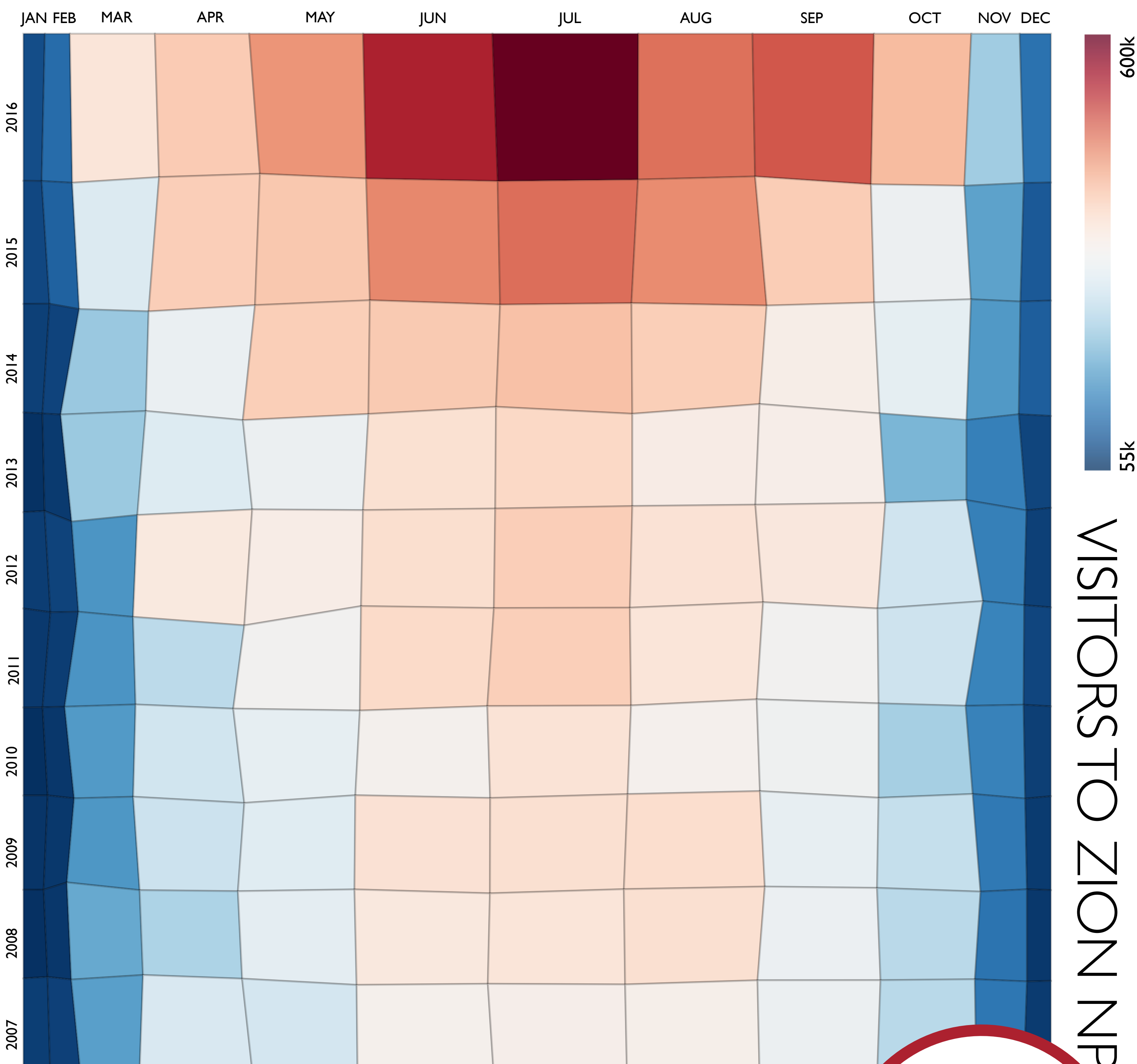
DESIGN & ANALYSIS OF TABLE CARTOGRAMS

Simultaneous Multipurpose Tabular Area-Encoding Displays

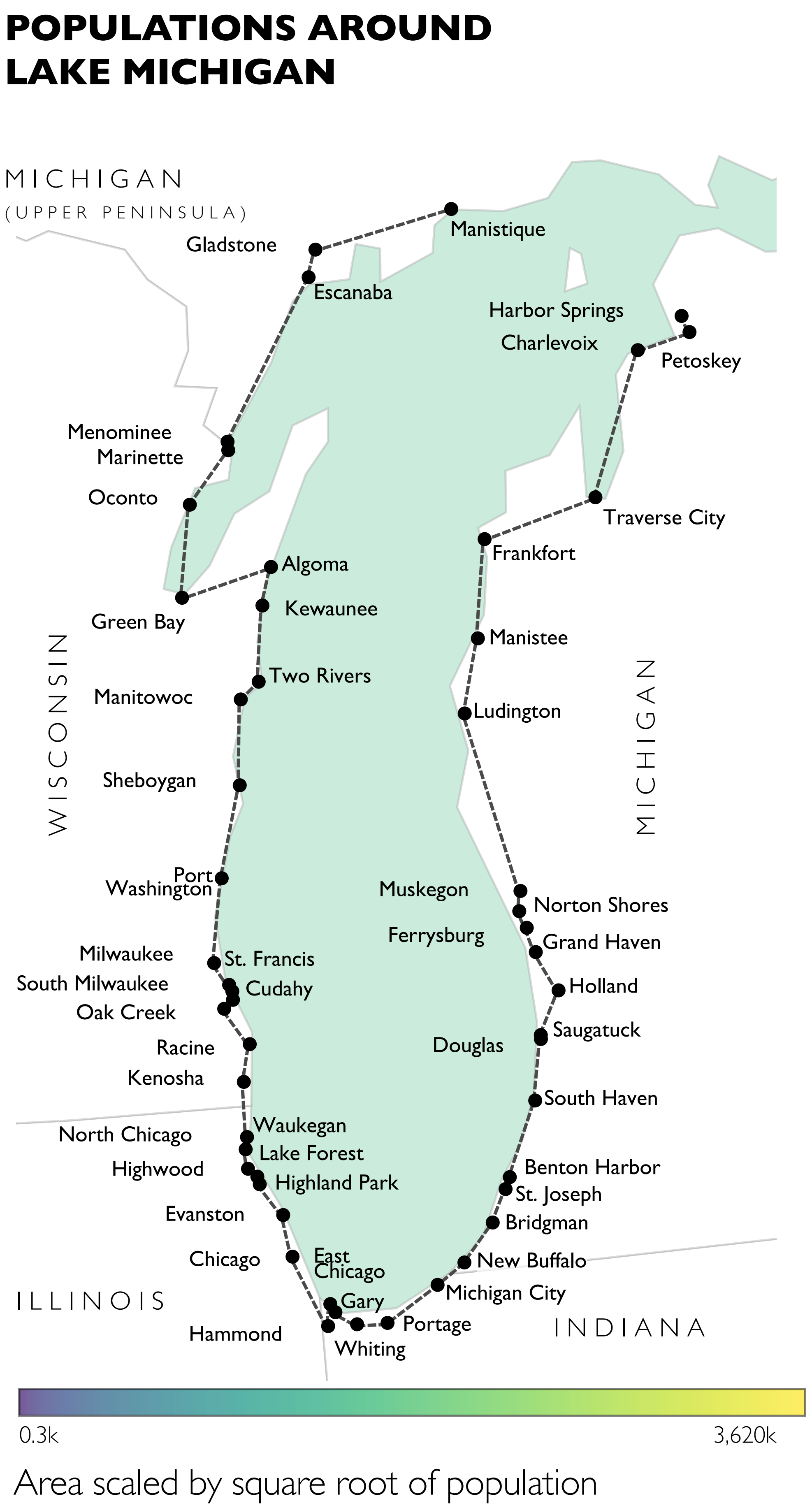


DEFENSE EDITION

A Master's Thesis Zine

ANDREW MCNUTT

What is a table cartogram?



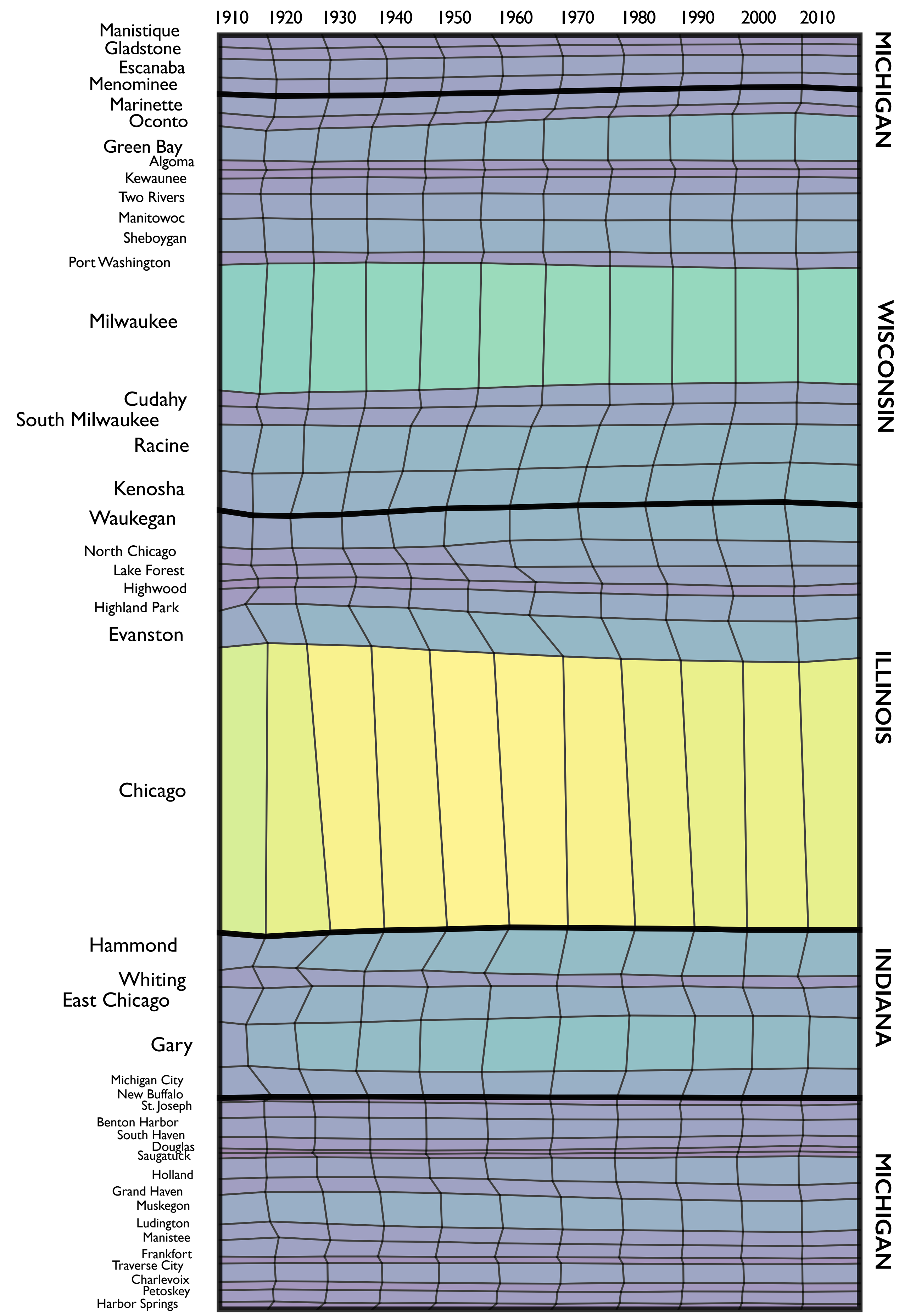


Table cartograms are a type of data visualization that represent tables of numbers in a graphically intriguing and theoretically rich way.

They represent a table of numbers as a grid of quadrilateral cells, whose areas are changed to fit the data. **Like a heatmap that has been area-ed rather than colored**

They have two defining properties:

1. They possesses an **Accurate Embedding** of Data as area.

2. They have a **Planar Grid-like Topology** that is constrained to a rectangle

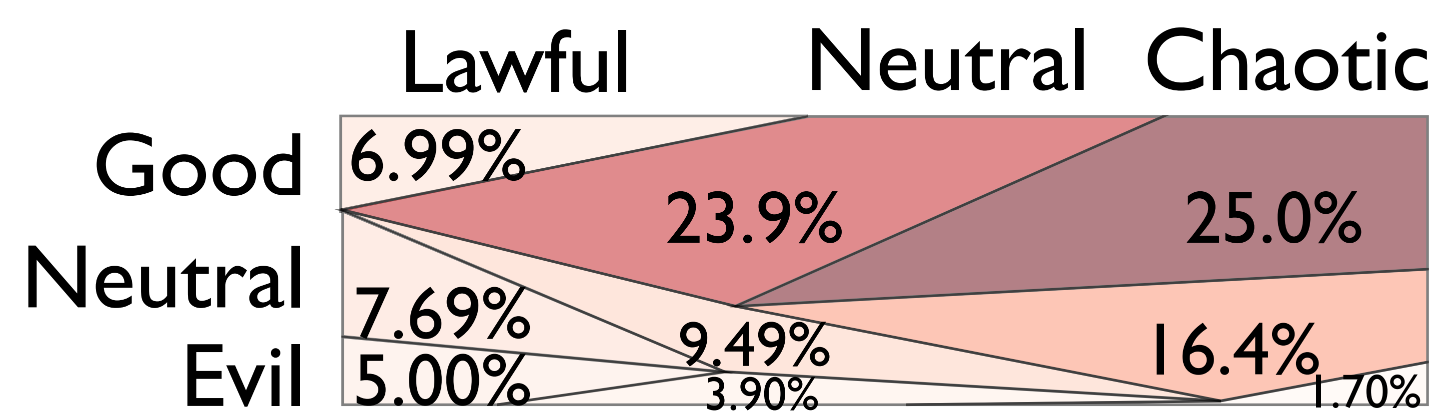
Moving violations detected by red light cameras in Chicago 2016

Notice how violations drop off exactly the moment that public schools go on summer break

Color is relative to full year, size is relative to month

Table cartograms are great at showing **part-to-whole relationships**, especially when the input data has meaningfully ordered rows and columns.

This makes them good for calendar displays and as enhancements to tables that already have a canonical ordering. In the year/month calendar to the left we are clearly able to make ordinal comparisons (the weekends have few tickets,and some particular weeks have very few). In the DnD alignment table below we can make easy part-to-whole comparisons without needing to tediously examine each individual pair of cells.

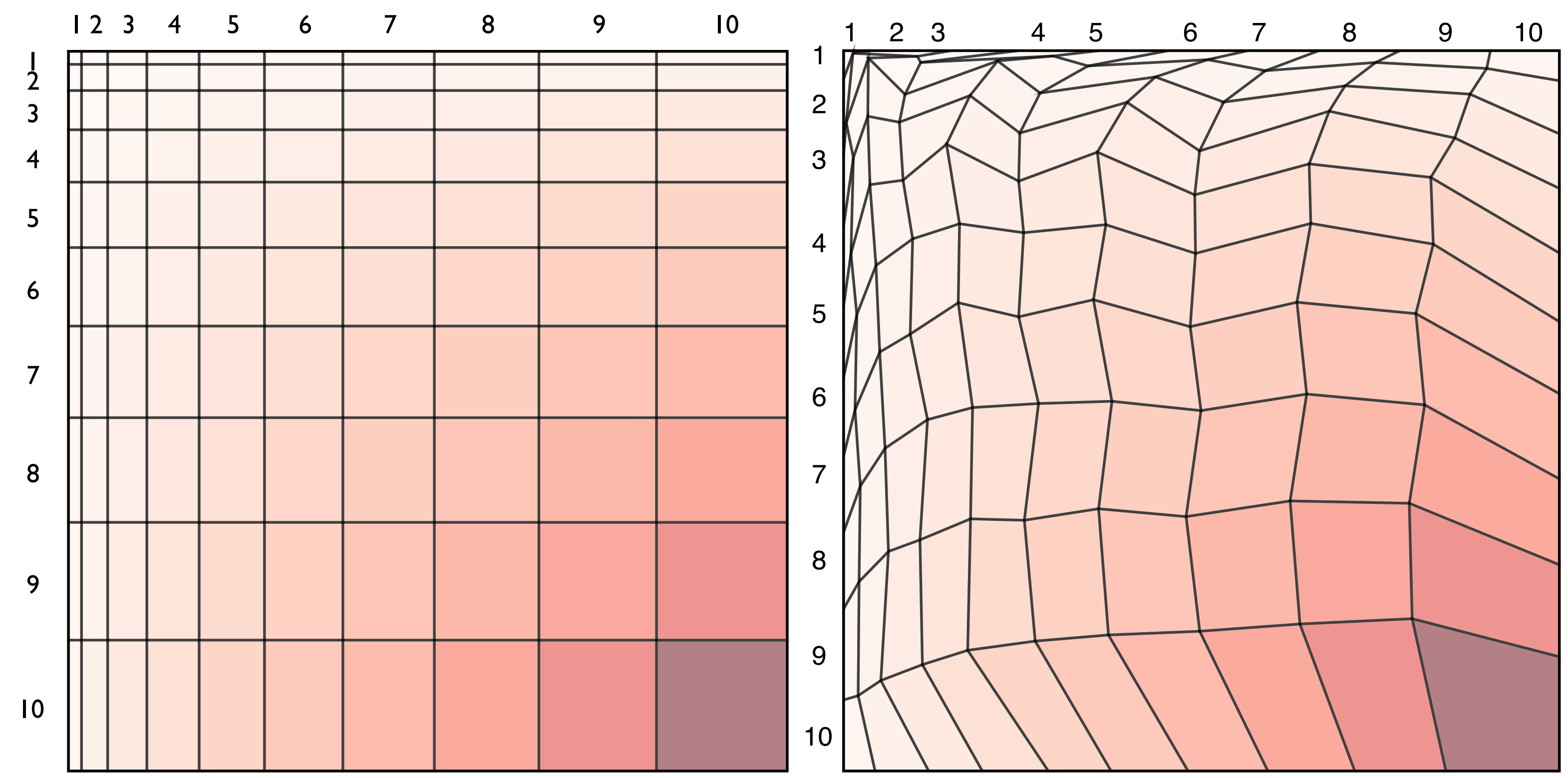


	Lawful	Neutral	Chaotic
Good	6.99%	23.9%	25.0%
Neutral	7.69%	9.49%	16.4%
Evil	5.00%	3.90%	1.70%

Popularity of DnD alignments, src informal online poll

Would you agree that a **good visualization will reflect changes to its data in a way that makes sense to a human**?

This is the basic principle behind Algebraic Vis Design (AVD). We use this lens as a mechanism to theoretically motivate our understanding of the table cartogram's properties. Taking this approach (rather than conducting user studies) allows us to economically scan the space of unknown unknowns about table cartograms, reducing them to known unknowns (which users studies are great at addressing).

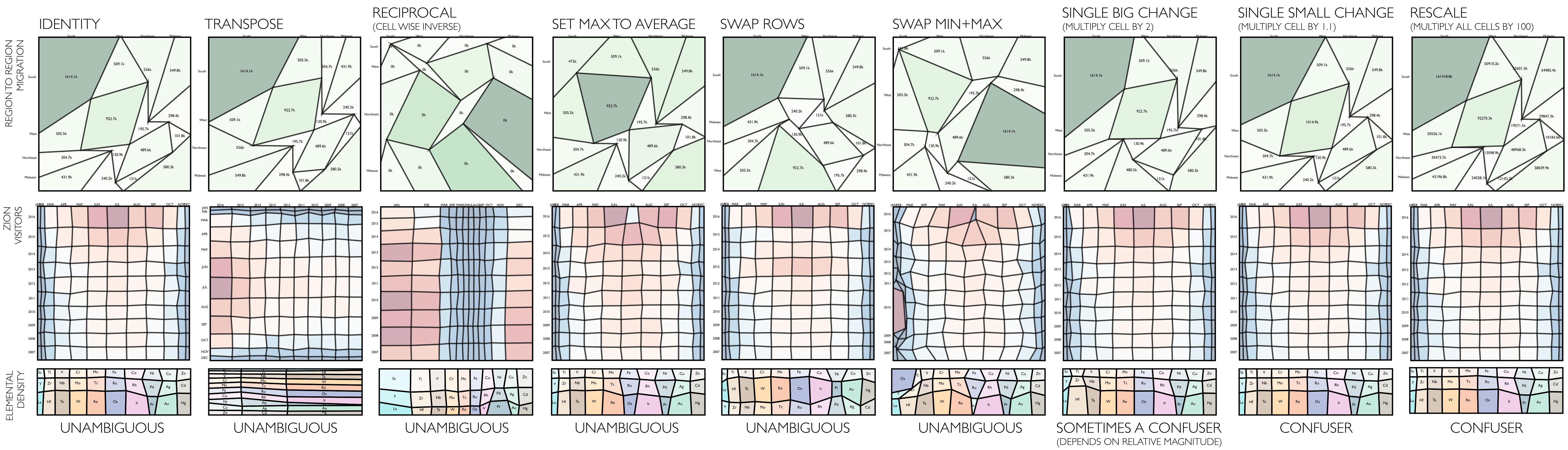


These two multiplication tables shown as table cartograms are equally right! A hallucinator!! This type of property makes it difficult to use table cartograms in visual analytics contexts where repeatable and consistent observations are important for gaining an understanding of the data.

Data change \ Visual change	*Obvious* ω	*None/illegible* ω
Insignificant α	**Hallucinator**	Invariant representation
Significant α	Unambiguous Visualization	**Confuser**

Failure mode *Success mode*

Here we show a summary of the basic failure modes in Algebraic Vis Design A significant α might change something important about the data, while an insignificant α might change something about the representation (like changing the order of a data in a scatterplot).



In the above chart we consider three tables of data from across this zine and apply a variety of data transforms (αs). Some of these transforms remain legible across transformation. While others do not! Here are the important takeaways from the analysis:

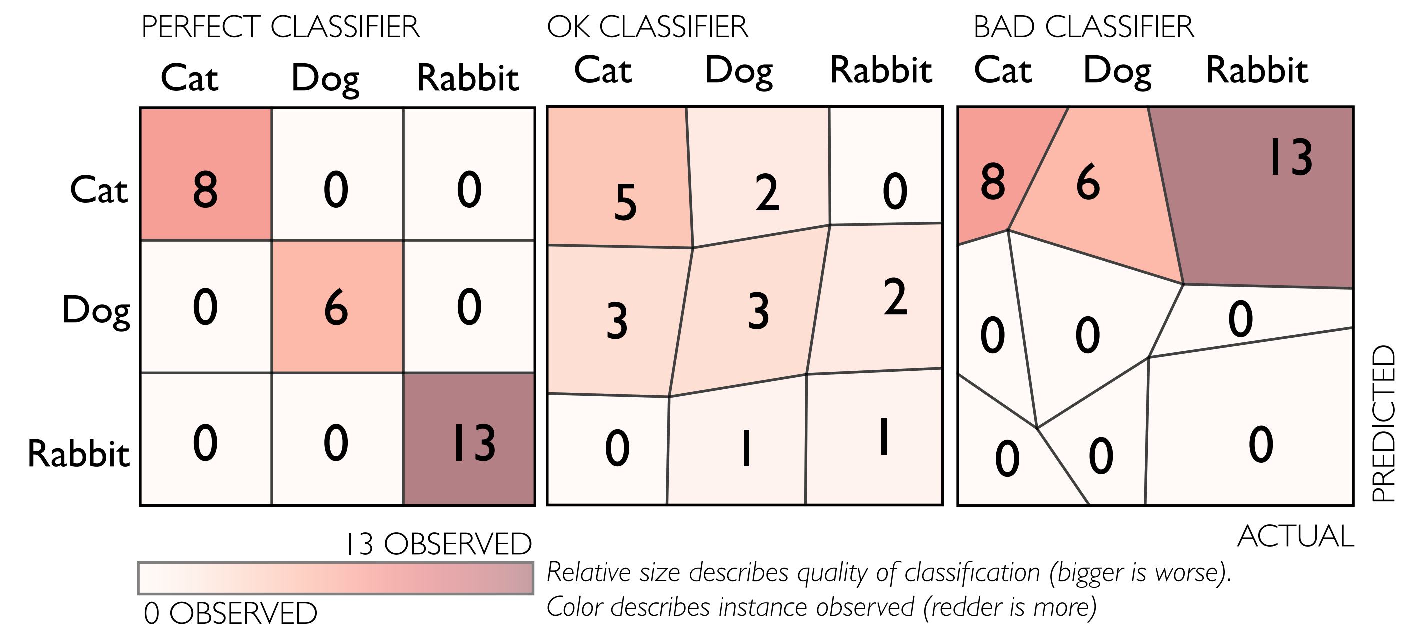
Changes to the scale are invisible. Table cartograms should not be used for presentations where scale matters

Reciprocal changes are visible. This shows that size order is visually maintained in table cartograms, which suggests that ordinal measurements (comparisons between discrete but ordered entities) are fair game. This confirms some of the part-to-whole analyses we've seen!

Small changes are invisible. This suggests that (when individual cells matter) data sets should be selected in such a way that they stay visible across transforms. A good rule of thumb is that tables should be selected that are no more than three orders of magnitude in range and contain no more than a few hundred cells.

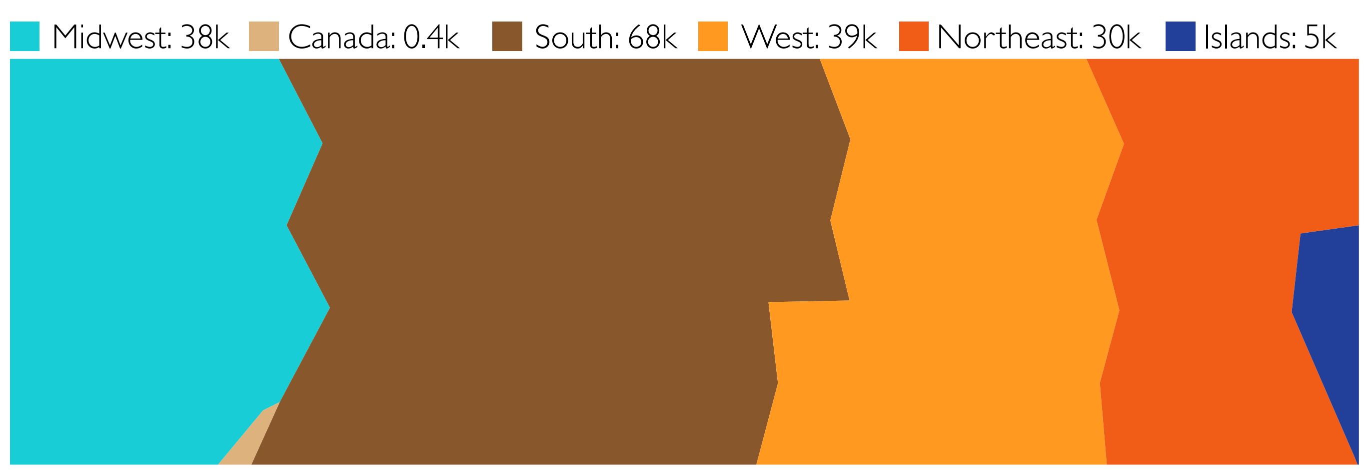
In addition to enhancing previous visual encodings we can make new ones!

Table Confusiongram



Characteristic:
Re-encoding of confusion matrices
Related to:
Confusion matrices, corrgrams, table corrgrams
Tasks it's good at:
Identify outliers, both on a row-whole and cell to whole level

Table Polygrams



The number of times an airplane struck a bird since 1991 by origin. Data from the FAA bird strikes database.

Characteristic:
Tabular encoding and layout of non-tabular data
Related to:
Pie chart, stacked bar chart, waffle plot
Tasks it's good at:
Part-to-whole and part-to-part for non-tabular data

or at least evocatively name some unusual chart types

Table Mosaicgram

Characteristic:

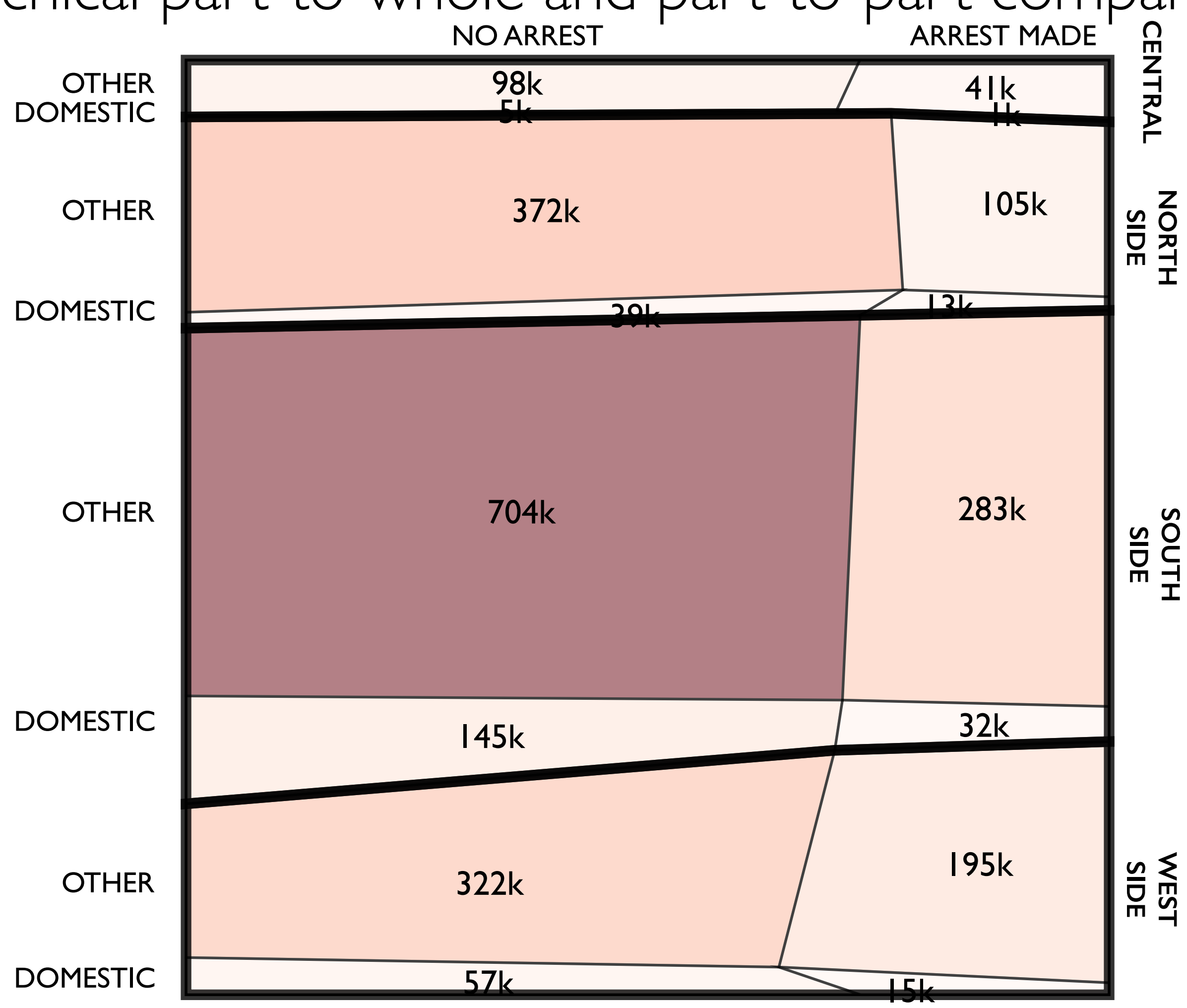
multiple hierarchical categorical variables

Related to:

Mosaic, treemap, pivot table, sunburst

Tasks it's good at:

Hierarchical part-to-whole and part-to-part comparison



CRIME IN CHICAGO SINCE 2006-2012

bigger/redder is more crime

Table corrgram

Characteristic:

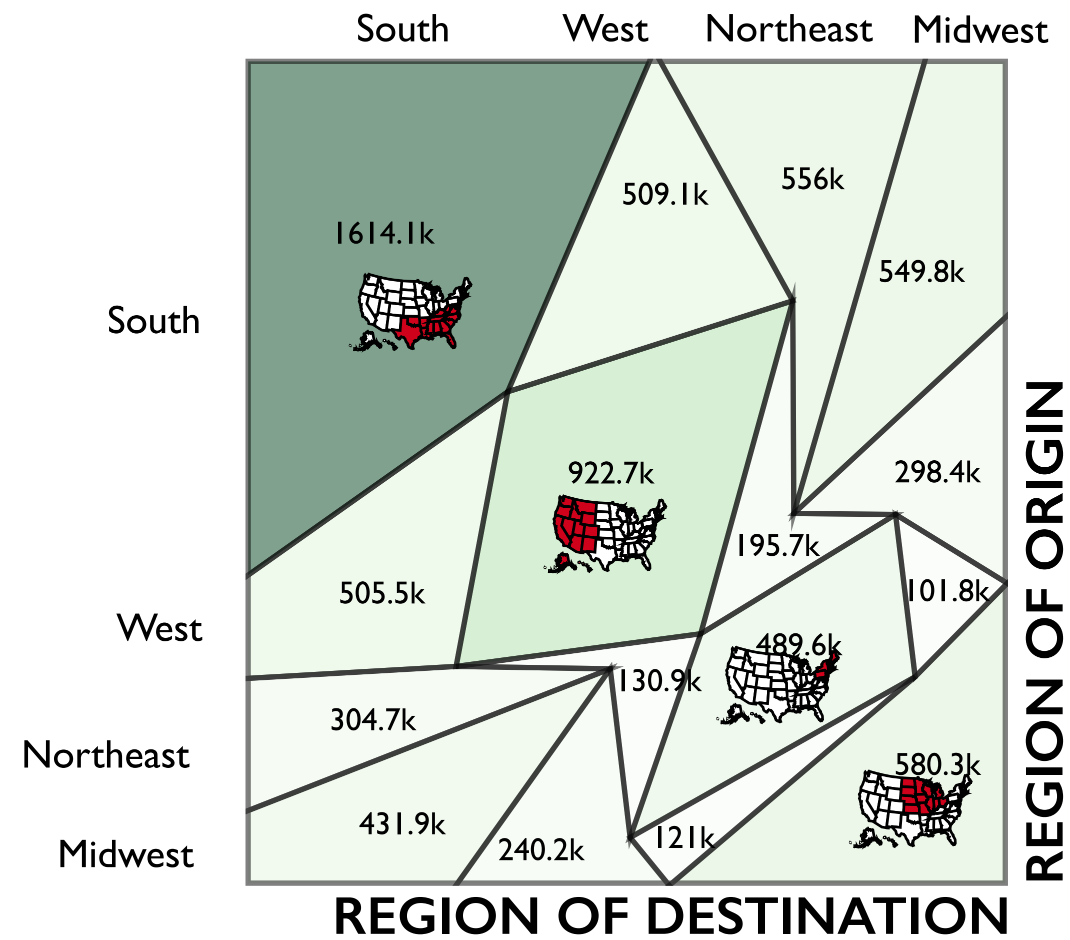
One categorical variable along rows and columns

Related to:

Confusion matrix, adjacency matrix, corrgrams

Tasks it's good at:

Summary, lookup, observe outlier



American Region to Region Migration in 2016 (Via the census)

Table Formogram

Characteristic:

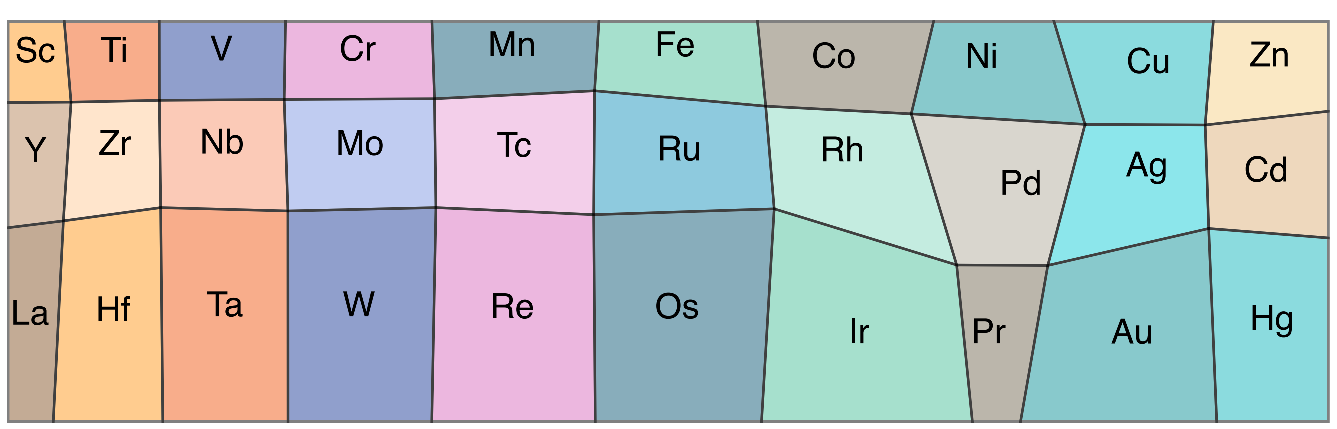
Matching an existing canonical table

Related to:

Shaded Matrix, the original table

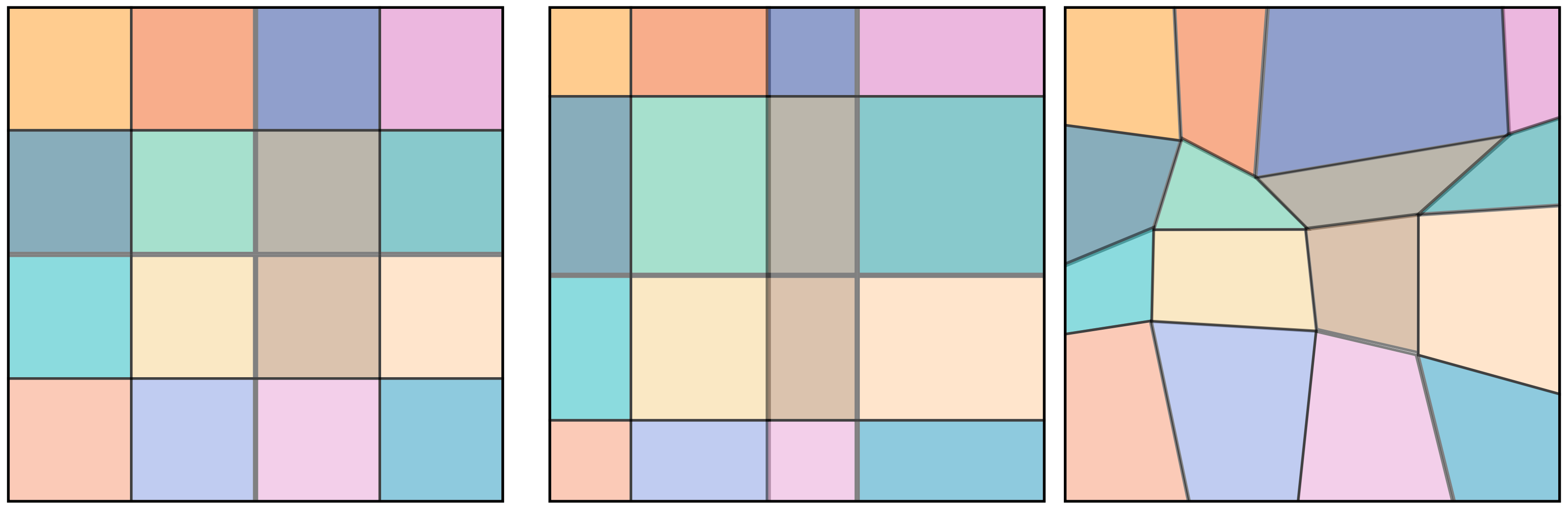
Tasks it's good at:

Lookup, Detect Change, Observe Outlier



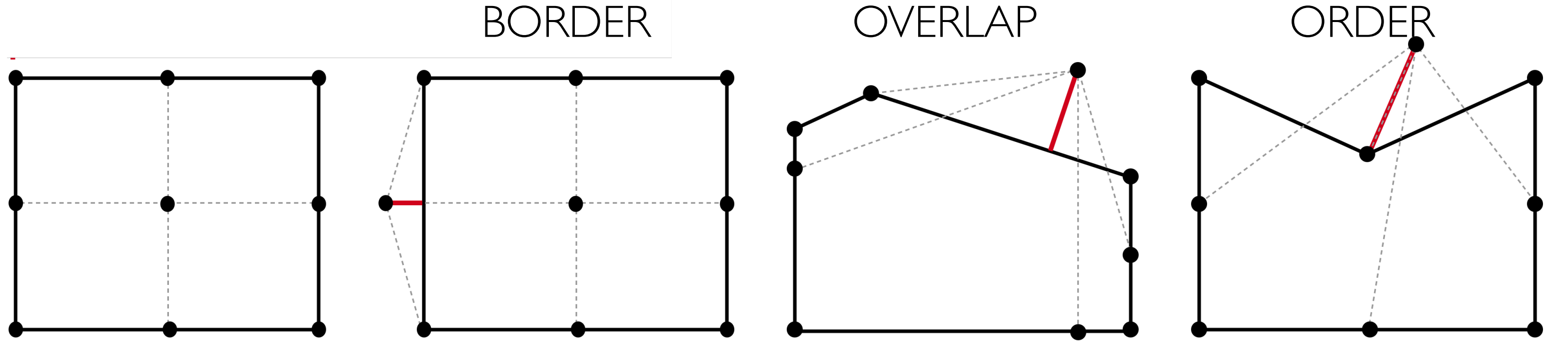
Groups 3-12 of the periodic table valued by elemental density

SETUP



1. Construct a table of (non-zero) numbers
2. Make an educated guess about the final layout
3. Run gradient descent to get the final layout

Our gradient descent minimizes **the relative error** between expected area as well as the penalties from the following constraints

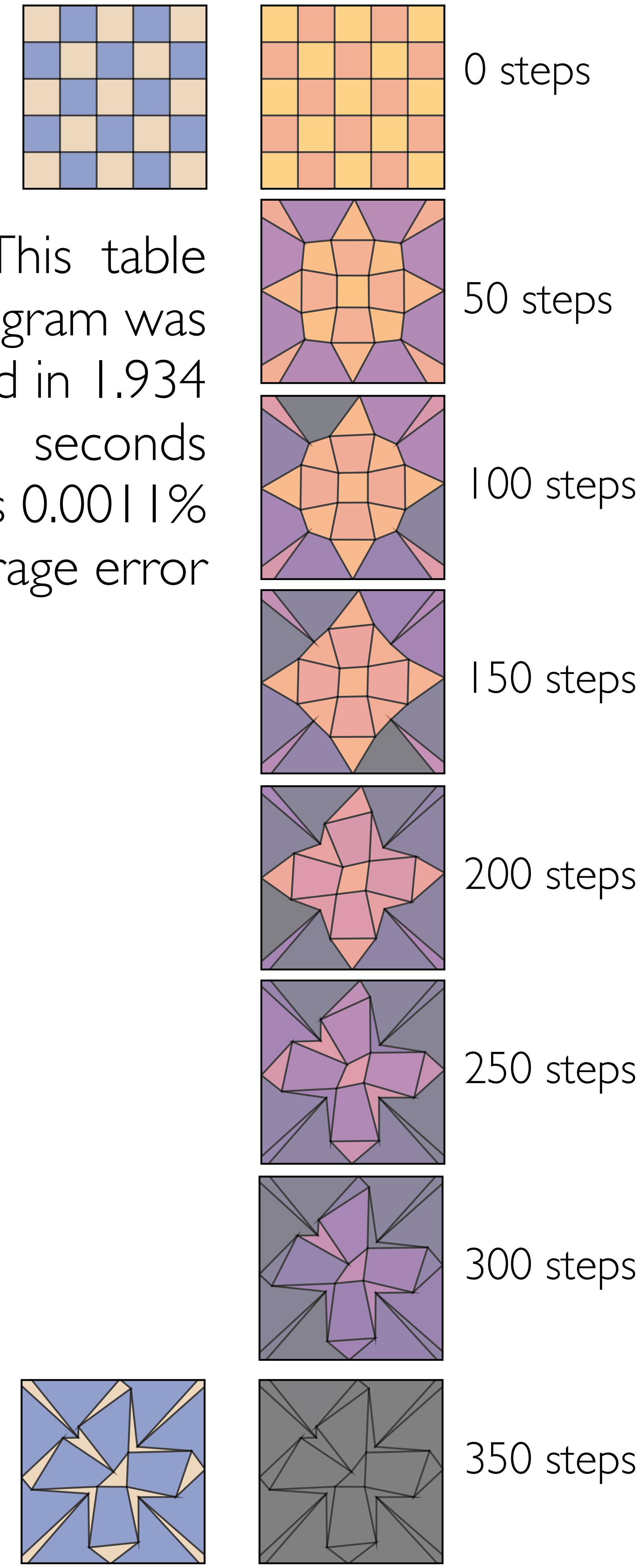


No penalty | Point is outside of border, penalize distance outside | Point is outside of containing rect, penalize distance | Point is out of order penalize distance outside of order

WATCH IT GO

Color by value purple is 5, brown is 1

Color by error Yellow is high error/ gray is no error



This table cartogram was computed in 1.934 seconds it has 0.0011% average error

How do we actually make a these things?

UNIVERSITY OF CHICAGO

Department of Computer Science

UC