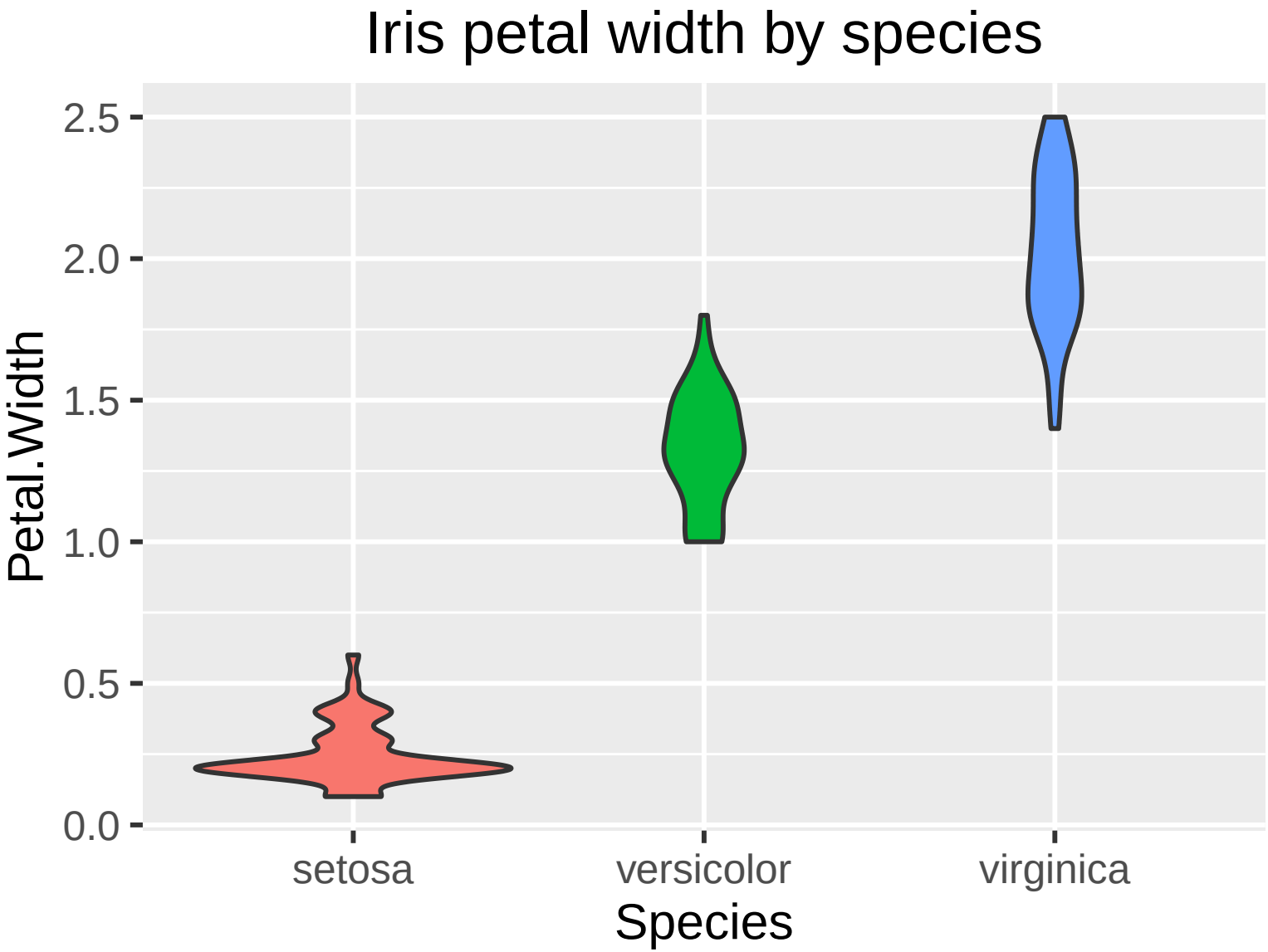
ggplot2 Introduction

Lingge Li

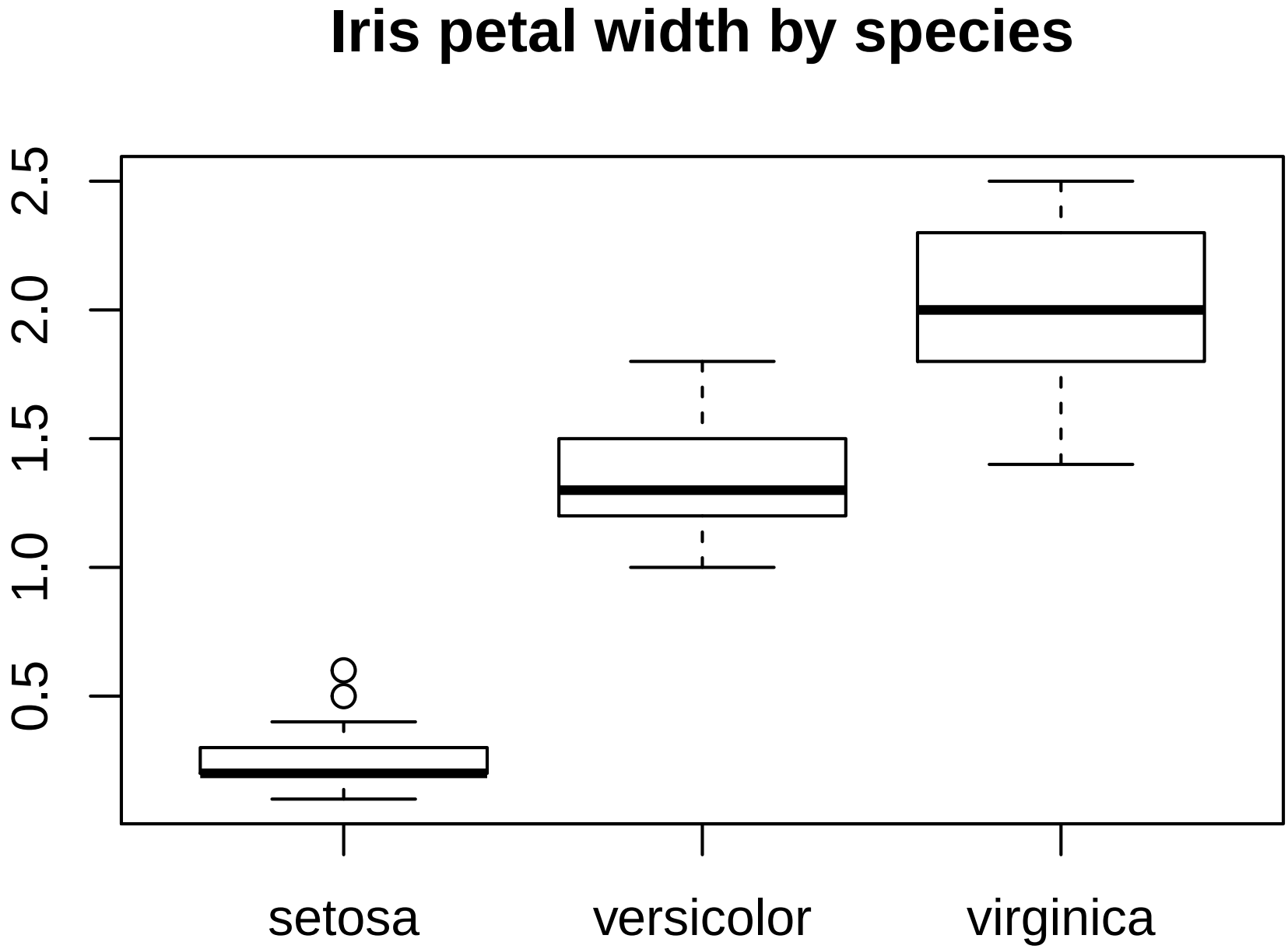
12/2/2016

ggplot2 looks awesome

- The beautiful style of ggplot2 graphics perhaps is most people's first impression



Base plot


Iris petal width by species
0.5
1.0
1.5
2.0
2.5
setosa
versicolor
virginica

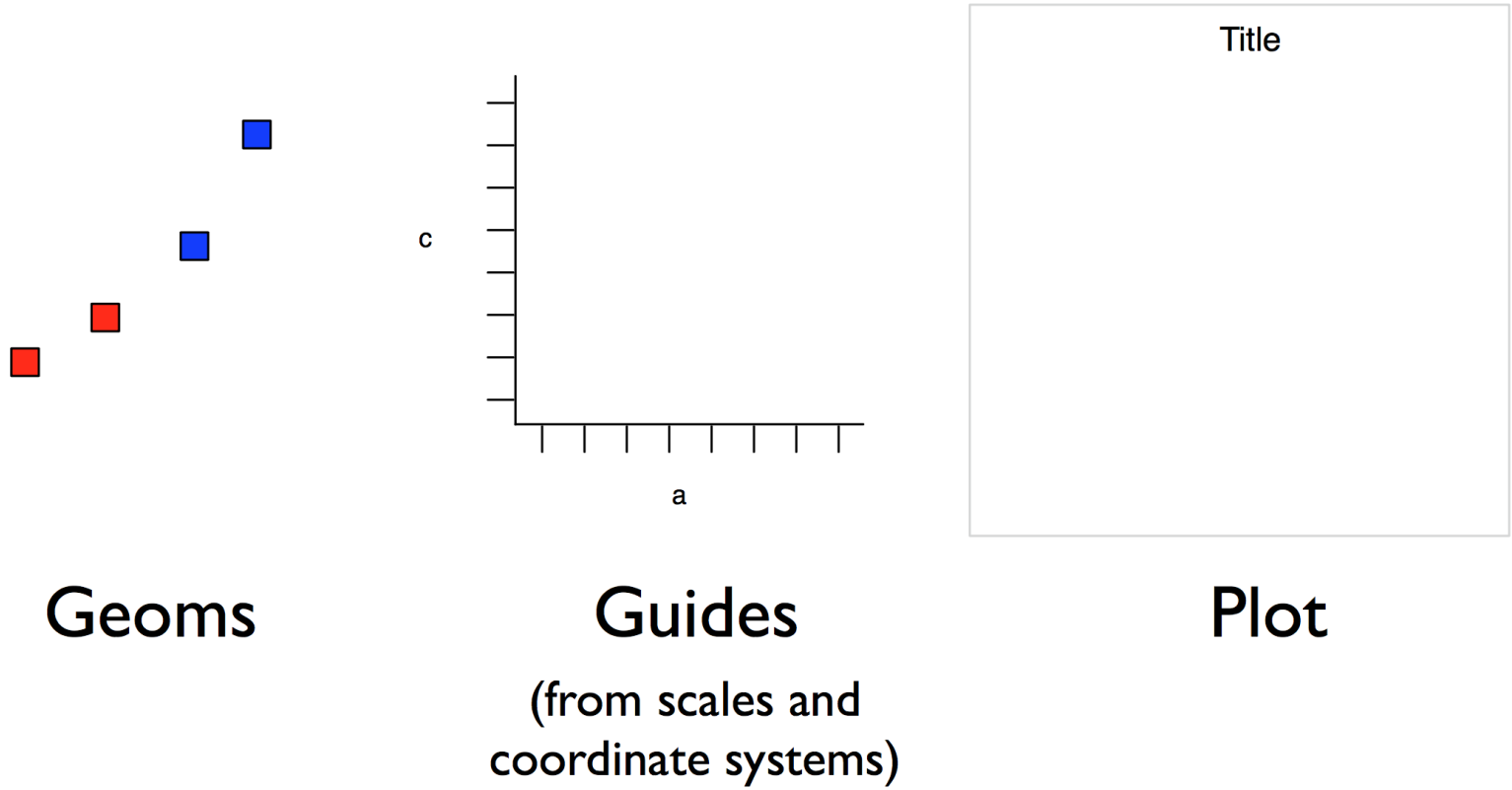
However...

- On second glance, the code seems slightly complicated

```
ggplot(data=iris, aes(x=Species, y=Petal.Width)) +
  geom_violin(aes(fill=Species)) +
  theme(legend.position='none') +
  labs(title='Iris petal width by species')
```

Layered grammar of graphics

- ggplot2 follows a specific grammar of graphics



Example taken from Hadley Wickham's book

`http://vita.had.co.nz/papers/layered-grammar.pdf`

How to make a plot

- Data
- Geometric objects (geom)
- Aesthetic mapping (aes)
- Statistical transformation (stat)
- Scales and coordinate system

Geom

- Wide range of geometric objects from basic points (geom_point) to statistical plots (geom_violin)
- Add geometric objects with $+$
- Multiple geometric objects on the same plot

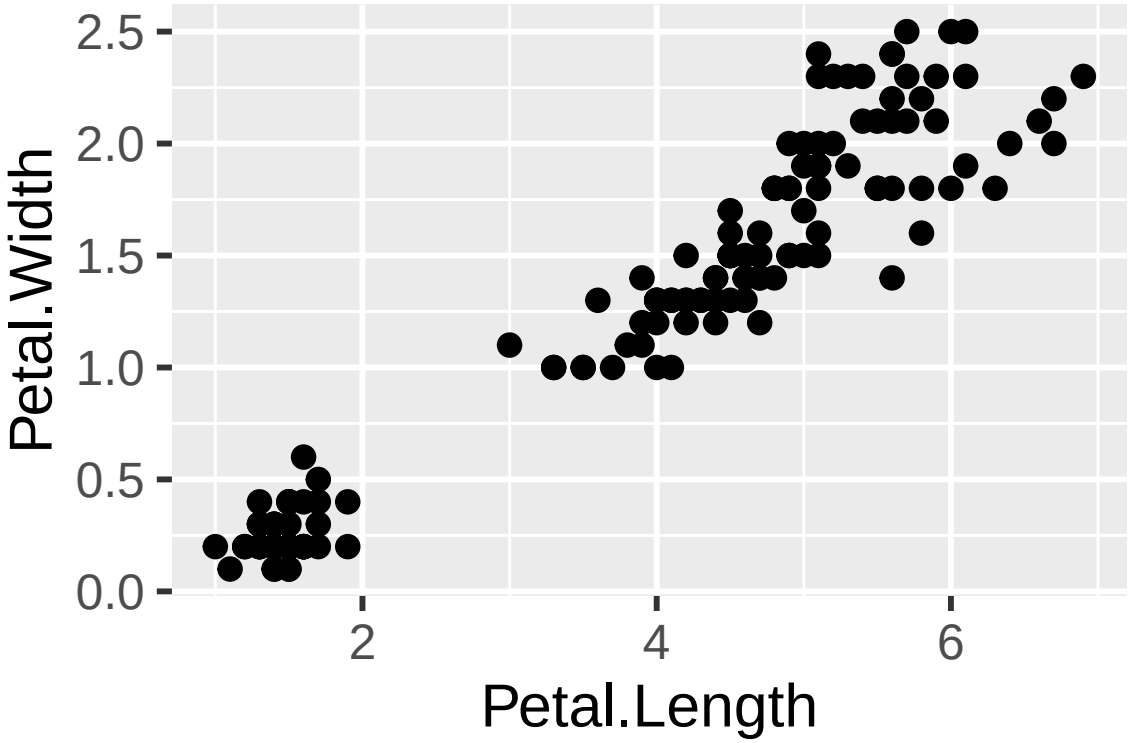
Data + mapping

- Must supply data and aes()
- Dataframe and xy positions always needed
- Each geometric object can have its own

```
geom_point(data, aes(x, y))
```

Scatterplot

```
ggplot() +
  geom_point(data=iris,
             aes(x=Petal.Length, y=Petal.Width))
```

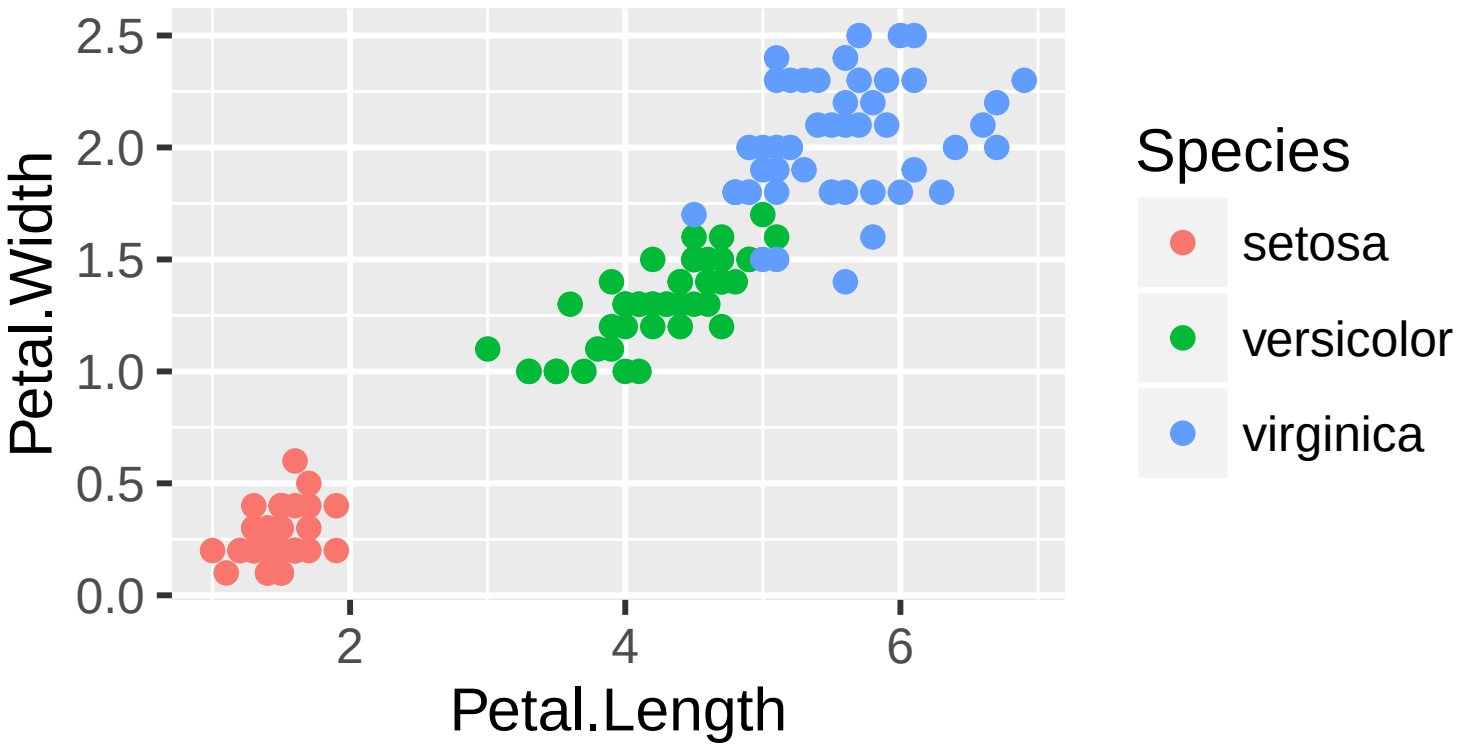



```
ggplot(data=iris, aes(x=Petal.Length, y=Petal.Width)) +
  geom_point()
```

Aesthetics

- Other aesthetic mapping arguments include colour, fill, shape, size. . .
- How would you change the shape

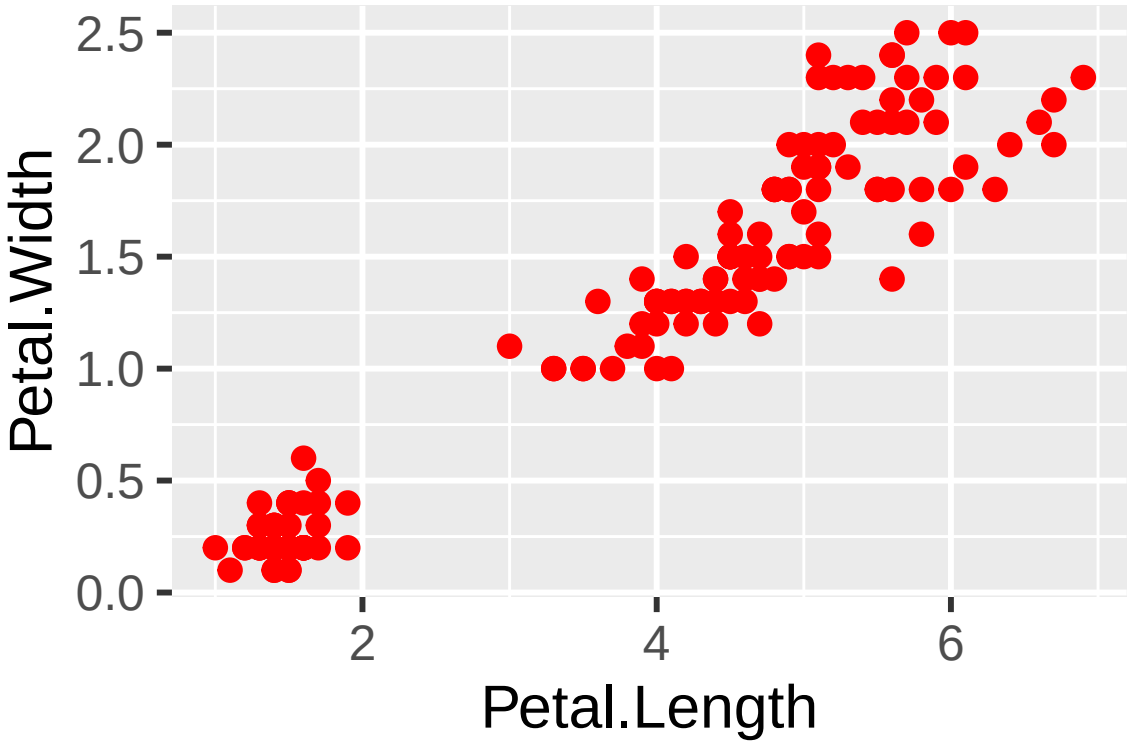
```
ggplot(data=iris, aes(x=Petal.Length, y=Petal.Width)) +
  geom_point(aes(colour=Species))
```



Mapping vs setting

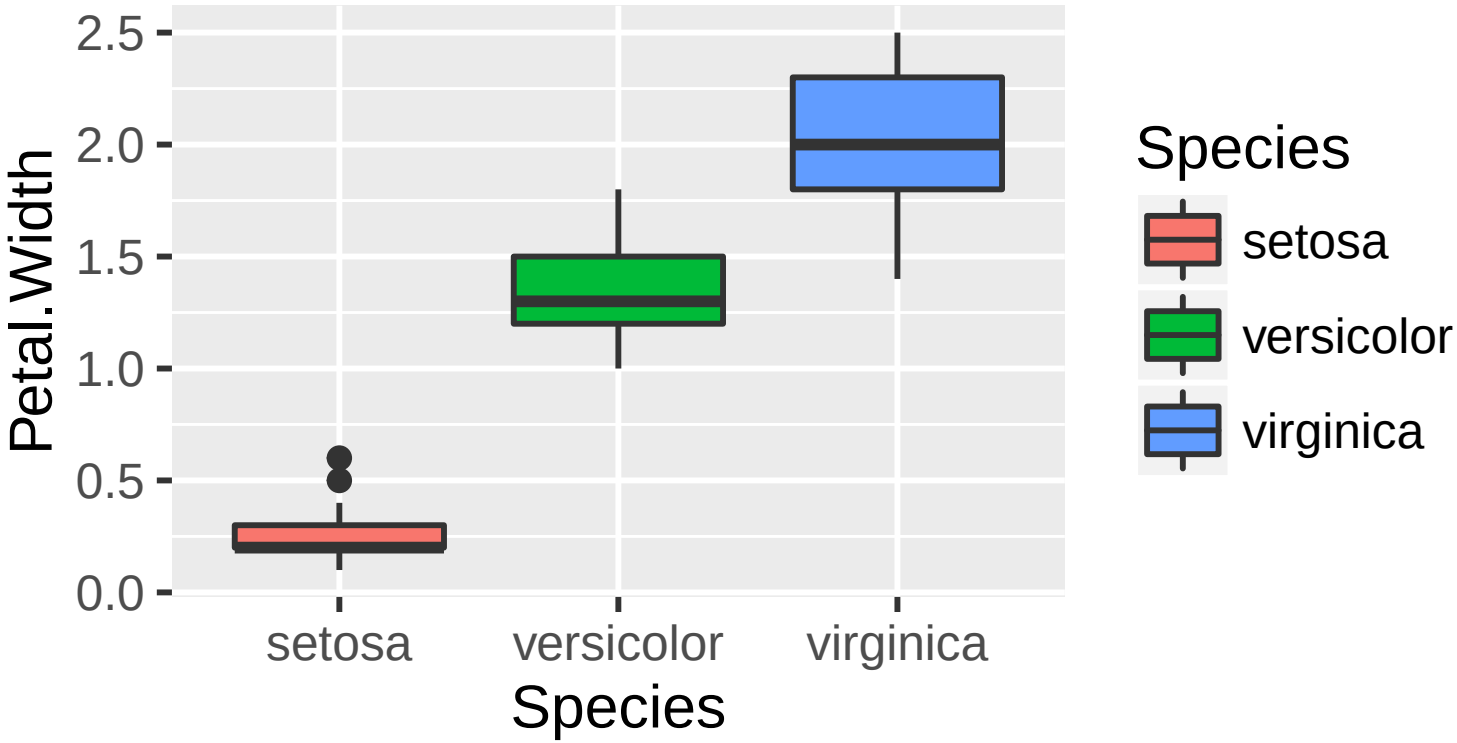
- What's the difference here

```
ggplot(data=iris, aes(x=Petal.Length, y=Petal.Width)) +
  geom_point(colour='red')
```



Boxplot

- How would you use geom_boxplot to create this

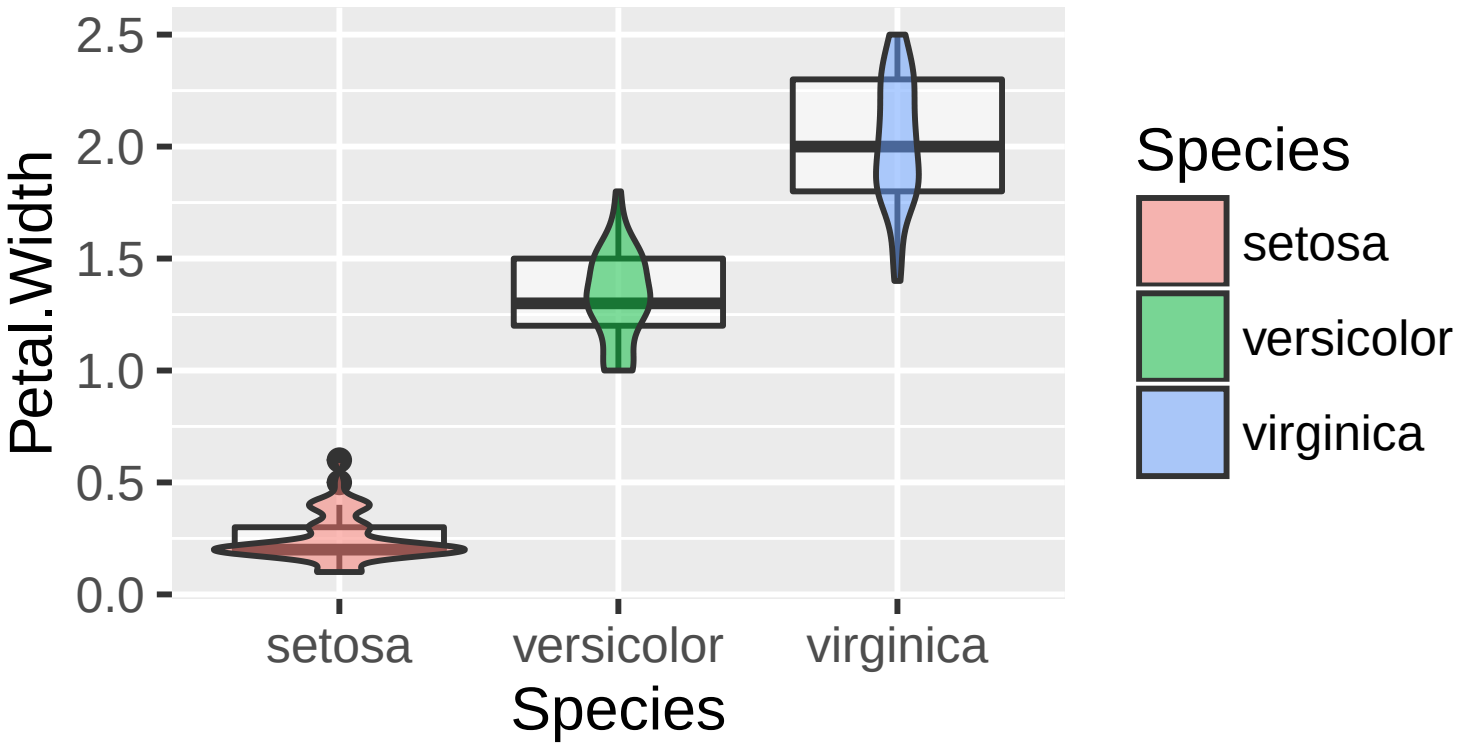


Layers

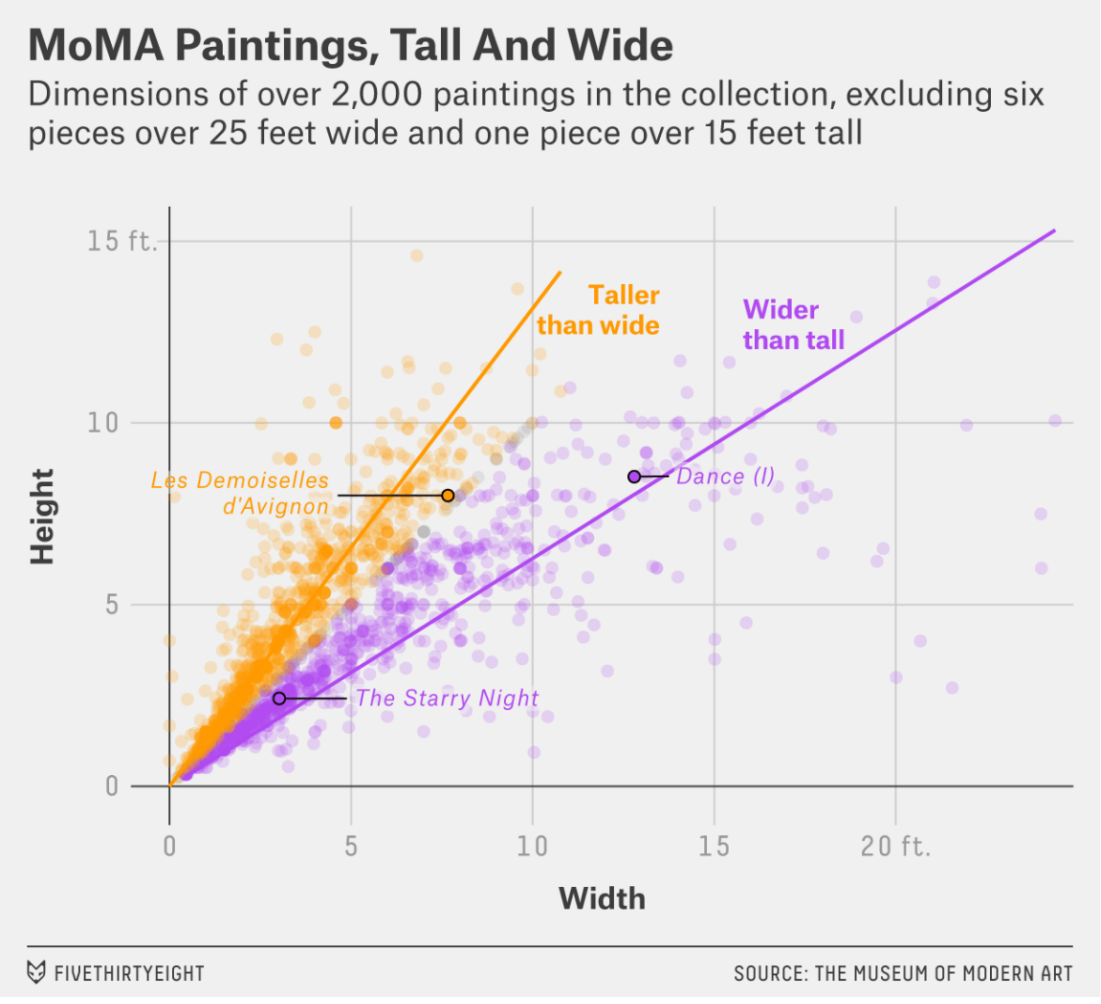
- Violin plots over boxplots

```
p <- ggplot(data=iris, aes(x=Species, y=Petal.Width)) +
  geom_boxplot(alpha=0.5)

p + geom_violin(aes(fill=Species), alpha=0.5)
```



Motivating example



http://fivethirtyeight.com/features/a-nerds-guide-to-the-2229-paintings-at-moma/

MoMA data

- Original dataset contains the entire MoMA collection

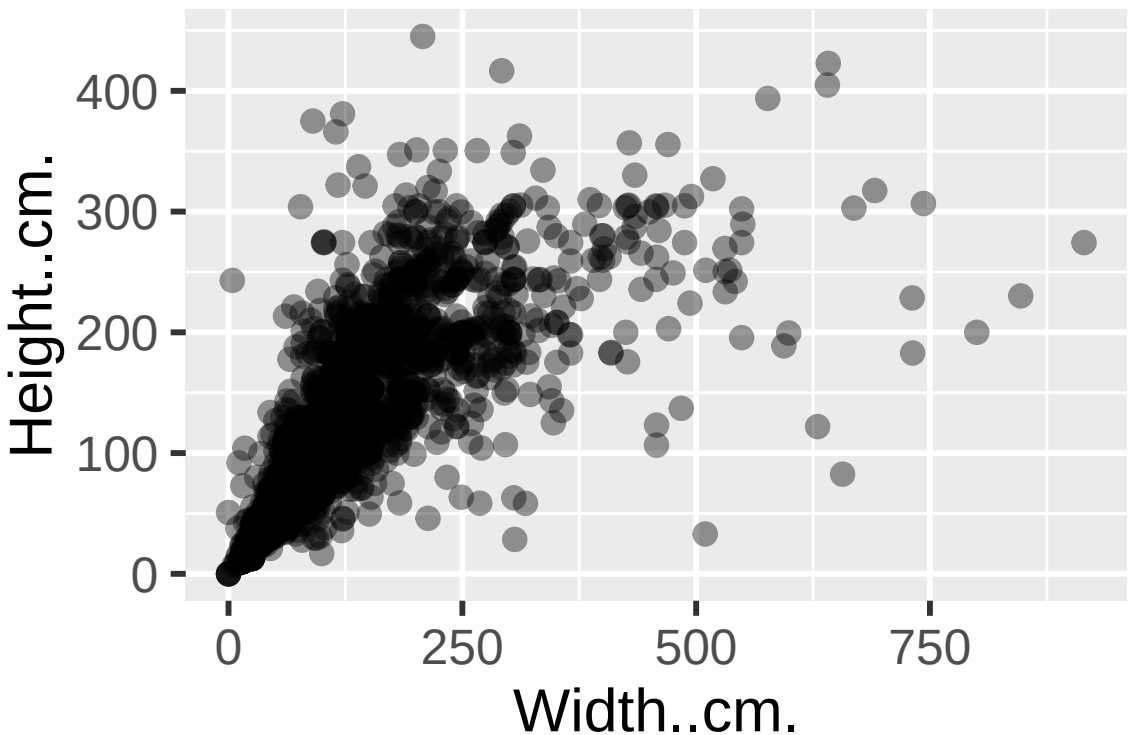
`https://github.com/MuseumofModernArt/collection`

- 2267 paintings by 989 artists
- Many interesting variables

Height vs width

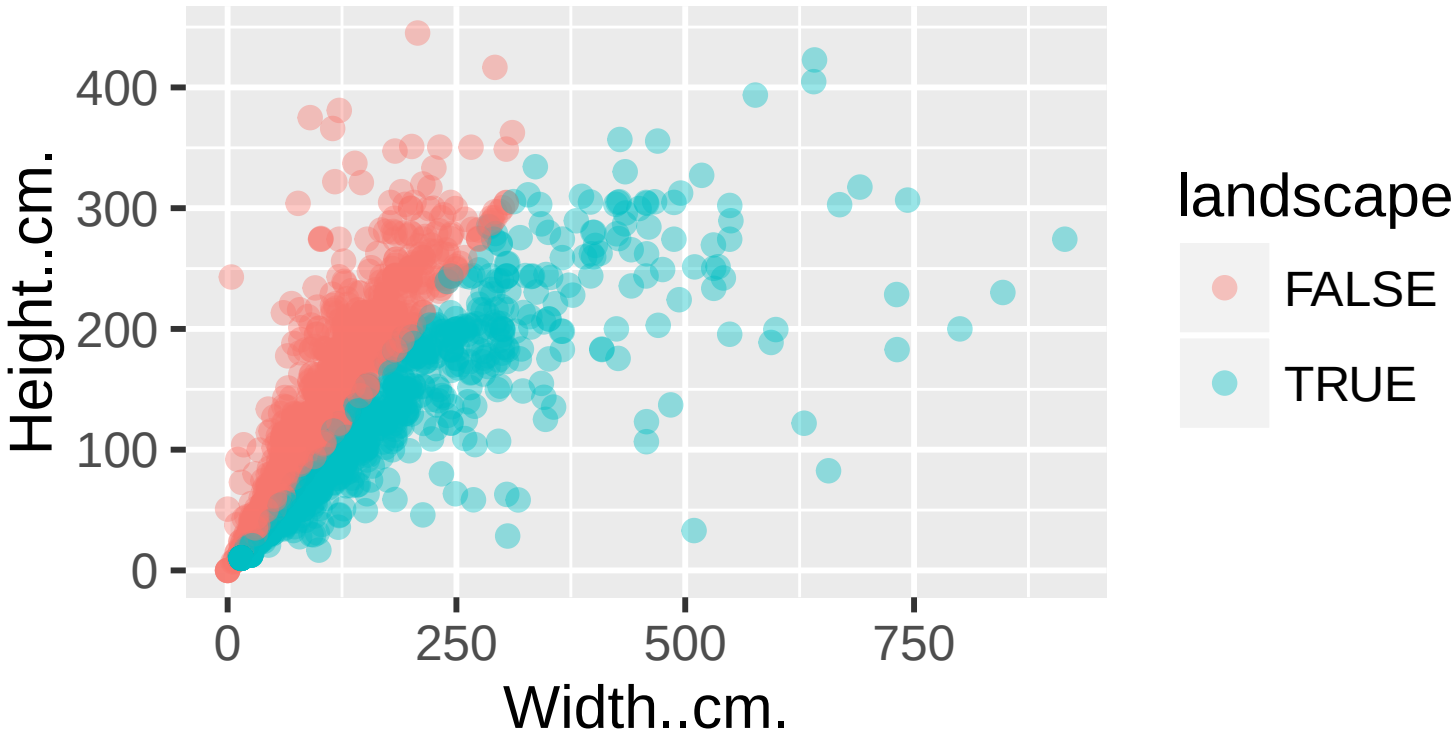
- Pre-process the data by removing outliers and missing values

```
Paintings <- read.csv("Paintings.csv")
Paintings <- Paintings[Paintings$Height..cm.<500,]
Paintings <- Paintings[Paintings$Width..cm.<1000,]
Paintings <- Paintings[!is.na(Paintings$Height..cm.),]
Paintings <- Paintings[!is.na(Paintings$Width..cm.),]
```



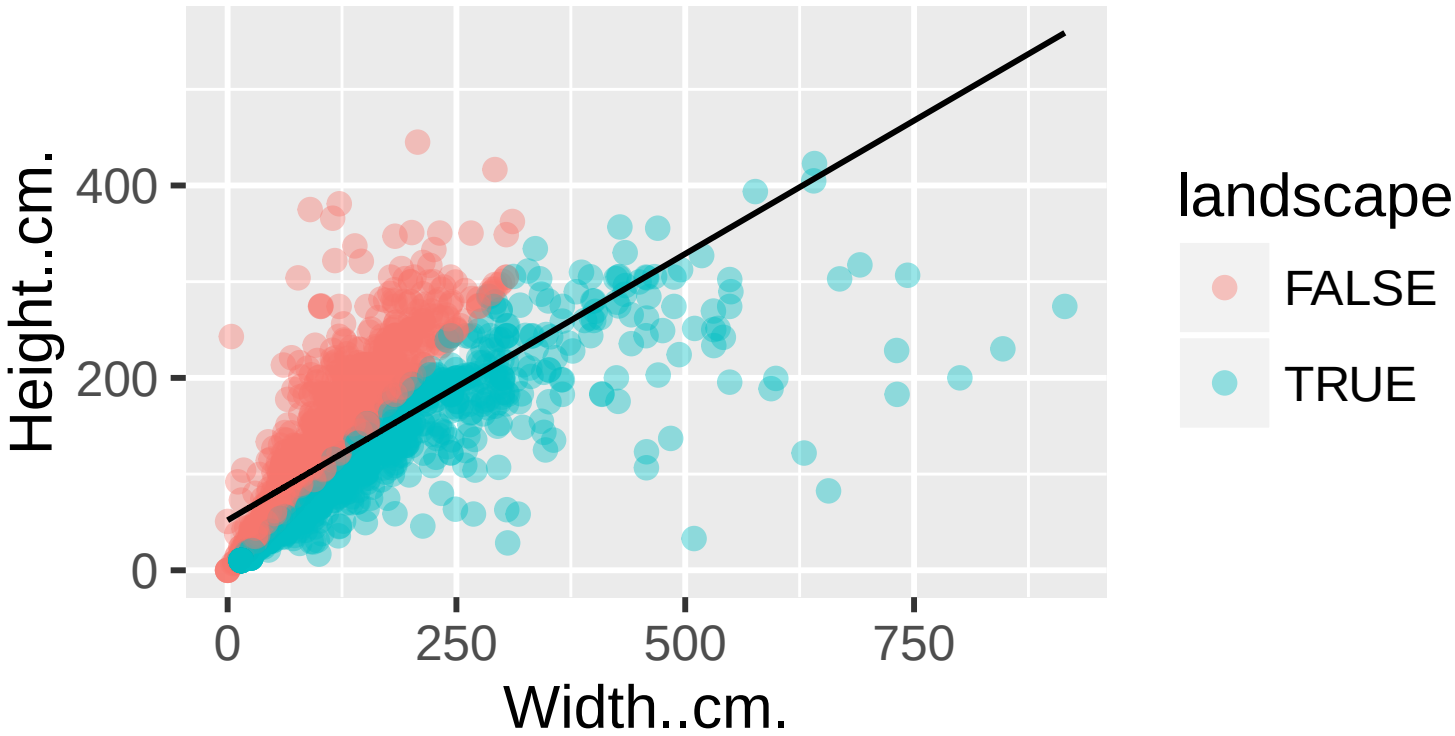
Color by orientation

- How can we create a new variable in the data frame



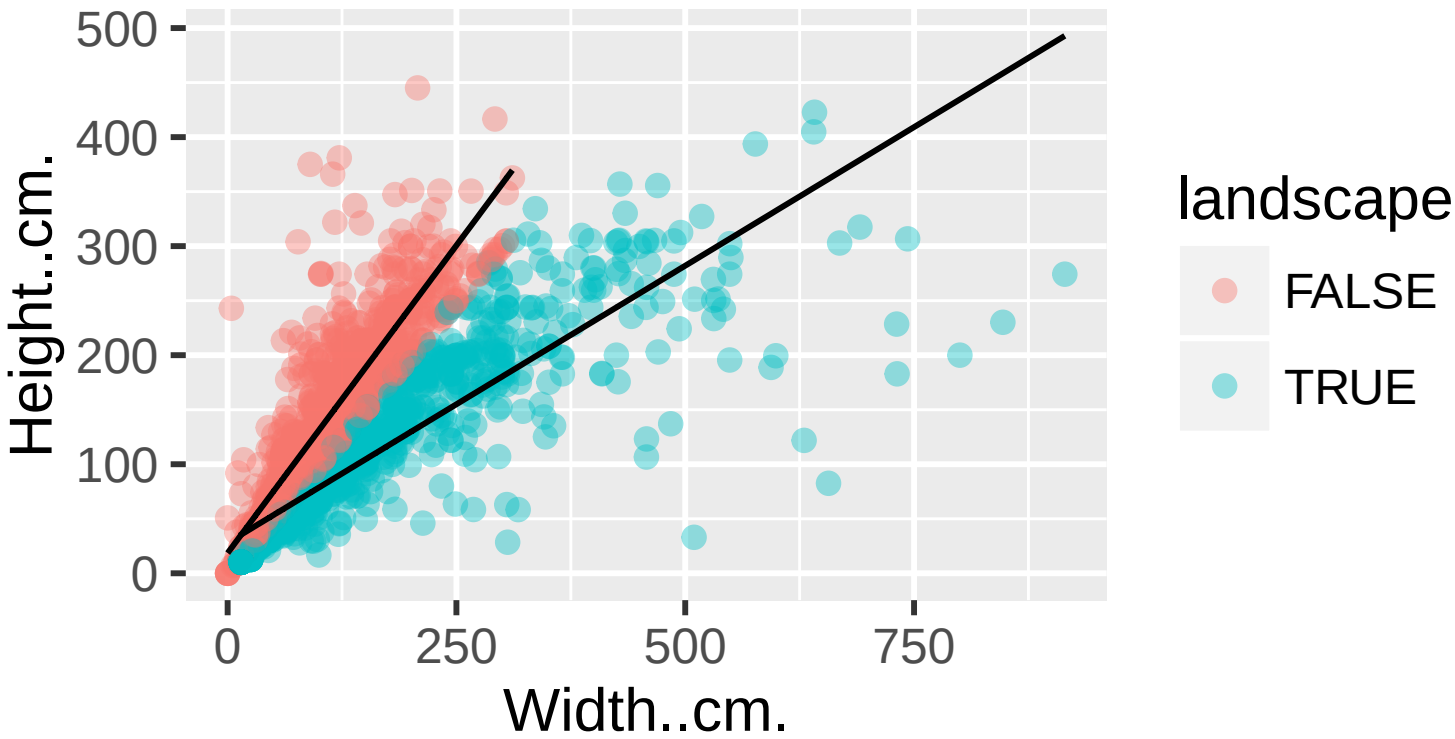
Regression line

- Plot the regression line of height~width with geom_line() (Hint: lm(Y~X)$fitted.values)



Separate lines

- How can we draw two separate regression lines (Hint: use different subsets of the data)

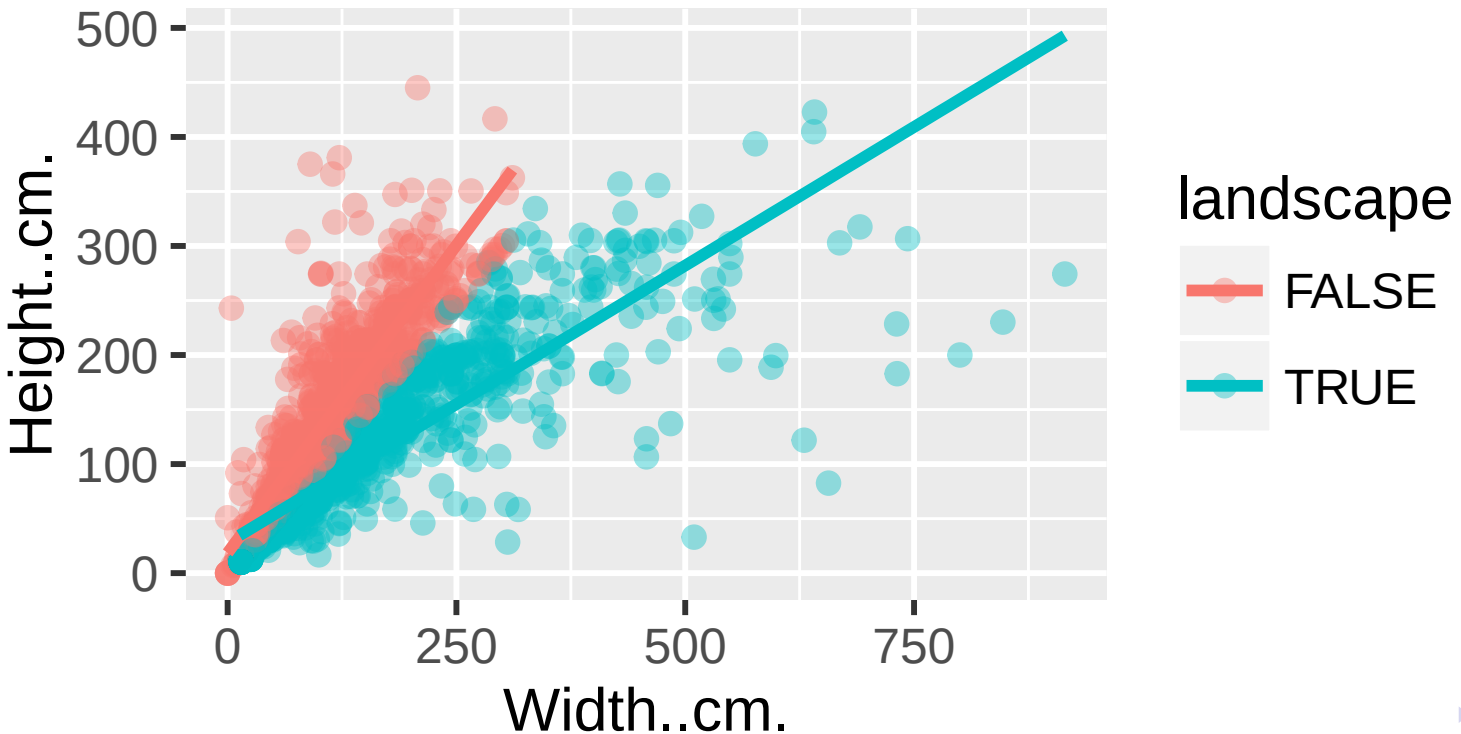


Smoother

- There is actually a much easier way with geom_smooth()

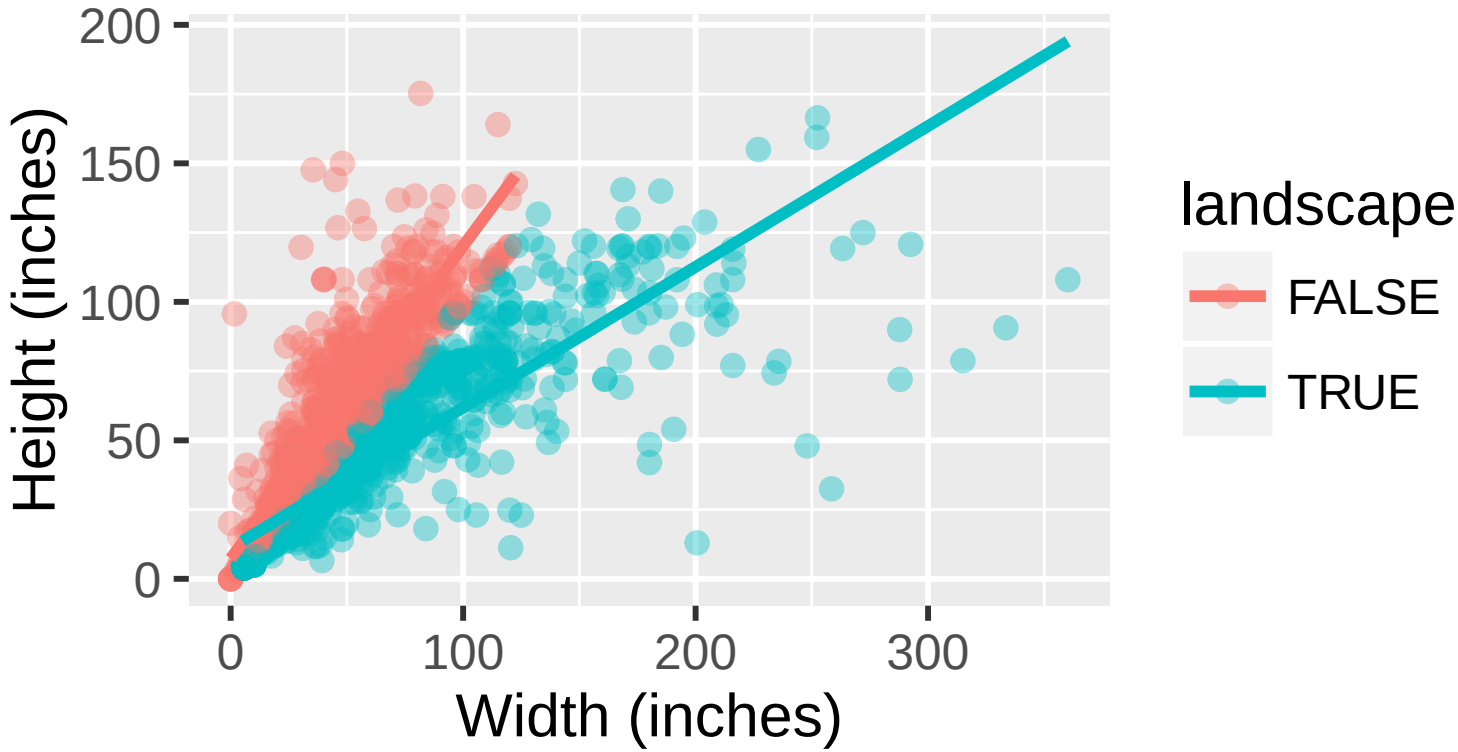
```
p <- ggplot(data=Paintings,
            aes(x=Width..cm., y=Height..cm.)) +
  geom_point(aes(colour=landscape), alpha=0.4)

p + geom_smooth(aes(colour=landscape),
                method='lm', se=FALSE)
```



Scales

- Maybe the unit should be inch instead of centimeter (Hint: `http://docs.ggplot2.org/0.9.3/scale_continuous.html`)

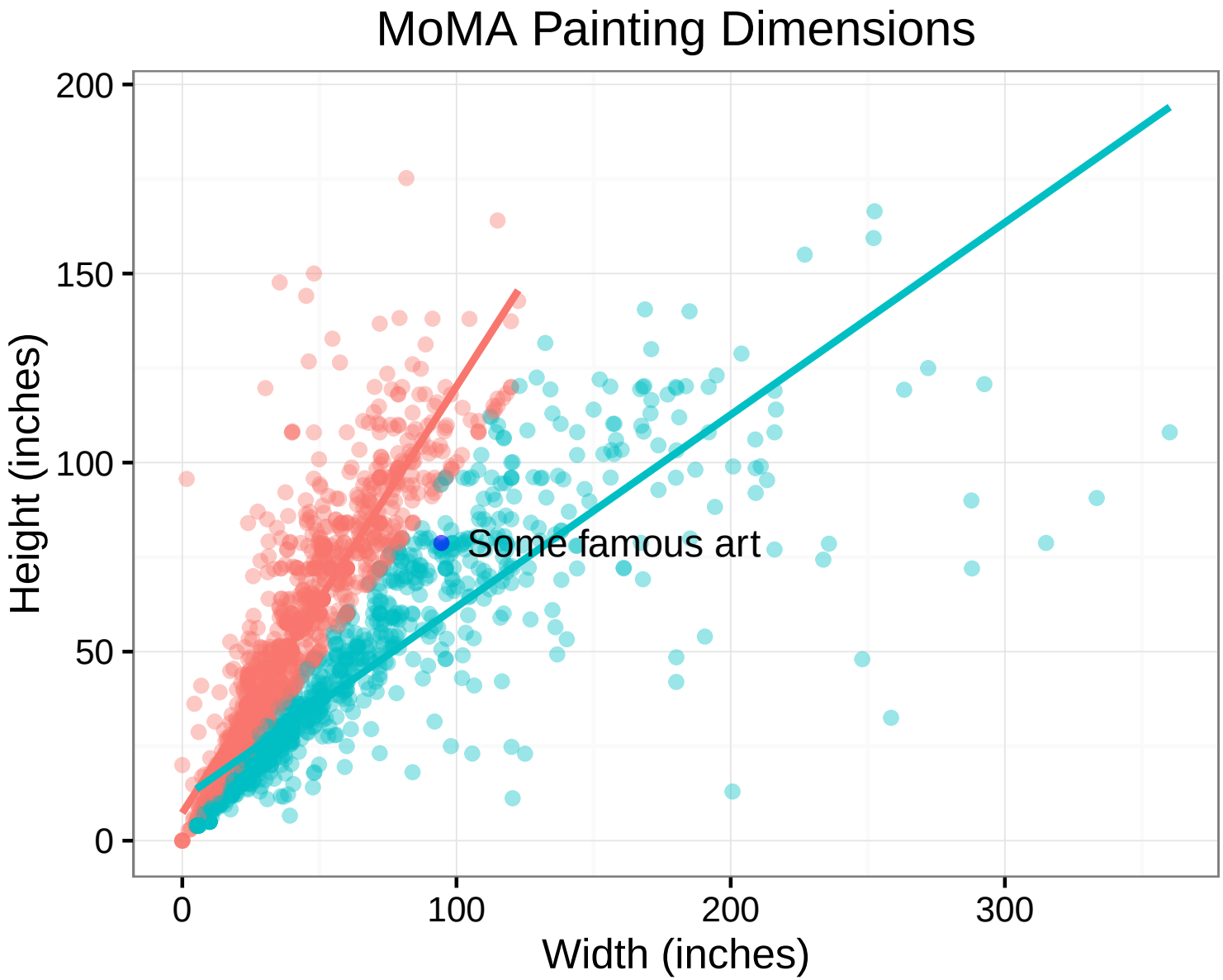


Details

- We can change many other details (http://docs.ggplot2.org/0.9.2.1/theme.html)

```
ggplot(data=Paintings,
       aes(x=0.393701*Width..cm., y=0.393701*Height..cm.))
  geom_point(aes(colour=landscape), alpha=0.4) +
  geom_smooth(aes(colour=landscape), method='lm', se=FALSE)
  scale_x_continuous(name='Width (inches)') +
  scale_y_continuous(name='Height (inches)') +
  theme_bw() + theme(legend.position='none') +
  annotate('point', x=0.393701*240, y=0.393701*200,
           colour='blue', alpha=0.6) +
  annotate('text', x=0.393701*400, y=0.393701*200,
           label='Some famous art') +
  ggtitle('MoMA Painting Dimensions')
```

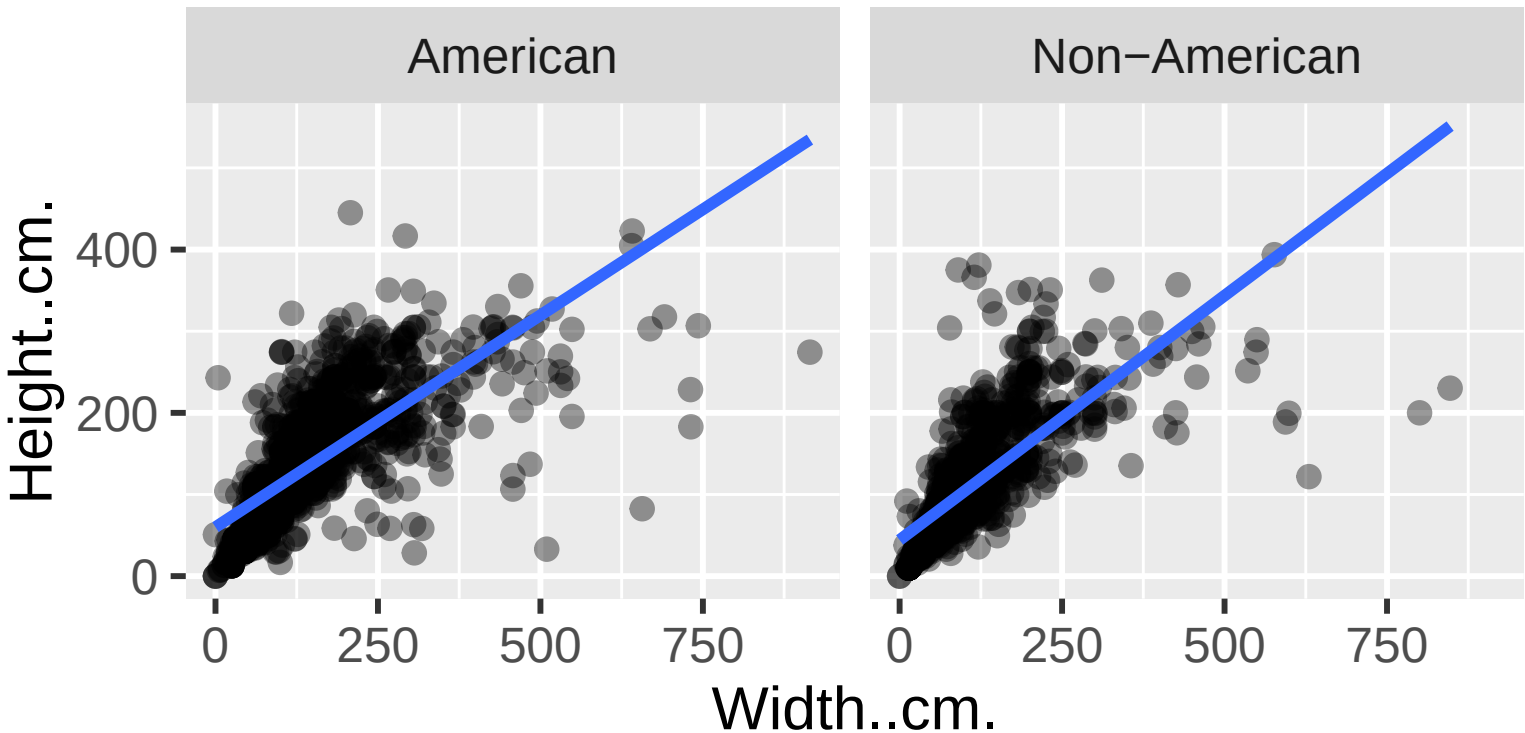
End result


MoMA Painting Dimensions
Height (inches)
Width (inches)
200
150
100
50
0
0
100
200
300
Some famous art

Facet

- We can make similar plots for different regions with facet

```
Paintings$American <- ifelse(Paintings$Nationality=='Americ
                             'American', 'Non-American')
ggplot(data=Paintings, aes(x=Width..cm., y=Height..cm.)) +
  geom_point(alpha=0.4) +
  geom_smooth(method='lm', se=FALSE) +
  facet_grid(.~American)
```



Multiple plots

- Multiplot function

`http://www.cookbook-r.com/Graphs/Multiple_graphs_on_one_page_(ggplot2)/`

- gridExtra package

`https://cran.r-project.org/web/packages/gridExtra/vignettes/arrangeGrob.html`

Last comment

- I hope you have gained a better understanding of ggplot2. There are plenty of tutorials and other resources online.

`https://www.rstudio.com/wp-content/uploads/2015/03/ggplot2-cheatsheet.pdf`
`http://www.cookbook-r.com/Graphs/`
`http://stackoverflow.com/questions/tagged/ggplot2`