Research Plumbing I - Solutions

Exercise A - (8 min)

1. Download https://ditraglia.com/data/STAR.csv and save this file on your local machine. Then load it with `read_csv()`. Note that this will require you to specify the path to this file on your local machine.
2. The file `final5.dta` from the Angrist data archive contains data from the article "Using Maimonides Rule to estimate the Effect of Class Size on Student Achievement" by Angrist & Lavy. Locate and download this file. Then try to load it with `read_dta()`. You will get an error. Consult the section "Character encoding" in the associated R help file and follow the instructions given there.

Solution

I can't show a solution with the path on your personallocal machine, so I'll read these datasets directly from the internet instead:

```
library(tidyverse)
library(haven)

# Part 1
star <- read_csv('https://ditraglia.com/data/STAR.csv')

# Part 2
final5 <- read_dta('https://ditraglia.com/data/final5.dta',
                   encoding = 'latin1')
```

Exercise B - (8 min)

1. Use `ends_with()` to select the columns `quiz2` and `midterm2` from `gradebook` with a minimum of typing.
2. Use `contains()` to select the columns whose names contain the abbreviation for "Empirical Research Methods."
3. May the 4th be with you (belatedly)! The `dplyr` package includes a built-in dataset called `starwars`. Use the `glimpse()` function to get a quick overview of this dataset, and then read the associated help file before completing the following:
   a. Select only the columns of `starwars` that contain character data.
   b. Select only the columns whose names contain an underscore.
   c. Select only the columns that are either numeric *or* whose names end with "color."

Solution

```
set.seed(92815)
gradebook <- tibble(
  student_id = c(192297, 291857, 500286, 449192, 372152, 627561),
  name = c('Alice', 'Bob', 'Charlotte', 'Dante',
           'Ethelburga', 'Felix'),
  quiz1 = round(rnorm(6, 65, 15)),
  quiz2 = round(rnorm(6, 88, 5)),
  quiz3 = round(rnorm(6, 75, 10)),
  midterm1 = round(rnorm(6, 75, 10)),
  midterm2 = round(rnorm(6, 80, 8)),
  final = round(rnorm(6, 78, 11)))


# Part 1
gradebook |>
  select(ends_with('2'))
```

```
# A tibble: 6 × 2
  quiz2 midterm2
  <dbl>    <dbl>
1    96       90
2    91       75
3    94       70
4    85       94
5    91       73
6    86       83
```

```
# Part 2
gradebook |>
  select(contains('erm'))
```

```
# A tibble: 6 × 2
  midterm1 midterm2
     <dbl>    <dbl>
1       81       90
2       75       75
3       81       70
4       83       94
5       63       73
6       78       83
```

```
# Part 3a
starwars |>
  select(where(is.character))
```

```
# A tibble: 87 × 8
   name           hair_color skin_color eye_color sex   gender homeworld species
   <chr>          <chr>      <chr>      <chr>     <chr> <chr>  <chr>     <chr>
 1 Luke Skywalker blond      fair       blue      male  mascu… Tatooine  Human
 2 C-3PO          <NA>       gold       yellow    none  mascu… Tatooine  Droid
 3 R2-D2          <NA>       white, bl… red       none  mascu… Naboo     Droid
 4 Darth Vader    none       white      yellow    male  mascu… Tatooine  Human
 5 Leia Organa    brown      light      brown     fema… femin… Alderaan  Human
 6 Owen Lars      brown, gr… light      blue      male  mascu… Tatooine  Human
 7 Beru Whitesun… brown      light      blue      fema… femin… Tatooine  Human
 8 R5-D4          <NA>       white, red red       none  mascu… Tatooine  Droid
 9 Biggs Darklig… black      light      brown     male  mascu… Tatooine  Human
10 Obi-Wan Kenobi auburn, w… fair       blue-gray male  mascu… Stewjon   Human
# ℹ 77 more rows
```

```
# Part 3b
starwars |>
  select(contains('_'))
```

```
# A tibble: 87 × 4
   hair_color    skin_color  eye_color birth_year
   <chr>         <chr>       <chr>          <dbl>
 1 blond         fair        blue            19
 2 <NA>          gold        yellow         112
 3 <NA>          white, blue red             33
 4 none          white       yellow          41.9
 5 brown         light       brown           19
 6 brown, grey   light       blue            52
 7 brown         light       blue            47
 8 <NA>          white, red  red             NA
 9 black         light       brown           24
10 auburn, white fair        blue-gray       57
# ℹ 77 more rows
```

```
# Part 3c
starwars |>
  select(ends_with('color') | where(is.numeric))
```

```
# A tibble: 87 × 6
   hair_color    skin_color  eye_color height  mass birth_year
   <chr>         <chr>       <chr>      <int> <dbl>      <dbl>
 1 blond         fair        blue         172    77       19
 2 <NA>          gold        yellow       167    75      112
 3 <NA>          white, blue red           96    32       33
 4 none          white       yellow       202   136       41.9
 5 brown         light       brown        150    49       19
 6 brown, grey   light       blue         178   120       52
 7 brown         light       blue         165    75       47
 8 <NA>          white, red  red           97    32       NA
 9 black         light       brown        183    84       24
10 auburn, white fair        blue-gray    182    77       57
# ℹ 77 more rows
```

Exercise C - (10 min)

1. Create a table of sample standard deviations for each of the quizzes in `gradebook`, where the columns are named according to `[COLUMN NAME]_sd`.
2. Read the help file for the function `n_distinct()` in `dplyr`. Use this function to count up the number of distinct values in each column of `starwars` that contains character data. Name your results according to `n_[COLUMN NAME]s`.
3. Read the help file for the `dplyr` function `n()`. Combine it with across() and other dplyr functions you have learned to display the following table. Each row should correspond to a `homeworld` that occurs at least twice in the `starwars` tibble. There should be three columns, counting up the number of distinct values of `sex`, `species`, and `eye_color`. What happens to the observations for which `homeworld` is missing?
4. For each species with at least two observations, calculate the sample median of all the numeric columns in `starwars`, dropping any missing observations. Why do we obtain the result that we do for members of the "Kaminoan" species?
5. Calculate the std. dev. and interquartile range of all numeric columns of `starwars`, dropping missing observations. Attach meaningful names to your results.

Solution

```
# Part 1
gradebook |>
  summarize(across(starts_with('quiz'), sd, .names = '{.col}_sd'))
```

```
# A tibble: 1 × 3
  quiz1_sd quiz2_sd quiz3_sd
     <dbl>    <dbl>    <dbl>
1     8.33     4.32     9.75
```

```
# Part 2
starwars |>
  summarize(across(where(is.character), n_distinct, .names = 'n_{.col}s'))
```

```
# A tibble: 1 × 8
  n_names n_hair_colors n_skin_colors n_eye_colors n_sexs n_genders n_homeworlds
    <int>         <int>         <int>        <int>  <int>     <int>        <int>
1      87            13            31           15      5         3           49
# ℹ 1 more variable: n_speciess <int>
```

```
# Part 3
starwars |>
  group_by(homeworld) |>
  filter(n() > 1) |>
  summarize(across(c(sex, species, eye_color), n_distinct))
```

```
# A tibble: 10 × 4
   homeworld   sex species eye_color
   <chr>     <int>   <int>     <int>
 1 Alderaan      2       1         1
 2 Corellia      1       1         2
 3 Coruscant     2       2         1
 4 Kamino        2       2         2
 5 Kashyyyk      1       1         1
 6 Mirial        1       1         1
 7 Naboo         4       4         5
 8 Ryloth        2       1         2
 9 Tatooine      3       2         4
10 <NA>          4       4         8
```

```
starwars |>
  group_by(homeworld) |>
  filter(n() > 1) |>
  summarize(across(c(sex, species, eye_color), n_distinct))
```

```
# A tibble: 10 × 4
   homeworld   sex species eye_color
   <chr>     <int>   <int>     <int>
 1 Alderaan      2       1         1
 2 Corellia      1       1         2
 3 Coruscant     2       2         1
 4 Kamino        2       2         2
 5 Kashyyyk      1       1         1
 6 Mirial        1       1         1
 7 Naboo         4       4         5
 8 Ryloth        2       1         2
 9 Tatooine      3       2         4
10 <NA>          4       4         8
```

```
# Part 4
starwars |>
  group_by(species) |>
  filter(n() > 1) |>
  summarize(across(where(is.numeric), \(x) median(x, na.rm = TRUE)))
```

```
# A tibble: 9 × 4
  species  height  mass birth_year
  <chr>     <dbl> <dbl>      <dbl>
1 Droid        97  53.5         33
2 Gungan      206  74           52
3 Human       180  79           48
4 Kaminoan    221  88           NA
5 Mirialan    168  53.1         49
6 Twi'lek     179  55           48
7 Wookiee     231 124          200
8 Zabrak      173  80           54
9 <NA>        183  48           62
```

```
starwars |>
  filter(species == 'Kaminoan')
```

```
# A tibble: 2 × 14
  name    height  mass hair_color skin_color eye_color birth_year sex    gender
  <chr>    <int> <dbl> <chr>      <chr>      <chr>          <dbl> <chr>  <chr>
1 Lama Su    229    88 none       grey       black             NA male   mascul…
2 Taun We    213    NA none       grey       black             NA female femini…
# ℹ 5 more variables: homeworld <chr>, species <chr>, films <list>,
#   vehicles <list>, starships <list>
```

```
# Part 5
SD_IQR <- list(
  SD = \(x) sd(x, na.rm = TRUE),
  IQR = \(x) IQR(x, na.rm = TRUE)
)
starwars |>
  summarize(across(where(is.numeric), SD_IQR, .names = '{.col}_{.fn}'))
```

```
# A tibble: 1 × 6
  height_SD height_IQR mass_SD mass_IQR birth_year_SD birth_year_IQR
      <dbl>      <dbl>   <dbl>    <dbl>          <dbl>          <dbl>
1      34.8         24    169.     28.9           155.             37
```

Exercise D - (3 min)

Recode the `race` and `hsgrad` variables from `star` as indicated above.

Solution

```
star <- star |>
  mutate(classtype = case_match(classtype,
                                1 ~ 'small',
                                2 ~ 'regular',
                                3 ~ 'regular+aid'),
         race = case_match(race,
                           1 ~ 'White',
                           2 ~ 'Black',
                           3 ~ 'Asian',
                           4 ~ 'Hispanic',
                           5 ~ 'Native American',
                           6 ~ 'Other'),
         hsgrad = if_else(hsgrad == 1, 'graduate', 'non-graduate'))
```