Uncertainty visualization in the grammar of graphics

SIADS 542: Presenting uncertainty – Week 1, Lecture 3

Matthew Kay
Assistant Professor
School of Information
University of Michigan

Today

We've established a vocabulary for uncertainty

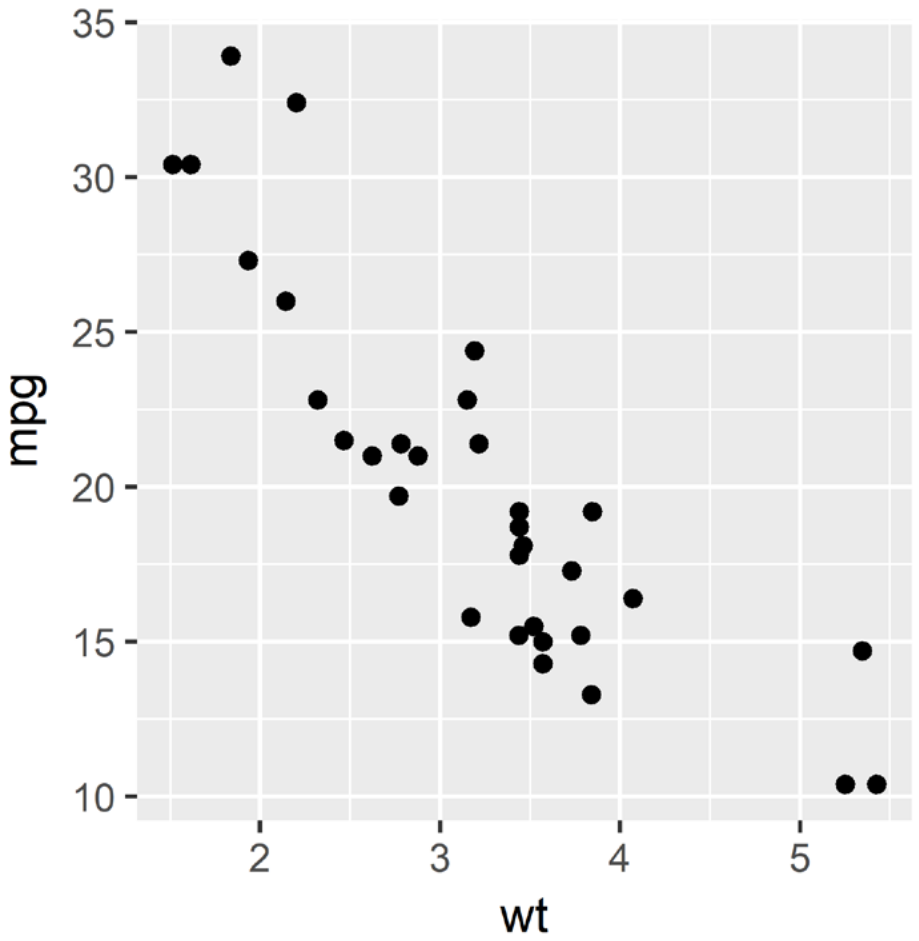
Today

We've established a vocabulary for uncertainty

Now let's connect it to the grammar of graphics

Grammar of graphics: A reminder

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0
Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0
Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0
Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0
Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0
Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0
Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0
Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0
Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0
Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1
Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1
Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1
Toyota Corona	21.5	4	120.1	97	3.70	2.465	20.01	1	0
Dodge Challenger	15.5	8	318.0	150	2.76	3.520	16.87	0	0
AMC Javelin	15.2	8	304.0	150	3.15	3.435	17.30	0	0
Camaro Z28	13.3	8	350.0	245	3.73	3.840	15.41	0	0
Pontiac Firebird	19.2	8	400.0	175	3.08	3.845	17.05	0	0



Let's systematize "turning data into a vis"

data -> ??? -> marks on the screen (or paper)

Let's systematize "turning data into a vis"

data -> ??? -> marks on the screen (or paper)

??? = some vis API

= some way of thinking about vis systematically

Let's systematize "turning data into a vis"

data -> ??? -> marks on the screen (or paper)

??? = New function for every chart type:

```
scatter_plot(data, ...)
bar_chart(data, ...)
...
```

Let's systematize "turning data into a vis"

data -> ??? -> marks on the screen (or paper)

??? = New function for every chart type:

```
scatter_plot(data, ...)
bar_chart(data, ...)
...
```

Every new chart is a new adventure!

Let's systematize "turning data into a vis"

data -> ??? -> marks on the screen (or paper)

??? = ~~New function for every chart type~~

= Low-level drawing functions

```
draw_point(...)
draw_rectangle(...)
```

Let's systematize "turning data into a vis"

data -> ??? -> marks on the screen (or paper)

??? = ~~New function for every chart type~~

= Low-level drawing functions

```
draw_point(...)
draw_rectangle(...)
```

Too low level!

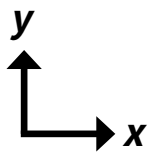
Let's systematize "turning data into a vis"

data -> ??? -> marks on the screen (or paper)

??? = ~~New function for every chart type~~
= ~~Low-level drawing functions~~
= Grammar of graphics
Encode data with visual channels
Display encodings with marks

Channels / encodings

Position



Length

Color

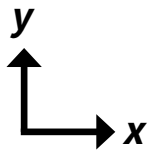
Angle

Area

etc

Channels / encodings -> Marks

Position



Length

Color

Angle

Area

etc

Points

Lines

Bars

etc

Grammar of graphics

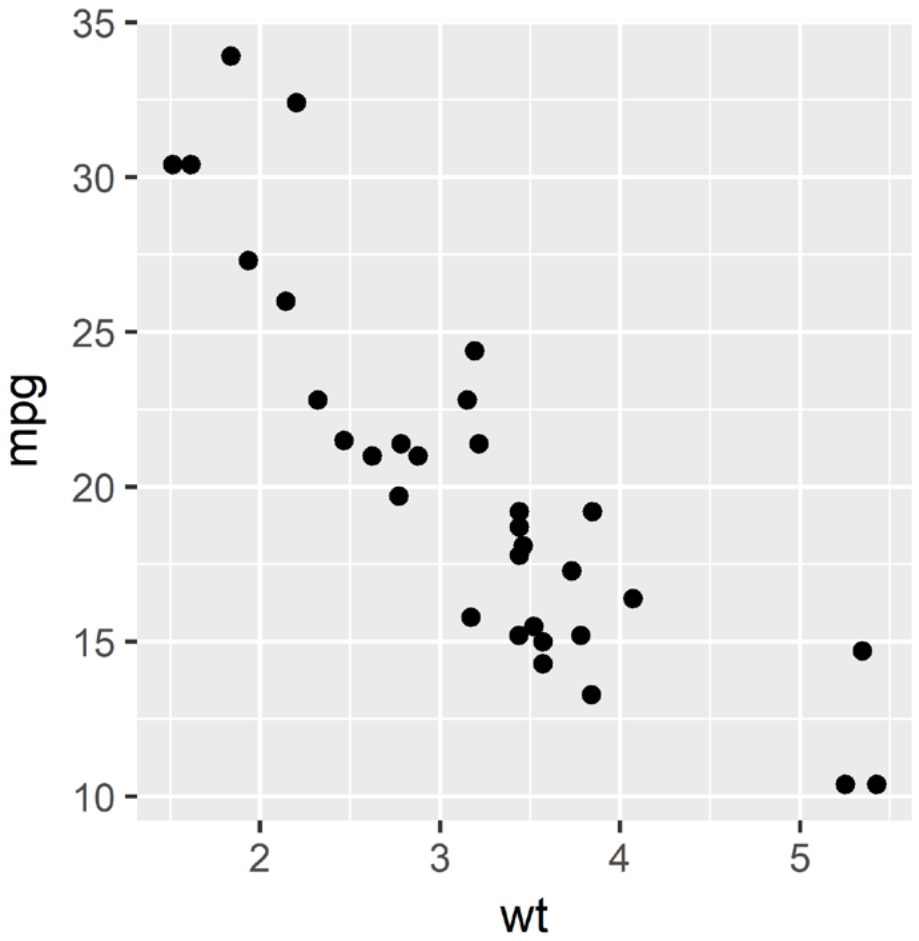
Codifies data types, encodings/channels, marks

Maps data -> channels -> marks

Makes visualization specification straightforward

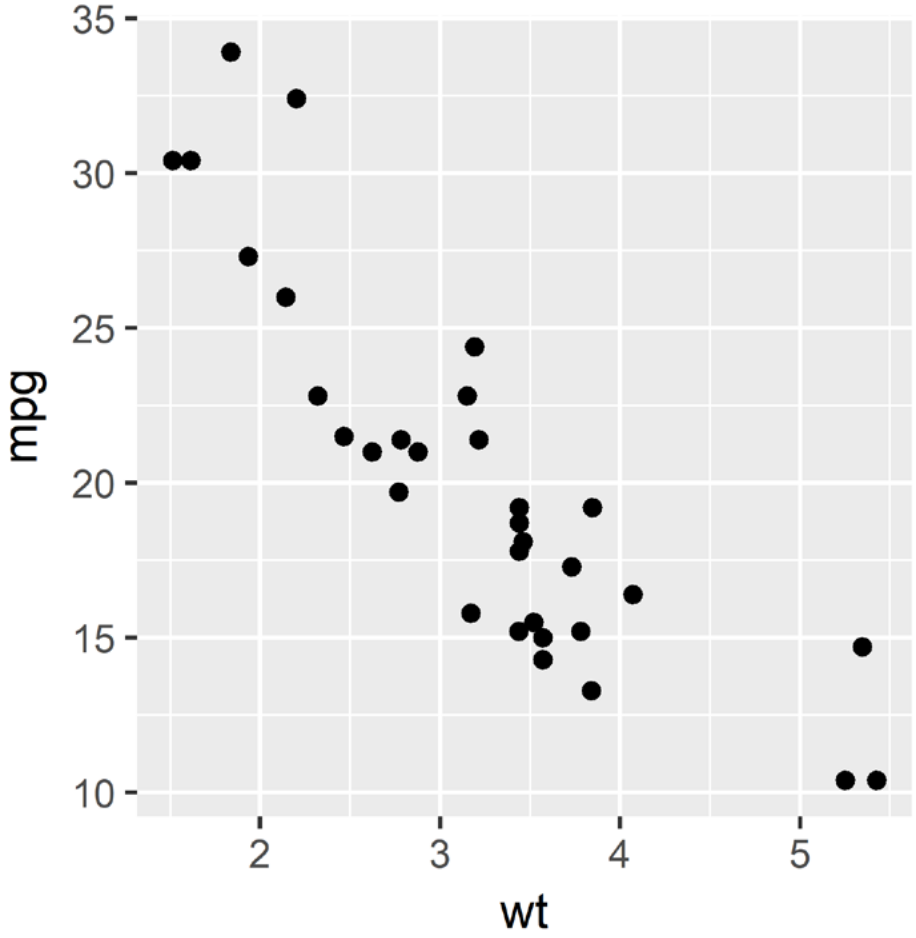
Undergirds Altair, ggplot, Tableau (terms may vary)

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0
Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0
Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0
Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0
Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0
Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0
Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0
Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0
Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0
Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1
Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1
Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1
Toyota Corona	21.5	4	120.1	97	3.70	2.465	20.01	1	0
Dodge Challenger	15.5	8	318.0	150	2.76	3.520	16.87	0	0
AMC Javelin	15.2	8	304.0	150	3.15	3.435	17.30	0	0
Camaro Z28	13.3	8	350.0	245	3.73	3.840	15.41	0	0
Pontiac Firebird	19.2	8	400.0	175	3.08	3.845	17.05	0	0



Grammar of graphics

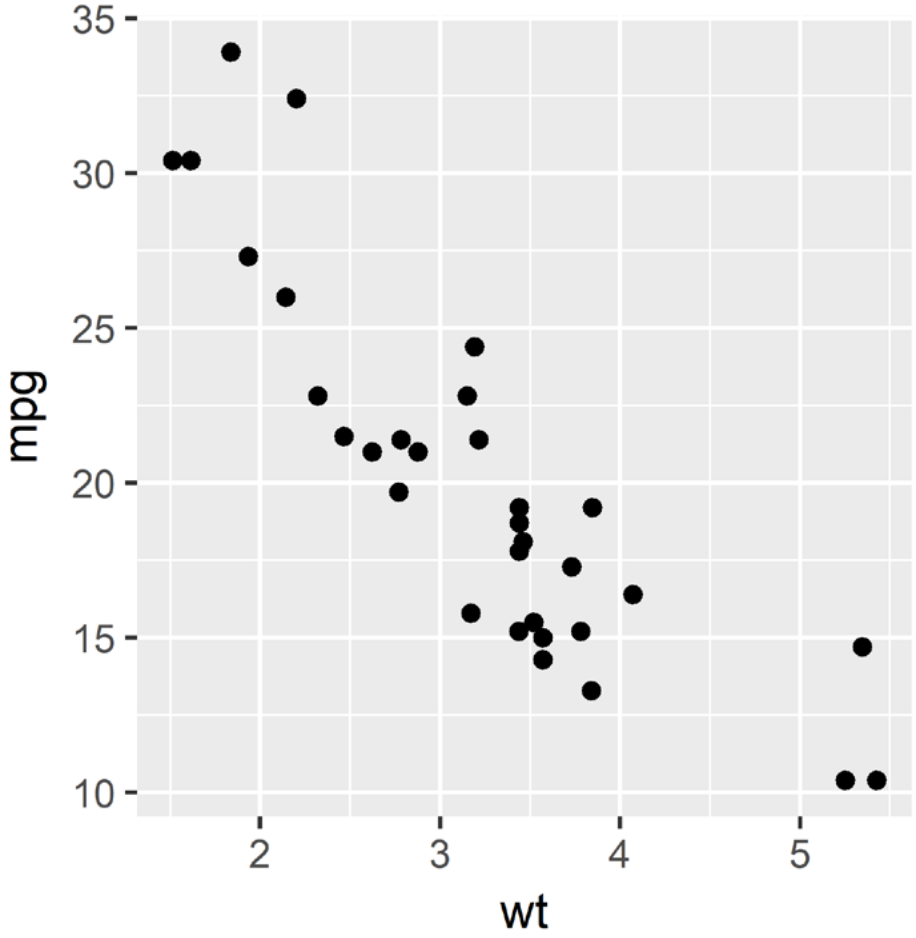
data types, channels, marks



Grammar of graphics

data types, channels, marks

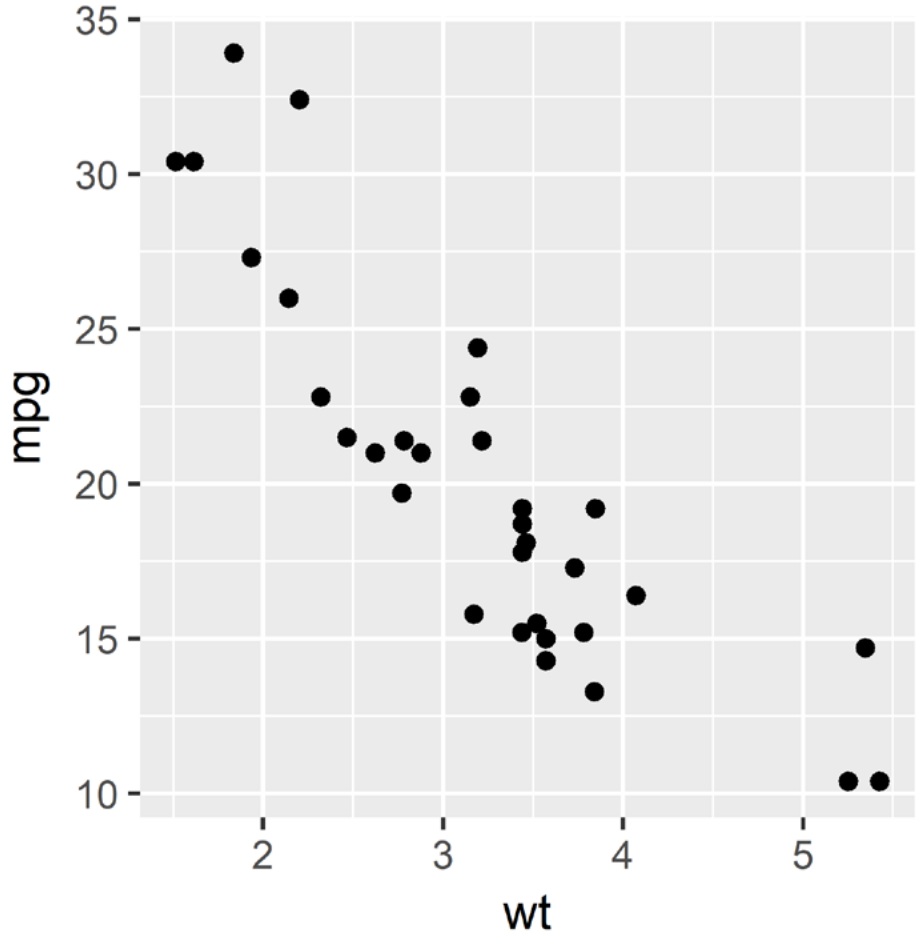
mpg: numeric
wt: numeric



Grammar of graphics

data types, channels, marks

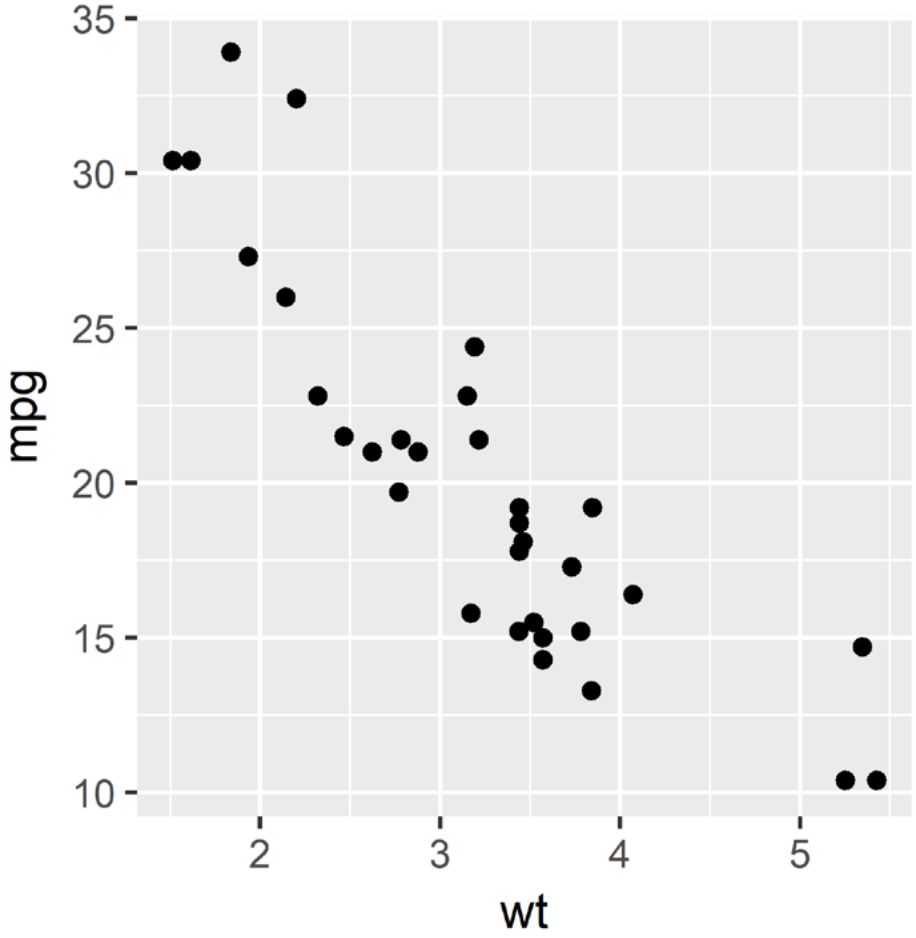
mpg: numeric
wt: numeric
wt → x position
mpg → y position



Grammar of graphics

data types, channels, marks

mpg: numeric
wt: numeric
wt → x position
mpg → y position
mark: point

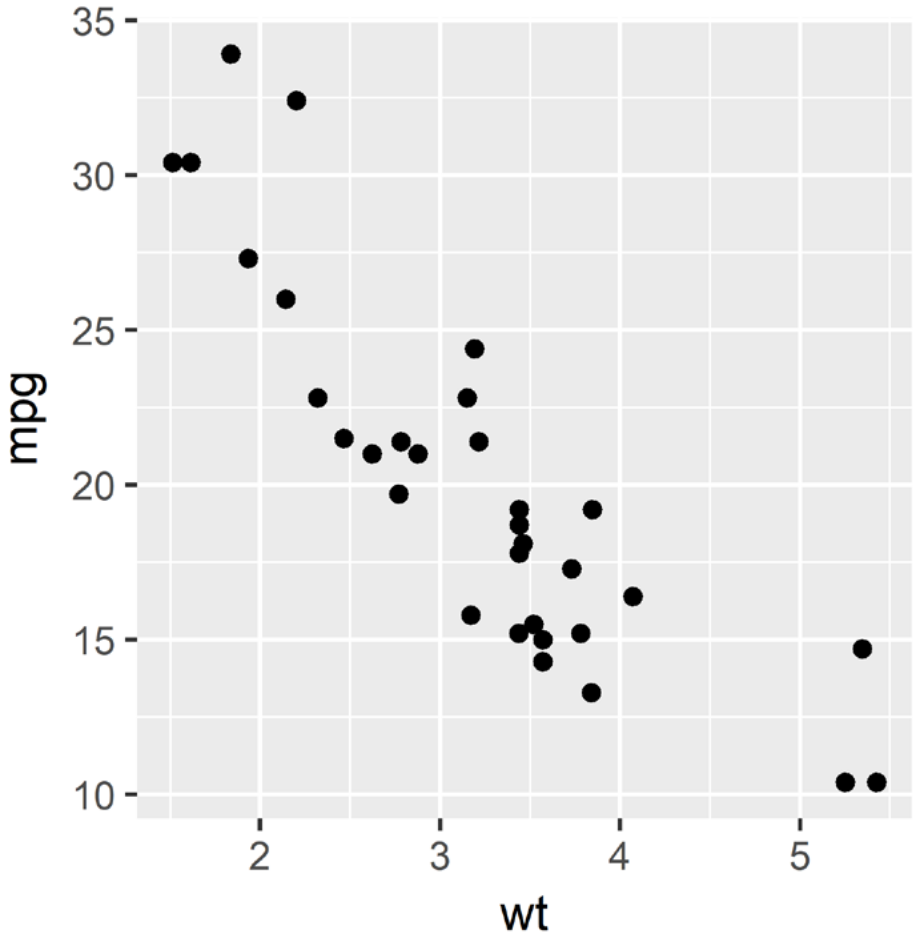


Grammar of graphics

mpg: numeric
wt: numeric

wt → x position
mpg → y position

mark: point



Grammar of graphics

mpg: numeric

wt: numeric

→ manual: nominal

wt → x position

mpg → y position

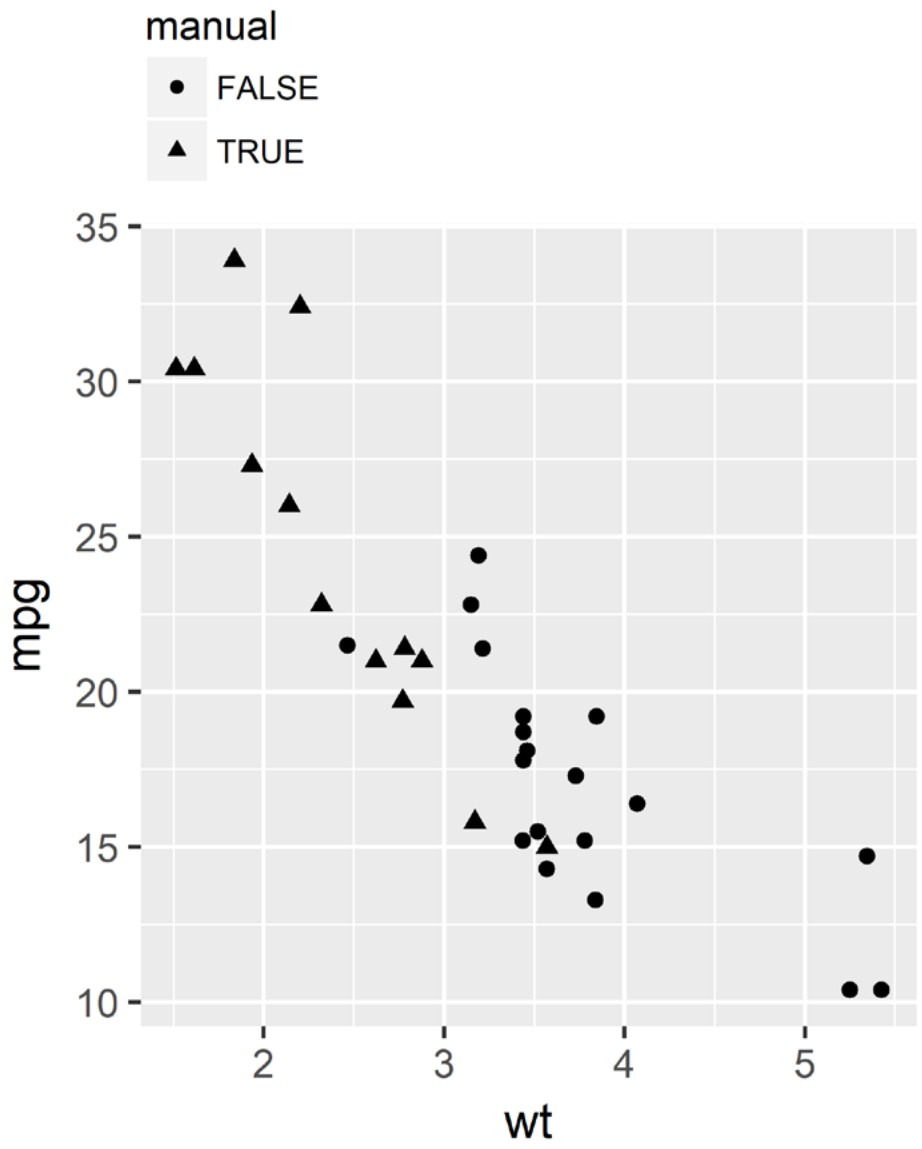
→ manual → color

mark: point

Grammar of graphics

mpg: numeric
wt: numeric
manual: nominal
wt → x position
mpg → y position
manual → shape ←

mark: point



Grammar of graphics

mpg: numeric
wt: numeric
manual: nominal
wt → x position
mpg → y position
manual → color ←
manual → shape ←
mark: point

Translating into Altair...

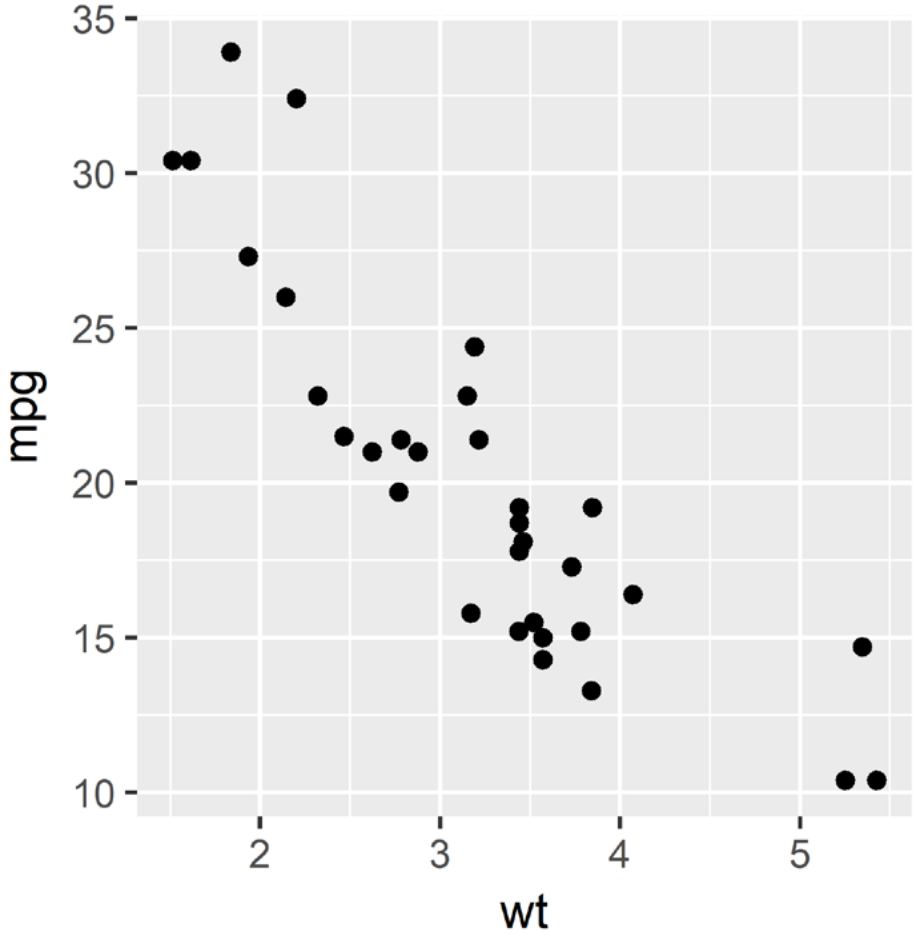
Translating into Altair…

(data, channels, marks)

mpg: numeric
wt: numeric

wt → x position
mpg → y position

mark: point



Translating into Altair…

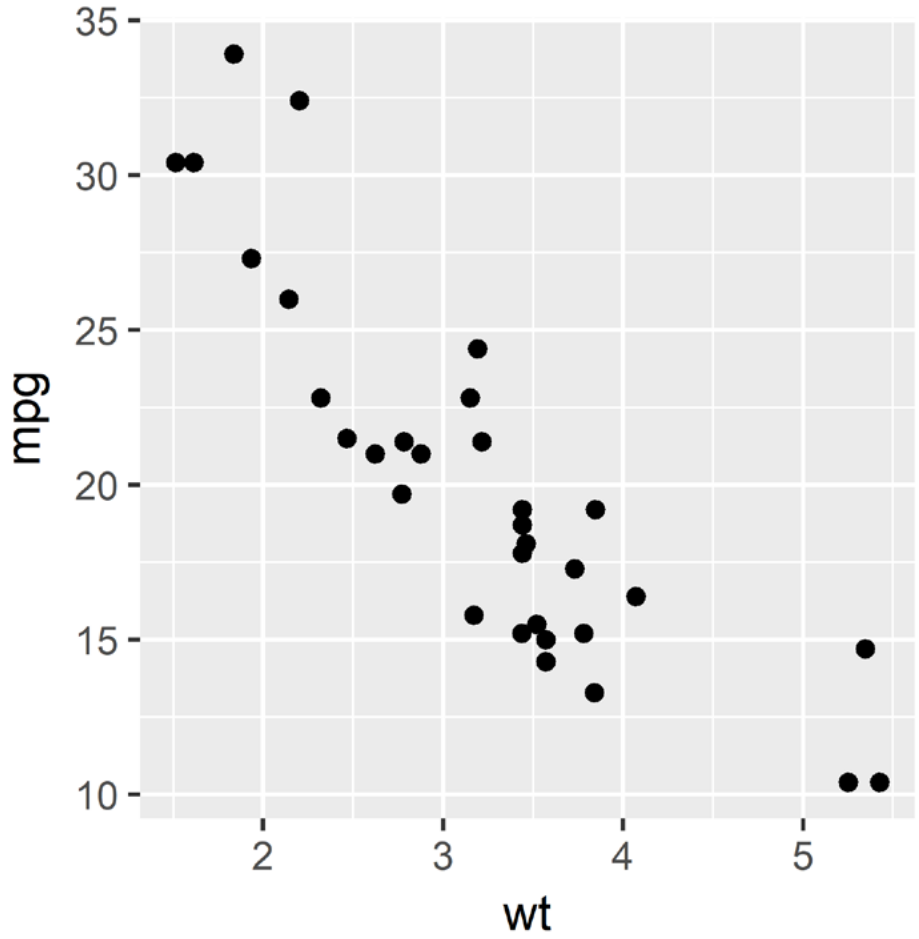
(data, channels, marks)

mpg: numeric
wt: numeric

wt → x position
mpg → y position

mark: point

```
alt.Chart(mtcars)
```



Translating into Altair…

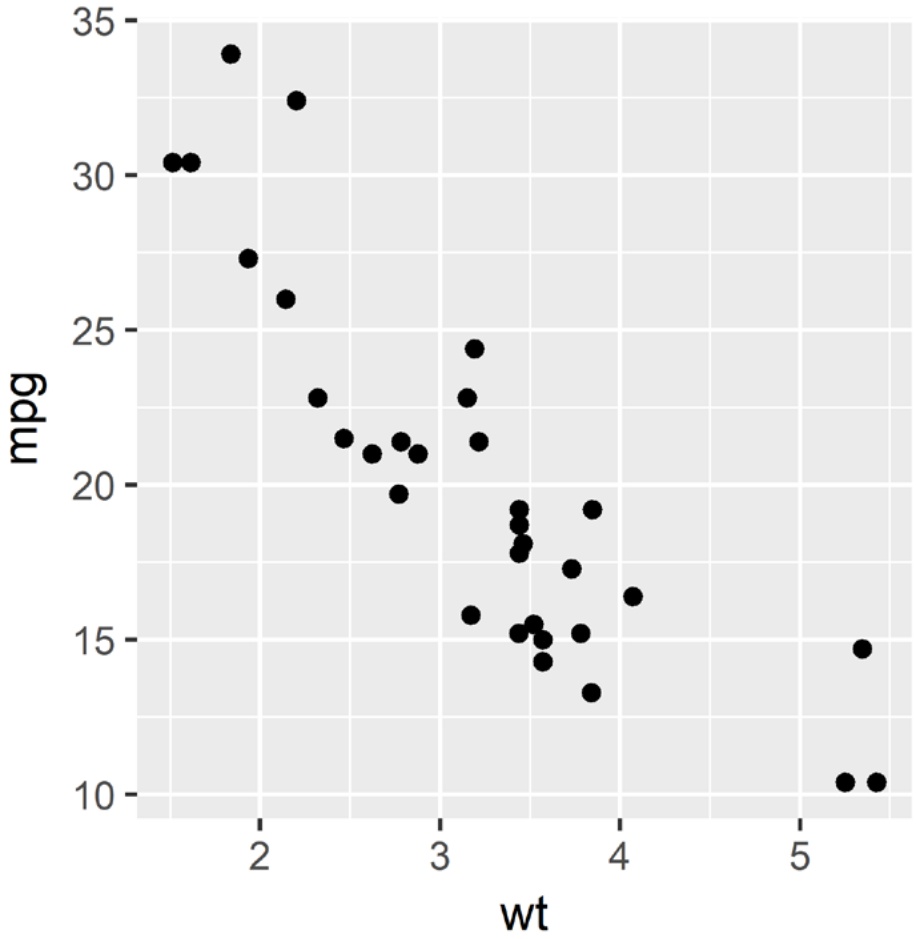
(data, channels, marks)

mpg: numeric
wt: numeric

wt → x position
mpg → y position

mark: point

```
alt.Chart(mtcars)\
    .encode(
        x = 'wt',
        y = 'mpg'
    )
```



Translating into Altair…

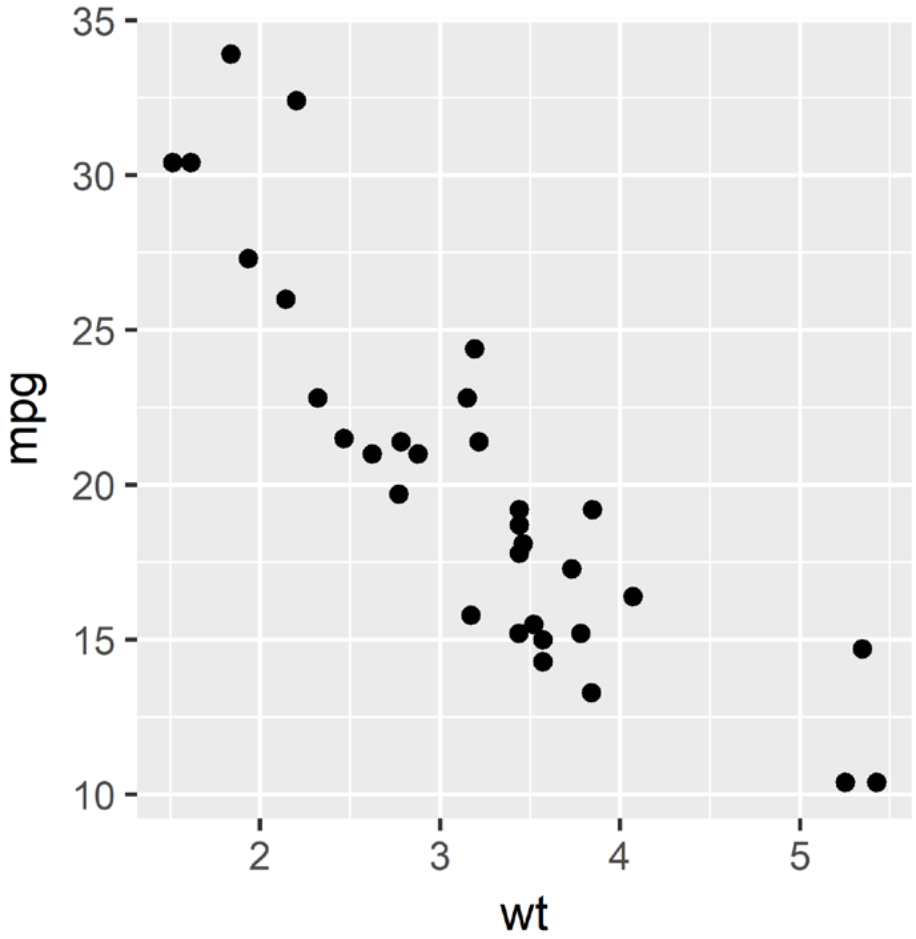
(data, channels, marks)

mpg: numeric
wt: numeric

wt → x position
mpg → y position

mark: point

```
alt.Chart(mtcars)\
    .encode(
        x = 'wt',
        y = 'mpg'
    )\
    .mark_point()
```

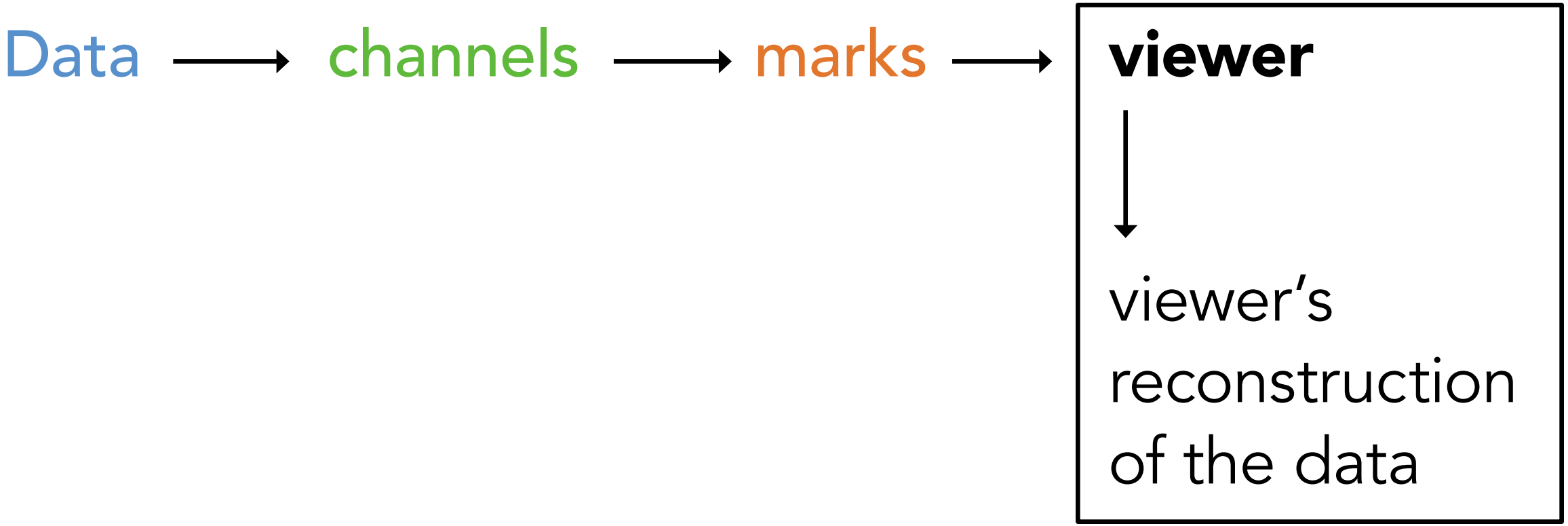


Why is the grammar of graphics useful?

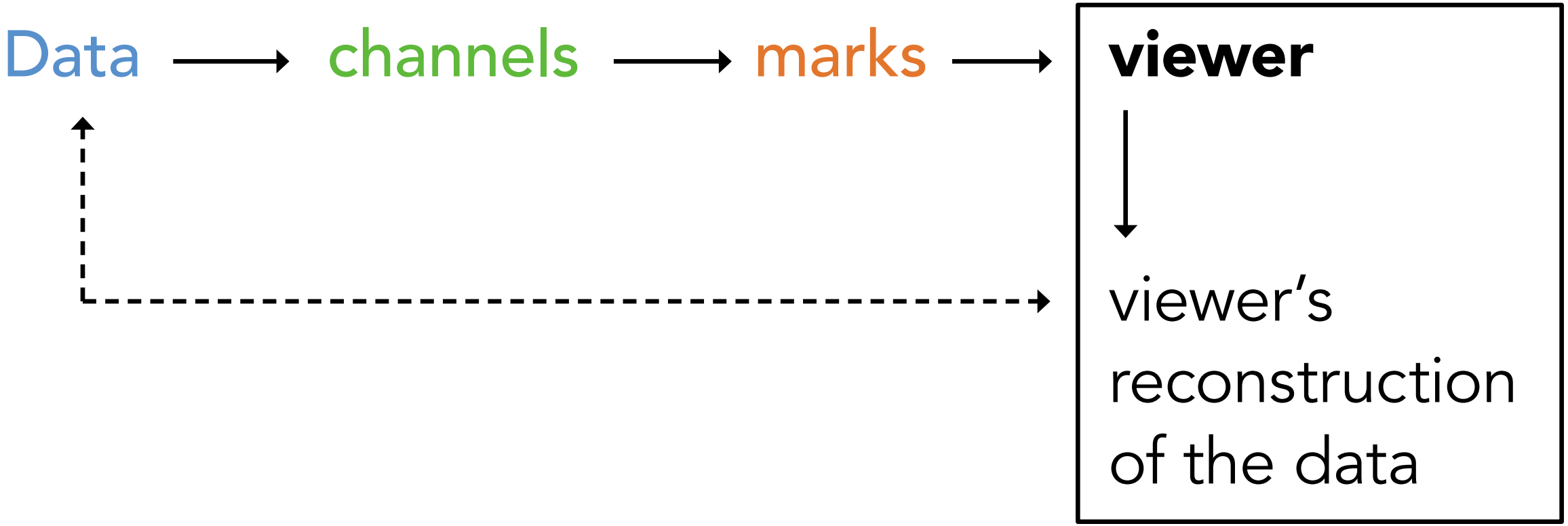
1. Easier to specify many charts, combinations

2. Helps you design/evaluate charts systematically

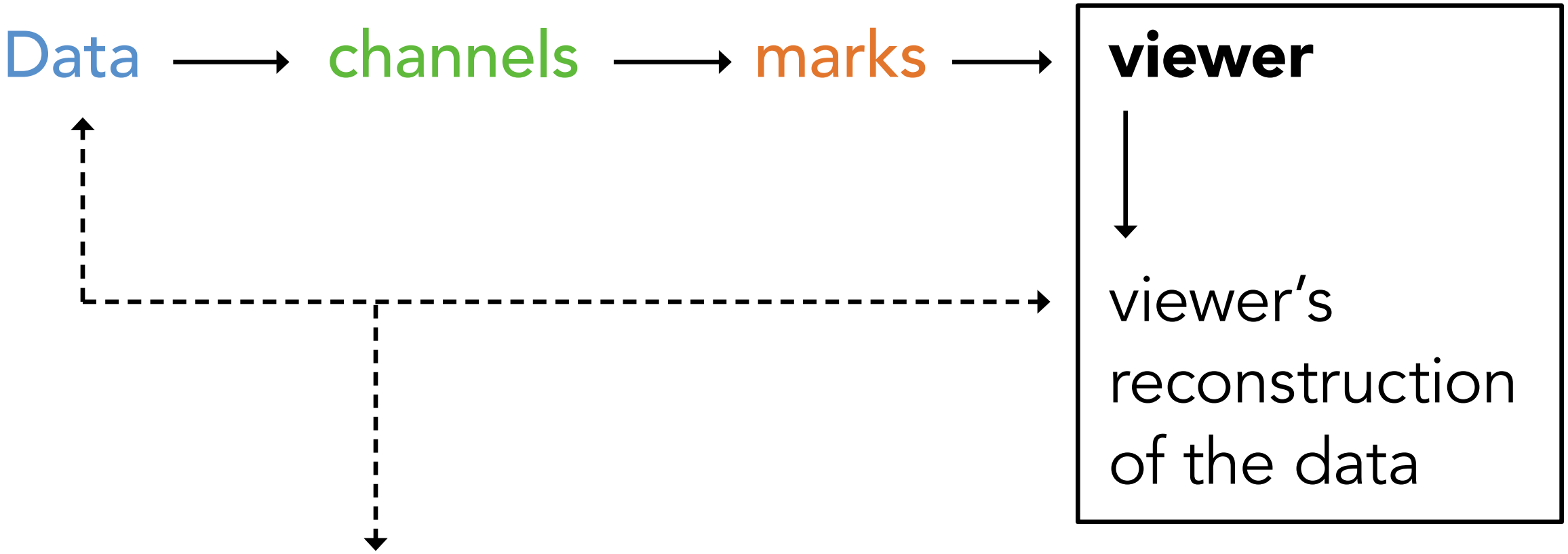
2. Helps you evaluate charts systematically



2. Helps you evaluate charts systematically



2. Helps you evaluate charts systematically



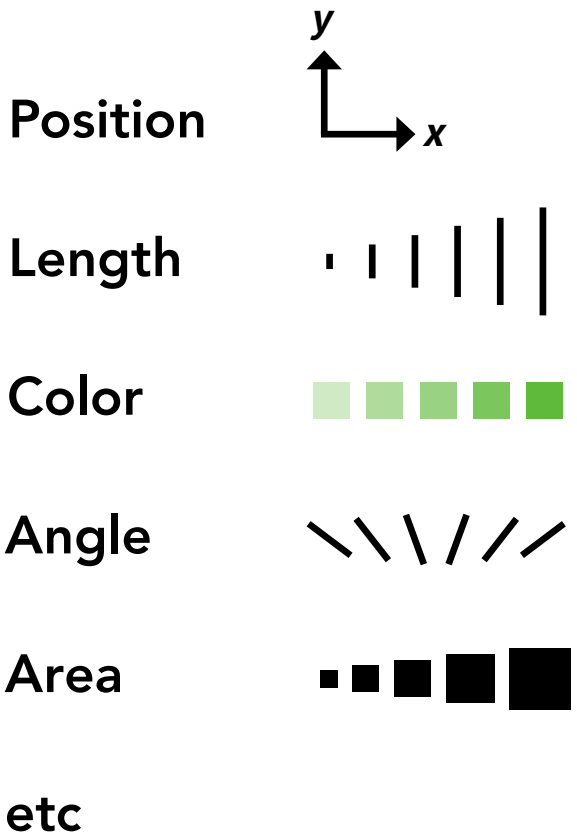
How well do these match, given the channel used?

2. Helps you evaluate charts systematically

e.g.,

How accurately do people perceive position?

How accurately do people perceive area?



Uncertainty in the grammar of graphics

Uncertainty in the grammar of graphics

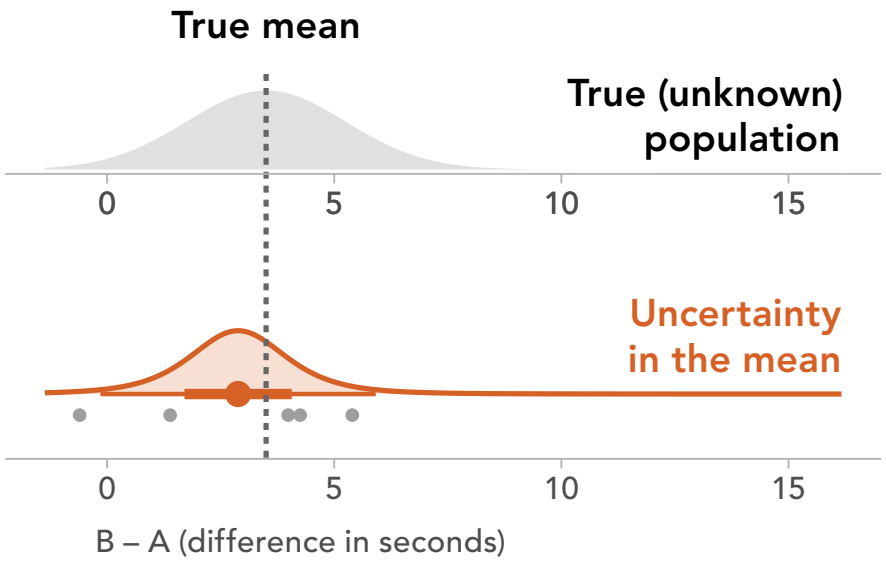
1. **Derive a distribution** describing your uncertainty: a confidence distribution, a posterior probability distribution, etc.

Uncertainty in the grammar of graphics

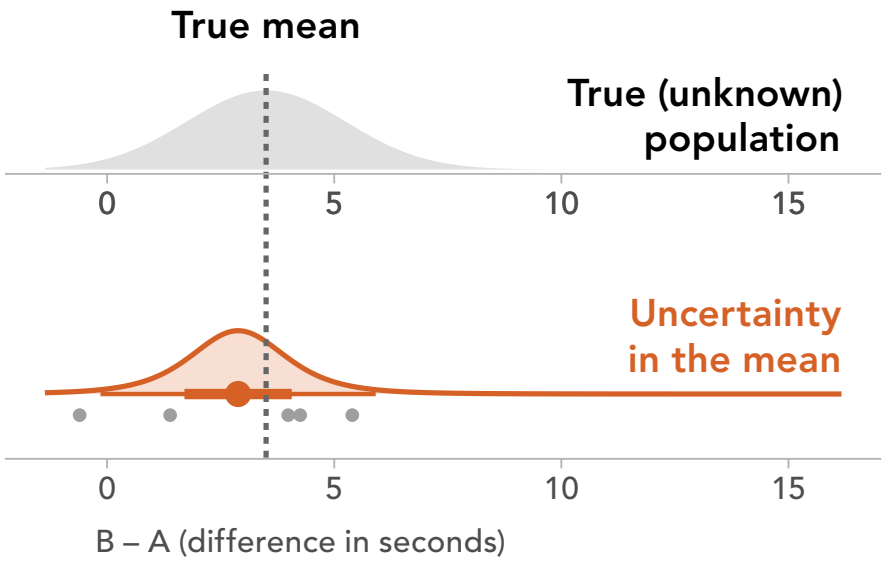
1. **Derive a distribution** describing your uncertainty: a confidence distribution, a posterior probability distribution, etc.

2. **Map properties of the distribution** (location, scale, quantiles, density) onto visual channels

Parameter uncertainty



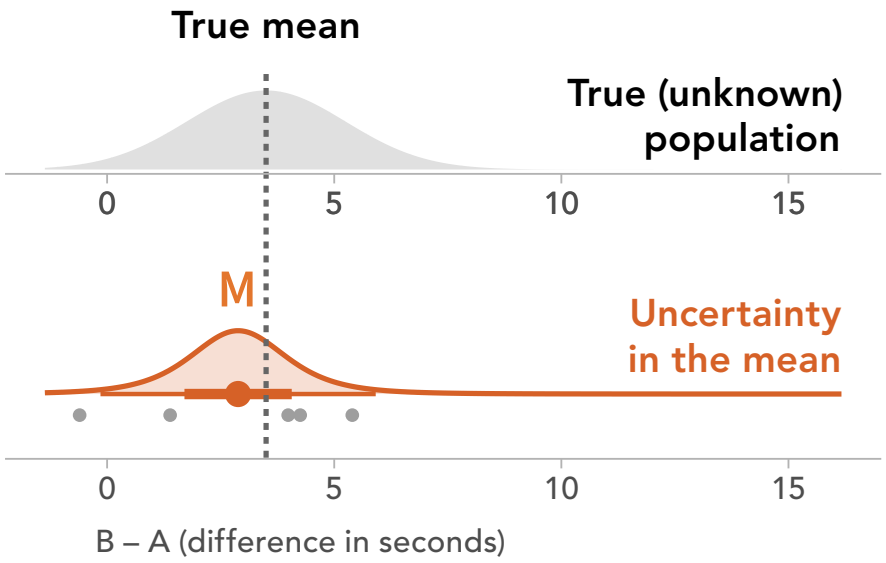
Parameter uncertainty



1. Derive a confidence distribution

2. Map distribution properties onto visual channels

Parameter uncertainty

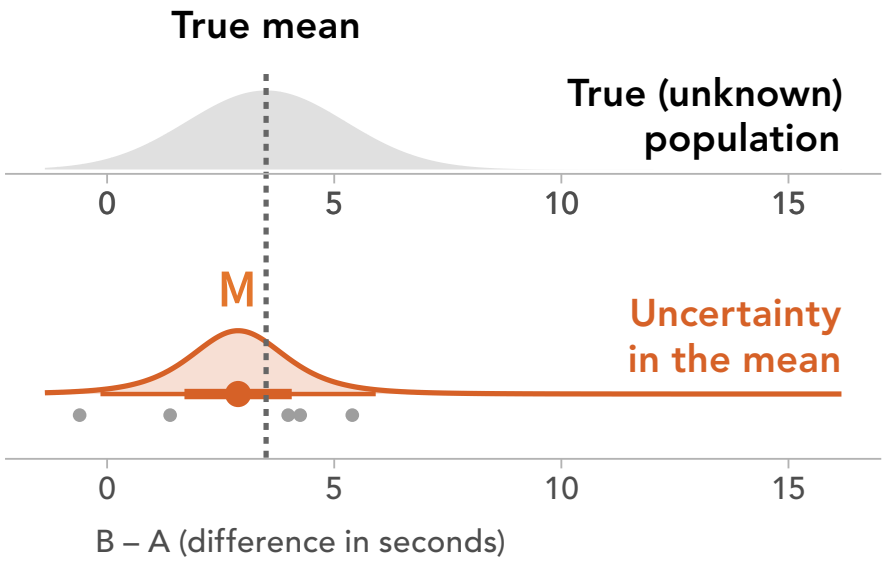


1. Derive a confidence distribution

   $M \sim t(df, \bar{x}, se)$ *scaled/shifted Student's t*

2. Map distribution properties onto visual channels

Parameter uncertainty



1. Derive a confidence distribution

   $M \sim t(df, \bar{x}, se)$ *scaled/shifted Student's t*

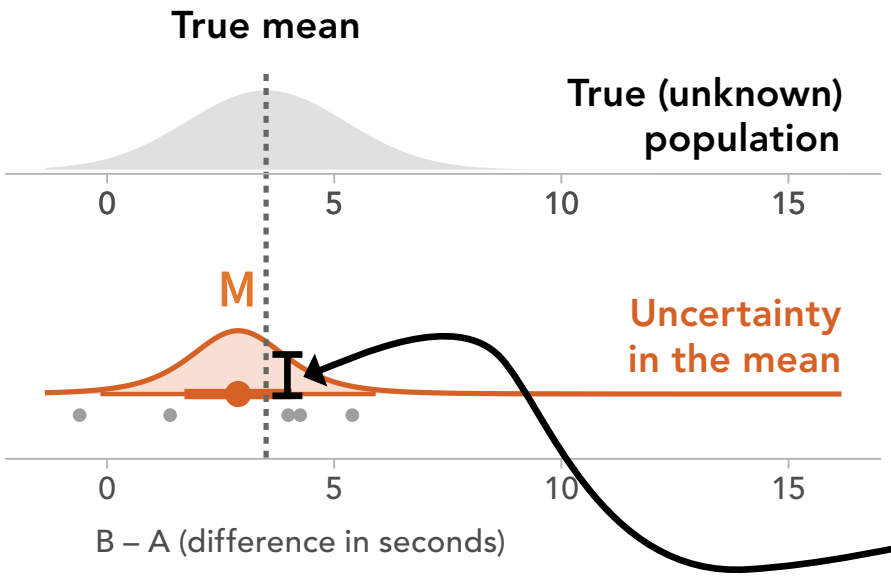
   $f_M(m) = f_t(m \mid df, \bar{x}, se)$ *density*

   $F_M(m) = F_t(m \mid df, \bar{x}, se)$ *CDF*

   $F_M^{-1}(p) = F_t^{-1}(p \mid df, \bar{x}, se)$ *quantile function*

2. Map distribution properties onto visual channels

Parameter uncertainty



1. Derive a confidence distribution

   $M \sim t(df, \bar{x}, se)$ *scaled/shifted Student's t*

   $f_M(m) = f_t(m \mid df, \bar{x}, se)$ *density*

   $F_M(m) = F_t(m \mid df, \bar{x}, se)$ *CDF*

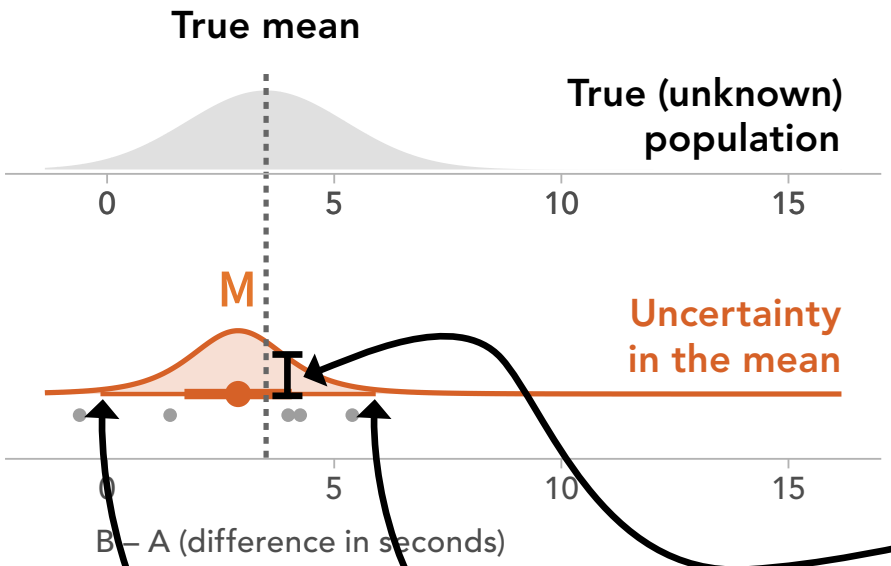
   $F_M^{-1}(p) = F_t^{-1}(p \mid df, \bar{x}, se)$ *quantile function*

2. Map distribution properties onto visual channels

   f_M(x position) → y position

   mark: area

Parameter uncertainty



1. Derive a confidence distribution

$M \sim t(df, \bar{x}, se)$ *scaled/shifted Student's t*

$f_M(m) = f_t(m \mid df, \bar{x}, se)$ *density*

$F_M(m) = F_t(m \mid df, \bar{x}, se)$ *CDF*

$F_M^{-1}(p) = F_t^{-1}(p \mid df, \bar{x}, se)$ *quantile function*

2. Map distribution properties onto visual channels

f_M(x position) → y position

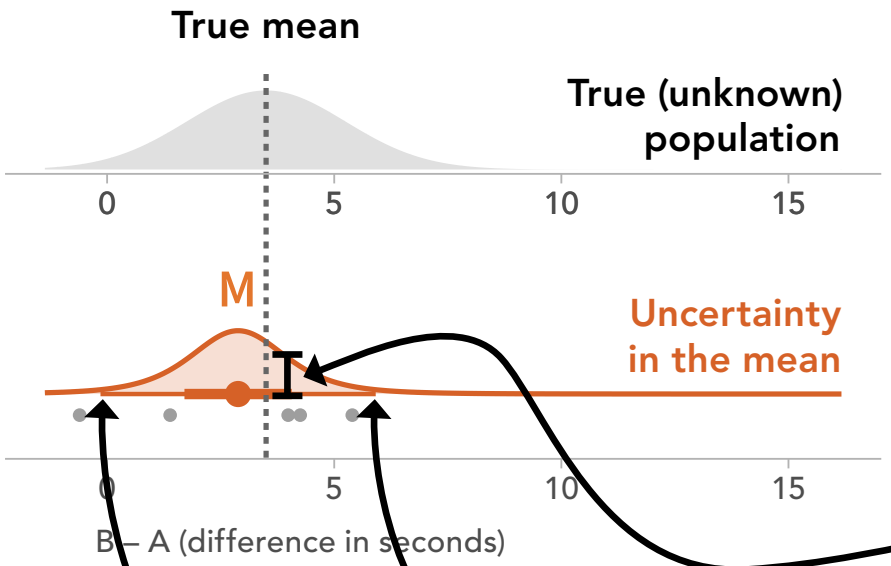
mark: area

$F_M^{-1}(0.975)$ → x_1 position

$F_M^{-1}(0.025)$ → x_2 position

mark: error bar

Parameter uncertainty



1. Derive a confidence distribution

$M \sim t(df, \bar{x}, se)$	*scaled/shifted Student's t*
$f_M(m) = f_t(m \mid df, \bar{x}, se)$	*density*
$F_M(m) = F_t(m \mid df, \bar{x}, se)$	*CDF*
$F_M^{-1}(p) = F_t^{-1}(p \mid df, \bar{x}, se)$	*quantile function*

2. Map distribution properties onto visual channels

f_M(x position) → y position

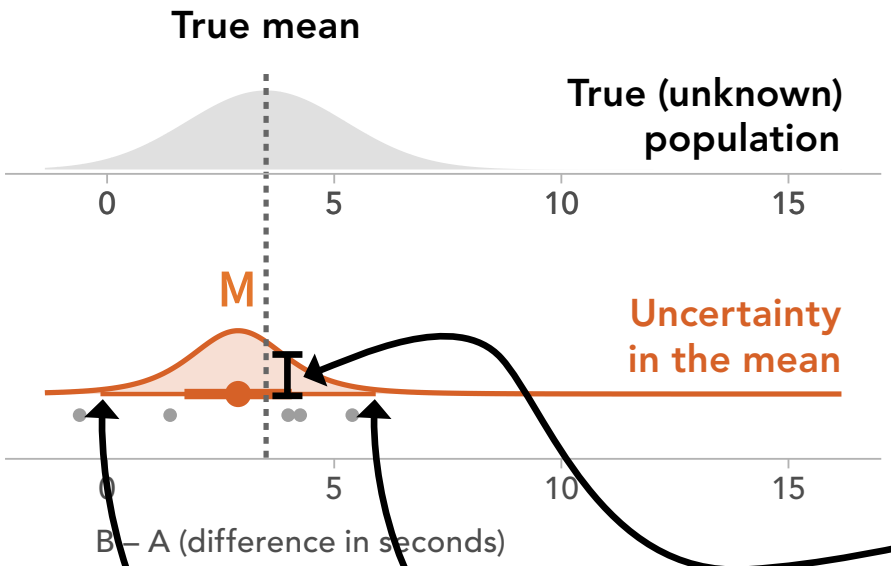
mark: area

$F_M^{-1}(0.975)$ → x_1 position

$F_M^{-1}(0.025)$ → x_2 position

mark: error bar

Parameter uncertainty



1. Derive a bootstrap sampling distribution

   Let $m^{(1)}, \ldots, m^{(k)}$ be bootstrap samples of the mean

   $f_M(m)$ = *kernel density estimator* of all $m^{(k)}$

   $F_M(m)$ = *empirical CDF* of all $m^{(k)}$

   $F_M^{-1}(p)$ = *empirical quantile function* of all $m^{(k)}$

2. Map distribution properties onto visual channels

   f_M(x position) → y position

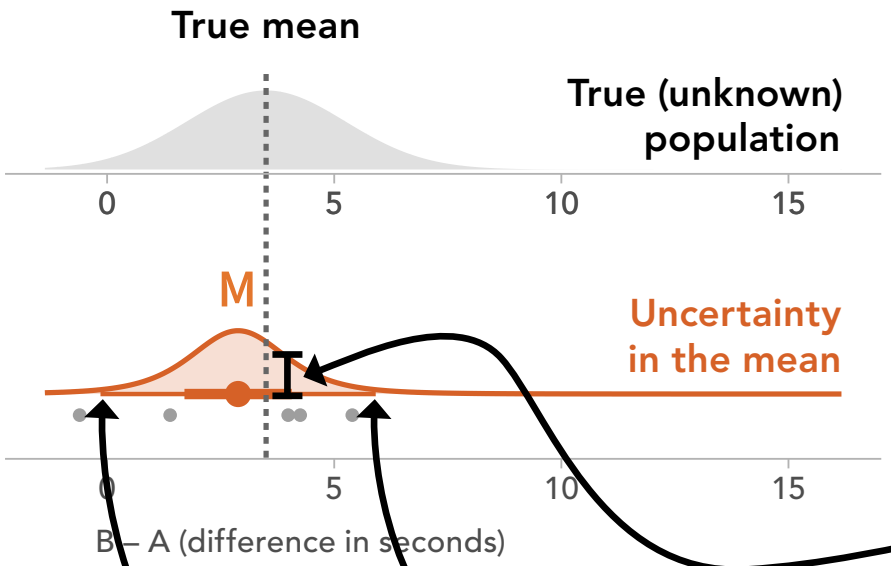
   mark: area

   $F_M^{-1}(0.975)$ → x_1 position

   $F_M^{-1}(0.025)$ → x_2 position

   mark: error bar

Parameter uncertainty



1. Derive a posterior distribution, $p(m \mid data)$

   Let $m^{(1)}, \ldots, m^{(k)}$ be samples from $p(m \mid data)$

   $f_M(m)$ = *kernel density estimator* of all $m^{(k)}$

   $F_M(m)$ = *empirical CDF* of all $m^{(k)}$

   $F_M^{-1}(p)$ = *empirical quantile function* of all $m^{(k)}$

2. Map distribution properties onto visual channels

   f_M(x position) → y position

   mark: area

   $F_M^{-1}(0.975)$ → x_1 position

   $F_M^{-1}(0.025)$ → x_2 position

   mark: error bar

Summing up

Uncertainty in the grammar of graphics

1. **Derive a distribution** describing your uncertainty: a confidence distribution, a posterior probability distribution, etc.

2. **Map properties of the distribution** (location, scale, quantiles, density) onto visual channels

This approach allows us to...

1. **Abstract away** differences in our small worlds (frequentist, Bayesian, parametric, nonparametric...)

2. **Reason about visualization effectiveness** using the grammar of graphics and visual perception (Week 2+)

Roadmap

Week 1: Introduction to uncertainty visualization
Types of uncertainty; small vs large world

Week 2 & 3: Small world uncertainty
Continuous uncertainty encodings
Frequency-framing uncertainty encodings

Week 4: Large world uncertainty

Uncertainty visualization in the grammar of graphics

SIADS 542: Presenting uncertainty – Week 1, Lecture 3

Matthew Kay
Assistant Professor
School of Information
University of Michigan