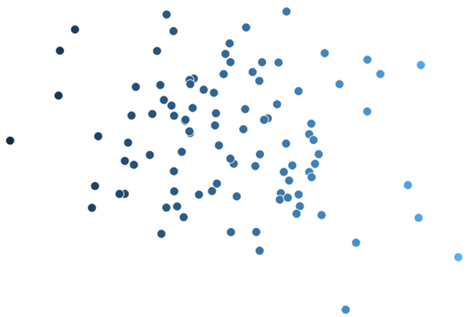
Grammar of Graphics

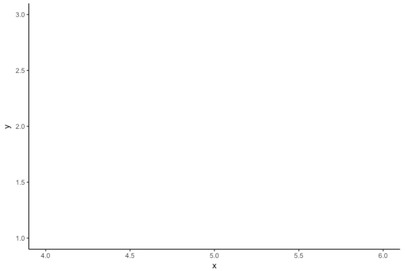
- Leland Wilkinson, *The Grammar of Graphics* (2nd edition, 2005)
- Why focus on grammar?
- More flexible, more room for growth
- ggplot2 is one implementation

Building Blocks

Layer(s)

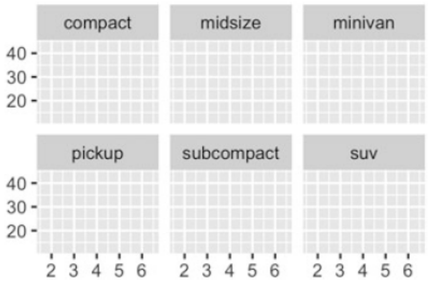


Scale(s)

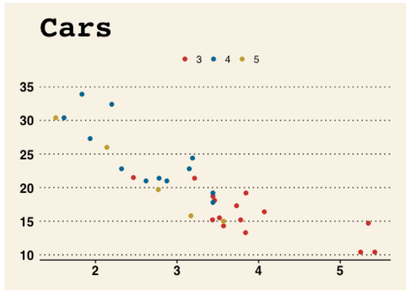


Coord

Facet

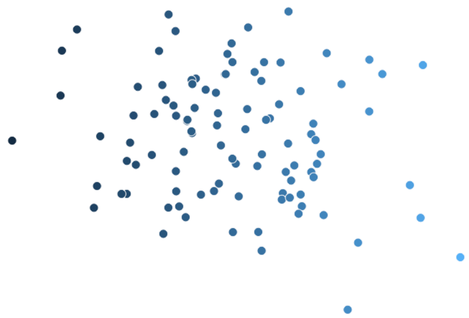


Theme

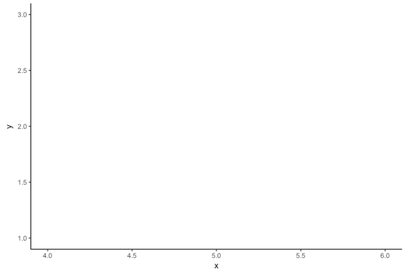


Building Blocks

Layer(s)

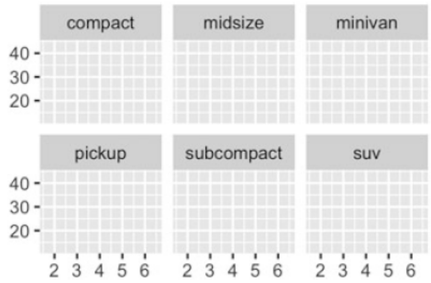


Scale(s)

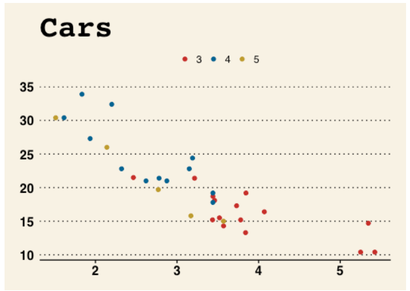


Coord

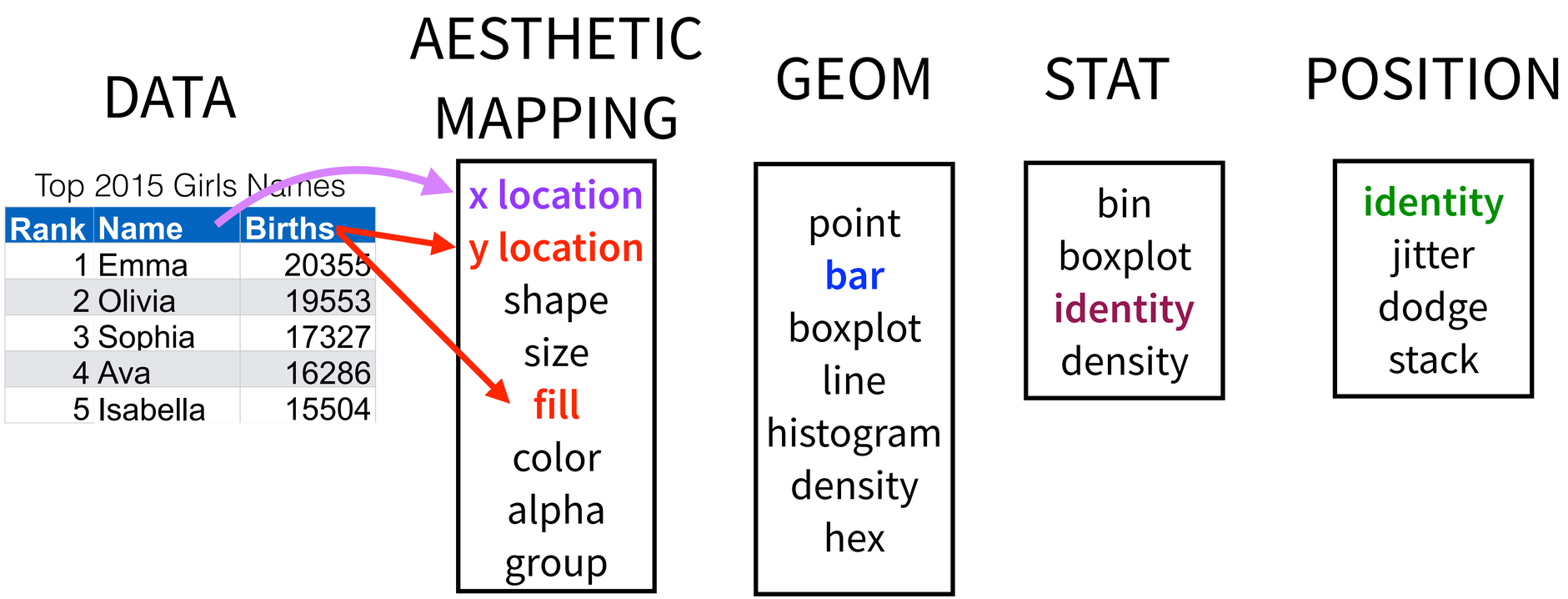
Facet



Theme



Layers


DATA
AESTHETIC MAPPING
GEOM
STAT
POSITION
Top 2015 Girls Names
Rank Name Births
1 Emma 20355
2 Olivia 19553
3 Sophia 17327
4 Ava 16286
5 Isabella 15504
x location
y location
shape
size
fill
color
alpha
group
point
bar
boxplot
line
histogram
density
hex
bin
boxplot
identity
density
identity
jitter
dodge
stack

Layer 1

```
df1 <- data.frame(x = rnorm(100), y = rnorm(100))
```

Data: df1

Mapping: x →x, y →y

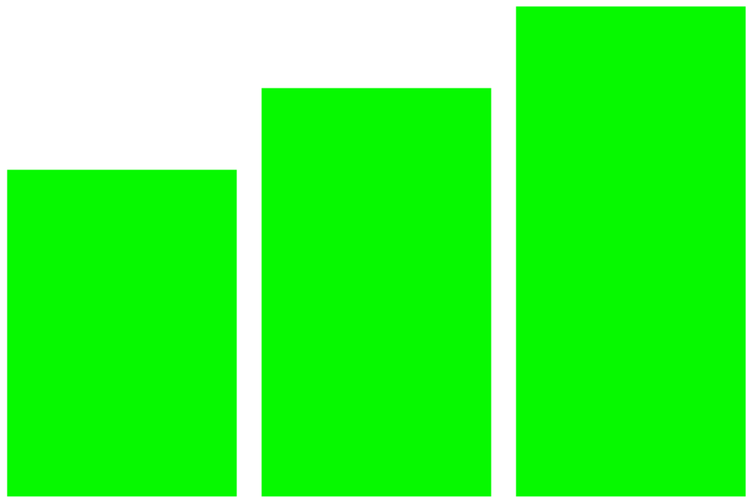
Geom: point

Stat: identity

Position: identity

Layer 2

```
df2 <- data.frame(num = 1:3, height = 4:6)
```



Data: df2

Mapping: num →x,
height →y

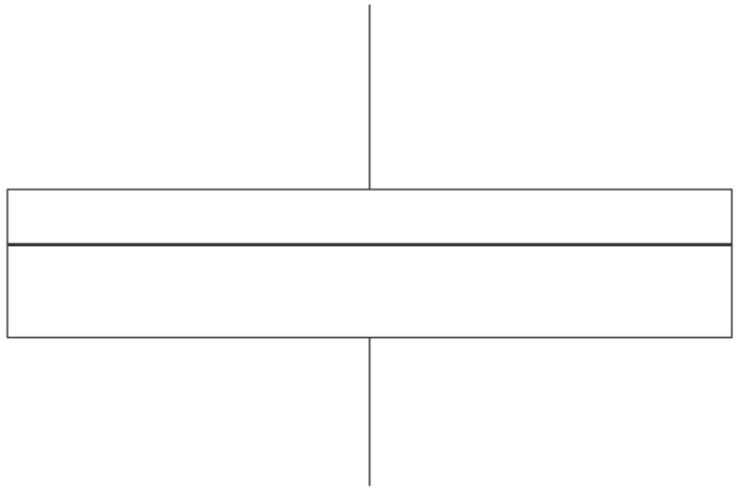
Geom: bar
setting: fill = green

Stat: identity

Position: identity

Layer 3

```
df3 <- data.frame(score = rnorm(25, mean = 15, sd = 3))
```



Data: df3

Mapping: 1 →x,
score →y

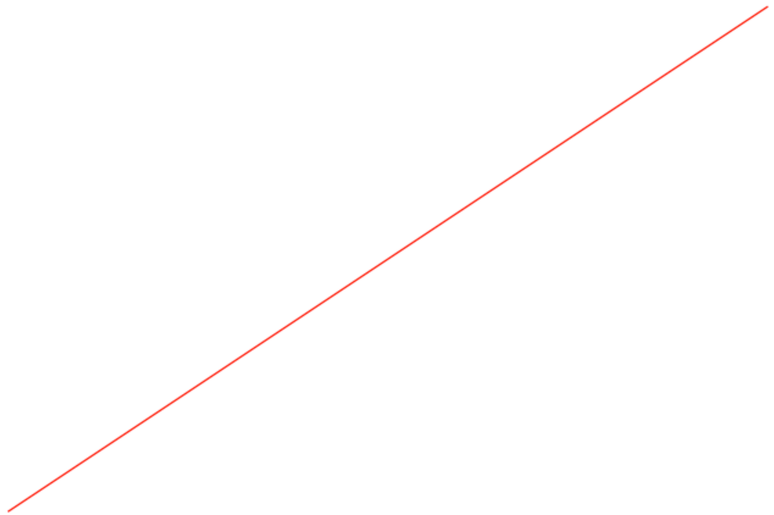
Geom: boxplot

Stat: boxplot

Position: dodge

Layer 4

```
df4 <- data.frame(time = 1:10, dist = 1:10)
```



Data: df4

Mapping: time→x

dist →y

Geom: line

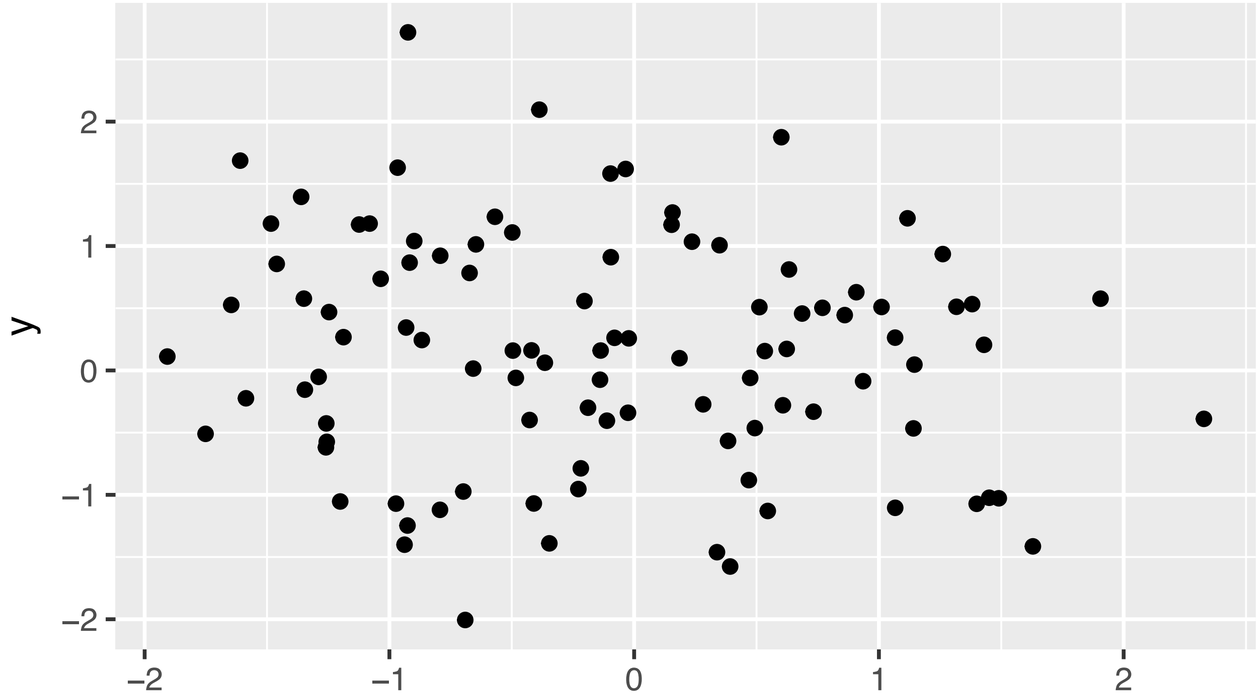
Stat: identity

Position: identity

Layer 1

(don't actually do this)

```
df1 <- data.frame(x = rnorm(100), y = rnorm(100))
ggplot() + layer(data = df1,
                 mapping = aes(x, y),
                 geom = "point",
                 stat = "identity",
                 position = "identity")
```



Data: df1

Mapping: x →x, y →y

Geom: point

Stat: identity

Position: identity

Layer 2

```
df2 <- data.frame(num = 1:3, height = 4:6)
ggplot() +
    layer(data = df2,
          mapping = aes(x = num, y = height),
          geom = "bar", params = list(fill = "green"),
          stat = "identity", position = "identity")
```

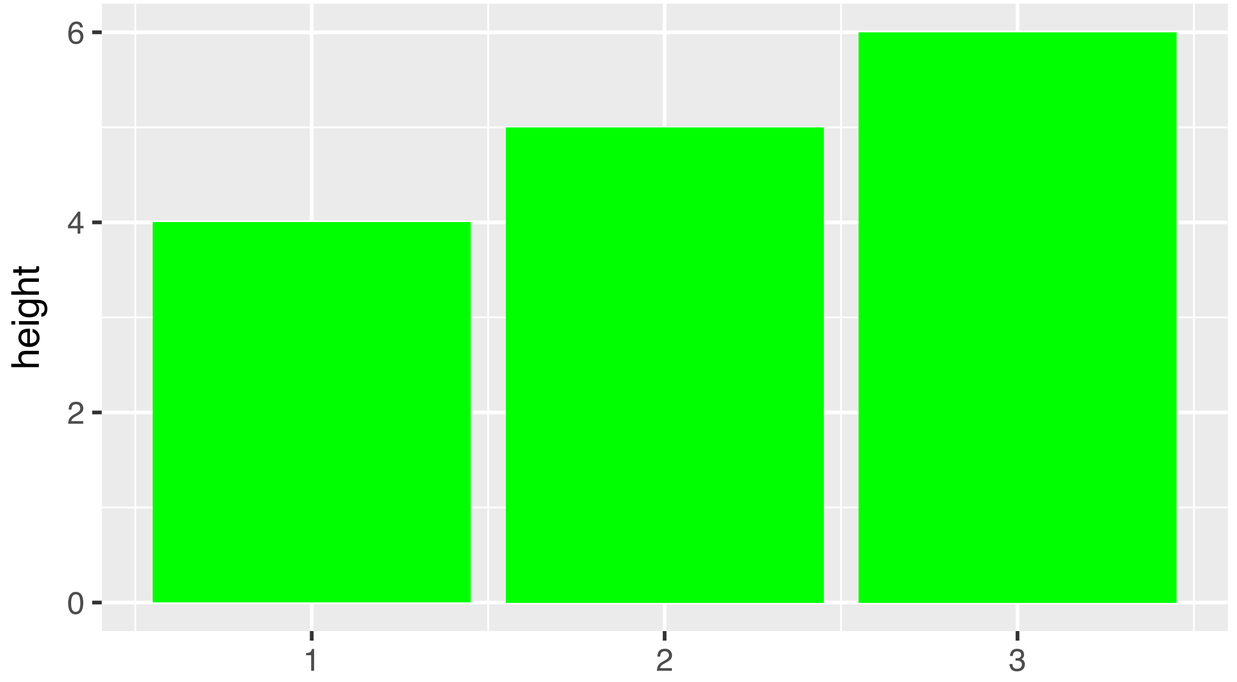
Data: df2

Mapping: num →x,
height →y

Geom: bar
setting: fill = green

Stat: identity

Position: identity



Layer 3

```
df3 <- data.frame(score = rnorm(25, mean = 15, sd = 3))
ggplot() + layer(data = df3,
                 mapping = aes(1, score),
                 geom = "boxplot",
                 stat = "boxplot",
                 position = "dodge")
```

Data: df3

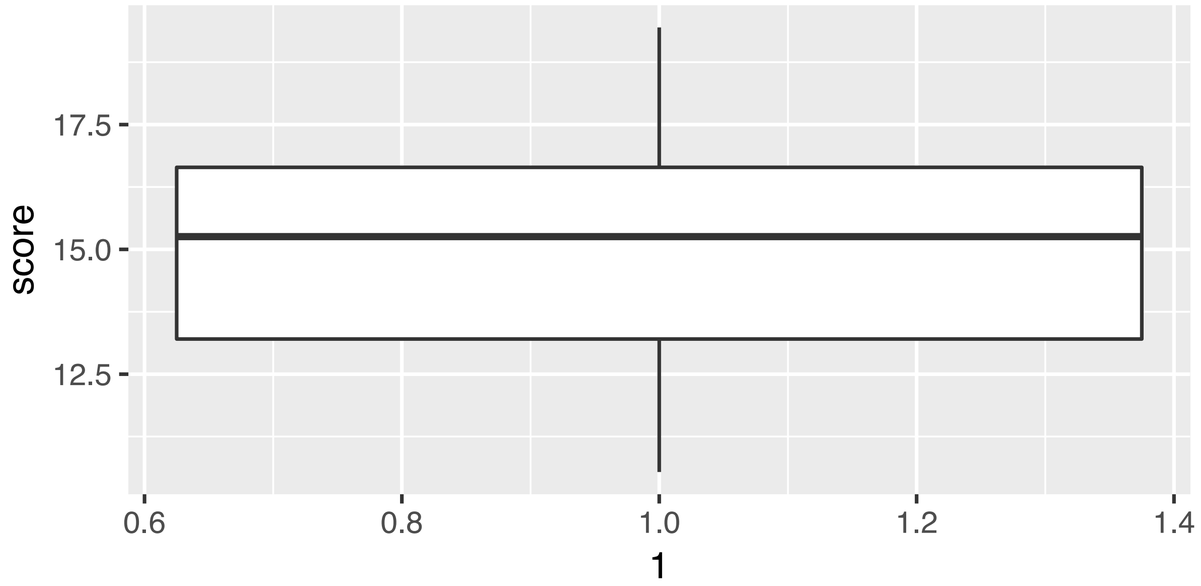
Mapping: 1 →x

score →y

Geom: boxplot

Stat: boxplot

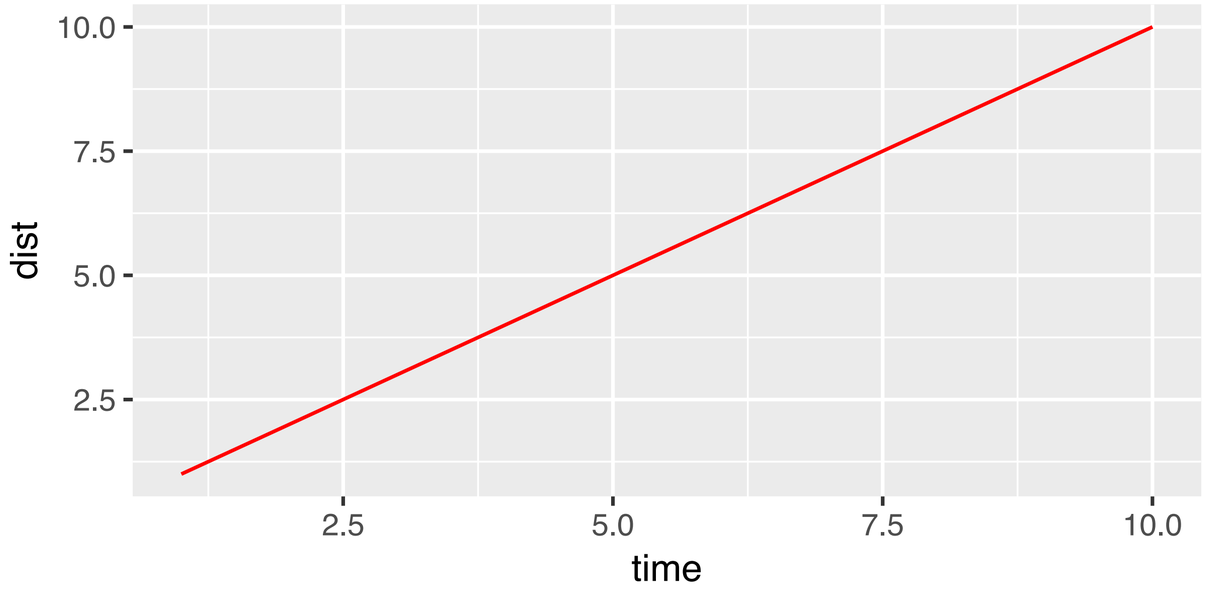
Position: dodge



Layer 4

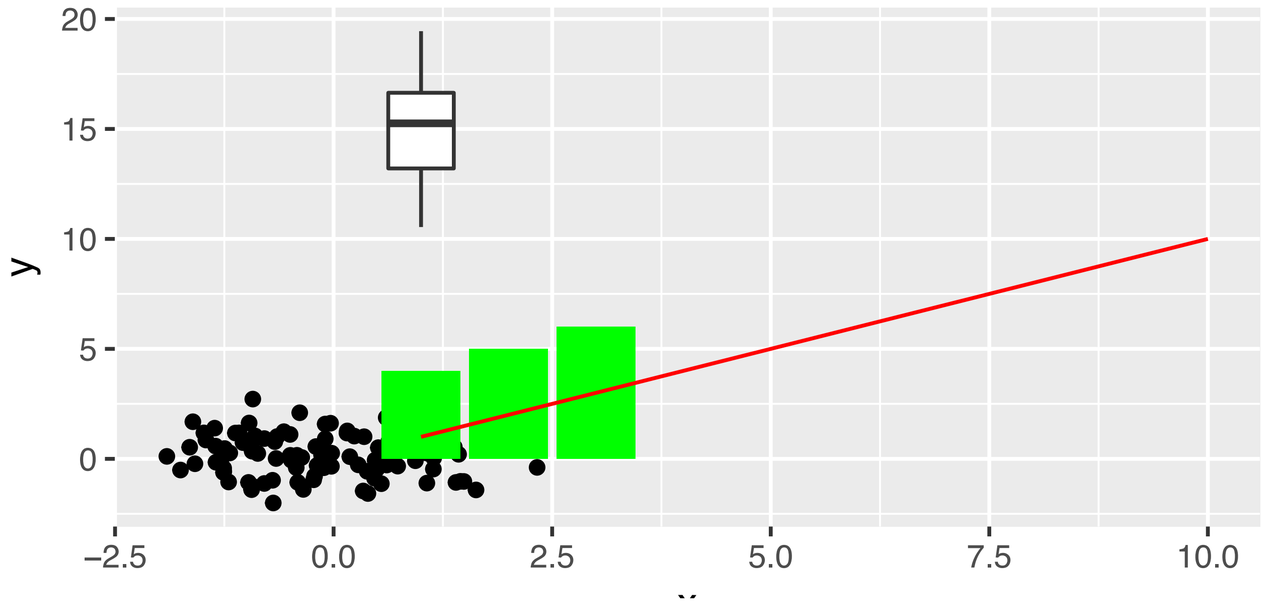
```
df4 <- data.frame(time = 1:10, dist = 1:10)
ggplot() + layer(data = df4,
                 mapping = aes(x = time, y = dist),
                 geom = "line",
                 params = list(color = "red"),
                 stat = "identity", position = "identity")
```

Data: df4
Mapping: time→x
dist →y
Geom: line
Stat: identity
Position: identity



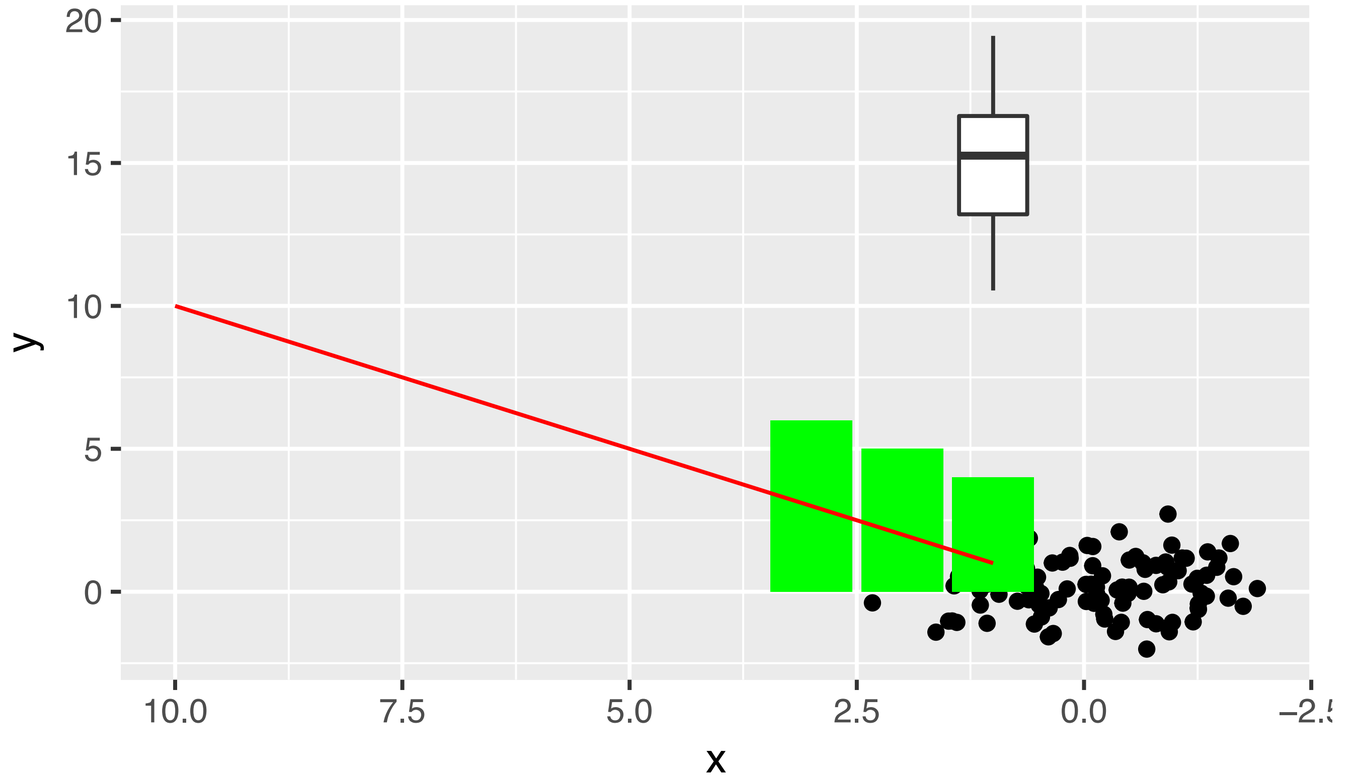
All layers

```
library (ggplot2)
g <- ggplot() + geom_point(data = df1, aes(x,y)) +
    geom_col(data = df2, aes(num, height),
             fill = "green") +
    geom_boxplot(data = df3, aes(1, score)) +
    geom_line(data = df4, aes(time, dist),
              color = "red")
g
```

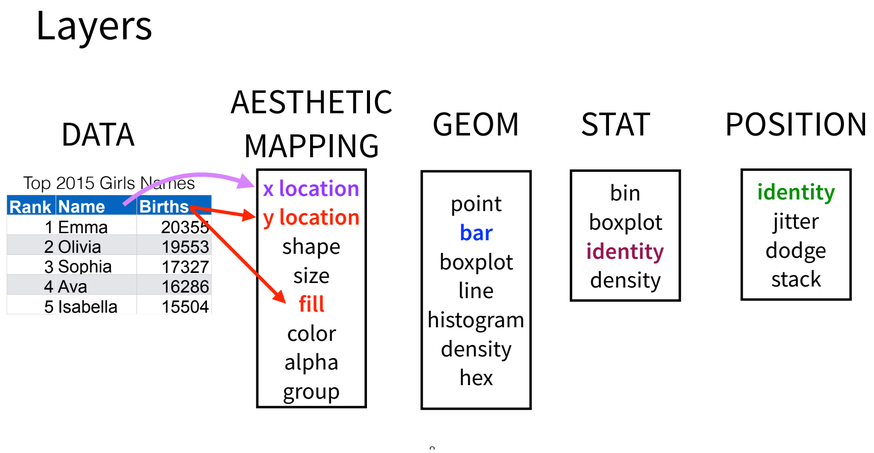


Scale

```
g + scale_x_reverse()
```



One scale per mapping

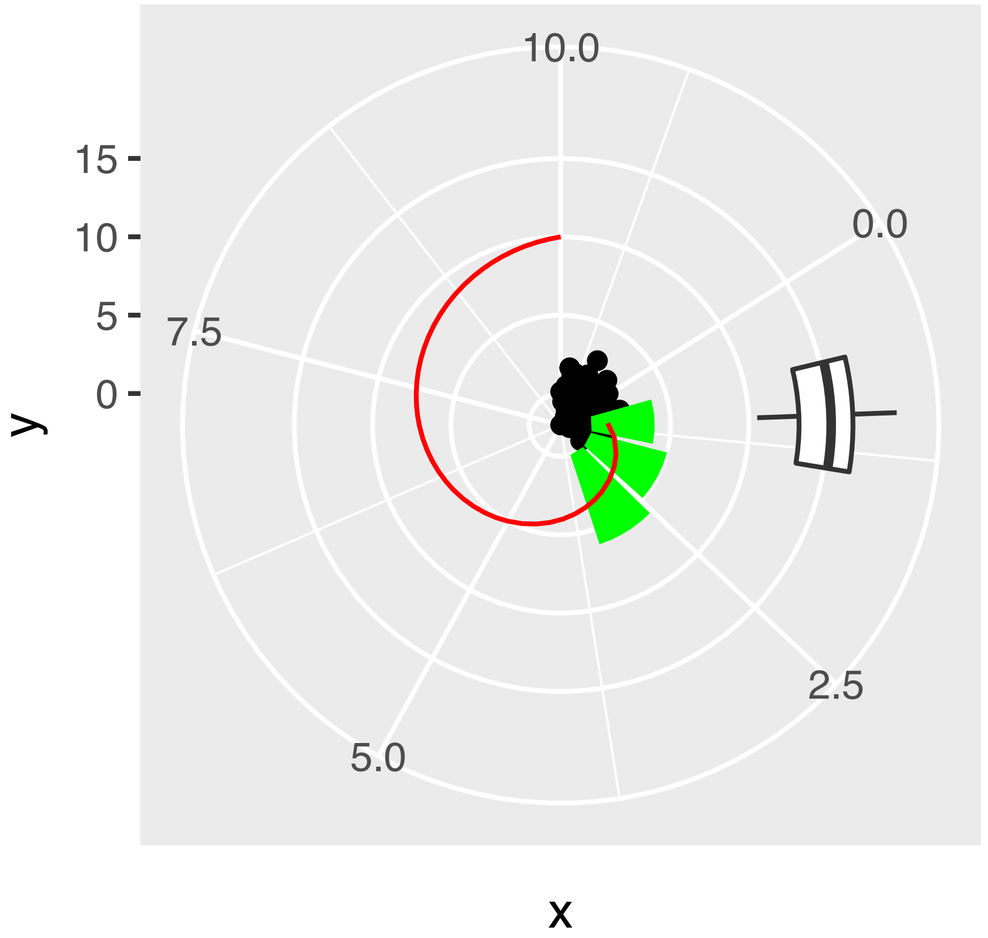


MAPPING	SCALE
x →	scale_x_date()
y →	scale_y_continuous()
color →	scale_color_manual()
fill →	scale_fill_viridis_c()

Coord

(only 1!)

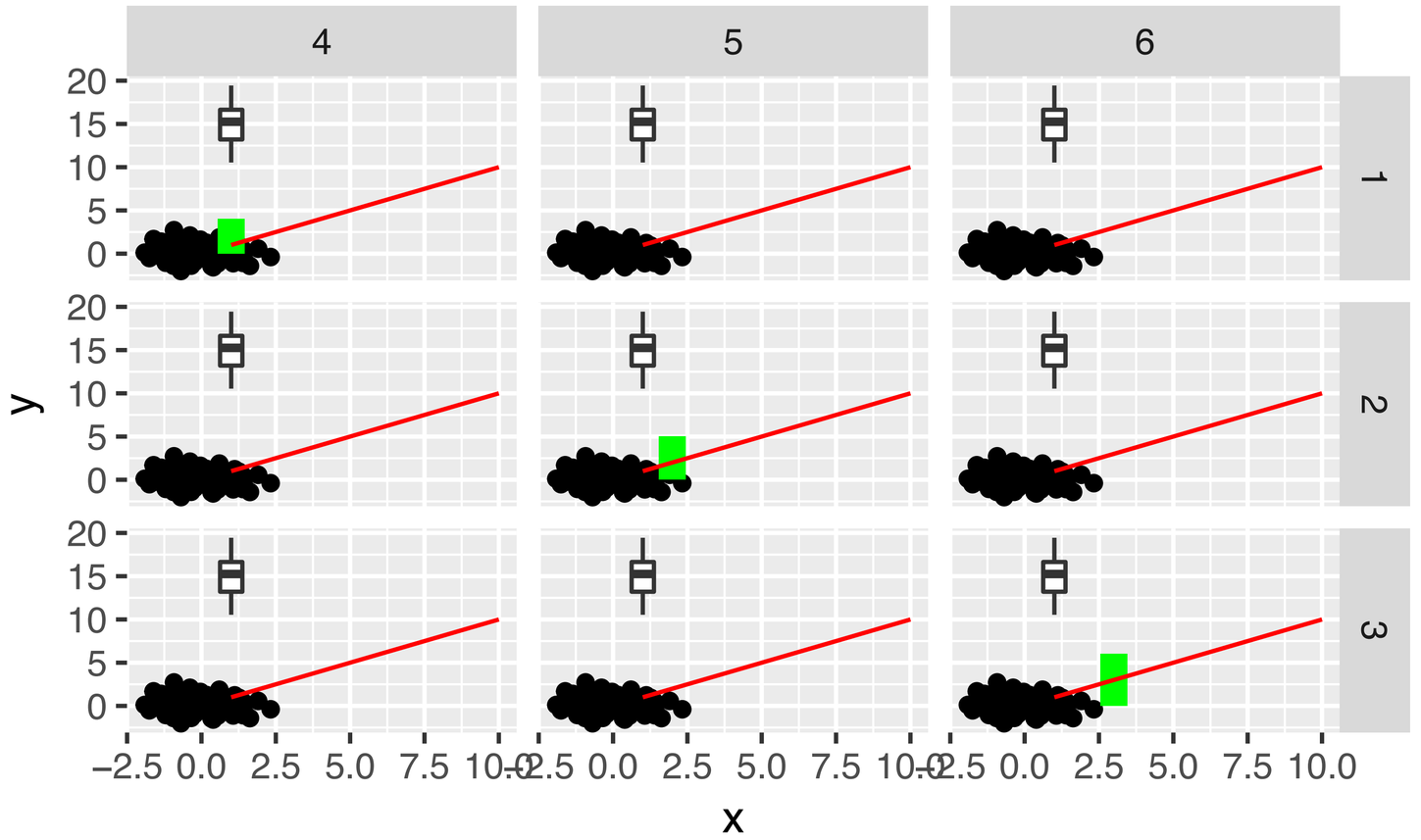
```
g + coord_polar()
```



Facet

(only 1!)

```
g + facet_grid(num~height)
```



Theme

(only 1!)

```
library(ggthemes)
g + theme_wsj()
```