Class 5: Data Viz with ggplot

Ashley Allen (PID: A14633373)

Intro to ggplot

There are many graphics systems in R (ways to make plots and figures). These include “base” R plots. Today we will focus mostly on the **ggplot2** package.

> Q1. For which phases is data visualization important in our scientific workflows?

-All of the above

> Q2. True or False? The ggplot2 package comes already installed with R?

-False

> Q3. Which plot types are typically NOT used to compare distributions of numeric variables?

-Network graphs

> Q4. Which statement about data visualization with ggplot2 is incorrect?

-gglplot2 is the only way to create plots in R

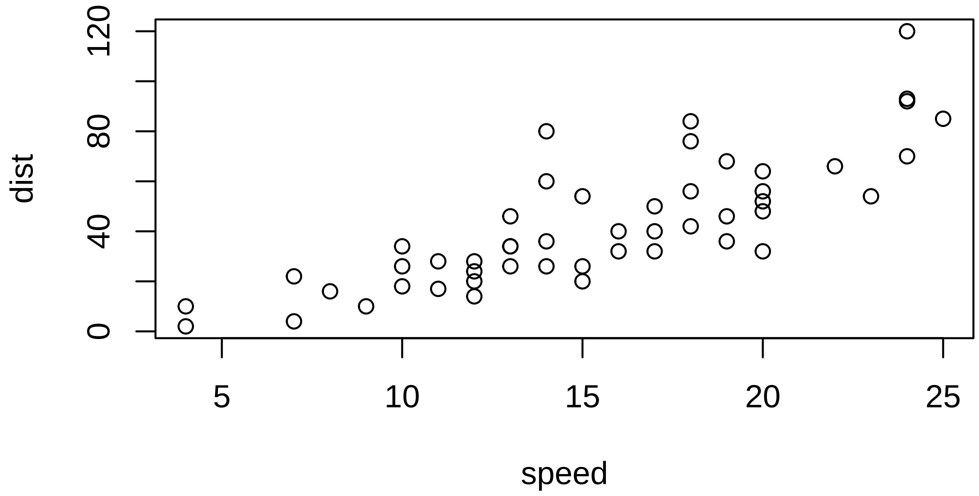
Let’s start with a plot of simple in-built dataset called `cars`.

```
cars
```

```
   speed dist
1      4    2
2      4   10
3      7    4
4      7   22
5      8   16
6      9   10
7     10   18
```

8	10	26
9	10	34
10	11	17
11	11	28
12	12	14
13	12	20
14	12	24
15	12	28
16	13	26
17	13	34
18	13	34
19	13	46
20	14	26
21	14	36
22	14	60
23	14	80
24	15	20
25	15	26
26	15	54
27	16	32
28	16	40
29	17	32
30	17	40
31	17	50
32	18	42
33	18	56
34	18	76
35	18	84
36	19	36
37	19	46
38	19	68
39	20	32
40	20	48
41	20	52
42	20	56
43	20	64
44	22	66
45	23	54
46	24	70
47	24	92
48	24	93
49	24	120
50	25	85

```
plot(cars)
```



Let's see how we can make this figure using **ggplot**. First I need to install this package on my computer. To install any R package I use the function `install.packages()`.

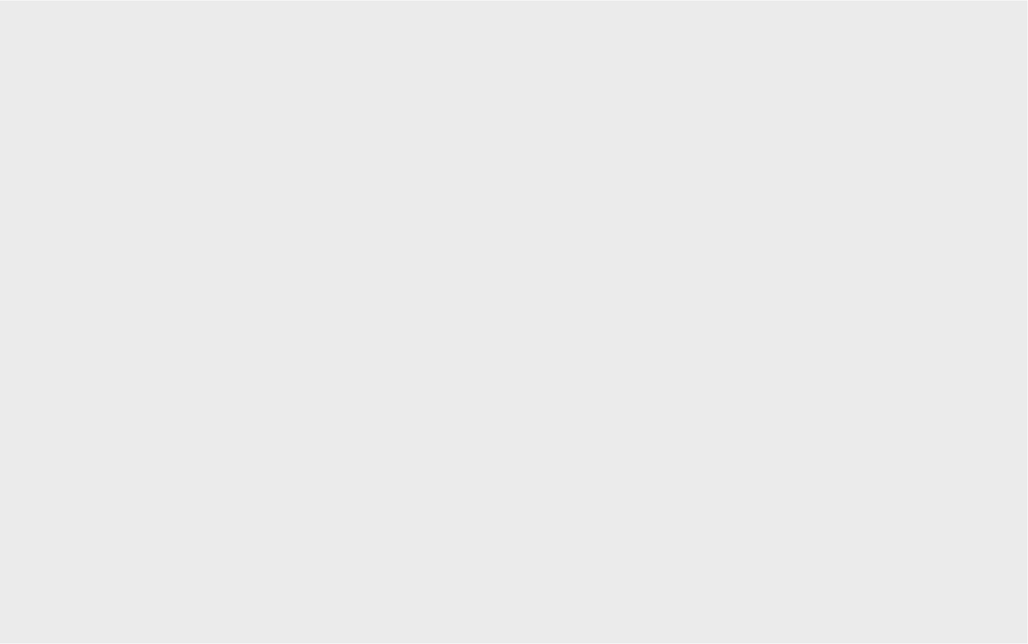
> I will run `install.packages("ggplot2")` in my R console not this quarto document!

Before I can use any functions from add on packages I need to load the package from my "library()" with the `library(ggplot2)` call.

```
library(ggplot2)
```

```
Warning: package 'ggplot2' was built under R version 4.3.3
```

```
ggplot(cars)
```



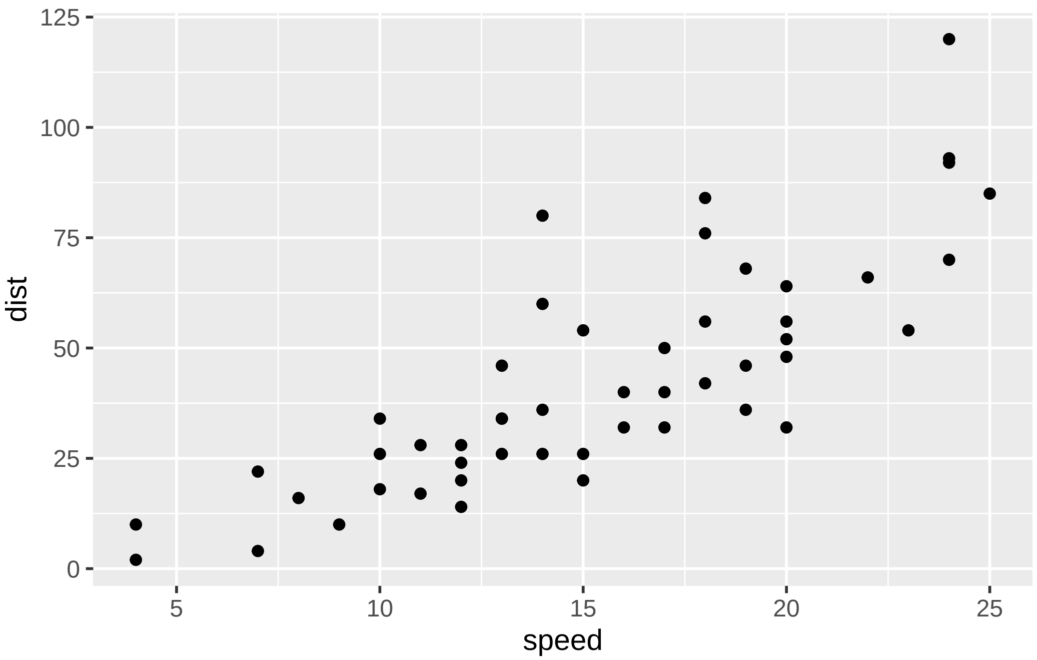
All ggplot figures have at least 3 things (called layers). These include:

-**data** (the input dataset I want to plot from), -**aes** (the aesthetic mapping of the data to my plot), -**geom** (the geom_point(), geom_line(), etc. that I want to draw)

> Q. Which geometric layer should be used to create scatter plots in ggplot2?

-geom_point()

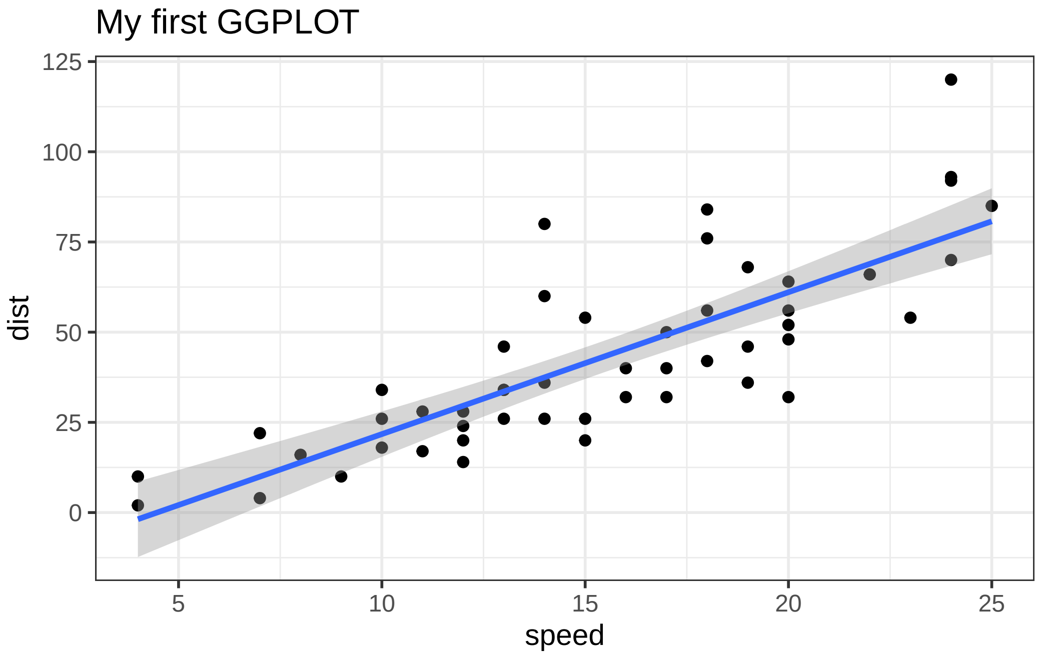
```r
ggplot(cars) + aes(x=speed, y=dist) + geom_point()
```



Let's add a line to show the relationship here:

```
ggplot(cars) + aes(x=speed, y=dist) + geom_point() + geom_smooth(method="lm") + theme_bw() +
```

```
`geom_smooth()` using formula = 'y ~ x'
```



The code to read the dataset

```
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
```

> Q. Use the nrow() function to find out how many genes are in this dataset. What is your answer?

```
nrow(genes)
```

```
[1] 5196
```

> Q. Use the colnames() function and the ncol() function on the genes data frame to find out what the column names are (we will need these later) and how many columns there are. How many columns did you find?

```
colnames(genes)
```

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

```
ncol(genes)
```

```
[1] 4
```

> Q. Use the table() function on the State column of this data.frame to find out how many 'up' regulated genes there are. What is your answer?

```
table(genes$State)
```

```

      down unchanging         up 
        72       4997        127 
```

> Q. Using your values above and 2 significant figures. What fraction of total genes is up-regulated in this dataset?

```
round(table(genes$State)/nrow(genes), 2)
```

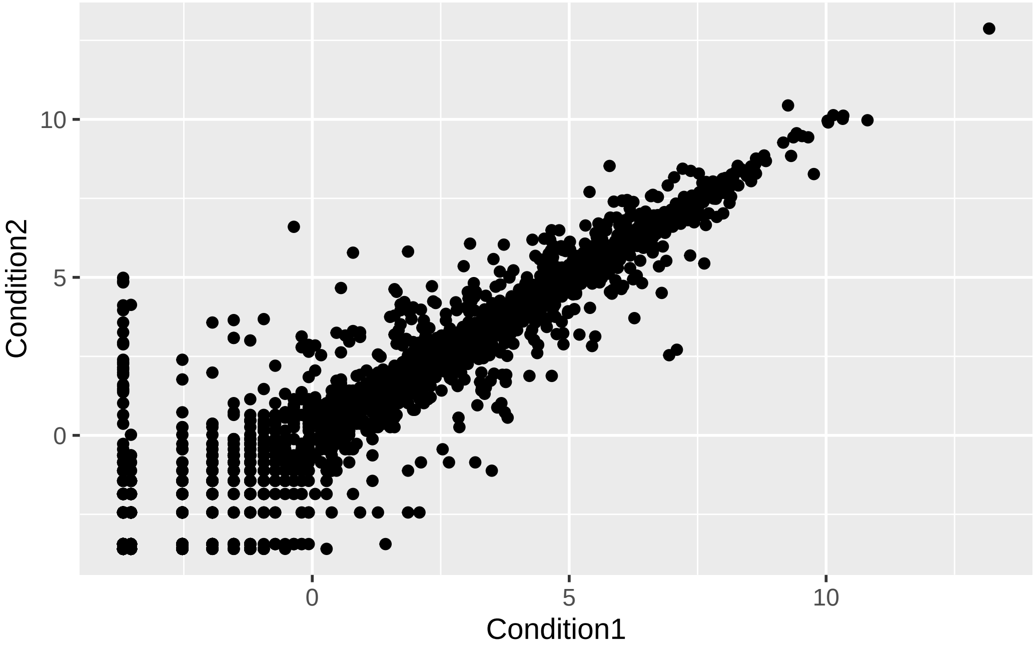
```

      down unchanging         up 
      0.01       0.96       0.02 
```

```
round(table(genes$State)/nrow(genes) * 100, 2)
```

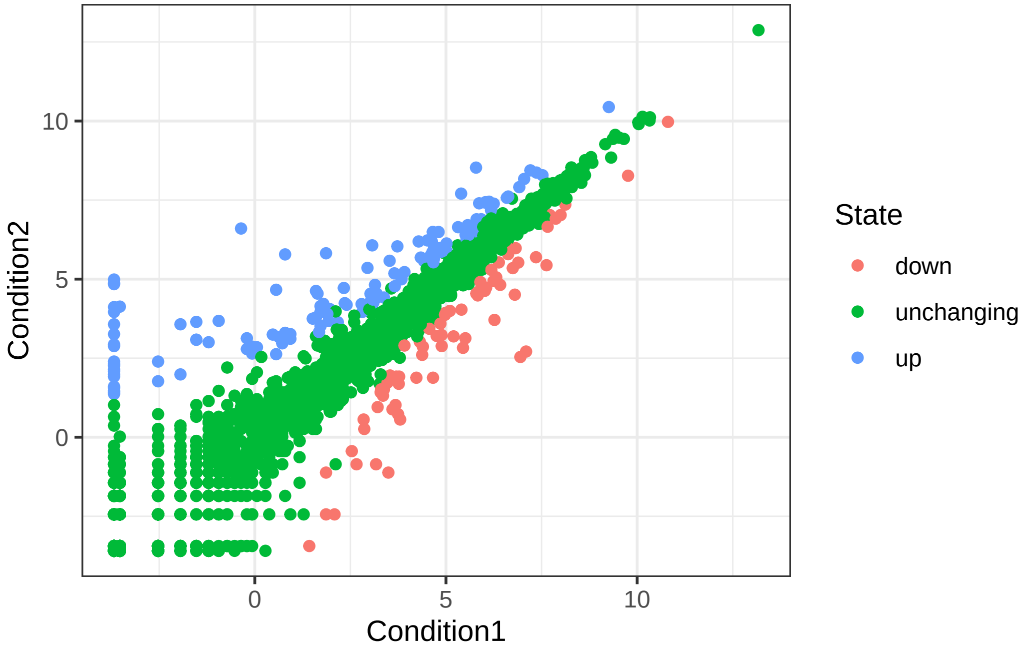
```

      down unchanging         up 
      1.39      96.17       2.44 
```

A first plot of this dataset

```
ggplot(genes) +
  aes(x=Condition1, y=Condition2) +
  geom_point()
```

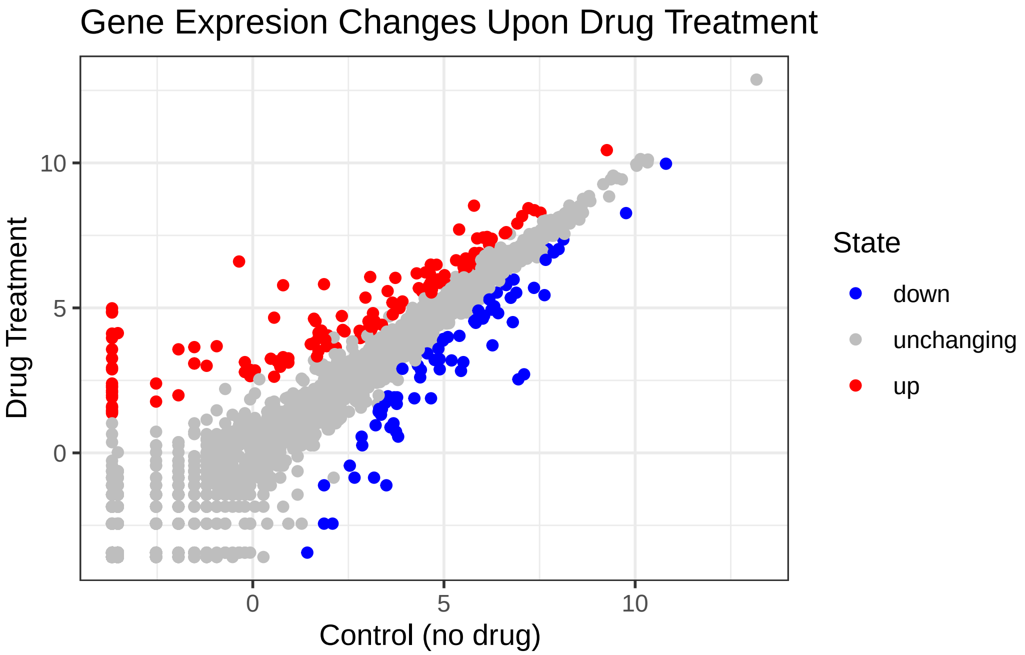


```
ggplot(genes) +
  aes(x=Condition1, y=Condition2, col=State) +
  geom_point() +
  theme_bw()
```



```
p <- ggplot(genes) +
  aes(x=Condition1, y=Condition2, col=State) +
  geom_point() +
  theme_bw()
```

```
p + scale_colour_manual(values=c("blue","gray","red")) +
    labs(title="Gene Expresion Changes Upon Drug Treatment",
         x="Control (no drug) ",
         y="Drug Treatment")
```



```
library(gapminder)
```

```
Warning: package 'gapminder' was built under R version 4.3.3
```

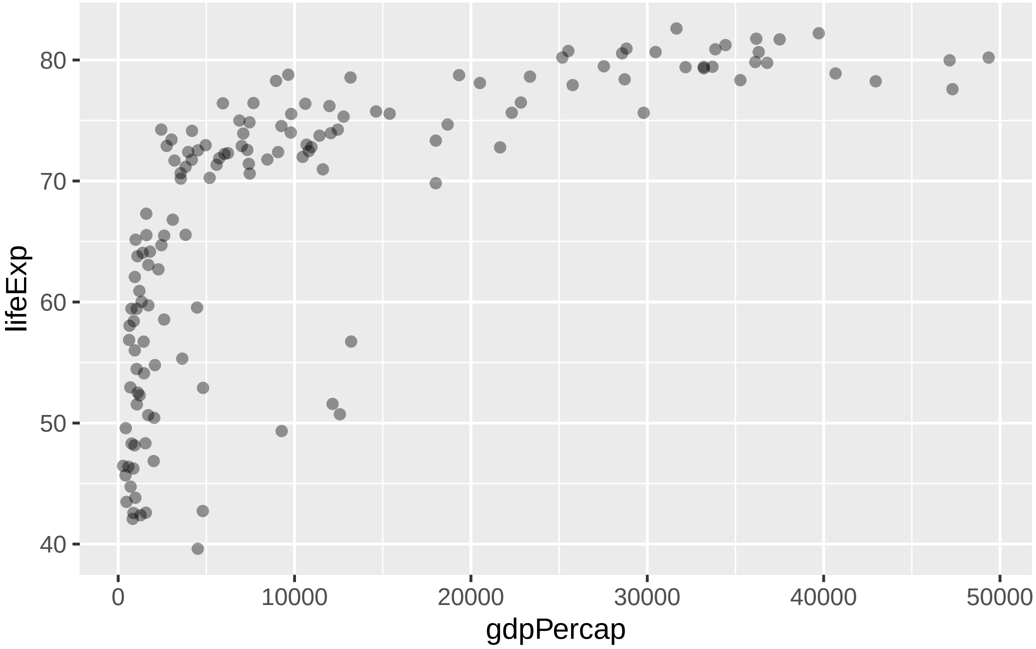
```
url <- "https://raw.githubusercontent.com/jennybc/gapminder/master/inst/extdata/gapminder.tsv

gapminder <- read.delim(url)
```

```
library(dplyr)
```

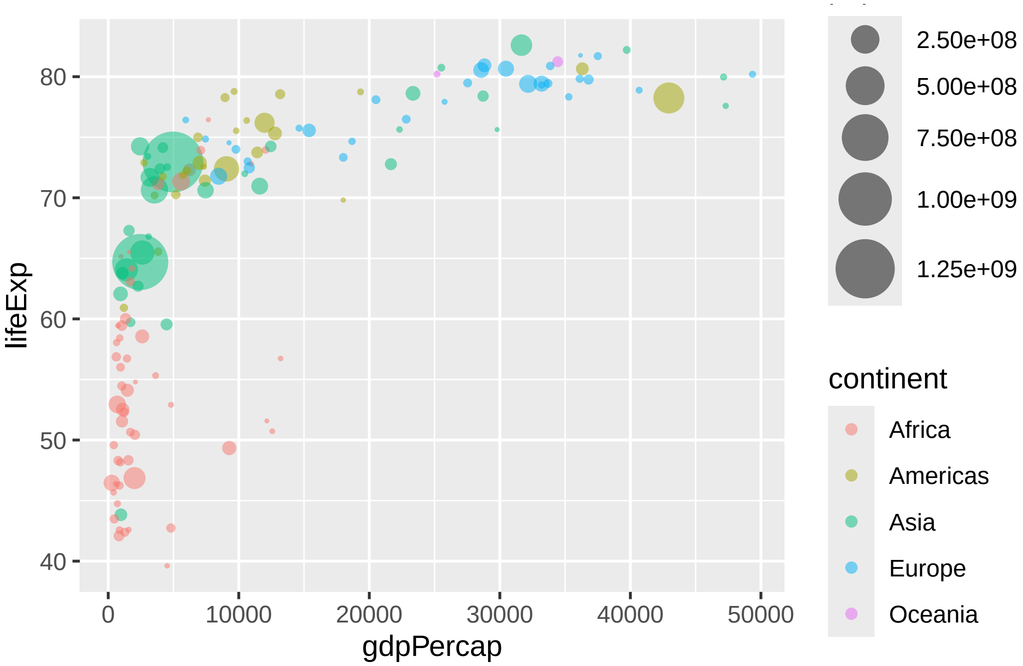
```

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
```

```r
gapminder_2007 <- gapminder %>% filter(year==2007)
```

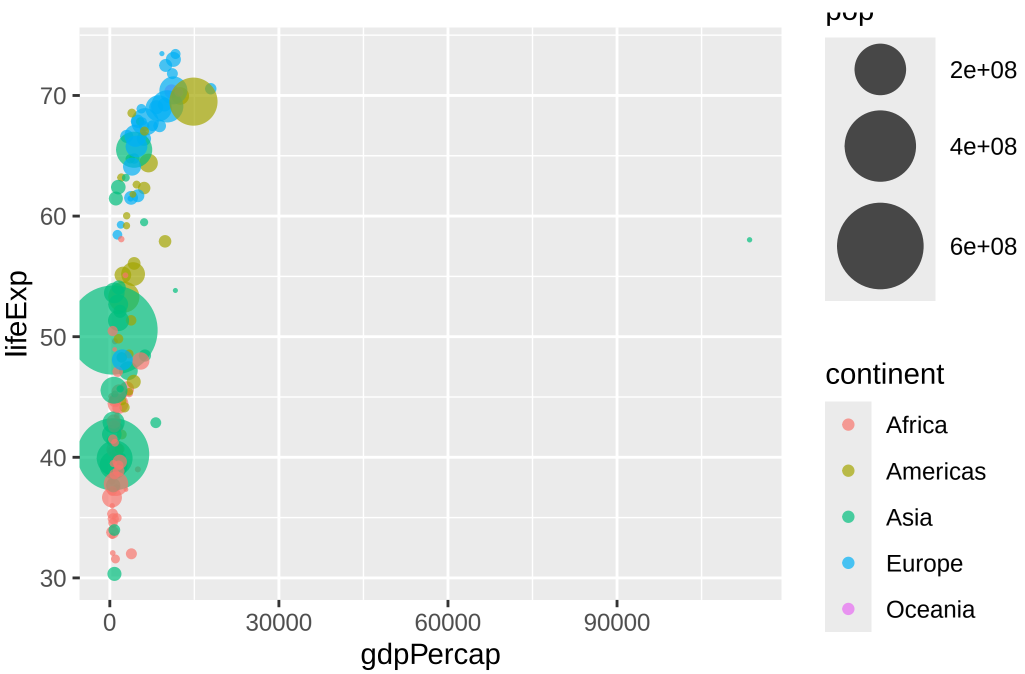
```r
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp) +
  geom_point(alpha=0.4)
```



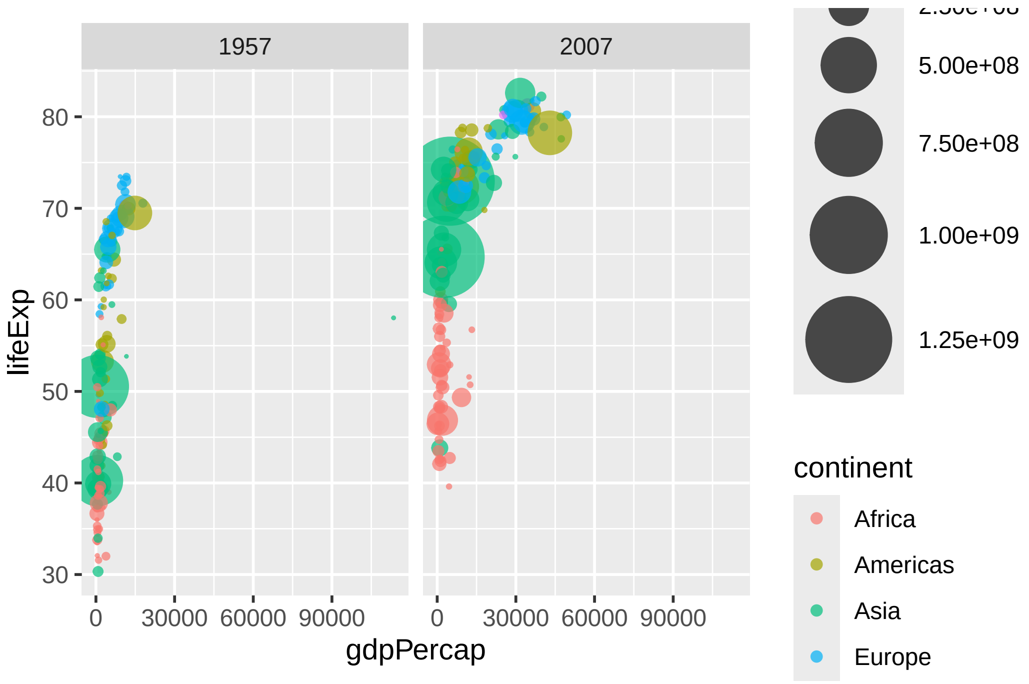
```r
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp, color=continent, size=pop) +
  geom_point(alpha=0.5)+
  scale_size_area(max_size = 10)
```



```
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=15)
```



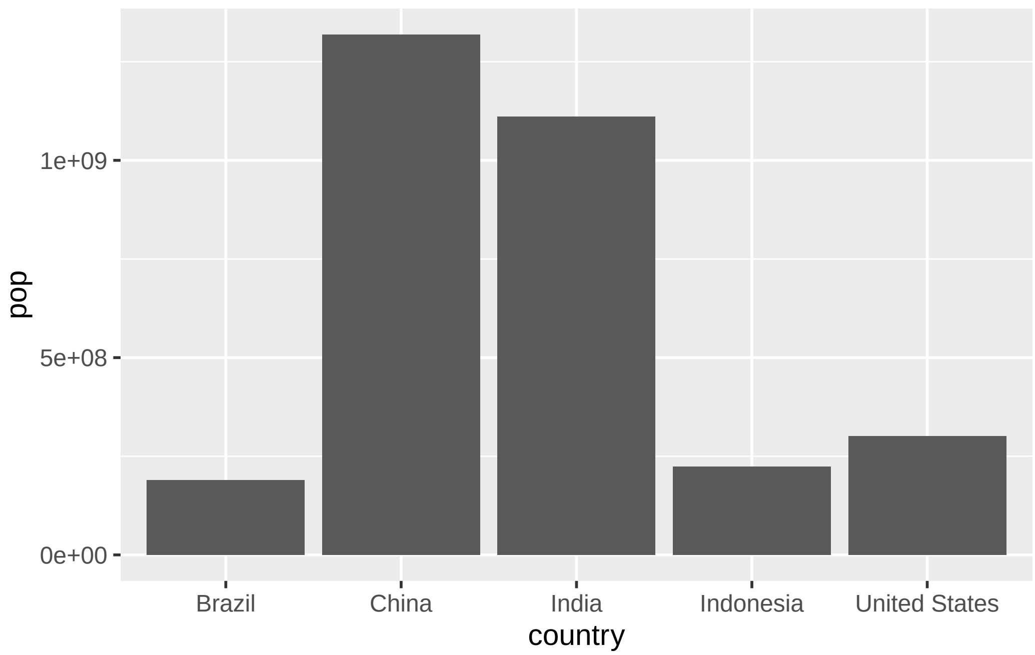
```
gapminder_1957 <- gapminder %>% filter(year==1957 | year==2007)
ggplot(gapminder_1957) +
  aes(x=gdpPercap, y=lifeExp, color=continent, size=pop) +
  geom_point(alpha=0.7) +
  scale_size_area(max_size=15) +
  facet_wrap(~year)
```



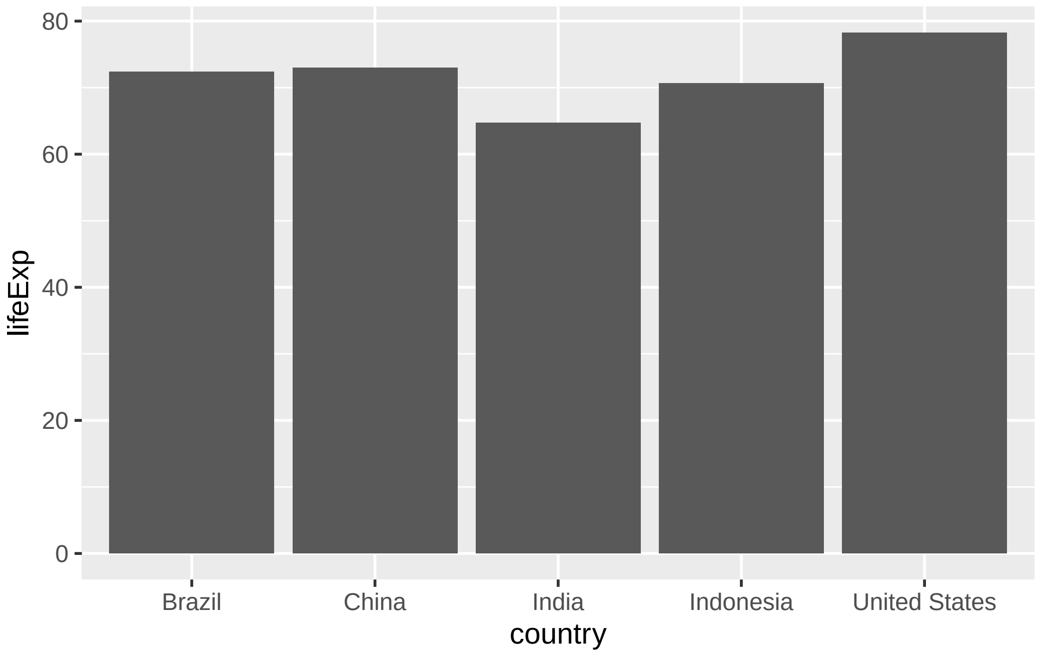
```
gapminder_top5 <- gapminder %>%
  filter(year==2007) %>%
  arrange(desc(pop)) %>%
  top_n(5, pop)

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
```

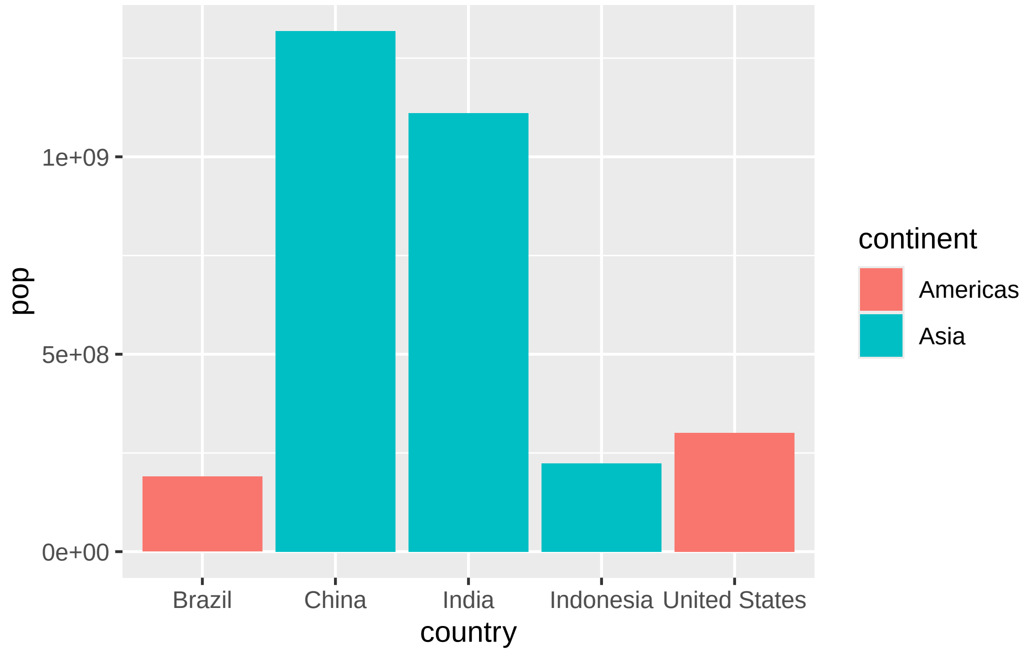
```
ggplot(gapminder_top5) +
  geom_col(aes(x = country, y = pop))
```



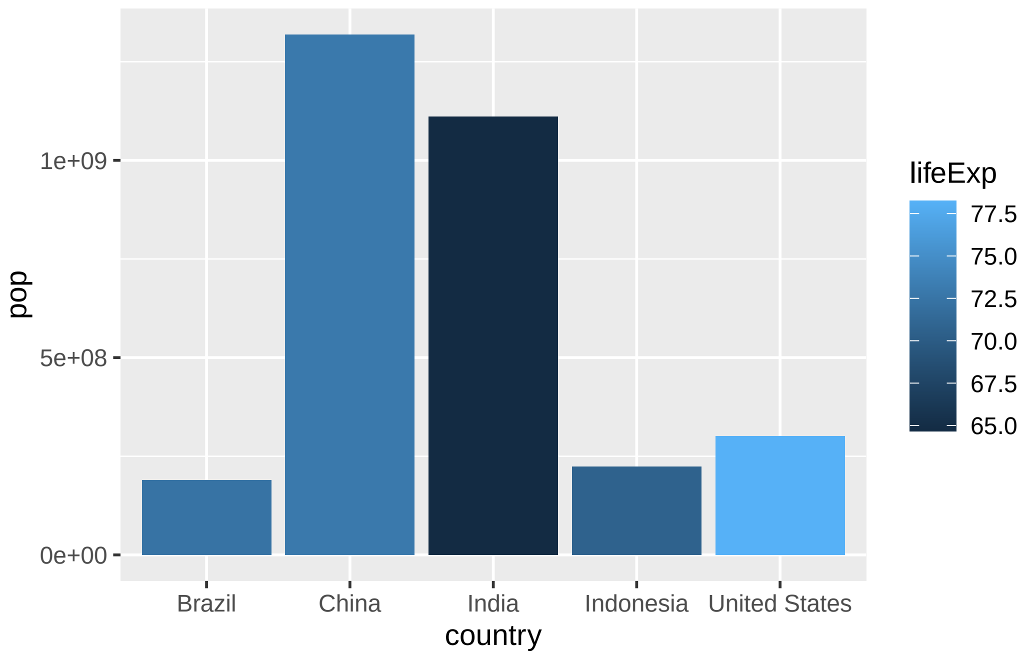
```r
ggplot(gapminder_top5) +
  geom_col(aes(x = country, y = lifeExp))
```



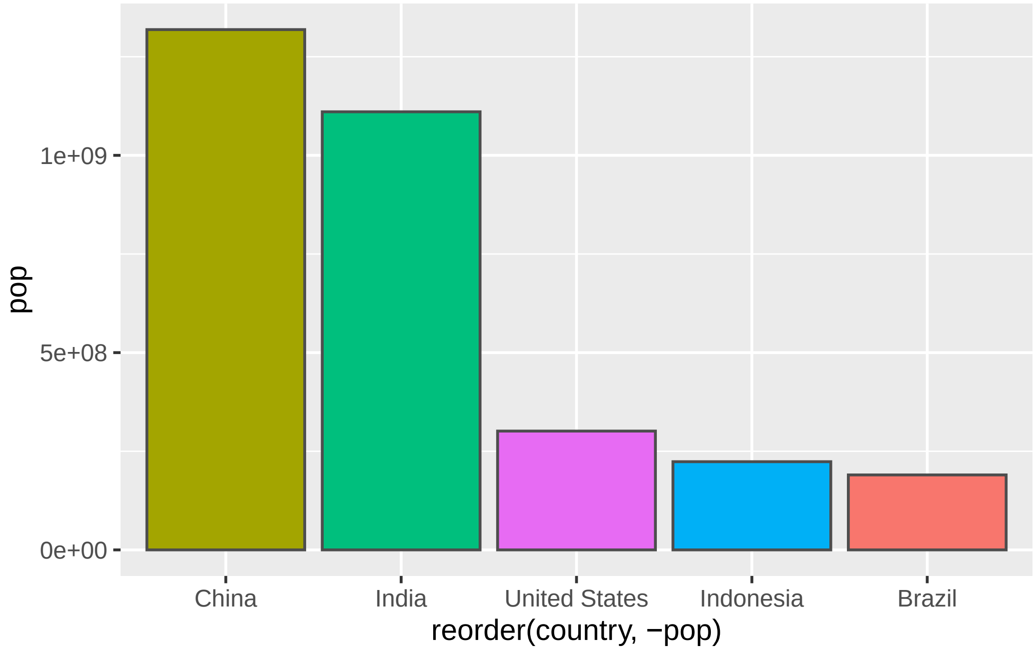
```
ggplot(gapminder_top5) +
  geom_col(aes(x = country, y = pop, fill = continent))
```



```
ggplot(gapminder_top5) +
  geom_col(aes(x = country, y = pop, fill = lifeExp))
```



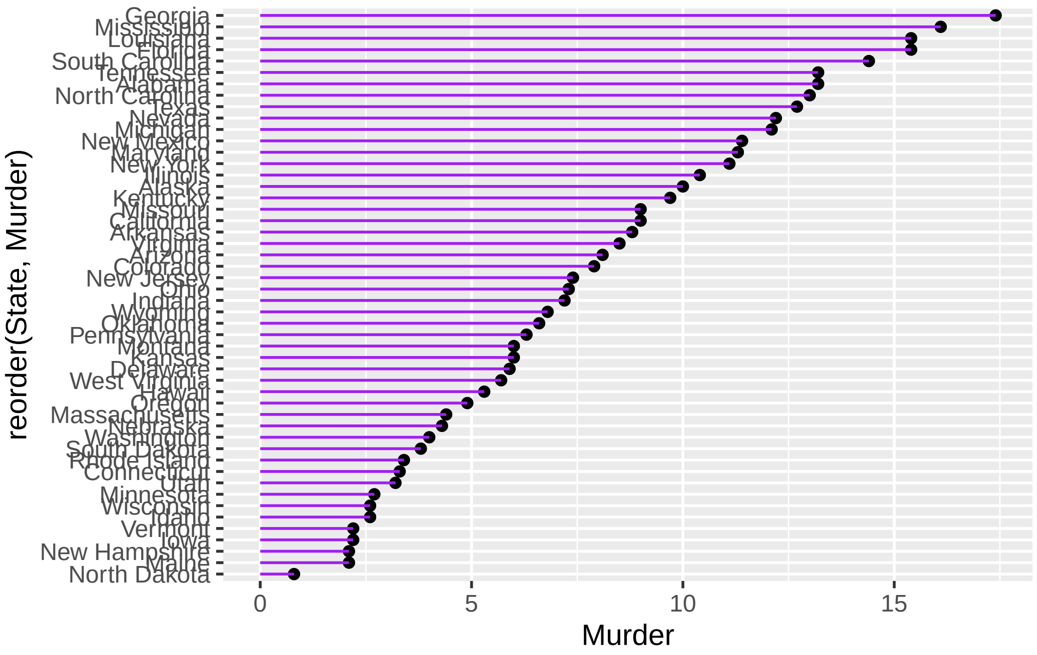
```
ggplot(gapminder_top5) +
  aes(x=reorder(country, -pop), y=pop, fill=country) +
  geom_col(col="gray30") +
  guides(fill="none")
```



```
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_point() +
  geom_segment(aes(x=State,
                   xend=State,
                   y=0,
                   yend=Murder), color="purple") +
  coord_flip()
```



```r
library(gapminder)
library(gganimate)
```

```
Warning: package 'gganimate' was built under R version 4.3.3
```

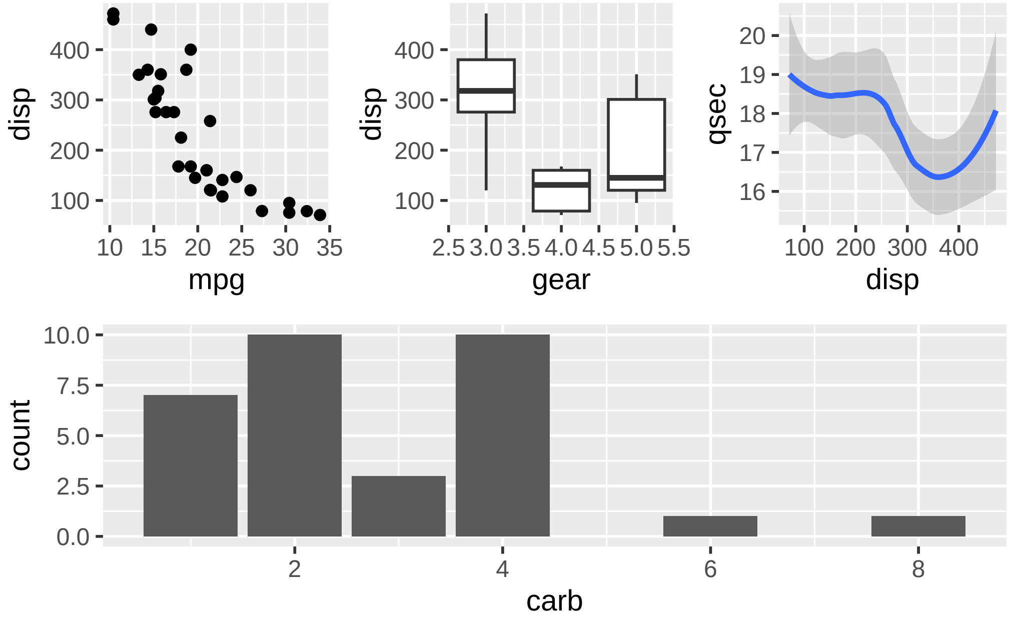
```r
#ggplot(gapminder, aes(gdpPercap, lifeExp, size = pop, colour = country)) +
#  geom_point(alpha = 0.7, show.legend = FALSE) +
#  scale_colour_manual(values = country_colors) +
#  scale_size(range = c(2, 12)) +
#  scale_x_log10() +

#  facet_wrap(~continent) +

#  labs(title = 'Year: {frame_time}', x = 'GDP per capita', y = 'life expectancy') +
#  transition_time(year) +
#  shadow_wake(wake_length = 0.1, alpha = FALSE)
```

```r
library(patchwork)
```

```
Warning: package 'patchwork' was built under R version 4.3.3
```

```r
# 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'
```


disp
mpg
disp
gear
qsec
disp
count
carb