12-Visualization of factors

Soyoung Park

Pusan National University
Department of Statistics

```
library(tidyverse)
library(gridExtra)
```

Factors

- A special type of variable to indicate categories
- both labels and their order (i.e. numbers)
- By default text variables are stored in factors during input
- numeric categorical variables have to be converted to factors manually
- `factor` creates a new factor with specified labels

Load fbi data

```
fbi<-read.csv('fbi.csv')
head(fbi)
```

```
##       State Abb Year Population
## 1 Alabama  AL 1961    3302000 Murder.and.nonnegligent.Mansl
## 2 Alabama  AL 1962    3358000 Murder.and.nonnegligent.Mansl
## 3 Alabama  AL 1963    3347000 Murder.and.nonnegligent.Mansl
## 4 Alabama  AL 1964    3407000 Murder.and.nonnegligent.Mansl
## 5 Alabama  AL 1965    3462000 Murder.and.nonnegligent.Mansl
## 6 Alabama  AL 1966    3517000 Murder.and.nonnegligent.Mansl
##   Violent.crime
## 1          TRUE
## 2          TRUE
## 3          TRUE
## 4          TRUE
## 5          TRUE
```

Your turn

1. Inspect the fbi object. How many variables are there? Which type does each of the variables have?
2. Make a summary of Year
3. Make Year a factor variable: `fbi$Year <- factor(fbi$Year)`
4. Compare summary of Year to the previous result
5. Are there other variables that should be factors (or vice versa)?

Note: factors in boxplots

boxplots in ggplot2 only work properly if the x variable is a character variable or a factor:

```
twoyear <- dplyr::filter(fbi, Year %in% c(1961, 2016))

ggplot(data = twoyear, aes(x = as.integer(as.character(Year)),
  geom_boxplot() -> p1

ggplot(data = twoyear,aes(x = factor(Year),y = Count)) +
  geom_boxplot()->p2

grid.arrange(p1, p2, nrow = 1)
```

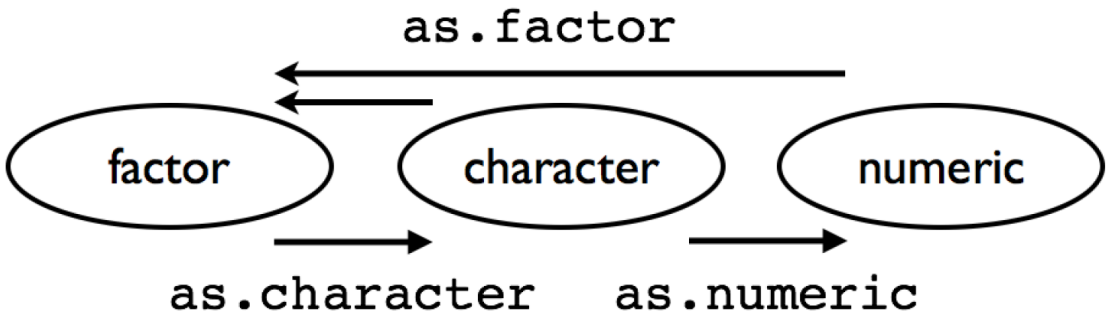
```
## Warning: Continuous x aesthetic -- did you forget aes(group

## Warning: Removed 50 rows containing non-finite values (stat
```

Data types: checking and casting

Checking for, and casting between types: - str, mode provide info on type - `is.XXX` (with XXX either factor, int, numeric, logical, character, ...) checks for specific type - as.XXX casts to specific type

Casting between types



Note: `as.numeric` applied to a factor retrieves order of labels, not labels, even if those could be interpreted as numbers. To get the labels of a factor as numbers, first cast to character and then to a number.

Levels of factor variables

- `levels(x)` shows us the levels of factor variable `x` in their current order
- factor variables often have to be re-ordered for ease of comparisons
- We can specify the order of the levels by explicitly listing them, see `help(factor)`
- We can make the order of the levels in one variable dependent on the summary statistic of another variable

Reordering factor levels - manual

```
levels(fbi$Type)
```

```
## NULL
```

```
#manually (extremely sensitive to typos):
levels(factor(fbi$Type, levels=c("Larceny.theft", "Burglary",
```

```
## [1] "Larceny.theft"
## [2] "Burglary"
## [3] "Motor.vehicle.theft"
## [4] "Robbery"
## [5] "Aggravated.assault"
## [6] "Legacy.rape"
## [7] "Rape"
## [8] "Murder.and.nonnegligent.Manslaughter"
```

```
fbi$Type<-factor(fbi$Type, levels=c("Larceny.theft", "Burglar
```

Reordering factor levels - using another variable

`reorder(factor, numbers, function)`

reorder levels in factor by values in numbers. Use function to summarize (average is used by default).

```
#missing values in numbers? make sure to use parameter na.rm=T
levels(reorder(fbi$Type, fbi$Count, na.rm=TRUE))
```

```
## [1] "Murder.and.nonnegligent.Manslaughter"
## [2] "Legacy.rape"
## [3] "Rape"
## [4] "Robbery"
## [5] "Aggravated.assault"
## [6] "Motor.vehicle.theft"
## [7] "Burglary"
## [8] "Larceny.theft"
```

Your turn

For this your turn use the `fbi` data.

- Introduce a rate of the number of reported offenses by population into the fbi data. You could use the Ames standard to make values comparable to a city of the size of Ames (population ~70,000).
- Plot boxplots of crime rates by different types of crime. How can you make axis text legible?
- Reorder the boxplots of crime rates, such that the boxplots are ordered by their medians.(Hint: use `reorder`)
- For one type of crime (subset!) plot boxplots of rates by state, reorder boxplots by median crime rates

Changing Levels' names

```
levels(fbi$Type)
```

```
## [1] "Larceny.theft"
## [2] "Burglary"
## [3] "Motor.vehicle.theft"
## [4] "Robbery"
## [5] "Aggravated.assault"
## [6] "Legacy.rape"
## [7] "Rape"
## [8] "Murder.and.nonnegligent.Manslaughter"
```

```
levels(fbi$Type)[8] <- "Murder"
```

Visualizing factors

- visualize factors directly: barcharts
- use factors in aesthetics (colour, fill, shape) or for facetting
- always make sure that the order in factors is sensible!

Note: factors for fill color

- In area plots (histograms and barcharts for now) use aesthetic fill for showing colored areas.
- Only factor variables can be mapped to fill

Example:

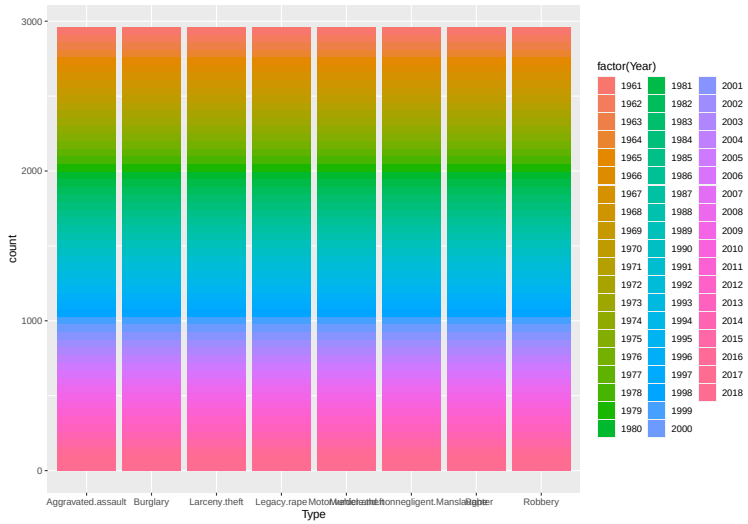
```
#reload data
fbi<-read.csv('fbi.csv')
str(fbi)
```

```
## 'data.frame':    23672 obs. of  7 variables:
##  $ State         : chr  "Alabama" "Alabama" "Alabama" "Alaba
##  $ Abb           : chr  "AL" "AL" "AL" "AL" ...
##  $ Year          : int  1961 1962 1963 1964 1965 1966 1967 1
##  $ Population    : int  3302000 3358000 3347000 3407000 3462
##  $ Type          : chr  "Murder.and.nonnegligent.Manslaughte
##  $ Count         : int  427 316 340 316 395 384 415 421 485
##  $ Violent.crime: logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
```

```
ggplot(fbi, aes(x = Type, fill=Year)) + geom_bar() # no color
```

Example:

```
ggplot(fbi, aes(x = Type, fill=factor(Year))) + geom_bar()
```



Example: Survival on the titanic

The object `titanic` is a table of a break down of survival of passengers and crew on board the titanic by gender and age.

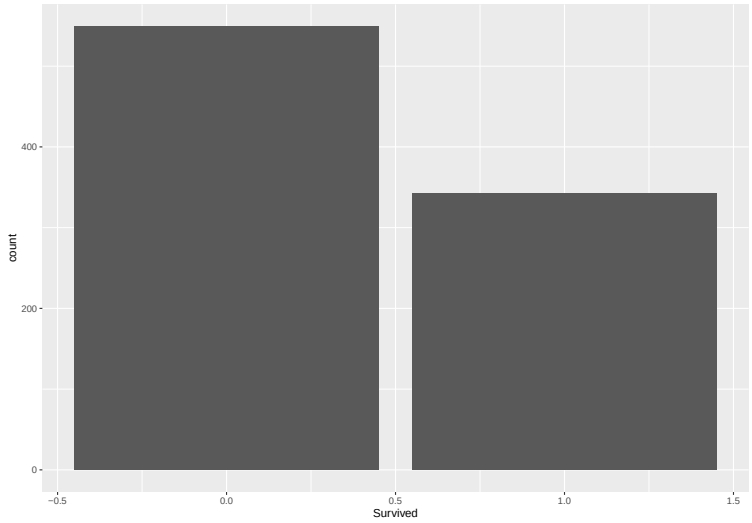
```
#install.packages("titanic")
library(titanic)
head(titanic_train)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                  Name    Se
## 1                             Braund, Mr. Owen Harris    mal
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) fema
```

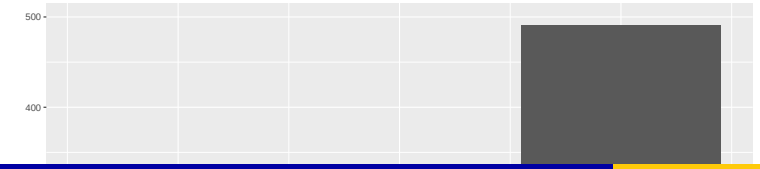
Barcharts of all variables

```
#?titanic_train

ggplot(titanic_train, aes(x = Survived)) +geom_bar()
```



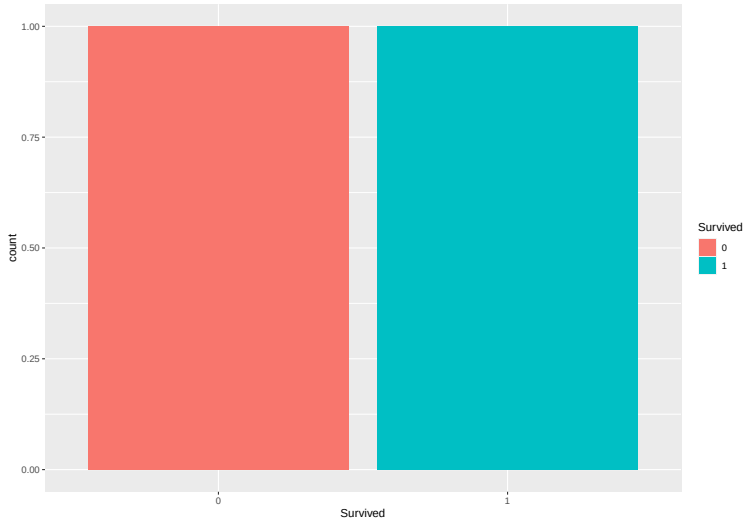

```
ggplot(titanic_train, aes(x = Pclass))+geom_bar()
```



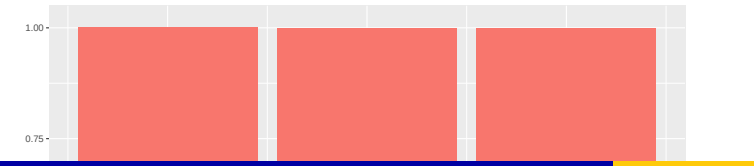
position="fill"

```
titanic_train$Survived<-factor(titanic_train$Survived)

ggplot(titanic_train, aes(x = Survived, fill=Survived)) + geom
```




```
ggplot(titanic_train, aes(x = Pclass, fill=Survived)) + geom_b
```



Two and more factor variables

- besides facetting and position, use mosaic plots
- there are extension packages for `ggplot2`, e.g. ggmosaic

Mosaicplots

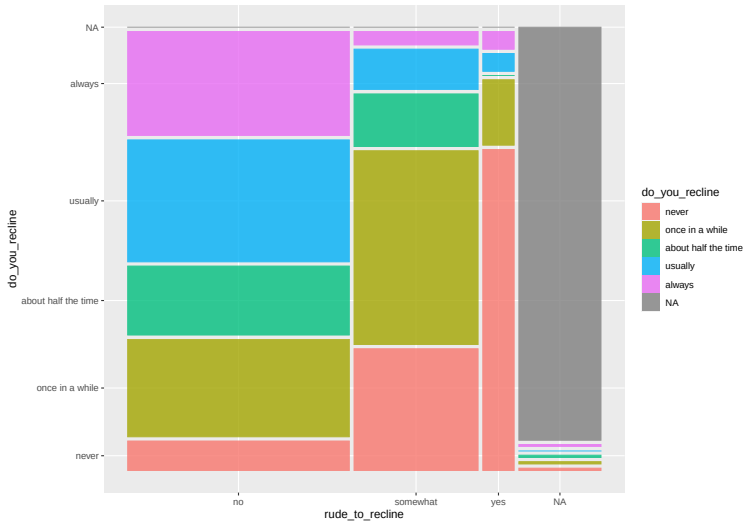
```
#install.packages('ggmosaic')
library(ggmosaic)
```

```
## Warning: package 'ggmosaic' was built under R version 4.0.5
```

```
titanic_train %>%
  mutate(Age2=ifelse(Age>20, 'Adult', 'Child')) %>%
 ggplot() +
 geom_mosaic(aes(x = product(Sex), fill=Survived, weight=1)) +
 facet_grid(Age2~Pclass)
```

Mosaicplots 2

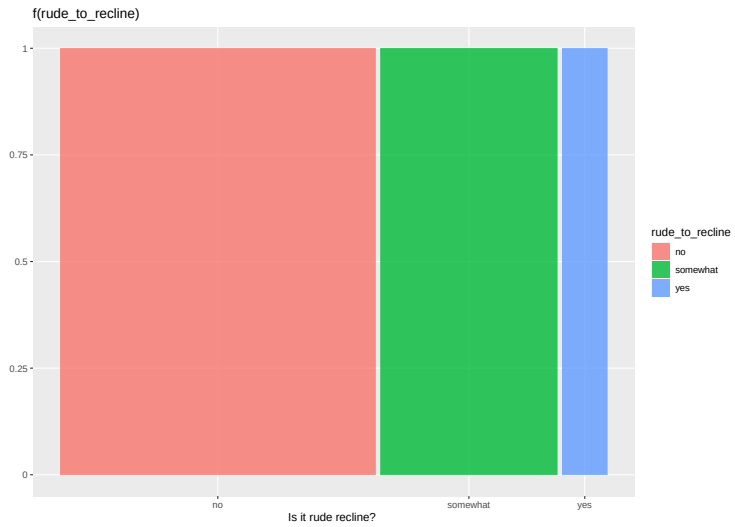
```
ggplot(data = fly) +
  geom_mosaic(aes(x = product(rude_to_recline), fill=do_you_re
```



Mosaicplots 3

$1 \sim X$

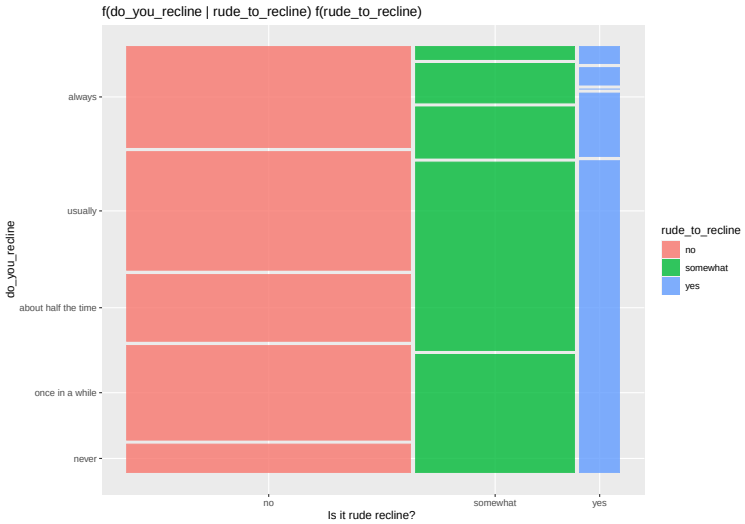
```
ggplot(data = fly) +
  geom_mosaic(aes(x = product(rude_to_recline), fill=rude_to_
  labs(x="Is it rude recline? ", title='f(rude_to_recline)')
```



Mosaicplots 4

$1 \sim Y+X$

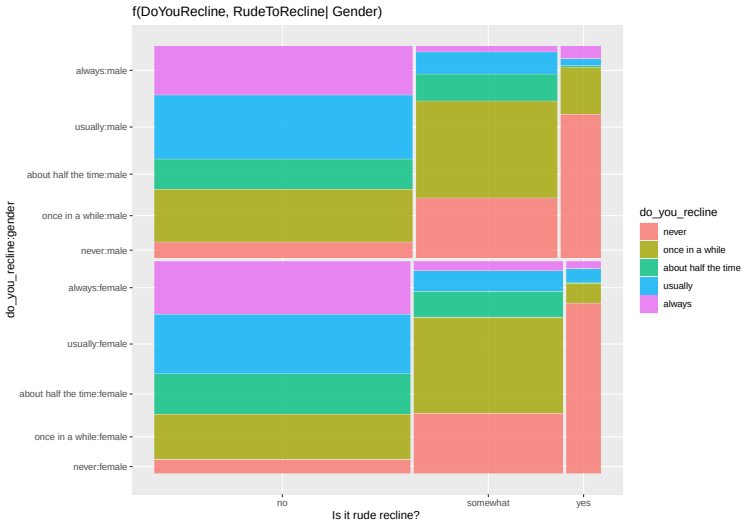
```
ggplot(data = fly) +
    geom_mosaic(aes(x = product(do_you_recline, rude_to_recline
  labs(x = "Is it rude recline? ", title='f(do_you_recline | r
```



Mosaicplots 5

$1 \sim X+Y/Z$

```
ggplot(data = fly) +
    geom_mosaic(aes(x = product(do_you_recline, rude_to_recline
```



Mosaicplots

More information about mosaic plot:

Here

Your turn

Use titanic_train data for following questions.

- Draw a barchart of Gender. Interpret.
- Map survival to fill color in the barchart of Gender. Interpret.
- In the previous barchart change the position parameter to "fill".Interpret.
- Read up on the position parameter in ?geom_bar. Try out other options for position.