More data manipulation with dplyr and tidy

Daniel Stoxreiter

NB: The worksheet has beed developed and prepared by Lincoln Mullen. Source: Lincoln A. Mullen, *Computational Historical Thinking: With Applications in R (2018)*: http://dh-r.lincolnmullen.com.

The best way to learn R or computational history is to practice. These worksheets contain a series of questions designed to teach you about R or different computational methods. The worksheets are R Markdown documents that include text and code together. The places where you are expected to answer questions are marked like this.

```
(@) Can you make a plot from this dataset?
```

Beneath each question is a space to either create a code block or write an answer.

Aims of this worksheet

In an earlier worksheet, you learned the basic data manipulation verbs from the dplyr package: `select()`, `filter()`, `mutate()`, `arrange()`, `group_by()`, and `summarize()`. In this worksheet you will learn additional data verbs from the dplyr and tidyr packages. These data verbs relate to window functions (`lead()` and `lag()`), data table joins (`left_join()` et al.), and data reshaping (`spread()` and `gather()`)

To begin, we will load the necessary packages, as well as the Methodist data.

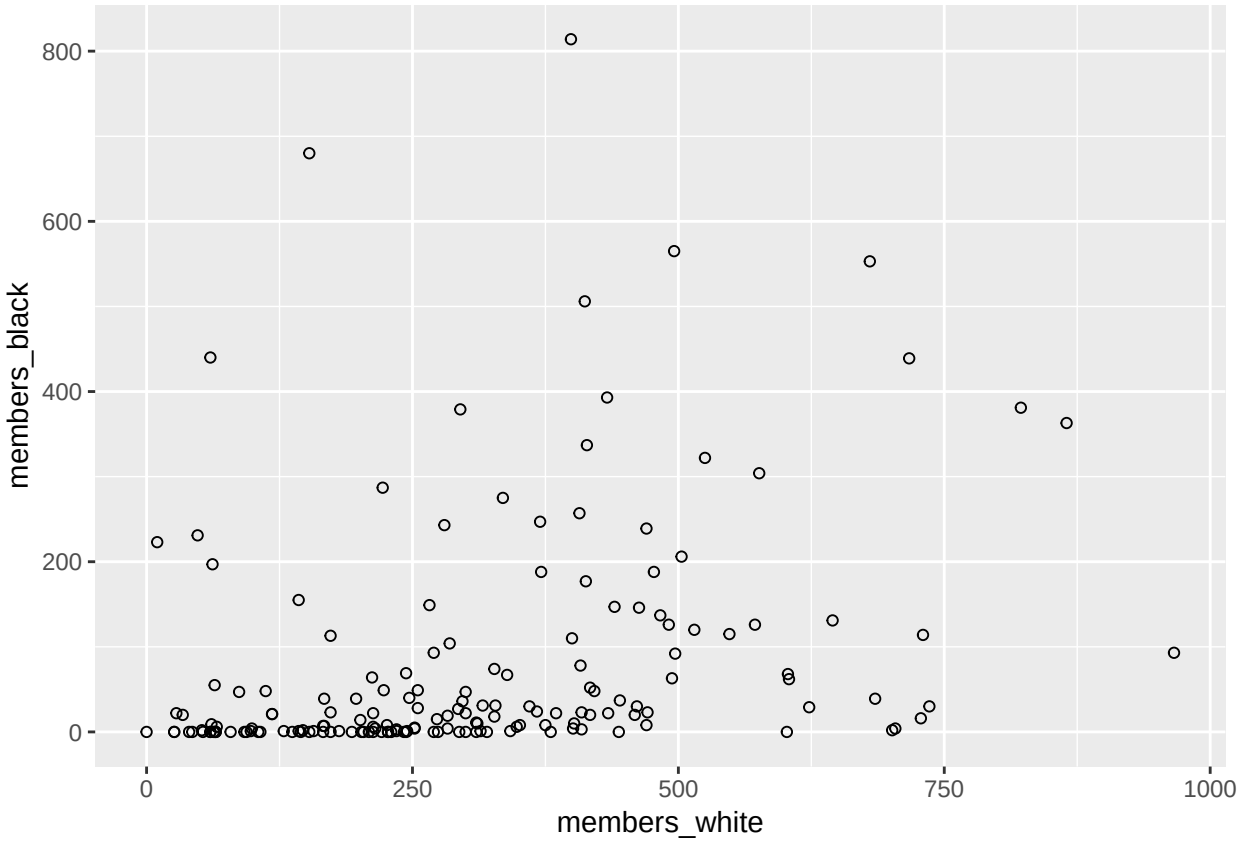
```
library(tidyverse)
library(historydata)
load("C:/Users/Daniel/Documents/R_course/R-univie/lession5/methodists.rda")
```

Data joining with two table verbs (`left_join()` et al.)

It is often the case that we want to use some variable in our data to create a new variable. Consider the Methodist data for the year 1800. Perhaps we are interested in the racial composition of the churches. Do they tend to be all white and all black, or do some churches have both white and black members in varying proportions? The simplest way to get a look at that question is to create a scatter plot of the figures for white and black membership.

```
methodists_1800 <- methodists %>%
  filter(year == 1800) %>%
  select(year, meeting, state, members_white, members_black)

ggplot(methodists_1800, aes(x = members_white, y = members_black)) +
  geom_point(shape = 1)
```



That scatterplot is interesting as far as it goes, but we might reasonably suspect that the racial composition of methodist meetings varies by region. We could use the `state` variable to facet the plot by state. However, this has two problems. There are 20 states represented in that year. Our faceted plot would have 20 panels, which is too many. But more important, by looking at individual states we might be getting *too* fine grained a look at the data. We have good reason to think that it is regions that matter more than states.

It is easy enough to describe what we would do to translate states into a new column with regions. We would look at each state name and assign it to a region. Connecticut would be in the Northeast, New York would be in the Mid-Atlantic, and so on. We can think of this problem as looking up a value in one table (our Methodist data) in another table. That other table will have a row for each state, where each state name is associated with a region. (In many cases, though, it would make more sense to create a CSV file with the data and read it in as a data frame.)

```
regions <- data_frame(
  state = c("Connecticut", "Delaware", "Georgia", "Kentucky", "Maine",
            "Maryland", "Massachusetts", "Mississippi", "New Hampshire",
            "New Jersey", "New York", "North Carolina",
            "Northwestern Territory", "Pennsylvania", "Rhode Island",
            "South Carolina", "Tennessee", "Upper Canada", "Vermont",
            "Virginia"),
  region = c("Northeast", "Atlantic South", "Atlantic South", "West",
             "Northeast", "Atlantic South", "Northeast", "Deep South",
             "Northeast", "Mid-Atlantic", "Mid-Atlantic", "Atlantic South",
             "West", "Mid-Atlantic", "Northeast", "Atlantic South", "West",
             "Canada", "Northeast", "Atlantic South")
)
```

```
## Warning: `data_frame()` is deprecated, use `tibble()`.
## This warning is displayed once per session.
```

And now we can inspect the table.

```
regions
```

```
## # A tibble: 20 x 2
##    state                  region
##    <chr>                  <chr>
##  1 Connecticut            Northeast
##  2 Delaware               Atlantic South
##  3 Georgia                Atlantic South
##  4 Kentucky               West
##  5 Maine                  Northeast
##  6 Maryland               Atlantic South
##  7 Massachusetts          Northeast
##  8 Mississippi            Deep South
##  9 New Hampshire          Northeast
## 10 New Jersey             Mid-Atlantic
## 11 New York               Mid-Atlantic
## 12 North Carolina         Atlantic South
## 13 Northwestern Territory West
## 14 Pennsylvania           Mid-Atlantic
## 15 Rhode Island           Northeast
## 16 South Carolina         Atlantic South
## 17 Tennessee              West
## 18 Upper Canada           Canada
## 19 Vermont                Northeast
## 20 Virginia               Atlantic South
```

We can do a look up where we take the `state` column in the `methodists_1800` data frame and associate it with the `states` column in our `regions` data frame. The result will be a new column `region`. Notice how we use the `by =` argument to specify which column in the left hand table matches which column in the right hand table.

```
methodists_region <- methodists_1800 %>%
  left_join(regions, by = "state")

methodists_region
```

```
##     year                   meeting                 state members_white
## 1   1800                   Augusta               Georgia            61
## 2   1800                     Burke               Georgia           297
## 3   1800                  Richmond               Georgia           548
## 4   1800                Washington               Georgia           497
## 5   1800               Broad River        South Carolina           604
## 6   1800                Bush River        South Carolina           328
## 7   1800                Charleston        South Carolina            60
## 8   1800                  Cherokee        South Carolina            79
## 9   1800                    Edisto        South Carolina           572
## 10  1800                Georgetown        South Carolina            10
```

```
## 11  1800              Great Pee Dee        South Carolina          212
## 12  1800   Little Pee Dee and Anson        South Carolina          603
## 13  1800         Santee and Catawba        South Carolina          470
## 14  1800                    Seleuda        South Carolina          461
## 15  1800     Banks and Mattamuskeet        North Carolina          213
## 16  1800                     Bertie        North Carolina          371
## 17  1800                     Bladen        North Carolina          730
## 18  1800                     Camden        North Carolina          412
## 19  1800                    Caswell        North Carolina          515
## 20  1800                  Contentney        North Carolina          167
## 21  1800                      Goshen        North Carolina          235
## 22  1800                    Guilford        North Carolina          685
## 23  1800                   Haw River        North Carolina          244
## 24  1800                     Newbern        North Carolina          280
## 25  1800                     Pamlico        North Carolina          173
## 26  1800                     Roanoke        North Carolina          222
## 27  1800                   Salisbury        North Carolina          471
## 28  1800                     Swanino        North Carolina          226
## 29  1800                   Tar River        North Carolina          491
## 30  1800                       Union        North Carolina          421
## 31  1800                  Wilmington        North Carolina           48
## 32  1800                      Yadkin        North Carolina          459
## 33  1800                  Cumberland             Tennessee          247
## 34  1800                       Green             Tennessee          434
## 35  1800                  Alexandria              Virginia           64
## 36  1800          Alleghany and Bath              Virginia          283
## 37  1800                      Amelia              Virginia          445
## 38  1800                     Amherst              Virginia          400
## 39  1800                     Bedford              Virginia          440
## 40  1800                     Berkley              Virginia          417
## 41  1800                  Bottetourt              Virginia          197
## 42  1800                   Brunswick              Virginia          413
## 43  1800                  Clarksburg              Virginia          401
## 44  1800                  Cumberland              Virginia          300
## 45  1800                     Fairfax              Virginia          300
## 46  1800                    Franklin              Virginia          409
## 47  1800                  Gloucester              Virginia          966
## 48  1800                  Greenbrier              Virginia          348
## 49  1800 Greensville and Mecklenburg              Virginia          865
## 50  1800                     Hanover              Virginia          255
## 51  1800                     Holston              Virginia          385
## 52  1800                   Lancaster              Virginia          266
## 53  1800              Little Kanawha              Virginia           60
## 54  1800                   New River              Virginia          118
## 55  1800      Norfolk and Portsmouth              Virginia          143
## 56  1800                 Northampton              Virginia          335
## 57  1800                        Ohio              Virginia          311
## 58  1800                     Orange              Virginia          367
## 59  1800                   Pendleton              Virginia           99
## 60  1800                  Portsmouth              Virginia          503
## 61  1800                    Richmond              Virginia           28
## 62  1800                  Rockingham              Virginia          293
## 63  1800                     Russell              Virginia          118
## 64  1800                    Stafford              Virginia          255
```

```
## 65  1800                  Sussex            Virginia          463
## 66  1800              Williamsburg            Virginia          327
## 67  1800                Winchester            Virginia          285
## 68  1800                  Danville            Kentucky          339
## 69  1800                 Hinkstone            Kentucky          283
## 70  1800                 Lexington            Kentucky          273
## 71  1800                 Limestone            Kentucky          417
## 72  1800                Salt River            Kentucky          147
## 73  1800                    Shelby            Kentucky          167
## 74  1800                Annamessex            Maryland          173
## 75  1800                 Annapolis            Maryland           62
## 76  1800          Baltimore Circuit            Maryland          408
## 77  1800                 Baltimore            Maryland          576
## 78  1800            Town and Point            Maryland          112
## 79  1800                   Calvert            Maryland          399
## 80  1800                  Caroline            Maryland          477
## 81  1800                     Cecil            Maryland          525
## 82  1800                Dorchester            Maryland          680
## 83  1800                   Federal            Maryland          414
## 84  1800                 Frederick            Maryland          223
## 85  1800                   Harford            Maryland          270
## 86  1800                      Kent            Maryland          295
## 87  1800                Montgomery            Maryland          370
## 88  1800          Prince George's            Maryland          153
## 89  1800              Queen Ann's            Maryland          496
## 90  1800                 Somerset            Maryland          483
## 91  1800                   Talbot            Maryland          433
## 92  1800                    Dover            Delaware          717
## 93  1800                  Milford            Delaware          822
## 94  1800                Wilmington            Delaware           87
## 95  1800                   Bristol        Pennsylvania          166
## 96  1800                  Carlisle        Pennsylvania          213
## 97  1800       Chester and Strasburg        Pennsylvania          402
## 98  1800                Huntingdon        Pennsylvania          215
## 99  1800             Northumberland        Pennsylvania          244
## 100 1800                  Pittsburg        Pennsylvania          470
## 101 1800               Philadelphia        Pennsylvania          407
## 102 1800                   Redstone        Pennsylvania          375
## 103 1800                      Tioga        Pennsylvania          202
## 104 1800                    Wyoming        Pennsylvania          193
## 105 1800                     Bethel          New Jersey          736
## 106 1800                  Burlington          New Jersey          623
## 107 1800                Elizabethtown          New Jersey          252
## 108 1800                   Flanders          New Jersey          235
## 109 1800                   Freehold          New Jersey          316
## 110 1800                      Salem          New Jersey          494
## 111 1800                    Trenton          New Jersey          201
## 112 1800                Albany City            New York           40
## 113 1800              Albany Circuit            New York          704
## 114 1800                   Brooklyn            New York           34
## 115 1800                  Cambridge            New York          701
## 116 1800                   Chenango            New York          227
## 117 1800                   Columbia            New York          143
## 118 1800                   Delaware            New York          380
```

```
## 119 1800                   Dutchess               New York          310
## 120 1800                   Herkimer               New York          294
## 121 1800                Long Island               New York          360
## 122 1800                     Mohawk               New York          242
## 123 1800                    Newburg               New York          351
## 124 1800     New Rochelle and Croton               New York          728
## 125 1800                   New York               New York          645
## 126 1800           Oneida and Cayuga               New York          209
## 127 1800                  Plattsburg               New York          107
## 128 1800                    Saratoga               New York          444
## 129 1800                      Seneca               New York          221
## 130 1800                  Litchfield            Connecticut          314
## 131 1800                  Middletown            Connecticut          252
## 132 1800                  New London            Connecticut          327
## 133 1800                     Pomfret            Connecticut          181
## 134 1800                     Redding            Connecticut          227
## 135 1800                     Tolland            Connecticut          245
## 136 1800                   Greenwich           Rhode Island           43
## 137 1800                Rhode Island           Rhode Island           52
## 138 1800                      Warren           Rhode Island          129
## 139 1800                      Boston          Massachusetts           66
## 140 1800                   Granville          Massachusetts          300
## 141 1800                        Lynn          Massachusetts           94
## 142 1800                  Marblehead          Massachusetts           26
## 143 1800           Martha's Vineyard          Massachusetts            0
## 144 1800                   Merrimack          Massachusetts           65
## 145 1800                     Needham          Massachusetts          153
## 146 1800                   Nantucket          Massachusetts           65
## 147 1800                  Pittsfield          Massachusetts          602
## 148 1800                Provincetown          Massachusetts          137
## 149 1800                    Sandwich          Massachusetts           63
## 150 1800                Chesterfield          New Hampshire          145
## 151 1800                       Hawke          New Hampshire           26
## 152 1800                Bath and Union                  Maine          173
## 153 1800                   Norridgwock                  Maine          166
## 154 1800                     Penobscot                  Maine          213
## 155 1800                      Portland                  Maine          230
## 156 1800                     Readfield                  Maine          310
## 157 1800                   Union River                  Maine          105
## 158 1800                         Essex                Vermont          274
## 159 1800                       Landaff                Vermont           53
## 160 1800                     Vergennes                Vermont          342
## 161 1800                      Vershire                Vermont          270
## 162 1800                   Wethersfield                Vermont           64
## 163 1800                     Whitingham                Vermont           92
## 164 1800                          Miami Northwestern Territory           98
## 165 1800                         Scioto Northwestern Territory          157
## 166 1800                       Natchez            Mississippi           60
## 167 1800                   Bay Quintie           Upper Canada          409
## 168 1800                       Niagara           Upper Canada          204
## 169 1800                   Oswegotchie           Upper Canada          320
##     members_black         region
## 1               9 Atlantic South
## 2              36 Atlantic South
```

```
## 3             115 Atlantic South
## 4              92 Atlantic South
## 5              62 Atlantic South
## 6              31 Atlantic South
## 7             440 Atlantic South
## 8               0 Atlantic South
## 9             126 Atlantic South
## 10            223 Atlantic South
## 11             64 Atlantic South
## 12             68 Atlantic South
## 13            239 Atlantic South
## 14             30 Atlantic South
## 15             22 Atlantic South
## 16            188 Atlantic South
## 17            114 Atlantic South
## 18            506 Atlantic South
## 19            120 Atlantic South
## 20             39 Atlantic South
## 21              3 Atlantic South
## 22             39 Atlantic South
## 23             69 Atlantic South
## 24            243 Atlantic South
## 25             23 Atlantic South
## 26            287 Atlantic South
## 27             23 Atlantic South
## 28              8 Atlantic South
## 29            126 Atlantic South
## 30             48 Atlantic South
## 31            231 Atlantic South
## 32             20 Atlantic South
## 33             40           West
## 34             22           West
## 35             55 Atlantic South
## 36             19 Atlantic South
## 37             37 Atlantic South
## 38            110 Atlantic South
## 39            147 Atlantic South
## 40             52 Atlantic South
## 41             39 Atlantic South
## 42            177 Atlantic South
## 43              4 Atlantic South
## 44             22 Atlantic South
## 45             47 Atlantic South
## 46             23 Atlantic South
## 47             93 Atlantic South
## 48              6 Atlantic South
## 49            363 Atlantic South
## 50             49 Atlantic South
## 51             22 Atlantic South
## 52            149 Atlantic South
## 53              0 Atlantic South
## 54             21 Atlantic South
## 55            155 Atlantic South
## 56            275 Atlantic South
```

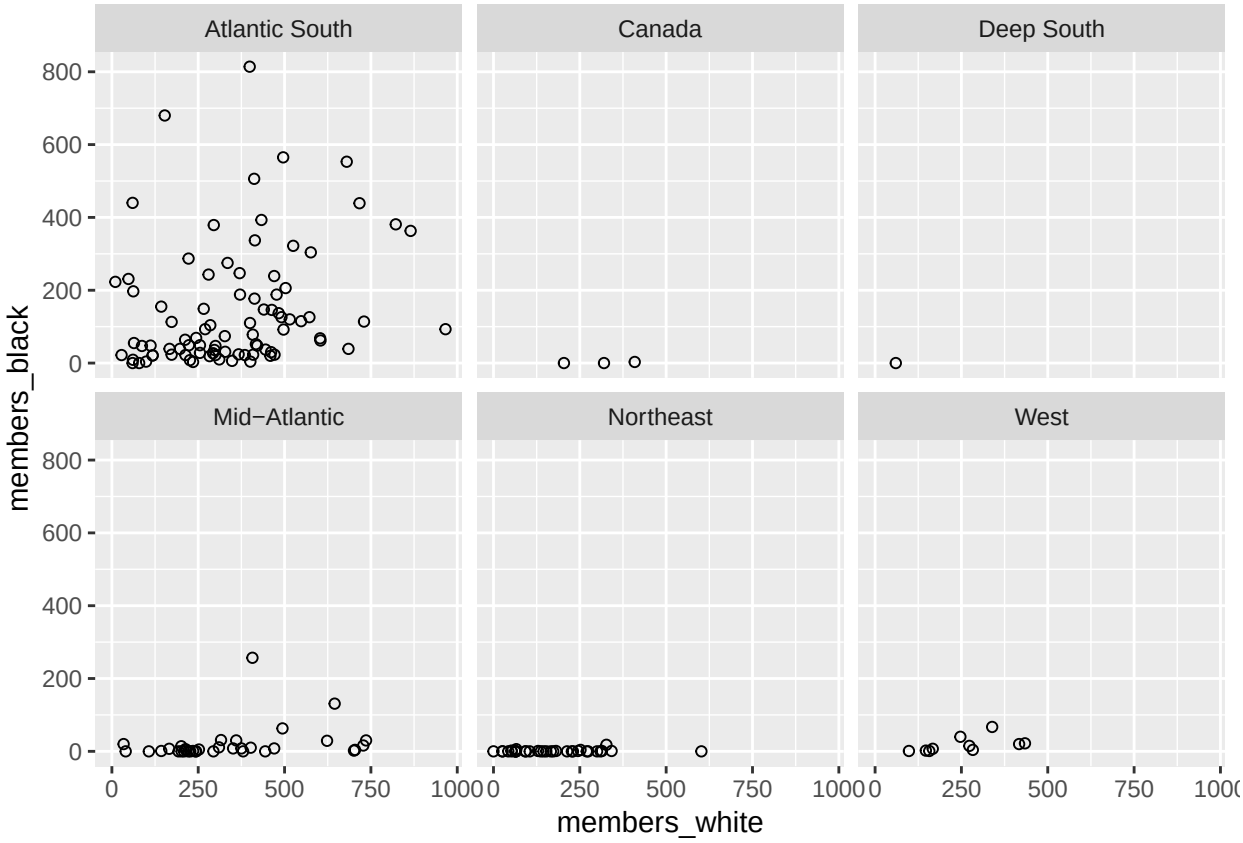
```
## 57             10 Atlantic South
## 58             24 Atlantic South
## 59              4 Atlantic South
## 60            206 Atlantic South
## 61             22 Atlantic South
## 62             27 Atlantic South
## 63             21 Atlantic South
## 64             28 Atlantic South
## 65            146 Atlantic South
## 66             74 Atlantic South
## 67            104 Atlantic South
## 68             67           West
## 69              4           West
## 70             15           West
## 71             20           West
## 72              2           West
## 73              7           West
## 74            113 Atlantic South
## 75            197 Atlantic South
## 76             78 Atlantic South
## 77            304 Atlantic South
## 78             48 Atlantic South
## 79            814 Atlantic South
## 80            188 Atlantic South
## 81            322 Atlantic South
## 82            553 Atlantic South
## 83            337 Atlantic South
## 84             49 Atlantic South
## 85             93 Atlantic South
## 86            379 Atlantic South
## 87            247 Atlantic South
## 88            680 Atlantic South
## 89            565 Atlantic South
## 90            137 Atlantic South
## 91            393 Atlantic South
## 92            439 Atlantic South
## 93            381 Atlantic South
## 94             47 Atlantic South
## 95              7   Mid-Atlantic
## 96              6   Mid-Atlantic
## 97             10   Mid-Atlantic
## 98              4   Mid-Atlantic
## 99              0   Mid-Atlantic
## 100             8   Mid-Atlantic
## 101           257   Mid-Atlantic
## 102             8   Mid-Atlantic
## 103             0   Mid-Atlantic
## 104             0   Mid-Atlantic
## 105            30   Mid-Atlantic
## 106            29   Mid-Atlantic
## 107             5   Mid-Atlantic
## 108             1   Mid-Atlantic
## 109            31   Mid-Atlantic
## 110            63   Mid-Atlantic
```

```
## 111            14   Mid-Atlantic
## 112             0   Mid-Atlantic
## 113             4   Mid-Atlantic
## 114            20   Mid-Atlantic
## 115             2   Mid-Atlantic
## 116             0   Mid-Atlantic
## 117             1   Mid-Atlantic
## 118             0   Mid-Atlantic
## 119            11   Mid-Atlantic
## 120             0   Mid-Atlantic
## 121            30   Mid-Atlantic
## 122             0   Mid-Atlantic
## 123             8   Mid-Atlantic
## 124            16   Mid-Atlantic
## 125           131   Mid-Atlantic
## 126             0   Mid-Atlantic
## 127             0   Mid-Atlantic
## 128             0   Mid-Atlantic
## 129             0   Mid-Atlantic
## 130             1      Northeast
## 131             4      Northeast
## 132            18      Northeast
## 133             1      Northeast
## 134             0      Northeast
## 135             1      Northeast
## 136             0      Northeast
## 137             2      Northeast
## 138             1      Northeast
## 139             6      Northeast
## 140             0      Northeast
## 141             0      Northeast
## 142             0      Northeast
## 143             0      Northeast
## 144             0      Northeast
## 145             0      Northeast
## 146             0      Northeast
## 147             0      Northeast
## 148             0      Northeast
## 149             0      Northeast
## 150             0      Northeast
## 151             0      Northeast
## 152             0      Northeast
## 153             0      Northeast
## 154             0      Northeast
## 155             0      Northeast
## 156             0      Northeast
## 157             0      Northeast
## 158             0      Northeast
## 159             0      Northeast
## 160             1      Northeast
## 161             0      Northeast
## 162             0      Northeast
## 163             0      Northeast
## 164             1           West
```

```
## 165             1           West
## 166             0     Deep South
## 167             3         Canada
## 168             0         Canada
## 169             0         Canada
```

Then we can plot the results. As we suspected, there is a huge regional variation.

```
ggplot(methodists_region, aes(x = members_white, y = members_black)) +
  geom_point(shape = 1) +
  facet_wrap(~ region)
```



(1) Can you summarize the racial composition of the different regions by year (i.e., a region had a certain percentage white and black members for a given year) and create a plot of the changing racial composition in each region over time?

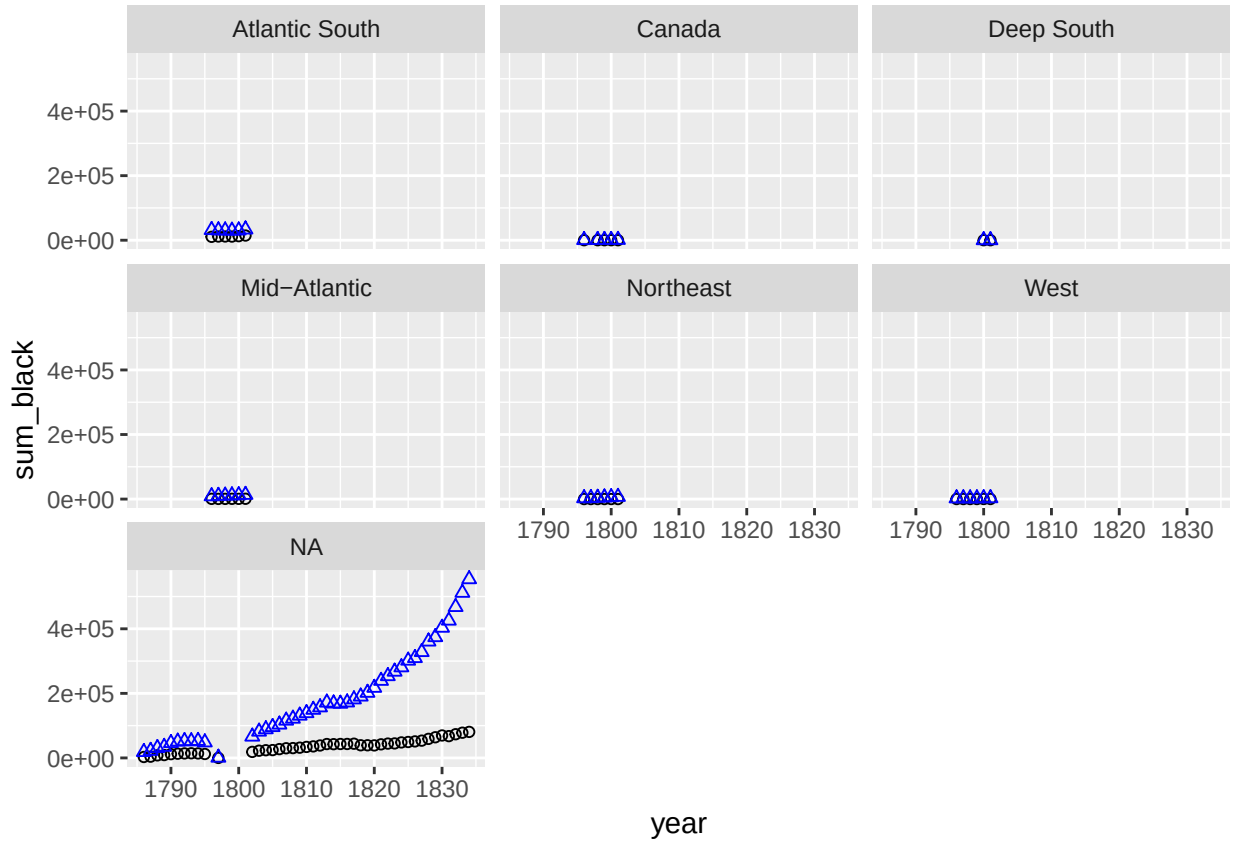
```
methodists_region <- methodists %>%
  left_join(regions, by = "state")

racial_composition <- methodists_region %>%
  group_by(year, region) %>%
  summarise(sum_white = sum(members_white), sum_black = sum(members_black))
racial_composition
```

```
## # A tibble: 75 x 4
```

```
## # Groups:   year [49]
##     year region sum_white sum_black
##    <int> <chr>      <int>     <int>
##  1  1786 <NA>       18291      2890
##  2  1787 <NA>       21949      3883
##  3  1788 <NA>       30557      7991
##  4  1789 <NA>       34425      8840
##  5  1790 <NA>       45983     11682
##  6  1791 <NA>       50580     13098
##  7  1792 <NA>       52079     13871
##  8  1793 <NA>       51486     14420
##  9  1794 <NA>       52794     13906
## 10  1795 <NA>       48121     12171
## # ... with 65 more rows
```

```
ggplot(racial_composition, aes(x = year, y = sum_black)) +
  geom_point(shape = 1, col = "black") +
  geom_point(aes(y = sum_white), shape = 2, col = "blue") +
  facet_wrap(~ region)
```



(2) In the europop package there are two data frames, `europop` with the historical populations of European cities, and `city_coords` which has the latitudes and longitudes of those cities. Load that package and join the two tables together. Can you get the populations of cities north of 48° of latitude?

```
library(europop)
joined_europop <- europop %>%
  left_join(city_coords, by = "city")
joined_europop
```

```
## # A tibble: 2,653 x 6
##    city       region             year population    lon   lat
##    <chr>      <chr>             <int>      <int>  <dbl> <dbl>
##  1 BERGEN     Scandinavia        1500          0   5.33  60.4
##  2 COPENHAGEN Scandinavia        1500         NA  12.6   55.7
##  3 GOTEBORG   Scandinavia        1500          0  12.0   57.7
##  4 KARLSKRONA Scandinavia        1500          0  15.6   56.2
##  5 OSLO       Scandinavia        1500          0  10.7   59.9
##  6 STOCKHOLM  Scandinavia        1500          0  18.1   59.3
##  7 BATH       England and Wales  1500          0  -2.36  51.4
##  8 BIRMINGHAM England and Wales  1500          0  -1.90  52.5
##  9 BLACKBURN  England and Wales  1500          0  -2.48  53.8
## 10 BOLTON     England and Wales  1500          0  -2.43  53.6
## # ... with 2,643 more rows
```

```
joined_europop %>%
  filter(lat > 48) %>%
  arrange(desc(lat)) %>%
  group_by(city) %>%
  summarise(sum_pop = sum(population, na.rm=TRUE))
```

```
## # A tibble: 202 x 2
##    city             sum_pop
##    <chr>              <int>
##  1 'S HERTOGENBOSCH     112
##  2 AACHEN                66
##  3 AALST                 48
##  4 ABBEVILLE             48
##  5 ABERDEEN              56
##  6 ALENCON               37
##  7 ALKMAAR               67
##  8 ALTONA                55
##  9 AMIENS               126
## 10 AMSTERDAM            911
## # ... with 192 more rows
```

(3) In the historydata package there are two tables, `judges_people` and `judges_appointments`. Join them together. What are the names of black judges who were appointed to the Supreme Court?

```
library(historydata)

joined_judges <- judges_people %>%
  left_join(judges_appointments, by ="judge_id")

black_supreme <- joined_judges %>%
  mutate(name = paste(name_first, name_last, sep = " ")) %>%
  filter(race == "African American", court_name == "Supreme Court of the United States") %>%
```

```
  select(name, race, court_name)
black_supreme
```

```
## # A tibble: 2 x 3
##   name              race             court_name
##   <chr>             <chr>            <chr>
## 1 Thurgood Marshall African American Supreme Court of the United States
## 2 Clarence Thomas   African American Supreme Court of the United States
```

(4) What courts did those justices serve on before the Supreme Court?

```
joined_judges %>%
  mutate(name = paste(name_first, name_last, sep = " ")) %>%
  filter(court_name != "Supreme Court of the United States", name == "Clarence Thomas") %>%
  select(name, court_name)
```

```
## # A tibble: 1 x 2
##   name            court_name
##   <chr>           <chr>
## 1 Clarence Thomas U. S. Court of Appeals for the District of Columbia Circ~
```

```
joined_judges %>%
  mutate(name = paste(name_first, name_last, sep = " ")) %>%
  filter(court_name != "Supreme Court of the United States", name == "Thurgood Marshall") %>%
  select(name, court_name)
```

```
## # A tibble: 1 x 2
##   name              court_name
##   <chr>             <chr>
## 1 Thurgood Marshall U. S. Court of Appeals for the Second Circuit
```

Data reshaping (`spread()` and `gather()`)

It can be helpful to think of tabular data as coming in two forms: wide data, and long data. Let's load in a table of data. This data contains total membership figures for the Virginia conference of the Methodist Episcopal Church for the years 1812 to 1830.

```
va_wide <- read_csv("http://dh-r.lincolnmullen.com/data/va-methodists-wide.csv")
va_wide
```

```
## # A tibble: 10 x 21
##    conference district `1812` `1813` `1814` `1815` `1816` `1817` `1818`
##    <chr>      <chr>     <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
##  1 Virginia   James R~   5348   4691   4520   4209   4118   3888   3713
##  2 Virginia   Meherren   4882   4486   4771   4687   4702     NA     NA
##  3 Virginia   Meherrin     NA     NA     NA     NA     NA   4435   3964
##  4 Virginia   Neuse        NA     NA   3474   3475   3448   2702   3340
##  5 Virginia   Newbern    3511   3558     NA     NA     NA     NA     NA
##  6 Virginia   Norfolk    4686   6196   6127   6001   5661   6495   6471
##  7 Virginia   Raleigh    3822   4018     NA     NA     NA     NA     NA
```

```
##  8 Virginia   Roanoke      NA     NA     NA     NA   3049     NA   1507
##  9 Virginia   Tar Riv~     NA     NA   3834   3466     NA     NA     NA
## 10 Virginia   Yadkin     3174   3216   3528   3323   3374   3323   4689
## # ... with 12 more variables: `1819` <dbl>, `1820` <dbl>, `1821` <dbl>,
## #   `1822` <dbl>, `1823` <dbl>, `1824` <dbl>, `1825` <dbl>, `1826` <dbl>,
## #   `1827` <dbl>, `1828` <dbl>, `1829` <dbl>, `1830` <dbl>
```

The first thing we can notice about this data frame is that it is very wide because it has a column for each of the years. The data is also suitable for reading because it like a table in a publication. We can read from left to right and see when certain districts begin and end and get the values for each year. The difficulties of computing on or plotting the data will also become quickly apparent. How would you make a plot of the change over time in the number of members in each district? Or how would you filter by year, or summarize by year? For that matter, what do the numbers in the table represent, since they are not given an explicit variable name?

The problem with the table is that it is not *tidy data*, because the variables are not in columns and observations in rows. One of the variables is the year, but its values are in the column headers. And another of the variables is total membership, but its values are spread across rows and columns and it is not explicitly named.

The `gather()` function from the tidyr package lets us turn wide data into long data. We need to tell the function two kinds of information. First we need to tell it the name of the column to create from the column headers and the name of the implicit variable in the rows. In the example below, we create to new columns `minutes_year` and `total_membership`. Then we also have to tell the function if there are any columns which should remain unchanged. In this case, the `conference` and `district` variables should remain the same, so we remove them from the gathering using the same syntax as the `select()` function.

```
va_wide %>%
  gather(year, members_total, -conference, -district)
```

```
## # A tibble: 190 x 4
##    conference district    year  members_total
##    <chr>      <chr>       <chr>         <dbl>
##  1 Virginia   James River 1812           5348
##  2 Virginia   Meherren    1812           4882
##  3 Virginia   Meherrin    1812             NA
##  4 Virginia   Neuse       1812             NA
##  5 Virginia   Newbern     1812           3511
##  6 Virginia   Norfolk     1812           4686
##  7 Virginia   Raleigh     1812           3822
##  8 Virginia   Roanoke     1812             NA
##  9 Virginia   Tar River   1812             NA
## 10 Virginia   Yadkin      1812           3174
## # ... with 180 more rows
```

We can see the results above. There are two ways that this result is not quite what we want. Because the years were column headers they are treated as character vectors rather than integers. We can manually convert them in a later step, but we can also let `gather()` do the right thing with the `convert =` argument. Then we have a lot of `NA` values which were explicit in the wide table but which can be removed from the long table with `na.rm =`.

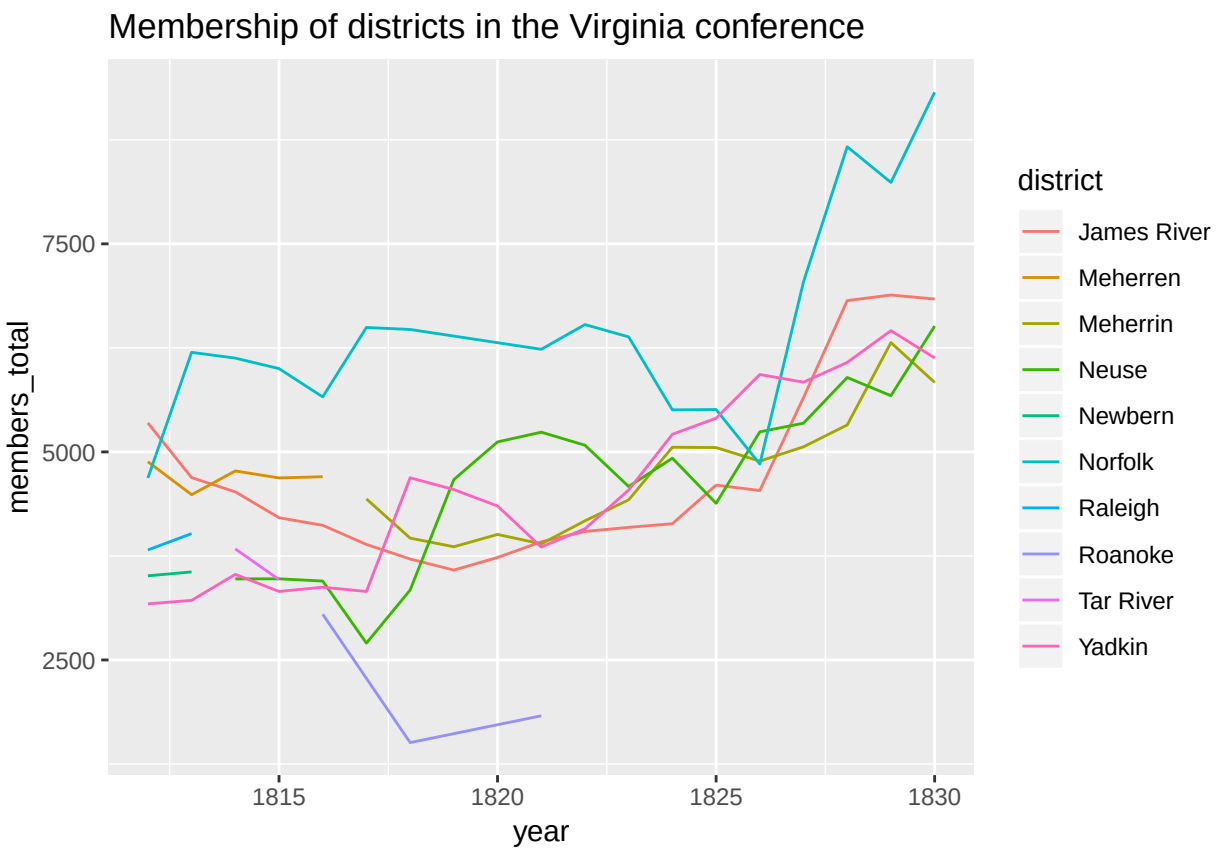
```
va_long <- va_wide %>%
  gather(year, members_total, -conference, -district,
         convert = TRUE, na.rm = TRUE)
```

```
va_long
```

```
## # A tibble: 100 x 4
##    conference district     year members_total
##    <chr>      <chr>       <int>         <dbl>
##  1 Virginia   James River  1812          5348
##  2 Virginia   Meherren     1812          4882
##  3 Virginia   Newbern      1812          3511
##  4 Virginia   Norfolk      1812          4686
##  5 Virginia   Raleigh      1812          3822
##  6 Virginia   Yadkin       1812          3174
##  7 Virginia   James River  1813          4691
##  8 Virginia   Meherren     1813          4486
##  9 Virginia   Newbern      1813          3558
## 10 Virginia   Norfolk      1813          6196
## # ... with 90 more rows
```

Notice that now we can use the data in ggplot2 without any problem.

```
ggplot(va_long,
       aes(x = year, y = members_total, color = district)) +
  geom_line() +
  ggtitle("Membership of districts in the Virginia conference")
```



The inverse operation of `gather()` is `spread()`. With `spread()` we specify the name of the column which should become the new column headers (in this case `minutes_year`), and then the name of the column to fill in underneath those new column headers (in this case, `total_membership`). We can see the results below.

```
va_wide2 <- va_long %>%
  spread(year, members_total)

va_wide2
```

```
## # A tibble: 10 x 21
##    conference district `1812` `1813` `1814` `1815` `1816` `1817` `1818`
##    <chr>      <chr>     <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
##  1 Virginia   James R~   5348   4691   4520   4209   4118   3888   3713
##  2 Virginia   Meherren   4882   4486   4771   4687   4702     NA     NA
##  3 Virginia   Meherrin     NA     NA     NA     NA     NA   4435   3964
##  4 Virginia   Neuse        NA     NA   3474   3475   3448   2702   3340
##  5 Virginia   Newbern    3511   3558     NA     NA     NA     NA     NA
##  6 Virginia   Norfolk    4686   6196   6127   6001   5661   6495   6471
##  7 Virginia   Raleigh    3822   4018     NA     NA     NA     NA     NA
##  8 Virginia   Roanoke      NA     NA     NA     NA   3049     NA   1507
##  9 Virginia   Tar Riv~     NA     NA   3834   3466     NA     NA     NA
## 10 Virginia   Yadkin     3174   3216   3528   3323   3374   3323   4689
## # ... with 12 more variables: `1819` <dbl>, `1820` <dbl>, `1821` <dbl>,
## #   `1822` <dbl>, `1823` <dbl>, `1824` <dbl>, `1825` <dbl>, `1826` <dbl>,
## #   `1827` <dbl>, `1828` <dbl>, `1829` <dbl>, `1830` <dbl>
```

By looking at the data we can see that we got back to where we started.

Turning long data into wide is often useful when you want to create a tabular representation of data. (And once you have a data frame that can be a table, the `knitr::kable()` function is quite nice.) And some algorithms, such as clustering algorithms, expect wide data rather than tidy data.

For the exercise, we will use summary statistics of the number of white and black members in the Methodists by year.

```
methodists_by_year_race <- methodists %>%
  group_by(year) %>%
  summarize(white = sum(members_white, na.rm = TRUE),
            black = sum(members_black, na.rm = TRUE),
            indian = sum(members_indian, na.rm = TRUE))
methodists_by_year_race
```

```
## # A tibble: 49 x 4
##     year white black indian
##    <int> <int> <int>  <int>
##  1  1786 18291  2890      0
##  2  1787 21949  3883      0
##  3  1788 30557  7991      0
##  4  1789 34425  8840      0
##  5  1790 45983 11682      0
##  6  1791 50580 13098      0
##  7  1792 52079 13871      0
##  8  1793 51486 14420      0
##  9  1794 52794 13906      0
```

```
## 10  1795 48121 12171      0
## # ... with 39 more rows
```

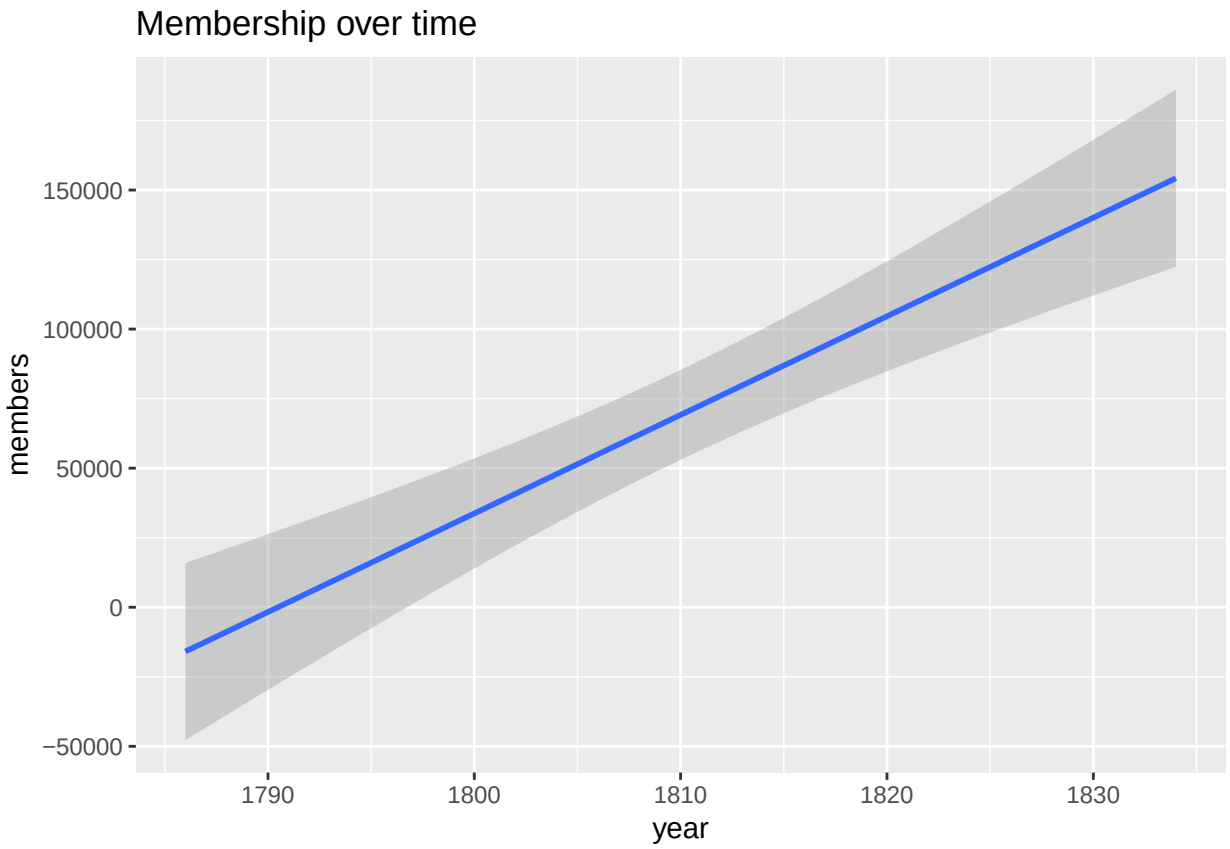
(5) The data in `methodists_by_year_race` could be tidier still. While `white`, `black`, and `indian` are variables, it is perhaps better to think of them as two different variables. One variable would be `race`, containing the racial descriptions that the Methodists used, and another would be `members`, containing the number of members. Using the `gather()` function, create that data frame.

```
methodists_tidy <- methodists_by_year_race %>%
  gather(race, members, -year)
methodists_tidy
```

```
## # A tibble: 147 x 3
##     year race  members
##    <int> <chr>   <int>
##  1  1786 white   18291
##  2  1787 white   21949
##  3  1788 white   30557
##  4  1789 white   34425
##  5  1790 white   45983
##  6  1791 white   50580
##  7  1792 white   52079
##  8  1793 white   51486
##  9  1794 white   52794
## 10  1795 white   48121
## # ... with 137 more rows
```

(6) Use the data frame you created in the previous step to create a line plot of membership over time, mapping the `race` column to the `color` aesthetic.

```
ggplot(methodists_tidy, aes(x = year, y = members), color = "aesthetic") +
  geom_smooth(method = lm) +
  ggtitle("Membership over time")
```



(7) Now use that newly tidied data frame to create a wide data frame, where the years are the column headers and the racial descriptions are the rows.

```
methodists_tidy %>%
  spread(year, members)
```

```
## # A tibble: 3 x 50
##   race  `1786` `1787` `1788` `1789` `1790` `1791` `1792` `1793` `1794`
##   <chr>  <int>  <int>  <int>  <int>  <int>  <int>  <int>  <int>  <int>
## 1 black   2890   3883   7991   8840  11682  13098  13871  14420  13906
## 2 indi~      0      0      0      0      0      0      0      0      0
## 3 white  18291  21949  30557  34425  45983  50580  52079  51486  52794
## # ... with 40 more variables: `1795` <int>, `1796` <int>, `1797` <int>,
## #   `1798` <int>, `1799` <int>, `1800` <int>, `1801` <int>, `1802` <int>,
## #   `1803` <int>, `1804` <int>, `1805` <int>, `1806` <int>, `1807` <int>,
## #   `1808` <int>, `1809` <int>, `1810` <int>, `1811` <int>, `1812` <int>,
## #   `1813` <int>, `1814` <int>, `1815` <int>, `1816` <int>, `1817` <int>,
## #   `1818` <int>, `1819` <int>, `1820` <int>, `1821` <int>, `1822` <int>,
## #   `1823` <int>, `1824` <int>, `1825` <int>, `1826` <int>, `1827` <int>,
## #   `1828` <int>, `1829` <int>, `1830` <int>, `1831` <int>, `1832` <int>,
## #   `1833` <int>, `1834` <int>
```

(8) Now use the same tidied data to create a wide data frame where the racial descriptions are column headers and the years are rows.

```
methodists_tidy %>%
  spread(race, members)
```

```
## # A tibble: 49 x 4
##     year black indian white
##    <int> <int>  <int> <int>
##  1  1786  2890      0 18291
##  2  1787  3883      0 21949
##  3  1788  7991      0 30557
##  4  1789  8840      0 34425
##  5  1790 11682      0 45983
##  6  1791 13098      0 50580
##  7  1792 13871      0 52079
##  8  1793 14420      0 51486
##  9  1794 13906      0 52794
## 10  1795 12171      0 48121
## # ... with 39 more rows
```