

# cognostics metrics for data visualization

Barret Schloerke  
Statistics PhD Candidate  
Purdue University

# About myself

# About myself

- **Purdue University**
  - 4th Year PhD Candidate in Statistics
  - Research in large data visualization using R -  
[www.tessera.io](http://www.tessera.io)
    - Dr. William Cleveland and Dr. Ryan Hafen

# About myself

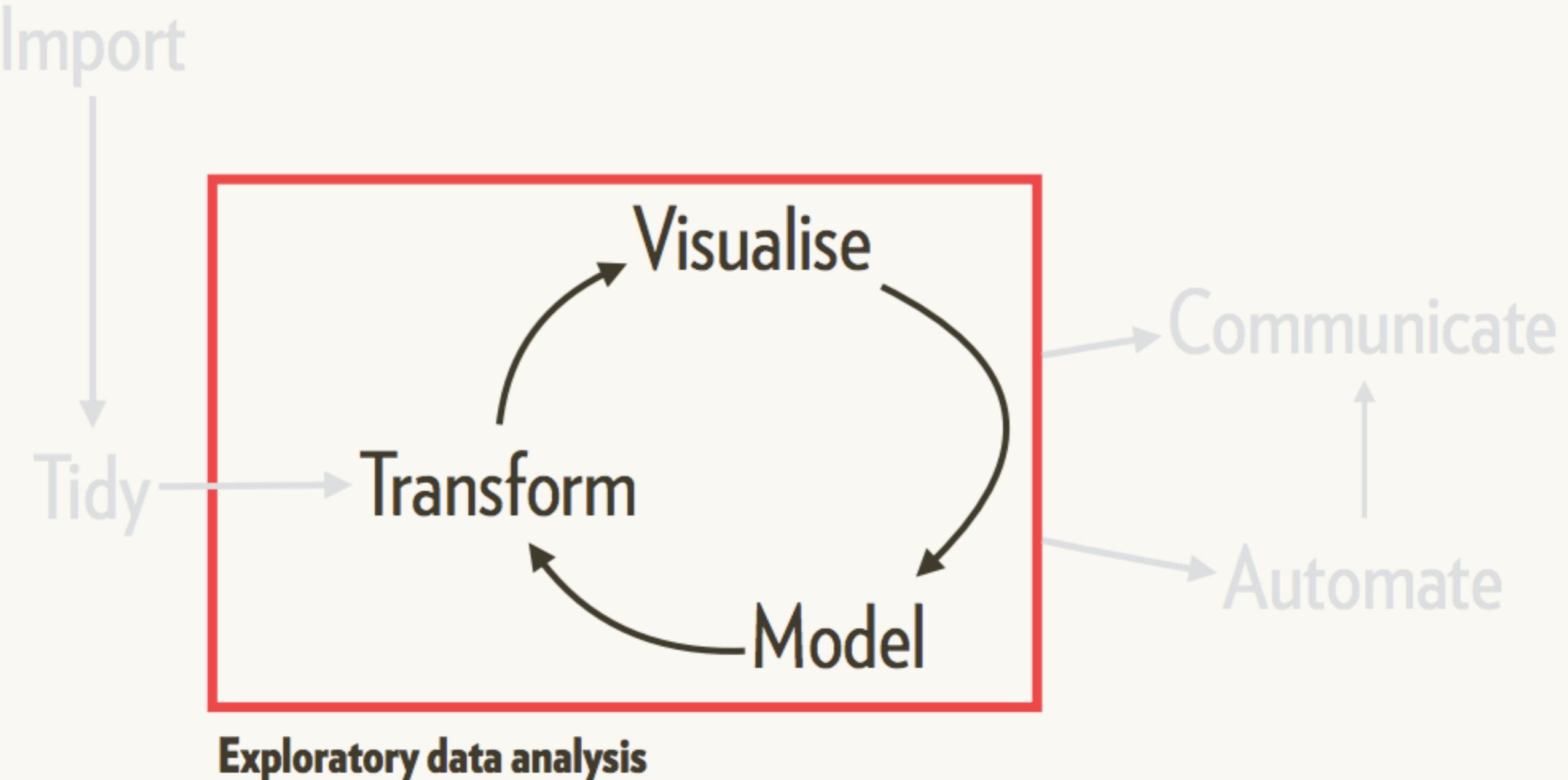
- **Purdue University**
  - 4th Year PhD Candidate in Statistics
  - Research in large data visualization using R -  
[www.tessera.io](http://www.tessera.io)
    - Dr. William Cleveland and Dr. Ryan Hafen
- [Metamarkets.com](http://Metamarkets.com) - 1.5 years
  - Front end engineer - node.js

# About myself

- **Purdue University**
  - 4th Year PhD Candidate in Statistics
  - Research in large data visualization using R -  
[www.tessera.io](http://www.tessera.io)
    - Dr. William Cleveland and Dr. Ryan Hafen
- [Metamarkets.com](http://Metamarkets.com) - 1.5 years
  - Front end engineer - node.js
- **Iowa State University**
  - B.S. in Computer Engineering
  - Research in statistical data visualization with R
    - Dr. Di Cook, Dr. Hadley Wickham, and Dr. Heike Hofmann

# Exploratory Data Analysis

# Exploratory Data Analysis



# Visualization of Large Data

# Visualization of Large Data

# Visualization of Large Data

- Most large data visualization tools or approaches either
  - Summarize the large data to create a single plot
  - Are very specialized and heavily engineered for a particular domain

# Visualization of Large Data

- Most large data visualization tools or approaches either
  - Summarize the large data to create a single plot
  - Are very specialized and heavily engineered for a particular domain
- Summaries are critical but can hide very interesting phenomena (e.g. Simpson's Paradox)

# Visualization of Large Data

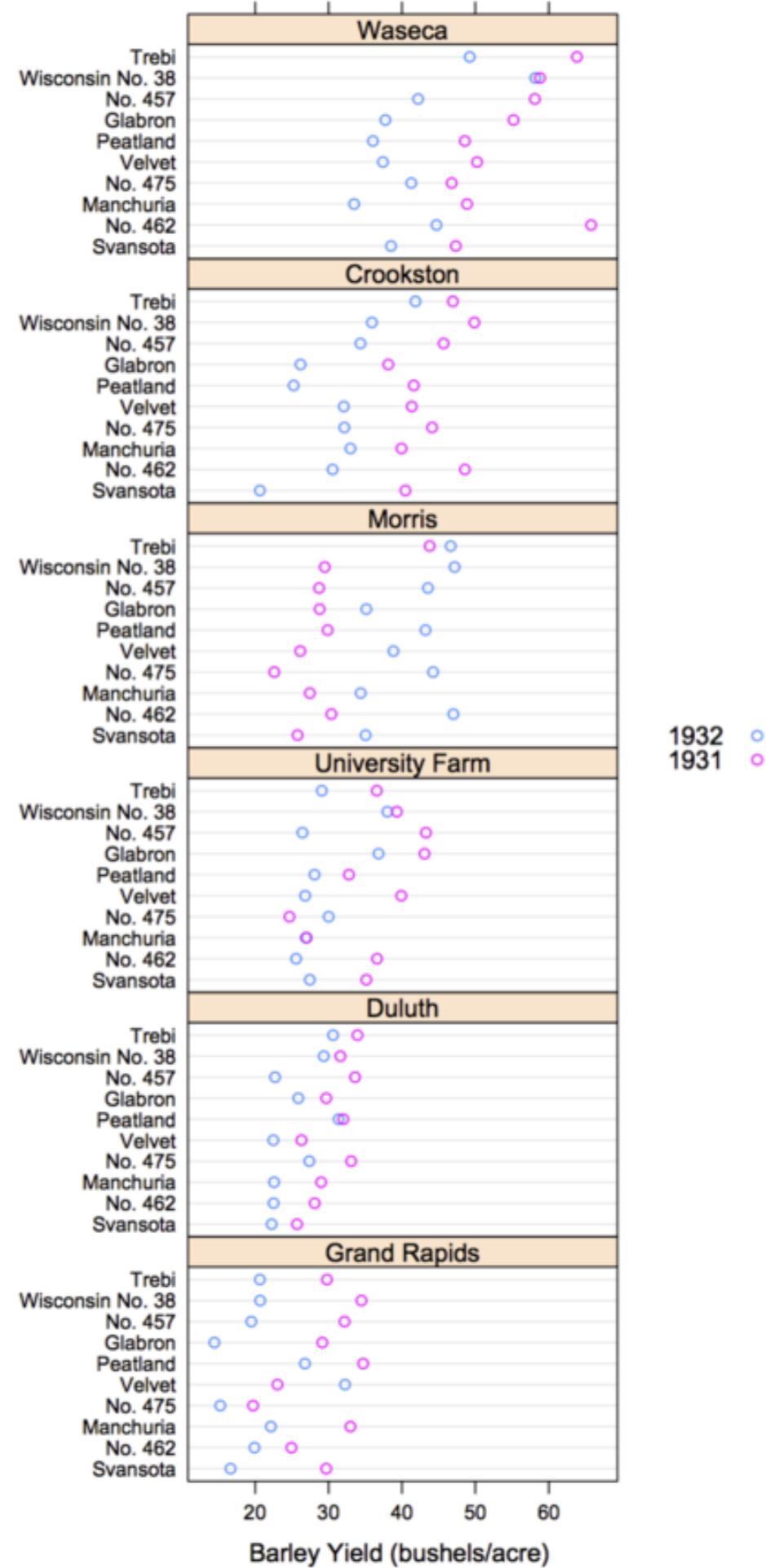
- Most large data visualization tools or approaches either
  - Summarize the large data to create a single plot
  - Are very specialized and heavily engineered for a particular domain
- Summaries are critical but can hide very interesting phenomena (e.g. Simpson's Paradox)
- Specialized tools can be useful but do not fit Exploratory Data Analysis paradigm (slow implementation)

# Visualization of Large Data

- Most large data visualization tools or approaches either
  - Summarize the large data to create a single plot
  - Are very specialized and heavily engineered for a particular domain
- Summaries are critical but can hide very interesting phenomena (e.g. Simpson's Paradox)
- Specialized tools can be useful but do not fit Exploratory Data Analysis paradigm (slow implementation)

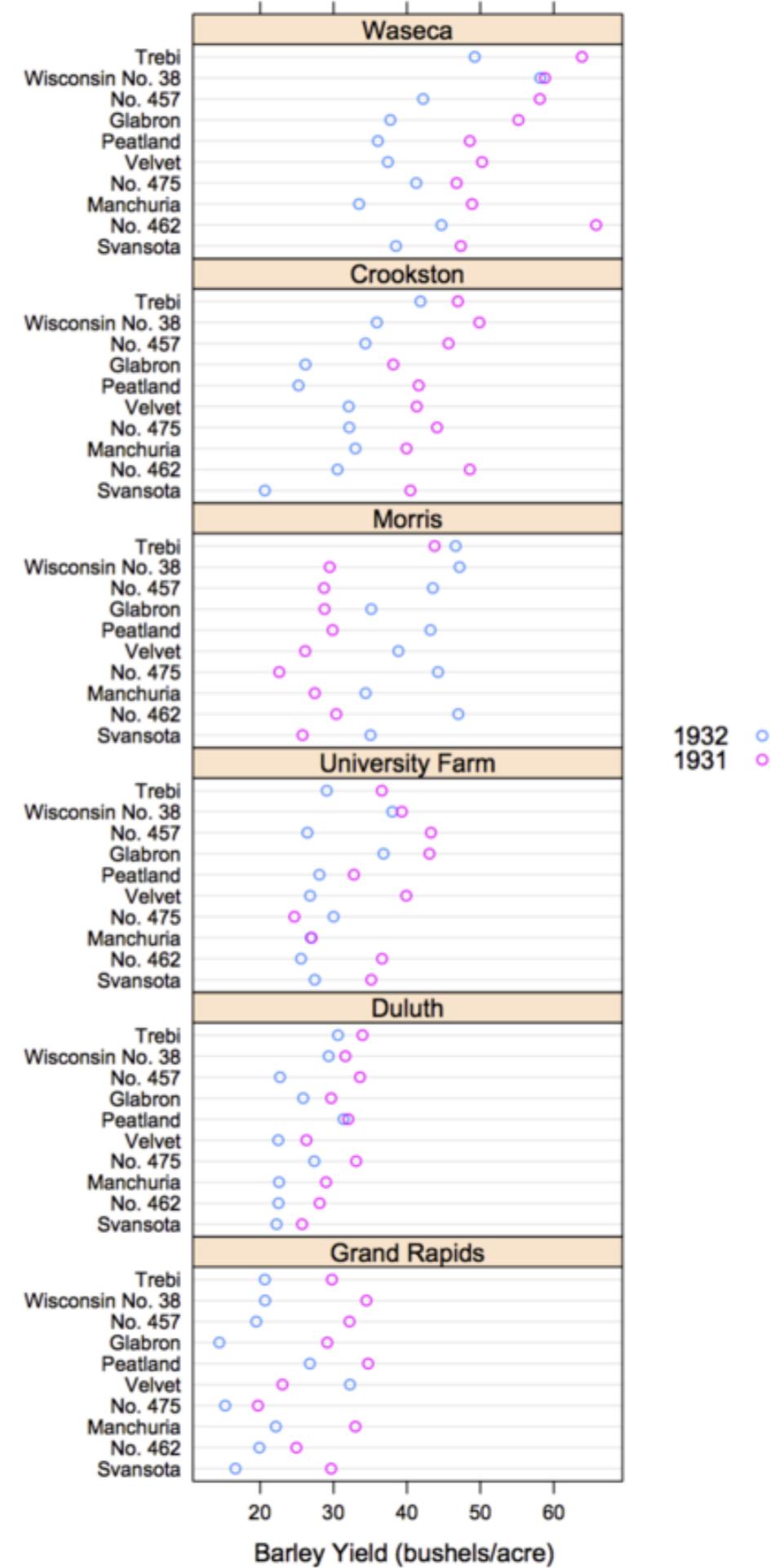
**We must be able to flexibly visualize complex data in detail even when the data is large!**

# Trellis Display



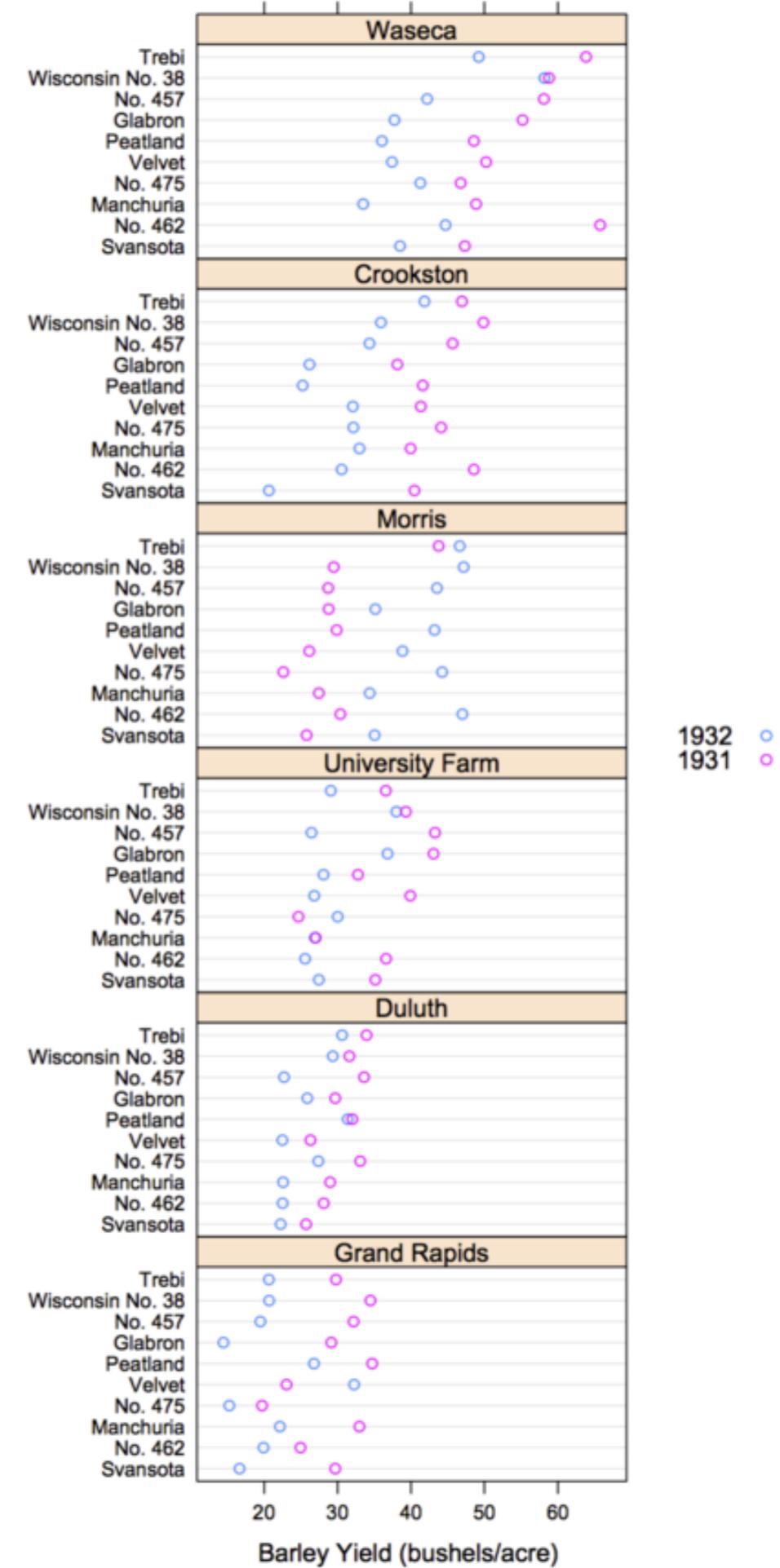
# Trellis Display

- Data is split into meaningful subsets, usually conditioning on variables of the dataset



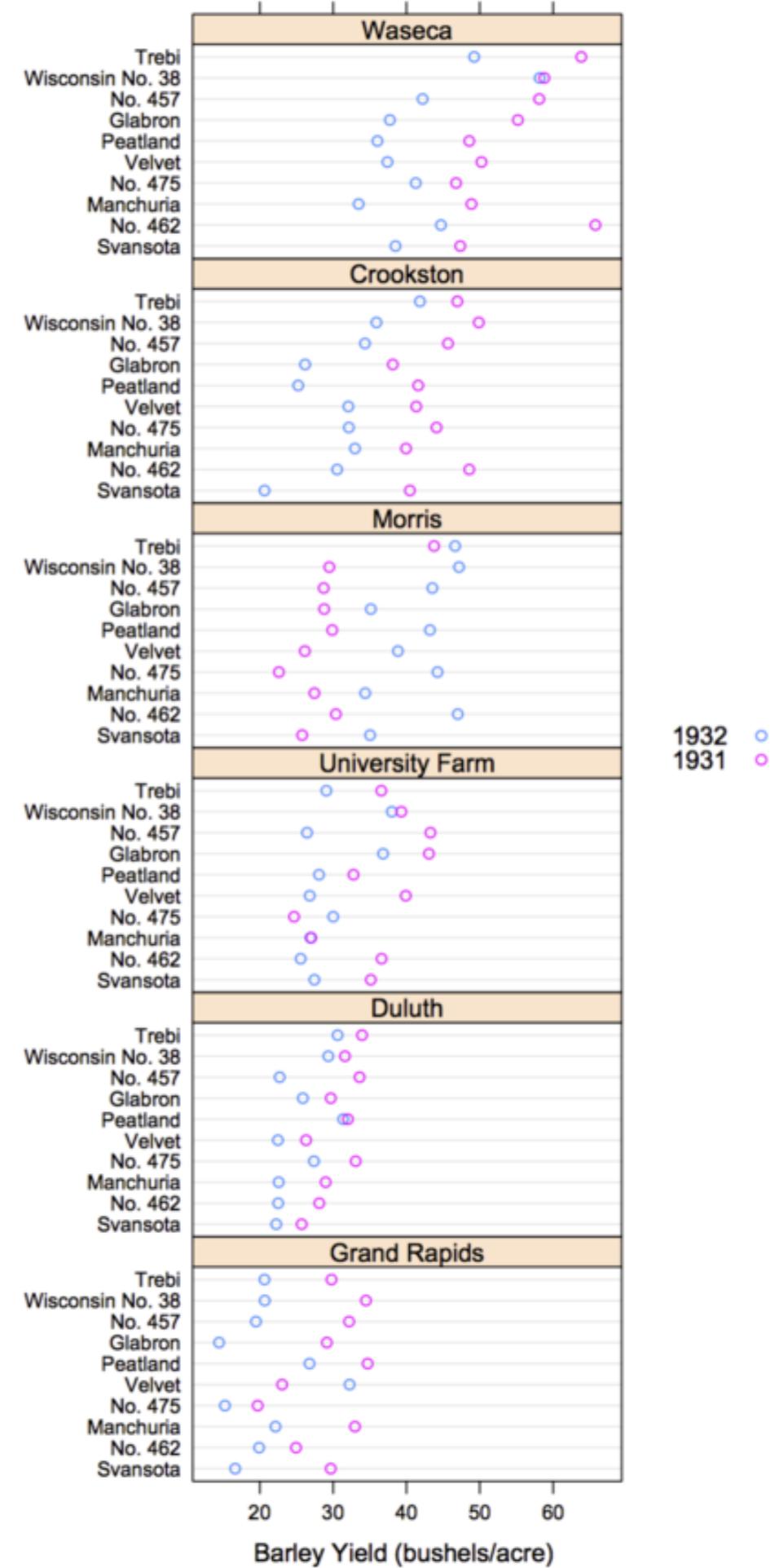
# Trellis Display

- Data is split into meaningful subsets, usually conditioning on variables of the dataset
- A visualization method is applied to each subset



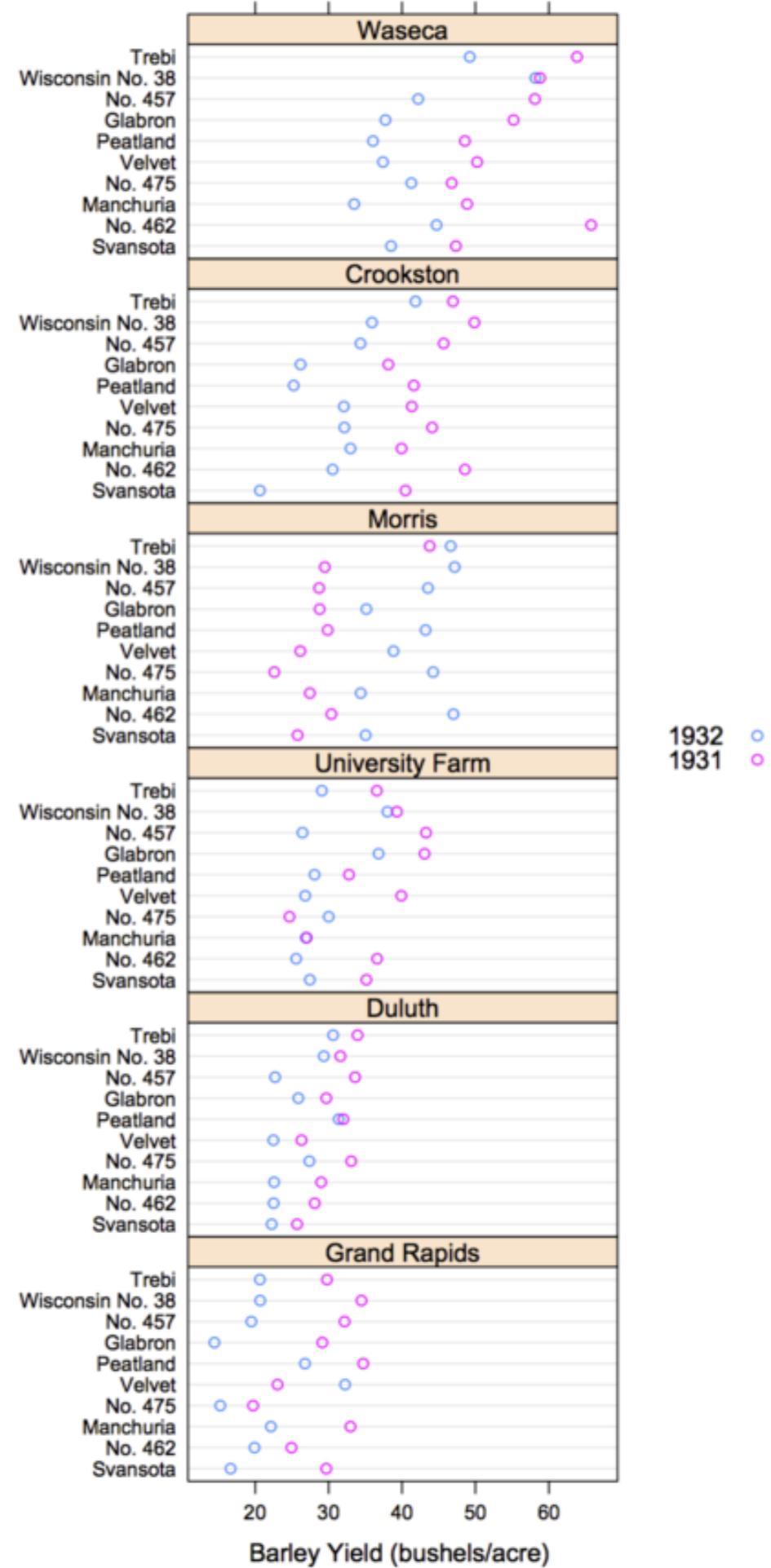
# Trellis Display

- Data is split into meaningful subsets, usually conditioning on variables of the dataset
- A visualization method is applied to each subset
- The image for each subset is called a "panel"



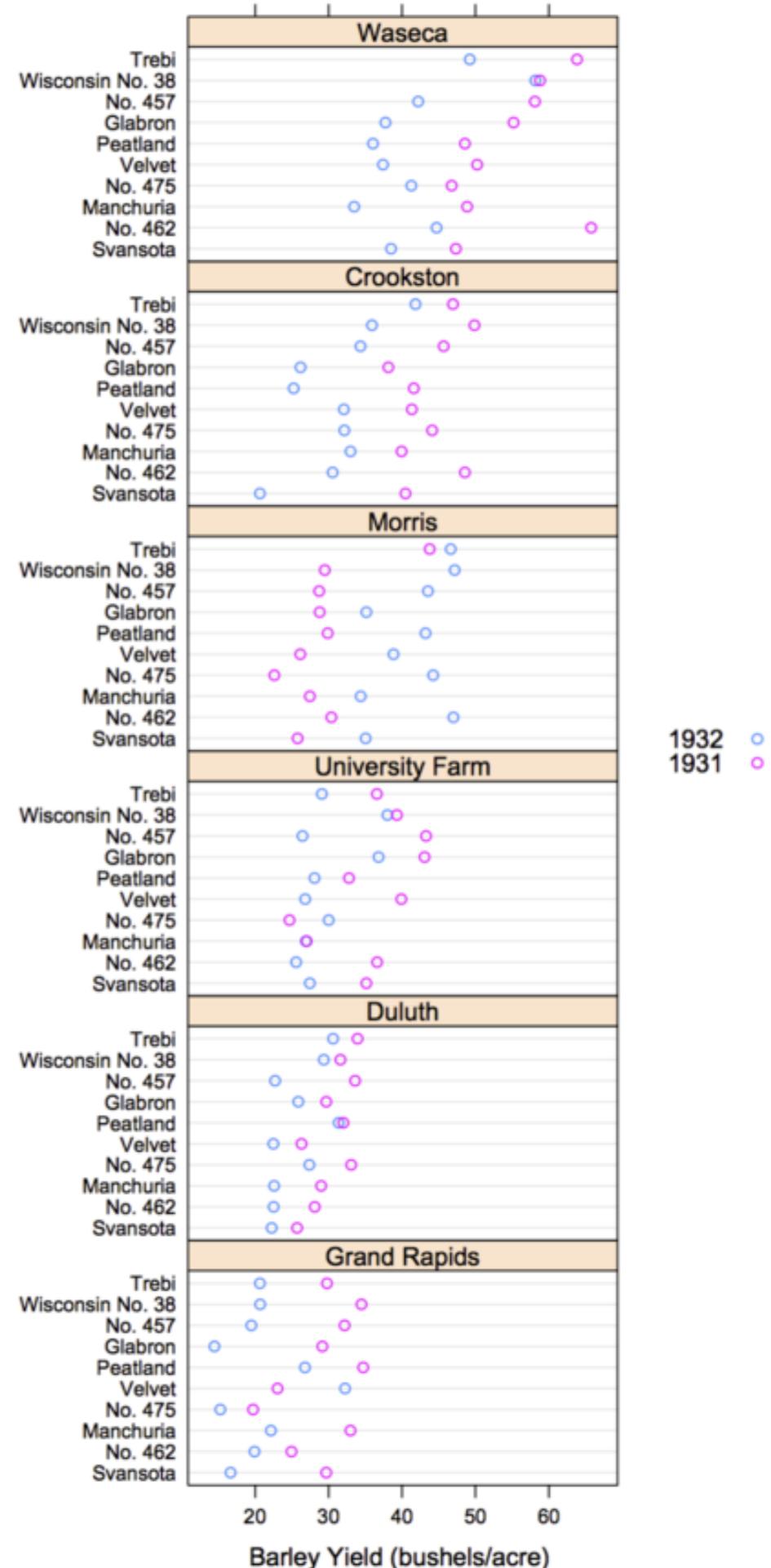
# Trellis Display

- Data is split into meaningful subsets, usually conditioning on variables of the dataset
- A visualization method is applied to each subset
- The image for each subset is called a "panel"
- Panels are arranged in an array of rows, columns, and pages, resembling a garden trellis



# Trellis Display

- Data is split into meaningful subsets, usually conditioning on variables of the dataset
- A visualization method is applied to each subset
- The image for each subset is called a "panel"
- Panels are arranged in an array of rows, columns, and pages, resembling a garden trellis
- `facet()`'ing in `ggplot2`



# Why Trellis is Effective

# Why Trellis is Effective

- Flexible to create
  - Data complexity / dimensionality / size can be handled by splitting the data into subsets
  - Complete freedom with what is plotted in every panel

# Why Trellis is Effective

- Flexible to create
  - Data complexity / dimensionality / size can be handled by splitting the data into subsets
  - Complete freedom with what is plotted in every panel
- Effective to consume
  - Understand one panel → Understand every panel
  - Scanning across panels elicits comparisons to reveal repetition and change, pattern and surprise

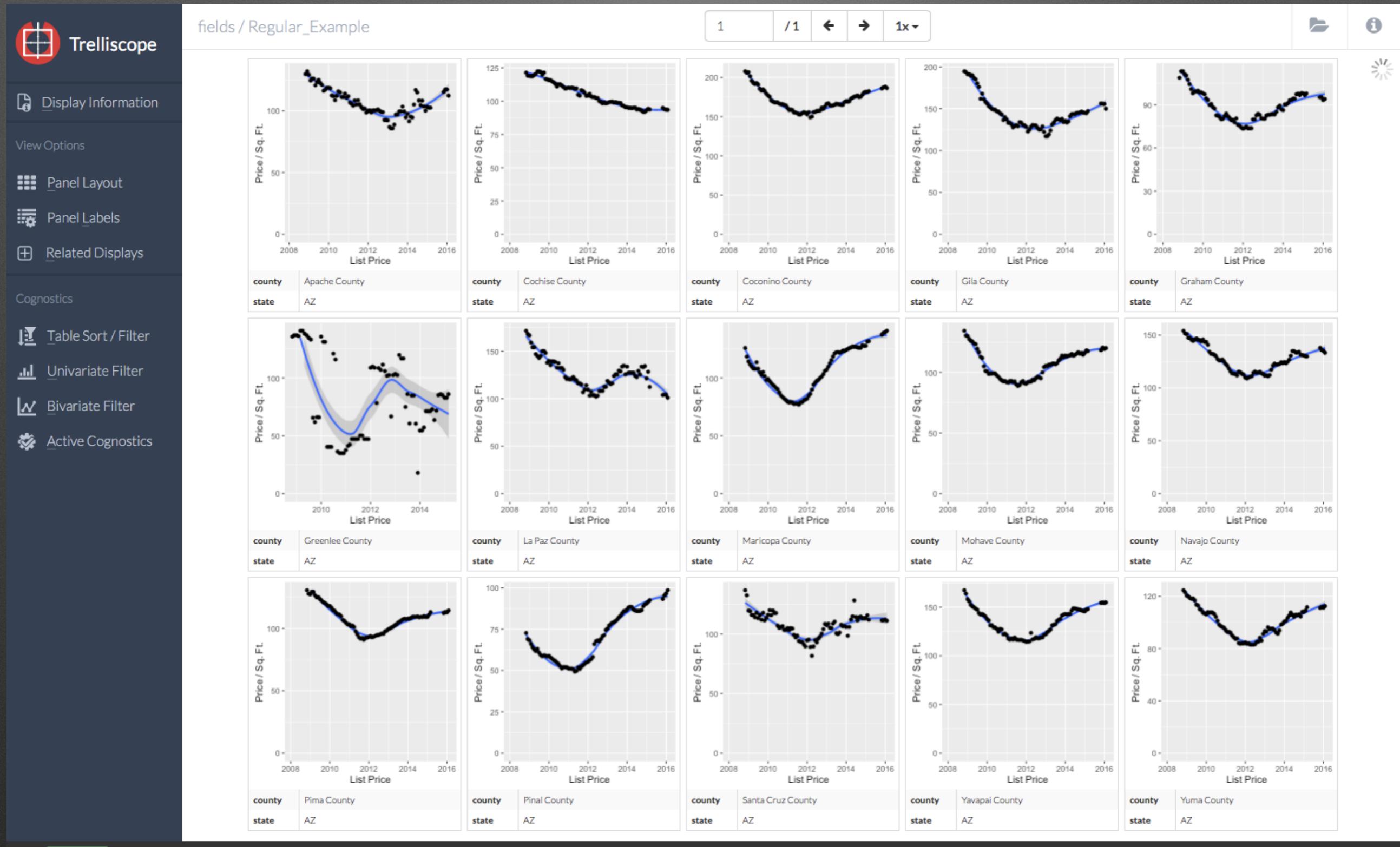
# Example / Data Description

- Monthly median home listing and number of units sold for 2,984 counties in the contiguous United States From 2008 to January 2016,
- Harvested from Quandl's Zillow

```
> housing %>% dplyr::group_by(county, state)
Source: local data frame [247,082 x 7]
Groups: county, state [2,984]
```

	fips	county	state	time	nSold	medListPriceSqft
*	<fctr>	<fctr>	<fctr>	<date>	<dbl>	<dbl>
1	06037	Los Angeles County	CA	2008-01-31	505900	NA
2	06037	Los Angeles County	CA	2008-02-29	497100	NA
3	06037	Los Angeles County	CA	2008-03-31	487300	NA
4	06037	Los Angeles County	CA	2008-04-30	476400	NA
5	06037	Los Angeles County	CA	2008-05-31	465900	NA
6	06037	Los Angeles County	CA	2008-06-30	456000	NA
7	06037	Los Angeles County	CA	2008-07-31	445700	NA
8	06037	Los Angeles County	CA	2008-08-31	435300	NA
9	06037	Los Angeles County	CA	2008-09-30	426700	NA
10	06037	Los Angeles County	CA	2008-10-31	419800	273.3073
# ...	with 247,072 more rows, and 1 more variables:	medSoldPriceSqft	<dbl>			

# Arizona



# Georgia



fields / Regular\_Example

/ 1 ← → 1x -



 Display Information



## Related Displays

Cognostics



 Active Cognostics

State:  state 

# Scaling Trellis

# Scaling Trellis

- Large data lends itself nicely to the idea of Trellis Display
  - Typically comprised of collections of smaller data from many subjects, sensors, locations, time periods, etc.
  - It is natural to break the data up based on these dimensions and make a plot for each subset

# Scaling Trellis

- Large data lends itself nicely to the idea of Trellis Display
  - Typically comprised of collections of smaller data from many subjects, sensors, locations, time periods, etc.
  - It is natural to break the data up based on these dimensions and make a plot for each subset
- Potentially thousands or millions of panels
  - Will never be able to (or want to) view all of them!

# Scaling Trellis with Cognostics

# Scaling Trellis with Cognostics

- Scaling Trellis:
  - Data are split into meaningful subsets, usually conditioning on variables of the dataset
  - A visualization method is applied to each subset
  - A set of cognostics that measure attributes of interest for each subset is computed
  - Panels are arranged in an array of rows, columns, and pages, resembling a garden trellis, with the arrangement being specified through interactions with the cognostics

# Scaling Trellis with Cognostics

# Scaling Trellis with Cognostics

- Scaling Trellis:
  - Data are split into meaningful subsets, usually conditioning on variables of the dataset
  - A visualization method is applied to each subset
  - A set of cognostics that measure attributes of interest for each subset is computed
  - Panels are arranged in an array of rows, columns, and pages, resembling a garden trellis, with the arrangement being specified through interactions with the cognostics

# Scaling Trellis with Cognostics

- Scaling Trellis:
  - Data are split into meaningful subsets, usually conditioning on variables of the dataset
  - A visualization method is applied to each subset
  - A set of cognostics that measure attributes of interest for each subset is computed
  - Panels are arranged in an array of rows, columns, and pages, resembling a garden trellis, with the arrangement being specified through interactions with the cognostics
- Can be achieved with the Trelliscope package

# Trelliscope Demo

# Calculated Cognostics

```
advanced_cog <- function(x) {  
  zillow_string <- gsub(" ", "-", do.call(paste, getSplitVars(x)))  
  
  model <- loess(  
    medListPriceSqft ~ as.numeric(time),  
    data = subset(x, !is.na(medListPriceSqft))  
  )  
  residuals <- model$residuals  
  list(  
    res_std_err = cog(model$s, desc = "residual standard error"),  
    enp = cog(model$enp, desc = "effective number of parameters"),  
    mean_list = cogMean(x$medListPriceSqft),  
    n_obs_list = cog(  
      length(which(!is.na(x$medListPriceSqft))),  
      desc = "number of non-NA list prices"  
    ),  
    zillow_href = cogHref(  
      sprintf("http://www.zillow.com/homes/%s_rb/", zillow_string),  
      desc = "zillow link"  
    )  
  )  
}
```

# Automatic Cognostics

# Automatic Cognostics

- Cumbersome to manually specify many cognostics for a Trelliscope display

# Automatic Cognostics

- Cumbersome to manually specify many cognostics for a Trelliscope display
- Should be able to automatically compute cognostics based on the context of what is being plotted
  - Help foster a scalable Trellis system

# Automatic Cognostics

- Cumbersome to manually specify many cognostics for a Trelliscope display
- Should be able to automatically compute cognostics based on the context of what is being plotted
  - Help foster a scalable Trellis system
- Analyze the plot objects and choose "best" cognostics based on the plot specification
  - `ggplot2`
  - `rbokeh`
  - `plotly`

For example...

# For example...

- For scatterplot layers:
  - Number of observations
  - Number of missing values
  - Summary statistics of x and y-axis variables

# For example...

- For scatterplot layers:
  - Number of observations
  - Number of missing values
  - Summary statistics of x and y-axis variables
- For statistical layers (such as geom\_smooth)
  - RMSE of fit
  - Number of outliers

# For example...

- For scatterplot layers:
  - Number of observations
  - Number of missing values
  - Summary statistics of x and y-axis variables
- For statistical layers (such as geom\_smooth)
  - RMSE of fit
  - Number of outliers
- Etc. (much research going on / to be done here...)

# Future work

# Future work

- Continually adding more cognostics to be created for each plotting layer

# Future work

- Continually adding more cagnostics to be created for each plotting layer
- Implement
  - Fully integrate into trelliscope
  - Parse ggplot2, rbokeh, and plotly objects

# Questions?

[www.tessera.io](http://www.tessera.io)

[github.com/tesseradata](https://github.com/tesseradata)

[github.com/schloerke](https://github.com/schloerke)