class 05

Christopher Levinger, A17390693

Table of contents

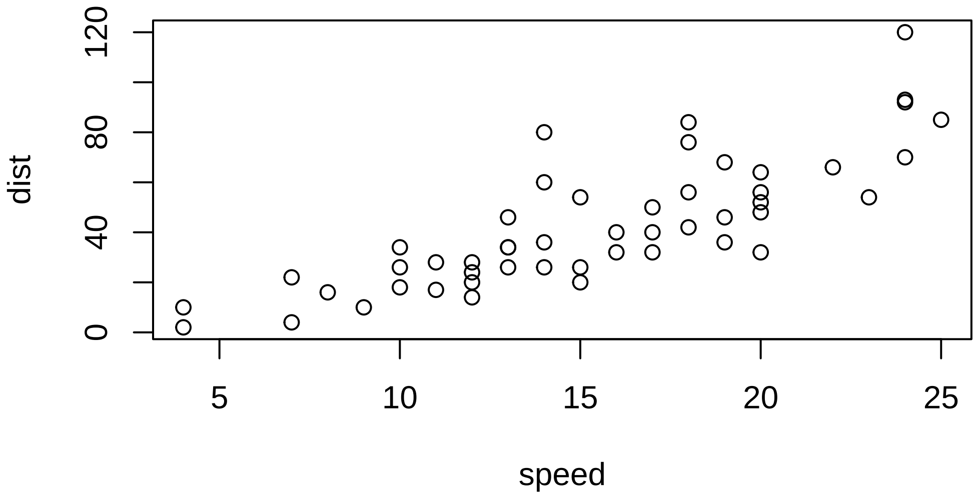
background

There are many graphics systems available in “R”. These include “base” R an tones of add on packages like **ggplot2**

Let’s compare “base” and **ggplot2** briefly. We can use some data that is built into R called cars.

In base R I can just call plot()

```
plot(cars)
```

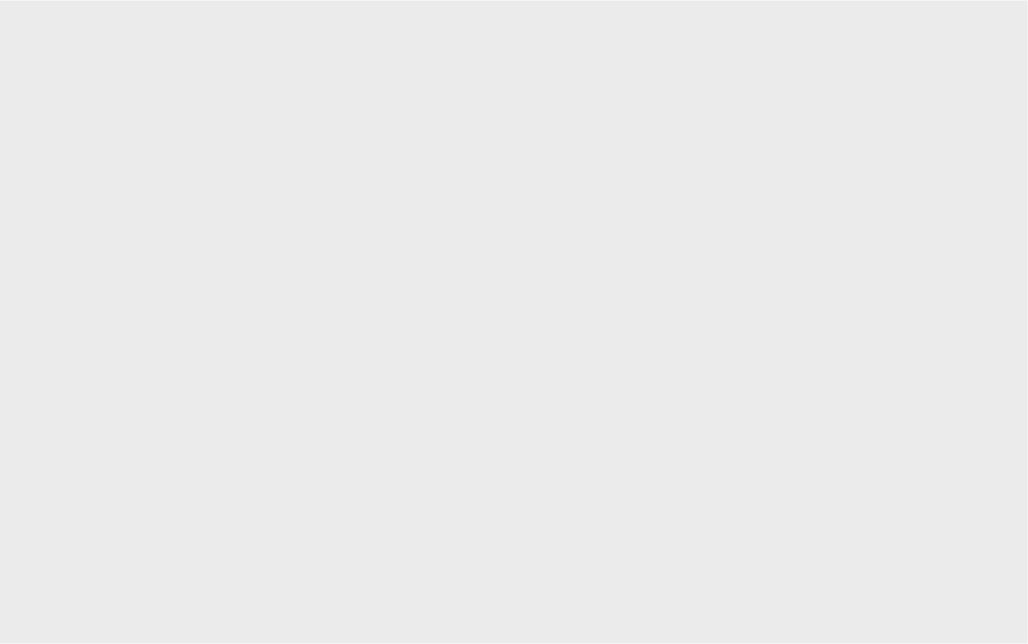


How can we do this with **ggplot2*

First we need to instal the package. We do this 'install.packages("ggplot2")'. I only need to do this once and then it will be available on my computer from then on.

> Key point: I only install packages in the R console not within quarto docs or R scripts. Before I use any add-on package I must load it up with a call to 'library()'

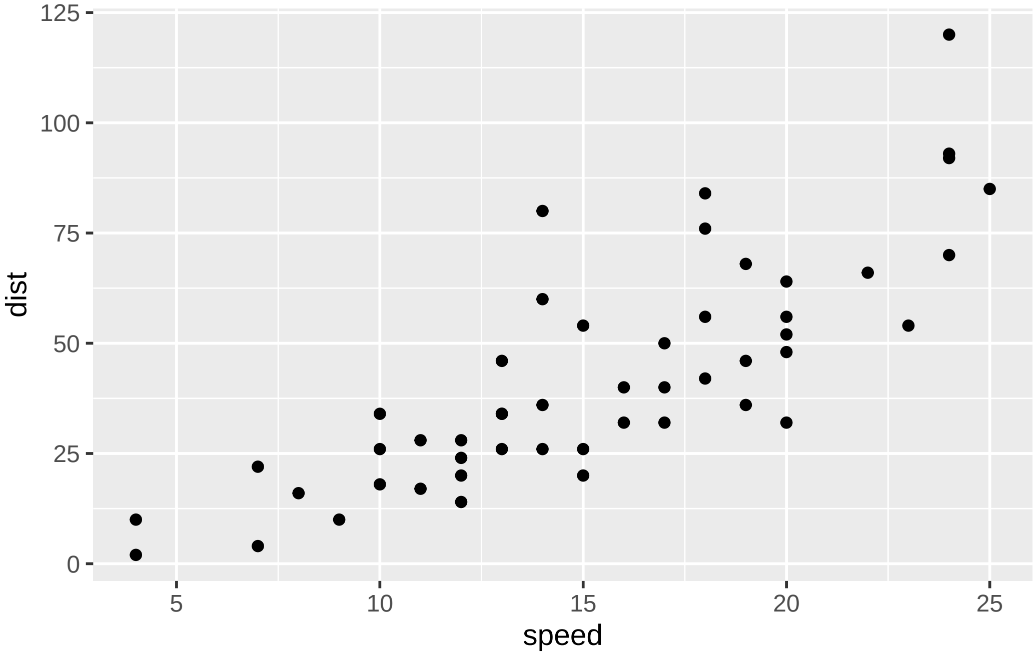
```
library(ggplot2)
ggplot(cars)
```



Every ggplot has at least 3 things:

- the **data** (in our case ‘cars’)
- the *aes*thetics (how the data map to the plot)
- the **geom**s that determine how the plot is drawn (lines, points, columns, etc.)

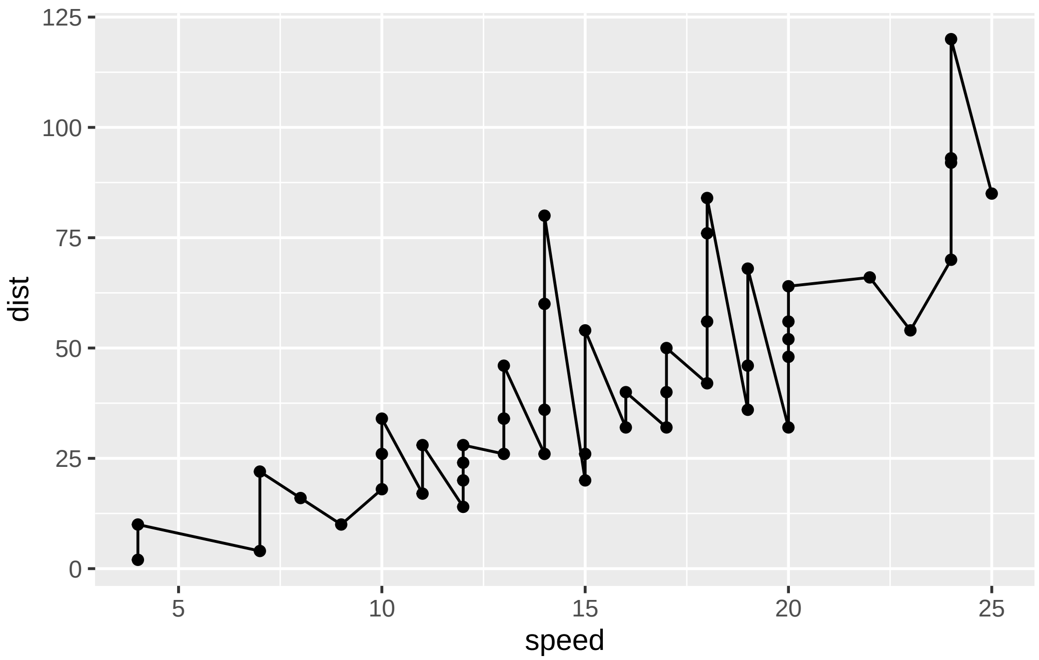
```r
ggplot(cars) +
 aes(x=speed, y=dist) +
 geom_point()
```



For “simple” plots ggplot is much more verbose than base R but the defaults are nicer and for complicated plots it becomes much more efficient and structured.

Add a line to show the relationship of speed to stopping distance (i.e. add another “layer”)

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

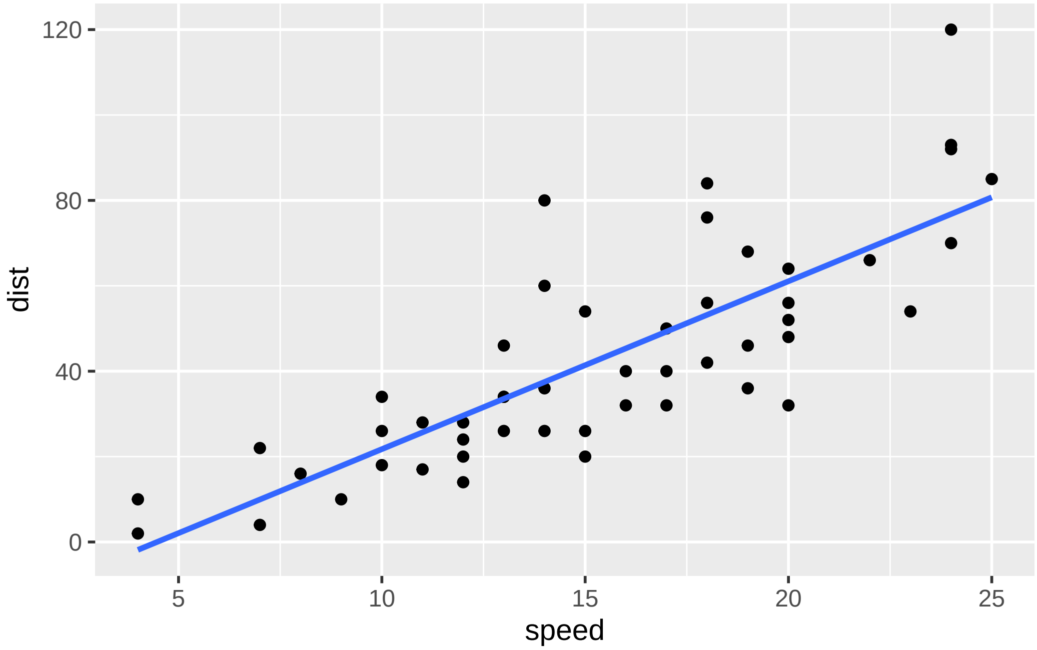


```
p <- ggplot(cars) +
  aes(x=speed, y=dist) +
  geom_point()+
  geom_smooth(se=FALSE, method="lm")
```

I can always save any ggplot object (i.e. plot)

```
p
```

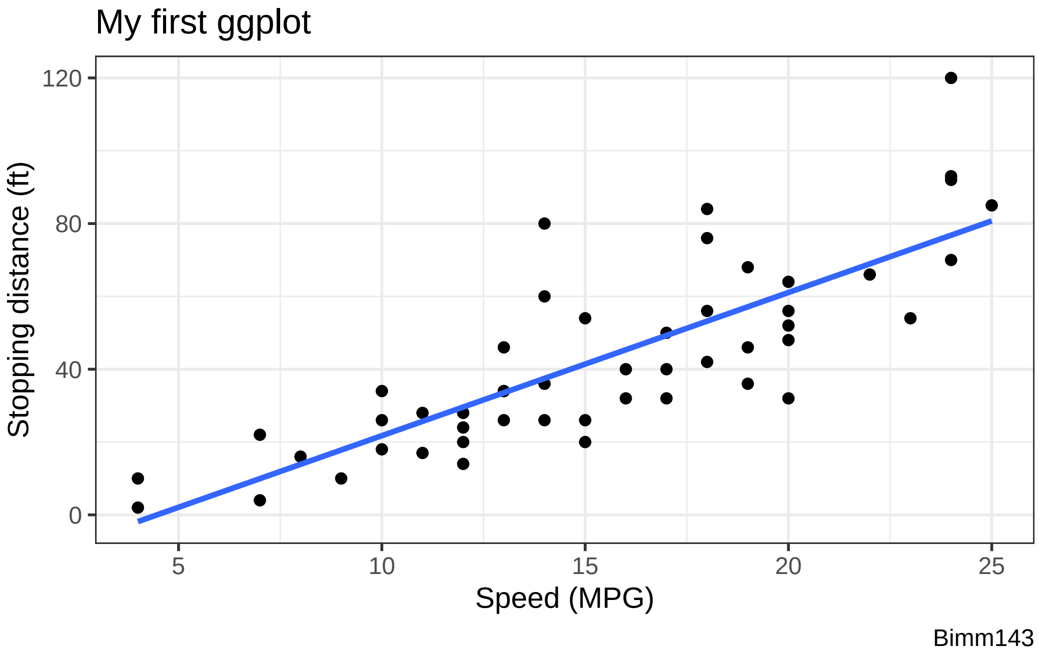
```
`geom_smooth()` using formula = 'y ~ x'
```



Q. Add a title and subtitle to the plot.

```
p + labs(title="My first ggplot",
         subtittle="Stopping Distance of old cars",
         caption = "Bimm143",
         x="Speed (MPG)",
         y="Stopping distance (ft)") +
  theme_bw()
```

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



Gene expression Plot

Read input data into R

```
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: how many genes are in this wee dataset?

```
nrow(genes)
```

```
[1] 5196
```

> Q: How many columns are there?

```
ncol(genes)
```

```
[1] 4
```

> Q: What are the column names?

```
colnames(genes)
```

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

> Q. How many “up” and “down” regulated genes are there?

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

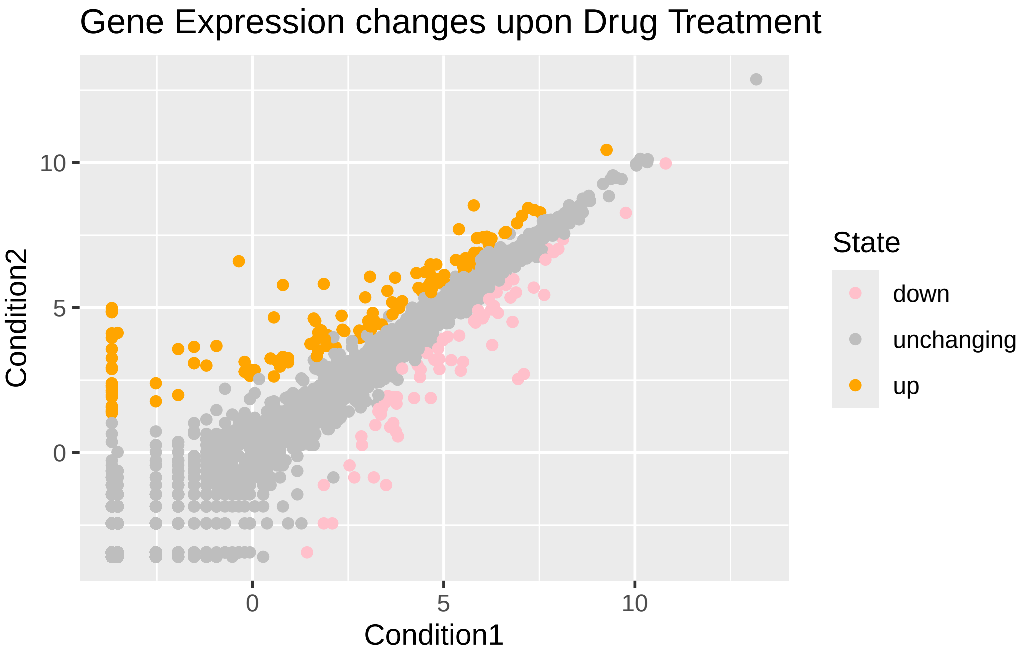
```

      down unchanging         up 
        72       4997        127 
```

Custom Color Plot

> Q. Make a first plot of this data

```
ggplot(genes) +
  aes(x= Condition1, y=Condition2, col=State) +
  scale_color_manual( values=c("pink","gray","orange")) +
  geom_point() +
  labs(title="Gene Expression changes upon Drug Treatment")
```



Using different aes and geoms

Let's plot some aspects of the in-built `mtcars` dataset.

```
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
```

Q. Scatterplot of `mpg` vs. `disp`

```
p1 <- ggplot(mtcars) +
  aes(x=disp, y=mpg) +
  geom_point()
```

Q. Boxplot of `gear` vs. `disp`

```
p2 <- ggplot(mtcars) +
  aes(gear, disp, group=gear) +
  geom_boxplot()
```

Q. Barplot of `carb`

```
p3 <- ggplot(mtcars) +
  aes(carb) +
  geom_bar()
```

Smooth of `disp` vs `qsec`

```
p4 <- ggplot(mtcars) +
  aes(disp, qsec) +
  geom_point() +
  geom_smooth()
```

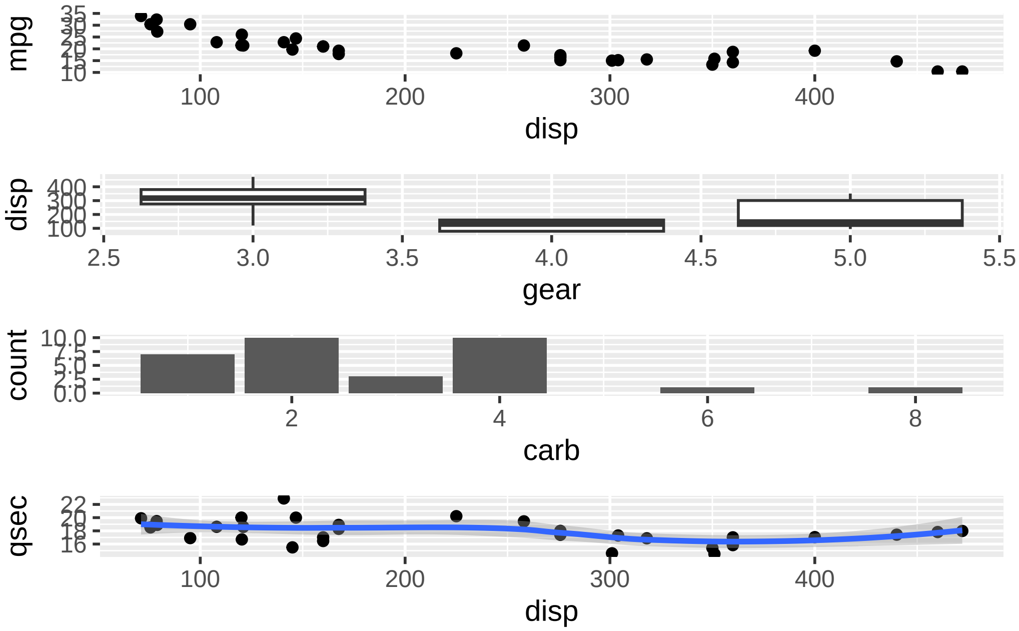
I want to combine all these plots into one figure with multiple pannels.

We can use the **patchwork** to do this.

```
library(patchwork)

((p1 / p2 / p3 / p4))
```

```
`geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



```r
ggsave(filename="myplot.png", width = 10, height = 10)
```

```
`geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```

```r
url <- "https://raw.githubusercontent.com/jennybc/gapminder/master/inst/extdata/gapminder.tsv
gapminder <- read.delim(url)
```

Faceting

```r
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
```

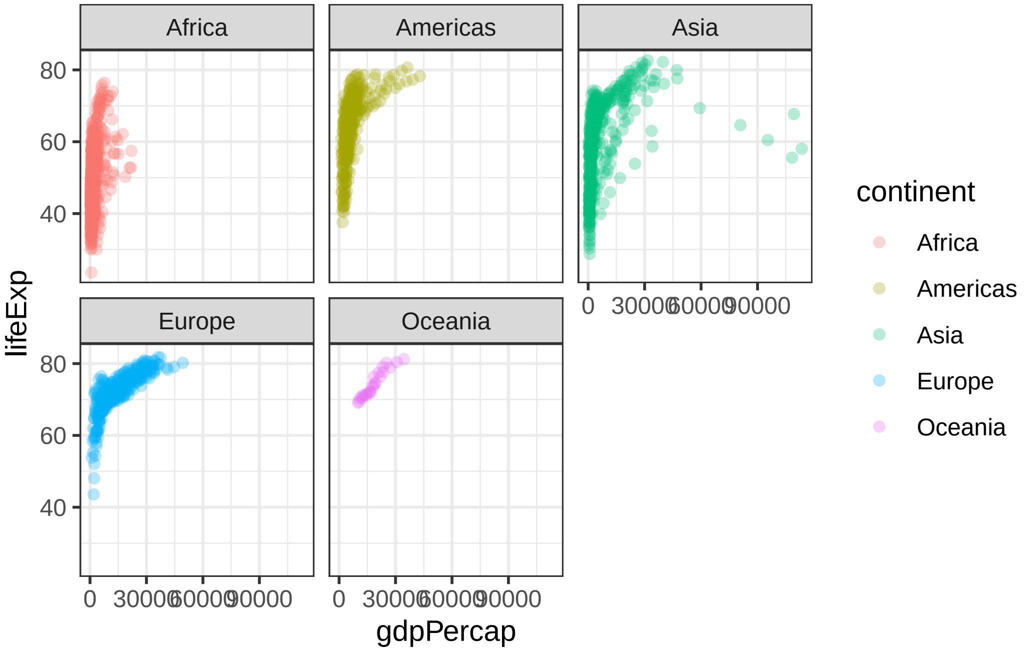
Q: How many countries are in this dataset?

```
length(table(gapminder$country) )
```

```
[1] 142
```

Q plot GDP vs. Life Expectancy colored by continent

```
ggplot(gapminder) +
  aes(x=gdpPercap, y=lifeExp, col=continent) +
  geom_point(alpha=0.3) +
  facet_wrap(~continent) +
  theme_bw()
```


Africa
Americas
Asia
Europe
Oceania
lifeExp
gdpPercap
continent
● Africa
● Americas
● Asia
● Europe
● Oceania