

Good news, everyone!



A very short intro to ggplot2



ROBOT
WASH
STREAKLESS



Can't I just do everything in Excel?



VS



Powerful plots



Easy replication



Quick to learn



<https://cran.uni-muenster.de/>



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](#)

[Contributed](#)

The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms


Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (Tuesday 2016-06-21, Bug in Your Hair) [R-3.3.1.tar.gz](#), read [what's new](#) in the latest version.
- Sources of [R alpha and beta releases](#) (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are [available here](#). Please read about [new features and bug fixes](#) before filing corresponding feature requests or bug reports.
- Source code of older versions of R is [available here](#).
- Contributed extension [packages](#)

Questions About R

- If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

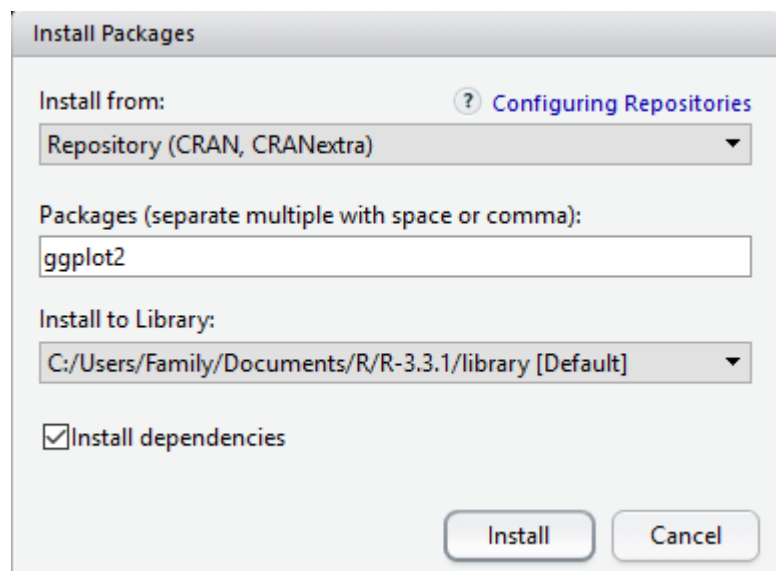
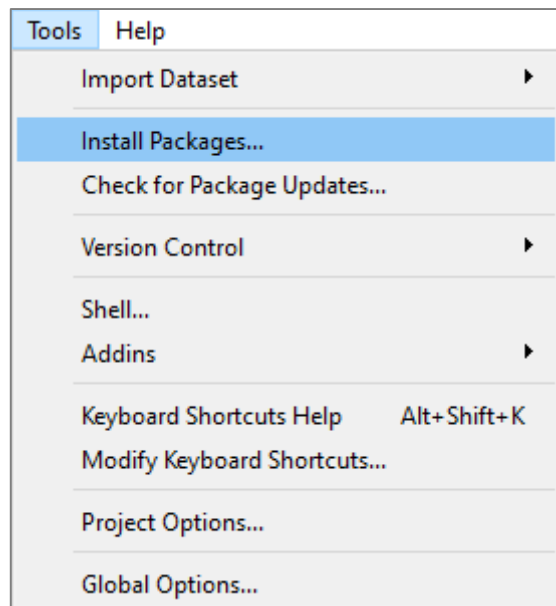
https://www.rstudio.com/products/rstudio/

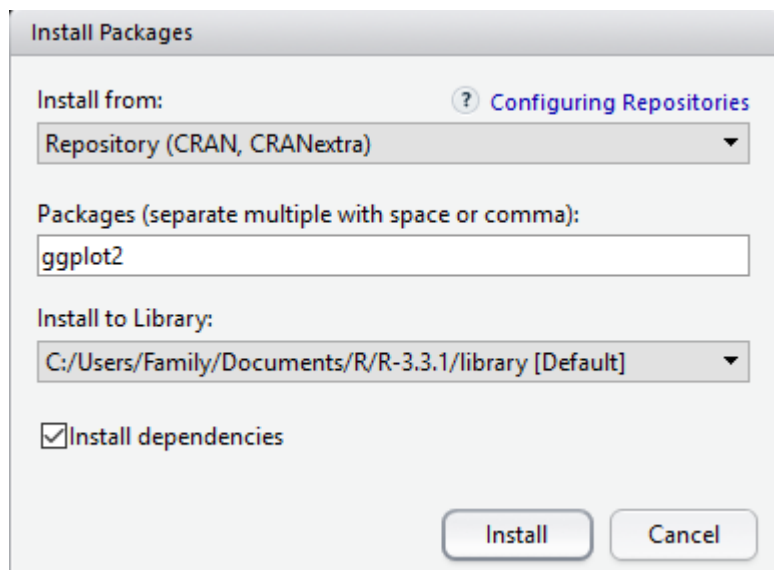
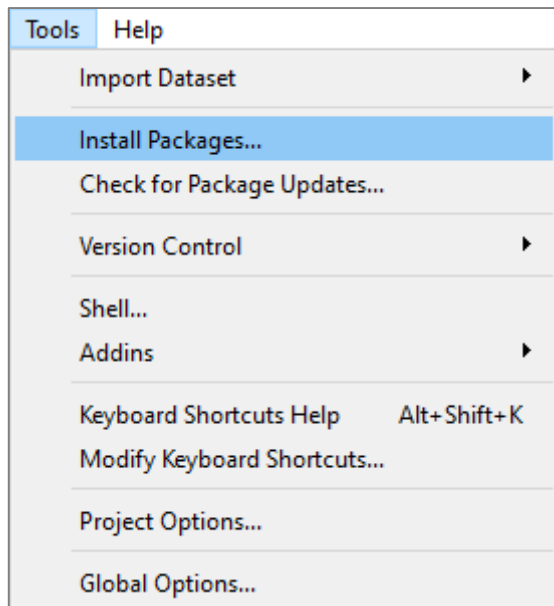


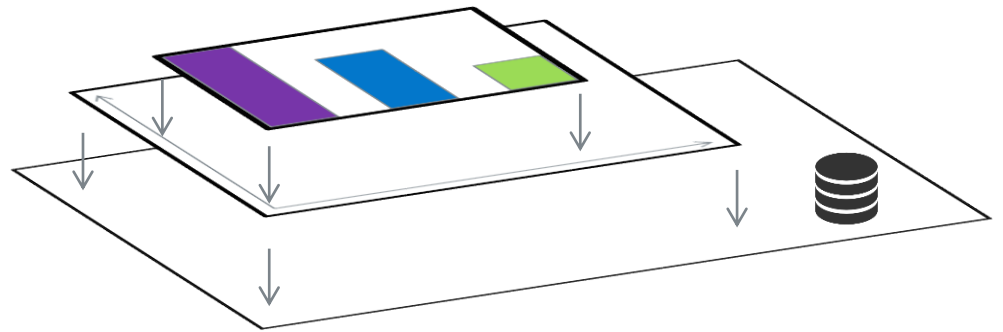
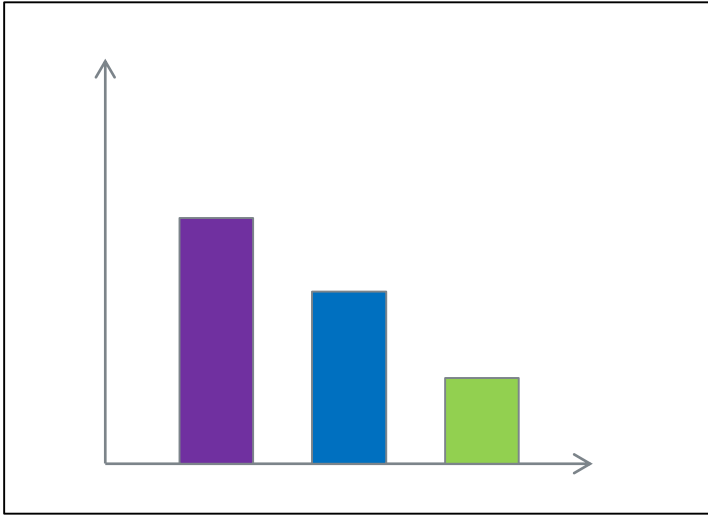
ProductsResourcesPricingAbout UsBlog

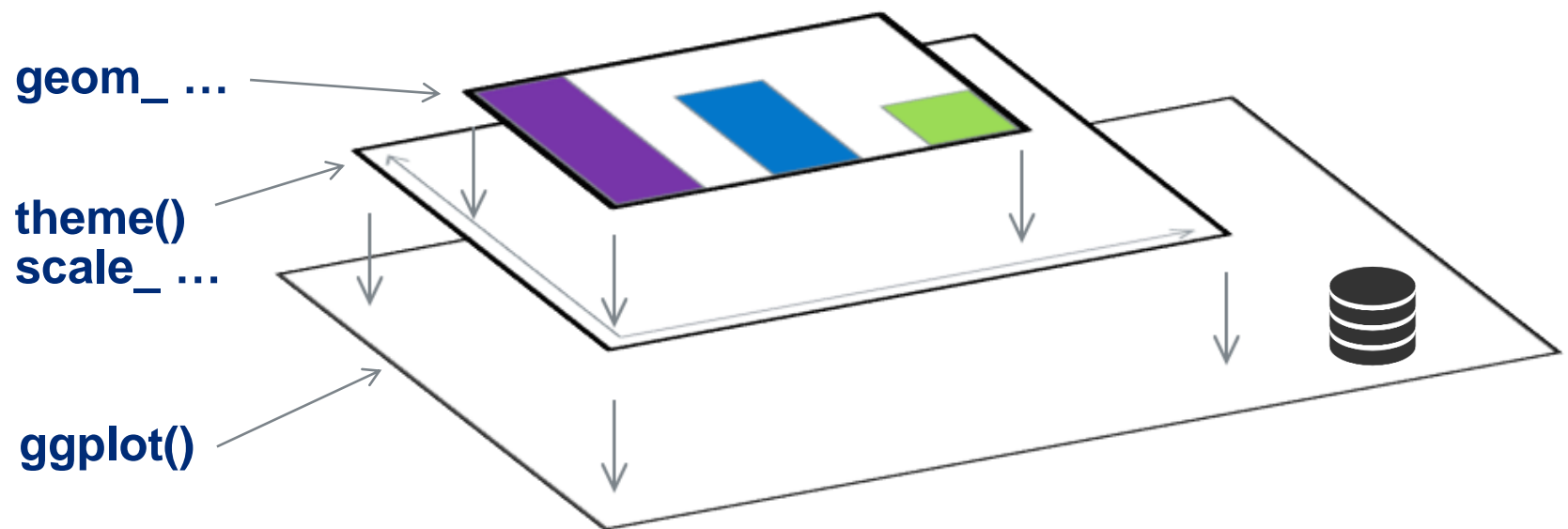
RStudio Desktop

	Open Source Edition	Commercial License
Overview	<ul style="list-style-type: none">• Access RStudio locally• Syntax highlighting, code completion, and smart indentation• Execute R code directly from the source editor• Quickly jump to function definitions• Easily manage multiple working directories using projects• Integrated R help and documentation• Interactive debugger to diagnose and fix errors quickly• Extensive package development tools	<p>All of the features of open source; plus:</p> <ul style="list-style-type: none">• A commercial license for organizations not able to use AGPL software• Access to priority support
Support	Community forums only	<ul style="list-style-type: none">• Priority Email Support• 8 hour response during business hours (ET)
License	AGPL v3	RStudio License Agreement
Pricing	Free	\$995/year
	DOWNLOAD RSTUDIO DESKTOP	BUY NOW









1. Create base ggplot layer

p **<-** **ggplot(** **clean_data** **)**

↑ new base ggplot-object

↑ function

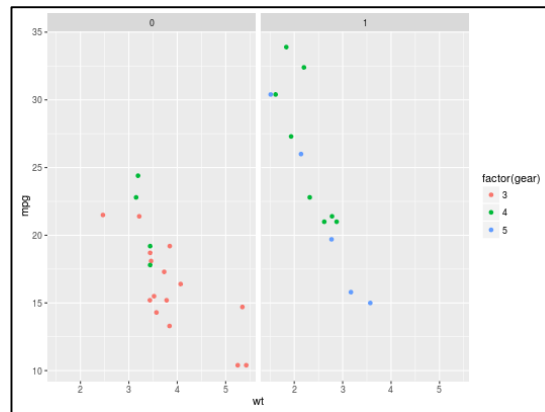
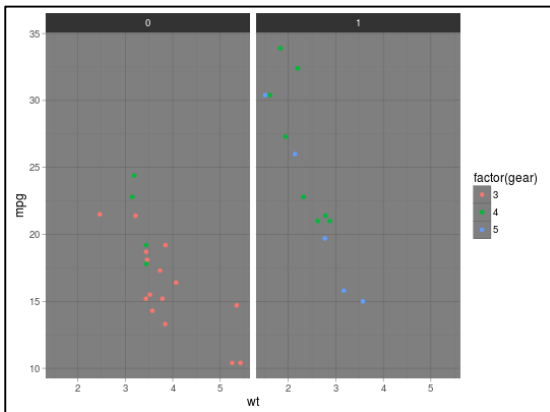
↑ data used for plotting

2. Choosing a theme layer

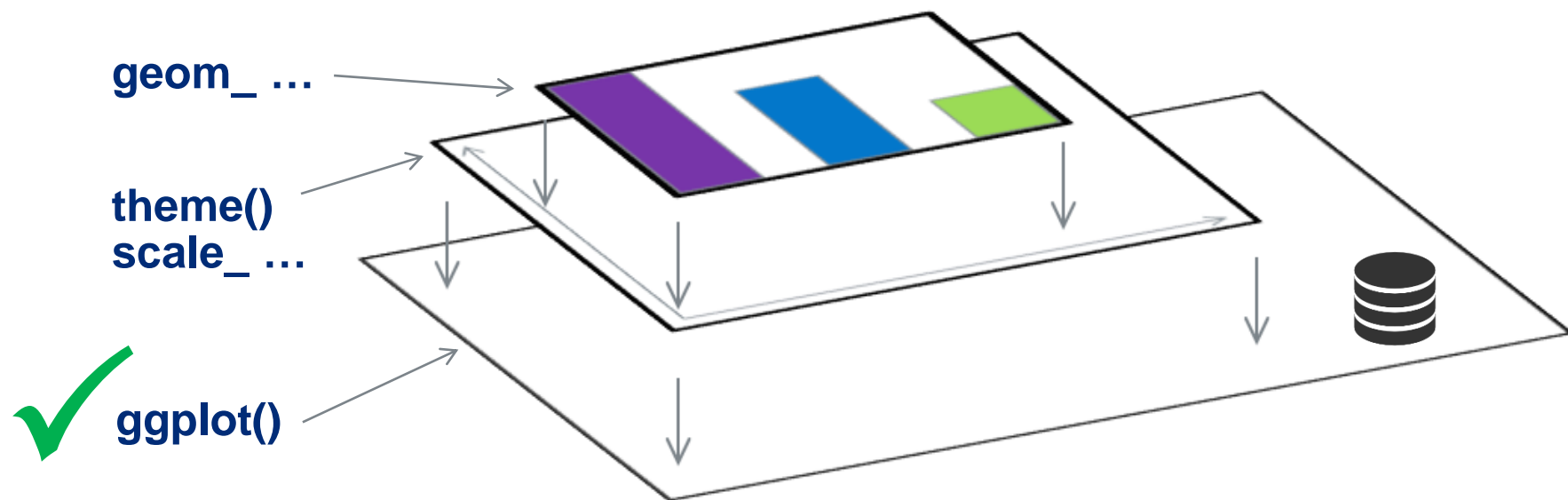
p **<-** **p** **+** **theme_dark()**

↑ ggplot-object

↑ ggplot theme



<http://docs.ggplot2.org/current/ggtheme.html>



Type `?theme` in R to read the help page:

<code>panel.grid.minor.y</code>	horizontal minor grid lines (<code>element_line</code> ; inherits from <code>panel.grid.minor</code>)
<code>panel.ontop</code>	option to place the panel (background, gridlines) over the data layers. Usually used with a transparent or blank <code>panel.background</code> . (logical)
<code>plot.background</code>	background of the entire plot (<code>element_rect</code> ; inherits from <code>rect</code>)
<code>plot.title</code>	plot title (text appearance) (<code>element_text</code> ; inherits from <code>text</code>)
<code>plot.margin</code>	margin around entire plot (unit with the sizes of the top, right, bottom, and left margins)
<code>strip.background</code>	background of facet labels (<code>element_rect</code> ; inherits from <code>rect</code>)
<code>strip.text</code>	facet labels (<code>element_text</code> ; inherits from <code>text</code>)
<code>strip.text.x</code>	facet labels along horizontal direction (<code>element_text</code> ; inherits from <code>strip.text</code>)

`?element_text`

Arguments

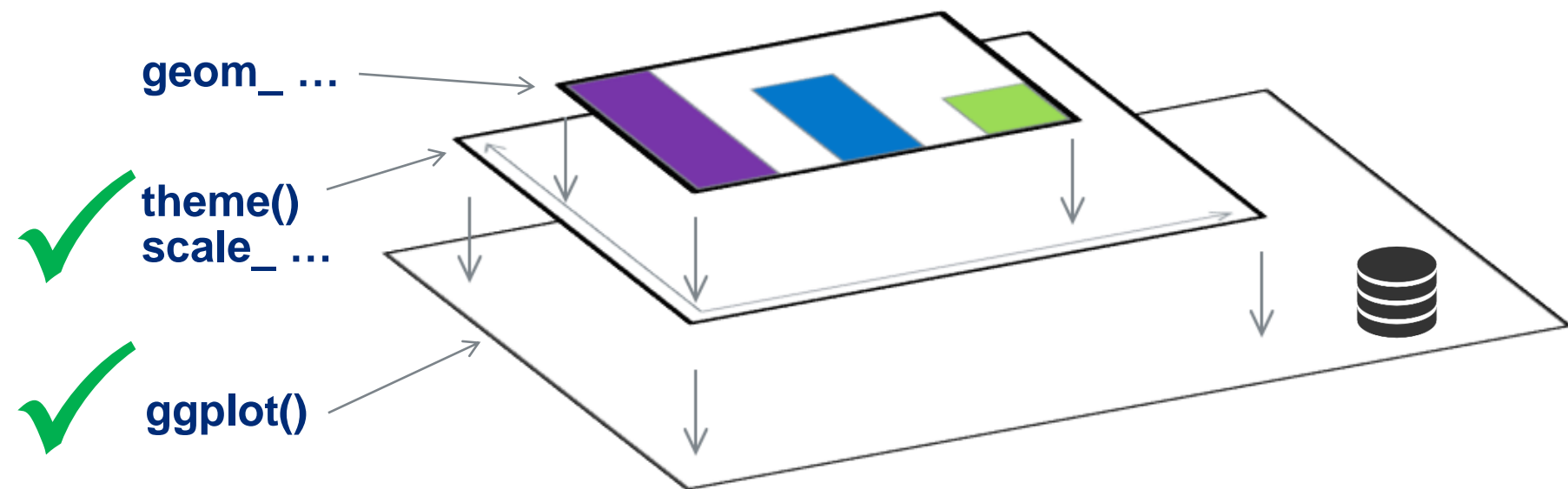
<code>family</code>	font family
<code>face</code>	font face ("plain", "italic", "bold", "bold.italic")
<code>colour</code>	text colour
<code>size</code>	text size (in pts)
<code>hjust</code>	horizontal justification (in [0, 1])
<code>vjust</code>	vertical justification (in [0, 1])
<code>angle</code>	angle (in [0, 360])
<code>lineheight</code>	line height

3. Customizing your theme

`p` `<-` `p` `+` `theme(plot.title = element_text(face="bold"))`

↑ ↑ ↑ ↑

ggplot-object Function to modify theme Part of the theme you want to change How do you want to change the theme?



4. Create your plot



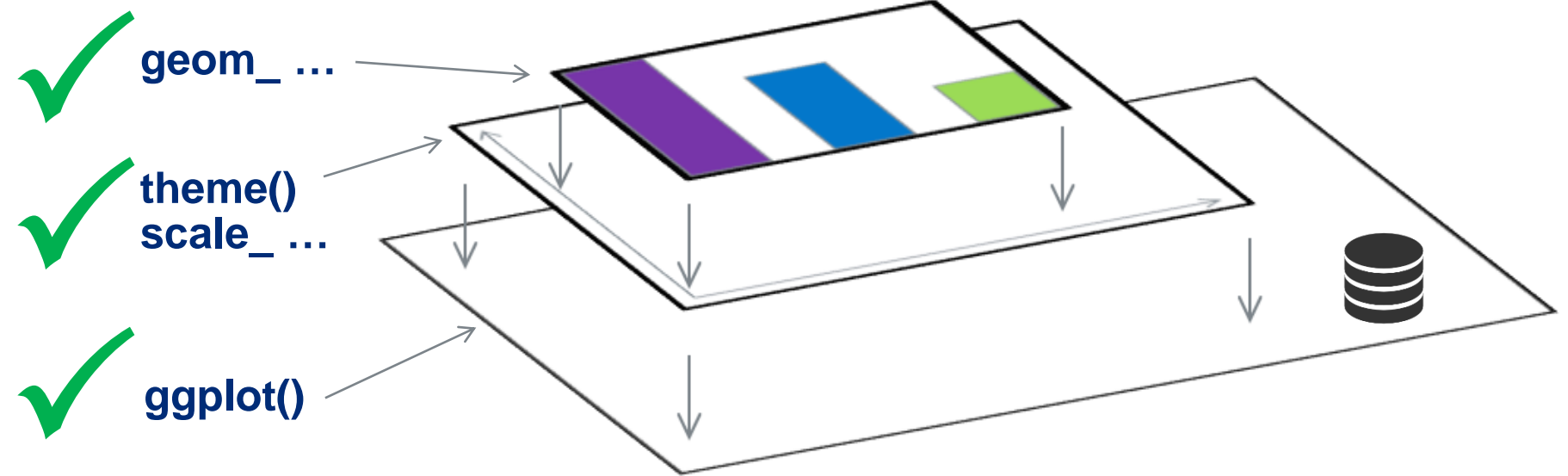
Geoms

Geoms, short for geometric objects, describe the type of plot you will produce.

- `geom_abline` (geom_hline, geom_vline)
Lines: horizontal, vertical, and specified by slope and intercept.
- `geom_bar` (stat_count)
Bars, rectangles with bases on x-axis
- `geom_bin2d` (stat_bin2d, stat_bin_2d)
Add heatmap of 2d bin counts.



<http://docs.ggplot2.org/current/index.html>



5. Making the plot look pretty

Description	Example
Changing the label of an axis: ylab(), xlab():	... + <code>ylab("New Name")</code>
Changing the title of the plot: ggtitle()	... + <code>ggtitle("New title")</code>
Changing the scale of y-axis: <code>scale_y_continuous()</code> <code>scale_y_discrete()</code> <code>scale_y_log10()</code> ...	<code>library(scales)</code> ... + <code>scale_y_continuous(labels = comma)</code>
Changing fill in plot: <code>scale_fill_manual()</code> <code>scale_fill_continuous()</code> <code>scale_fill_discrete()</code> + <code>scale_fill_manual(values=c("#006D9E", "#625BC4", "#AD208E"))</code>
Splitting plot into several plots: <code>facet_grid(.~ COLUMN_NAME)</code> <code>facet_wrap(.~ COLUMN_NAME)</code>	... + <code>facet_grid(. ~ COUNTRY)</code>
... many more options	http://docs.ggplot2.org/current/index.html



Define the packages we want to use.

```
1. library(fueleconomy) #dataset
2. library(ggplot2) #plotting
3. library(RColorBrewer) #color-scheme
4.
```

Get some data.

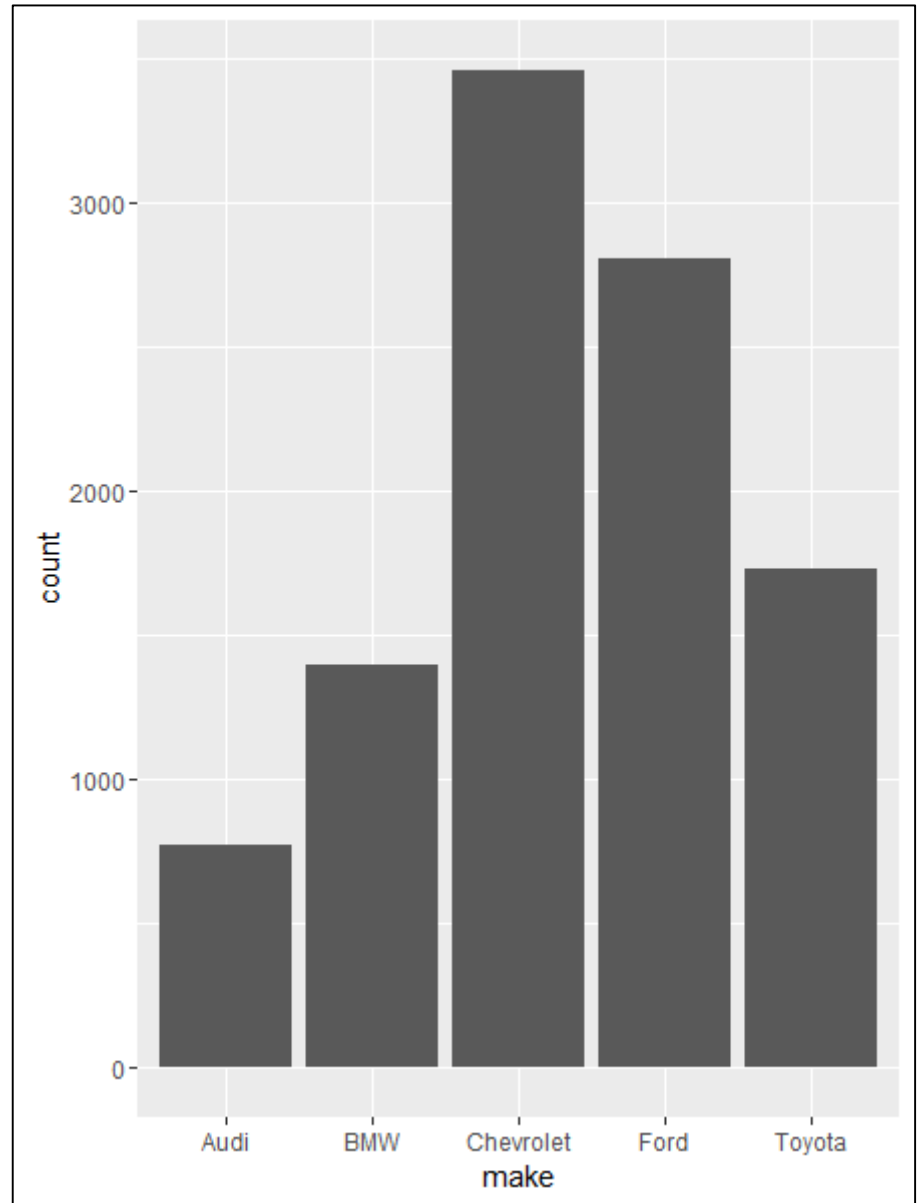
```
5. data(vehicles)
6. ?vehicles
7. car_data <- subset(vehicles, make %in% c("Audi", "BMW", "Ford", "Chevrolet", "Toyota"))
```

```
10. p <- ggplot(car_data)
```

```
11. p + geom_bar(aes(make))
```

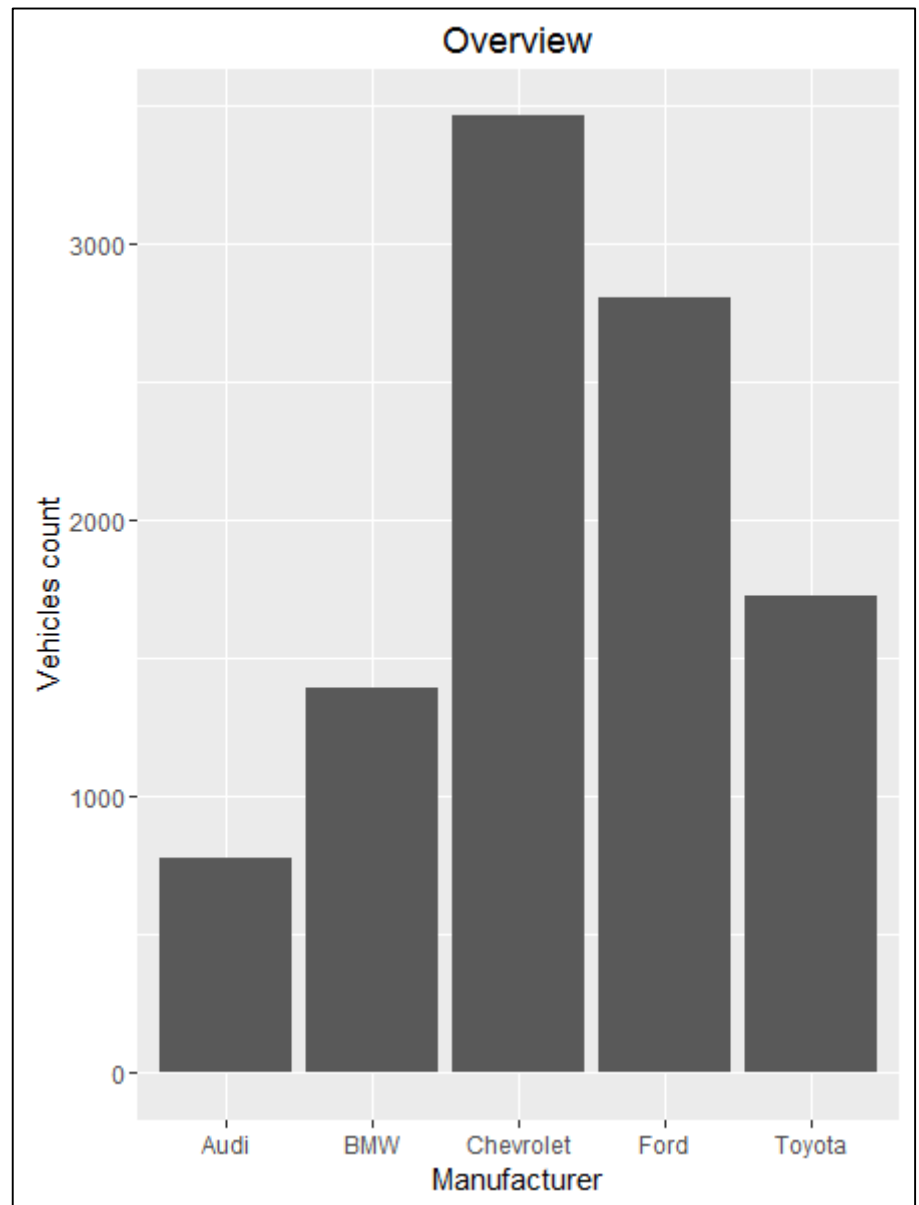
```
12. 
```

Create a plot.



```
10. p <- ggplot(car_data)
11. p + geom_bar(aes(make))
12.
13. p + geom_bar(aes(make)) +
14.   ylab("Vehicles count") +
15.   xlab("Manufacturer") +
16.   ggtitle("Overview")
```

Add titles.

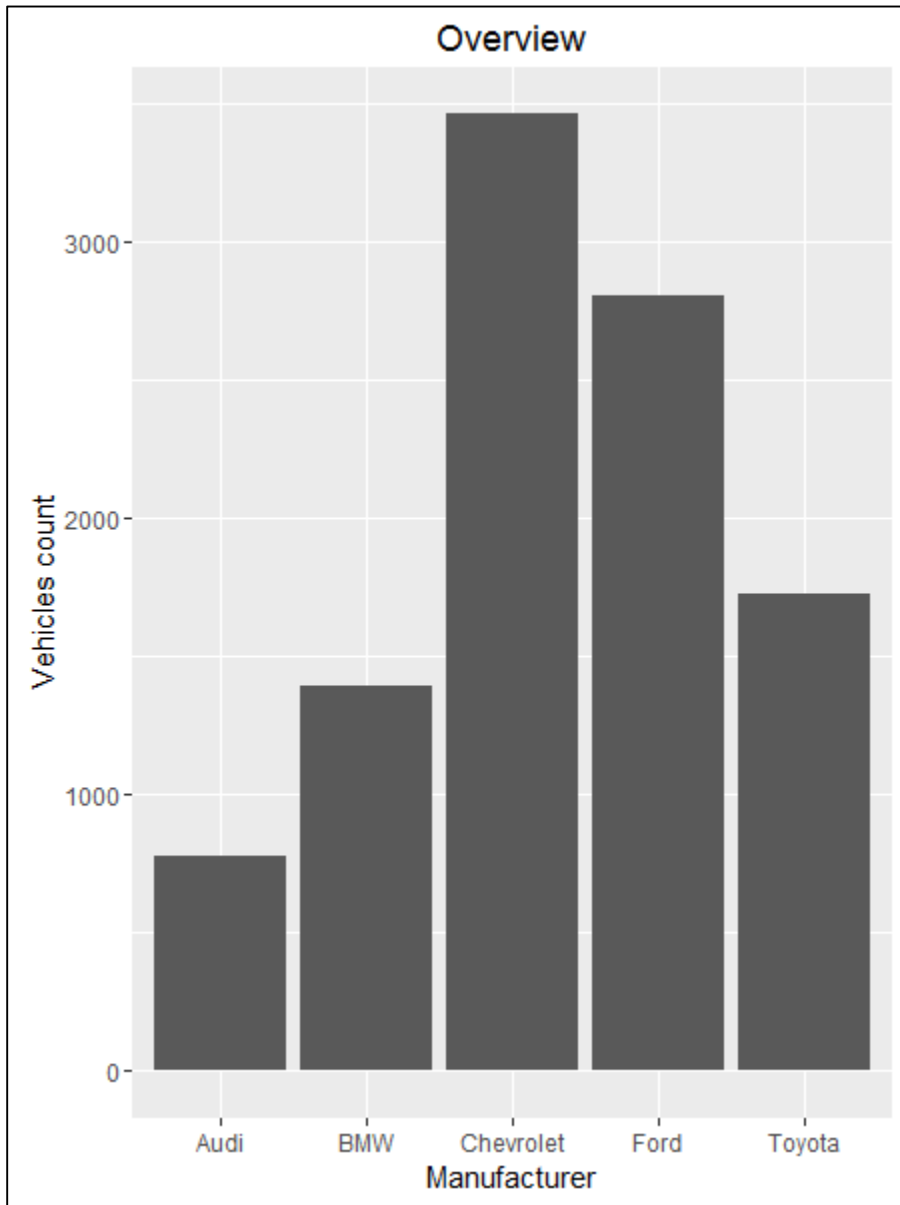


```
22. p <- ggplot(car_data) +  
23.   theme_light() +  
24.   theme(plot.title = element_text(size = 16, face = "bold"),  
25.         axis.title = element_text(face = "bold"))
```

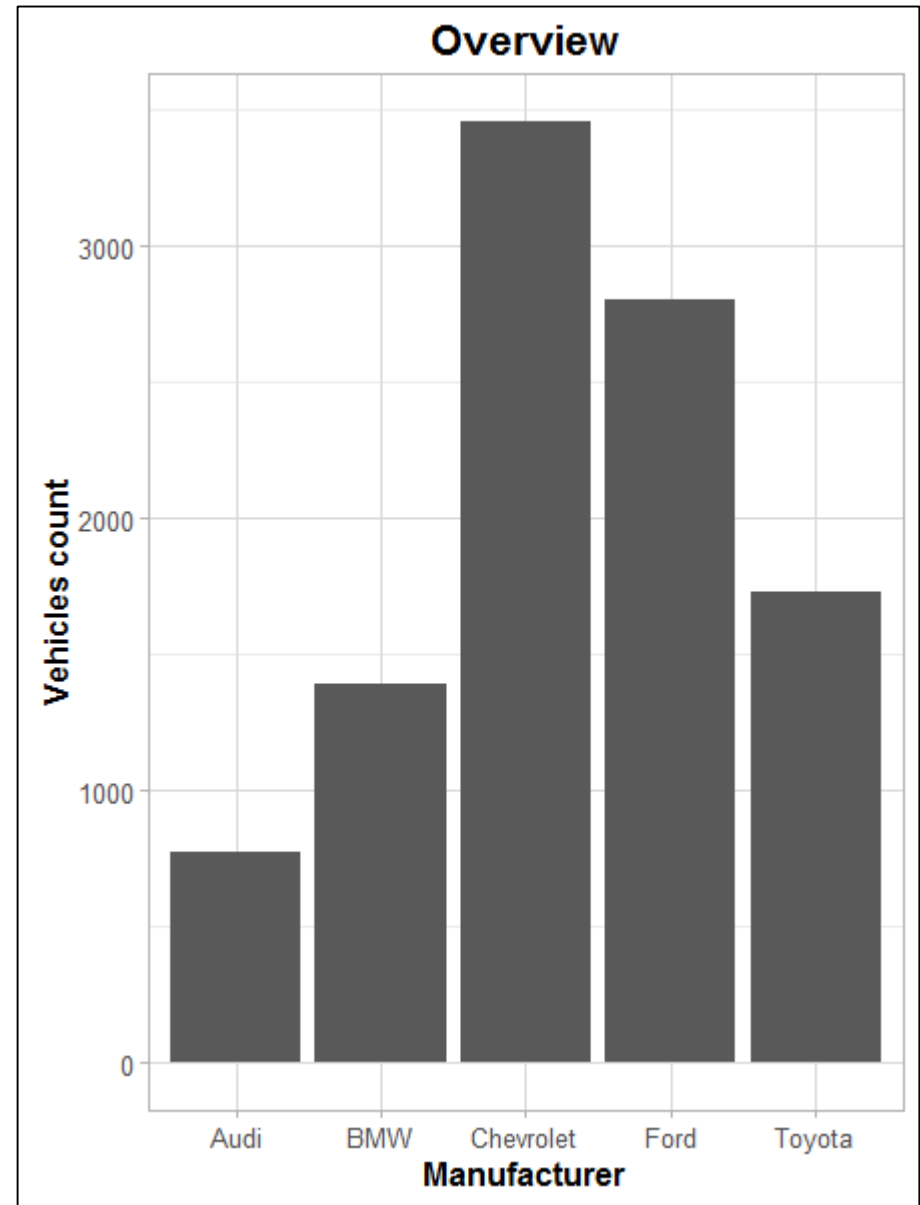
Adjust theme.

```
26.  
27. p + geom_bar(aes(make)) +  
28.   ylab("Vehicles count") +  
29.   xlab("Manufacturer") +  
30.   ggtitle("Overview")
```

OLD

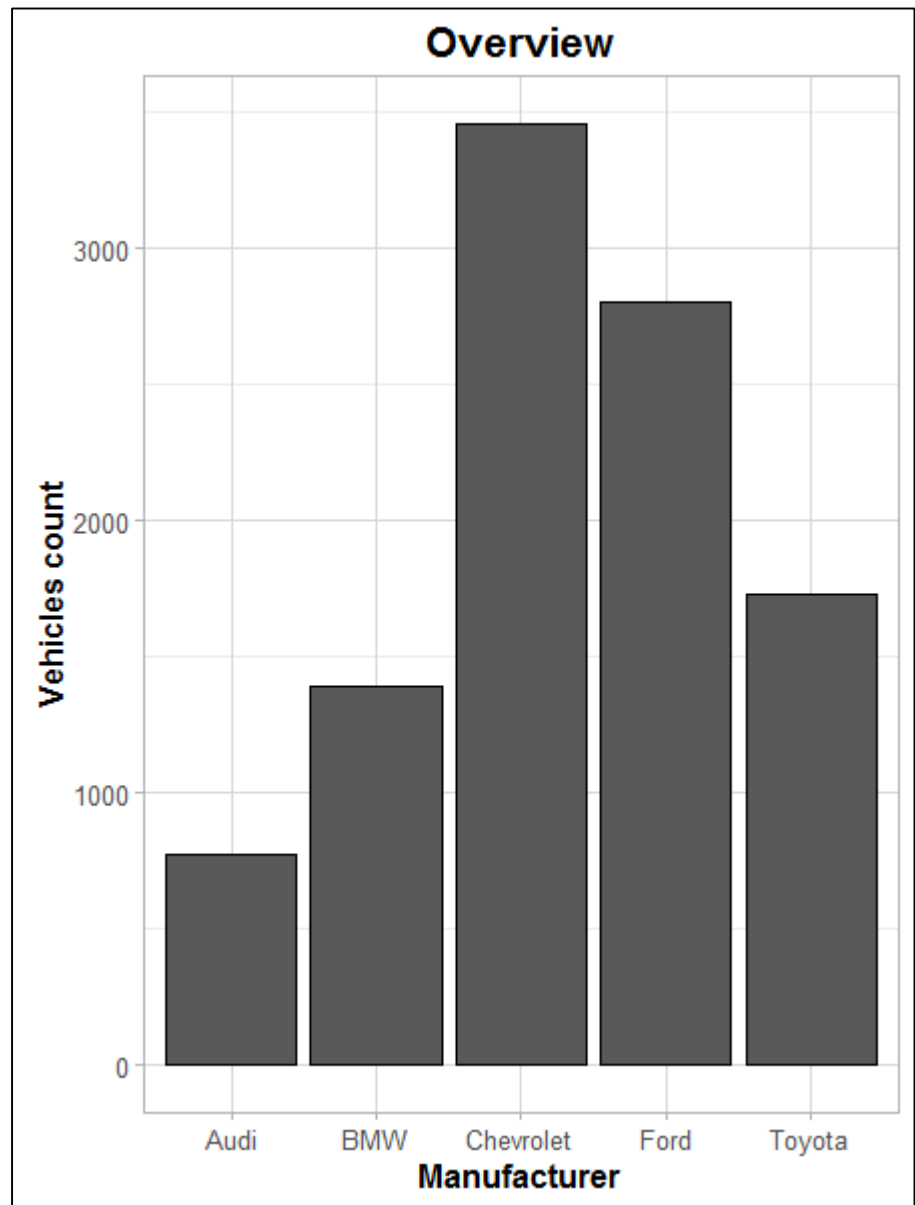


NEW



Set exterior color

```
34. p + geom_bar(aes(make), color = "black") +  
35.   ylab("Vehicles count") +  
36.   xlab("Manufacturer") +  
37.   ggtitle("Overview")
```

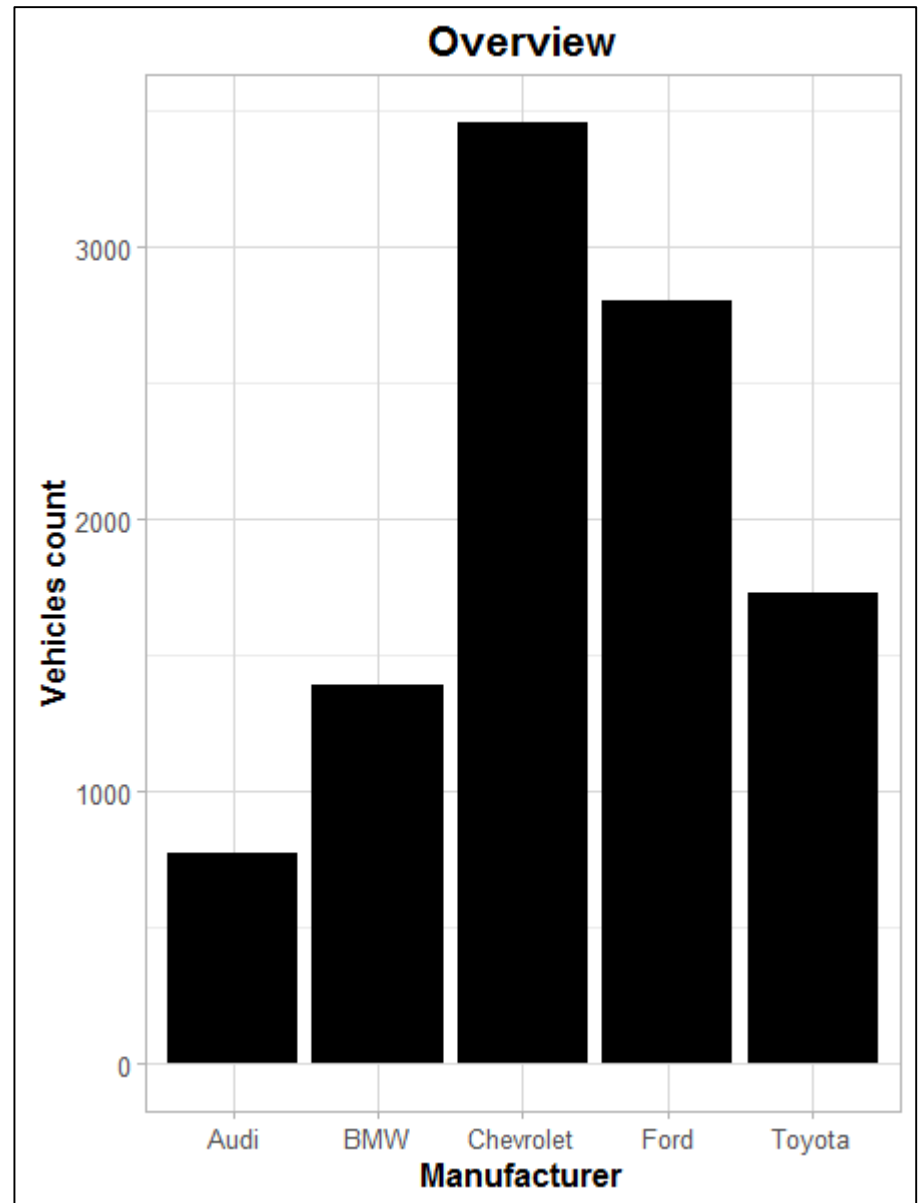


```

34. p + geom_bar(aes(make), color = "black") +
35.   ylab("Vehicles count") +
36.   xlab("Manufacturer") +
37.   ggtitle("Overview")
38.
39. p + geom_bar(aes(make), fill = "black") +
40.   ylab("Vehicles count") +
41.   xlab("Manufacturer") +
42.   ggtitle("Overview")

```

Set interior color

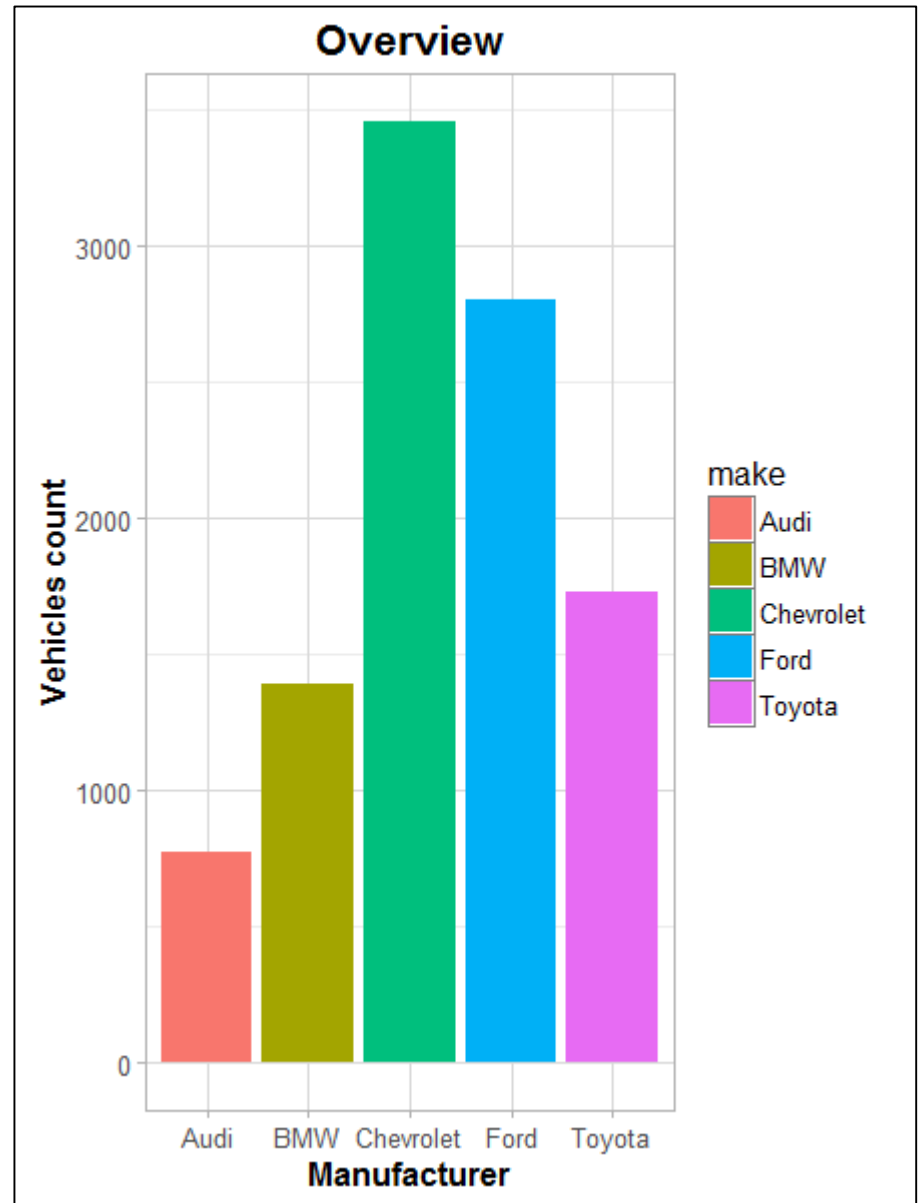


```

34. p + geom_bar(aes(make), color = "black") +
35.   ylab("Vehicles count") +
36.   xlab("Manufacturer") +
37.   ggtitle("Overview")
38.
39. p + geom_bar(aes(make), fill = "black") +
40.   ylab("Vehicles count") +
41.   xlab("Manufacturer") +
42.   ggtitle("Overview")
43.
44. p + geom_bar(aes(make, fill = make)) +
45.   ylab("Vehicles count") +
46.   xlab("Manufacturer") +
47.   ggtitle("Overview")

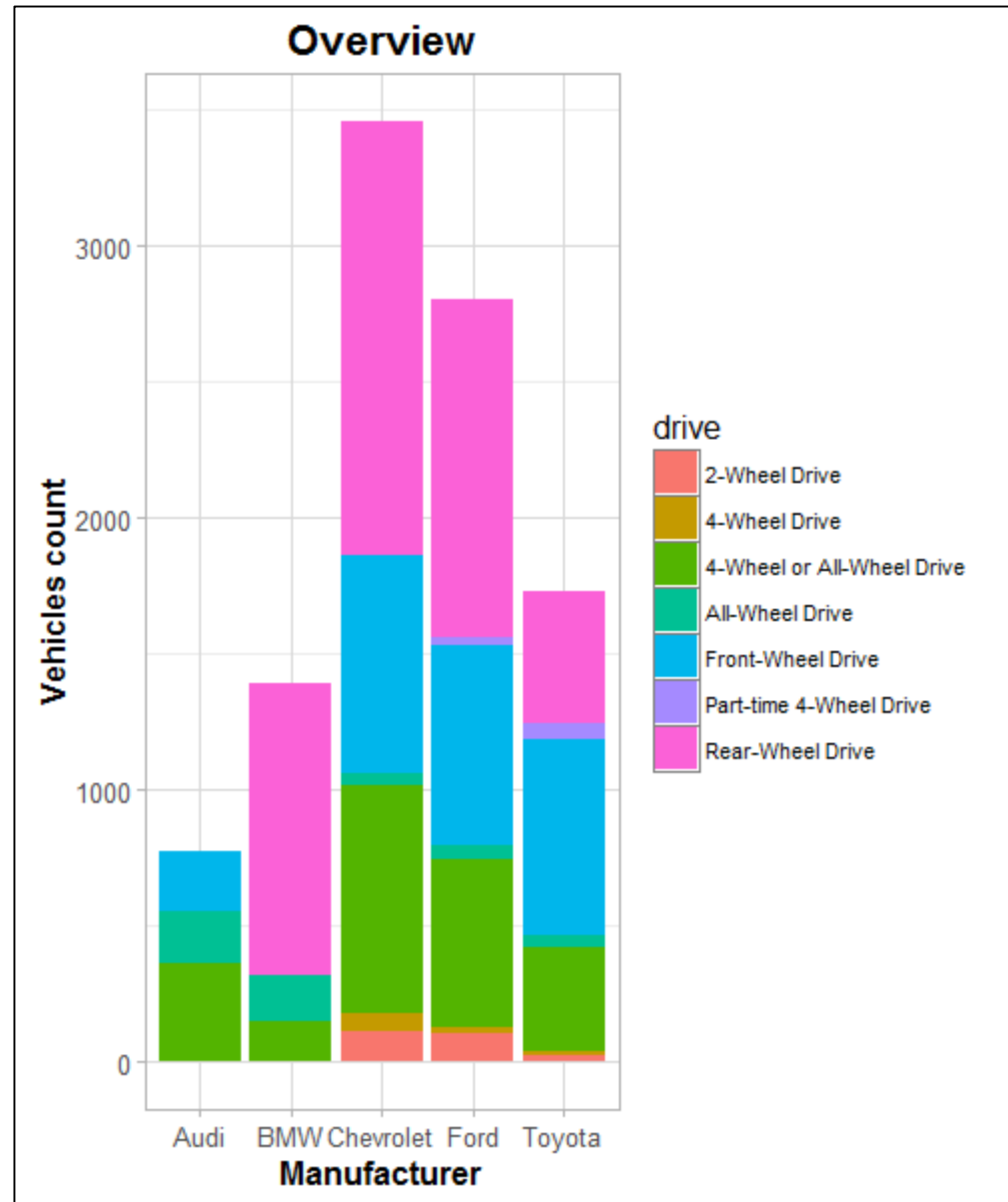
```

Set fill by factor



Set fill by factor

```
49. p + geom_bar(aes(make, fill = drive)) +  
50.   ylab("Vehicles count") +  
51.   xlab("Manufacturer") +  
52.   ggtitle("Overview")
```

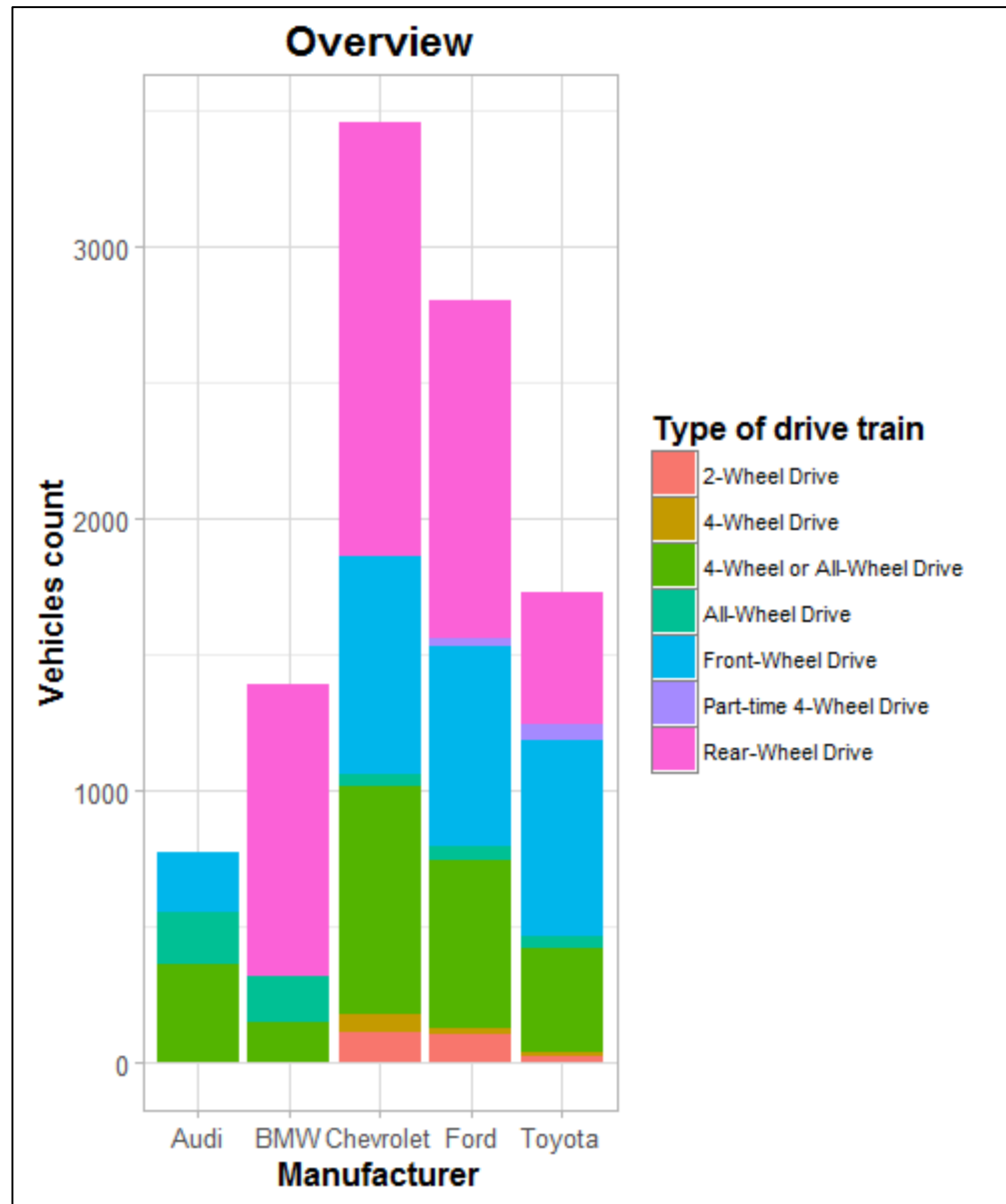


```

49. p + geom_bar(aes(make, fill = drive)) +
50.   ylab("Vehicles count") +
51.   xlab("Manufacturer") +
52.   ggtitle("Overview")
53.
54. p + geom_bar(aes(make, fill = drive)) +
55.   ylab("Vehicles count") +
56.   xlab("Manufacturer") +
57.   ggtitle("Overview") +
58.   labs(fill='Type of drive train') +
59.   theme(legend.title = element_text(face = "bold"))

```

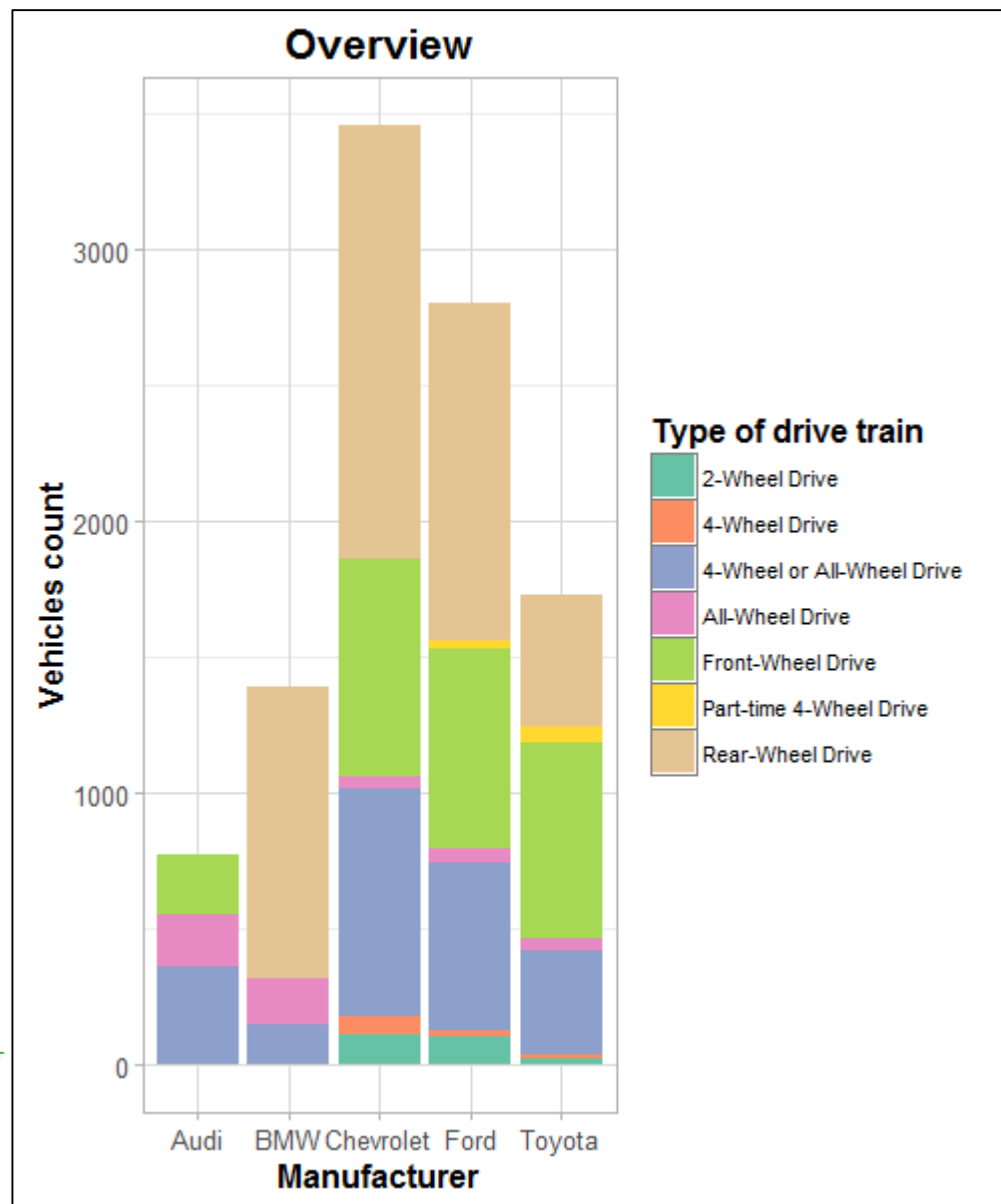
Adjust legend

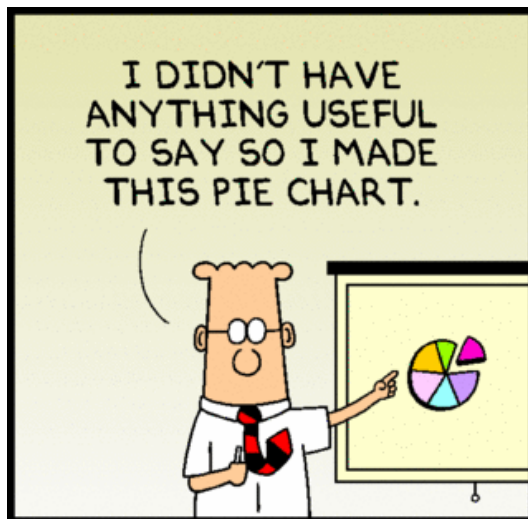



```

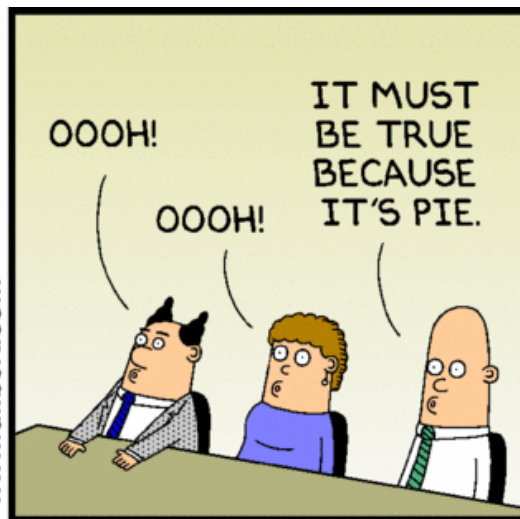
49. p + geom_bar(aes(make, fill = drive)) +
50.   ylab("Vehicles count") +
51.   xlab("Manufacturer") +
52.   ggtitle("Overview")
53.
54. p + geom_bar(aes(make, fill = drive)) +
55.   ylab("Vehicles count") +
56.   xlab("Manufacturer") +
57.   ggtitle("Overview") +
58.   labs(fill='Type of drive train') +
59.   theme(legend.title = element_text(face = "bold"))
60.
61. nicer_colors <- brewer.pal(7, "Set2")
62. p + geom_bar(aes(make, fill = drive)) +
63.   ylab("Vehicles count") +
64.   xlab("Manufacturer") +
65.   ggtitle("Overview") +
66.   labs(fill='Type of drive train') +
67.   theme(legend.title = element_text(face = "bold")) +
68.   scale_fill_manual(values = nicer_colors)

```





www.dilbert.com scottadams@aol.com



3-7-09 ©2009 Scott Adams, Inc./Dist. by UFS, Inc.

