



# ggplot Wizardry

---

My Favorite Tricks and Secrets for Beautiful Plots in R

**Dr. Cédric Scherer**  
Freelancing Data Visualization Designer



# **Scientist by Training**



Computational Ecology at the Leibniz Institute for Zoo and Wildlife Research

# **DataVizard by Heart**



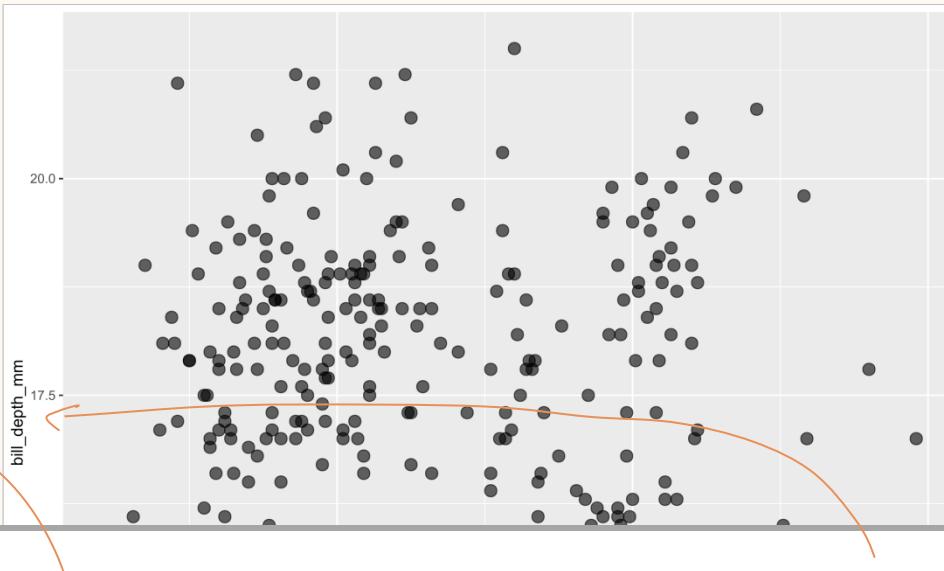
Self-employed Data Visualization Designer & Workshop Instructor

# ggplot2

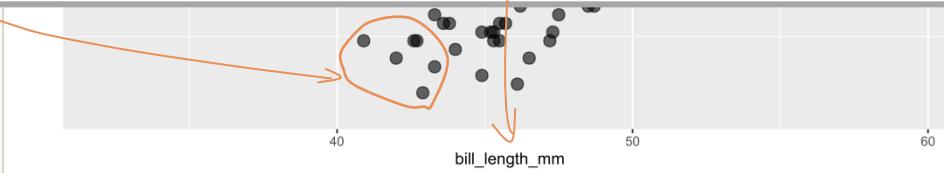
[gee-gee-plot-two]

*an R package*

is a system for declaratively creating graphics  
based on “The Grammar of Graphics”.



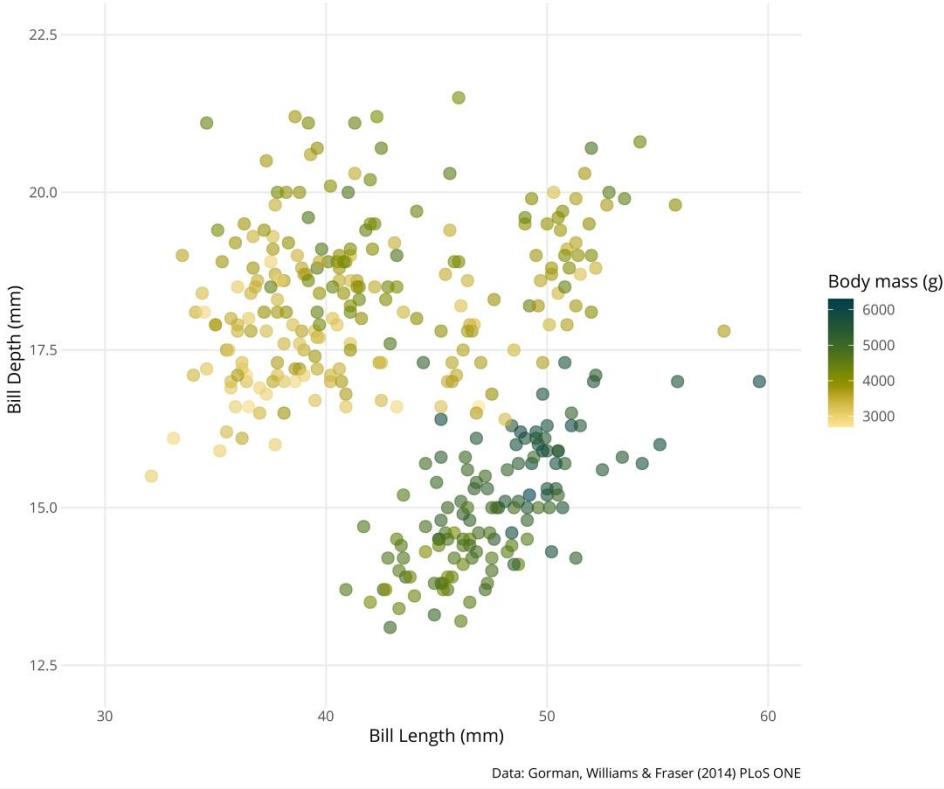
```
ggplot(data = penguins, aes(x = bill_length_mm, y = bill_depth_mm)) +  
  geom_point(alpha = .6)
```



You provide the data, tell ggplot2 how to map variables to aesthetics, what graphical primitives to use, and it takes care of the details.

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



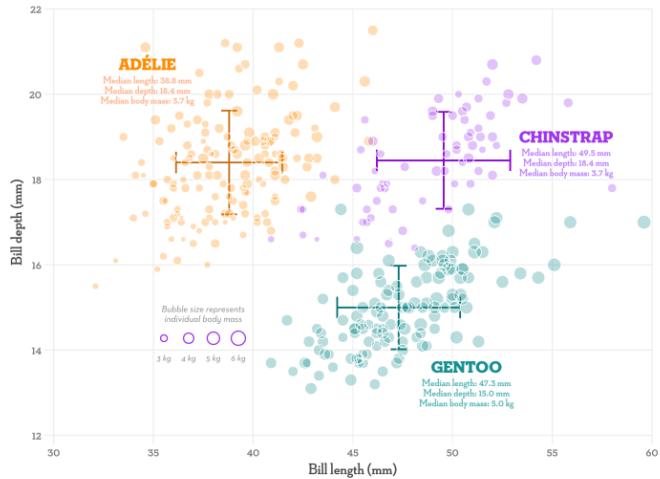
You provide the data, tell ggplot2 how to map variables to aesthetics, what graphical primitives to use, and it takes care of the details.

## BILL DIMENSIONS OF BRUSH-TAILED PENGUINS

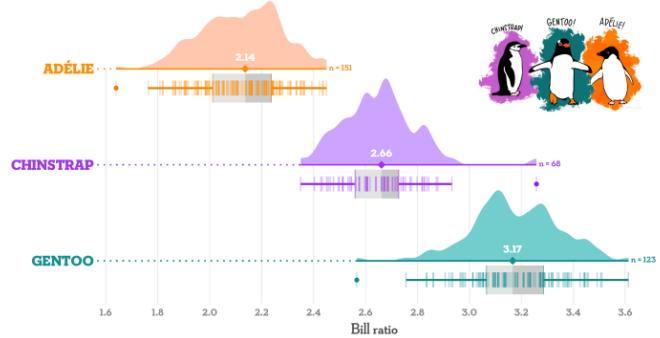
*Pygoscelis adeliae* (Adélie penguin) • *P. antarctica* (Chinstrap penguin) • *P. papua* (Gentoo penguin)



A. Scatterplot of bill length versus bill depth (median +/- sd)



B. Distribution of the bill ratio, estimated as bill length divided by bill depth

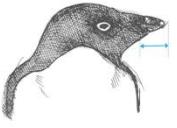


Note: In the original data, bill dimensions are recorded as "culmen length" and "culmen depth". The culmen is the dorsal (upper) ridge of a bird's bill.  
Visualization: Cédric Scherer • Data: Gorman, Williams & Fraser (2014) DOI: 10.1371/journal.pone.0090001 • Illustrations: Alison Horst

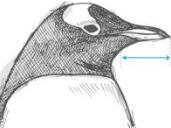
## Palmer Penguins Bill Length

Palmer Archipelago is a group of islands off the northwestern coast of the Antarctic Peninsula.  
The histograms show that females have shorter bills than males in every species.

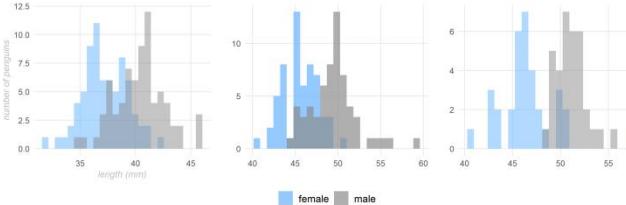
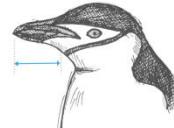
ADELIE



GENTOO



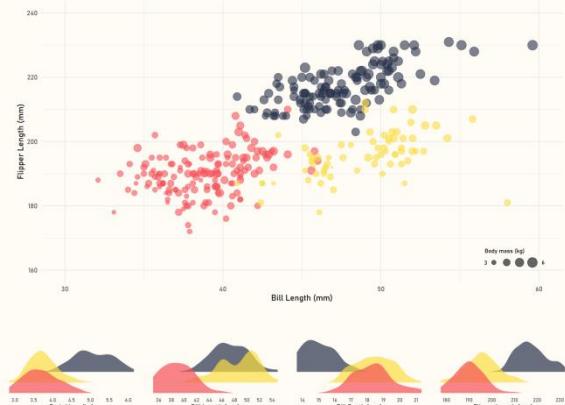
CHINSTRAP



Visualization: Laura Navarro Soler | Data: Gorman, Williams & Fraser (2014)

Measurements of body mass, bill length, bill depth, and flipper length in **Adélie**, **Gentoo**, and **Chinstrap** penguins, collected from 3 islands in the Palmer Archipelago, Antarctica.

Data: Dr. K. Gorman, Dr. A. Horst, & Dr. A. Hill. alisonhorst/palmerpenguins  
Visualization: Joseph Shaw / @JosephShaw.



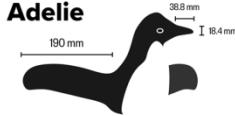
#TidyTuesday Contributions by

- ← myself
- ↖ Laura Navarro
- ↙ Joe Shaw
- ↓ Georgios Karamanis

## Palmer Penguins

Median length of flipper, length and depth of bill,  
of 342 penguins recorded between 2007 and 2009

Adelie



Recorded penguins by species

151

Chinstrap



68

Gentoo



123

Source: Dr. Kristen Gorman and the Palmer Station, Antarctica LTER | Graphic: Georgios Karamanis

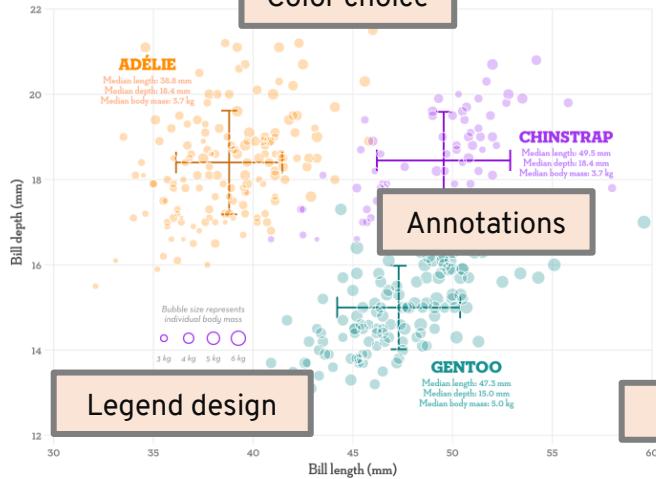
## BILL DIMENSIONS OF BRUSH-TAILED PENGUINS

*Pygoscelis adeliae* (Adélie penguin) • *P. antarctica* (Chinstrap penguin) • *P. papua* (Gentoo penguin)

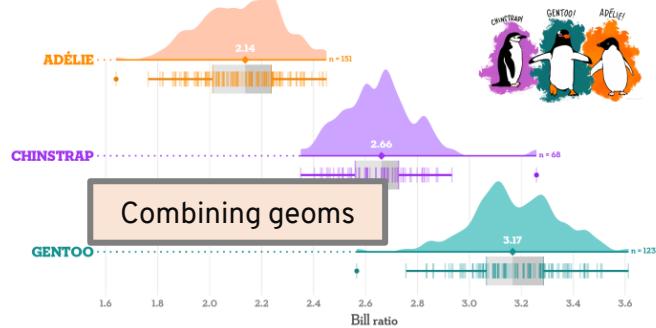


A. Scatterplot of bill length versus bill depth.

Color choice



B. Distribution of the bill ratio, estimated as bill length divided by bill depth



Note: In the original data, bill dimensions are recorded as "culmen length" and "culmen depth". The culmen is the dorsal (upper) ridge of a bird's bill.

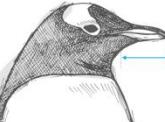
## Palmer Penguins Bill Length

Palmer Archipelago is a group of islands off the northwestern coast of the Antarctic Peninsula. The histograms show that females have shorter bills than males in every species.

ADELIE



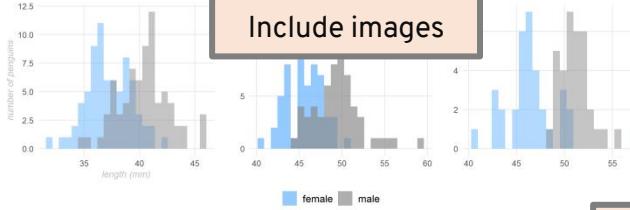
GENTOO



CHINSTRAP



Include images



Text styling

Measurements of body mass, bill length, bill depth, and flipper length in **Adelie**, **Gentoo**, and **Chinstrap** penguins, collected from 3 islands in the Palmer Archipelago, Antarctica.

Data: Dr. K. Gorman, Dr. A. Horst, & Dr. A. Hill. [allisonhorst/palmerpenguins](#)

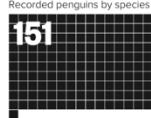
Visualization: Joseph Shaw / [@JosephShaw](#)

Spacing

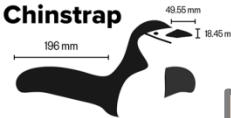
## Palmer Penguins

Median length of flipper, length and depth of bill, of 342 penguins recorded between 2007 and 2009

Adelie



Chinstrap



Gentoo



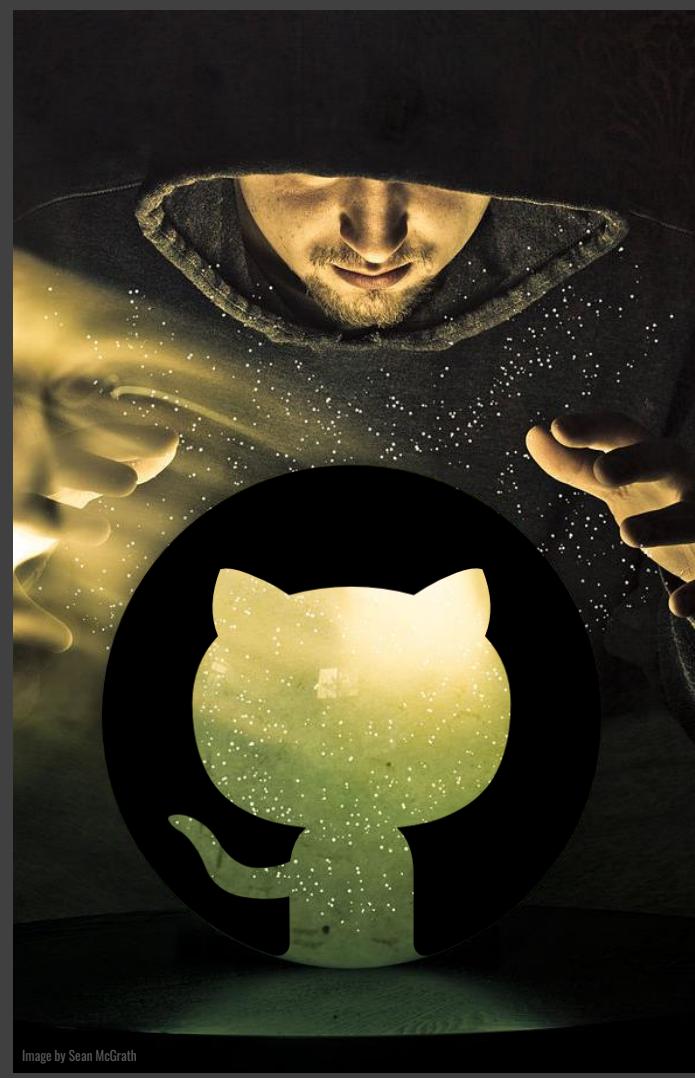
Unusual geoms



Source: Dr. Kristen Gorman and the Palmer Station, Antarctica LTER | Graphic: Georgios Karamanis

#TidyTuesday Contributions by

- ← myself
- ↖ Laura Navarro
- ↙ Joe Shaw
- ↓ Georgios Karamanis



Laura Navarro  
@LauraNavarroSol

Replying to @LauraNavarroSol

Code here! [github.com/lau-cloud/TidyTuesday](https://github.com/lau-cloud/TidyTuesday)

5:15 PM · Jul 31, 2020 · Twitter Web App

color choice

Joe Shaw  
@JosephShaw\_

Attempt #2 of this weeks #TidyTuesday, penguins labelled correctly this time...

Code: [tinyurl.com/yyswk2xt](https://tinyurl.com/yyswk2xt)  
HD: [github.com/josephshaw/tidyTuesday](https://github.com/josephshaw/tidyTuesday)

2:42 PM · Jul 28, 2020 · Twitter Web App

Spacing

e plots

#TidyTuesday Contributions by

- ← myself
- ↖ Laura Navarro
- ↙ Joe Shaw
- ↓ Georgios Karamanis

Georgios Karamanis  
@geokaramanis

Palmer penguins for #TidyTuesday, happy to get the penguins almost right with geom\_bspline\_closed() from {ggforce} 😊

code: [github.com/gkaramanis/tidyTuesday](https://github.com/gkaramanis/tidyTuesday)

#dataviz #RStats

2:21 PM · Jul 28, 2020 · Twitter Web App

# Tidy Tuesday

A weekly data project in R from the  
R4DS online learning community

#TIDYTUESDAY on Twitter • RFORDATA SCIENCE/TIDYTUESDAY on GitHub • R4DS on Slack

“I’m not looking to necessarily practice my skills as much as I am **to be inspired and know what I can do** based on what other people share.”

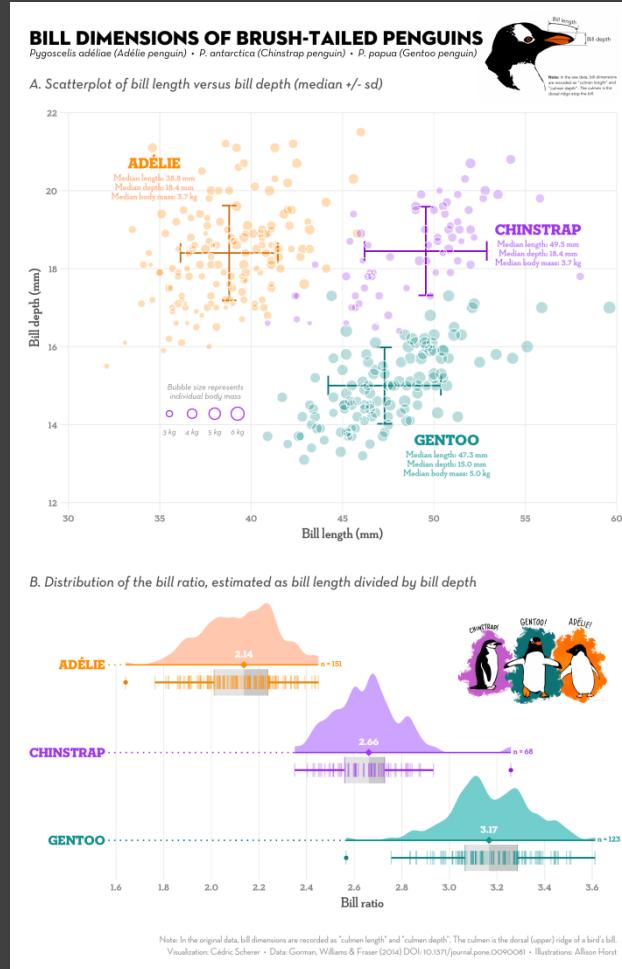
P3 in *Proc. ACM Hum.-Comput. Interact.* 37, 4, Article 111

“#TidyTuesday became a ‘choose your own adventure game’, which allowed participants **to ‘pursue something really weird’ beyond traditional visualizations.**”

I17 and P16 in *Proc. ACM Hum.-Comput. Interact.* 37, 4, Article 111



# The Showcase



with the help of  
{ggdist} & {ggttext} & {patchwork}

mjskay.github.io/ggdist

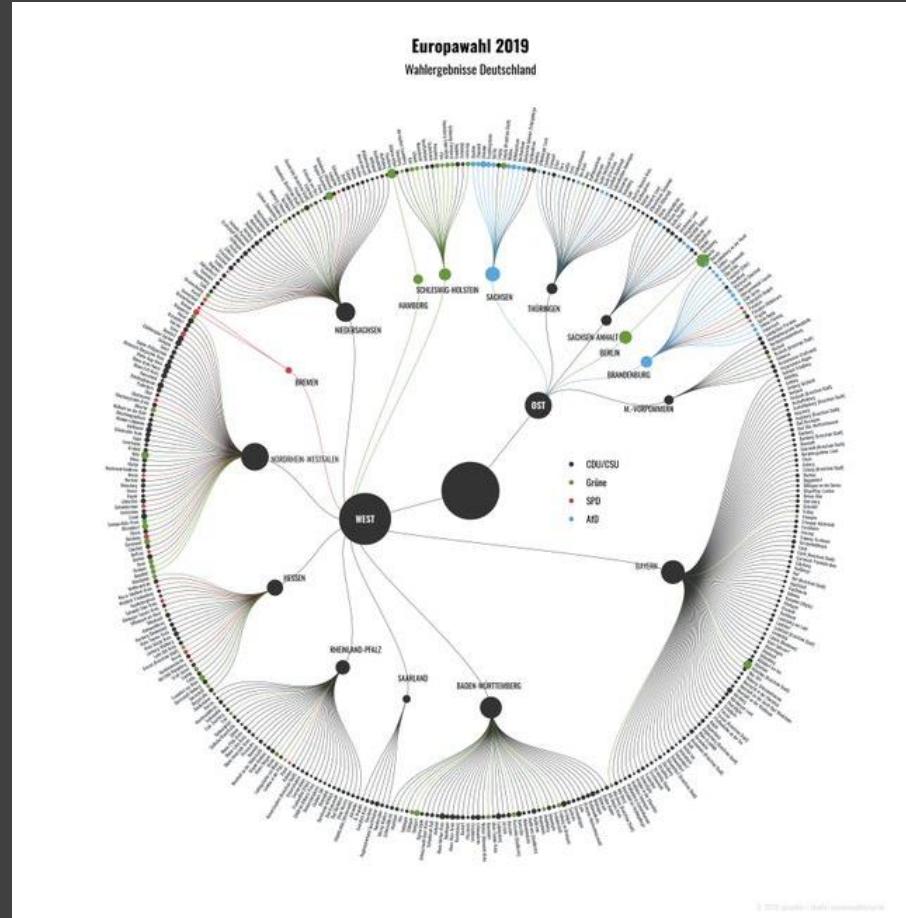
wilkelab.org/ggttext/

patchwork.data-imaginist.com

My Contribution to #TidyTuesday 2020/31

# The Showcase

Tree Diagrams + Networks



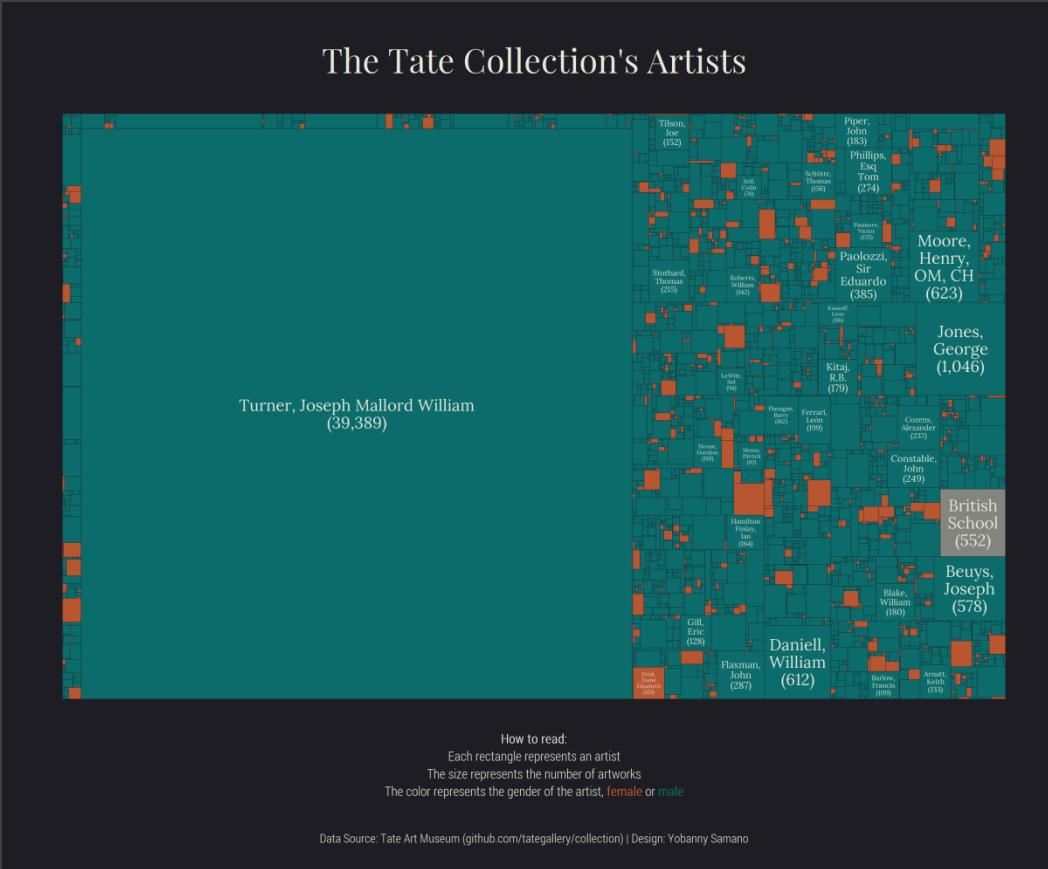
with the help of  
{ggraph}

graph.data-imaginist.com

Thorsten Sprenger

# The Showcase

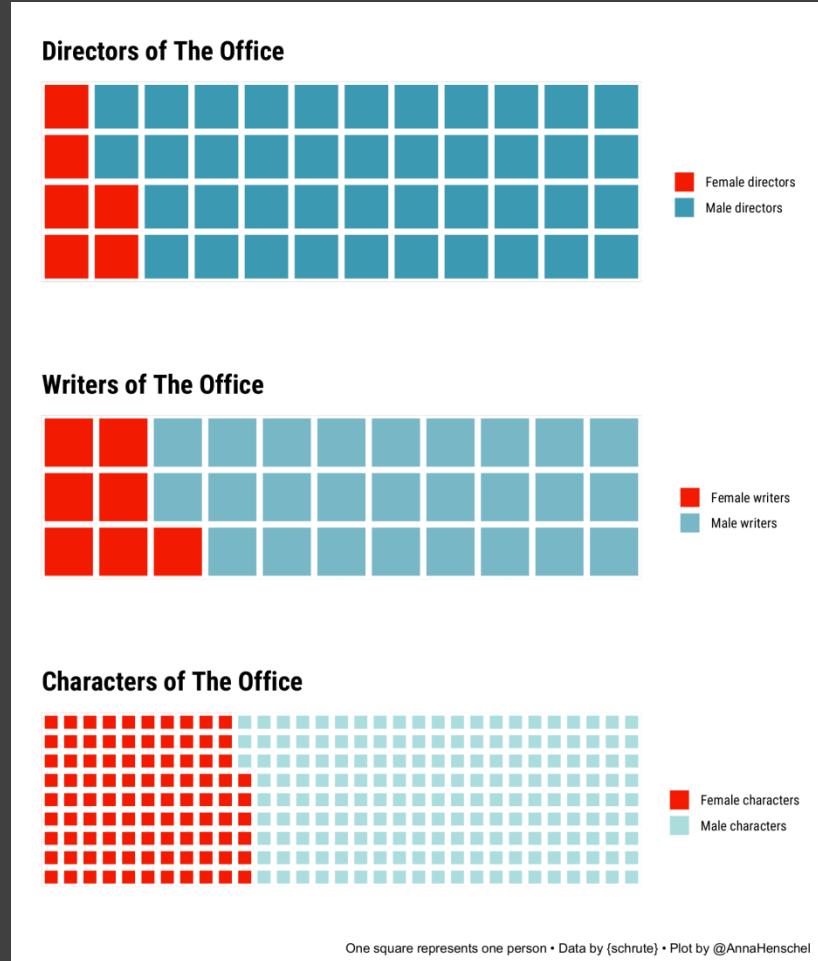
## Treemaps



with the help of  
{treemapify}  
[github.com/gwilkox/treemapify](https://github.com/gwilkox/treemapify)

# The Showcase

Waffle Charts



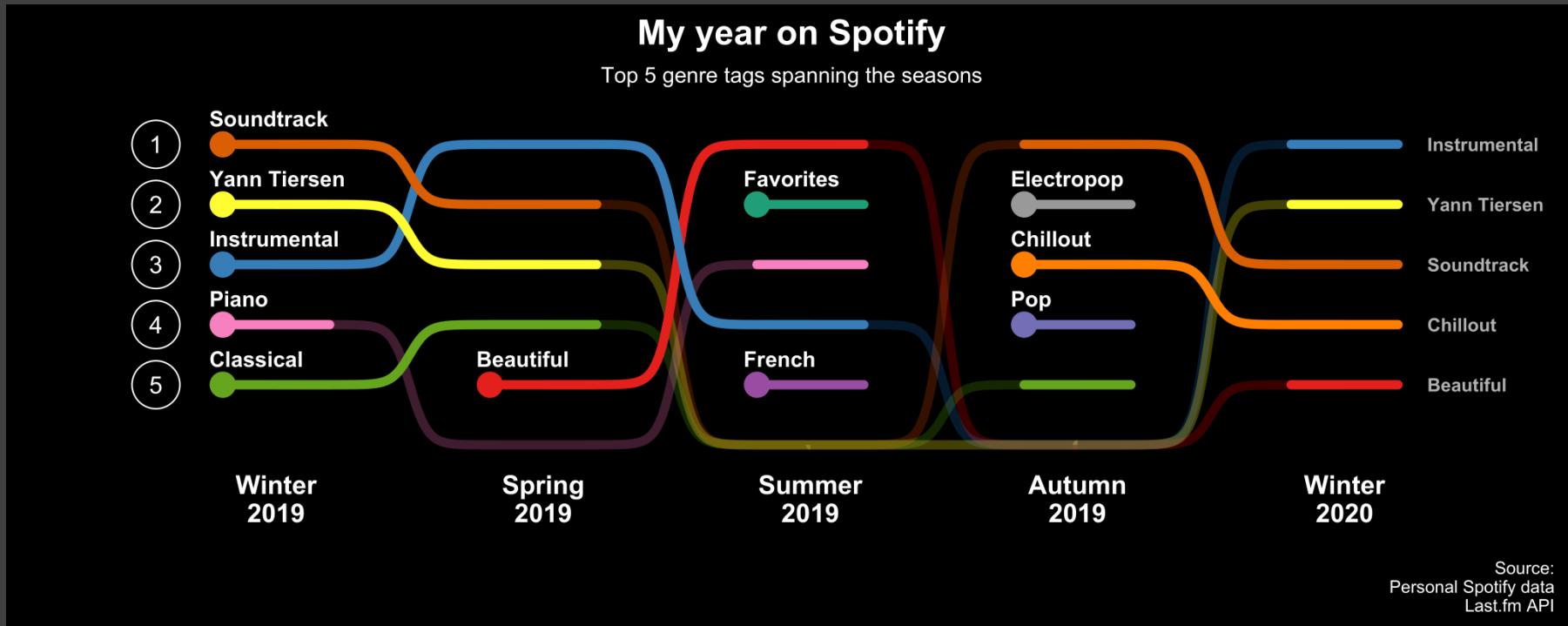
with the help of  
{waffle}

[github.com/hrbrmstr/waffle](https://github.com/hrbrmstr/waffle)

Anna Henschel, #TidyTuesday 2020/12

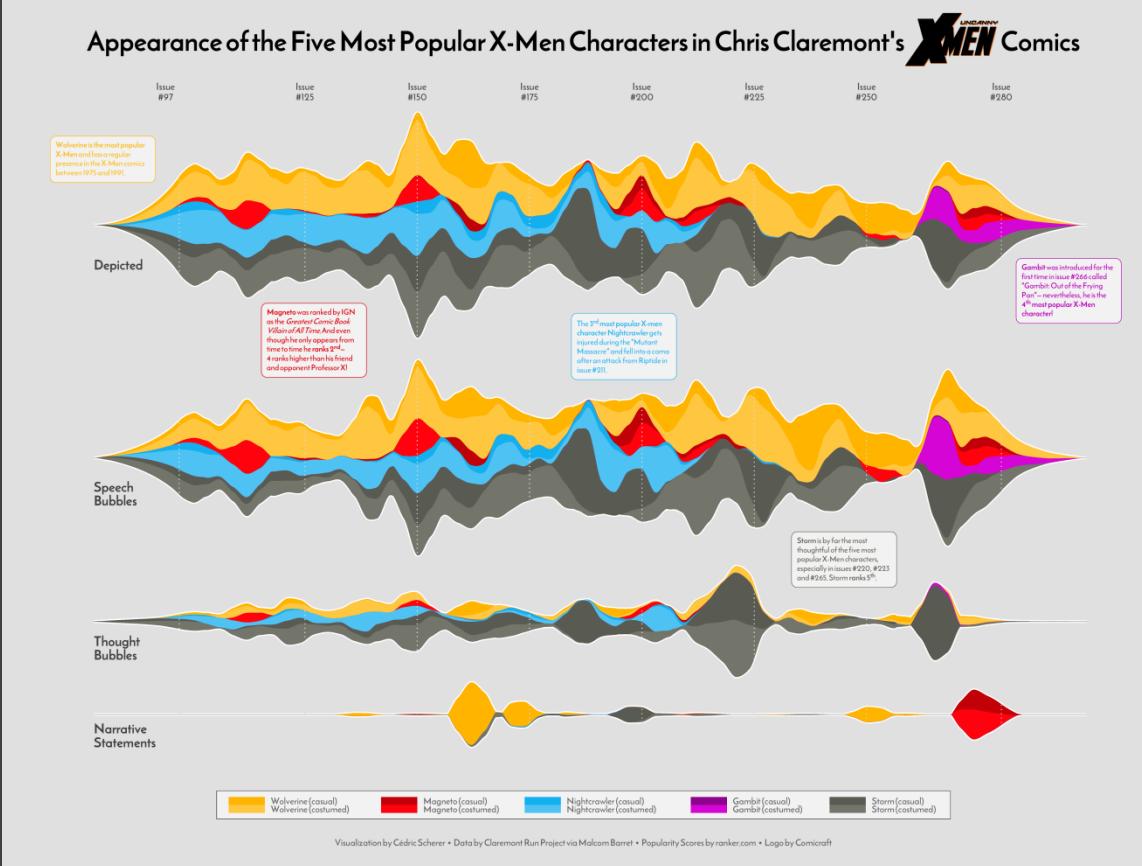
# The Showcase

Bump Charts



# The Showcase

## Streamgraphs

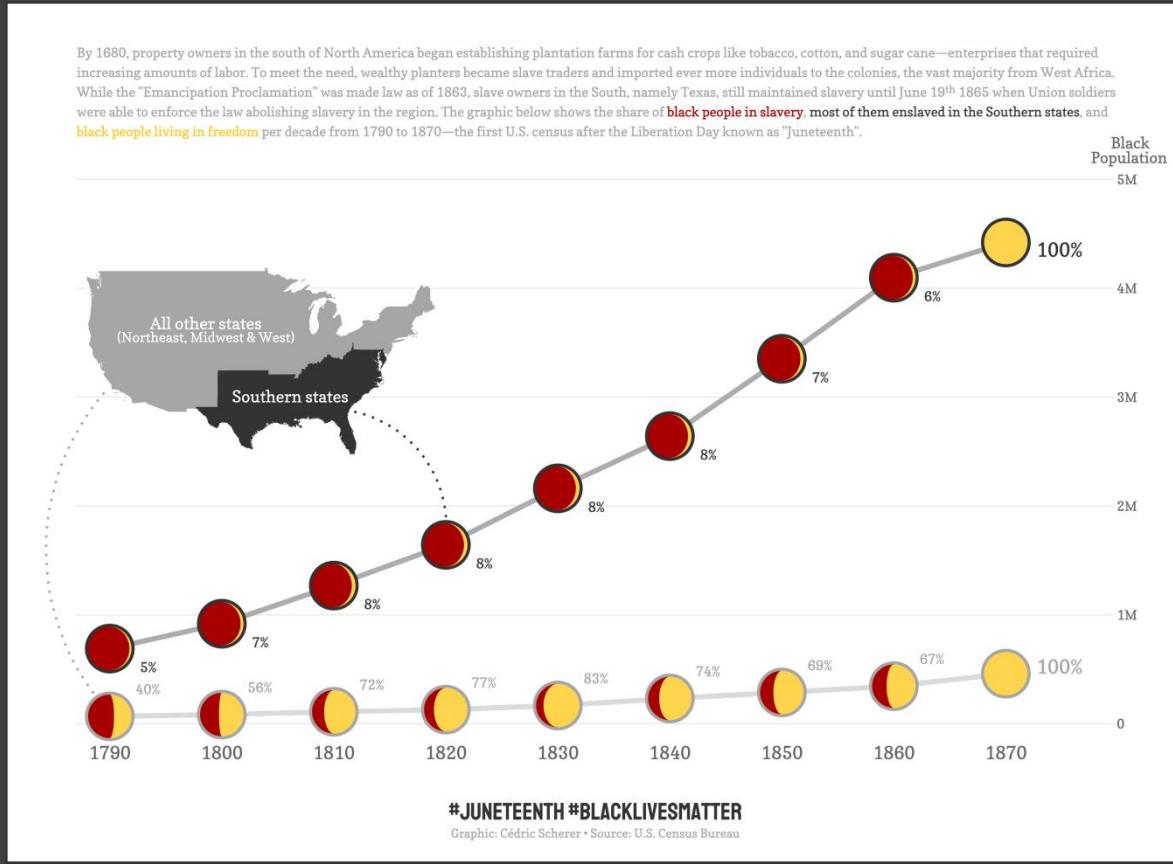


with the help of  
{ggstream}  
github.com/davidsjoberg/ggstream

My Contribution to #TidyTuesday 2020/27

# The Showcase

## Streamgraphs

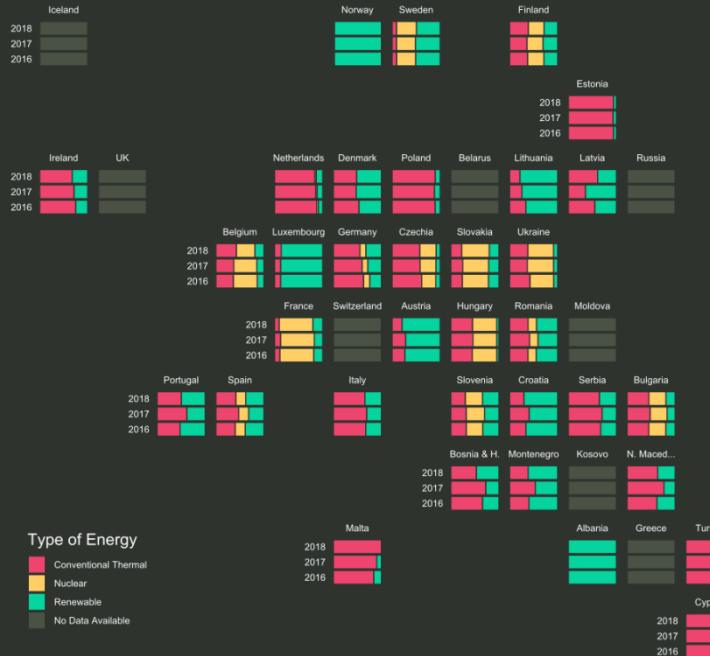


# The Showcase

Geofacets

## EUROPEAN ENERGY GENERATION

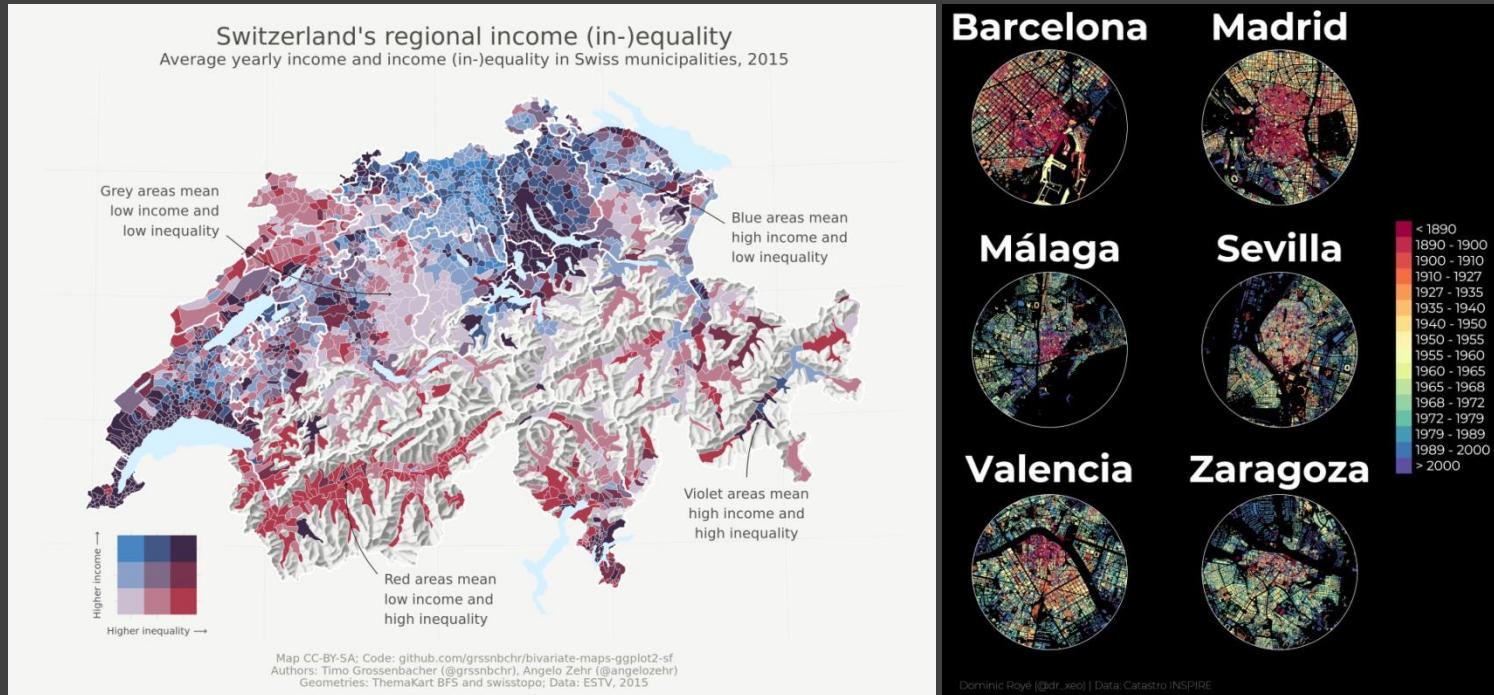
Each bar represents the **total energy generation** for each country per year.  
The colours represent the proportion of energy generated a) using **conventional thermal power plants**, which is to say those that use coal, oil or natural gas,  
b) using **nuclear power stations**, and c) using other **renewable sources**.



Data from 'Electricity generation statistics - First Results' (ec.europa.eu/eurostat/statistics-explained)  
Visualisation by Jack Davison (@JDavison\_ )  
Code found at [github.com/jack-davison](https://github.com/jack-davison)

# The Showcase

## Maps



# The Showcase

## Cartograms

What do most people die from?

Cardiovascular Diseases



Cancers

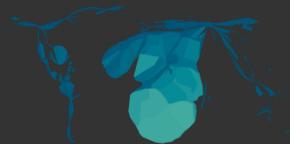


Diabetes



The leading causes of death across the world still vary significantly.  
These cartograms show causes of deaths in 2016 that exceeded 20 percent of total deaths in at least 1 country.

HIV Infections & Aids



Malaria Infections



Wars & Conflicts



The data refers to the specific cause of death, which is distinguished from risk factors for death, such as air pollution, diet and other lifestyle factors.

with the help of  
{cartogram}

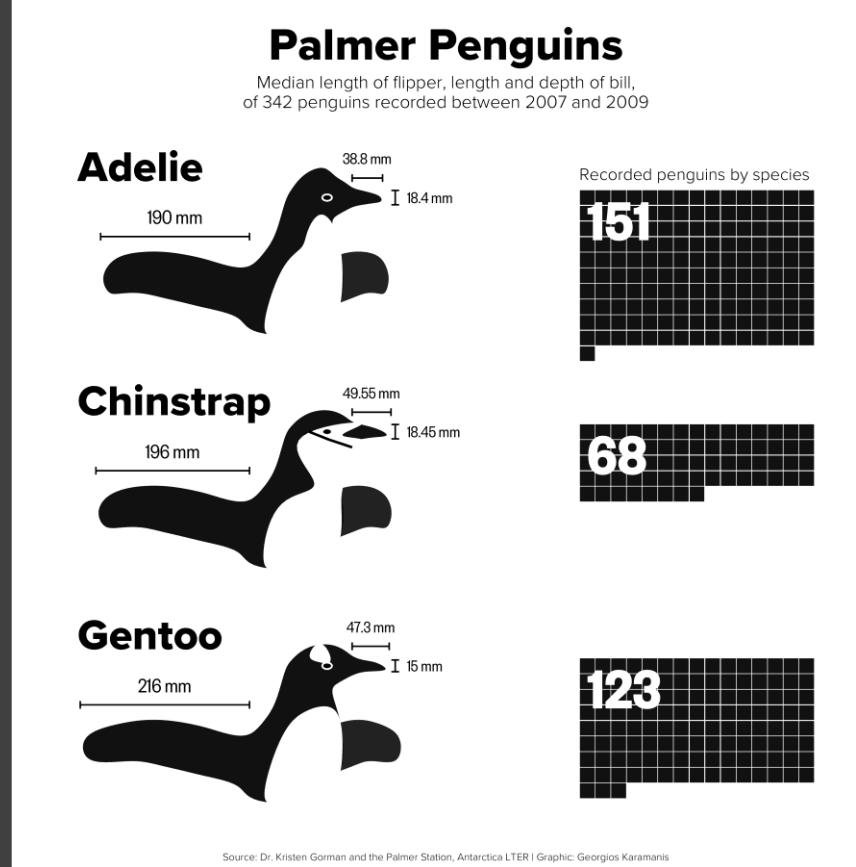
[github.com/sjewo/cartogram](https://github.com/sjewo/cartogram)

Visualization by Cédric Schefer • Data by OurWorldInData.org

My Contribution to the #30DayMapChallenge 2019, Topic *Polygons*

# The Showcase

Drawings



# The Showcase

Tables

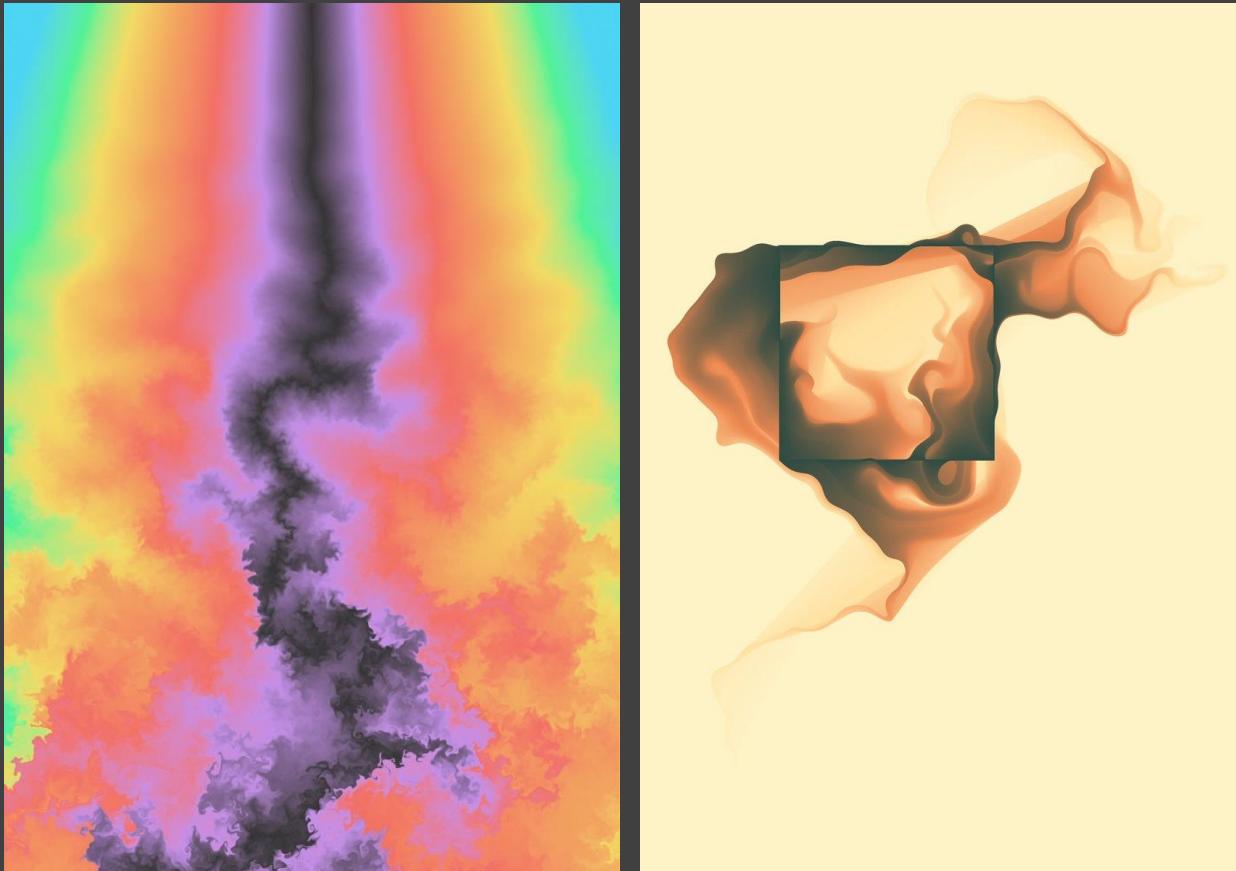
## Tour de France Winners

Source: alastairrushworth/tdf & kaggle.com/jaminliu | Graphic: Georgios Karamanis

YEAR	DISTANCE	WINNER	TEAM	AVERAGE SPEED	TOTAL TIME	YEAR
1985	0	8000 km				
1986	2428	René Gavet (FRA)	La Française	0	60 km/h	1986
1987	4488	Henri Cornet (FRA)	Gemini	25.3	+ 96.1	1987
1988	4754	Louis Trousselier (FRA)	Peugeot-Motul			1988
1989	4754	René Pottier (FRA)	Peugeot-Motul			1989
1990	4754	Lucien Petit-Breton (FRA)	Peugeot-Motul			1990
1991	4754	Ugo Fagioli (ITA)	Peugeot-Motul			1991
1992	5287	François Faber (LUX)	Acylon-Dunlop			1992
1993	5287	Gustave Serrigno (FRA)	Acylon-Dunlop			1993
1994	5287	Olivie Decroix (FRA)	Acylon-Dunlop			1994
1995	5287	Philippe Thys (BEL)	Peugeot-Motul	26.7	+ 197.9	1995
1996	5287	Philippe Thys (BEL)	Peugeot-Motul			1996
1997	5287	Philippe Thys (BEL)	Peugeot-Motul			1997
1998	5287	Philippe Thys (BEL)	Peugeot-Motul			1998
1999	5287	Philippe Thys (BEL)	Peugeot-Motul			1999
2000	5287	Philippe Thys (BEL)	Peugeot-Motul			2000
2001	5287	Philippe Thys (BEL)	Peugeot-Motul			2001
2002	5287	Philippe Thys (BEL)	Peugeot-Motul			2002
2003	5287	Philippe Thys (BEL)	Peugeot-Motul			2003
2004	5287	Philippe Thys (BEL)	Peugeot-Motul			2004
2005	5287	Philippe Thys (BEL)	Peugeot-Motul			2005
2006	5287	Philippe Thys (BEL)	Peugeot-Motul			2006
2007	5287	Philippe Thys (BEL)	Peugeot-Motul			2007
2008	5287	Philippe Thys (BEL)	Peugeot-Motul			2008
2009	5287	Philippe Thys (BEL)	Peugeot-Motul			2009
2010	5287	Philippe Thys (BEL)	Peugeot-Motul			2010
2011	5287	Philippe Thys (BEL)	Peugeot-Motul			2011
2012	5287	Philippe Thys (BEL)	Peugeot-Motul			2012
2013	5287	Philippe Thys (BEL)	Peugeot-Motul			2013
2014	5287	Philippe Thys (BEL)	Peugeot-Motul			2014
2015	5287	Philippe Thys (BEL)	Peugeot-Motul			2015
2016	5287	Philippe Thys (BEL)	Peugeot-Motul			2016
2017	5287	Philippe Thys (BEL)	Peugeot-Motul			2017
2018	5287	Philippe Thys (BEL)	Peugeot-Motul			2018
2019	5287	Philippe Thys (BEL)	Peugeot-Motul			2019
1915-1918 Tour suspended because of World War I						
1919	5583	Pierre Lefèvre (FRA)	La Sportive	24.1	+ 228.6	1919
1920	5583	Philippe Thys (BEL)	La Sportive			1920
1921	5583	León Scieur (BEL)	La Sportive			1921
1922	5386	Emile Georget (FRA)	Peugeot	24.2	+ 222.3	1922
1923	5386	Henri Pélissier (FRA)	Autosport-Hutchinson			1923
1924	5386	Ottavio Bottecchia (ITA)	Autosport-Hutchinson			1924
1925	5745	Lucien Buysse (BEL)	Autosport-Hutchinson	24.1	+ 238.7	1925
1926	5745	Nicolas Frantz (LUX)	Autosport-Hutchinson			1926
1927	5745	Nicolas Frantz (LUX)	Autosport-Hutchinson			1927
1928	5745	Nicolas Frantz (LUX)	Autosport-Hutchinson			1928
1929	5287	André Leducq (FRA)	Acylon-Dunlop	28.3	+ 184.7	1929
1930	5287	André Leducq (FRA)	Acylon-Dunlop			1930
1931	5287	André Leducq (FRA)	Acylon-Dunlop			1931
1932	6479	André Leducq (FRA)	France	29	+ 154.2	1932
1933	6479	Georges Lautru (FRA)	France			1933
1934	6479	Antonio Rigo (ITA)	France			1934
1935	4338	Rouelin Haes (BEL)	Belgium	38.7	+ 141.4	1935
1936	4338	Sylvère Maes (BEL)	Belgium			1936
1937	4338	Roger Lapebie (FRA)	France			1937
1938	4338	Sylvère Maes (BEL)	Belgium	31.6	+ 148.1	1938
1939	4338	Sylvère Maes (BEL)	Belgium			1939
1940-1946 Tour suspended because of World War II						
1947	4222	Jean Robic (FRA)	France	33.4	+ 167.2	1947
1948	4222	Giaco Bartali (ITA)	Italy			1948
1949	4222	Fausto Coppi (ITA)	Italy			1949
1950	4222	René Pottier (FRA)	France			1950
1951	4499	Hugo Koblet (SUI)	Switzerland	32.9	+ 142.5	1951
1952	4499	Pierre Goux (FRA)	France			1952
1953	4499	Louis Bobet (FRA)	France			1953
1954	4458	Louis Bobet (FRA)	France	33.2	+ 148.1	1954
1955	4458	Roger Wileman (FRA)	France			1955
1956	4458	Jacques Anquetil (FRA)	France	34.4	+ 135.7	1956
1957	4469	Charly Gaul (LUX)	Luxembourg			1957
1958	4469	Paul Deschampt (ESP)	Spain			1958
1959	4469	Gaston Menzini (ITA)	Italy	37.2	+ 112.1	1959
1960	4373	Jacques Anquetil (FRA)	France			1960
1961	4373	Jacques Anquetil (FRA)	France			1961
1962	4138	Jacques Anquetil (FRA)	France			1962
1963	4138	Jacques Anquetil (FRA)	France	36.5	+ 113.5	1963
1964	4138	Jacques Anquetil (FRA)	France			1964
1965	4138	Fausto Coppi (ITA)	Italy			1965
1966	4138	Fausto Coppi (ITA)	Italy	36.0	+ 117.6	1966
1967	4138	Fausto Coppi (ITA)	Italy			1967
1968	4138	Roger Pinçon (FRA)	Peugeot-GP-Michelin			1968
1969	4117	Eddy Merckx (BEL)	Peugeot-GP-Michelin	35.4	+ 116.3	1969
1970	4117	Eddy Merckx (BEL)	Peugeot-GP-Michelin			1970
1971	3846	Eddy Merckx (BEL)	Molteni	35.5	+ 108.3	1971
1972	3846	Eddy Merckx (BEL)	Molteni			1972
1973	3846	Eddy Merckx (BEL)	Molteni			1973
1974	3846	Eddy Merckx (BEL)	Molteni			1974
1975	3846	Eddy Merckx (BEL)	Molteni			1975
1976	3846	Eddy Merckx (BEL)	Molteni			1976
1977	3846	Eddy Merckx (BEL)	Molteni			1977
1978	3846	Eddy Merckx (BEL)	Molteni			1978
1979	3846	Eddy Merckx (BEL)	Molteni			1979
1980	3908	Bernard Thévenet (FRA)	Peugeot-Essaïe-Kiehl	34.9	+ 114.6	1980
1981	3908	Bernard Hinault (FRA)	Peugeot-Essaïe-Kiehl	34.1	+ 108.3	1981
1982	3753	Bernard Hinault (FRA)	Renault-Gitane-Campagnolo	39	+ 96.3	1982
1983	3753	Bernard Hinault (FRA)	Renault-Gitane-Campagnolo			1983
1984	4821	Laurent Fignon (FRA)	Renault-Gitane	35.9	+ 112.1	1984
1985	4821	Laurent Fignon (FRA)	Renault-Gitane			1985
1986	4821	Greg LeMond (USA)	La Vie Claire	36.6	+ 115.5	1986
1987	4231	Pedro Delgado (ESP)	Ad Merviel-W Cup-Bottecchia			1987
1988	4231	Pedro Delgado (ESP)	Z-Torosso	38.6	+ 99.7	1988
1989	3564	Greg LeMond (USA)	Team Telekom			1989
1990	3714	Miguel Indurain (ESP)	Banesto	58.7	+ 96	1990
1991	3714	Miguel Indurain (ESP)	Banesto			1991
1992	3714	Miguel Indurain (ESP)	Banesto			1992
1993	3714	Miguel Indurain (ESP)	Banesto			1993
1994	3714	Miguel Indurain (ESP)	Banesto			1994
1995	3714	Miguel Indurain (ESP)	Banesto			1995
1996	3714	Miguel Indurain (ESP)	Team Telekom	39.2	+ 96	1996
1997	3714	Miguel Indurain (ESP)	Team Telekom			1997
1998	3714	Miguel Indurain (ESP)	Team Telekom			1998
1999	3714	Miguel Indurain (ESP)	Team Telekom			1999
2000	3467	Lance Armstrong (USA)	U.S. Postal Service	48.3	+ 91.5	2000
2001	3467	Lance Armstrong (USA)	U.S. Postal Service			2001
2002	5772	Lance Armstrong (USA)	U.S. Postal Service	39.9	+ 82.1	2002
2003	5772	Lance Armstrong (USA)	U.S. Postal Service			2003
2004	5772	Lance Armstrong (USA)	U.S. Postal Service			2004
2005	3593	Oscar Pereiro (ESP)	U.S. Postal Service	41.7	+ 86.3	2005
2006	3593	Oscar Pereiro (ESP)	Caisse d'Épargne-Illes Balears			2006
2007	3593	Alberto Contador (ESP)	Discovery Channel	40.5	+ 87.9	2007
2008	3593	Alberto Contador (ESP)	Astana			2008
2009	3438	Cadel Evans (AUS)	Team BMC Racing Team			2009
2010	3438	Alberto Contador (ESP)	Team CSC			2010
2011	3438	Alberto Contador (ESP)	Team Sky	39.8	+ 86.2	2011
2012	3438	Alberto Contador (ESP)	Team Sky			2012
2013	3438	Chris Froome (GBR)	Team Sky			2013
2014	3668.5	Chris Froome (GBR)	Team Sky	40.7	+ 99	2014
2015	3668.5	Chris Froome (GBR)	Team Sky			2015
2016	3548	Chris Froome (GBR)	Team Sky			2016
2017	3548	Chris Froome (GBR)	Team Sky	41	+ 86.3	2017
2018	3548	Graeme Obree (GBR)	Team Sky			2018
2019	3548	Graeme Obree (GBR)	Team Sky			2019

# *The Showcase*

Generative Art



*with the help of*



Thomas Lin Pedersen



## More Text Labels, Less Bar Charts.

— More often used in 2020 — More often used in 2019

Compared to 2019, I used more geom's to add text labels in my #TidyTuesday scripts in 2020<sup>†</sup> while the number of bar charts per script decreased tremendously. Besides text labelling, points were by far the most used geometry (even though less used than in 2019). The most used extension packages were (ggtext) and (ggforce) but also (ggdist), (ggstream), and (gggibbous) played an important role for the first time in 2020. Interestingly, I did not use as many tiles and not any beeswarms at all.

Visualization: Cédric Scherer • Data: My 66 #TidyTuesday Scripts from 2019 and 2020<sup>†</sup>

<sup>†</sup> I extracted all functions starting with `geom` or `stat` from my Rmd files containing the code for all my #TidyTuesday contributions (thanks Georgios for the idea and script). For the contributions from 2019 ( $n = 26$ ) and 2020 ( $n = 40$ ) I calculated the frequency of usage per year for each geom/stat as times used divided by the number of contributions. Note that some geom's which usually appear together (e.g. `treemapify::geom_treemap` functions) or behave very similarly (e.g. `ggforce::geommark` functions) were grouped together.



## More Text Labels, Less Bar Charts.

— More often used in 2020 — More often used in 2019

Compared to 2019, I used more geom's to add text labels in my #TidyTuesday scripts in 2020<sup>†</sup> while the number of bar charts per script decreased tremendously. Besides text labelling, points were by far the most used geometry (even though less used than in 2019). The most used extension packages were (ggtext) and (ggforce) but also (ggdist), (ggstream), and (gggibbous) played an important role for the first time in 2020. Interestingly, I did not use as many tiles and not any beeswarms at all.

Visualization: Cédric Scherer • Data: My 66 #TidyTuesday Scripts from 2019 and 2020<sup>†</sup>

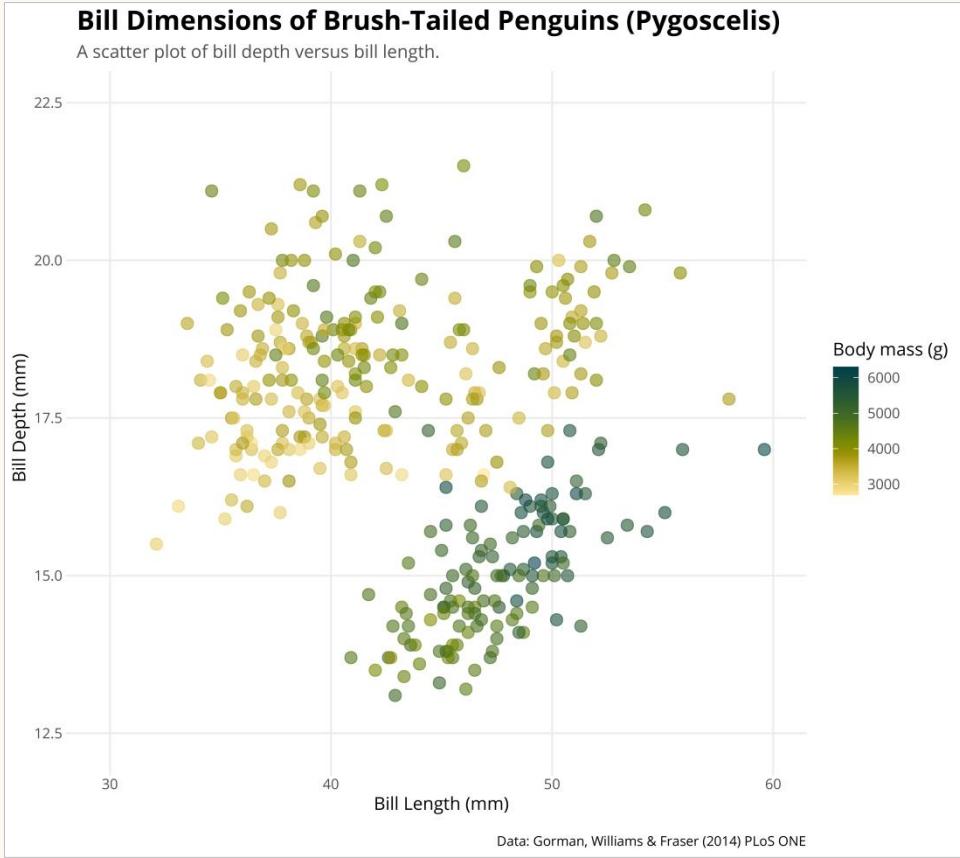
<sup>†</sup> I extracted all functions starting with `geom` or `stat` from my Rmd files containing the code for all my #TidyTuesday contributions (thanks Georgios for the idea and script). For the contributions from 2019 ( $n = 26$ ) and 2020 ( $n = 40$ ) I calculated the frequency of usage per year for each geomist as times used divided by the number of contributions. Note that some geom's which usually appear together (e.g. `treemapify::geom_treemap` functions) or behave very similarly (e.g. `ggforce::geommark` functions) were grouped together.

# {ggtext}

---

Improved Text Rendering Support

# {ggtext} Improved Text Rendering Support



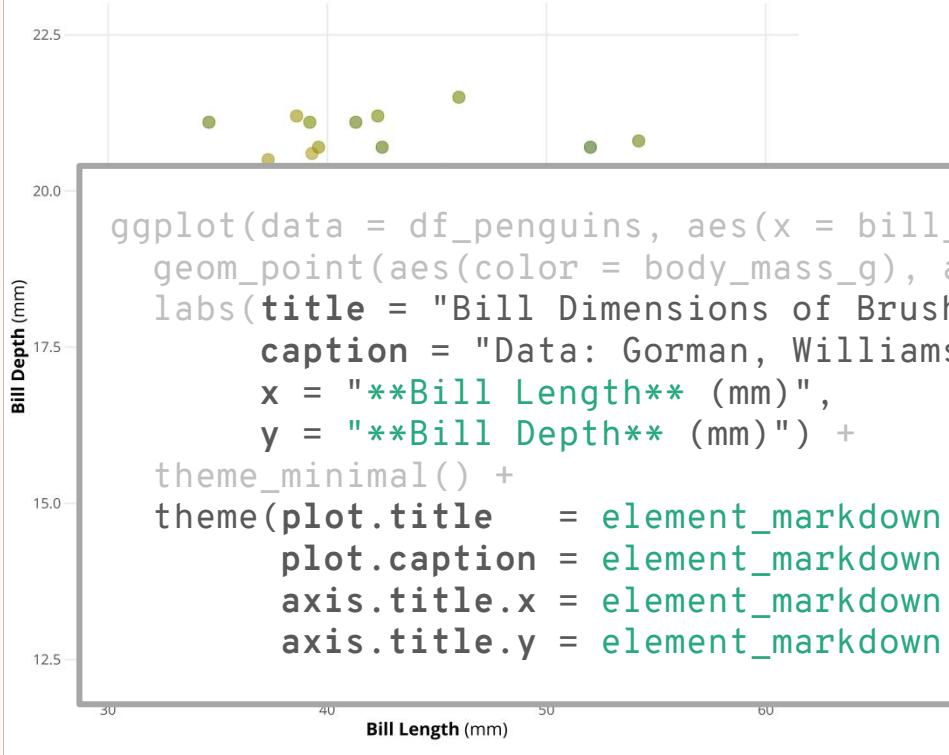
`element_markdown()`

- formatted text elements,  
e.g. titles, caption, axis text, striptext

# {ggtext} Improved Text Rendering Support

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



## element\_markdown()

→ formatted text elements,  
e.g. titles, caption, axis text, striptext

# {ggtext} Improved Text Rendering Support

```
<i style='color:#28A87D;'>Pygoscelis</i>
```

**Bill Dimensions of Brush-Tailed Penguins** *Pygoscelis*

```
<b style='font-size:32pt;font-family:blacksword;'>Pygoscelis</b>
```

**Bill Dimensions of Brush-Tailed Penguins** *Pygoscelis*

```
<img src='https://cedricscherer.com/img/pygoscelis.jpg', width='10' />
```

**Bill Dimensions of Brush-Tailed Penguins**



# Chats about Friends and their Past, Present, and Future Partners

Mentions of the main characters and their most popular partners in dialogues\* during the ten seasons of Friends.



Font Color  
Font Face

**<span style='color:#d86b1d;'>Ross & Rachel</span>**  
**<span style='color:#ba2a22;'>Rachel & Barry</span>**

\*For each of the 67,373 dialogues in 236 episodes it was determined whether the two names occur in the same text.  
The area and luminance of the squares is mapped to the number of overall mentions of the two names per season.

Visualization by Cédric Scherer • Data by Emil Hvitfeldt via the [\(friends\) R package](#)

Contribution to #TidyTuesday 2020/37

## Chats about Friends and their Past, Present, and Future Partners

Mentions of the main characters and their most popular partners in dialogues\* during the ten seasons of Friends.



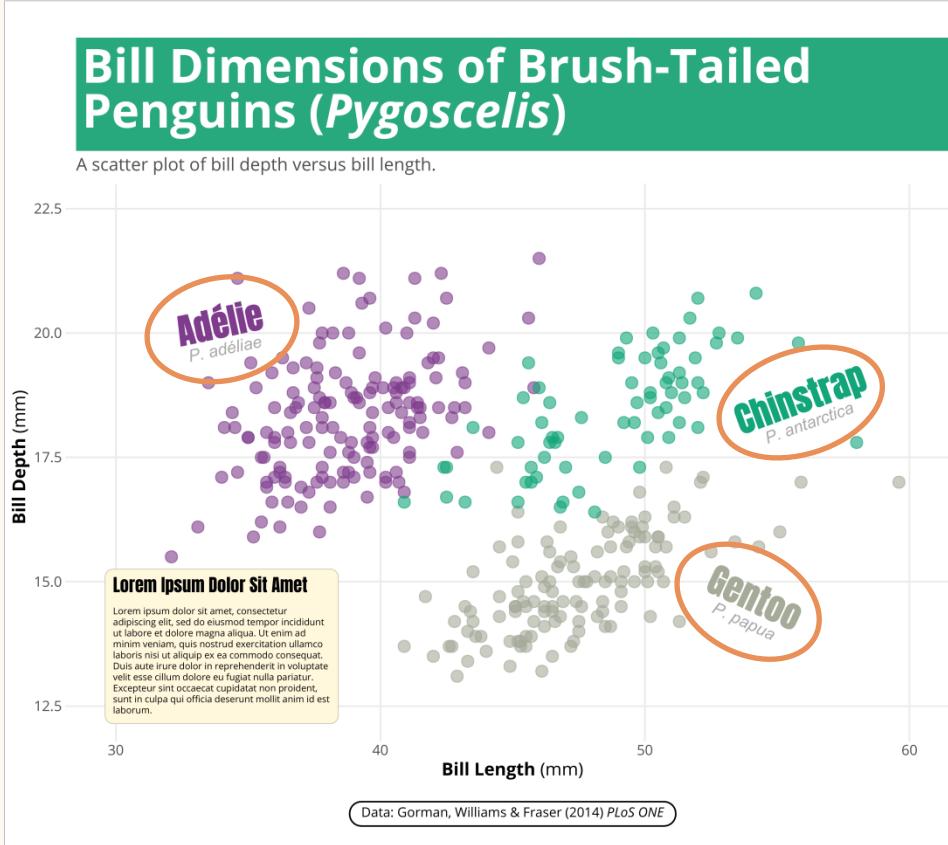
Font Color  
Font Face

```
label = if_else(
  key %in% c("Ross & Rachel", "Rachel & Joey", "Monica & Chandler"),
  glue::glue("<b style='color:{color};'>{partners}</b>"),
  stringr::str_replace(partners, key,
    glue::glue("<b style='color:{color};'>{key}</b>"))
)
```

partners	key	color	label
Ross & Rachel	Ross & Rachel	#d86b1d	<b style='color:#d86b1d;'>Ross & Rachel</b>
Chandler & Janice	Chandler	#48508c	<b style='color:#48508c;'>Chandler</b> & Janice
Ross & Carol	Ross	#f6ab18	<b style='color:#f6ab18;'>Ross</b> & Carol

Contribution to #TidyTuesday 2020/37

# {ggtext} Improved Text Rendering Support



`element_markdown()`

- formatted text elements,  
e.g. titles, caption, axis text, striptext

`geom_richtext()`

- formatted text labels with 360° rotation

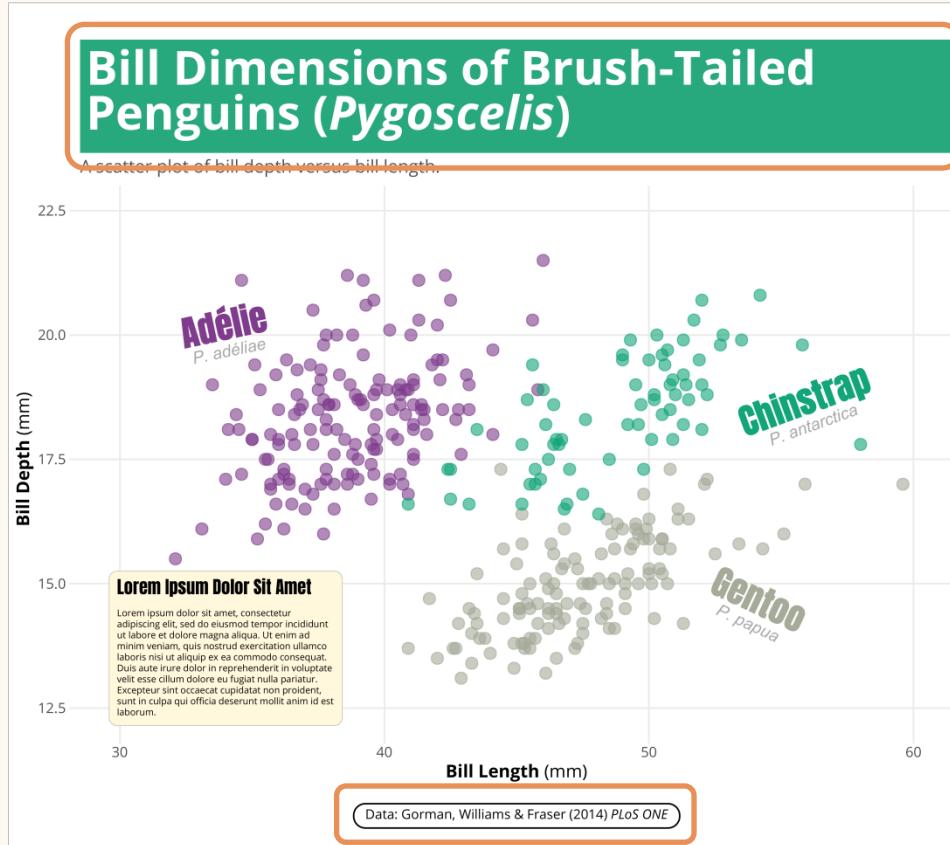
`element_textbox()` and  
`element_textbox_simple()`

- formatted text boxes with word wrapping

`geom_textbox()`

- formatted text boxes with word wrapping

# {ggttext} Improved Text Rendering Support



## element\_markdown()

- formatted text elements,  
e.g. titles, caption, axis text, striptext

## geom\_richtext()

- formatted text labels with 360° rotation

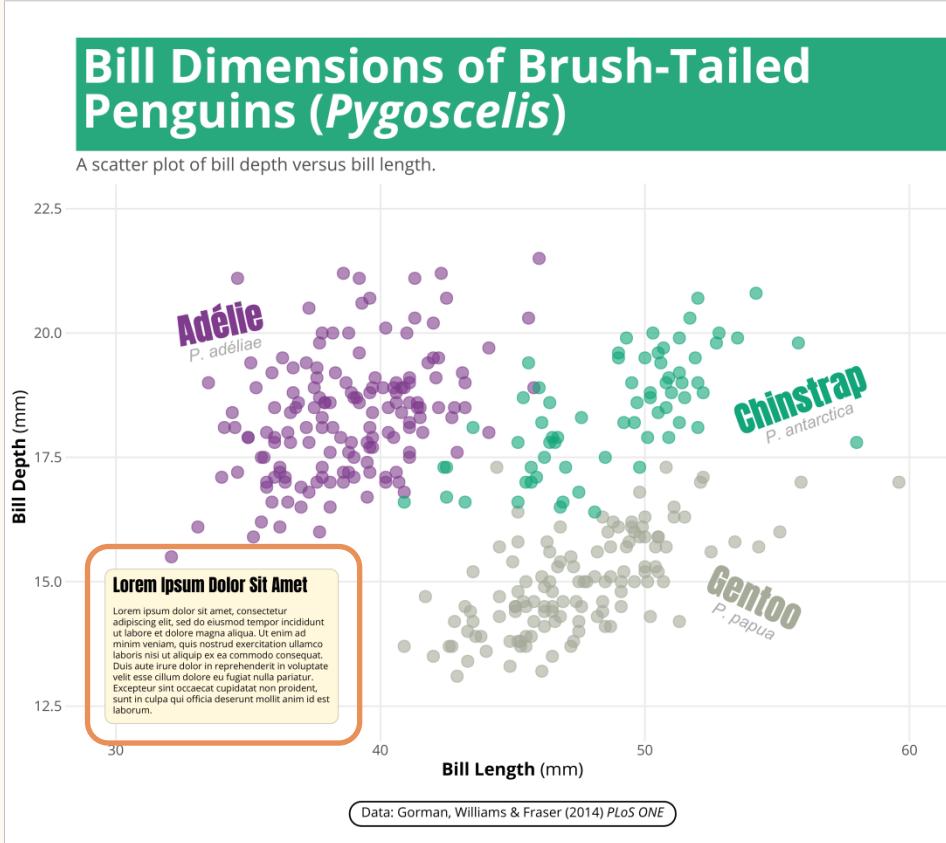
`element_textbox()` and  
`element_textbox_simple()`

- formatted text boxes with word wrapping

```
geom_textbox()
```

- formatted text boxes with word wrapping

# {ggtext} Improved Text Rendering Support



`element_markdown()`

→ formatted text elements,  
e.g. titles, caption, axis text, striptext

`geom_richtext()`

→ formatted text labels with 360° rotation

`element_textbox()` and  
`element_textbox_simple()`

→ formatted text boxes with word wrapping

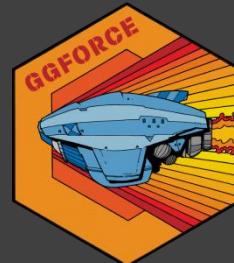
`geom_textbox()`

→ formatted text boxes with word wrapping

# {ggforce}

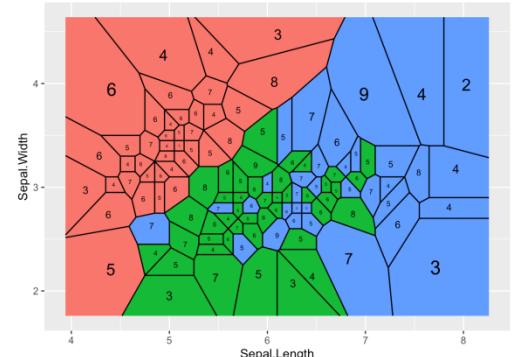
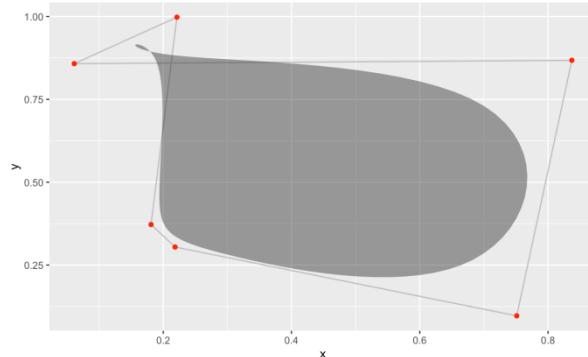
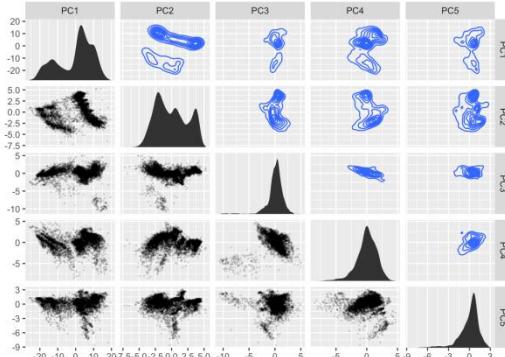
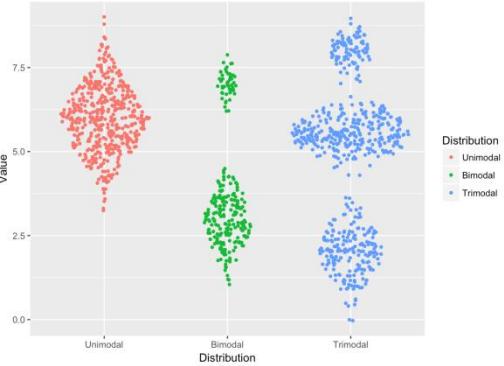
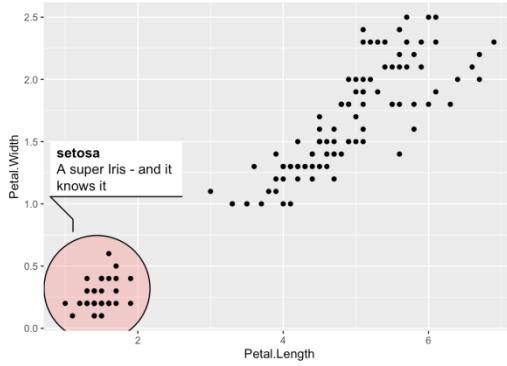
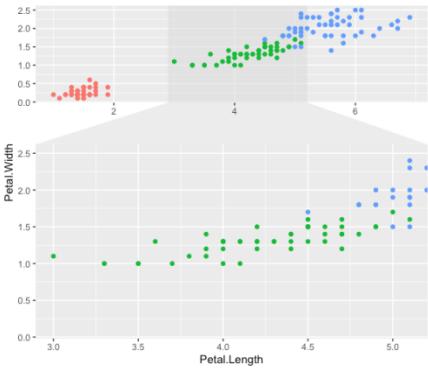
---

Providing Missing Functionality to `ggplot2`



[ggforce.data-imaginist.com](http://ggforce.data-imaginist.com)

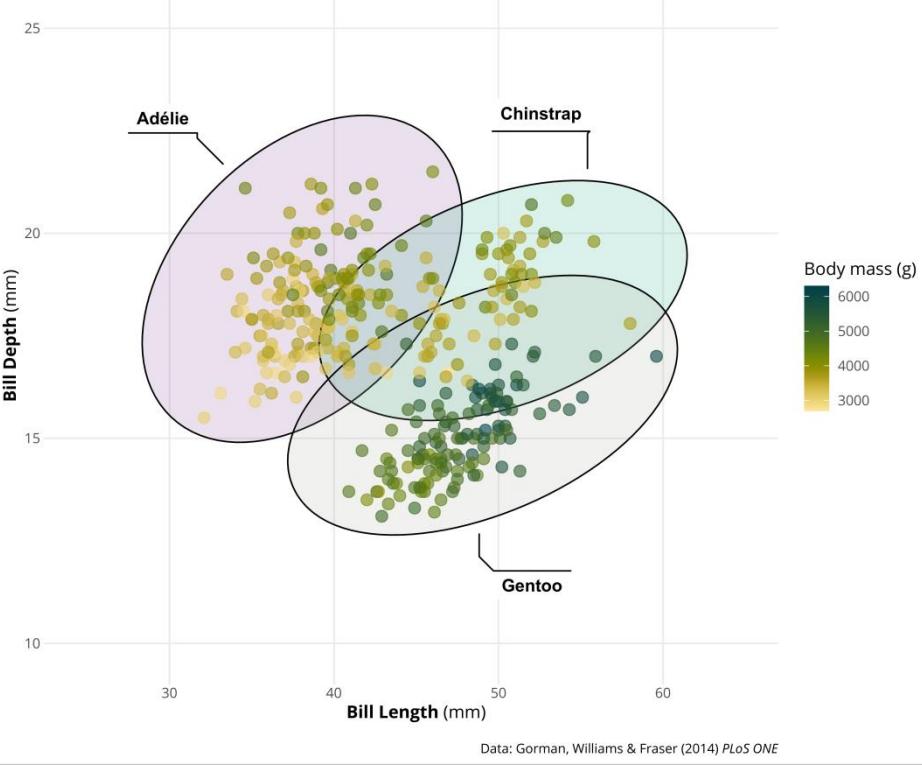
# {ggforce} Providing Missing Functionality



# {ggforce} Fancy Annotations

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



`geom_mark_*`

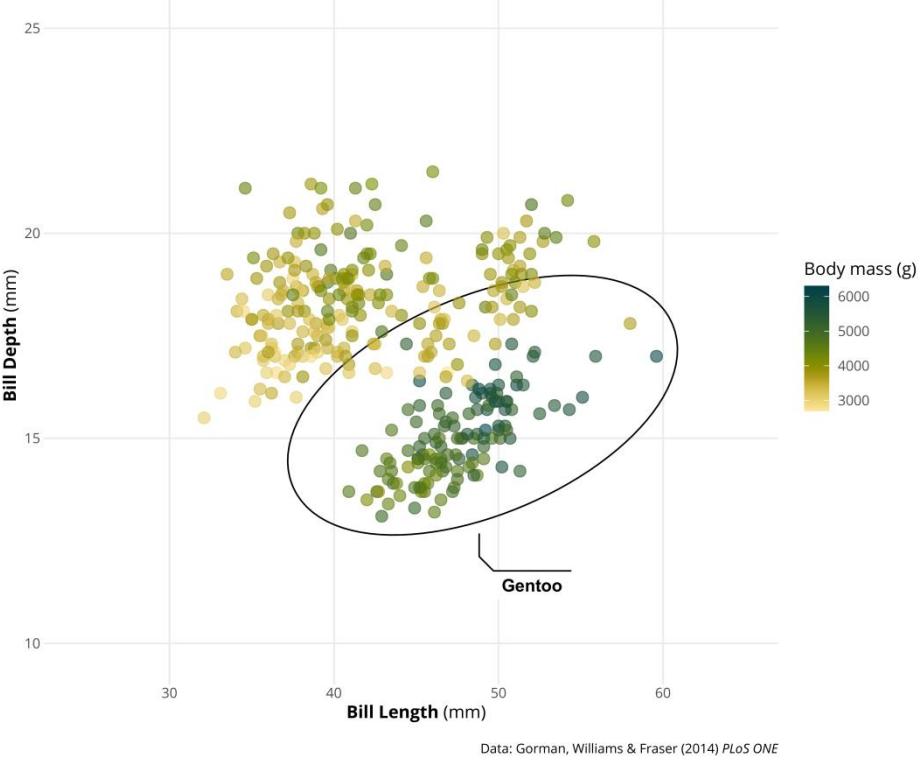
→ advanced labels for single or multiple points

```
geom_mark_ellipsoid(aes(fill = species, label = species))
```

# {ggforce} Fancy Annotations

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



```
geom_mark_ellipsoid(aes(fill = species, label = species,  
filter = species == 'Gentoo'))
```

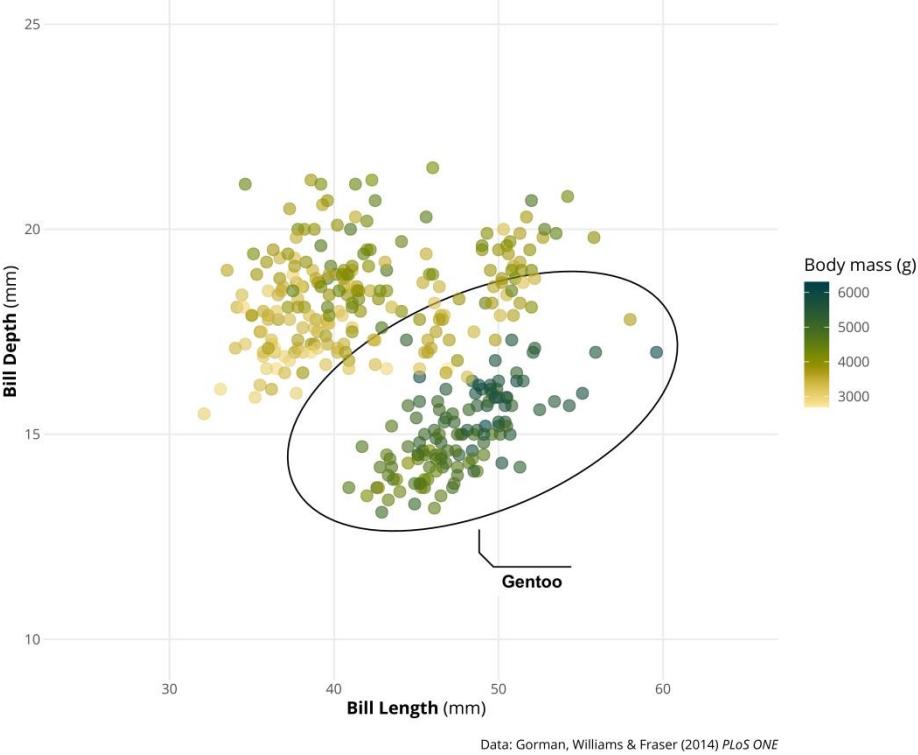
## geom\_mark\_\*

- advanced labels for single or multiple points
- show all groups or highlight interesting parts

# {ggforce} Fancy Annotations

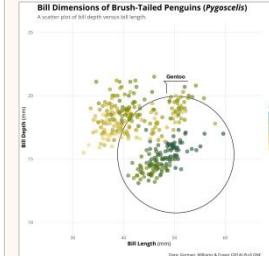
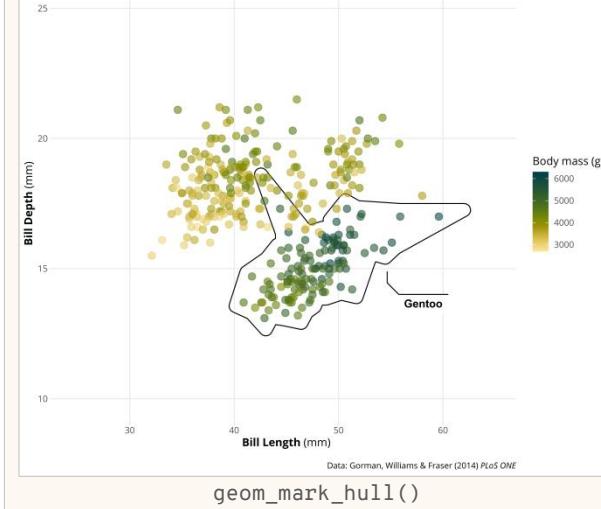
## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.

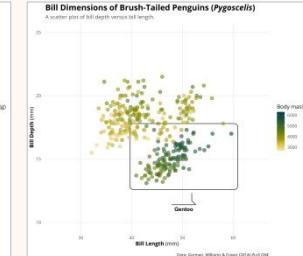


## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

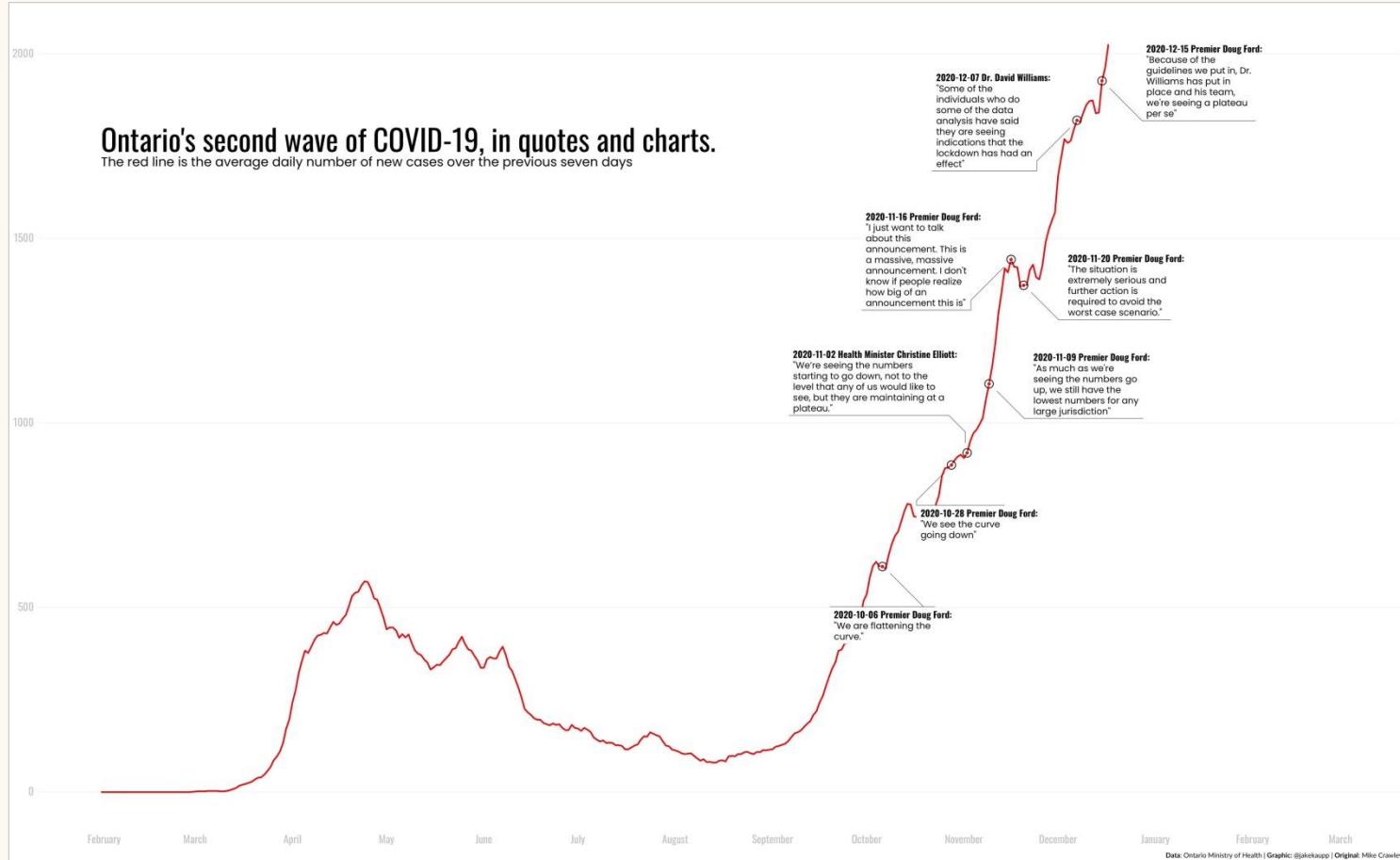
A scatter plot of bill depth versus bill length.



`geom_mark_circle()`

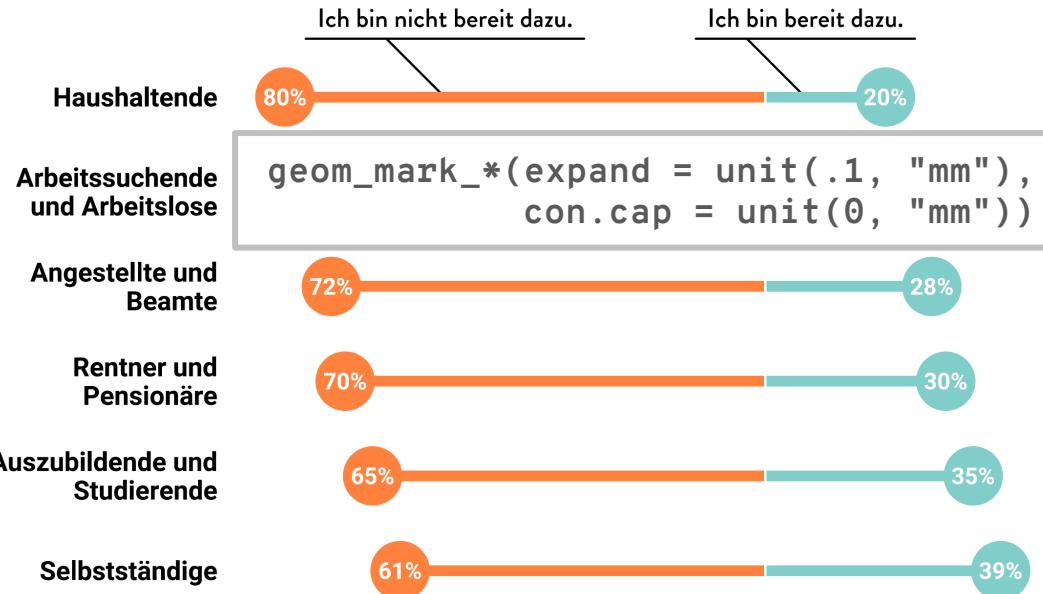


`geom_mark_rect()`



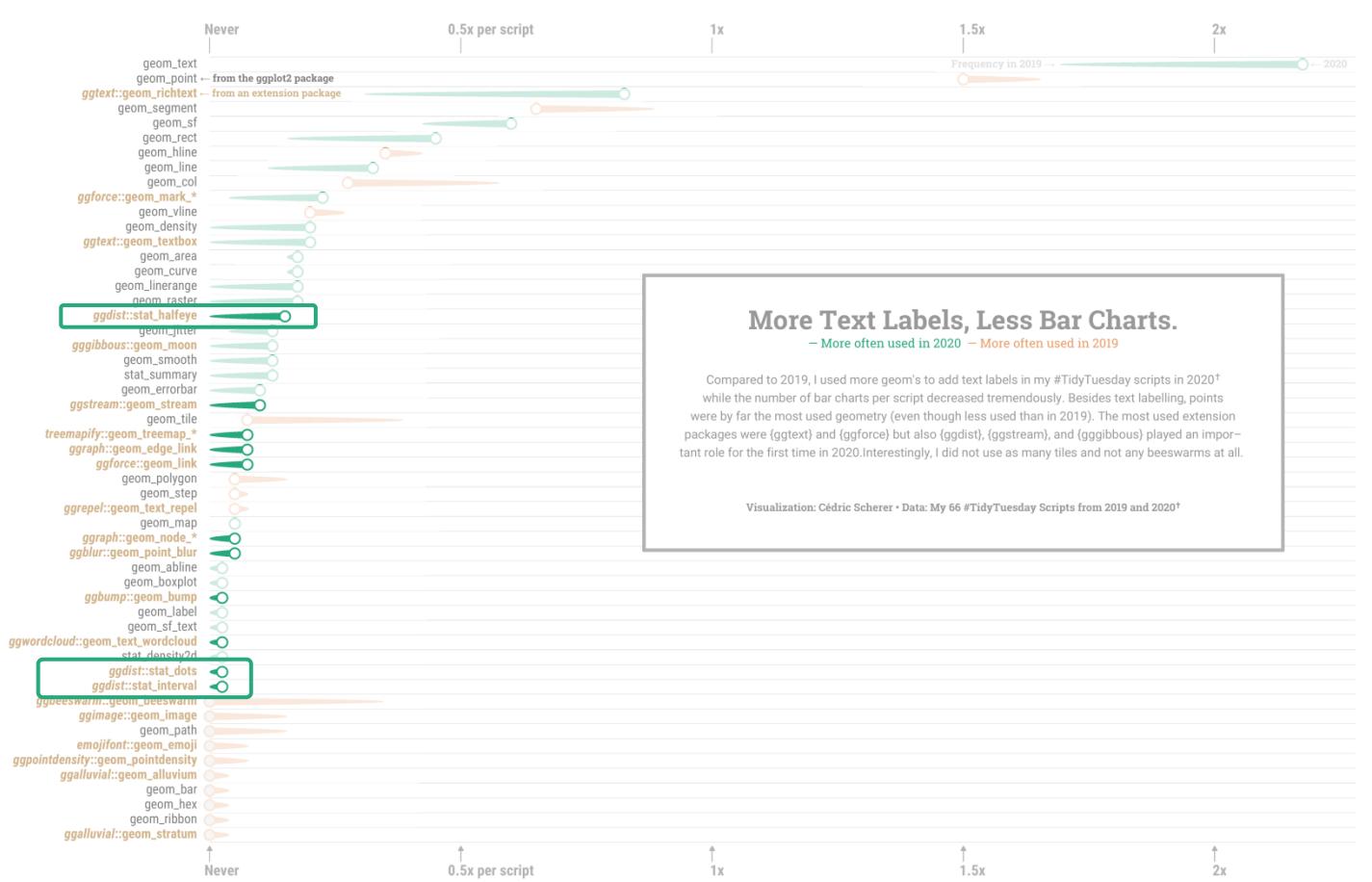
Jake Kaupp

Die Präsidentin der EU Ursula von der Leyen bittet "finanziell nicht notleidende Kunden" ihr Recht auf Rückerstattung aus Solidarität nicht in Anspruch zu nehmen.



Basierend auf 1057 Antworten auf eine Umfrage von KUENDIGUNG.ORG

Customer survey Kuendigung.org  
(kuendigung.org/studien/verbraucherumfrage-zur-zukunft-nach-der-krise)



## More Text Labels, Less Bar Charts.

— More often used in 2020 — More often used in 2019

Compared to 2019, I used more geom's to add text labels in my #TidyTuesday scripts in 2020<sup>†</sup> while the number of bar charts per script decreased tremendously. Besides text labelling, points were by far the most used geometry (even though less used than in 2019). The most used extension packages were (ggtext) and (ggeforce) but also (ggdist), (ggstream), and (gggibbous) played an important role for the first time in 2020. Interestingly, I did not use as many tiles and not any beeswarms at all.

Visualization: Cédric Scherer • Data: My 66 #TidyTuesday Scripts from 2019 and 2020<sup>†</sup>

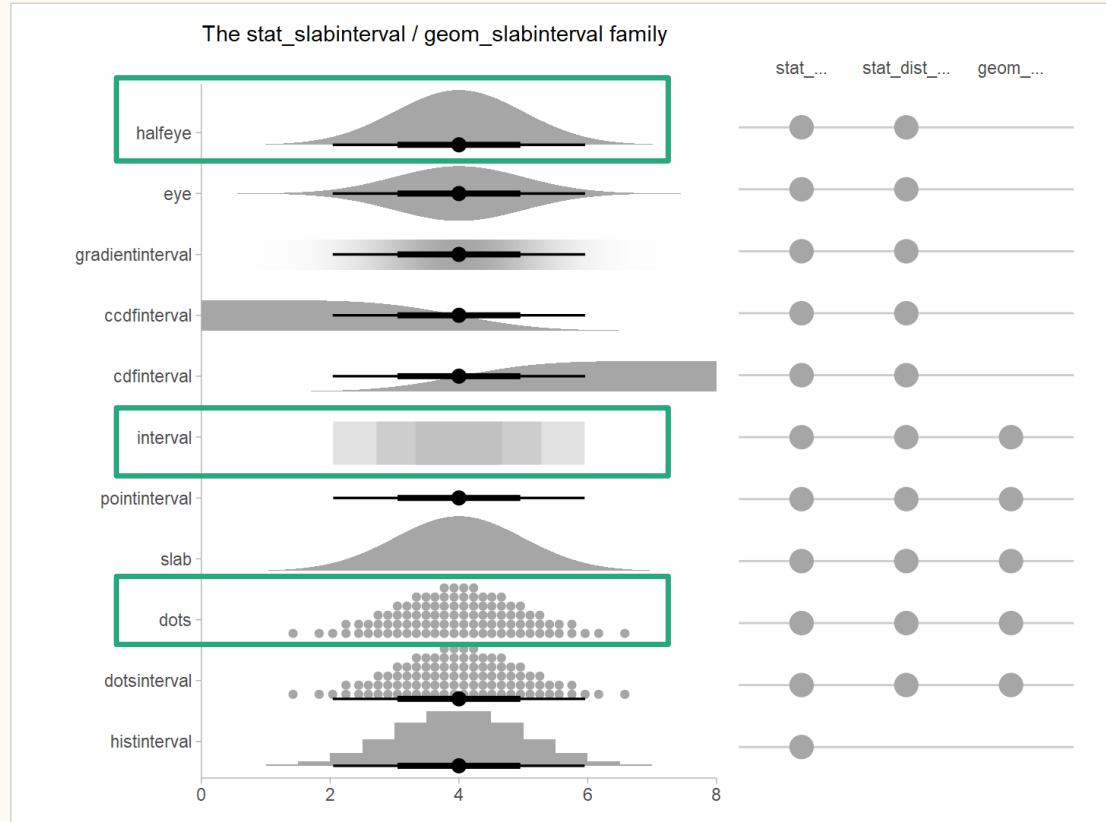
<sup>†</sup> I extracted all functions starting with `geom` or `stat` from my Rmd files containing the code for all my #TidyTuesday contributions (thanks Georgios for the idea and script). For the contributions from 2019 ( $n = 26$ ) and 2020 ( $n = 40$ ) I calculated the frequency of usage per year for each geom/stat as times used divided by the number of contributions. Note that some geom's which usually appear together (e.g. `treemapify::geom_treemap` functions) or behave very similarly (e.g. `ggforce::geommark` functions) were grouped together.

# {ggdist}

---

Visualizations of Distributions and Uncertainty

# {ggdist} Visualizations of Distributions and Uncertainty



# Not my cup of coffee...

Each dot depicts one coffee bean rated by Coffee Quality Institute's trained reviewers. In addition, the multiple interval stripes show where 25%, 50%, 95%, and 100% of the beans fall along the rating gradient from 0 to 100 points. The rated coffee beans range from 59.8 points (Guatemala) to 89.9 (Ethiopia). Only countries of origin with 25 or more tested beans are shown. The red empty triangle marks the minimum rating, the black filled triangle indicates each country's median score.

Visualization by Cédric Scherer

Coffee stain: © paperwerk.

60 POINTS

70 POINTS

## GUATEMALA

△ 59.8 POINTS  
The coffee bean with the lowest rating has its origin in Guatemala.



## NICARAGUA

△ 63.1 POINTS

## COLOMBIA

△ 72.8 POINTS

The best coffee—in terms of both median and maximum rating—is shipped to you from Ethiopia!

## ETHIOPIA

△ 80.3 POINTS

The best coffee—in terms of both median and maximum rating—is shipped to you from Ethiopia!

## KENYA

△ 79.8 POINTS

△ 84.6 POINTS

## UGANDA

△ 80.5 POINTS

△ 83.2 POINTS

## COSTA RICA

△ 71.8 POINTS

80 POINTS

90 POINTS

## UNITED STATES

△ 73.7 POINTS

△ 82.8 POINTS

## BRAZIL

△ 73.2 POINTS

△ 82.4 POINTS

## TANZANIA

△ 80.3 POINTS

△ 82.2 POINTS

## TAIWAN

△ 77.7 POINTS

△ 81.9 POINTS

## HONDURAS

△ 69.2 POINTS

△ 81.7 POINTS

## MEXICO

△ 68.3 POINTS

△ 81.6 POINTS

With 218 tested beans, Mexico is the country with the most reviews.

Contribution to #TidyTuesday 2020/28

# COLOMBIA

△ 72.8 POINTS

# KENYA

△ 79.8 POINTS

stat\_dots()

stat\_interval()

▲ 84.6 POINTS

geom\_text()  
geom\_point()

△ 80.5 POINTS

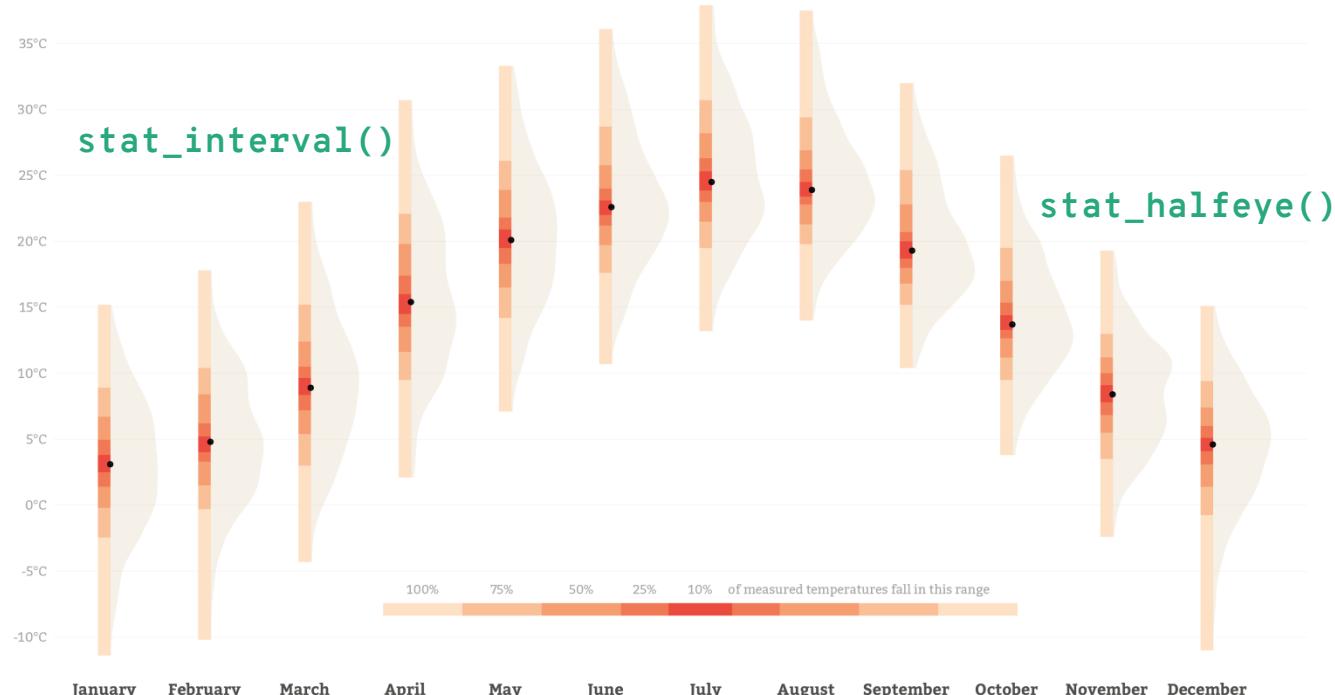
# UGANDA

▲ 83.2 POINTS

Contribution to #TidyTuesday 2020/28

# Daily Temperatures in Berlin, Germany

Range and distribution of maximum daily temperatures in Celsius per month from 2000 to 2018 measured in Berlin-Dahlem, Germany



Contribution to the SWD Challenge September 2019

# {ggplot2}

---

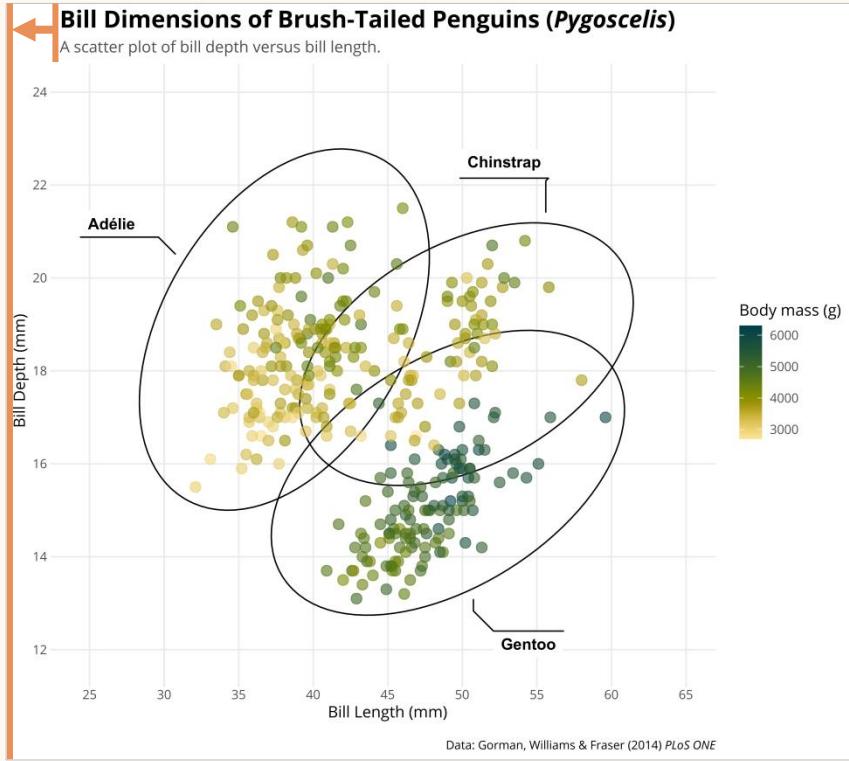
Details You May Not Know Yet  
(or have never thought about)



[ggplot2.tidyverse.org](https://ggplot2.tidyverse.org)

# Left-Aligned Title?

```
theme(plot.title = element_text(hjust = 0))
```



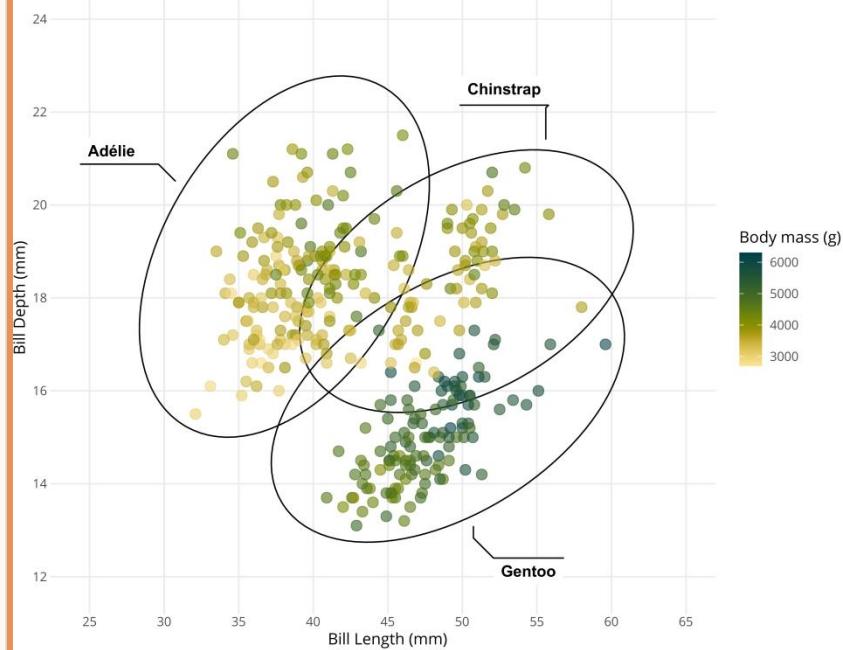
# Left-Aligned Title

```
theme(plot.title.position = 'panel')
```

Default

Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.

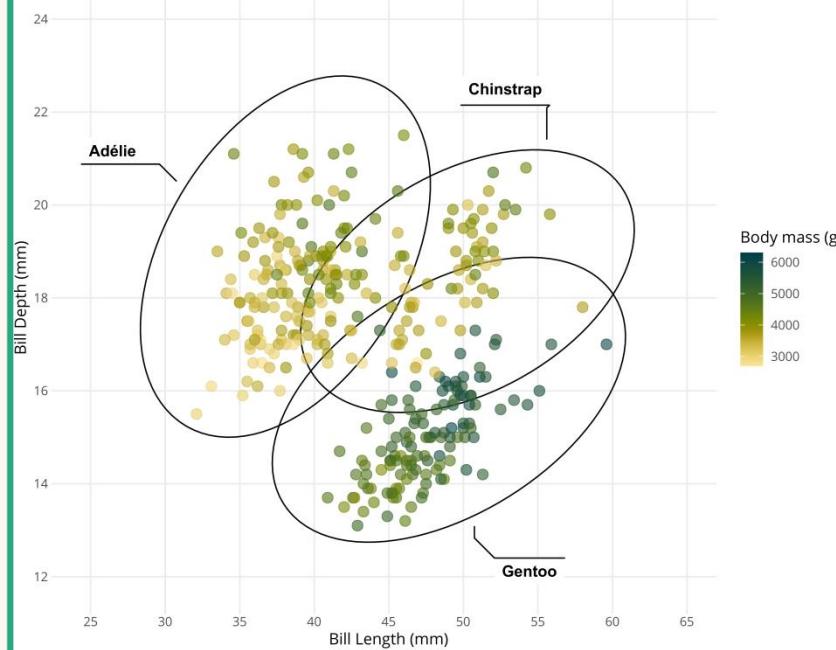


```
theme(plot.title.position = 'plot')
```

Better

Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



Data: Gorman, Williams & Fraser (2014) *PLoS ONE*

Data: Gorman, Williams & Fraser (2014) *PLoS ONE*

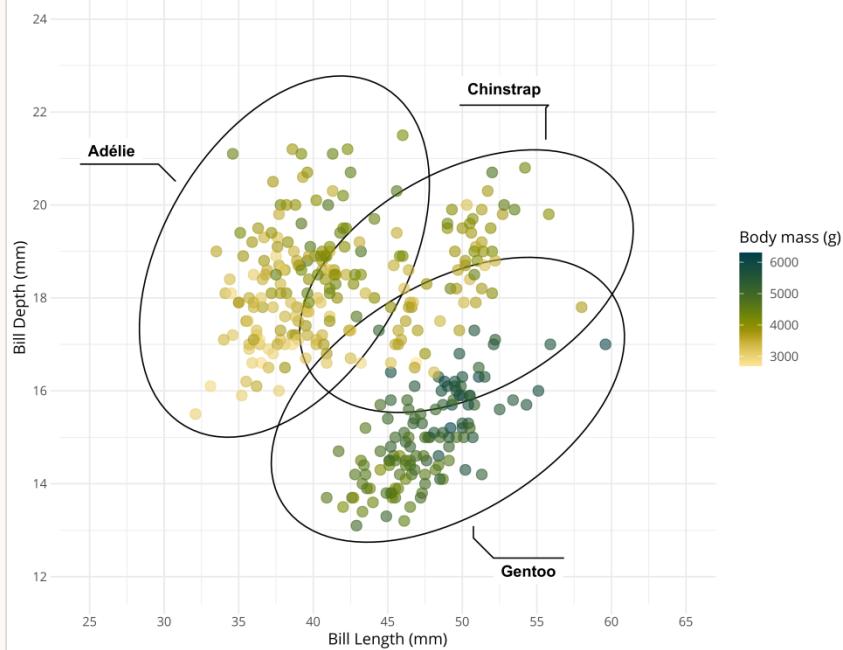
# Right-Aligned Caption

```
theme(plot.caption.position = 'panel')
```

Default

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



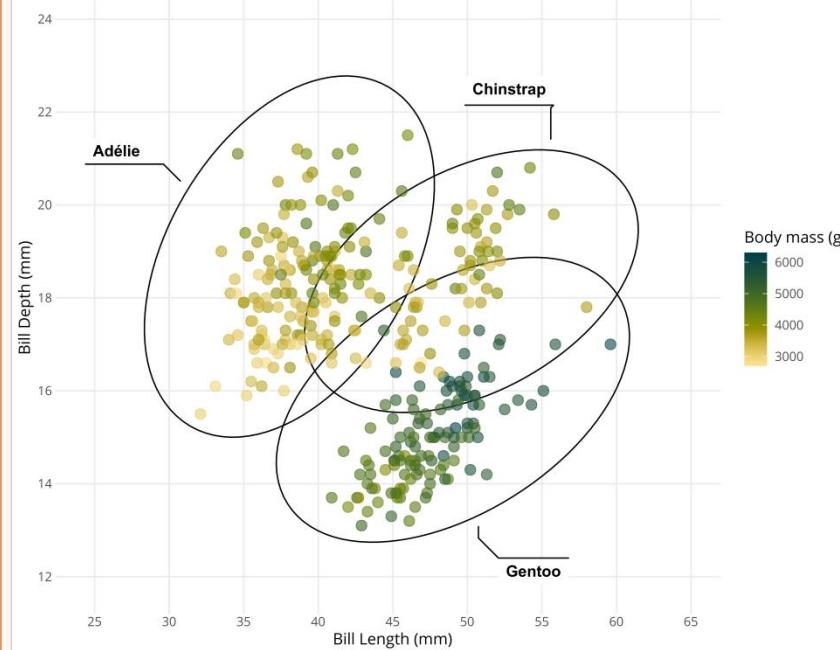
Data: Gorman, Williams & Fraser (2014) *PLoS ONE*

Better (?)

```
theme(plot.caption.position = 'plot')
```

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



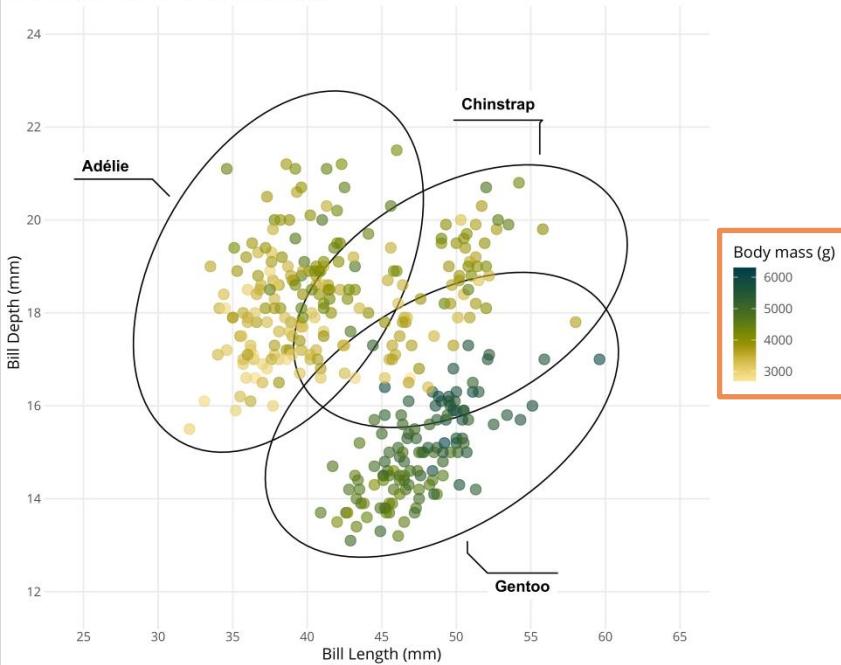
Data: Gorman, Williams & Fraser (2014) *PLoS ONE*

# Legend Design

theme(legend.position = 'right') *Default*

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

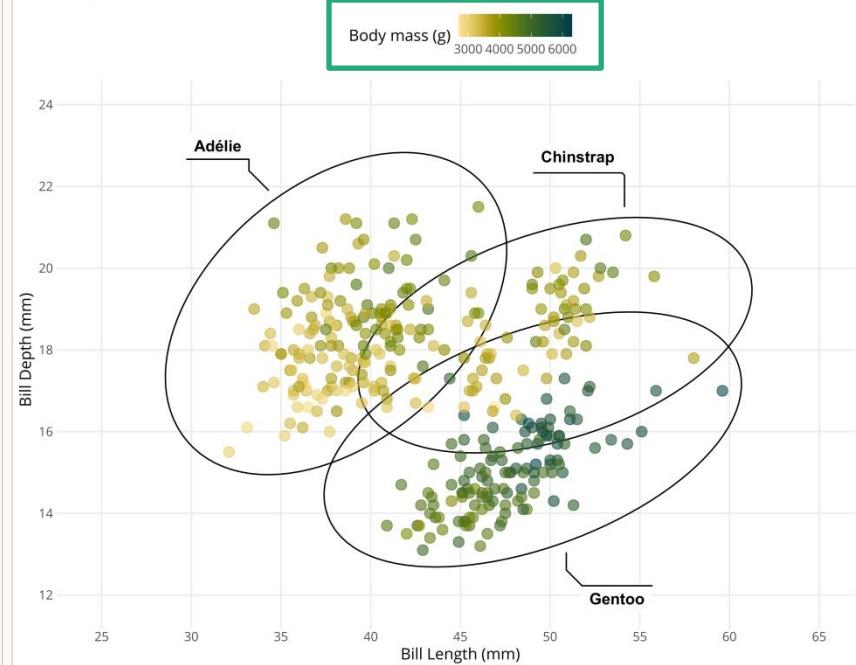
A scatter plot of bill depth versus bill length.



Well...  
theme(legend.position = 'plot')

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



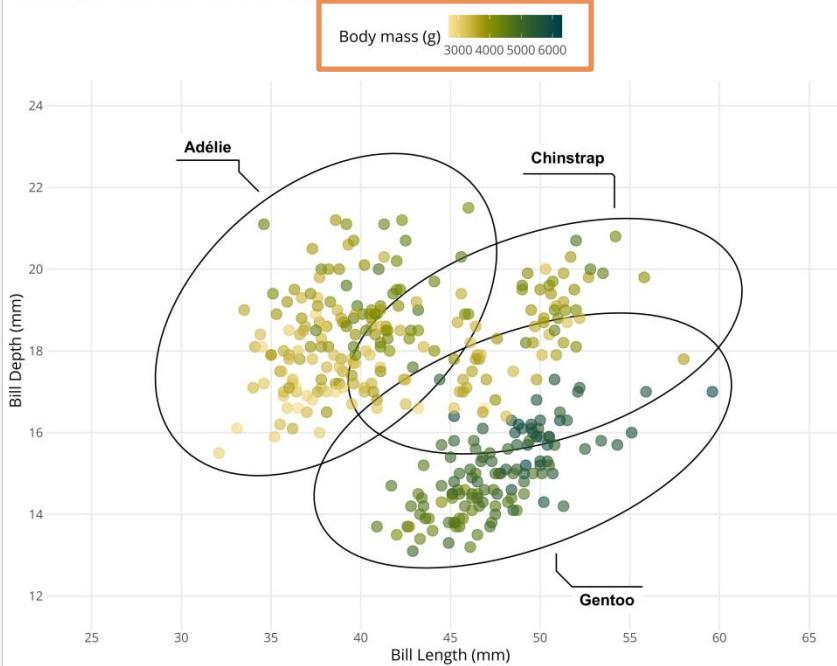
# Legend Design

*Default*

```
guides(color = guide_colorbar(...))
```

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



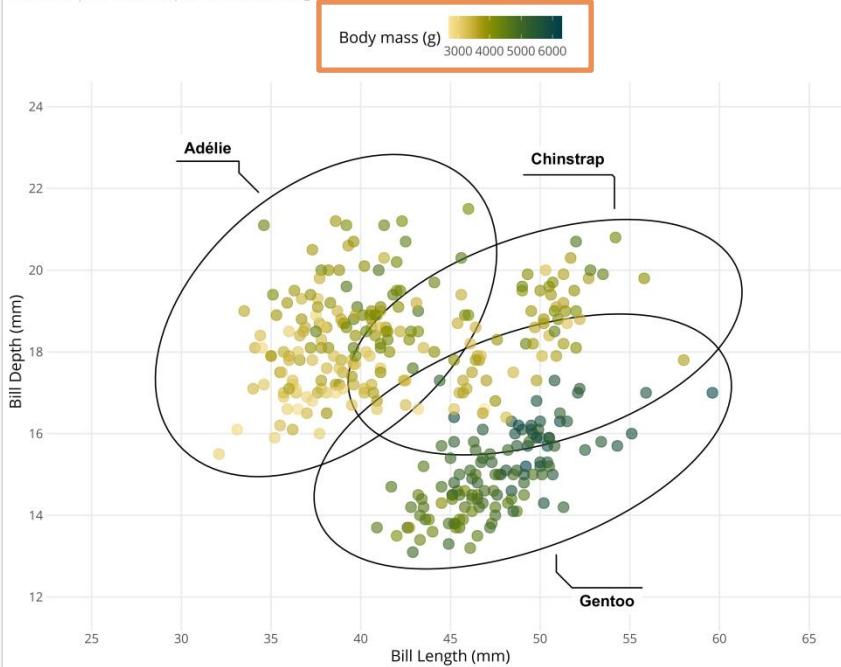
# Legend Design

```
guides(color = guide_colorbar(title.position = 'top', title.hjust = .5,  
                                barwidth = unit(20, 'lines'), barheight = unit(.5, 'lines')))
```

Better!

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

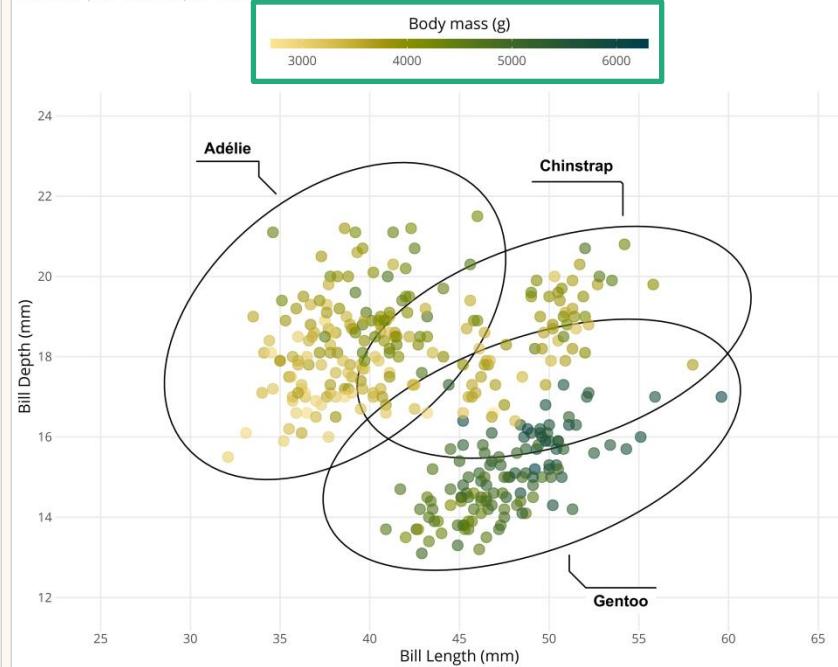
A scatter plot of bill depth versus bill length.



Data: Gorman, Williams & Fraser (2014) *PLoS ONE*

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



Data: Gorman, Williams & Fraser (2014) *PLoS ONE*

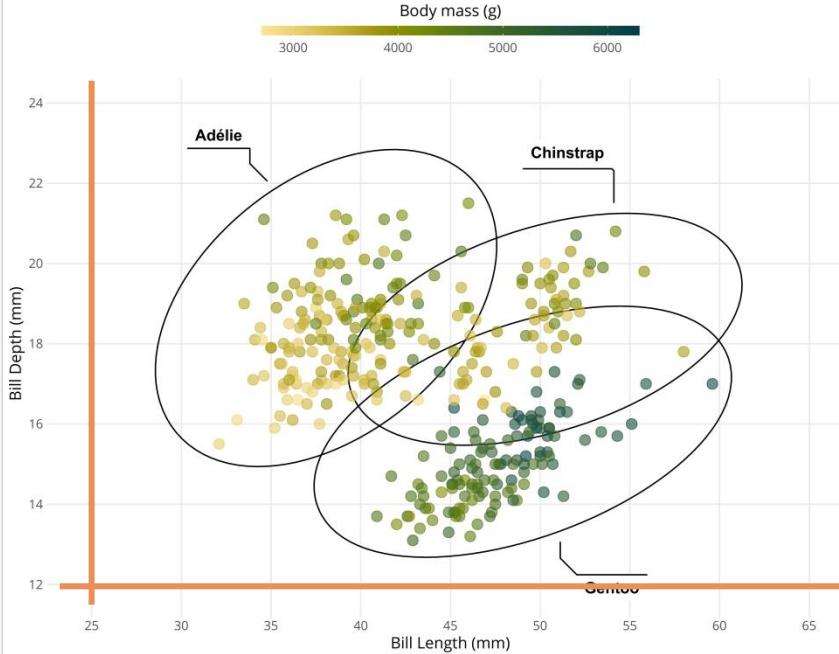
# Limit Expansion

`coord_cartesian(expand = TRUE)`

*Default*

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.

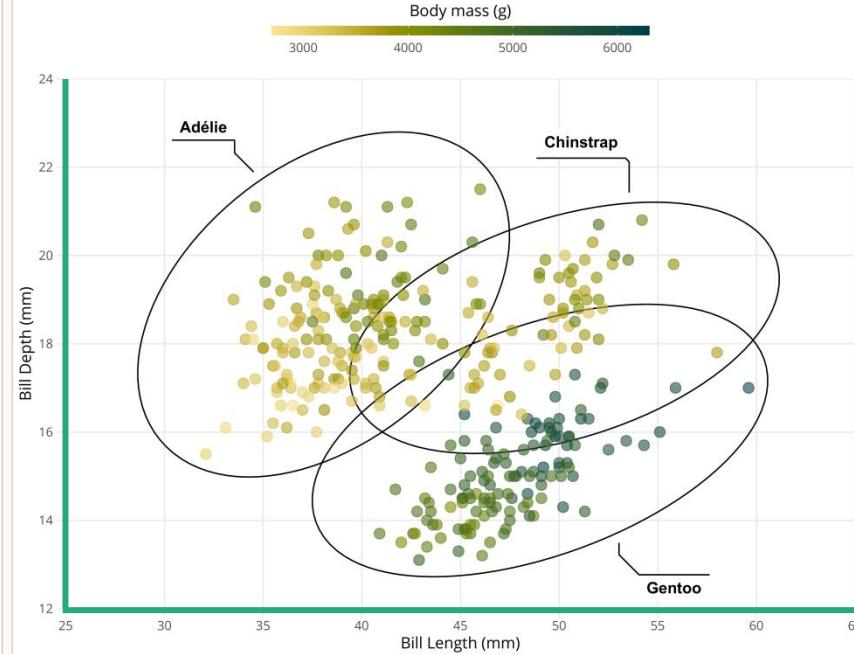


`coord_cartesian(expand = FALSE)`

*Without*

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



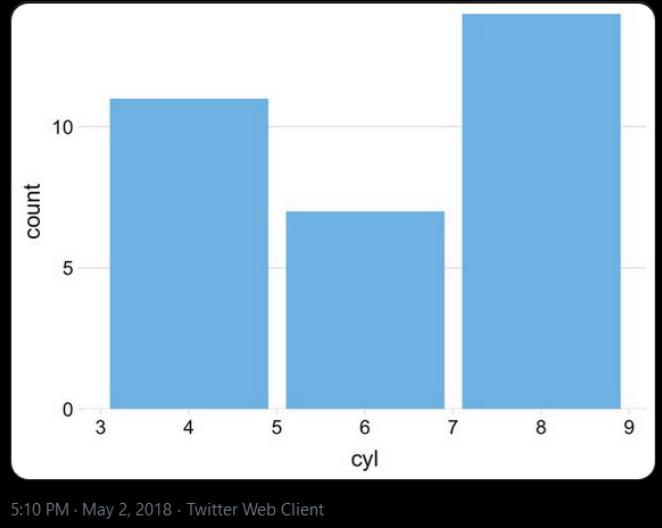
# Geeky Details



Claus Wilke  
@ClausWilke

Replies to @thomas85 and @hrbrmstr

This kind of stuff just really gets me. One of the main reasons I was motivated to add clip = "off" is plots like the attached, which look terrible in my opinion.



5:10 PM · May 2, 2018 · Twitter Web Client



gvdr @ipnosimmia · May 2, 2018

Replies to @ClausWilke @thomas85 and @hrbrmstr

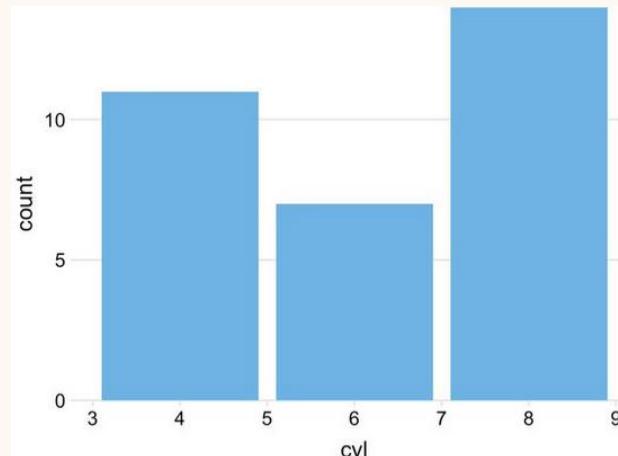
I must admit my limits. I can't see any difference between incorrect and correct. May I ask you to spell it out? (here to learn!)



1



2



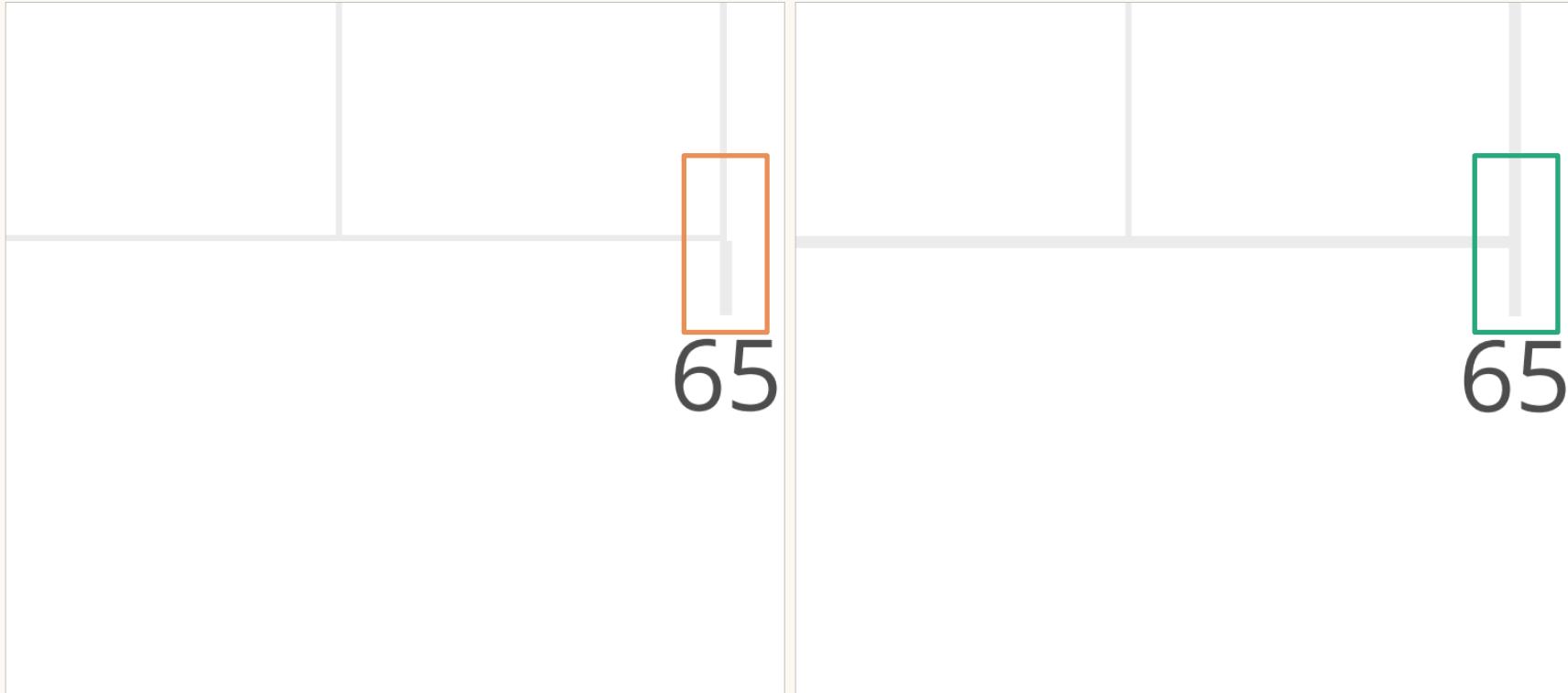
# Geeky Details

`coord_cartesian(clip = 'on')`

*Default*

*Without*

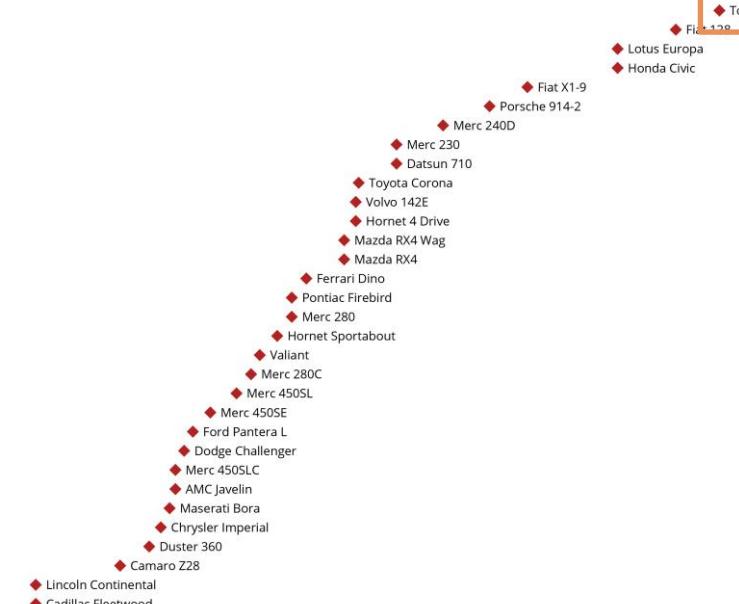
`coord_cartesian(clip = 'off')`



# Geeky Details

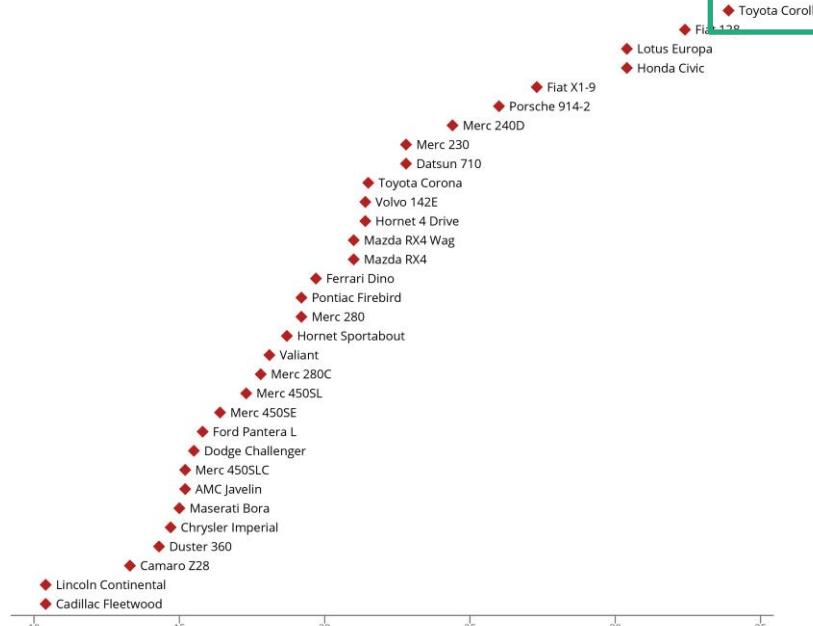
```
coord_cartesian(clip = 'on')
```

*Default*



```
coord_cartesian(clip = 'off')
```

*Without*



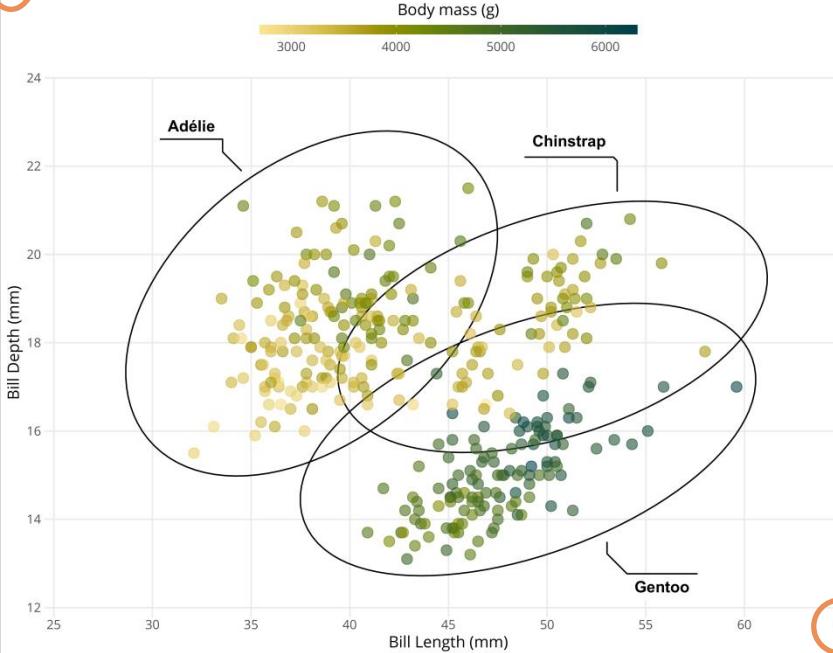
# White Space

```
theme(plot.margin = margin(rep(base_size/2, 4)))
```

Default

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.

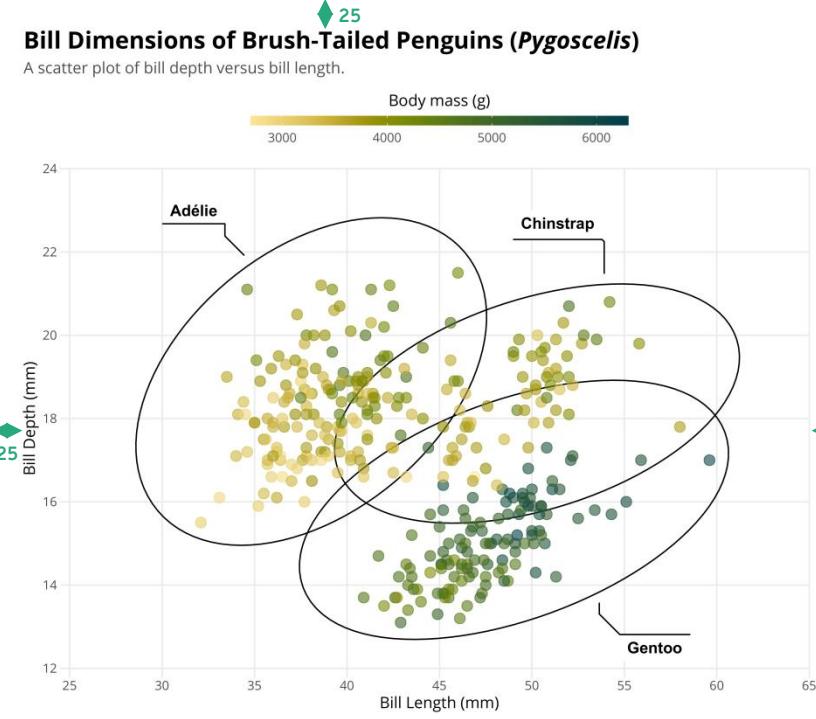


```
theme(plot.margin = margin(25, 25, 10, 25))
```

Better

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.

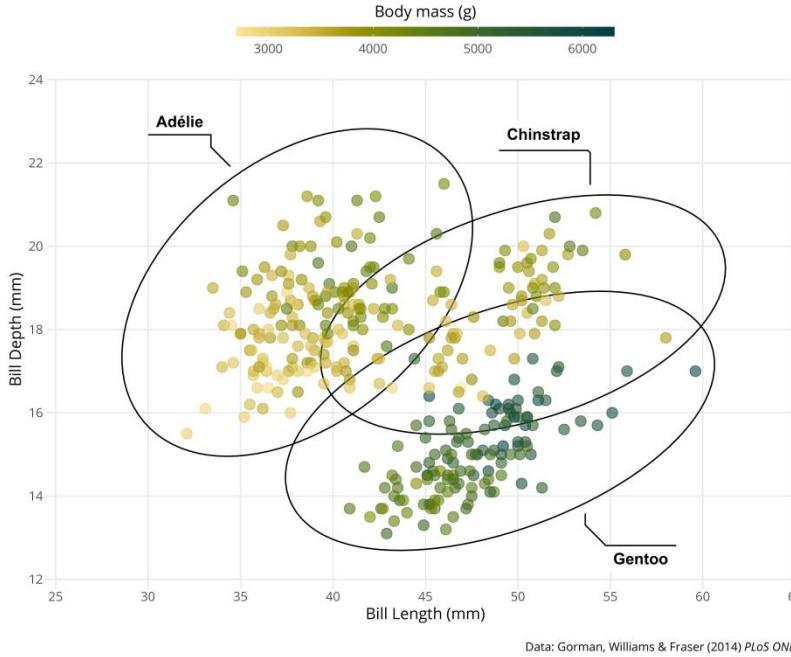


# Adding Images

```
png <- magick::image_read("https://raw.githubusercontent.com/allisonhorst/.../culmen_depth.png")
img <- grid::rasterGrob(png, interpolate = TRUE)
annotation_custom(img, ymin = 22, ymax = 31, xmin = 55, xmax = 65.5)
```

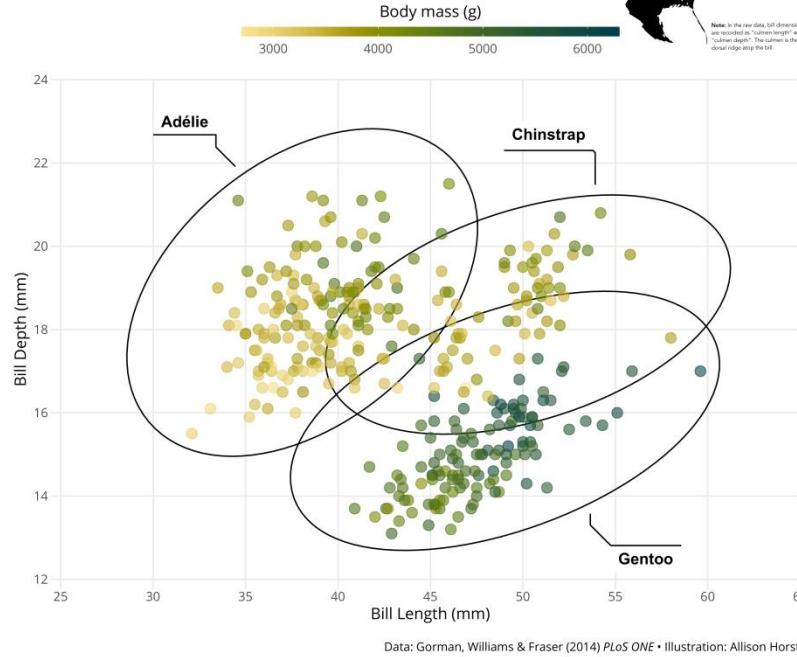
## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A scatter plot of bill depth versus bill length.



# {patchwork}

---

The Composer of ggplots



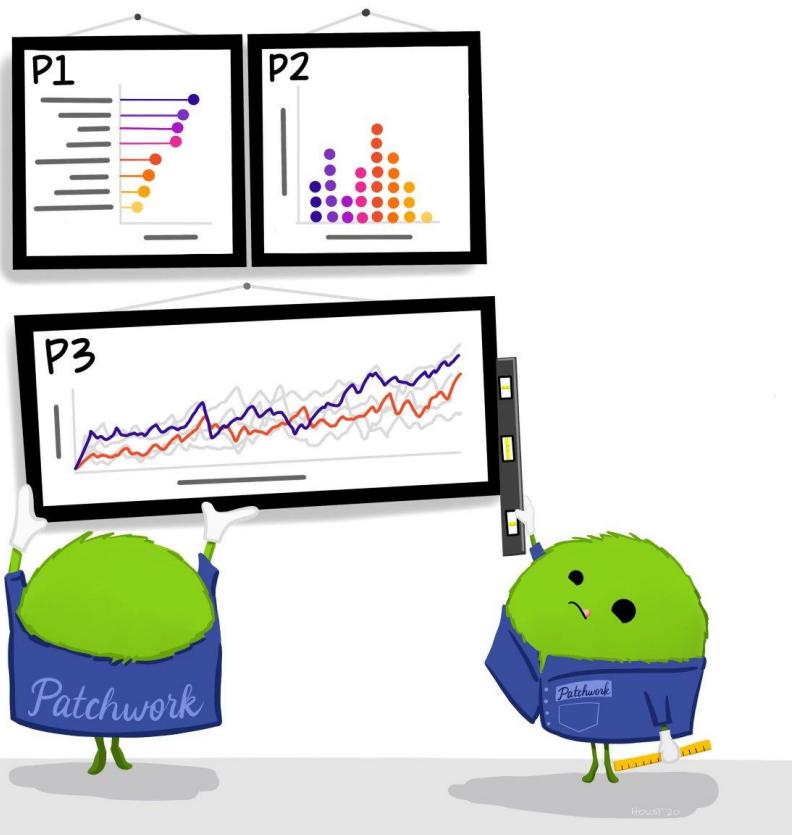
[patchwork.data-imaginist.com](http://patchwork.data-imaginist.com)

# patchwork

Combine + arrange  
your ggplots!

PLAN:  
 $(P1+P2)/P3$

P1	P2
P3	

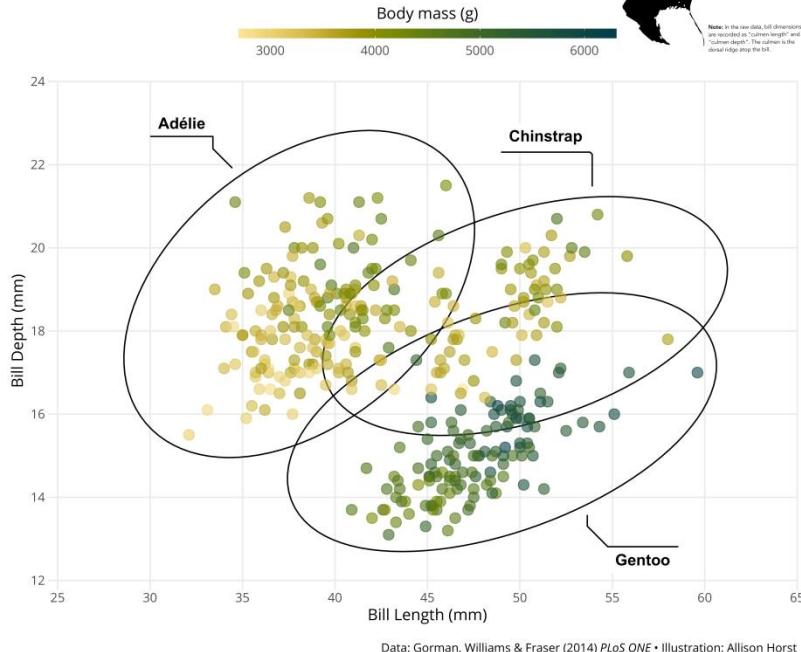


Artwork by Allison Horst

# {patchwork} The Composer of ggplots

## Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

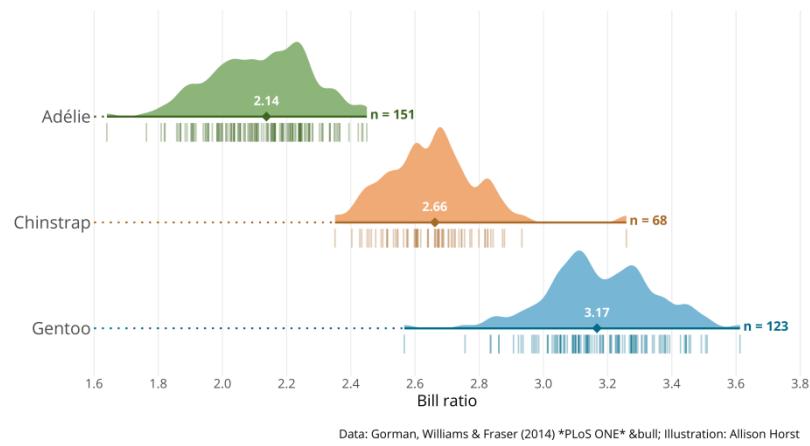
A scatter plot of bill depth versus bill length.



p1

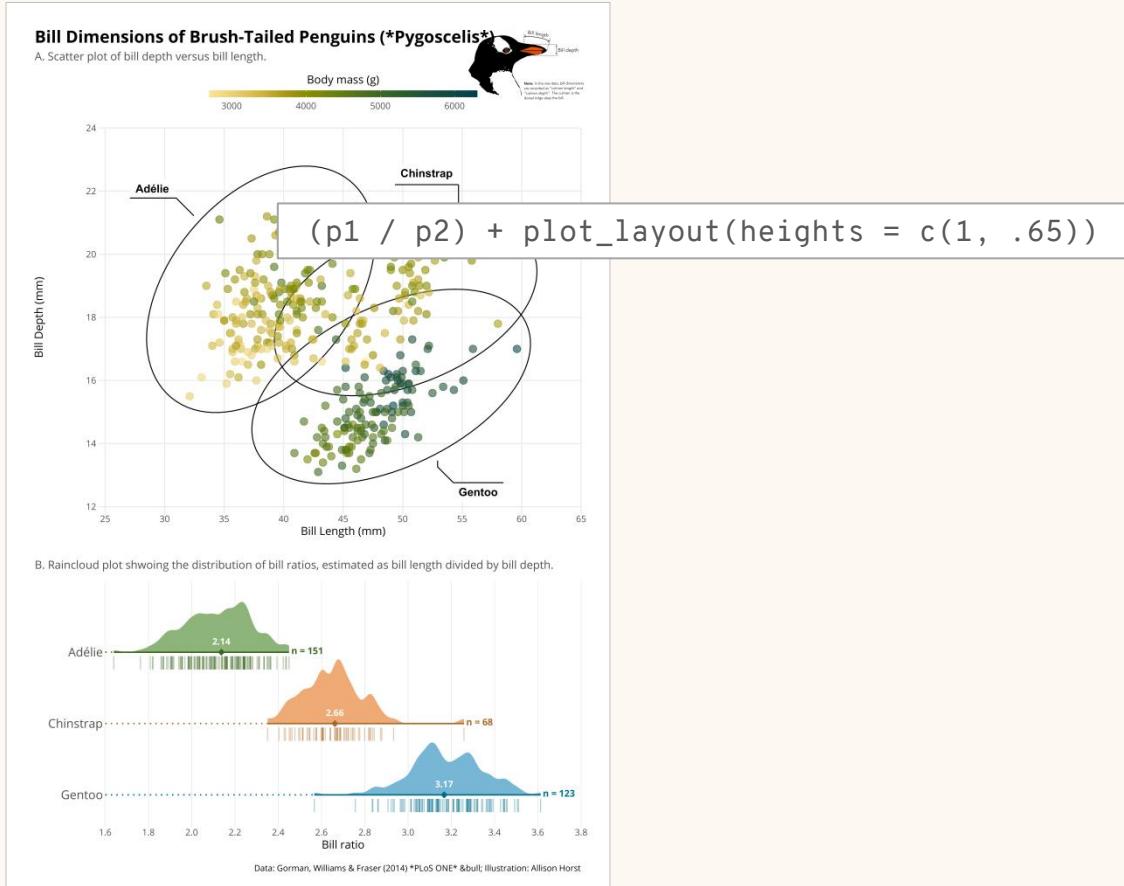
```
(p1 / p2) + plot_layout(heights = c(1, .65))
```

B. Raincloud plot showing the distribution of bill ratios, estimated as bill length divided by bill depth.



p2

# {patchwork} The Composer of ggplots



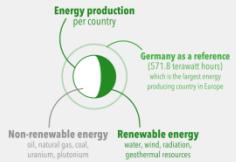
# {patchwork} The Composer of ggplots

## How European countries generated electricity in 2018

Germany is the largest energy producing country in Europe. It generates the most renewable and conventional thermal energy, representing 3% and 96% of its overall production respectively. France is the second largest energy European producer and by far the largest nuclear energy provider; 75% of its production is based on nuclear fission to generate heat.

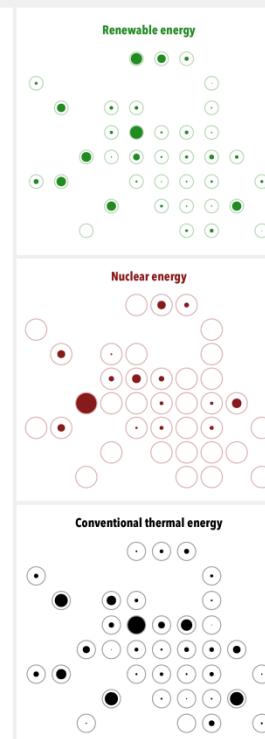
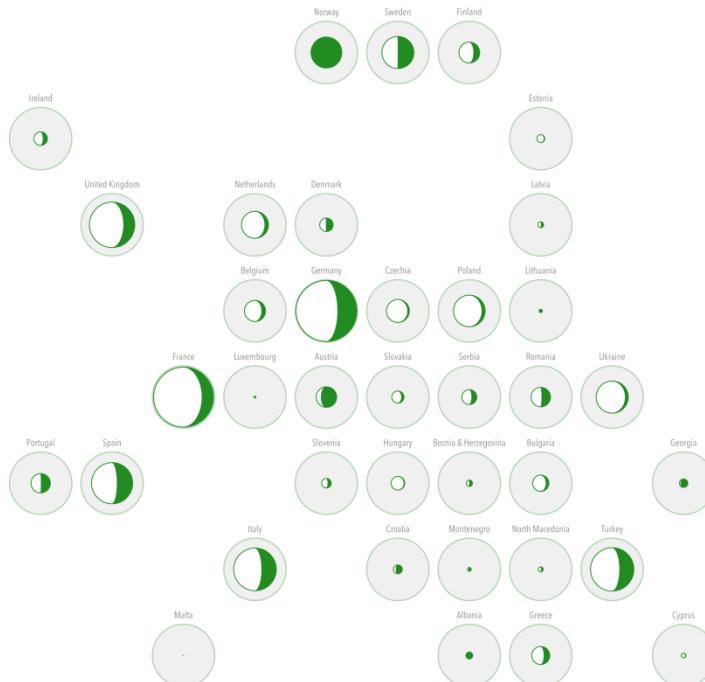


Renewable energy is energy that comes from resources that are naturally replenished such as sunlight, wind, water, and geothermal heat. Unlike fossil fuels, such as oil, natural gas and coal, or nuclear power sources such as uranium and plutonium, renewable energy regenerates naturally in a short period of time.



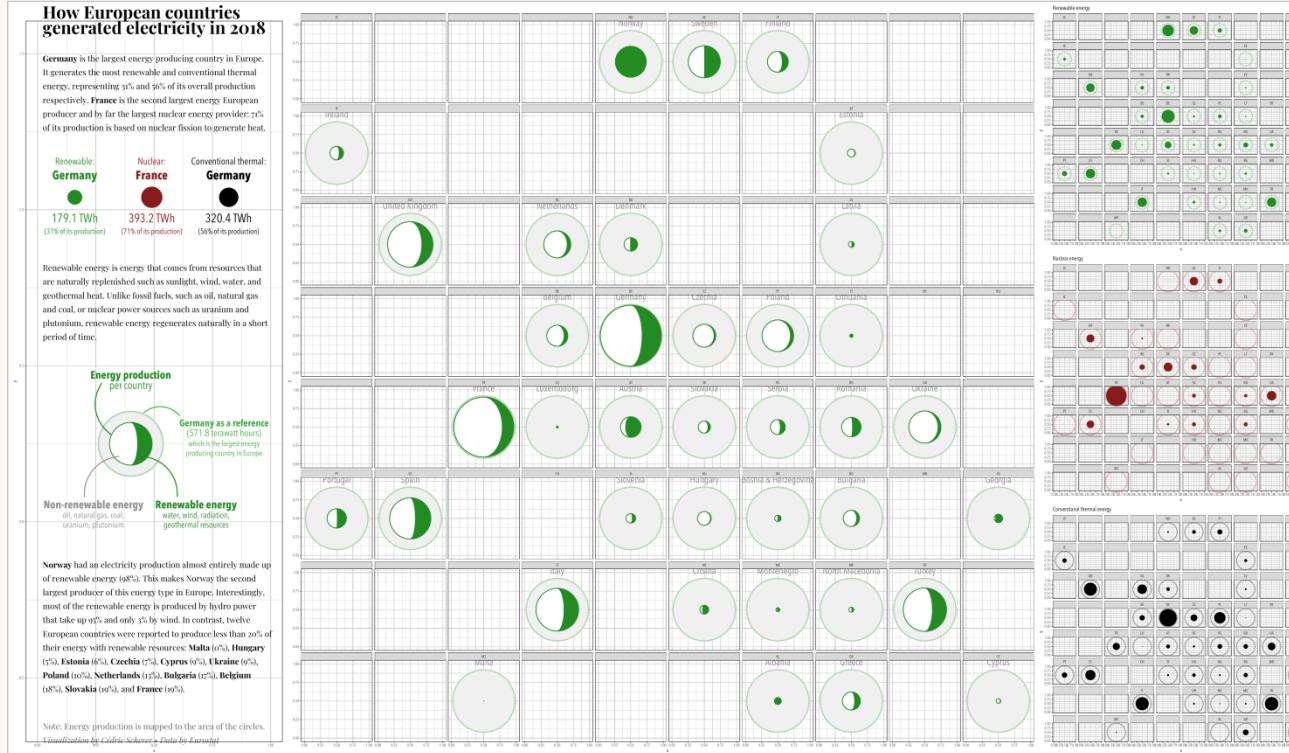
Norway had an electricity production almost entirely made up of renewable energy (68%). This makes Norway the second largest producer of this energy type in Europe. Interestingly, most of the renewable energy is produced by hydro power that take up 95% and only 5% by wind. In contrast, twelve European countries were reported to produce less than 20% of their energy with renewable resources: Malta (0%), Hungary (2%), Estonia (6%), Czechia (7%), Cyprus (9%), Ukraine (9%), Poland (10%), Netherlands (11%), Bulgaria (17%), Belgium (18%), Slovakia (19%), and France (20%).

Note: Energy production is mapped to the area of the circles.  
Visualisation by Cedric Scherer • Data by Eurostat



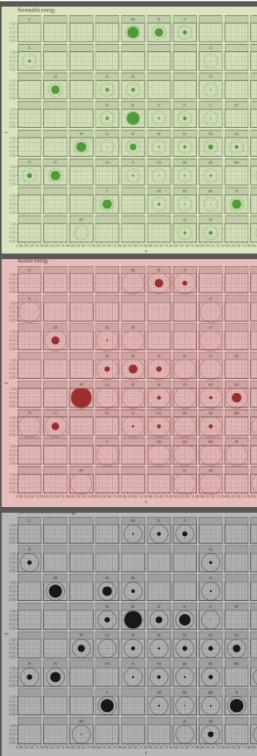
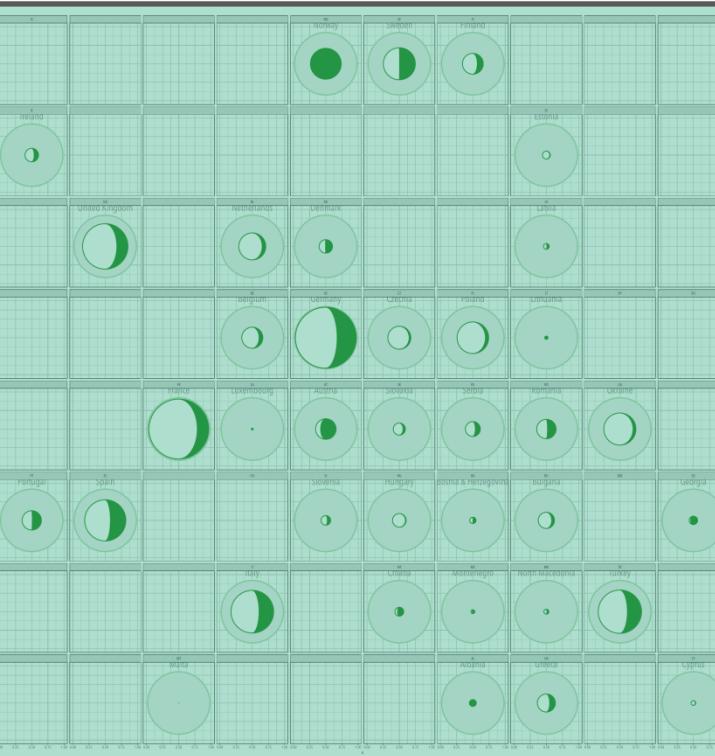
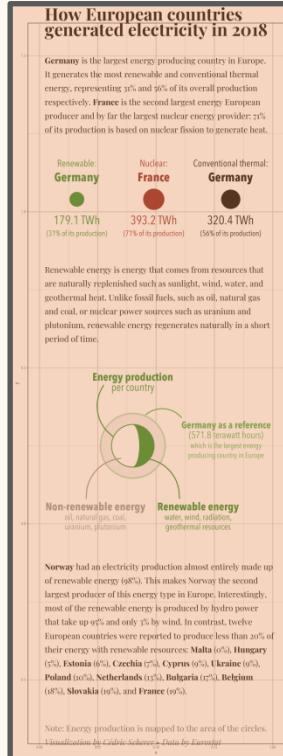
# {patchwork} The Composer of ggplots

legend | main | (renewable / nuclear / thermal) + plot\_layout(widths = c(.35, 1, .35))



# {patchwork} The Composer of ggplots

legend | main | (renewable / nuclear / thermal) + plot\_layout(widths = c(.35, 1, .35))



```
ggtitle('*Pygoscelis*') + theme(plot.title = element_markdown())
```



```
theme(plot.position = 'plot')
```

```
theme(legend.position = 'top') +  
  guide(color = guide_colorbar())
```

```
theme(plot.margin = margin(t, r, b, 1))
```

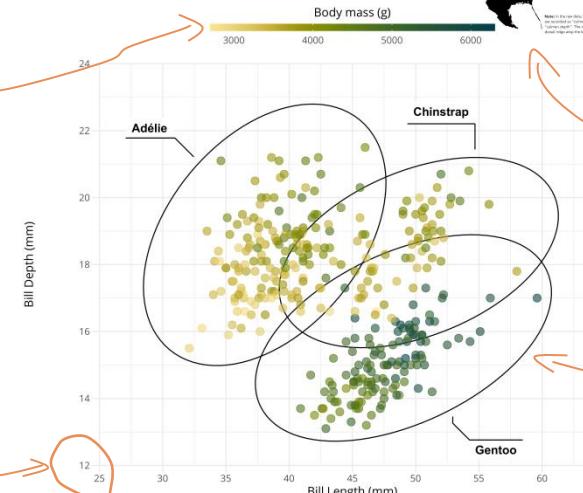
```
coord_cartesian(expand = c(0, 0),  
  clip = 'off')
```

```
ggdist::stat_halfeye()
```

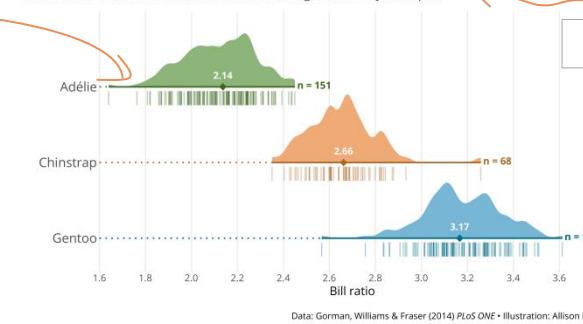


### Bill Dimensions of Brush-Tailed Penguins (*Pygoscelis*)

A. Scatter plot of bill depth versus bill length.



### B. Distribution of the bill ratio, estimated as bill length divided by bill depth



```
annotation_custom(grid::rasterGrob(img))
```



```
ggforce::geom_mark_*
```



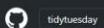
```
(p1 / p2) + plot_layout(heights = c(1, .65))
```

# Palmer Penguins

[github.com/allisonhorst/palmerpenguins](https://github.com/allisonhorst/palmerpenguins)



ggplot2  
[ggplot2.tidyverse.org](https://ggplot2.tidyverse.org)



Repositories

1,483 repository results

Sort: Most stars

Code 17K+

Commits 2K+

Issues 347

Discussions Beta

Packages 0

Marketplace 0

Topics 4

Wikis 7

Users 3

Languages

R 706

HTML 279

Jupyter Notebook 27

Python 6

JavaScript 4

CSS 1

Mathematica 1

Shell 1

TeX 1

Advanced search

Cheat sheet

fordatascience/tidytuesday

Official repo for the #tidytuesday project  
3.2k HTML CC0-1.0 license Updated 20 hours ago

Z3tt/TidyTuesday

My contributions to the #tidytuesday challenge  
ggplot2 dative tidyverse rstats r4ds tidytuesday tidytuesday-challenge  
384 R Updated 5 days ago

gkaramanis/tidytuesday

Code and plots for submissions to the #tidytuesday challenge  
249 HTML MIT license Updated 4 days ago

jkaupp/tidytuesdays

Entries for tidytuesday  
r4ds r tidytuesday  
134 R Updated 13 hours ago

cienciadedatos/datos-de-miercoles

Primo latinoamericano de #tidytuesday  
98 HTML Updated on 12 Jul 2020

sprenger/tidytuesday

Code snippets for weekly #tidytuesday project.  
64 R Updated on 25 Oct 2020

zhiliyang/tidytuesday

Data visualization collection  
tidytuesday dative rstats datavisualization  
63 R Updated on 5 Jan 2020

abichat/tidytuesday

Codes and plots for the #TidyTuesday challenge  
42 R Updated on 12 May 2020

thebioengineer/tidytuesdayR

Extract weekly TidyTuesday Data/Readme  
39 R Updated 15 days ago

jack-davison/TidyTuesday

Collection of #TidyTuesday Visualisations!  
tidytuesday-visualisations ggplot2 r tidyverse rstats tidytuesday tidytuesday-challenge  
23 HTML Updated 1 hour ago

&lt; Previous 1 2 3 4 5 ... 99 100 Next &gt;



Z3tt/TidyTuesday

Unwatch 49 Star 384 Fork 74  
Code Issues Pull requests Actions Security Insights Settings

master TidyTuesday / R / 2021\_02\_TransitCosts.Rmd

Z3tt remove unnecessary code parts

Latest commit 9a8b54c 11 days ago History

All 1 contributor

249 lines (227 sloc) 7.51 KB

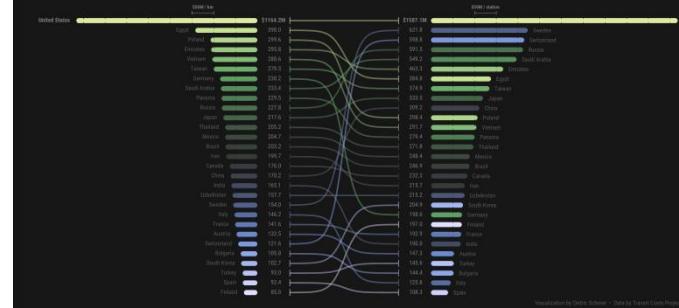
```

1 ---
2   title: "TidyTuesday 2021/02 - Transit Costs by Transit Cost Project"
3   author: "Cédric Scherer"
4   date: "7th of January 2021"
5   output:
6     html_document:
7       theme: paper
8       highlight: kate
9       editor_options:
10         chunk_output_type: console
11   ---
12
13   ```{r setup, include=FALSE}
14   knitr::opts_chunk$set(echo = TRUE, warning = FALSE, fig.showtext = T, fig.retina = 1)
15   ```
16
17   ```{r prep, message=FALSE, warning=FALSE}
18   ## packages
19   library(tidyverse)
20   library(ggplot2)
21   library(ggtext)
22   library(systemfonts)
23   library(pdftools)
24
25   theme_set(theme_void(base_family = "Roboto Condensed"))
26
27   theme_update(
28     legend.position = "none",
29     plot.margin = margin(25, 35, 25, 25),
30     plot.background = element_rect(fill = "#1f77b4"),
31     plot.title = element_markdown(color = "grey70", size = 24,
32                                   family = "Roboto Black", face = "bold",
33                                   lineheight = 1.1),
34     plot.subtitle = element_markdown(color = "grey50", size = 13,
35                                     lineheight = 1.1,
36                                     margin = margin(t = 15, b = 35)),
37     plot.caption = element_text(color = "grey50", size = 10,
38                                 margin = margin(t = 0))
39   )
40
41   ````{r data}
42   df_transit <- reader::read_csv("https://raw.githubusercontent.com/fordatascience/tidytuesday/master/data/2021/2021-02-transit_cost.csv", col_types = list(real_cost = col_double()))
43
44   ````{r prep-data}
45   df_transit_ranking <-
46     df_transit %>%
47     filter(!is.na(stations), stations > 0, real_cost > 0, end_year < 2021) %>%
48     mutate(
49       cost_per_station = real_cost / stations,
50       country = case_when(country ~ "UK" ~ "GB",
51                           TRUE ~ country)
52     ) %>%
53     group_by(country) %>%
54     filter(n() > 1) %>%
55     summarize(across(c(length, stations, real_cost, cost_in_millions, cost_per_station),

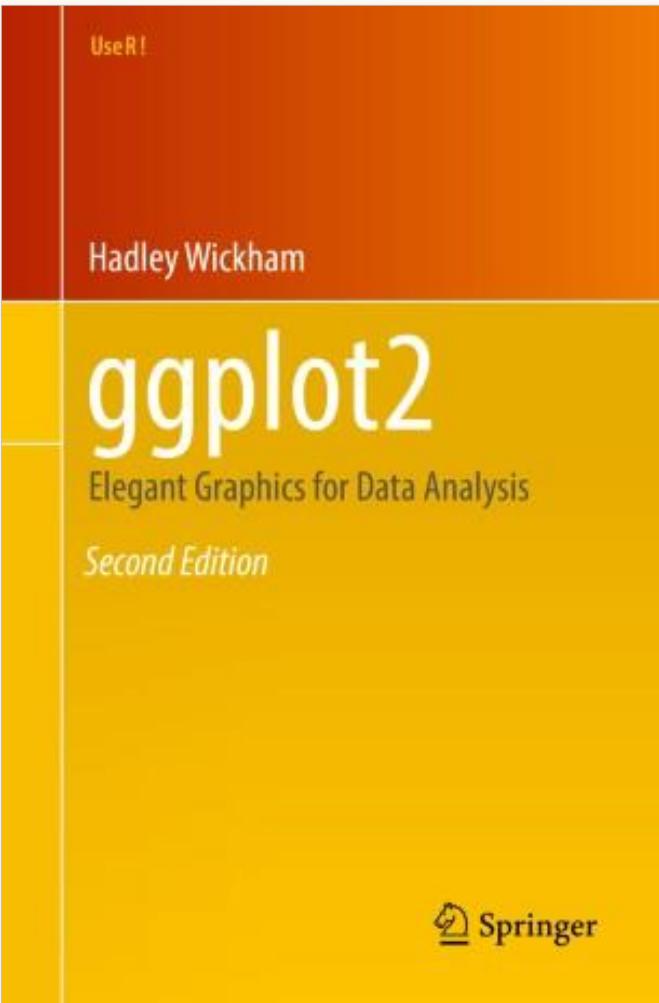
```

The United States spend by far the most improving the state of public transit-infrastructure not only when it comes to transit length but also costs per station.

Based on a database that spans more than 50 countries provided by the Transit Costs Project, the graphic shows the costs for urban rail projects that have been finished since the late 1990s. Each wagon represents a cost of 50 million USD per km (left) and per station (right), each ranked by highest costs. The connections highlight the changes in the ranking with the United States holding the first place in both categories.



Visualization by Gertjan Verbeek - Data by Transit Costs Project



The image is a screenshot of the R Graph Gallery website. At the top, there is a navigation bar with links for "CHART TYPES", "QUICK", "TOOLS", "ALL", "D3.JS", "PYTHON", "DATA TO VIZ", and "ABOUT". Below the navigation bar, the text "The R Graph Gallery" is centered, followed by social media sharing icons for Twitter, GitHub, LinkedIn, and others. A welcome message from the website's creator is present, encouraging users to explore the gallery and provide feedback. The main content area is organized into several sections, each featuring a grid of circular thumbnails representing different types of charts. The sections include "Distribution", "Correlation", "Ranking", and "Part of a whole". Each section contains five thumbnails, each showing a different type of chart such as a violin plot, density plot, histogram, boxplot, or word cloud.

CHART TYPES    QUICK    TOOLS    ALL    D3.JS    PYTHON    DATA TO VIZ    ABOUT

The R Graph Gallery

Welcome to the R graph gallery, a collection of charts made with the [R programming language](#). Hundreds of charts are displayed in several sections, always with their reproducible code available. The gallery makes a focus on the tidyverse and [ggplot2](#). Feel free to suggest a chart or report a bug; any feedback is highly welcome. Stay in touch with the gallery by following it on [Twitter](#) or [Github](#). If you're new to R, consider following this [course](#).

Distribution

Violin   Density   Histogram   Boxplot   Ridgeline

Correlation

Scatter   Heatmap   Correlogram   Bubble   Connected scatter   Density 2d

Ranking

Barplot   Spider / Radar   Wordcloud   Parallel   Lollipop   Circular Barplot

Part of a whole

[DATAVERSE](#) [TUTORIAL](#) [TOYVERSE](#) [GGPLOT2](#)

# A GGPLOT2 TUTORIAL FOR BEAUTIFUL PLOTTING IN R

POSTED BY CÉDRIC ON MONDAY, AUGUST 5, 2013

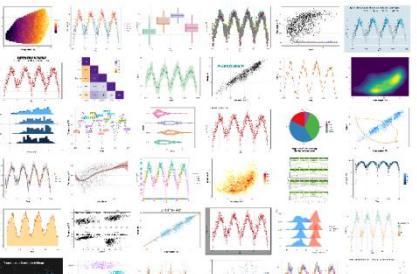
Last update: 2020-12-07

## INTRODUCTORY WORDS

I don't care, just show me the content!

Back in 2016, I had to prepare my PhD introductory talk and I started using [\(ggplot2\)](#) to visualize my data. I never liked the syntax and style of base plots in R, so I was quickly in love with ggplot. Especially useful was its faceting utility. But because I was short on time, I plotted these figures by trial and error and with the help of lots of googling. The resource I came always back to was a blog entry called [Beautiful plotting in R: A ggplot2 cheatsheet](#) by Zev Ross, updated last in January 2016. After giving the talk which contained some decent plots thanks to the blog post, I decided to go through this tutorial step-by-step. I learned so much from it and directly started modifying the codes and over the time I added additional code snippets, chart types and resources.

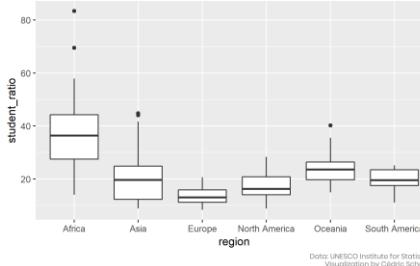
Since the blog entry by Zev Ross was not updated for some years and step by step this became a unique version of a tutorial, I decided to host the updated version on my GitHub. Now it finds its proper place on this homepage! (Plus I added a ton of other updates—just to name a few: The fantastic [\(patchwork\)](#), [\(ggttext\)](#) and [\(ggeforce\)](#) packages. How to deal with custom fonts and colors. A collection of R packages tailored to create interactive charts. And several other chart types including pie charts because everyone looooves pie charts!)


[DATAVERSE](#) [TUTORIAL](#) [ANIMATIONS](#) [GGPLOT EVOLUTION](#) [TOY](#) [GGPLOT2](#) [TOYVERSE](#) [TOYTUESDAY](#)

# THE EVOLUTION OF A GGPLOT (EP. I)

POSTED BY CÉDRIC ON FRIDAY, MAY 17, 2019

## The Evolution of a ggplot



- Aim of this Tutorial
- Data Preparation
- The Default Boxplot
- Sort Your Data!
- Let Your Plot Shine—Get Rid of the Default Settings
- The Choice of the Chart Type
- More Geoms, More Fun, More Info!
- Add Text Boxes to Let The Plot Speak for Itself
- Bonus: Add a Tile Map as Legend
- The Final Evolved Visualization
- Complete Code for Final Plot
- Post Scriptum: Mean versus Median

### AIM OF THIS TUTORIAL

In this series of blog posts, I aim to show you how to turn a default ggplot into a plot that visualizes information in an appealing and easily understandable way. The goal of each blog post is to provide a step-by-step tutorial explaining how my visualization have evolved from a typical basic ggplot. All plots are going to be created with 100% [\(ggplot2\)](#) and 0% Inkscape.

In the first episode, I transform a basic boxplot into a colorful and self-explanatory combination of a jittered dot strip plot and a lollipop plot. I am going to use [data](#) provided by the UNESCO on global student to teacher ratios that was selected as data for the #TidyTuesday challenge 19 of 2019.

- [www.cedricscherer.com](http://www.cedricscherer.com)

[cedricphilippscherer@gmail.com](mailto:cedricphilippscherer@gmail.com)

[www.twitter.com/CedScherer](https://www.twitter.com/CedScherer)

[www.github.com/z3tt](https://www.github.com/z3tt)

**Outlier**

*Thank you!*