Class05: Data Viz with GGPLOT

Henry Li (A16354124)

Today we are gonna play with plotting and graphics in R.

There are lots of ways to make cool figures in R.

The most popular package is `ggplot2`, which is part of the `tidyverse` collection of packages. –copilot

There is "base" R graphics (`plot()`, `hits()`, `boxplot()`, etc.).

There is also add-on packages like **ggplot**

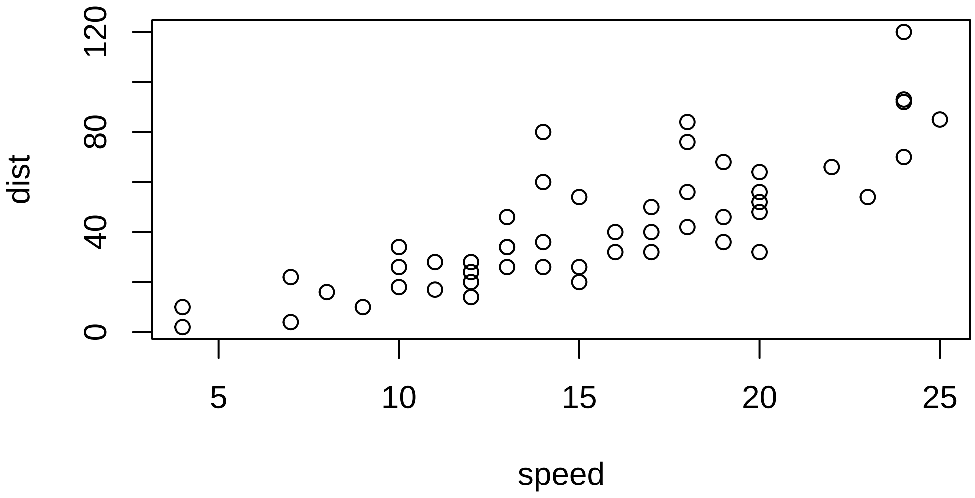
To insert any code, click three 'back-tick's to mark the start of the code, or just click the green button above, that is to the left of the button `Run`.

```
head(cars,3)
```

```
  speed dist
1     4    2
2     4   10
3     7    4
```

Let's plot this with "base" R: *option + command + I* would directly insert a code block.

```
plot(cars)
```

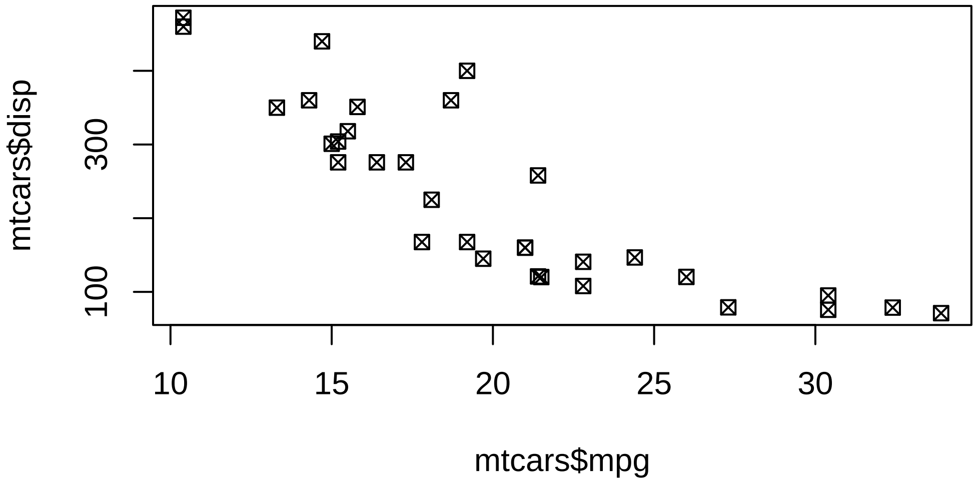


```
head(mtcars)
```

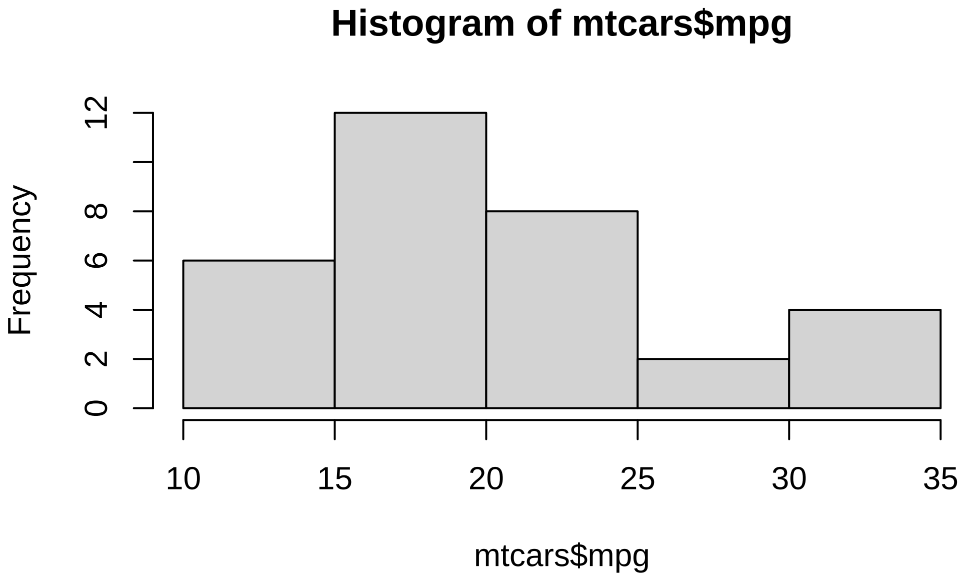
```
                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

Let’s plot `mpg` vs `disp`: `pch` is plotting character, each representing a symbol.

```
plot(mtcars$mpg, mtcars$disp, pch=7)
```



```
hist(mtcars$mpg)
```



GGPLOT (the pkg itself is called ggplot2)

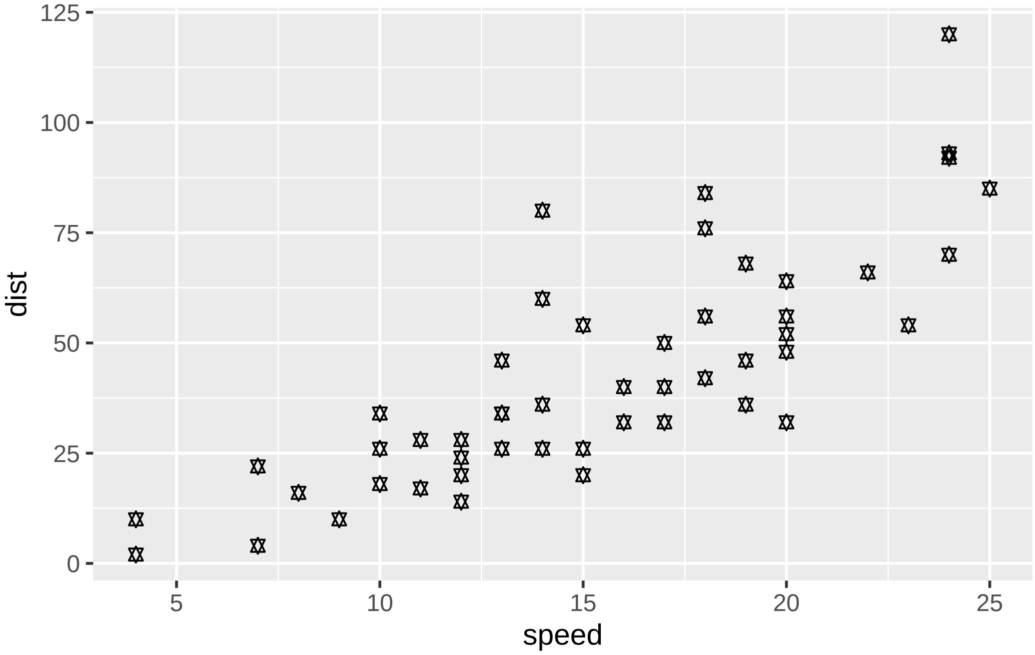
The main function in the ggplot2 package is **ggplot()**. First, I need to install the **ggplot2** package. To do this, I can install any package with the function `install.packages("ggplot2")`.

Then I need to load the package with `library(ggplot2)`.

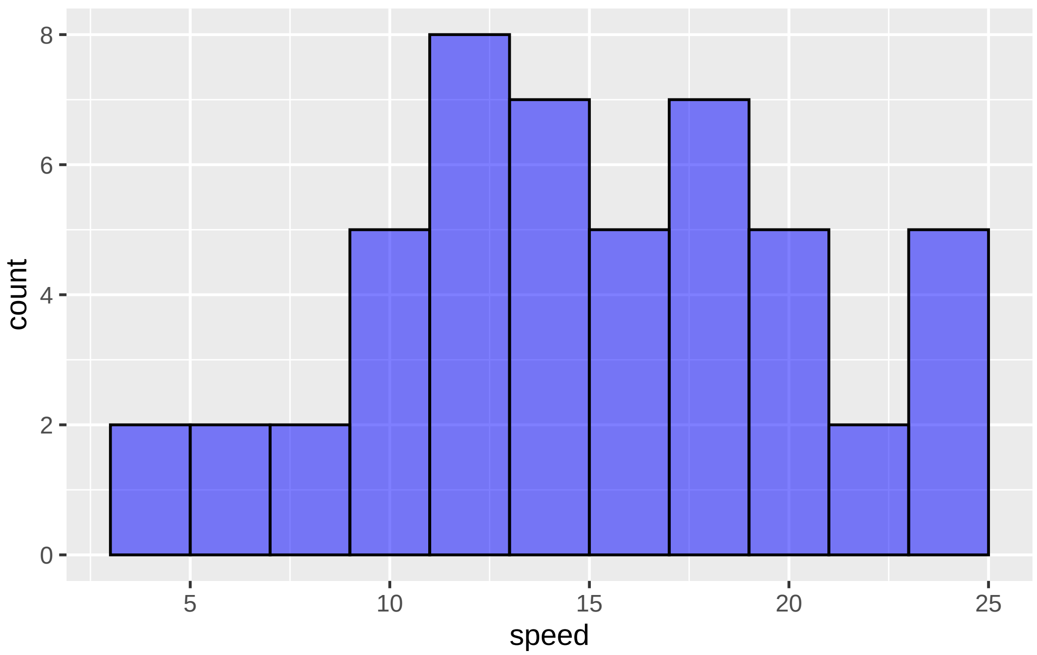
> **N.B** I never want to run `install.packages()` in my quarto source document!! If I do that, every time I render the document, it will try to install the package again. I only need to install a package once on my computer.

`library()` loads the package into my current R session.

```
library(ggplot2)
# simply ggplot(cars) would not work. We need data, aesthetics, and geometries.
ggplot(cars)+
  aes(x=speed, y=dist)+
  geom_point(pch = 11)
```



```
ggplot(cars) +
  aes(speed) +
  geom_histogram(binwidth = 2, fill="blue", color="black", alpha=0.5)
```



Every ggplot needs at least 3 things: - Data (given with ggplot(cars)) - Aesthetics (given with aes(x=speed, y=dist)) - Geometries (given with geom_point())

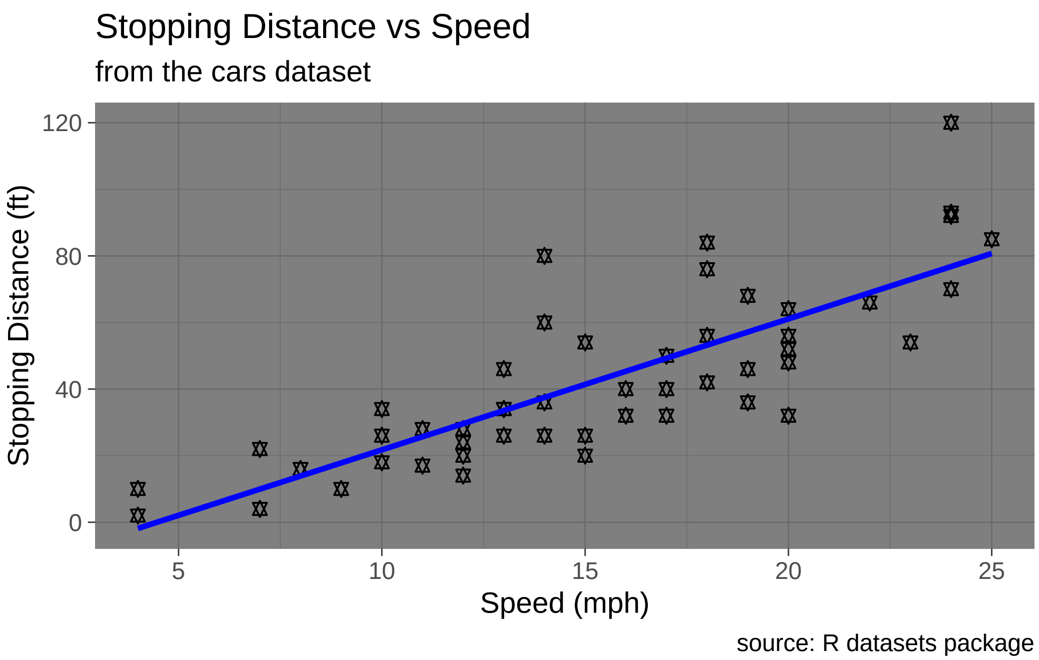
> For simple canned graphs “base” R is nearly always faster.

Adding more layers

Let’s add a line and a tile, subtitle, and caption as well as custom axis labels.

```
ggplot(cars)+
  aes(x=speed, y=dist)+
  geom_point(pch = 11)+
  geom_smooth(method = "lm", se=FALSE, color="blue")+
  labs(title = "Stopping Distance vs Speed",
       subtitle = "from the cars dataset",
       caption = "source: R datasets package",
       x = "Speed (mph)",
       y = "Stopping Distance (ft)")+
  theme_dark()
```

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


Stopping Distance vs Speed

from the cars dataset

source: R datasets package

Plot some 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
```

```
nrow(genes)
```

```
[1] 5196
```

```r
colnames(genes)
```

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

```r
ncol(genes)
```

```
[1] 4
```

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

```

      down unchanging         up 
        72       4997        127 
```

```r
sum(genes$State == "up")
```

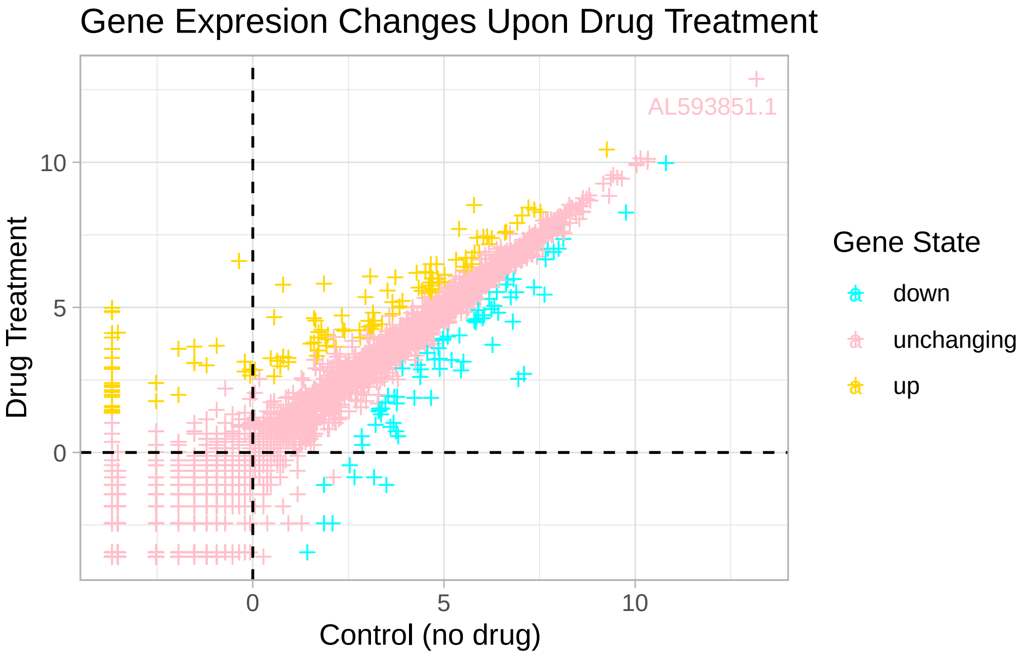
```
[1] 127
```

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

```

      down unchanging         up 
      1.39      96.17       2.44 
```

```r
p <- ggplot(genes) +
  aes(x=Condition1, y=Condition2, color=State) +
  geom_point(pch = 3) +
  theme_minimal() +
  geom_hline(yintercept = 0, linetype="dashed") +
  geom_vline(xintercept = 0, linetype="dashed") +
  labs(title="Gene Expression Changes",
       x="Condition 1",
       y="Condition 2",
       color="Gene State") +
  theme_light()
p <- p + scale_colour_manual(values=c("cyan","pink","gold")) +
    labs(title="Gene Expresion Changes Upon Drug Treatment",
         x="Control (no drug) ",
         y="Drug Treatment")
library(ggrepel)
p <- p + geom_text_repel(data=subset(genes, abs(Condition1) > 2 | abs(Condition2) > 2),aes(la
p
```

```
Warning: ggrepel: 4578 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



Going further

```
library(gapminder)
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
```

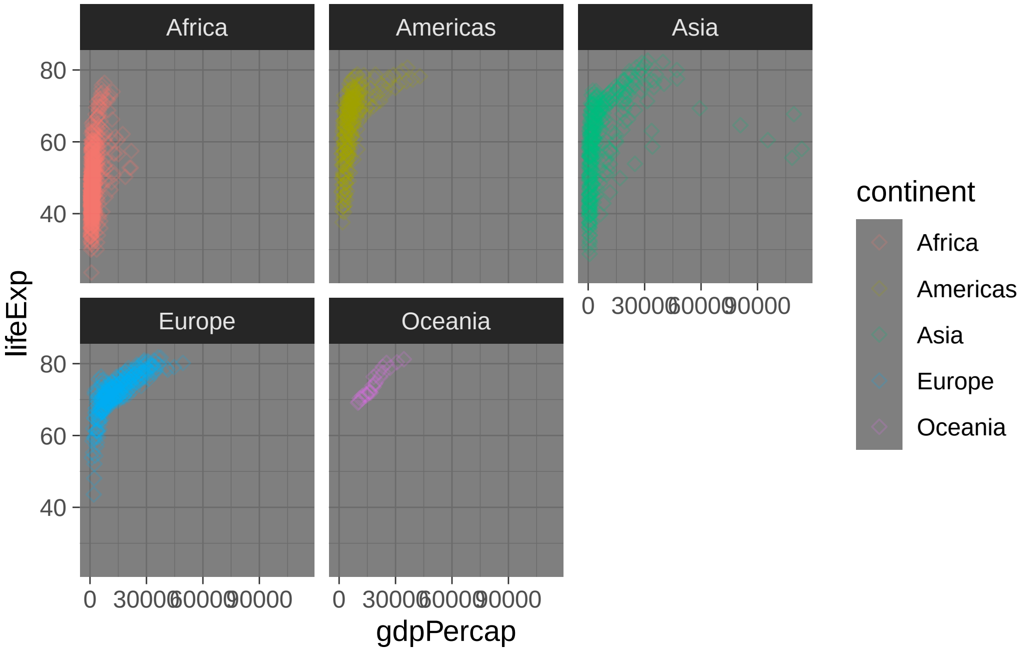
```
# File location online
url <- "https://raw.githubusercontent.com/jennybc/gapminder/master/inst/extdata/gapminder.tsv

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

```
      country continent year lifeExp      pop gdpPercap
1 Afghanistan      Asia 1952  28.801  8425333  779.4453
2 Afghanistan      Asia 1957  30.332  9240934  820.8530
3 Afghanistan      Asia 1962  31.997 10267083  853.1007
4 Afghanistan      Asia 1967  34.020 11537966  836.1971
5 Afghanistan      Asia 1972  36.088 13079460  739.9811
6 Afghanistan      Asia 1977  38.438 14880372  786.1134
```

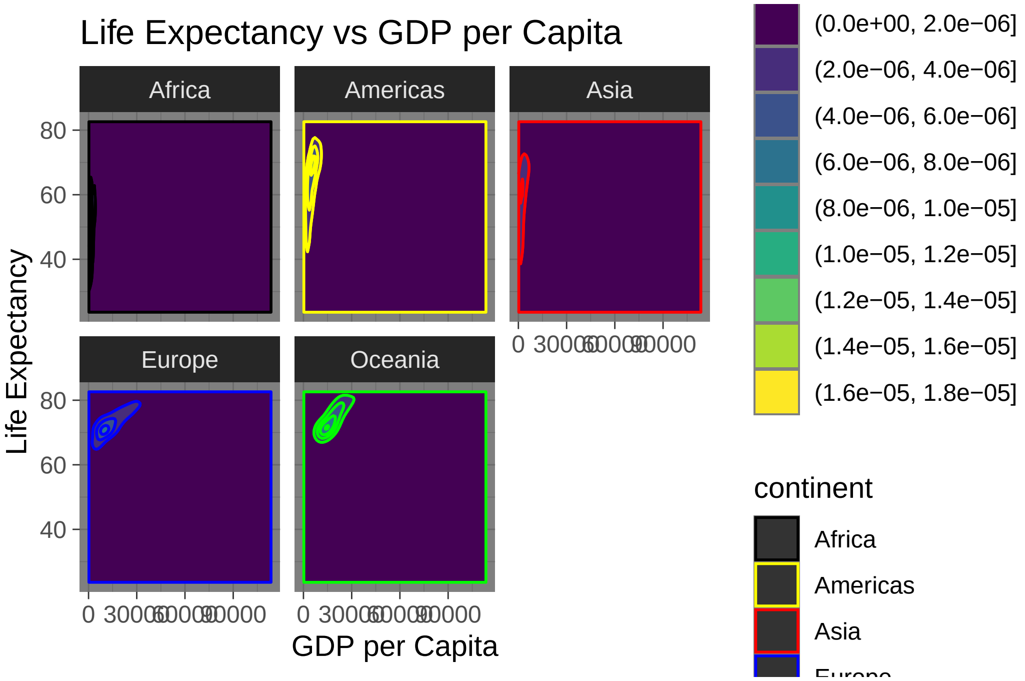
```
tail(gapminder)
```

```
      country continent year lifeExp      pop gdpPercap
1699 Zimbabwe    Africa 1982  60.363  7636524  788.8550
1700 Zimbabwe    Africa 1987  62.351  9216418  706.1573
1701 Zimbabwe    Africa 1992  60.377 10704340  693.4208
1702 Zimbabwe    Africa 1997  46.809 11404948  792.4500
1703 Zimbabwe    Africa 2002  39.989 11926563  672.0386
1704 Zimbabwe    Africa 2007  43.487 12311143  469.7093
```

```
ggplot(gapminder) +
  aes(y = lifeExp, x = gdpPercap, color = continent) +
  geom_point(pch = 5, alpha = 0.2)+
  facet_wrap(~continent) +
  theme_dark()
```

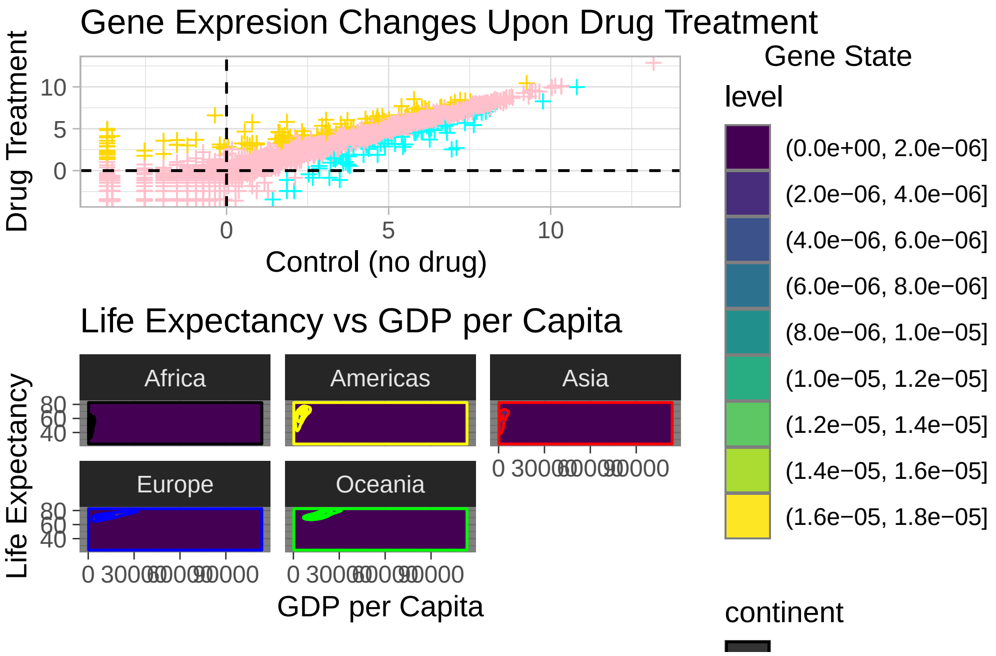


```
p3 <- ggplot(gapminder) +
  aes(y = lifeExp, x = gdpPercap, color = continent) +
  geom_density2d_filled()+
  facet_wrap(~continent) +
  scale_color_manual(values = c("black", "yellow", "red", "blue", "green")) +
  labs(title = "Life Expectancy vs GDP per Capita",
       x = "GDP per Capita",
       y = "Life Expectancy",) +
  theme_dark()
p3
```



```
library(patchwork)
options(repr.plot.width = 8, repr.plot.height = 10)
p / p3
```

```
Warning: ggrepel: 4579 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```


Gene Expresion Changes Upon Drug Treatment
Drug Treatment
10
5
0
0
5
10
Control (no drug)
Gene State
level
(0.0e+00, 2.0e−06]
(2.0e−06, 4.0e−06]
(4.0e−06, 6.0e−06]
(6.0e−06, 8.0e−06]
(8.0e−06, 1.0e−05]
(1.0e−05, 1.2e−05]
(1.2e−05, 1.4e−05]
(1.4e−05, 1.6e−05]
(1.6e−05, 1.8e−05]
Life Expectancy vs GDP per Capita
Africa
Americas
Asia
Europe
Oceania
Life Expectancy
80
60
40
0 30000 60000 90000
GDP per Capita
continent