1. Initial data set up

1.1 Summary of data:

head()

names()

str()

summary()

dim()

1.2 Reshaping data

* Melting dataset – Wide to Long format

library(reshape2)

Malidb = melt(Mali\_data, id = "Country")

* Transpose

t(dataset)

* Subset

male\_pf <- subset(pf, pf$gender == "male")

- tbl\_df

* Long to Wide format

pf.fc\_by\_age\_gender.wide <- **dcast**(pf.fc\_by\_age\_gender, age ~ gender, value.var = "median\_friend\_count")

* Categorical vs. continuous variables

c1 <- cut(clinical.trial$age, breaks = 4) > table(c1)

specify breaks using sequence seq()

If specifying values: pf$year\_joined.bucket = cut(pf$year\_joined, c(2004, 2009, 2011, 2012, 2014))

* Transforming data/ adding column

yo <- transform(yo, all.purchases = yo$strawberry+yo$blueberry+ yo$pina.colada+ yo$plain + yo$mixed.berry)

**Define a new transformation**

cuberoot\_trans = function() trans\_new('cuberoot', transform = function(x) x^(1/3), inverse = function(x) x^3)

- Sampling data

sample.ids <- sample(levels(yo$id,15))

ggplot(aes(x = time, y = price), data = subset(yo, id %in%sample.ids)) +

facet\_wrap(~id) + geom\_line() +

geom\_point(aes(size = all.purchases), pch =1)

Sample data, also remove a few columns if more than 10 columns

Syntax: 10000 rows

diamond\_samp <- diamonds[sample(1:length(diamonds$price), 10000),]

1.3 Importing Files

- Excel

library(readxl)

?read\_excel

gapminder <- read\_excel("~/Documents/Files/gapminder.xlsx", sheet = "Data", col\_names = TRUE)

names(gapminder)

csv

yo <- read.csv("yogurt.csv") ``

changing data type

yo$id <- factor(yo$id)

library(ggplot2)

data(diamonds)

1.4 Statistics/ Operations

table

count

max value:

count\_db <- na.omit(count(birthdays, vars = "datemonth"))

count\_db[which.max(count\_db$freq),]

Rounding

Round

Ceiling

Floor

Dplyr package

First argument is a dataframe. Second argument tells you what to do with the dataframe. This command always returns a dataframe.

* SUMMARIZE
* FILTER – select rows that match some criterion {kind of functions like subset}
* SELECT – columns
* ARRANGE – order data (Desc, Ascen)
* MUTATE – transform data (multiply all values in columns etc., add new columns)
* SUMMAIRIZE – min, median, max, quantile, n , ndistinct, sum, mean, stdev, var
* GROUP\_BY
* MIN\_RANK/ DENSE\_RANK/ ROW\_NUMBER
* LAG (lag + order\_by)
* Left\_JOIN, Inner\_Join, Semi\_Join, Anti\_join,
* Do
* Explain

Pipe Operator:

%>%

example:

age\_groups\_by\_months = group\_by(pf, age\_with\_months)

pf.fc\_by\_age\_months <- summarise(age\_groups\_by\_months, friend\_count\_mean = mean(friend\_count), friend\_count\_median = median(friend\_count), n = n())

pf.fc\_by\_age\_months <- arrange(pf.fc\_by\_age\_months, desc(age\_with\_months))

browse\_vignettes

1.5 Dates

strptime(birthdays$dates, format = "%d/%m/%Y")

birthdays$month = strftime(birthdays$dates, "%m")

1.6 Setting working directory

getwd()

setwd("~/Documents/Bakbak/Uninstalls")

2. Plots

2.1 qplot ( x = , data = , xlim = c( , ) , binwidth = ) +

scale\_x\_continuous(trans = …, limits = c(a,b), breaks = seq(start point, endpoint, jump) or c(actual points)

2.2 ggplot2(x = , data = ) +

scale\_x\_discrete(breaks = : )

or

2.3 ggplot(aes(x = ), data = ) +

geom\_histogram(binwidth = 1) +

scale\_x\_continuous(breaks = : )

ggplot(aes(x = age, y = friend\_count), data = pf) + geom\_jitter(alpha = 1/20) +

xlim(13,90) + ylim(1,5000) + coord\_trans(y = "sqrt")

also,

ggplot(aes(x = age, y = friend\_count), data = pf) + geom\_point(alpha = 1/20, position = position\_jitter(h = 0) )+

xlim(13,90) + coord\_trans(y = "sqrt")

also,

ylim(0, quantile(pf$likes\_received, 0.95))

or scale\_y\_continuous(lim = c(0, quantile(pf$likes\_received, 0.95))

2.4 facets

facet\_grid (vertical ~ horizontal #for blank put dot#)

facet\_wrap (~ variable, ncol = )

scales = “free” for different axes

or free\_x or free\_y

2.5 omitting na values

subset(dataset, !is.na(colname))

data = na.omit(dbname)

2.6 Adding color

qplot(x = , data = , color = I(“color”#for boundary color#), fill = I(color))

2. 7 adding lables

xlab = “” ylab = “”

2.8 Arranging multiple plots in a single window

library(gridExtra)

grid.arrange(“p1”, “p2, …., ncols = 2/3/4…)

2.9 Log scale or Sqrt scale

Only on ggplot

+ scale\_y\_sqrt()

+ scale\_y\_log10

2.10 frequency polygon

geom = “freqpoly”

Add color = “variable” for breaking by variable

2.11 To plot on proportions on y axis

Add y - ..count../sum(..count..)

2.12 boxplots

geom = “boxplot”

x and y interchanged

coord\_cartesian(ylim = c(0,..) for adjusting plot height

2.13 ggplot geomline

ggplot(aes(y= friend\_count\_mean, x = age), data = pf.fc\_by\_age) +

geom\_line(size = 0.5, arrow = arrow(ends = "both", type = "closed"), colour = "yellow")

Adding summaries to raw data:

+ geom\_line(stat = ‘summary’, fun.y = mean)

+ geom\_line(stat = ‘summary’, fun.y = quantile, probs = 0.1)

+ geom\_line(stat = ‘summary’, fun.y = quantile, probs = 0.9 + linetype = 2 + color = ‘blue)

or

+ stat\_summary(fun.y = "mean", colour = "red", size = 2, geom = "point")

2.14 Modulus operator (repeat a period)

ggplot(aes(x=(Month%%12),y=Temp),data=Mitchell)+

geom\_point()

2.15 To smooth a plot

geom\_smooth()

2.16 Bar chart

ggplot(aes(x = clarity, y = mean\_price), data = diamonds\_mp\_by\_clarity) + geom\_bar(stat = "identity")

Arranging in ascending/descending order – use reorder

reorder\_size <- function(x) {factor(x, levels = names(sort(table(x), decreasing = TRUE)))}

2.17 Multivariable plots

ggplot(aes(x = age, y = median\_friend\_count), data = pf.fc\_by\_age\_gender) +

+ geom\_line(aes(color = gender))

ggplot(aes(y = friend\_count, x = age), data = subset(pf, !is.na(year\_joined.bucket))) +

geom\_line(aes(color = year\_joined.bucket), stat = 'summary', fun.y = median)

2.18 Scatterplot matrix

install.packages('GGally')

ggpairs

]

ggpairs(diamond\_samp, lower = list(continuous = wrap("points", shape = I('.'))), upper = list(combo = wrap("box", outlier.shape = I('.'))))

heatmaps

3. Statistics

3.1 table(database$column)

3.2 by( x, by\_variable, f(x))

also using dplyr add as column

melt\_wines <- quality9wines %>% group\_by(variable) %>% mutate(value.new = mean(value))

3.3 True and False for a variable

summary(variable > 0 #or any condition)

3.4 length()

Gives the total number of rows

3.5 sum (variable = “value”)

3.6 corelation coefficient

cor.test(x = pf$age, y = pf$friend\_count, data = pf, method = "pearson")

with(pf,cor.test(age, friend\_count, method = "pearson"))

Adding correlation line to a scatterplot:

+ geom\_smooth(method = ‘lm’, = “red”)

4. Functions

<https://www.youtube.com/watch?v=29TdKkUmcA4&index=18&list=PLOU2XLYxmsIK9qQfztXeybpHvru-TrqAP>