Homework 04 Submission

Jimmy Liu

2019-10-08

Contents

1 Exercise 1

1.1 Widen gapminder dataset by creating a lifeExp column for each country

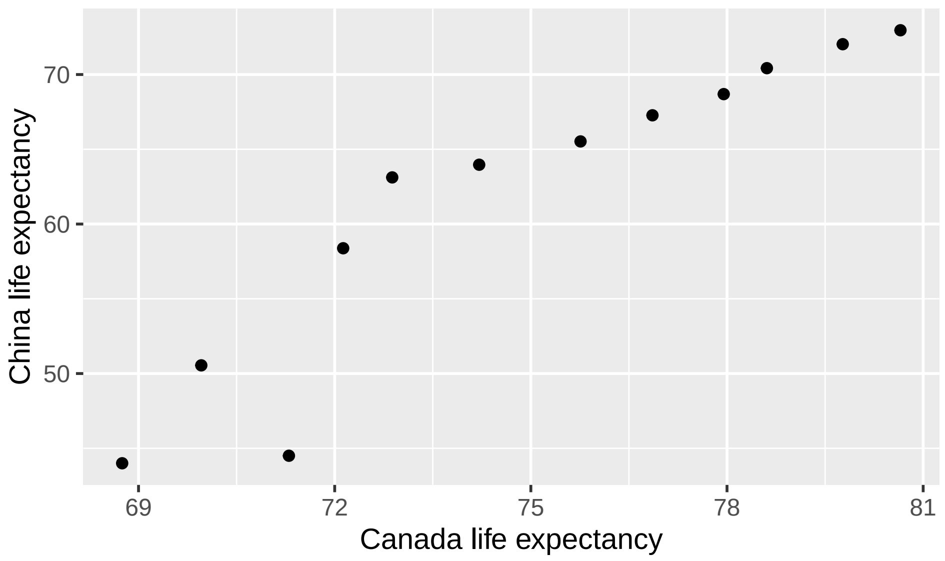
```
gapminder_wide <- gapminder %>%
  filter(country %in% c("Canada", "China")) %>% # Filter for rows with Canada or China
  pivot_wider(id_cols = c(country, year), # Keep countries and year columns
              values_from = lifeExp, # Cells of new columns will contain lifeExp values
              names_from = country) # Newly generated columns are named by the values
                                    # of the country column

gapminder_wide %>%
  knitr::kable() # Present the transformed tibble in a nice table
```

year	Canada	China
1952	68.750	44.00000
1957	69.960	50.54896
1962	71.300	44.50136
1967	72.130	58.38112
1972	72.880	63.11888
1977	74.210	63.96736
1982	75.760	65.52500
1987	76.860	67.27400
1992	77.950	68.69000
1997	78.610	70.42600
2002	79.770	72.02800
2007	80.653	72.96100

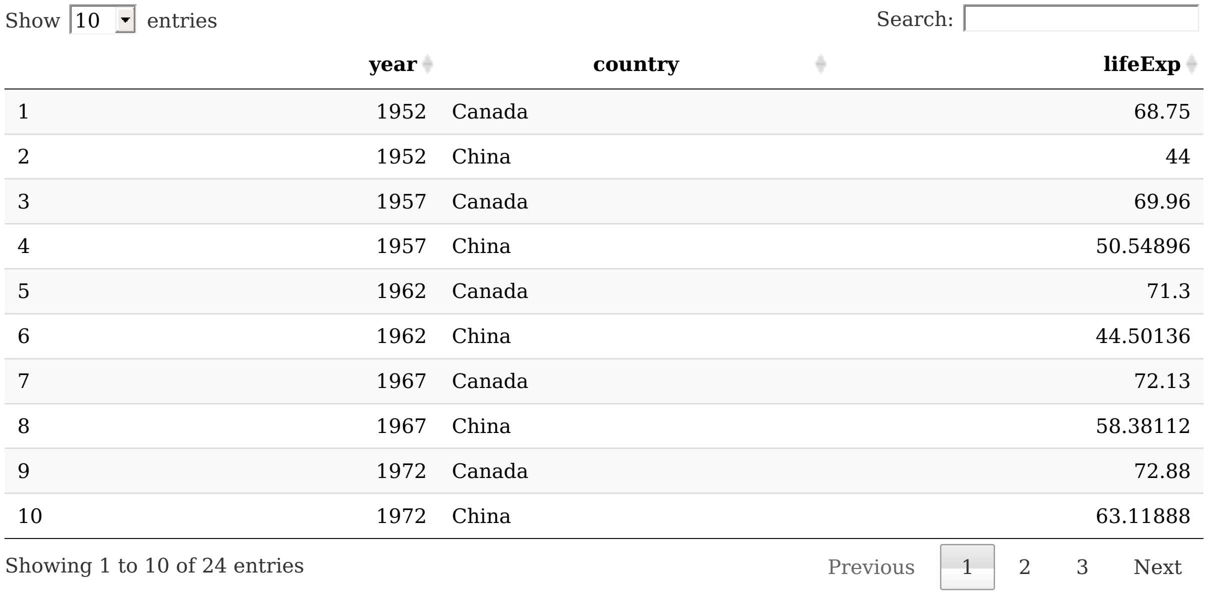
1.2 Create a scatterplot of life expectancies in Canada vs China

```
gapminder_wide %>%
  ggplot(aes(x = Canada, y = China)) + # Specify the x and y variables for ggplot
  geom_point() + # Add point aesthetics for each data point
  xlab("Canada life expectancy") + # label x axis
  ylab("China life expectancy") # label y axis
```



1.3 Re-lengthen the widen gapminder dataset

```r
gapminder_wide %>%
  pivot_longer(cols = (-year), # Select all columns except year to be collapsed
               names_to = "country", # Collapse Canada and China columns to
                                     # a column called "country"
               values_to = "lifeExp") %>% # Values of Canda and China columns
                                          # will be under a column called "lifeExp"
  DT::datatable() # Present the transformed tibble in a nice table
```



	year	country	lifeExp
1	1952	Canada	68.75
2	1952	China	44
3	1957	Canada	69.96
4	1957	China	50.54896
5	1962	Canada	71.3
6	1962	China	44.50136
7	1967	Canada	72.13
8	1967	China	58.38112
9	1972	Canada	72.88
10	1972	China	63.11888

Showing 1 to 10 of 24 entries

2 Exercise 2

2.1 Widen gapminder dataset by creating lifeExp and gdpPerCap columns for each country

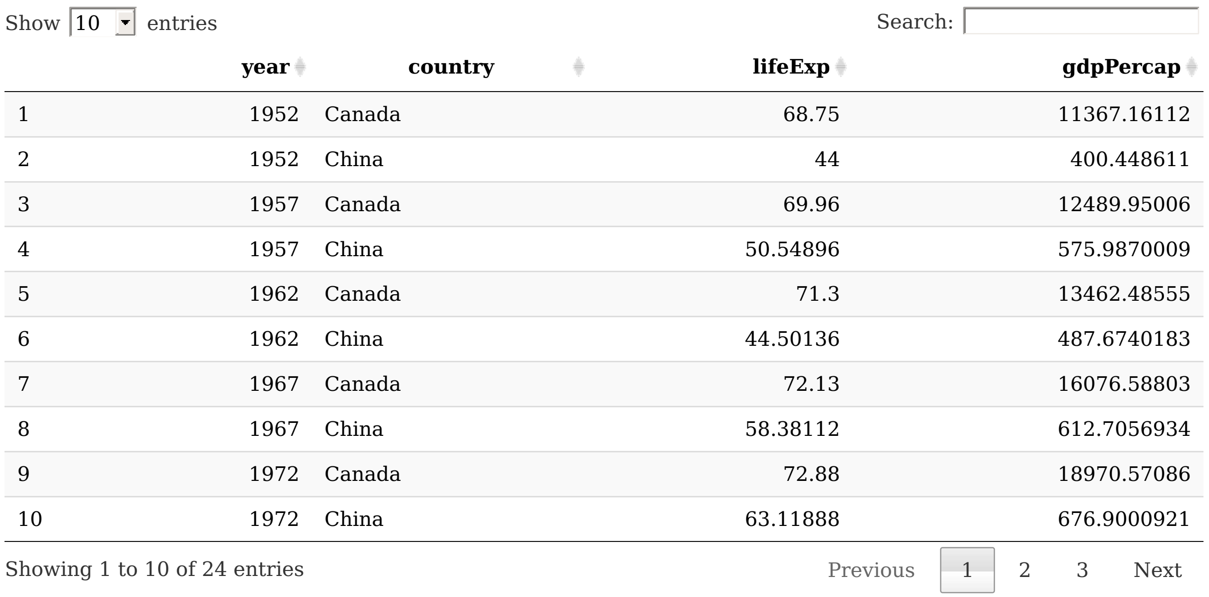
```
gapminder_wide <- gapminder %>%
 filter(country %in% c("Canada", "China")) %>% # Filter for rows with Canada or China
 pivot_wider(id_cols = c(country, year), # Keep countries and year columns
             names_from = country, # Newly generated columns are named by
                                   # the values of the country column
             names_sep = "_", # Each new column name is separated by "_"
             values_from = c(lifeExp, gdpPercap)) # Cells of the new columns will contain
                                                  # either lifeExp or gdpPerCap values

gapminder_wide %>%
  knitr::kable() # Present the transformed tibble in a nice table
```

year	lifeExp_Canada	lifeExp_China	gdpPercap_Canada	gdpPercap_China
1952	68.750	44.00000	11367.16	400.4486
1957	69.960	50.54896	12489.95	575.9870
1962	71.300	44.50136	13462.49	487.6740
1967	72.130	58.38112	16076.59	612.7057
1972	72.880	63.11888	18970.57	676.9001
1977	74.210	63.96736	22090.88	741.2375
1982	75.760	65.52500	22898.79	962.4214
1987	76.860	67.27400	26626.52	1378.9040
1992	77.950	68.69000	26342.88	1655.7842
1997	78.610	70.42600	28954.93	2289.2341
2002	79.770	72.02800	33328.97	3119.2809
2007	80.653	72.96100	36319.24	4959.1149

2.1.1 Re-lengthen the widen gapminder dataset

```
gapminder_wide %>%
  pivot_longer(cols = (-year), # Collapse all columns except the year column
               names_sep = "_", # Split the column names by "_"
               names_to = c(".value", "country")) %>% # Collapse selected columns
                                                      # and create 3 new columns
  DT::datatable() # Present the transformed tibble in a nice table
```



	year	country	lifeExp	gdpPercap
1	1952	Canada	68.75	11367.16112
2	1952	China	44	400.448611
3	1957	Canada	69.96	12489.95006
4	1957	China	50.54896	575.9870009
5	1962	Canada	71.3	13462.48555
6	1962	China	44.50136	487.6740183
7	1967	Canada	72.13	16076.58803
8	1967	China	58.38112	612.7056934
9	1972	Canada	72.88	18970.57086
10	1972	China	63.11888	676.9000921

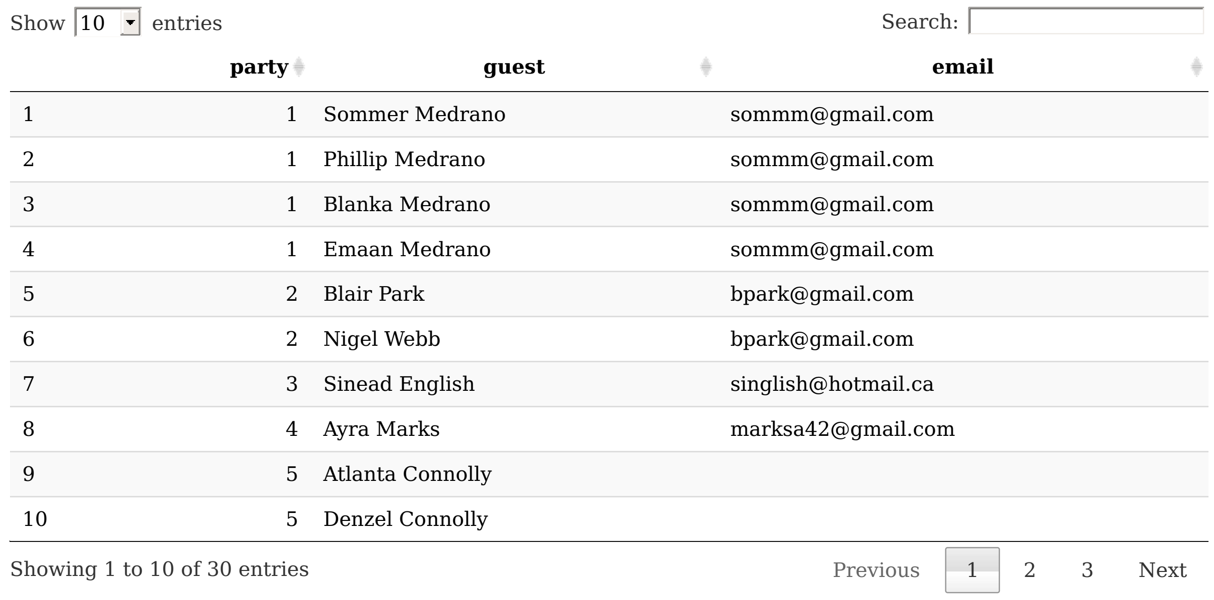
Showing 1 to 10 of 24 entries

3 Exercise 3

```
guest <- read_csv("../data/wedding/attend.csv")
email <- read_csv("../data/wedding/emails.csv")
```

3.1 Merge the email column to the guest dataset

```
guest %>%
  rename("guest" = name) %>% # rename the name variable to "guest"
  left_join((email %>%
               separate_rows("guest", sep = ", ")), # split values in the guest
                                                     # variable into separate rows
            by = "guest") %>% # merge by the guest variable
  select(party, guest, email) %>% # Only keep the party, guest and email columns
  DT::datatable() # Present the new tibble in a nice table
```



	party	guest	email
1	1	Sommer Medrano	sommm@gmail.com
2	1	Phillip Medrano	sommm@gmail.com
3	1	Blanka Medrano	sommm@gmail.com
4	1	Emaan Medrano	sommm@gmail.com
5	2	Blair Park	bpark@gmail.com
6	2	Nigel Webb	bpark@gmail.com
7	3	Sinead English	singlish@hotmail.ca
8	4	Ayra Marks	marksa42@gmail.com
9	5	Atlanta Connolly	
10	5	Denzel Connolly	

Showing 1 to 10 of 30 entries

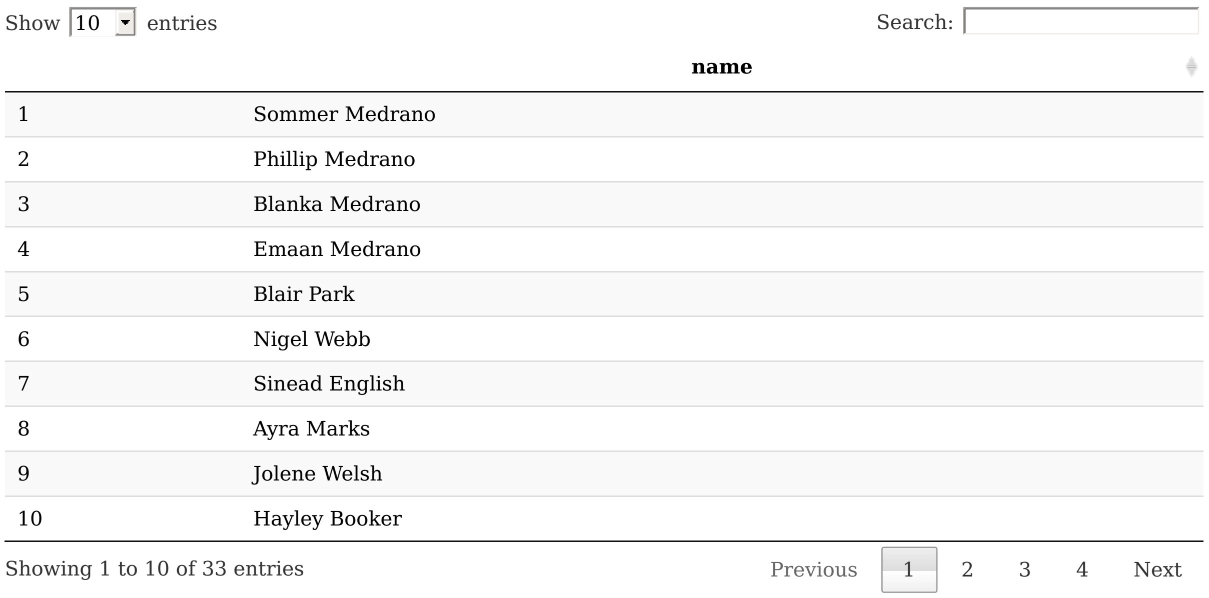
3.2 Report individuals with known emails but not on guest list

```
email %>%
  rename("name" = guest) %>% # rename the guest variable to "name"
  select(name) %>% # keep the name variable only
  separate_rows("name", sep = ", ") %>% # split values in the name
                                        # variable into separate rows
  setdiff(guest %>% # find values in the email tibble not
                    # in the guest tibble
            select(name)) %>%
  knitr::kable() # present the new tibble in a nice table
```

name
Turner Jones
Albert Marshall
Vivian Marshall

3.3 Report everyone in the guest and email datasets

```
email %>%
  rename("name" = guest) %>% # rename the guest variable to "name"
  select(name) %>% # keep the name variable only
  separate_rows("name", sep = ", ") %>% # split values in the name variable into separate rows
  union(guest %>% # join all values found in the email and guest tibble into a single tibble
            select(name)) %>%
DT::datatable() # present the new tibble in a nice table
```



	name
1	Sommer Medrano
2	Phillip Medrano
3	Blanka Medrano
4	Emaan Medrano
5	Blair Park
6	Nigel Webb
7	Sinead English
8	Ayra Marks
9	Jolene Welsh
10	Hayley Booker

Showing 1 to 10 of 33 entries