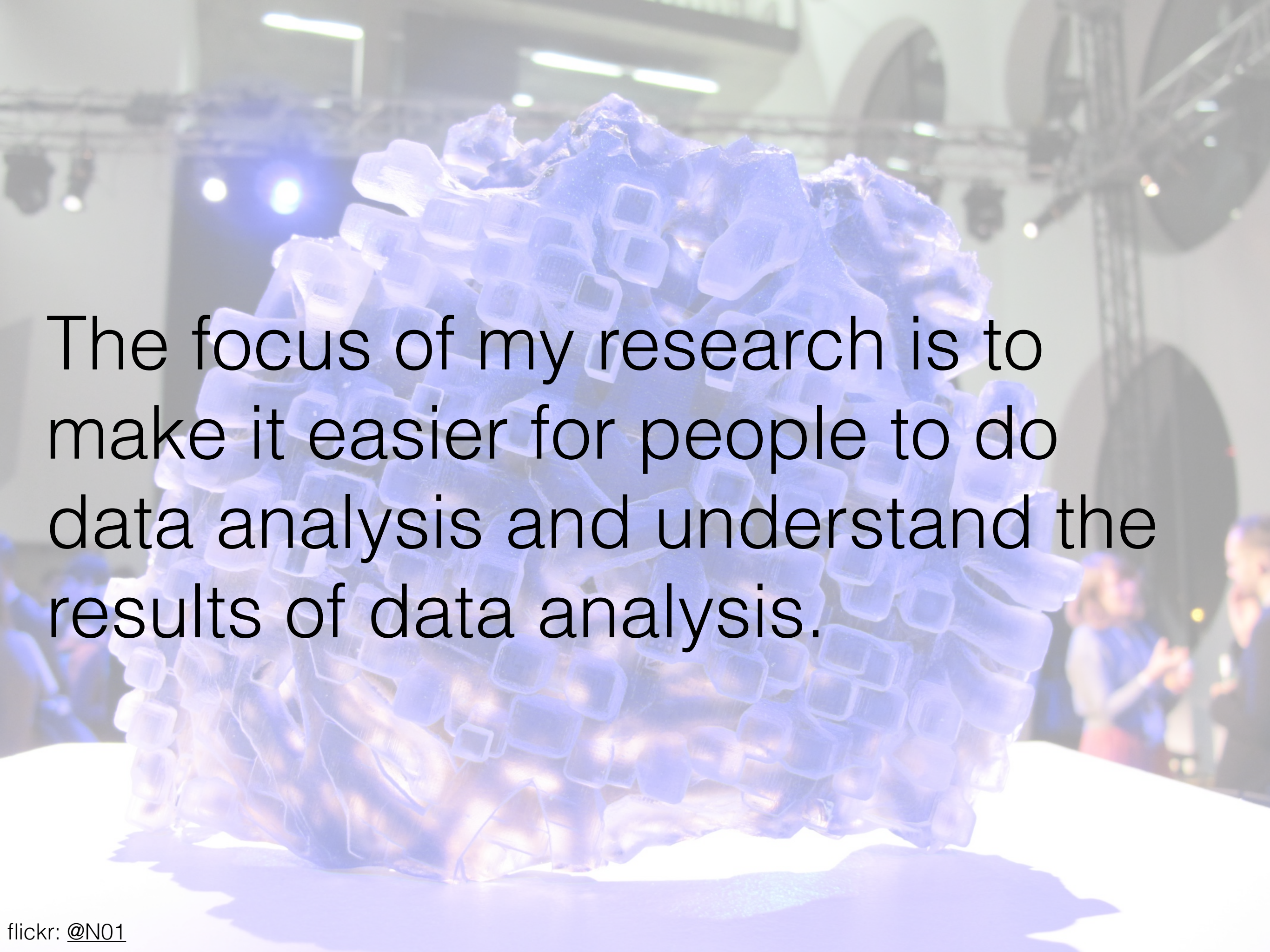




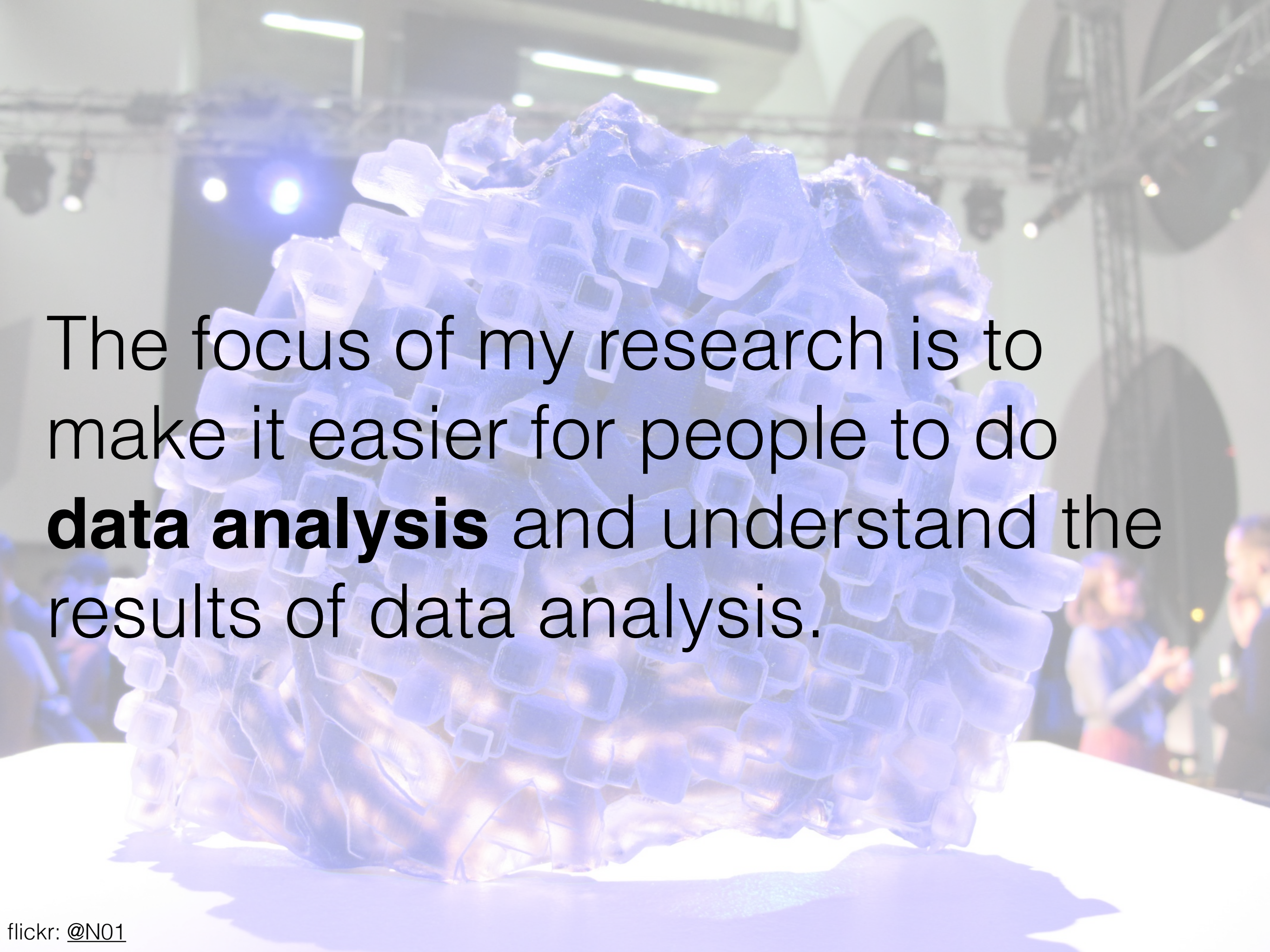
Interfacing with data

Amelia McNamara ([@AmeliaMN](#))

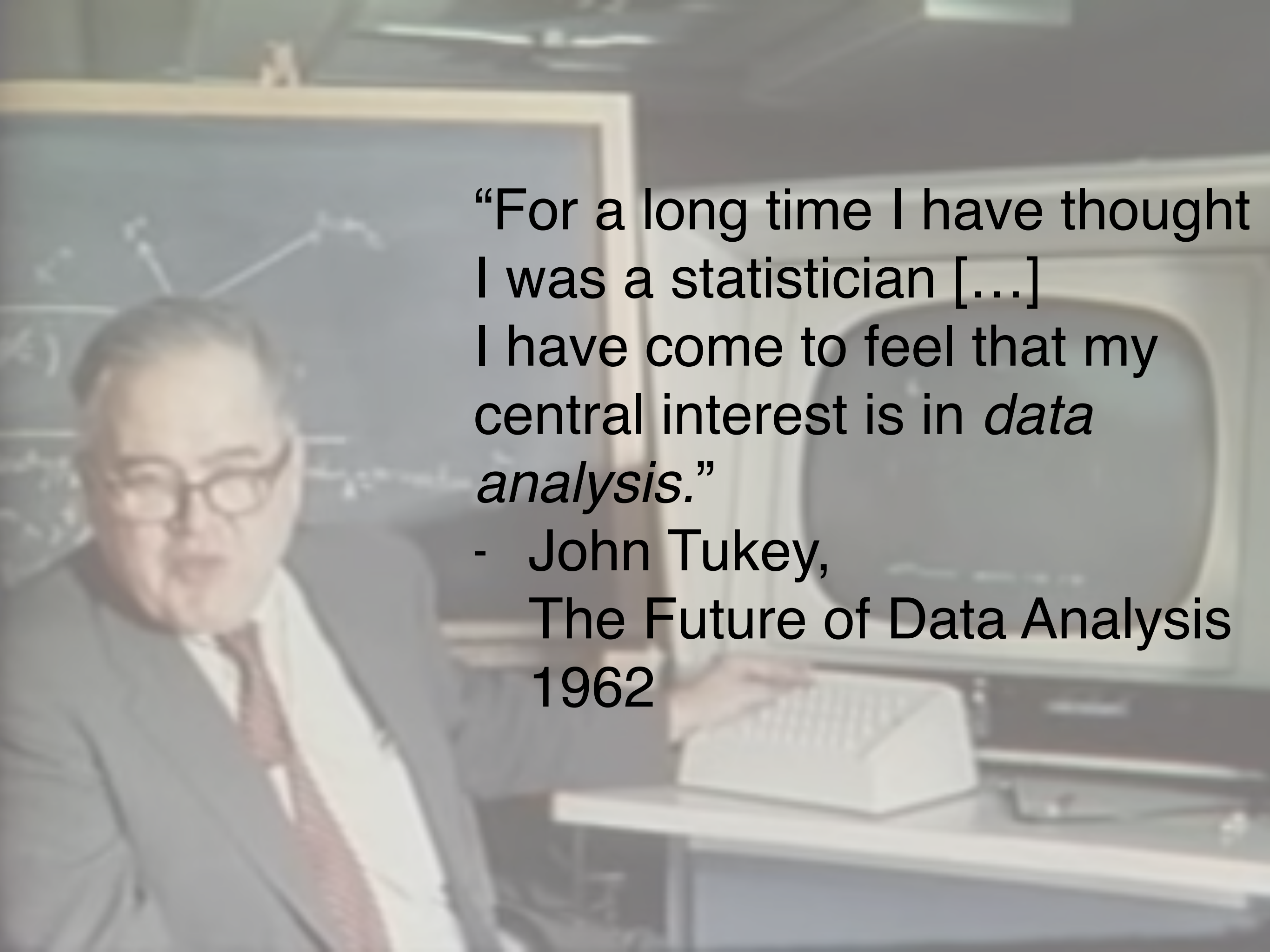
Visiting Assistant Professor of Statistical and Data Sciences
Smith College Northampton, MA, USA



The focus of my research is to make it easier for people to do data analysis and understand the results of data analysis.



The focus of my research is to make it easier for people to do **data analysis** and understand the results of data analysis.



“For a long time I have thought I was a statistician [...] I have come to feel that my central interest is in *data analysis*.”

- John Tukey,
The Future of Data Analysis
1962

CONTENTS

I INTRODUCTORY

| | |
|--|----|
| 1. The Scope of Statistics | 1 |
| 2. General Method, Calculation of Statistics | 6 |
| 3. The Qualifications of Satisfactory Statistics | 11 |
| 4. Scope of this Book | 16 |
| 5. Historical Note | 20 |

II DIAGRAMS

| | |
|---|----|
| 7. Diagrams | 24 |
| 8. Time Diagrams, Growth Rate, and Relative Growth Rate | 24 |
| 9. Correlation Diagrams | 29 |
| 10. Frequency Diagrams | 33 |
| 10.1 Transformed Frequencies | 37 |

III DISTRIBUTIONS

| | |
|--|----|
| 11. Distributions | 41 |
| 12. The Normal Distribution | 43 |
| 13. Fitting the Normal Distribution (Ex. 2) | 45 |
| 14. Test of Departure From Normality (Ex. 3) | 52 |
| 15. Discontinuous Distributions | 54 |
| 16. Small Samples of a Poisson Series (Ex. 4) | 57 |
| 17. Presence and Absence of Organisms in Samples | 61 |
| 18. The Binomial Distribution (Ex. 5, 6) | 63 |
| 19. Small Samples of the Binomial Series (Ex. 7) | 68 |
| Appendix on Technical Notation and Formulae | 70 |

IV TESTS OF GOODNESS OF FIT, INDEPENDENCE AND HOMOGENEITY; WITH TABLE OF χ^2

| | |
|--|----|
| 20. The χ^2 Distribution (Ex. 8, 9) | 78 |
| 21. Tests of Independence, Contingency Tables (Ex. 10, 11, 12, 13) | 85 |
| 21.01 Yates' Correction for Continuity (Ex. 13.1) | 92 |
| 21.02 The Exact Treatment of 2×2 Tables | 96 |

xiv

CONTENTS

| | |
|---|-----|
| 21.03 Exact Tests based on the χ^2 Distribution (Ex. 14) | 97 |
| 21.1 The Combination of Probabilities from Tests of Significance (Ex. 14.1) | 99 |
| 22. Partition of χ^2 into its Components (Ex. 15, 15.1) | 101 |

V TESTS OF SIGNIFICANCE OF MEANS, DIFFERENCES OF MEANS, AND REGRESSION COEFFICIENTS

| | |
|--|-----|
| 23. The Standard Error of the Mean (Ex. 16, 17, 18) | 114 |
| 24. The Significance of the Mean of a Unique Sample (Ex. 19) | 119 |
| 24.1 Comparison of Two Means (Ex. 20, 21) | 122 |
| 25. Regression Coefficients | 129 |
| 26. Sampling Errors of Regression Coefficients (Ex. 22) | 132 |
| 26.1 The Comparison of Regression Coefficients (Ex. 23) | 140 |
| 26.2 The Ratio of Means and Regression Coefficients (Ex. 23.1, 23.2) | 142 |
| 27. The Fitting of Curved Regression Lines | 147 |
| 28. The Arithmetical Procedure of Fitting | 151 |
| 28.1 The Calculation of the Polynomial Values | 154 |
| 29. Regression with Several Independent Variates (Ex. 24) | 156 |
| 29.1 The Omission of an Independent Variate | 166 |
| 29.2 Polynomial Fitting when the Frequencies are Unequal | 168 |

VI THE CORRELATION COEFFICIENT

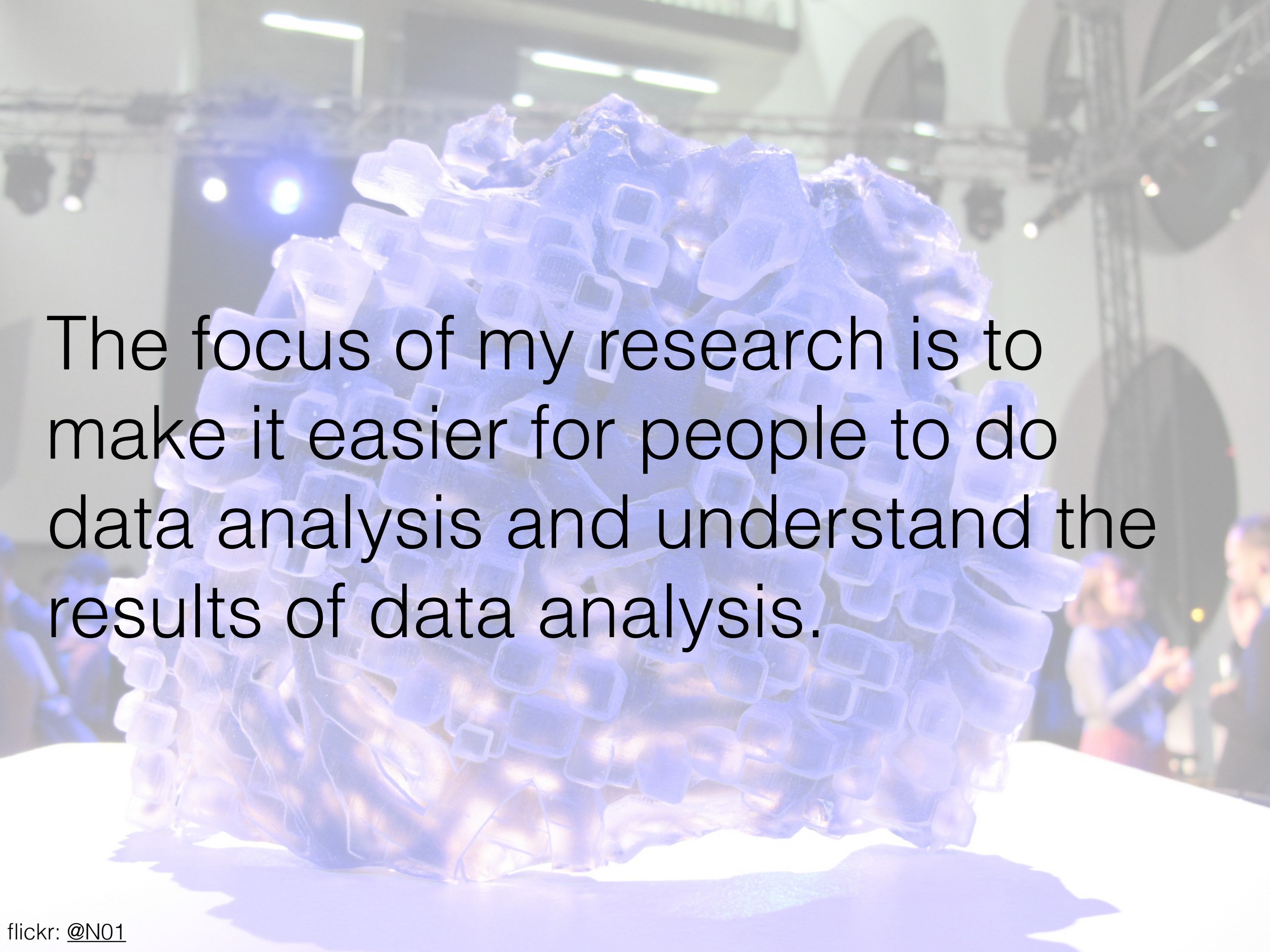
| | |
|--|-----|
| 30. The Correlation Coefficient | 177 |
| 31. The Statistical Estimation of the Correlation (Ex. 25) | 185 |
| 32. Partial Correlations (Ex. 26) | 189 |
| 33. Accuracy of the Correlation Coefficient | 194 |
| 34. The Significance of an Observed Correlation (Ex. 27, 28) | 195 |
| 35. Transformed Correlations (Ex. 29, 30, 31, 32, 33) | 199 |
| 36. Systematic Errors | 207 |
| 37. Correlation between Series | 208 |

VII INTRACLASS CORRELATIONS AND THE ANALYSIS OF VARIANCE

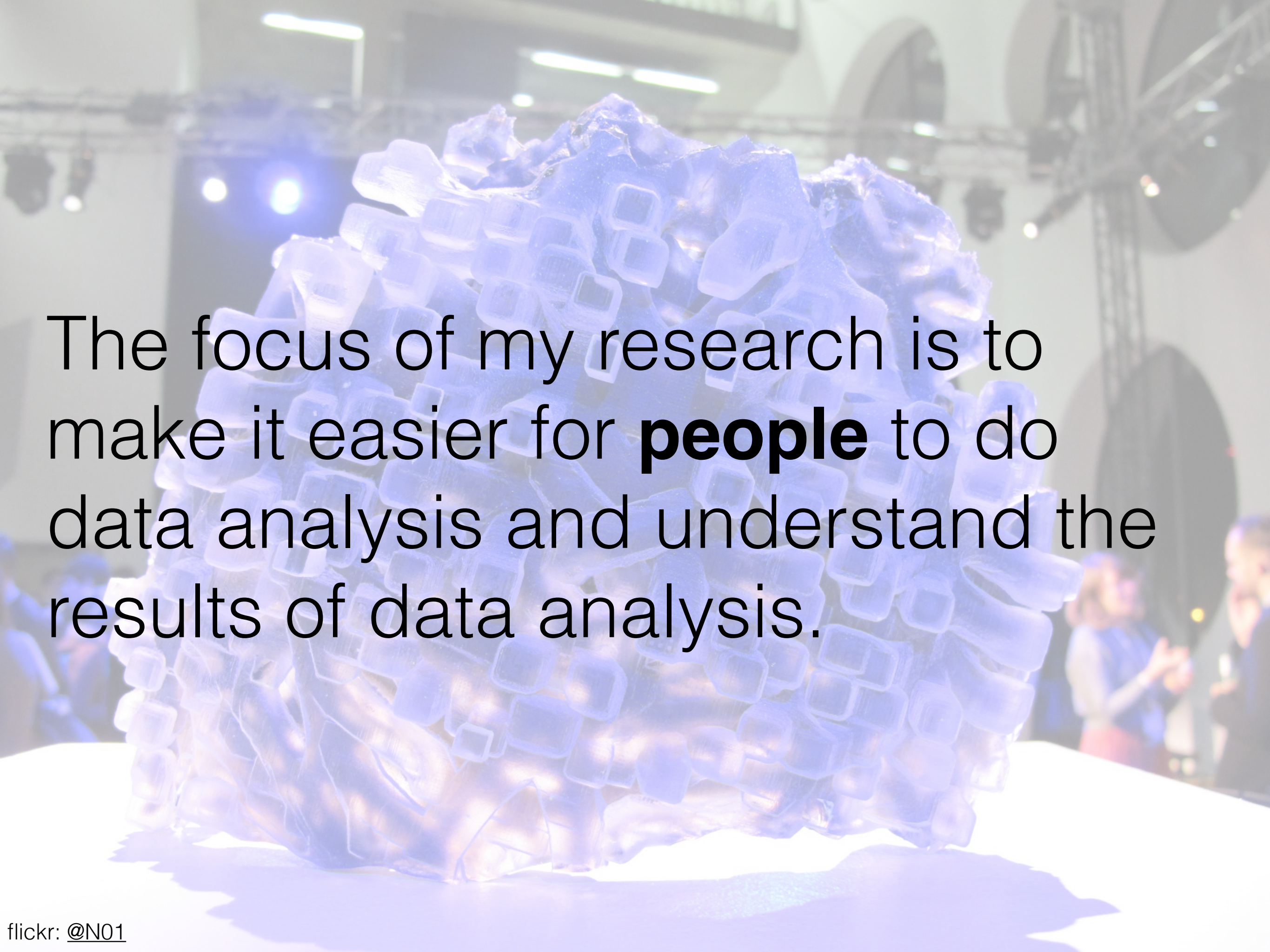
| | |
|--|-----|
| 38. Intraclass Correlations | 213 |
| 39. Sampling Errors of Intraclass Correlations (Ex. 34, 35, 36) | 217 |
| 40. Intraclass Correlation as an Example of the Analysis of Variance | 223 |
| 41. Test of Significance of Difference of Variance (Ex. 37, 38, 39) | 227 |

Contents

| | | | |
|--|------------|--|--|
| Preface | VII | | |
| Chapter 1 Introduction | 1 | | |
| 1.1 Statistics and the Life Sciences | 1 | | |
| 1.2 Examples and Overview | 1 | | |
| Chapter 2 Description of Populations and Samples | 9 | | |
| 2.1 Introduction | 9 | | |
| 2.2 Frequency Distributions: Techniques for Data | 12 | | |
| 2.3 Frequency Distributions: Shapes and Examples | 21 | | |
| 2.4 Descriptive Statistics: Measures of Center | 26 | | |
| 2.5 Boxplots | 32 | | |
| 2.6 Measures of Dispersion | 40 | | |
| 2.7 Effect of Transformation of Variables (Optional) | 50 | | |
| 2.8 Samples and Populations: Statistical Inference | 57 | | |
| 2.9 Perspective | 63 | | |
| Chapter 3 Random Sampling, Probability, and the Binomial Distribution | 71 | | |
| 3.1 Probability and the Life Sciences | 71 | | |
| 3.2 Random Sampling | 71 | | |
| 3.3 Introduction to Probability | 78 | | |
| 3.4 Probability Trees | 83 | | |
| 3.5 Probability Rules (Optional) | 88 | | |
| 3.6 Density Curves | 93 | | |
| 3.7 Random Variables | 96 | | |
| 3.8 The Binomial Distribution | 102 | | |
| 3.9 Fitting a Binomial Distribution to Data (Optional) | 112 | | |
| Chapter 4 The Normal Distribution | 119 | | |
| 4.1 Introduction | 119 | | |
| 4.2 The Normal Curves | 122 | | |
| 4.3 Areas Under a Normal Curve | 123 | | |
| 4.4 Assessing Normality | 133 | | |
| 4.5 The Continuity Correction (Optional) | 141 | | |
| 4.6 Perspective | 145 | | |
| Chapter 9 Comparison of Paired Samples | 347 | | |
| 9.1 Introduction | 347 | | |
| 9.2 The Paired-Sample t test and Confidence Interval | 348 | | |
| 9.3 The Paired Design | 358 | | |
| 9.4 The Sign Test | 364 | | |
| 9.5 The Wilcoxon Signed-Rank Test | 372 | | |
| 9.6 Further Considerations in Paired Experiments | 377 | | |
| 9.7 Perspective | 381 | | |
| Chapter 10 Analysis of Categorical Data | 391 | | |
| 10.1 Inference for Proportions: The Chi-Square Goodness-of-Fit Test | 391 | | |
| 10.2 The Chi-Square Test for the 2×2 Contingency Table | 402 | | |
| 10.3 Independence and Association in the 2×2 Contingency Table | 412 | | |
| 10.4 Fisher's Exact Test (Optional) | 422 | | |
| 10.5 The $r \times k$ Contingency Table | 428 | | |
| 10.6 Applicability of Methods | 434 | | |
| 10.7 Confidence Interval for Difference Between Probabilities | 439 | | |
| 10.8 Paired Data and 2×2 Tables (Optional) | 441 | | |
| 10.9 Relative Risk and the Odds Ratio (Optional) | 444 | | |
| 10.10 Summary of Chi-Square Tests | 454 | | |
| Chapter 11 Comparing the Means of Many Independent Samples | 463 | | |
| 11.1 Introduction | 463 | | |
| 11.2 The Basic Analysis of Variance | 467 | | |
| 11.3 The Analysis of Variance Model (Optional) | 476 | | |
| 11.4 The Global F Test | 478 | | |
| 11.5 Applicability of Methods | 484 | | |
| 11.6 Two-Way ANOVA (Optional) | 487 | | |
| 11.7 Linear Combinations of Means (Optional) | 498 | | |
| 11.8 Multiple Comparisons (Optional) | 507 | | |
| 11.9 Perspective | 516 | | |
| Chapter 12 Linear Regression and Correlation | 525 | | |
| 12.1 Introduction | 525 | | |
| 12.2 The Fitted Regression Line | 527 | | |
| 12.3 Parametric Interpretation of Regression: The Linear Model | 541 | | |
| 12.4 Statistical Inference Concerning β_1 | 548 | | |
| 12.5 The Correlation Coefficient | 553 | | |
| 12.6 Guidelines for Interpreting Regression and Correlation | 565 | | |
| 12.7 Perspective | 576 | | |
| 12.8 Summary of Formulas | 586 | | |



The focus of my research is to make it easier for people to do data analysis and understand the results of data analysis.



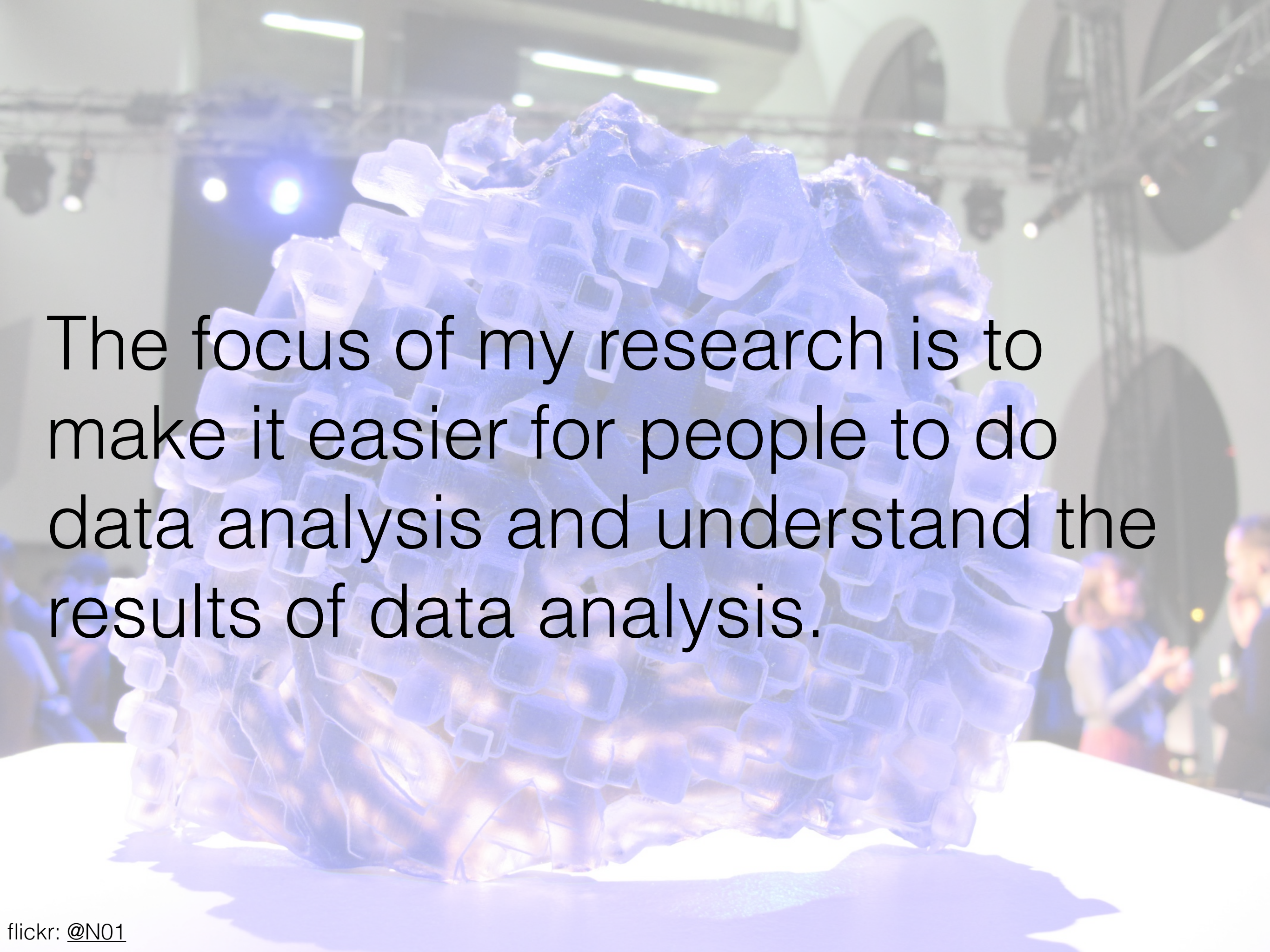
The focus of my research is to make it easier for **people** to do data analysis and understand the results of data analysis.

In particular

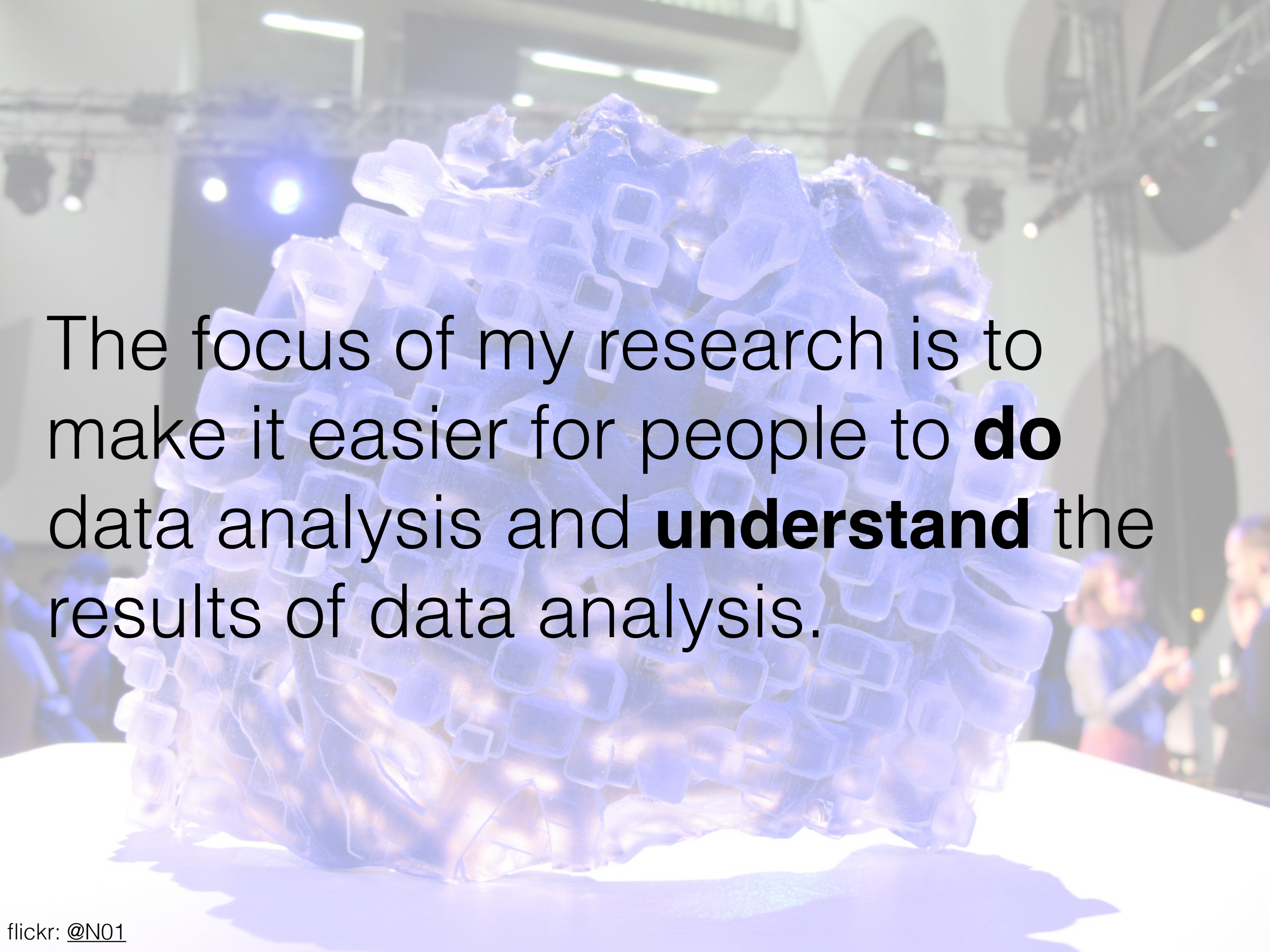
High school teachers/students

College students





The focus of my research is to make it easier for people to do data analysis and understand the results of data analysis.

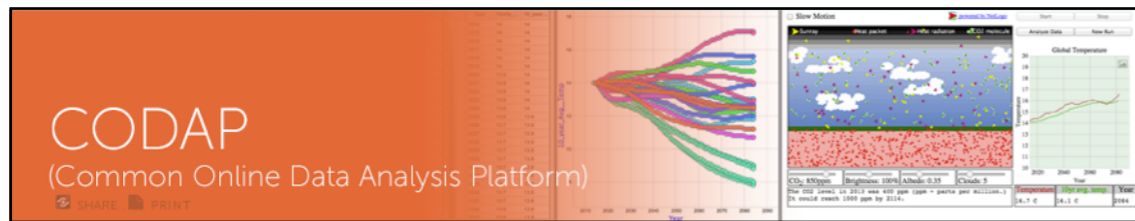
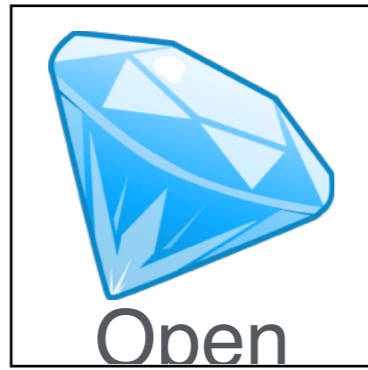


The focus of my research is to make it easier for people to **do** data analysis and **understand** the results of data analysis.

tools

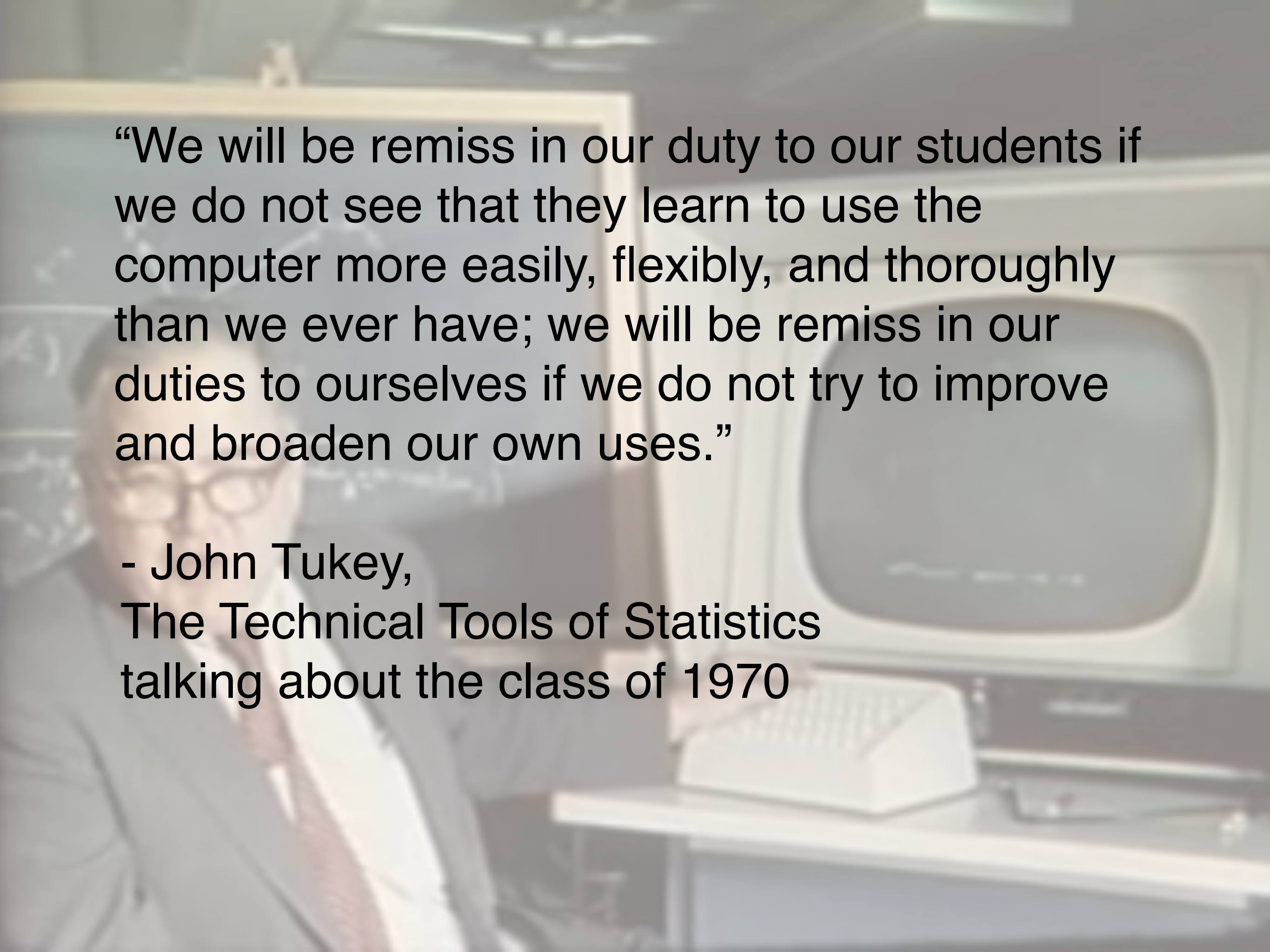


we can push the
boundary with
tools
and/or
curriculum



**curriculum
(and, tools)**





“We will be remiss in our duty to our students if we do not see that they learn to use the computer more easily, flexibly, and thoroughly than we ever have; we will be remiss in our duties to ourselves if we do not try to improve and broaden our own uses.”

- John Tukey,
The Technical Tools of Statistics
talking about the class of 1970

**a modern
statistical
computing
tool**

- Accessibility
- Easy entry for novice users
- Data as a first-order persistent object
- Support for a cycle of exploratory and confirmatory analysis
- Flexible plot creation
- Support for randomization throughout
- Interactivity at every level
- Inherent visual documentation
- Simple support for narrative, publishing, and reproducibility
- Flexibility to build extensions

tools designed for **learning** statistics are typically:

- graphical
- interactive
- intuitive
- supportive of EDA

but:

- don't support reproducibility
- can't handle real data

The logo for Fathom, featuring the word "Fathom" in a bold, blue, sans-serif font. The letter "o" is replaced by a 3D-rendered yellow sphere with a grid pattern.The logo for TinkerPlots, with "Tinker" in orange and "Plots" in blue. A dotted line arches over the text, and a 3D orange sphere with blue dots is positioned between the two words.

StatKey

to accompany [Statistics: Unlocking the Power of Data](#)
by Lock, Lock, Lock, Lock, and Lock

Rossman/Chance Applet Collection

tools designed for **doing** statistics are typically:

- powerful
- flexible
- reproducible
- supportive of extensions

but:

- hard to get started using
- not interactive

The logo for Stata, consisting of the word "Stata" in white, bold, sans-serif font on a dark teal rectangular background.The logo for the R programming language, featuring a stylized blue letter "R" inside a grey oval.The logo for SPSS, with "SPSS" in white, bold, sans-serif font on a red square background. Below it, the text "AN IBM COMPANY" is written in smaller white font.The logo for SAS, featuring a blue stylized "S" followed by "sas" in black, lowercase, sans-serif font. Below it is the tagline "THE POWER TO KNOW" in grey.The logo for MATLAB, with "MATLAB" in black, bold, sans-serif font. Below it is the tagline "The Language of Technical Computing" in a smaller, italicized font.

Easy entry— TinkerPlots

The screenshot displays the TinkerPlots software interface. At the top is a menu bar with 'TinkerPlots', 'File', 'Edit', 'Object', 'Data', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons for data manipulation and visualization. The main workspace is divided into three panels:

- Top Left Panel (Table View):** Shows a table titled 'Snack.csv' with columns for 'Attribute', 'Value', 'Unit', and 'Form...'. The data includes snack costs, locations, and other attributes.
- Bottom Left Panel (Faceted Plot):** A plot showing the distribution of snack costs across different categories. The y-axis is labeled 'SnackCostlabel' and the x-axis is labeled 'SnackLocationlabel'. The plot uses colored circles to represent data points, with percentages shown above each category.
- Right Panel (Grid Plot):** A grid plot showing the relationship between 'SnackCostlabel' (y-axis) and 'SnackLocationlabel' (x-axis). The y-axis categories are 'NOT_DISPLAYED', 'More than \$10.00', 'Less than \$1.00', '\$7.00-\$10.00', '\$5.00-\$7.00', '\$3.00-\$5.00', and '\$1.00-\$3.00'. The x-axis categories are 'Friends' houses', 'Home', 'NOT_DISPLAYED', 'Other', 'Party', 'Restaurant', 'School', 'Vehicle', and 'Work'. The plot uses colored circles to represent data points, with a legend on the right side.

At the bottom left, the text 'TinkerPlots® version 2.1' is visible.

Extensible—

```
Terminal Shell Edit View Window Help
amelia — R — 80x24
Last login: Wed Apr 16 15:39:29 on ttys000
Amelias-MacBook-Air:~ amelia$ R

R version 3.0.2 (2013-09-25) -- "Frisbee Sailing"
Copyright (C) 2013 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin10.8.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

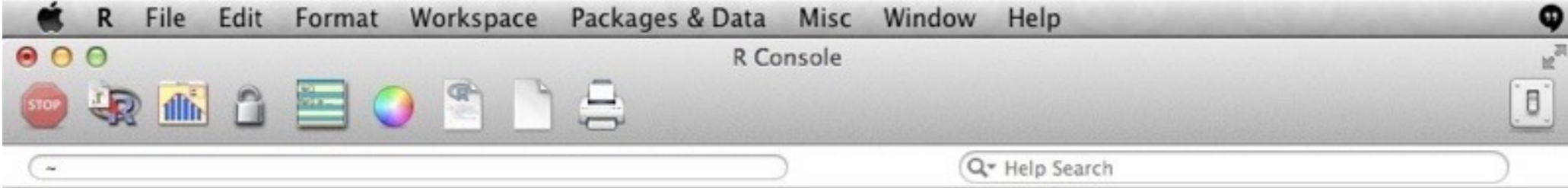
Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> █
```

Extensible—



```
R version 3.0.2 (2013-09-25) -- "Frisbee Sailing"
Copyright (C) 2013 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin10.8.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

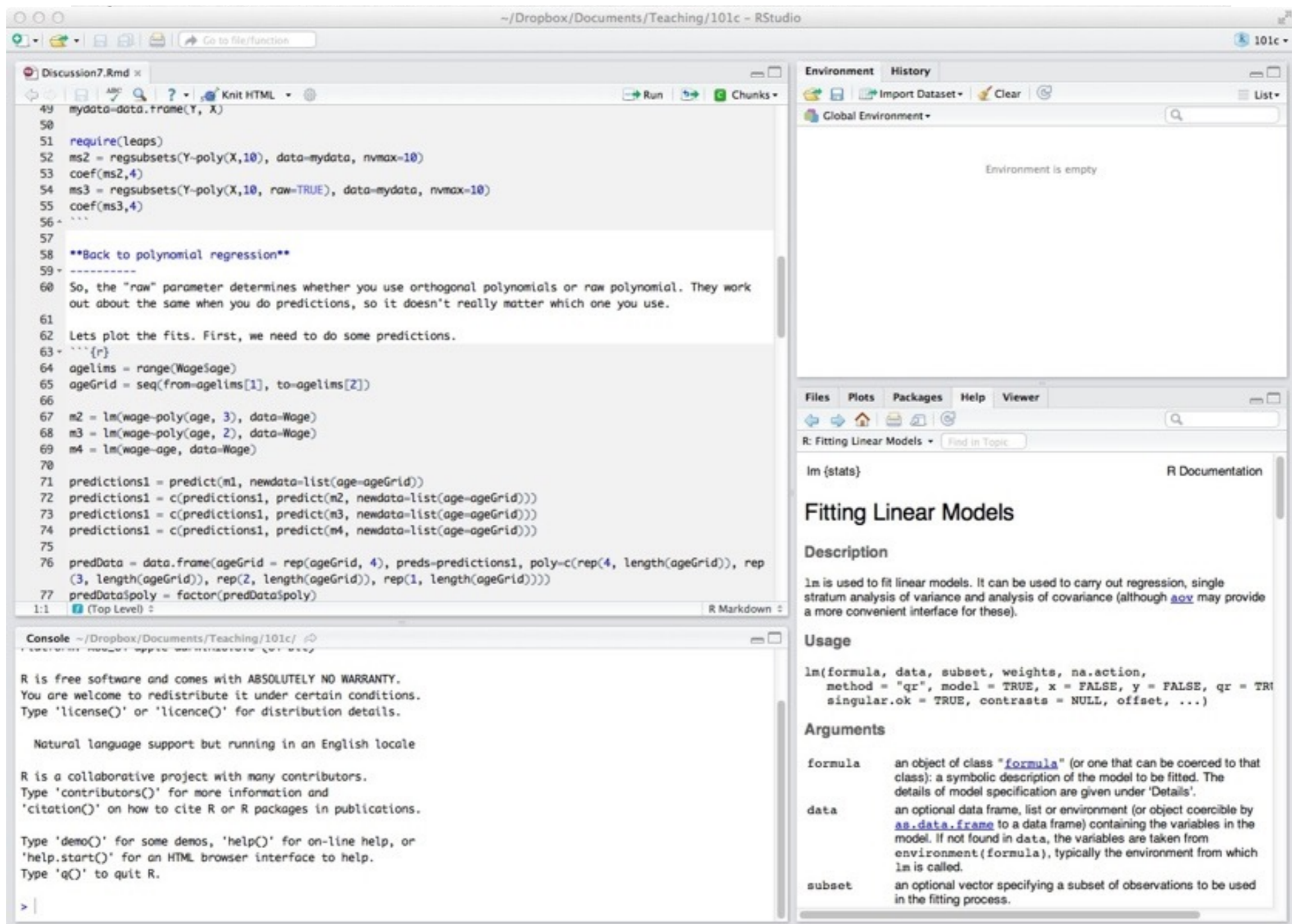
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.62 (6558) x86_64-apple-darwin10.8.0]

[History restored from /Users/amelia/.Rapp.history]

> |
```

Extensible



The screenshot displays the RStudio interface with three main panels:

- Source Editor (Top Left):** Contains R code for polynomial regression and a text block explaining the 'raw' parameter. The code includes:

```
49 mydata=data.frame(Y, X)
50
51 require(leaps)
52 ms2 = regsubsets(Y~poly(X,10), data=mydata, nvmax=10)
53 coef(ms2,4)
54 ms3 = regsubsets(Y~poly(X,10, raw=TRUE), data=mydata, nvmax=10)
55 coef(ms3,4)
56 ...
57
58 **Back to polynomial regression**
59 -----
60 So, the "raw" parameter determines whether you use orthogonal polynomials or raw polynomial. They work
61 out about the same when you do predictions, so it doesn't really matter which one you use.
62 Lets plot the fits. First, we need to do some predictions.
63 ```{r}
64 agelims = range(Wage$age)
65 ageGrid = seq(from=agelims[1], to=agelims[2])
66
67 m2 = lm(wage~poly(age, 3), data=Wage)
68 m3 = lm(wage~poly(age, 2), data=Wage)
69 m4 = lm(wage~age, data=Wage)
70
71 predictions1 = predict(m1, newdata=list(age=ageGrid))
72 predictions1 = c(predictions1, predict(m2, newdata=list(age=ageGrid)))
73 predictions1 = c(predictions1, predict(m3, newdata=list(age=ageGrid)))
74 predictions1 = c(predictions1, predict(m4, newdata=list(age=ageGrid)))
75
76 predData = data.frame(ageGrid = rep(ageGrid, 4), preds=predictions1, poly=c(rep(4, length(ageGrid)), rep
77 (3, length(ageGrid)), rep(2, length(ageGrid)), rep(1, length(ageGrid))))
78 predData$poly = factor(predData$poly)
```
- Environment (Top Right):** Shows 'Global Environment' with the message 'Environment is empty'.
- Console (Bottom Left):** Displays the R startup message:

```
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```
- Viewer (Bottom Right):** Shows the R documentation for the `lm` function, titled 'Fitting Linear Models'. It includes sections for 'Description', 'Usage', and 'Arguments'.

```
lm(formula, data, subset, weights, na.action,
   method = "qr", model = TRUE, x = FALSE, y = FALSE, qr = TRUE,
   singular.ok = TRUE, contrasts = NULL, offset, ...)
```

Interactivity— Fathom

The screenshot displays the Fathom software interface. At the top, there is a toolbar with various tool icons: Collection, Table, Graph, Summary, Estimate, Test, Model, Slider, Meter, and Text. Below the toolbar, a file named 'tgSpending.csv' is open, showing a table with the following data:

| | Attr1 | Attr2 | <new> |
|-----|-------|---------|-------|
| 429 | 429 | 50.1943 | |
| 430 | 430 | 41.7233 | |
| 431 | 431 | 203.553 | |
| 432 | 432 | 92.1924 | |
| 433 | 433 | 52.1701 | |
| 434 | 434 | 21.2527 | |
| 435 | 435 | 66.7804 | |
| 436 | 436 | 149.905 | |
| | | | |

The bottom of the window shows the text 'Fathom Dynamic Data Software'.

Interactivity— Fathom

The screenshot displays the Fathom software interface. At the top, there is a toolbar with various tool icons: Collection, Table, Graph, Summary, Estimate, Test, Model, Slider, Meter, and Text. Below the toolbar, a file named 'tgSpending.csv' is open, showing a table with the following data:

| | Attr1 | Attr2 | <new> |
|-----|-------|---------|-------|
| 429 | 429 | 50.1943 | |
| 430 | 430 | 41.7233 | |
| 431 | 431 | 203.553 | |
| 432 | 432 | 92.1924 | |
| 433 | 433 | 52.1701 | |
| 434 | 434 | 21.2527 | |
| 435 | 435 | 66.7804 | |
| 436 | 436 | 149.905 | |
| | | | |

The bottom of the window shows the text 'Fathom Dynamic Data Software'.

R/ggplot2

RStudio

lab-intro.Rmd * R data sets * R data sets *

ncbirths North Carolina births
oscars Oscar winners, 1929 to 2012
poker Poker winnings during 50 sessions
possum possum
prRace08 Election results for the 2008 U.S. Presidential race
president United States Presidential History
run10 Cherry Blossom 10 mile run data, 2009
run10Samp Cherry Blossom 10 mile run data, 2009
run10_09 Cherry Blossom 10 mile run data, 2009
satGPA SAT and GPA data
senateRace10 Election results for the 2010 U.S. Senate races
smoking UK Smoking Data
textbooks Textbook data for UCLA Bookstore and Amazon
tgSpending Thanksgiving spending, simulated based on Gallup poll.
tips Tip data
unempl Annual unemployment since 1890

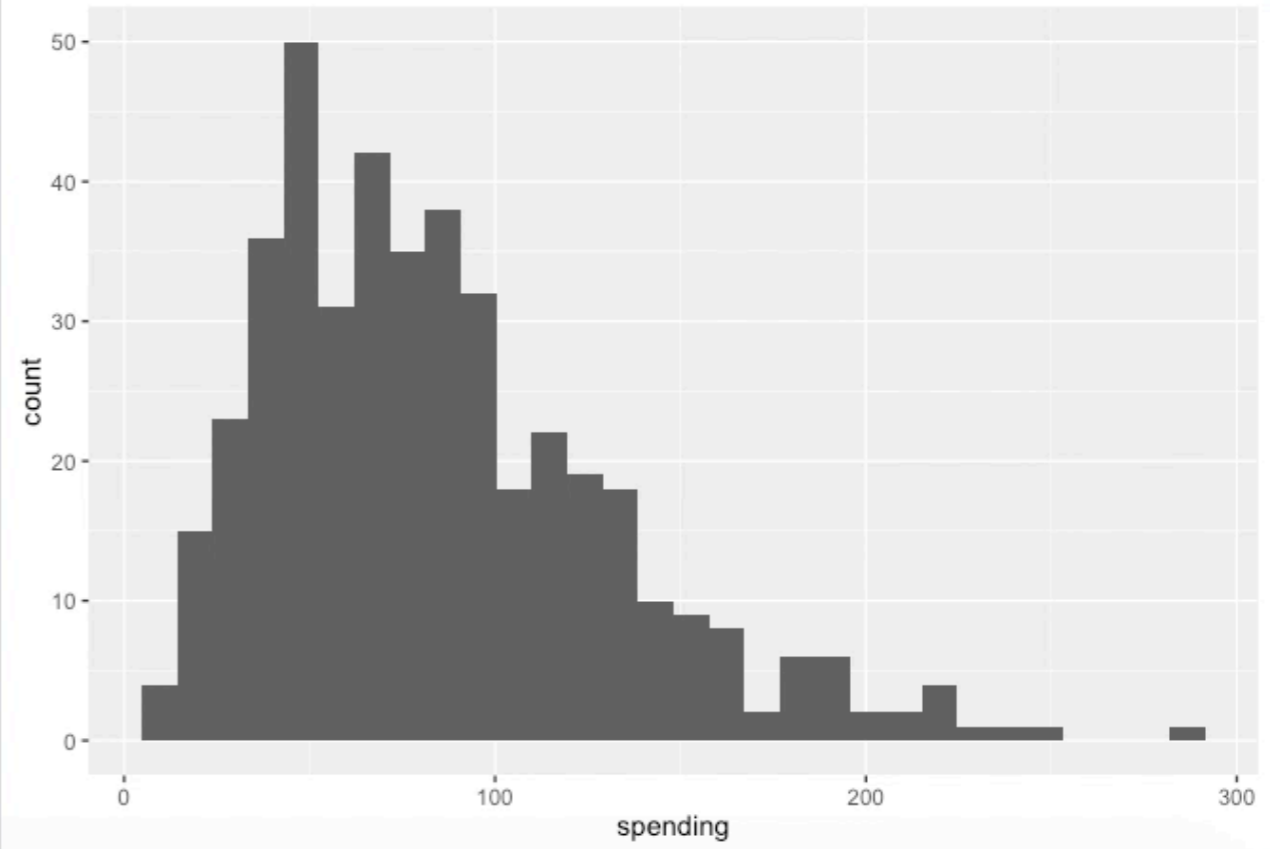
Use 'data(package = .packages(all.available = TRUE))'
to list the data sets in all *available* packages.

Environment History
Global Environment
Data
tgSpending 436 obs. of 1 variable

Files Plots Packages Help Viewer
Zoom Export Publish

Console ~/

```
Attaching package: 'openintro'  
  
The following object is masked from 'package:mosaic':  
  
  dotPlot  
  
The following object is masked from 'package:datasets':  
  
  cars  
  
> ggplot(tgSpending) + geom_histogram(aes(x=spending))  
'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.  
> |
```



R/ggplot2

RStudio

lab-intro.Rmd * R data sets * R data sets *

```
ncbirths      North Carolina births
oscars        Oscar winners, 1929 to 2012
poker         Poker winnings during 50 sessions
possum        possum
prRace08      Election results for the 2008 U.S.
               Presidential race
president     United States Presidential History
run10         Cherry Blossom 10 mile run data, 2009
run10Samp     Cherry Blossom 10 mile run data, 2009
run10_09      Cherry Blossom 10 mile run data, 2009
satGPA        SAT and GPA data
senateRace10  Election results for the 2010 U.S. Senate
               races
smoking       UK Smoking Data
textbooks     Textbook data for UCLA Bookstore and Amazon
tgSpending    Thanksgiving spending, simulated based on
               Gallup poll.
tips          Tip data
unempl        Annual unemployment since 1890
```

Use 'data(package = .packages(all.available = TRUE))'
to list the data sets in all *available* packages.

Environment History

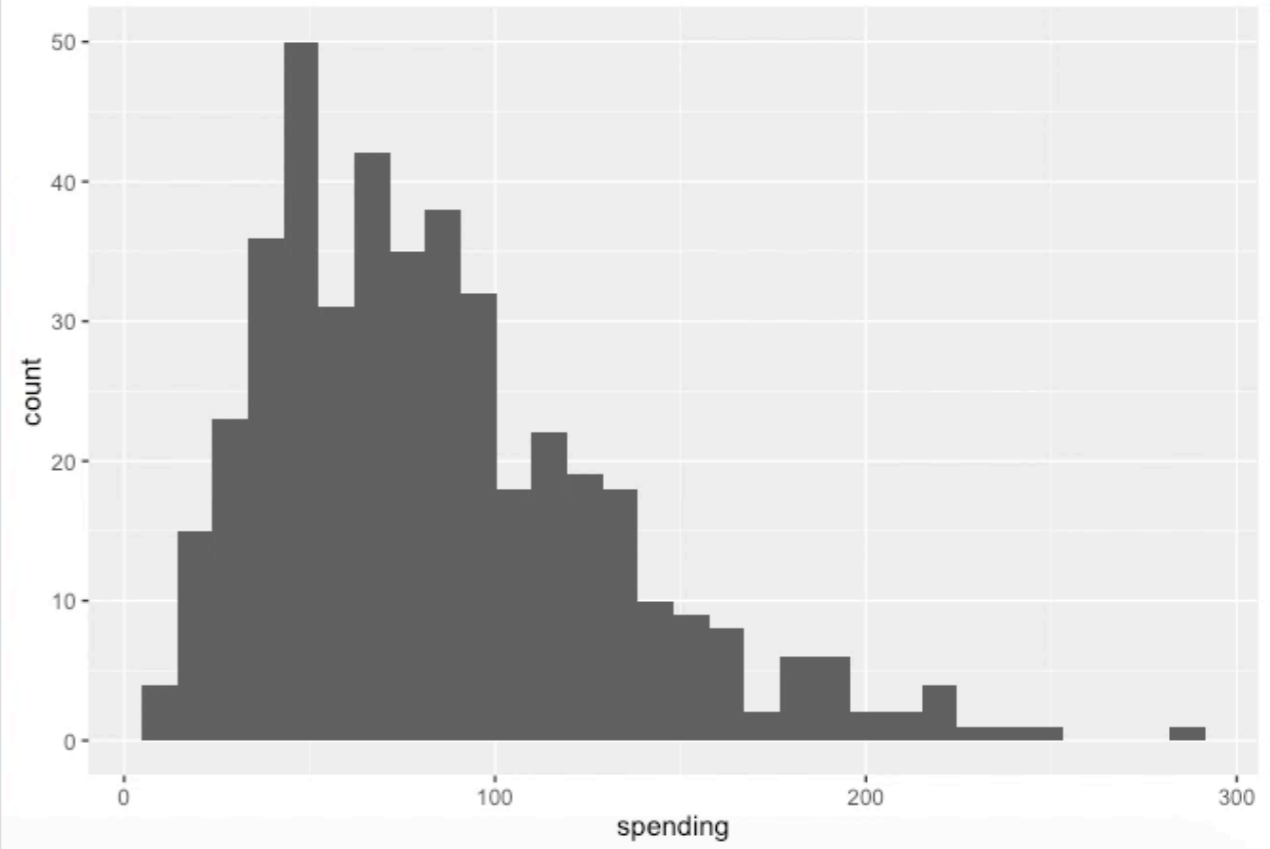
Global Environment

Data

tgSpending 436 obs. of 1 variable

Files Plots Packages Help Viewer

Zoom Export Publish



count

spending

```
Console ~/
```

```
Attaching package: 'openintro'
```

```
The following object is masked from 'package:mosaic':
```

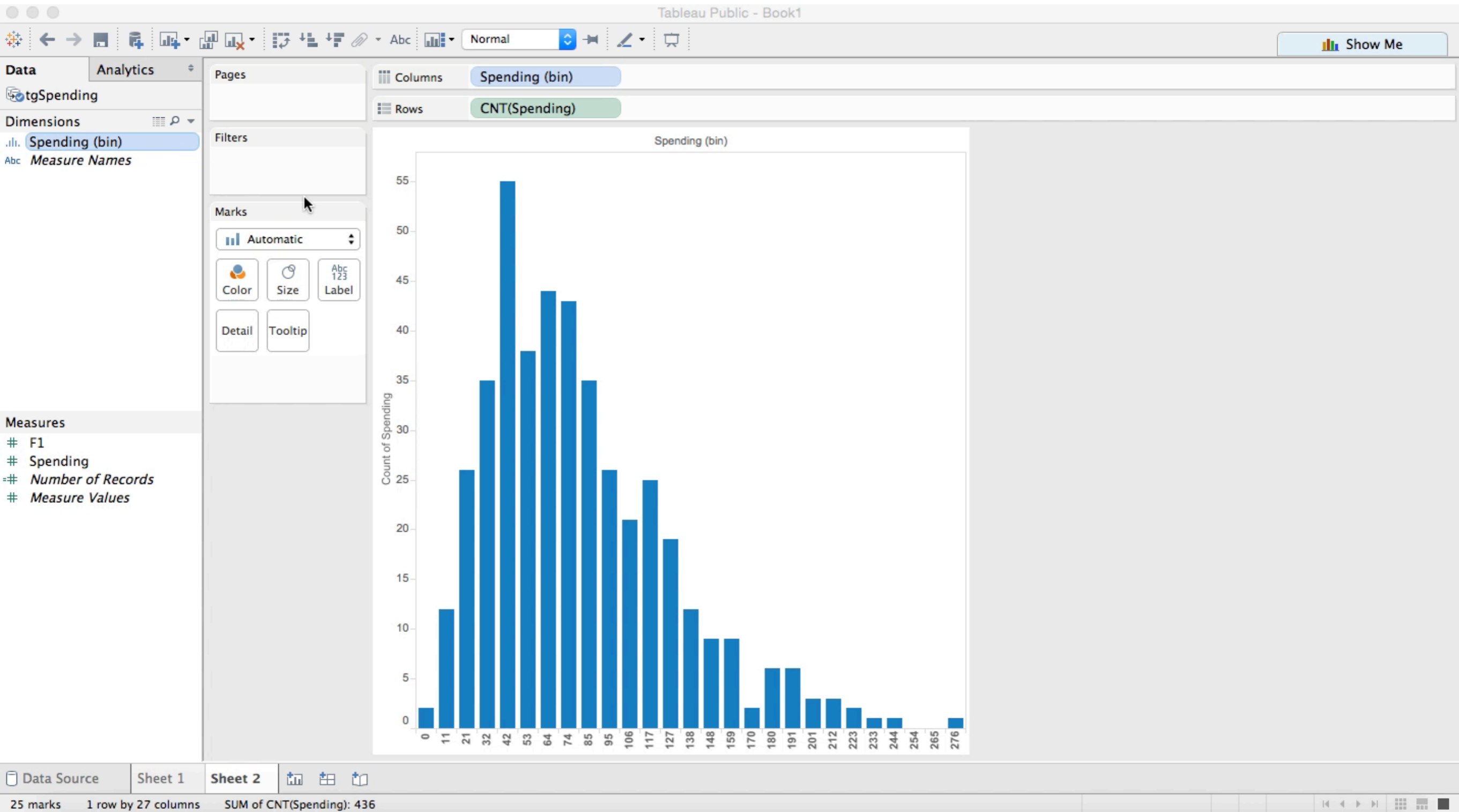
```
  dotPlot
```

```
The following object is masked from 'package:datasets':
```

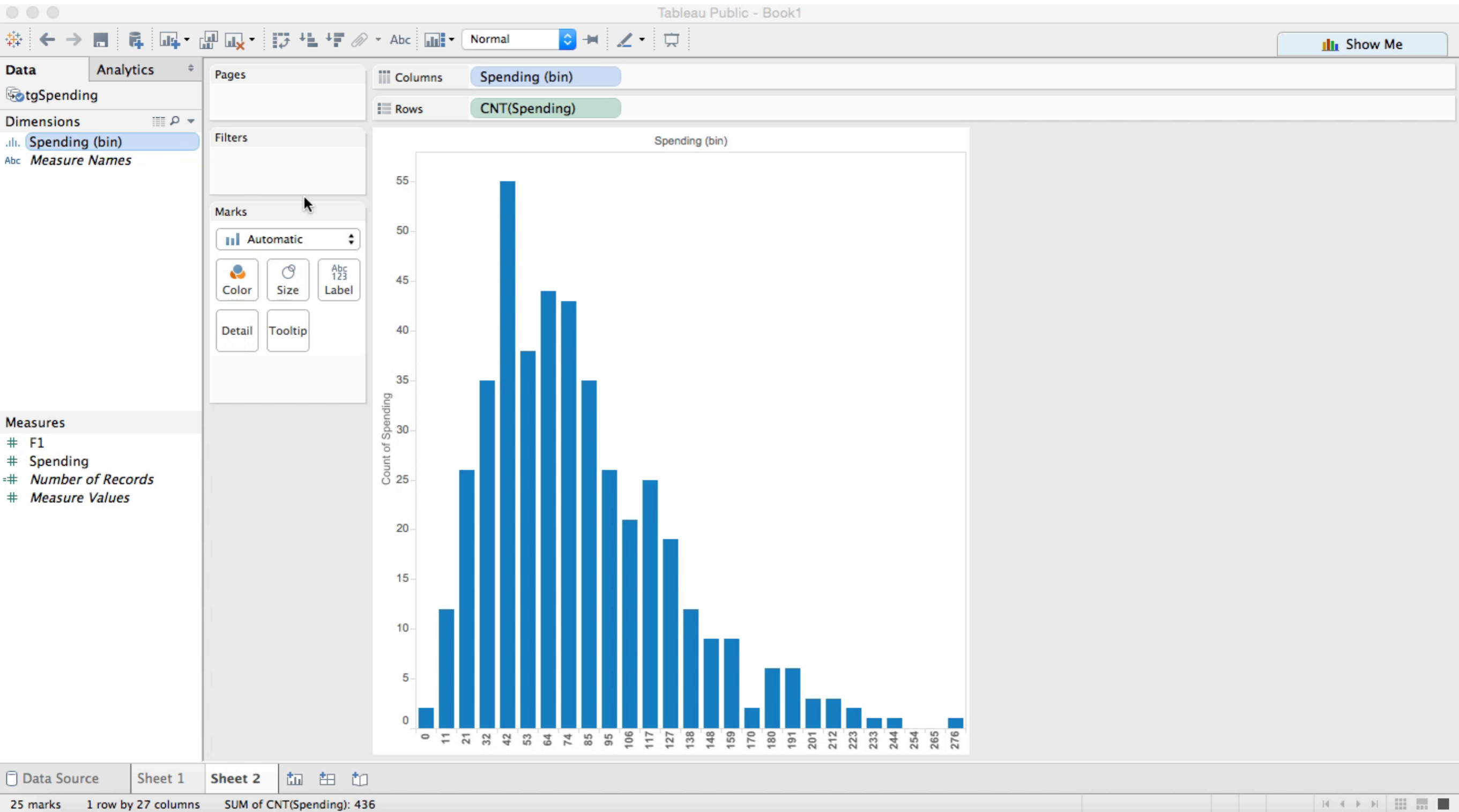
```
  cars
```

```
> ggplot(tgSpending) + geom_histogram(aes(x=spending))
  `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
> |
```

Tableau



Tableau



Gather your data

A histogram is based on a collection of data about a numeric variable. Our first step is to gather some values for that variable. The initial dataset we will consider consists of fuel consumption (in miles per gallon) from a sample of car models available in 1974 (yes, rather out of date). We can visualize the dataset as a pool of items, with each item identified by its value—which in theory lets us "see" all the items, but makes it hard to get the gestalt of the variable. What are some common values? Is there a lot of variation?

Sort into an ordered list

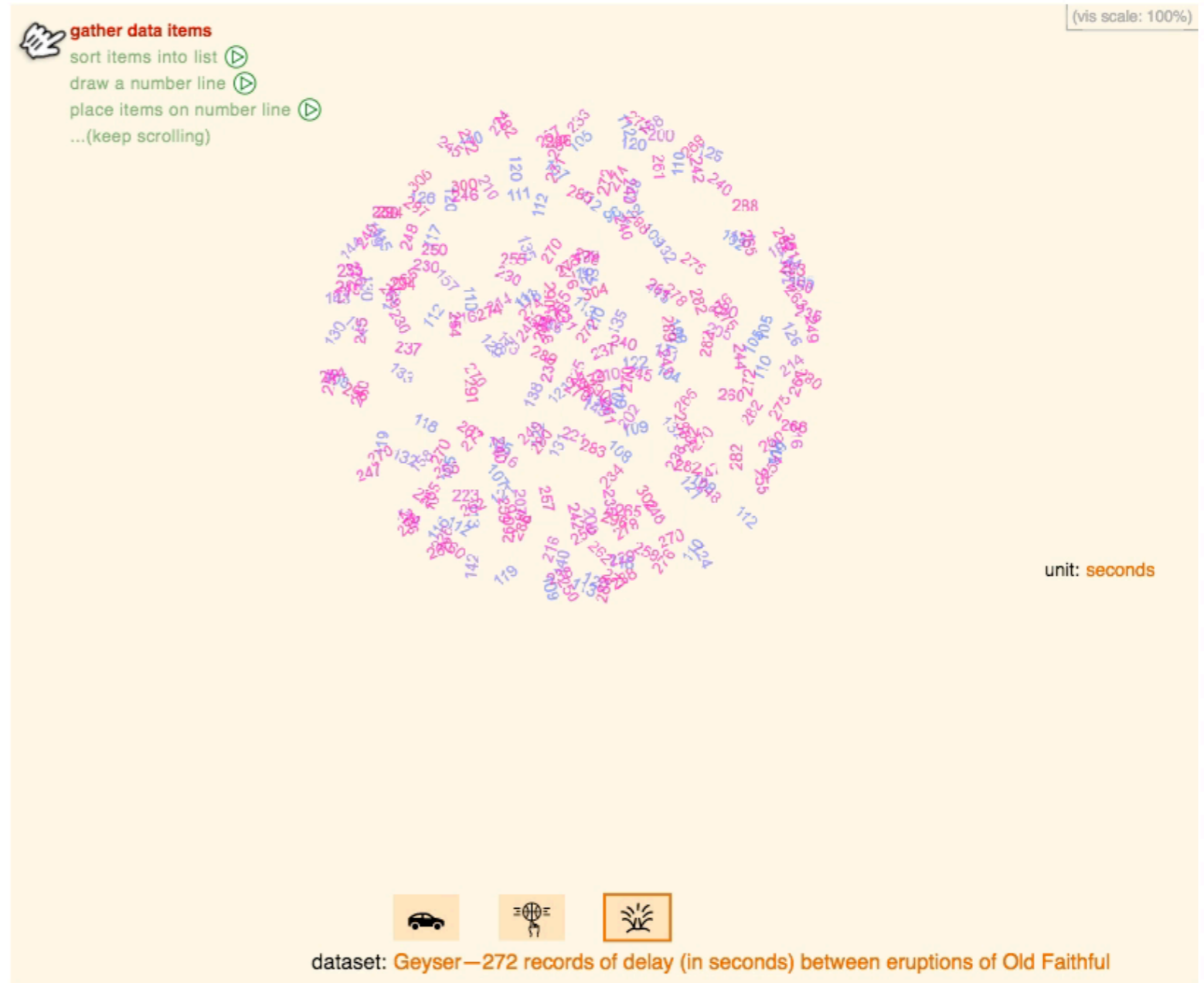
A useful first step towards describing the variable's distribution is to sort the items into a list. Now we can see the maximum value and the minimum value. Beyond that, it is hard to say much about the center, shape, and spread of the distribution. Part of the problem is that the list is completely filled; the space between any two items is the same, no matter how dissimilar their values may be. We need a way to see how the items relate to each other. Are they clustered around a few specific values? Is there one lonely item, with a value far removed from all the others?

Draw the number line

A common convention is to use a number line, on which higher values are displayed to the right and smaller (or negative) values to the left. We can draw a line representing all possible numbers between the minimum and maximum data values.

Add data to the number line

Now, we map each item to a dot at the appropriate point along the number line. In our visualization we draw the path followed by each item on its way from the list to the line, helping to reveal how adjacent list items end up close or far apart on the number line



Gather your data

A histogram is based on a collection of data about a numeric variable. Our first step is to gather some values for that variable. The initial dataset we will consider consists of fuel consumption (in miles per gallon) from a sample of car models available in 1974 (yes, rather out of date). We can visualize the dataset as a pool of items, with each item identified by its value—which in theory lets us "see" all the items, but makes it hard to get the gestalt of the variable. What are some common values? Is there a lot of variation?

Sort into an ordered list

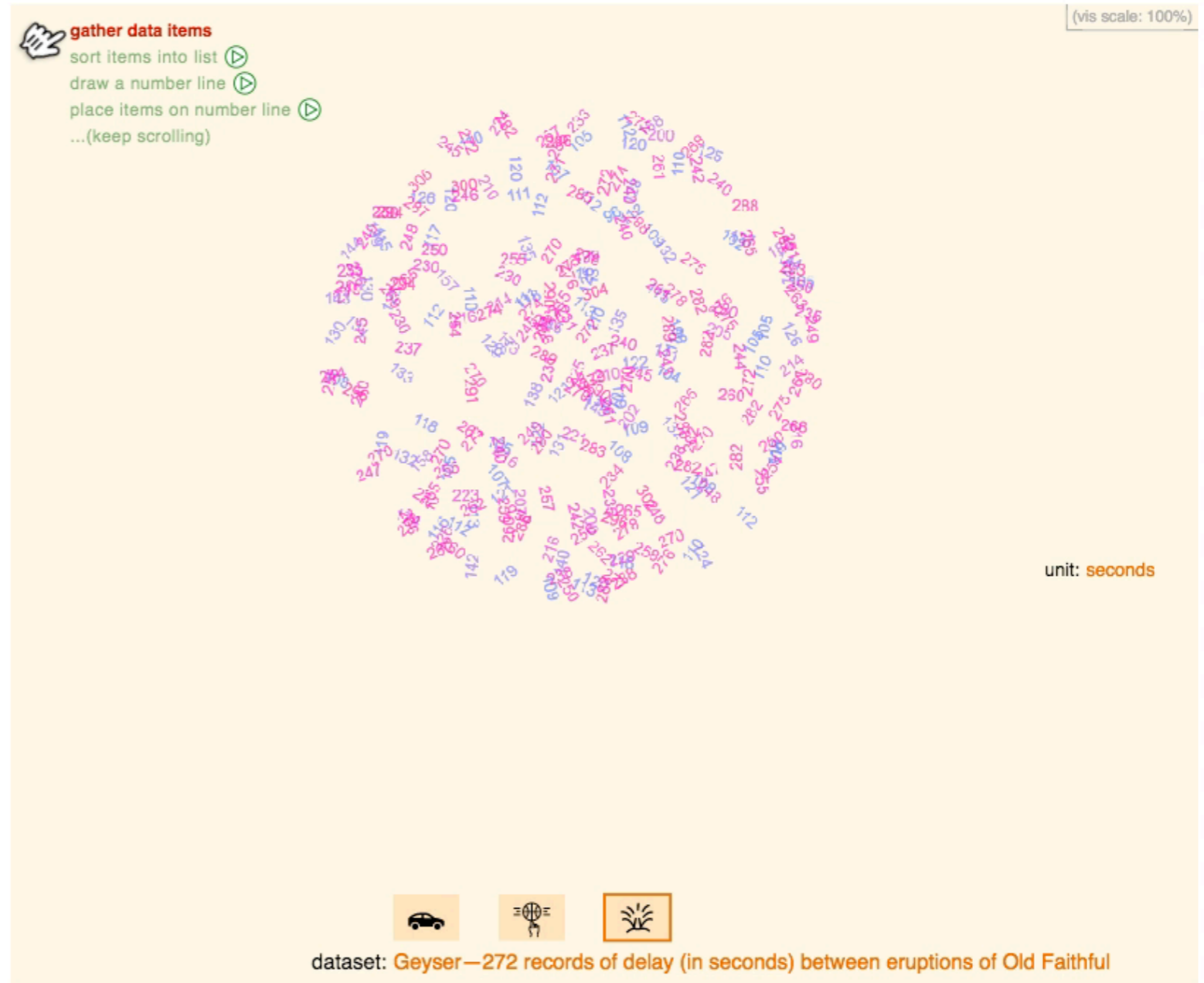
A useful first step towards describing the variable's distribution is to sort the items into a list. Now we can see the maximum value and the minimum value. Beyond that, it is hard to say much about the center, shape, and spread of the distribution. Part of the problem is that the list is completely filled; the space between any two items is the same, no matter how dissimilar their values may be. We need a way to see how the items relate to each other. Are they clustered around a few specific values? Is there one lonely item, with a value far removed from all the others?







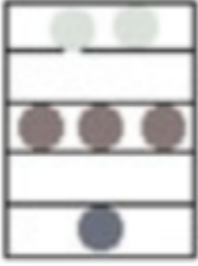
Draw the number line

A common convention is to use a number line, on which higher values are displayed to the right and smaller (or negative) values to the left. We can draw a line representing all possible numbers between the minimum and maximum data values.

Add data to the number line

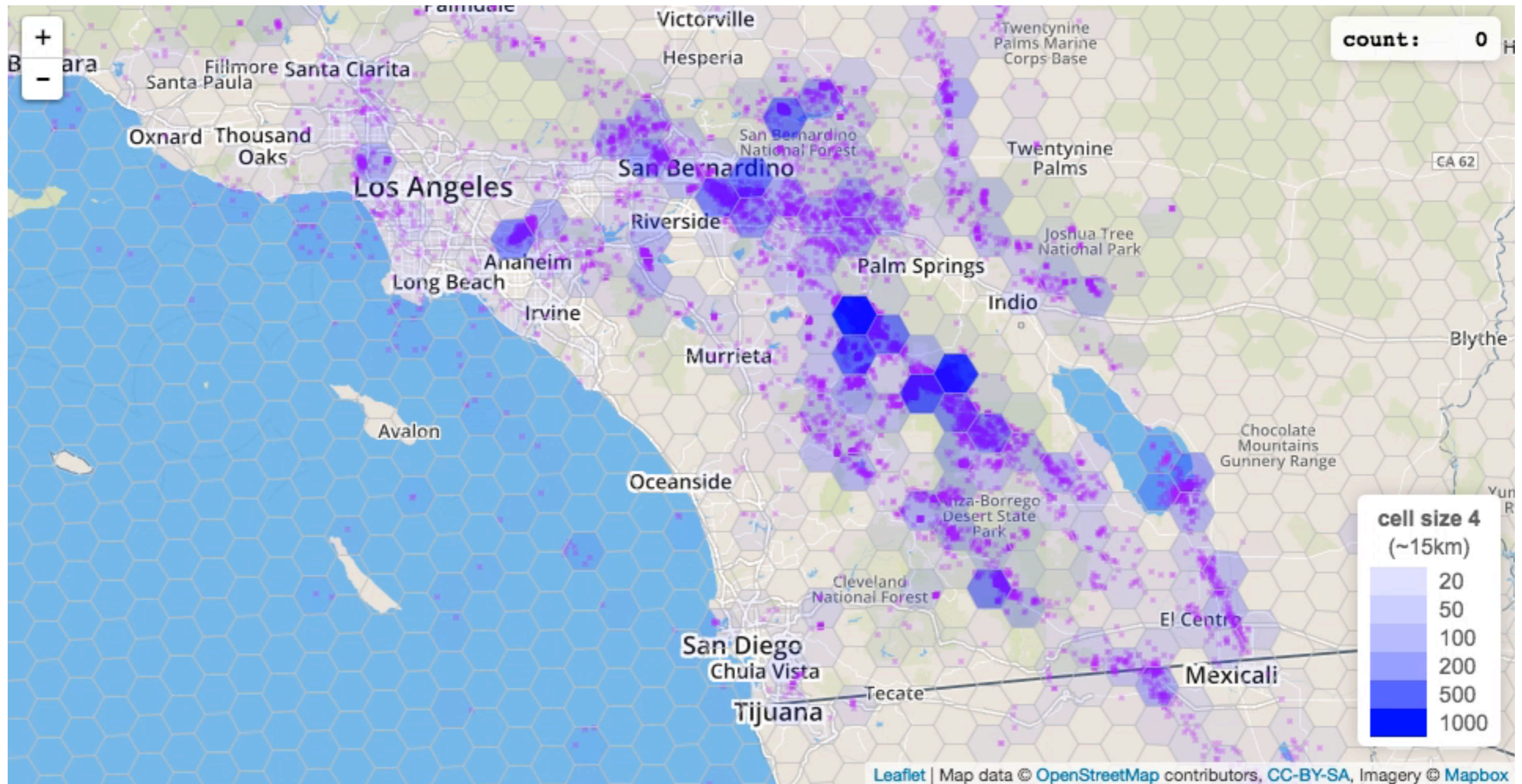
Now, we map each item to a dot at the appropriate point along the number line. In our visualization we draw the path followed by each item on its way from the list to the line, helping to reveal how adjacent list items end up close or far apart on the number line



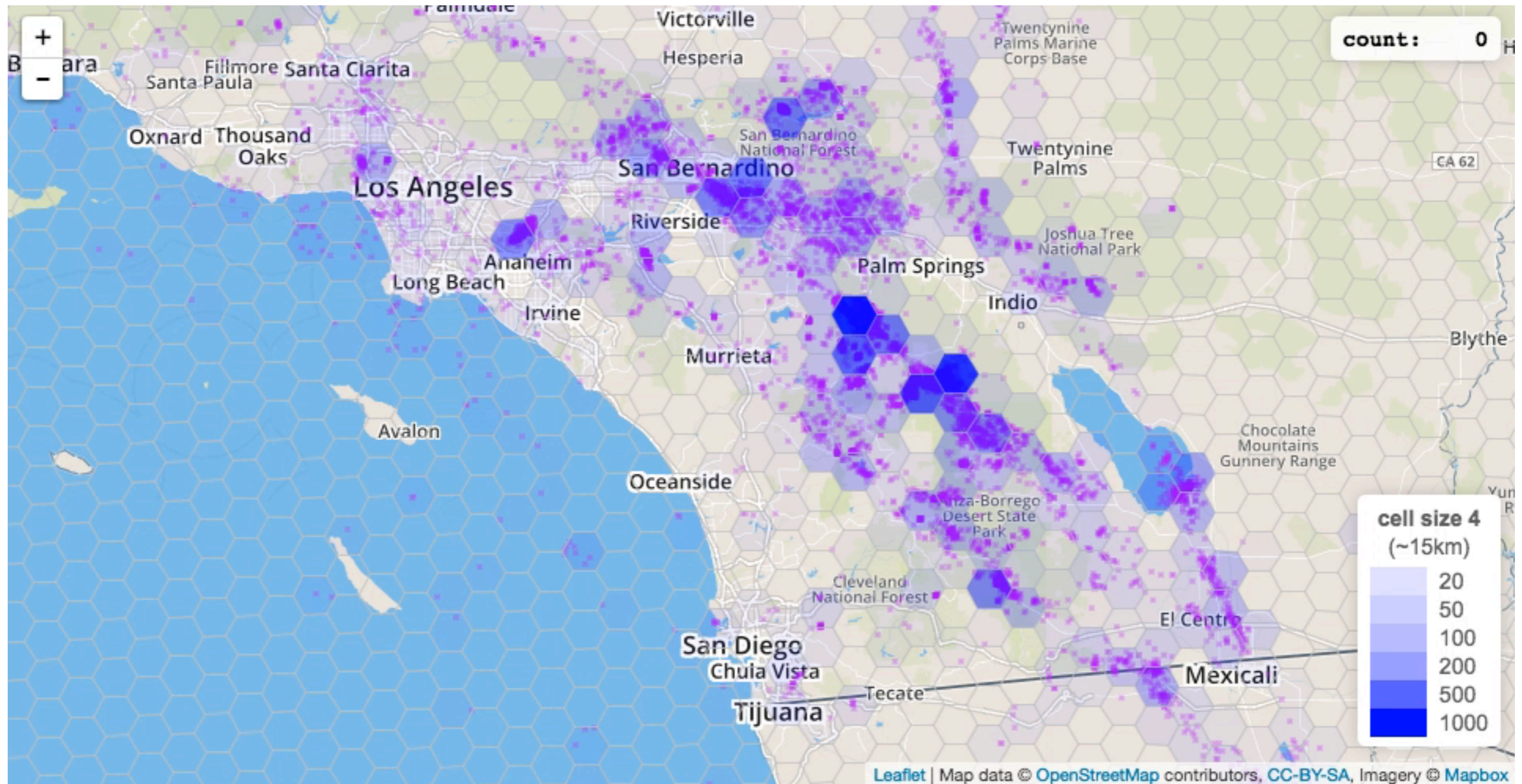
| View of Data | Perceptual Unit | Data Structure | Student Observation |
|--------------|---|---|-----------------------------|
| Pointer | ? |  | We said our favorite colors |
| Case Value |  |  | Juan likes red |
| Classifier |  |  | Three like red |
| Aggregate |  |  | Half like red |

Konold, C. et al. "Data seen through different lenses." *Educational Studies in Mathematics*, 2014.

Spatial aggregation toy



Spatial aggregation toy



Interactivity in published work

Choose a Ranking (choose a weighting or make your own)

IEEE Spectrum

Trending

Jobs

Open

Custom

Edit Ranking

Add a Comparison

Language Types (click to hide)



Web



Mobile



Enterprise



Embedded

Language Rank

Types

Spectrum Ranking

1. Java



100.0

2. C



99.2

3. C++



95.5

4. Python



93.4

5. C#



92.2

6. PHP



84.6

7. Javascript



84.3

8. Ruby



78.6

9. R



74.0

10. MATLAB



72.6

Show Extended Ranking

Interactivity in published work

Choose a Ranking (choose a weighting or make your own)

IEEE Spectrum

Trending

Jobs


Open


Custom


Edit Ranking


Add a Comparison

Language Types (click to hide)

 Web

 Mobile

 Enterprise

 Embedded

The ranking is calculated using 12 weighted data sources. Click a data source to toggle its inclusion in the ranking and drag its slider to reweight it.

Google (search)



Google (trends)



Github (active)



Github (created)



Stack Overflow (?s)



Stack Overflow (views)



Reddit



Hacker News



Career Builder



Dice



Topsy



IEEE Xplore



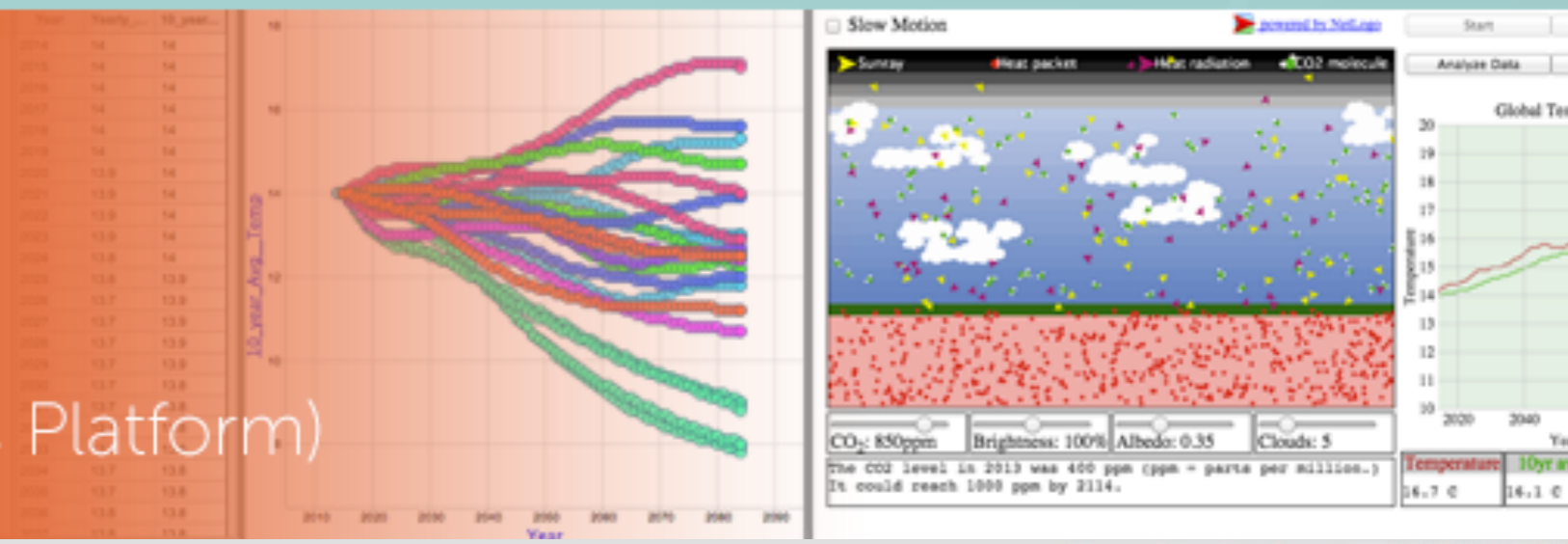
Cancel

Save as Custom

CODAP

(Common Online Data Analysis Platform)

SHARE PRINT



Untitled Document

User: guest Version 1.1 (0283 IS)

File Game Table Graph Map Slider Calc Text Options

Clear Data... Login

Next Gen MW

Share About

Achieved Terminal Velocity

Distance vs. Time

| Time (s) | Distance (m) |
|----------|--------------|
| 0.0 | 8.0 |
| 0.50 | 7.5 |
| 1.0 | 7.0 |
| 1.5 | 6.5 |
| 2.0 | 6.0 |
| 2.5 | 5.5 |

Velocity vs. Time

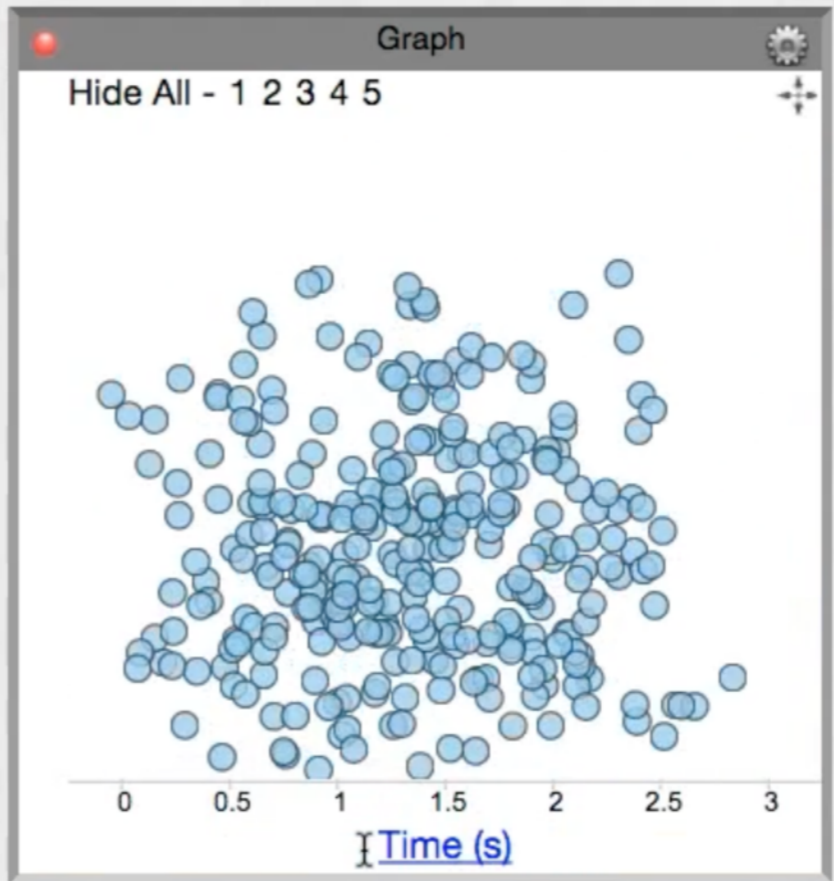
| Time (s) | Velocity (m/s) |
|----------|----------------|
| 0.0 | 0.0 |
| 0.50 | 1.5 |
| 1.0 | 1.75 |
| 1.5 | 1.78 |
| 2.0 | 1.78 |
| 2.5 | 1.78 |

Mass of jumper (g) 200 Parachute size (cm²) 900

12.5 s Start Stop Analyze Data New Run

5 runs/305 measurements Case Table

| Row | mass_o... | parach... | termina... | Time (s) | Distanc... | Velocit... |
|-----|-----------|-----------|------------|----------|------------|------------|
| 1 | 200 | 700 | 2.29 | 2.33 | 4.58 | 1.78 |
| 2 | 200 | 800 | 2 | 2.37 | 4.51 | 1.78 |
| 3 | 200 | 1000 | 1.6 | 2.41 | 4.44 | 1.78 |
| 4 | 200 | 1100 | 1.45 | 2.45 | 4.36 | 1.78 |
| 5 | 200 | 900 | 1.78 | 2.5 | 4.29 | 1.78 |



Curriculum



mobilize

Our data. Our lives.





Introduction to Data Science

Robert Gould

Suyen Moncada-Machado

Terri Anna Johnson

James Molyneux

- Year-long course
- Validates Algebra II requirement
- “Data science”
- Taught in R within RStudio server
- Participatory sensing
- Content includes:
 - Exploratory data analysis
 - Randomization, simulation, bootstrapping
 - Simple linear regression, multiple regression
 - Decision trees, clustering, k-means



Introduction to Data Science

Robert Gould
Suyen Moncada-Machado
Terri Anna Johnson
James Molyneux

RStudio Lab Codes and Functions

Contents

| | |
|--|----|
| Loading, saving and viewing data | 1 |
| Scraping web data | 3 |
| Data manipulation functions | 3 |
| Numerical summaries and frequency tables | 5 |
| Plotting functions | 7 |
| Maps | 11 |
| Statistical modeling | 12 |
| Sampling and permutation functions | 13 |
| Probability functions | 13 |
| Symbols and operators | 13 |
| Creating functions | 13 |

Loading, saving and viewing data

`data()`: Loads and displays a pre-loaded data file from RStudio.

Example:

```
data(cdc)
```

`read.csv()`: Imports data from a `.csv` formatted file into R.

Example:

```
read.csv("Time Use.csv")
```

`View()`: Displays the data as a preview in a new tab.

Example:

```
View(cdc)
```

`head()`: Prints the first 6 values or rows of data in the console.

Examples:

```
# Observations of a dataset
```

```
head(cdc)
```

```
# Observations of a variable
```

```
head(-gender, data = cdc)
```

`tail()`: Prints the last 6 values or rows of data in the console.

Examples:

```
# Observations of a dataset
```

```
tail(cdc)
```



DataCamp

OpenIntro 



{swirl}

Learn R, in R.



Introduction to Data Science

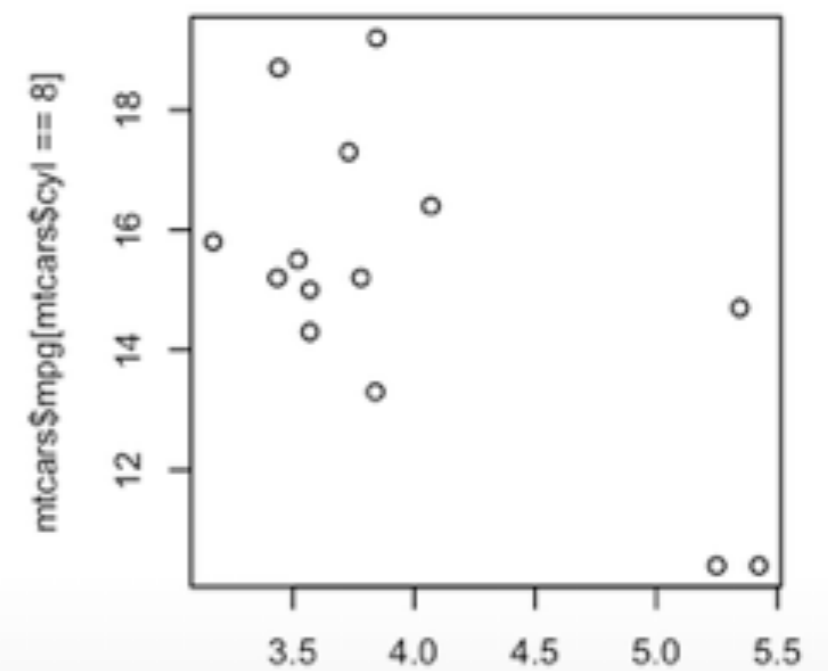
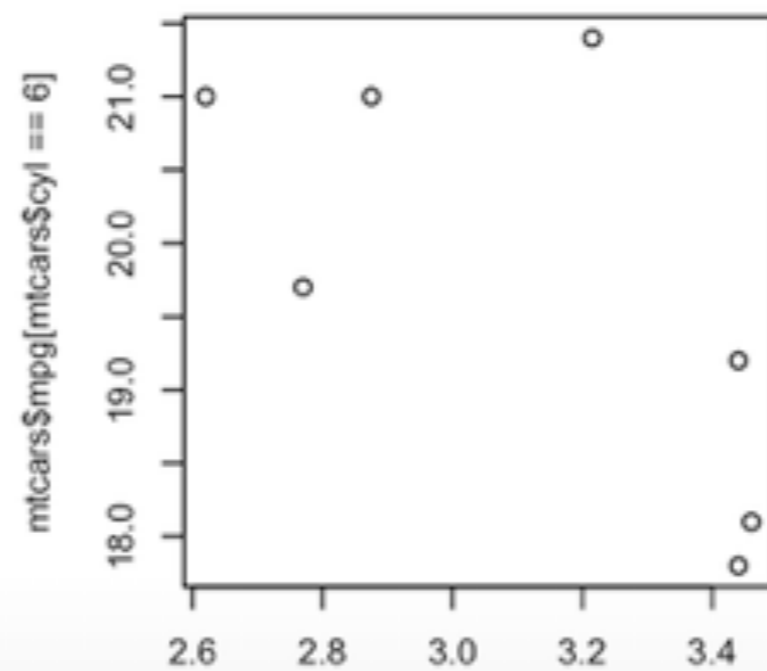
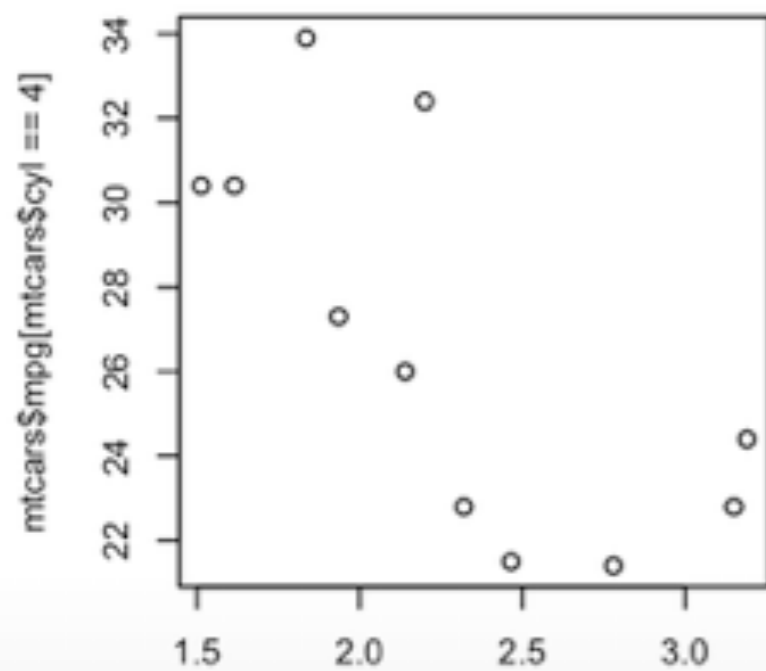
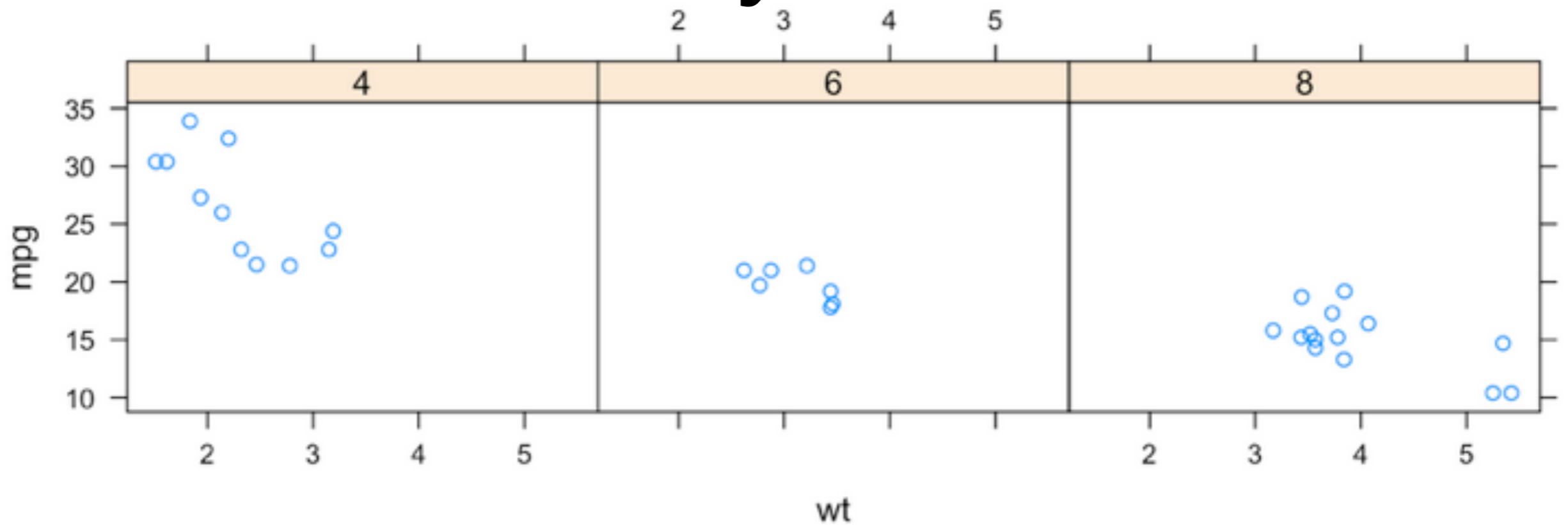
R Syntax

```
xyplot(mpg ~ wt | as.factor(cyl), data = mtcars)
```

vs.

```
par(mfrow = c(1,3))  
plot(mtcars$wt[mtcars$cyl == 4], mtcars$mpg[mtcars$cyl == 4])  
plot(mtcars$wt[mtcars$cyl == 6], mtcars$mpg[mtcars$cyl == 6])  
plot(mtcars$wt[mtcars$cyl == 8], mtcars$mpg[mtcars$cyl == 8])
```

R Syntax



`mtcars$wt[mtcars$cyl == 4]`

`mtcars$wt[mtcars$cyl == 6]`

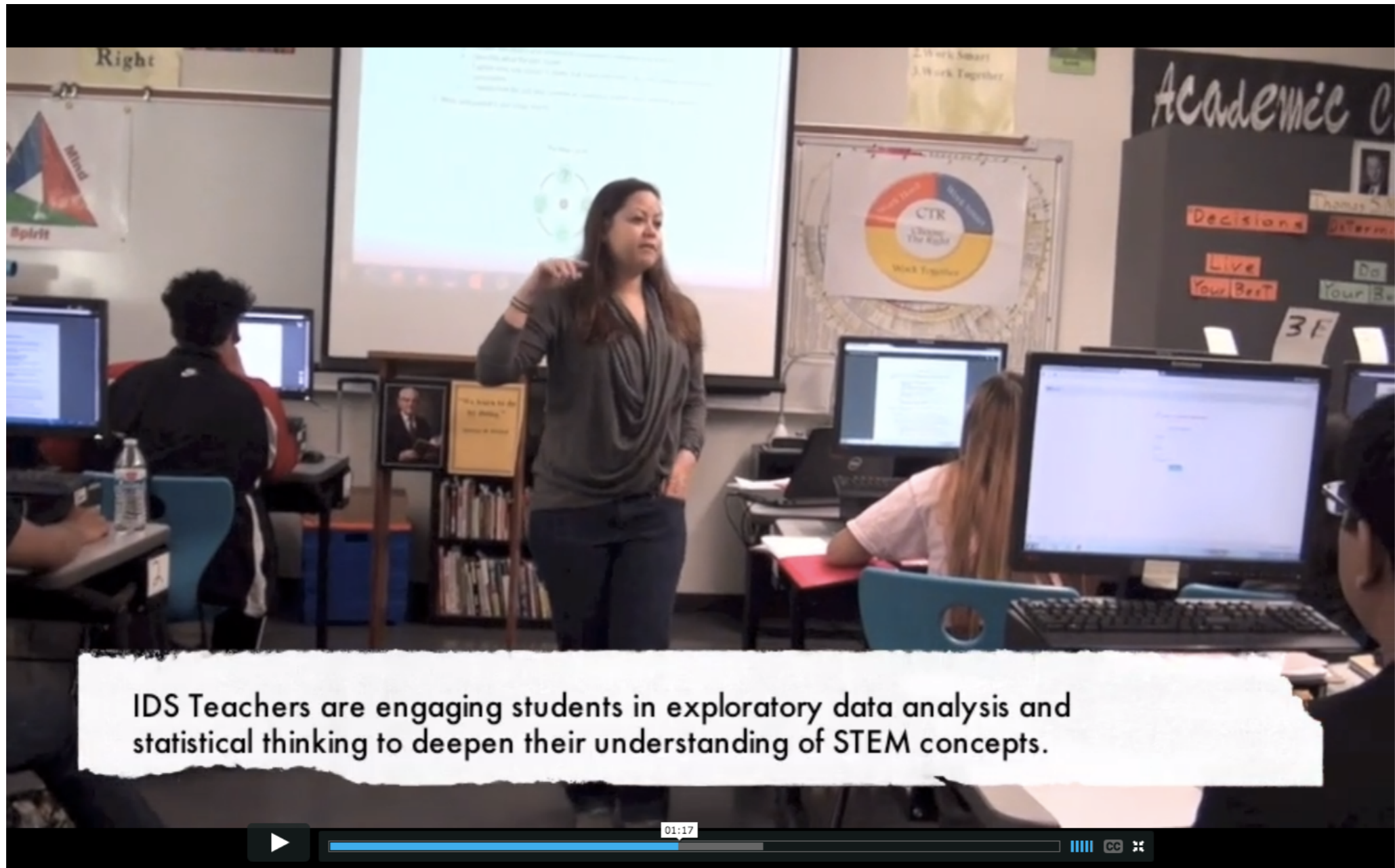
`mtcars$wt[mtcars$cyl == 8]`

Formula-based syntax

- `lattice` graphics
- `mosaic` package for statistics
- `mobilizr` additional functions

Want to check out the labs?

```
library(devtools)
install_github("mobilizingcs/mobilizr")
library(mobilizr)
load_labs()
```



IDS Teachers are engaging students in exploratory data analysis and statistical thinking to deepen their understanding of STEM concepts.

Data science at Smith



Introductory course

Introductory Statistics with
Randomization and
Simulation.

David M Diez, Christopher
D Barr, Mine Çetinkaya-
Rundel

www.openintro.org

Introductory Statistics with Randomization and Simulation

First Edition

OpenIntro[®]

David M Diez
Christopher D Barr
Mine Çetinkaya-Rundel

R syntax comparison

Cheat Sheet



Syntax

Syntax is the set of rules that govern what code works and doesn't work. Most programming languages offer one standardized syntax, but R has many.

Most people use some combination of all the syntaxes available to them.

1. Dollar sign syntax uses the dollar sign to locate a variable within a dataset. It is expected by most **base** R functions.
2. Formula syntax uses the **data=** argument at the end of a list of function arguments. The formula syntax is used by modeling functions like **lm()**, **lattice** graphics like **xyplot()**, and **mosaic** summary statistics like **mean()**.
3. Tidyverse syntax uses data as the first argument to function calls. It is used by the packages **dplyr** and **tidyr**, among others. The associated graphics library is **ggplot2**.

Dollar sign syntax

```
goal(data$x, data$y)
```

Summary statistics:

one continuous variable:
`mean(mtcars$mpg)`

one categorical variable:
`table(mtcars$cyl)`

two categorical variables:
`table(mtcars$cyl, mtcars$am)`

one continuous, one categorical:
`mean(mtcars$mpg[mtcars$cyl==4])`
`mean(mtcars$mpg[mtcars$cyl==6])`
`mean(mtcars$mpg[mtcars$cyl==8])`

Plotting:

one continuous variable:
`hist(mtcars$disp)`

`boxplot(mtcars$disp)`

one categorical variable:
`barplot(table(mtcars$cyl))`

two continuous variables:
`plot(mtcars$disp, mtcars$mpg)`

two categorical variables:
`mosaicplot(table(mtcars$am, mtcars$cyl))`

one continuous, one categorical:
`histogram(mtcars$disp[mtcars$cyl==4])`
`histogram(mtcars$disp[mtcars$cyl==6])`
`histogram(mtcars$disp[mtcars$cyl==8])`

`boxplot(mtcars$disp[mtcars$cyl==4])`
`boxplot(mtcars$disp[mtcars$cyl==6])`
`boxplot(mtcars$disp[mtcars$cyl==8])`

Wrangling:

subsetting:
`mtcars[mtcars$mpg>30,]`

making a new variable:
`mtcars$efficient[mtcars$mpg>30] <- TRUE`
`mtcars$efficient[mtcars$mpg<30] <- FALSE`

Formula syntax

```
goal(y~x|z, data=data, group=w)
```

Summary statistics:

one continuous variable:
`mosaic::mean(~mpg, data=mtcars)`

one categorical variable:
`mosaic::tally(~cyl, data=mtcars)`

two categorical variables:
`mosaic::tally(cyl~am, data=mtcars)`

one continuous, one categorical:
`mosaic::mean(mpg~cyl, data=mtcars)`

tilde

Plotting:

one continuous variable:
`lattice::histogram(~disp, data=mtcars)`

`lattice::bwplot(~disp, data=mtcars)`

one categorical variable:
`mosaic::bargraph(~cyl, data=mtcars)`

two continuous variables:
`lattice::xyplot(mpg~disp, data=mtcars)`

two categorical variables:
`mosaic::bargraph(~am, data=mtcars, group=cyl)`

one continuous, one categorical:
`lattice::histogram(~disp|cyl, data=mtcars)`

`lattice::bwplot(cyl~disp, data=mtcars)`

Tidyverse syntax

```
data %>% goal(x)
```

Summary statistics:

one continuous variable:
`mtcars %>% dplyr::summarize(mean(mpg))`

one categorical variable:
`mtcars %>% dplyr::group_by(cyl) %>% dplyr::summarize(n())`

the pipe

two categorical variables:
`mtcars %>% dplyr::group_by(cyl, am) %>% dplyr::summarize(n())`

one continuous, one categorical:
`mtcars %>% dplyr::group_by(cyl) %>% dplyr::summarize(mean(mpg))`

Plotting:

one continuous variable:
`ggplot2::qplot(x=mpg, data=mtcars, geom="histogram")`

`ggplot2::qplot(y=disp, x=1, data=mtcars, geom="boxplot")`

one categorical variable:
`ggplot2::qplot(x=cyl, data=mtcars, geom="bar")`

two continuous variables:
`ggplot2::qplot(x=disp, y=mpg, data=mtcars, geom="point")`

two categorical variables:
`ggplot2::qplot(x=factor(cyl), data=mtcars, geom="bar") + facet_grid(~am)`

one continuous, one categorical:
`ggplot2::qplot(y=disp, x=factor(cyl), data=mtcars, geom="boxplot")`

`ggplot2::qplot(x=disp, data=mtcars, geom="histogram") + facet_grid(~cyl)`

Wrangling:

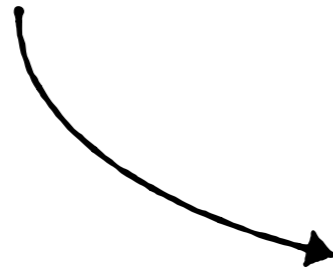
subsetting:
`mtcars %>% dplyr::filter(mpg>30)`

making a new variable:
`mtcars <- mtcars %>% dplyr::mutate(efficient = if_else(mpg>30, TRUE, FALSE))`

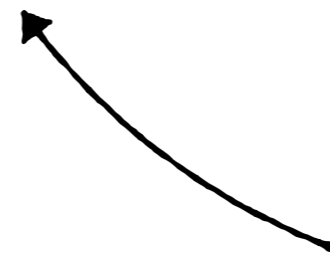
- Graphical literacy skills should be taught in the context of science and social science.
- Translating between representations may be beneficial
- Explicitly focus on the links between visual features and meaning.
- Make graph reading metacognitive

Shah, P. and Hoeffner, J. "Review of Graph Comprehension Research: Implications for Instruction." *Educational Psychology Review*, 2002.

New faculty



- SDS 136: Communicating with Data
- SDS 192: Introduction to Data Science
- SDS 235: Visual Analytics
- SDS 236: Data Journalism
- SDS 293: Machine Learning



New courses

ASA DataFest™

- Started at UCLA
- Now a national and international event
- Groups of undergraduate students compete to find insight in data
- Past data sponsors: LAPD, Kiva.com, eHarmony, Travelocity



ASA DataFest™



Data Science as a Science

Practical Data Science for Stats

<http://bit.ly/practical-data-sci>





Figure 3. Prestructural representation from Group 5. (The caption at the top reads "This is a graph of a group of kids who showed their ages".)

Chick, H. and Watson, J. "Data representation and interpretation by primary school students working in groups." *Mathematics Education Research Journal*, 2001.



Thank you

Amelia McNamara ([@AmeliaMN](#))