Class 5 Data Visualization Lab

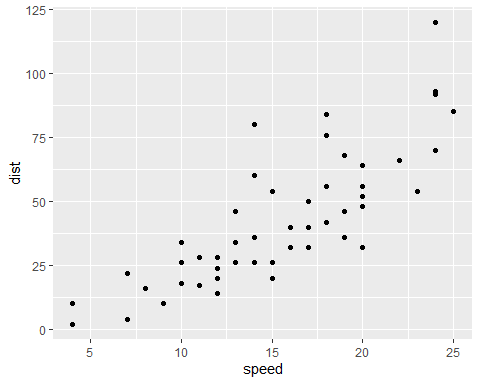
Cindy Tran (A15830581)

Winter 2022

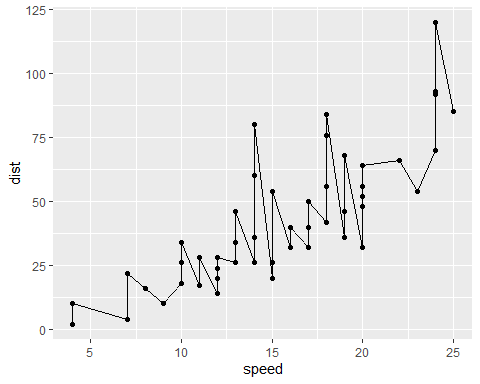
# Week 4 Data Visualization Lab  
  
# Install the package ggplot2  
#install.packages("ggplot2")  
  
# Any time I want to use this package, I need to load it  
library(ggplot2)  
  
View(cars)  
  
# A quick baseR plot - this is not ggplot  
plot(cars)



# Our first ggplot  
#We need data + aes + geom  
ggplot(data = cars) +  
 aes(x = speed, y = dist) +  
 geom\_point()

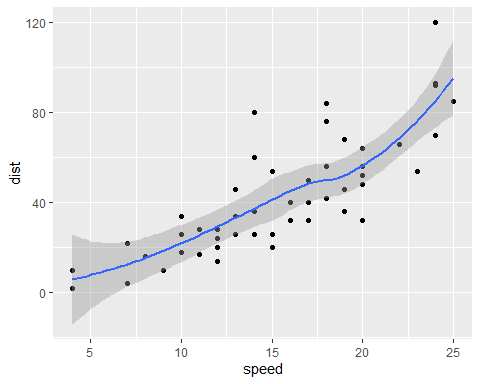


p <- ggplot(data = cars) +  
 aes(x = speed, y = dist) +  
 geom\_point()   
  
# Add a line geom with geom\_line()  
p + geom\_line()



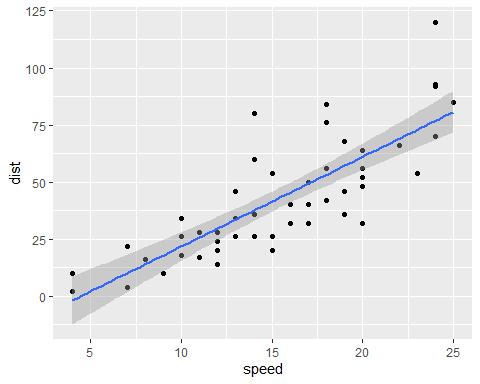
# Add a trend line close to the data  
  
p + geom\_smooth()

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



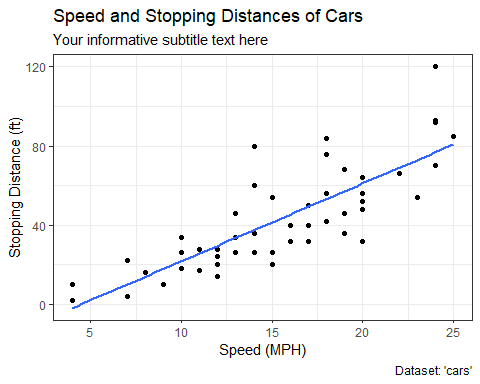
p + geom\_smooth(method = "lm")

## `geom\_smooth()` using formula 'y ~ x'



# Adding labels  
p + geom\_smooth(method = "lm", se = FALSE) +  
 labs(title = "Speed and Stopping Distances of Cars",  
 x = "Speed (MPH)",  
 y = "Stopping Distance (ft)",  
 subtitle = "Your informative subtitle text here",  
 caption = "Dataset: 'cars'") +   
 theme\_bw()

## `geom\_smooth()` using formula 'y ~ x'



##  
  
#Read in drug expression data  
  
url <- "https://bioboot.github.io/bimm143\_S20/class-material/up\_down\_expression.txt"  
  
genes <- read.delim(url)  
head(genes)

## Gene Condition1 Condition2 State  
## 1 A4GNT -3.6808610 -3.4401355 unchanging  
## 2 AAAS 4.5479580 4.3864126 unchanging  
## 3 AASDH 3.7190695 3.4787276 unchanging  
## 4 AATF 5.0784720 5.0151916 unchanging  
## 5 AATK 0.4711421 0.5598642 unchanging  
## 6 AB015752.4 -3.6808610 -3.5921390 unchanging

# How many genes  
nrow(genes)

## [1] 5196

# Column names and number of columns  
colnames(genes)

## [1] "Gene" "Condition1" "Condition2" "State"

ncol(genes)

## [1] 4

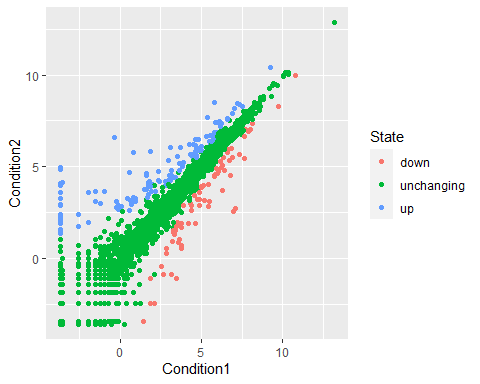
#How many upreglated genes  
table(genes$State)

##   
## down unchanging up   
## 72 4997 127

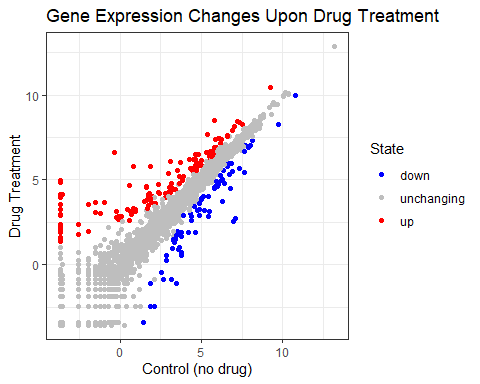
#Fraction of genes up-regulated  
round ( (table(genes$State) / nrow(genes)) \* 100, 2)

##   
## down unchanging up   
## 1.39 96.17 2.44

# Let's make a first plot attempt  
ggplot(data = genes) +   
 aes(x = Condition1, y = Condition2, col = State) +   
 geom\_point()



#Change colors  
ggplot(data = genes) +  
 aes(x = Condition1, y = Condition2, col = State) +  
 geom\_point() +  
 scale\_color\_manual(values = c("blue", "gray", "red")) +  
 labs(title = "Gene Expression Changes Upon Drug Treatment",  
 x = "Control (no drug)",  
 y = "Drug Treatment") +  
 theme\_bw()



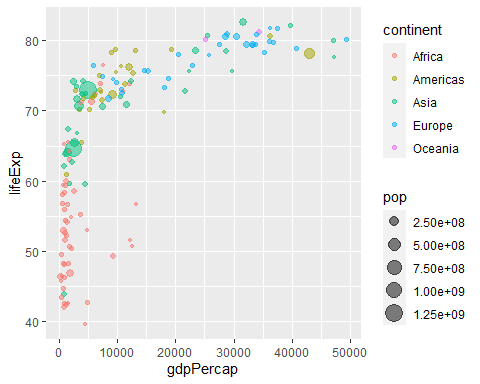
##  
  
#Optional Part 6  
  
#install.packages("gapminder")  
library(gapminder)  
  
# File location online  
url2 <- "https://raw.githubusercontent.com/jennybc/gapminder/master/inst/extdata/gapminder.tsv"  
gapminder <- read.delim(url2)  
  
#install.packages(dplyr)  
library(dplyr)

##   
## Attaching package: 'dplyr'

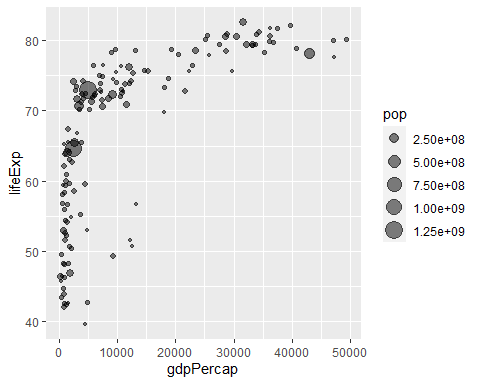
## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

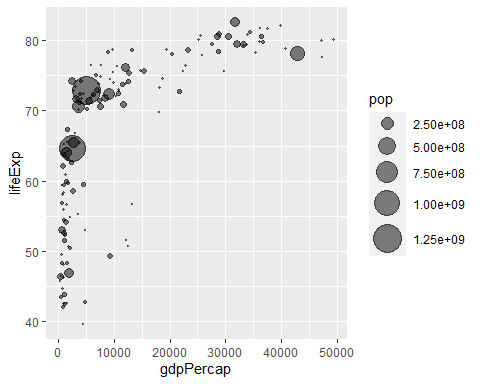
gapminder\_2007 <- gapminder %>%  
 filter(year == 2007)  
  
ggplot(gapminder\_2007) +  
 aes(x = gdpPercap, y = lifeExp, color = continent, size = pop) +  
 geom\_point(alpha = 0.5)



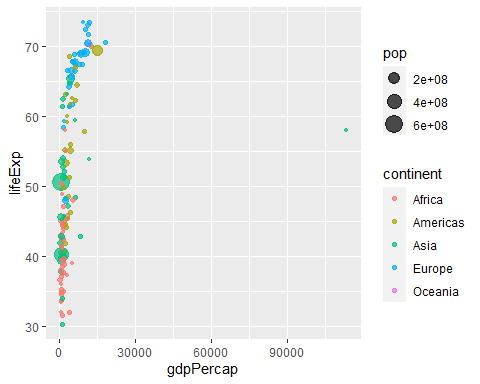
# Color by pop  
ggplot(gapminder\_2007) +   
 aes(x = gdpPercap, y = lifeExp,  
 size = pop) +  
 geom\_point(alpha = 0.5)



#Scale to reflect actual population differences  
ggplot(gapminder\_2007) +   
 geom\_point(aes(x = gdpPercap, y = lifeExp,  
 size = pop),  
 alpha = 0.5) +  
 scale\_size\_area(max\_size = 10)



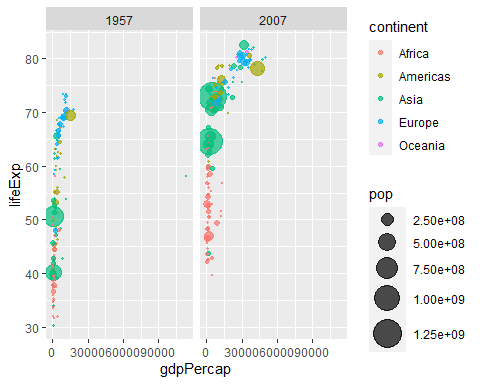
#1957 Plot  
gapminder\_1957 <- gapminder %>%   
 filter(year == 1957)  
  
ggplot(gapminder\_1957) +  
 aes(x = gdpPercap, y = lifeExp,  
 color = continent,  
 size = pop) +  
 geom\_point(alpha = 0.7)



scale\_size\_area(max\_size = 10)

## <ScaleContinuous>  
## Range:   
## Limits: 0 -- 1

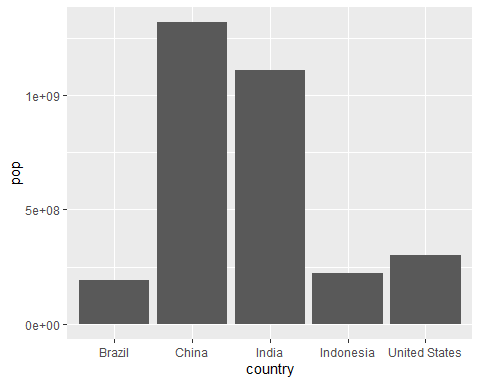
#Combine 1957 and 2007  
gapminder\_combined <- gapminder %>%  
 filter(year == 1957 | year == 2007)  
  
ggplot(gapminder\_combined) +   
 geom\_point(aes(x = gdpPercap, y = lifeExp,  
 color=continent,  
 size = pop), alpha=0.7) +  
 scale\_size\_area(max\_size = 10) +  
 facet\_wrap(~year)



##  
  
#Optional Part 7  
  
gapminder\_top5 <- gapminder %>%   
 filter(year == 2007) %>%   
 arrange(desc(pop)) %>%   
 top\_n(5, pop)  
gapminder\_top5

## country continent year lifeExp pop gdpPercap  
## 1 China Asia 2007 72.961 1318683096 4959.115  
## 2 India Asia 2007 64.698 1110396331 2452.210  
## 3 United States Americas 2007 78.242 301139947 42951.653  
## 4 Indonesia Asia 2007 70.650 223547000 3540.652  
## 5 Brazil Americas 2007 72.390 190010647 9065.801

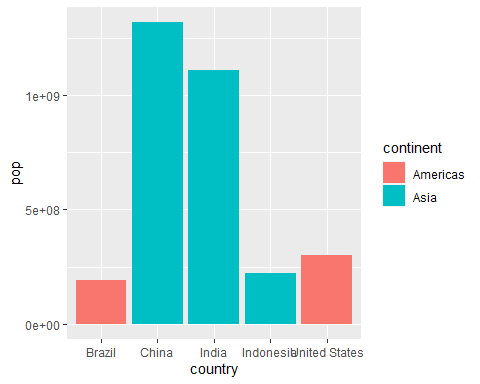
#Creating a bar chart  
ggplot(gapminder\_top5) +  
 geom\_col(aes(x = country, y = pop))



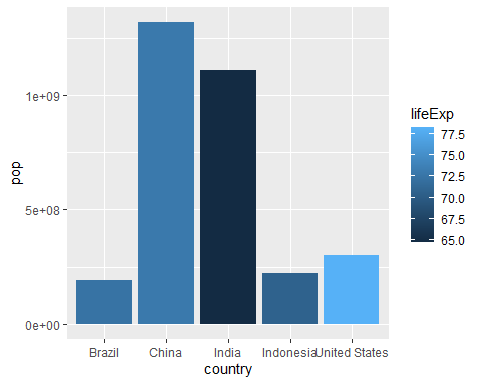
ggplot(gapminder\_top5) +  
 geom\_col(aes(x = country, y = lifeExp))



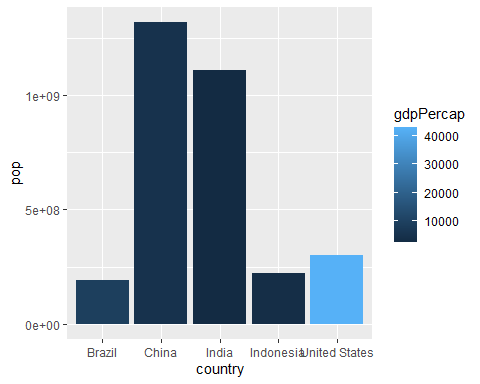
#Filling bars with color  
ggplot(gapminder\_top5) +   
 geom\_col(aes(x = country, y = pop, fill = continent))



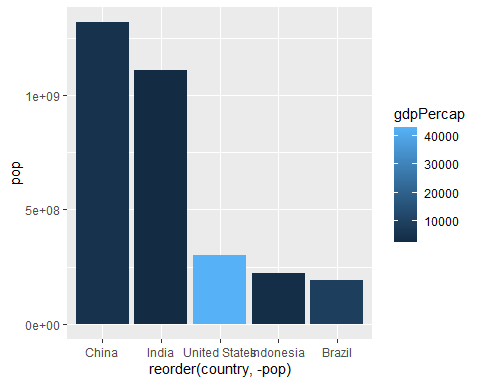
ggplot(gapminder\_top5) +   
 geom\_col(aes(x = country, y = pop, fill = lifeExp))



#Population size by country  
ggplot(gapminder\_top5) +  
 aes(x = country, y = pop, fill = gdpPercap) +  
 geom\_col()

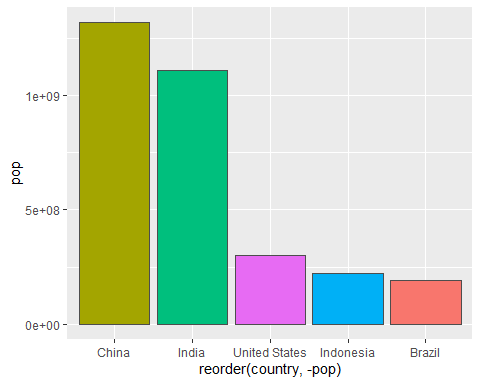


#Change order of bars  
ggplot(gapminder\_top5) +  
 aes(x = reorder(country, -pop), y=pop,  
 fill = gdpPercap) +  
 geom\_col()



ggplot(gapminder\_top5) +  
 aes(x = reorder(country, -pop), y = pop,  
 fill = country) +  
 geom\_col(col = "gray30") +  
 guides(fill = FALSE)

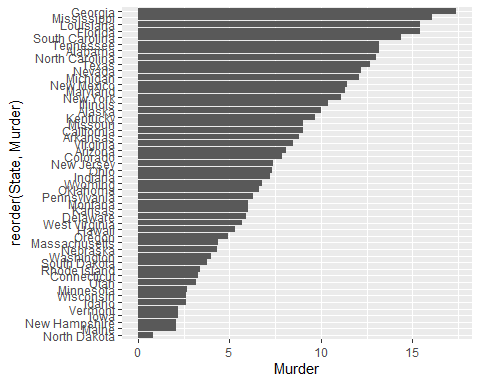
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =  
## "none")` instead.



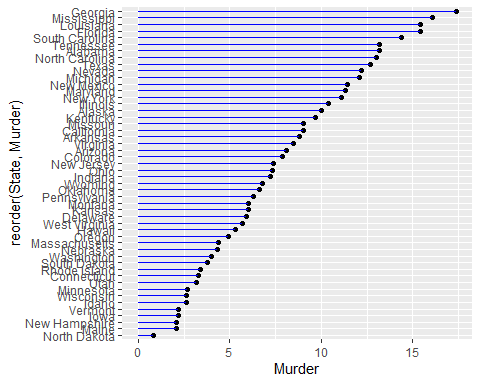
#Flipping bar charts  
head(USArrests)

## Murder Assault UrbanPop Rape  
## Alabama 13.2 236 58 21.2  
## Alaska 10.0 263 48 44.5  
## Arizona 8.1 294 80 31.0  
## Arkansas 8.8 190 50 19.5  
## California 9.0 276 91 40.6  
## Colorado 7.9 204 78 38.7

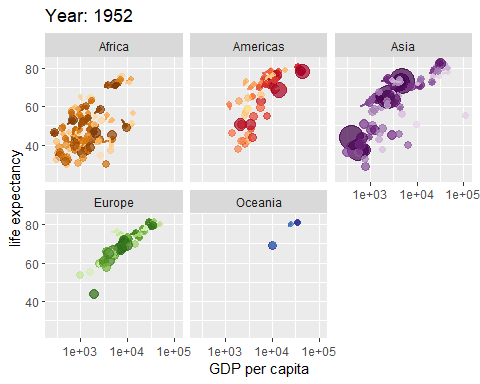
USArrests$State <- rownames(USArrests)  
  
ggplot(USArrests) +  
 aes(x = reorder(State, Murder), y = Murder) +  
 geom\_col() +  
 coord\_flip()



ggplot(USArrests) +  
 aes(x = reorder(State, Murder), y = Murder) +  
 geom\_point() +  
 geom\_segment(aes(x=State,   
 xend=State,   
 y=0,   
 yend=Murder),  
 color = "blue") +  
 coord\_flip()

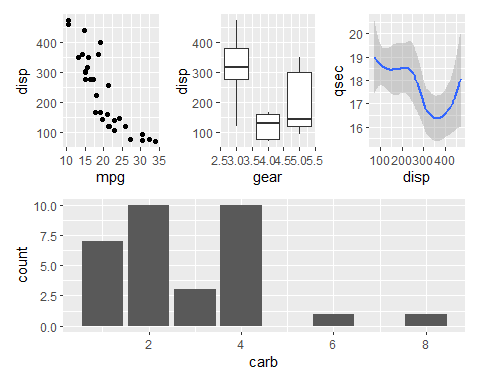


##  
  
#Optional Part 8  
  
#install.packages("gifski")  
#install.packages("gganimate")  
  
library(gapminder)  
library(gganimate)  
  
# Setup nice regular ggplot of the gapminder data  
ggplot(gapminder, aes(gdpPercap, lifeExp, size = pop, colour = country)) +  
 geom\_point(alpha = 0.7, show.legend = FALSE) +  
 scale\_colour\_manual(values = country\_colors) +  
 scale\_size(range = c(2, 12)) +  
 scale\_x\_log10() +  
 # Facet by continent  
 facet\_wrap(~continent) +  
 # Here comes the gganimate specific bits  
 labs(title = 'Year: {frame\_time}', x = 'GDP per capita', y = 'life expectancy') +  
 transition\_time(year) +  
 shadow\_wake(wake\_length = 0.1, alpha = FALSE)



##  
  
#Optional Part 9  
  
#Combining Plots  
  
#install.packages("patchwork")  
library(patchwork)  
  
# Setup some example plots   
p1 <- ggplot(mtcars) + geom\_point(aes(mpg, disp))  
p2 <- ggplot(mtcars) + geom\_boxplot(aes(gear, disp, group = gear))  
p3 <- ggplot(mtcars) + geom\_smooth(aes(disp, qsec))  
p4 <- ggplot(mtcars) + geom\_bar(aes(carb))  
  
# Use patchwork to combine them here:  
(p1 | p2 | p3) /   
 p4

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



sessionInfo()

## R version 4.1.2 (2021-11-01)  
## Platform: x86\_64-w64-mingw32/x64 (64-bit)  
## Running under: Windows 10 x64 (build 19043)  
##   
## Matrix products: default  
##   
## locale:  
## [1] LC\_COLLATE=English\_United States.1252   
## [2] LC\_CTYPE=English\_United States.1252   
## [3] LC\_MONETARY=English\_United States.1252  
## [4] LC\_NUMERIC=C   
## [5] LC\_TIME=English\_United States.1252   
##   
## attached base packages:  
## [1] stats graphics grDevices utils datasets methods base   
##   
## other attached packages:  
## [1] patchwork\_1.1.1 gganimate\_1.0.7 dplyr\_1.0.7 gapminder\_0.3.0  
## [5] ggplot2\_3.3.5   
##   
## loaded via a namespace (and not attached):  
## [1] Rcpp\_1.0.8 plyr\_1.8.6 pillar\_1.6.4 compiler\_4.1.2   
## [5] highr\_0.9 prettyunits\_1.1.1 progress\_1.2.2 tools\_4.1.2   
## [9] digest\_0.6.27 lattice\_0.20-45 nlme\_3.1-155 evaluate\_0.14   
## [13] lifecycle\_1.0.1 tibble\_3.1.6 gtable\_0.3.0 mgcv\_1.8-38   
## [17] pkgconfig\_2.0.3 rlang\_0.4.11 Matrix\_1.4-0 DBI\_1.1.2   
## [21] yaml\_2.2.1 xfun\_0.29 withr\_2.4.3 stringr\_1.4.0   
## [25] knitr\_1.37 hms\_1.1.1 generics\_0.1.1 vctrs\_0.3.8   
## [29] grid\_4.1.2 tidyselect\_1.1.1 glue\_1.6.0 R6\_2.5.1   
## [33] gifski\_1.4.3-1 fansi\_0.5.0 rmarkdown\_2.11 tweenr\_1.0.2   
## [37] purrr\_0.3.4 farver\_2.1.0 magrittr\_2.0.1 splines\_4.1.2   
## [41] scales\_1.1.1 ellipsis\_0.3.2 htmltools\_0.5.1.1 assertthat\_0.2.1   
## [45] colorspace\_2.0-2 labeling\_0.4.2 utf8\_1.2.2 stringi\_1.7.6   
## [49] munsell\_0.5.0 crayon\_1.4.2