

# Intro to R

Data Visualization with Esquisse

# Esquisse Package

```
# install.packages("esquisse")  
library(esquisse)
```

# Esquisse Package

The [esquisse package](#) is helpful for getting used to creating plots in R.

It is an interactive tool to help you in RStudio.

It's super **nifty**!



# Starting a plot

Using the `esquisser()` function you can start creating a plot for a `data.frame` or `tibble`. That's it!

```
esquisser(mtcars)
```

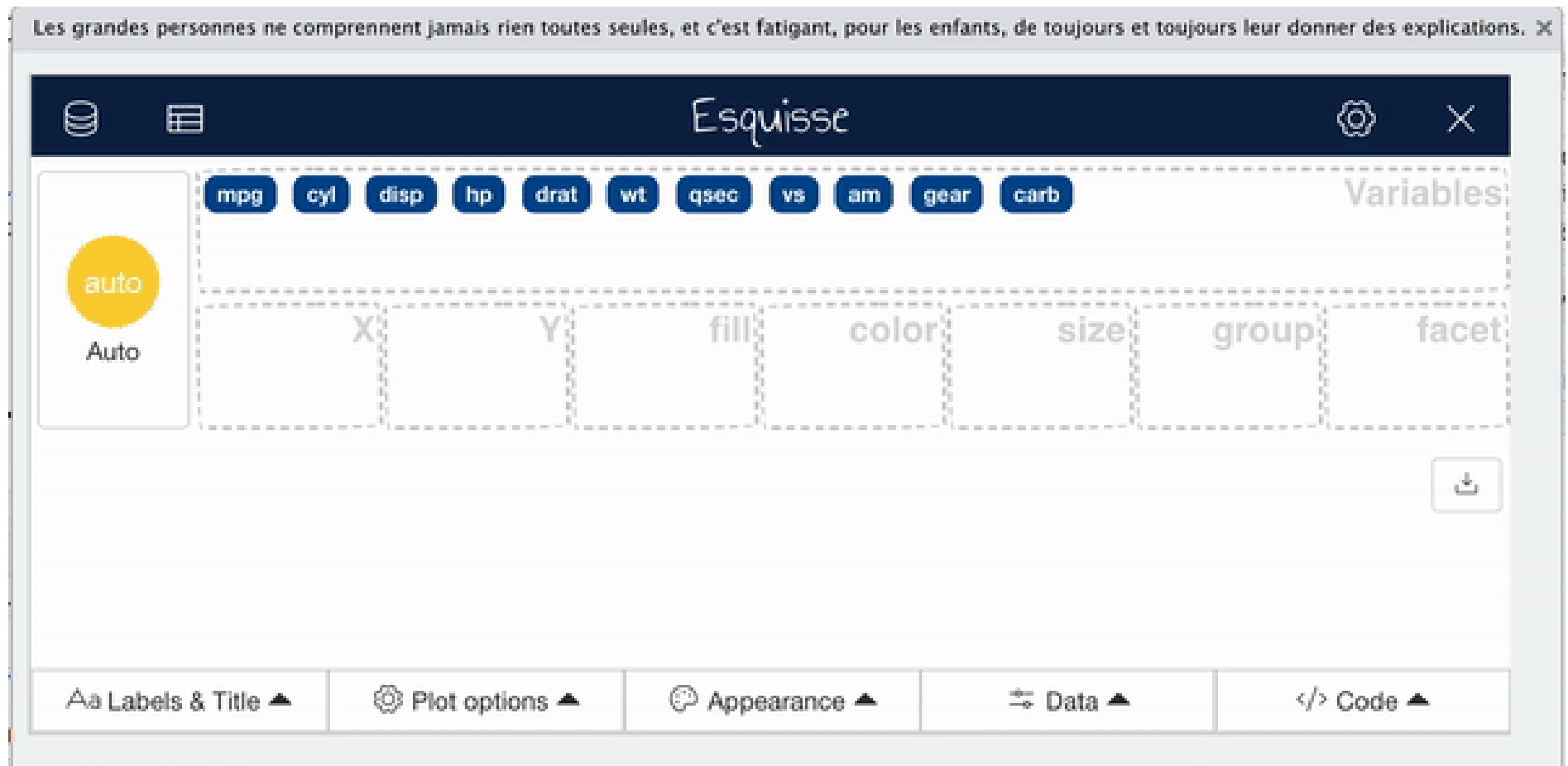


## Show the plot in the browser

```
esquisse::esquisser(iris, viewer = "browser")
```

# Select Variables

To select variables you can drag and drop variables to the respective axis that you would like the variable to be plotted on.



## Find code

To select variables you can drag and drop variables to the respective axis that you would like the variable to be plotted on.



# Change plot type

esquisse automatically assumes a plot type, but you might want to change this.





## Add Facets

Facets create multiple plots based on the different values of a variable.



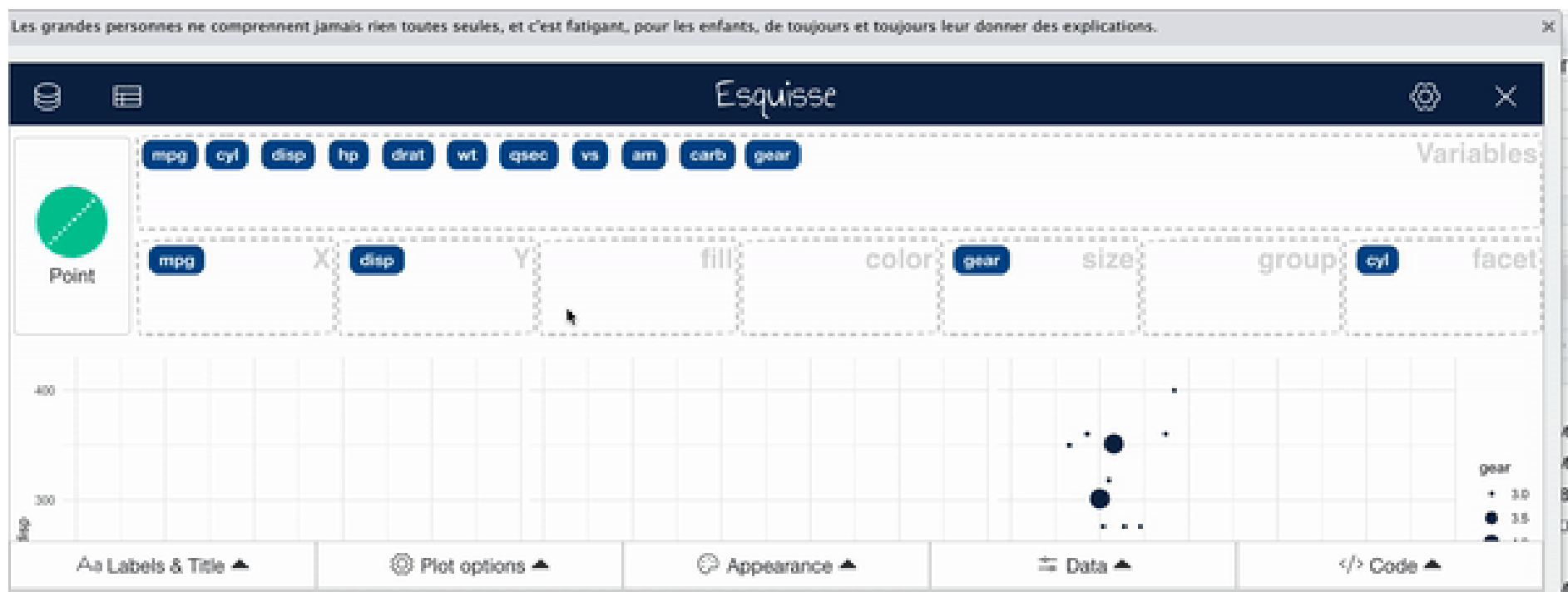
# Add size

Sometimes it is useful to change the way points are plotted so that size represents a variable. This can especially be helpful if you need your plot to be black and white.



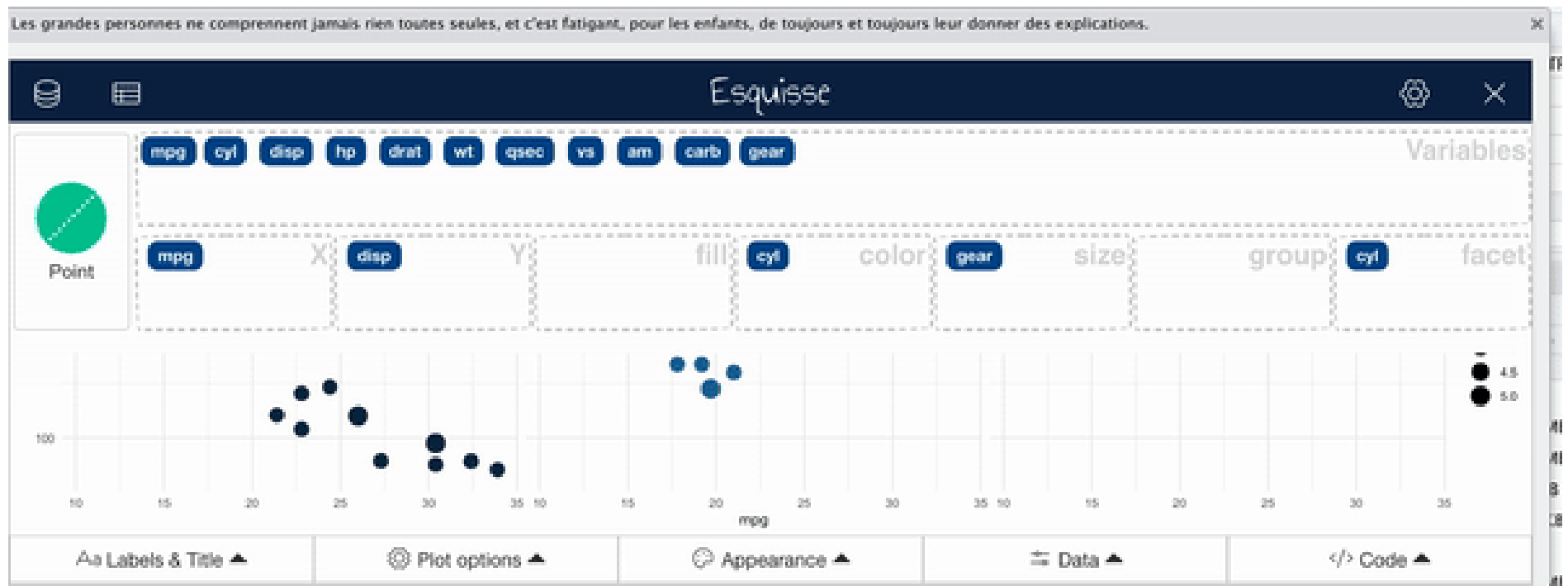
# Add color

For plots with points use the color region to change coloring according to a variable. (use “fill” for bar plots)



# Appearance

You can change the overall appearance with the appearance tab.



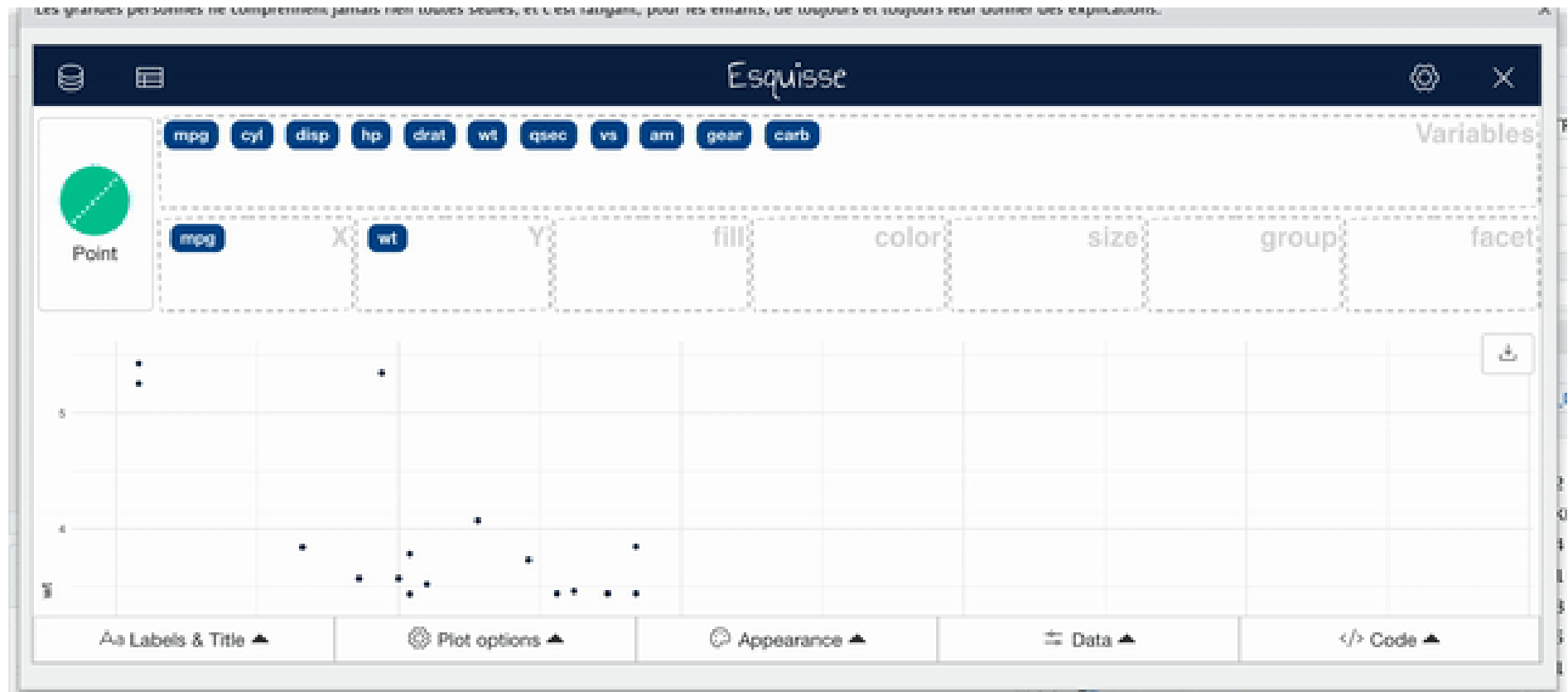
# Smooth Lines

Especially when you have a scatter plot, it can be helpful to add a smooth/trend line.



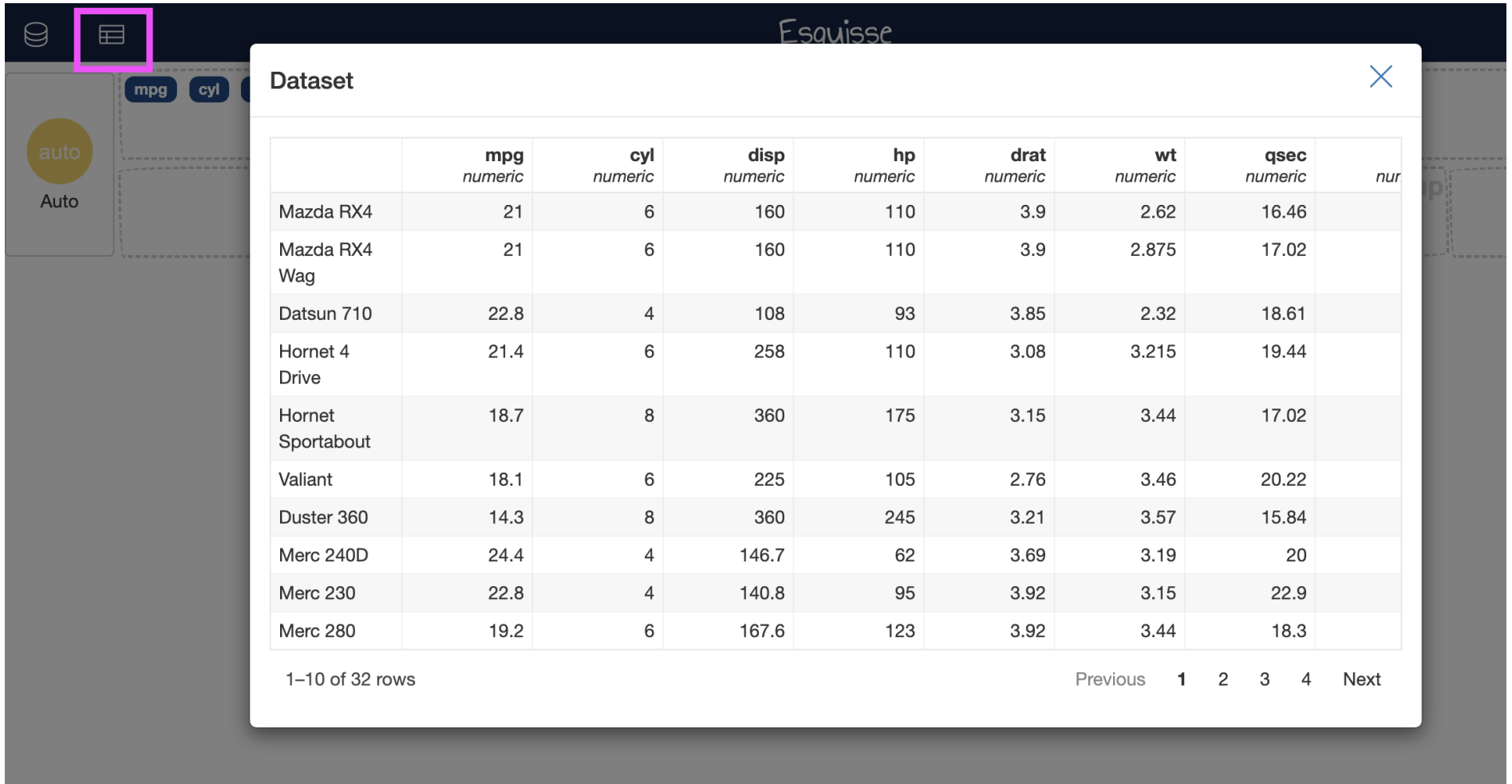
# Change titles

To change titles on your plot, use the titles tab.



# View data

You can also easily view data



The screenshot shows a data visualization interface with a dark blue header. On the left, there is a sidebar with a database icon and a table icon (highlighted with a pink box). Below the sidebar, there is a section labeled 'auto' with a yellow circle and the text 'Auto'. The main area displays a 'Dataset' modal window with a table of car data. The table has columns for car name, mpg, cyl, disp, hp, drat, wt, qsec, and a partially visible 'nur' column. The data rows show various car models and their specifications. At the bottom of the modal, there is a pagination bar showing '1-10 of 32 rows' and navigation buttons for 'Previous', '1', '2', '3', '4', and 'Next'.

	<b>mpg</b> <i>numeric</i>	<b>cyl</b> <i>numeric</i>	<b>disp</b> <i>numeric</i>	<b>hp</b> <i>numeric</i>	<b>drat</b> <i>numeric</i>	<b>wt</b> <i>numeric</i>	<b>qsec</b> <i>numeric</i>	<b>nur</b>
Mazda RX4	21	6	160	110	3.9	2.62	16.46	
Mazda RX4 Wag	21	6	160	110	3.9	2.875	17.02	
Datsun 710	22.8	4	108	93	3.85	2.32	18.61	
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	
Hornet Sportabout	18.7	8	360	175	3.15	3.44	17.02	
Valiant	18.1	6	225	105	2.76	3.46	20.22	
Duster 360	14.3	8	360	245	3.21	3.57	15.84	
Merc 240D	24.4	4	146.7	62	3.69	3.19	20	
Merc 230	22.8	4	140.8	95	3.92	3.15	22.9	
Merc 280	19.2	6	167.6	123	3.92	3.44	18.3	

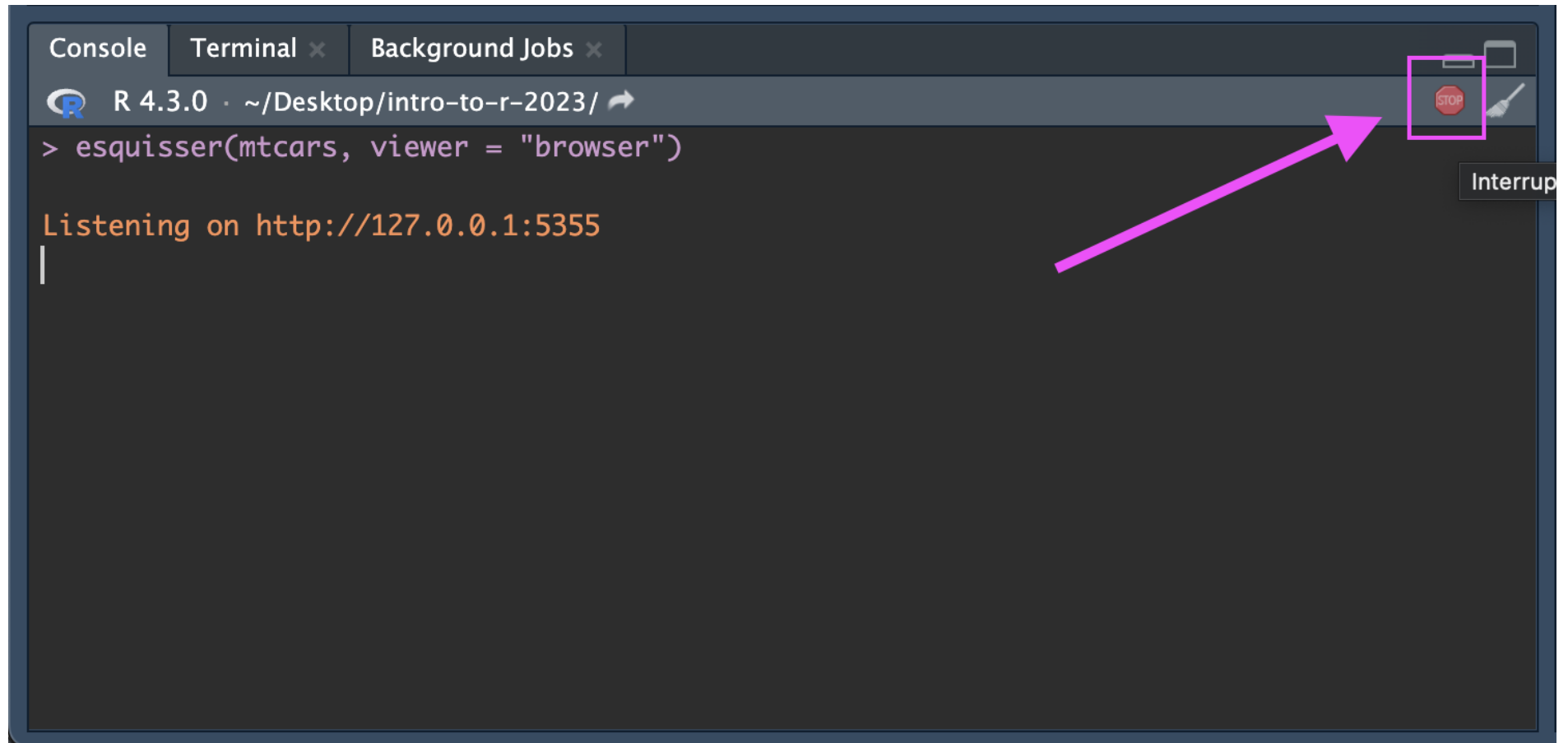
1-10 of 32 rows

Previous 1 2 3 4 Next

# Interrupting Esquisse

You'll need to "interrupt" Esquisse to launch it with a new dataset.

Use the stop button or press ctrl+c to stop the Esquisse app.





## Wide & Long Data Example

```
library(jhur)
wide_circ <- read_circulator()
```

```
## Rows: 1146 Columns: 15
## — Column specification —————
## Delimiter: ","
## chr (2): day, date
## dbl (13): orangeBoardings, orangeAlightings, orangeAverage, purpleBoardings
##
## [ Use `spec()` to retrieve the full column specification for this data.
## [ Specify the column types or set `show_col_types = FALSE` to quiet this message.]
```

# Wide Data

```
library(dplyr)
glimpse(wide_circ)
```

```
## Rows: 1,146
## Columns: 15
## $ day      <chr> "Monday", "Tuesday", "Wednesday", "Thursday", "Fri
## $ date     <chr> "01/11/2010", "01/12/2010", "01/13/2010", "01/14/2
## $ orangeBoardings <dbl> 877, 777, 1203, 1194, 1645, 1457, 839, 999, 1023,
## $ orangeAlightings <dbl> 1027, 815, 1220, 1233, 1643, 1524, 938, 1000, 1047
## $ orangeAverage <dbl> 952.0, 796.0, 1211.5, 1213.5, 1644.0, 1490.5, 888.
## $ purpleBoardings <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ purpleAlightings <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ purpleAverage <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ greenBoardings <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ greenAlightings <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ greenAverage <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ bannerBoardings <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ bannerAlightings <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ bannerAverage <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ daily <dbl> 952.0, 796.0, 1211.5, 1213.5, 1644.0, 1490.5, 888.
```

# Long Data

```
library(tidyr)
long_circ <- wide_circ %>%
  pivot_longer(
    cols = contains(c("boarding")),
    names_to = "Route",
    values_to = "Boardings"
  )
```

# Long Data

```
glimpse(long_circ)
```

```
## Rows: 4,584
## Columns: 13
## $ day      <chr> "Monday", "Monday", "Monday", "Monday", "Tuesday",
## $ date     <chr> "01/11/2010", "01/11/2010", "01/11/2010", "01/11/2010",
## $ orangeAlightings <dbl> 1027, 1027, 1027, 1027, 815, 815, 815, 815, 1220,
## $ orangeAverage   <dbl> 952.0, 952.0, 952.0, 952.0, 796.0, 796.0, 796.0, 796.0,
## $ purpleAlightings <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
## $ purpleAverage    <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
## $ greenAlightings  <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
## $ greenAverage     <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
## $ bannerAlightings <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
## $ bannerAverage    <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
## $ daily            <dbl> 952.0, 952.0, 952.0, 952.0, 796.0, 796.0, 796.0, 796.0,
## $ Route            <chr> "orangeBoardings", "purpleBoardings", "greenBoardings",
## $ Boardings        <dbl> 877, NA, NA, NA, 777, NA, NA, NA, 1203, NA, NA, NA, NA,
```

## Make a plot of boardings by day for different routes

```
esquisser(wide_circ) # days as x...? Tricky!  
esquisser(long_circ) # day as x, Boardings as y, Route as fill
```

## Summary

- Use the `esquisser()` function on a dataset
- Use the `viewer = "browser"` argument to launch in your browser.
- Code from Esquisse can be copied into code chunks to be generated in the "Plots" pane
- It's easier if your code is in "long" form!

# Lab

[Class Website](#)

[Lab](#)



Image by [Gerd Altmann](#) from [Pixabay](#)