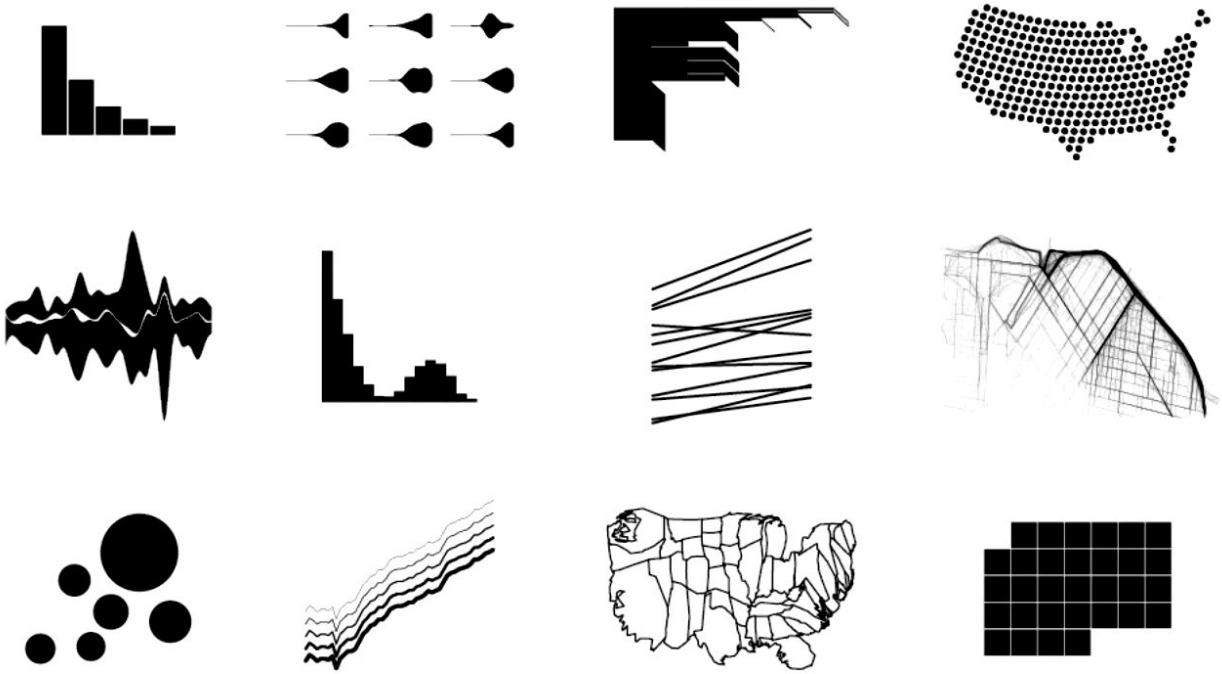
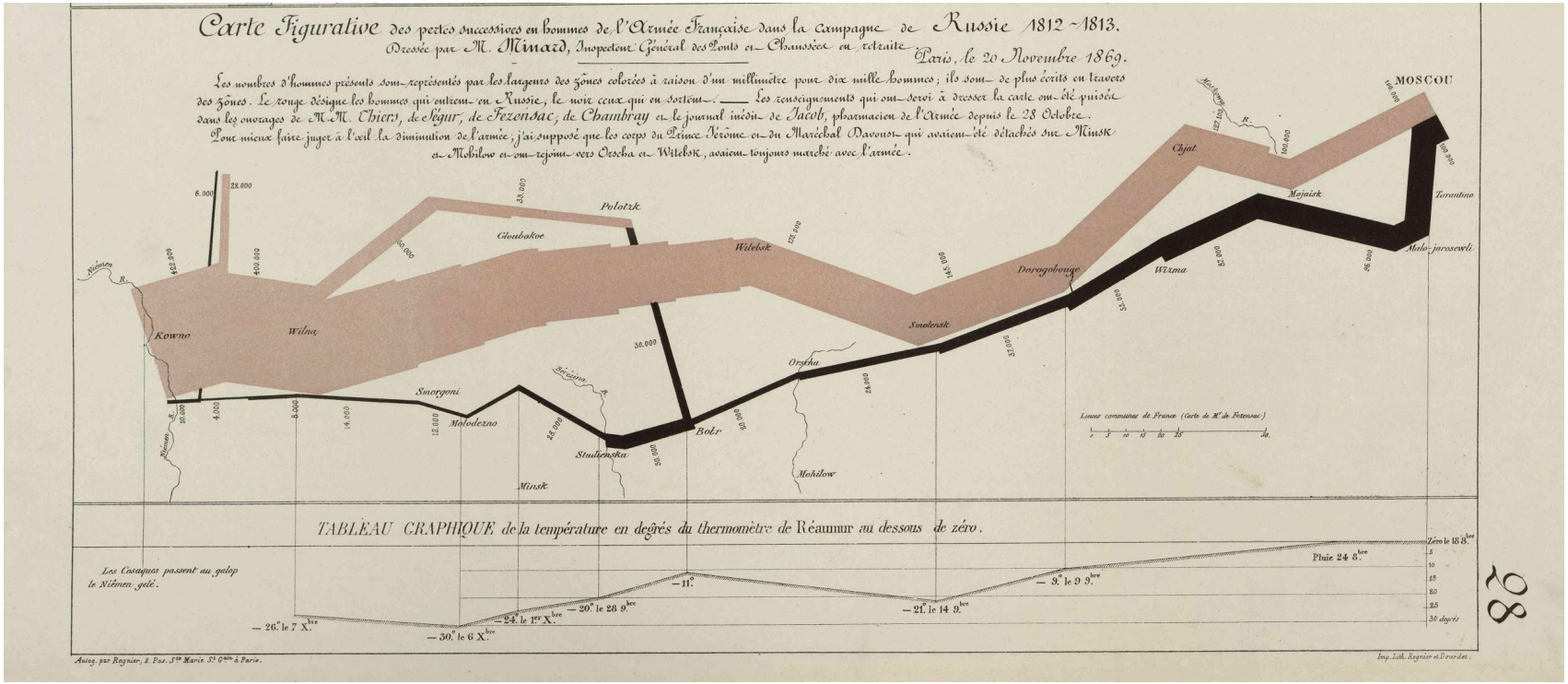
Data Visualization in R.

What is data visualization?

What is Data Visualization?

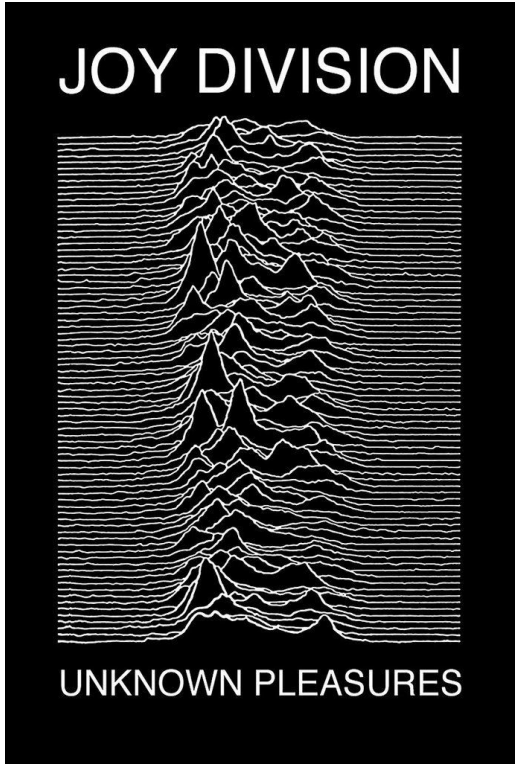


What is Data Visualization?



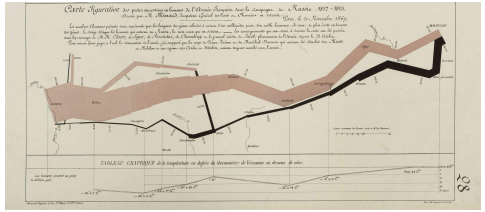
Charles Joseph Minard's *Carte Figurative* illustrates facts related to French Invasion of Russia 1812. (1869)

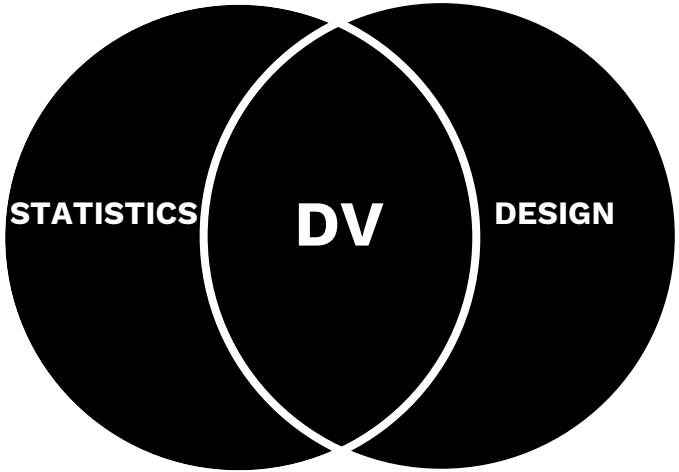
What is Data Visualization?

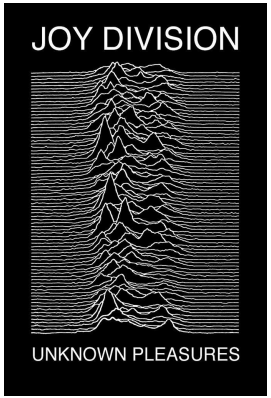


- Artwork of the album *Unknown Pleasures* (Joy Division, 1979)
- *From "Radio Observations of the Pulse Profiles and Dispersion Measures of Twelve Pulsars," by Harold D. Craft, Jr. (September 1970)*
- https://www.youtube.com/watch?v=reEQye0EOAw

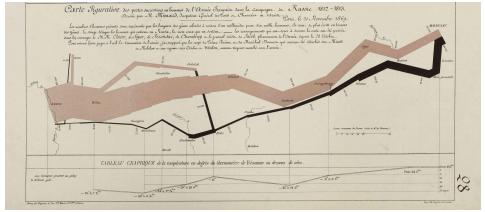
What is Data Visualization?

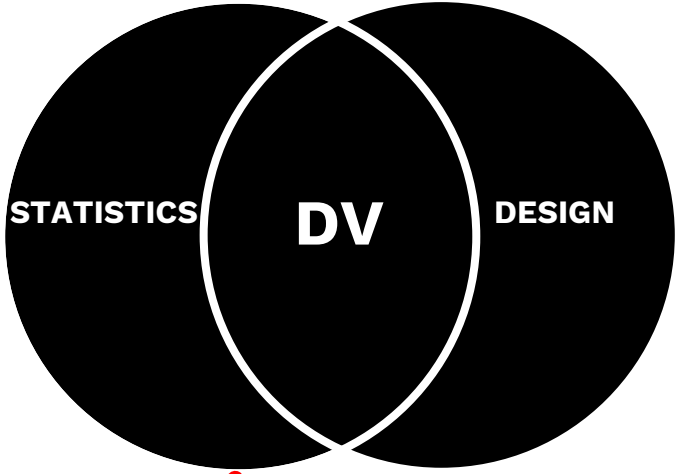




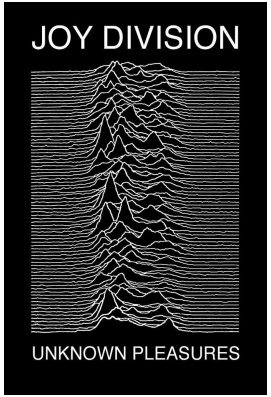


What is Data Visualization?

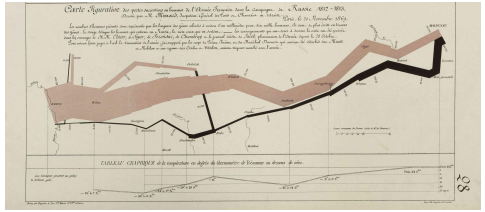


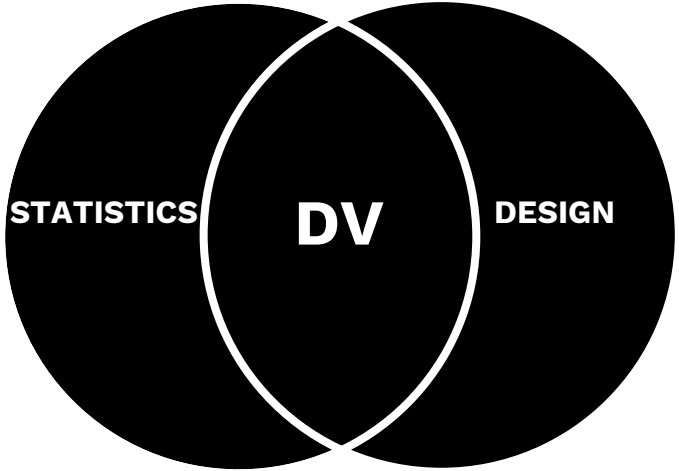


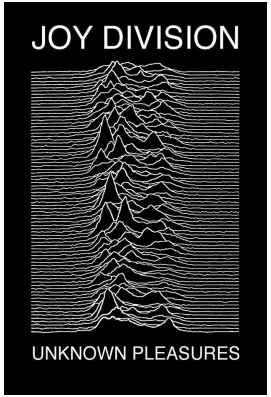
EXPLORATORY PLOTS



What is Data Visualization?



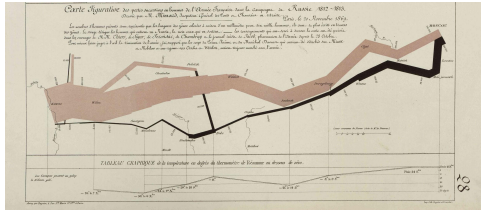


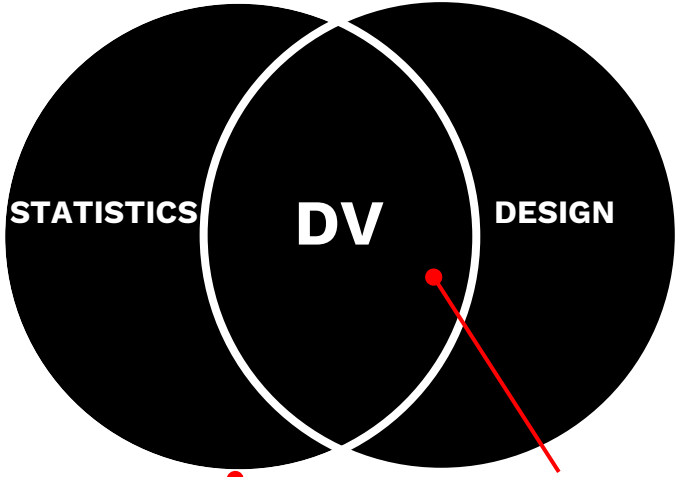


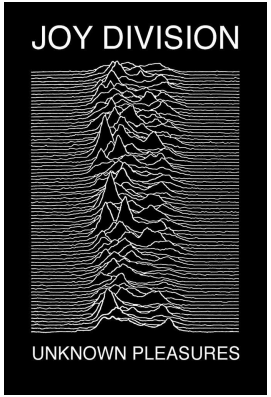
EXPLORATORY PLOTS

- **EASILY-GENERATED**
- **DATA-HEAVY**
- **INTENDED FOR SPECIALIST AUDIENCE**

What is Data Visualization?





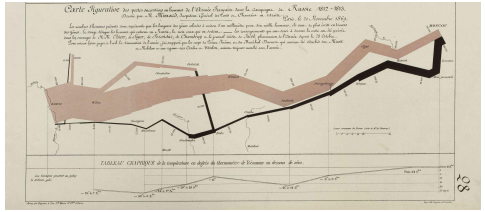


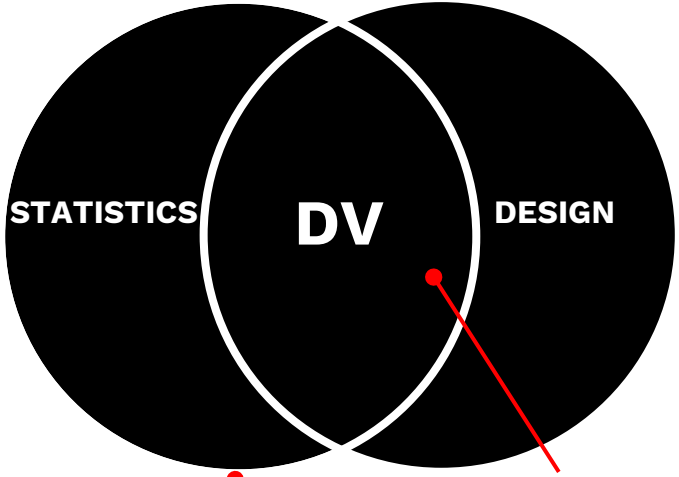
EXPLORATORY PLOTS

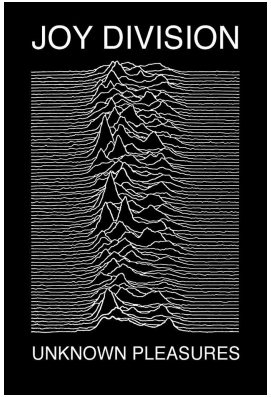
- **EASILY-GENERATED**
- **DATA-HEAVY**
- **INTENDED FOR SPECIALIST AUDIENCE**

EXPLANATORY PLOTS

What is Data Visualization?







EXPLORATORY PLOTS

- EASILY-GENERATED
- DATA-HEAVY
- INTENDED FOR SPECIALIST AUDIENCE

EXPLANATORY PLOTS

- LABOR-INTENSE
- DATA-SPECIFIC
- INTENDED FOR A BROAD AUDIENCE

Example

Example

```r
# read data
library(data.table)
dt <- fread('data/weather_madrid_2017.csv')
```

Example

```
> dplyr::as_tibble(dt)
# A tibble: 6,202 x 15
   date        hour icon  summary temperature apparentTempera~ cloudCover dewPoint humidity precipIntensity precipProbabili~
   <date>     <int> <fct> <fct>         <dbl>            <dbl>      <dbl>    <dbl>    <dbl>           <dbl>            <dbl>
 1 2017-07-22     7 part~ Partly~        17.3             17.3      0.310     5.60    0.460               0                0
 2 2017-07-22     8 part~ Partly~        18.5             18.5      0.310     5.92    0.440               0                0
 3 2017-07-22     9 part~ Partly~        21.8             21.8      0.310     6.57    0.370               0                0
 4 2017-07-22    10 part~ Partly~        23.8             23.8      0.310     6.71    0.330               0                0
 5 2017-07-22    11 clea~ Clear          25.5             25.5      0.190     6.31    0.290               0                0
 6 2017-07-22    12 clea~ Clear          27.1             27.1      0         6.16    0.260               0                0
 7 2017-07-22    13 clea~ Clear          28.6             28.6      0         5.08    0.220               0                0
 8 2017-07-22    14 part~ Partly~        29.1             29.1      0.270     4.26    0.210               0                0
 9 2017-07-22    15 clea~ Clear          30.5             30.5      0         3.95    0.190               0                0
10 2017-07-22    16 part~ Partly~        31.2             31.2      0.310     3.27    0.170               0                0
# ... with 6,192 more rows, and 4 more variables: pressure <dbl>, visibility <dbl>, windBearing <int>, month <fct>
```

Example

```
> dplyr::as_tibble(dt)
# A tibble: 6,202 x 15
   date        hour icon  summary temperature apparentTempera~ cloudCover dewPoint humidity precipIntensity precipProbabili~
   <date>     <int> <fct> <fct>         <dbl>            <dbl>      <dbl>    <dbl>    <dbl>           <dbl>            <dbl>
 1 2017-07-22     7 part~ Partly~        17.3             17.3      0.310     5.60    0.460               0                0
 2 2017-07-22     8 part~ Partly~        18.5             18.5      0.310     5.92    0.440               0                0
 3 2017-07-22     9 part~ Partly~        21.8             21.8      0.310     6.57    0.370               0                0
 4 2017-07-22    10 part~ Partly~        23.8             23.8      0.310     6.71    0.330               0                0
 5 2017-07-22    11 clea~ Clear          25.5             25.5      0.190     6.31    0.290               0                0
 6 2017-07-22    12 clea~ Clear          27.1             27.1      0         6.16    0.260               0                0
 7 2017-07-22    13 clea~ Clear          28.6             28.6      0         5.08    0.220               0                0
 8 2017-07-22    14 part~ Partly~        29.1             29.1      0.270     4.26    0.210               0                0
 9 2017-07-22    15 clea~ Clear          30.5             30.5      0         3.95    0.190               0                0
10 2017-07-22    16 part~ Partly~        31.2             31.2      0.310     3.27    0.170               0                0
# ... with 6,192 more rows, and 4 more variables: pressure <dbl>, visibility <dbl>, windBearing <int>, month <fct>
```

```
> str(dt)
Classes 'data.table' and 'data.frame':  6202 obs. of  15 variables:
 $ date               : Date, format: "2017-07-22" "2017-07-22" "2017
 $ hour               : int  7 8 9 10 11 12 13 14 15 16 ...
 $ icon               : Factor w/ 8 levels "clear-day","clear-night",
 $ summary            : Factor w/ 15 levels "Breezy","Breezy and Most
 $ temperature        : num  17.3 18.5 21.8 23.8 25.5 ...
 $ apparentTemperature: num  17.3 18.5 21.8 23.8 25.5 ...
 $ cloudCover         : num  0.31 0.31 0.31 0.31 0.19 0 0 0.27 0 0.31
 $ dewPoint           : num  5.6 5.92 6.57 6.71 6.31 6.16 5.08 4.26 3
 $ humidity           : num  0.46 0.44 0.37 0.33 0.29 0.26 0.22 0.21
 $ precipIntensity    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ precipProbability  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ pressure           : num  1013 1013 1013 1013 1013 ...
 $ visibility         : num  14.2 15.7 14.2 14.2 15.6 ...
 $ windBearing        : int  311 25 224 177 219 199 213 230 228 224 .
 $ month              : Factor w/ 12 levels "December","November",..:
 - attr(*, ".internal.selfref")=<externalptr>
```

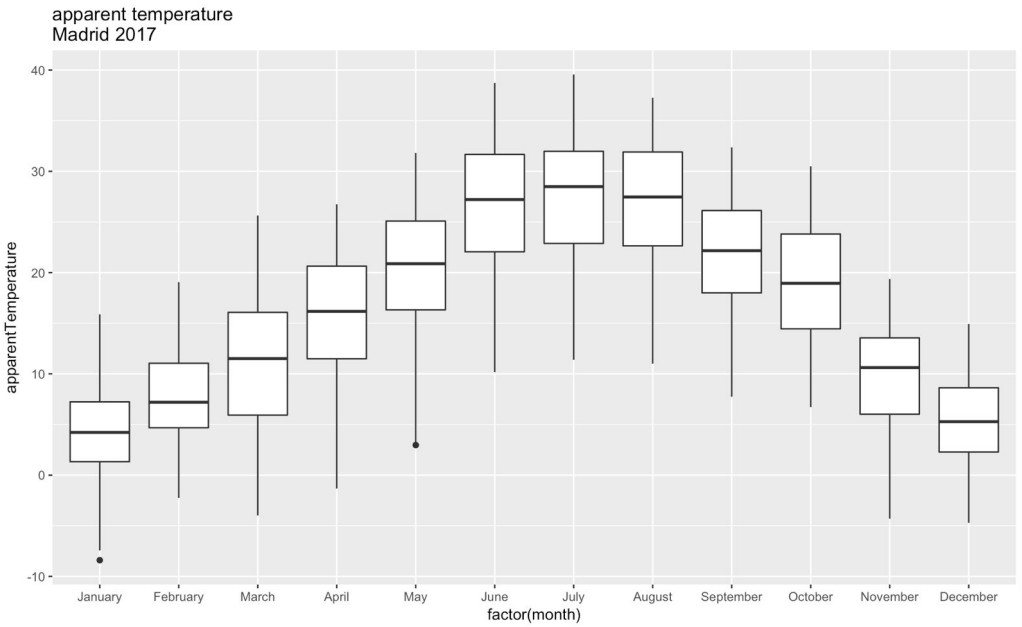
Example

```r
# quick descriptive analysis
dt[,as.list(summary(apparentTemperature)), by=month][order(-month)]
```

	month	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
1:	January	-8.40	1.3300	4.220	4.366279	7.2400	15.88
2:	February	-2.25	4.6750	7.200	7.898592	11.0500	19.06
3:	March	-3.98	5.9250	11.510	11.129639	16.0750	25.64
4:	April	-1.32	11.4925	16.175	15.738804	20.6400	26.75
5:	May	2.97	16.3200	20.880	20.509658	25.0900	31.82
6:	June	10.17	22.0550	27.215	26.899157	31.6700	38.73
7:	July	11.39	22.8850	28.490	27.555522	31.9750	39.56
8:	August	11.01	22.6400	27.470	27.136224	31.9100	37.27
9:	September	7.74	18.0000	22.165	21.845569	26.1300	32.37
10:	October	6.72	14.4450	18.940	18.976964	23.8100	30.50
11:	November	-4.30	6.0175	10.620	9.838784	13.5525	19.37
12:	December	-4.71	2.2850	5.280	5.455863	8.6250	14.93

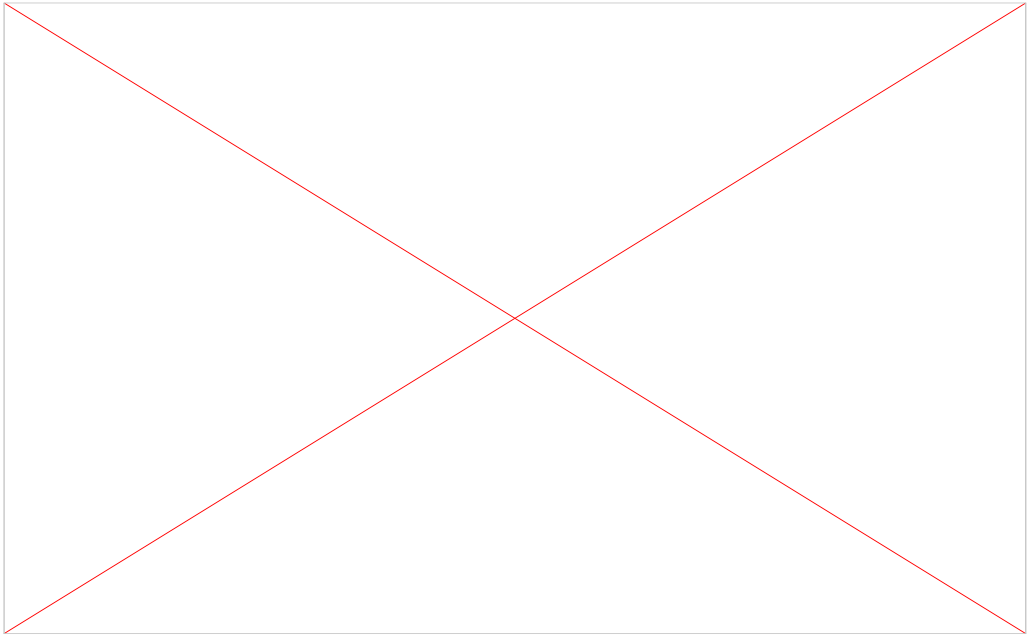
Example

```
# exploratory plot (boxplots)
ggplot(dt,aes(y = apparentTemperature, x = factor(month))) +
  geom_boxplot() + labs(title = 'apparent temperature \nMadrid 2017')
```

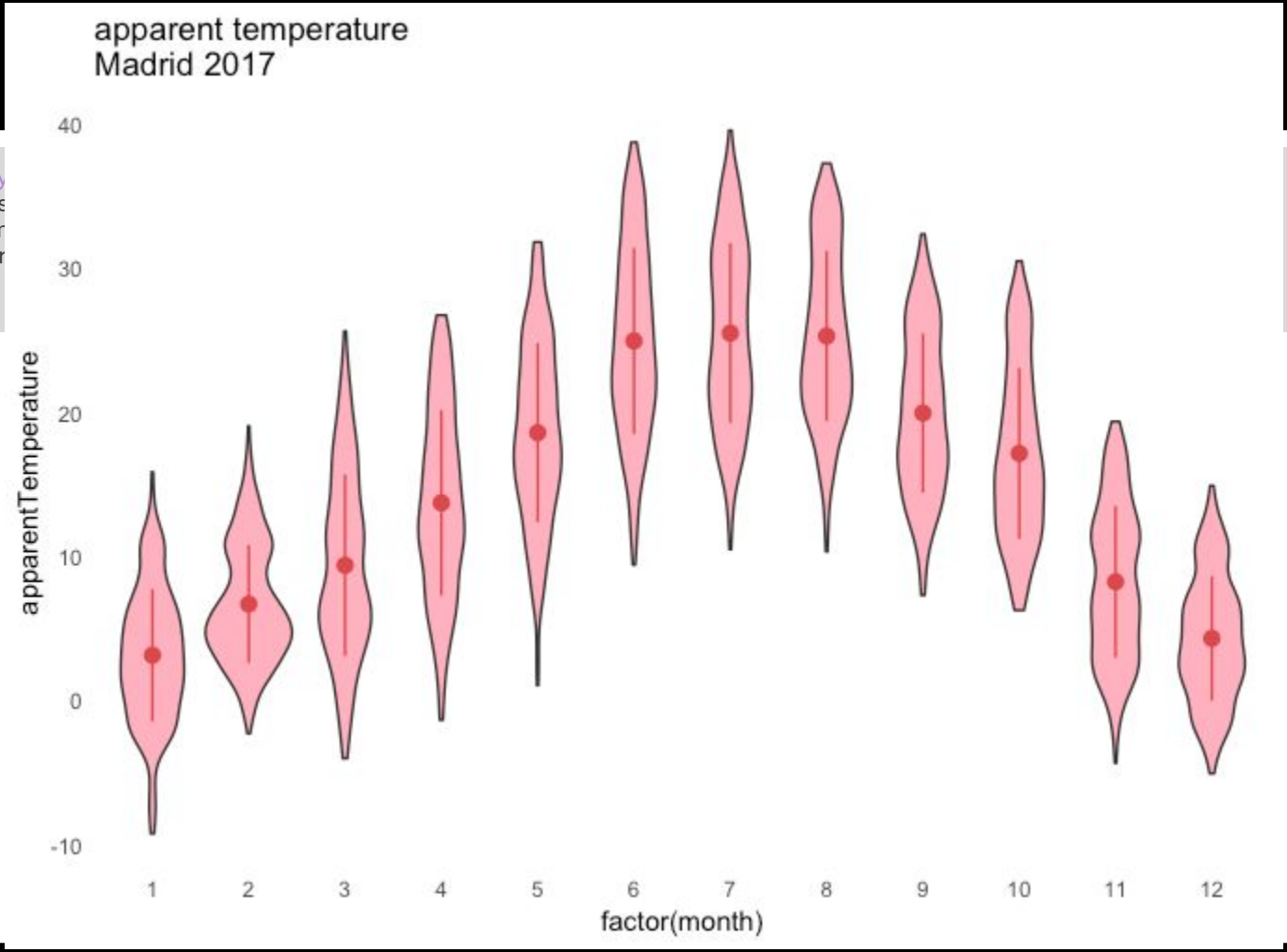


Example

```
# exploratory plot (violin plots)
ggplot(dt,aes(y = apparentTemperature, x = factor(month))) +
  geom_violin() + labs(title = 'apparent temperature \nMadrid 2017')
```

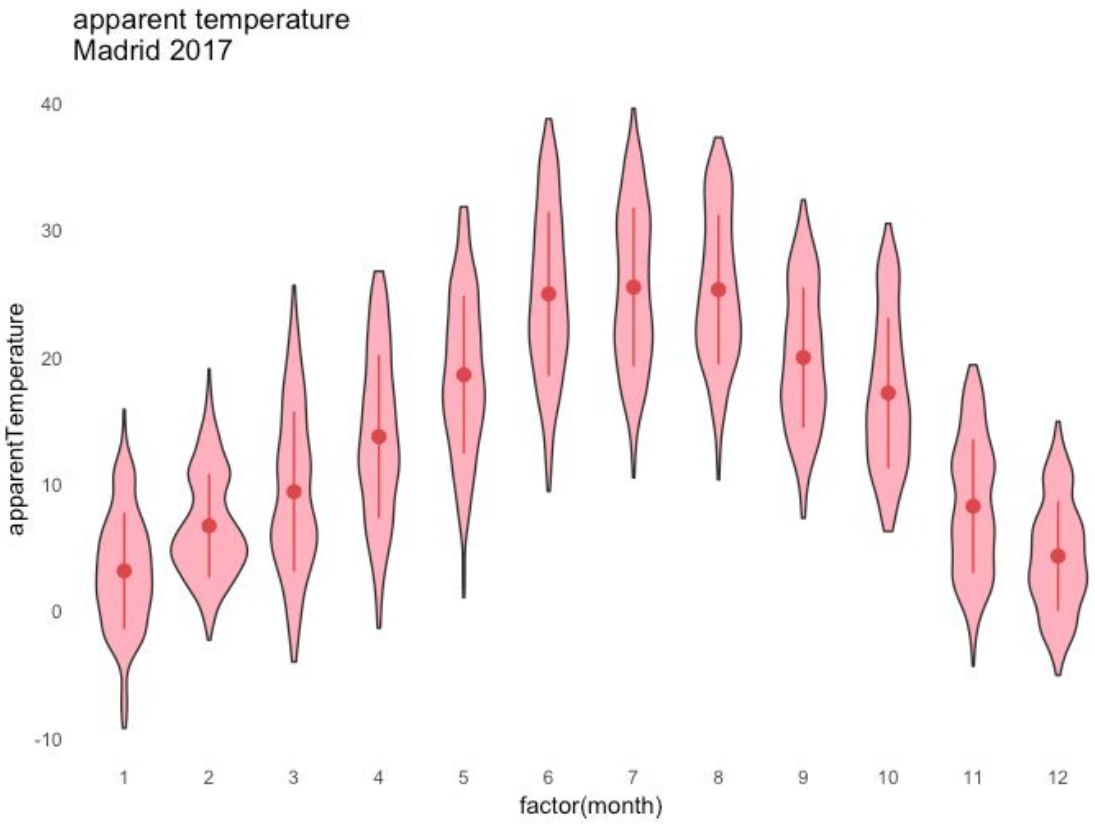


```
# exploratory
ggplot(dt,aes
  geom_violin
  stat_summar
```



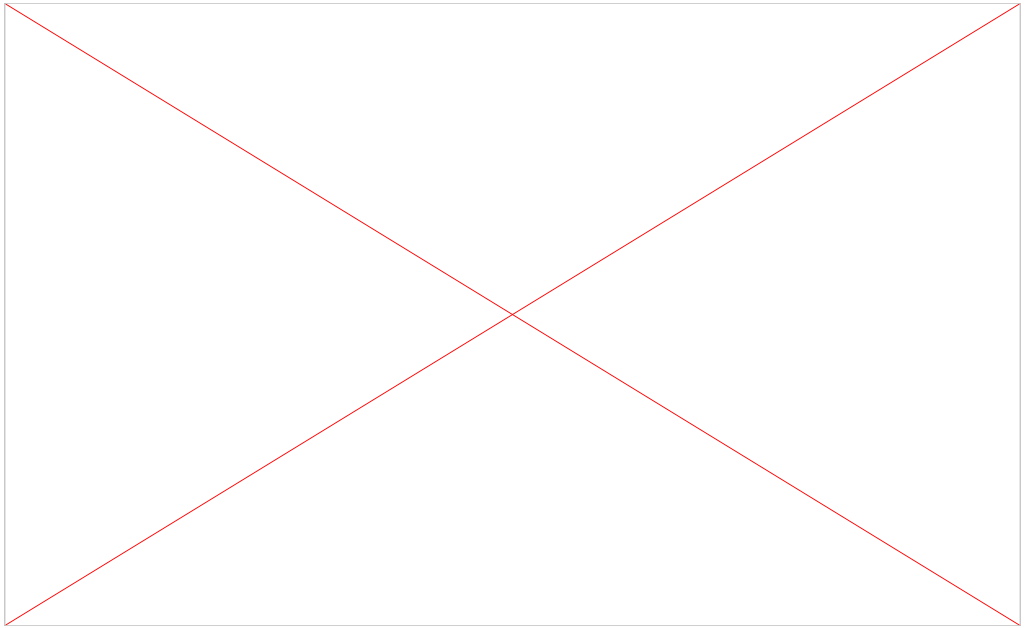
Example

```
# exploratory plot (violin plots)
ggplot(dt,aes(y = apparentTemperature, x =
factor(month))) +
  geom_violin() + labs(title = 'apparent
temperature \nMadrid 2017')+
  stat_summary(fun.data="mean_sdl",
fun.args = list(mult=1),
geom="pointrange", color = "black")
```



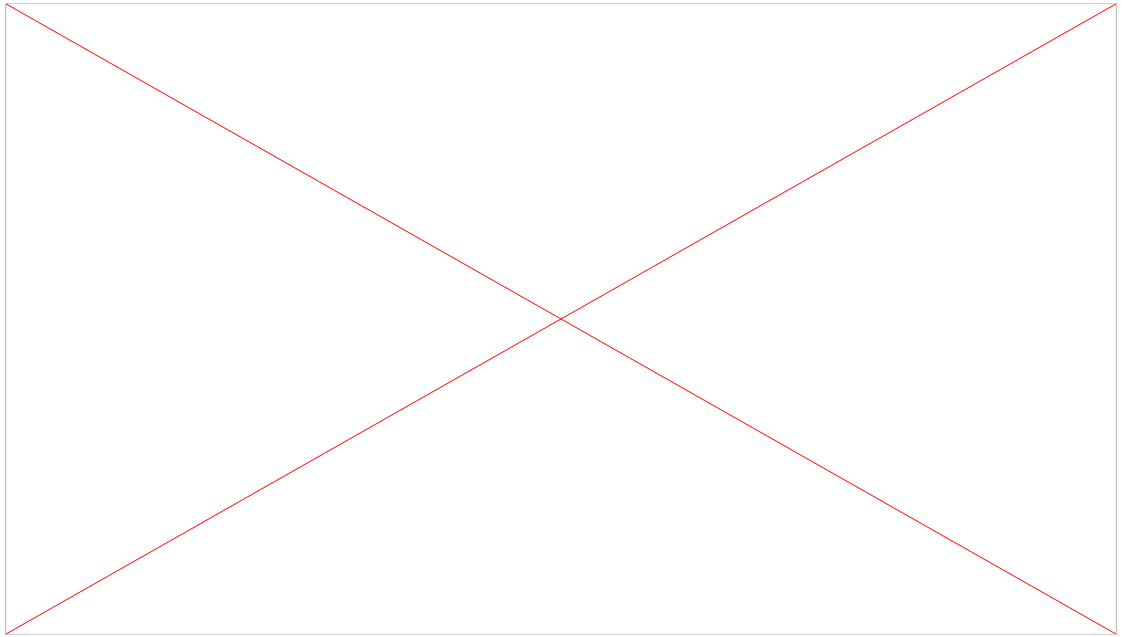
Example

```
# exploratory plot (boxplots)
# fct_rev function is used the flip the order of the levels in a factor
ggplot(dt,aes(y = apparentTemperature, x = fct_rev(month))) +
  geom_boxplot() + labs(title = 'apparent temperature \nMadrid 2017')+
  coord_flip()
```



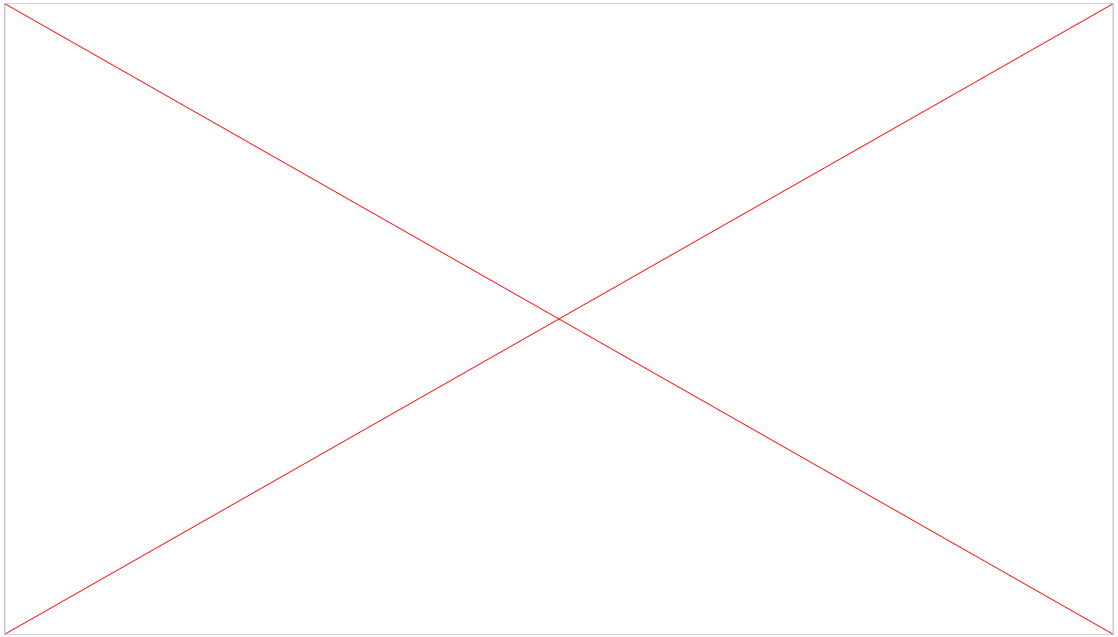
Example

```
# explanatory plot
ggplot(dt,aes(x = apparentTemperature, y = factor(month))) +
  geom_density_ridges2(fill='black',color = 'white',size = 1,scale=2) +  theme_minimal(base_size = 15) +
  theme(axis.title = element_blank(), plot.title = element_text(family='Verdana',hjust = 0.5)) +
  labs(title = 'Apparent Temperature (Celsius) \nMadrid 2017')
```



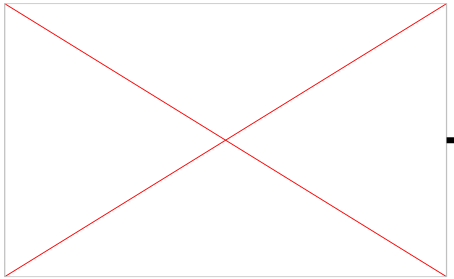
Example

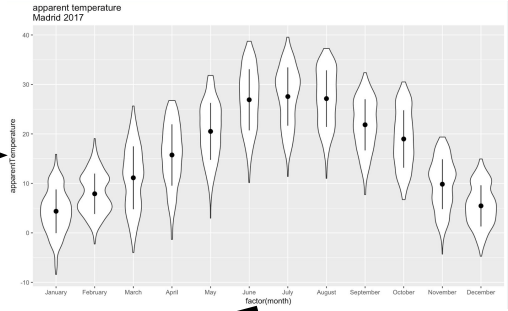
```
# joy division plot
ggplot(dt,aes(x = apparentTemperature,  y = factor(month))) +
  geom_density_ridges2(fill='black',color = 'white',size = 1,scale=2) +
  theme_void(base_size = 15) +
  theme(axis.title = element_blank(), plot.background = element_rect(fill='black'))
```

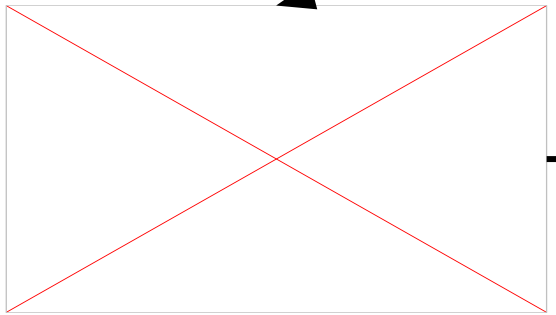


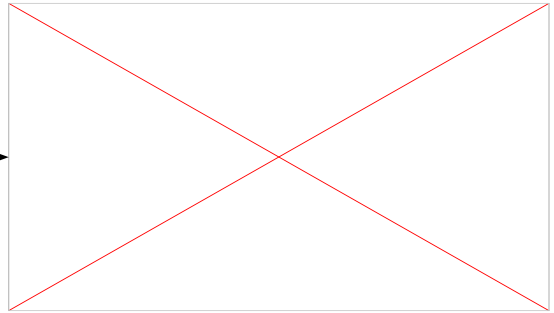
Example

STATS





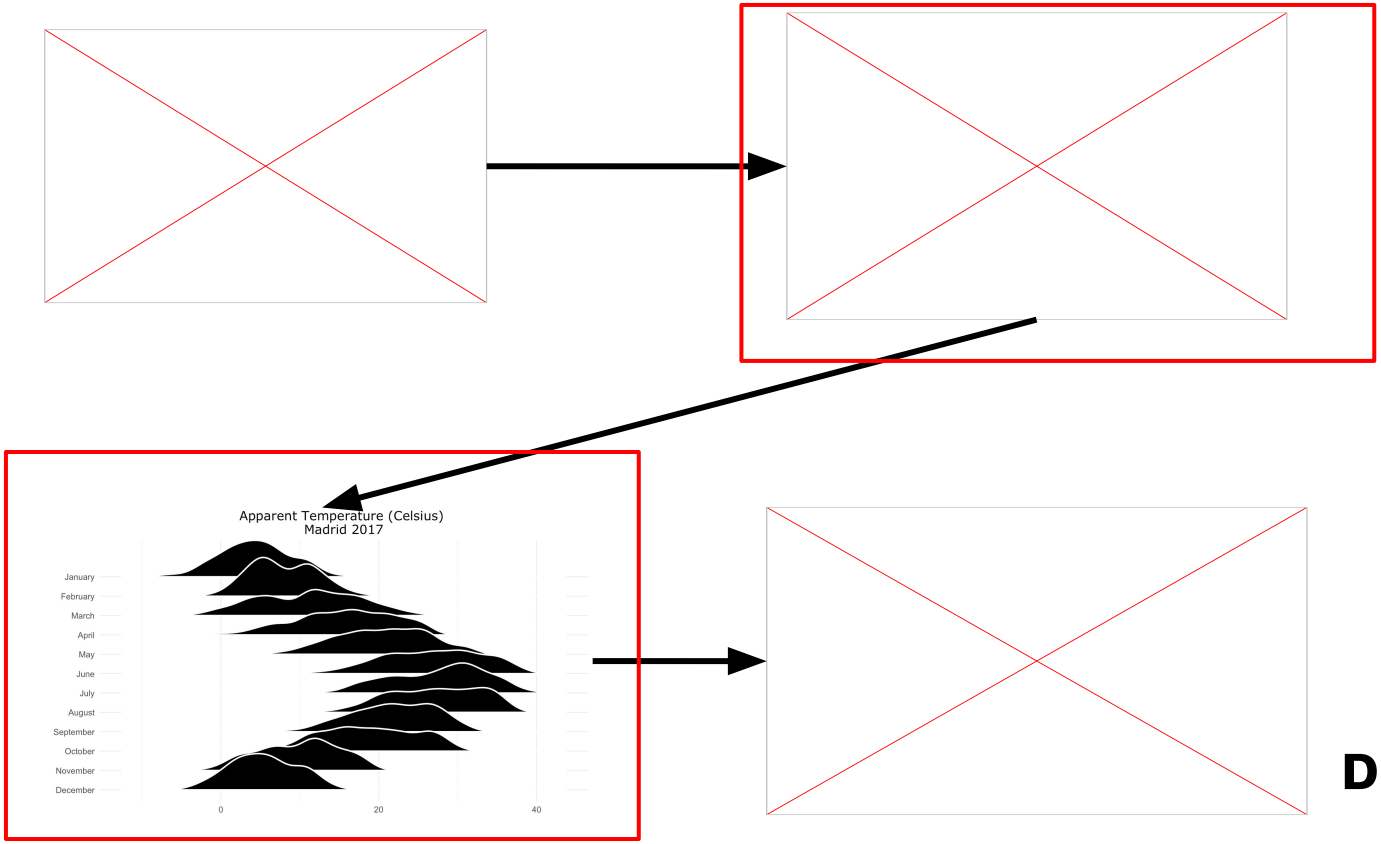




DESIGN

Example

STATS



DESIGN

Tidy Data

Tidy Data

Tidy Data

- real world dataset are not tidy.

Tidy Data

- real world dataset are not tidy.
- ‘tidy datasets are all alike but every messy dataset is messy in its own way’.(Hadley Wickham, *Tidy Data*).

Tidy Data

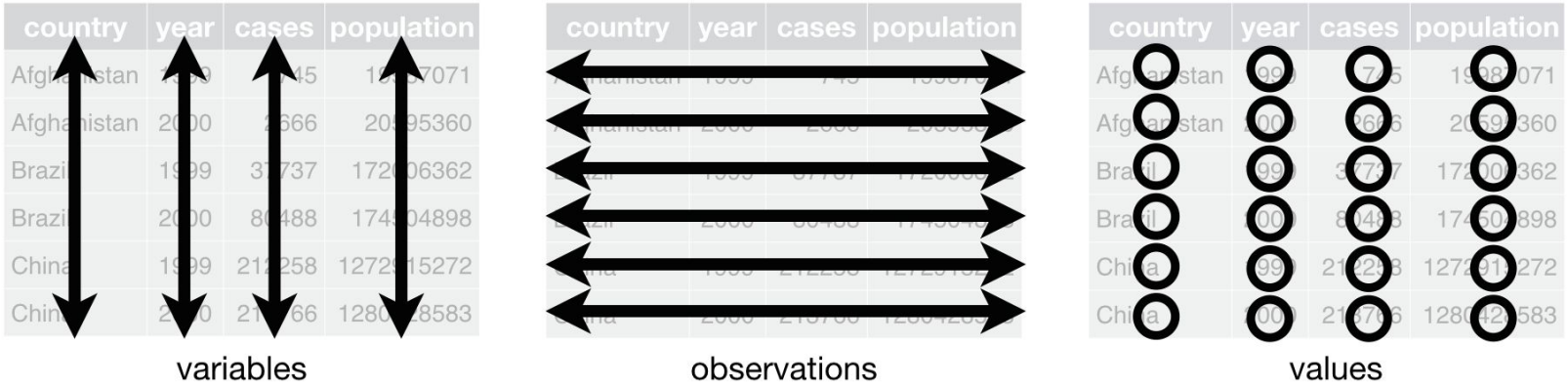
- real world dataset are not tidy.
- ‘tidy datasets are all alike but every messy dataset is messy in its own way’.(Hadley Wickham, *Tidy Data*).
- Tidy data manifesto:

1. each variable forms a column
2. each observation forms a row
3. each type of observational unit forms a table

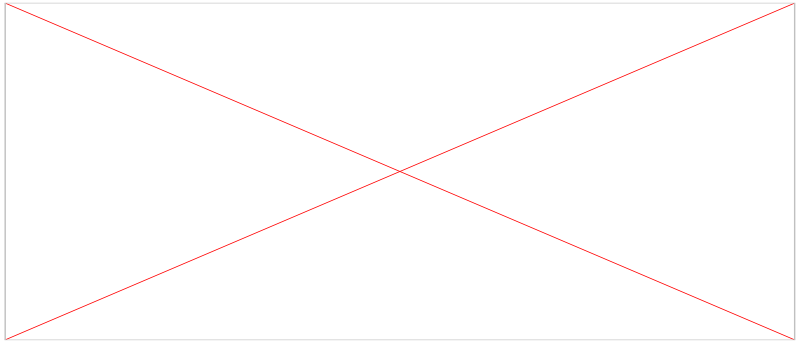
Tidy Data

- Tidy data manifesto:

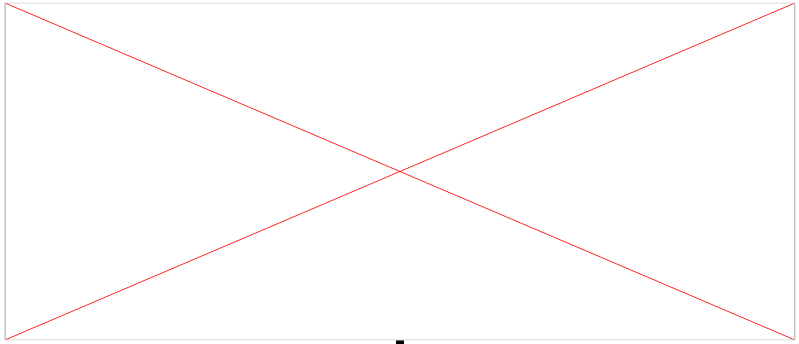
  1. each variable forms a column
  2. each observation forms a row
  3. each type of observational unit forms a table



Tidy Data

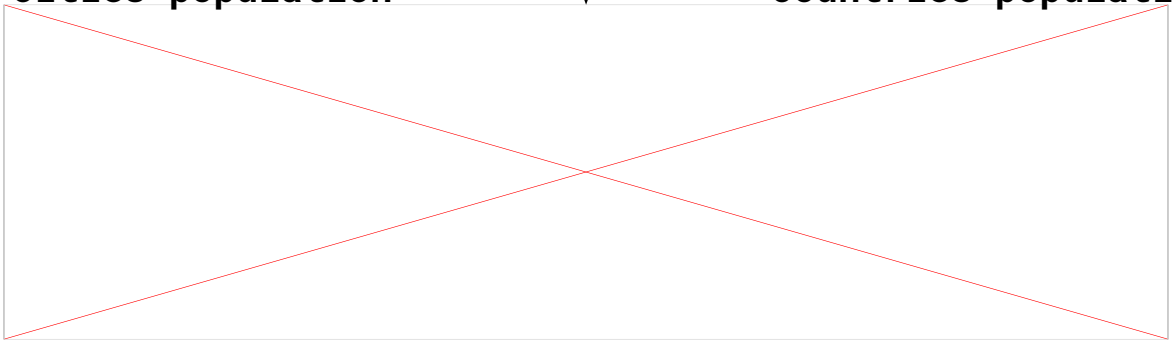


Tidy Data

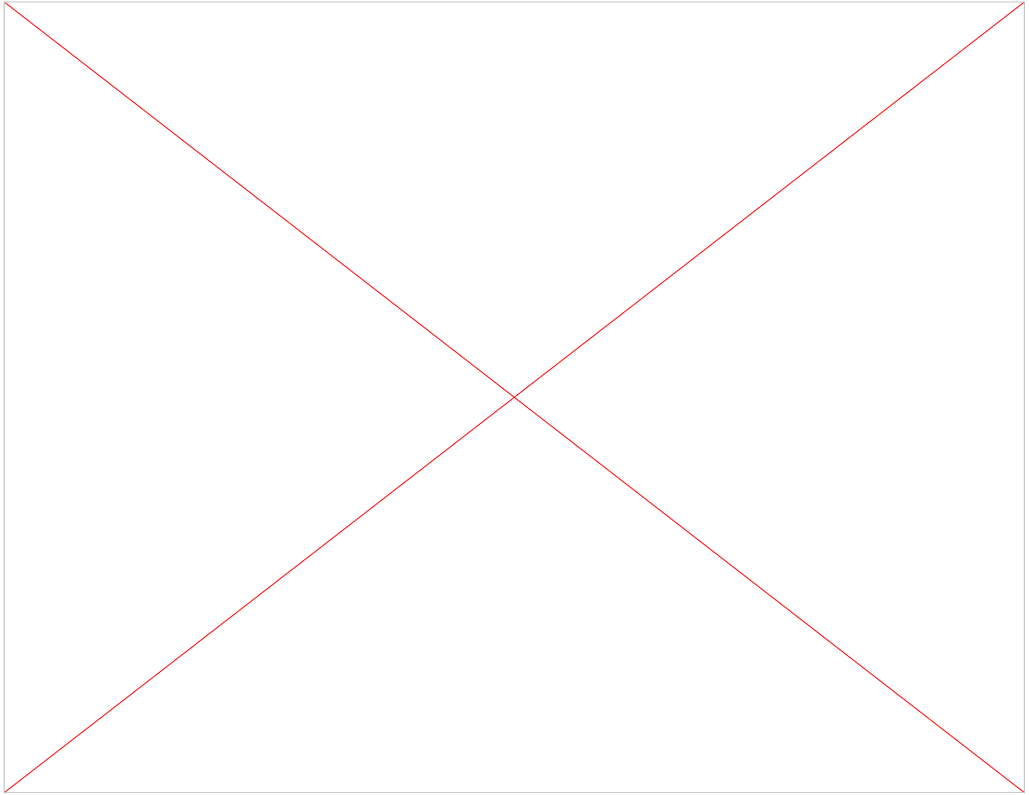


cities population

countries population



Tidy Data



https://www.billboard.com/charts/hot-100 (2018/04/09)

Tidy Data

	artist.inverted	track	time	genre	date.entered	date.peaked	x1st.week	x2nd.week	x29th.week	x76th.week
1	Destiny's Child	Independent Women Part I - (2000)	3:38	Rock	2000-09-23	2000-11-18	78	63		
2	Santana	Maria, Maria - (2000)	4:18	Rock	2000-02-12	2000-04-08	15	8		
3	Savage Garden	I Knew I Loved You - (2000)	4:07	Rock	1999-10-23	2000-01-29	71	48	30	
4	Madonna	Music - (2000)	3:45	Rock	2000-08-12	2000-09-16	41	23		
5	Aguilera, Christina	Come On Over Baby (All I Want Is You) - (2000)	3:38	Rock	2000-08-05	2000-10-14	57	47		

Tidy Data

	artist.inverted	track	time	genre	date.entered	date.peaked	x1st.week	x2nd.week	x29th.week	x76th.week
1	Destiny's Child	Independent Women Part I - (2000)	3:38	Rock	2000-09-23	2000-11-18	78	63		
2	Santana	Maria, Maria - (2000)	4:18	Rock	2000-02-12	2000-04-08	15	8		
3	Savage Garden	I Knew I Loved You - (2000)	4:07	Rock	1999-10-23	2000-01-29	71	48	30	
4	Madonna	Music - (2000)	3:45	Rock	2000-08-12	2000-09-16	41	23		
5	Aguilera, Christina	Come On Over Baby (All I Want Is You) - (2000)	3:38	Rock	2000-08-05	2000-10-14	57	47		

DESCRIPTION

- the name of the artist (***artist.inverted***)
- the name of the song (and the released year) (***track***)
- the date on which the song enters the Billboard Top 100 list (***data.entered***)
- the date on which the song reach it best rank (***date.peaked***)
- the duration of the song (in a character variable) (***time***)
- the genre or the song (***genre***)
- the date of a song first entered the Billboard Top 100 (***date.entered***)
- the date in which this songs reach it best rank (***date.peaked***)
- the rank of the song during the week in which the song enters the Billboard Top 100 list (***x1st.week***)
- the rank of the song during the next week (***x2nd.week***)
- and so on (***x3rd.week***) ... until the 76th week (***x76th.week***)
- In most cases, the song leaves the top 100 list before the week 76, in those cases the cell is filled with a blank (or NA in R)

Tidy Data

	artist.inverted	track	time	genre	date.entered	date.peaked	x1st.week	x2nd.week	x29th.week	x76th.week
1	Destiny's Child	Independent Women Part I - (2000)	3:38	Rock	2000-09-23	2000-11-18	78	63		
2	Santana	Maria, Maria - (2000)	4:18	Rock	2000-02-12	2000-04-08	15	8		
3	Savage Garden	I Knew I Loved You - (2000)	4:07	Rock	1999-10-23	2000-01-29	71	48	30	
4	Madonna	Music - (2000)	3:45	Rock	2000-08-12	2000-09-16	41	23		
5	Aguilera, Christina	Come On Over Baby (All I Want Is You) - (2000)	3:38	Rock	2000-08-05	2000-10-14	57	47		

Tidy Data

	artist.inverted	track	time	genre	date.entered	date.peaked	x1st.week	x2nd.week	x29th.week	x76th.week
1	Destiny's Child	Independent Women Part I - (2000)	3:38	Rock	2000-09-23	2000-11-18	78	63		
2	Santana	Maria, Maria - (2000)	4:18	Rock	2000-02-12	2000-04-08	15	8		
3	Savage Garden	I Knew I Loved You - (2000)	4:07	Rock	1999-10-23	2000-01-29	71	48	30	
4	Madonna	Music - (2000)	3:45	Rock	2000-08-12	2000-09-16	41	23		
5	Aguilera, Christina	Come On Over Baby (All I Want Is You) - (2000)	3:38	Rock	2000-08-05	2000-10-14	57	47		

PROBLEMS

- **column header are values, not variable names**
  - the header of the columns with the value of the rank of each song per week, are actually values of a variable instead of column names.
- **multiple variables stored in one column**
  - the column track gives information about the name of the song and the released year.
- **names of the columns**
  - the names of the columns are not adequate nor well formatted
- **data types**
  - the column time must be numeric
  - the columns date.entered and date.peaked must have a date format (not a character)

Tidy Data

```r
##### READ DATA
# read the billboard dataset and take a look at it
dt <- fread('dtviz/data/billboard.csv')
dt
```

```
       artist.inverted                                      track time genre date.entered date.peaked x1st.week x2nd.week x3rd.week x4th.week x5th.week x6th.week :
  1:     Destiny's Child                 Independent Women Part I - 2000 3:38  Rock   2000-09-23  2000-11-18        78        63        49        33        23        15
  2:             Santana                            Maria, Maria - 2000 4:18  Rock   2000-02-12  2000-04-08        15         8         6         5         2         3
  3:       Savage Garden                      I Knew I Loved You - 2000 4:07  Rock   1999-10-23  2000-01-29        71        48        43        31        20        13
  4:             Madonna                                  Music - 2000 3:45  Rock   2000-08-12  2000-09-16        41        23        18        14         2         1
  5: Aguilera, Christina Come On Over Baby (All I Want Is You) - 2000 3:38  Rock   2000-08-05  2000-10-14        57        47        45        29        23        18
 ---
313:    Ghostface Killah                       Cherchez LaGhost - 2000 3:04   R&B   2000-08-05  2000-08-05        98        NA        NA        NA        NA        NA
314:         Smith, Will                            Freakin' It - 2000 3:58   Rap   2000-02-12  2000-02-12        99        99        99        99        NA        NA
315:       Zombie Nation                          Kernkraft 400 - 2000 3:30  Rock   2000-09-02  2000-09-02        99        99        NA        NA        NA        NA
316:      Eastsidaz, The                                Got Beef - 2000 3:58   Rap   2000-07-01  2000-07-01        99        99        NA        NA        NA        NA
317:              Fragma                         Toca's Miracle - 2000 3:22   R&B   2000-10-28  2000-10-28        99        NA        NA        NA        NA        NA
```

Tidy Data

```r
##### READ DATA
# read the billboard dataset and take a look at it
dt <- fread('dtviz/data/billboard.csv')
dt
```

melt and cast

melt

row	a	b	c
A	1	4	7
B	2	5	8
C	3	6	9

cast

row	column	value
A	a	1
B	a	2
C	a	3
A	b	4
B	b	5
C	b	6
A	c	7
B	c	8
C	c	9

Melt
Turn columns into rows
From wide to long

Cast
Turn rows into columns
From long to wide

Tidy Data

```r
##### MELT THE DATASET
# take the names of the columns we want to melt
week_column_names <- names(dt)[grepl('.week$',names(dt))]
# store the rest of the column names because they are going to be id.vars
id_column_names <- setdiff(colnames(dt),week_column_names)
# melt the dataset using melt() function from data.table package
dt_melted <- melt(dt,id.vars = id_column_names,measure.vars = week_column_names) week_column_names)
```

```
        artist.inverted                                     track time genre date.entered date.peaked x1st.week x2nd.week x3rd.week x4th.week x5th.week x6th.week :
  1:     Destiny's Child                Independent Women Part I - 2000 3:38  Rock   2000-09-23  2000-11-18        78        63        49        33        23        15
  2:             Santana                            Maria, Maria - 2000 4:18  Rock   2000-02-12  2000-04-08        15         8         6         5         2         3
  3:       Savage Garden                     I Knew I Loved You - 2000 4:07  Rock   1999-10-23  2000-01-29        71        48        43        31        20        13
  4:             Madonna                                  Music - 2000 3:45  Rock   2000-08-12  2000-09-16        41        23        18        14         2         1
  5: Aguilera, Christina Come On Over Baby (All I Want Is You) - 2000 3:38  Rock   2000-08-05  2000-10-14        57        47        45        29        23        18
 ---
313:    Ghostface Killah                       Cherchez LaGhost - 2000 3:04   R&B   2000-08-05  2000-08-05        98        NA        NA        NA        NA        NA
314:         Smith, Will                            Freakin' It - 2000 3:58   Rap   2000-02-12  2000-02-12        99        99        99        99        NA        NA
315:       Zombie Nation                          Kernkraft 400 - 2000 3:30  Rock   2000-09-02  2000-09-02        99        99        NA        NA        NA        NA
316:      Eastsidaz, The                                Got Beef - 2000 3:58   Rap   2000-07-01  2000-07-01        99        99        NA        NA        NA        NA
317:              Fragma                         Toca's Miracle - 2000 3:22   R&B   2000-10-28  2000-10-28        99        NA        NA        NA        NA        NA
```

Tidy Data

```r
##### MELT THE DATASET
# take the names of the columns we want to melt
week_column_names <- names(dt)[grepl('.week$',names(dt))]
# store the rest of the column names because they are going to be id.vars
id_column_names <- setdiff(colnames(dt),week_column_names)
# melt the dataset using melt() function from data.table package
dt_melted <- melt(dt,id.vars = id_column_names,measure.vars = week_column_names)
```

```
        artist.inverted                                      track time genre date.entered date.peaked   variable value
    1:     Destiny's Child                 Independent Women Part I - 2000 3:38  Rock   2000-09-23  2000-11-18  x1st.week    78
    2:             Santana                              Maria, Maria - 2000 4:18  Rock   2000-02-12  2000-04-08  x1st.week    15
    3:       Savage Garden                       I Knew I Loved You - 2000 4:07  Rock   1999-10-23  2000-01-29  x1st.week    71
    4:             Madonna                                   Music - 2000 3:45  Rock   2000-08-12  2000-09-16  x1st.week    41
    5: Aguilera, Christina Come On Over Baby (All I Want Is You) - 2000 3:38  Rock   2000-08-05  2000-10-14  x1st.week    57
   ---
24088:    Ghostface Killah                        Cherchez LaGhost - 2000 3:04   R&B   2000-08-05  2000-08-05 x76th.week    NA
24089:         Smith, Will                             Freakin' It - 2000 3:58   Rap   2000-02-12  2000-02-12 x76th.week    NA
24090:       Zombie Nation                           Kernkraft 400 - 2000 3:30  Rock   2000-09-02  2000-09-02 x76th.week    NA
24091:      Eastsidaz, The                                Got Beef - 2000 3:58   Rap   2000-07-01  2000-07-01 x76th.week    NA
24092:              Fragma                           Toca's Miracle - 2000 3:22   R&B   2000-10-28  2000-10-28 x76th.week    NA
```

Tidy Data

```r
##### FILTER NAs
# remove those rows with no value (because the song left the list before the week 76)
dt_melted <- dt_melted[!is.na(value)]
dt_melted
# we have removed the 78% of rows
```

```
         artist.inverted                                      track time   genre date.entered date.peaked   variable value
   1:     Destiny's Child                Independent Women Part I - 2000 3:38    Rock   2000-09-23  2000-11-18  x1st.week    78
   2:             Santana                            Maria, Maria - 2000 4:18    Rock   2000-02-12  2000-04-08  x1st.week    15
   3:       Savage Garden                     I Knew I Loved You - 2000 4:07    Rock   1999-10-23  2000-01-29  x1st.week    71
   4:             Madonna                                  Music - 2000 3:45    Rock   2000-08-12  2000-09-16  x1st.week    41
   5: Aguilera, Christina Come On Over Baby (All I Want Is You) - 2000 3:38    Rock   2000-08-05  2000-10-14  x1st.week    57
  ---
5303:            Lonestar                                 Amazed - 2000 4:25 Country   1999-06-05  2000-03-04 x63rd.week    45
5304:               Creed                                 Higher - 2000 5:16    Rock   1999-09-11  2000-07-22 x63rd.week    50
5305:            Lonestar                                 Amazed - 2000 4:25 Country   1999-06-05  2000-03-04 x64th.week    50
5306:               Creed                                 Higher - 2000 5:16    Rock   1999-09-11  2000-07-22 x64th.week    50
5307:               Creed                                 Higher - 2000 5:16    Rock   1999-09-11  2000-07-22 x65th.week    49
```

Tidy Data

```r
##### QUICK EXPLORATION
# to know what we have to do next
str(dt_melted)
```

```
Classes 'data.table' and 'data.frame':  5307 obs. of  8 variables:
 $ artist.inverted: chr  "Destiny's Child" "Santana" "Savage Garden'
 $ track          : chr  "Independent Women Part I - 2000" "Maria, M
 $ time           : chr  "3:38" "4:18" "4:07" "3:45" ...
 $ genre          : chr  "Rock" "Rock" "Rock" "Rock" ...
 $ date.entered   : chr  "2000-09-23" "2000-02-12" "1999-10-23" "200
 $ date.peaked    : chr  "2000-11-18" "2000-04-08" "2000-01-29" "200
 $ variable       : Factor w/ 76 levels "x1st.week","x2nd.week",..:
 $ value          : int  78 15 71 41 57 59 83 63 77 81 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

Tidy Data

```r
##### TWO VARIABLE IN THE SAME COLUMN (column 'track')
# separate the track name from the year (using tidyr package)
dt_melted <- tidyr::separate(dt_melted,track,sep=' - ', into = c('track','year'))
# remove year variable (it is redundant)
dt_melted[,year := NULL]
dt_melted
```

```
> dt_melted
         artist.inverted                                track time   genre date.entered date.peaked   variable value
   1:     Destiny's Child              Independent Women Part I 3:38    Rock   2000-09-23  2000-11-18  x1st.week    78
   2:             Santana                           Maria, Maria 4:18    Rock   2000-02-12  2000-04-08  x1st.week    15
   3:       Savage Garden                   I Knew I Loved You 4:07    Rock   1999-10-23  2000-01-29  x1st.week    71
   4:             Madonna                                 Music 3:45    Rock   2000-08-12  2000-09-16  x1st.week    41
   5: Aguilera, Christina Come On Over Baby (All I Want Is You) 3:38    Rock   2000-08-05  2000-10-14  x1st.week    57
  ---
5303:            Lonestar                               Amazed 4:25 Country   1999-06-05  2000-03-04 x63rd.week    45
5304:               Creed                               Higher 5:16    Rock   1999-09-11  2000-07-22 x63rd.week    50
5305:            Lonestar                               Amazed 4:25 Country   1999-06-05  2000-03-04 x64th.week    50
5306:               Creed                               Higher 5:16    Rock   1999-09-11  2000-07-22 x64th.week    50
5307:               Creed                               Higher 5:16    Rock   1999-09-11  2000-07-22 x65th.week    49
```

Tidy Data

```r
##### TWO VARIABLE IN THE SAME COLUMN (column 'time')
# separate the track name from the year (using tidyr package)
dt_melted <- tidyr::separate(dt_melted,time,sep=':', into = c('minutes','seconds'))
# create variable duration (we need to know that a minute has 60 seconds)
dt_melted[,duration_min := as.numeric(minutes) + as.numeric(seconds)/60]
# remove redundants variable
dt_melted[,c('time','minutes','seconds') := NULL]
dt_melted
```

```
> dt_melted
         artist.inverted                                 track   genre date.entered date.peaked   variable value duration_min
   1:     Destiny's Child                Independent Women Part I    Rock   2000-09-23  2000-11-18  x1st.week    78     3.633333
   2:             Santana                            Maria, Maria    Rock   2000-02-12  2000-04-08  x1st.week    15     4.300000
   3:       Savage Garden                     I Knew I Loved You    Rock   1999-10-23  2000-01-29  x1st.week    71     4.116667
   4:             Madonna                                   Music    Rock   2000-08-12  2000-09-16  x1st.week    41     3.750000
   5: Aguilera, Christina Come On Over Baby (All I Want Is You)    Rock   2000-08-05  2000-10-14  x1st.week    57     3.633333
  ---
5303:            Lonestar                                Amazed Country   1999-06-05  2000-03-04 x63rd.week    45     4.416667
5304:               Creed                                Higher    Rock   1999-09-11  2000-07-22 x63rd.week    50     5.266667
5305:            Lonestar                                Amazed Country   1999-06-05  2000-03-04 x64th.week    50     4.416667
5306:               Creed                                Higher    Rock   1999-09-11  2000-07-22 x64th.week    50     5.266667
5307:               Creed                                Higher    Rock   1999-09-11  2000-07-22 x65th.week    49     5.266667
```

Tidy Data

```
##### CLEAN VARIABLE WITH THE INFORMATION OF THE NUMBER OF THE WEEK
# using gsub function and regular expressions
dt_melted[,week := gsub('[^0-9]','',variable)]
dt_melted[,week := as.integer(week)]
# remove variable column
dt_melted[,variable := NULL]
dt_melted
```

Tidy Data

```r
##### CHANGE THE NAMES OF SOME COLUMNS
setnames(x = dt_melted,
         old = c('artist.inverted','date.entered','date.peaked'),
         new = c('artist','date','date_peaked'))
dt_melted
```

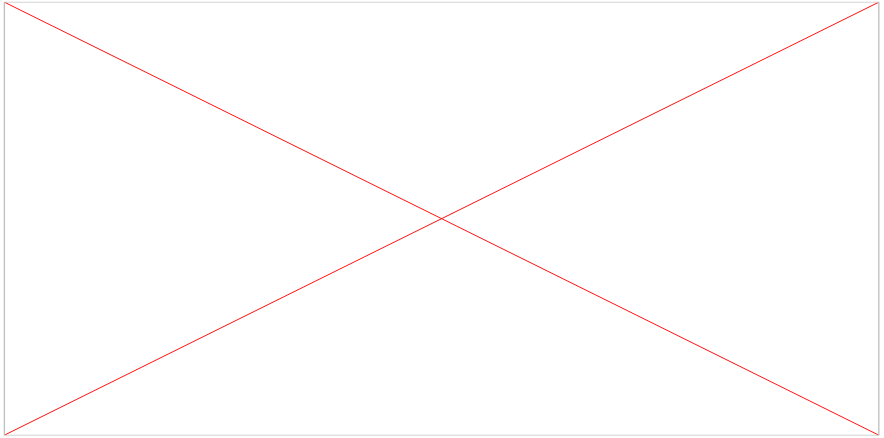
```
> dt_melted
                artist                                track   genre       date date_peaked value duration_min week
   1:     Destiny's Child              Independent Women Part I    Rock 2000-09-23  2000-11-18    78     3.633333    1
   2:            Santana                           Maria, Maria    Rock 2000-02-12  2000-04-08    15     4.300000    1
   3:       Savage Garden                   I Knew I Loved You    Rock 1999-10-23  2000-01-29    71     4.116667    1
   4:            Madonna                                  Music    Rock 2000-08-12  2000-09-16    41     3.750000    1
   5: Aguilera, Christina Come On Over Baby (All I Want Is You)    Rock 2000-08-05  2000-10-14    57     3.633333    1
  ---
5303:           Lonestar                               Amazed Country 1999-06-05  2000-03-04    45     4.416667   63
5304:              Creed                               Higher    Rock 1999-09-11  2000-07-22    50     5.266667   63
5305:           Lonestar                               Amazed Country 1999-06-05  2000-03-04    50     4.416667   64
5306:              Creed                               Higher    Rock 1999-09-11  2000-07-22    50     5.266667   64
5307:              Creed                               Higher    Rock 1999-09-11  2000-07-22    49     5.266667   65
```

Tidy Data

```
##### CHANGE THE DATA TYPES
factor_variables <- c('genre')
character_variables <- c('artist','track')
date_variables <- c('date','date_peaked')
integer_variable <- c('value','week')
numeric_variables <- c('duration_min')
```

Tidy Data

```r
# using a trick of data.table
dt_melted[,c(factor_variables) := map(.SD,as.factor),.SDcols = c(factor_variables)]
dt_melted[,c(character_variables) := map(.SD,as.character),.SDcols = c(character_variables)]
dt_melted[,c(date_variables) := map(.SD,as.Date),.SDcols = c(date_variables)]
dt_melted[,c(integer_variable) := map(.SD,as.integer),.SDcols = c(integer_variable)]
dt_melted[,c(numeric_variables) := map(.SD,as.numeric),.SDcols = c(numeric_variables)]
str(dt_melted)
```



Tidy Data

```
##### DATE VARIABLE IS A STATIC VARIABLE
head(dt_melted[artist == 'Santana' & track == 'Maria, Maria'],10)
```

Tidy Data

```r
##### MAKE DATE A DYNAMIC VARIABLE
# number of days to add
dt_melted[,n_days_to_add := (week -1)*7]
# coerce into date
dt_melted[,date:=date + lubridate::days(n_days_to_add)]
# remove auxiliary variable
dt_melted[,n_days_to_add := NULL]
```

Tidy Data

```r
##### STATIC AND DYNAMIC VARIABLE
static_variables <- c('artist','track','genre','duration_min','date_peaked')
dynamic_variable <- c('date','value','week')
setcolorder(dt_melted,c(static_variables,dynamic_variable))
head(dt_melted[artist == 'Santana' & track == 'Maria, Maria'],10)
```

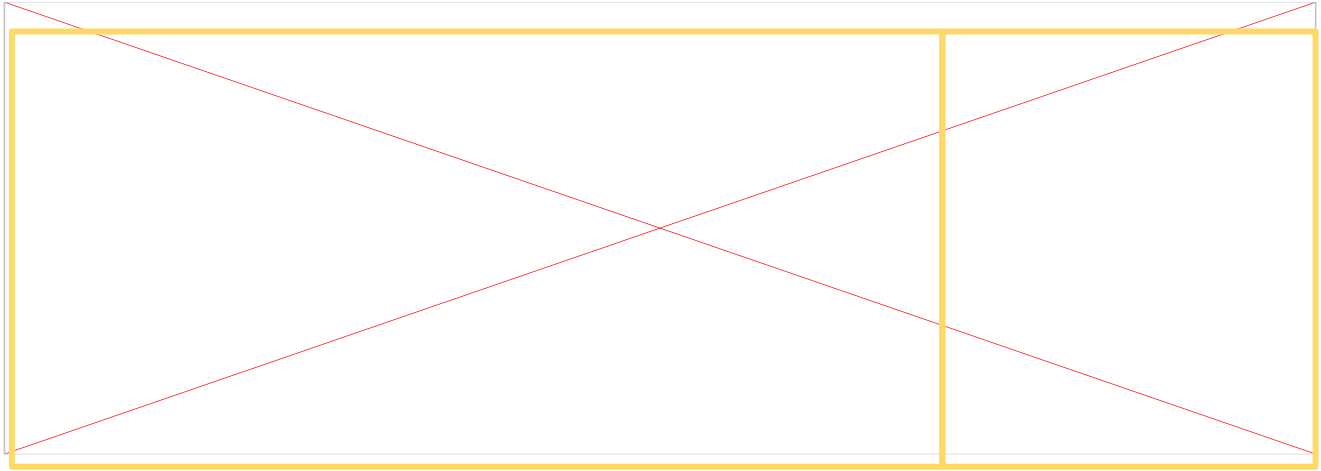
```
> head(dt_melted[artist == 'Santana' & track == 'Maria, Maria'],10)
     artist        track genre duration_min date_peaked       date value week
 1: Santana Maria, Maria  Rock          4.3  2000-04-08 2000-02-12    15    1
 2: Santana Maria, Maria  Rock          4.3  2000-04-08 2000-02-19     8    2
 3: Santana Maria, Maria  Rock          4.3  2000-04-08 2000-02-26     6    3
 4: Santana Maria, Maria  Rock          4.3  2000-04-08 2000-03-04     5    4
 5: Santana Maria, Maria  Rock          4.3  2000-04-08 2000-03-11     2    5
 6: Santana Maria, Maria  Rock          4.3  2000-04-08 2000-03-18     3    6
 7: Santana Maria, Maria  Rock          4.3  2000-04-08 2000-03-25     2    7
 8: Santana Maria, Maria  Rock          4.3  2000-04-08 2000-04-01     2    8
 9: Santana Maria, Maria  Rock          4.3  2000-04-08 2000-04-08     1    9
10: Santana Maria, Maria  Rock          4.3  2000-04-08 2000-04-15     1   10
```

Tidy Data

```r
##### STATIC AND DYNAMIC VARIABLE
static_variables <- c('artist','track','genre','duration_min','date_peaked')
dynamic_variable <- c('date','value','week')
setcolorder(dt_melted,c(static_variables,dynamic_variable))
head(dt_melted[artist == 'Santana' & track == 'Maria, Maria'],10)
```

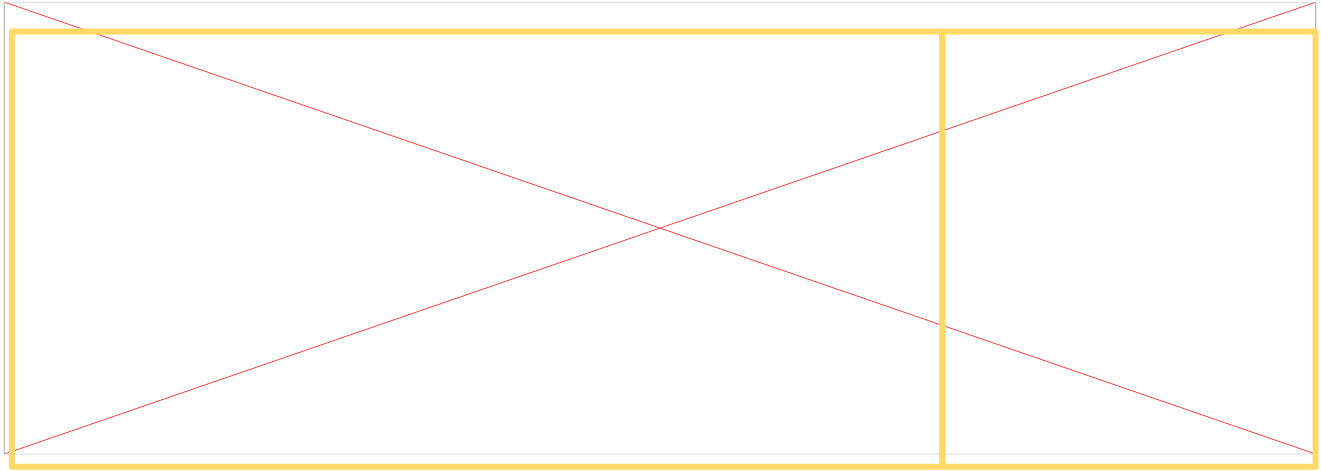
Tidy Data

```
##### TWO OBSERVATIONAL UNITS IN THE SAME TABLE
head(dt_melted[artist == 'Santana' & track == 'Maria, Maria'],10)
```



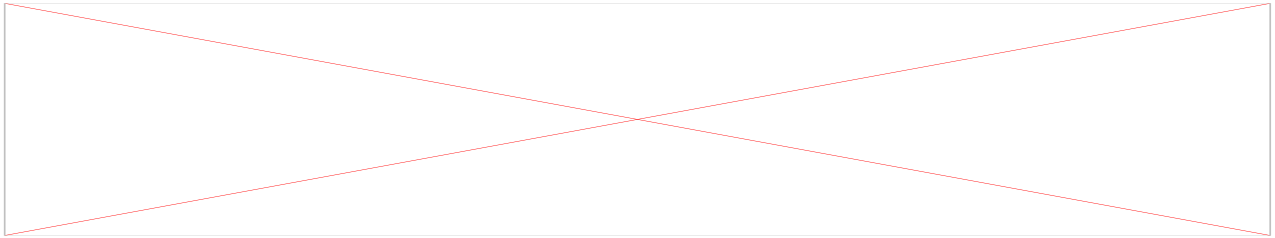
Tidy Data

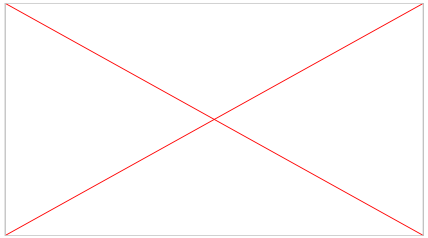
```r
##### CREATE A DATASET
# first we need an id per track (data.table GRP function)
dt_melted[,track_id := .GRP,by =c('track','artist')]
# create two separate tables
track <- dt_melted[,c('track_id',static_variables),with=F]
rank <- dt_melted[,c('track_id',dynamic_variable),with=F]
```



Tidy Data

```
##### CREATE A DATASET
# remove duplicates
track <- unique(track)
rank <- unique(rank)
# look at inside
head(track)
head(rank)
```





Tidy Data

```r
##### STORE_TABLE IN LIST
billboard <- list()
billboard[['track']] <- track
billboard[['rank']] <- rank
##### WHAT IS INSIDE THE LIST
billboard$track
billboard$rank
```

Tidy Data

```r
##### STORE_TABLE IN LIST
billboard <- list()
billboard[['track']] <- track
billboard[['rank']] <- rank
##### WHAT IS INSIDE THE LIST
billboard$track
billboard$rank
```

Exercise

Number of tuberculosis cases in Afghanistan, Brazil and China during 1999 and 2000

1

V1	V2	V3
country	1999	2000
Afghanistan	19987071	20595360
Brazil	172006362	174504898
China	1272915272	1280428583

2

3

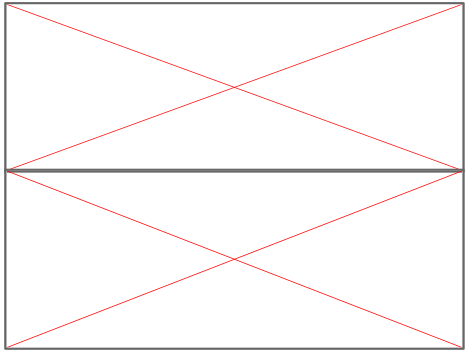
4

5

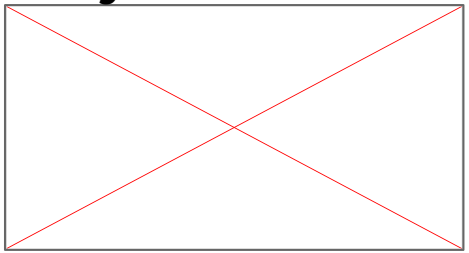
Exercise

Number of tuberculosis cases in Afghanistan, Brazil and China during 1999 and 2000

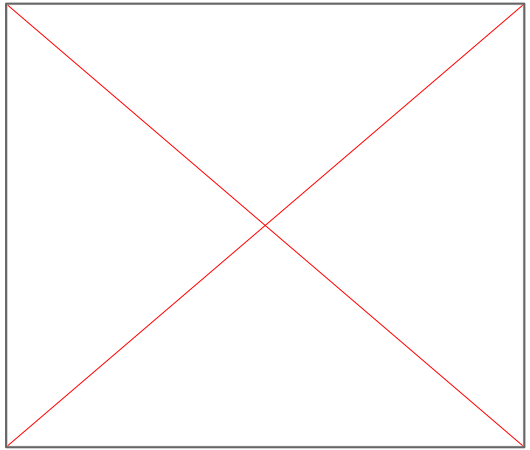
1



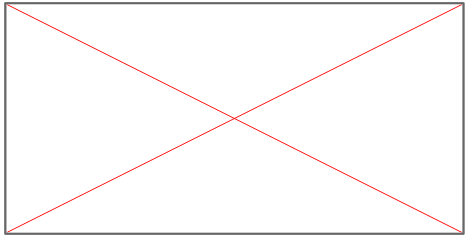
Tidy data



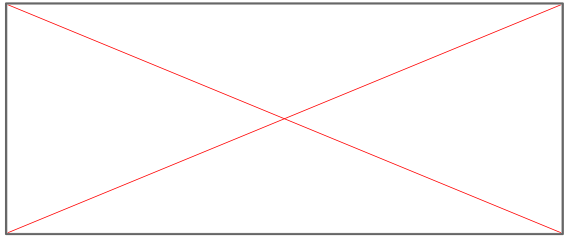
3



4



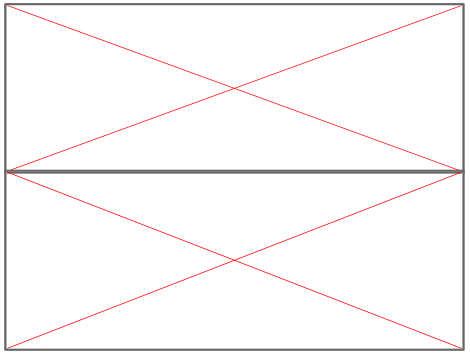
5



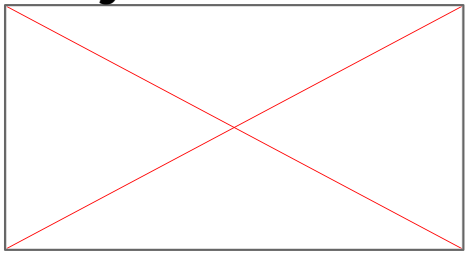
Exercise

Number of tuberculosis cases in Afghanistan, Brazil and China during 1999 and 2000

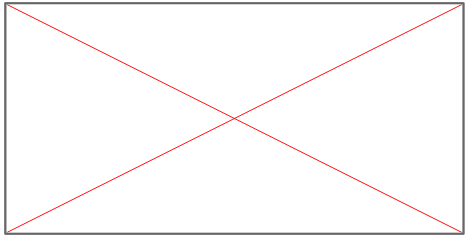
Columns instead of rows



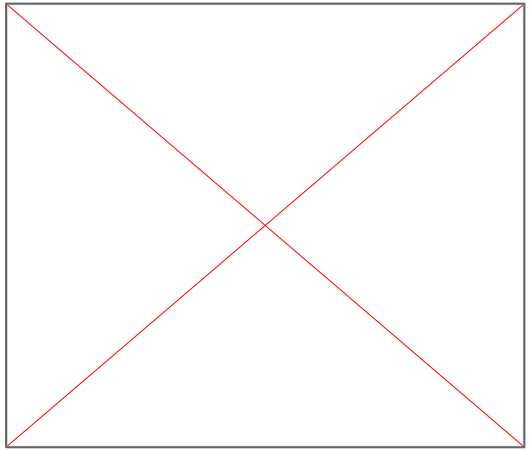
Tidy data



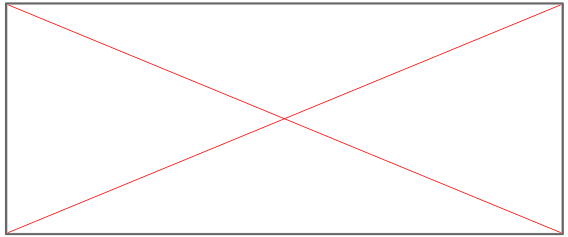
4



3



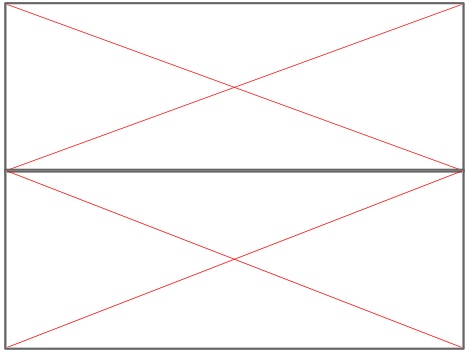
5



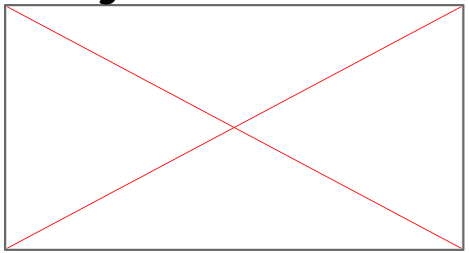
Exercise

Number of tuberculosis cases in Afghanistan, Brazil and China during 1999 and 2000

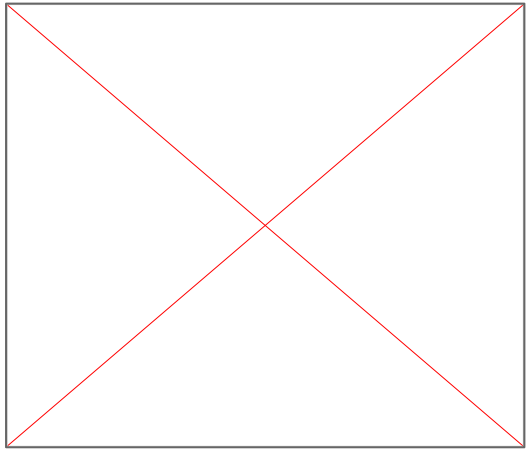
Columns instead of rows



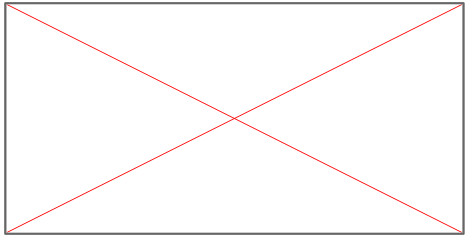
Tidy data



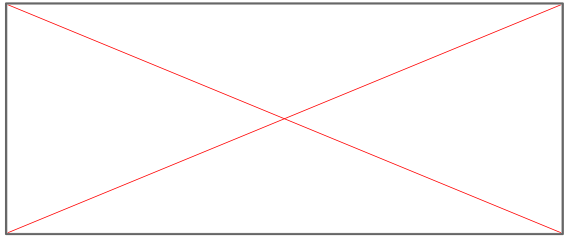
Rows instead of columns



4



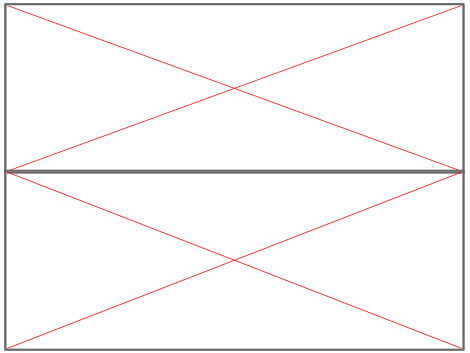
5



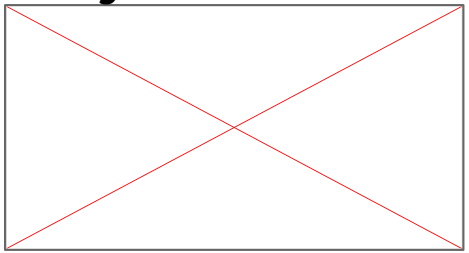
Exercise

Number of tuberculosis cases in Afghanistan, Brazil and China during 1999 and 2000

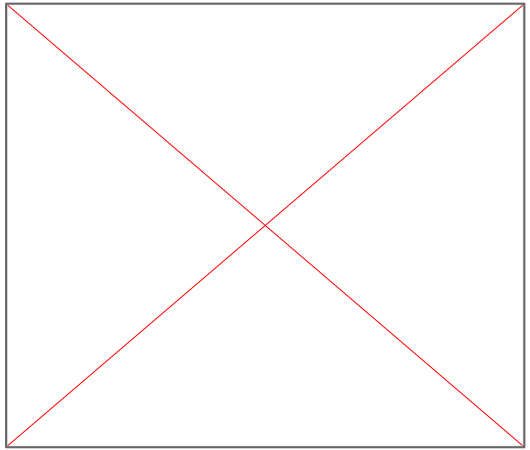
Columns instead of rows



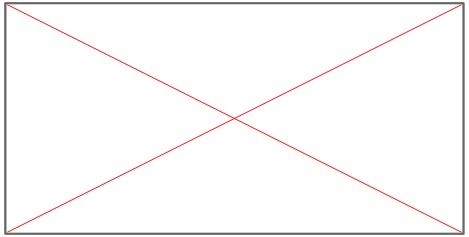
Tidy data



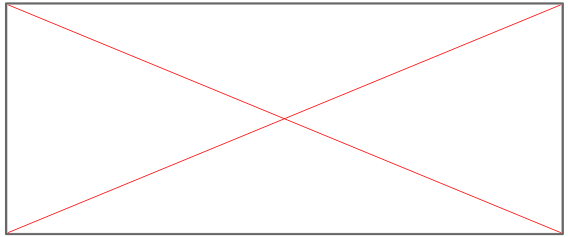
Rows instead of columns



Two variables in the same column



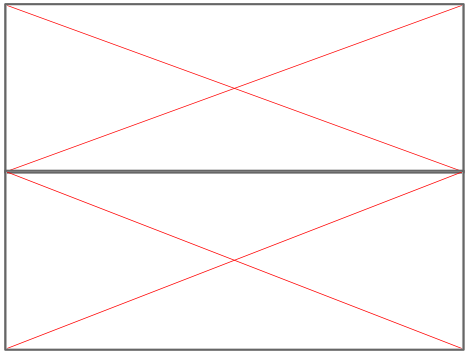
5



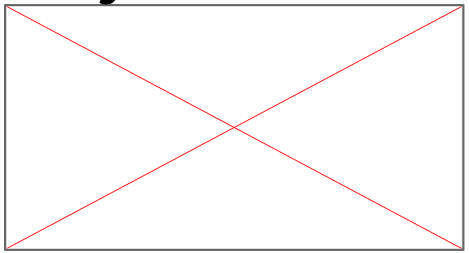
Exercise

Number of tuberculosis cases in Afghanistan, Brazil and China during 1999 and 2000

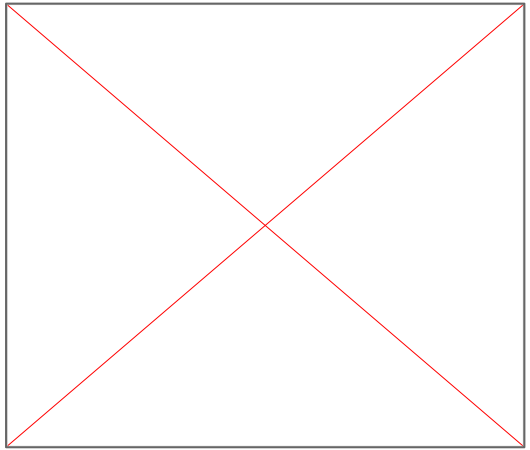
Columns instead of rows



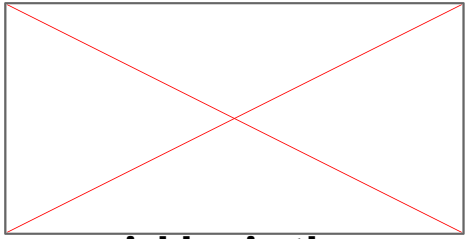
Tidy data



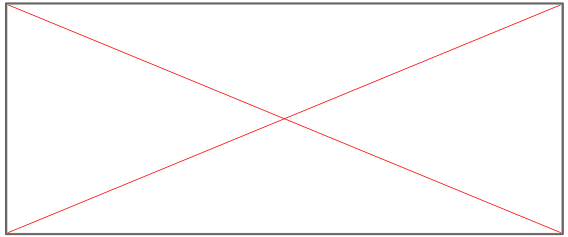
Rows instead of columns



Two variables in the same column



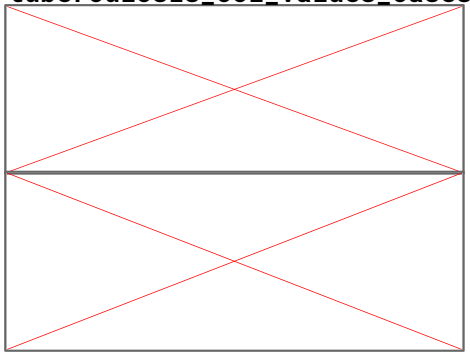
Two variables in the same column &
One variable in two columns



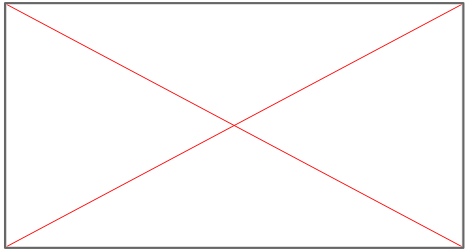
Exercise

Number of tuberculosis cases in Afghanistan, Brazil and China during 1999 and 2000

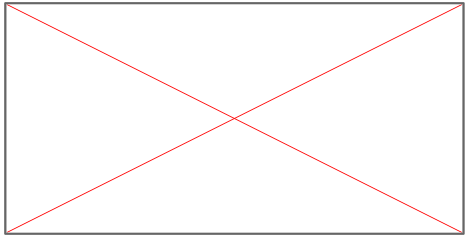
`tuberculosis_col_values_population.csv`
`tuberculosis_col_values_cases.csv`



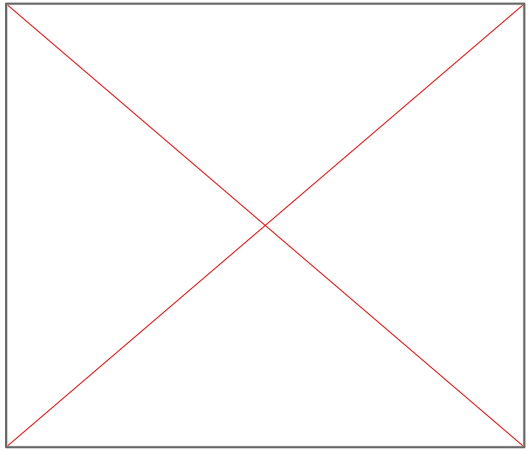
`tuberculosis_tidy.csv`



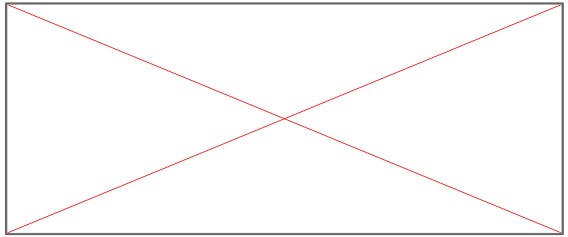
`tuberculosis_mult_values.csv`



`tuberculosis_mult_rows.csv`



`tuberculosis_mult_columns.csv`



Exercise
Column instead of rows

```r
# COLUMNS INSTEAD OF ROWS ------------------------------------------------
dt_pop <- tub$col_values_population
dt_cas <- tub$col_values_cases
# change the column names. The column names are stored in the firs row.
colnames(dt_pop) <- as.character(dt_pop[1,])
dt_pop <- dt_pop[-1]
colnames(dt_cas) <- as.character(dt_cas[1,])
dt_cas <- dt_cas[-1]
```

Exercise
Column instead of rows

```r
# COLUMNS INSTEAD OF ROWS ------------------------------------------------
dt_pop <- tub$col_values_population
dt_cas <- tub$col_values_cases
# change the column names. The column names are stored in the firs row.
colnames(dt_pop) <- as.character(dt_pop[1,])
dt_pop <- dt_pop[-1]
colnames(dt_cas) <- as.character(dt_cas[1,])
dt_cas <- dt_cas[-1]
```

country	1999	2000
Afghanistan	745	2666
Brazil	37737	80488
China	212258	213766

country	1999	2000
Afghanistan	19987071	20595360
Brazil	172006362	174504898
China	1272915272	1280428583

Exercise
Column instead of rows

```
##### tidyverse way
dt_pop %>%
  gather(year,value,'1999':'2000') %>%
    rename(population = value) -> dt_pop_melted
dt_cas %>%
  gather(year,value,'1999':'2000') %>%
  rename(cases = value)  -> dt_cas_melted


dt_tidy <- dt_pop_melted %>% inner_join(dt_cas_melted,by=c('country','year'))
dt_tidy <- dt_tidy %>% mutate(year = as.integer(year)) %>% arrange(country,year)
```

country	year	cases	population
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20595360
Brazil	1999	37737	172006362
Brazil	2000	80488	174504898
China	1999	212258	1272915272
China	2000	213766	1280428583

Exercise
Column instead of rows

```r
##### data.table way
dt_pop_melted <- melt(dt_pop,id.vars = 'country',variable.name = 'year',value.name = 'population')
dt_cas_melted <- melt(dt_cas,id.vars = 'country',variable.name = 'year',value.name = 'cases')
dt_tidy <- merge(dt_pop_melted,dt_cas_melted)
```

country	year	cases	population
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20595360
Brazil	1999	37737	172006362
Brazil	2000	80488	174504898
China	1999	212258	1272915272
China	2000	213766	1280428583

Exercise
Column instead of rows

```r
##### data.table way
dt_pop_melted <- melt(dt_pop,id.vars = 'country',variable.name = 'year',value.name = 'population')
dt_cas_melted <- melt(dt_cas,id.vars = 'country',variable.name = 'year',value.name = 'cases')
dt_tidy <- merge(dt_pop_melted,dt_cas_melted)
```

country	year	cases	population
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20595360
Brazil	1999	37737	172006362
Brazil	2000	80488	174504898
China	1999	212258	1272915272
China	2000	213766	1280428583

Exercise
Rows instead of columns

```r
# ROWS INSTEAD OF COLUMNS ----------------------------------------------
dt <- tub$mult_rows
dt
```

country	year	key	value
Afghanistan	1999	cases	745
Afghanistan	1999	population	19987071
Afghanistan	2000	cases	2666
Afghanistan	2000	population	20595360
Brazil	1999	cases	37737
Brazil	1999	population	172006362
Brazil	2000	cases	80488
Brazil	2000	population	174504898
China	1999	cases	212258
China	1999	population	1272915272
China	2000	cases	213766
China	2000	population	1280428583

Exercise
Two variables in the same column

```r
##### tidyverse way
dt_tidy <- separate(dt,col='rate',into = c('cases','population'),sep = '/',convert = T)


##### data.table way
dt_tidy <- copy(dt)
dt_tidy[,c('cases','population') := tstrsplit(rate,'/',type.convert = T,fixed=T)]
dt_tidy[,rate:=NULL]
```

country	year	cases	population
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20595360
Brazil	1999	37737	172006362
Brazil	2000	80488	174504898
China	1999	212258	1272915272
China	2000	213766	1280428583

Exercise
Rows instead of columns

```r
# ROWS INSTEAD OF COLUMNS ------------------------------------------------
dt <- tub$mult_rows
dt
```

Exercise
Rows instead of columns

```r
##### tidyverse way
dt_tidy <- spread(dt,key,value)



##### data.table way
dt_tidy <- dcast(dt,country+year~key)
```

country	year	cases	population
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20595360
Brazil	1999	37737	172006362
Brazil	2000	80488	174504898
China	1999	212258	1272915272
China	2000	213766	1280428583

Exercise
Two variables in the same column

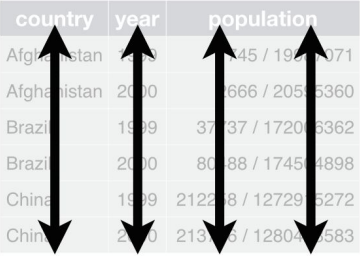
```r
# MULTIPLES VALUES ----------------------------------------------
dt  <- copy(tub$mult_values)
dt
```

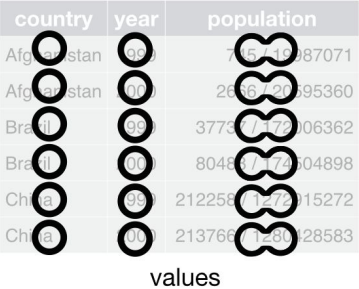
country	year	rate
Afghanistan	1999	745/19987071
Afghanistan	2000	2666/20595360
Brazil	1999	37737/172006362
Brazil	2000	80488/174504898
China	1999	212258/1272915272
China	2000	213766/1280428583

Exercise
Two variables in the same column

```r
# MULTIPLES VALUES ----------------------------------------------
dt  <- copy(tub$mult_values)
dt
```

country	year	population
Afghanistan	1999	745 / 19987071
Afghanistan	2000	2666 / 20595360
Brazil	1999	37737 / 172006362
Brazil	2000	80488 / 174504898
China	1999	212258 / 1272915272
China	2000	213766 / 1280428583





Exercise
One variable in two columns

```r
# MULTIPLES COLUMNS ----------------------------------------------
dt <- copy(tub$mult_columns)
```

country	century	year	rate
Afghanistan	19	99	745/19987071
Afghanistan	20	0	2666/20595360
Brazil	19	99	37737/172006362
Brazil	20	0	80488/174504898
China	19	99	212258/1272915272
China	20	0	213766/1280428583

Exercise
One variable in two columns

```
##### tidyverse way
dt %>% mutate(year = str_pad(year,width = 2,side='left',pad='0')) %>%
       unite('year',c('century','year'),sep='') %>%
       mutate(year=as.integer(year)) %>%
       separate('rate',c('cases','population'),convert = T) -> dt_tidy


##### data.table way
dt_tidy <- copy(dt)
dt_tidy[,year := paste0(century,str_pad(year,width = 2,side = 'left',pad = '0'))]
dt_tidy[,c('cases','population') := tstrsplit(rate,'/',type.convert = T,fixed=T)]
```

country	year	rate
Afghanistan	1999	745/19987071
Afghanistan	2000	2666/20595360
Brazil	1999	37737/172006362
Brazil	2000	80488/174504898
China	1999	212258/1272915272
China	2000	213766/1280428583