Jr, Jr, Jr, Jr, Jr,~
$ Status     <fct> Full-time, Full-time, Full-time, Full-time, Full-time, Full~
$ Gender     <fct> Female, Female, Female, Female, Female, Female, Female, Fem~
$ BirthYear  <dbl> 1995, 1995, 1995, 1995, 1995, 1995, 1995, 1995, 1995, 1995,~
$ FinPL      <ord> No, No, No, No, No, No, No, No, No, No, No, No, No, No, No,~
$ FinSch     <ord> Yes, Yes, Yes, Yes, Yes, Yes, Yes, Yes, Yes, Yes, Yes, Yes,~
$ FinGov     <ord> No, No, No, No, No, No, No, No, No, No, No, No, No, No, No,~
$ FinSelf    <ord> Yes, Yes, Yes, Yes, Yes, Yes, Yes, Yes, Yes, Yes, No, No, N~
$ FinPar     <ord> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, No, No, No, No, No,~
$ FinOther   <ord> No, No, No, No, No, No, No, No, No, No, No, No, No, No, No,~
$ Question   <chr> "BadAdvising", "ClassTooBig", "FinAid", "Grades", "NotRelev~
$ Response   <ord> NA, Neutral, Agree, NA, NA, Strongly Agree, Strongly Agree,~
$ NumResp    <dbl> NA, 3, 4, NA, NA, 5, 5, 2, 2, 2, 1, 3, 4, 1, NA, 5, NA, 3, ~
```

```
skim(survey)
```

Table 4: Data summary

Name	survey
Number of rows	335240
Number of columns	17
Column type frequency:	
character	2
factor	11
numeric	4
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
ID	0	1	12	12	0	33524	0
Question	0	1	5	12	0	10	0

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
Field	0	1.00	FALSE	16	Soc: 71580, Lif: 63180, Und: 41150, Eng: 36790
ClassLevel	5510	0.98	TRUE	4	Fre: 109270, Sop: 83110, Jr: 71340, Sr: 66010
Status	6270	0.98	FALSE	3	Ful: 251240, Par: 61730, Oth: 16000
Gender	8750	0.97	FALSE	3	Fem: 159880, Mal: 137130, Oth: 29480
FinPL	16740	0.95	TRUE	2	No: 222110, Yes: 96390
FinSch	16150	0.95	TRUE	2	No: 160240, Yes: 158850
FinGov	15860	0.95	TRUE	2	No: 286960, Yes: 32420
FinSelf	16220	0.95	TRUE	2	No: 190670, Yes: 128350
FinPar	16990	0.95	TRUE	2	No: 255370, Yes: 62880
FinOther	16990	0.95	TRUE	2	No: 301970, Yes: 16280
Response	66759	0.80	TRUE	5	Str: 62765, Agr: 61927, Neu: 55456, Dis: 48919

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Year	0	1.0	2014.50	1.71	2012	2013	2014	2016	2017	
NPS	0	1.0	5.95	1.52	4	4	6	7	8	
BirthYear	0	1.0	1994.01	3.76	1988	1991	1994	1997	2000	
NumResp	66759	0.8	3.22	1.37	1	2	3	4	5	

0.9 Summarize by question

Now that we have the new `NumResp` column, we can calculate numerical data on the survey responses.

```r
survey |>
  group_by(Question) |>
  summarize(Median = median(NumResp, na.rm = TRUE),
            Avg = mean(NumResp, na.rm = TRUE))
```

```
# A tibble: 10 x 3
   Question     Median   Avg
   <chr>         <dbl> <dbl>
 1 BadAdvising       2  2.33
 2 ClassTooBig       2  2.52
 3 FinAid            4  3.83
 4 Grades            3  3.02
 5 NotRelevant       3  2.73
 6 OverallValue      4  4.11
 7 PoorTeaching      4  3.42
 8 Sched             4  4.00
 9 TooDifficult      3  2.99
10 UnsuppFac         3  3.28
```

0.10 Remove columns that we do not need

This table is quite large. We can get rid of the `ID` field if nothing else since we will not be doing any analysis related to the `ID` of the survey.

```
survey <-
  survey |>
    select(Year, NPS, Field, ClassLevel, Status, Gender, BirthYear,
           FinPL, FinSch, FinGov, FinSelf, FinPar, FinOther,
           Question, Response, NumResp)
```

0.11 Export data

Export the data but don't include the row number.

```
write.csv(survey,"data/survey-output.csv",
          row.names=FALSE)
```