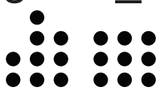
Grammar	ggplot2	PGoG
Defaults		
Data	A	P(A|B,...)
Aesthetics	x ← A	height ← P(A|B,...)
Layer		
Data		
Aesthetics		
Geom	geom_bar	geom_bloc
Stat		
Position	geom_density	geom_icon
Scale		
	geom_points	
Coord		
Facet	geom_rect	
		
	geom_...	

(Wickham 2010)

What is the Probabilistic Grammar of Graphics?

1. The PGoG **grammar** is an extension to *Grammar of Graphics*
2. Probability distributions are first class citizens (data) and other grammar components (aesthetics and geometries) are defined around them

PGoG Grammar/*data*

```
                   mpg cyl am
Mazda RX4         21.0   6  1
Mazda RX4 Wag     21.0   6  1
Datsun 710        22.8   4  1
Hornet 4 Drive    21.4   6  0
Hornet Sportabout 18.7   8  0
Valiant           18.1   6  0
```

Column variable

`mpg`

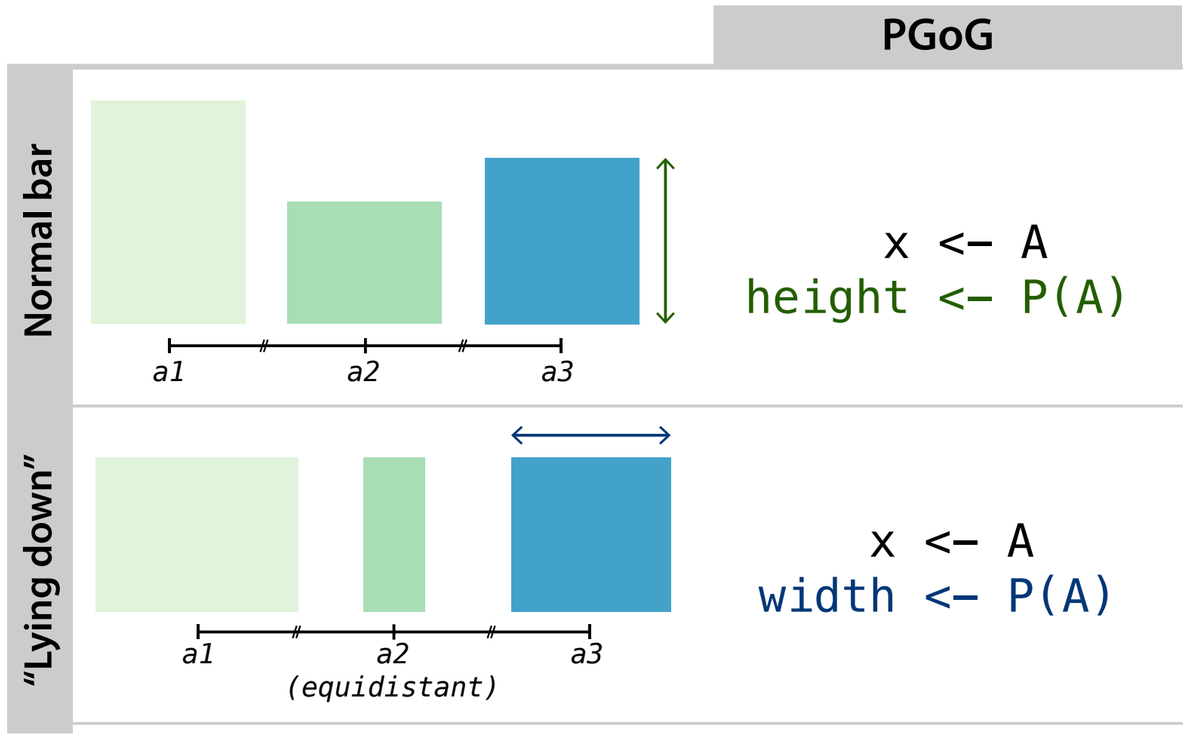
A column in tidy dataset

Probabilistic variable

`P(mpg|cyl)`

In the form of P(A...IB...), where A, B and ... are variables in columns

PGoG Grammar/*aesthetics 1/3*

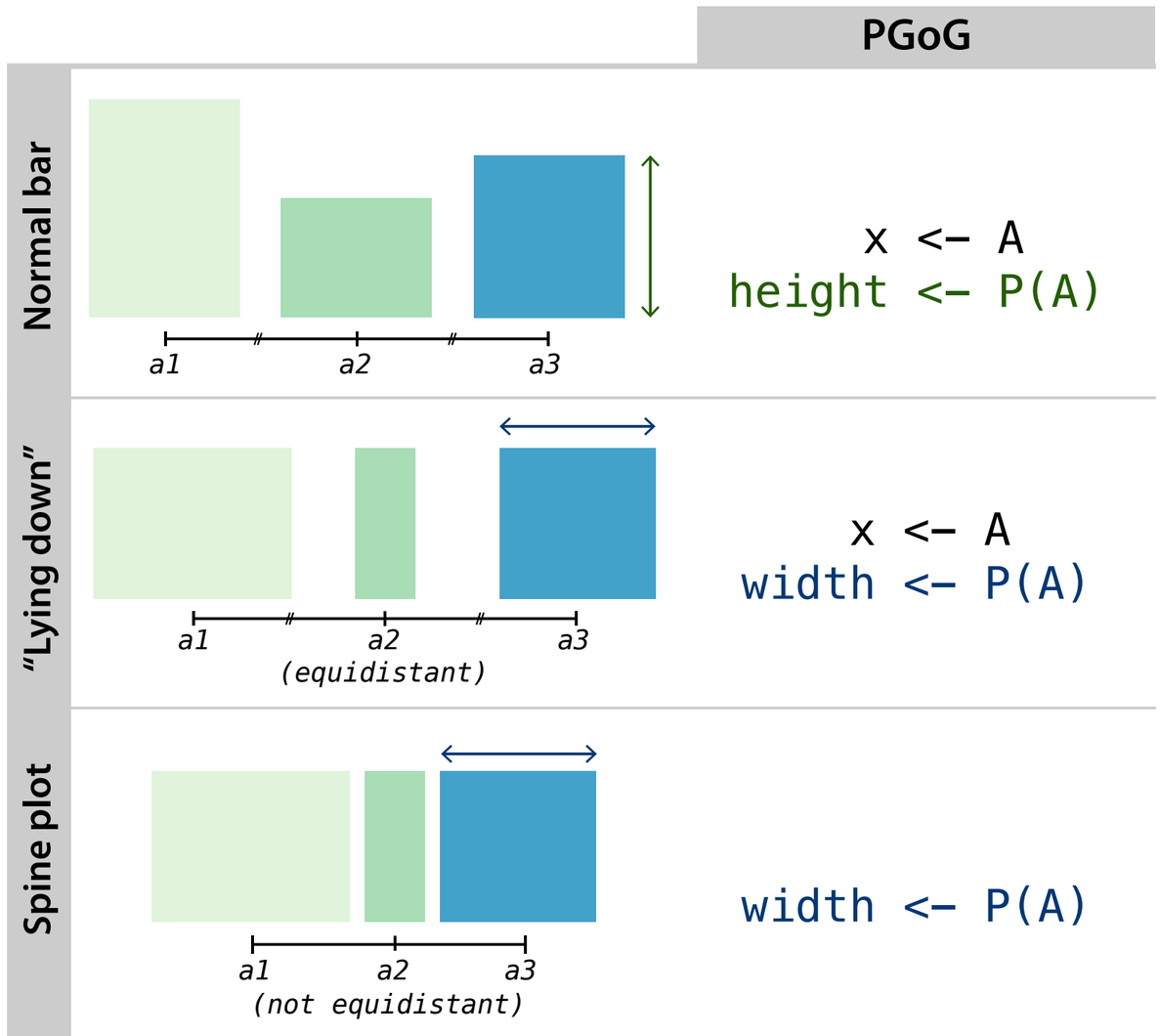


Probabilistic aesthetics

`width, height`

- Works with probabilistic variables only
- Expresses the probability value by length

PGoG Grammar/*aesthetics 2/3*



Probabilistic aesthetics

Coordinate aesthetics

x, y

- For discrete vars: equidistant partitions
- For continuous vars: as one would expect

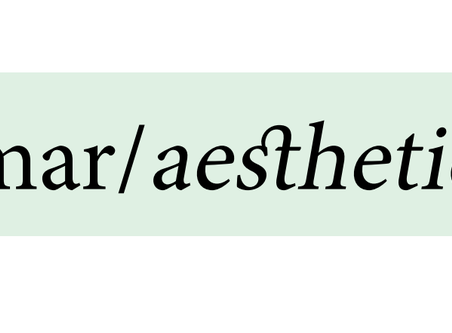
PGoG Grammar/*aesthetics 3/3*

Probabilistic aesthetics

Coordinate aesthetics

Visual aesthetics

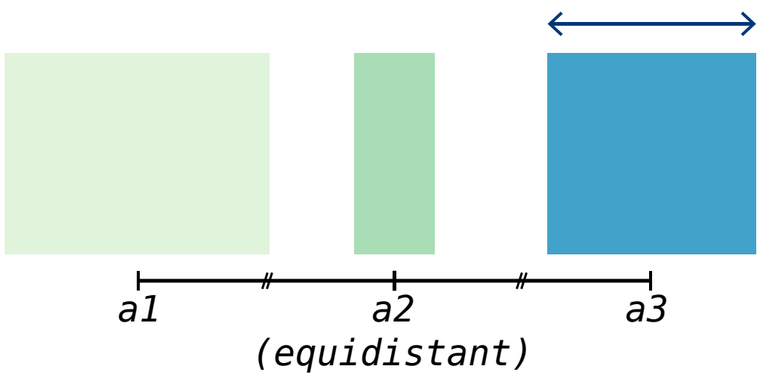
Uses x, y aesthetics



A scatter plot

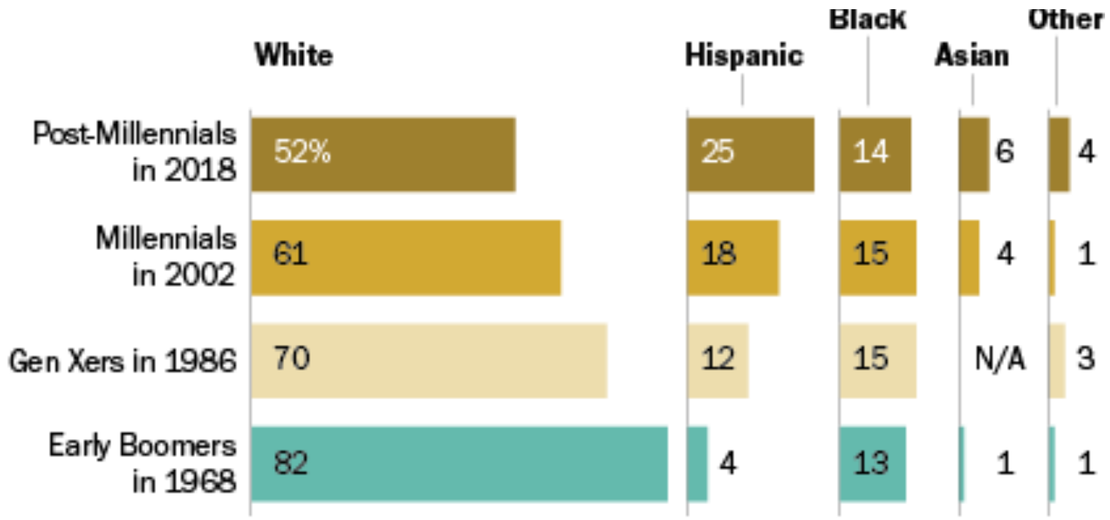
```
fill,color,alpha,...
```

PGoG Grammar/*Example for conditional*




```
x <- A
width <- P(A)
```

http://www.pewresearch.org/fact-tank/2018/12/13/18-striking-findings-from-2018/




```
x <- race
y <- generation
width <- P(race|generation)
```

ggplot2	PGoG
geom_bar 	**geom_bloc**
geom_mosaic* 	
geom_density 	
geom_violin 	
geom_density_ridges* 	
geom_waffle* 	**geom_icon**
geom_dotplot 	

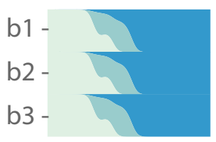
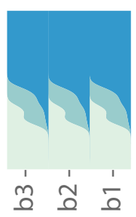
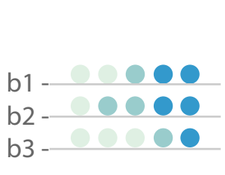
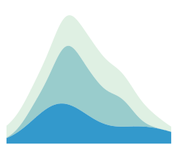
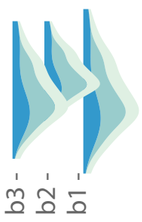
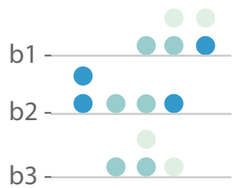
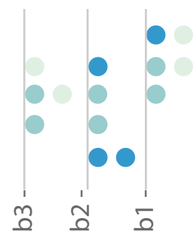
Comparison: look at all those geometries in `ggplot2` we have replaced

* `ggplot2` extensions

Evaluation of the Grammar

- Expressive?
- Generative?
- Cognitively ergonomic?

Expressiveness of the grammar

	geom_bloc				geom_icon			
	x ← A h ← ...	y ← A w ← ...	x ← A y ← B h ← ...	y ← A x ← B w ← ...	x ← A h ← ...	y ← A w ← ...	x ← A y ← B h ← ...	y ← A x ← B w ← ...
P(A)	 density plot		NA	NA	 dotplot		NA	NA
P(B|A) P(B|A, C)								
P(B|A) P(A) P(B|A,C) P(A|C)		 onion plot*						

Generativeness from the combination of aesthetics

Onion plot

```
geom_bloc:
          y ← mpg
      width ← P(mpg) P(cyl|mpg)
direction ← both
```

Cognitive ergonomics (Blackwell et al. 2001)

Pro:
Short edit distances
- *Low viscosity*
- *No premature committment*

Close to probability expressions

Con:
Specifying probability expressions can be difficult
- *Hidden dependencies*
- *Error prone-ness*

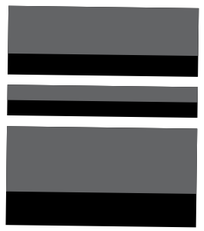
Cognitive ergonomics

Pro: Short edit distances, close to probability expressions

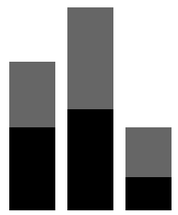
- *Low viscosity*
- *No premature committment*

Existing ggplot2 packages

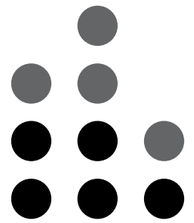
Changes



```
geom:mosaic→bar
+fill +y
-divider
```



```
geom:bar→dotplot
+stackgroup +method
-y
```



Syntax

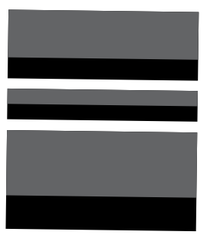
```
geom_mosaic
      x = cyl,
          mpg*
divider = hspine,
          hspine
```

```
geom_bar
   x = mpg*
   y = stat(prop)
fill = cyl
```

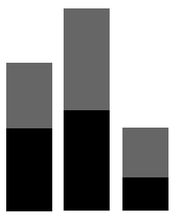
```
geom_dotplot
        x = mpg*
     fill = cyl
stackgrp = TRUE
  method = histodot
```

Probabilistic Grammar of Graphics (PGoG)

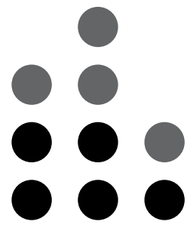
Changes



```
+x
```



```
geom:bloc→icon
```



Syntax

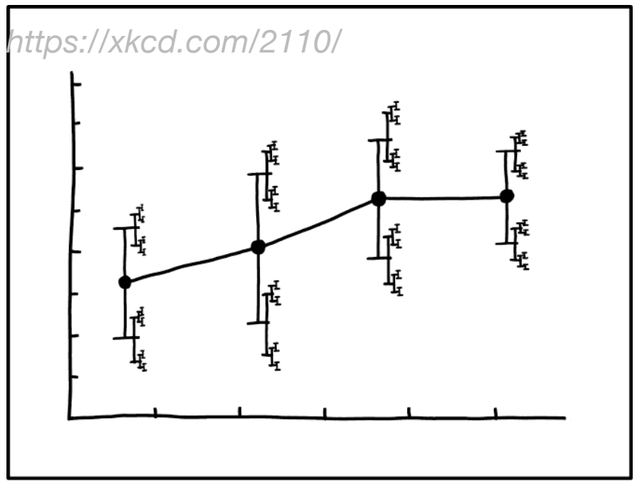
```
geom_bloc
  h <- P(mpg*)
       P(cyl|mpg*)
fill<- cyl
```

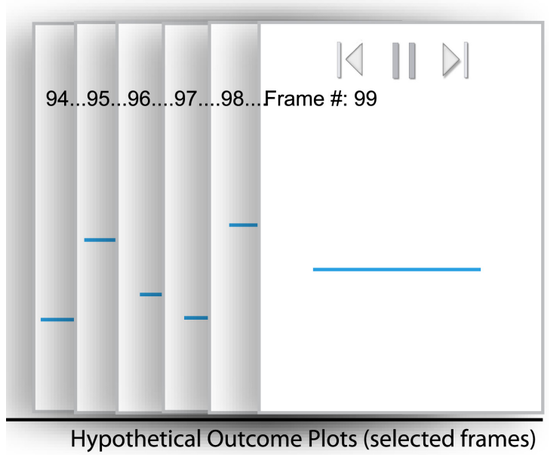
`mpg*`: discretized miles per gallon

```
geom_bloc
  x <- mpg*
  h <- P(mpg*)
       P(cyl|mpg*)
fill<- cyl
```

```
geom_icon
  x <- mpg
  h <- P(mpg*)
       P(cyl|mpg*)
fill<- cyl
```

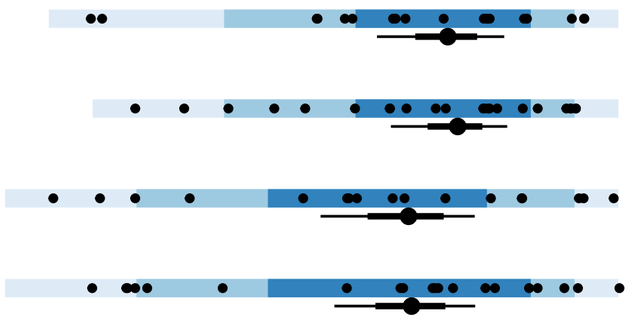
Future work: more uncertainty vizes & systemization



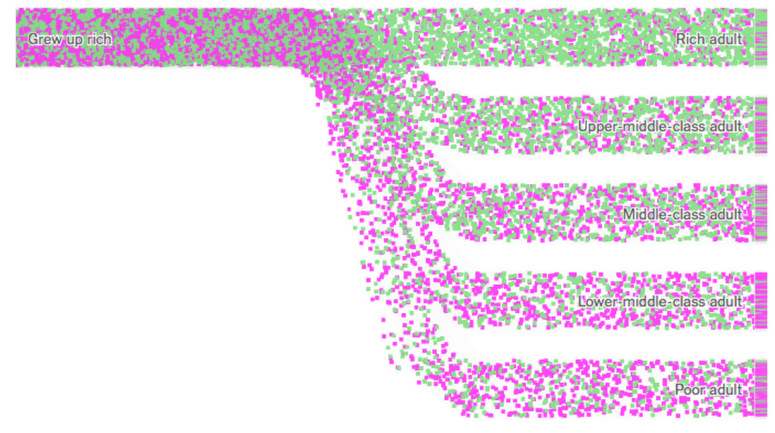


(Hullman, Resnick, and Adar 2015)

- Uncertainty sources: aleotory or epistemic
- Data structure: hierarchical, sequential, etc.
- Summary statistics, confidence intervals, etc.
- Visualization techniques such as linking



https://www.nytimes.com/interactive/2018/03/27/up-shot/make-your-own-mobility-animation.html



Conclusions

PGoG is a visualization grammar for **probabilistic visualizations**, treating probability distributions as first-class citizens. It **shifts our thinking** about specifications for probabilistic visualizations and could facilitate **uncertainty** communication in the future.