Data Wrangling

Jo Hardin

September 11 & 13, 2017

Goals

- Piping / chaining
- Basic data verbs
- Higher level data verbs

Datasets

starwars is from `dplyr`, although originally from SWAPI, the Star Wars API, http://swapi.co/.

NHANES From `?NHANES`: This is survey data collected by the US National Center for Health Statistics (NCHS) which has conducted a series of health and nutrition surveys since the early 1960's. Since 1999 approximately 5,000 individuals of all ages are interviewed in their homes every year and complete the health examination component of the survey. The health examination is conducted in a mobile examination centre (MEC).

babynames Each year, the US Social Security Administration publishes a list of the most popular names given to babies. In 2014, http://www.ssa.gov/oact/babynames/#ht=2 shows Emma and Olivia leading for girls, Noah and Liam for boys.

The `babynames` data table in the `babynames` package comes from the Social Security Administration's listing of the names givens to babies in each year, and the number of babies of each sex given that name. (Only names with 5 or more babies are published by the SSA.)

Examples of Chaining

```
babynames %>% nrow()
```

```
## [1] 1858689
```

```
babynames %>% names()
```

```
## [1] "year" "sex"  "name" "n"    "prop"
```

```
babynames %>% glimpse()
```

```
## Observations: 1,858,689
## Variables: 5
## $ year <dbl> 1880, 1880, 1880, 1880, 1880, 1880, 1880, 1880, 1880, 188...
## $ sex  <chr> "F", "F", "F", "F", "F", "F", "F", "F", "F", "F", "F", "F...
## $ name <chr> "Mary", "Anna", "Emma", "Elizabeth", "Minnie", "Margaret"...
## $ n    <int> 7065, 2604, 2003, 1939, 1746, 1578, 1472, 1414, 1320, 128...
## $ prop <dbl> 0.072384329, 0.026679234, 0.020521700, 0.019865989, 0.017...
```

```
babynames %>% head()
```

```
## # A tibble: 6 x 5
##    year   sex      name     n       prop
##   <dbl> <chr>     <chr> <int>      <dbl>
```

```
## 1  1880     F      Mary  7065 0.07238433
## 2  1880     F      Anna  2604 0.02667923
## 3  1880     F      Emma  2003 0.02052170
## 4  1880     F Elizabeth  1939 0.01986599
## 5  1880     F    Minnie  1746 0.01788861
## 6  1880     F  Margaret  1578 0.01616737
```

```
babynames %>% tail()
```

```
## # A tibble: 6 x 5
##    year   sex   name     n         prop
##   <dbl> <chr>  <chr> <int>        <dbl>
## 1  2015     M   Zyah     5 2.466855e-06
## 2  2015     M Zykell     5 2.466855e-06
## 3  2015     M Zyking     5 2.466855e-06
## 4  2015     M  Zykir     5 2.466855e-06
## 5  2015     M  Zyrus     5 2.466855e-06
## 6  2015     M   Zyus     5 2.466855e-06
```

```
babynames %>% sample_n(size=5)
```

```
## # A tibble: 5 x 5
##    year   sex     name     n         prop
##   <dbl> <chr>    <chr> <int>        <dbl>
## 1  2009     F   Ashlea     8 3.956872e-06
## 2  1999     F   Brania     5 2.569595e-06
## 3  2000     F Sharanya     9 4.512497e-06
## 4  2006     M   Jallen     8 3.652858e-06
## 5  1972     F  Elspeth     5 3.100729e-06
```

```
babynames %>% mosaic::favstats(n ~ sex, data = .)
```

```
##   sex min Q1 median Q3   max     mean       sd       n missing
## 1   F   5  7     11 31 99680 153.3909 1196.259 1100858       0
## 2   M   5  7     12 34 94763 226.9508 1962.471  757831       0
```

Data Verbs

Taken from the dplyr tutorial: http://dplyr.tidyverse.org/

```
library(dplyr)

starwars %>% dim()
```

```
## [1] 87 13
```

```
starwars %>% names()
```

```
##  [1] "name"       "height"     "mass"       "hair_color" "skin_color"
##  [6] "eye_color"  "birth_year" "gender"     "homeworld"  "species"
## [11] "films"      "vehicles"   "starships"
```

```
starwars %>% head()
```

```
## # A tibble: 6 x 13
##             name height  mass  hair_color  skin_color eye_color birth_year
##            <chr>  <int> <dbl>       <chr>       <chr>     <chr>      <dbl>
## 1 Luke Skywalker    172    77       blond        fair      blue       19.0
## 2          C-3PO    167    75        <NA>        gold    yellow      112.0
## 3          R2-D2     96    32        <NA> white, blue       red       33.0
## 4    Darth Vader    202   136        none       white    yellow       41.9
## 5    Leia Organa    150    49       brown       light     brown       19.0
## 6      Owen Lars    178   120 brown, grey       light      blue       52.0
## # ... with 6 more variables: gender <chr>, homeworld <chr>, species <chr>,
## #   films <list>, vehicles <list>, starships <list>
```

```
starwars %>%
  mosaic::favstats(mass~gender, data = .)
```

```
##          gender  min      Q1 median      Q3  max       mean       sd  n
## 1        female   45   49.25   52.5   55.90   75   54.02000  8.37215 10
## 2 hermaphrodite 1358 1358.00 1358.0 1358.00 1358 1358.00000       NA  1
## 3          male   15   76.50   80.0   87.25  159   81.00455 28.22371 44
## 4          none  140  140.00  140.0  140.00  140  140.00000       NA  1
##   missing
## 1       9
## 2       0
## 3      18
## 4       1
```

```
starwars %>%
  dplyr::filter(species == "Droid")
```

```
## # A tibble: 5 x 13
##    name height  mass hair_color  skin_color eye_color birth_year gender
##   <chr>  <int> <dbl>      <chr>       <chr>     <chr>      <dbl>  <chr>
## 1 C-3PO    167    75       <NA>        gold    yellow        112   <NA>
## 2 R2-D2     96    32       <NA> white, blue       red         33   <NA>
## 3 R5-D4     97    32       <NA>  white, red       red         NA   <NA>
## 4 IG-88    200   140       none       metal       red         15   none
## 5   BB8     NA    NA       none        none     black         NA   none
## # ... with 5 more variables: homeworld <chr>, species <chr>, films <list>,
## #   vehicles <list>, starships <list>
```

```
starwars %>%
  dplyr::filter(species != "Droid") %>%
  mosaic::favstats(mass~gender, data = .)
```

```
##          gender  min     Q1 median      Q3  max       mean         sd  n
## 1        female   45   50.0     55   56.20   75   54.68889  8.591921  9
## 2 hermaphrodite 1358 1358.0   1358 1358.00 1358 1358.00000         NA  1
## 3          male   15   76.5     80   87.25  159   81.00455 28.223707 44
##   missing
## 1       7
## 2       0
## 3      16
```

```
starwars %>%
  dplyr::select(name, ends_with("color"))
```

```
## # A tibble: 87 x 4
##                  name    hair_color  skin_color eye_color
##                 <chr>         <chr>       <chr>     <chr>
##  1     Luke Skywalker         blond        fair      blue
##  2              C-3PO          <NA>        gold    yellow
##  3              R2-D2          <NA> white, blue       red
##  4        Darth Vader          none       white    yellow
##  5        Leia Organa         brown       light     brown
##  6          Owen Lars   brown, grey       light      blue
##  7 Beru Whitesun lars         brown       light      blue
##  8              R5-D4          <NA>  white, red       red
##  9  Biggs Darklighter         black       light     brown
## 10     Obi-Wan Kenobi auburn, white        fair blue-gray
## # ... with 77 more rows
```

```
starwars %>%
  dplyr::mutate(name, bmi = mass / ((height / 100)  ^ 2)) %>%
  dplyr::select(name:mass, bmi)
```

```
## # A tibble: 87 x 4
##                  name height  mass      bmi
##                 <chr>  <int> <dbl>    <dbl>
##  1     Luke Skywalker    172    77 26.02758
##  2              C-3PO    167    75 26.89232
##  3              R2-D2     96    32 34.72222
##  4        Darth Vader    202   136 33.33007
##  5        Leia Organa    150    49 21.77778
##  6          Owen Lars    178   120 37.87401
##  7 Beru Whitesun lars    165    75 27.54821
##  8              R5-D4     97    32 34.00999
##  9  Biggs Darklighter    183    84 25.08286
## 10     Obi-Wan Kenobi    182    77 23.24598
## # ... with 77 more rows
```

```
starwars %>%
  dplyr::arrange(desc(mass))
```

```
## # A tibble: 87 x 13
##                    name height  mass  hair_color       skin_color
##                   <chr>  <int> <dbl>       <chr>            <chr>
```

```
##  1 Jabba Desilijic Tiure    175  1358        <NA> green-tan, brown
##  2              Grievous    216   159        none     brown, white
##  3                 IG-88    200   140        none            metal
##  4           Darth Vader    202   136        none            white
##  5               Tarfful    234   136       brown            brown
##  6             Owen Lars    178   120 brown, grey            light
##  7                 Bossk    190   113        none            green
##  8             Chewbacca    228   112       brown          unknown
##  9      Jek Tono Porkins    180   110       brown             fair
## 10       Dexter Jettster    198   102        none            brown
## # ... with 77 more rows, and 8 more variables: eye_color <chr>,
## #   birth_year <dbl>, gender <chr>, homeworld <chr>, species <chr>,
## #   films <list>, vehicles <list>, starships <list>
```

```
starwars %>%
  dplyr::group_by(species) %>%
  dplyr::summarise(
    n = n(),
    mass = mean(mass, na.rm = TRUE)
  ) %>%
  dplyr::filter(n > 1)
```

```
## # A tibble: 9 x 3
##    species     n      mass
##      <chr> <int>     <dbl>
## 1    Droid     5  69.75000
## 2   Gungan     3  74.00000
## 3    Human    35  82.78182
## 4 Kaminoan     2  88.00000
## 5 Mirialan     2  53.10000
## 6  Twi'lek     2  55.00000
## 7  Wookiee     2 124.00000
## 8   Zabrak     2  80.00000
## 9     <NA>     5  48.00000
```

```
require(NHANES)
names(NHANES)
```

```
##  [1] "ID"               "SurveyYr"         "Gender"
##  [4] "Age"              "AgeDecade"        "AgeMonths"
##  [7] "Race1"            "Race3"            "Education"
## [10] "MaritalStatus"    "HHIncome"         "HHIncomeMid"
## [13] "Poverty"          "HomeRooms"        "HomeOwn"
## [16] "Work"             "Weight"           "Length"
## [19] "HeadCirc"         "Height"           "BMI"
## [22] "BMICatUnder20yrs" "BMI_WHO"          "Pulse"
## [25] "BPSysAve"         "BPDiaAve"         "BPSys1"
## [28] "BPDia1"           "BPSys2"           "BPDia2"
## [31] "BPSys3"           "BPDia3"           "Testosterone"
## [34] "DirectChol"       "TotChol"          "UrineVol1"
## [37] "UrineFlow1"       "UrineVol2"        "UrineFlow2"
## [40] "Diabetes"         "DiabetesAge"      "HealthGen"
## [43] "DaysPhysHlthBad"  "DaysMentHlthBad"  "LittleInterest"
## [46] "Depressed"        "nPregnancies"     "nBabies"
## [49] "Age1stBaby"       "SleepHrsNight"    "SleepTrouble"
```

```
## [52] "PhysActive"       "PhysActiveDays"   "TVHrsDay"
## [55] "CompHrsDay"       "TVHrsDayChild"    "CompHrsDayChild"
## [58] "Alcohol12PlusYr"  "AlcoholDay"       "AlcoholYear"
## [61] "SmokeNow"         "Smoke100"         "Smoke100n"
## [64] "SmokeAge"         "Marijuana"        "AgeFirstMarij"
## [67] "RegularMarij"     "AgeRegMarij"      "HardDrugs"
## [70] "SexEver"          "SexAge"           "SexNumPartnLife"
## [73] "SexNumPartYear"   "SameSex"          "SexOrientation"
## [76] "PregnantNow"
```

```
# find the sleep variables
NHANESsleep <- NHANES %>% select(Gender, Age, Weight, Race1, Race3, Education, SleepTrouble,
                                 SleepHrsNight, TVHrsDay, TVHrsDayChild, PhysActive)
names(NHANESsleep)
```

```
##  [1] "Gender"        "Age"           "Weight"        "Race1"
##  [5] "Race3"         "Education"     "SleepTrouble"  "SleepHrsNight"
##  [9] "TVHrsDay"      "TVHrsDayChild" "PhysActive"
```

```
dim(NHANESsleep)
```

```
## [1] 10000    11
```

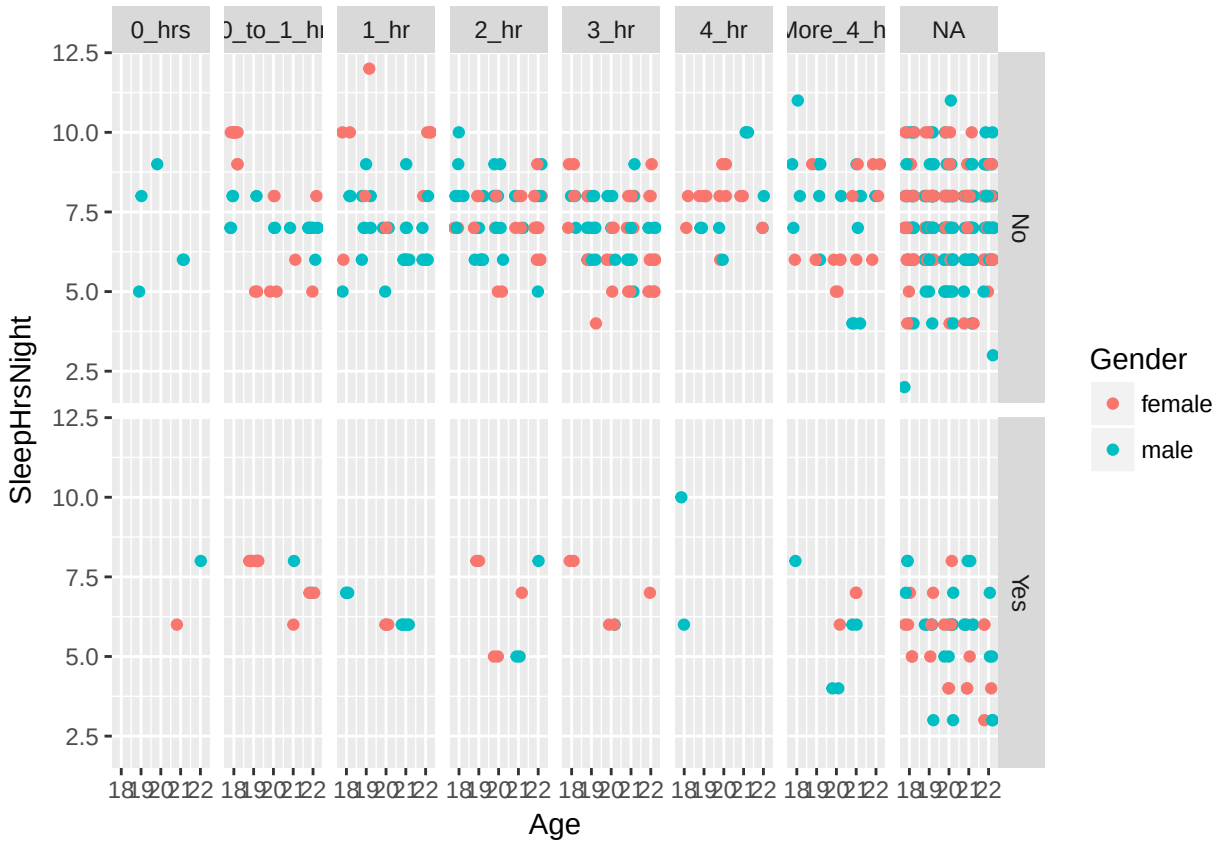
```
# subset for college students
NHANESsleep <- NHANESsleep %>% filter(Age %in% c(18:22)) %>%
  mutate(Weightlb = Weight*2.2)

names(NHANESsleep)
```

```
##  [1] "Gender"        "Age"           "Weight"        "Race1"
##  [5] "Race3"         "Education"     "SleepTrouble"  "SleepHrsNight"
##  [9] "TVHrsDay"      "TVHrsDayChild" "PhysActive"    "Weightlb"
```

```
dim(NHANESsleep)
```

```
## [1] 655  12
```

```
NHANESsleep %>% ggplot(aes(x=Age, y=SleepHrsNight, color=Gender)) +
  geom_point(position=position_jitter(width=.25, height=0) ) +
  facet_grid(SleepTrouble ~ TVHrsDay)
```



summarise and group_by

```
# Using summarise() and group_by()

# number of people (cases) in NHANES
NHANES %>% summarise(n())
```

```
## # A tibble: 1 x 1
##   `n()`
##   <int>
## 1 10000
```

```
# total weight of all the people in NHANES (silly)
NHANES %>% mutate(Weightlb = Weight*2.2) %>% summarise(sum(Weightlb, na.rm=TRUE))
```

```
## # A tibble: 1 x 1
##   `sum(Weightlb, na.rm = TRUE)`
##                           <dbl>
## 1                       1549419
```

```
# mean weight of all the people in NHANES
NHANES %>% mutate(Weightlb = Weight*2.2) %>% summarise(mean(Weightlb, na.rm=TRUE))
```

```
## # A tibble: 1 x 1
##   `mean(Weightlb, na.rm = TRUE)`
##                            <dbl>
## 1                         156.16
```

```
# repeat the above but for groups

# males versus females
NHANES %>% group_by(Gender) %>% summarise(n())
```

```
## # A tibble: 2 x 2
##   Gender `n()`
##   <fctr> <int>
## 1 female  5020
## 2   male  4980
```

```
NHANES %>% group_by(Gender) %>% mutate(Weightlb = Weight*2.2) %>%
  summarise(mean(Weightlb, na.rm=TRUE))
```

```
## # A tibble: 2 x 2
##   Gender `mean(Weightlb, na.rm = TRUE)`
##   <fctr>                          <dbl>
## 1 female                       145.6586
## 2   male                       166.7421
```

```
# smokers and non-smokers
NHANES %>% group_by(SmokeNow) %>% summarise(n())
```

```
## # A tibble: 3 x 2
##   SmokeNow `n()`
##     <fctr> <int>
## 1       No  1745
## 2      Yes  1466
## 3     <NA>  6789
```

```
NHANES %>% group_by(SmokeNow) %>% mutate(Weightlb = Weight*2.2) %>%
  summarise(mean(Weightlb, na.rm=TRUE))
```

```
## # A tibble: 3 x 2
##   SmokeNow `mean(Weightlb, na.rm = TRUE)`
##     <fctr>                          <dbl>
## 1       No                       185.9033
## 2      Yes                       177.1771
## 3     <NA>                       143.9869
```

```
# people with and without diabetes
NHANES %>% group_by(Diabetes) %>% summarise(n())
```

```
## # A tibble: 3 x 2
##   Diabetes `n()`
##     <fctr> <int>
## 1       No  9098
## 2      Yes   760
## 3     <NA>   142
```

```
NHANES %>% group_by(Diabetes) %>% mutate(Weightlb = Weight*2.2) %>%
  summarise(mean(Weightlb, na.rm=TRUE))
```

```
## # A tibble: 3 x 2
##   Diabetes `mean(Weightlb, na.rm = TRUE)`
##     <fctr>                          <dbl>
## 1       No                      154.51245
## 2      Yes                      201.53850
```

```
## 3     <NA>                       21.58324
```

```
# break down the smokers versus non-smokers further, by sex
NHANES %>% group_by(SmokeNow, Gender) %>% summarise(n())
```

```
## # A tibble: 6 x 3
## # Groups:   SmokeNow [?]
##   SmokeNow Gender `n()`
##     <fctr> <fctr> <int>
## 1       No female   764
## 2       No   male   981
## 3      Yes female   638
## 4      Yes   male   828
## 5     <NA> female  3618
## 6     <NA>   male  3171
```

```
NHANES %>% group_by(SmokeNow, Gender) %>% mutate(Weightlb = Weight*2.2) %>%
  summarise(mean(Weightlb, na.rm=TRUE))
```

```
## # A tibble: 6 x 3
## # Groups:   SmokeNow [?]
##   SmokeNow Gender `mean(Weightlb, na.rm = TRUE)`
##     <fctr> <fctr>                          <dbl>
## 1       No female                       166.5945
## 2       No   male                       201.0432
## 3      Yes female                       166.7475
## 4      Yes   male                       185.2018
## 5     <NA> female                       137.5011
## 6     <NA>   male                       151.3725
```

```
# break down the people with diabetes further, by smoking
NHANES %>% group_by(Diabetes, SmokeNow) %>% summarise(n())
```

```
## # A tibble: 8 x 3
## # Groups:   Diabetes [?]
##   Diabetes SmokeNow `n()`
##     <fctr>   <fctr> <int>
## 1       No       No  1476
## 2       No      Yes  1360
## 3       No     <NA>  6262
## 4      Yes       No   267
## 5      Yes      Yes   106
## 6      Yes     <NA>   387
## 7     <NA>       No     2
## 8     <NA>     <NA>   140
```

```
NHANES %>% group_by(Diabetes, SmokeNow) %>% mutate(Weightlb = Weight*2.2) %>%
  summarise(mean(Weightlb, na.rm=TRUE))
```

```
## # A tibble: 8 x 3
## # Groups:   Diabetes [?]
##   Diabetes SmokeNow `mean(Weightlb, na.rm = TRUE)`
##     <fctr>   <fctr>                          <dbl>
## 1       No       No                       182.66279
## 2       No      Yes                       175.13073
## 3       No     <NA>                       143.40996
## 4      Yes       No                       203.78500
```

```
## 5      Yes      Yes                     203.54610
## 6      Yes     <NA>                     199.42305
## 7     <NA>       No                     192.72000
## 8     <NA>     <NA>                      19.13843
```

Back to the babynames

```
babynames %>% group_by(sex) %>%
  summarise(total=sum(n))
```

```
## # A tibble: 2 x 2
##     sex     total
##   <chr>     <int>
## 1     F 168861581
## 2     M 171990331
```

```
babynames %>% group_by(year, sex) %>%
  summarise(name_count = n_distinct(name)) %>% head()
```

```
## # A tibble: 6 x 3
## # Groups:   year [3]
##    year   sex name_count
##   <dbl> <chr>      <int>
## 1  1880     F        942
## 2  1880     M       1058
## 3  1881     F        938
## 4  1881     M        997
## 5  1882     F       1028
## 6  1882     M       1099
```

```
babynames %>% group_by(year, sex) %>%
  summarise(name_count = n_distinct(name)) %>% tail()
```

```
## # A tibble: 6 x 3
## # Groups:   year [3]
##    year   sex name_count
##   <dbl> <chr>      <int>
## 1  2013     F      19203
## 2  2013     M      14026
## 3  2014     F      19150
## 4  2014     M      14026
## 5  2015     F      18993
## 6  2015     M      13959
```

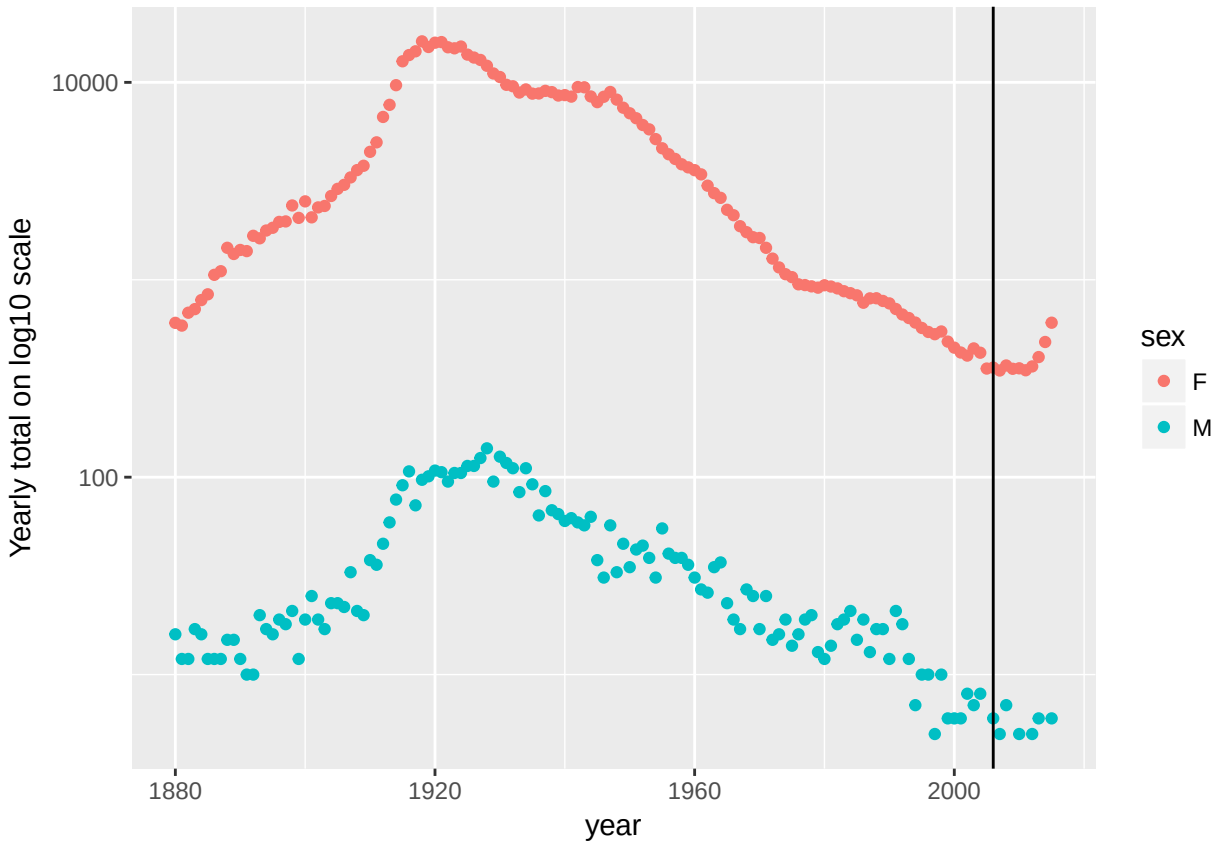
```
babysamp <- babynames %>% sample_n(size=50)
babysamp %>% select(year) %>% distinct() %>% table()
```

```
## .
## 1881 1900 1901 1912 1913 1914 1923 1928 1936 1937 1944 1945 1947 1951 1956
##    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
## 1964 1966 1970 1975 1976 1977 1979 1980 1988 1990 1991 1992 1993 1994 1995
##    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
## 1997 1999 2003 2004 2005 2006 2008 2010 2011 2013 2015
##    1    1    1    1    1    1    1    1    1    1    1
```

```
babysamp %>% distinct() %>% select(year) %>% table()
```

```
## .
## 1881 1900 1901 1912 1913 1914 1923 1928 1936 1937 1944 1945 1947 1951 1956
##    1    1    1    1    1    1    1    1    1    1    2    1    1    1    1
## 1964 1966 1970 1975 1976 1977 1979 1980 1988 1990 1991 1992 1993 1994 1995
##    1    1    1    1    1    1    3    1    1    1    1    1    2    1    1
## 1997 1999 2003 2004 2005 2006 2008 2010 2011 2013 2015
##    1    2    1    2    1    3    1    1    1    2    1
```

```
Frances <- babynames %>%
  filter(name== "Frances") %>%
  group_by(year, sex) %>%
  summarise(yrTot = sum(n))

Frances %>% ggplot(aes(x=year, y=yrTot)) +
  geom_point(aes(color=sex)) +
  geom_vline(xintercept=2006) + scale_y_log10() +
  ylab("Yearly total on log10 scale")
```



Next level of verbs (gather, spread, join, etc.)

gather

gather the military paygrade? https://docs.google.com/spreadsheets/d/1Ow6Cm4z-Z1Yybk3i352msulYCEDOUaOghmo9ALajyHo/edit#gid=1811988794

```
navy.url = "https://docs.google.com/spreadsheets/d/1Ow6Cm4z-Z1Yybk3i352msulYCEDOUaOghmo9ALajyHo/edit#gi
navy.temp = gs_url(navy.url, visibility="public")
navy = gs_read(navy.temp)
glimpse(navy)
```

```
## Observations: 37
## Variables: 17
## $ X1                    <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ `Active Duty Family` <chr> NA, "Marital Status Report", NA, "Data Re...
## $ X3                    <chr> NA, NA, NA, NA, NA, NA, "Single Without C...
## $ X4                    <chr> NA, NA, NA, NA, NA, NA, NA, "Female", "2,...
## $ X5                    <chr> NA, NA, NA, NA, NA, NA, NA, "Total", "10,...
## $ X6                    <chr> NA, NA, NA, NA, NA, NA, "Single With Chil...
## $ X7                    <chr> NA, NA, NA, NA, NA, NA, NA, "Female", "34...
## $ X8                    <chr> NA, NA, NA, NA, NA, NA, NA, "Total", "151...
## $ X9                    <chr> NA, NA, NA, NA, NA, NA, "Joint Service Ma...
## $ X10                   <chr> NA, NA, NA, NA, NA, NA, NA, "Female", "57...
## $ X11                   <chr> NA, NA, NA, NA, NA, NA, NA, "Total", "87"...
## $ X12                   <chr> NA, NA, NA, NA, NA, NA, "Civilian Marriag...
## $ X13                   <chr> NA, NA, NA, NA, NA, NA, NA, "Female", "16...
## $ X14                   <chr> NA, NA, NA, NA, NA, NA, NA, "Total", "968...
## $ X15                   <chr> NA, NA, NA, NA, NA, NA, "Total", "Male", ...
## $ X16                   <chr> NA, NA, NA, NA, NA, NA, NA, "Female", "2,...
## $ X17                   <chr> NA, NA, NA, NA, NA, NA, NA, "Total", "11,...
```

```
names(navy) = c("X","pay.grade", "male.sing.wo", "female.sing.wo", "tot.sing.wo",
                "male.sing.w", "female.sing.w", "tot.sing.w",
                "male.joint.NA", "female.joint.NA", "tot.joint.NA", "male.civ.NA",
                "female.civ.NA", "tot.civ.NA",
                "male.tot.NA", "female.tot.NA", "tot.tot.NA")
navy = navy[-c(1:8), -1]
glimpse(navy)
```

```
## Observations: 29
## Variables: 16
## $ pay.grade       <chr> "E-1", "E-2", "E-3", "E-4", "E-5", "E-6", "E-7...
## $ male.sing.wo    <chr> "7,820", "11,198", "28,163", "23,285", "18,856...
## $ female.sing.wo  <chr> "2,275", "2,718", "6,396", "4,266", "3,649", "...
## $ tot.sing.wo     <chr> "10,095", "13,916", "34,559", "27,551", "22,50...
## $ male.sing.w     <chr> "117", "237", "681", "967", "2,837", "2,725", ...
## $ female.sing.w   <chr> "34", "87", "788", "949", "1,787", "1,284", "3...
## $ tot.sing.w      <chr> "151", "324", "1,469", "1,916", "4,624", "4,00...
## $ male.joint.NA   <chr> "30", "113", "861", "1,526", "2,191", "1,654",...
## $ female.joint.NA <chr> "57", "164", "1,288", "1,691", "2,099", "1,461...
## $ tot.joint.NA    <chr> "87", "277", "2,149", "3,217", "4,290", "3,115...
## $ male.civ.NA     <chr> "806", "2,474", "11,297", "16,277", "31,491", ...
## $ female.civ.NA   <chr> "162", "388", "1,791", "1,768", "2,272", "1,56...
## $ tot.civ.NA      <chr> "968", "2,862", "13,088", "18,045", "33,763", ...
## $ male.tot.NA     <chr> "8,773", "14,022", "41,002", "42,055", "55,375...
```

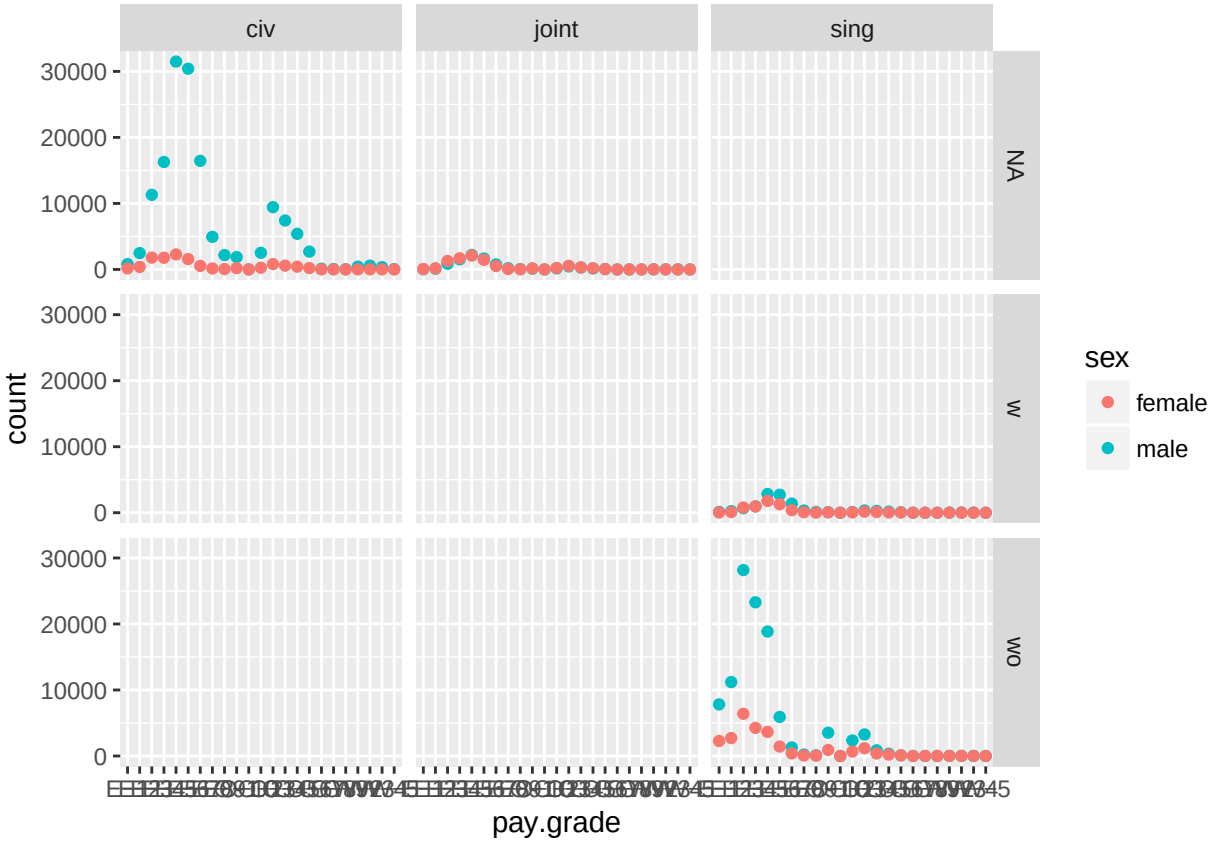
```
## $ female.tot.NA   <chr> "2,528", "3,357", "10,263", "8,674", "9,807", ...
## $ tot.tot.NA      <chr> "11,301", "17,379", "51,265", "50,729", "65,18...
```

```
# get rid of total columns & rows:

navyWR = navy %>% dplyr::select(-contains("tot")) %>%
  filter(substr(pay.grade, 1, 5) != "TOTAL" & substr(pay.grade, 1, 5) != "GRAND" ) %>%
  tidyr::gather(status,numPeople,-pay.grade) %>%
  tidyr::separate(status, into=c("sex", "marital", "kids", sep=".")) %>%
  dplyr::select(c(1:4,6)) %>% mutate(count=readr::parse_number(numPeople))

navyWR %>% head()
```

```
## # A tibble: 6 x 6
##   pay.grade   sex marital  kids numPeople count
##       <chr> <chr>   <chr> <chr>     <chr> <dbl>
## 1       E-1  male    sing    wo     7,820  7820
## 2       E-2  male    sing    wo    11,198 11198
## 3       E-3  male    sing    wo    28,163 28163
## 4       E-4  male    sing    wo    23,285 23285
## 5       E-5  male    sing    wo    18,856 18856
## 6       E-6  male    sing    wo     5,917  5917
```

Does a graph tell us if we did it right? what if we had done it wrong...?

```
navyWR %>% ggplot(aes(x=pay.grade, y=count, color=sex)) +
  geom_point()  + facet_grid(kids~marital)
```



spread

```
babynames %>% spread(sex, n) %>% head()
```

```
## # A tibble: 6 x 5
##    year      name       prop     F     M
##   <dbl>     <chr>      <dbl> <int> <int>
## 1  1880      Mary 0.07238433  7065    NA
## 2  1880      Anna 0.02667923  2604    NA
## 3  1880      Emma 0.02052170  2003    NA
## 4  1880 Elizabeth 0.01986599  1939    NA
## 5  1880    Minnie 0.01788861  1746    NA
## 6  1880  Margaret 0.01616737  1578    NA
```

```
babynames %>%
  dplyr::select(year, name, sex, n) %>%
  tidyr::spread(sex, n) %>%
  dplyr::filter(!is.na(F) & !is.na(M)) %>%
  arrange(desc(year))
```

```
## # A tibble: 163,187 x 4
##     year    name     F     M
##    <dbl>   <chr> <int> <int>
##  1  2015 Aalijah    12    17
##  2  2015 Aaliyah  4836     5
##  3  2015  Aamari     9     7
##  4  2015  Aarian     8    13
##  5  2015  Aarion     7    15
##  6  2015   Aaron    18  7113
##  7  2015   Aarya   180    20
##  8  2015   Aaryn    46    29
##  9  2015  Abbott     6    45
## 10  2015   Abeer    10     9
## # ... with 163,177 more rows
```

```
babynames %>%
  tidyr::spread(sex, n) %>%
  dplyr::filter(!is.na(F) & !is.na(M))
```

```
## # A tibble: 0 x 5
## # ... with 5 variables: year <dbl>, name <chr>, prop <dbl>, F <int>,
## #   M <int>
```

join (use join to merge two datasets)

First get the data

Both of the following datasets come from GapMinder. The first represents country, year, and female literacy rate. The second represents country, year, and GDP (in fixed 2000 US$).

```
litF_url = "https://docs.google.com/spreadsheets/d/1hDinTIRHQIaZg1RUn6Z_6mo12PtKwEPFIz_mJVF6P5I/pub?gid=
#pulling in the URL & keeping track of how big it is
litFurl = gs_url(litF_url, visibility="public")
litF_nrow = litFurl$ws$row_extent[1]
litF_ncol = litFurl$ws$col_extent[1]
#reading in the dataset
```

```r
litF = gs_read(litFurl, range=cell_limits(c(1,1), c(litF_nrow,litF_ncol)))

litF = litF %>% dplyr::select(country=starts_with("Adult"), starts_with("1"), starts_with("2")) %>%
  tidyr::gather(year, litRateF, -country) %>%
  dplyr::mutate( year = readr::parse_number(year)) %>%
  dplyr::filter(!is.na(litRateF))

gdp_url = "https://docs.google.com/spreadsheets/d/1RctTQmKB0hzbm1E8rGcufYdMshRdhmYdeL29nXqmvsc/pub?gid=0
gdpurl = gs_url(gdp_url, visibility="public")
gdp_nrow = gdpurl$ws$row_extent[1]
gdp_ncol = gdpurl$ws$col_extent[1]
gdp = gs_read(gdpurl, range=cell_limits(c(1,1), c(gdp_nrow, gdp_ncol)))
gdp = gdp %>% dplyr::select(country = starts_with("Income"), starts_with("1"), starts_with("2")) %>%
  tidyr::gather(year, gdp, - country) %>%
  dplyr::mutate(year=readr::parse_number(year)) %>%
  dplyr::filter(!is.na(gdp))
```

```r
head(litF)
```

```
## # A tibble: 6 x 3
##                country  year  litRateF
##                  <chr> <dbl>     <dbl>
## 1         Burkina Faso  1975  3.182766
## 2 Central African Rep.  1975  8.399576
## 3               Kuwait  1975 48.015214
## 4               Turkey  1975 45.098921
## 5 United Arab Emirates  1975 38.124870
## 6              Uruguay  1975 94.304522
```

```r
head(gdp)
```

```
## # A tibble: 6 x 3
##      country  year        gdp
##        <chr> <dbl>      <dbl>
## 1    Algeria  1960  1280.3848
## 2  Argentina  1960  5251.8768
## 3  Australia  1960  9407.6851
## 4    Austria  1960  7434.1837
## 5    Bahamas  1960 11926.4610
## 6 Bangladesh  1960   254.8251
```

```r
# left
litGDPleft = left_join(litF, gdp, by=c("country", "year"))
dim(litGDPleft)
```

```
## [1] 571   4
```

```r
sum(is.na(litGDPleft$gdp))
```

```
## [1] 66
```

```r
# inner
litGDPinner = inner_join(litF, gdp, by=c("country", "year"))
dim(litGDPinner)
```

```
## [1] 505   4
```

```
sum(is.na(litGDPinner$gdp))
```

```
## [1] 0
```

```
# full
litGDPfull = full_join(litF, gdp, by=c("country", "year"))
dim(litGDPfull)
```

```
## [1] 8054    4
```

```
sum(is.na(litGDPfull$gdp))
```

```
## [1] 66
```

lubridate

Fun with dates!

```
require(lubridate)
rightnow <- now()

day(rightnow)
```

```
## [1] 13
```

```
week(rightnow)
```

```
## [1] 37
```

```
month(rightnow, label=FALSE)
```

```
## [1] 9
```

```
month(rightnow, label=TRUE)
```

```
## [1] Sep
## 12 Levels: Jan < Feb < Mar < Apr < May < Jun < Jul < Aug < Sep < ... < Dec
```

```
year(rightnow)
```

```
## [1] 2017
```

```
minute(rightnow)
```

```
## [1] 8
```

```
hour(rightnow)
```

```
## [1] 10
```

```
yday(rightnow)
```

```
## [1] 256
```

```
mday(rightnow)
```

```
## [1] 13
```

```
wday(rightnow, label=FALSE)
```

```
## [1] 4
```

```
wday(rightnow, label=TRUE)
```

```
## [1] Wed
## Levels: Sun < Mon < Tues < Wed < Thurs < Fri < Sat
```

But how do I create a date object?

```
jan31 <- ymd("2013-01-31")
jan31 + months(0:11)
```

```
##  [1] "2013-01-31" NA           "2013-03-31" NA           "2013-05-31"
##  [6] NA           "2013-07-31" "2013-08-31" NA           "2013-10-31"
## [11] NA           "2013-12-31"
```

```
floor_date(jan31, "month")
```

```
## [1] "2013-01-01"
```

```
floor_date(jan31, "month") + months(0:11) + days(31)
```

```
##  [1] "2013-02-01" "2013-03-04" "2013-04-01" "2013-05-02" "2013-06-01"
##  [6] "2013-07-02" "2013-08-01" "2013-09-01" "2013-10-02" "2013-11-01"
## [11] "2013-12-02" "2014-01-01"
```

```
jan31 + months(0:11) + days(31)
```

```
##  [1] "2013-03-03" NA           "2013-05-01" NA           "2013-07-01"
##  [6] NA           "2013-08-31" "2013-10-01" NA           "2013-12-01"
## [11] NA           "2014-01-31"
```

```
jan31 %m+% months(0:11)
```

```
##  [1] "2013-01-31" "2013-02-28" "2013-03-31" "2013-04-30" "2013-05-31"
##  [6] "2013-06-30" "2013-07-31" "2013-08-31" "2013-09-30" "2013-10-31"
## [11] "2013-11-30" "2013-12-31"
```

Using the flight data...

```
names(flights)
```

```
##  [1] "year"           "month"          "day"            "dep_time"
##  [5] "sched_dep_time" "dep_delay"      "arr_time"       "sched_arr_time"
##  [9] "arr_delay"      "carrier"        "flight"         "tailnum"
## [13] "origin"         "dest"           "air_time"       "distance"
## [17] "hour"           "minute"         "time_hour"
```

```
flightsWK <- flights %>% mutate(ymdday = ymd(paste(year,"-", month,"-",day))) %>%
  mutate(weekdy = wday(ymdday, label=TRUE), whichweek = week(ymdday))
flightsWK %>% select(year, month, day, ymdday, weekdy, whichweek, dep_time,
                     arr_time, air_time) %>%  head()
```

```
## # A tibble: 6 x 9
##    year month   day     ymdday weekdy whichweek dep_time arr_time air_time
##   <int> <int> <int>     <date>  <ord>     <dbl>    <int>    <int>    <dbl>
## 1  2013     1     1 2013-01-01   Tues         1      517      830      227
## 2  2013     1     1 2013-01-01   Tues         1      533      850      227
## 3  2013     1     1 2013-01-01   Tues         1      542      923      160
## 4  2013     1     1 2013-01-01   Tues         1      544     1004      183
## 5  2013     1     1 2013-01-01   Tues         1      554      812      116
## 6  2013     1     1 2013-01-01   Tues         1      554      740      150
```