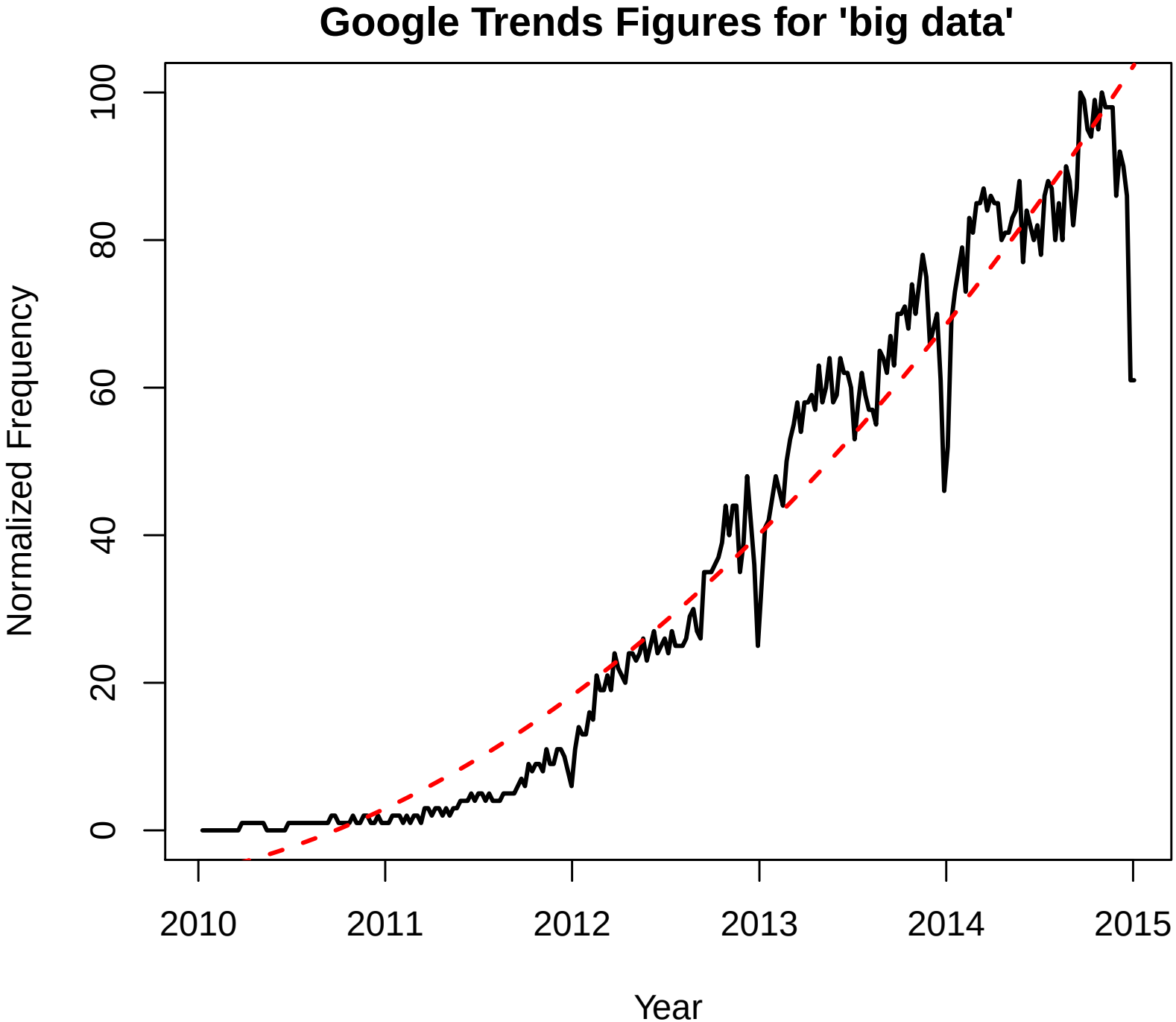
'Big Data' in the Social Sciences

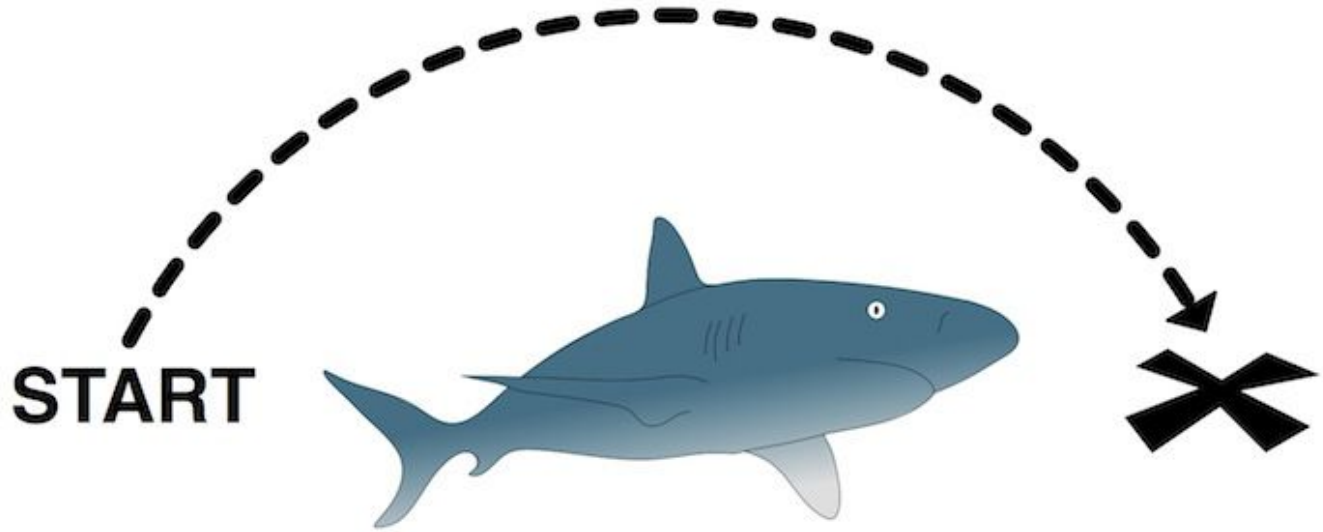
Christopher Zorn
zorn@psu.edu

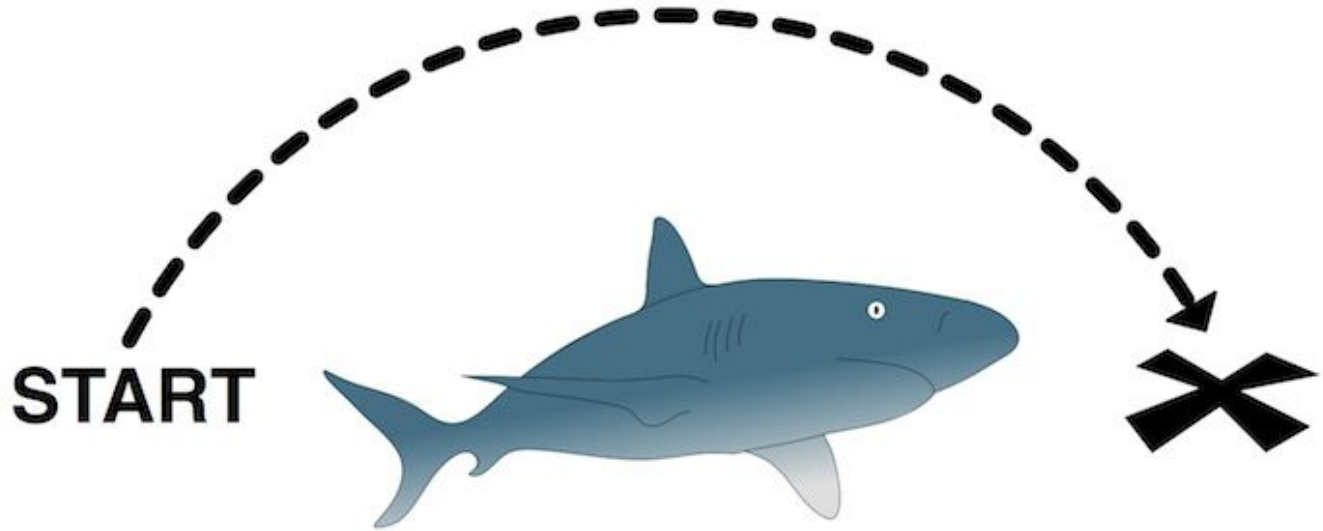
Workshop for Teachers of Quantitative Methods for Social Scientists

University of Oxford, January 8, 2015

Big


Google Trends Figures for 'big data'
Normalized Frequency
0
20
40
60
80
100
2010
2011
2012
2013
2014
2015
Year

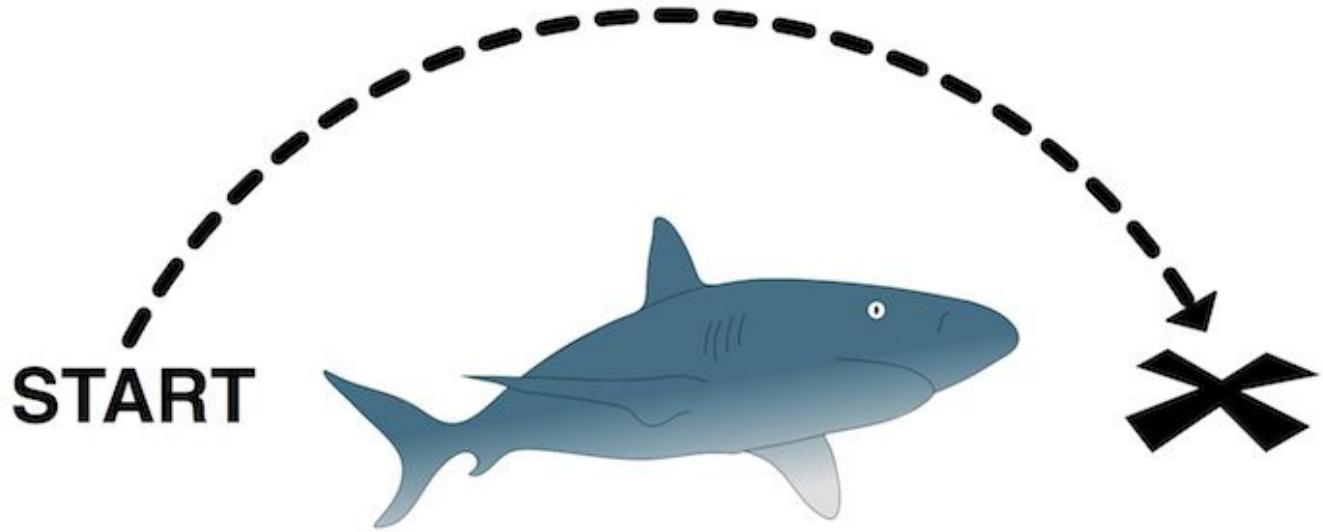

START



Google "big data is dead"

Web News Videos Images Shopping More Search tools

About 65,200 results (0.31 seconds)


START


Big Data
Pictures of Big Data
Big Data is visualized in so many ways...all of them blue and with numbers and lens flare.

Overview

Three things:

- Data
- Tools
- Philosophy

1. Data

Data: Format

What we think of:

name	age	height
Chris	46	72
Zaryab	41	60
Evan	5	44

Data: Format

What we think of:

```
name     age    height
Chris     46      72
Zaryab    41      60
Evan       5      44
```

JSON:

```
[{"name":"Chris","age":46,"height":72},
 {"name":"Zaryab","age":41,"height":60},
 {"name":"Evan","age":5,"height":44}]
```

Data: Format

What we think of:

```
name     age   height
Chris     46      72
Zaryab    41      60
Evan       5      44
```

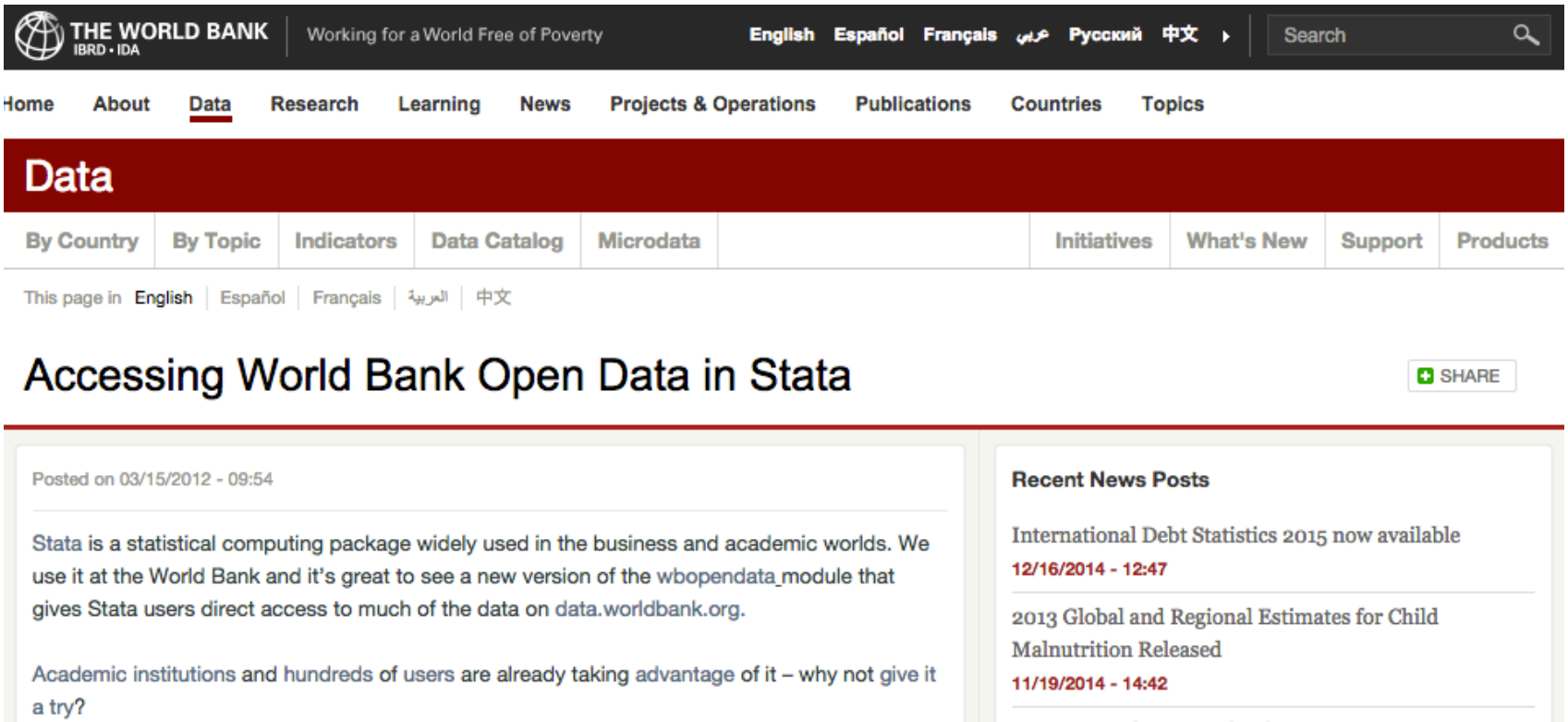
XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<rows>
  <row name age height="Chris 46 72" ></row>
  <row name age height="Zaryab 41 60" ></row>
  <row name age height="Evan 5 44" ></row>
</rows>
```

Data: Sources

- Open government /official data sources
- APIs / streaming data (Google, Twitter, etc.)
- MTurk / crowdsourcing
- “Data exhaust”

Official Data: E.g., Stata's `wbopendata`



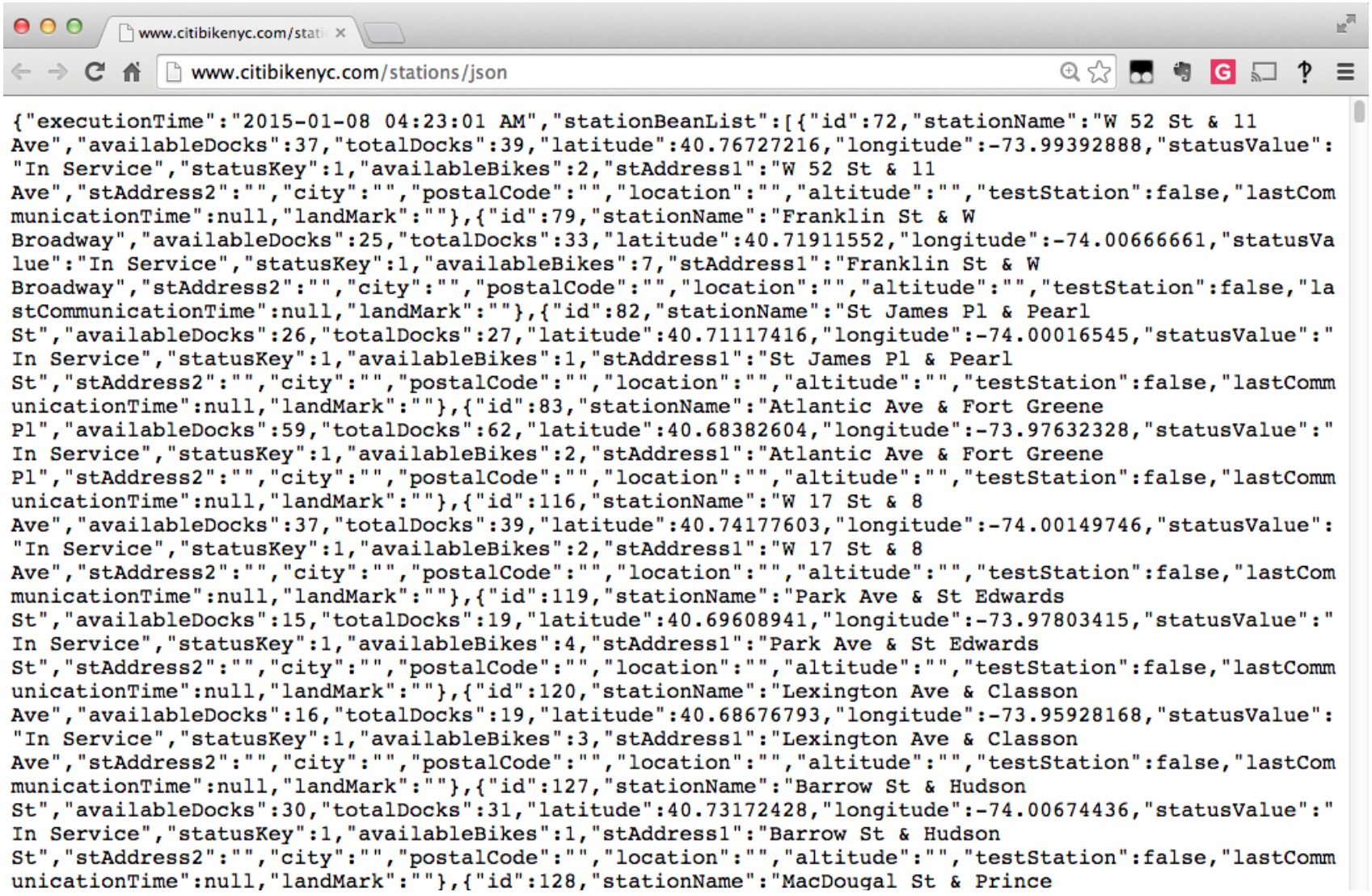
`http://data.worldbank.org/news/accessing-world-bank-open-data-in-stata`

Sources: APIs / Streaming Data

API = *Application programming interface*

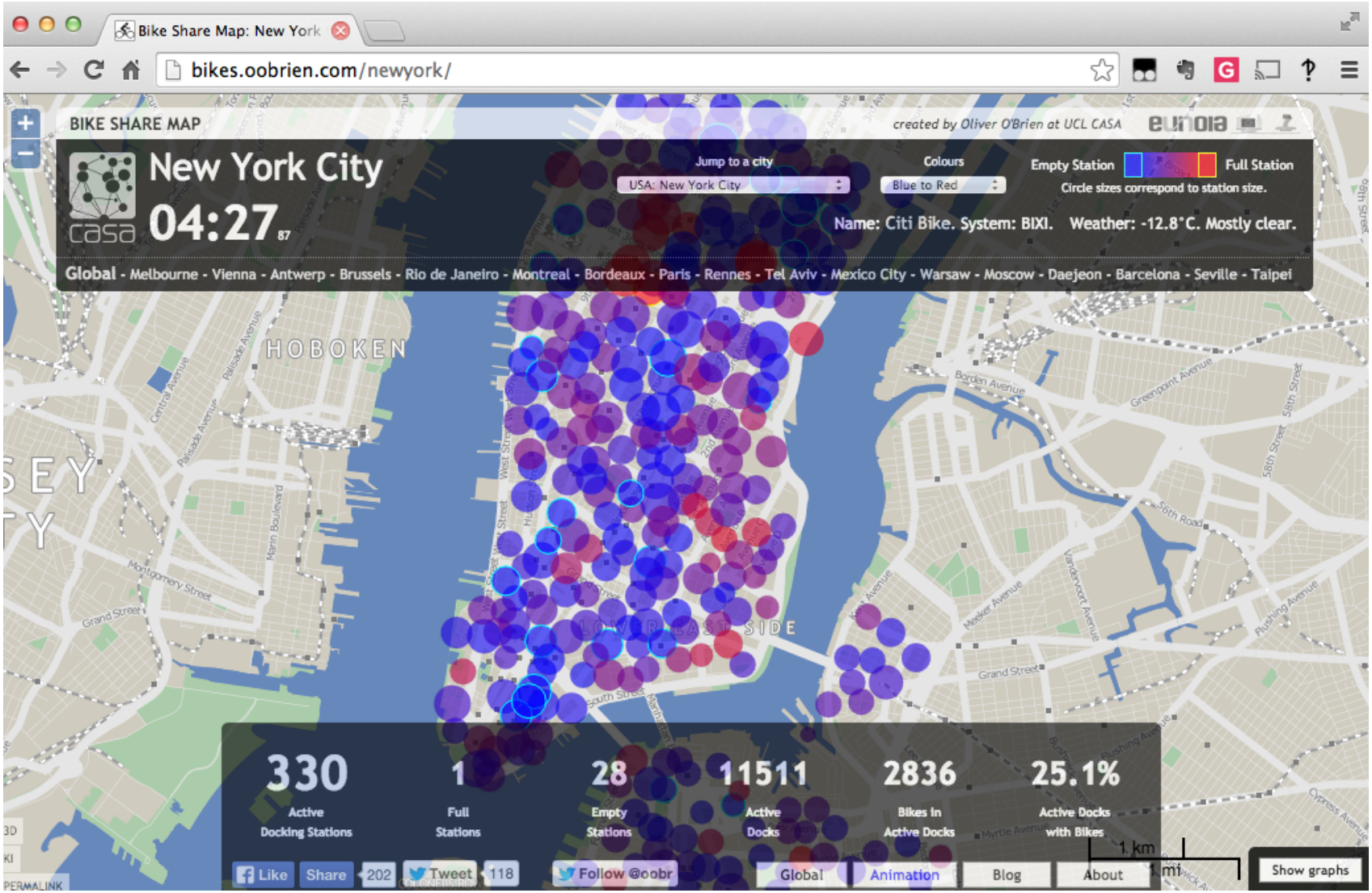
- "...a set of programming instructions and standards for accessing a Web-based software application or Web tool."
- Data portal / stream
- Examples:
    - United Nations Data
    - CitiBikeNYC API (real-time JSON data)
    - Others...

CitiBikeNYC Data

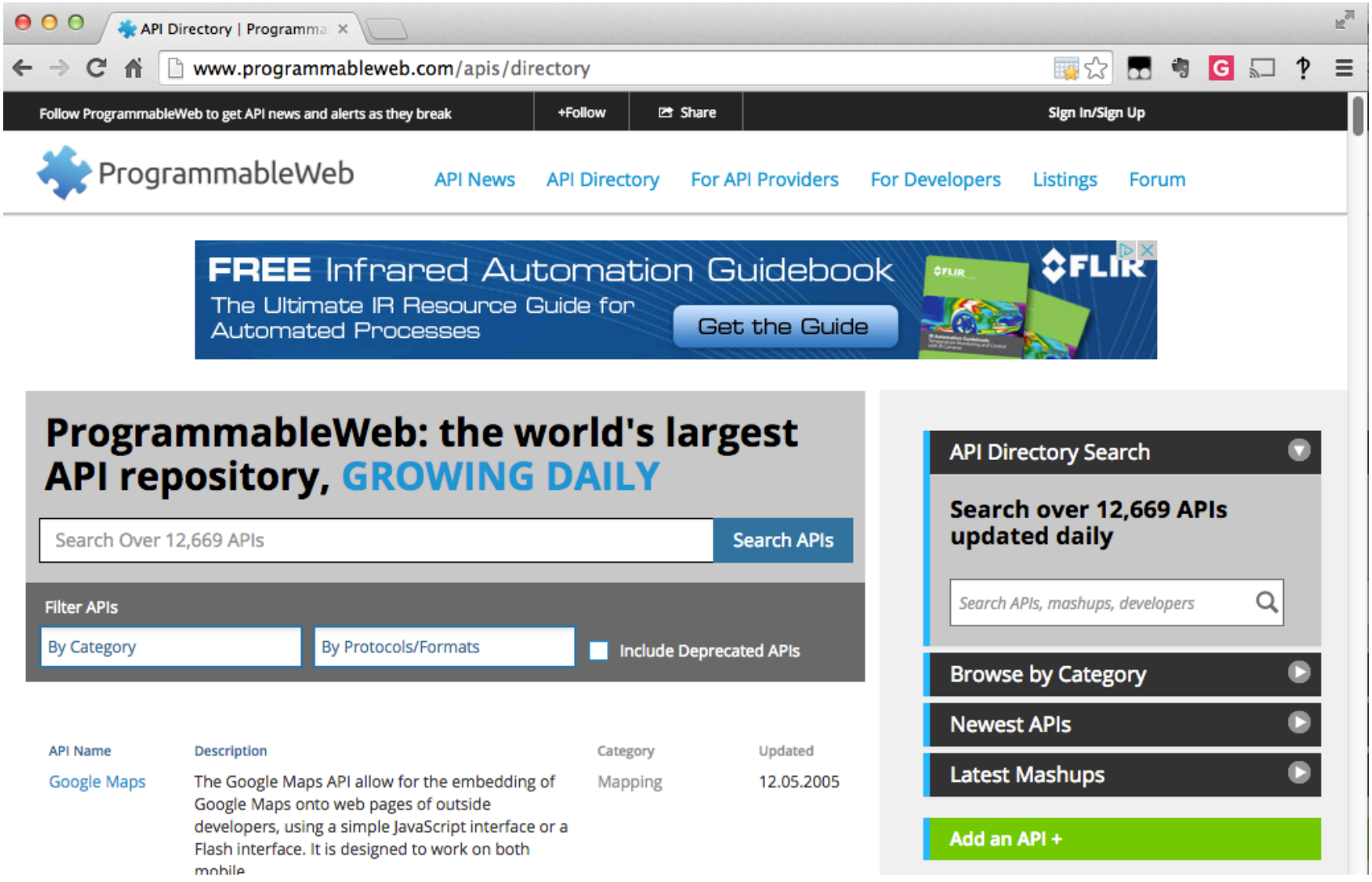



```
{"executionTime":"2015-01-08 04:23:01 AM","stationBeanList":[{"id":72,"stationName":"W 52 St & 11
Ave","availableDocks":37,"totalDocks":39,"latitude":40.76727216,"longitude":-73.99392888,"statusValue":
"In Service","statusKey":1,"availableBikes":2,"stAddress1":"W 52 St & 11
Ave","stAddress2":"","city":"","postalCode":"","location":"","altitude":"","testStation":false,"lastCom
municationTime":null,"landMark":""},{"id":79,"stationName":"Franklin St & W
Broadway","availableDocks":25,"totalDocks":33,"latitude":40.71911552,"longitude":-74.00666661,"statusVa
lue":"In Service","statusKey":1,"availableBikes":7,"stAddress1":"Franklin St & W
Broadway","stAddress2":"","city":"","postalCode":"","location":"","altitude":"","testStation":false,"la
stCommunicationTime":null,"landMark":""},{"id":82,"stationName":"St James Pl & Pearl
St","availableDocks":26,"totalDocks":27,"latitude":40.71117416,"longitude":-74.00016545,"statusValue":"
In Service","statusKey":1,"availableBikes":1,"stAddress1":"St James Pl & Pearl
St","stAddress2":"","city":"","postalCode":"","location":"","altitude":"","testStation":false,"lastComm
unicationTime":null,"landMark":""},{"id":83,"stationName":"Atlantic Ave & Fort Greene
Pl","availableDocks":59,"totalDocks":62,"latitude":40.68382604,"longitude":-73.97632328,"statusValue":"
In Service","statusKey":1,"availableBikes":2,"stAddress1":"Atlantic Ave & Fort Greene
Pl","stAddress2":"","city":"","postalCode":"","location":"","altitude":"","testStation":false,"lastComm
unicationTime":null,"landMark":""},{"id":116,"stationName":"W 17 St & 8
Ave","availableDocks":37,"totalDocks":39,"latitude":40.74177603,"longitude":-74.00149746,"statusValue":
"In Service","statusKey":1,"availableBikes":2,"stAddress1":"W 17 St & 8
Ave","stAddress2":"","city":"","postalCode":"","location":"","altitude":"","testStation":false,"lastCom
municationTime":null,"landMark":""},{"id":119,"stationName":"Park Ave & St Edwards
St","availableDocks":15,"totalDocks":19,"latitude":40.69608941,"longitude":-73.97803415,"statusValue":"
In Service","statusKey":1,"availableBikes":4,"stAddress1":"Park Ave & St Edwards
St","stAddress2":"","city":"","postalCode":"","location":"","altitude":"","testStation":false,"lastComm
unicationTime":null,"landMark":""},{"id":120,"stationName":"Lexington Ave & Classon
Ave","availableDocks":16,"totalDocks":19,"latitude":40.68676793,"longitude":-73.95928168,"statusValue":
"In Service","statusKey":1,"availableBikes":3,"stAddress1":"Lexington Ave & Classon
Ave","stAddress2":"","city":"","postalCode":"","location":"","altitude":"","testStation":false,"lastCom
municationTime":null,"landMark":""},{"id":127,"stationName":"Barrow St & Hudson
St","availableDocks":30,"totalDocks":31,"latitude":40.73172428,"longitude":-74.00674436,"statusValue":"
In Service","statusKey":1,"availableBikes":1,"stAddress1":"Barrow St & Hudson
St","stAddress2":"","city":"","postalCode":"","location":"","altitude":"","testStation":false,"lastComm
unicationTime":null,"landMark":""},{"id":128,"stationName":"MacDougal St & Prince
```

CitiBikeNYC: Data → Maps



Programmable Web (API clearinghouse)



`http://www.programmableweb.com/apis/directory`

R Tools

General packages:

- `RCurl`, `httr`
- JSON: `jsonlite`, `rjson`
- XML: `XML`, `R4X`

R Tools

General packages:

- `RCurl`, `httr`
- JSON: `jsonlite`, `rjson`
- XML: `XML`, `R4X`

Specific API data tools:

- Google: `RGA`, `googleVis`, `translate`, others
- `twitteR` (Twitter)
- `Rook` / `Rfacebook` (Facebook)
- `tumblR` (Tumblr)
- Others: `dvn` (Dataverse), `WDI` (World Bank), etc. (see `http://cran.r-project.org/web/views/WebTechnologies.html`)

Simple API Example: Twitter

Setup:

```
library(httr)
library(twitteR)

reqURL <- "https://api.twitter.com/oauth/request_token"
accessURL <- "https://api.twitter.com/oauth/access_token"
authURL <- "https://api.twitter.com/oauth/authorize"
APIKey<-"ieDpmBzQ1khjVlbrR5RoAZL09"
APISecret<-"PnhfyeA3XbNOxMswN68z9hxThm5e5Z1AQ9dTYjvQllf0VGQQwk"
twitCred <- OAuthFactory$new(consumerKey=APIKey,
                             consumerSecret=APISecret,
                             requestURL=reqURL,
                             accessURL=accessURL,
                             authURL=authURL)
twitCred$handshake
```

Simple API Example: Twitter

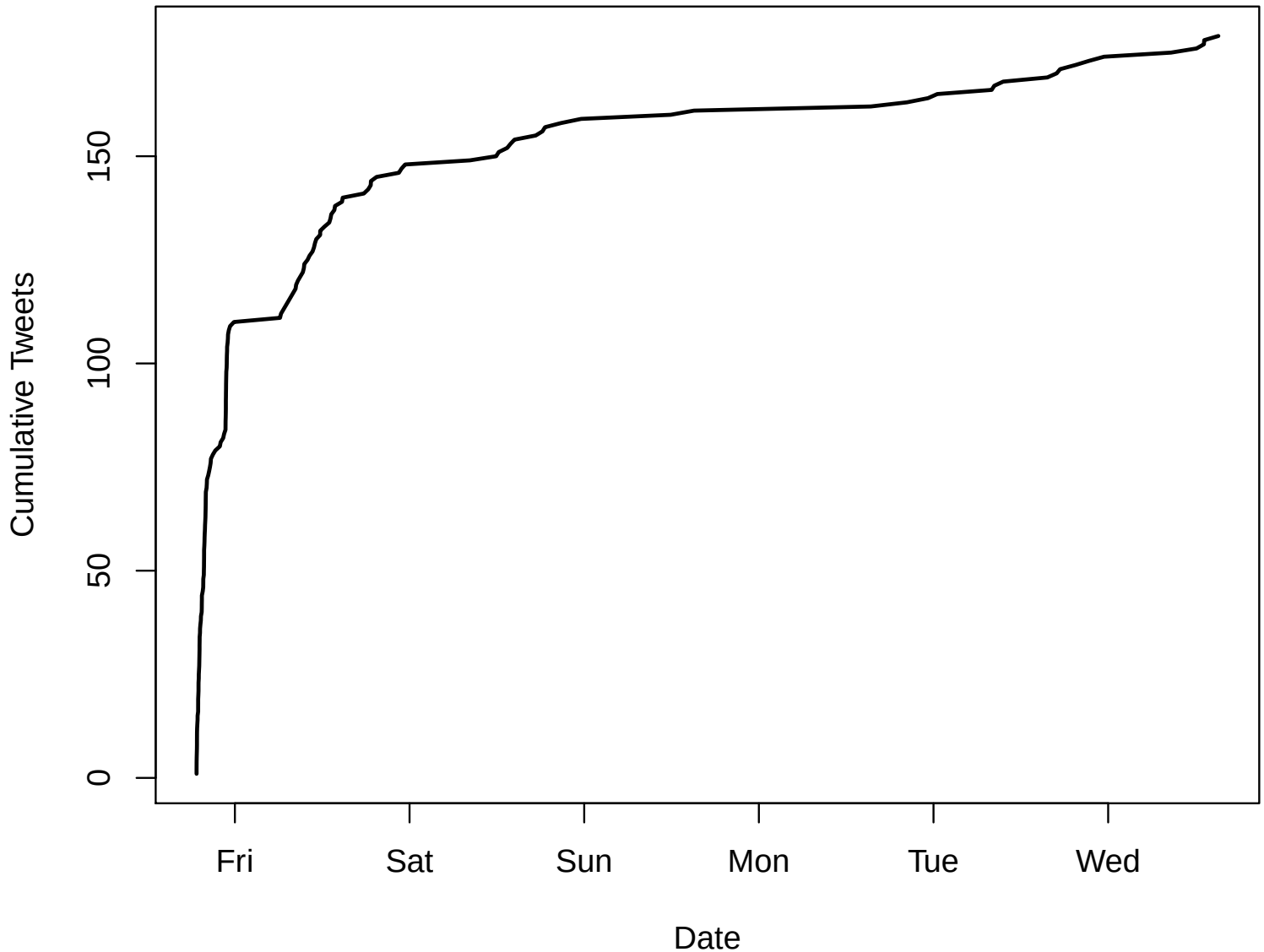
Simple API Example: Twitter

Pull "local" tweets:

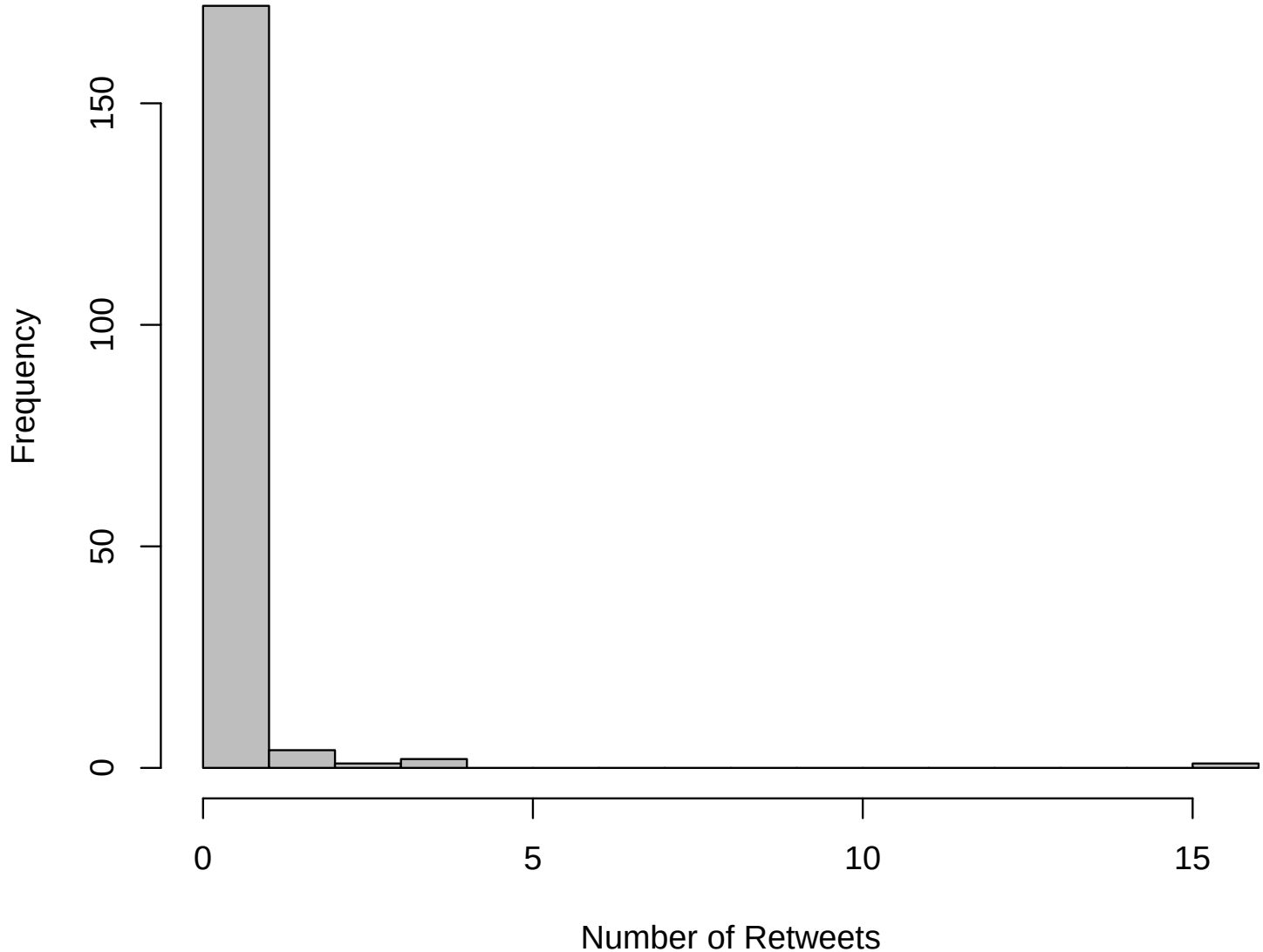
```
KaneTweets<-searchTwitter('Kane',geocode='51.603211,
    -0.066430, 10mi', since='2014-12-31',
    n=5000, retryOnRateLimit=1)

KaneTweetsData<-twListToDF(KaneTweets)
```

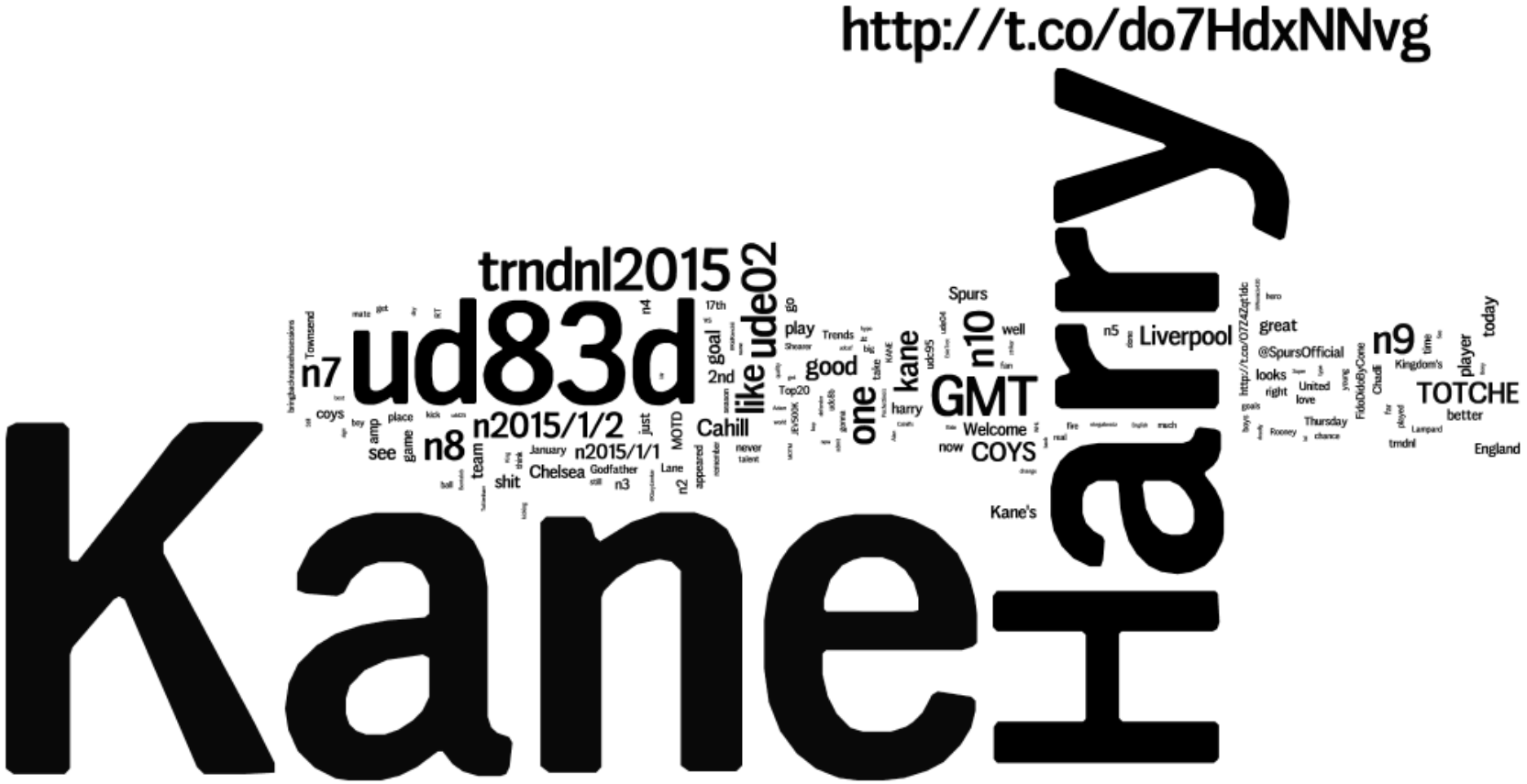
Cumulative Number of Tweets



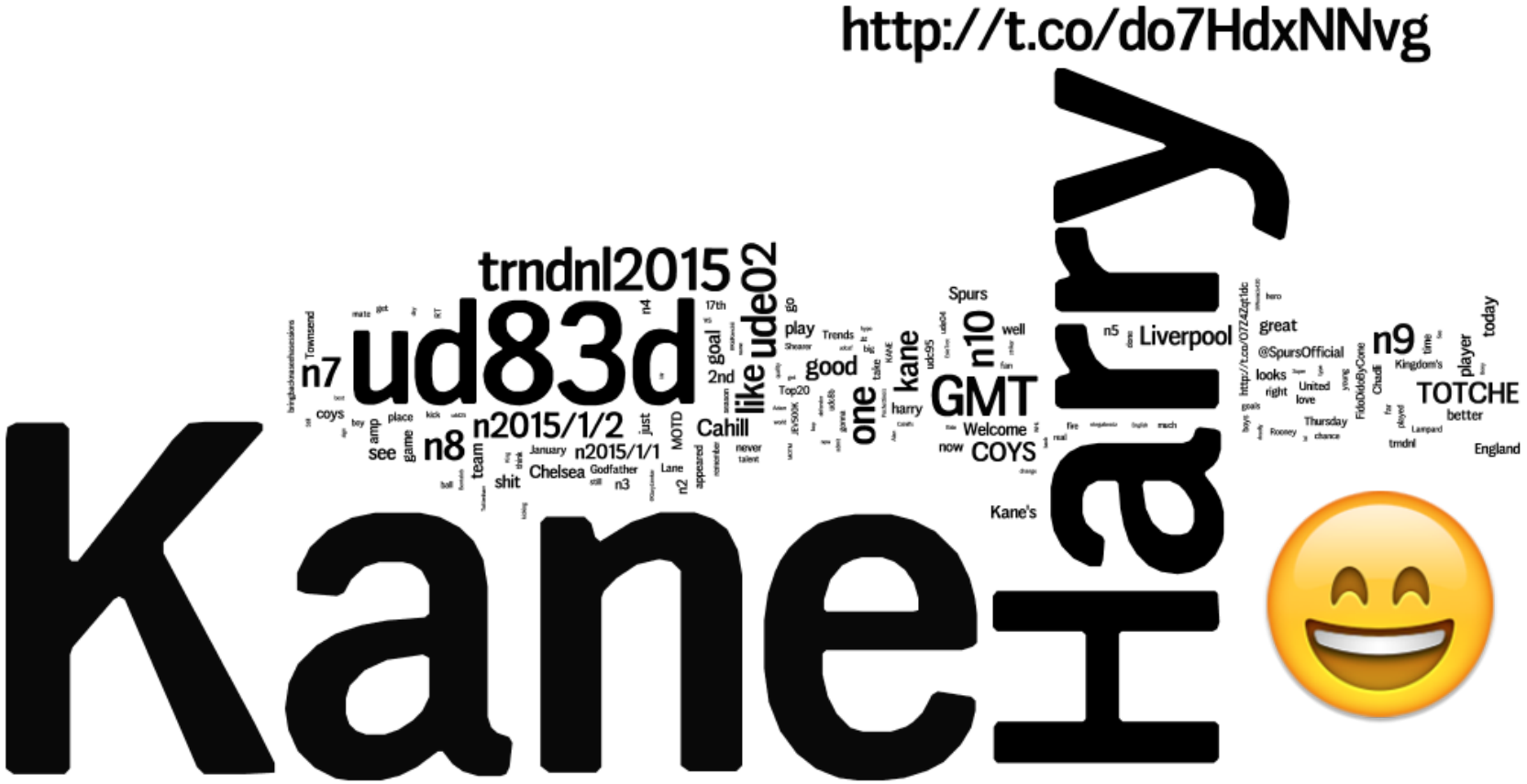
Histogram: Number of Retweets



Harry Kane Word Cloud



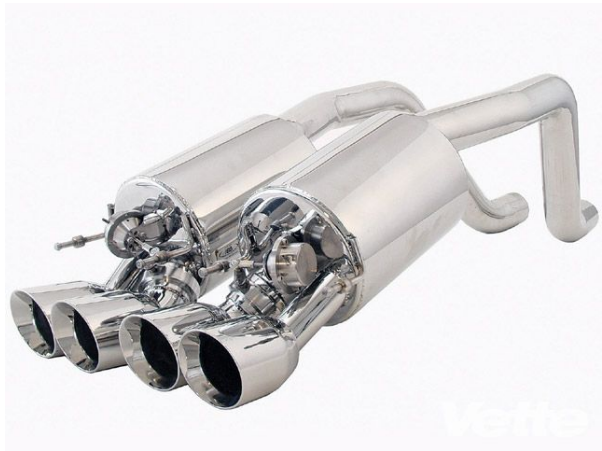
Harry Kane Word Cloud



“Data Exhaust”

“Data Exhaust”

- Large public records



“Data Exhaust”

- Large public records
- Geotags / date stamps / etc.

“Data Exhaust”

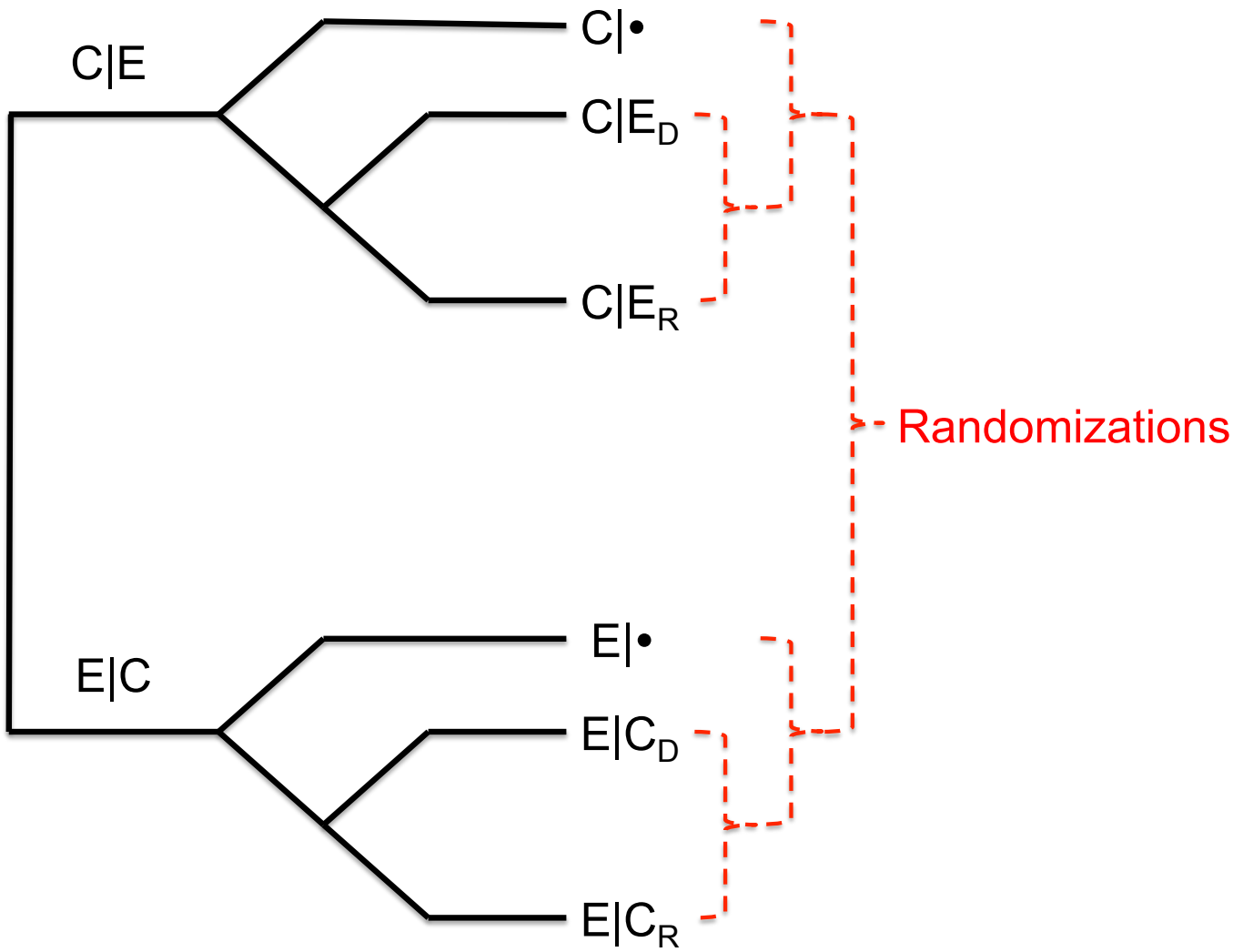
- Large public records
- Geotags / date stamps / etc.
- Commercial / financial transactions

“Data Exhaust”

- Large public records
- Geotags / date stamps / etc.
- Commercial / financial transactions
- **Linkage** data (e.g., *your friends’* locations)

Crowdsourcing

Crowdsourcing

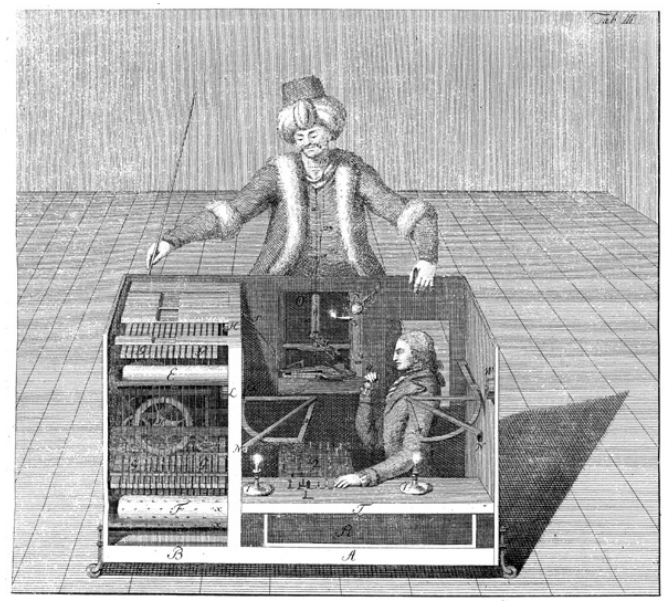


MTurk + Qualtrics

MTurk + Qualtrics

Details:

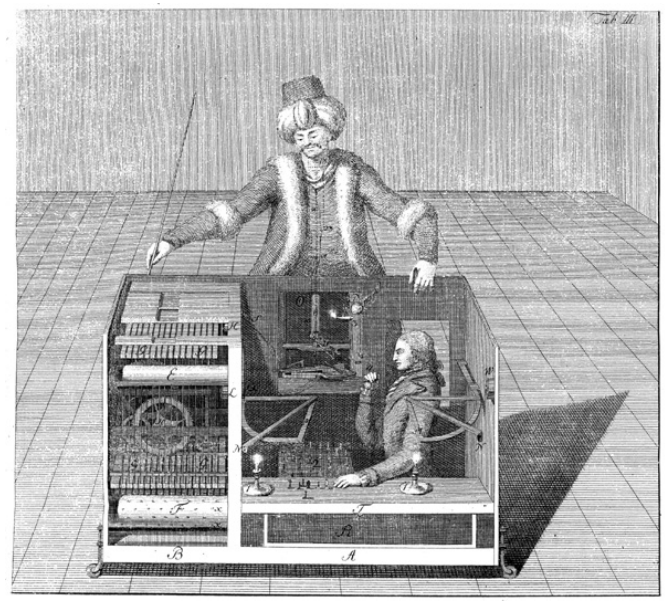
- 25 question, 10-minute survey
- 703 respondents
- *One week*



MTurk + Qualtrics

Details:

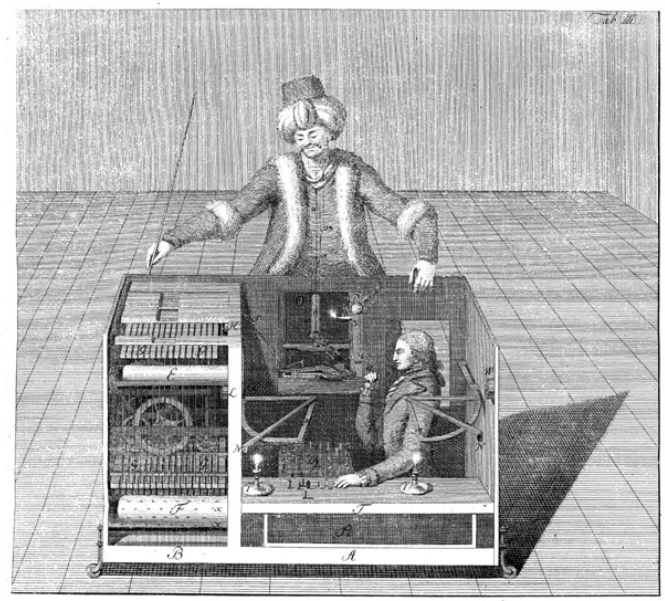
- 25 question, 10-minute survey
- 703 respondents
- *One week*
- **£100 (U.S. $150)**



MTurk + Qualtrics

Details:

- 25 question, 10-minute survey
- 703 respondents
- *One week*
- **£100 (U.S. $150)**

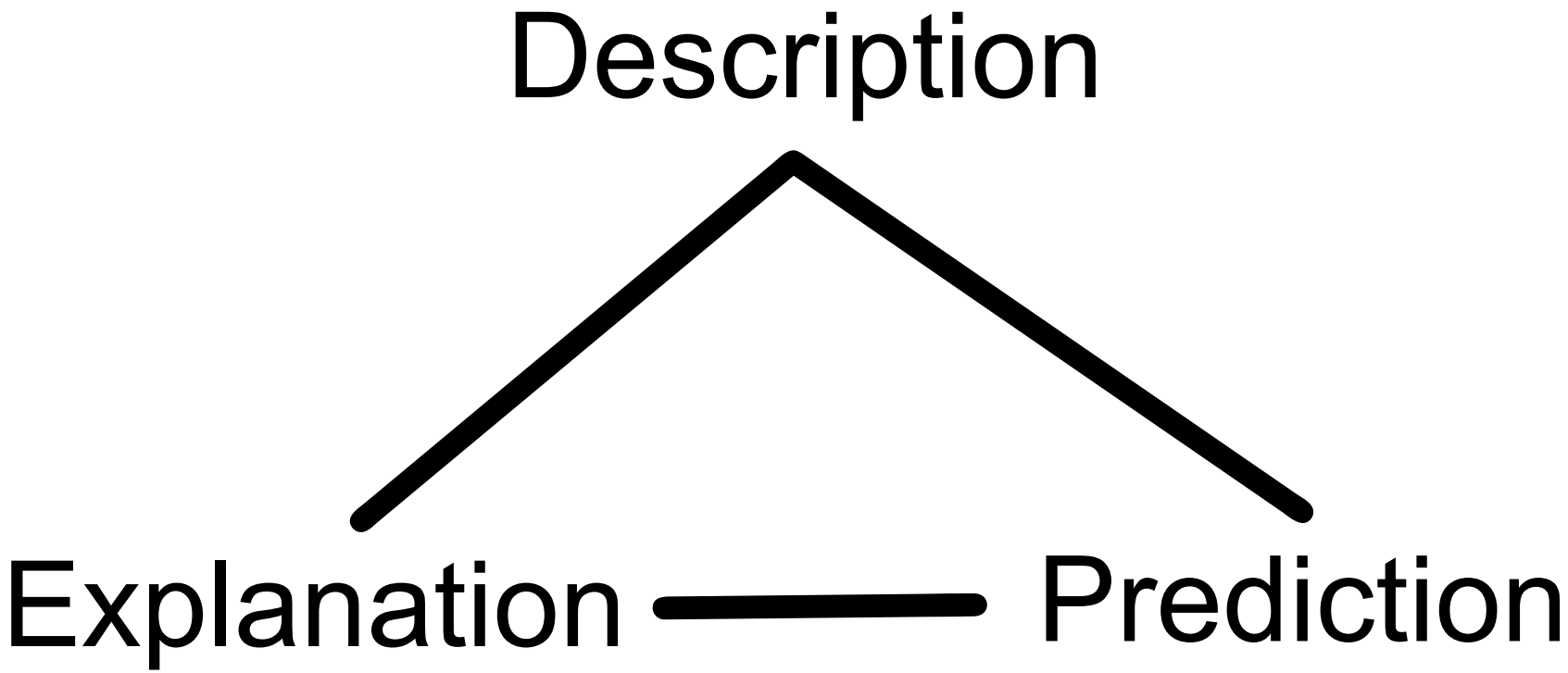


Other Advantages:

- Theory → Operationalization → Measurement → Analysis
- Hands-on Design Experience
- "Real" Challenges To Inference

2. Tools

Three Things



Explanation > Description > Prediction

Descriptive Tools: Visualization

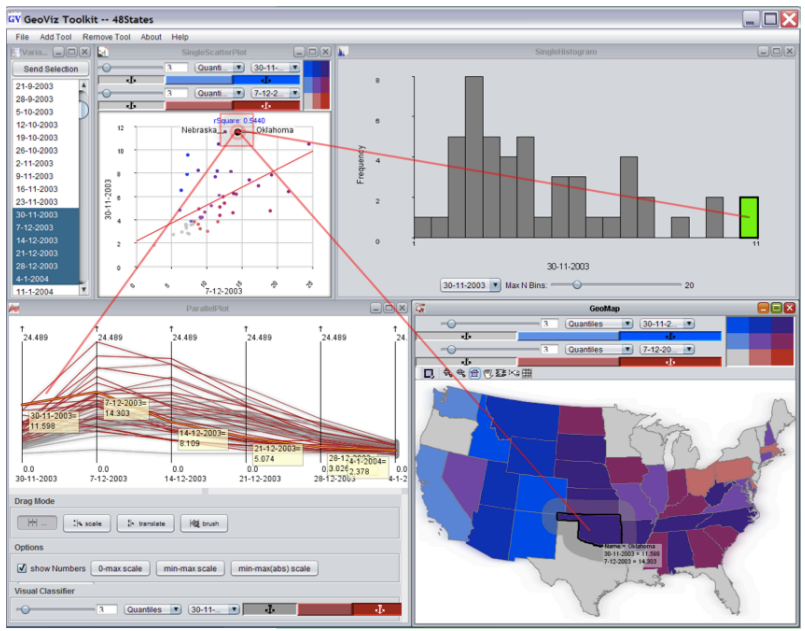
Descriptive Tools: Visualization

Some facts:

Descriptive Tools: Visualization

Some facts:

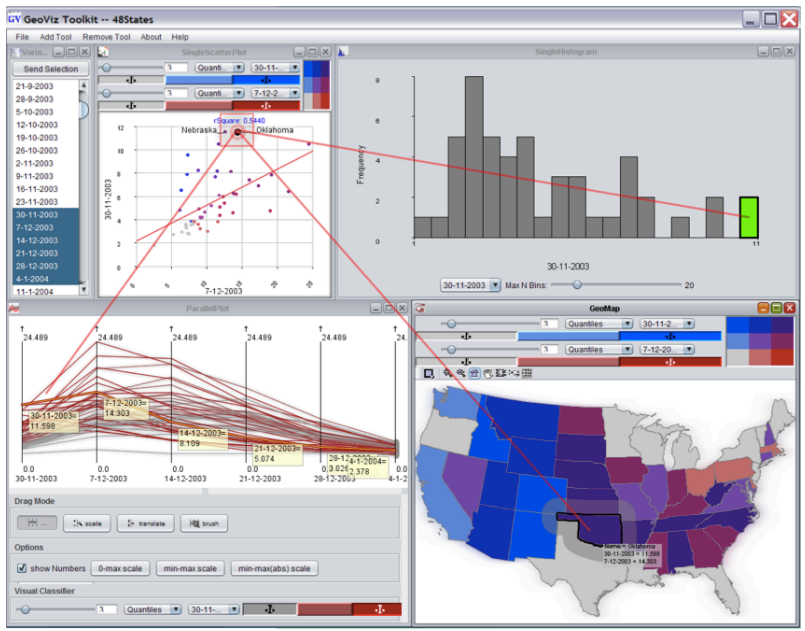
- People like pictures



Descriptive Tools: Visualization

Some facts:

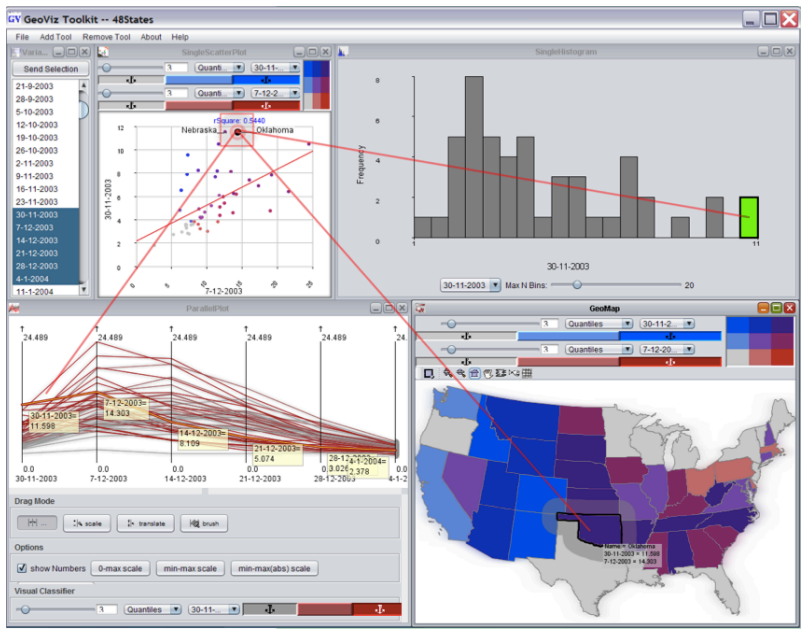
- People like pictures
- People love maps



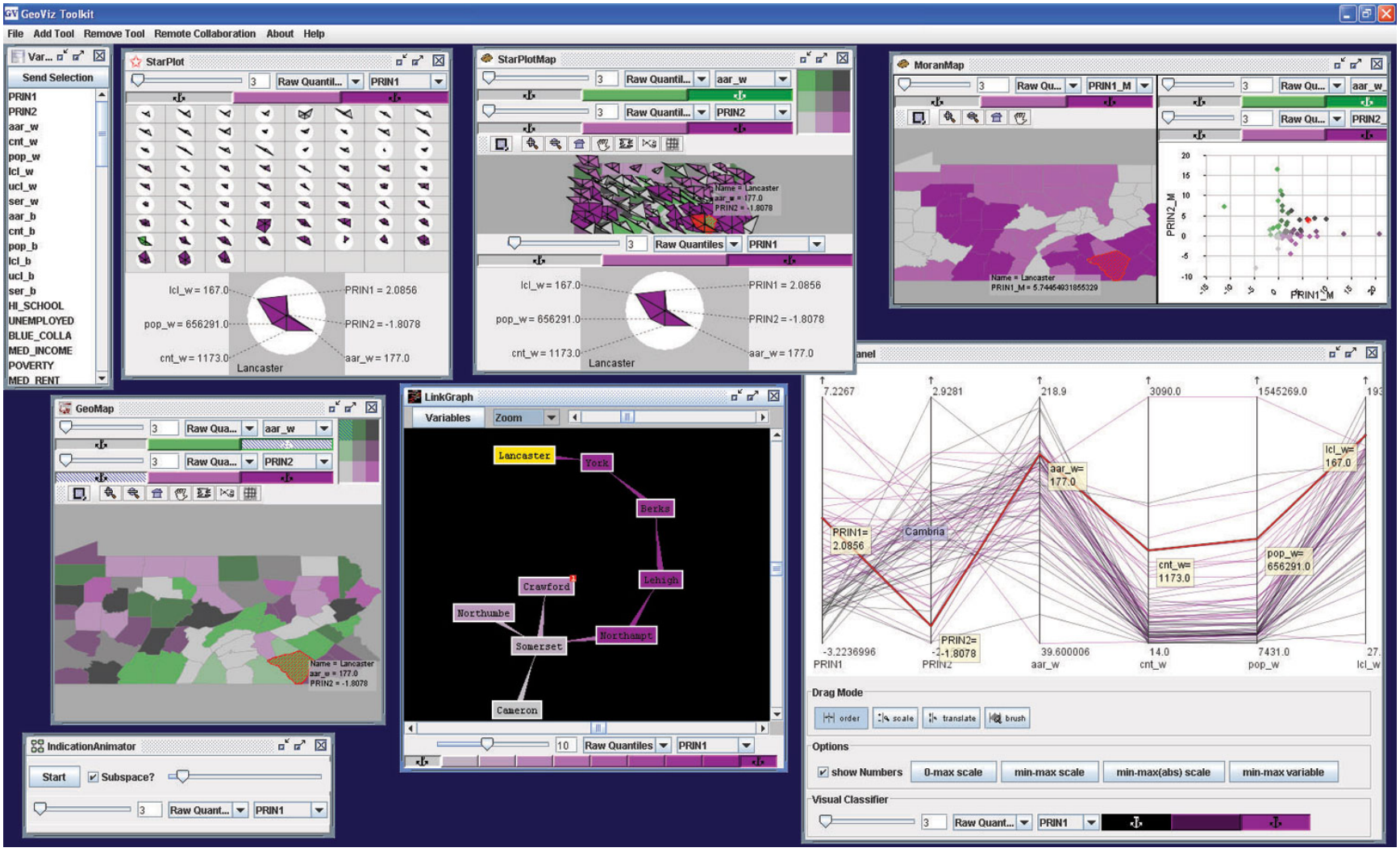
Descriptive Tools: Visualization

Some facts:

- People like pictures
- People love maps
- Making both is easier than ever



High-Dimensional Visualization



http://www.geovista.psu.edu

2D and 3D Mapping

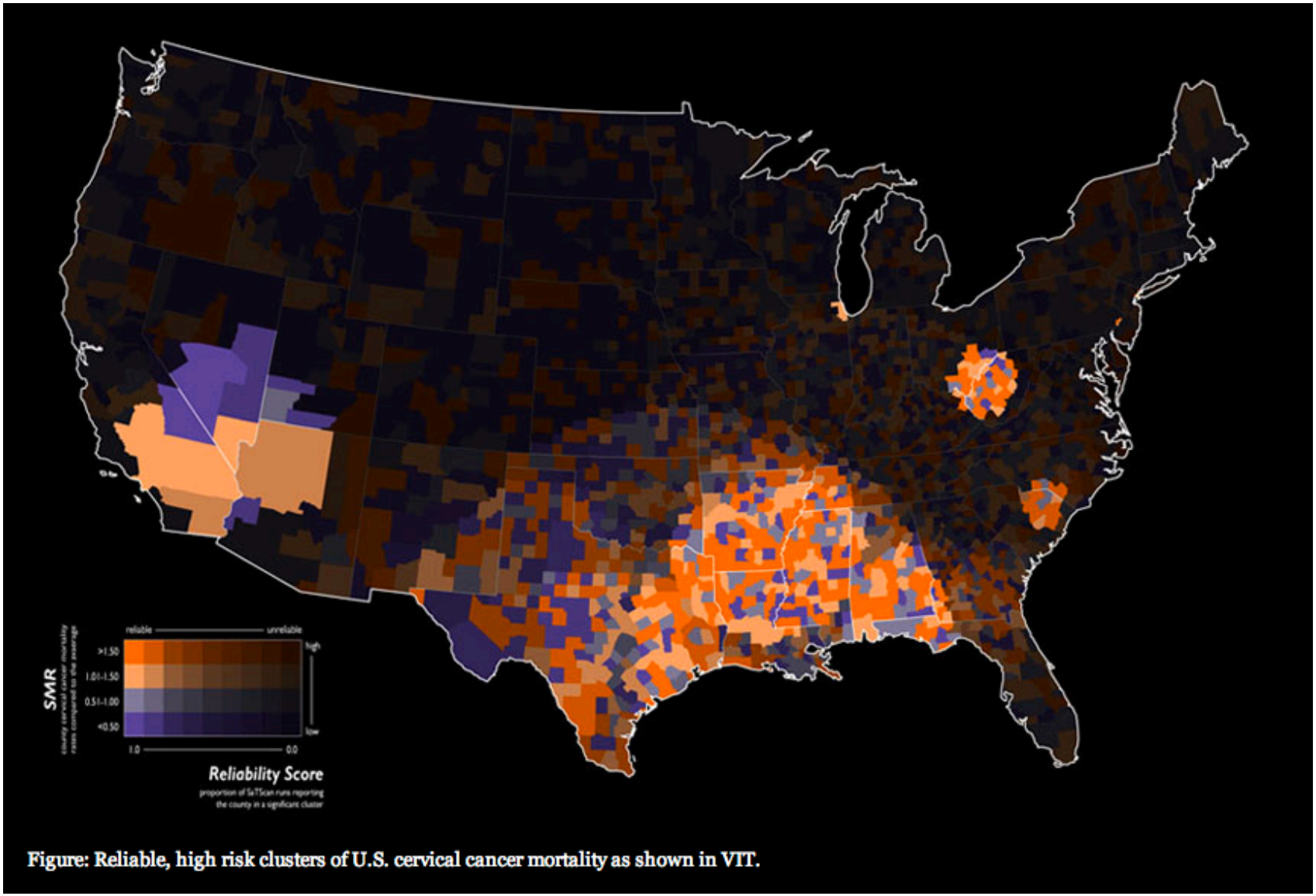


Figure: Reliable, high risk clusters of U.S. cervical cancer mortality as shown in VIT.

Tools: Prediction

Tools: Prediction

Opportunities:

- Real-Time Data = Real-Time Out-Of-Sample Predictions

Tools: Prediction

Opportunities:

- Real-Time Data = Real-Time Out-Of-Sample Predictions
- Cross-Validation

Tools: Prediction

Opportunities:

- Real-Time Data = Real-Time Out-Of-Sample Predictions
- Cross-Validation
- Kaggle, etc.

Tools: Prediction

Opportunities:

- Real-Time Data = Real-Time Out-Of-Sample Predictions
- Cross-Validation
- Kaggle, etc.

Challenges:

- Timing

Tools: Prediction

Opportunities:

- Real-Time Data = Real-Time Out-Of-Sample Predictions
- Cross-Validation
- Kaggle, etc.

Challenges:

- Timing
- Uncertainty

Tools: Prediction

Opportunities:

- Real-Time Data = Real-Time Out-Of-Sample Predictions
- Cross-Validation
- Kaggle, etc.

Challenges:

- Timing
- Uncertainty
- The “Arms Race”

3. Theory

TECHNOLOGY

Big data and the death of the theorist

By Ian Steadman | 25 January 13

TECHNOLOGY

Big data and the death of the theorist

By Ian Steadman 6 January 13

Whither (Wither?) Theory?

Whither (Wither?) Theory?

- The Counterargument: "It's more important than ever!"

Whither (Wither?) Theory?

- The Counterargument: "It's more important than ever!"
- The Power of Description

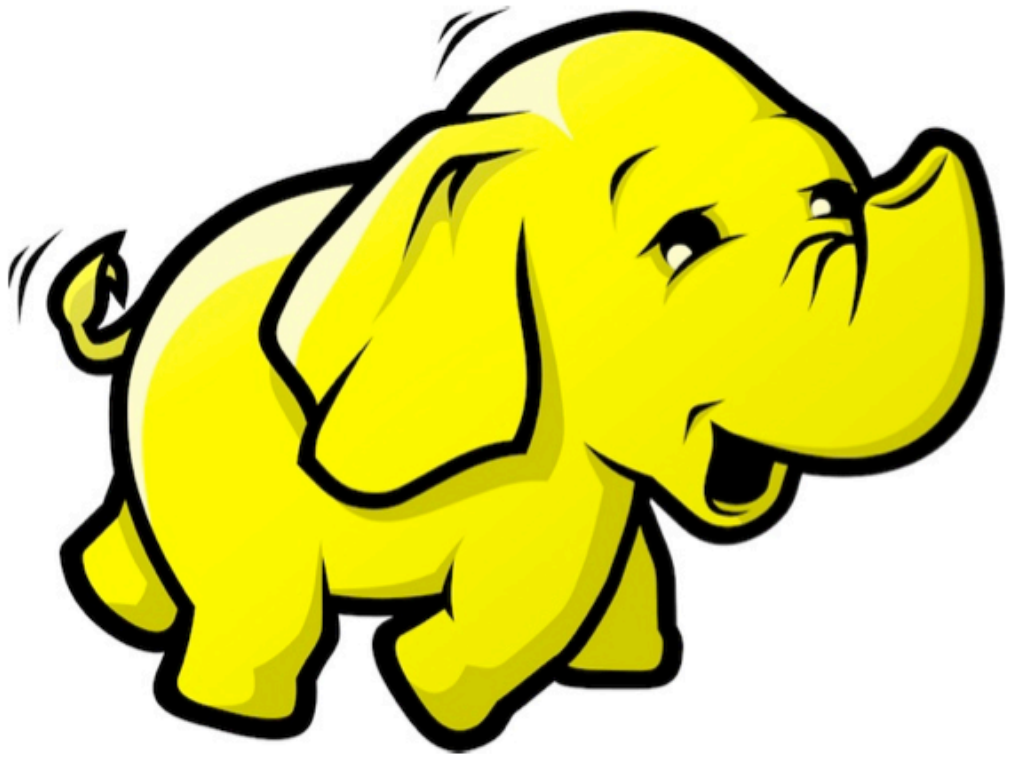
Whither (Wither?) Theory?

- The Counterargument: "It's more important than ever!"
- The Power of Description
- Learn From Data

Whither (Wither?) Theory?

- The Counterargument: "It's more important than ever!"
- The Power of Description
- Learn From Data
- Change your Borrowing Habits

Think different



Theorize

~~Think~~ different

What Ought We Do (Differently)?

What Ought We Do (Differently)?

- Learn some (new?) things
  - Python/Perl/whatever
  - JSON and/or XML data formats

What Ought We Do (Differently)?

- Learn some (new?) things
  - Python/Perl/whatever
  - JSON and/or XML data formats
- Teach (a little) **programming**

What Ought We Do (Differently)?

- Learn some (new?) things
  - Python/Perl/whatever
  - JSON and/or XML data formats
- Teach (a little) **programming**
- Emphasis on **description** / **visualization** and **exploration**

What Ought We Do (Differently)?

- Learn some (new?) things
  - Python/Perl/whatever
  - JSON and/or XML data formats
- Teach (a little) **programming**
- Emphasis on **description** / **visualization** and **exploration**
- Greater attention to **prediction**

What Ought We Do (Differently)?

- Learn some (new?) things
  - Python/Perl/whatever
  - JSON and/or XML data formats
- Teach (a little) **programming**
- Emphasis on **description** / **visualization** and **exploration**
- Greater attention to **prediction**
- **Theorize different(ly)**: A little more induction, a little more humility

Thank you

https://github.com/PrisonRodeo/QM2015