# Class 05: Data Visualization

AUTHOR
Hyeseung (Frankie) Son PID:16025601

## Basic R graphics vs ggplot2

There are many graphics systems available in R, including so-called "base" R graphics and the so-called **ggplot2** package.

To compare these let's play with the inbuilt cars dataset.

```
head(cars)
```

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

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

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

To use `ggplot2` package I first need to install it wih the function `install.packages("ggplot2")`.

I will run this in my R console (i.e. the R brain) as I do not want to re-install it every time I render my report

The main function in this package is called `ggplot()`. Can I just call it?

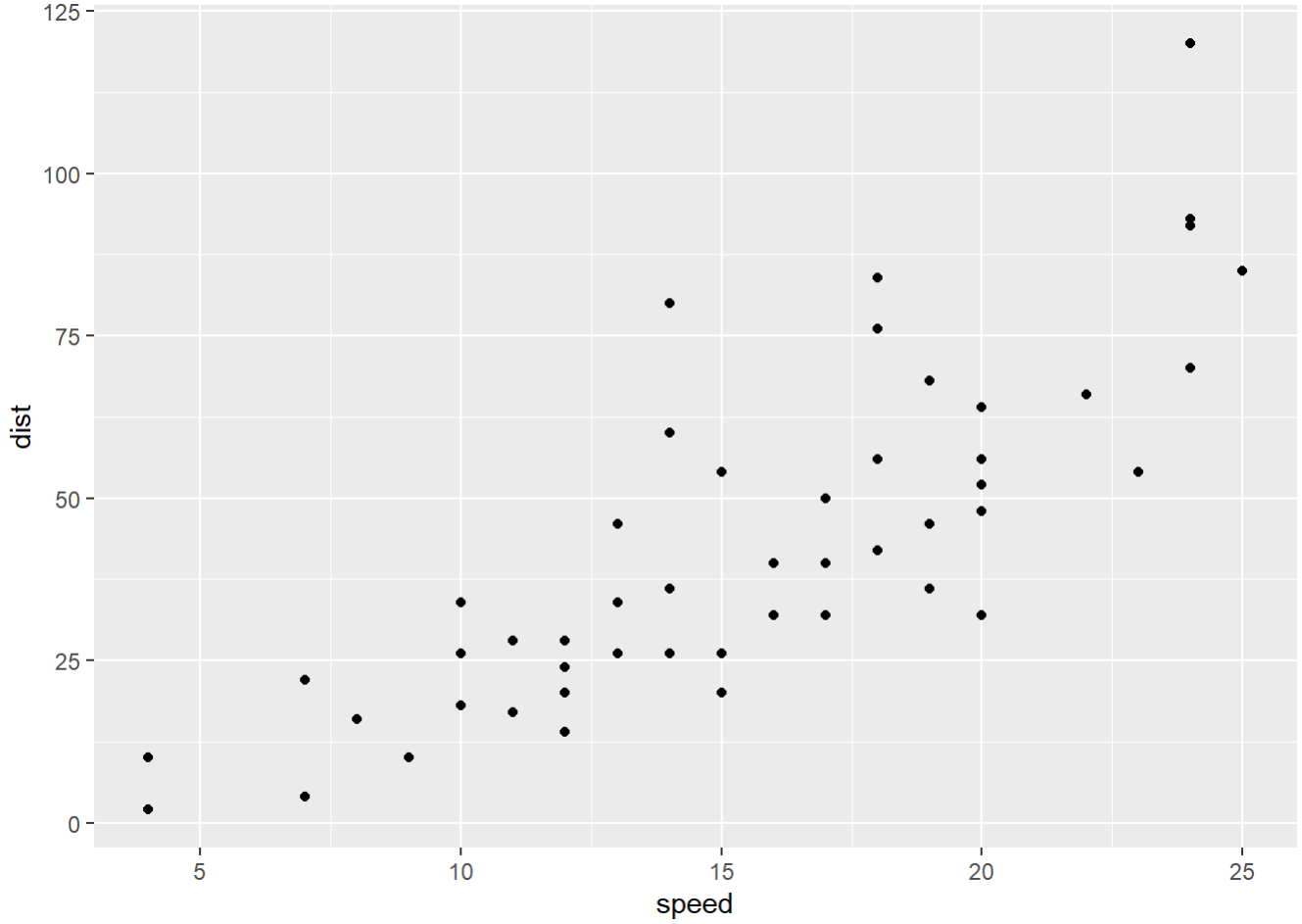
```
library(ggplot2)
```

```
Warning: package 'ggplot2' was built under R version 4.2.3
```

```
ggplot()
```

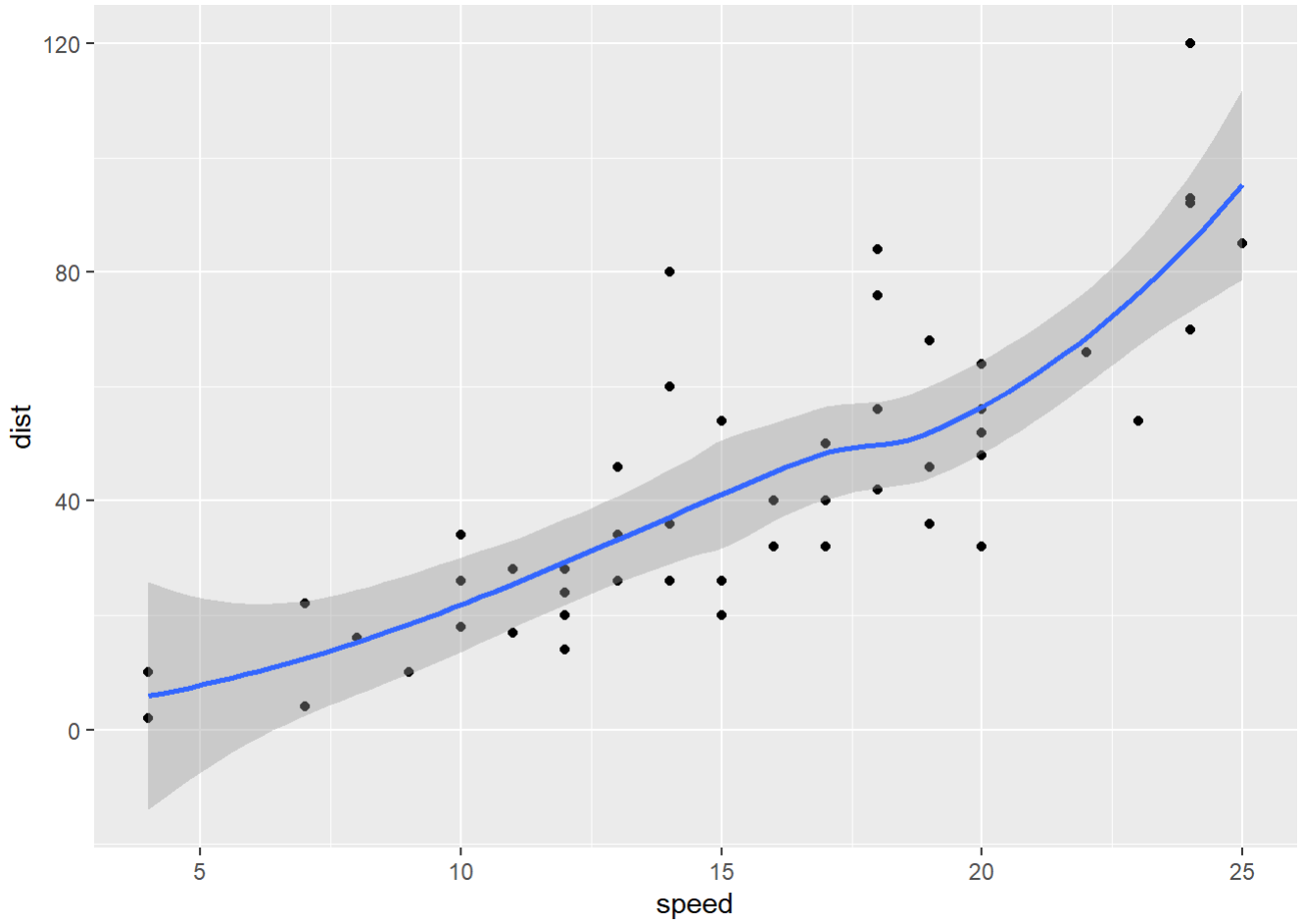
To make a figure with ggpot I need always at least 3 things: - data (i.e. what I want to plot) - aes: the aesthetic mapping of the data I want to plot - the geoms (i.e. how I want ot plot the data)

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




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

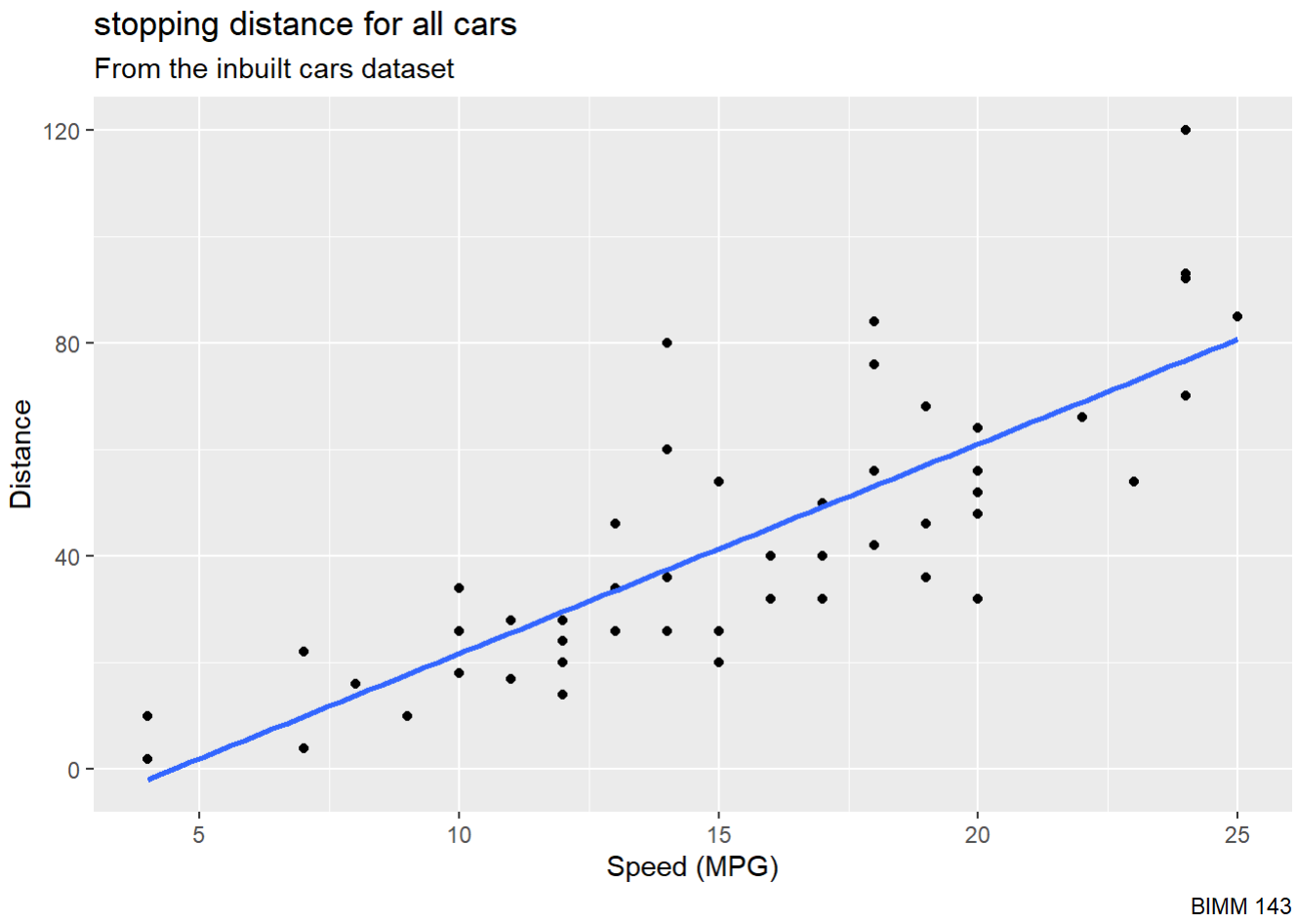
```
`geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```




If I want ot add more stuff, I can just keep adding layers such as

```
ggplot(data=cars)+
  aes (x=speed, y=dist)+
  geom_point()+
  geom_smooth(se=FALSE, method="lm")+
  labs(title="stopping distance for all cars", subtitle="From the inbuilt cars dataset", caption=
```

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




ggplot is much more verbose than base R plots but it has a consistent layer system that I can use to make just about any plot.

## A more complicated plot

Let's plot some gene 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
```

> Q. how many genes are in this dataset?

```
nrow(genes)
```

```
[1] 5196
```

> Q. How can we summarize the last column - the State column?

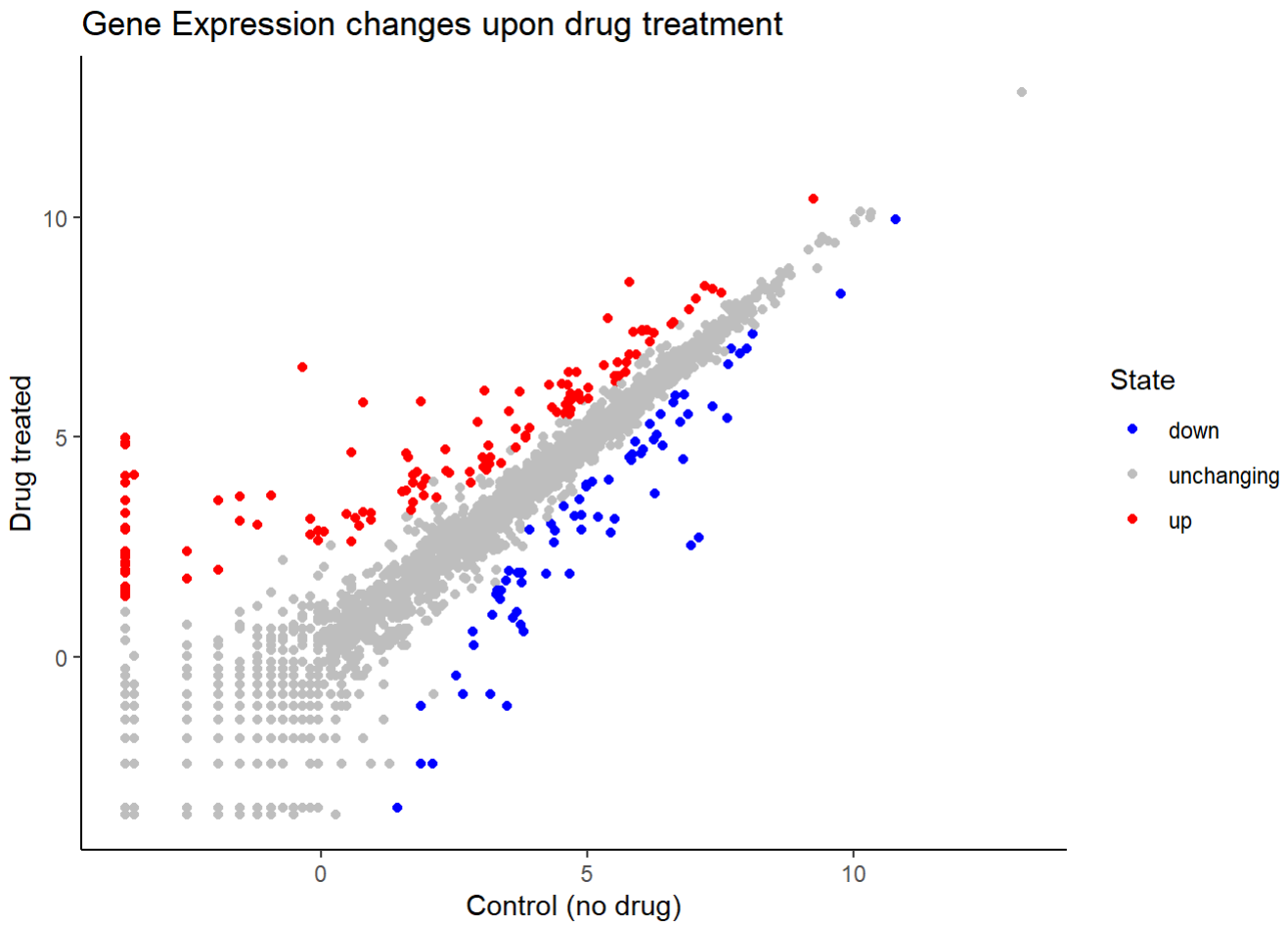
```
table(genes$State)
```

```
      down unchanging         up 
        72       4997        127 
```

```
library(ggplot2)
p <- ggplot(genes)+
  aes(x=Condition1, y=Condition2, color=State)+
  geom_point()+
  theme_classic()
```

I can now just call `p` when I want to plot or add to it

```
p + labs(title="Gene Expression changes upon drug treatment", x="Control (no drug)", y="Drug trea
  scale_colour_manual(values=c("blue", "gray", "red"))
```




##Going Further

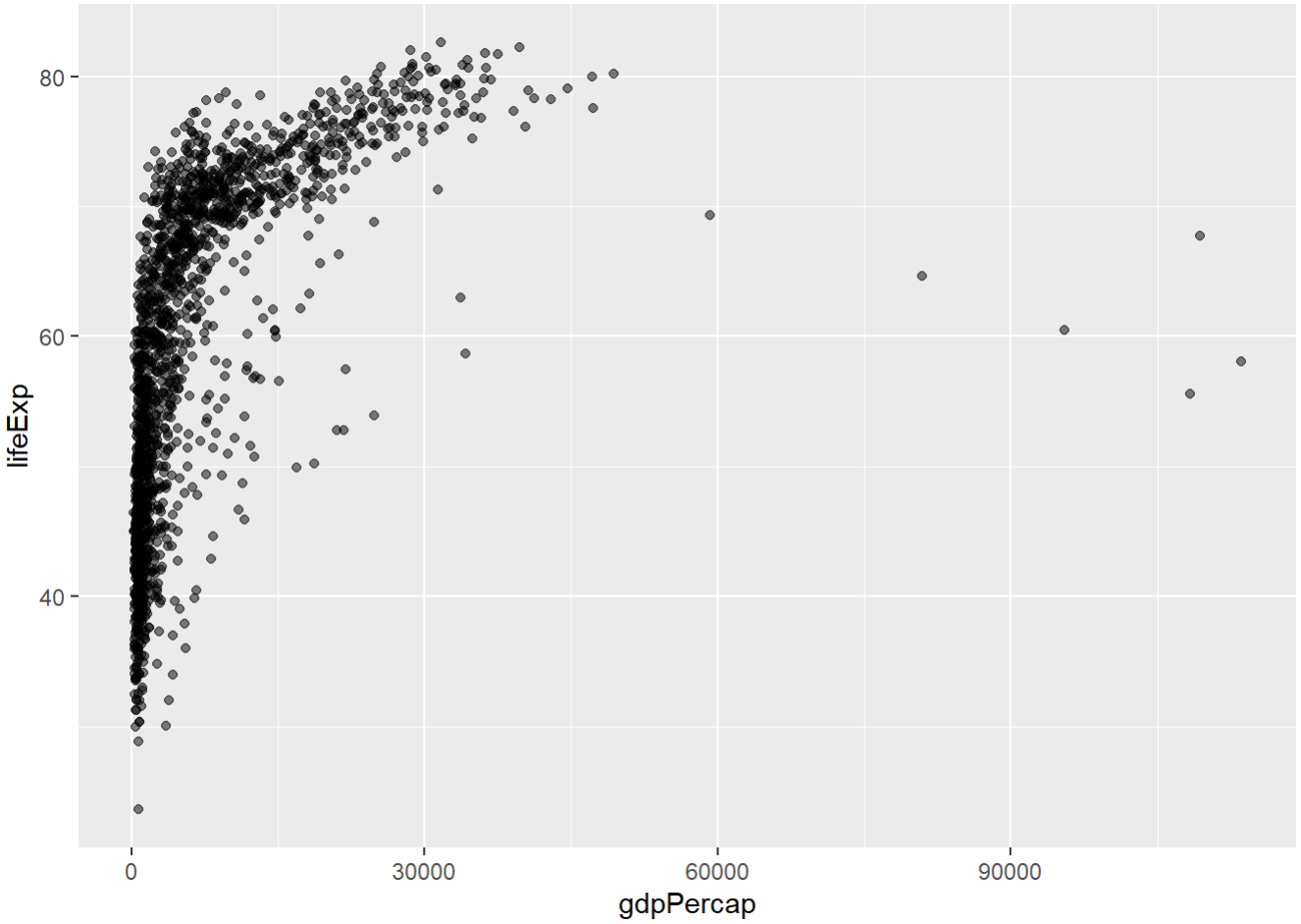
```
# File location online
url <- "https://raw.githubusercontent.com/jennybc/gapminder/master/inst/extdata/gapminder.tsv"

gapminder <- read.delim(url)

library(dplyr)
```

```
Warning: package 'dplyr' was built under R version 4.2.3
```

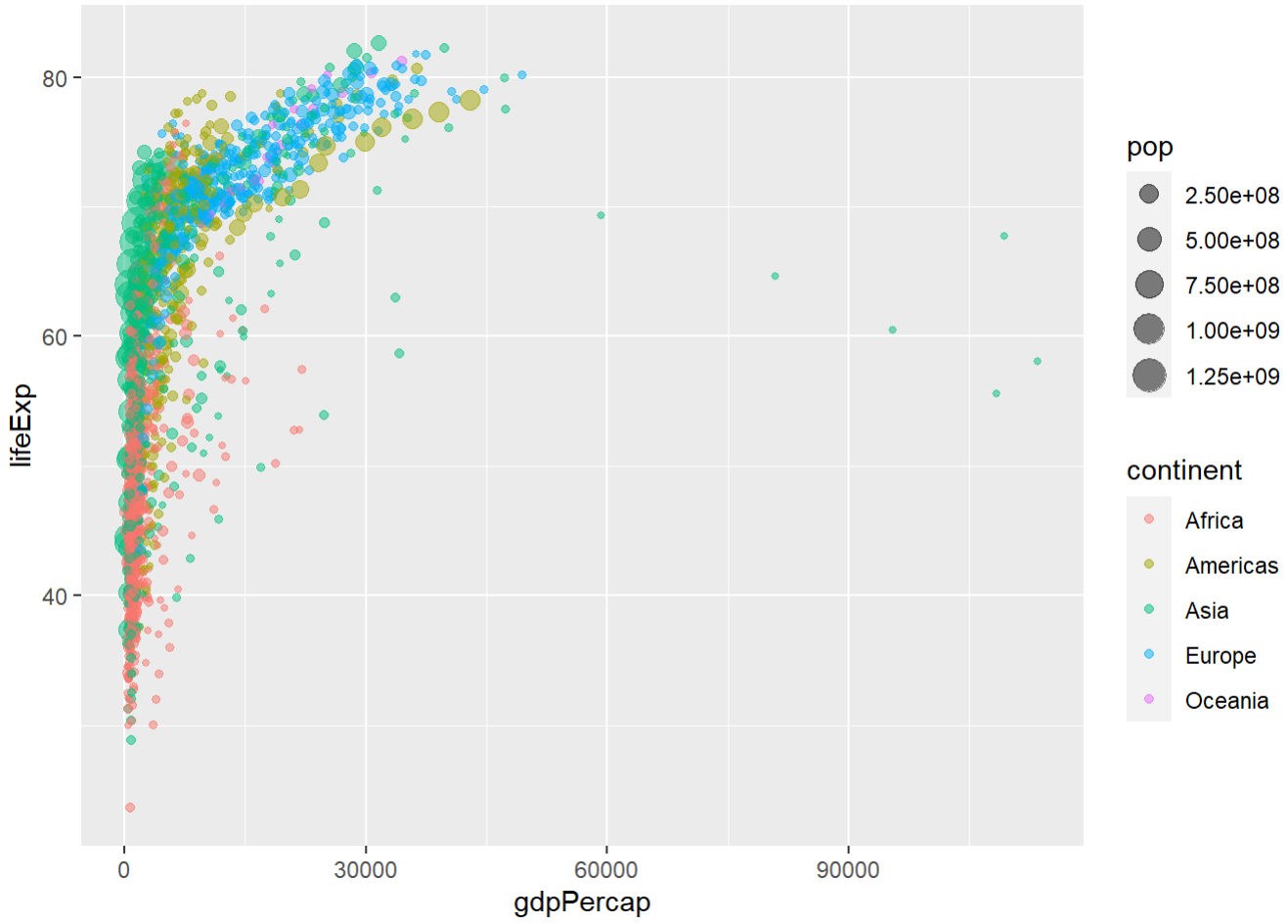
```
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
ggplot(gapminder) +
  aes(x=gdpPercap, y=lifeExp) +
```

```
  geom_point(alpha=0.5)
```

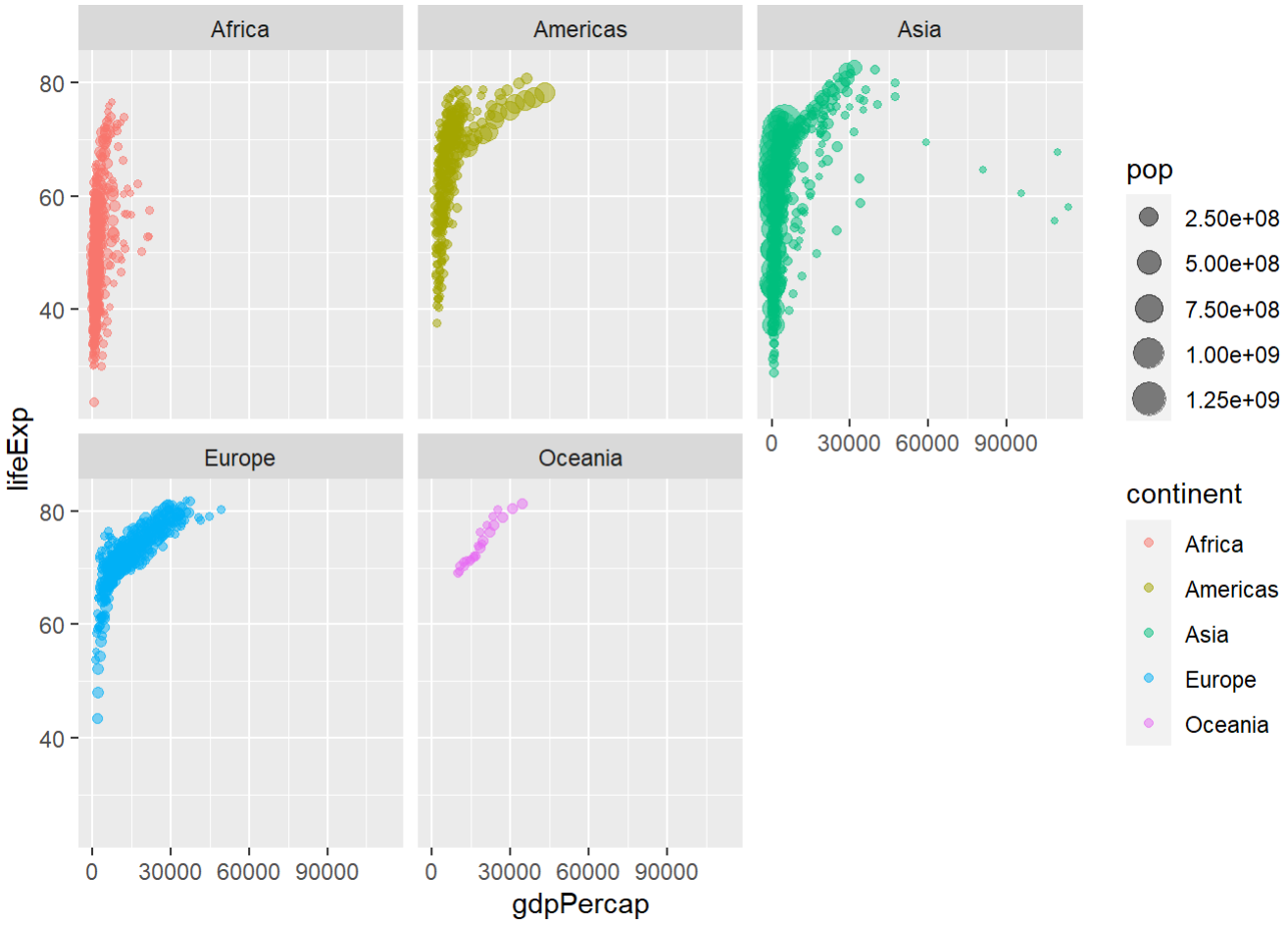



```
ggplot(gapminder) +
  aes(x=gdpPercap, y=lifeExp, color=continent, size=pop) +
  geom_point(alpha=0.5)
```



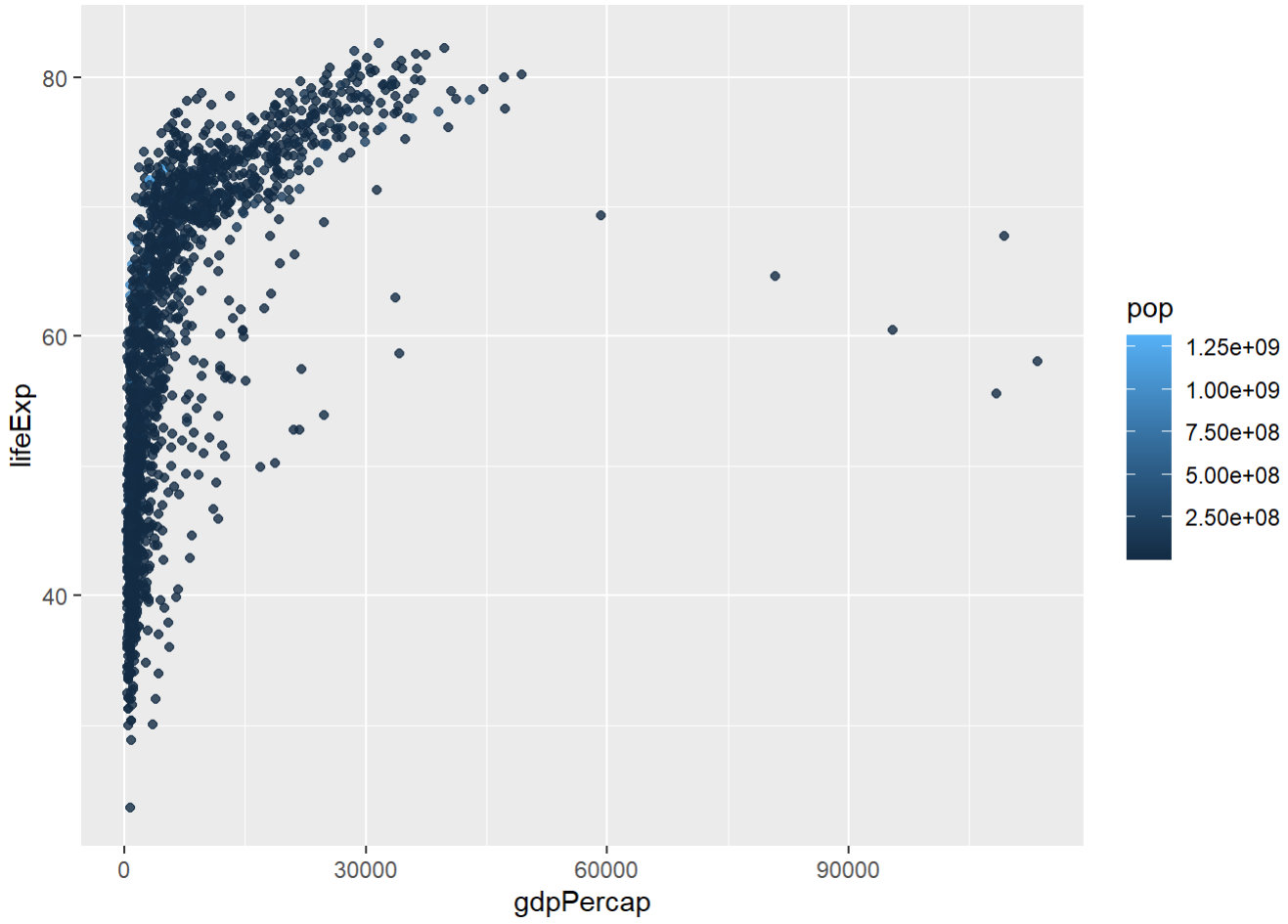

A very useful layer to add sometimes is for “faceting”

```
ggplot(gapminder) +
  aes(x=gdpPercap, y=lifeExp, color=continent, size=pop) +
  geom_point(alpha=0.5)+
  facet_wrap(~continent)
```




Let's see how the plot looks like if we color the points by the numeric variable population pop

```
ggplot(gapminder)+
  aes(x = gdpPercap, y = lifeExp, color = pop) +
  geom_point(alpha=0.8)
```


80
60
40
lifeExp
0
30000
60000
90000
gdpPercap
pop
1.25e+09
1.00e+09
7.50e+08
5.00e+08
2.50e+08