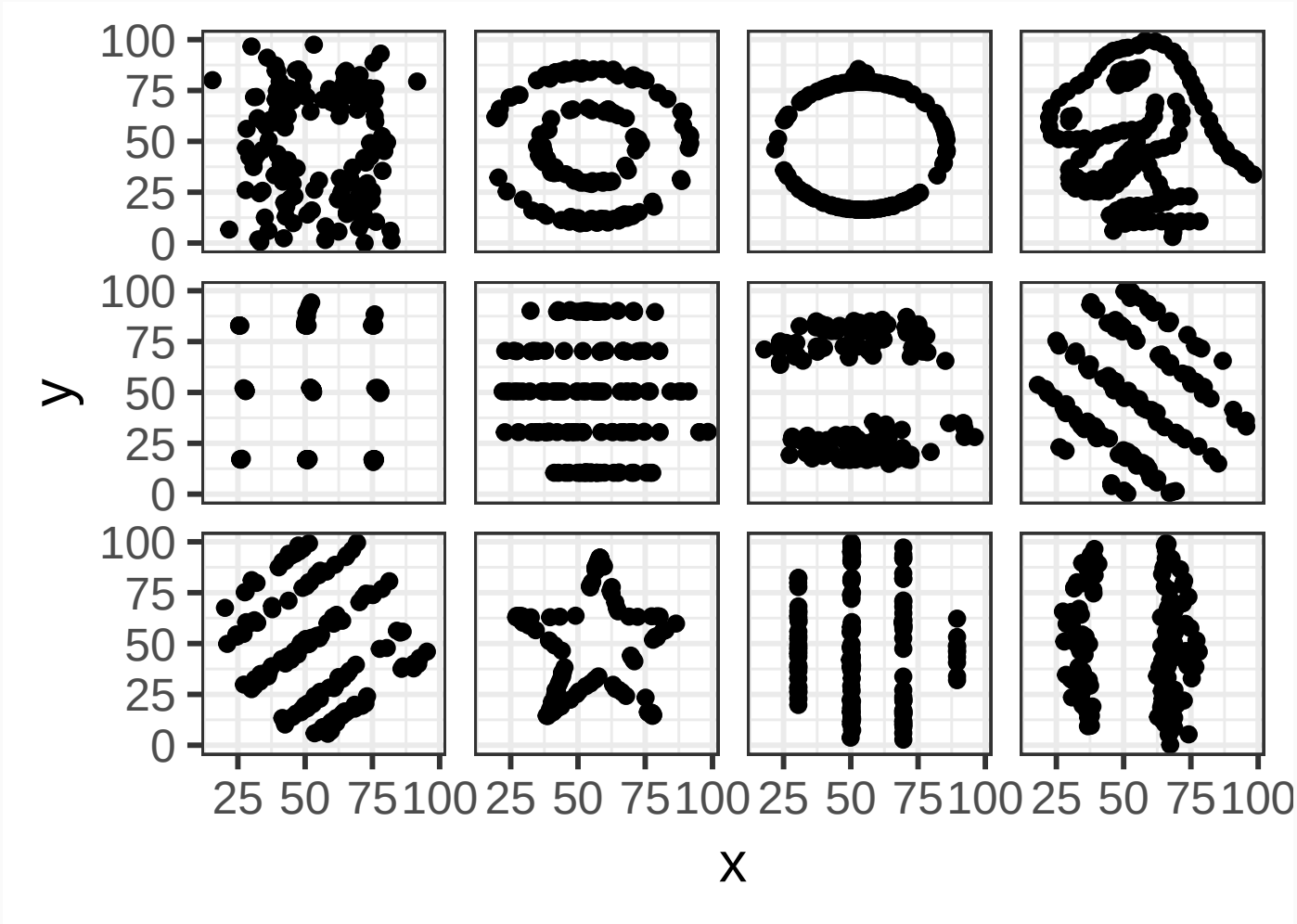
Data visualisation with ggplot2

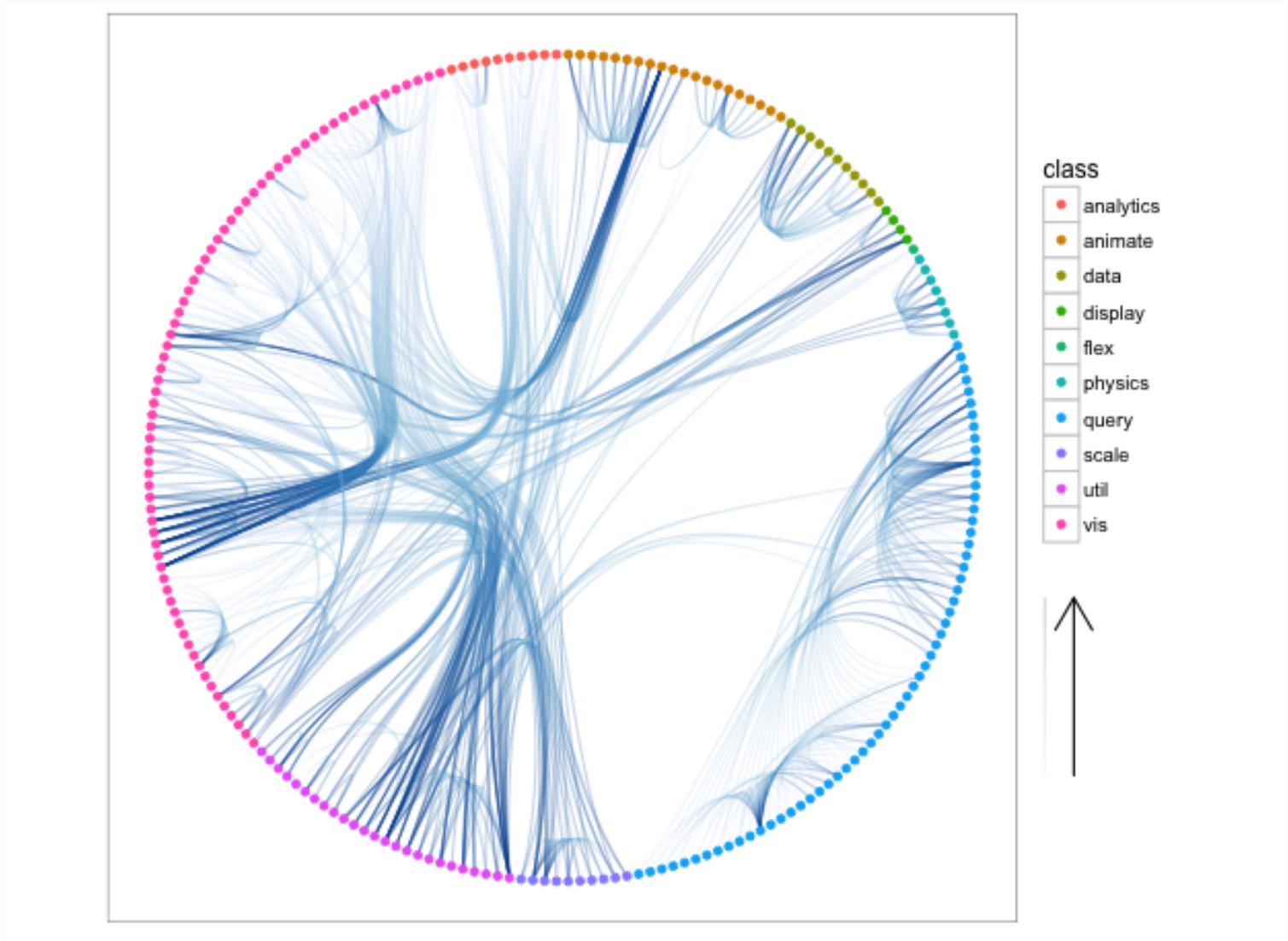
Francisco Rodriguez-Sanchez (@frod_san)

Always plot data!



https://github.com/stephlocke/datasauRus

Made with ggplot

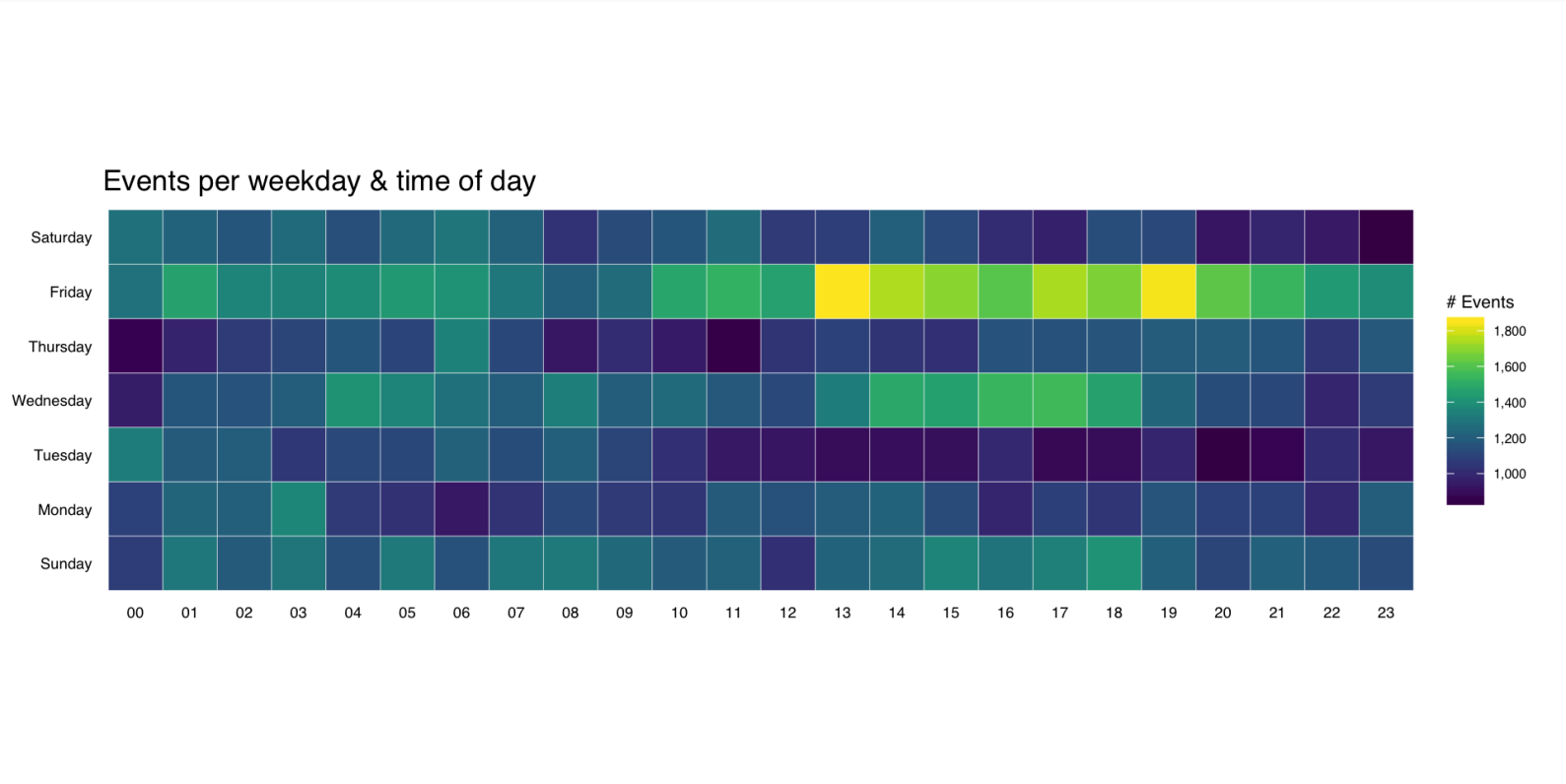


https://github.com/thomasp85/ggraph

Made with ggplot

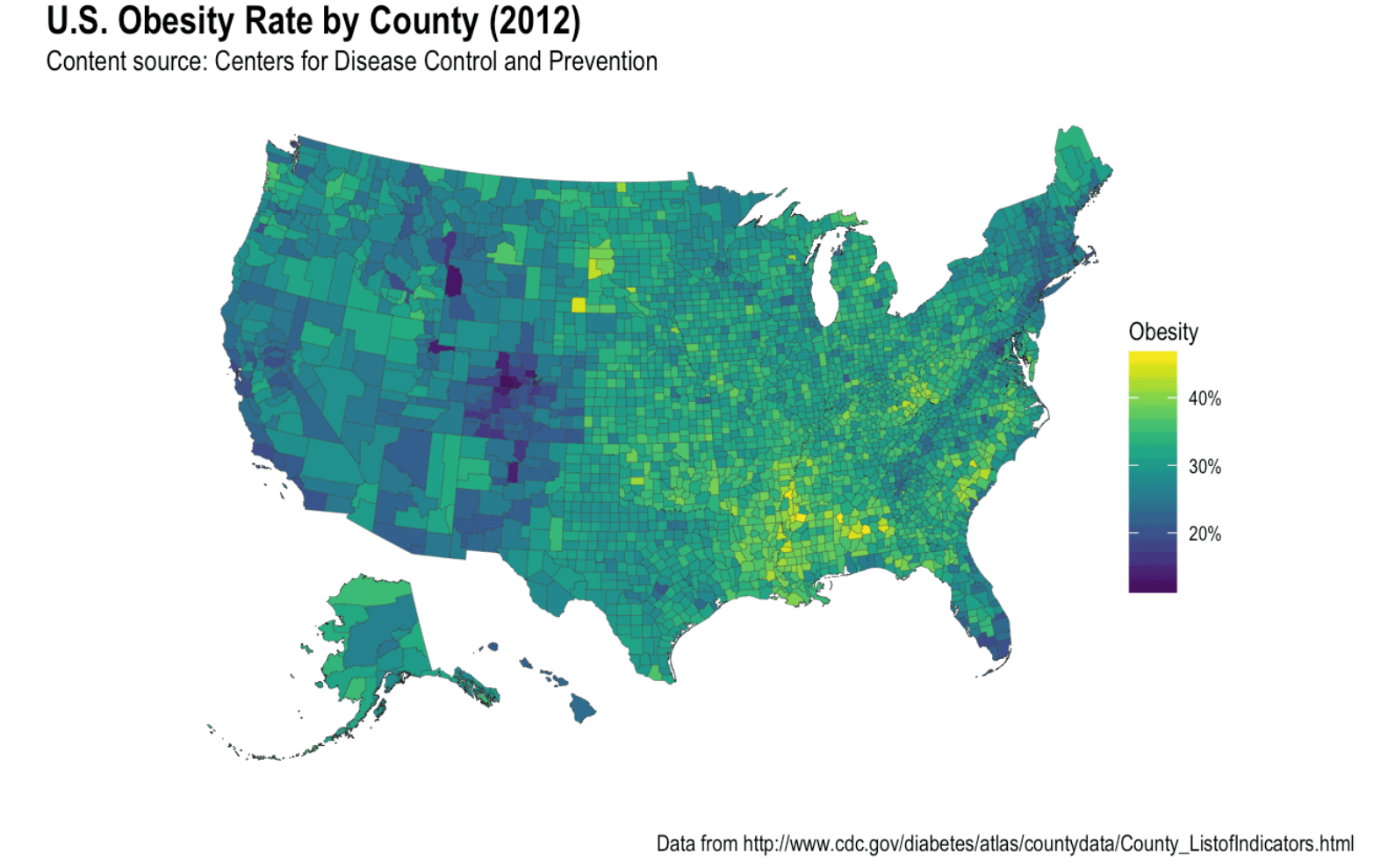
http://spatial.ly/2012/02/great-maps-ggplot2/

Made with ggplot



https://rud.is/b/2016/02/14/making-faceted-heatmaps-with-ggplot2/

Made with ggplot

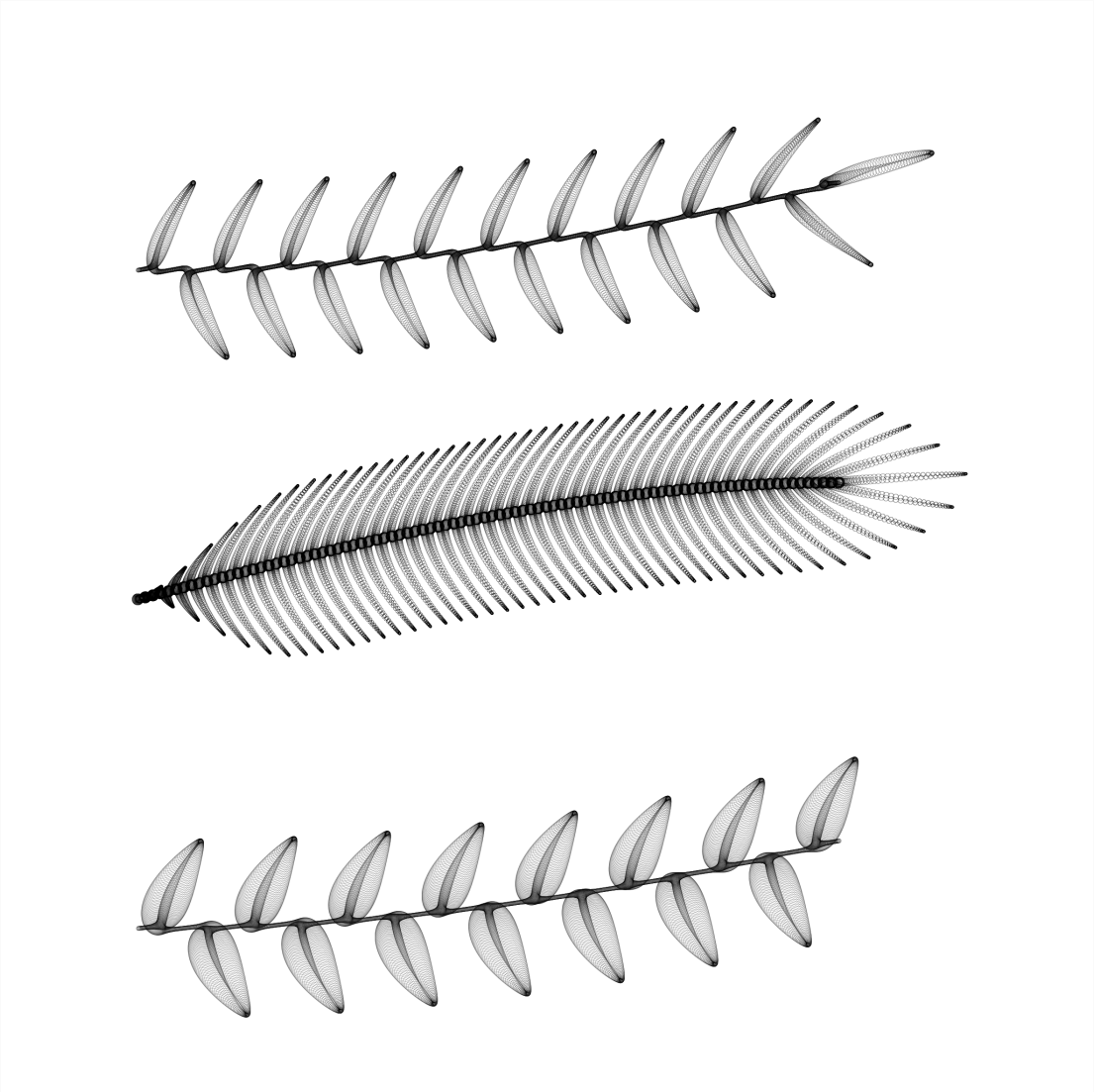


https://rud.is/b/2016/03/29/easier-composite-u-s-choropleths-with-albersusa/

Made with ggplot

https://guangchuangyu.github.io/ggtree/

Made with ggplot

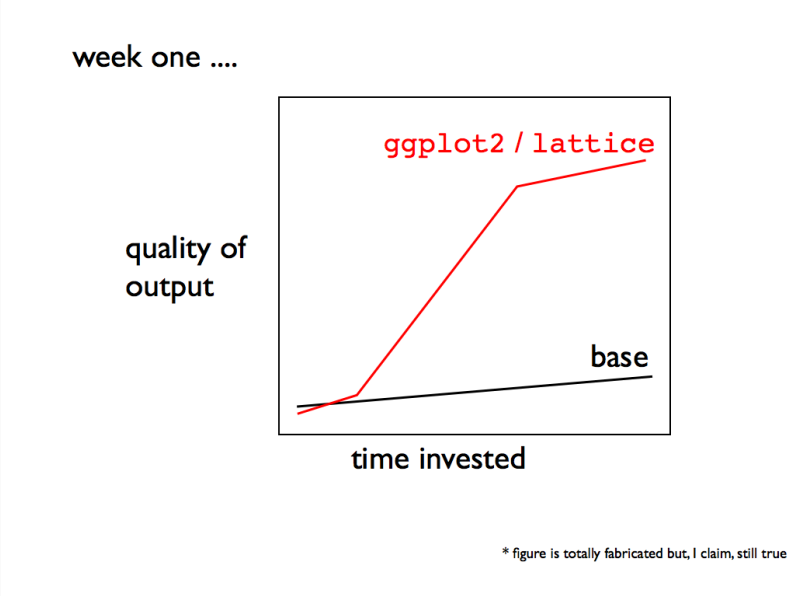


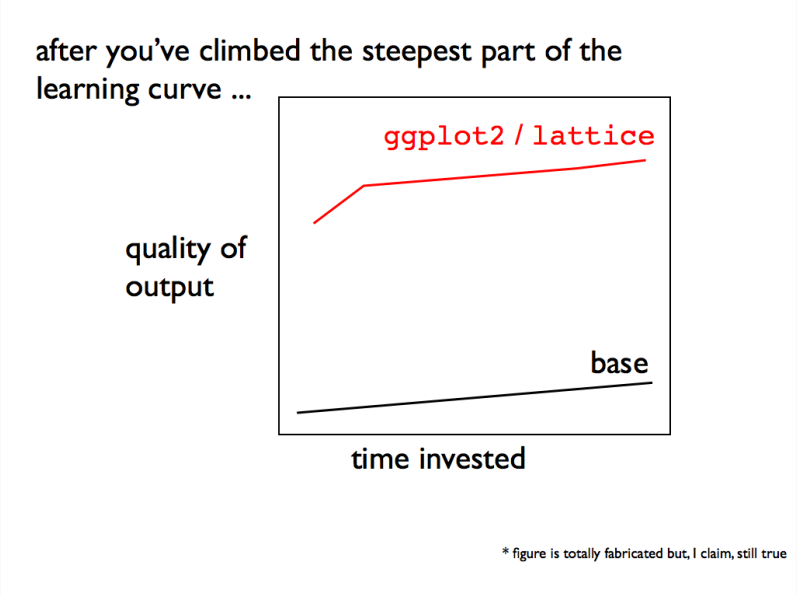
https://github.com/marcusvolz/mathart

Why ggplot

- Extremely powerful and flexible
- Consistent (grammar of graphics)
- Very powerful user base and active development

At the beginnning it's hard, but then it pays off



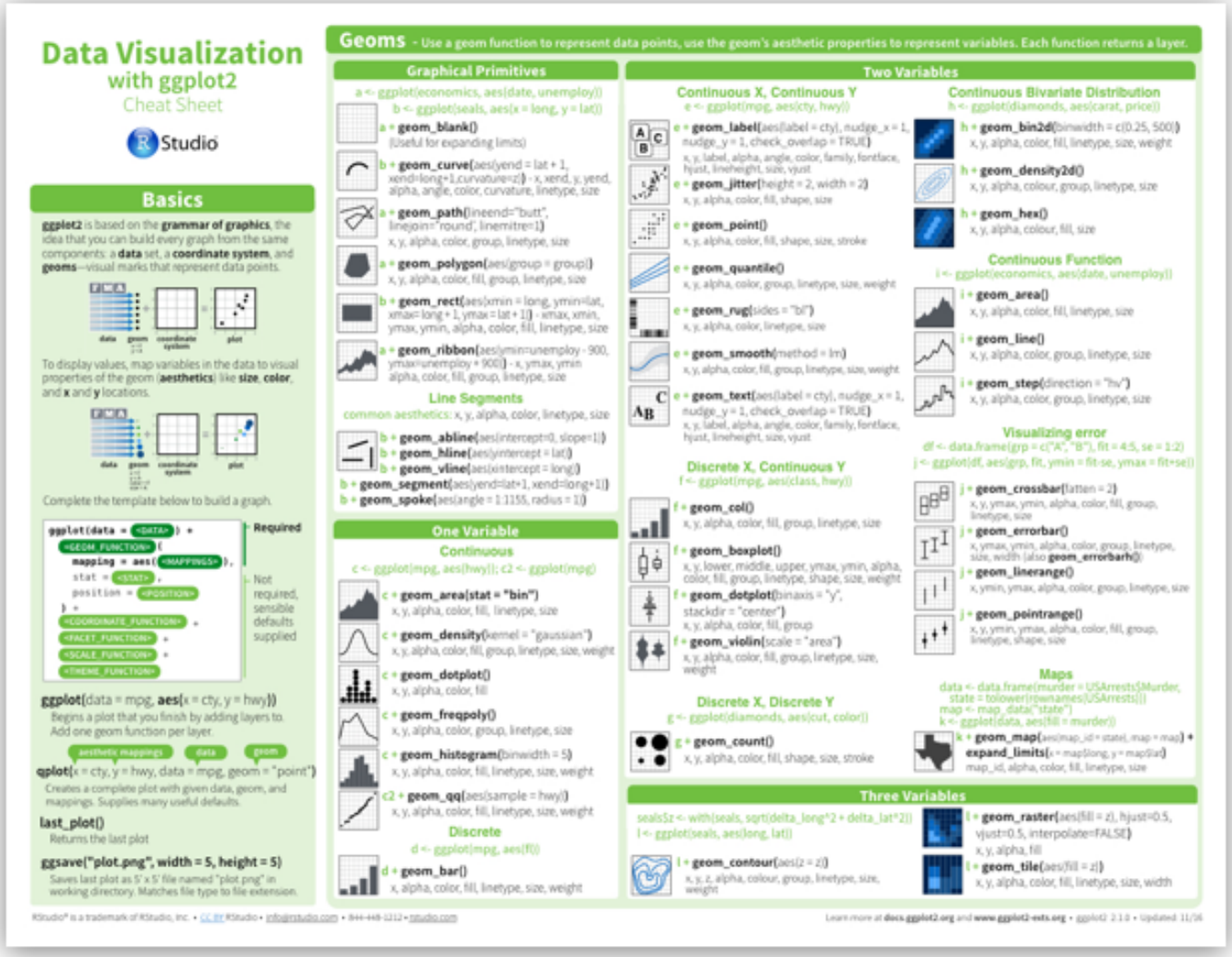


Source: https://github.com/jennybc/ggplot2-tutorial

Very good documentation and tutorials

- Official ggplot2 documentation
- ggplot2 book
- R graphics cookbook and Cookbook for R
- Beautiful plotting in R: A ggplot2 cheatsheet
- Introduction to ggplot2
- Tutorial: ggplot2
- How to format plots for publication using ggplot2
- Visualising data with ggplot2
- Data Visualization with R and ggplot2
- ggplot2 tutorial
- Data visualisation chapter in R for Data Science
- The complete ggplot2 tutorial
- Data visualization: a practical introduction (K. Healy)
- Fundamentals of data visualization (C. Wilke)

Cheatsheet



https://www.rstudio.com/resources/cheatsheets/

Repos of figures + code

- R graph catalog
- From Data to Viz
- The R graph gallery
- R graph gallery
- Cookbook for R: Graphs
- Graphical data analysis with R
- IEG figures

Find answers for all your questions in Stack Overflow

stackoverflow

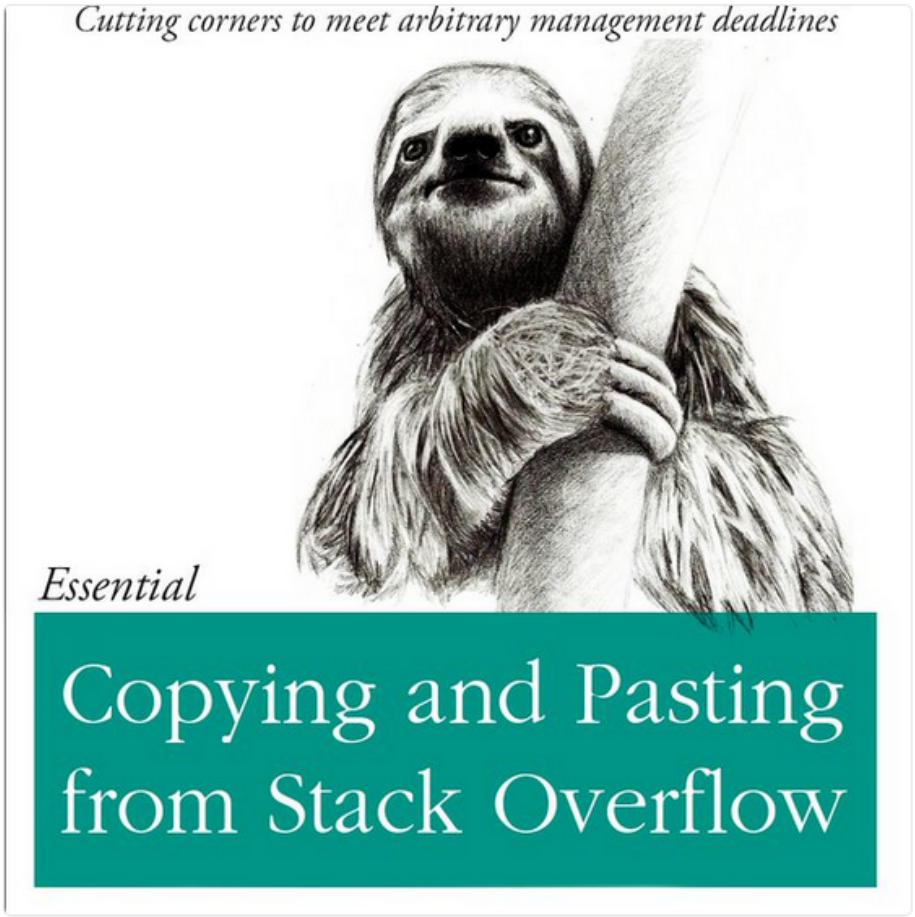
Search

ggplot2

36,854 results

DEV The Practical Dev
@ThePracticalDev
Follow

The last programming book you'll ever need



Building a ggplot figure

Our example dataset: tree heights and DBH

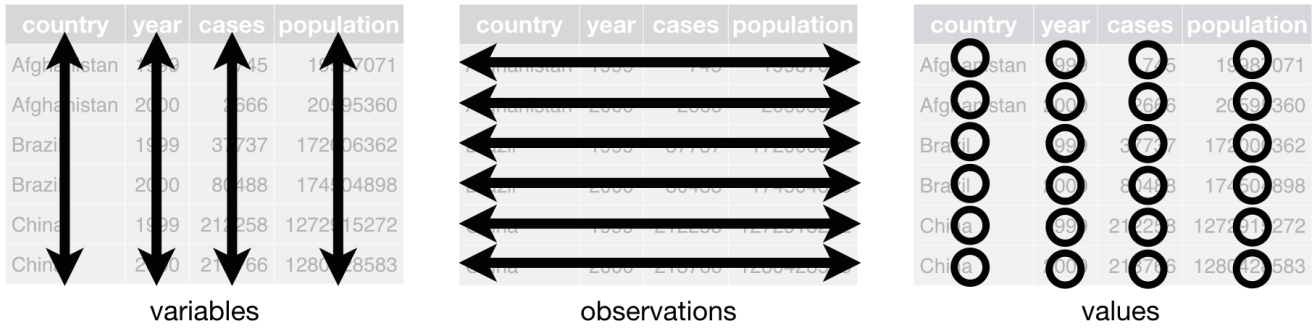
http://tinyurl.com/treesdata

- One species
- 10 plots
- 1000 trees
- Number of trees per plot ranging from 4 to 392

```
trees <- read.csv("data/trees.csv")
summary(trees[, 1:3])
```

```
      plot            dbh              height     
 Min.   : 1.0   Min.   : 5.06   Min.   :13.40  
 1st Qu.: 1.0   1st Qu.:17.69   1st Qu.:29.68  
 Median : 2.0   Median :28.62   Median :36.55  
 Mean   : 2.7   Mean   :27.88   Mean   :36.51  
 3rd Qu.: 4.0   3rd Qu.:38.97   3rd Qu.:43.33  
 Max.   :10.0   Max.   :49.92   Max.   :59.30  
```

Data must be a tidy data frame



tidyr::gather(table4, key = "year", value = "cases", "1999", "2000")

country	year	cases
Afghanistan	1999	745
Afghanistan	2000	2666
Brazil	1999	37737
Brazil	2000	80488
China	1999	212258
China	2000	213766

country	1999	2000
Afghanistan	745	2666
Brazil	37737	80488
China	212258	213766

table4

http://r4ds.had.co.nz/tidy-data.html

Calling ggplot

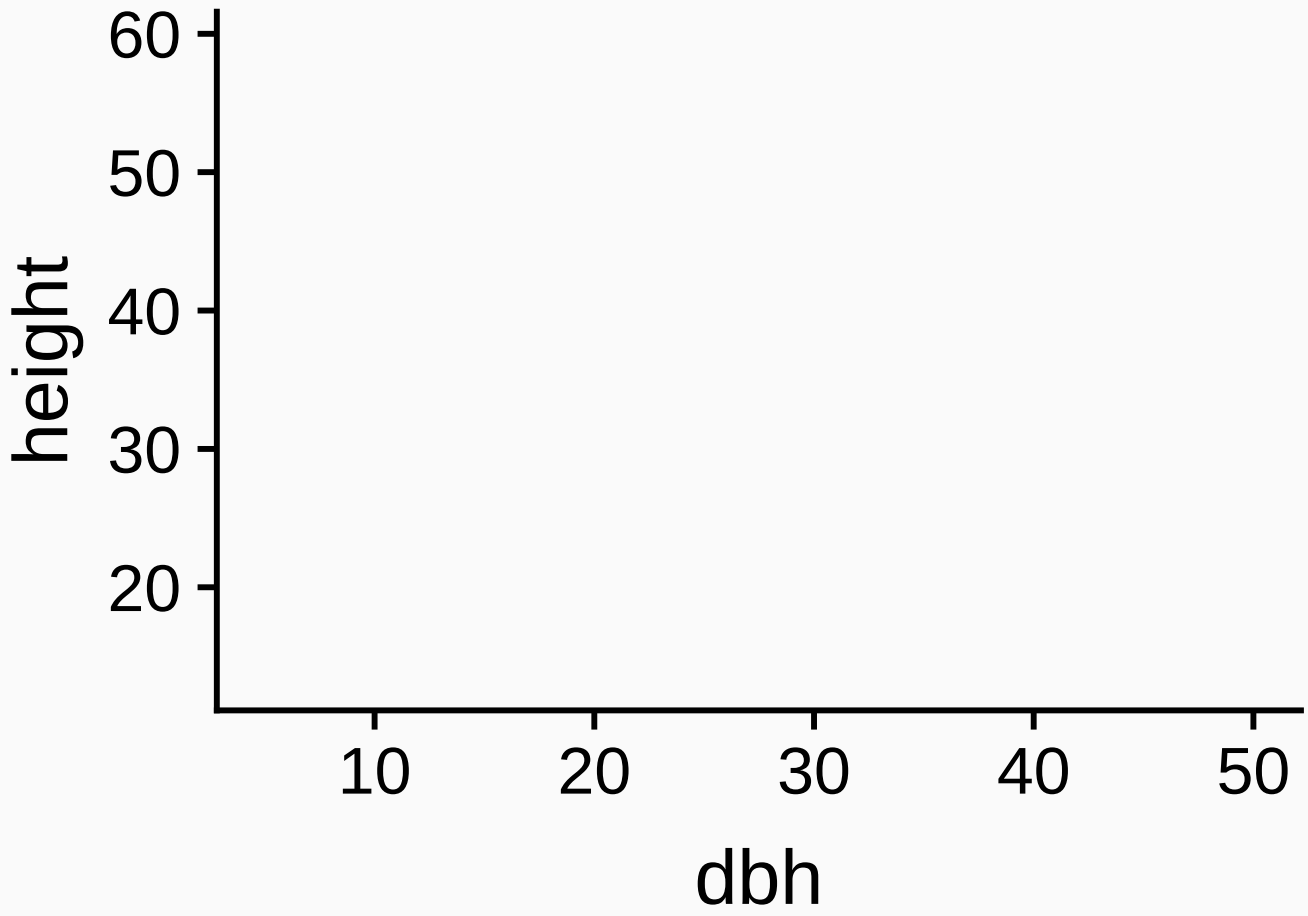
```
library(ggplot2)
ggplot(trees)
```

```
ggplot(trees)
```

First argument is a tidy data frame

What variables as axes?

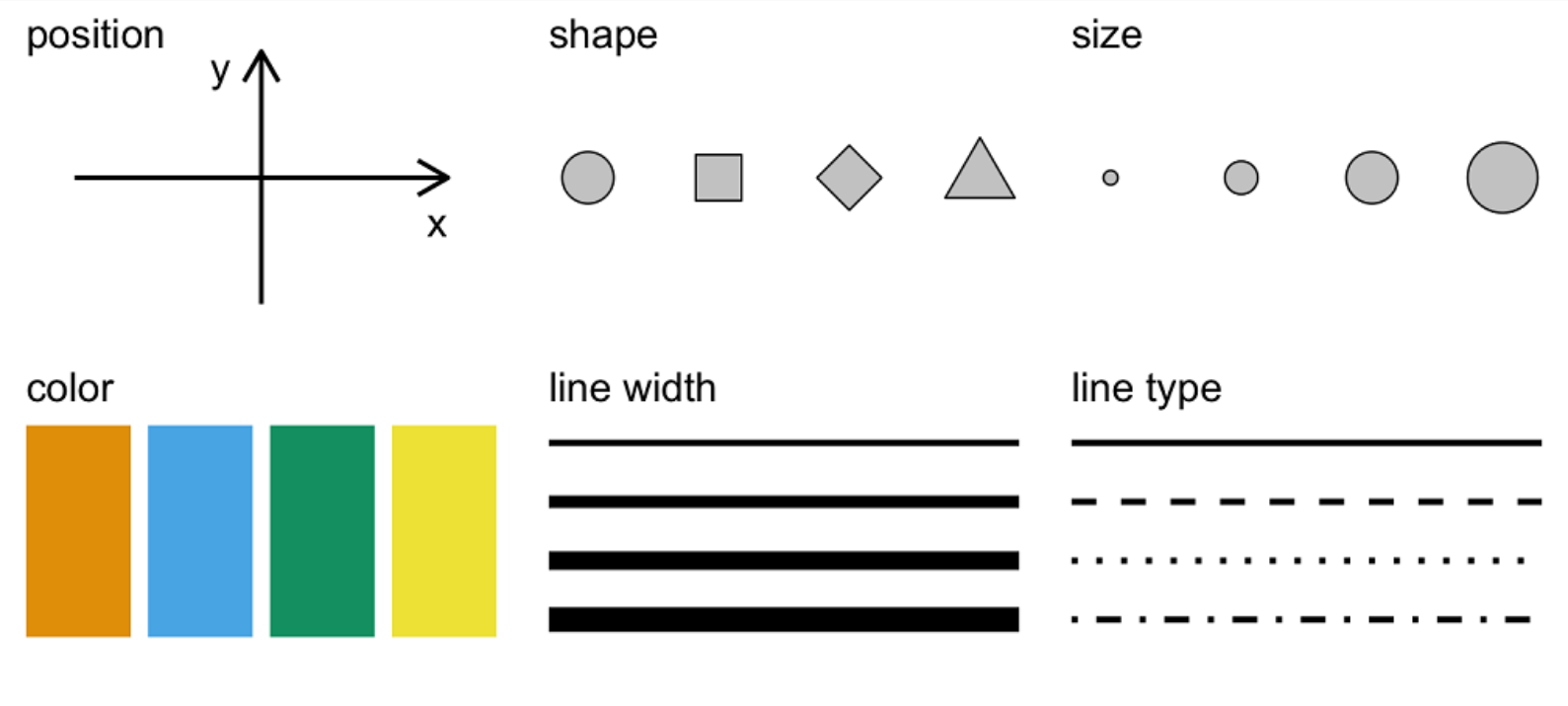
```
ggplot(trees) +
  aes(x = dbh, y = height)
```



Note syntax: + followed by new line

```
ggplot(trees) +
  aes(x = dbh, y = height)
```

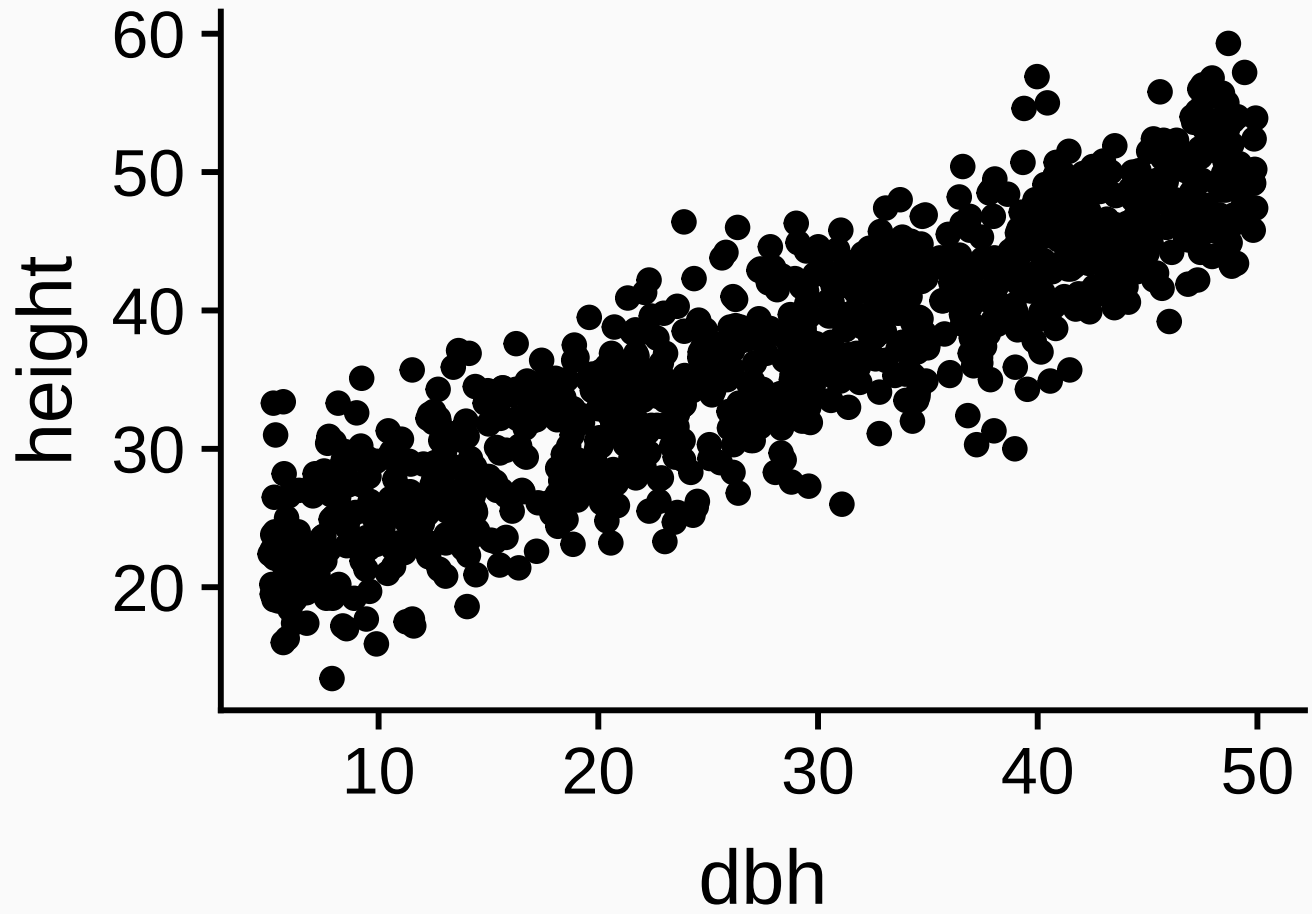
Aesthetics (aes) map data variables (*dbh*, *height*) to graphic elements (*axes*)



http://serialmentor.com/dataviz/aesthetic-mapping.html

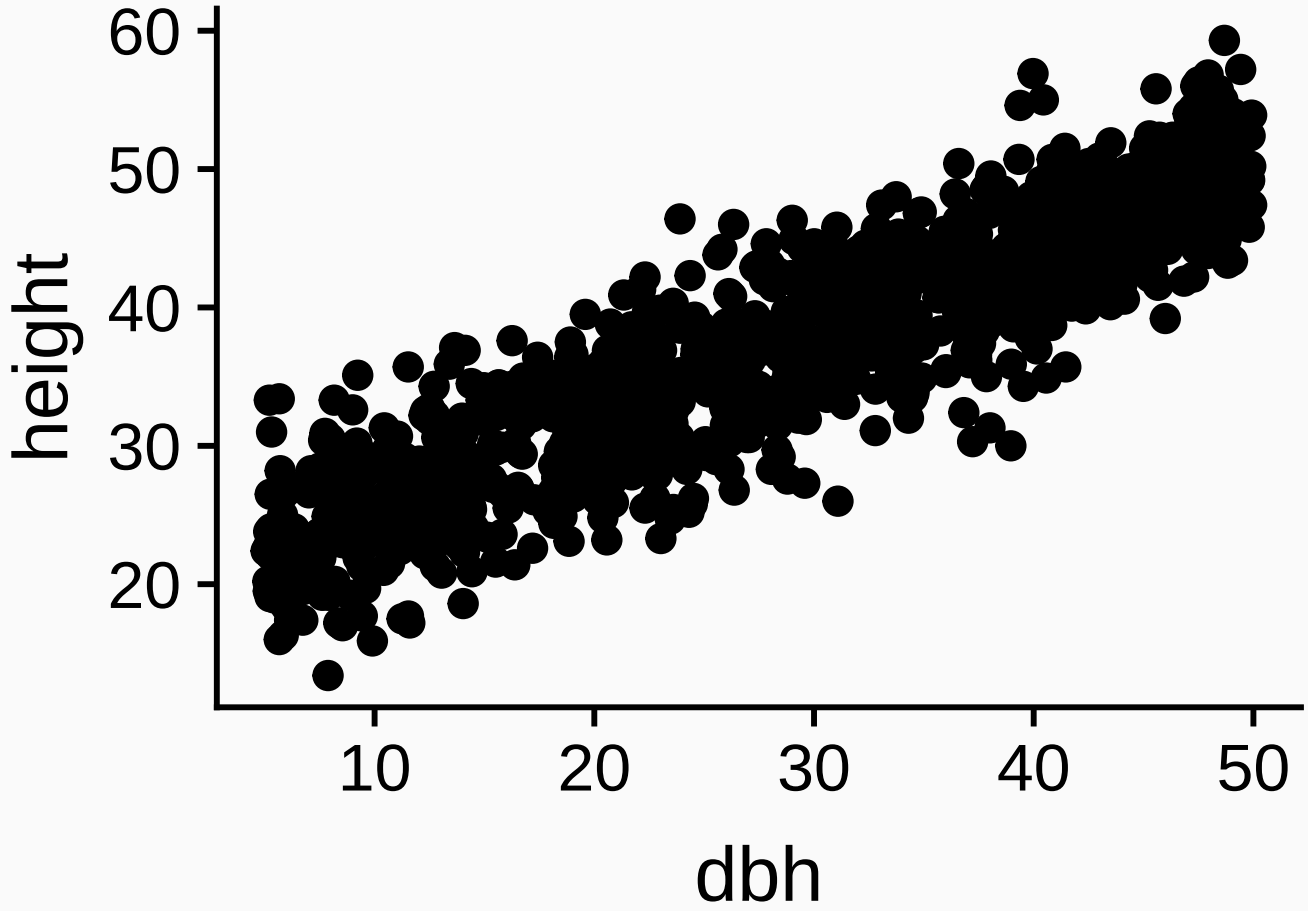
Adding layers (geoms)

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point()
```



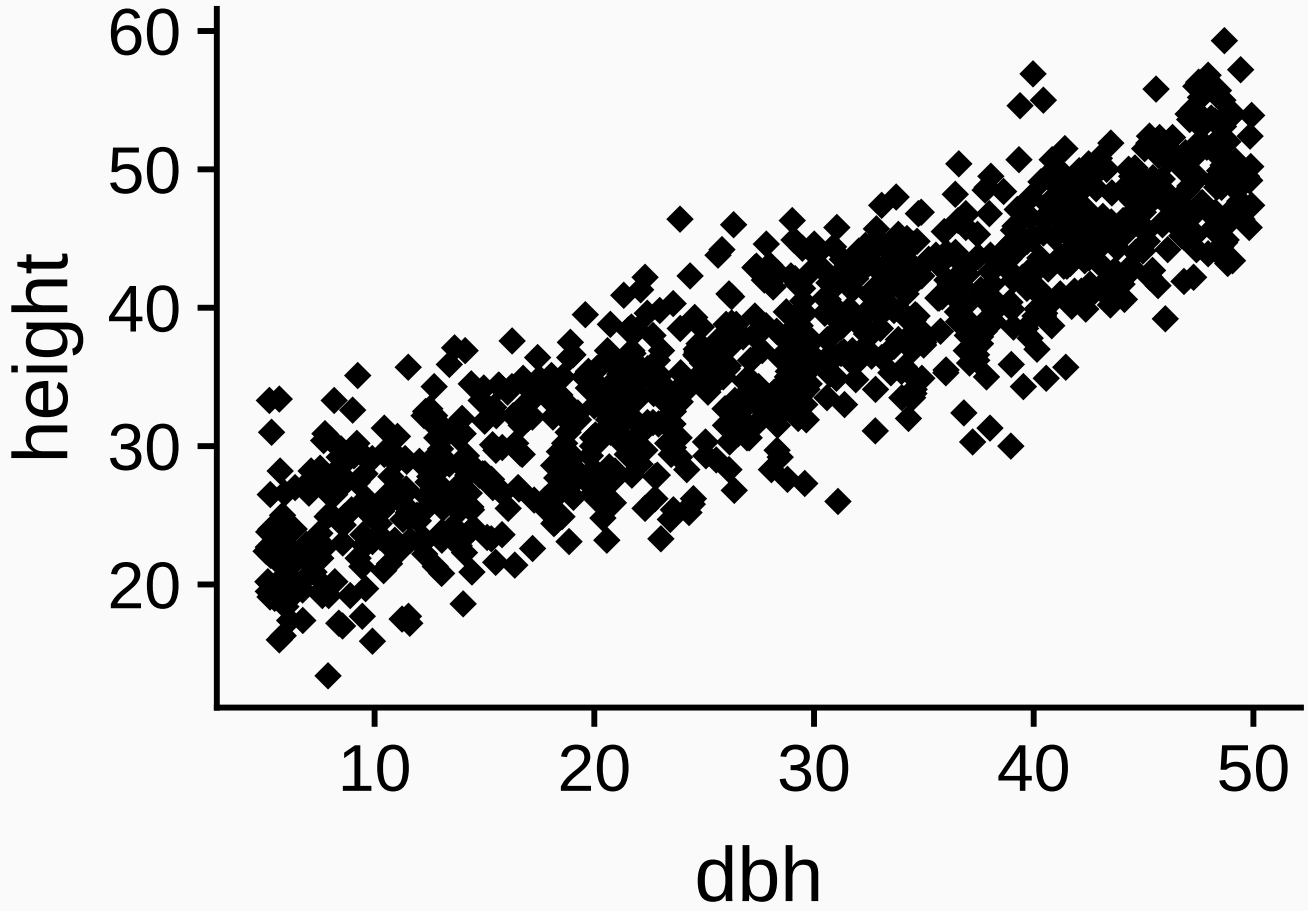
Changing point size

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point(size = 2)
```



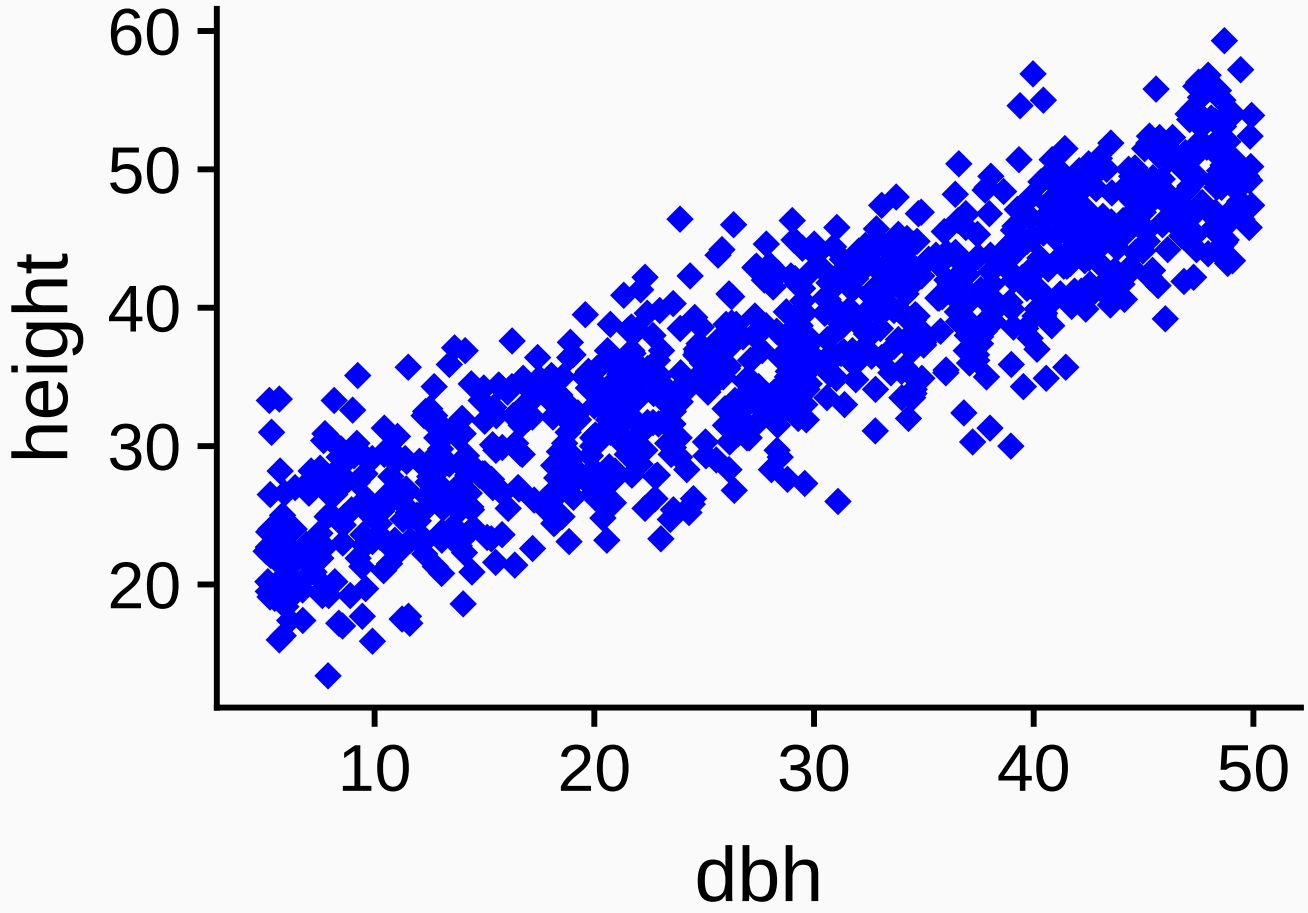
Changing point size and type

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point(size = 2, shape = 18)
```



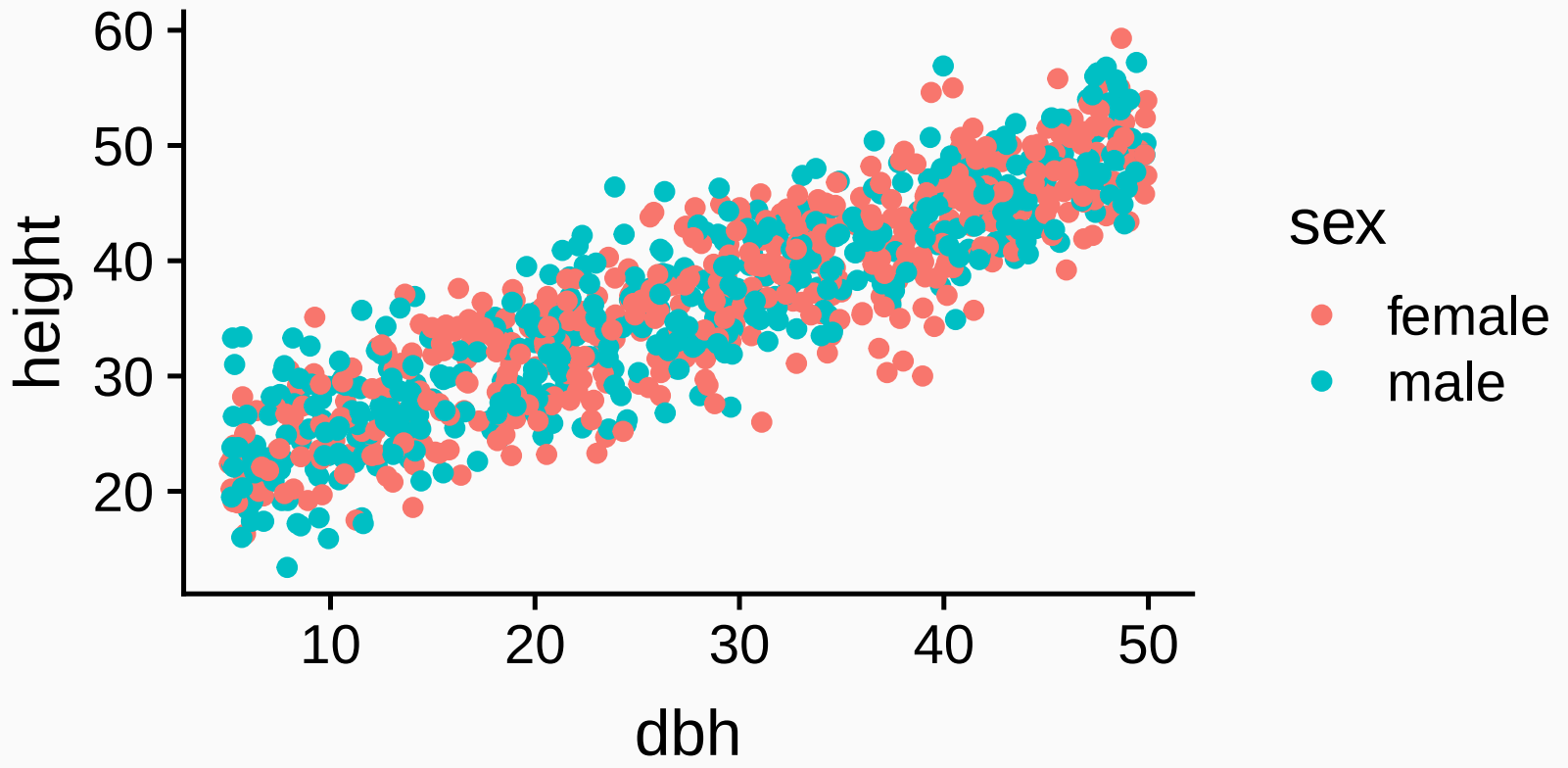
Changing point size and type

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point(size = 2, shape = 18, colour = 'blue')
```



Map geom aesthetics (e.g. colour) to variable

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point(aes(colour = sex))
```



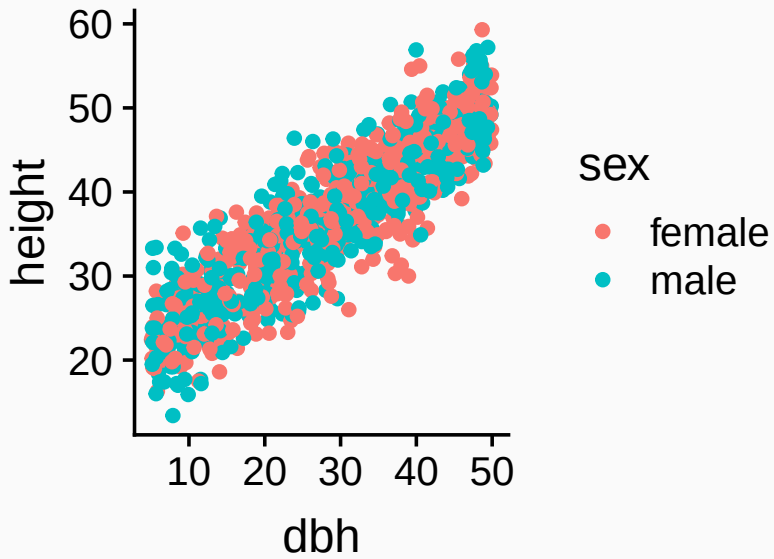
Note difference between

```
geom_point(colour = "blue")
# colour is given a concrete value ('blue')
```

```
geom_point(aes(colour = sex))
# colour maps a *variable* (using `aes`)
```

This works:

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point(aes(colour = sex))
```



This doesn't work:

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point(colour = sex)
```

Error in layer(data = data, mapping = mapping, stat = stat, geom = GeomPoint, : object 'sex' not found

.

.

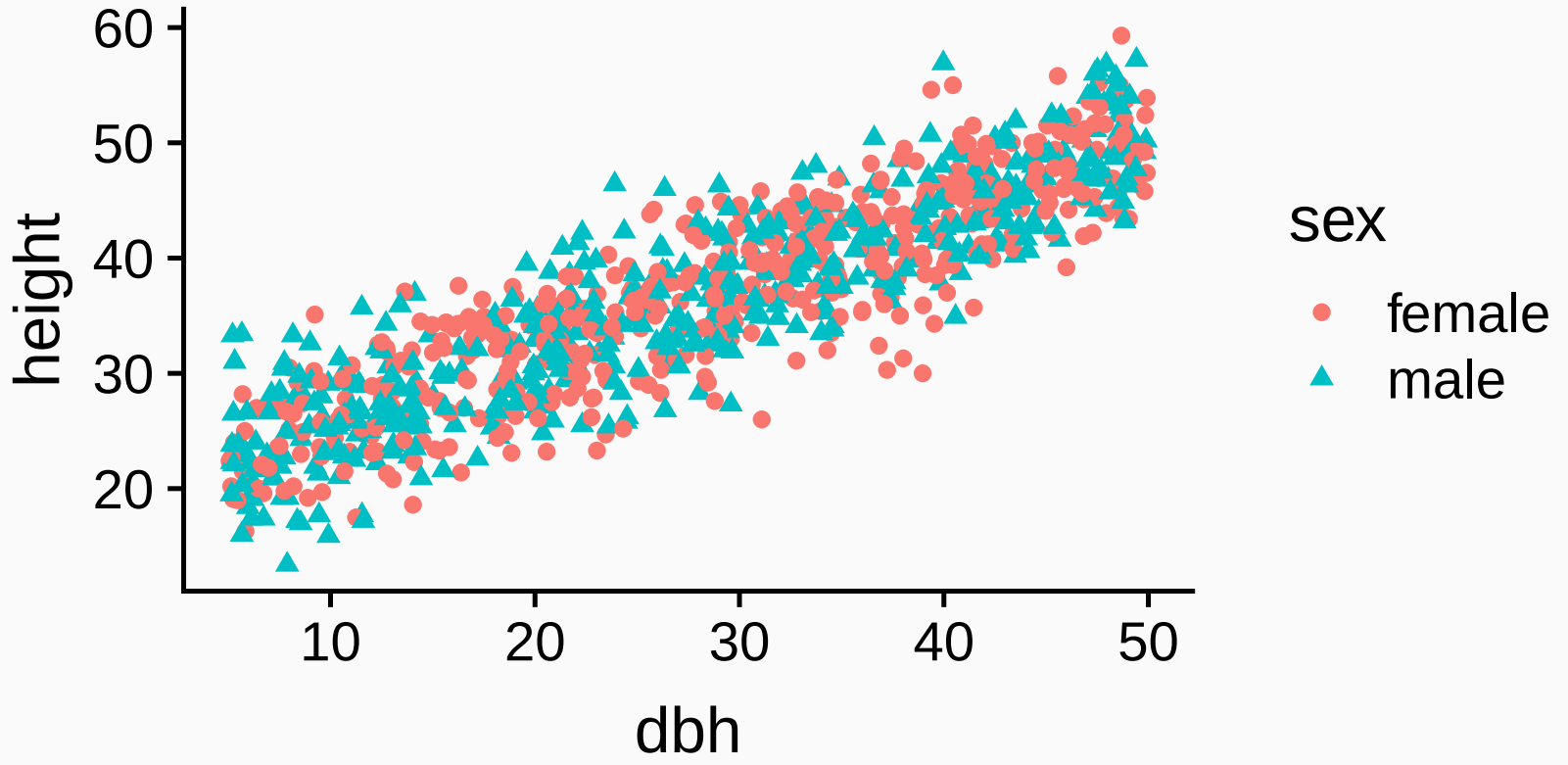
'sex' is a variable in dataframe

.

Must use `aes`

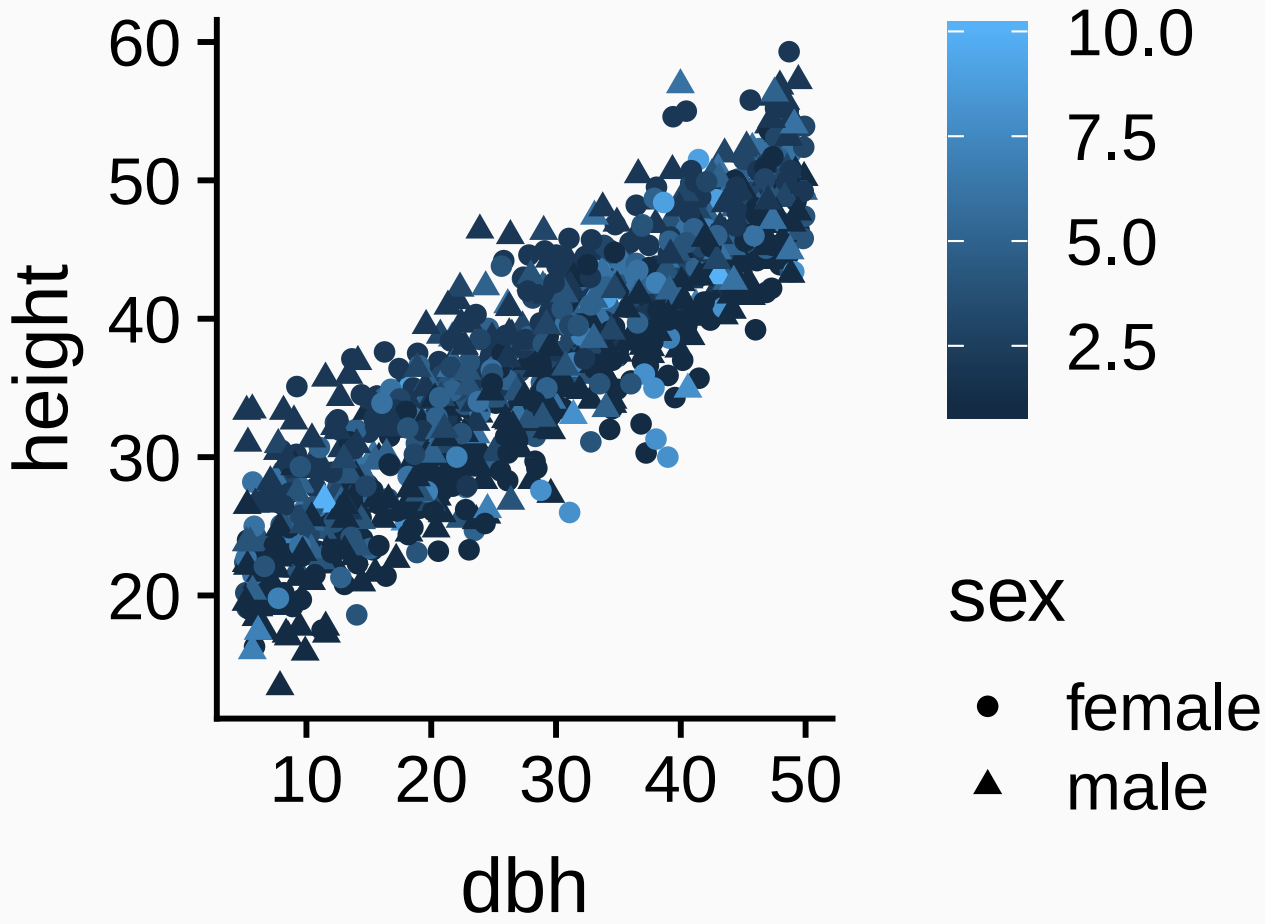
Map geom aesthetics (colour, shape) to variable

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point(aes(colour = sex, shape = sex))
```



Map geom aesthetics (colour, shape) to variable

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point(aes(colour = plot, shape = sex))
```

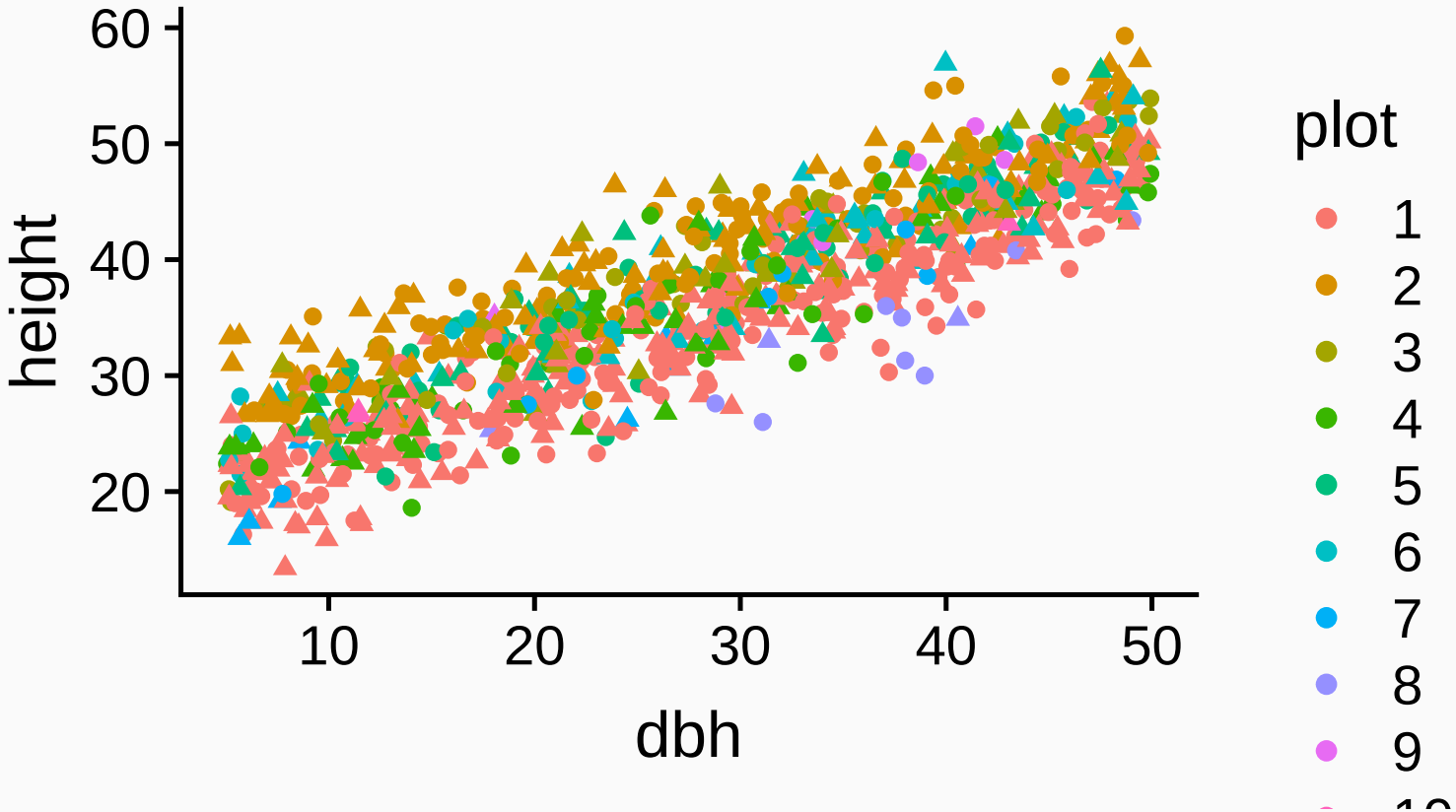


Ensuring ‘plot’ is a factor, not numeric

```
trees$plot <- as.factor(trees$plot)
```

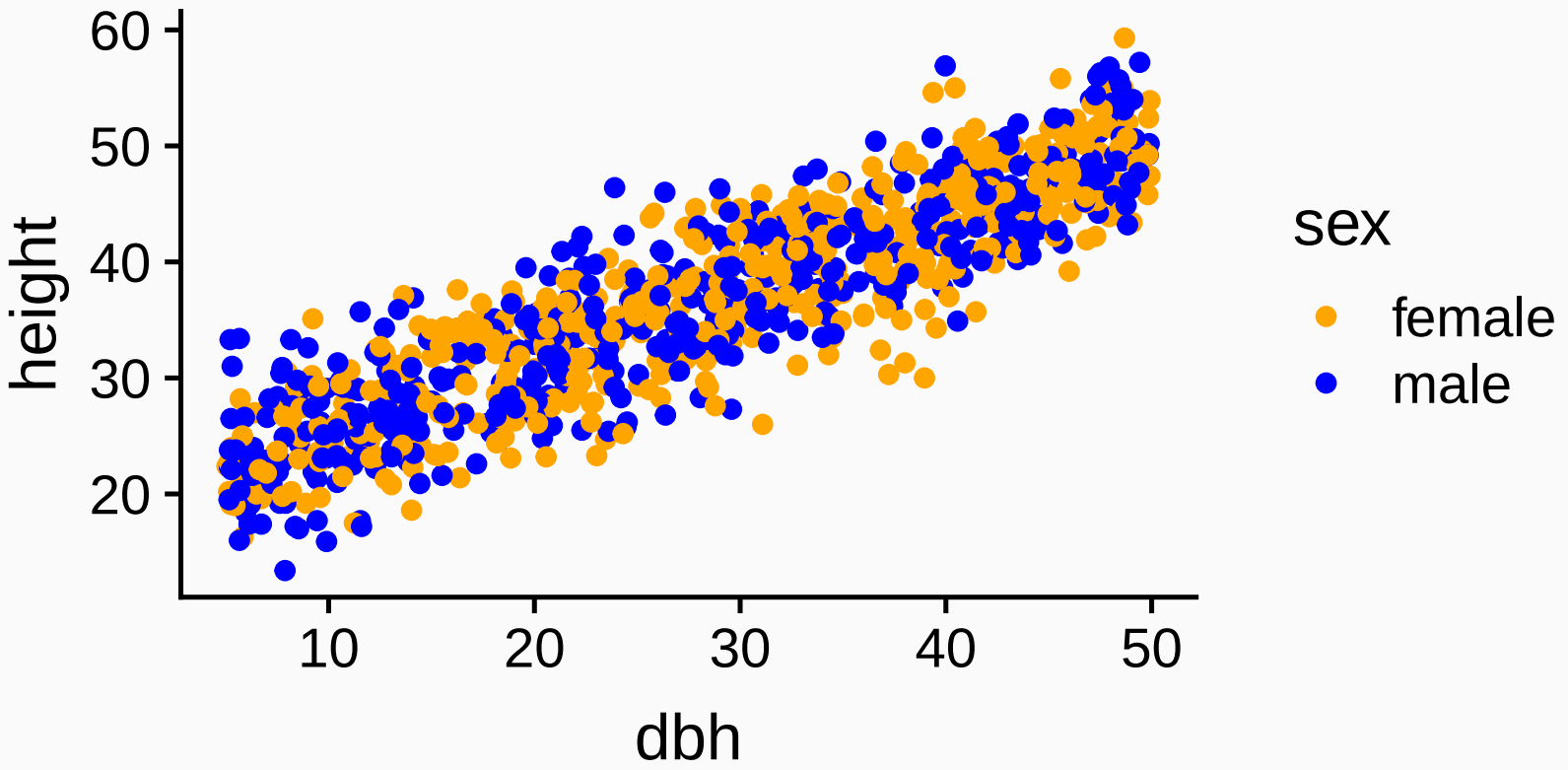
Map geom aesthetics (colour, shape) to variable

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point(aes(colour = plot, shape = sex))
```



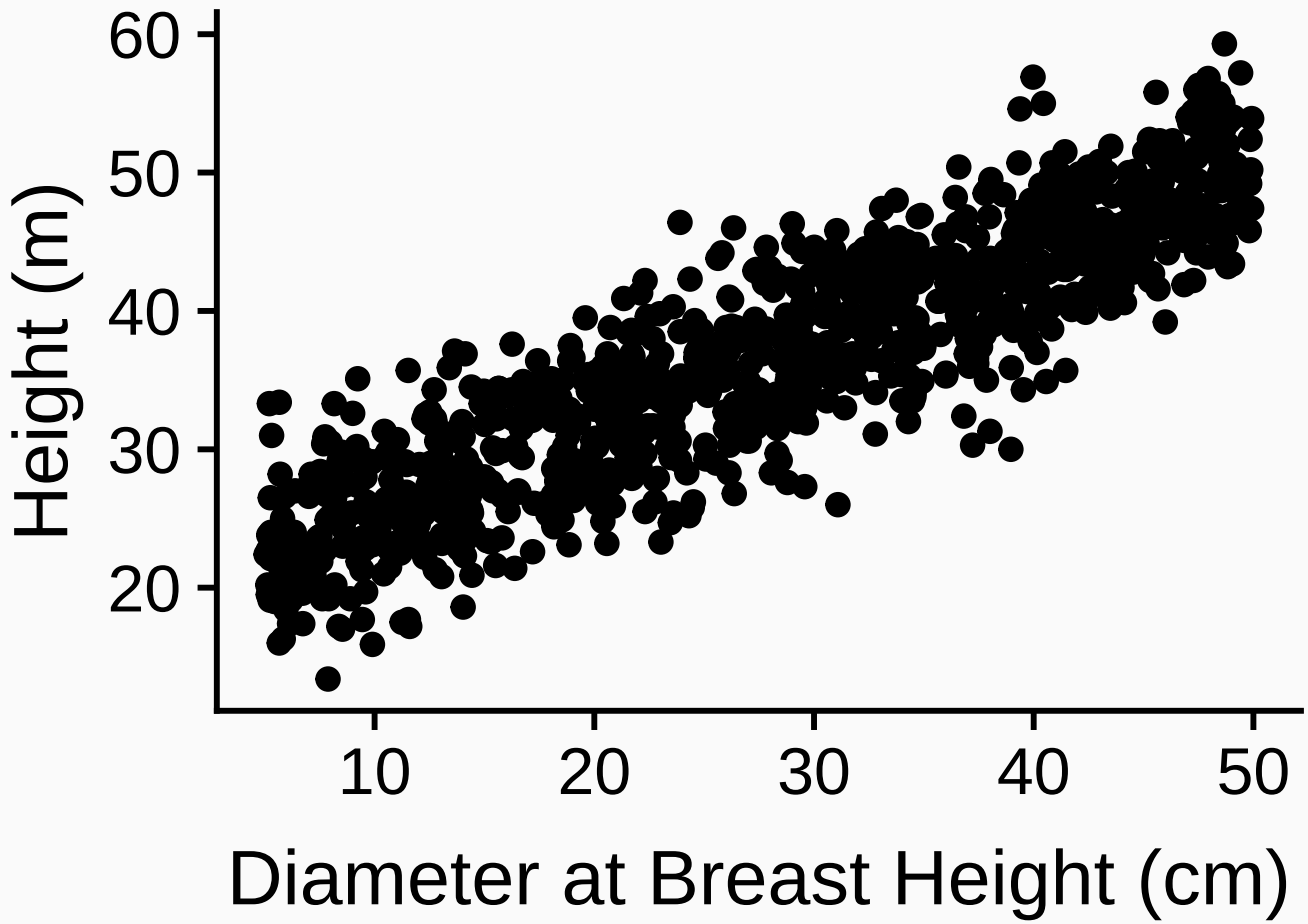
Change colour scale

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point(aes(colour = sex)) +
  scale_colour_manual(values = c("orange", "blue"))
```



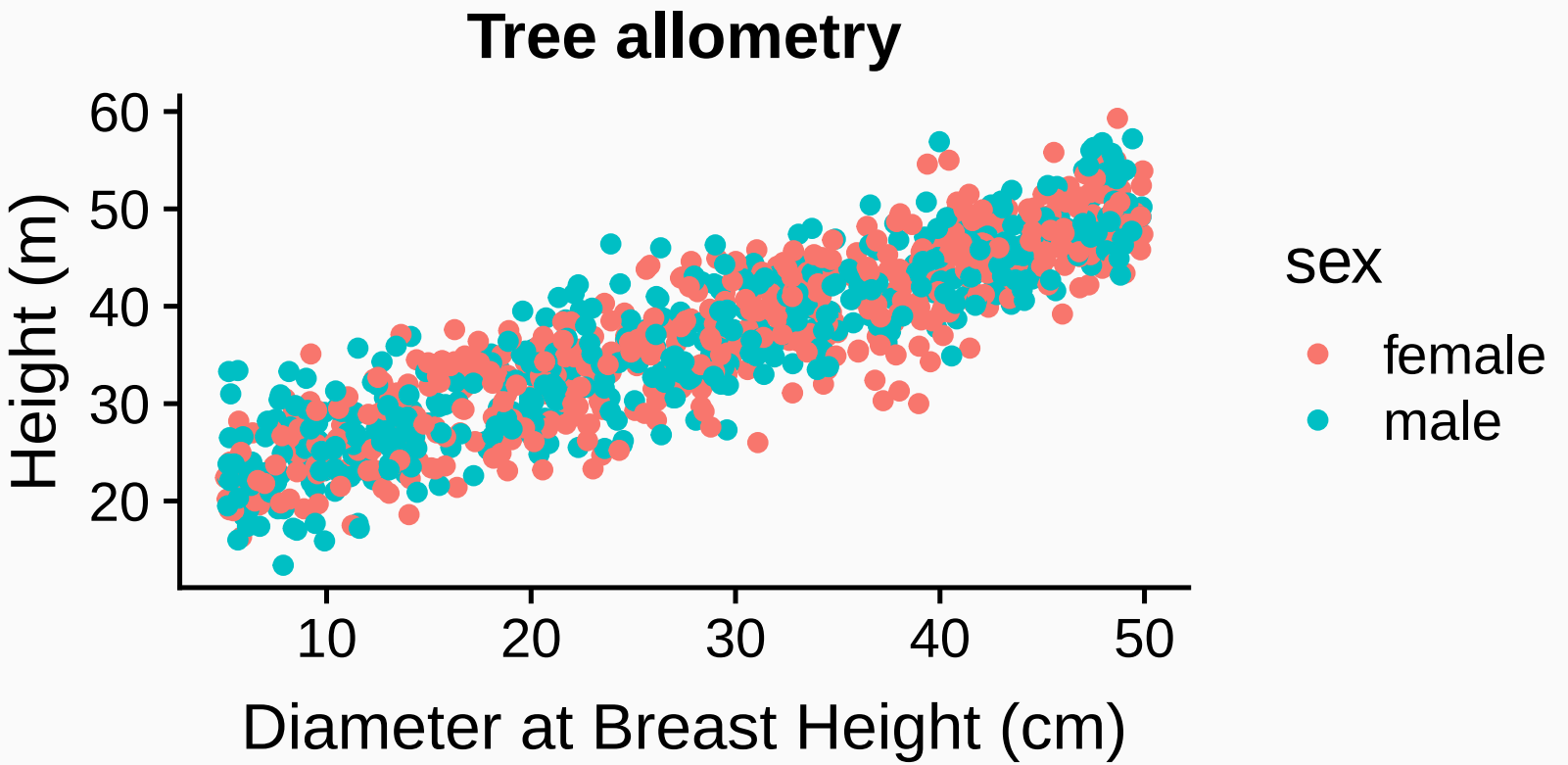
Change axis labels: xlab & ylab

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point() +
  labs(x = "Diameter at Breast Height (cm)", y = "Height (m)")
```



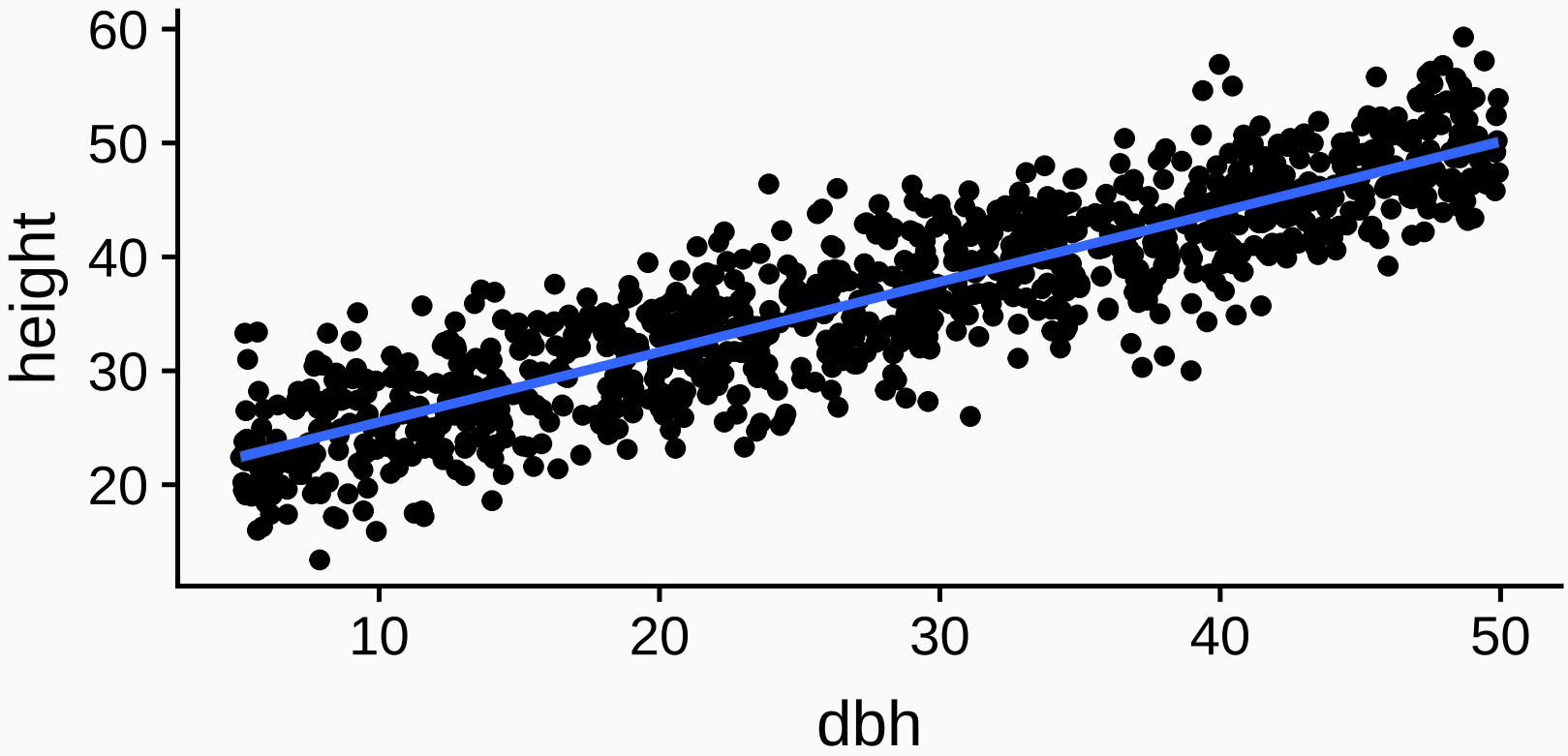
Set title

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point(aes(colour = sex)) +
  labs(x = "Diameter at Breast Height (cm)", y = "Height (m)") +
  labs(title = "Tree allometry")
```



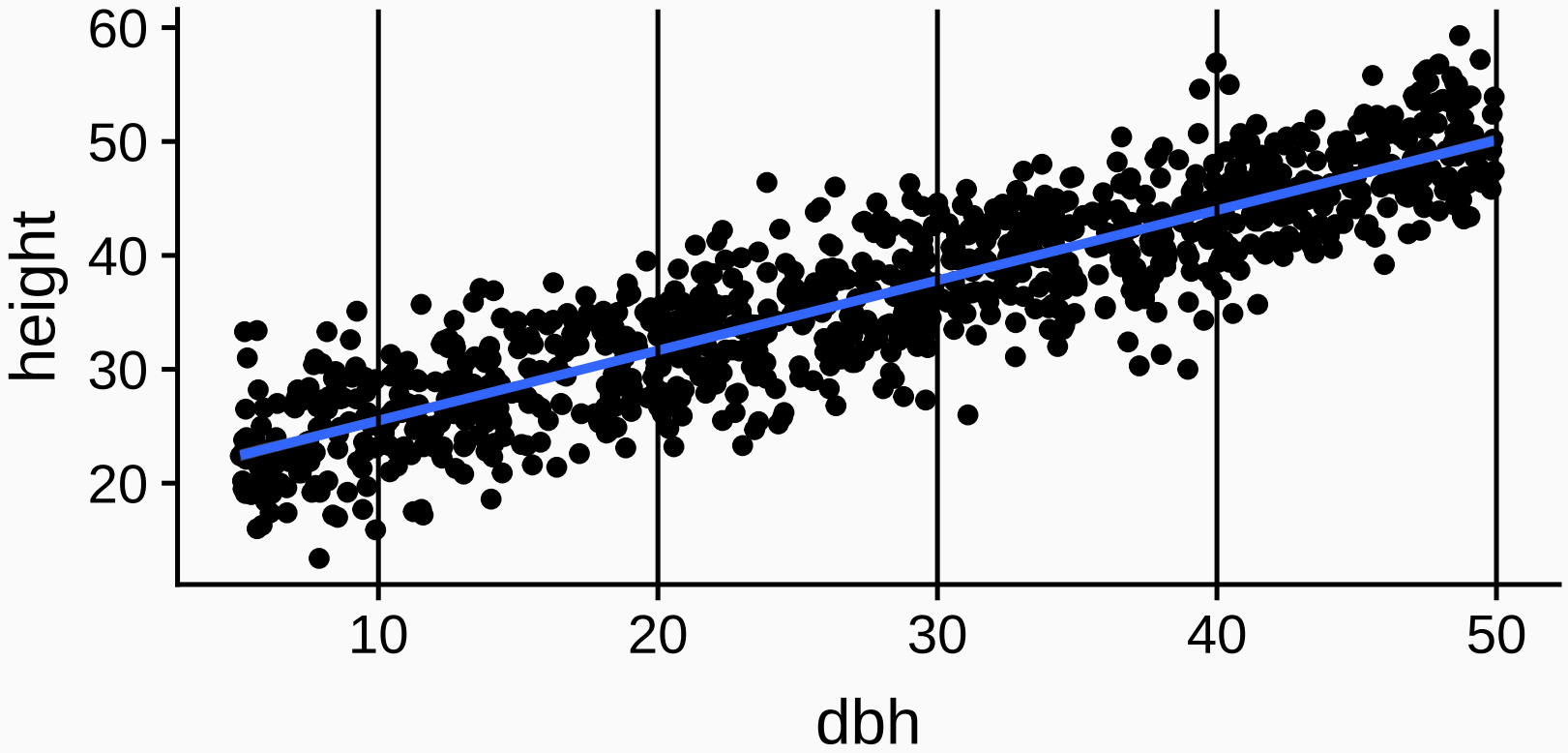
Adding more layers

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point() +
  geom_smooth(method = "lm")
```



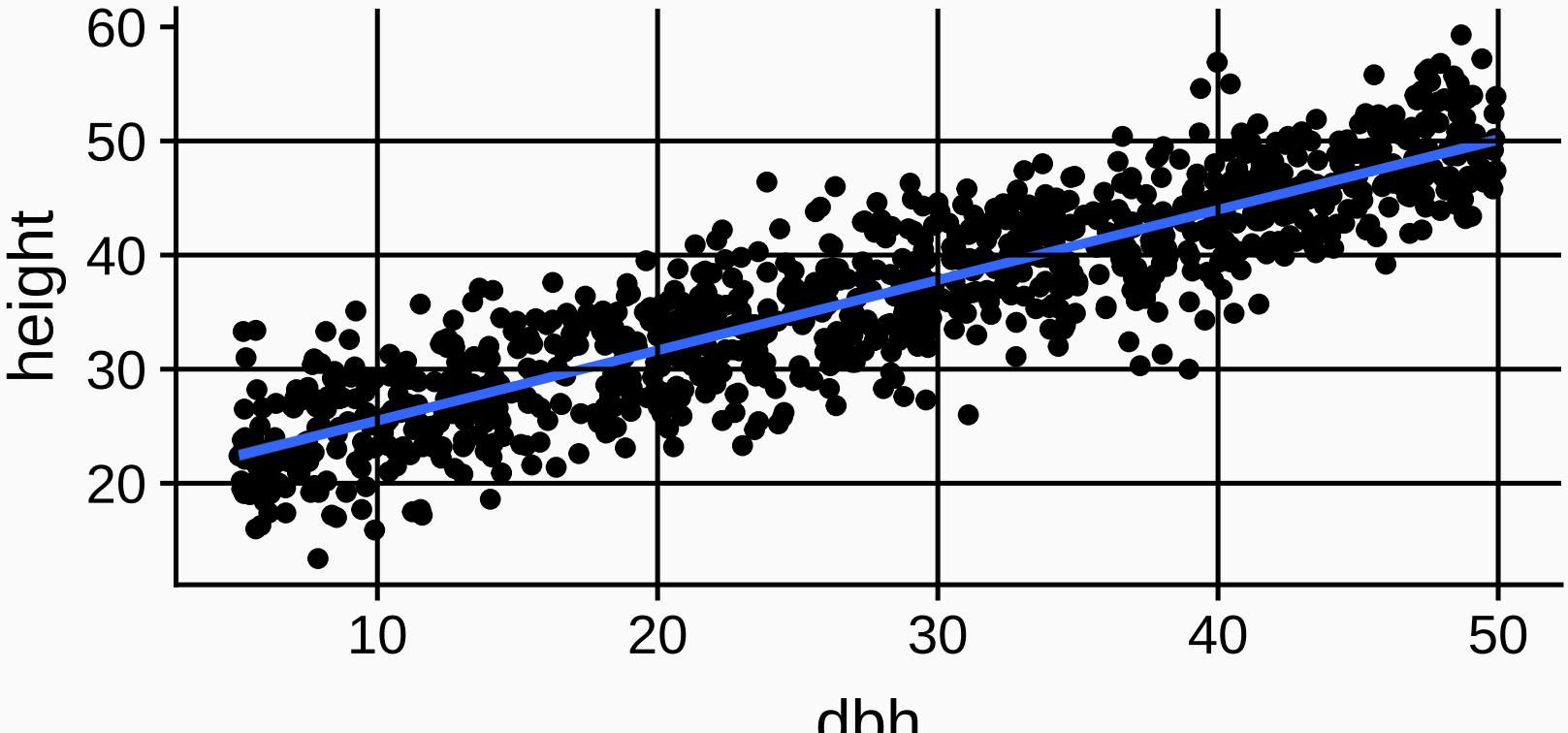
Adding more layers

```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point() +
  geom_smooth(method = "lm") +
  geom_vline(xintercept = c(10, 20, 30, 40, 50))
```



Adding more layers

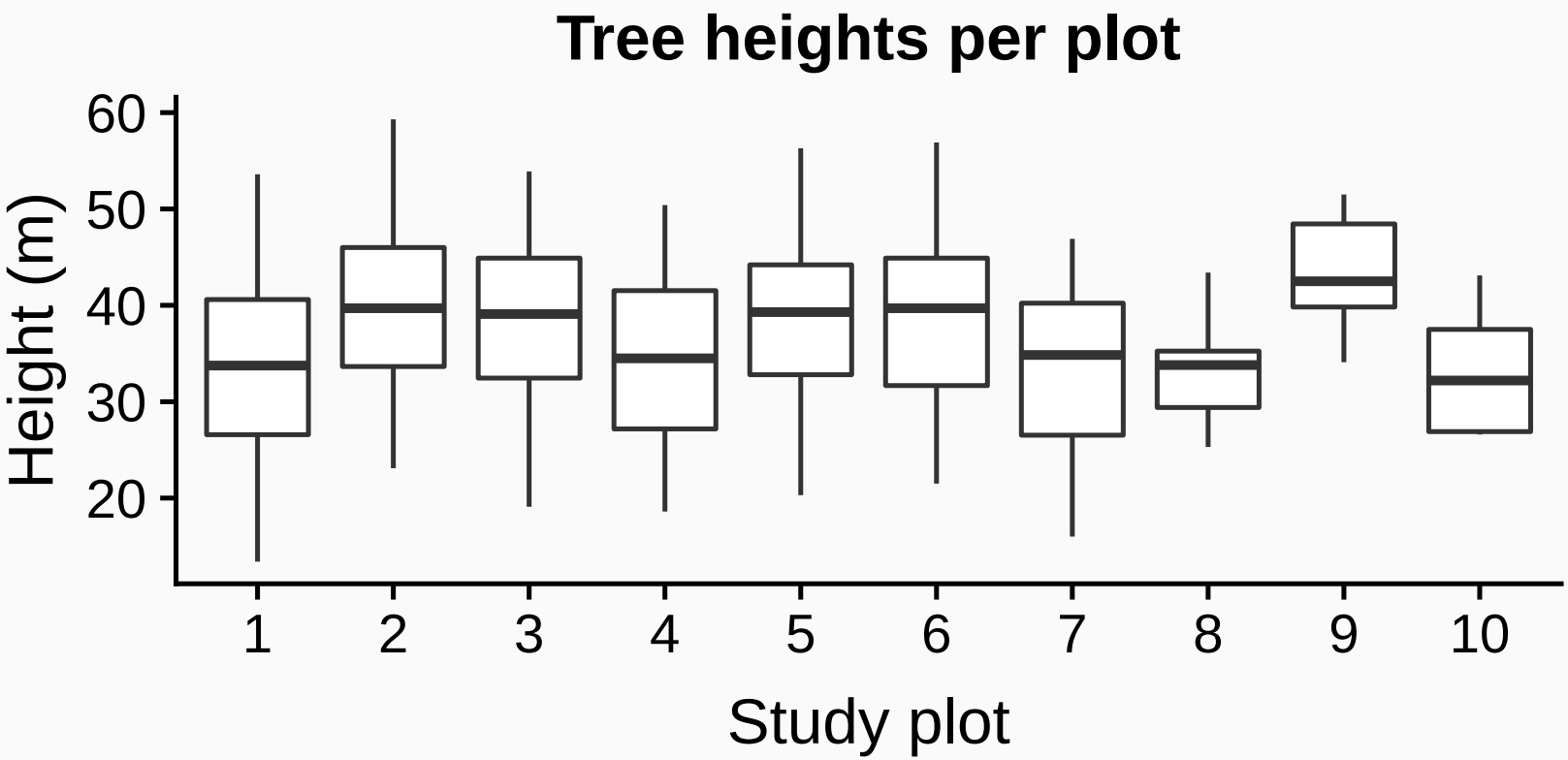
```
ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point() +
  geom_smooth(method = "lm") +
  geom_vline(xintercept = c(10, 20, 30, 40, 50)) +
  geom_hline(yintercept = c(20, 30, 40, 50))
```



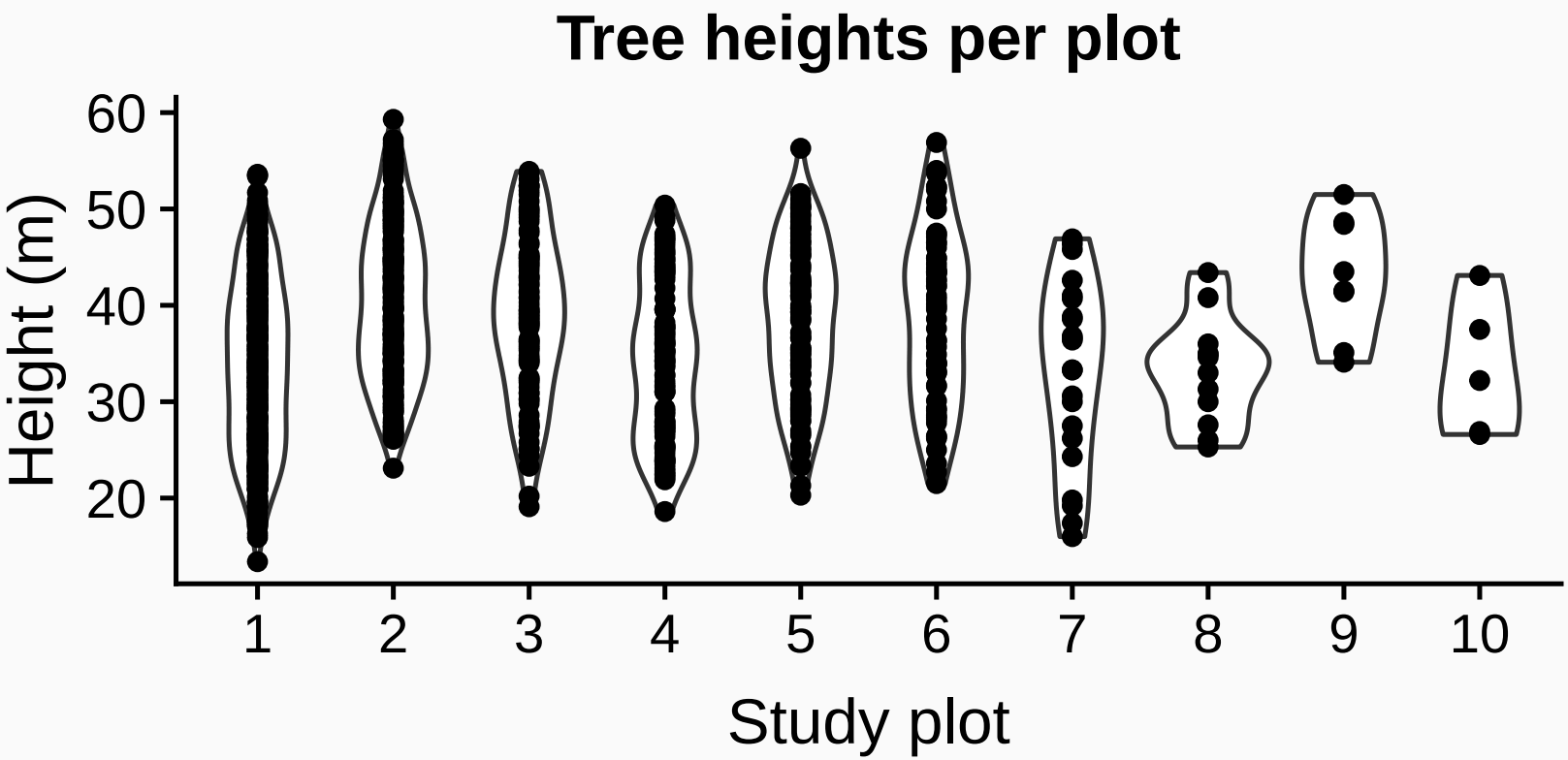
Summary

```
ggplot(trees) +                    # Name of (tidy) data frame
  aes(x = dbh, y = height) +       # Aesthetics (variables to map in axes)
  geom_point()                     # Geoms: geometric objects
```

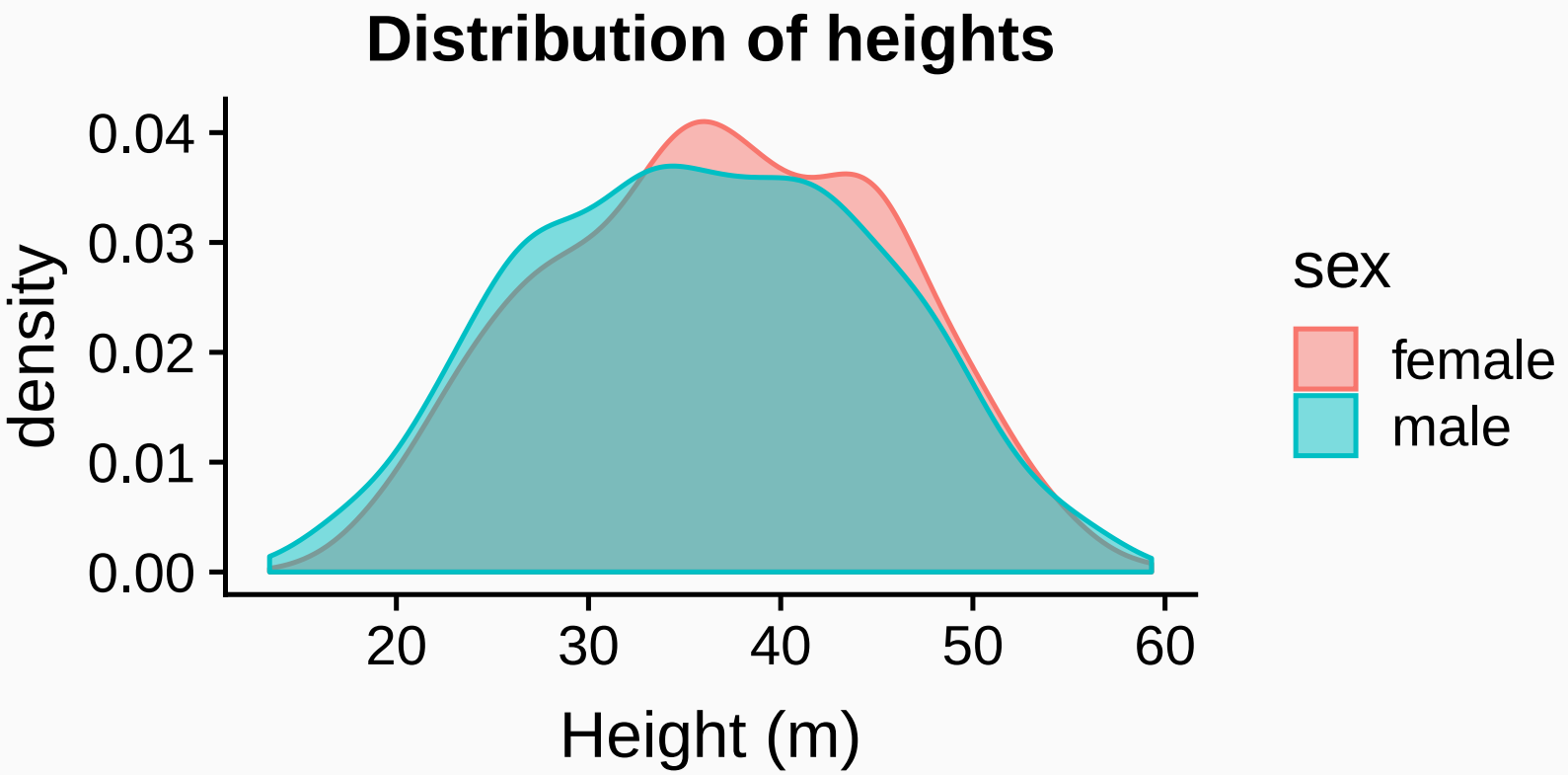
Exercise: Make a plot like this one



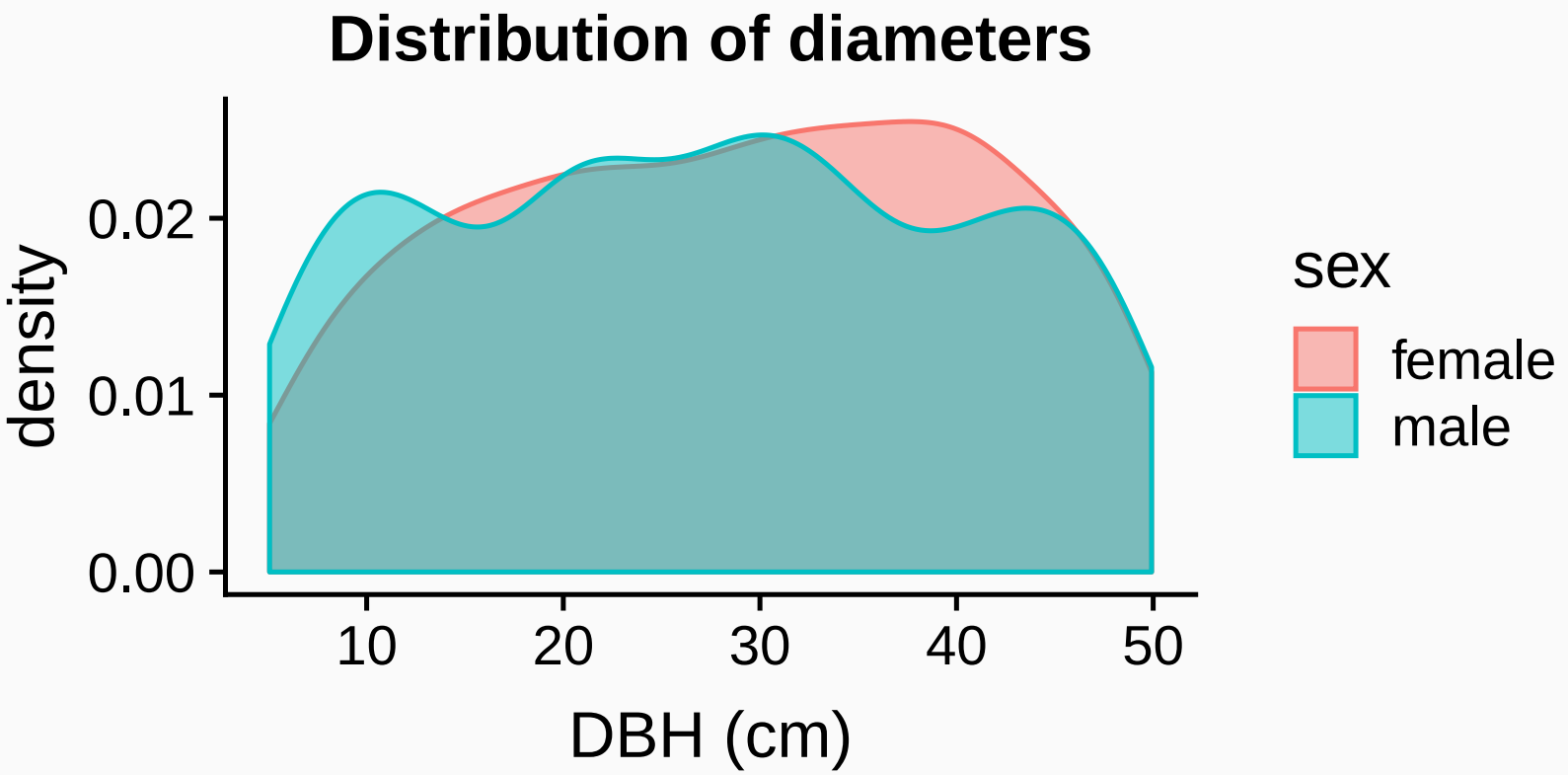
Exercise: Make a plot like this one



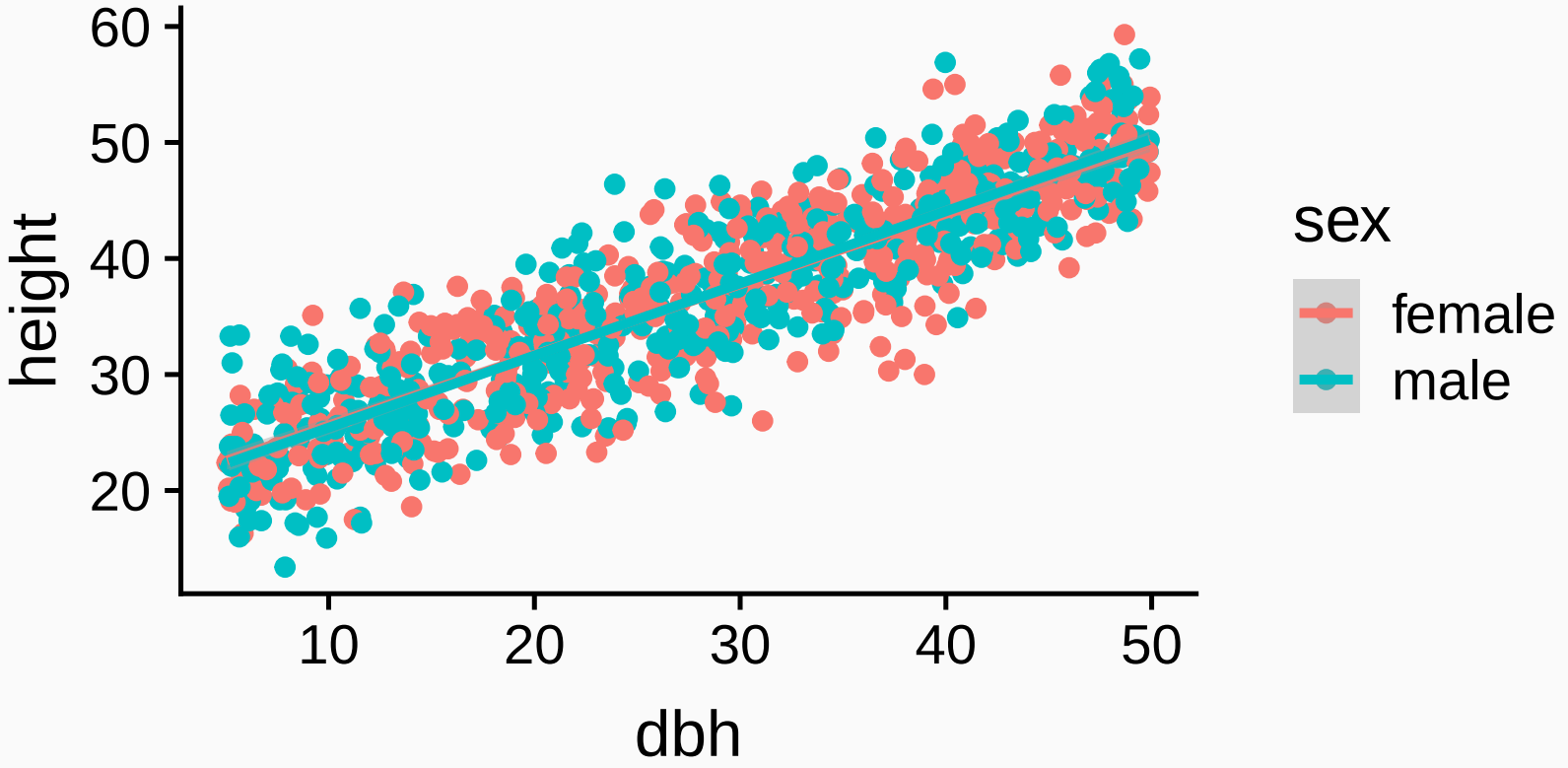
Exercise: Make a plot like this one



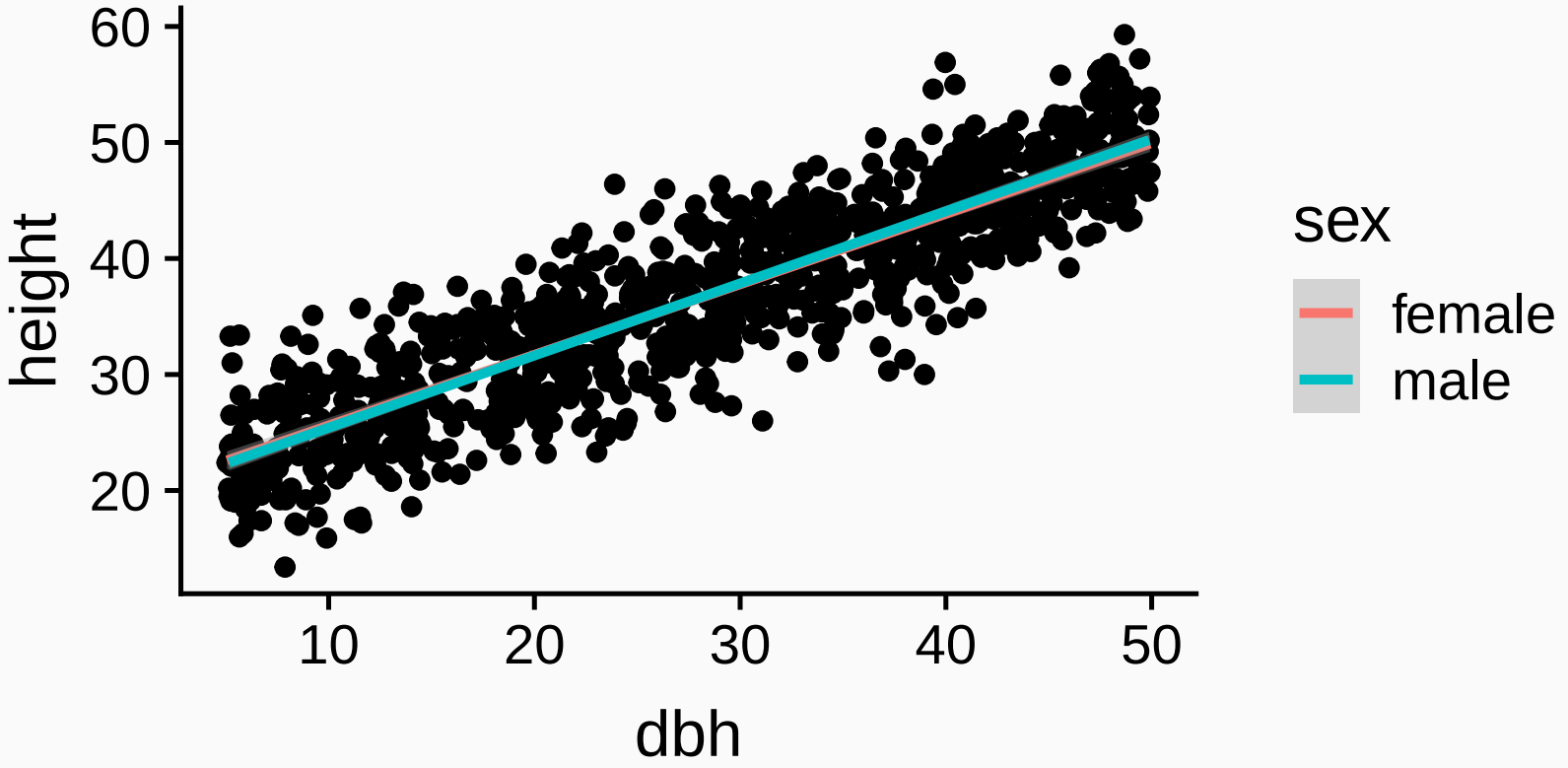
Exercise: Make a plot like this one



Exercise: Make a plot like this one



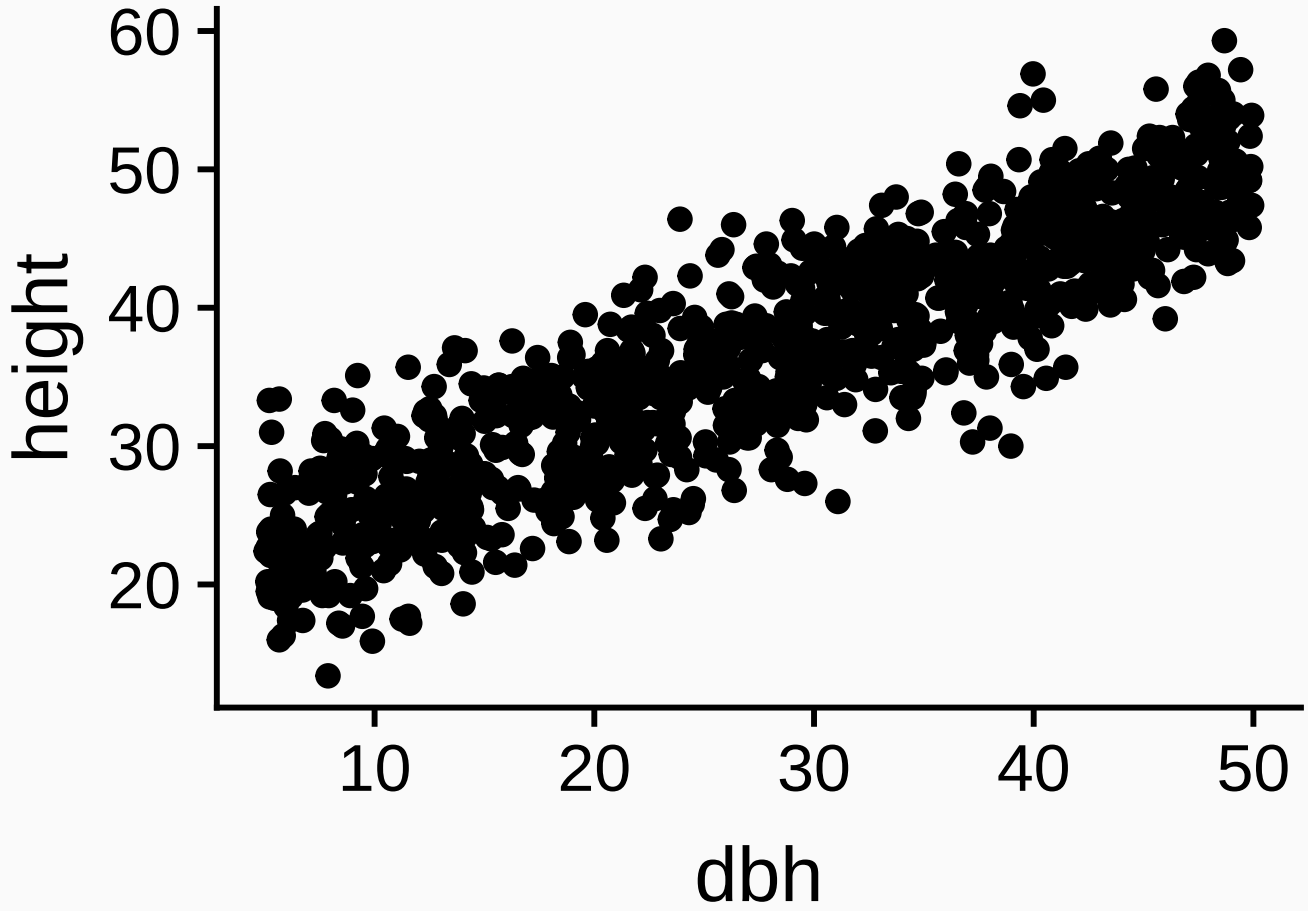
Exercise: Make a plot like this one



ggplot2 figures can be assigned to R objects

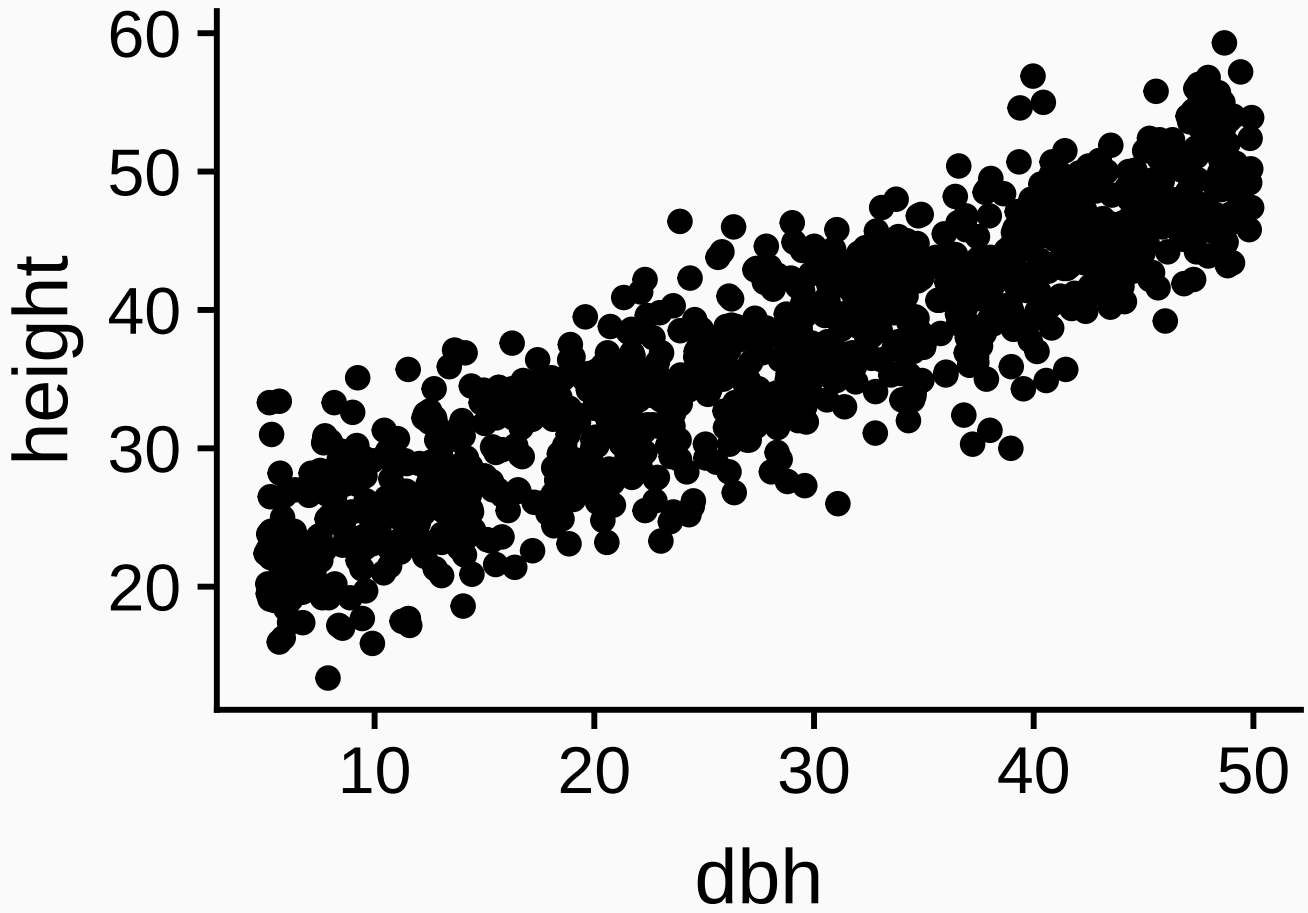
Assigning ggplot objects

```
myplot <- ggplot(trees) +
  aes(x = dbh, y = height)
myplot + geom_point()
```



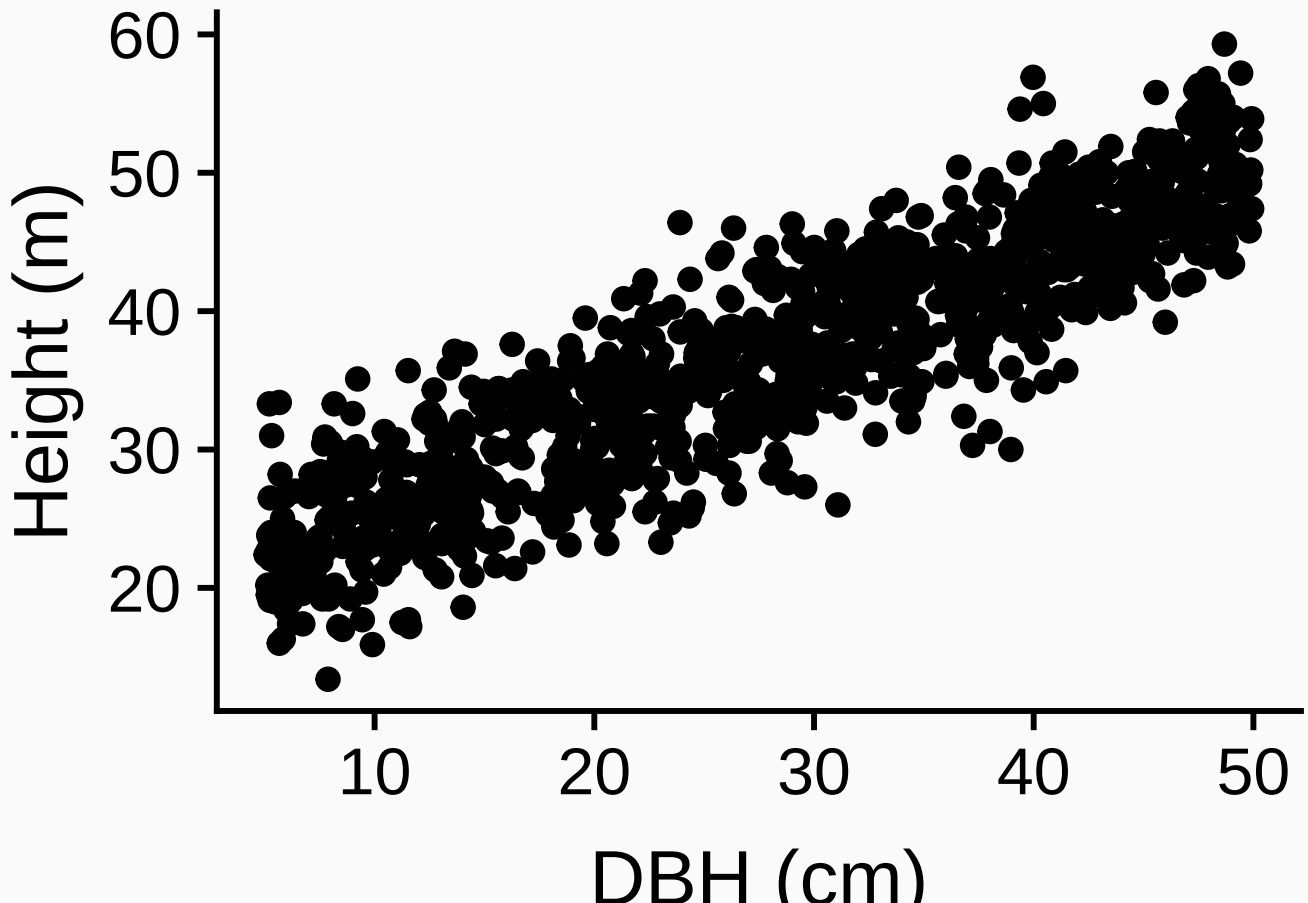
Assigning ggplot objects

```
myplot <- ggplot(trees) +
  aes(x = dbh, y = height)
myplot <- myplot + geom_point()
myplot
```



Assigning ggplot objects

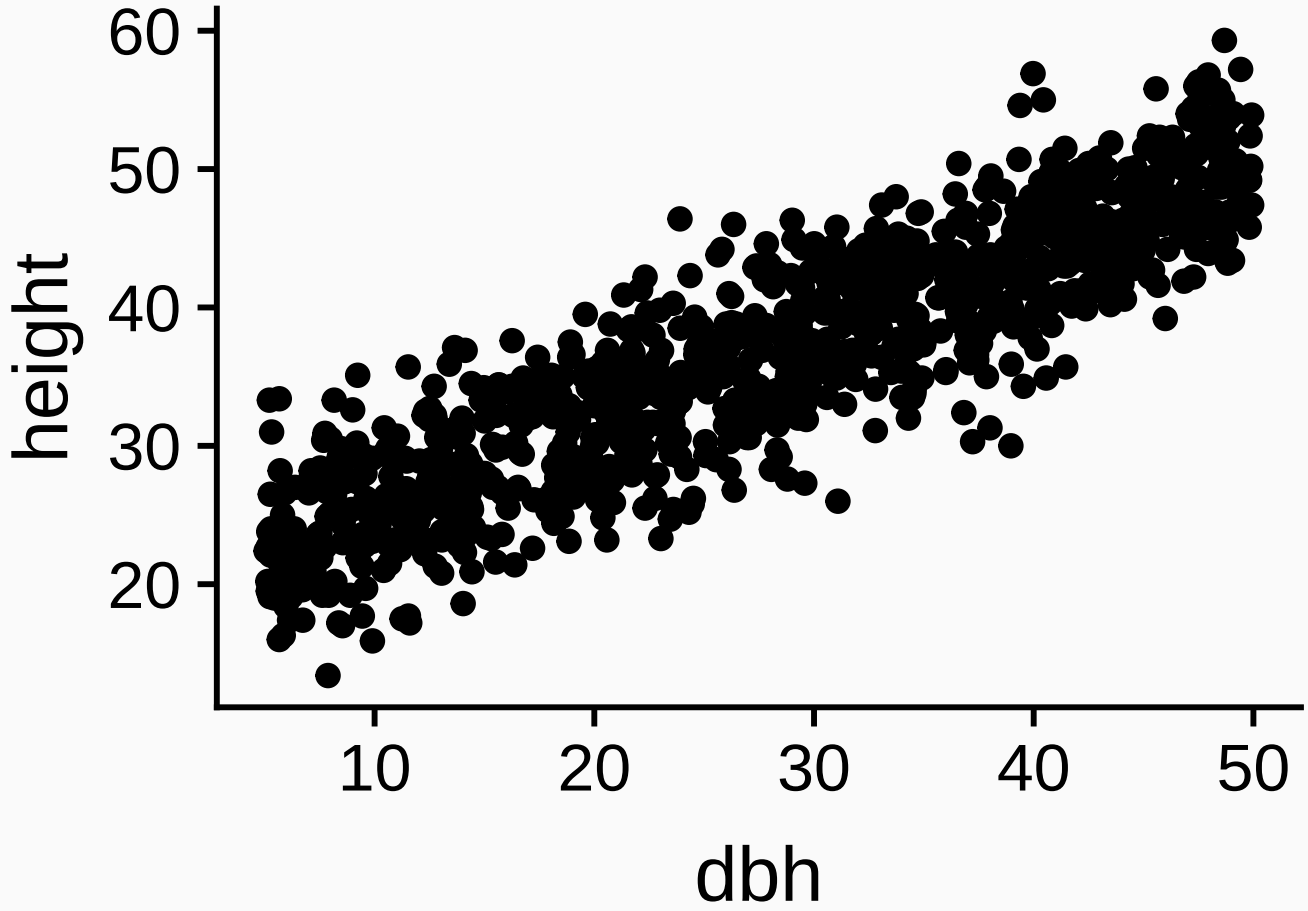
```
baseplot <- ggplot(trees) +
  aes(x = dbh, y = height)
scatterplot <- baseplot + geom_point()
labelled <- scatterplot + labs(x = "DBH (cm)", y = "Height (m)")
labelled
```



Themes: changing plot appearance

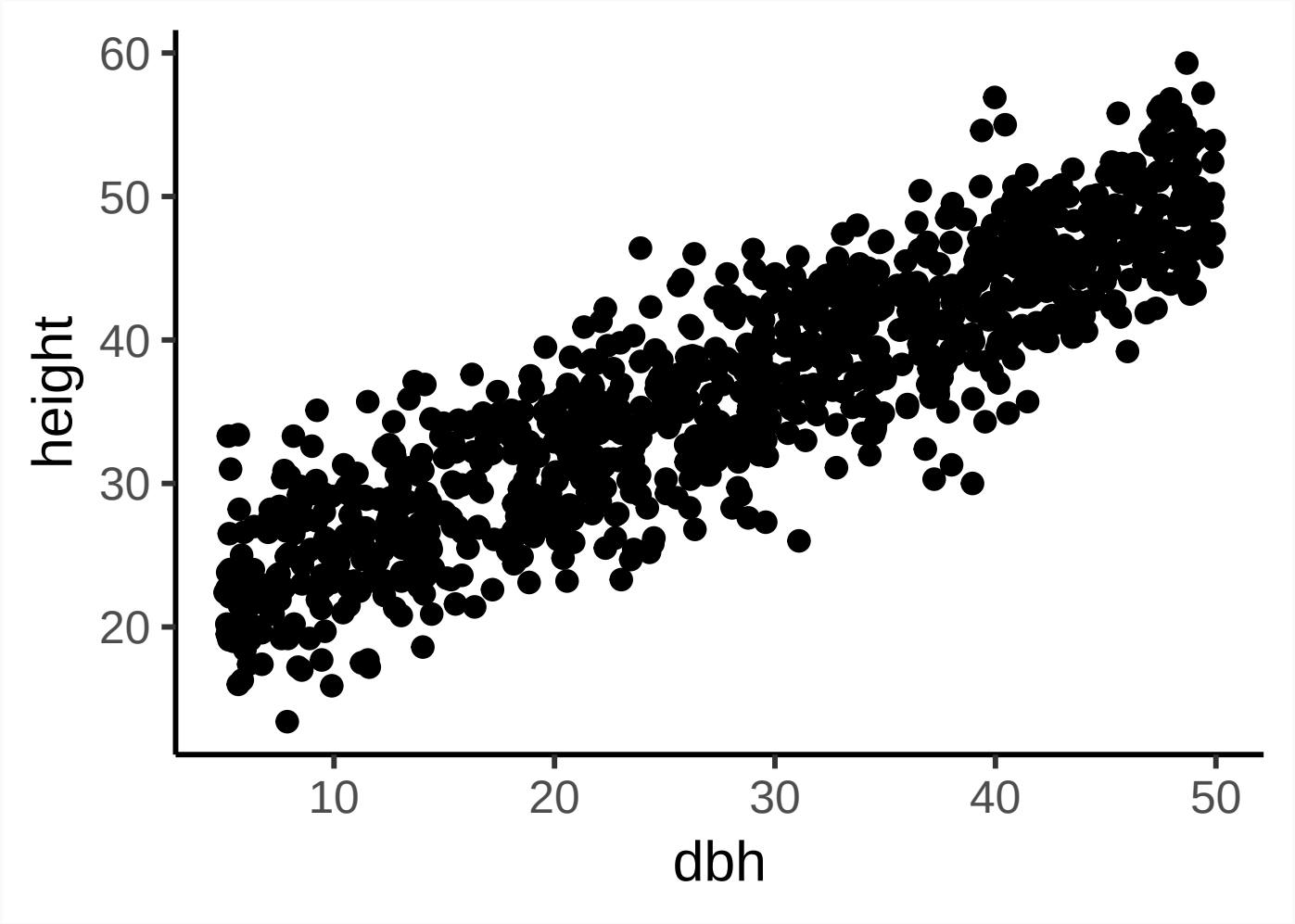
myplot

```
myplot <- ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point()
```



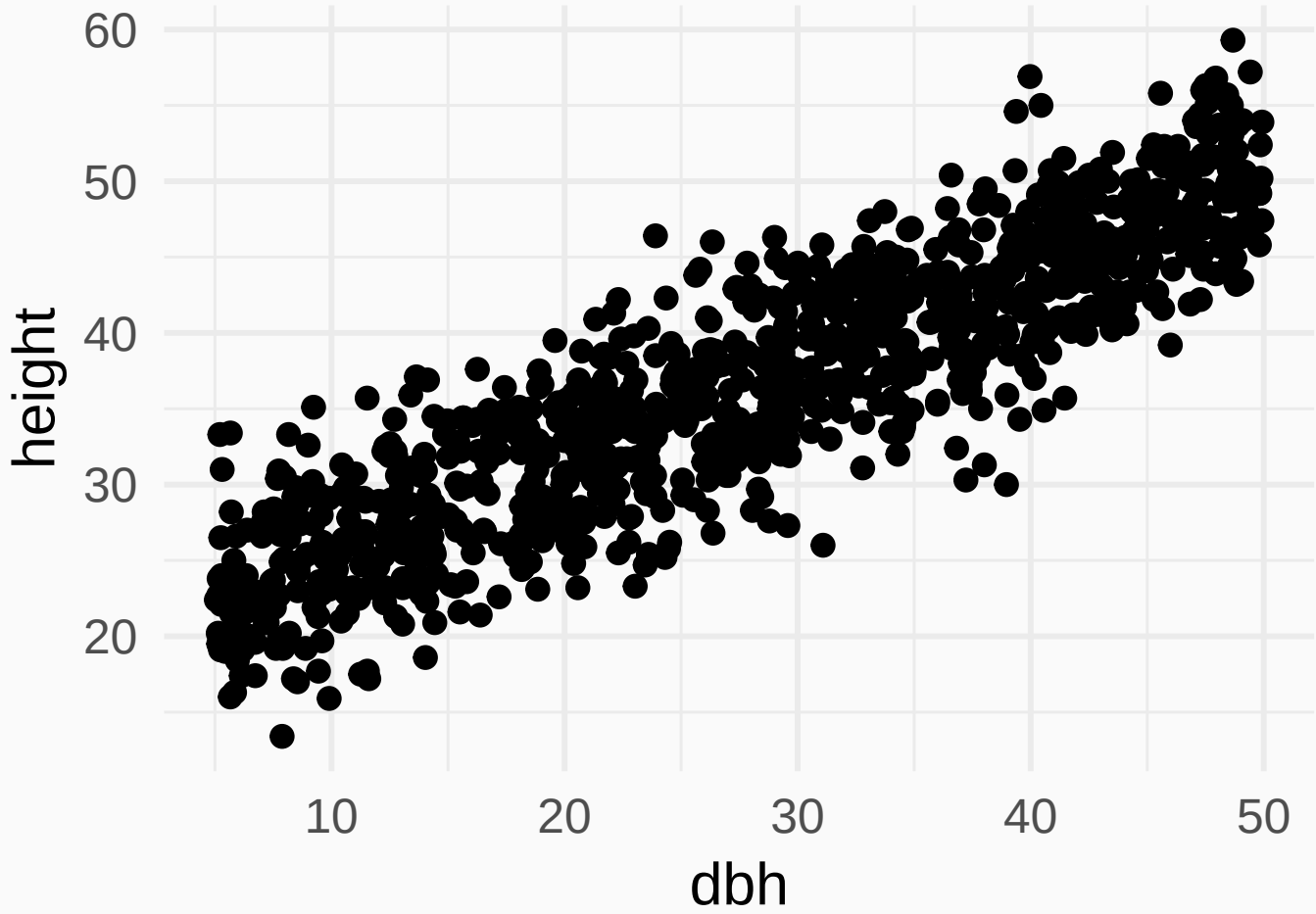
theme_classic

```
myplot + theme_classic()
```



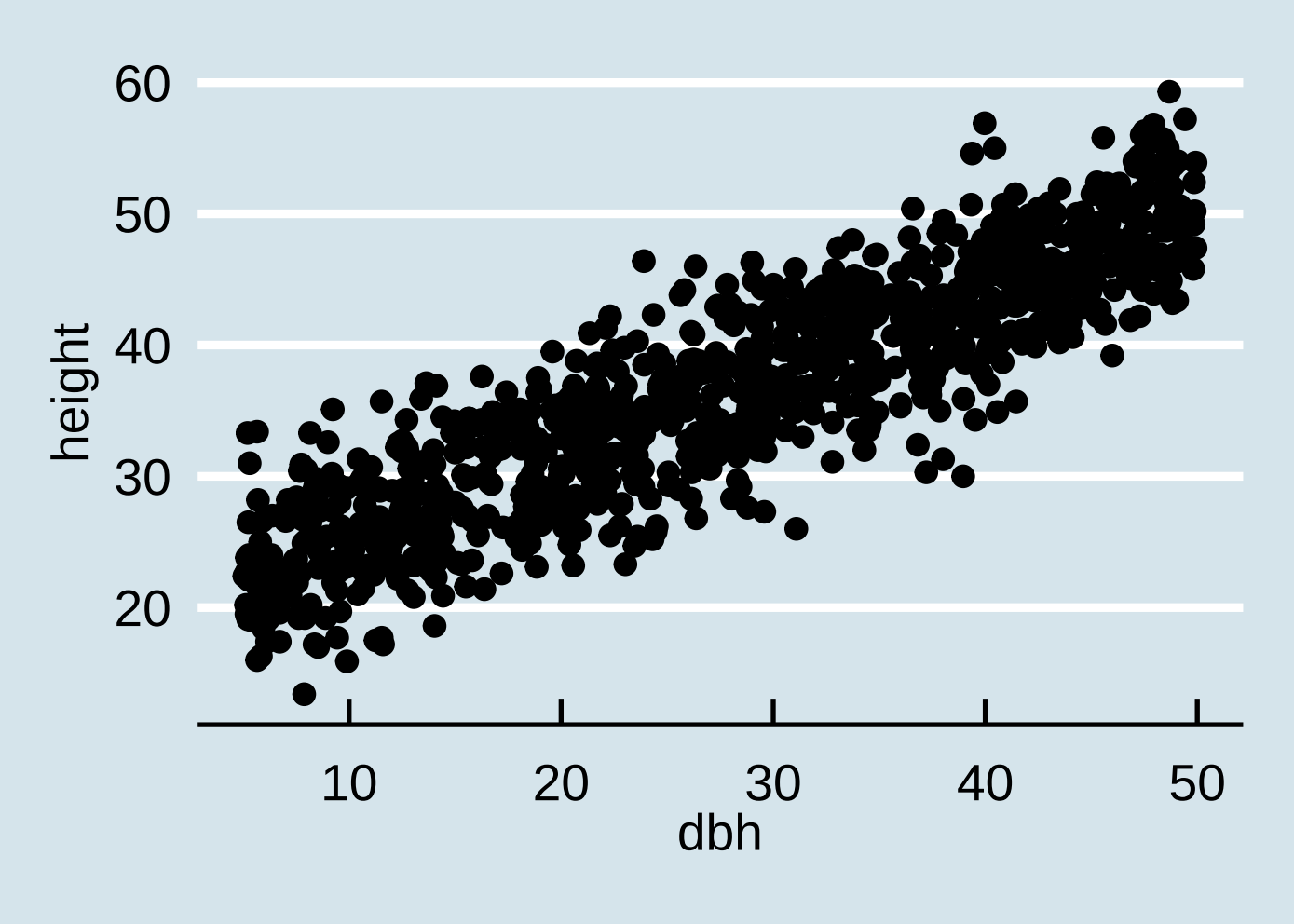
theme_minimal

```
myplot + theme_minimal()
```



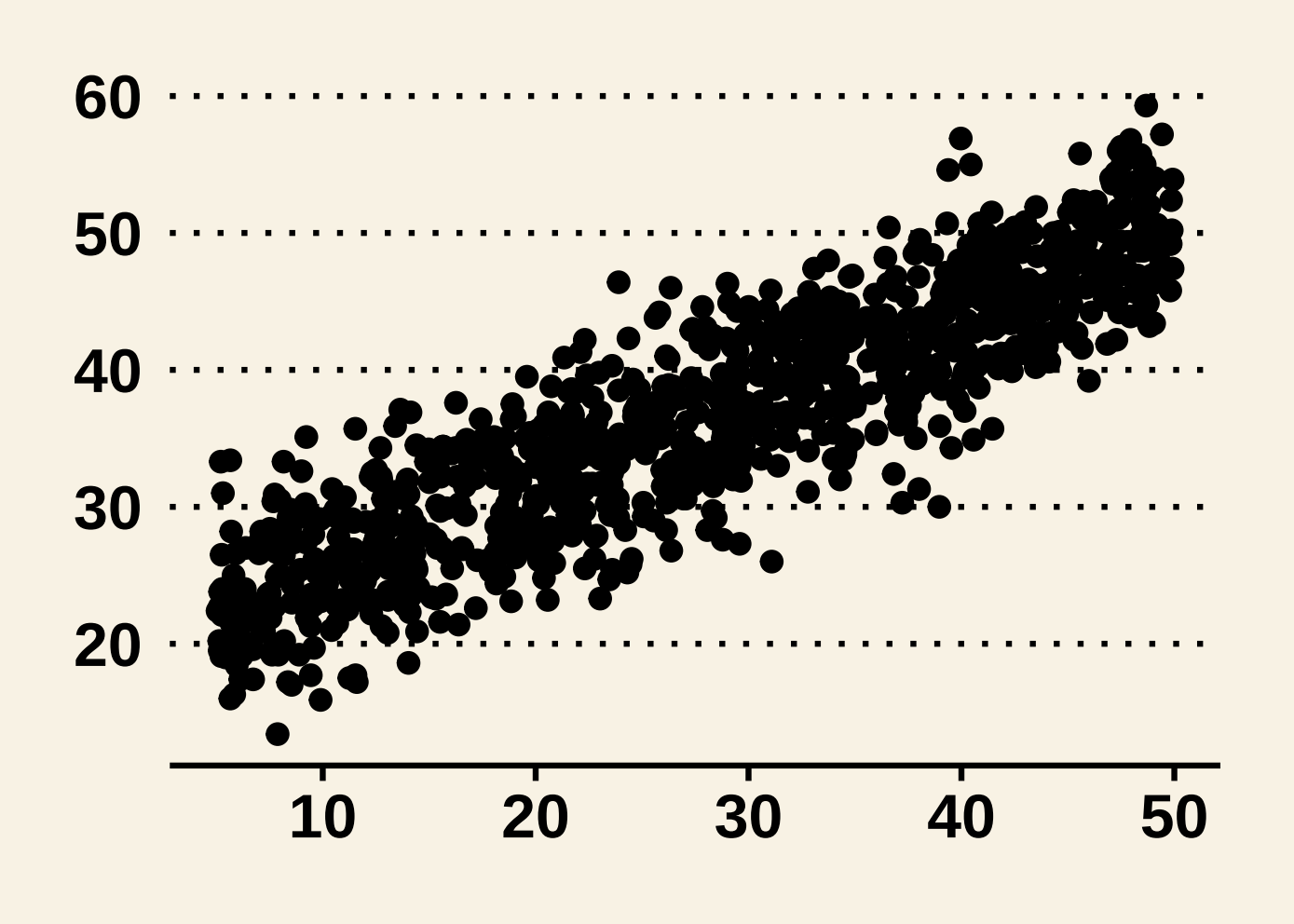
Lots of themes out there

```
library(ggthemes)
myplot + theme_economist()
```



Lots of themes out there

```
myplot + theme_wsj()
```

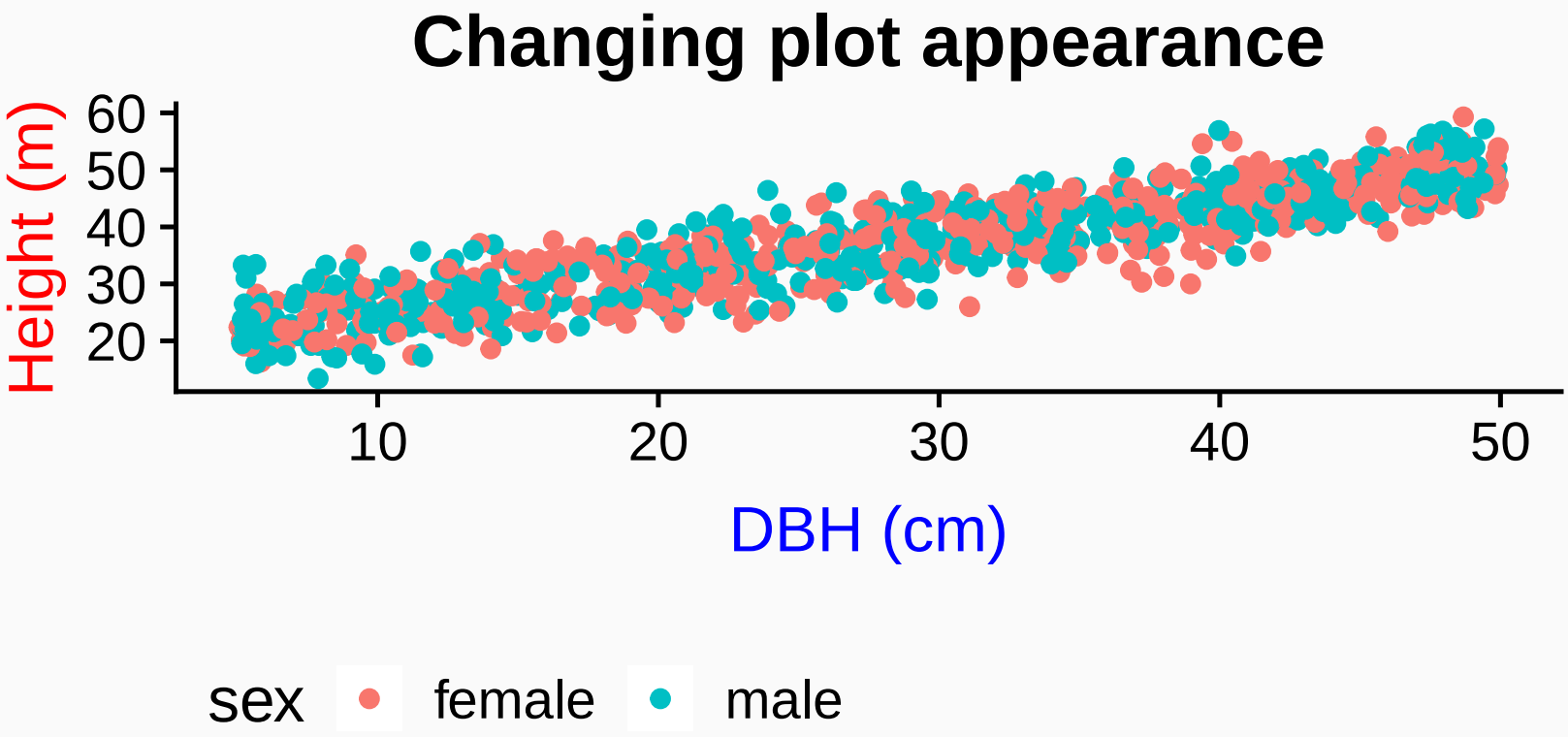


Editing themes

```
?theme
```

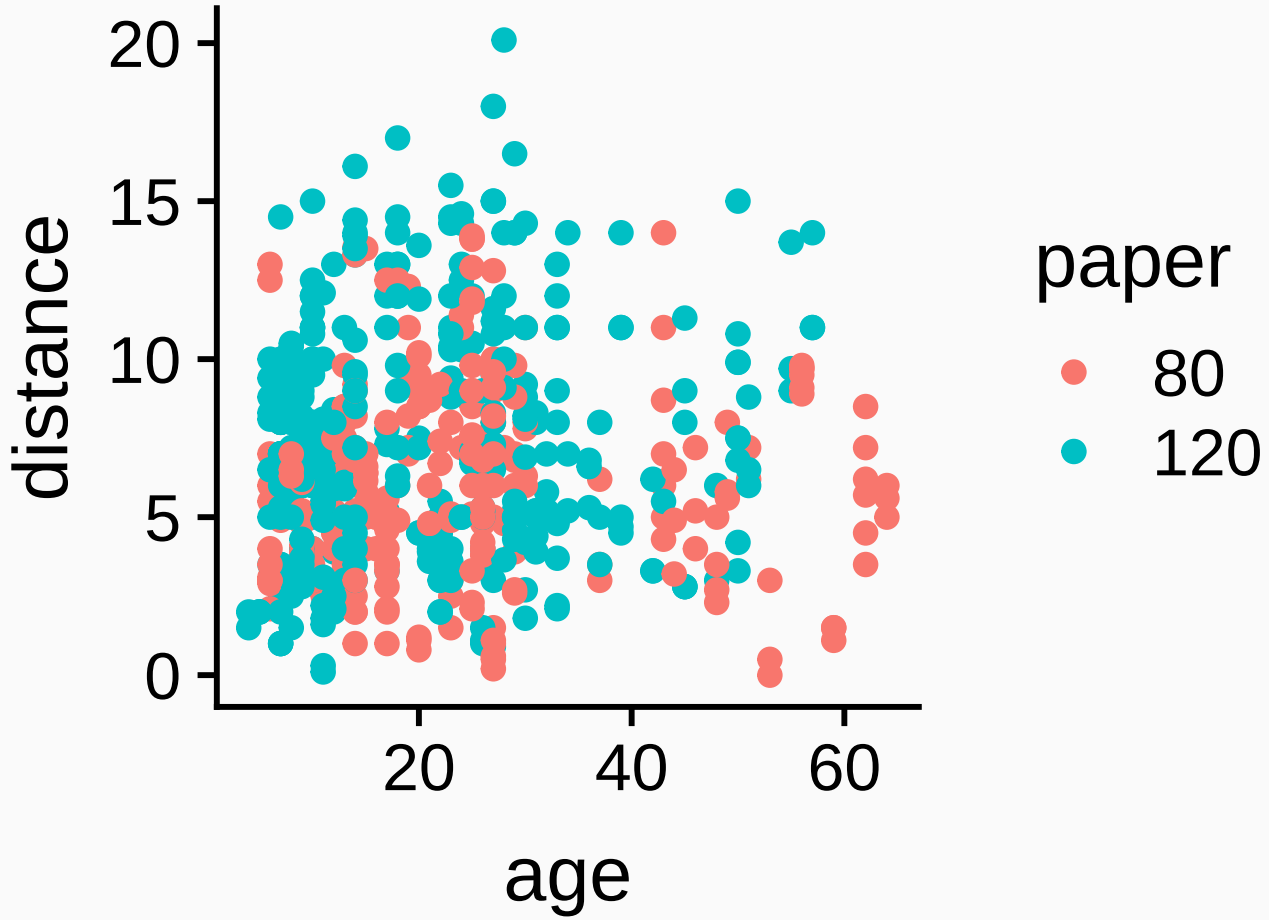
- `element_blank`
- `element_text`
- `element_line`
- `element_rect` (borders & backgrounds)

Exercise: make a plot like this one



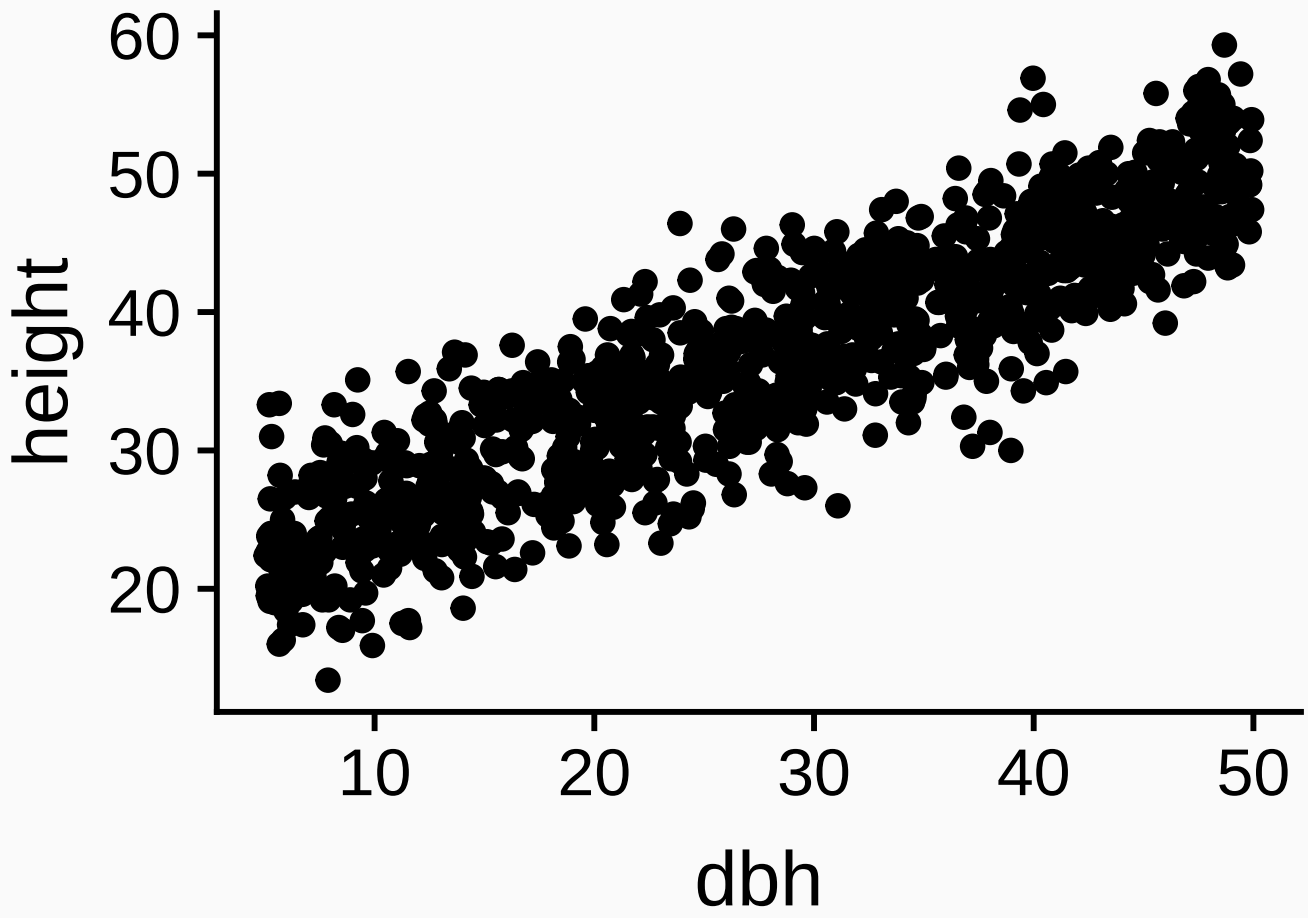
Easily changing appearance with ggthemeassist (Rstudio addin)

https://github.com/calligross/ggthemeassist



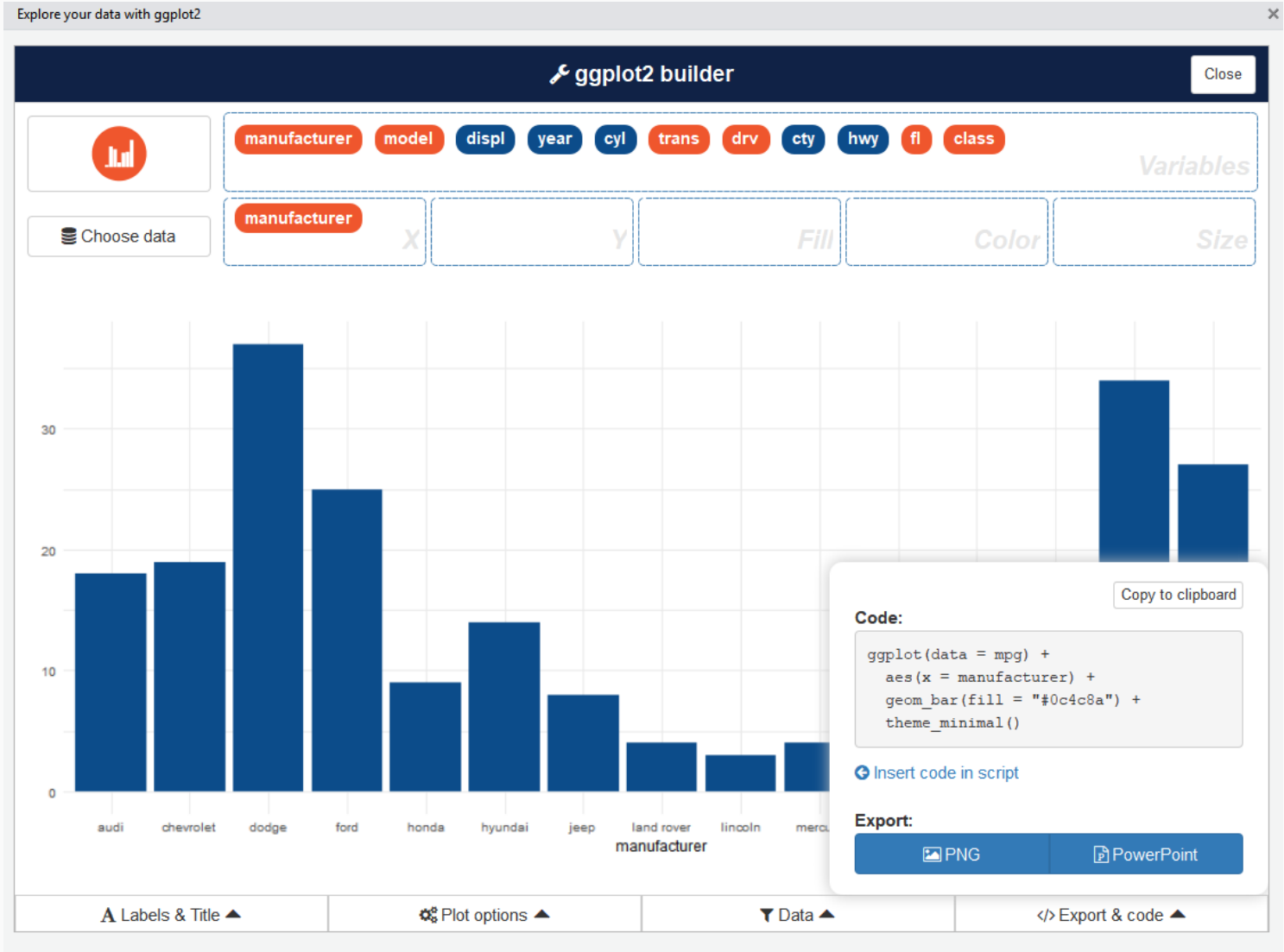
Easily changing appearance with ggedit

https://github.com/metrumresearchgroup/ggedit

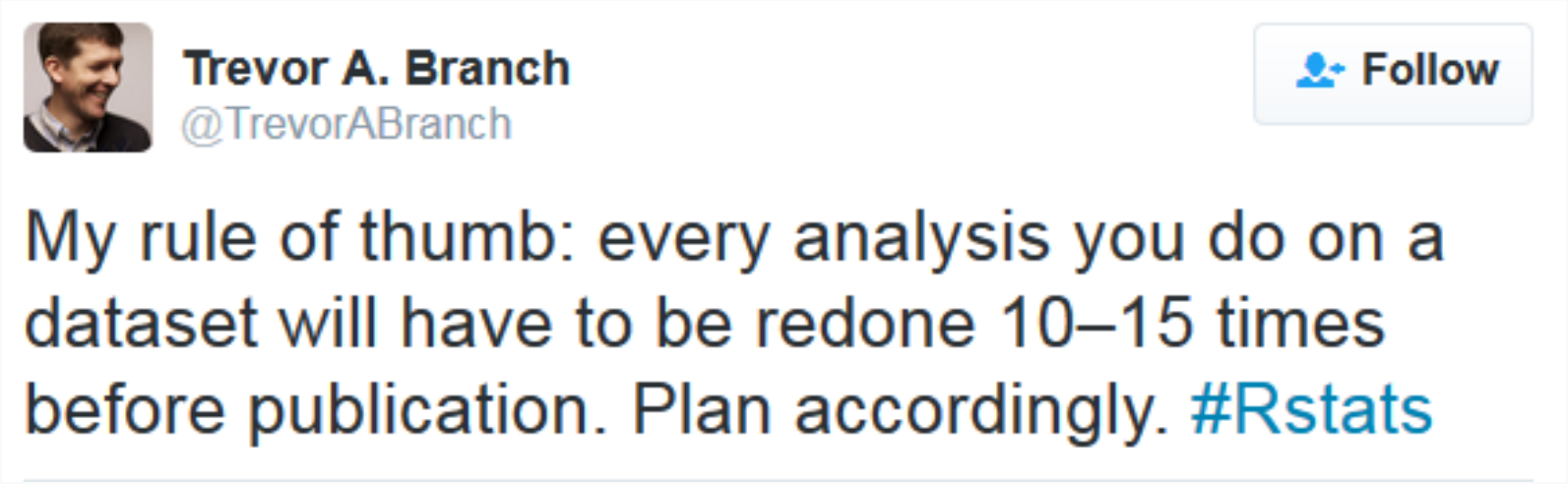


esquisse: ggplot2 builder addin

https://github.com/dreamRs/esquisse



Think twice before editing plots out of R



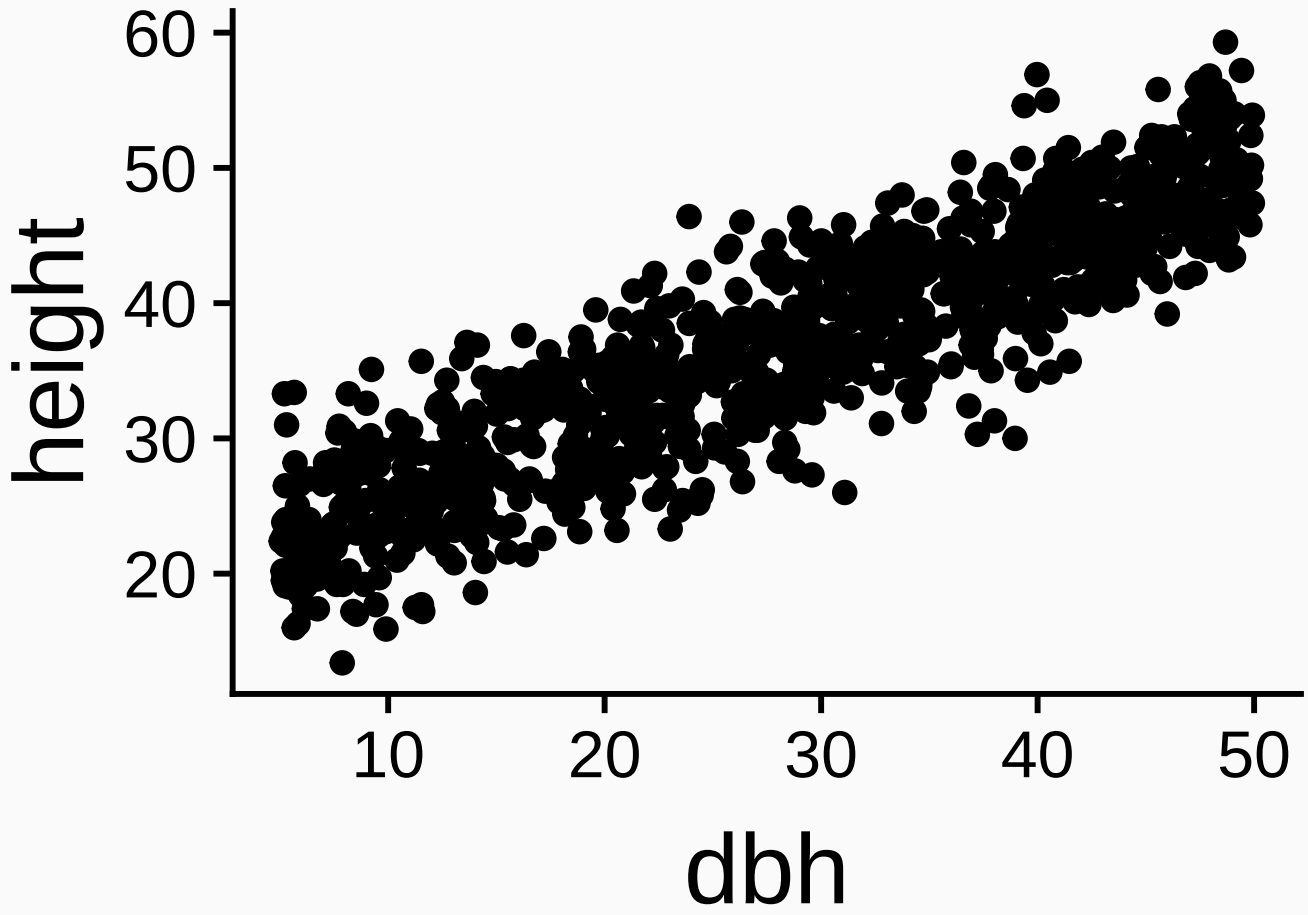
http://mbjoseph.github.io/2015/02/26/plotting.html

serialmentor.com/dataviz/choosing-the-right-visualization-software.html

Think twice before editing plots out of R

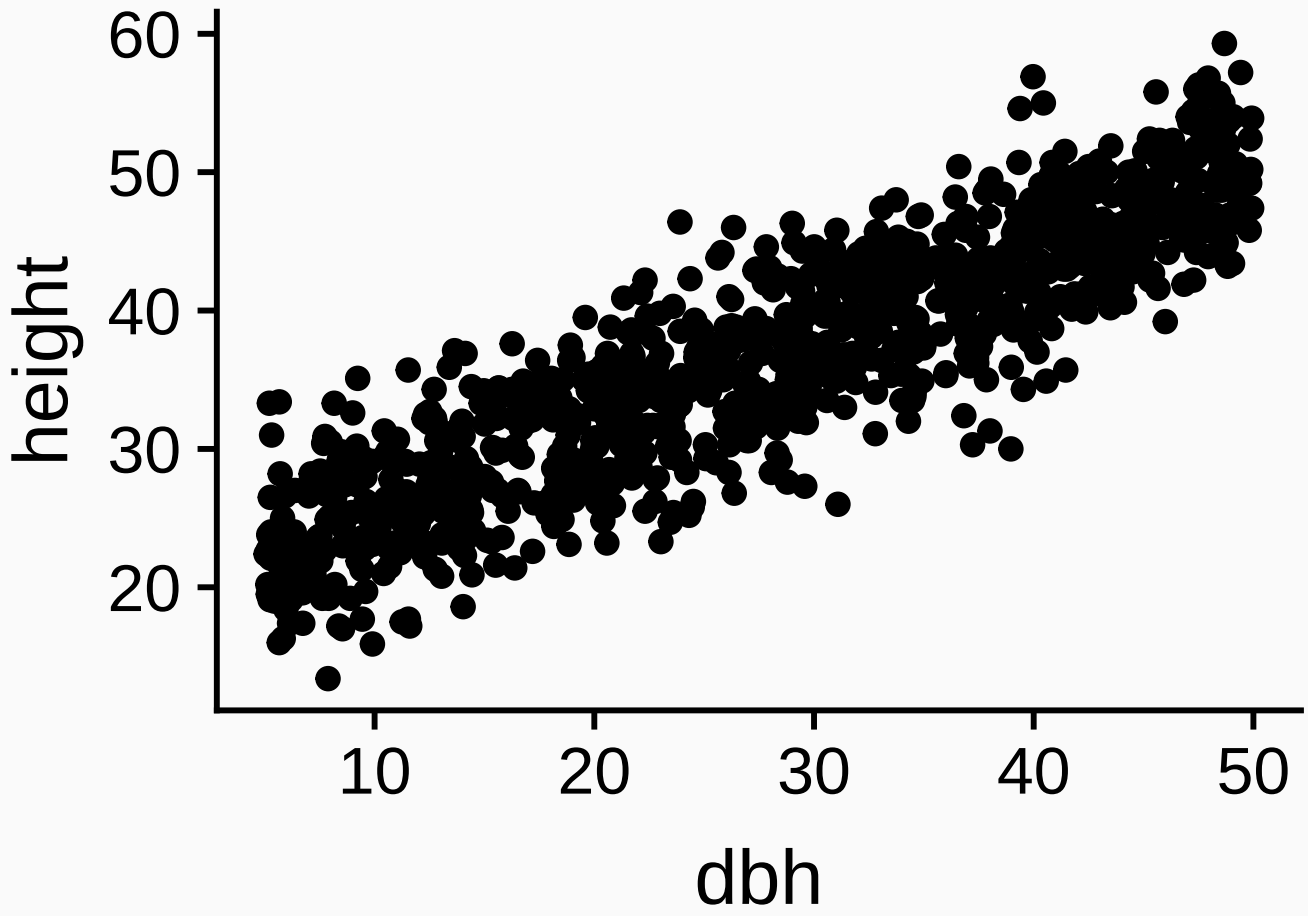
Referee #3: "Please increase font size in all figures"

```
myplot +
  theme(axis.title = element_text(size = 18))
```



Publication-quality plots

```
library(cowplot)
myplot
```



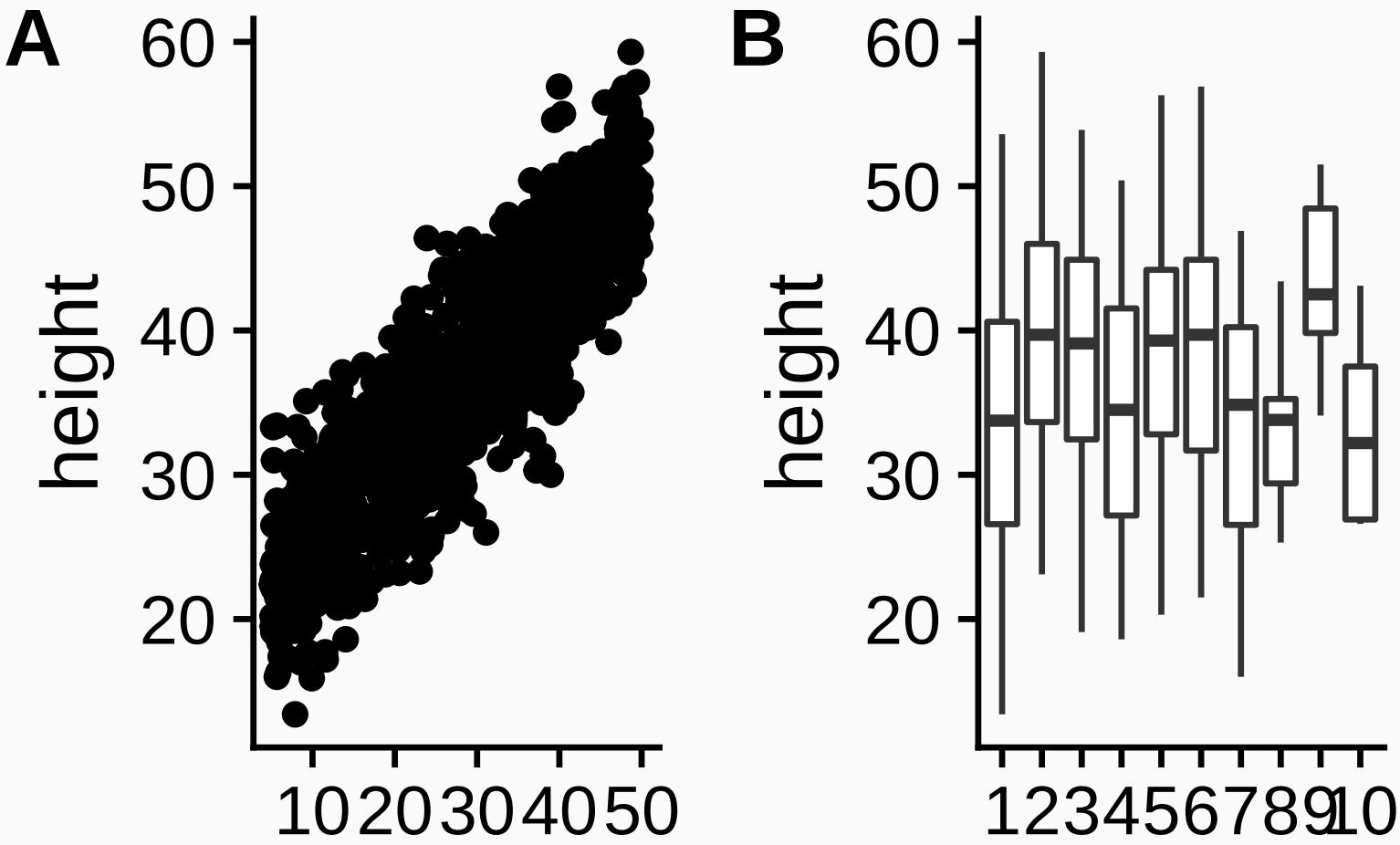
Some publication themes:

https://gist.github.com/Pakillo/c2c7ea11c528cc2ee20f#themes

Composite figures

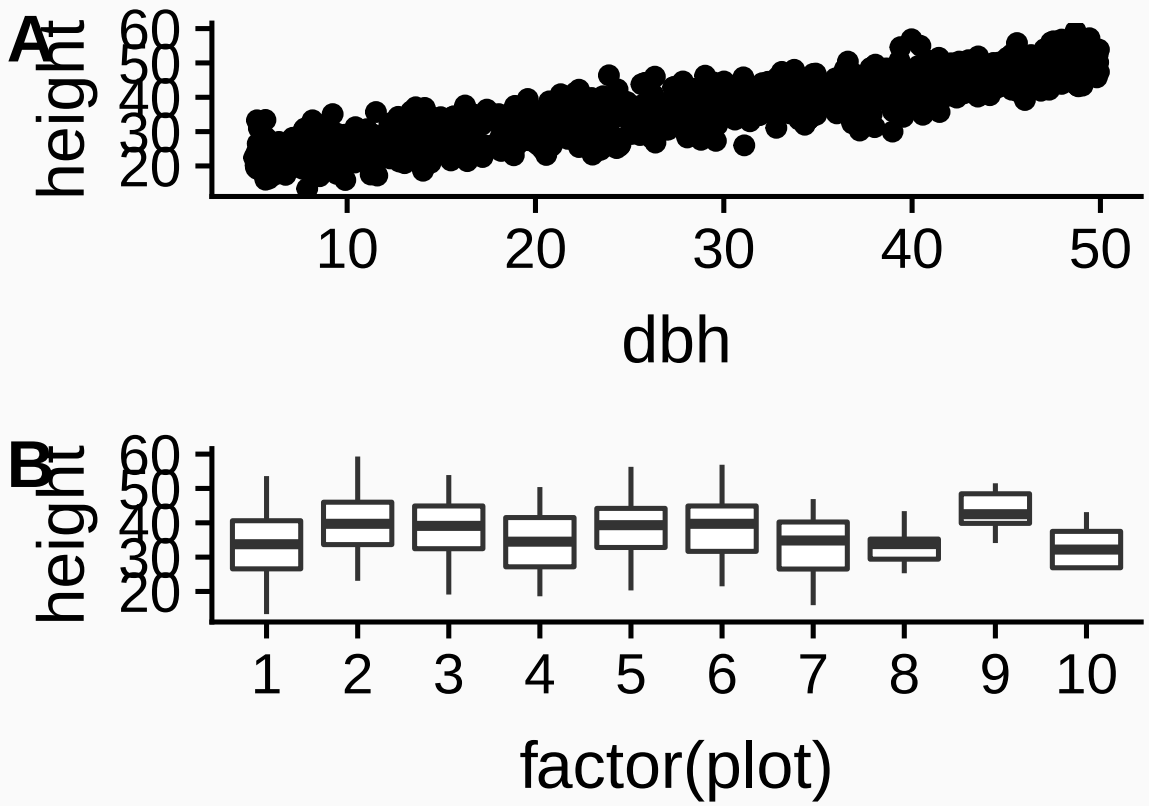
Composite figures: cowplot

```
library(cowplot)
plot1 <- ggplot(trees) + aes(dbh, height) + geom_point()
plot2 <- ggplot(trees) + aes(factor(plot), height) + geom_boxplot()
plot_grid(plot1, plot2, labels = "AUTO")
```



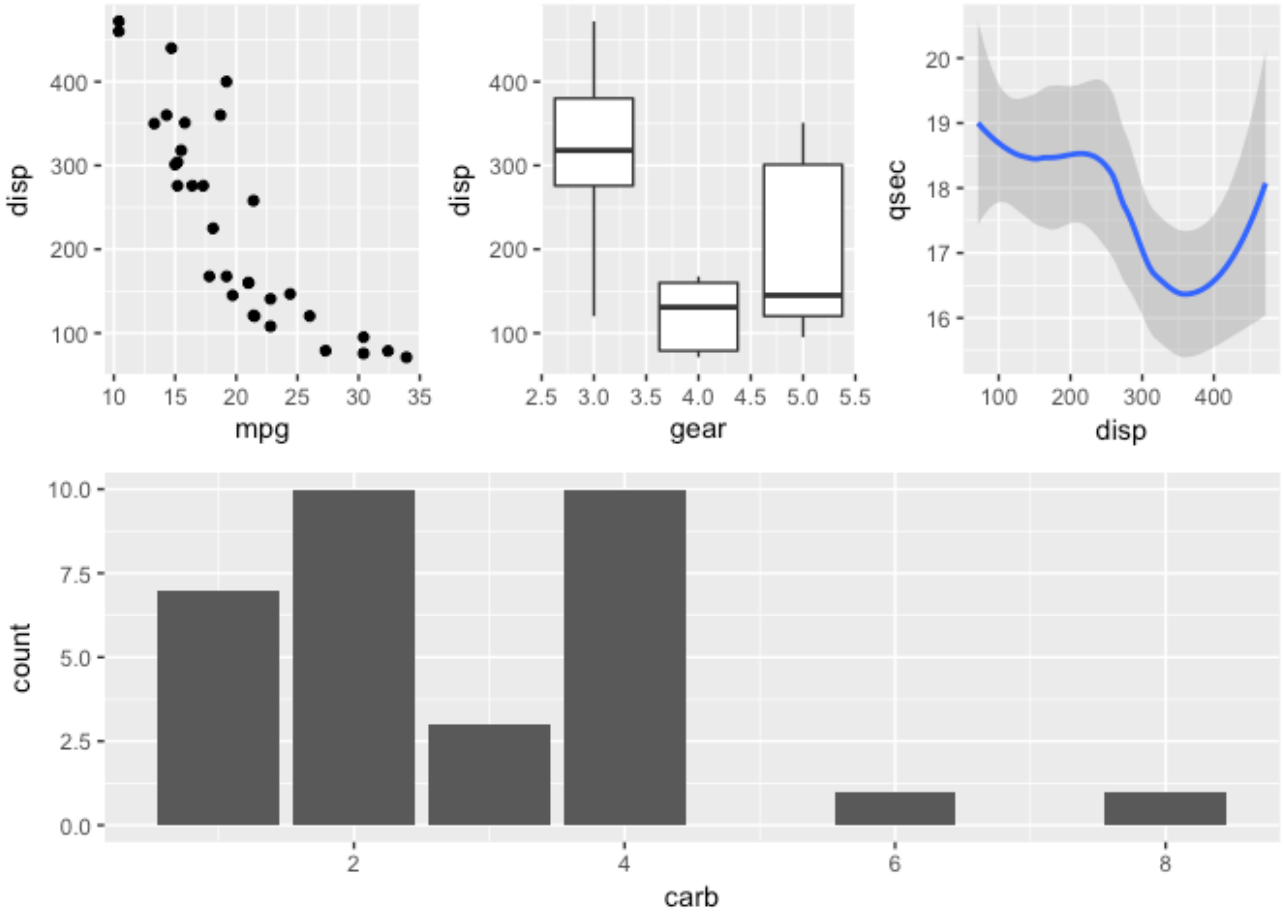
Composite figures

```
plot_grid(plot1, plot2, labels = "AUTO", ncol = 1)
```



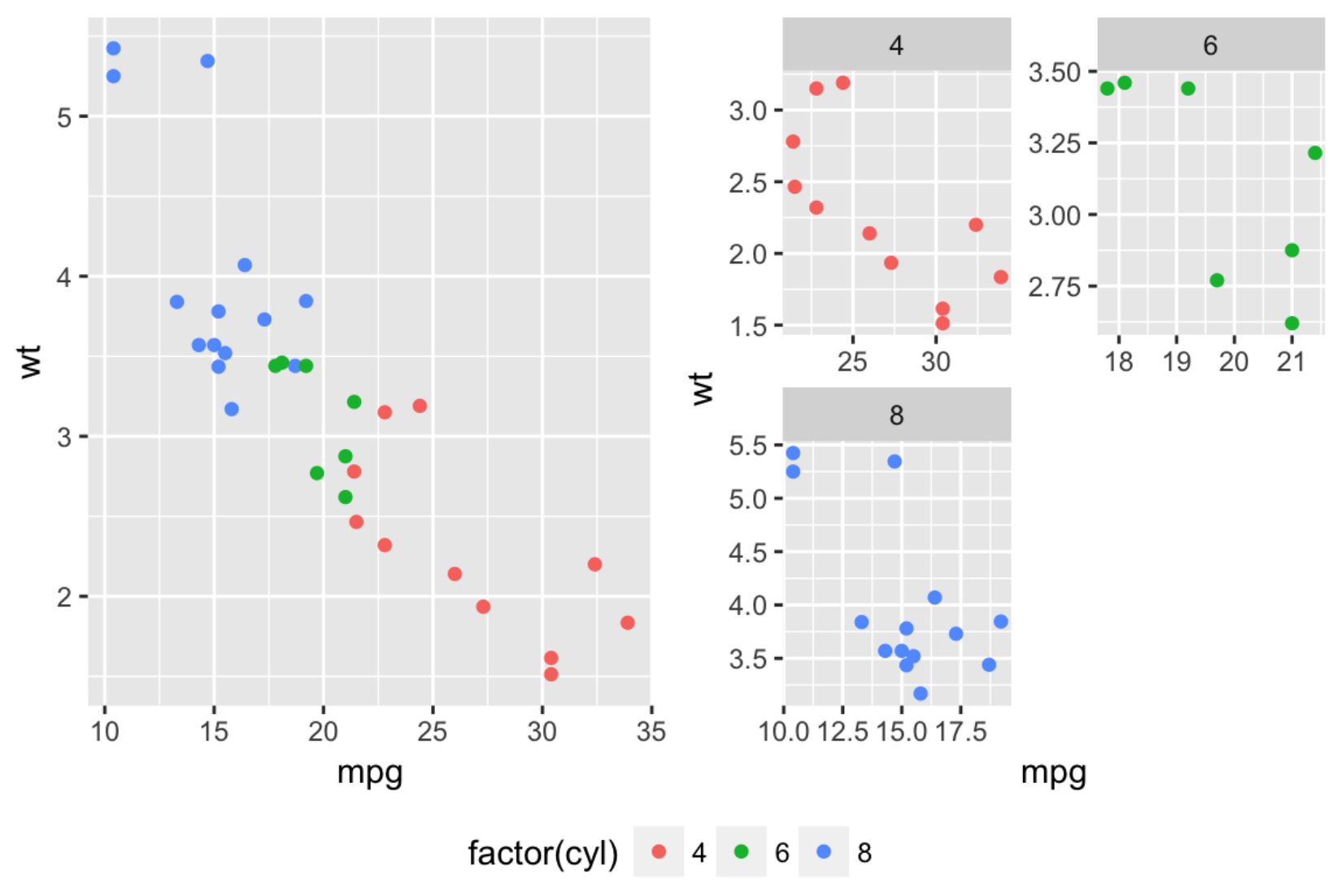
Composite figures: patchwork

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



https://github.com/thomasp85/patchwork

Composite figures: egg



https://cran.r-project.org/web/packages/egg/index.html

Saving plot

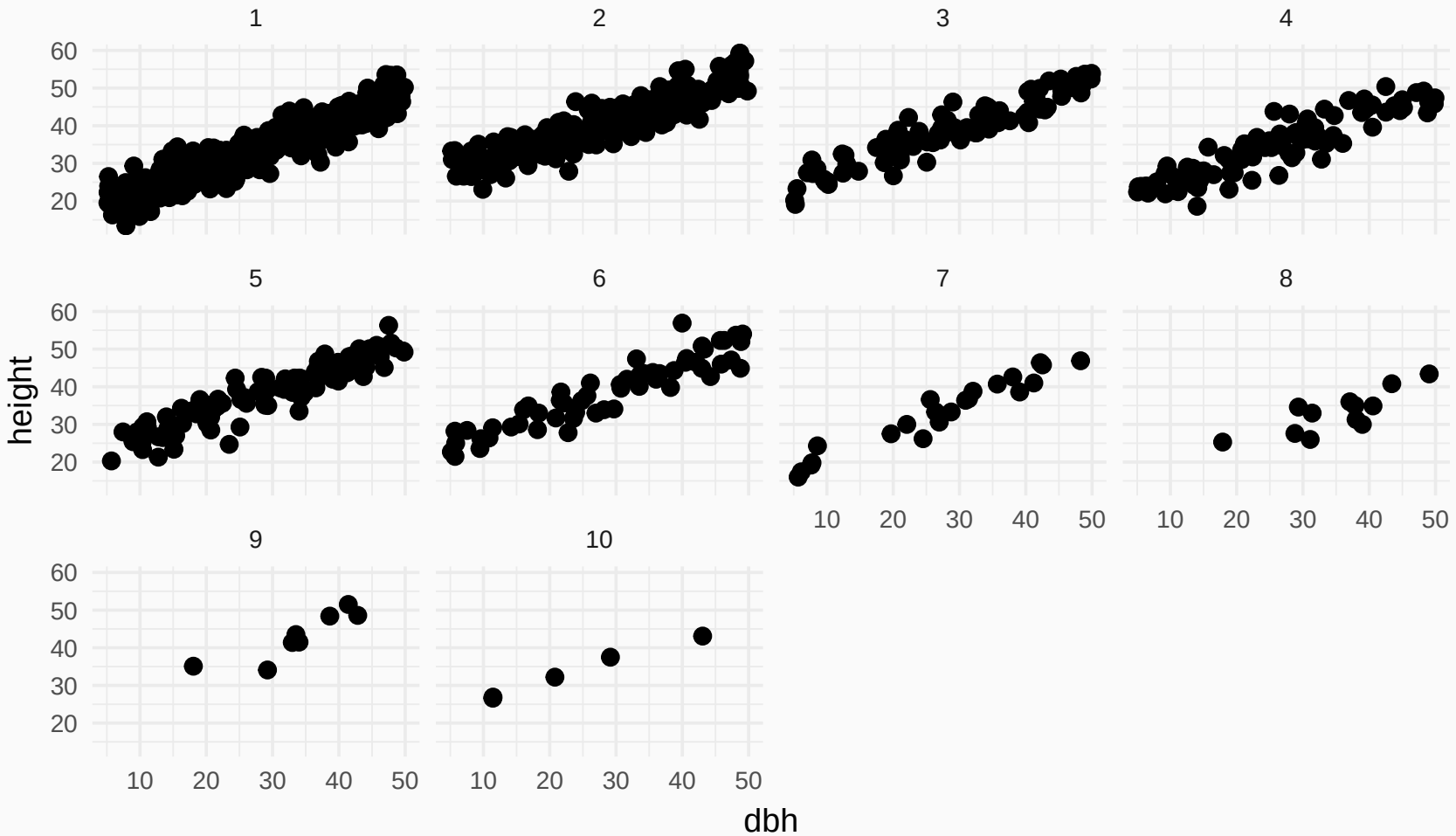
```
ggsave("myplot.pdf")
```

```
save_plot("myplot.pdf")
```

Facetting (small multiples)

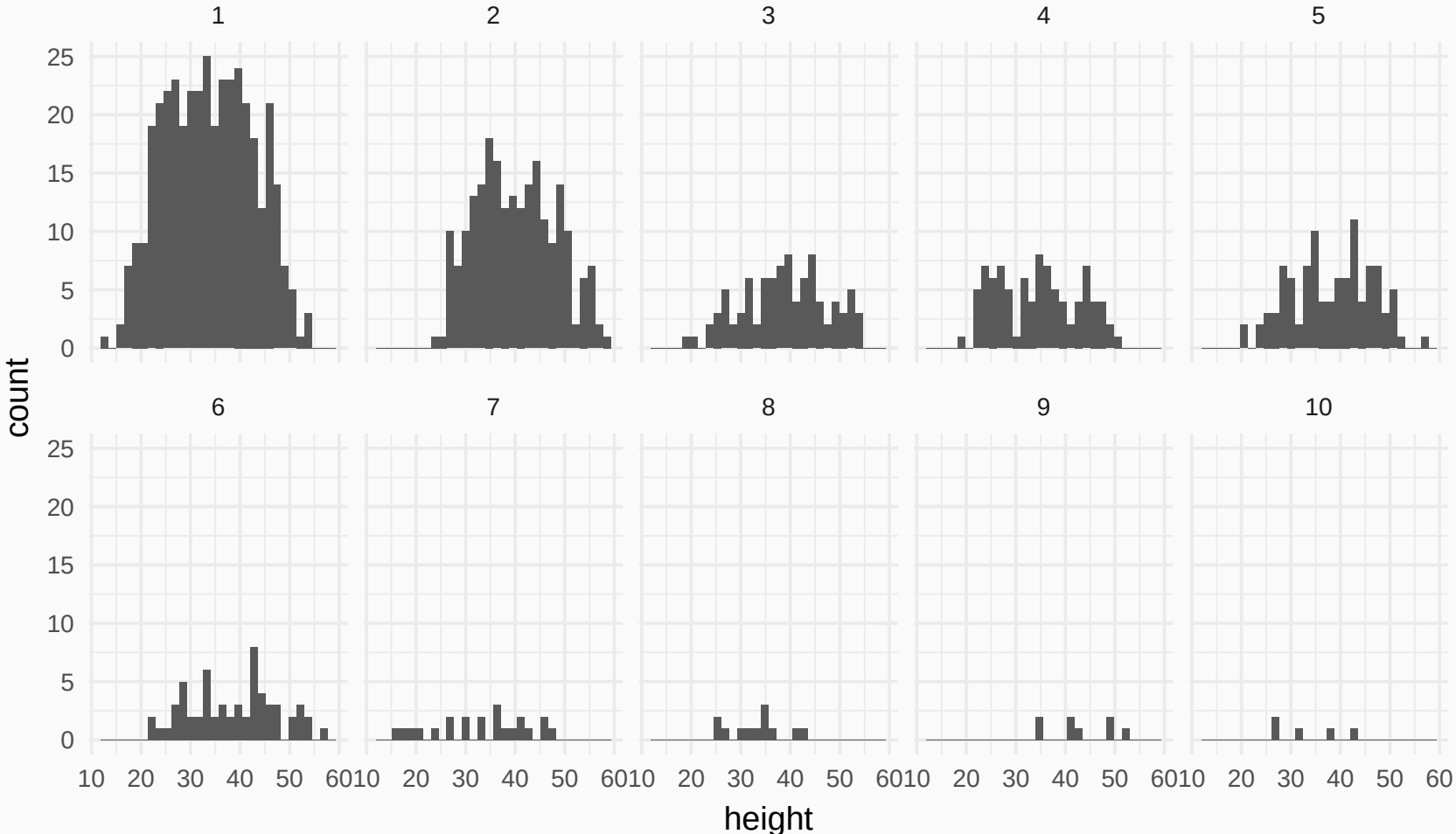
Facetting

```
ggplot(trees) + aes(dbh, height) +
  geom_point() + theme_minimal(base_size = 8) +
  facet_wrap(~plot)
```



Facetting

```
ggplot(trees) +
  geom_histogram(aes(height)) + theme_minimal(base_size = 8) +
  facet_wrap(~plot, nrow = 2)
```

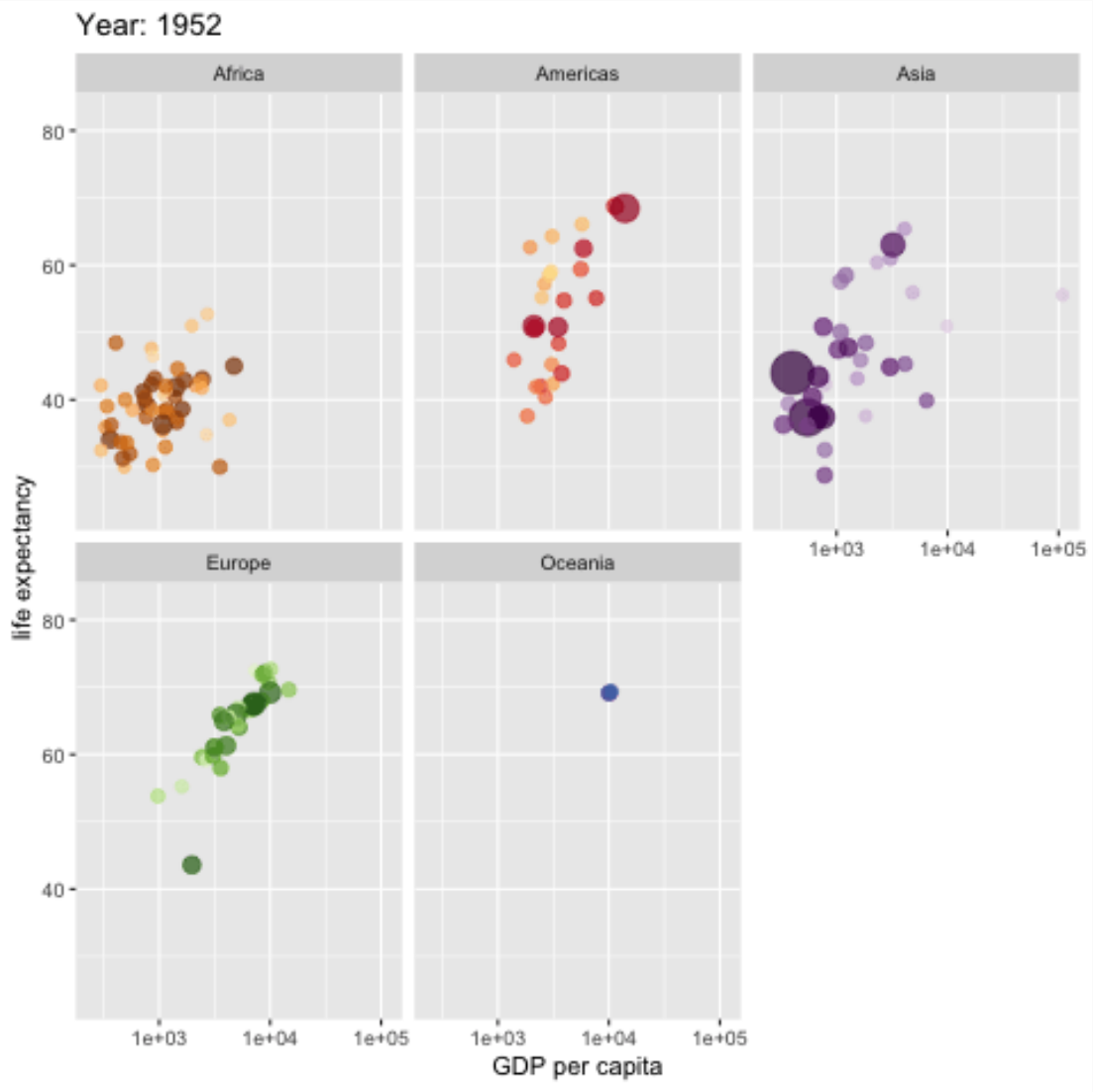


Interactivity: plotly

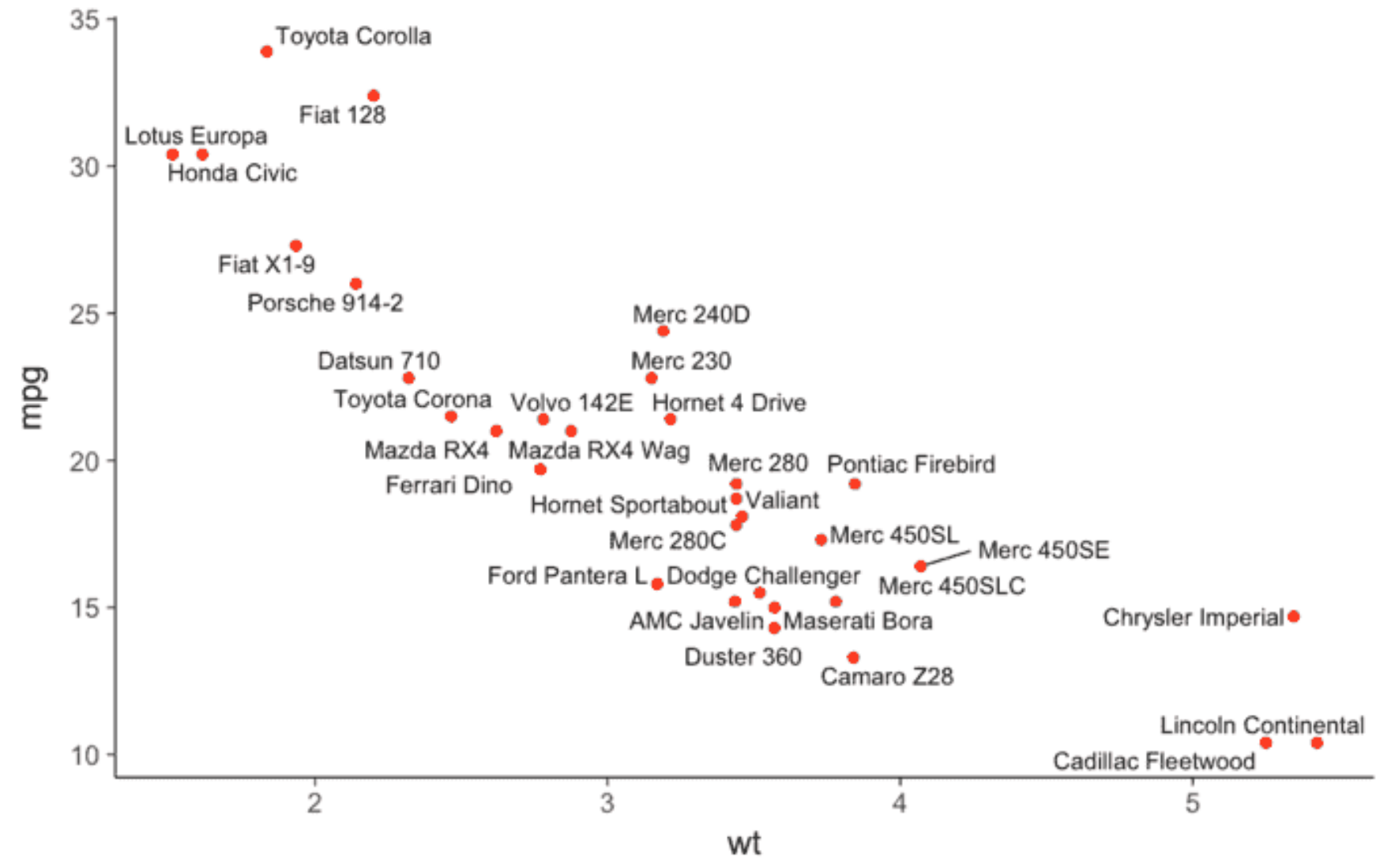
```
library(plotly)
myplot <- ggplot(trees) +
  aes(x = dbh, y = height) +
  geom_point()
ggplotly(myplot)
```

Animated graphs

https://github.com/thomasp85/gganimate



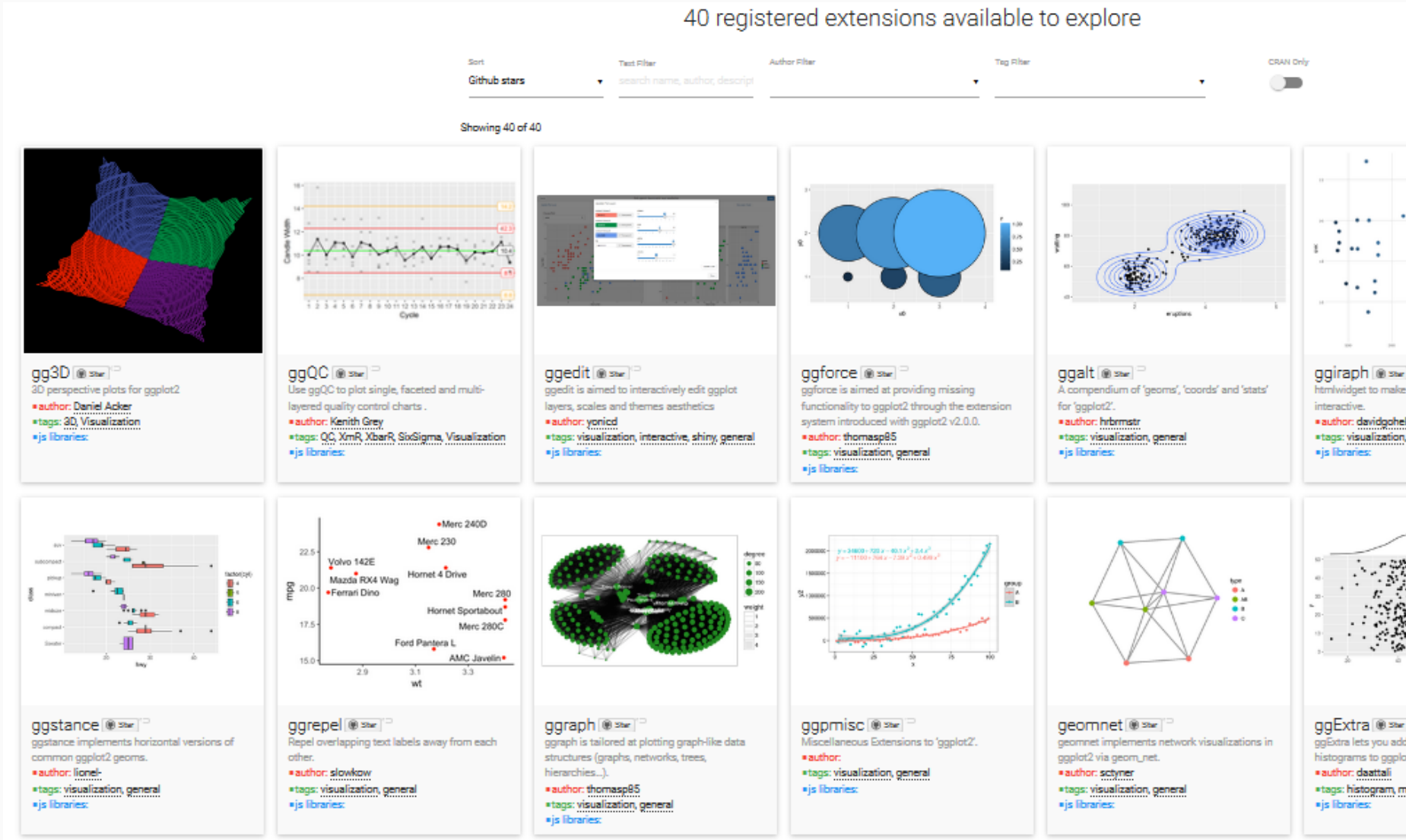
Automatic label placement



https://cran.r-project.org/package=ggrepel

Many extensions!

https://www.ggplot2-exts.org/

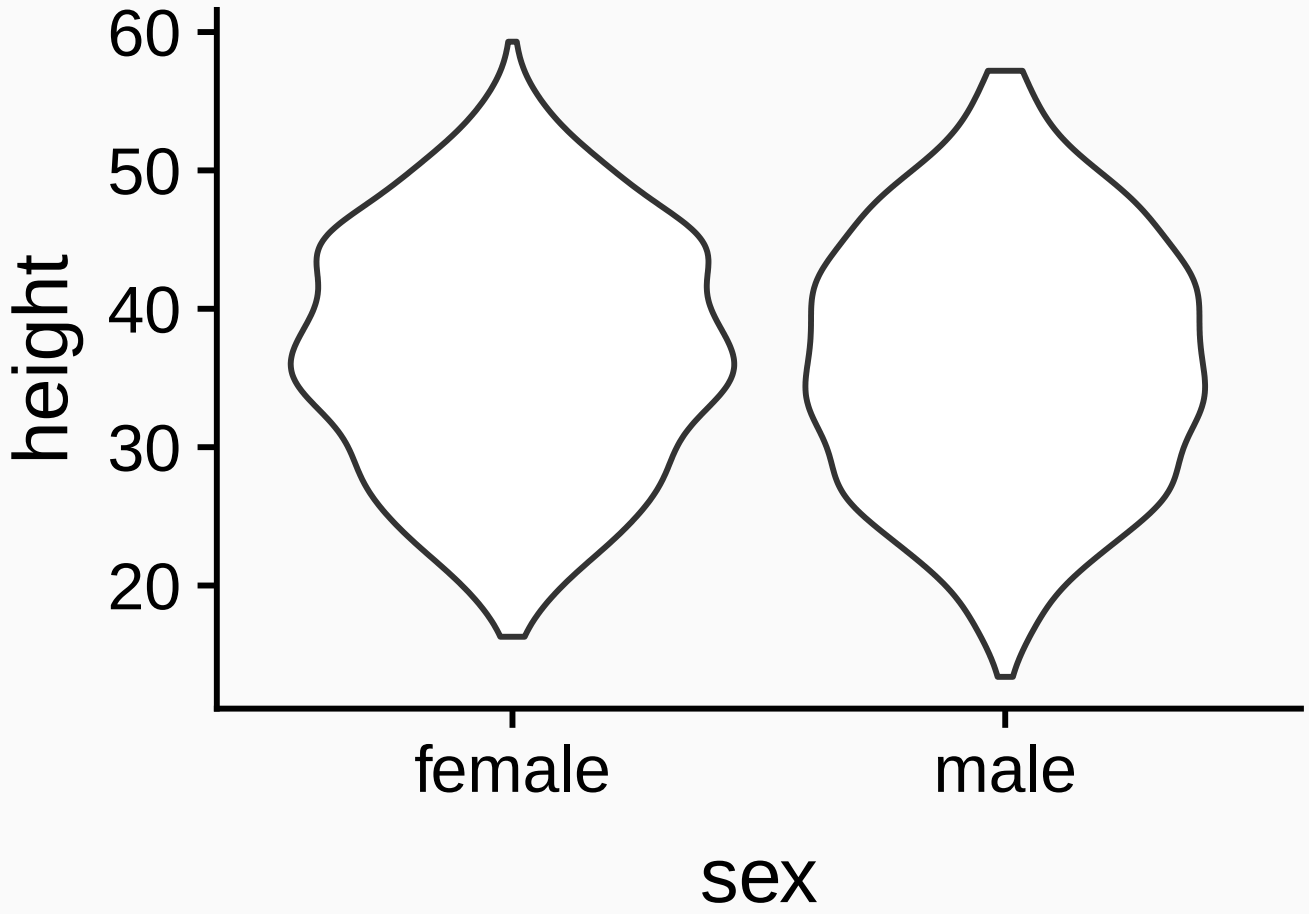


Summary

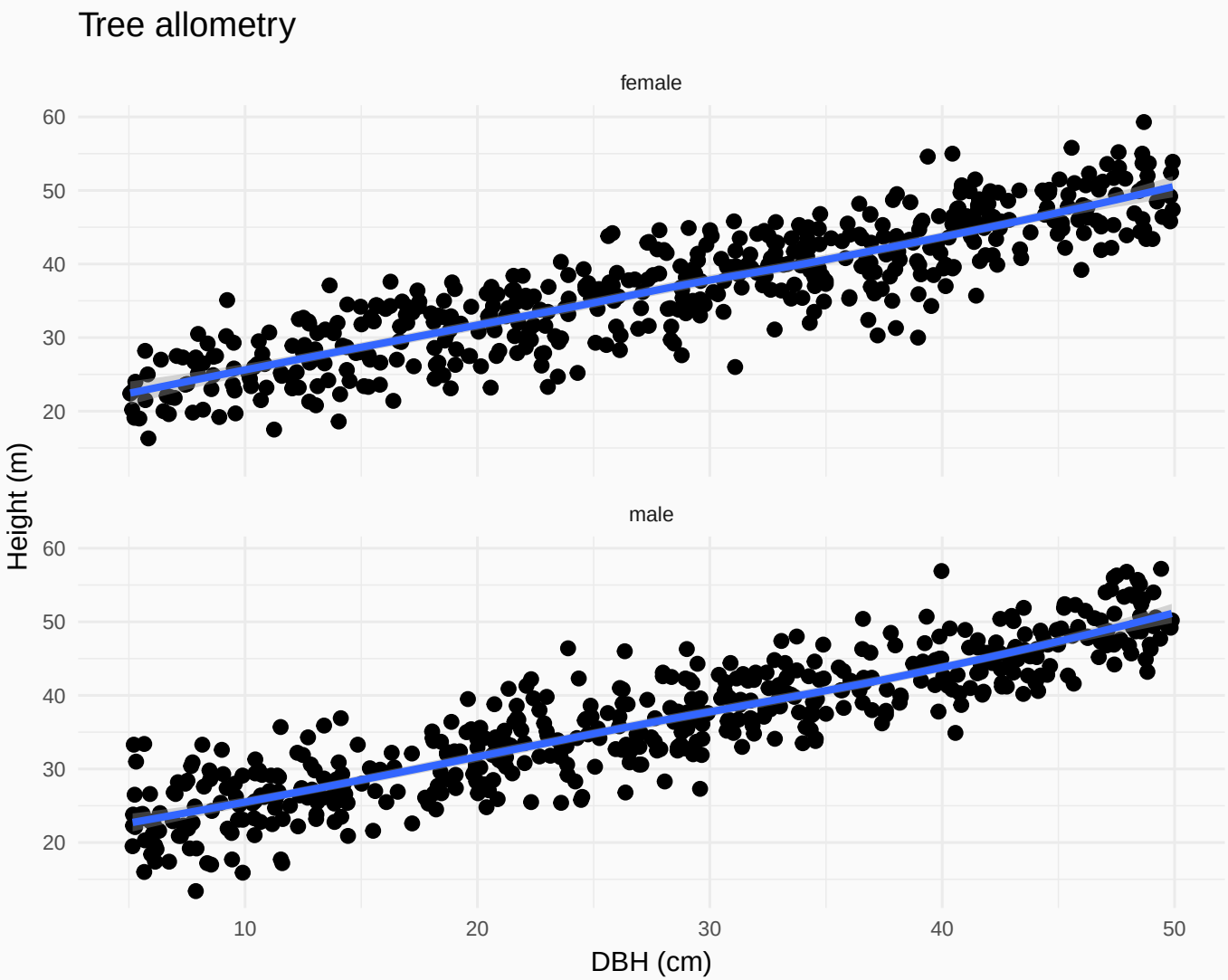
Grammar of graphics

- **Data** (tidy data frame)
- **Layers** (*geoms*: points, lines, polygons. . .)
- **Aesthetics** mappings (x, y, size, colour. . .)
- **Scales** (colour, size, shape. . .)
- **Facets** (small multiples)
- **Themes** (appearance)
- **Coordinate system** (Cartesian, polar, map projections. . .)

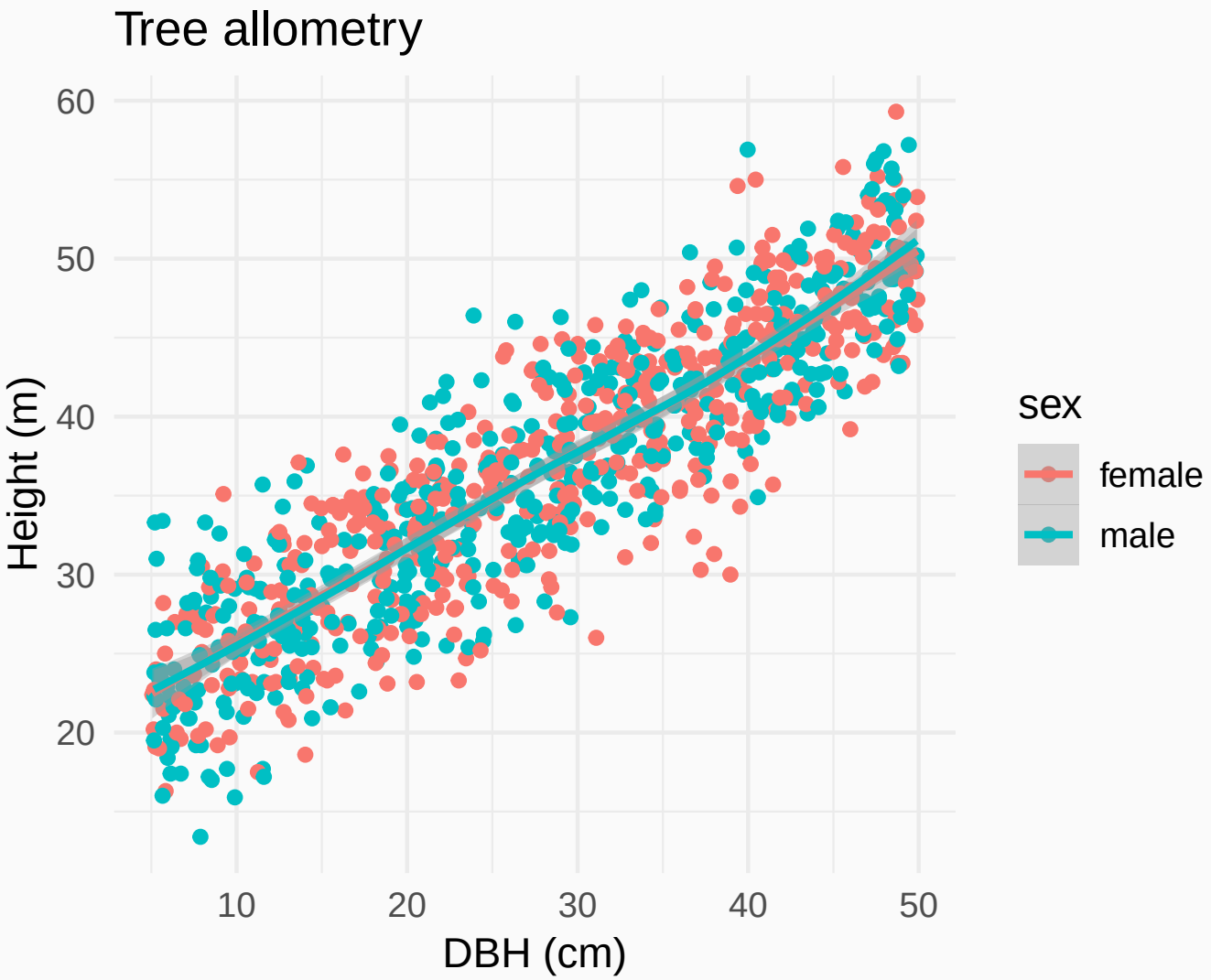
Exercise: make a plot like this one



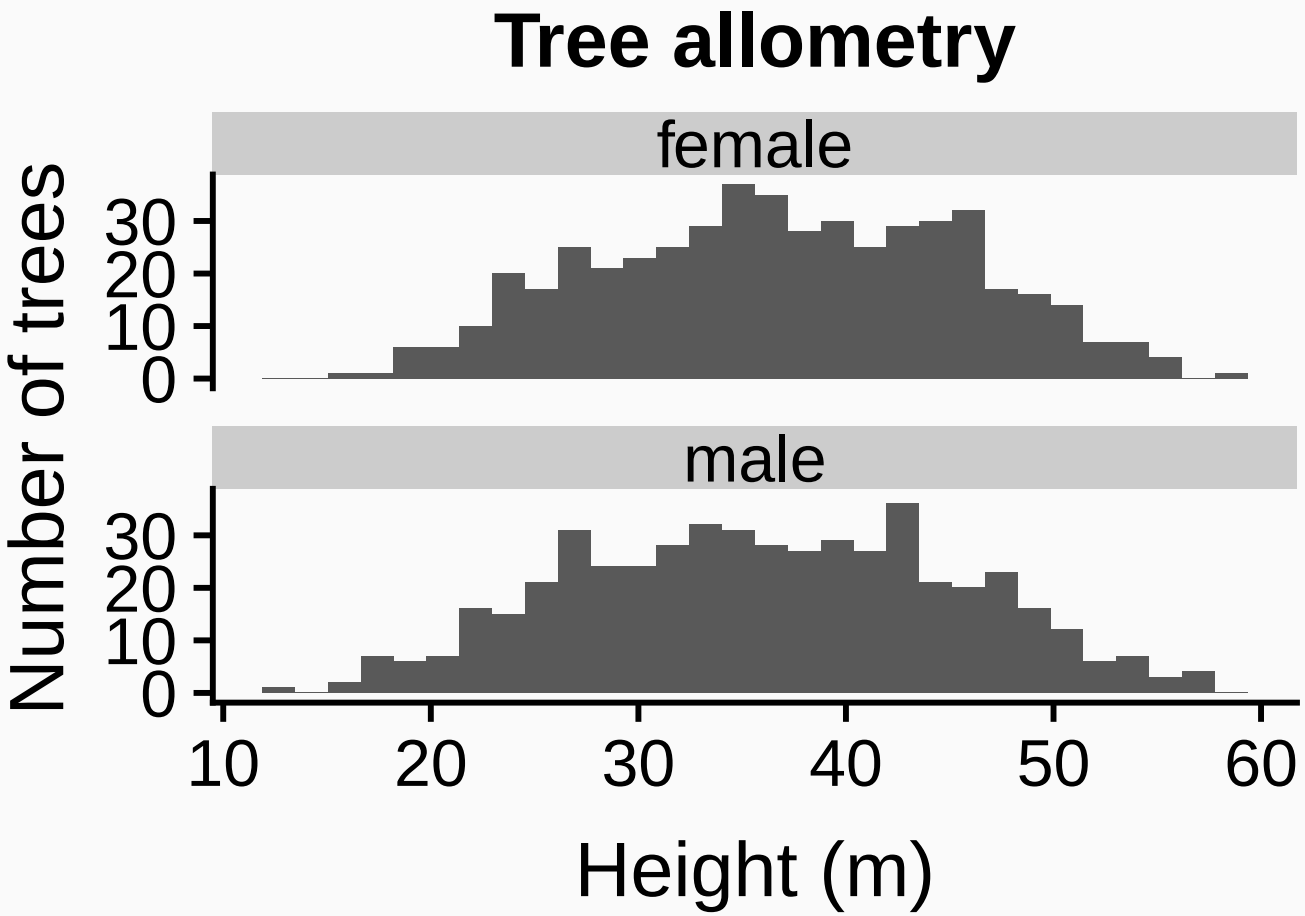
Exercise: make a plot like this one



Exercise: make a plot like this one



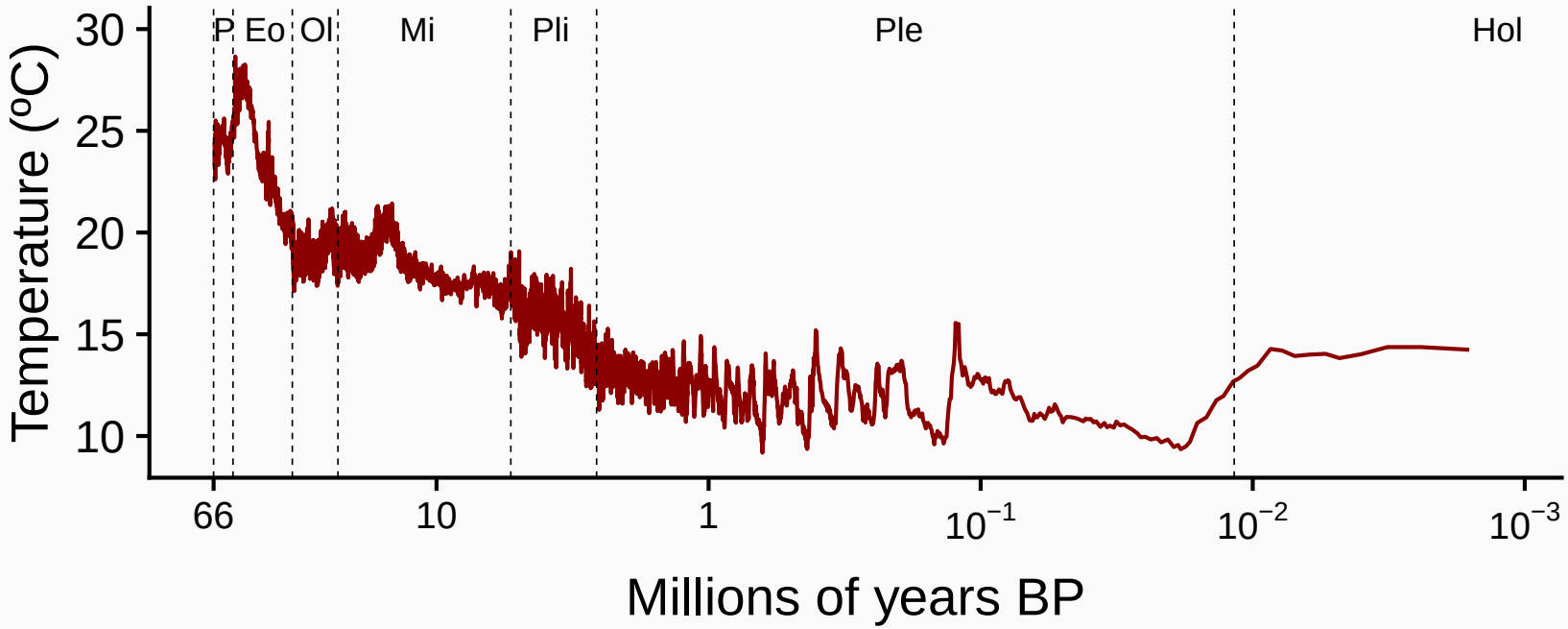
Exercise: make a plot like this one



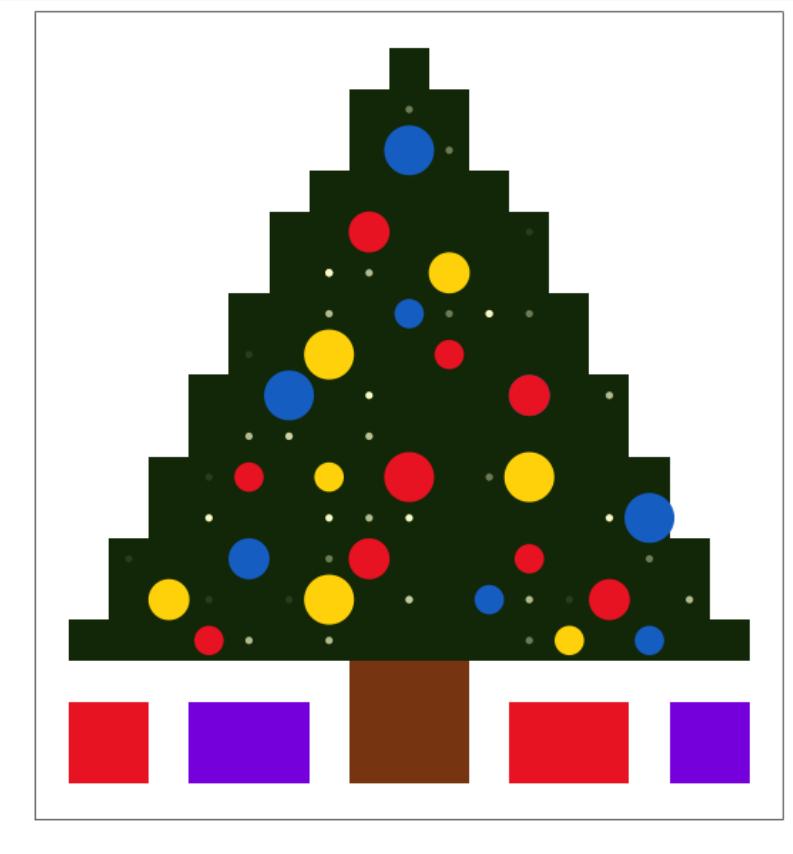
Exercise: make a plot like this one

Exercise: make a plot like this one

Data from http://www.columbia.edu/~mhs119/Sensitivity+SL+CO2/Table.txt



Exercise: make a plot like this one



END

Slides and source code available at https://github.com/Pakillo/ggplot-intro