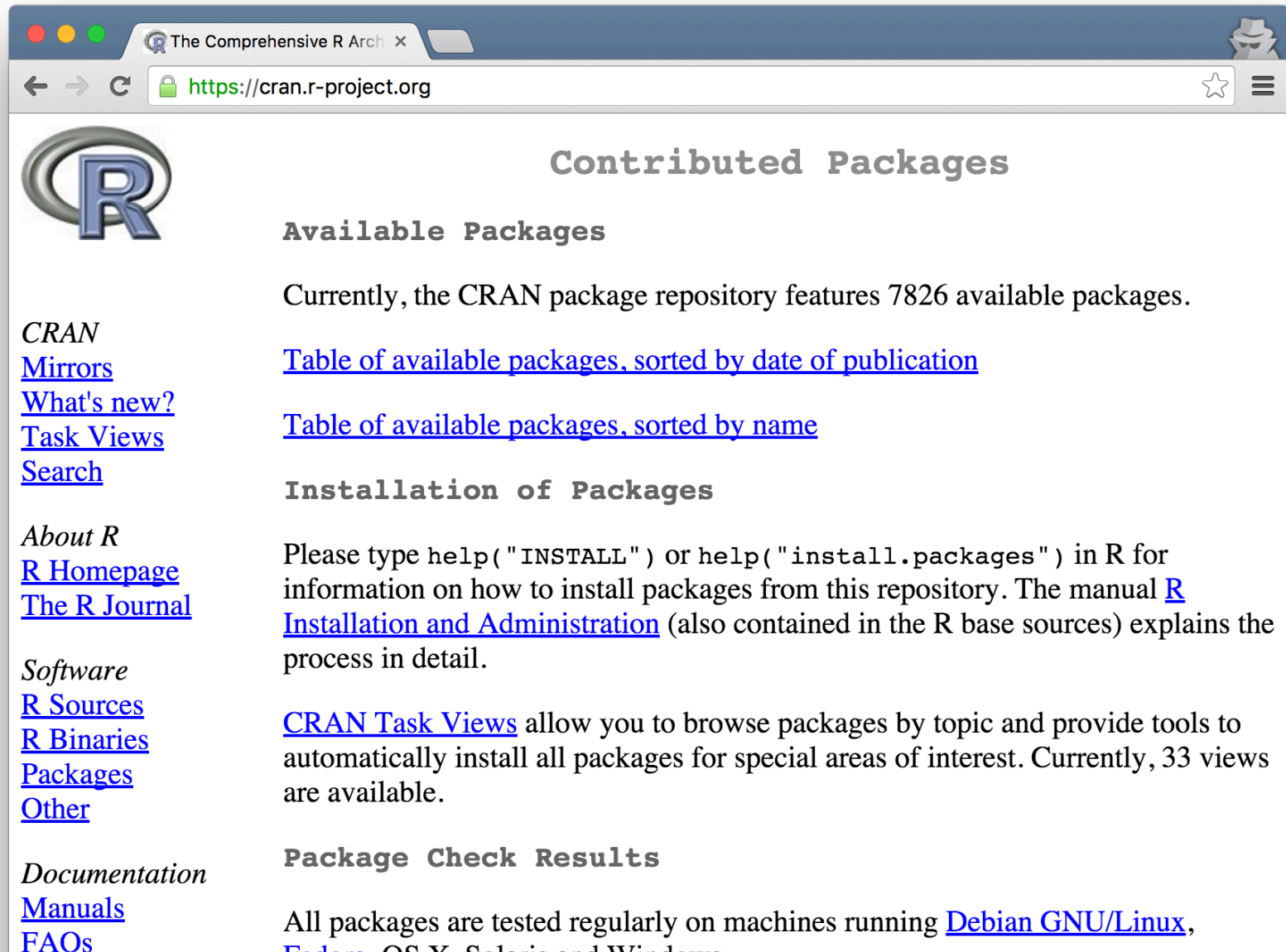



Extending R through packages:
There's a package for everything

R packages are available on CRAN (Comprehensive R Archive Network)

A screenshot of a web browser displaying the CRAN (Comprehensive R Archive Network) website. The browser's address bar shows the URL "https://cran.r-project.org". The page features the R logo on the left and the title "Contributed Packages" at the top. Below the title, there are sections for "Available Packages", "Installation of Packages", and "Package Check Results". The "Available Packages" section states that there are 7826 available packages and provides two links: "Table of available packages, sorted by date of publication" and "Table of available packages, sorted by name". The "Installation of Packages" section explains how to install packages using the R help system and provides a link to the "CRAN Task Views". The "Package Check Results" section mentions that all packages are tested regularly on various operating systems. On the left side of the page, there is a vertical menu with links to "CRAN Mirrors", "What's new?", "Task Views", "Search", "About R", "R Homepage", "The R Journal", "Software", "R Sources", "R Binaries", "Packages", "Other", "Documentation", "Manuals", and "FAQs".

The Comprehensive R Arch x

https://cran.r-project.org



Contributed Packages

Available Packages

Currently, the CRAN package repository features 7826 available packages.

[Table of available packages, sorted by date of publication](#)

[Table of available packages, sorted by name](#)

Installation of Packages

Please type `help("INSTALL")` or `help("install.packages")` in R for information on how to install packages from this repository. The manual [R Installation and Administration](#) (also contained in the R base sources) explains the process in detail.

[CRAN Task Views](#) allow you to browse packages by topic and provide tools to automatically install all packages for special areas of interest. Currently, 33 views are available.

Package Check Results

All packages are tested regularly on machines running [Debian GNU/Linux](#), [Fedora](#), [OS X](#), [Solaris](#) and [Windows](#).

CRAN

- [Mirrors](#)
- [What's new?](#)
- [Task Views](#)
- [Search](#)

About R

- [R Homepage](#)
- [The R Journal](#)

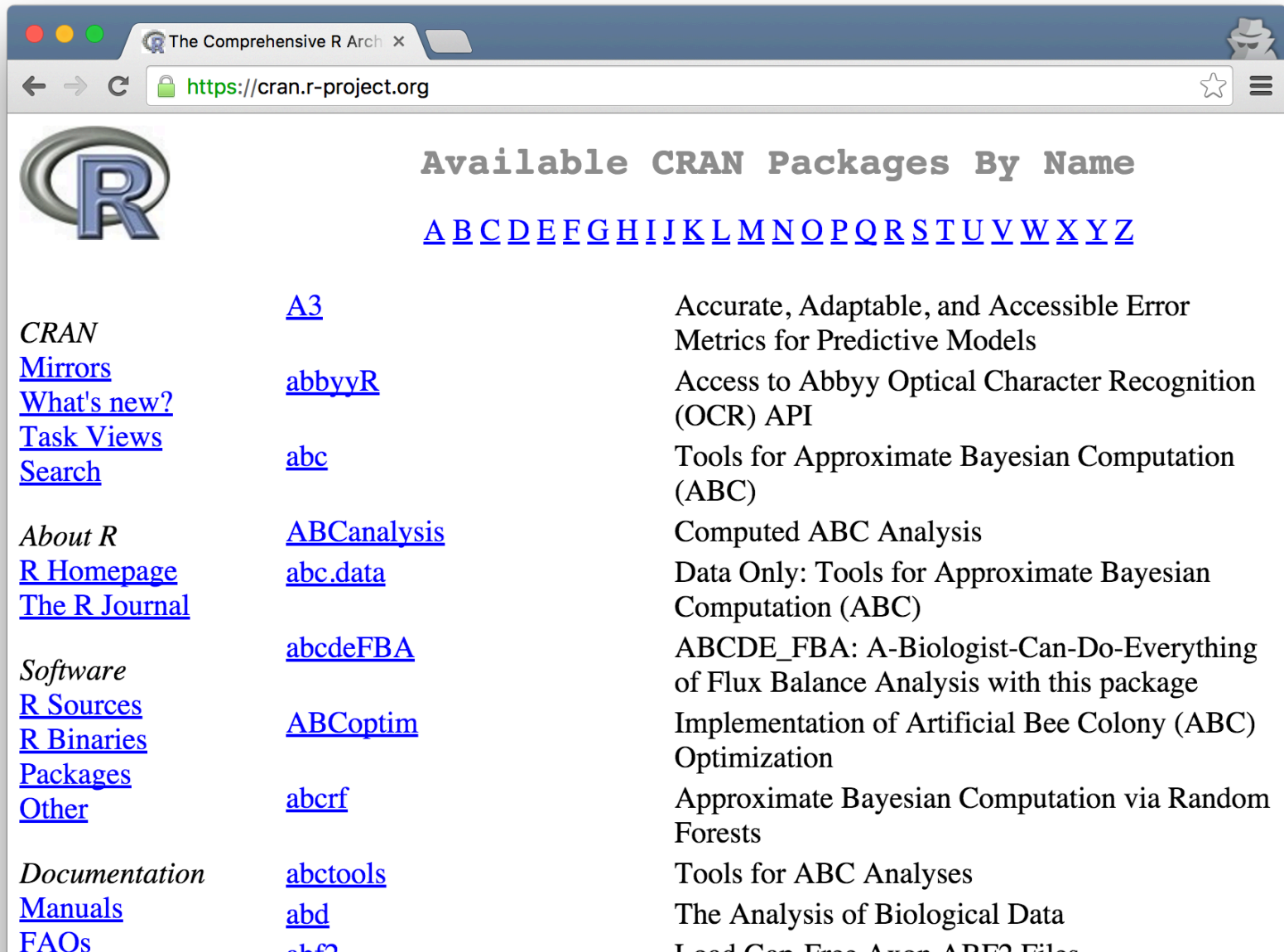
Software

- [R Sources](#)
- [R Binaries](#)
- [Packages](#)
- [Other](#)


Documentation

- [Manuals](#)
- [FAQs](#)

R packages are available on CRAN (Comprehensive R Archive Network)



The screenshot shows a web browser window with the address bar displaying <https://cran.r-project.org>. The page title is "The Comprehensive R Arch x". The main heading is "Available CRAN Packages By Name" with a list of letters [A](#)[B](#)[C](#)[D](#)[E](#)[F](#)[G](#)[H](#)[I](#)[J](#)[K](#)[L](#)[M](#)[N](#)[O](#)[P](#)[Q](#)[R](#)[S](#)[T](#)[U](#)[V](#)[W](#)[X](#)[Y](#)[Z](#) in blue underlined text. On the left, there is a sidebar with links: [CRAN](#), [Mirrors](#), [What's new?](#), [Task Views](#), [Search](#), [About R](#), [R Homepage](#), [The R Journal](#), [Software](#), [R Sources](#), [R Binaries](#), [Packages](#), [Other](#), [Documentation](#), [Manuals](#), and [FAQs](#). The main content area displays a list of packages with their names and descriptions. The packages shown are [A3](#), [abbyyR](#), [abc](#), [ABCanalysis](#), [abc.data](#), [abcdeFBA](#), [ABCoptim](#), [abcrf](#), [abctools](#), [abd](#), and [abf2](#). The descriptions for these packages are: "Accurate, Adaptable, and Accessible Error Metrics for Predictive Models", "Access to Abbyy Optical Character Recognition (OCR) API", "Tools for Approximate Bayesian Computation (ABC)", "Computed ABC Analysis", "Data Only: Tools for Approximate Bayesian Computation (ABC)", "ABCDE_FBA: A-Biologist-Can-Do-Everything of Flux Balance Analysis with this package", "Implementation of Artificial Bee Colony (ABC) Optimization", "Approximate Bayesian Computation via Random Forests", "Tools for ABC Analyses", "The Analysis of Biological Data", and "Load Cap Free Anon ABF2 Files".

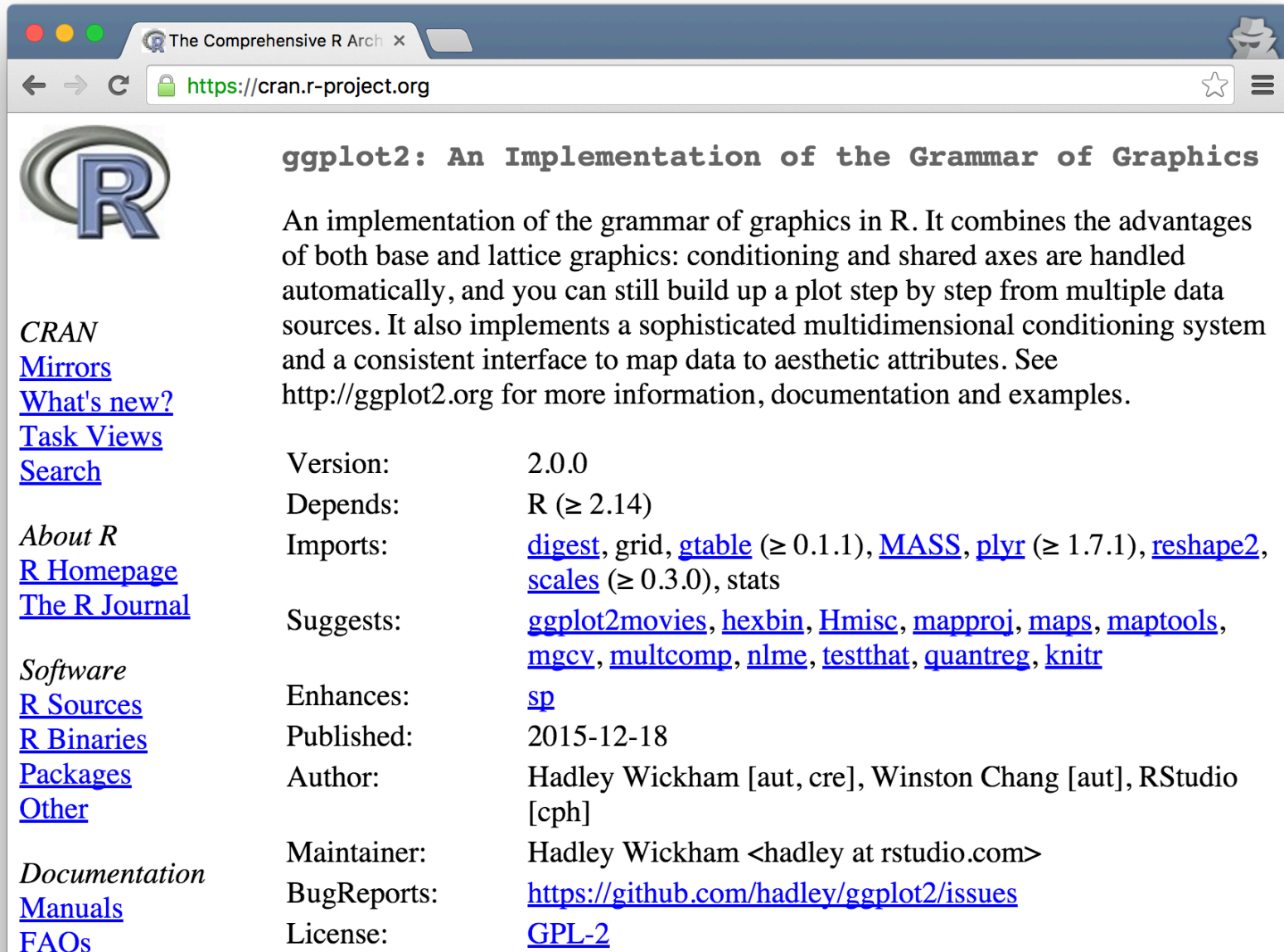


Available CRAN Packages By Name

[A](#)[B](#)[C](#)[D](#)[E](#)[F](#)[G](#)[H](#)[I](#)[J](#)[K](#)[L](#)[M](#)[N](#)[O](#)[P](#)[Q](#)[R](#)[S](#)[T](#)[U](#)[V](#)[W](#)[X](#)[Y](#)[Z](#)

CRAN	A3	Accurate, Adaptable, and Accessible Error Metrics for Predictive Models
Mirrors	abbyyR	Access to Abbyy Optical Character Recognition (OCR) API
What's new?	abc	Tools for Approximate Bayesian Computation (ABC)
Task Views	ABCanalysis	Computed ABC Analysis
Search	abc.data	Data Only: Tools for Approximate Bayesian Computation (ABC)
About R	abcdeFBA	ABCDE_FBA: A-Biologist-Can-Do-Everything of Flux Balance Analysis with this package
R Homepage	ABCoptim	Implementation of Artificial Bee Colony (ABC) Optimization
The R Journal	abcrf	Approximate Bayesian Computation via Random Forests
Software	abctools	Tools for ABC Analyses
R Sources	abd	The Analysis of Biological Data
R Binaries	abf2	Load Cap Free Anon ABF2 Files
Packages		
Other		
Documentation		
Manuals		
FAQs		

We'll be working with the package ggplot2



ggplot2: An Implementation of the Grammar of Graphics

An implementation of the grammar of graphics in R. It combines the advantages of both base and lattice graphics: conditioning and shared axes are handled automatically, and you can still build up a plot step by step from multiple data sources. It also implements a sophisticated multidimensional conditioning system and a consistent interface to map data to aesthetic attributes. See <http://ggplot2.org> for more information, documentation and examples.

Version: 2.0.0

Depends: R (≥ 2.14)

Imports: [digest](#), grid, [gtable](#) (≥ 0.1.1), [MASS](#), [plyr](#) (≥ 1.7.1), [reshape2](#), [scales](#) (≥ 0.3.0), stats

Suggests: [ggplot2movies](#), [hexbin](#), [Hmisc](#), [mapproj](#), [maps](#), [maptools](#), [mgcv](#), [multcomp](#), [nlme](#), [testthat](#), [quantreg](#), [knitr](#)

Enhances: [sp](#)

Published: 2015-12-18

Author: Hadley Wickham [aut, cre], Winston Chang [aut], RStudio [cph]

Maintainer: Hadley Wickham <hadley at rstudio.com>

BugReports: <https://github.com/hadley/ggplot2/issues>

License: [GPL-2](#)

CRAN
[Mirrors](#)
[What's new?](#)
[Task Views](#)
[Search](#)

About R
[R Homepage](#)
[The R Journal](#)

Software
[R Sources](#)
[R Binaries](#)
[Packages](#)
[Other](#)

Documentation
[Manuals](#)
[FAQs](#)

You can install this package using `install.packages()` in RStudio

Console ~/

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> `install.packages("ggplot2")`

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload Upload	Total	Spent	Left	Speed
0	0	0	0	0	--:--:--	--:--:--	0 38 1932k
38	751k	0	0	1529k	0 0:00:01	--:--:--	0:00:01 1527k100 1932k
0	0	2918k	0	--:--:--	--:--:--	--:--:--	2918k

The downloaded binary packages are in
/var/folders/q8/wptgtbdn1pz0cfgrz39gq00m0000gn/T//RtmpvQgw1u/downloaded_packages

> |

ggplot2: A grammar of graphics

Traditional plotting: You **are** a painter

- Manually place individual graphical elements

ggplot2: You **employ** a painter

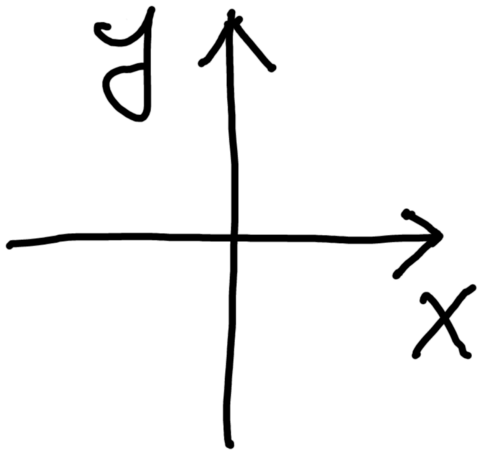
- Describe conceptually how data should be visualized

Most confusing key concept: aesthetic mapping

Maps data values to visual elements of the plot

A few examples of aesthetics

position



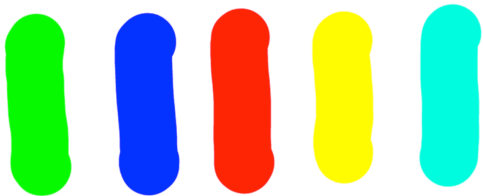
shape



size



color



angle



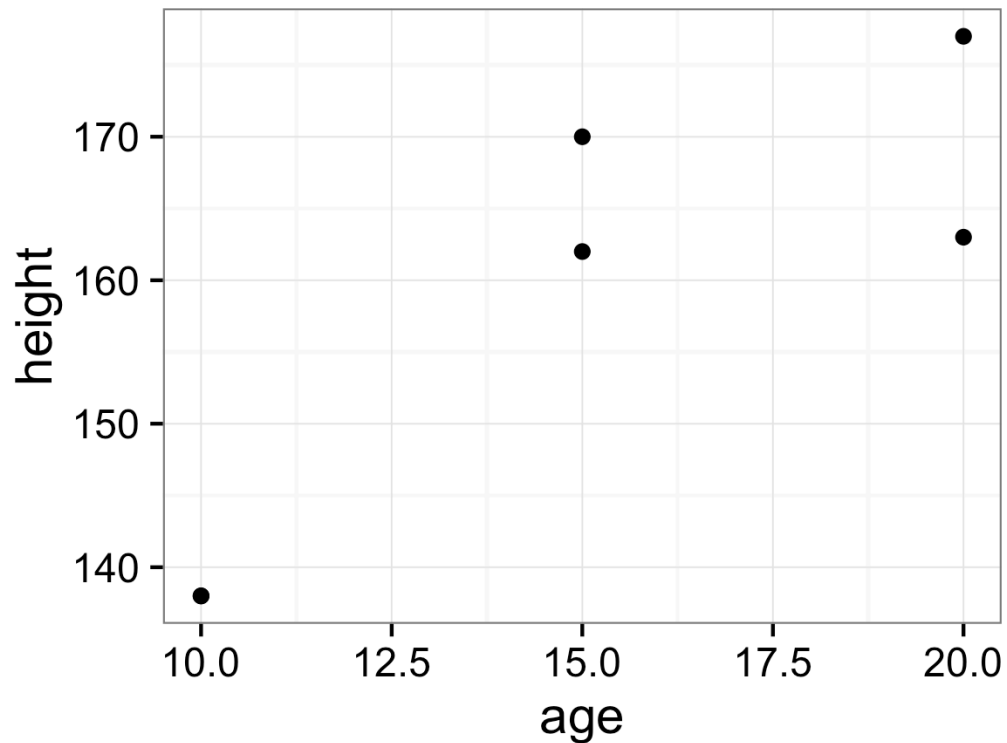
Let's go over a simple example: mean height and weight of boys/girls ages 10-20

age (yrs)	height (cm)	weight (kg)	sex
10	138	32	M
15	170	56	M
20	177	71	M
10	138	33	F
15	162	52	F
20	163	53	F

Data from: <http://www.cdc.gov/growthcharts/>

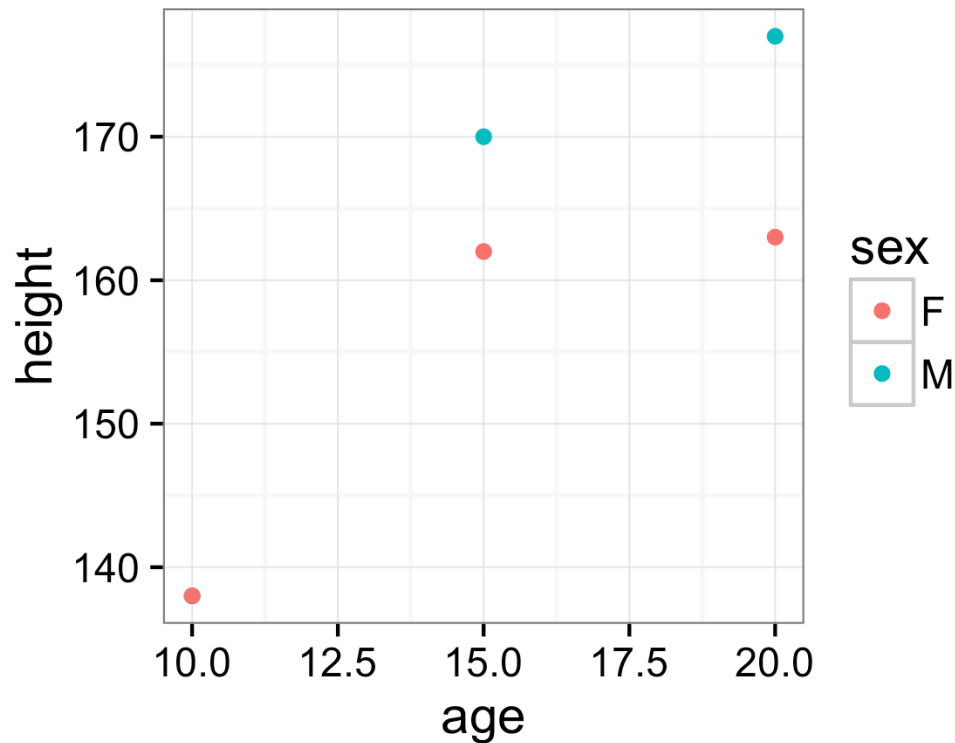
Map age to x, height to y, visualize using points

```
ggplot(data, aes(x=age, y=height)) +  
  geom_point()
```



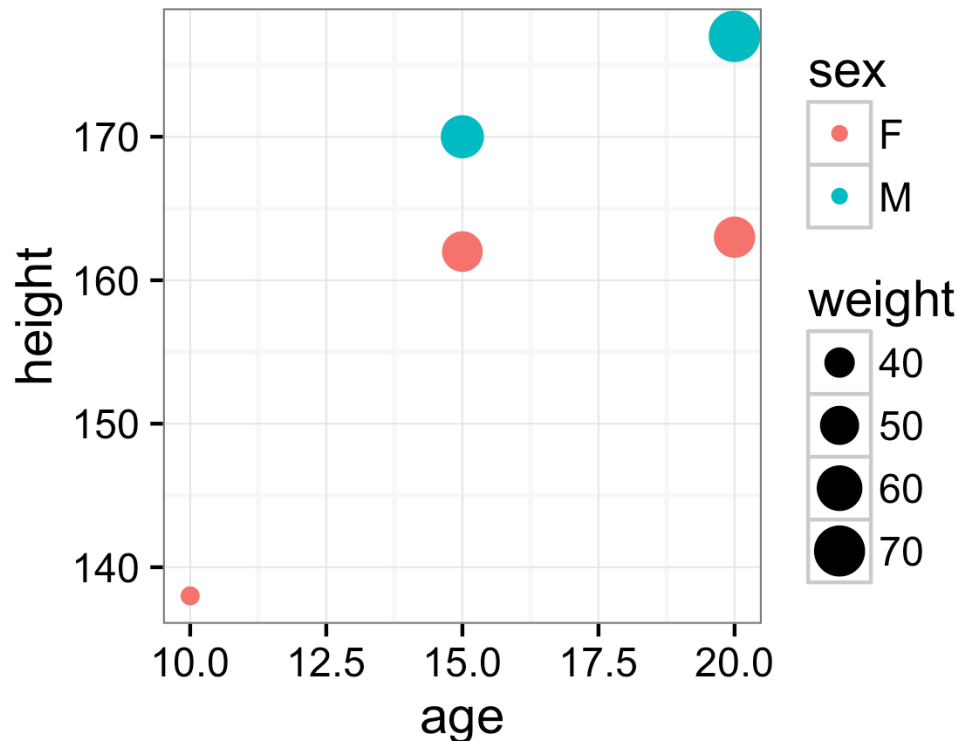
Let's color the points by sex

```
ggplot(data, aes(x=age, y=height,  
                 color=sex)) + geom_point()
```



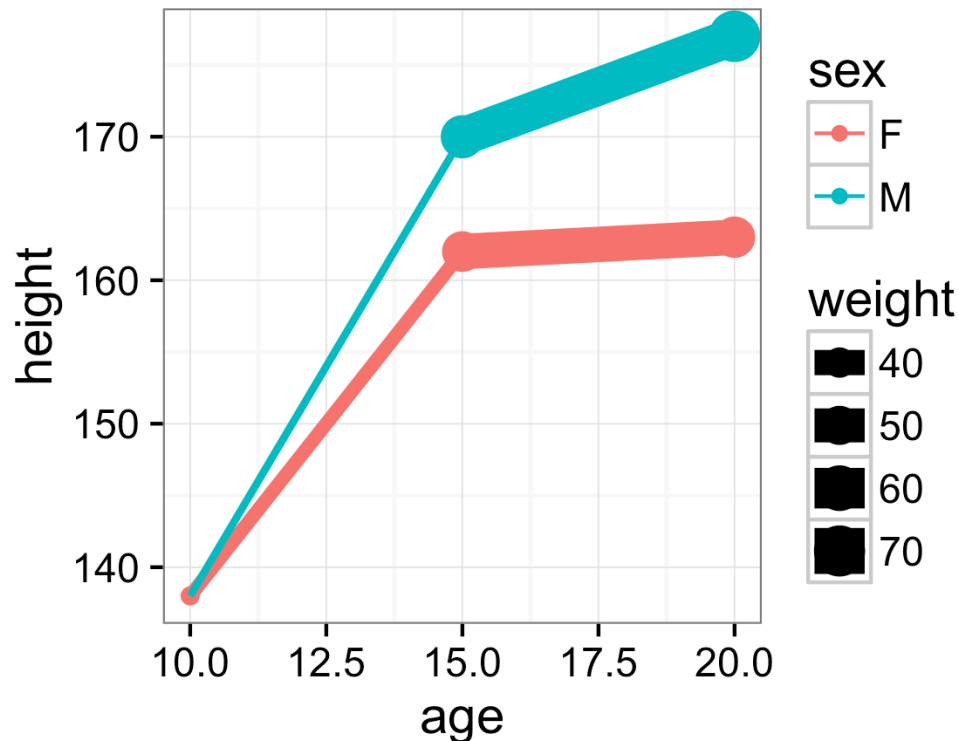
And change point size by weight

```
ggplot(data, aes(x=age, y=height,  
  color=sex, size=weight)) + geom_point()
```



And connect the points with lines

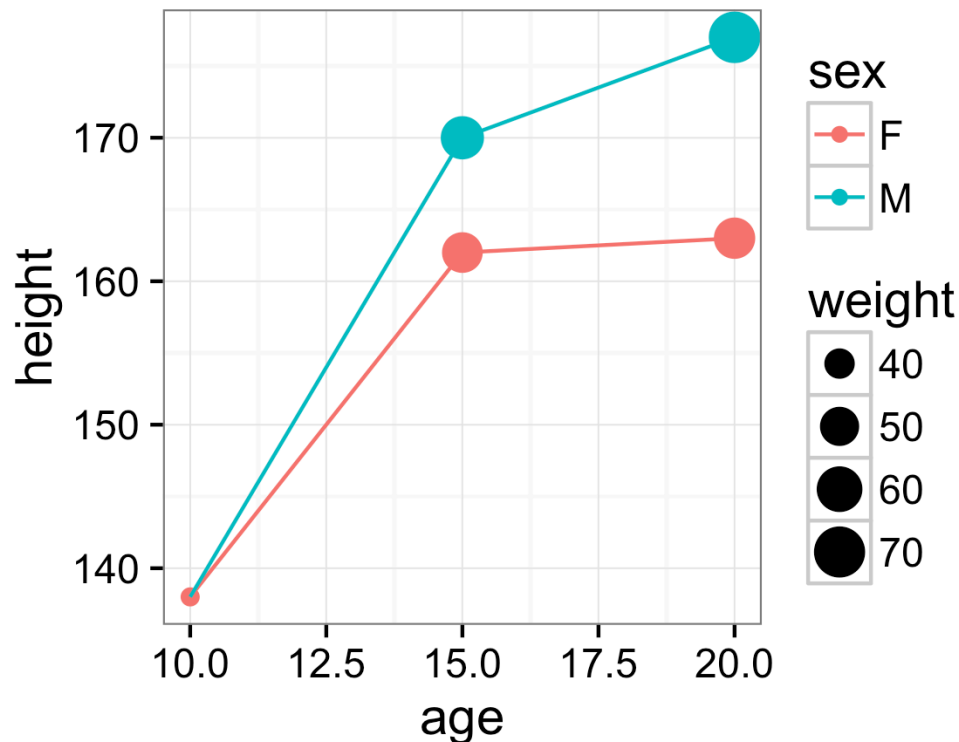
```
ggplot(data, aes(x=age, y=height,  
  color=sex, size=weight)) +  
  geom_point() + geom_line()
```



Oops!

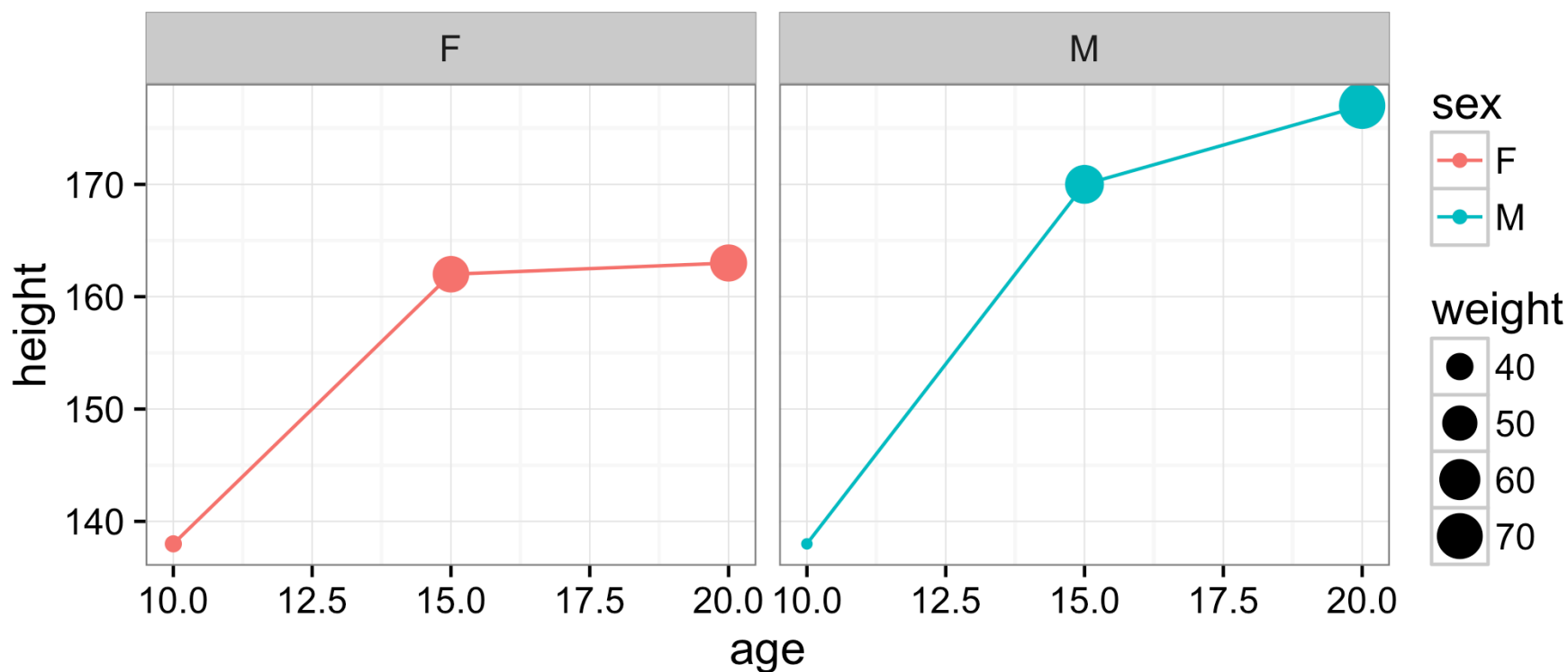
The weight-to-size mapping should only be applied to points

```
ggplot(data, aes(x=age, y=height,  
  color=sex)) + geom_point(aes(size=weight)) +  
  geom_line()
```



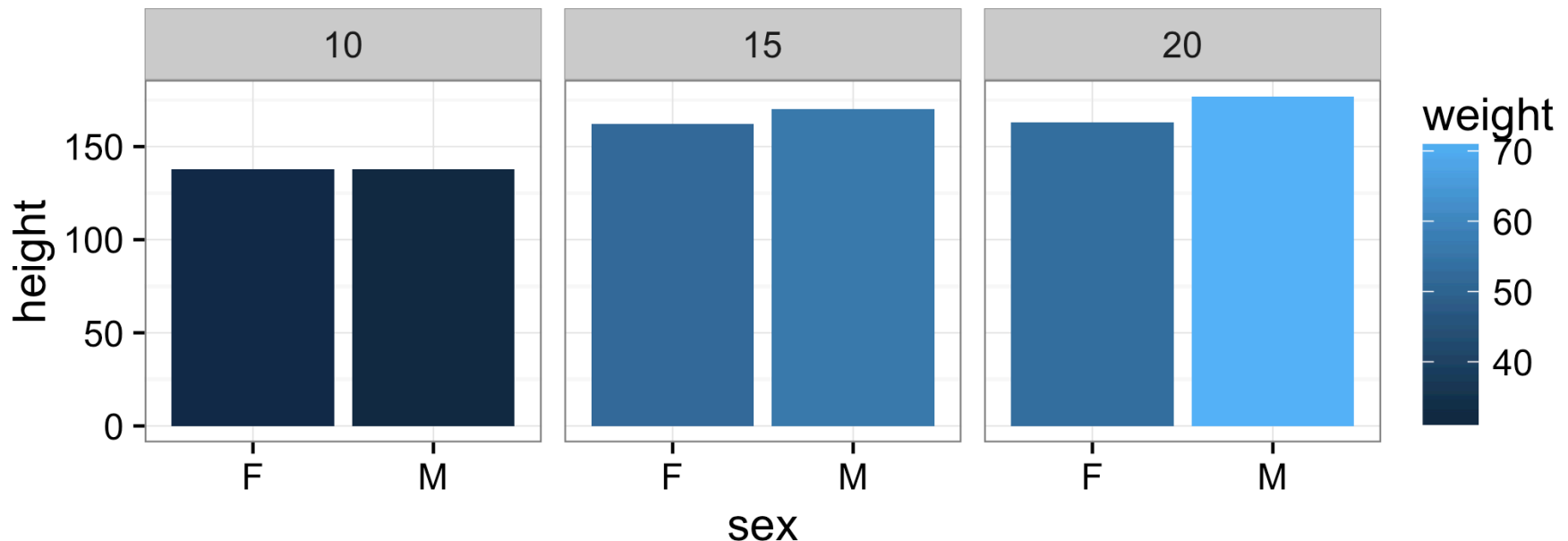
We can also make side-by-side plots (called facets)

```
ggplot(data, aes(x=age, y=height,  
  color=sex)) + geom_point(aes(size=weight)) +  
  geom_line() + facet_wrap(~sex)
```



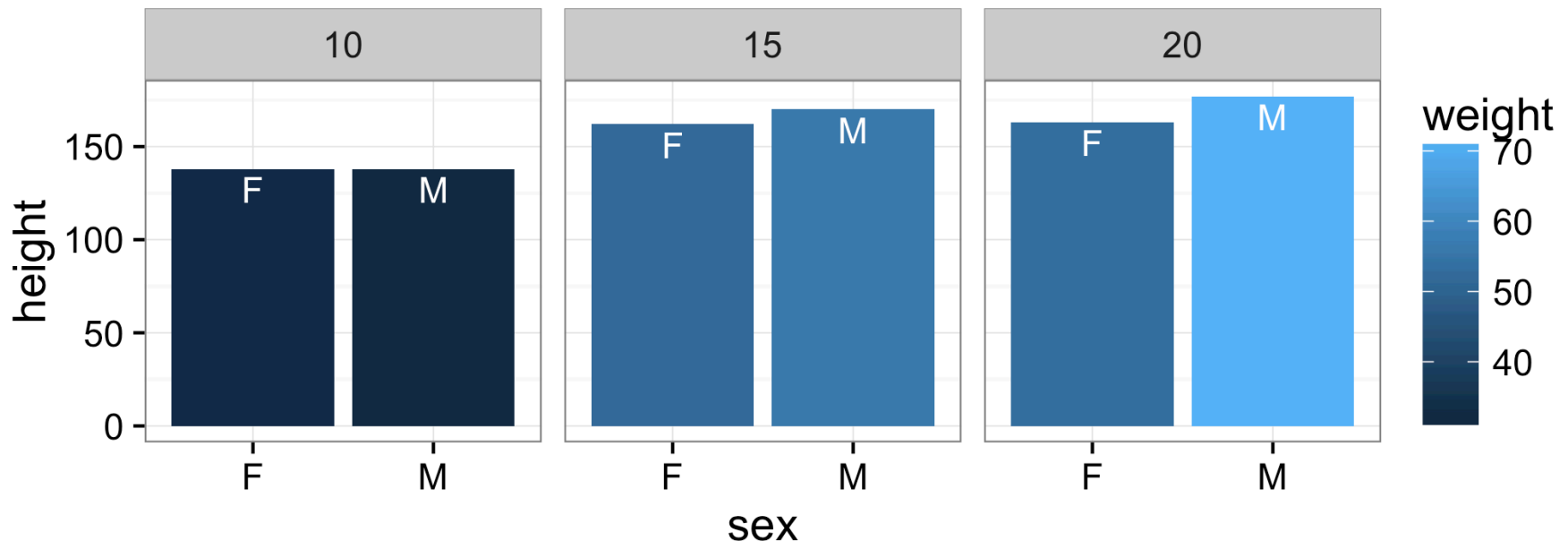
Now let's facet by age, color by weight, and use bars (columns) to plot height

```
ggplot(data, aes(x=sex, y=height, fill=weight)) +  
  geom_col() + facet_wrap(~age)
```



Let's plot the sex also at the top of the bar

```
ggplot(data, aes(x=sex, y=height, fill=weight)) +  
  geom_col() +  
  geom_text(aes(label=sex), vjust=1.3, color='white') +  
  facet_wrap(~age)
```



All the geom's with all their options are described on the ggplot2 web page

<http://ggplot2.tidyverse.org/reference/>