# Last update: 07/28/2022

cd

cd Documents

mkdir data\_viz

cd data\_viz

wget <https://github.com/clauswilke/dviz.supp/archive/refs/heads/master.zip>

gunzip master.zip

# ==== R のコンソールで下記を実行 =============

#ライブラリの読み込み

library(readr)

library(here)

library(lubridate)

library(dplyr)

library(tidyr)

library(forcats)

library(patchwork)

library(ggplot2)

library(cowplot)

#pallette の定義

palette\_OkabeIto <- c("#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7", "#999999")

theme\_dviz\_grid <- function(font\_size = 14, font\_family = "") {

color = "grey90"

line\_size = 0.5

# Starts with theme\_cowplot and then modify some parts

cowplot::theme\_cowplot(font\_size = font\_size, font\_family = font\_family) %+replace%

theme(

# make horizontal grid lines

panel.grid.major = element\_line(colour = color,

size = line\_size),

# adjust axis tickmarks

axis.ticks = element\_line(colour = color, size = line\_size),

# no x or y axis lines

axis.line.x = element\_blank(),

axis.line.y = element\_blank()

)

}

#テストデータの読み込み

ncdc\_normals <- read\_fwf(here::here("Desktop","data\_viz","dviz.supp-master","data-raw", "ncdc\_normals", "dly-tavg-normal.txt"),fwf\_positions(c(1, 13, 19 + 7\*(0:30)),c(11, 14, 24 + 7\*(0:30)),c("station\_id", "month", paste0("day", 1:31)))) %>%

gather(day, value, -station\_id, -month) %>%

extract(day, "day", regex = "([[:digit:]]+)") %>%

extract(value, c("temperature", "flag"), regex = "(-\*[[:digit:]]+)([[:alpha:]]\*)") %>%

mutate(temperature = ifelse(temperature == "-8888", NA, as.numeric(temperature))/10) %>%

na.omit() %>% mutate(date = ymd(paste("0000", month, day, sep = '-'))) %>%

arrange(station\_id, as.numeric(month), as.numeric(day))

#テストデータの整形

temps\_long <- filter(ncdc\_normals,

station\_id %in% c(

"USW00014819", # Chicago, IL 60638

#"USC00516128", # Honolulu, HI 96813

#"USW00027502", # Barrow, AK 99723, coldest point in the US

"USC00042319", # Death Valley, CA 92328 hottest point in the US

"USW00093107", # San Diego, CA 92145

#"USC00427606" # Salt Lake City, UT 84103

"USW00012918" # Houston, TX 77061

)) %>%

mutate(location = fct\_recode(factor(station\_id),

"Chicago" = "USW00014819",

#"Honolulu, HI" = "USC00516128",

#"Barrow, AK" = "USW00027502",

"Death Valley" = "USC00042319",

"San Diego" = "USW00093107",

#"Salt Lake City, UT" = "USC00427606",

"Houston" = "USW00012918")) %>%

mutate(location = factor(location, levels = c("Death Valley", "Houston", "San Diego", "Chicago")))

#Line plotの描画

ggplot(temps\_long, aes(x = date, y = temperature, color = location)) +

geom\_line(size = 1) +

scale\_x\_date(name = "month", limits = c(ymd("0000-01-01"), ymd("0001-01-04")),

breaks = c(ymd("0000-01-01"), ymd("0000-04-01"), ymd("0000-07-01"),

ymd("0000-10-01"), ymd("0001-01-01")),

labels = c("Jan", "Apr", "Jul", "Oct", "Jan"), expand = c(1/366, 0)) +

scale\_y\_continuous(limits = c(19.9, 107),

breaks = seq(20, 100, by = 20),

name = "temperature (°F)") +

theme\_dviz\_grid() +scale\_color\_manual(values=palette\_OkabeIto)+

theme(legend.title.align = 0.5)

#ヒートマップの準備

month\_names <- c("01" = "Jan", "02" = "Feb", "03" = "Mar", "04" = "Apr", "05" = "May", "06" = "Jun",

"07" = "Jul", "08" = "Aug", "09" = "Sep", "10" = "Oct", "11" = "Nov", "12" = "Dec")

mean\_temps <- temps\_long %>%

group\_by(location, month) %>%

summarize(mean = mean(temperature)) %>%

ungroup() %>%

mutate(month = month\_names[month]) %>%

mutate(month = factor(month, levels = unname(month\_names)))

#ヒートマップ描画

p <- ggplot(mean\_temps, aes(x = month, y = location, fill = mean)) +

geom\_tile(width = .95, height = 0.95) +

scale\_fill\_viridis\_c(option = "B", begin = 0.15, end = 0.98,

name = "temperature (°F)") +

scale\_y\_discrete(name = NULL) +

coord\_fixed(expand = FALSE) +

theme\_bw() +

theme(axis.line = element\_blank(),

axis.ticks = element\_blank(),

#axis.text.y = element\_text(size = 14),

legend.title = element\_text(size = 12)

)

ggdraw(p)

#散布図

p\_mtcars <- ggplot(mtcars, aes(disp, mpg, fill = hp, shape = factor(cyl), size = wt)) +

geom\_point(color = "white") +

scale\_shape\_manual(values = c(23, 24, 21), name = "cylinders") +

scale\_fill\_viridis\_c()+

xlab("displacement (cu. in.)") +

ylab("fuel efficiency (mpg)") +

guides(

shape = guide\_legend(override.aes = list(size = 4, fill = "#329D84")),

size = guide\_legend(override.aes = list(shape = 21, fill = "#329D84"),

title = "weight (1000 lbs)")

) + background\_grid() +

theme(

#legend.title = element\_text(size = 12),

legend.box.background = element\_rect(fill = "white", color = "white"),

legend.position = "top",

legend.direction = "vertical",

legend.justification = "center",

legend.box.margin = margin(7, 7, 7, 7)

)

ggdraw() +

draw\_plot(p\_mtcars) +theme\_bw()

#デフォルトの描画

P = ggplot(temps\_long, aes(x = date, y = temperature, color = location,label=location)) +

geom\_line(size = 1) +

scale\_x\_date(name = "month", limits = c(ymd("0000-01-01"), ymd("0001-01-04")),

breaks = c(ymd("0000-01-01"), ymd("0000-04-01"), ymd("0000-07-01"),

ymd("0000-10-01"), ymd("0001-01-01")),

labels = c("Jan", "Apr", "Jul", "Oct", "Jan"), expand = c(1/366, 0)) +

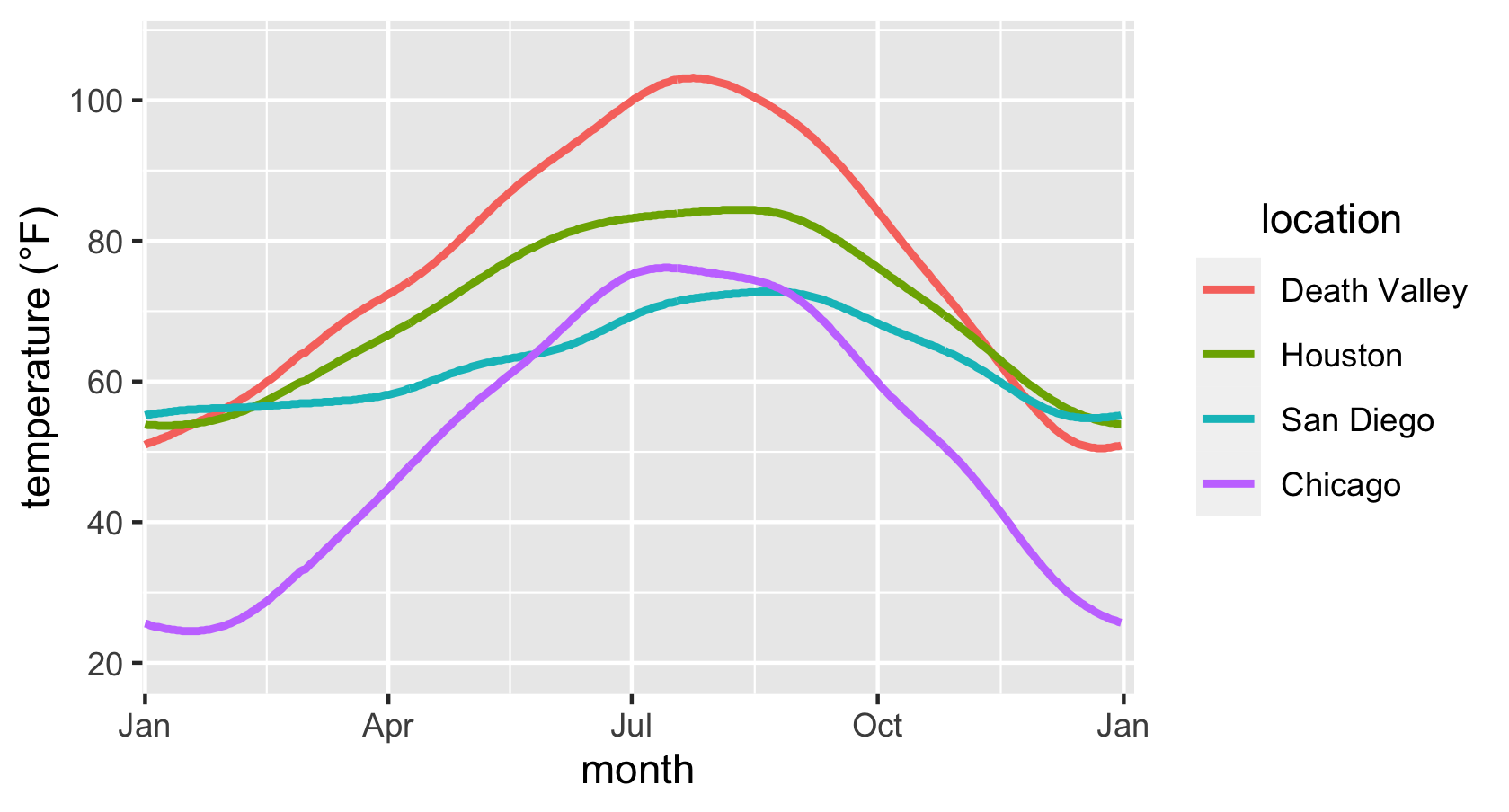
scale\_y\_continuous(limits = c(19.9, 107),

breaks = seq(20, 100, by = 20),

name = "temperature (°F)") +

theme(legend.title.align = 0.5)

P



#識別しやすい背景

P+ theme\_bw() + theme( legend.key = element\_blank(),

strip.background = element\_rect(   
 color= "#FFFFFF",fill="#FFFFFF"),

panel.grid.minor = element\_blank(),

panel.grid.major = element\_blank(),

panel.background = element\_blank()

)

#枠線・tickの太さを統一

P+ theme\_bw() + theme(

panel.border = element\_rect(size=1.0),

axis.ticks = element\_line(color = "#000000", size = 0.5),

)

#テキストサイズ・フォント・色を統一

P+ theme\_bw() + theme(

axis.text.x = element\_text(color="#000000",size=9.5),

axis.text.y = element\_text(color="#000000",size=9.5),

strip.text.y = element\_text(size = 9.5, color = "#000000", angle = 270),

strip.text.x = element\_text(size = 9.5, color = "#000000"),

axis.title=element\_text(size=9.5, color = "#000000")

)

#色を指定

cbPalette <- c("#999999", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7")

group.colors <- c( "Death Valley" = "#E69F00",

"Houston" = "#56B4E9",

"San Diego" = "#009E73",

"Chicago" = "#F0E442"

)

P +scale\_colour\_manual(values=group.colors)

#全部モリモリ

P = ggplot(temps\_long, aes(x = date, y = temperature, color =location)) +

geom\_line(size = 1) +

scale\_x\_date(name = "Month", limits = c(ymd("0000-01-01"), ymd("0001-01-04")),

breaks = c(ymd("0000-01-01"), ymd("0000-04-01"), ymd("0000-07-01"),

ymd("0000-10-01"), ymd("0001-01-01")),

labels = c("Jan", "Apr", "Jul", "Oct", "Jan"), expand = c(1/366, 0)) +

scale\_y\_continuous(limits = c(19.9, 107),

breaks = seq(20, 100, by = 20),

name = "Temperature (°F)") +

theme\_bw() +   
 theme( panel.spacing = unit(2, "lines"),

legend.title.align = 0.5,

legend.key = element\_blank(),

strip.background = element\_rect(  
　　　　　　　 color="#FFFFFF",fill="#FFFFFF"),

panel.border = element\_rect(size=1.0),

panel.grid.minor = element\_blank(),

panel.grid.major = element\_blank(),

panel.background = element\_blank(),

aspect.ratio=0.8,

axis.ticks = element\_line(color = "#000000", size = 0.5),

axis.text.x = element\_text(color="#000000",size=9.5),

axis.text.y = element\_text(color="#000000",size=9.5),

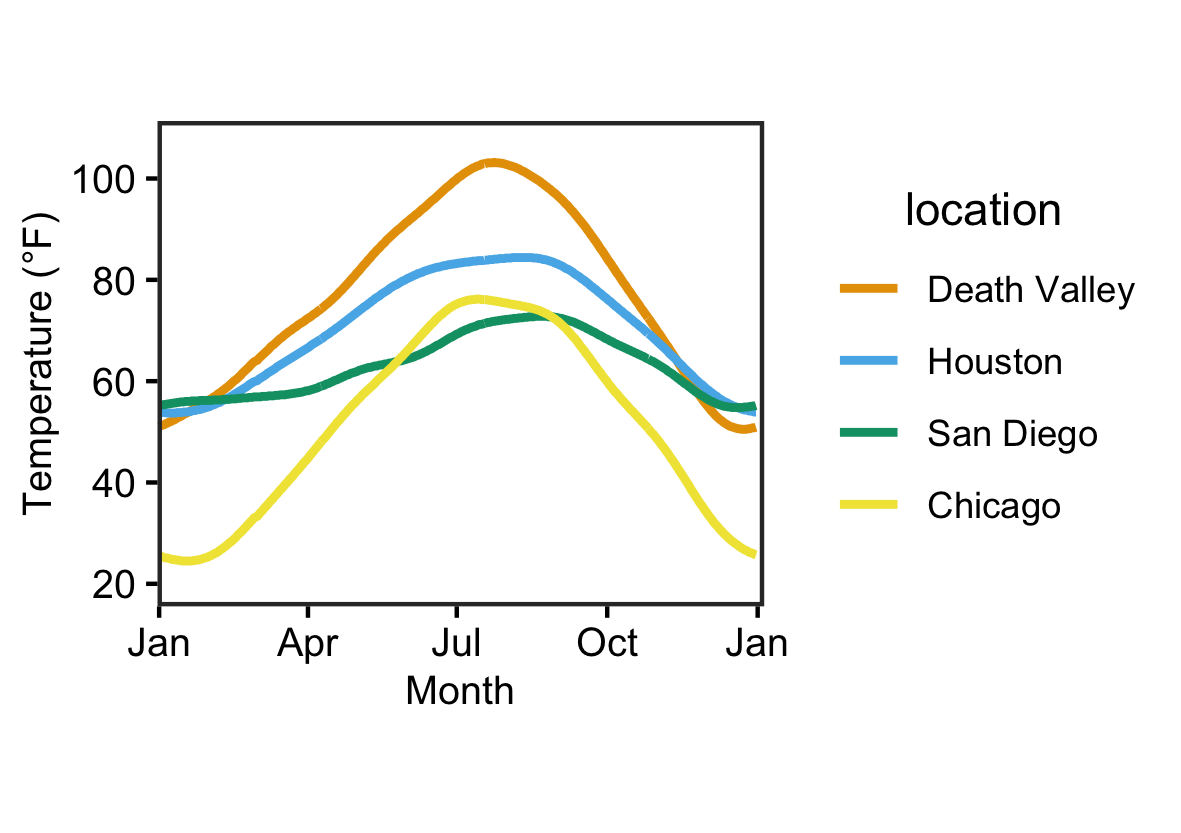
strip.text.y = element\_text(size = 9.5, color = "#000000", angle = 270),

strip.text.x = element\_text(size = 9.5, color = "#000000"),

axis.title=element\_text(size=9.5, color = "#000000")

)+  
scale\_colour\_manual(values=group.colors)

P



#ヒートマップの準備

month\_names <- c("01" = "Jan", "02" = "Feb", "03" = "Mar", "04" = "Apr", "05" = "May", "06" = "Jun",

"07" = "Jul", "08" = "Aug", "09" = "Sep", "10" = "Oct", "11" = "Nov", "12" = "Dec")

mean\_temps <- temps\_long %>%

group\_by(location, month) %>%

summarize(mean = mean(temperature)) %>%

ungroup() %>%

mutate(month = month\_names[month]) %>%

mutate(month = factor(month, levels = unname(month\_names)))

#ヒートマップ描画

p <- ggplot(mean\_temps, aes(x = month, y = location, fill = mean)) +

geom\_tile(width = .95, height = 0.95) +

scale\_fill\_viridis\_c(option = "B", begin = 0.15, end = 0.98,

name = "temperature (°F)") +

scale\_y\_discrete(name = NULL) +

coord\_fixed(expand = FALSE) +

theme\_bw() +

theme(axis.line = element\_blank(),

axis.ticks = element\_blank(),

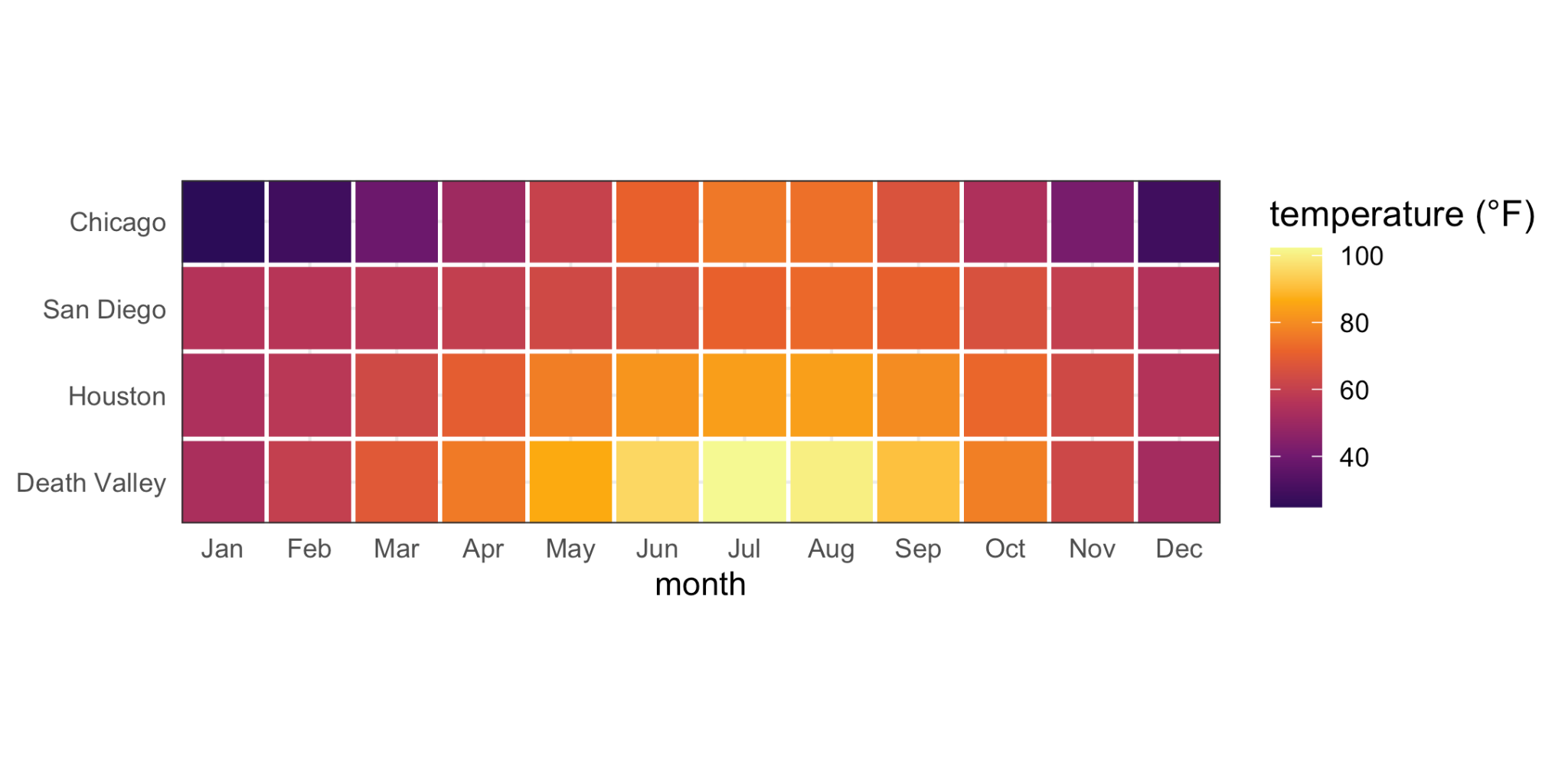
#axis.text.y = element\_text(size = 14),

legend.title = element\_text(size = 12)

)

p

#デフォルト



#全部モリモリ

heatmap <- ggplot(mean\_temps, aes(x = month, y = location, fill = mean)) +

geom\_tile(width = 1, height = 1) +

scale\_fill\_viridis\_c(option = "B", begin = 0.15, end = 0.98,

name = "Temperature (°F)") +

scale\_y\_discrete(expand=c(0,0)) +  
 scale\_x\_discrete(expand=c(0,0))+

coord\_fixed(expand = FALSE) +

theme\_bw() +

theme(legend.position='right',  
 legend.key = element\_blank(),   
 strip.background = element\_rect(color="#FFFFFF",fill="#FFFFFF"),  
 panel.border = element\_rect(size=1.0),  
 panel.grid.minor = element\_blank(),  
 panel.grid.major = element\_blank(),  
 panel.background =element\_blank(),  
 axis.ticks = element\_blank(),  
 axis.text.x = element\_text(color="#000000",size=9.5, angle = 90),

axis.text.y = element\_text(color="#000000",size=9.5),

strip.text.y = element\_text(size = 9.5, color = "#000000", angle = 270),

strip.text.x = element\_text(size = 9.5, color = "#000000"),

)

heatmap

#デフォルト

p\_mtcars <- ggplot(mtcars, aes(disp, mpg, fill = hp, shape = factor(cyl), size = wt)) +

geom\_point(color = "white") +

scale\_shape\_manual(values = c(23, 24, 21), name = "cylinders") +

scale\_fill\_viridis\_c()+

xlab("Displacement (cu. in.)") +

ylab("Fuel efficiency (mpg)") +

guides(

shape = guide\_legend(override.aes = list(size = 4, fill = "#329D84")),

size = guide\_legend(override.aes = list(shape = 21, fill = "#329D84"),

title = "weight (1000 lbs)")

) + background\_grid() +

theme(

#legend.title = element\_text(size = 12),

legend.box.background = element\_rect(fill = "white", color = "white"),

legend.position = "top",

legend.direction = "vertical",

legend.justification = "center",

legend.box.margin = margin(7, 7, 7, 7)

)

p\_mtcars

＃設定

p\_mtcars +

theme(legend.position='none',

legend.key = element\_blank(),

strip.background = element\_rect(color="#FFFFFF",fill="#FFFFFF"),

panel.grid.minor = element\_blank(),

panel.grid.major = element\_blank(),

panel.spacing = unit(0.75, "lines"),

panel.background = element\_blank(),

aspect.ratio=1.0,

axis.ticks =element\_line(color = "#000000", size = 0.5),

axis.text.x = element\_text(color="#000000",size=9.5),

axis.text.y = element\_text(color="#000000",size=9.5) ,

panel.border = element\_rect(color="#000000", fill =NA,size=1),

axis.title=element\_text(size=9.5))

#X/Y軸の設定

scatter = p\_mtcars +

theme(legend.position='bottom',

legend.key = element\_blank(),

strip.background = element\_rect(color="#FFFFFF",fill="#FFFFFF"),

panel.grid.minor = element\_blank(),

panel.grid.major = element\_blank(),

panel.spacing = unit(0.75, "lines"),

panel.background = element\_blank(),

aspect.ratio=1.0,

axis.ticks =element\_line(color = "#000000", size = 0.5),

axis.text.x = element\_text(color="#000000",size=9.5),

axis.text.y = element\_text(color="#000000",size=9.5) ,

panel.border = element\_rect(color="#000000", fill =NA,size=1),

axis.title=element\_text(size=9.5))+

scale\_y\_continuous(expand=c(0,0),limits=c(-1,41)) +  
 scale\_x\_continuous(expand=c(0,0),limits=c(-10,510))

library(grid)

library(gridExtra)

library(egg)

#サイズの指定

p <-set\_panel\_size(P,width = unit(5, "cm"),height = unit(3, "cm"))

　　#このサイズと最後に保存する際のサイズは違うので注意。

grid.newpage()

grid.draw(p)

#ファイルに保存するサイズの指定

ggsave(plot=p,"line\_size\_adjusted.pdf" ,width=unit(4,"cm"),height=unit(2,"cm"))

　　#この保存サイズは大まかに　1ページ幅=14cm, 半ページ幅=7cm くらいです。

library(ggrepel)

#テストデータの読み込み

ncdc\_normals <- read\_fwf(here::here("Desktop","data\_viz","dviz.supp-master","data-raw", "ncdc\_normals", "dly-tavg-normal.txt"),fwf\_positions(c(1, 13, 19 + 7\*(0:30)),c(11, 14, 24 + 7\*(0:30)),c("station\_id", "month", paste0("day", 1:31)))) %>%

gather(day, value, -station\_id, -month) %>%

extract(day, "day", regex = "([[:digit:]]+)") %>%

extract(value, c("temperature", "flag"), regex = "(-\*[[:digit:]]+)([[:alpha:]]\*)") %>%

mutate(temperature = ifelse(temperature == "-8888", NA, as.numeric(temperature))/10) %>%

na.omit() %>% mutate(date = ymd(paste("0000", month, day, sep = '-'))) %>%

arrange(station\_id, as.numeric(month), as.numeric(day))

#データの用意

temps\_wide <- filter(ncdc\_normals,

station\_id %in% c(

"USW00014819", # Chicago, IL 60638

"USC00516128", # Honolulu, HI 96813

"USW00027502", # Barrow, AK 99723, coldest point in the US

"USC00042319", # Death Valley, CA 92328 hottest point in the US

"USW00093107", # San Diego, CA 92145

"USW00012918", # Houston, TX 77061

"USC00427606" # Salt Lake City, UT 84103

)) %>%

mutate(location = fct\_recode(factor(station\_id),

"Chicago" = "USW00014819",

"Honolulu" = "USC00516128",

"Barrow, AK" = "USW00027502",

"Death Valley" = "USC00042319",

"San Diego" = "USW00093107",

"Houston" = "USW00012918",

"Salt Lake City, UT" = "USC00427606")) %>%

select(-station\_id, -flag) %>%

spread(location, temperature) %>%

arrange(date)

temps\_wide\_label <- mutate(

temps\_wide,

label = ifelse(

date %in% c(ymd("0000-01-01"), ymd("0000-04-01"), ymd("0000-07-01"), ymd("0000-10-01")),

format(date, "%b 1st"),

""

),

nudge\_x = ifelse(

date %in% c(ymd("0000-01-01"), ymd("0000-04-01"), ymd("0000-07-01"), ymd("0000-10-01")),

c(-1, -2, -2, 1)[round(month(date)/3)+1],

0

),

nudge\_y = ifelse(

date %in% c(ymd("0000-01-01"), ymd("0000-04-01"), ymd("0000-07-01"), ymd("0000-10-01")),

c(-2, 1, 0.5, -2)[round(month(date)/3)+1],

0

)

)

#描画

temp\_plot <- ggplot(temps\_wide\_label, aes(x = date, y = `Houston`)) +

geom\_line(size = 1, color = "#0072B2") +

scale\_x\_date(name = "month", limits = c(ymd("0000-01-01"), ymd("0001-01-03")),

breaks = c(ymd("0000-01-01"), ymd("0000-04-01"), ymd("0000-07-01"),

ymd("0000-10-01"), ymd("0001-01-01")),

labels = c("Jan", "Apr", "Jul", "Oct", "Jan"), expand = c(2/366, 0)) +

scale\_y\_continuous(limits = c(50, 90),

name = "temperature (°F)") +

theme(plot.margin = margin(3, 5, 3, 1.5))

#好きな縦/横比に指定

temp\_plot+theme(aspect.ratio=1.5)

temp\_plot+theme(aspect.ratio=0.5)

temp\_plot+theme(aspect.ratio=1.0)

temp\_plot+theme(aspect.ratio=3.0)

#対数軸の指定

set.seed(3878)

library(devtools)

install\_github("OpenIntroStat/openintro")

library(openintro)

#データ読み込み

getdata <- function(...)

{

e <- new.env()

name <- data(..., envir = e)[1]

e[[name]]

}

# now load your data calling getdata()

US\_census <- getdata("county\_complete")

head(US\_census )

#関数の定義

label\_log10 <- function(x) {

NAs <- is.na(x)

x[NAs] <- 1 # remove NAs

neg <- x < 0

x[neg] <- -1\*x[neg] # remove negative numbers

zero <- x == 0

x[zero] <- 1 # remove zeros

exp <- floor(log10(x))

coef <- signif(x / 10^exp, 2)

sign\_str <- ifelse(neg, "-", "")

zero\_str <- ifelse(zero, "0", "")

coef\_str <- ifelse(coef == 1, "", paste0(coef, " %\*% "))

exp\_str <- ifelse(zero, "", paste0("10^", exp))

labels <- paste0(sign\_str, zero\_str, coef\_str, exp\_str)

labels[NAs] <- "NULL"

parse(text = labels)

}

US\_census %>% filter(state == "Texas") %>%

select(name, pop2010) %>%

extract(name, "county", regex = "(.+) County") %>%

mutate(popratio = pop2010/median(pop2010)) %>%

arrange(desc(popratio)) %>%

mutate(index = 1:n(),

label = ifelse(index <= 3 | index > n()-3 | runif(n()) < .04, county, ""),

label\_large = ifelse(index <= 6, county, "")) -> tx\_counties

#描画

#指数にしない場合

ggplot(tx\_counties, aes(x = index, y = popratio)) +

geom\_hline(yintercept = 1, linetype = 2, color = "grey40") +

geom\_point(size = 0.5, color = "#0072B2") +

geom\_text\_repel(aes(label = label), point.padding = .4, color = "black",

min.segment.length = 0) +

scale\_y\_continuous(expand = c(0, 0)) +

scale\_x\_continuous(limits = c(.5, nrow(tx\_counties) + .5), expand = c(0, 0),

breaks = NULL, #c(1, 50\*(1:5)),

name = "Texas counties, from most to least populous") +

theme(axis.line = element\_blank(),

plot.margin = margin(3, 7, 3, 1.5))

#指数軸 (上の関数を定義する必要あり)

ggplot(tx\_counties, aes(x = index, y = popratio)) +

geom\_hline(yintercept = 1, linetype = 2, color = "grey40") +

geom\_point(size = 0.5, color = "#0072B2") +

geom\_text\_repel(aes(label = label), point.padding = .4, color = "black",

min.segment.length = 0) +

scale\_y\_log10(breaks = c(.01, .1, 1, 10, 100),

name = "population number / median",

labels = label\_log10) +

scale\_x\_continuous(limits = c(.5, nrow(tx\_counties) + .5), expand = c(0, 0),

breaks = NULL, #c(1, 50\*(1:5)),

name = "Texas counties, from most to least populous") +

theme(axis.line = element\_blank(),

plot.margin = margin(3, 7, 3, 1.5))

#Yの指数をとる

ggplot(tx\_counties, aes(x = index, y = log10(popratio))) +

geom\_hline(yintercept = 1, linetype = 2, color = "grey40") +

geom\_point(size = 0.5, color = "#0072B2") +

geom\_text\_repel(aes(label = label), point.padding = .4, color = "black",

min.segment.length = 0) +

scale\_y\_continuous(expand = c(0, 0)) +

scale\_x\_continuous(limits = c(.5, nrow(tx\_counties) + .5), expand = c(0, 0),

breaks = NULL, #c(1, 50\*(1:5)),

name = "Texas counties, from most to least populous") +

theme(axis.line = element\_blank(),

plot.margin = margin(3, 7, 3, 1.5))

#軸のラベルを綺麗にする

ggplot(tx\_counties, aes(x = index, y = log10(popratio))) +

geom\_hline(yintercept = 1, linetype = 2, color = "grey40") +

geom\_point(size = 0.5, color = "#0072B2") +

geom\_text\_repel(aes(label = label), point.padding = .4, color = "black",

min.segment.length = 0) +

scale\_y\_continuous(expand = c(0, 0)) +

scale\_x\_continuous(limits = c(.5, nrow(tx\_counties) + .5), expand = c(0, 0),

breaks = NULL, #c(1, 50\*(1:5)),

name = "Texas counties, from most to least populous") +

theme(axis.line = element\_blank(),

plot.margin = margin(3, 7, 3, 1.5))+

ylab(bquote( Log[10]~(~"Population ratio")))

#データに0がある場合は log(x+1)ともできる

ggplot(tx\_counties, aes(x = index, y = log10(pop2010+1))) +

geom\_hline(yintercept = 1, linetype = 2, color = "grey40") +

geom\_point(size = 0.5, color = "#0072B2") +

geom\_text\_repel(aes(label = label), point.padding = .4, color = "black",

min.segment.length = 0) +

scale\_y\_continuous(expand = c(0, 0)) +

scale\_x\_continuous(expand = c(0, 0),

breaks = NULL, #c(1, 50\*(1:5)),

name = "Texas counties, from most to least populous") +

theme(axis.line = element\_blank(),

plot.margin = margin(3, 7, 3, 1.5))+

ylab(bquote( Log[10]~(~"Population 2010 +1")))

install.packages(c("FactoMineR", "factoextra"))

library("FactoMineR")

library("factoextra")

library(ggplot2)

library(scales)

library(grid)

library(plyr)

library(gridExtra)

#

#### 整えtheme　 ###########

dantheme = theme\_bw() +   
 theme( panel.spacing = unit(2, "lines"),

legend.title.align = 0.5,

legend.key = element\_blank(),

strip.background = element\_rect(  
　　　　　　　 color="#FFFFFF",fill="#FFFFFF"),

panel.border = element\_rect(size=1.0),

panel.grid.minor = element\_blank(),

panel.grid.major = element\_blank(),

panel.background = element\_blank(),

aspect.ratio=0.8,

axis.ticks = element\_line(color = "#000000", size = 0.5),

axis.text.x = element\_text(color="#000000",size=9.5),

axis.text.y = element\_text(color="#000000",size=9.5),

strip.text.y = element\_text(size = 9.5, color = "#000000", angle = 270),

strip.text.x = element\_text(size = 9.5, color = "#000000"),

axis.title=element\_text(size=9.5, color = "#000000")

)

# start with a clean slate

rm(list=ls(all=TRUE))

setwd("~/Desktop/data\_viz")

# load example data

data(decathlon2)

head(decathlon2)

# compute PCA

decathlon2.active <- decathlon2[1:23, 1:10]

res.pca <- PCA(decathlon2.active, graph = FALSE)

#res.pca <- PCA(decathlon, quanti.sup = 11:12, quali.sup=13, graph = FALSE)

#

print(res.pca)

# 関数一覧

**get\_eigenvalue(res.pca)**

# 主成分の固有ベクトルを取得します。

**fviz\_eig(res.pca)**

# 固有ベクトルを可視化します。

**get\_pca\_ind(res.pca), get\_pca\_var(res.pca)**

# 各データと変数の結果をそれぞれ抽出します。

**fviz\_pca\_ind(res.pca), fviz\_pca\_var(res.pca)**

#各データと変数の結果をそれぞれ描画します。

**fviz\_pca\_biplot(res.pca)**

# 個体と変数のプロットを作成します。

library("factoextra")

eig.val <- get\_eigenvalue(res.pca)

eig.val

fviz\_eig(res.pca, addlabels = TRUE, ylim = c(0, 50))

fviz\_eig(res.pca, addlabels = FALSE, ylim = c(0, 50))+dantheme

###　変数一覧　####

fviz\_eig(

X,

choice = c("variance", "eigenvalue"),

geom = c("bar", "line"),

barfill = "steelblue",

barcolor = "steelblue",

linecolor = "black",

ncp = 10,

addlabels = FALSE,

hjust = 0,

main = NULL,

xlab = NULL,

ylab = NULL,

ggtheme = theme\_minimal(),

...

)

fviz\_eig(res.pca, addlabels = FALSE, ylim = c(1.8, 52), geom=c("bar"), barfill="#ababab",barcolor="#000000")+dantheme+scale\_y\_continuous(limits=c(0,52))

eig\_plot =fviz\_eig(res.pca, addlabels = FALSE, ylim = c(1.8, 52), geom=c("bar"), barfill="#ababab",barcolor="#000000")+dantheme+scale\_y\_continuous(limits=c(0,52))

library(egg)

p <-set\_panel\_size(eig\_plot,width = unit(5, "cm"),height = unit(5, "cm"))

#このサイズと最後に保存する際のサイズは違うので注意。

grid.newpage()

grid.draw(p)

#ファイルに保存するサイズの指定

ggsave(plot=p,"pca\_eig\_plot.pdf" ,width=unit(7,"cm"),height=unit(7,"cm"))

var <- get\_pca\_var(res.pca)

var

# Coordinates　位置

head(var$coord)

# Cos2: quality on the factore map

head(var$cos2)

# Contributions to the principal components　主成分への寄与率

head(var$contrib)

# Coordinates of variables　#相関

head(var$coord, 4)

#Correlation circleの描画

fviz\_pca\_var(res.pca, col.var = "black")

fviz\_pca\_var(res.pca, col.var = "black")+dantheme+theme(aspect.ratio=1.0)

# 引数一覧

fviz\_pca\_var(

X,

axes = c(1, 2),

geom = c("arrow", "text"),

geom.var = geom,

repel = FALSE,

col.var = "black",

fill.var = "white",

alpha.var = 1,

col.quanti.sup = "blue",

col.circle = "grey70",

select.var = list(name = NULL, cos2 = NULL, contrib = NULL),

...

)

head(var$cos2, 4)

library(ggcorrplot)

ggcorrplot(var$cos2, method = "circle")

#このggcorrplotは　+theme() などでいじれます

res.pca

**# 固有ベクトルの描画**

fviz\_pca\_var(res.pca, col.var = "contrib",

gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07")

)

# 固有ベクトルを任意の数にクラスタリングして描画することもできます。

# Create 3 groups of variables (centers = 3)

set.seed(123)

res.km <- kmeans(var$coord, centers =2, nstart = 25)

grp <- as.factor(res.km$cluster)

# Color variables by groups

fviz\_pca\_var(res.pca, col.var = grp,

palette = c("#0073C2FF", "#EFC000FF", "#868686FF"),

legend.title = "Cluster")

**# データの描画**

ind <- get\_pca\_ind(res.pca)

ind

# データの確認

# Coordinates of individuals

head(ind$coord)

# Quality of individuals

head(ind$cos2)

# Contributions of individuals

head(ind$contrib)

#引数一覧

fviz\_pca\_ind(

X,

axes = c(1, 2),

geom = c("point", "text"),

geom.ind = geom,

repel = FALSE,

habillage = "none",

palette = NULL,

addEllipses = FALSE,

col.ind = "black",

fill.ind = "white",

col.ind.sup = "blue",

alpha.ind = 1,

select.ind = list(name = NULL, cos2 = NULL, contrib = NULL),

...

)

fviz\_pca\_ind(res.pca, col.ind = "#000000",

pallette= c("#ababab", "#E7B800"),geom = c("point"),

repel = TRUE # Avoid text overlapping (slow if many points)

)+dantheme

#主成分分析の結果のクラスタリング

# irisのデータを使います

head(iris, 3)

# The variable Species (index = 5) is removed

# before PCA analysis

iris.pca <- PCA(iris[,-5], graph = FALSE)

fviz\_pca\_ind(iris.pca,

geom.ind = "point", # show points only (nbut not "text")

col.ind = iris$Species, # color by groups

palette = c("#00AFBB", "#E7B800", "#FC4E07"),

addEllipses = TRUE, # Concentration ellipses

legend.title = "Groups"

)

fviz\_pca\_ind(iris.pca,

geom.ind = "point", # show points only (nbut not "text")

col.ind = iris$Species, # color by groups

palette = c("#00AFBB", "#E7B800", "#FC4E07"),

addEllipses = TRUE, # Concentration ellipses

legend.title = "Groups"

)+dantheme

pca\_plot = fviz\_pca\_ind(iris.pca,

geom.ind = "point", # show points only (nbut not "text")

col.ind = iris$Species, # color by groups

palette = c("#00AFBB", "#E7B800", "#FC4E07"),

addEllipses = TRUE, # Concentration ellipses

legend.title = "Groups"

)+dantheme

pca\_plot+theme(legend.position=”none”)

#サイズの指定

library(egg)

p <-set\_panel\_size(pca\_plot+dantheme,width = unit(5, "cm"),height = unit(5, "cm"))

#このサイズと最後に保存する際のサイズは違うので注意。

grid.newpage()

grid.draw(p)

#ファイルに保存するサイズの指定

ggsave(plot=p,"pca\_plot.pdf" ,width=unit(7,"cm"),height=unit(7,"cm"))

　　#この保存サイズは大まかに　1ページ幅=14cm, 半ページ幅=7cm くらいです。

library(sf)

library(ggplot2)

library(scales)

library(grid)

library(plyr)

library(dplyr)

library(gridExtra)

library(openintro)

library(devtools)

library(colorspace)

#データ読み込み

getdata <- function(...)

{

e <- new.env()

name <- data(..., envir = e)[1]

e[[name]]

}

# now load your data calling getdata()

US\_census <- getdata("county\_complete")

head(US\_census )

US\_regions <- read.csv("~/Desktop/data\_viz/dviz.supp-master/data-raw/US\_regions/US\_regions.csv", stringsAsFactors = FALSE)

popgrowth\_df <- left\_join(US\_census, US\_regions) %>%

group\_by(region, division, state) %>%

summarize(pop2000 = sum(pop2000, na.rm = TRUE),

pop2010 = sum(pop2010, na.rm = TRUE),

popgrowth = (pop2010-pop2000)/pop2000,

area = sum(area\_2010)) %>%

arrange(popgrowth) %>%

ungroup() %>%

mutate(state = factor(state, levels = state),

region = factor(region, levels = c("West", "South", "Midwest", "Northeast")))

region\_colors <- c("#E69F00", "#56B4E9", "#009E73", "#F0E442")

region\_colors\_dark <- darken(region\_colors, 0.4)

state\_colors <- region\_colors\_dark[as.numeric(popgrowth\_df$region[order(popgrowth\_df$state)])]

dantheme = theme\_bw() +   
 theme( panel.spacing = unit(2, "lines"),

legend.title.align = 0.5,

legend.key = element\_blank(),

strip.background = element\_rect(  
　　　　　　　 color="#FFFFFF",fill="#FFFFFF"),

panel.border = element\_rect(size=1.0),

panel.grid.minor = element\_blank(),

panel.grid.major = element\_blank(),

panel.background = element\_blank(),

axis.ticks = element\_line(color = "#000000", size = 0.5),

axis.text.x = element\_text(color="#000000",size=9.5),

axis.text.y = element\_text(color="#000000",size=9.5),

strip.text.y = element\_text(size = 9.5, color = "#000000", angle = 270),

strip.text.x = element\_text(size = 9.5, color = "#000000"),

axis.title=element\_text(size=9.5, color = "#000000")

)

ggplot(popgrowth\_df, aes(x = state, y = 100\*popgrowth, fill = region)) +

geom\_col() +

scale\_y\_continuous(

limits = c(-.6, 37.5), expand = c(0, 0),

labels = scales::percent\_format(accuracy = 1, scale = 1),

name = "population growth, 2000 to 2010"

) +

scale\_fill\_manual(values = region\_colors) +

coord\_flip() +

theme(axis.title.y = element\_blank(),

axis.line.y = element\_blank(),

axis.ticks.length = unit(0, "pt"),

axis.text.y = element\_text(size = 10, color = state\_colors),

legend.position = c(.56, .68),

#legend.text = element\_text(color = region\_colors),

legend.background = element\_rect(fill = "#ffffffb0"))+dantheme

#region\_colors <- c(“Northeast”="#E69F00",

“South”="#56B4E9",

”Midwest”= "#009E73",

“West”="#F0E442")

install.packages("rgdal", dependencies = TRUE)

library(tidycensus)

library(tidyverse)

census\_api\_key(key='04a67126af11175583422954745efa37fe61ceb0',install=TRUE)

readRenviron("~/.Renviron")

# get median income

texas\_income <- get\_acs(state = "TX", geography = "county", year = 2015,

variables = "B19013\_001", geometry = TRUE)

texas\_crs <- "+proj=aea +lat\_1=27.5 +lat\_2=35 +lat\_0=18 +lon\_0=-100 +x\_0=1500000 +y\_0=6000000 +ellps=GRS80 +datum=NAD83 +units=m +no\_defs"

# -110, -93.5 transformed using texas\_crs

texas\_xlim <- c(558298.7, 2112587)

texas\_income %>% st\_transform(crs = texas\_crs) %>%

ggplot(aes(fill = estimate)) +

geom\_sf(color = "white") +

coord\_sf(xlim = texas\_xlim, datum = NA) +

scale\_fill\_distiller(

palette = "Blues", type = 'seq', na.value = "grey60", direction = 1,

name = "annual median income (USD)",

limits = c(18000, 90000),

breaks = 20000\*c(1:4),

labels = c("$20,000", "$40,000", "$60,000", "$80,000"),

guide = guide\_colorbar(

direction = "horizontal",

label.position = "bottom",

title.position = "top",

ticks = FALSE,

barwidth = grid::unit(3.0, "in"),

barheight = grid::unit(0.2, "in")

)

) +

theme(

legend.title.align = 0.5,

legend.text.align = 0.5,

legend.justification = c(0, 0),

legend.position = c(0.02, 0.1)

)

racevars <- c("P005003", "P005004", "P005006", "P004003")

texas\_race\_raw <- get\_decennial(state = "TX", geography = "county", year = 2010,

variables = racevars, geometry = TRUE,

summary\_var = "P001001")

texas\_race <- texas\_race\_raw %>%

mutate(pct = 100 \* (value / summary\_value),

variable = fct\_recode(variable,

White = "P005003",

Black = "P005004",

Asian = "P005006",

Hispanic = "P004003"))

texas\_race %>% st\_sf() %>%

st\_transform(crs = texas\_crs) %>%

filter(variable == "White") %>%

ggplot(aes(fill = pct)) +

geom\_sf(color = "white") +

coord\_sf(xlim = texas\_xlim, datum = NA) +

scale\_fill\_continuous\_divergingx(

palette = "Earth",

mid = 50,

limits = c(0, 100),

breaks = 25\*(0:4),

labels = c("0% ", "25%", "50%", "75%", " 100%"),

name = "percent identifying as white",

guide = guide\_colorbar(

direction = "horizontal",

label.position = "bottom",

title.position = "top",

ticks = FALSE,

barwidth = grid::unit(3.0, "in"),

barheight = grid::unit(0.2, "in"))) +

theme(

legend.title.align = 0.5,

legend.text.align = 0.5,

legend.justification = c(0, 0),

legend.position = c(0.02, 0.1)

)

popgrowth\_hilight <- left\_join(US\_census, US\_regions) %>%

group\_by(region, division, state) %>%

summarize(pop2000 = sum(pop2000, na.rm = TRUE),

pop2010 = sum(pop2010, na.rm = TRUE),

popgrowth = (pop2010-pop2000)/pop2000,

area = sum(area\_2010)) %>%

arrange(popgrowth) %>%

ungroup() %>%

mutate(region = ifelse(state %in% c("Texas", "Louisiana"), "highlight", region)) %>%

mutate(state = factor(state, levels = state),

region = factor(region, levels = c("West", "South", "Midwest", "Northeast", "highlight")))

region\_colors\_bars <- c(desaturate(lighten(c("#E69F00", "#56B4E9", "#009E73", "#F0E442"), .4), .8), darken("#56B4E9", .3))

region\_colors\_axis <- c(rep("gray30", 4), darken("#56B4E9", .4))

region\_fontface <- c(rep("plain", 4), "bold")

state\_colors <- region\_colors\_axis[as.numeric(popgrowth\_hilight$region[order(popgrowth\_hilight$state)])]

state\_fontface <- region\_fontface[as.numeric(popgrowth\_hilight$region[order(popgrowth\_hilight$state)])]

ggplot(popgrowth\_hilight, aes(x = state, y = 100\*popgrowth, fill = region)) +

geom\_col() +

scale\_y\_continuous(

limits = c(-.6, 37.5), expand = c(0, 0),

labels = scales::percent\_format(accuracy = 1, scale = 1),

name = "population growth, 2000 to 2010"

) +

scale\_fill\_manual(

values = region\_colors\_bars,

breaks = c("West", "South", "Midwest", "Northeast")

) +

coord\_flip() +

theme(

text = element\_text(color = "gray30"),

axis.text.x = element\_text(color = "gray30"),

axis.title.y = element\_blank(),

axis.line.y = element\_blank(),

axis.ticks.length = unit(0, "pt"),

axis.text.y = element\_text(

size = 10, color = state\_colors,

face = state\_fontface

),

legend.position = c(.56, .68),

legend.background = element\_rect(fill = "#ffffffb0")

)

library(DAAG)

library(sf)

library(ggplot2)

library(scales)

library(grid)

library(plyr)

library(dplyr)

library(gridExtra)

library(openintro)

library(devtools)

library(colorspace)

Aus\_athletes = ais

male\_Aus <- filter(Aus\_athletes, sex=="m") %>%

filter(sport %in% c("B\_Ball", "Field", "Swim", "T\_400m",

"T\_Sprnt", "Row"))

male\_Aus$sport <- factor(male\_Aus$sport,

levels = c("T\_400m","T\_Sprnt", "Field", "Row", "B\_Ball", "Swim"))

colors <- c("#BD3828", rep("#808080", 5))

fills <- c("#BD3828D0", rep("#80808080", 5))

ggplot(male\_Aus, aes(x=ht, y=pcBfat, shape=sport, color = sport, fill = sport)) +

geom\_point(size = 3) +

scale\_shape\_manual(values = 21:26) +

scale\_color\_manual(values = colors) +

scale\_fill\_manual(values = fills) +

xlab("height (cm)") +

ylab("% body fat") +

dantheme

group.colors <- c("T\_400m"="#ababab",

"T\_Sprnt"="#ababab",

"Field"="#ababab",

"Row"="#ababab",

"B\_Ball"="#ababab",

"Swim"= "#ababab")

#### ライブラリ読み込み　 ###########library(dplyr)

library(ggplot2)

library(forcats)

#### データ読み込み・整形####

boxoffice <- data.frame(rank = 1:5,

title = c("Star Wars: The Last Jedi", "Jumanji: Welcome to the Jungle", "Pitch Perfect 3", "The Greatest Showman", "Ferdinand"),

title\_short = c("Star Wars", "Jumanji", "Pitch Perfect 3", "Greatest Showman", "Ferdinand"),

amount = c(71565498, 36169328, 19928525, 8805843, 7316746),

amount\_text = c("$71,565,498", "$36,169,328", "$19,928,525", "$8,805,843", "$7,316,746"))

boxoffice\_display <- boxoffice %>%

mutate(A = " ", B = " ", C = " ") %>%

select(A, rank, title, amount\_text, B, C) %>%

rename(` ` = A,

Rank = rank,

Title = title,

`Weekend gross` = amount\_text,

` ` = B,

` ` = C)

#### 整えtheme　 ###########

dantheme = theme\_bw() +   
 theme( panel.spacing = unit(2, "lines"),

legend.title.align = 0.5,

legend.key = element\_blank(),

strip.background = element\_rect(  
　　　　　　　 color="#FFFFFF",fill="#FFFFFF"),

panel.border = element\_rect(size=1.0),

panel.grid.minor = element\_blank(),

panel.grid.major = element\_blank(),

panel.background = element\_blank(),

#aspect.ratio=0.8,

axis.ticks = element\_line(color = "#000000", size = 0.5),

axis.text.x = element\_text(color="#000000",size=9.5),

axis.text.y = element\_text(color="#000000",size=9.5),

strip.text.y = element\_text(size = 9.5, color = "#000000", angle = 270),

strip.text.x = element\_text(size = 9.5, color = "#000000"),

axis.title=element\_text(size=9.5, color = "#000000")

)

##### 描画

boxoffice %>%

ggplot(aes(x = fct\_reorder(title\_short, rank), y = amount)) +

geom\_col(fill = "#56B4E9", width = 0.6, alpha = 0.9) +

scale\_y\_continuous(expand = c(0, 1),

breaks = c(0, 2e7, 4e7, 6e7,8e7),

labels = c("0", "20", "40", "60","80"),

name = "weekend gross (million USD)") +

scale\_x\_discrete(name = NULL,

expand = c(0, 0.4)) +

coord\_cartesian(clip = "off") +

dantheme +

theme(

#axis.ticks.length = grid::unit(0, "pt"),

axis.line.x = element\_blank(),

axis.ticks.x = element\_blank()

)

boxoffice %>%

ggplot(aes(x = fct\_reorder(title\_short, rank), y = amount)) +

geom\_col(fill = "#56B4E9", alpha = 0.9) +

scale\_y\_continuous(expand = c(0, 0),

breaks = c(0, 2e7, 4e7, 6e7),

labels = c("0", "20", "40", "60"),

name = "weekend gross (million USD)") +

scale\_x\_discrete(name = NULL) +

coord\_cartesian(clip = "off") +

dantheme+

theme(

#axis.ticks.length = grid::unit(0, "pt"),

axis.line.x = element\_blank(),

axis.ticks.x = element\_blank(),

axis.text.x = element\_text(angle = 45, vjust = 1, hjust = 1),

plot.margin = margin(3, 7, 3, 1.5)

)

library(forcats)

ggplot(boxoffice, aes(x = fct\_reorder(title\_short, desc(rank)), y = amount)) +

geom\_col(fill = "#56B4E9", alpha = 0.9) +

scale\_y\_continuous(limits = c(0, 7.5e7),

expand = c(0, 0),

breaks = c(0, 2e7, 4e7, 6e7),

labels = c("0", "20", "40", "60"),

name = "weekend gross (million USD)") +

scale\_x\_discrete(name = NULL,

expand = c(0, 0.5)) +

coord\_flip(clip = "off") +

dantheme +

theme(

#axis.ticks.length = grid::unit(0, "pt"),

axis.line.y = element\_blank(),

axis.ticks.y = element\_blank()

)

ggplot(boxoffice, aes(x = factor(title\_short, levels = title\_short[c(2, 1, 5, 3, 4)]),

y = amount)) +

geom\_col(fill = "#56B4E9", alpha = 0.9) +

scale\_y\_continuous(limits = c(0, 7.5e7),

expand = c(0, 0),

breaks = c(0, 2e7, 4e7, 6e7),

labels = c("0", "20", "40", "60"),

name = "weekend gross (million USD)") +

scale\_x\_discrete(name = NULL,

expand = c(0, 0.5)) +

coord\_flip(clip = "off") +

dantheme+

theme(

#axis.ticks.length = grid::unit(0, "pt"),

axis.line.y = element\_blank(),

axis.ticks.y = element\_blank()

)

### データ読み込み

load("/Users/danyamamotoevans/Desktop/data\_viz/dviz.supp-master/data/income\_by\_age.rda")

Income\_by\_age

income\_by\_age %>% filter(race %in% c("white", "asian", "black", "hispanic")) %>%

mutate(race = fct\_relevel(race, c("asian", "white", "hispanic", "black")),

race = fct\_recode(race, Asian = "asian", Hispanic = "hispanic"),

age = fct\_recode(age, "≥ 75" = "> 74")) -> income\_df

# Take the darkest four colors from 5-class ColorBrewer palette "PuBu"

colors\_four = RColorBrewer::brewer.pal(5, "PuBu")[5:2]

colors\_seven = RColorBrewer::brewer.pal(8, "PuBu")[2:8]

income\_df %>%

mutate(age = fct\_recode(age, "15–24" = "15 to 24", "25–34" = "25 to 34", "35–44" = "35 to 44",

"45–54" = "45 to 54", "55–64" = "55 to 64", "65–74" = "65 to 74")) -> income\_age\_abbrev\_df

load("/Users/danyamamotoevans/Desktop/data\_viz/dviz.supp-master/data/titanic\_all.rda")

titanic\_groups <- titanic\_all %>% filter(class != "\*") %>%

select(class, sex) %>%

group\_by(class, sex) %>%

tally() %>% arrange(class, desc(sex)) %>%

mutate(sex = factor(sex, levels = c("female", "male"))) %>%

group\_by(class) %>%

mutate(nlabel = cumsum(n) - n/2) %>%

ungroup() %>%

mutate(class = paste(class, "class"))

### 描画

ggplot(income\_df, aes(x = age, y = median\_income, fill = race)) +

geom\_col(position = "dodge", alpha = 0.9) +

scale\_y\_continuous(

expand = c(0, 0),

name = "median income (USD)",

breaks = c(0, 20000, 40000, 60000, 80000, 100000),

labels = c("$0", "$20,000", "$40,000", "$60,000", "$80,000", "$100,000")

) +

scale\_fill\_manual(values = colors\_four, name = NULL) +

coord\_cartesian(clip = "off") +

xlab("age (years)") +

dantheme+

theme(

axis.line.x = element\_blank(),

axis.ticks.x = element\_blank()

)

ggplot(income\_df, aes(x = race, y = median\_income, fill = age)) +

geom\_col(position = "dodge", alpha = 0.9) +

scale\_y\_continuous(

expand = c(0, 0),

name = "median income (USD)",

breaks = c(0, 20000, 40000, 60000, 80000, 100000),

labels = c("$0", "$20,000", "$40,000", "$60,000", "$80,000", "$100,000")

) +

scale\_fill\_manual(values = colors\_seven, name = "age (yrs)") +

coord\_cartesian(clip = "off") +

xlab(label = NULL) +

dantheme +

theme(

axis.line.x = element\_blank(),

axis.ticks.x = element\_blank(),

legend.title.align = 0.5

)

ggplot(income\_age\_abbrev\_df, aes(x = age, y = median\_income)) +

geom\_col(fill = "#56B4E9", alpha = 0.9) +

scale\_y\_continuous(

expand = c(0, 0),

name = "median income (USD)",

breaks = c(0, 20000, 40000, 60000, 80000, 100000),

labels = c("$0", "$20,000", "$40,000", "$60,000", "$80,000", "$100,000")

) +

coord\_cartesian(clip = "off") +

xlab(label = "age (years)") +

facet\_wrap(~race, scales = "free\_x") +

dantheme +

theme(

#axis.ticks.length = grid::unit(0, "pt"),

axis.ticks.x = element\_blank(),

axis.line = element\_blank(),

strip.text = element\_text(size = 14),

panel.spacing.y = grid::unit(14, "pt")

)

ggplot(titanic\_groups, aes(x = class, y = n, fill = sex)) +

geom\_col(position = "stack", color = "white", size = 1, width = 1) +

geom\_text(

aes(y = nlabel, label = n), color = "white", size = 14/.pt,

) +

scale\_x\_discrete(expand = c(0, 0), name = NULL) +

scale\_y\_continuous(expand = c(0, 0), breaks = NULL, name = NULL) +

scale\_fill\_manual(

values = c("#D55E00", "#0072B2"),

breaks = c("female", "male"),

labels = c("female passengers ", "male passengers"),

name = NULL

) +

coord\_cartesian(clip = "off") +

dantheme+

theme(

panel.grid.major = element\_blank(),

axis.ticks = element\_blank(),

axis.text = element\_text(size = 14),

legend.position = "bottom",

legend.justification = "center",

legend.background = element\_rect(fill = "white"),

legend.spacing.x = grid::unit(4.5, "pt"),

legend.spacing.y = grid::unit(0, "cm"),

legend.box.spacing = grid::unit(7, "pt")

)

## データ読み込み・整形

install.packages("gapminder")

library(gapminder)

df\_Americas <- gapminder %>% filter(year == 2007, continent == "Americas")

#描画

ggplot(df\_Americas, aes(x = lifeExp, y = fct\_reorder(country, lifeExp))) +

geom\_point(color = "#0072B2", size = 3) +

scale\_x\_continuous(

name = "life expectancy (years)",

limits = c(59.7, 81.5),

expand = c(0, 0)

) +

scale\_y\_discrete(name = NULL, expand = c(0, 0.5)) +

dantheme +

theme(

#axis.ticks.length = grid::unit(0, "pt"),

#axis.title = element\_text(size = 12),

plot.margin = margin(18, 6, 3, 1.5)

)

ggplot(df\_Americas, aes(y = lifeExp, x = fct\_reorder(country, lifeExp))) +

geom\_col(fill = "#56B4E9", alpha = 0.9) +

scale\_y\_continuous(

name = "life expectancy (years)",

limits = c(0, 85),

expand = c(0, 0)

) +

scale\_x\_discrete(name = NULL, expand = c(0, 0.5)) +

coord\_flip(clip = "off") +

dantheme +

theme(

#axis.ticks.length = grid::unit(0, "pt"),

axis.line.y = element\_blank(),

axis.ticks.y = element\_blank(),

#axis.title = element\_text(size = 12),

plot.margin = margin(18, 6, 3, 1.5)

)

ggplot(df\_Americas, aes(x = lifeExp, y = fct\_rev(country))) +

geom\_point(color = "#0072B2", size = 3) +

scale\_x\_continuous(name = "life expectancy (years)",

limits = c(59.7, 81.5),

expand = c(0, 0)) +

scale\_y\_discrete(name = NULL, expand = c(0, 0.5)) +

dantheme +

theme(#axis.ticks.length = grid::unit(0, "pt"),

#axis.title = element\_text(size = 12),

plot.margin = margin(18, 6, 3, 1.5))

## データ読み込み・整形

load("/Users/danyamamotoevans/Desktop/data\_viz/dviz.supp-master/data/internet.rda")

country\_list = c("United States", "China", "India", "Japan", "Algeria",

"Brazil", "Germany", "France", "United Kingdom", "Italy", "New Zealand",

"Canada", "Mexico", "Chile", "Argentina", "Norway", "South Africa", "Kenya",

"Israel", "Iceland")

internet\_short <- filter(internet, country %in% country\_list) %>%

mutate(users = ifelse(is.na(users), 0, users))

internet\_summary <- internet\_short %>%

group\_by(country) %>%

summarize(year1 = min(year[users > 0]),

last = users[n()]) %>%

arrange(last, desc(year1))

internet\_short <- internet\_short %>%

mutate(country = factor(country, levels = internet\_summary$country))

##　描画

ggplot(filter(internet\_short, year > 1993),

aes(x = year, y = country, fill = users)) +

geom\_tile(color = "white", size = 0.25) +

scale\_fill\_viridis\_c(

option = "A", begin = 0.05, end = 0.98,

limits = c(0, 100),

name = "internet users / 100 people",

guide = guide\_colorbar(

direction = "horizontal",

label.position = "bottom",

title.position = "top",

ticks = FALSE,

barwidth = grid::unit(3.5, "in"),

barheight = grid::unit(0.2, "in")

)

) +

scale\_x\_continuous(expand = c(0, 0), name = NULL) +

scale\_y\_discrete(name = NULL, position = "right") +

dantheme+

theme(

axis.line = element\_blank(),

axis.ticks = element\_blank(),

axis.ticks.length = grid::unit(1, "pt"),

legend.position = "top",

legend.justification = "left",

legend.title.align = 0.5,

legend.title = element\_text(size = 12\*12/14)

)

internet\_summary <- internet\_short %>%

group\_by(country) %>%

summarize(year1 = min(year[users > 20]),

last = users[n()]) %>%

arrange(desc(year1), last)

internet\_short <- internet\_short %>%

mutate(country = factor(country, levels = internet\_summary$country))

ggplot(filter(internet\_short, year > 1993),

aes(x = year, y = country, fill = users)) +

geom\_tile(color = "white", size = 0.25) +

scale\_fill\_viridis\_c(

option = "A", begin = 0.05, end = 0.98,

limits = c(0, 100),

name = "internet users / 100 people",

guide = guide\_colorbar(

direction = "horizontal",

label.position = "bottom",

title.position = "top",

ticks = FALSE,

barwidth = grid::unit(3.5, "in"),

barheight = grid::unit(0.2, "in")

)

) +

scale\_x\_continuous(expand = c(0, 0), name = NULL) +

scale\_y\_discrete(name = NULL, position = "right") +

dantheme +

theme(

axis.line = element\_blank(),

axis.ticks = element\_blank(),

axis.ticks.length = grid::unit(1, "pt"),

legend.position = "top",

legend.justification = "left",

legend.title.align = 0.5,

legend.title = element\_text(size = 12\*12/14)

)