# GR5702
# Exploratory Data Analysis and Visualization

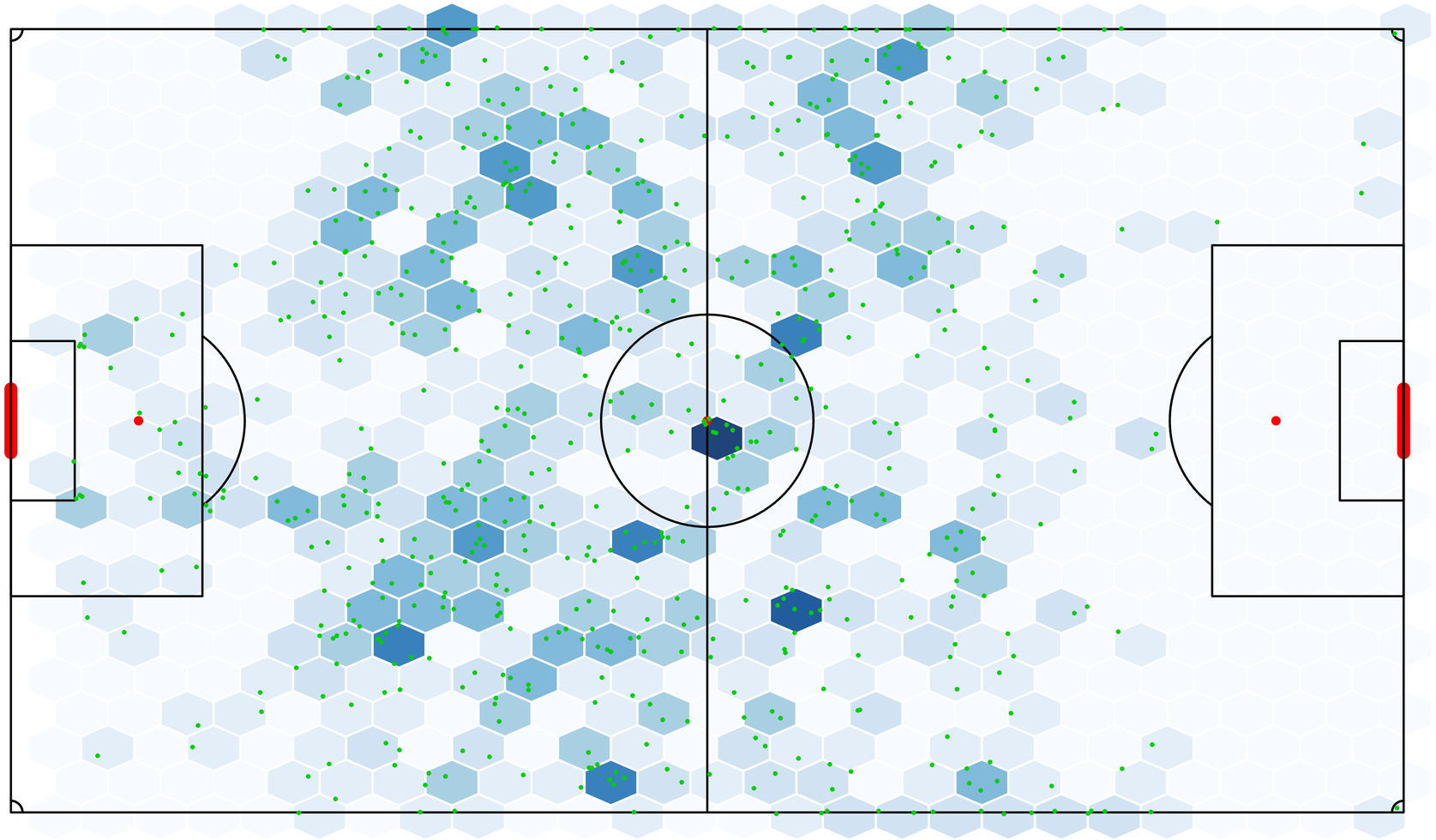
Prof. Joyce Robbins

# Today's Agenda (1/24/ 17)

- Announcements
  - Final Project
  - Homework http://flowingdata.com
  - DataCamp http://datacamp.com
- Grammar of Graphics / ggplot2

# Base Graphics

**passes (F): yes**

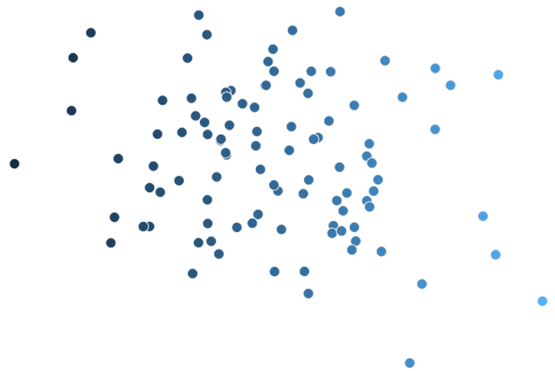


# Grammar of Graphics

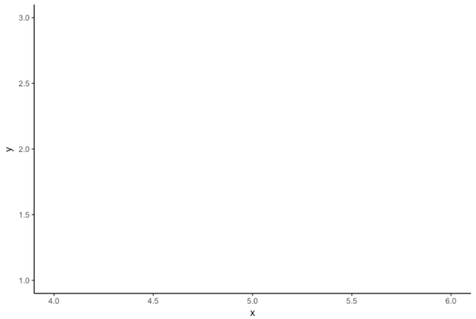
- Leland Wilkinson, *The Grammar of Graphics*, 1999 (2nd edition, 2005)
- Why focus on grammar?
- More flexible, more room for growth

# Building Blocks

## Layer



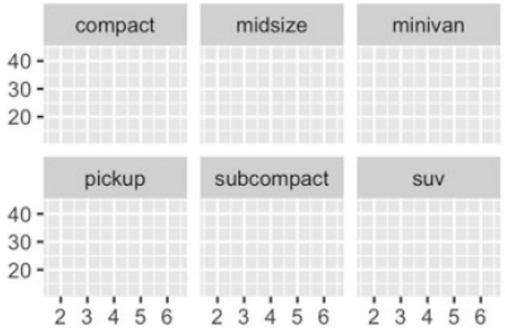
## Scale



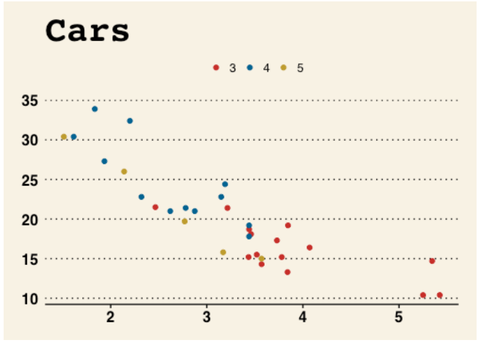

## Coord



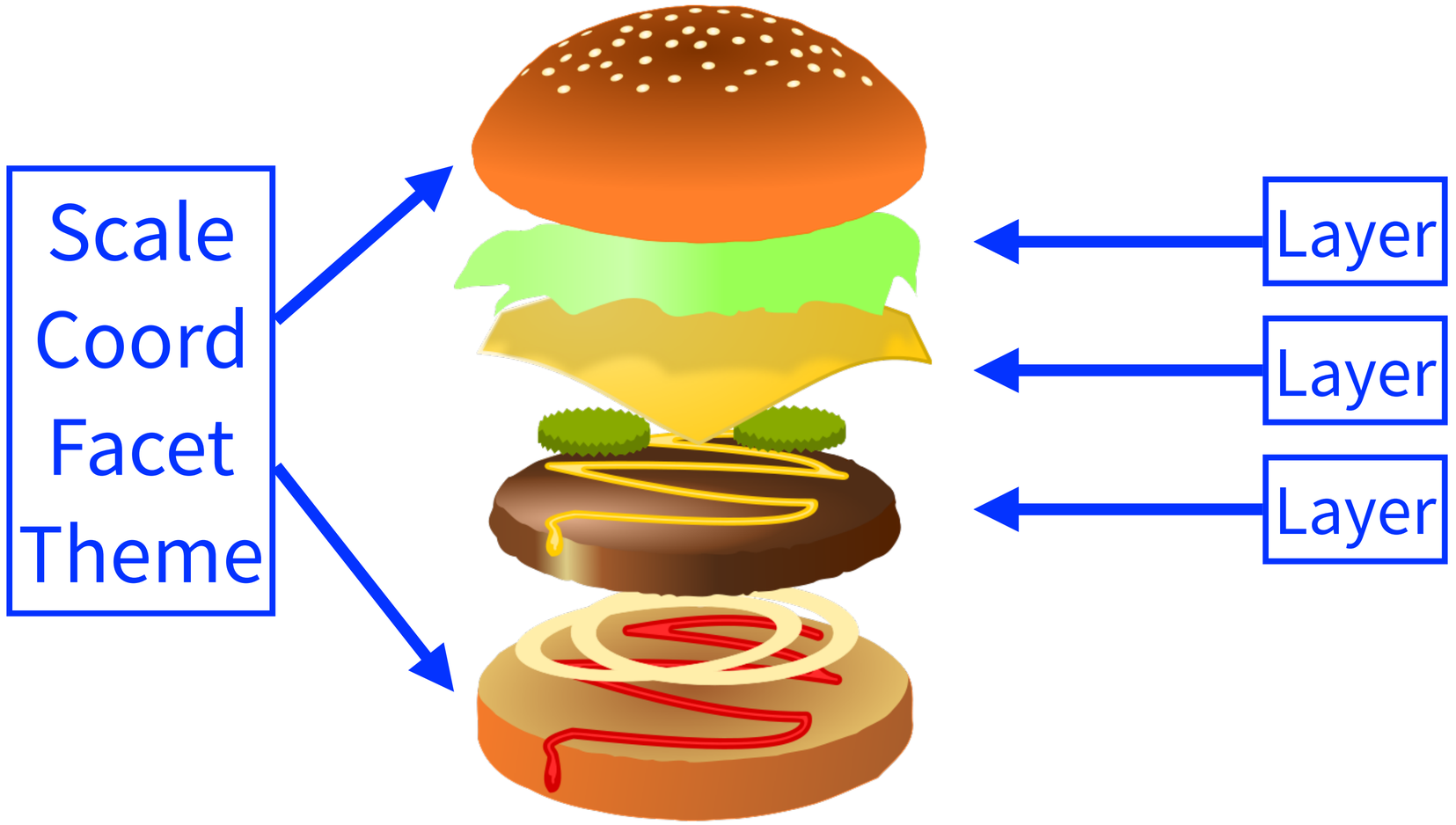
## Facet




## Theme



# Layered Approach



# No!

# Layers

## DATA

Top 2015 Girls Names

| Rank | Name | Births |
|---|---|---|
| 1 | Emma | 20355 |
| 2 | Olivia | 19553 |
| 3 | Sophia | 17327 |
| 4 | Ava | 16286 |
| 5 | Isabella | 15504 |

## AESTHETIC MAPPING

**x location**
**y location**
shape
size
**fill**
color
alpha
group

## GEOM

point
**bar**
boxplot
line
histogram
density
hex

## STAT

bin
boxplot
**identity**
density

## POSITION

**identity**
jitter
dodge
stack

# Layer 1

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

Data: df1

Mapping: x →x, y →y

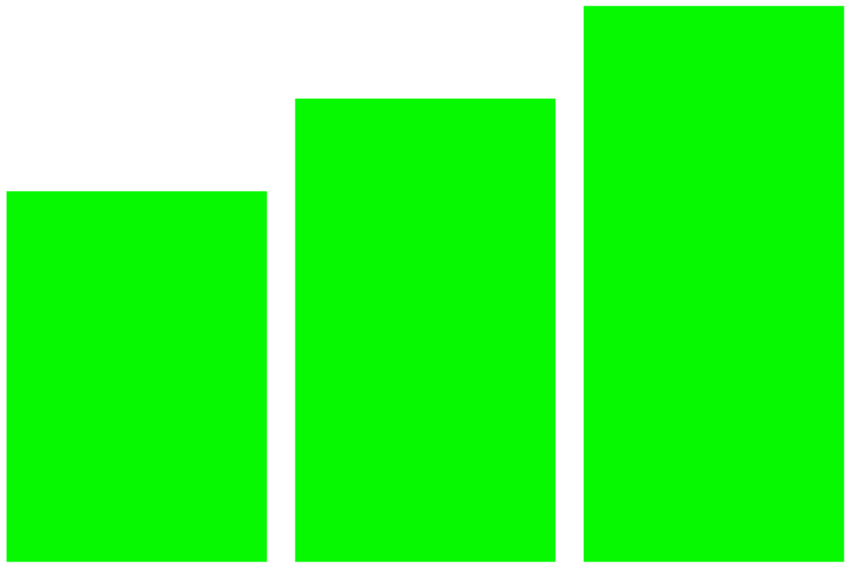
Geom: point

Stat: identity

Position: identity

# Layer 2

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



Data: df2

Mapping: num →x,
height →y

Geom: bar

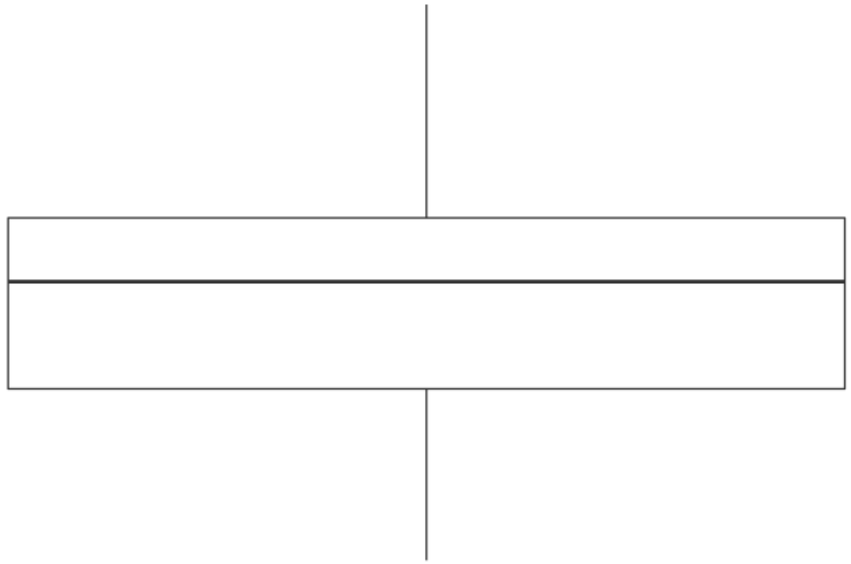
setting: fill = green

Stat: identity

Position: identity

# Layer 3

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



Data: df3

Mapping: 1 →x,
score →y

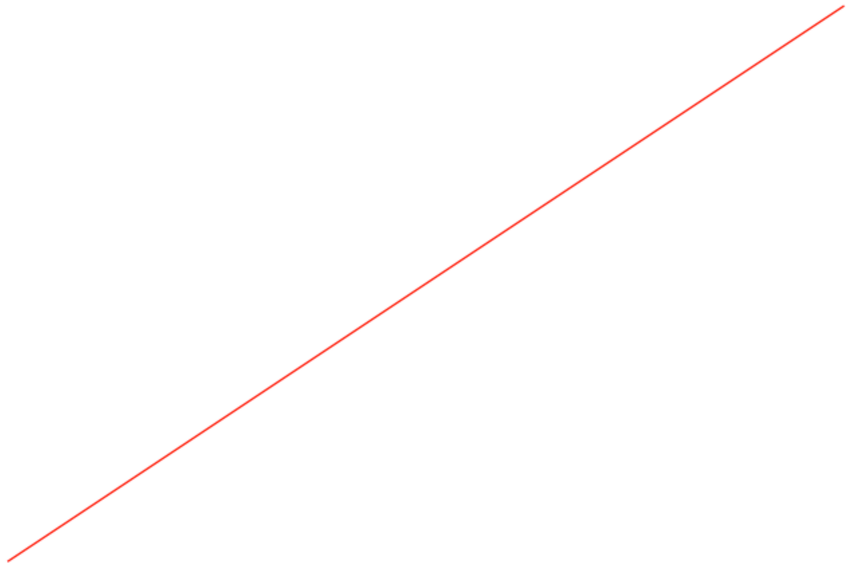
Geom: boxplot

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

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

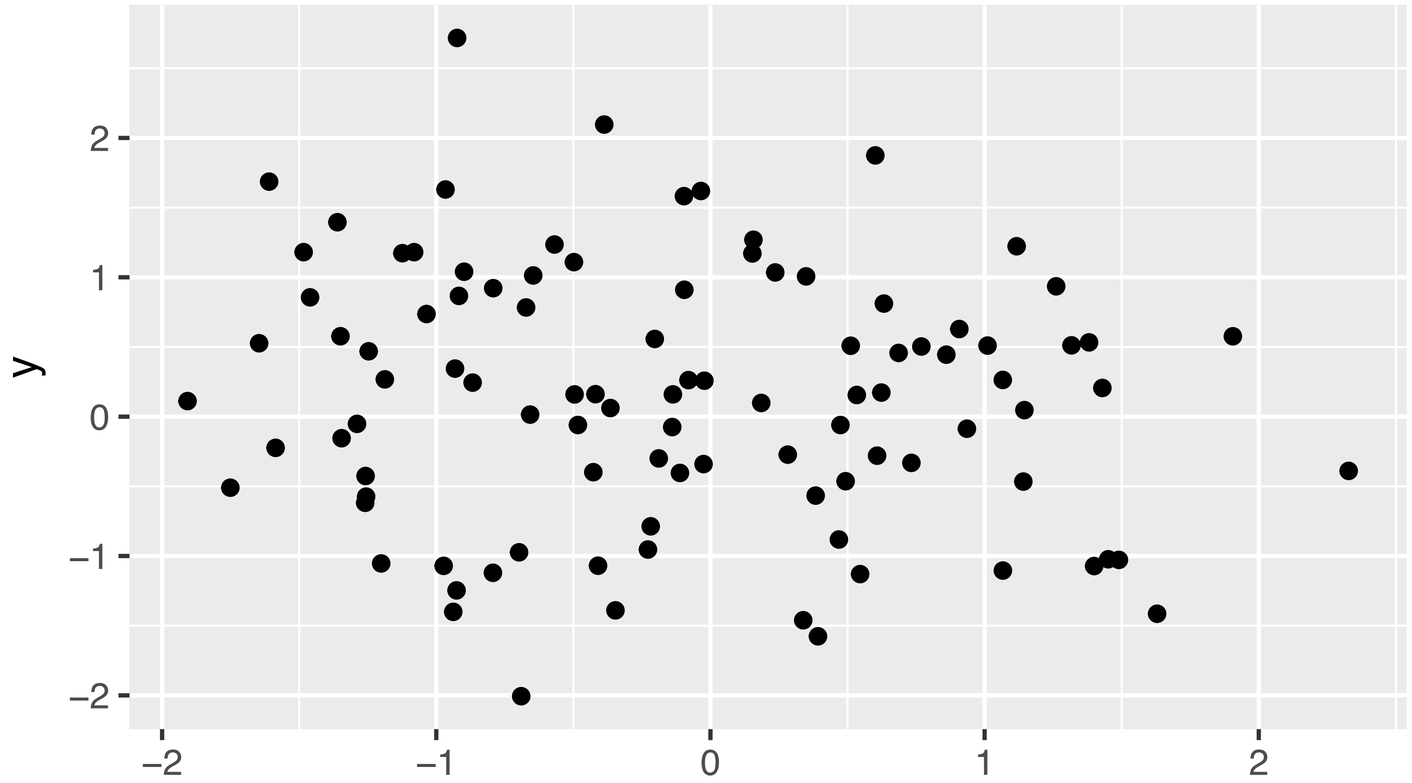
Data: df1

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")
```

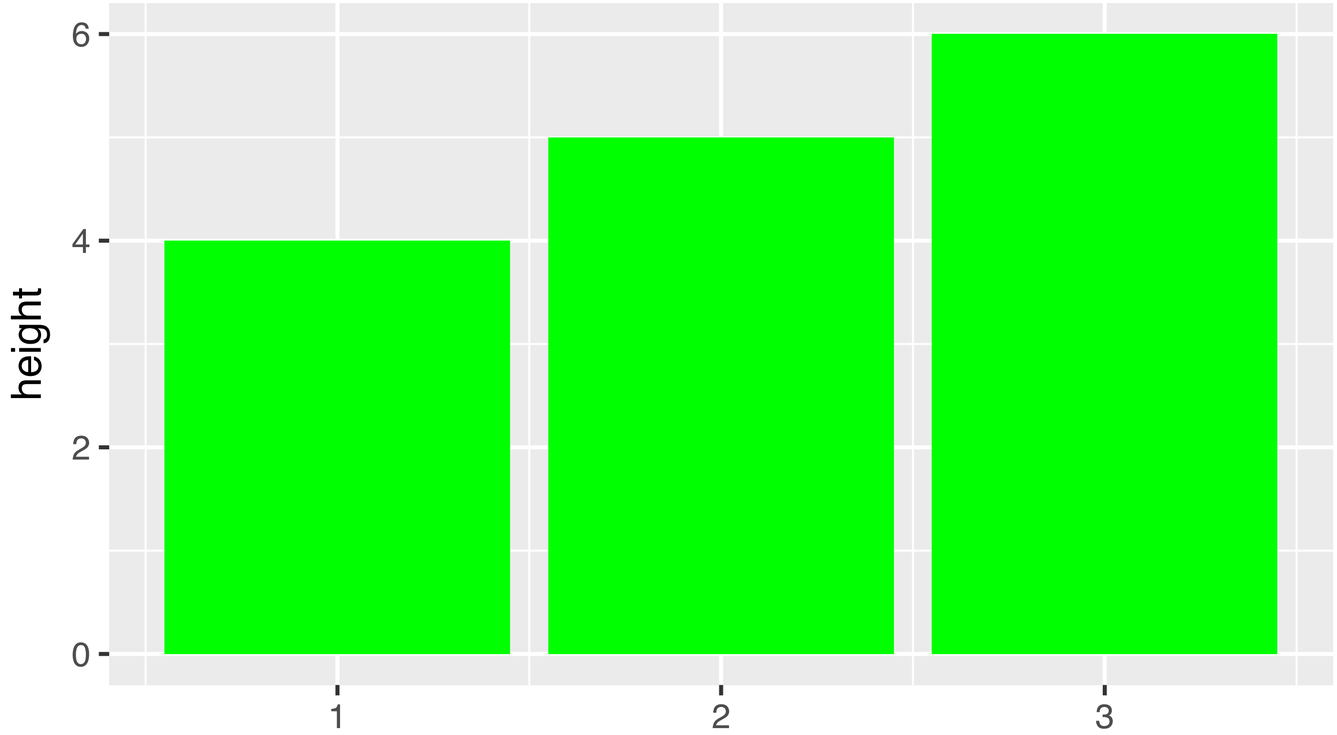
Data: df2

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

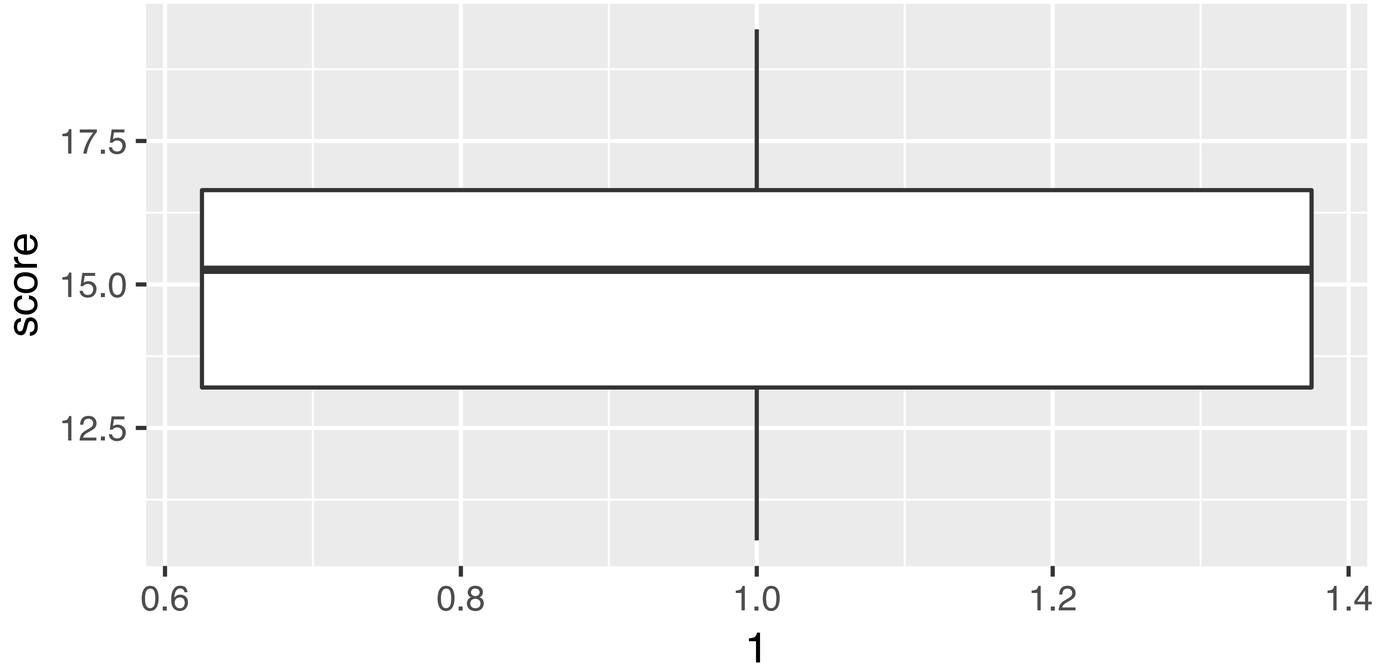
Mapping: 1 →x

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

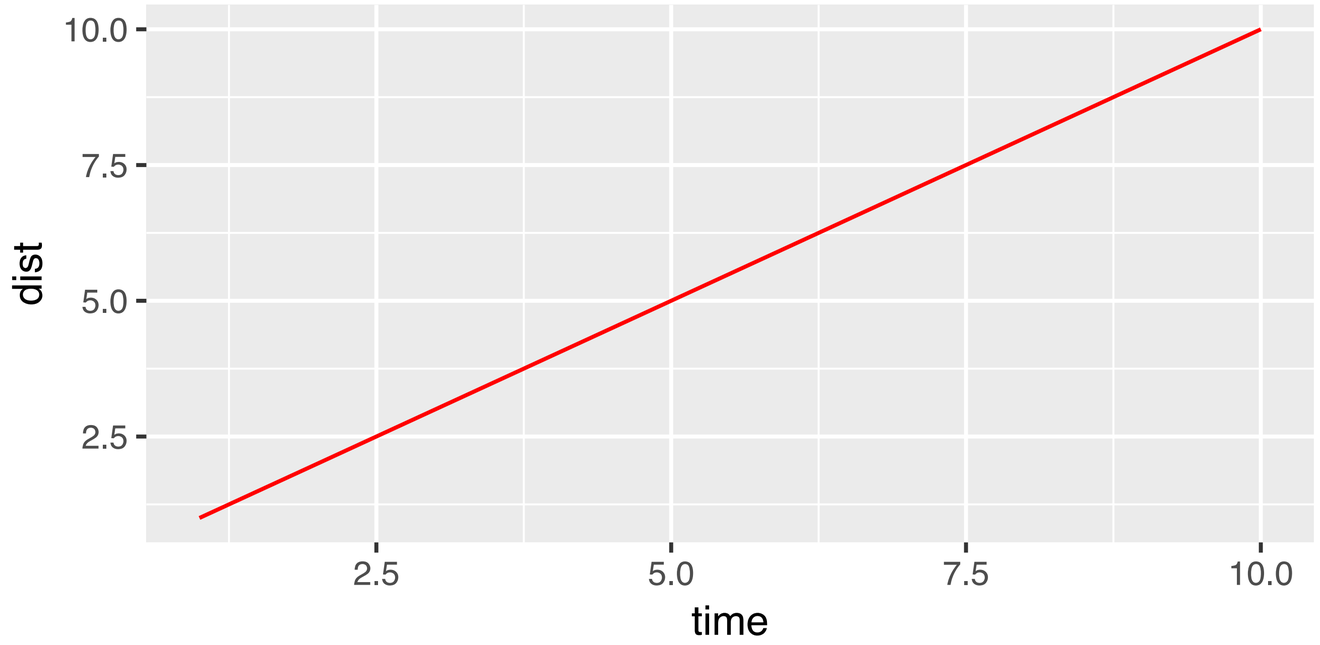
Mapping: time→x

dist →y

Geom: line

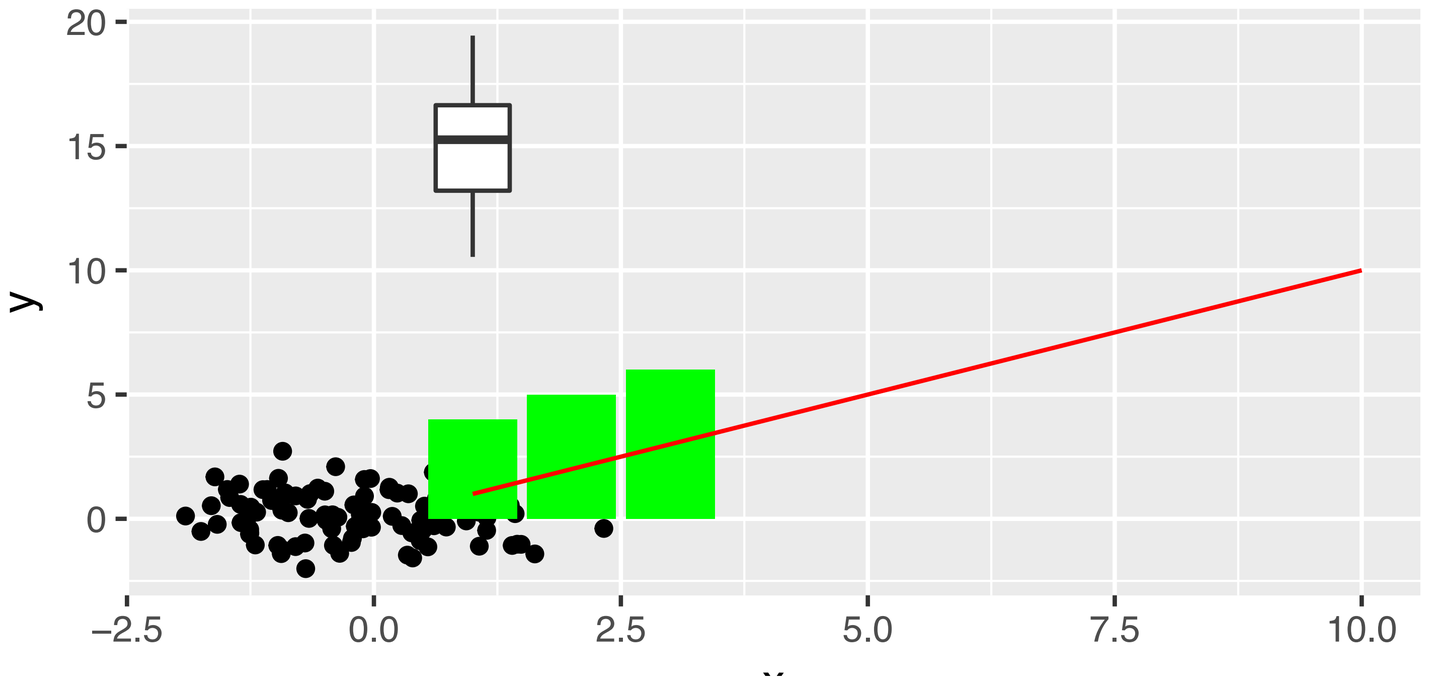
Stat: identity

Position: identity



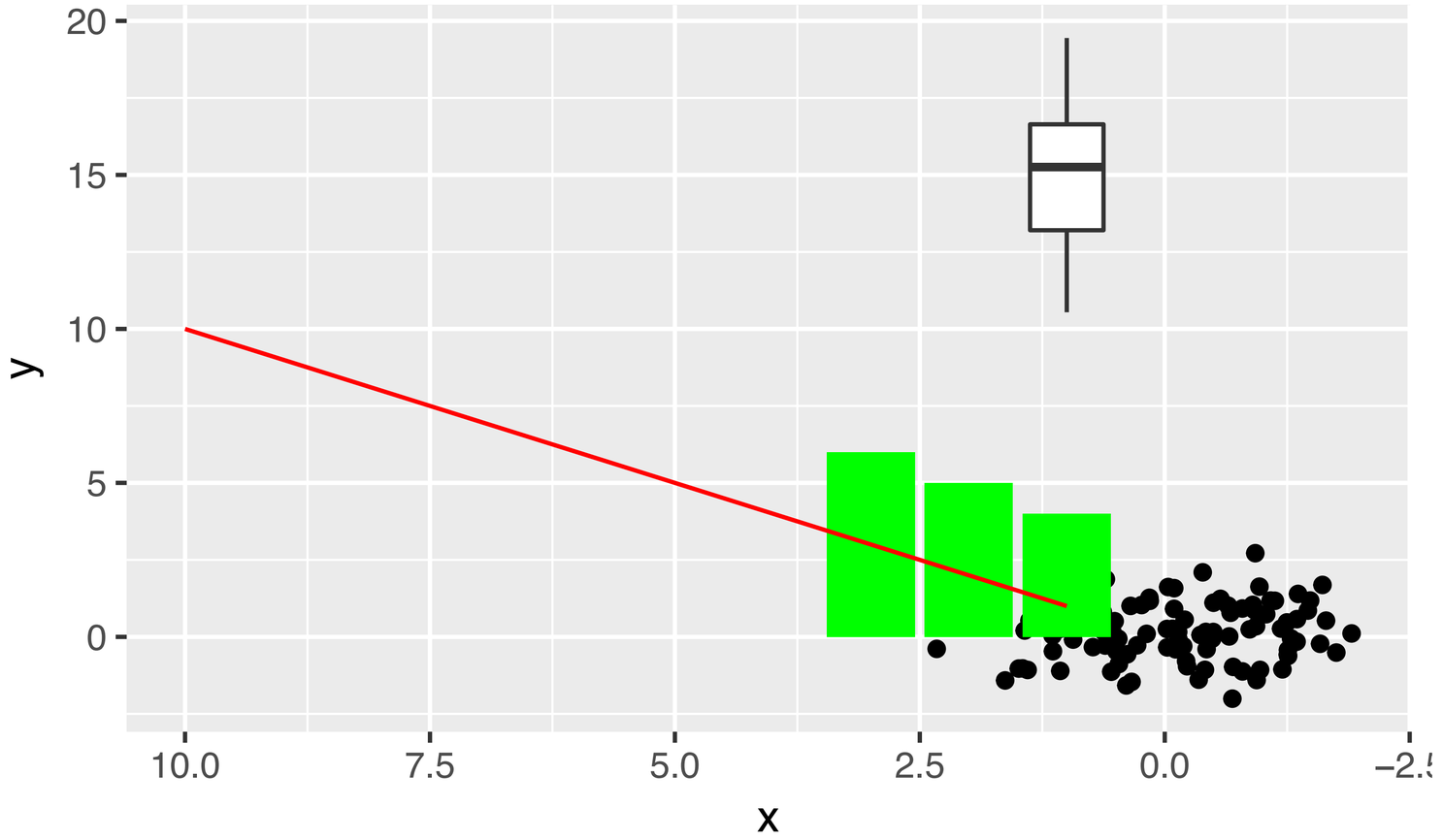
# All layers

```
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()
```



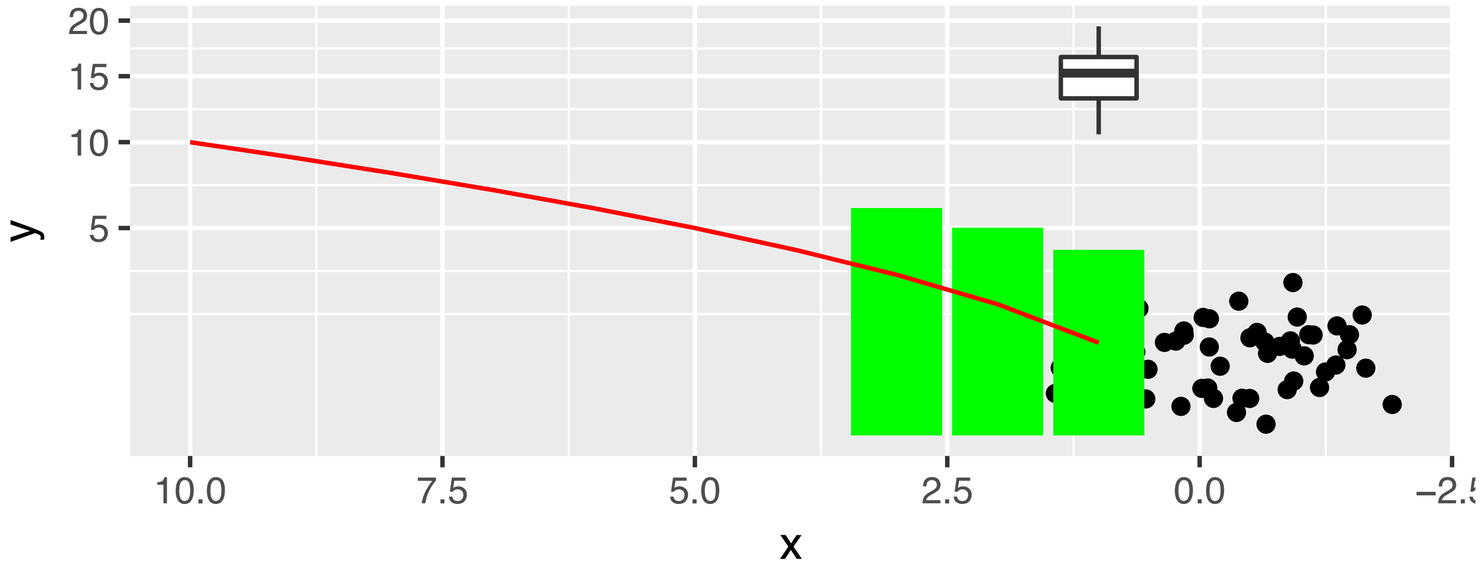
# Scale

```
g + scale_x_reverse() + scale_y_sqrt()
```

```
## Warning in self$trans$transform(x): NaNs produced
```

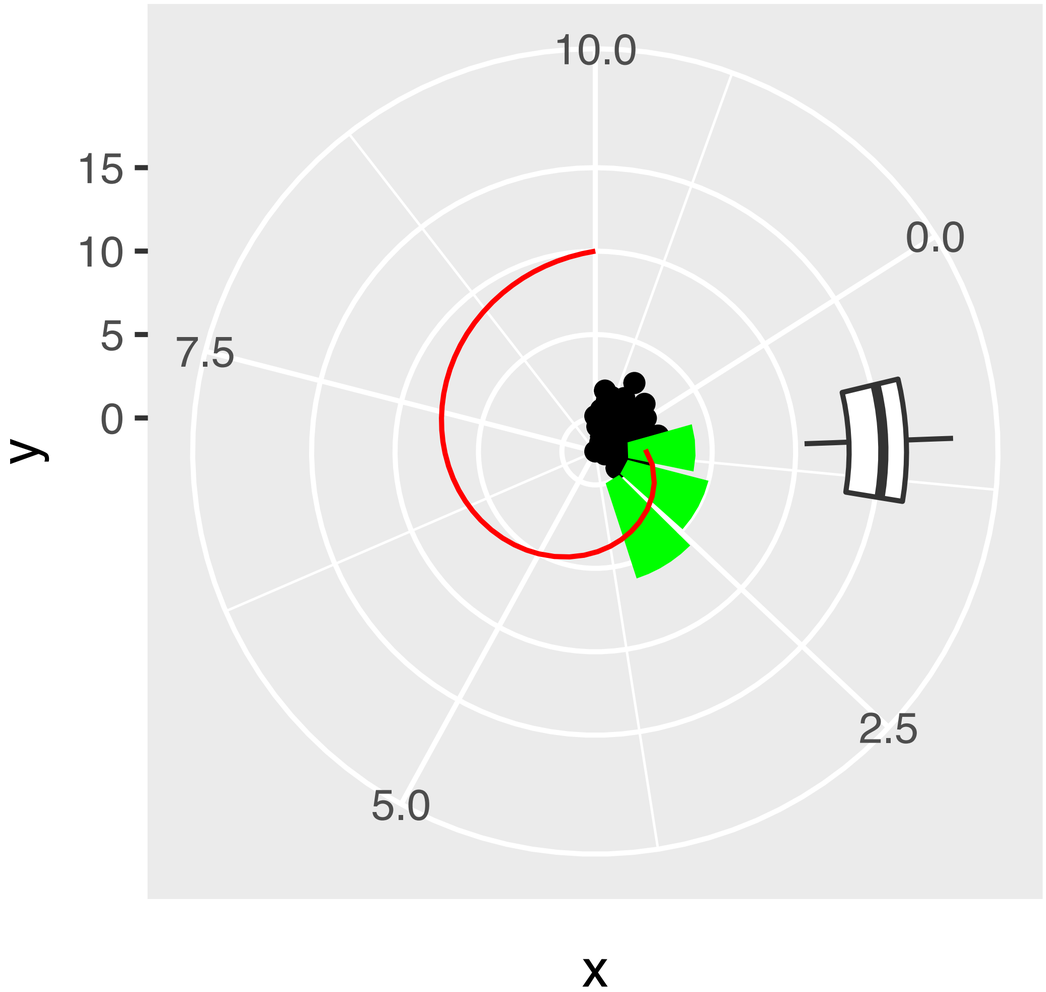
```
## Warning: Transformation introduced infinite values in c
```

```
## Warning: Removed 42 rows containing missing values (geo
```



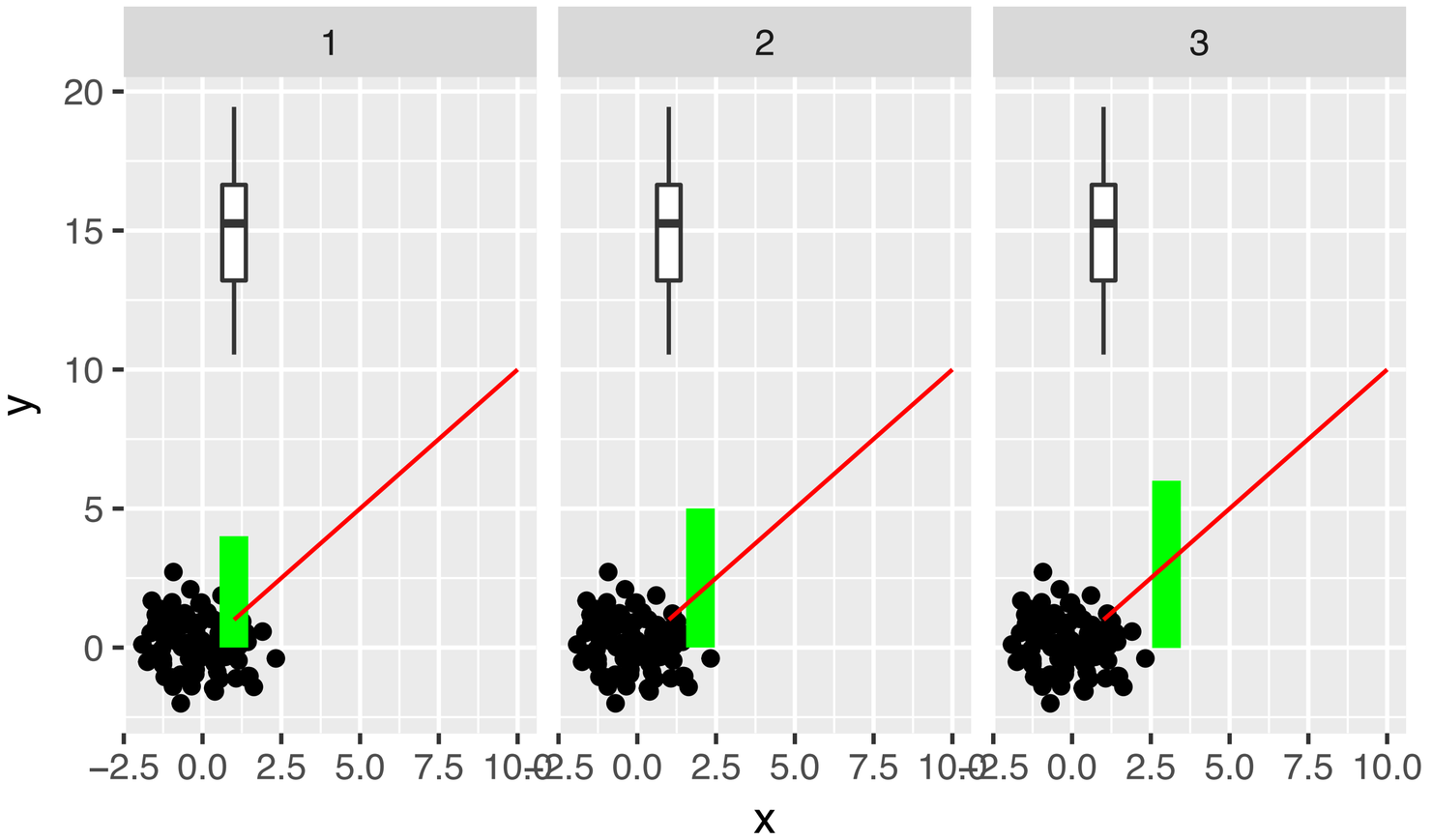
# Coord

```
g + coord_polar()
```



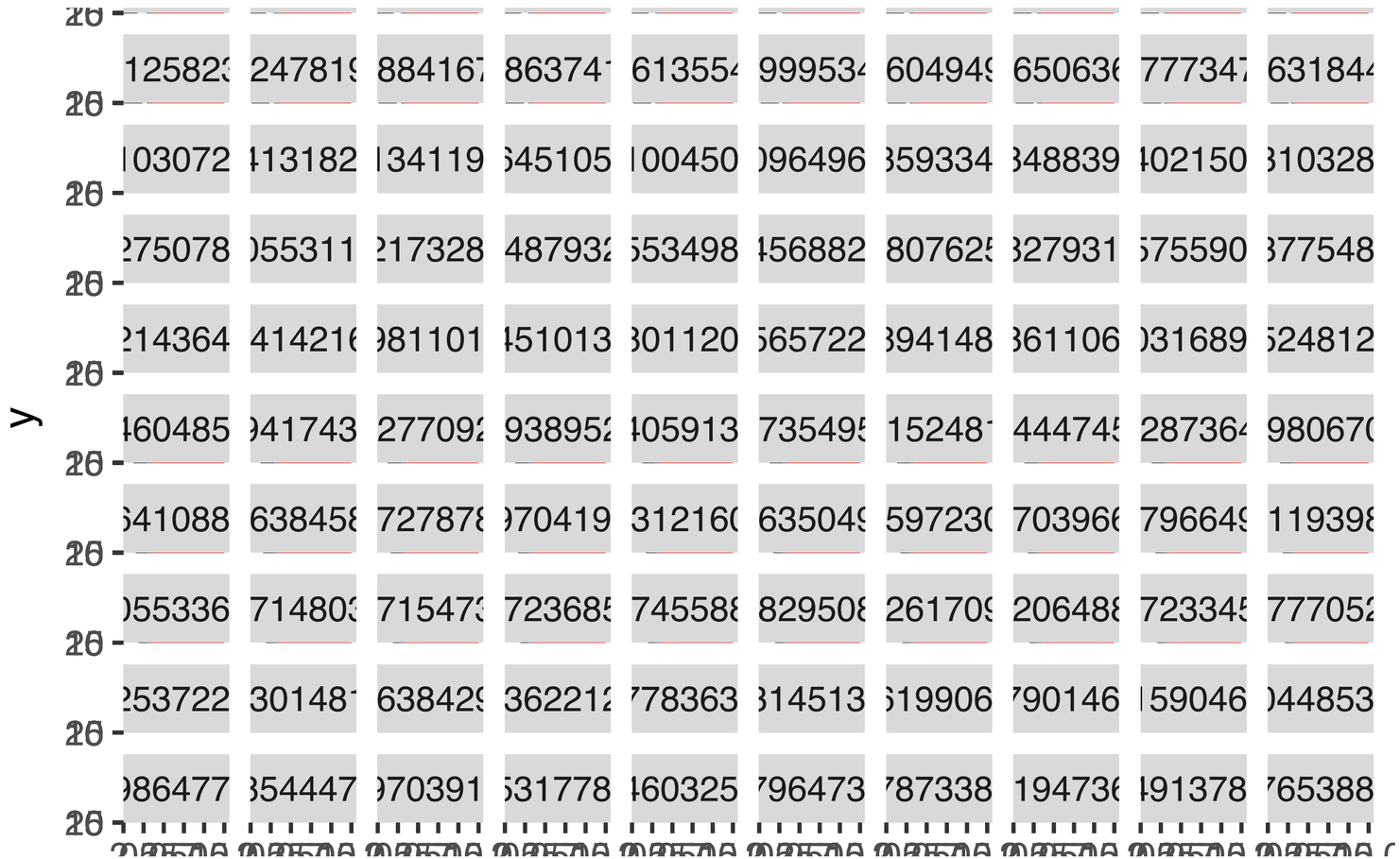
# Facet

```
g + facet_wrap(~num)
```



# Facet

```
g + facet_wrap(~x)
```



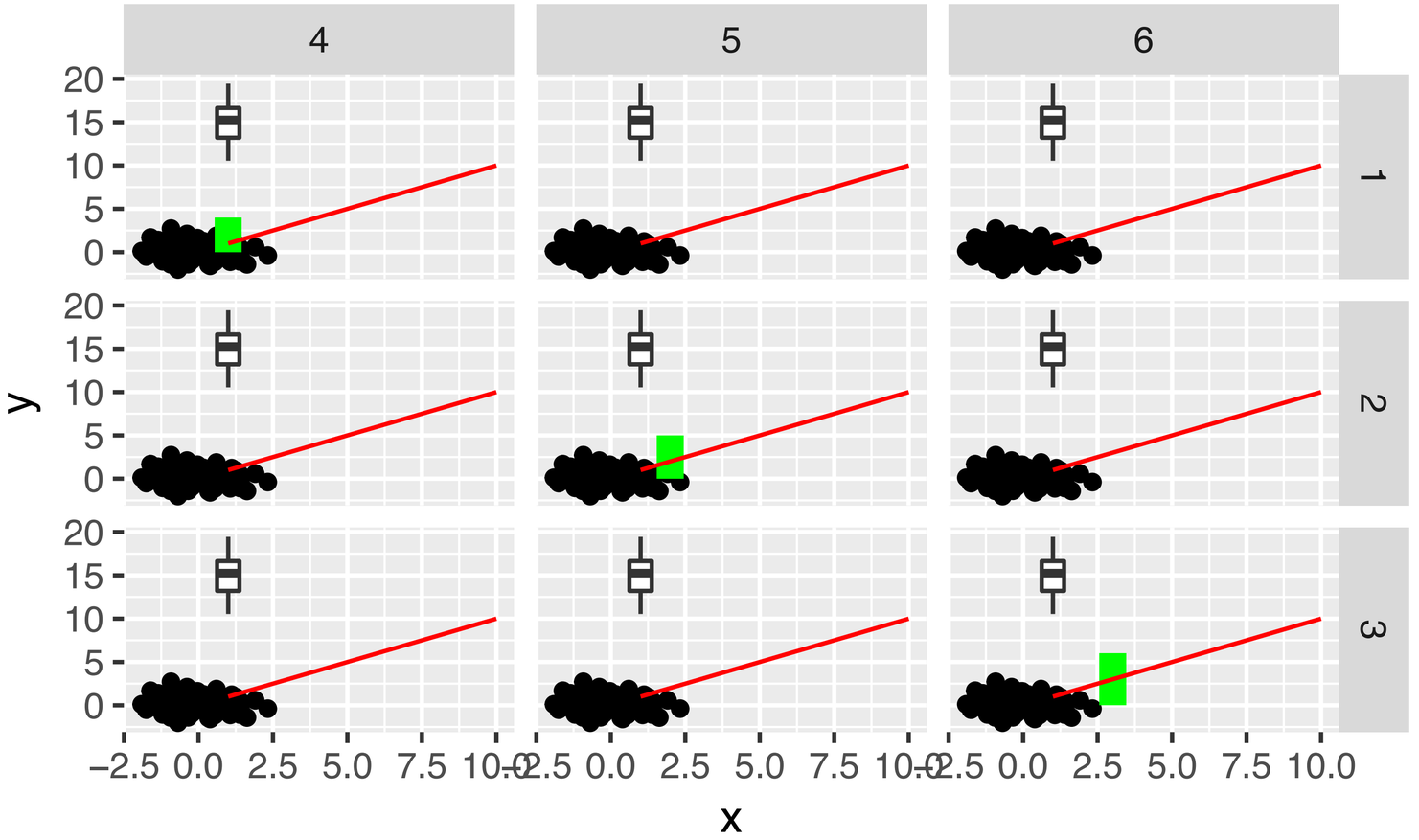
# Facet

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

```
## geom_path: Each group consists of only one observation.
## adjust the group aesthetic?
## geom_path: Each group consists of only one observation.
## adjust the group aesthetic?
## geom_path: Each group consists of only one observation.
## adjust the group aesthetic?
## geom_path: Each group consists of only one observation.
## adjust the group aesthetic?
## geom_path: Each group consists of only one observation.
## adjust the group aesthetic?
## geom_path: Each group consists of only one observation.
## adjust the group aesthetic?
## geom_path: Each group consists of only one observation.
## adjust the group aesthetic?
## geom_path: Each group consists of only one observation.
## adjust the group aesthetic?
## geom_path: Each group consists of only one observation.
## adjust the group aesthetic?
## geom_path: Each group consists of only one observation.
```

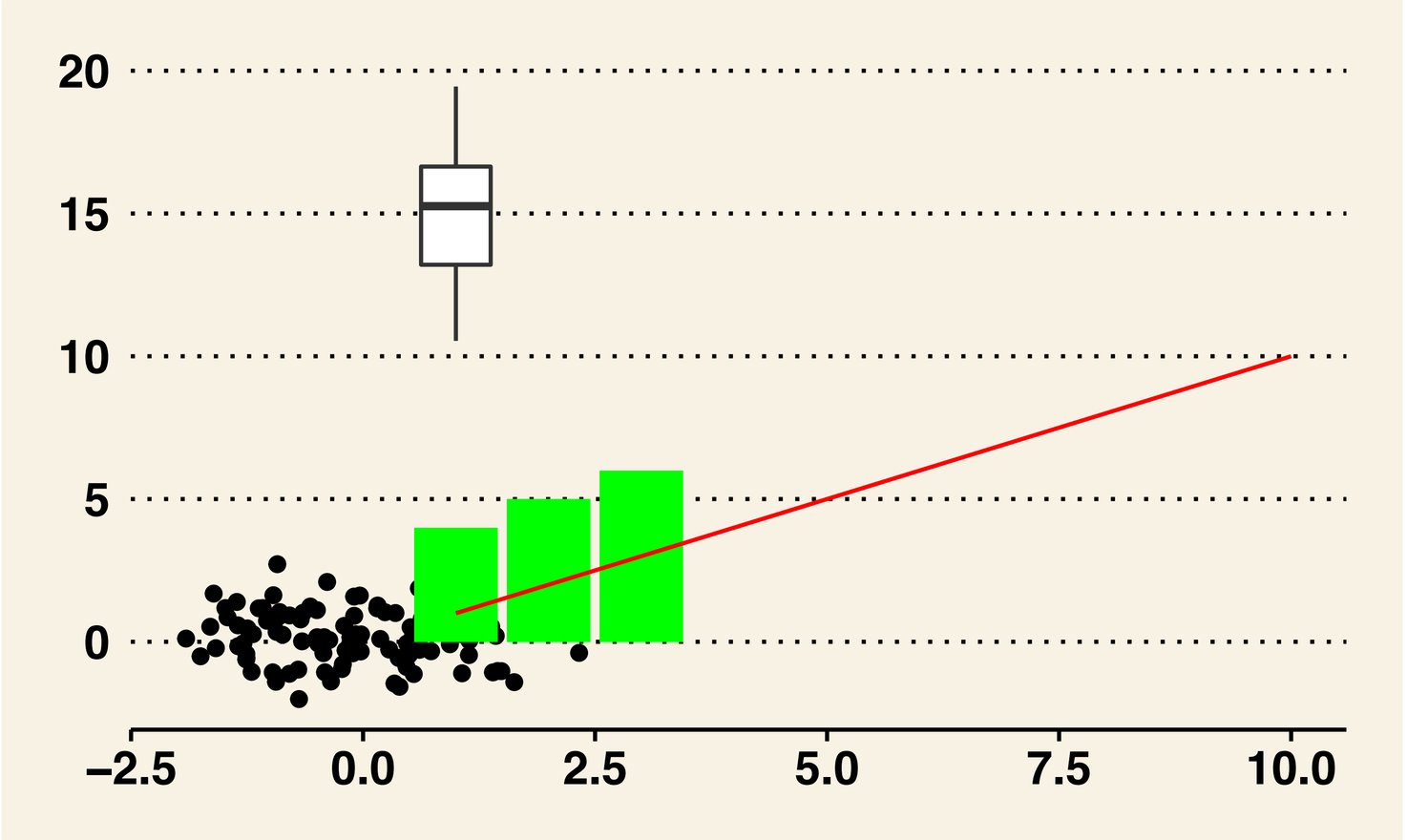
# Facet

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

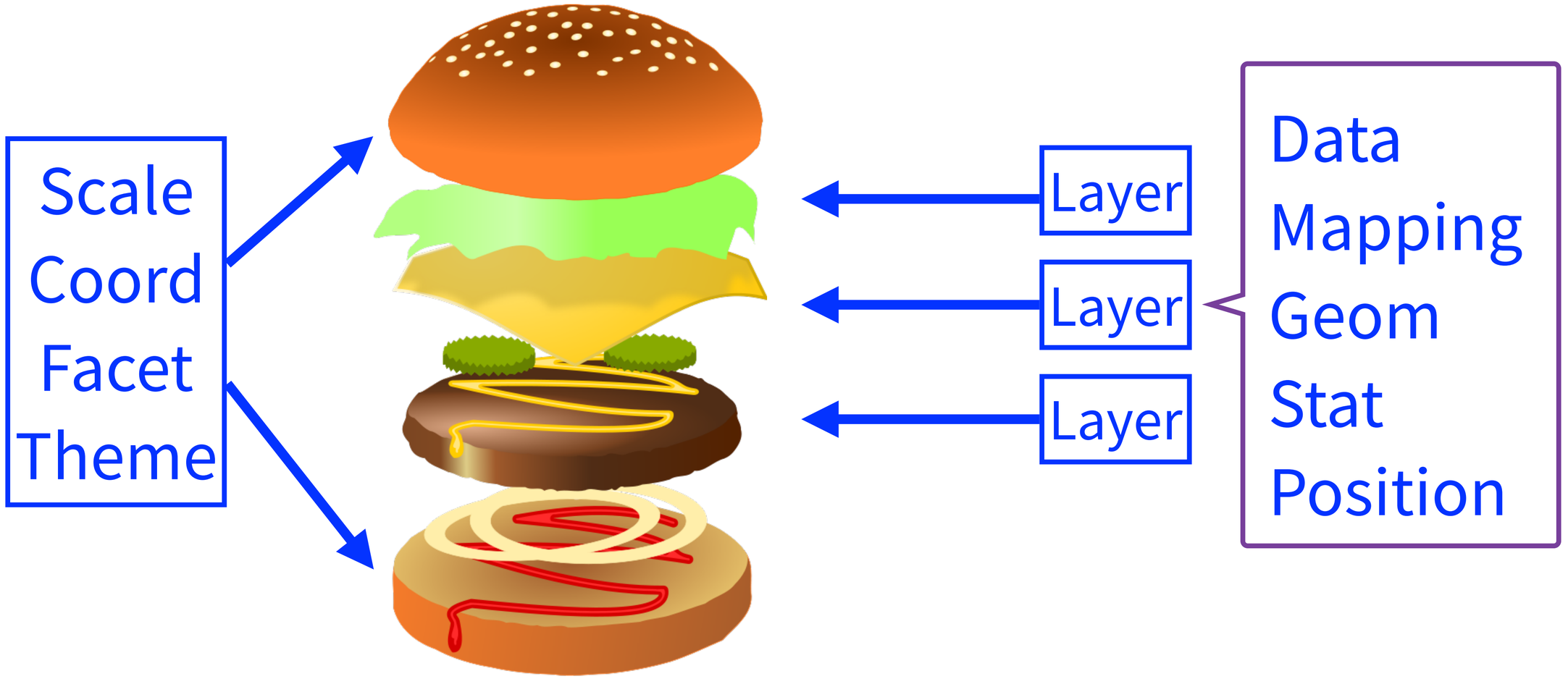


# Theme

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



# Layered Approach



# R & ggplot2  -- tips and tricks

- factors
- bar charts
- box plots
- histograms
- Rmarkdown