Introduction to scatterplots with ggplot2

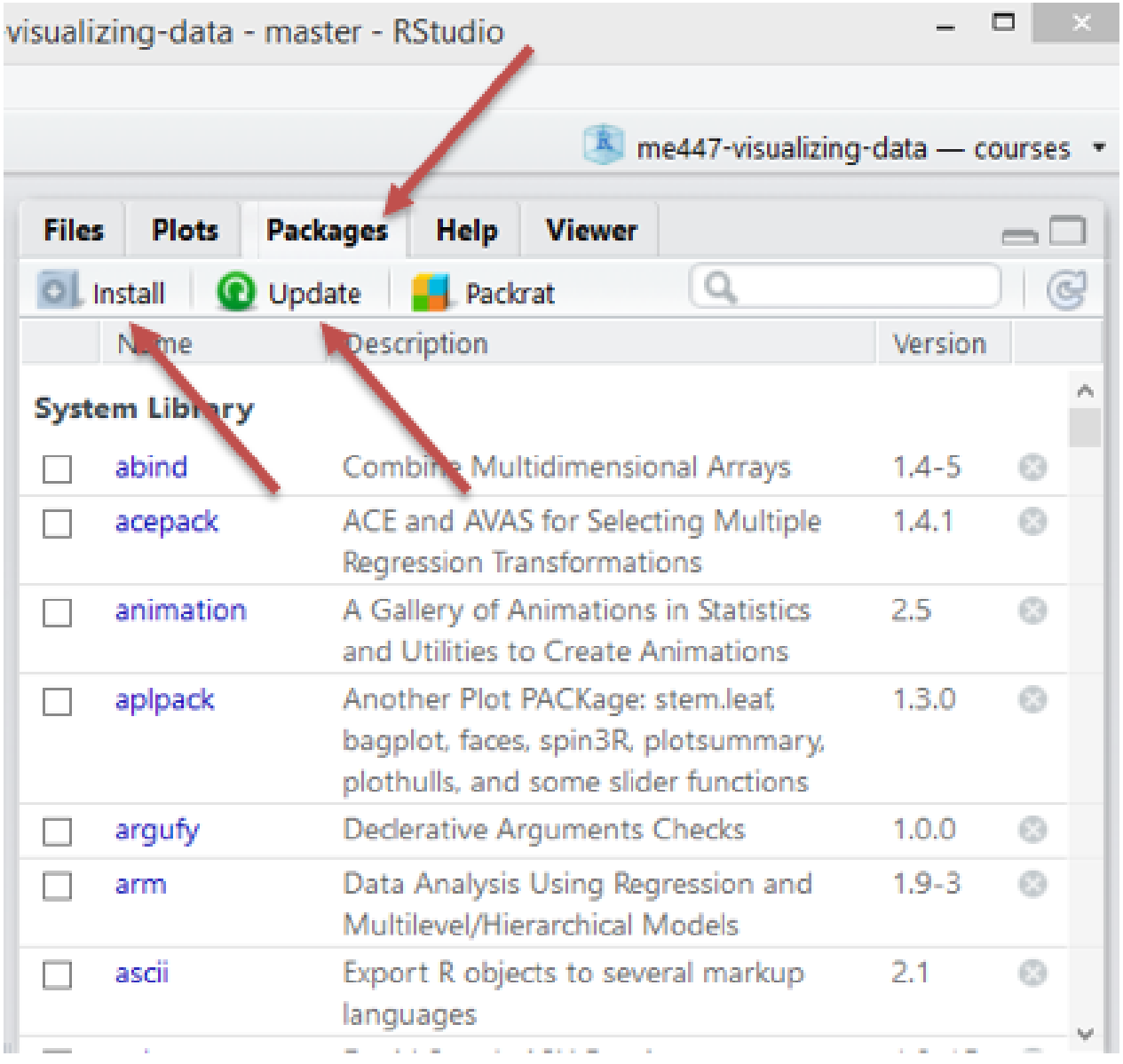
Richard Layton

2017-09-04

R package vocabulary

Term	Definition
package	An R 'app', the basic unit of shareable code, a collection of R functions, data, and code
library	The directory on your computer where packages are stored, e.g., C:/R/library
CRAN	Online home of 11,340 R packages (2017-09-02), the Comprehensive R Archive Network
install	Download a package, save it in your library
update	Bring the package in your library up to date
load	To use a package in an R session, you load the package using the library() command.

Managing packages in RStudio



The graphics package we'll use most often is ggplot2

The Tidyverse: R packages for data science

When you get a chance, install the `tidyverse` package

`tidyverse` includes `ggplot2` and many other packages we will find useful

- `ggplot2` is the package
- `ggplot()` is the function

The arguments of `ggplot()`

Arguments	Definition
data	a data frame
aesthetic	visual properties: color, shape, etc.
geom	geometric shape: points, lines, rectangles, etc.
scale	map data units to computer units
stat	summarize or transform data, e.g., a regression
facet	create small multiples

Example

Getting the example started

- I installed the `VGAMdata` package from CRAN
- Open a new R script
- Load the package into my workspace using `library()`

```
library(VGAMdata)
```

- The 2012 summer Olympic data about individual athletes is accessed using the `data()` function

```
data("oly12")
```

What objects?

What objects are in my workspace?

```
ls()
```

```
## [1] "oly12"
```

What class of object is it?

```
class(oly12)
```

```
## [1] "data.frame"
```

Good. It's a data frame.

Examine the data frame

What variables (columns) do I have?

```
names(oly12)
```

```
##  [1] "Name"    "Country" "Age"     "Height"
##  [5] "Weight"  "Sex"     "DOB"     "PlaceOB"
##  [9] "Gold"    "Silver"  "Bronze"  "Total"
## [13] "Sport"   "Event"
```

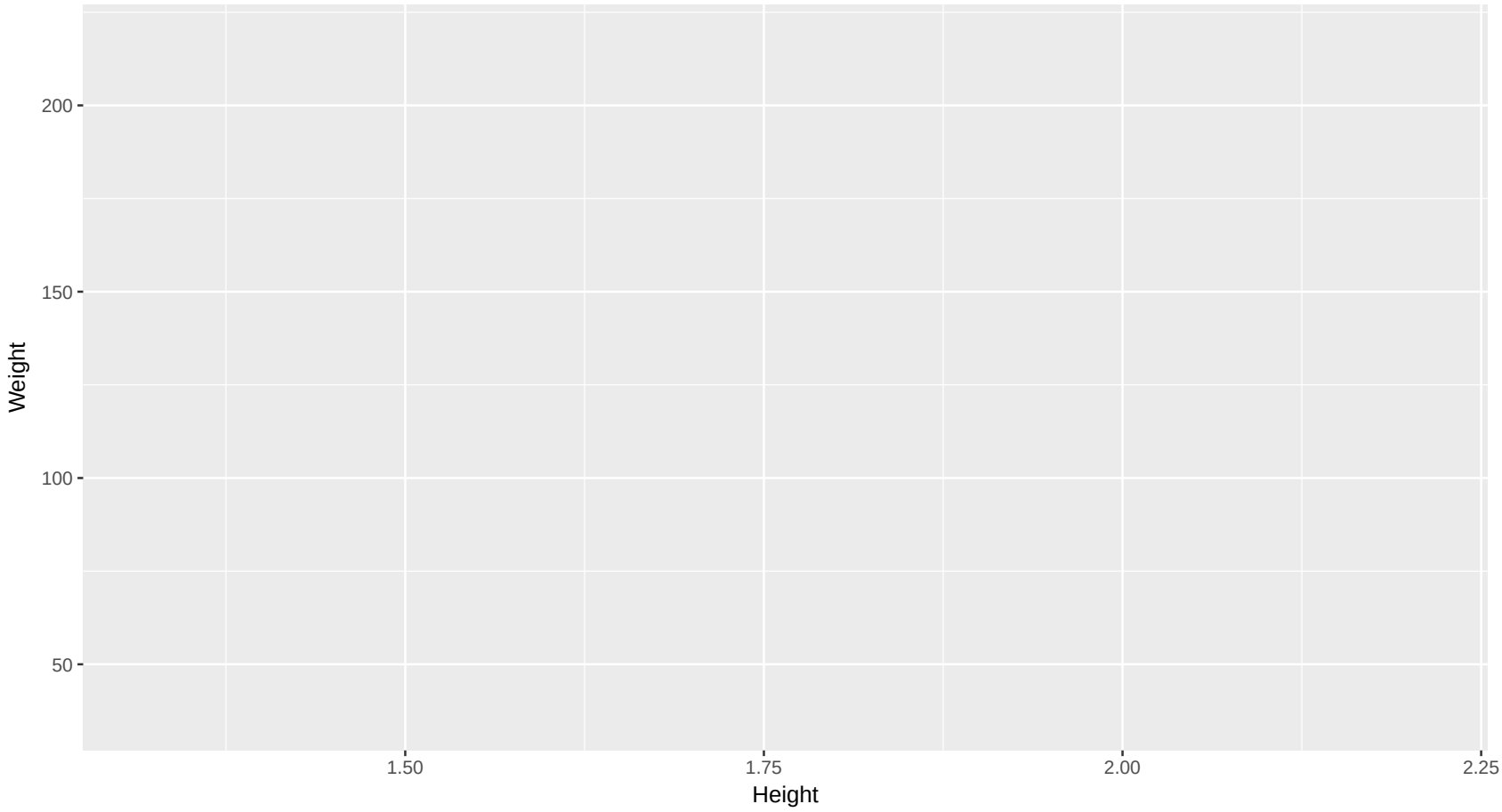
To see the structure of the data frame in more detail, use

```
glimpse(oly12)
```

```
## Observations: 10,384
## Variables: 14
## $ Name    <fctr> Lamusi A, A G Kruger, Jamale A...
## $ Country <fctr> People's Republic of China, Un...
## $ Age     <int> 23, 33, 30, 24, 26, 27, 30, 23,...
## $ Height  <dbl> 1.70, 1.93, 1.87, NA, 1.78, 1.8...
## $ Weight  <int> 60, 125, 76, NA, 85, 80, 73, 75...
## $ Sex     <fctr> M, M, M, M, F, M, F, M, M, M, ...
## $ DOB     <date> 1989-02-06, NA, NA, 1988-09-02...
## $ PlaceOB <fctr> NEIMONGGOL (CHN), Sheldon (USA...
## $ Gold    <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ Silver  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ Bronze  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ Total   <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ Sport   <fctr> Judo, Athletics, Athletics, Bo...
## $ Event   <fctr> Men's -60kg, Men's Hammer Thro...
```

Data and aesthetics

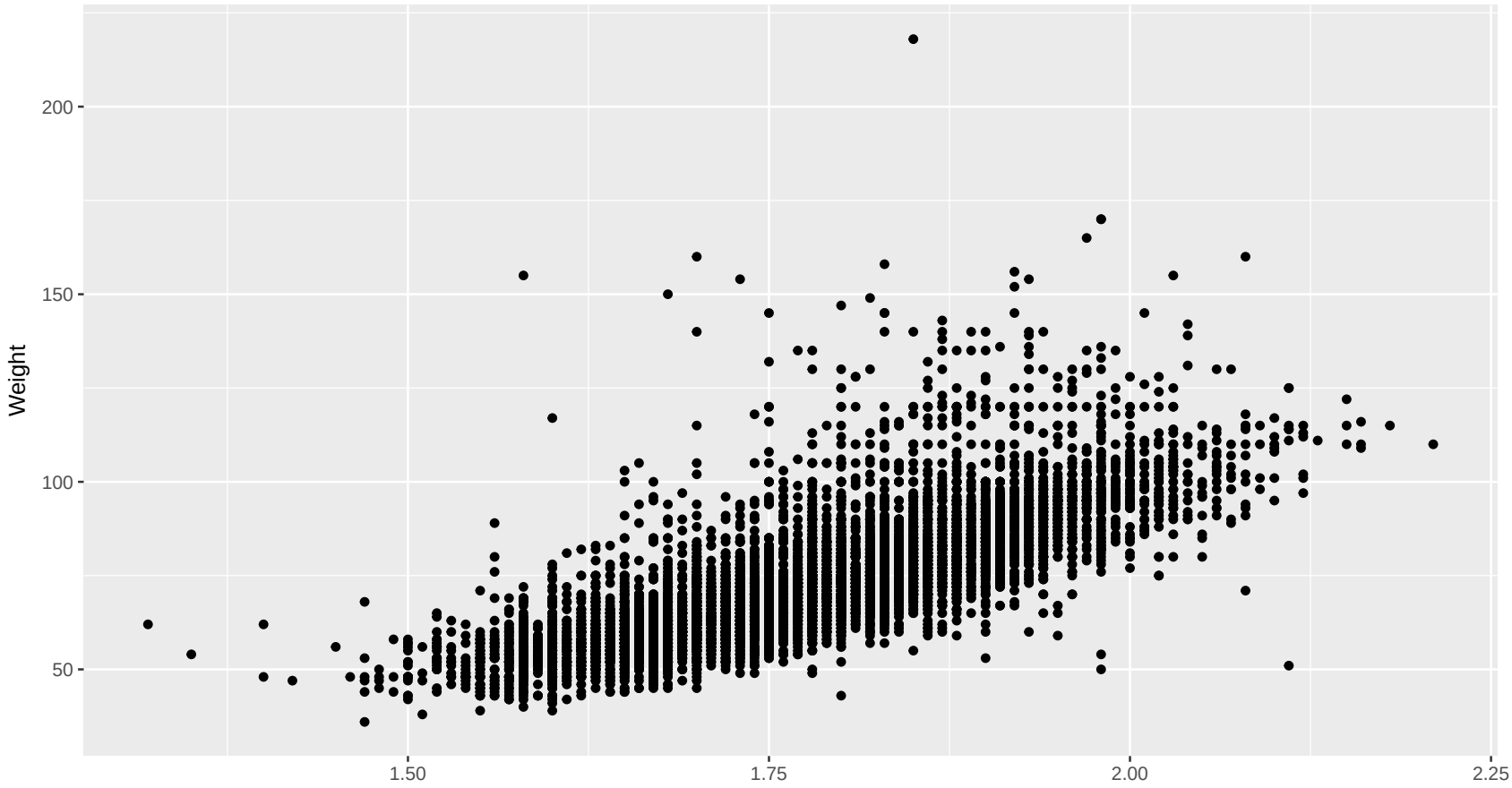
```
ggplot(data = oly12, aes(x = Height, y = Weight))
```



Add points

```
ggplot(data = oly12, aes(x = Height, y = Weight)) +
    geom_point()
```

```
## Warning: Removed 1346 rows containing missing values
## (geom_point).
```



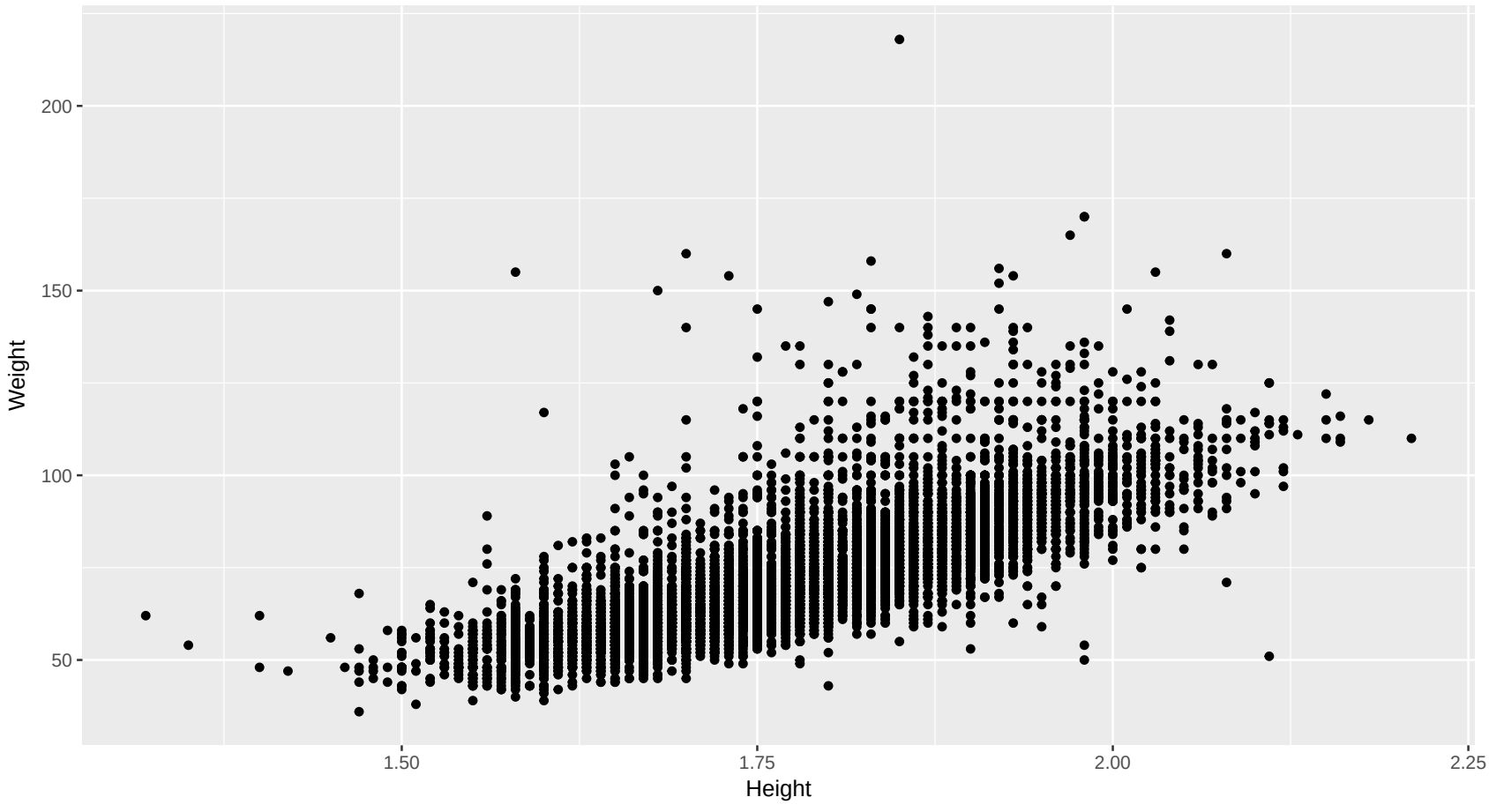
Select 4 columns and filter for complete cases only

```
oly12_sub <- oly12 %>%
    select(Height, Weight, Sex, Sport) %>%
    filter(complete.cases(.))
glimpse(oly12_sub)
```

```
## Observations: 9,038
## Variables: 4
## $ Height <dbl> 1.70, 1.93, 1.87, 1.78, 1.82, 1....
## $ Weight <int> 60, 125, 76, 85, 80, 73, 75, 80,...
## $ Sex    <fctr> M, M, M, F, M, F, M, M, F, F, M...
## $ Sport  <fctr> Judo, Athletics, Athletics, Ath...
```

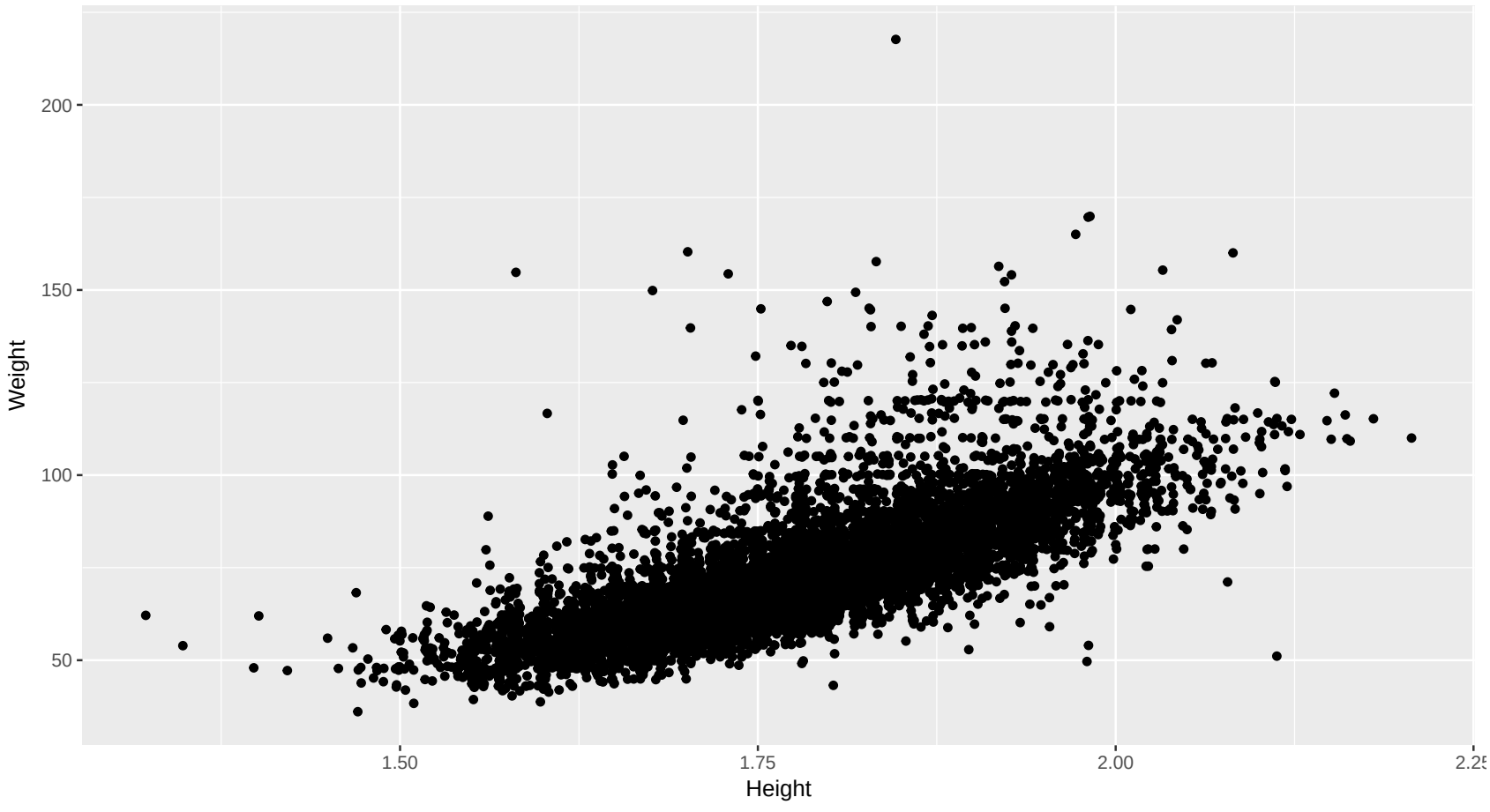
Graph it again (no warning message)

```
ggplot(data = oly12_sub, aes(x = Height, y = Weight)) +
    geom_point()
```



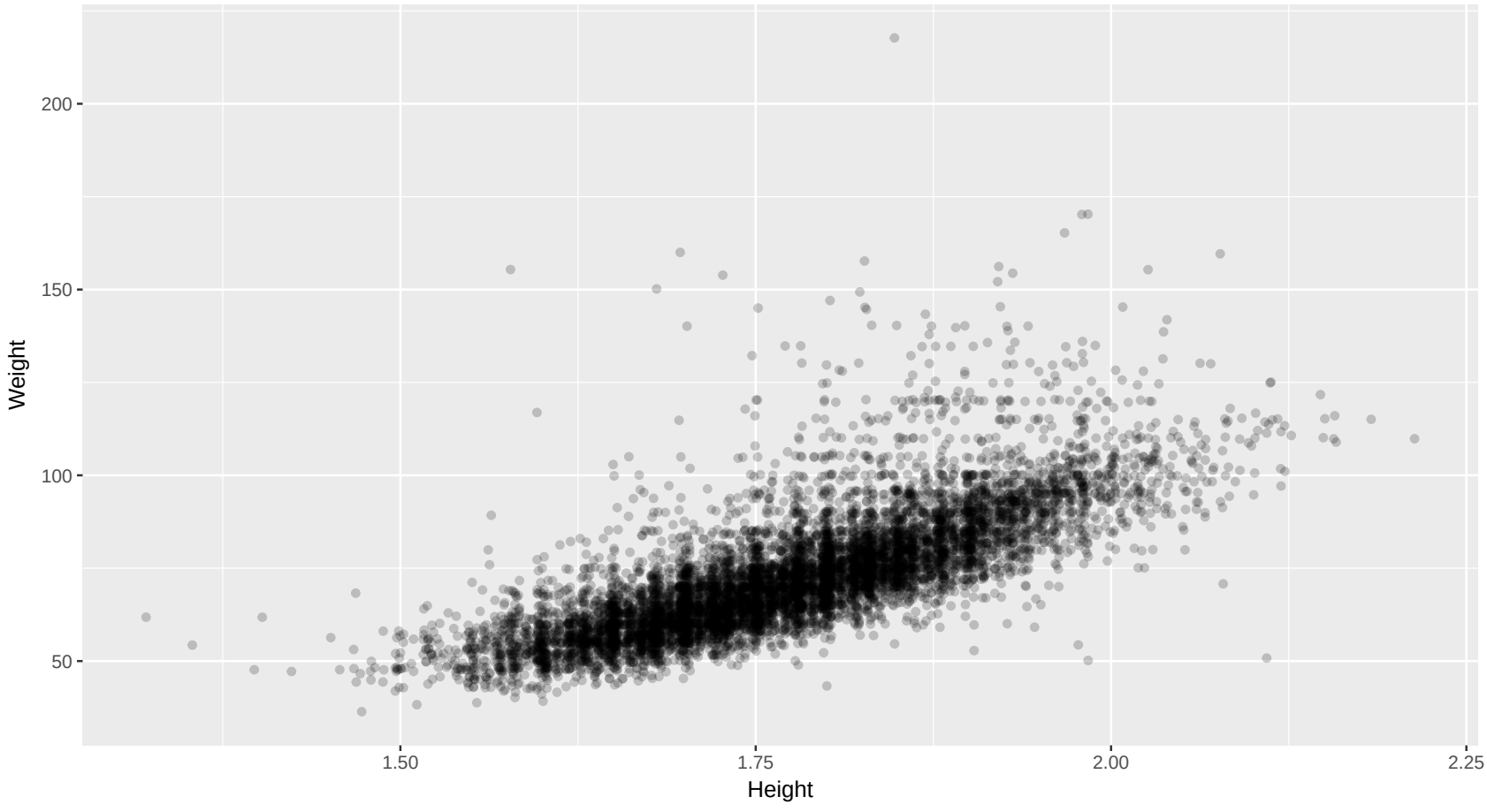
Jitter the data markers to reduce overprinting

```
ggplot(data = oly12_sub, aes(x = Height, y = Weight)) +
    geom_jitter()
```



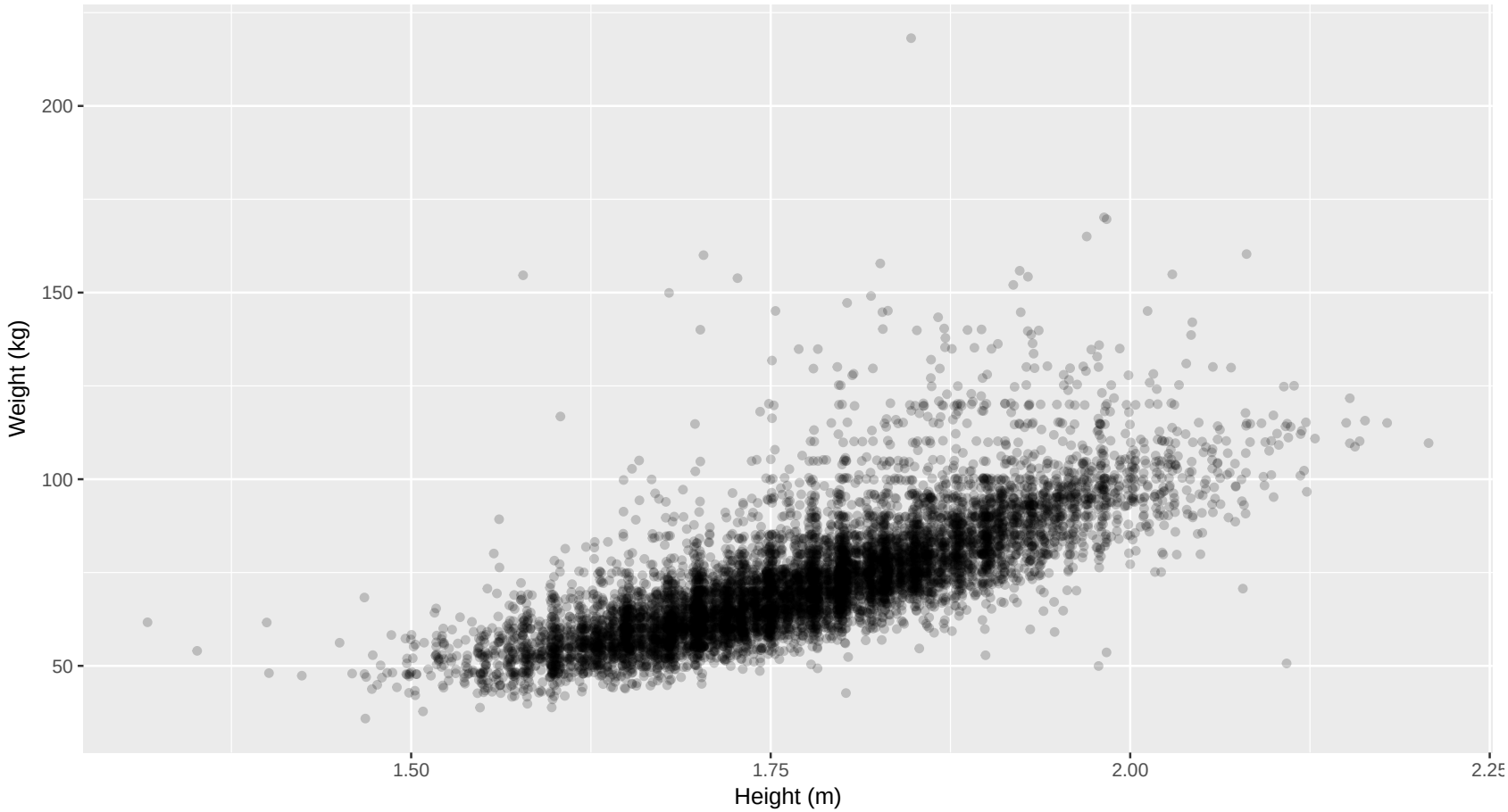
Edit the transparency of the data markers

```
ggplot(data = oly12_sub, aes(x = Height, y = Weight)) +
    geom_jitter(alpha = 0.2)
```



Edit the axis labels

```
ggplot(data = oly12_sub, aes(x = Height, y = Weight)) +
    geom_jitter(alpha = 0.2) +
    labs(x = "Height (m)", y = "Weight (kg)")
```



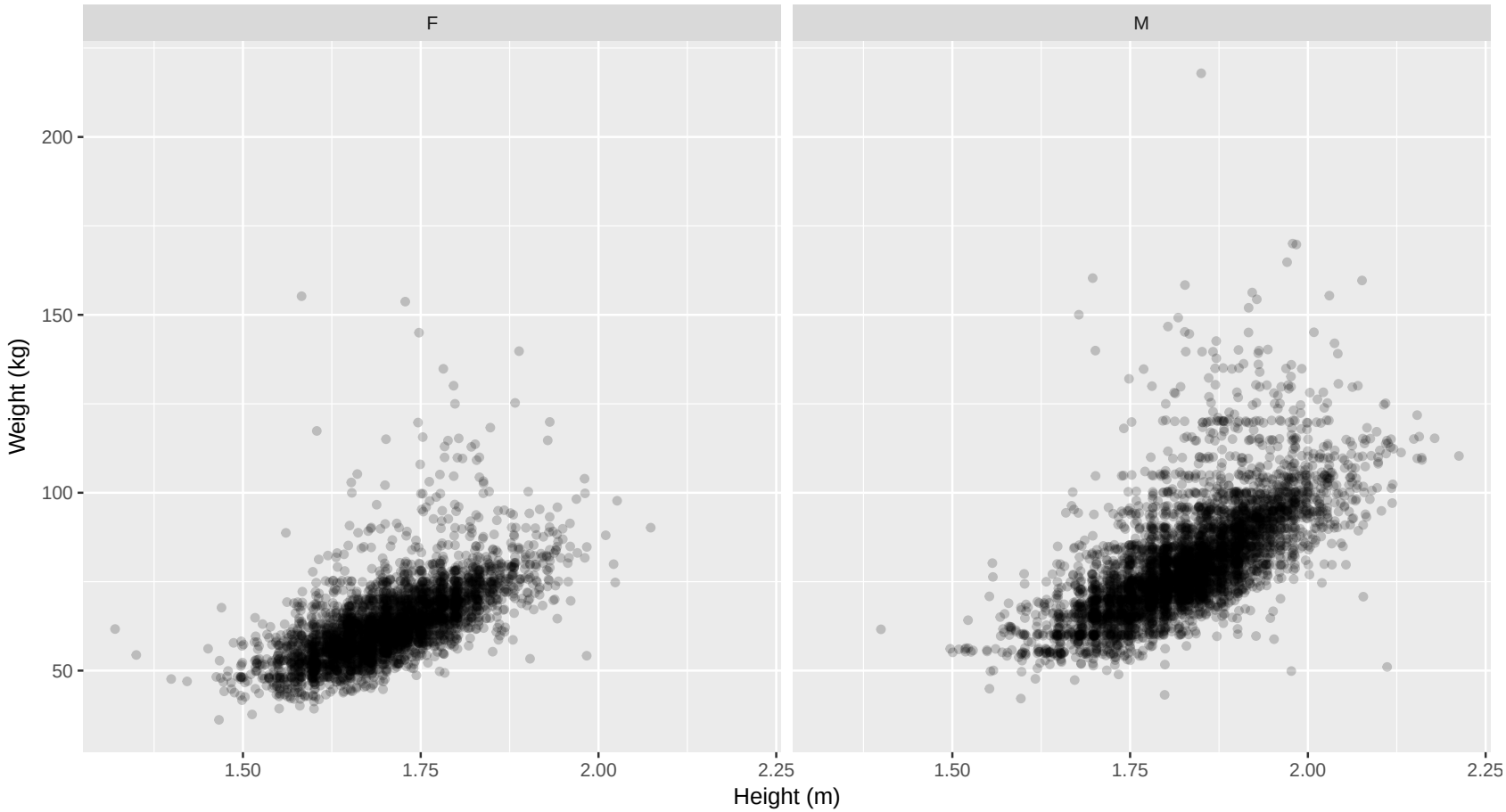
Bivariate? Trivariate?

- A scatterplot displays two continuous variables
- How can a scatterplot be tri-variate?

```
## Observations: 9,038
## Variables: 4
## $ Height <dbl> 1.70, 1.93, 1.87, 1.78, 1.82, 1....
## $ Weight <int> 60, 125, 76, 85, 80, 73, 75, 80,...
## $ Sex    <fctr> M, M, M, F, M, F, M, M, F, F, M...
## $ Sport  <fctr> Judo, Athletics, Athletics, Ath...
```

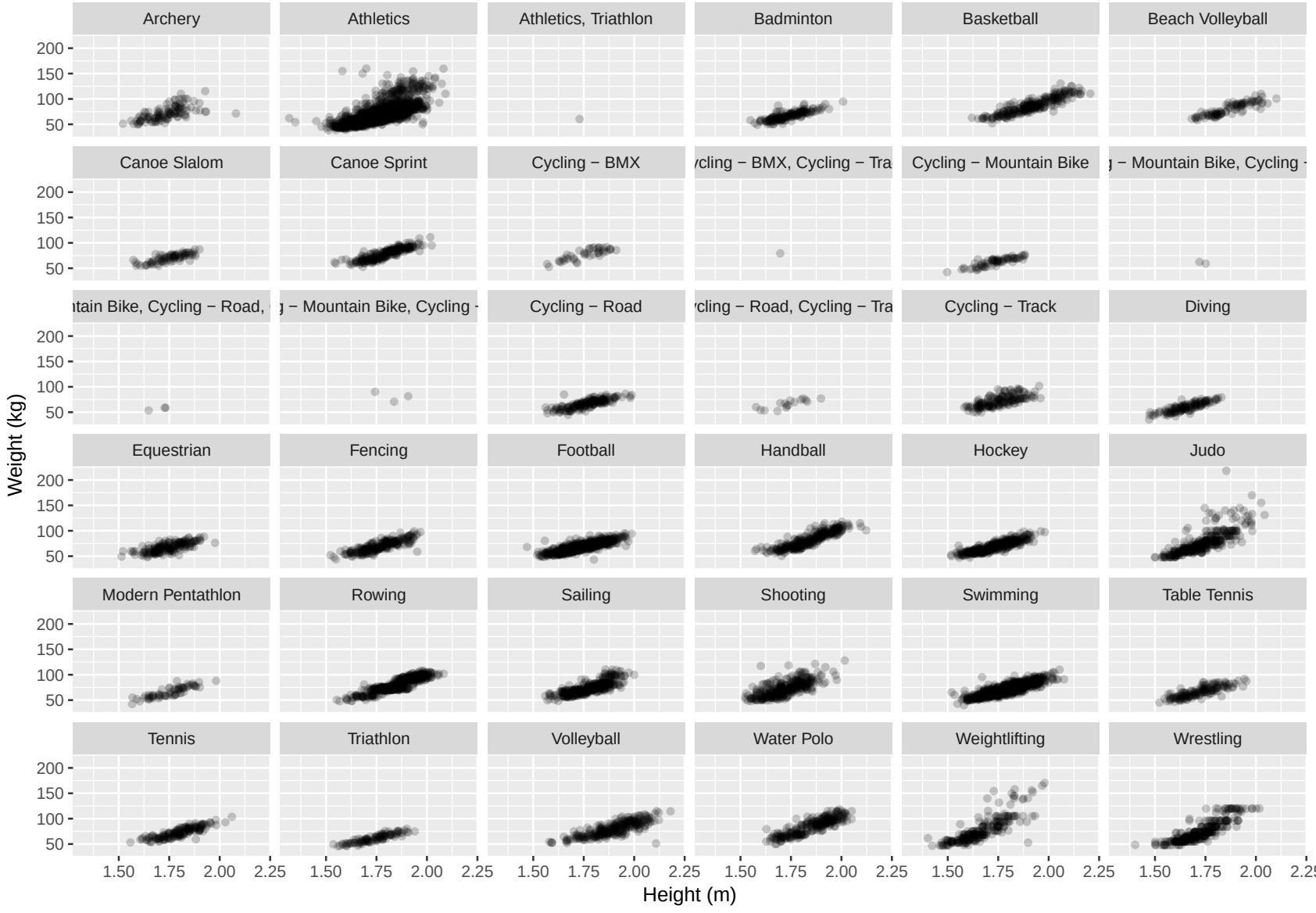
Add a facet by Sex

```
ggplot(data = oly12_sub, aes(x = Height, y = Weight)) +
    geom_jitter(alpha = 0.2) +
    labs(x = "Height (m)", y = "Weight (kg)") +
    facet_wrap(~Sex)
```



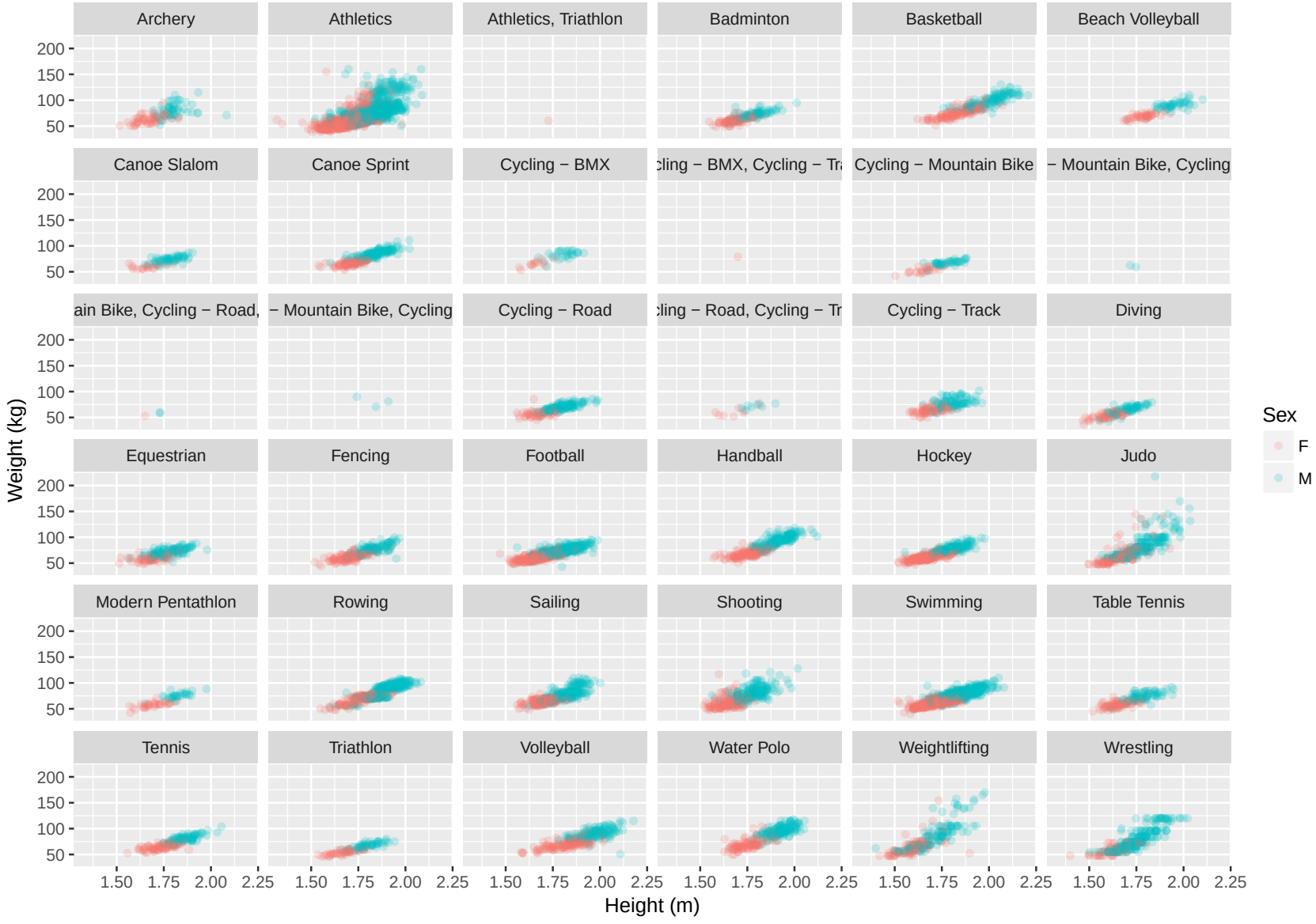
Or by Sport, creating a small-multiples graph

```
ggplot(data = oly12_sub, aes(x = Height, y = Weight)) +
    geom_jitter(alpha = 0.2) +
    labs(x = "Height (m)", y = "Weight (kg)") +
    facet_wrap(~Sport)
```


Archery
Athletics
Athletics, Triathlon
Badminton
Basketball
Beach Volleyball
Canoe Slalom
Canoe Sprint
Cycling − BMX
Cycling − Mountain Bike
Cycling − Road
Cycling − Track
Diving
Equestrian
Fencing
Football
Handball
Hockey
Judo
Modern Pentathlon
Rowing
Sailing
Shooting
Swimming
Table Tennis
Tennis
Triathlon
Volleyball
Water Polo
Weightlifting
Wrestling
Weight (kg)
50
100
150
200
Height (m)
1.50
1.75
2.00
2.25

We can include a fourth variable using a color aesthetic

```
ggplot(data = oly12_sub,
       aes(x = Height, y = Weight, col = Sex)) +
    geom_jitter(alpha = 0.2) +
    labs(x = "Height (m)", y = "Weight (kg)") +
    facet_wrap(~Sport)
```


Archery
Athletics
Athletics, Triathlon
Badminton
Basketball
Beach Volleyball
Canoe Slalom
Canoe Sprint
Cycling − BMX
Cycling − Mountain Bike
Cycling − Track
Diving
Equestrian
Fencing
Football
Handball
Hockey
Judo
Modern Pentathlon
Rowing
Sailing
Shooting
Swimming
Table Tennis
Tennis
Triathlon
Volleyball
Water Polo
Weightlifting
Wrestling
Weight (kg)
200
150
100
50
Height (m)
1.50
1.75
2.00
2.25
Sex
F
M

Review

Packages

- tidyverse
- VGAMdata

Tidyverse functions

- ggplot()
- glimpse()
- select()
- filter()

R functions

- library()
- data()
- ls()
- class()
- names()
- print()
- complete.cases()

ggplot arguments

- aes()
- geom_point()
- geom_jitter()
- labs()
- facet_wrap()