Ohio Election Analysis
Verification of Nov 2020 Results:
Robert Lee's Analysis of Ohio Data on GitHub

efg, 2022-05-01

Outline

- Reproducible Research
- Robert Lee's GitHub Page, Limitations
- Jupyter and RStudio Notebooks
    1. Download Ohio voter data and "jsonify"
    2. Single County analysis: Cuyahoga and Franklin Counties
        - Registered Voters, Active Voters, and Nov 2020 Voted Counts
        - Fraction voted (turnout)
        - "Normalized" fraction voted ("key")
    3. All Ohio Counties: Normalized voter turnout "keys" for Nov 2020 election
    4. Generate Ohio statewide normalized turnout "key"
    5. Polynomial curve fitting of statewide "key"
    6. Compare county and state normalized turnout keys using "heatmap"
    7. Compare correlations of number of registered voters and number of Nov 2020 voters
    8. Predictions: Predict Hamilton County turnout using Ohio "key"
- Technical Questions

Reproducible Research

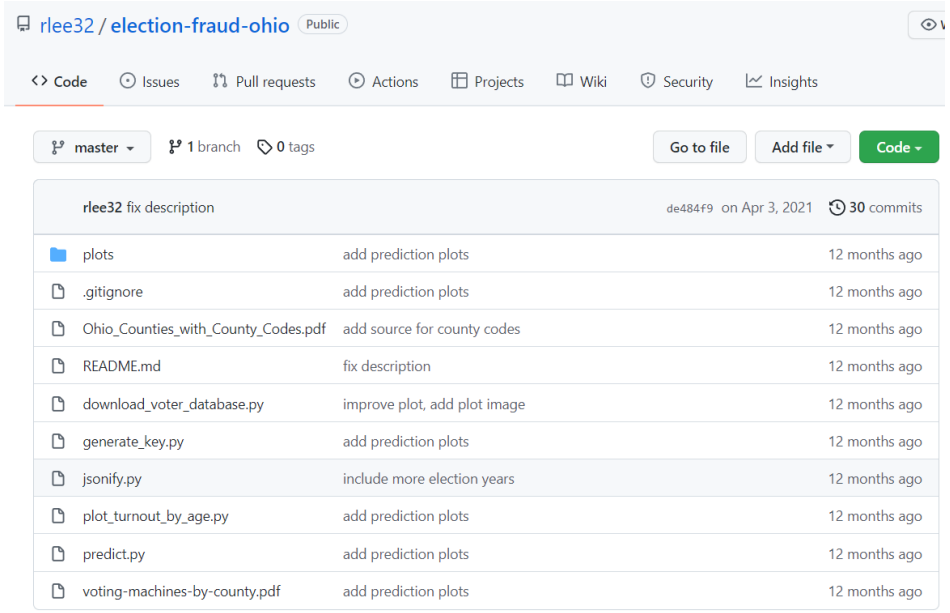
- Criticism of election process should not be another “black box.”

Any proper analysis should …

- Document data provenance starting with original government sources.
- Provide documented, step-by-step analysis, which can be reviewed and audited.

Robert Lee's Github Page

- The page compares the normalized turnout "keys" for several states: https://github.com/rlee32/election-fraud-national
- This page show details with python code for Ohio analysis: https://github.com/rlee32/election-fraud-ohio



Robert Lee's Github Page

- I converted Lee's Python scripts to be part of Jupyter notebooks to show documentation, code, results, graphics and comments in one place.
- Modifications to some of Lee's approaches and assumptions:
  - Used different file download process since Lee's approach didn't work for me.
  - Added curves for "Active" voters to Lee's plots of Registered and Nov 2020 Voters. Not necessary to compute key but adds additional insights.
  - Added county names to numbers. Clarified titles and axes labels.
  - Jupyter notebooks reject ages > 105 years instead of Lee's 150 year old limit.
  - Added legend to identify lines. Added grids to all plots.
- Added polynomial fits of statewide "normalized turnout" to reproduce Ohio "key".

Local directory: Election-Integrity/Ohio/rlee32/Analysis

Robert Lee's Github Page

Robert Lee's script	Corresponding Jupyter Notebooks
download_voter_database.py	Ohio-Analysis-1-setup.ipynb → .html
jsonify.py	
plot_turnout_by_age.py	Ohio-Analysis-2-Single-County.ipynb Ohio-Analysis-3-All-Counties.ipynb
generate_key.py	Ohio-Analysis-4-Generate-Key.ipynb
predict.py	Ohio-Analysis-5-Predictions-vs-Acutal-Votes.ipynb

Local directory: Election-Integrity/Ohio/rlee32/Analysis

Limitations

- I’m using latest Ohio data from 2022-03-25 since older Ohio files from the Nov. 2020 election are not readily available.
- Many Ohio voter records have been added, deleted or changed since the Nov. 2020 election.
  - Analysis here shows 7,431,918 Ohio registered voters with 5,656,585 voting in Nov. 2020.
  - Official Ohio statewide results for the Nov. 2020 election show 8,073,929 registered voters with 5,974,121 casting ballots.

Download Ohio voter data and “jsonify”

Notebook: **Ohio-Analysis-1-setup.html**

- State of Ohio online databases: https://www.ohiosos.gov/secretary-office/online-databases/
- Ohio County voter file download: https://www6.ohiosos.gov/ords/f?p=111:1
- Sample link to download county file 1, Adams County: https://www6.ohiosos.gov/ords/f?p=VOTERFTP:DOWNLOAD::FILE:NO:2:P2_PRODUCT_NUMBER:1
- Notebook downloaded 88 files 1.csv, 2.csv, …, 88.csv to local directory: Election-Integrity/Ohio/rlee32/Analysis/voter_database
- Data was captured from Ohio SOS site on 2022-03-25.

Adapted from Lee’s scripts: download_voter_database.py and jsonify.py

Download Ohio voter data and “jsonify”

Notebook: **Ohio-Analysis-1-setup.html**

- Ohio Counties with county number 1 .. 88
  https://tax.ohio.gov/static/excise/Ohio%20County%20Listing.pdf
- Map of Ohio Counties with county number
  https://jfs.ohio.gov/PerformanceCenter/FastFacts/Ohio_Counties_with_County_Codes.pdf
- Raw data looks like this for Adams County (1.csv; 17,046 records):

```
"SOS_VOTERID","COUNTY_NUMBER","COUNTY_ID","LAST_NAME","FIRST_NAME","MIDDLE_NAME","SUFFIX","DATE_OF_BIRTH","REGISTRATION_DATE","VOTER_STATUS","PARTY_A  ...
"OH0021963657","01","31838","KEATON","HAROLD","G","","1955-01-27","2012-09-12","ACTIVE","","14608 SR 41","","WEST UNION","OH","45693","","","","","","",
"OH0016296576","01","24925","MCCANN","NATASHA","DANIELLE","","1984-07-05","2004-09-29","ACTIVE","","3070 LOUISVILLE RD","","PEEBLES","OH","45660","","
"OH0021185869","01","30011","TENER","JOSHUA","L","","1989-06-25","2010-07-01","ACTIVE","","200 BEASLEY FORK RD","","WEST UNION","OH","45693","","","",""
"OH0016296517","01","2185","SIMMERS","WILMA","LEE","","1948-12-10","1969-09-29","ACTIVE","","942 COSBY ST","","WEST UNION","OH","45693","","","","","","
"OH0021525957","01","30996","WINTEROD","LARRY","L","JR","1972-11-18","2011-09-27","ACTIVE","","185 N MAIN ST","APT 10","PEEBLES","OH","45660","","","","
...
```

Download Ohio voter data and “jsonify”

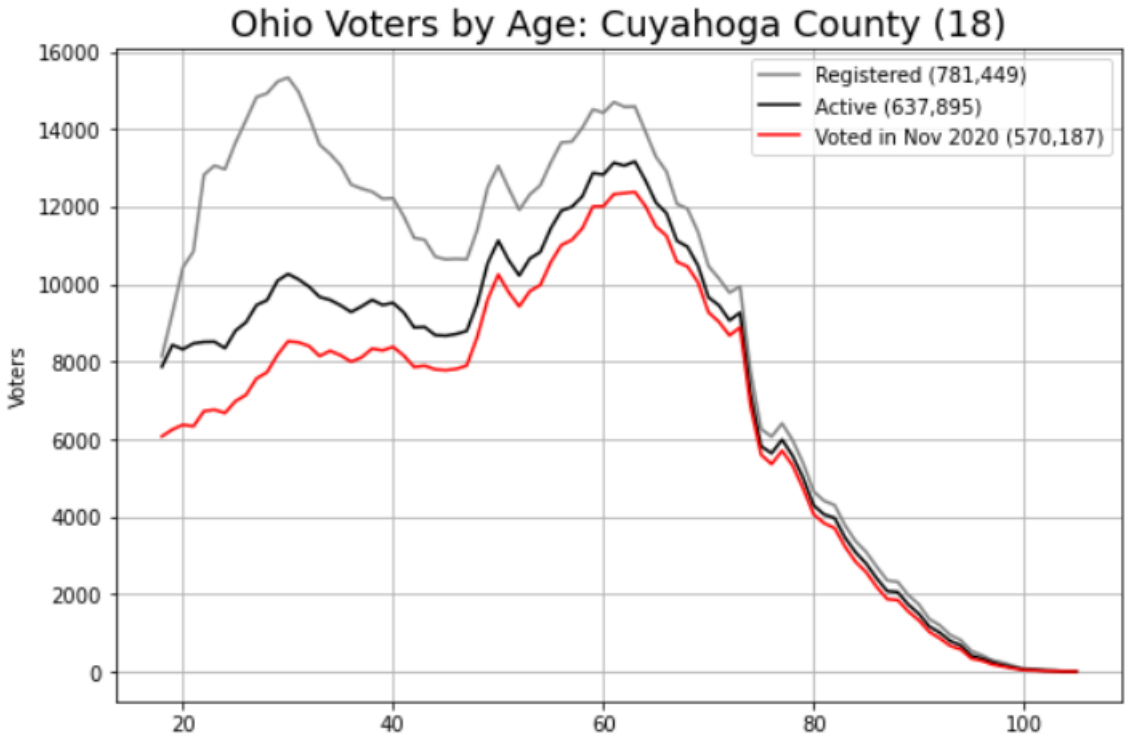
Notebook: **Ohio-Analysis-1-setup.html**

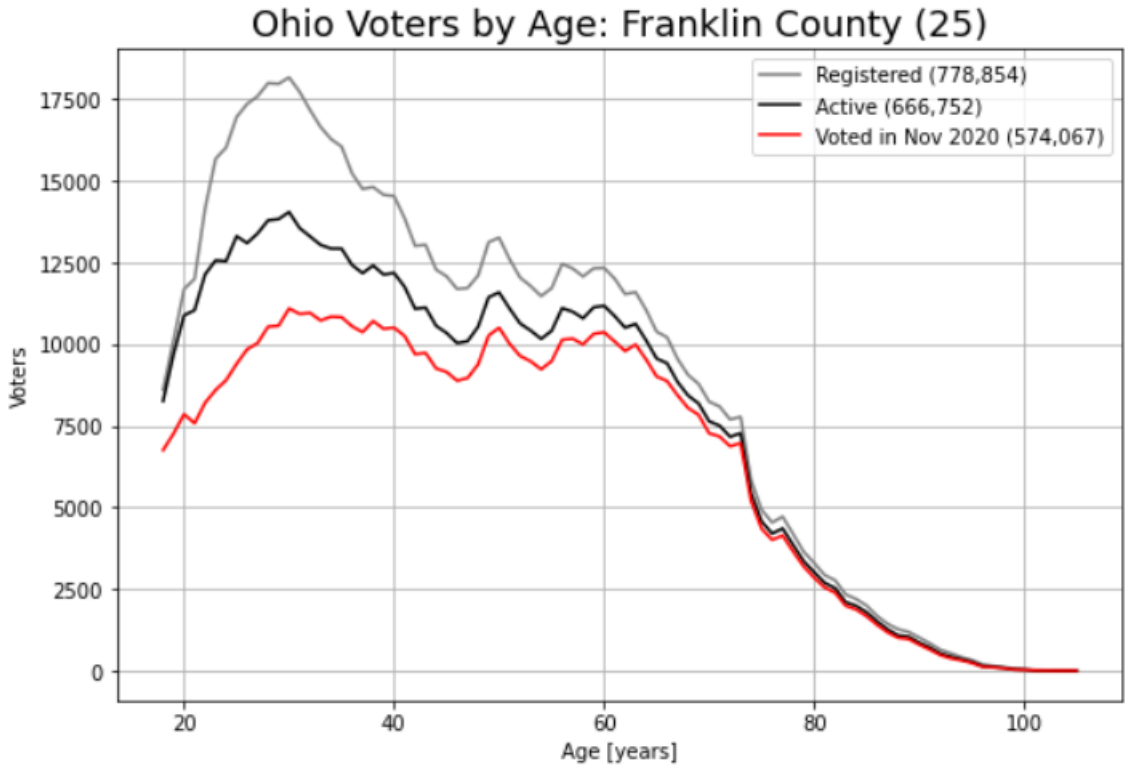
- Analysis could have been performed using .csv files with Python “pandas” data analysis library, but Lee chose to convert to a “JSON” file format instead. I followed Lee’s script using JSON.
- Notebook filtered data and converted files *n*.csv to *n*.json in local directory: Ohio/rlee32/Analysis/jsonified
- Raw JSON file (1.json; 187,497 records) looks like this:


```
[
  {
    "date_of_birth": "1955-01-27",
    "registration_date": "2012-09-12",
    "voter_status": "ACTIVE",
    "general_2000": "",
    "general_2016": "X",
    "general_2020": "X",
    "general_2004": "",
    "general_2008": "",
    "general_2012": "X"
  },
  {
    "date_of_birth": "1984-07-05",
    "registration_date": "2004-09-29",
    "voter_status": "ACTIVE",
    ...
```

Single County analysis: Cuyahoga and Franklin Counties

Ohio-Analysis-2-Single-County-Cuyahoga.html
Ohio-Analysis-2-Single-County-Franklin.html





Source: Ohio Secretary of State's web site; data captured on 2020-03-25.

Profiles of Ohio's two largest counties are very different based on age.

"Active" voter lines added to Lee's analysis to explore that factor.

Adapted from Lee's script: plot_turnout_by_age.py

Single County analysis

fraction voted (turnout):

Ohio-Analysis-2-Single-County-Cuyahoga.html
Ohio-Analysis-2-Single-County-Franklin.html

For each age interval, 18 to 105 years:

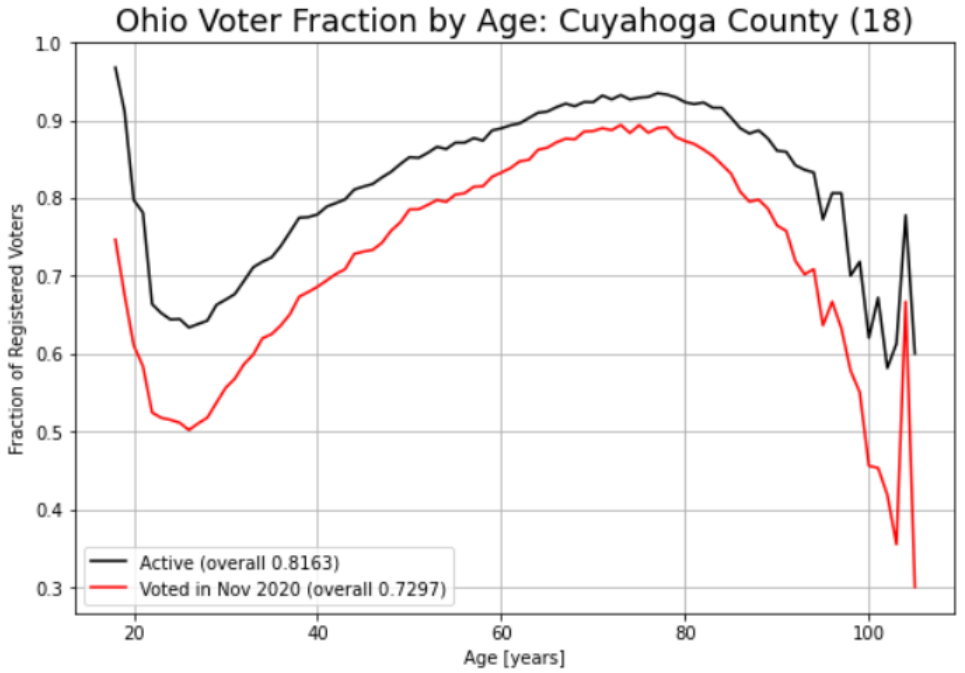
Fraction Active = $\frac{Active\ voters}{Registered\ voters}$

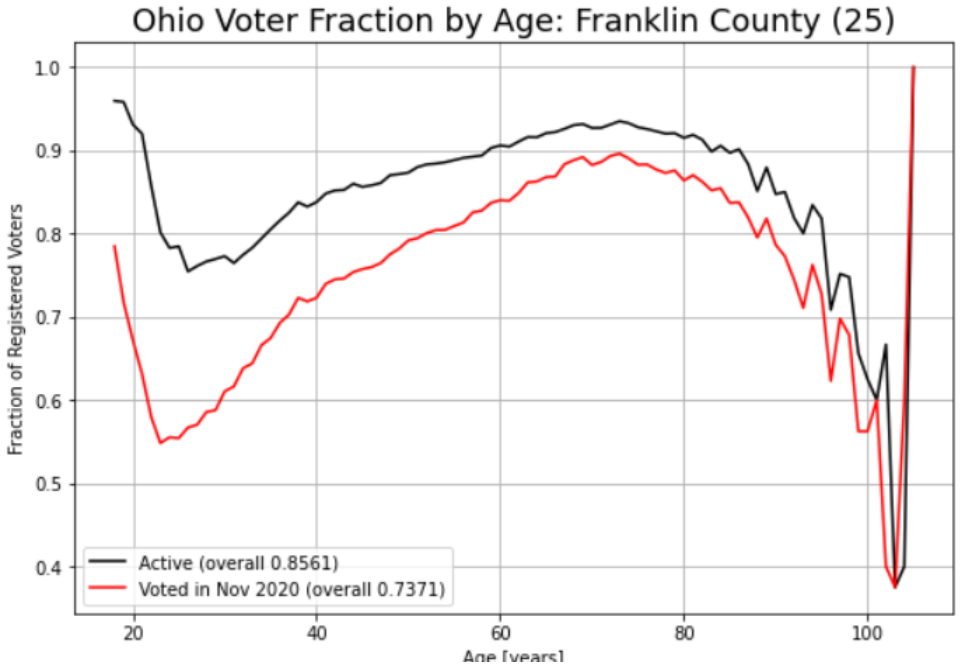
Fraction Voted = $\frac{Nov\ 2020\ voters}{Registered\ voters}$ = **Turnout**

Overall values *(not averages)*:

Overall Active Fraction = $\frac{\sum_{age} Active\ voters}{\sum_{age} Registered\ voters}$

Overall Voted Fraction = $\frac{\sum_{age} Nov\ 2020\ voters}{\sum_{age} Registered\ voters}$





Single County analysis

”normalized” fraction voted (“normalized” turnout):

Ohio-Analysis-2-Single-County-Cuyahoga.html

Ohio-Analysis-2-Single-County-Franklin.html

For each age interval, 18 to 105 years:

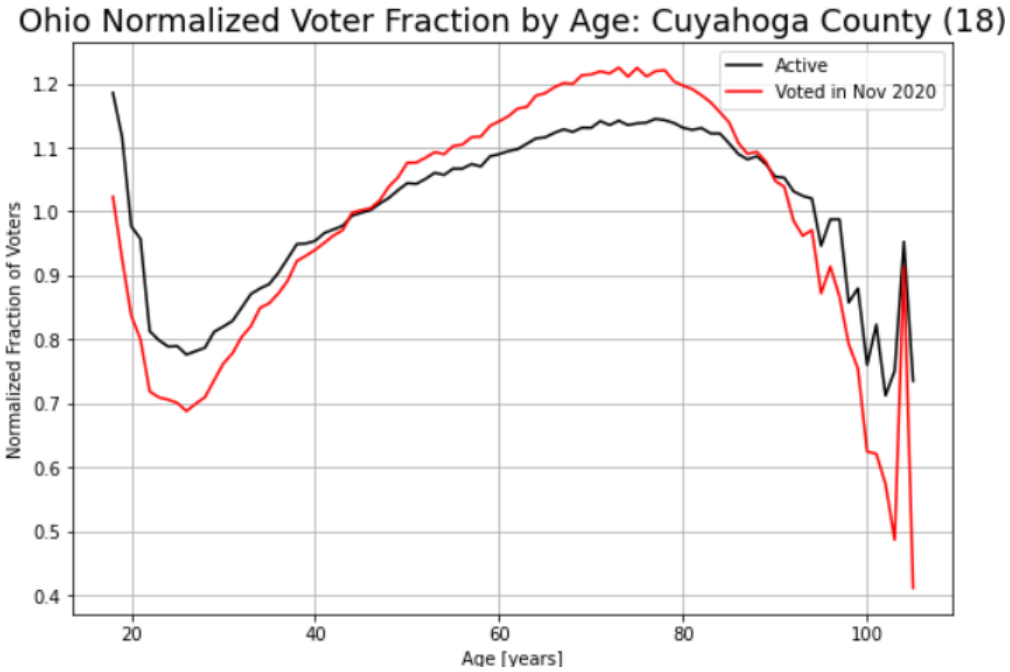
$$\textbf{Normalized Fraction Active} = \frac{\frac{Active\ voters}{Registered\ voters}}{Overall\ Active\ Fraction}$$

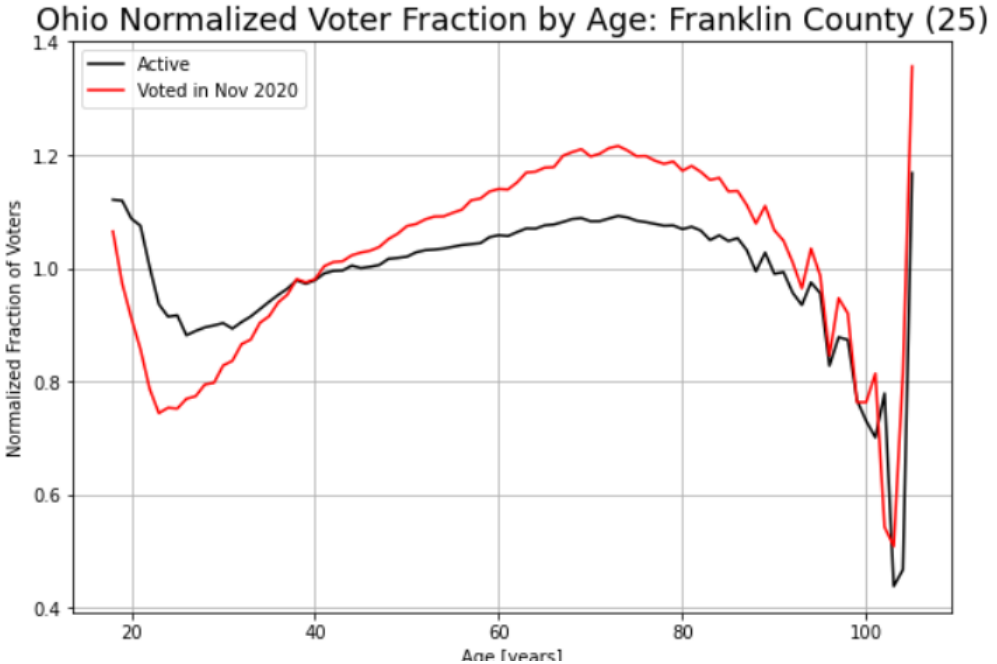
$$\textbf{Normalized Fraction Voted} = \frac{\frac{Nov\ 2020\ voters}{Registered\ voters}}{Overall\ Voted\ Fraction} = \text{Turnout Key}$$

where

$$\textbf{Overall Active Fraction} = \frac{\sum_{age} Active\ voters}{\sum_{age} Registered\ voters}$$

$$\textbf{Overall Voted Fraction} = \frac{\sum_{age} Nov\ 2020\ voters}{\sum_{age} Registered\ voters}$$





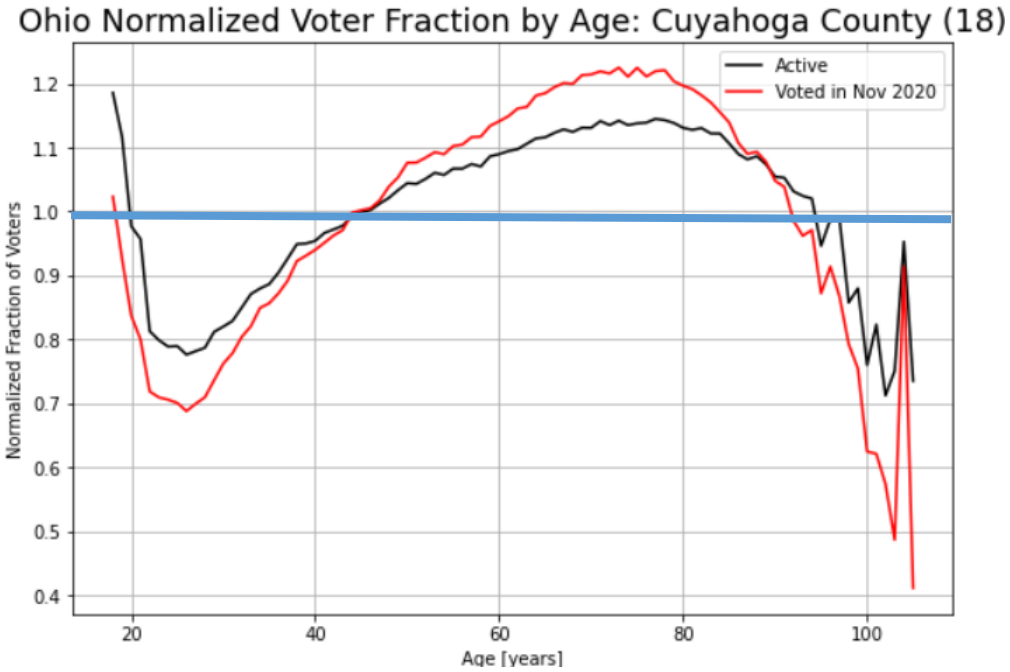
Single County analysis

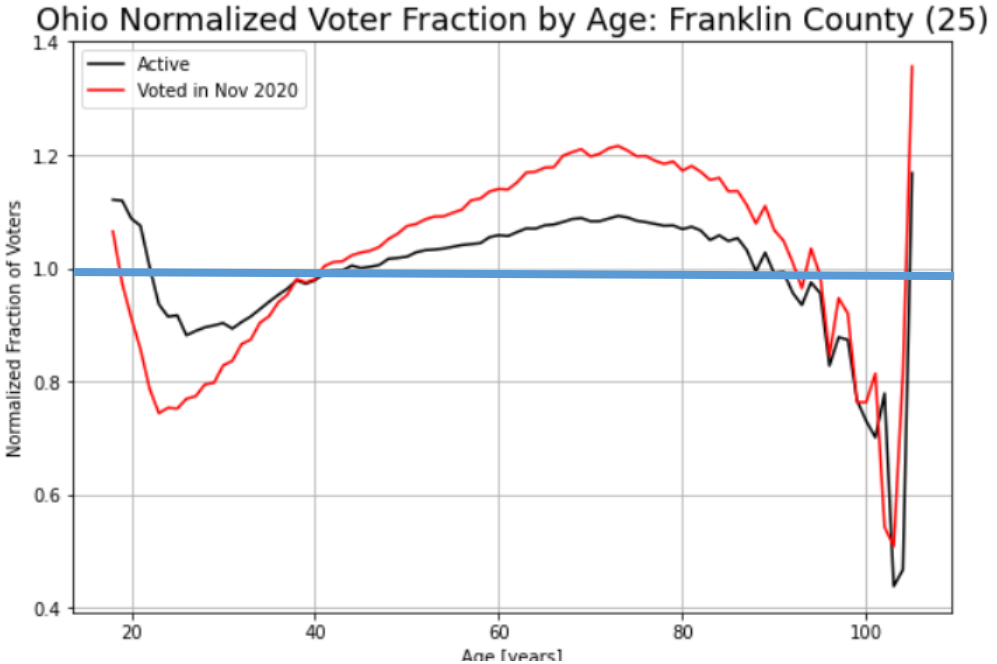
”normalized” fraction voted (“normalized” turnout):
Ohio-Analysis-2-Single-County-Cuyahoga.html
Ohio-Analysis-2-Single-County-Franklin.html

Comments:
The “normalized” fraction shows age intervals that are “above” 1 or “below” 1, where 1 is the overall fraction for all ages.

For example, in Cuyahoga County voter turnout was **below** the overall rate for ages ~18 to ~45 and above 95. Turnout was **above** the overall rate for ages ~45 to ~95.

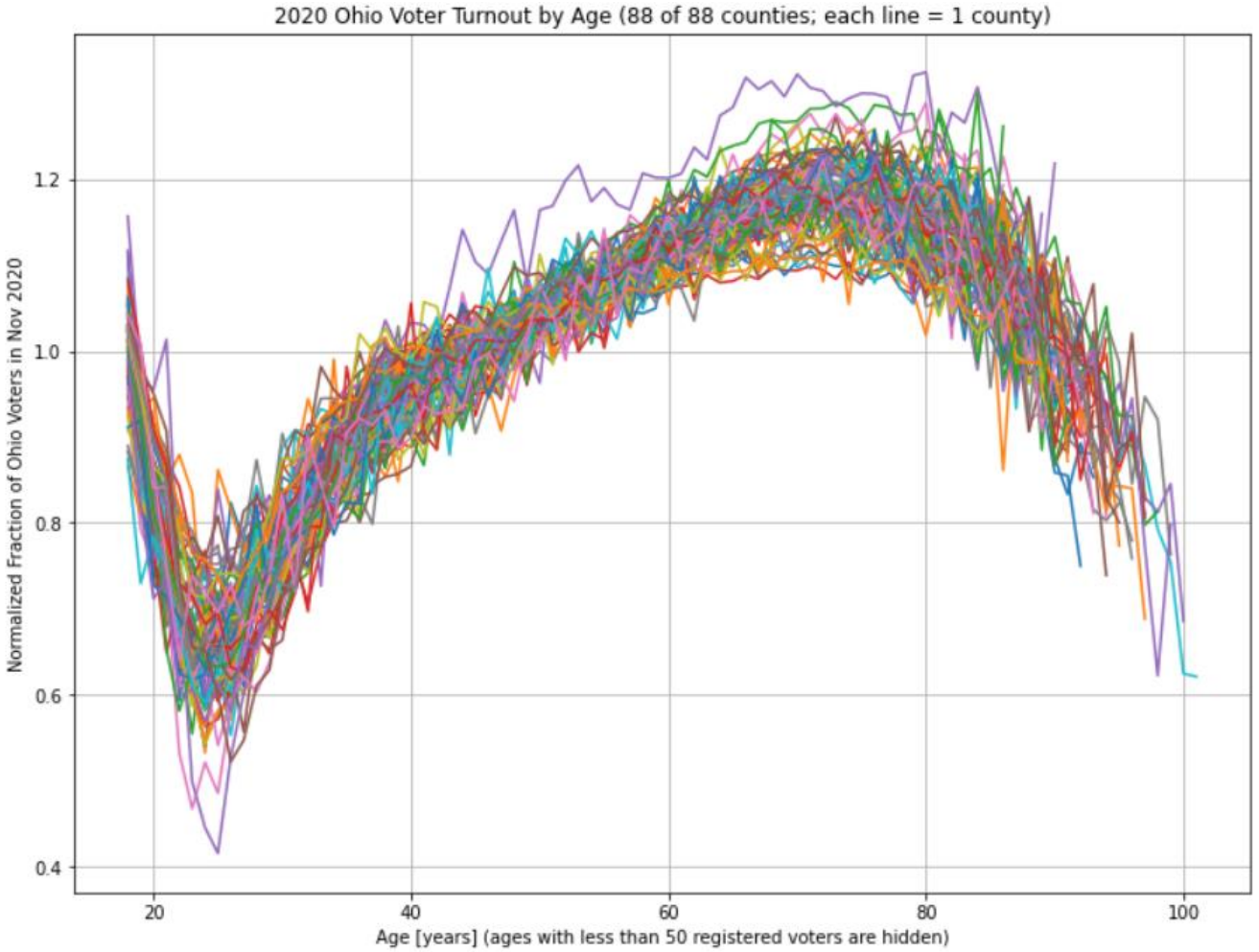
“Overall rate” is not the average by age.





All Ohio Counties: “Normalized” Voter Turnout Keys

Ohio-Analysis-3-All-Counties.html



Notes:

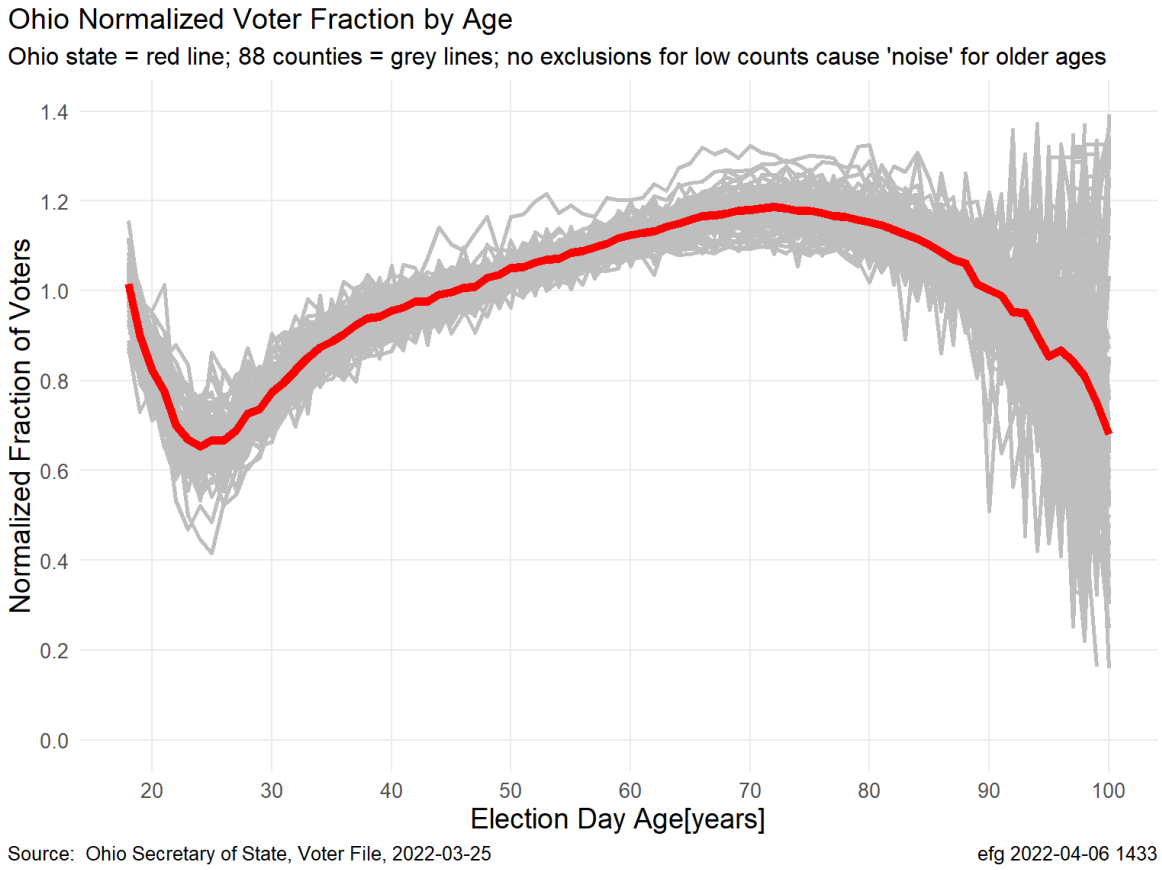
- These county plots include the red lines in the normalized turnout plots for Cuyahoa and Franklin Counties on the last slide
- Variation of normalized turnout varies by 15-20% across all counties for any given age.
- All of these county “keys” are highly correlated but not identical.

Adapted from Lee’s script: plot_turnout_by_age.py

All Ohio Counties: “Normalized” Voter Turnout Keys

Ohio-Key-Heatmap.html

Compare statewide “key” to separate “keys” by county



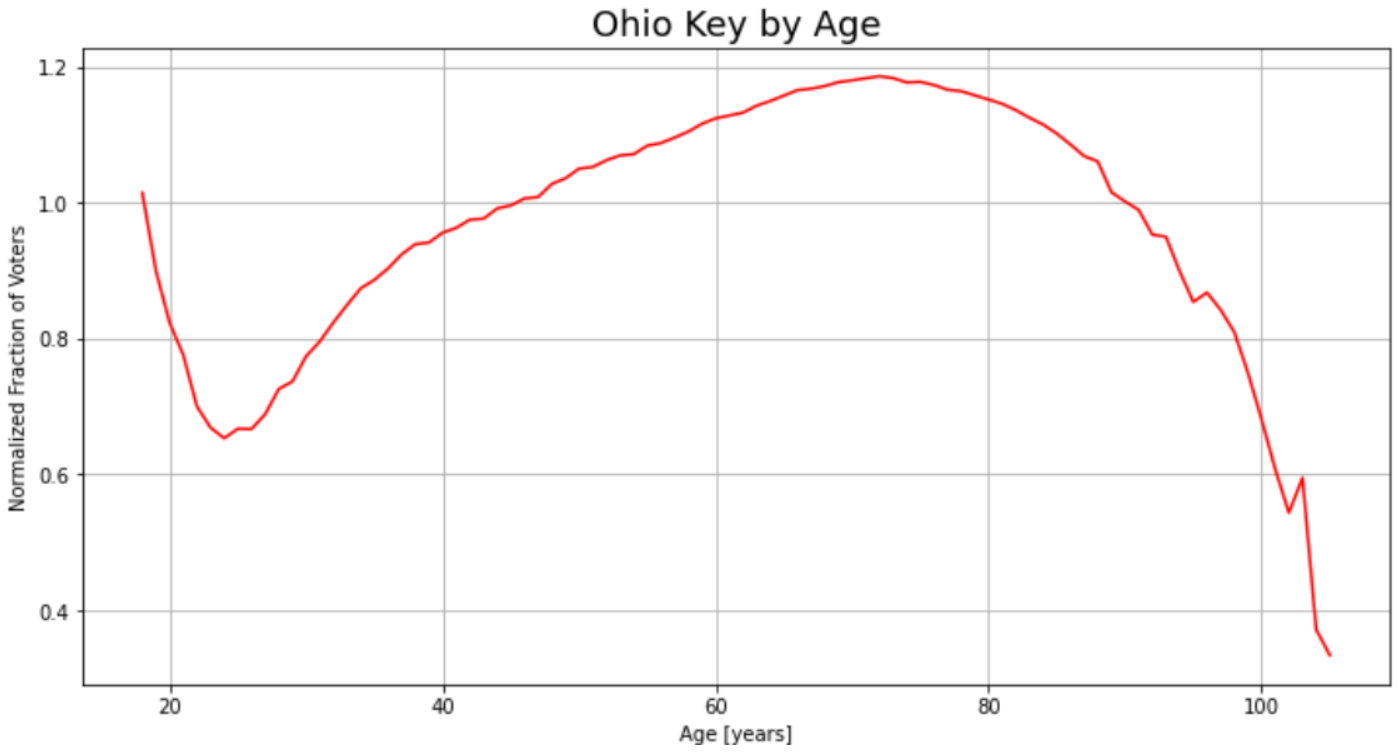
The normalized voter turnout by age varies Considerably by county

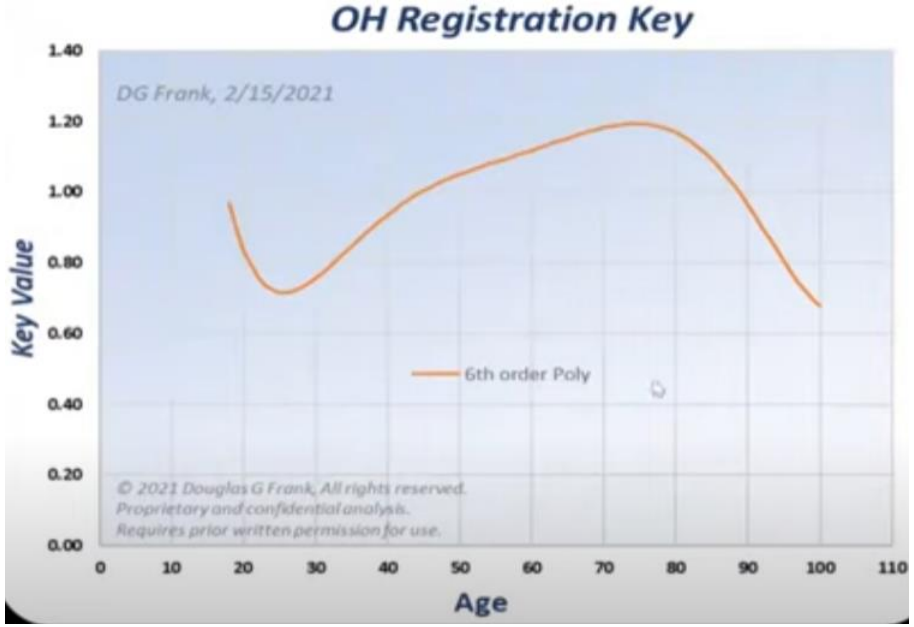
Small counts (small “n”) for ages 90+ result in “noisy” normalized key

Generate Ohio Normalized Turnout “Key”

Ohio-Analysis-4-Generate-Key.html

Apply methodology used for single county in previous slides to whole State of Ohio:





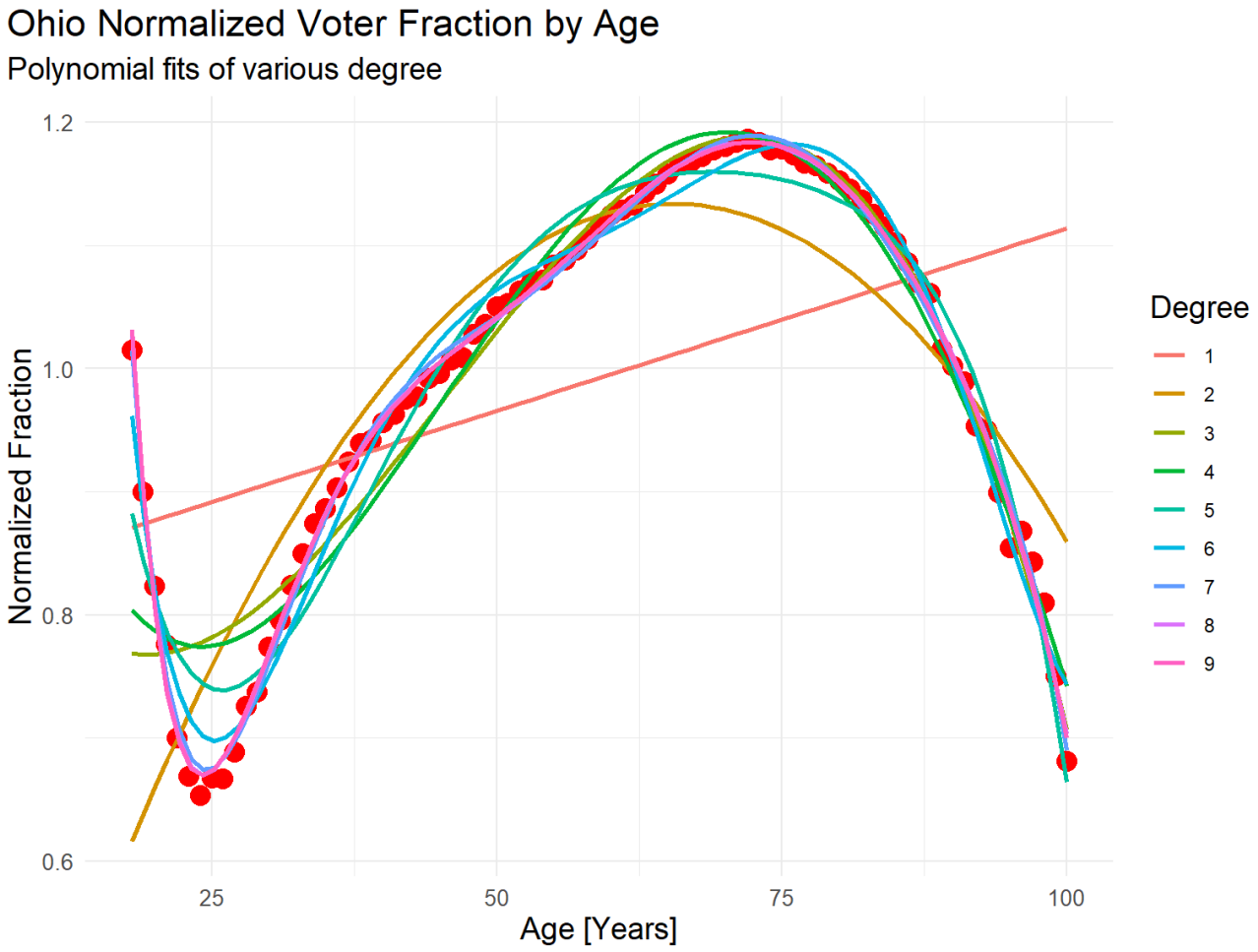
Dr. Frank’s version of OH Key from presentation to Kansas House Election Committee on 2022-03-15

Adapted from Lee’s script: generate_key.py

“Key” written to file **key.json**, which is not easy to review in a text editor or Excel since it’s over 200 characters wide, and not necessarily ordered.


```
{"18": 1.0148327400458397, "19": 0.8998751228239705,
. . .
"105": 0.3339083283101293, "104": 0.37172693302673665}
```

Fit Polynomials to Ohio Normalized Turnout Key

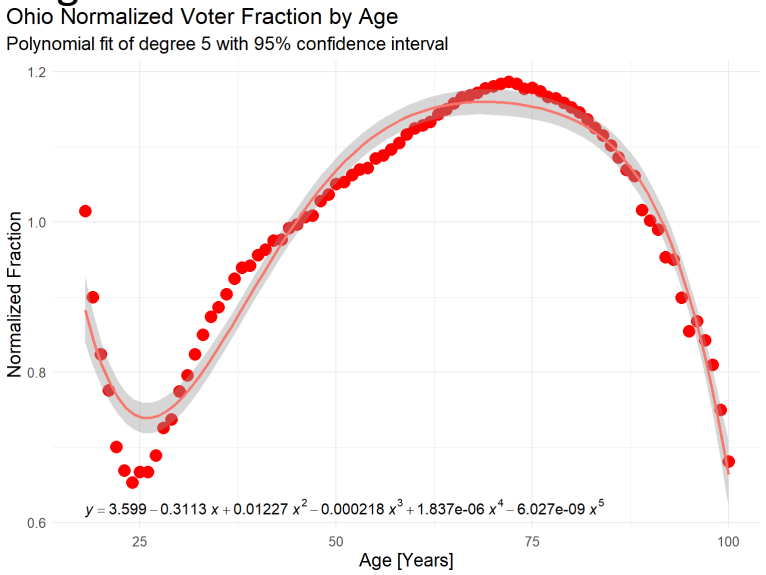


- Here age range limited to [18, 100].
- Fit Ohio Normalized Turnout curve (red dots) to polynomials of various degrees
- Higher degree provides curvature/ "wiggliness", but too high can lead to overfitting.
- Akaike Information Criterion (AIC) indicates highest degree over range 1 to 9 was the "best" model.
- R^2 approaches 1 as degree increases.
- There is nothing "remarkable" about these curve fits.
- Curve fits offer few insight about data but provide good numerical interpolation.

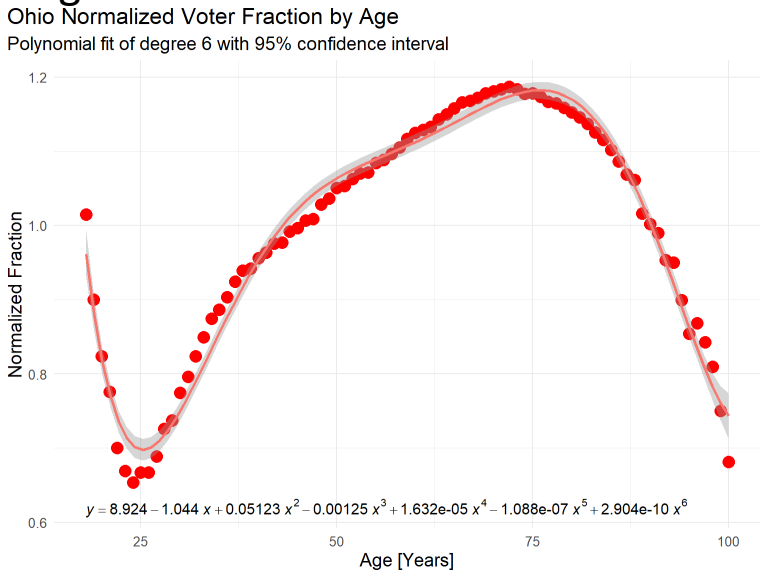
RStudio Notebook: Ohio-Key-Polynomials.html, Section 5.1 (AIC), Section 5.2 (R^2), polyfitsMany-1.png

Fit Polynomials to Ohio Normalized Turnout Key

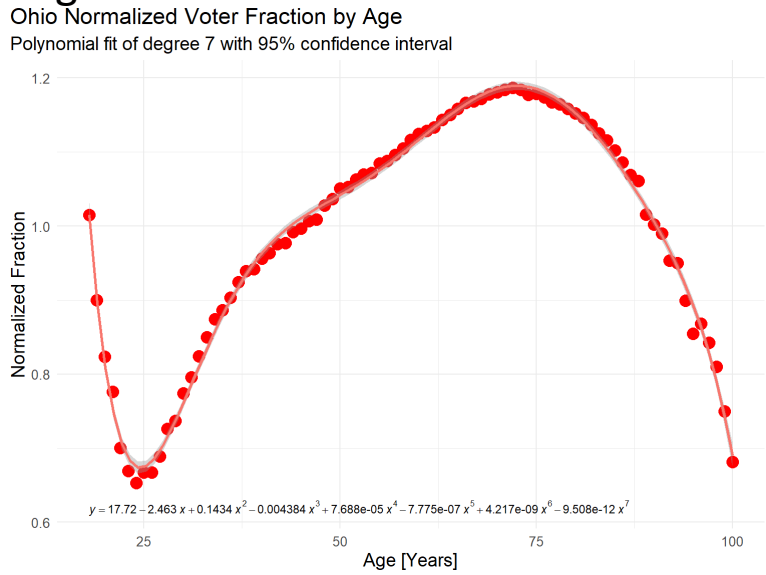
Degree 5



Degree 6



Degree 7



95% confidence interval is about width of line for 7-th degree fit, but this may be "overfitting"

In polynomial equations above, x = Age, y = Normalized Fraction

Largest residual over range: 0.132 (5^{th}), 0.062 (6^{th}), 0.038 (7^{th})

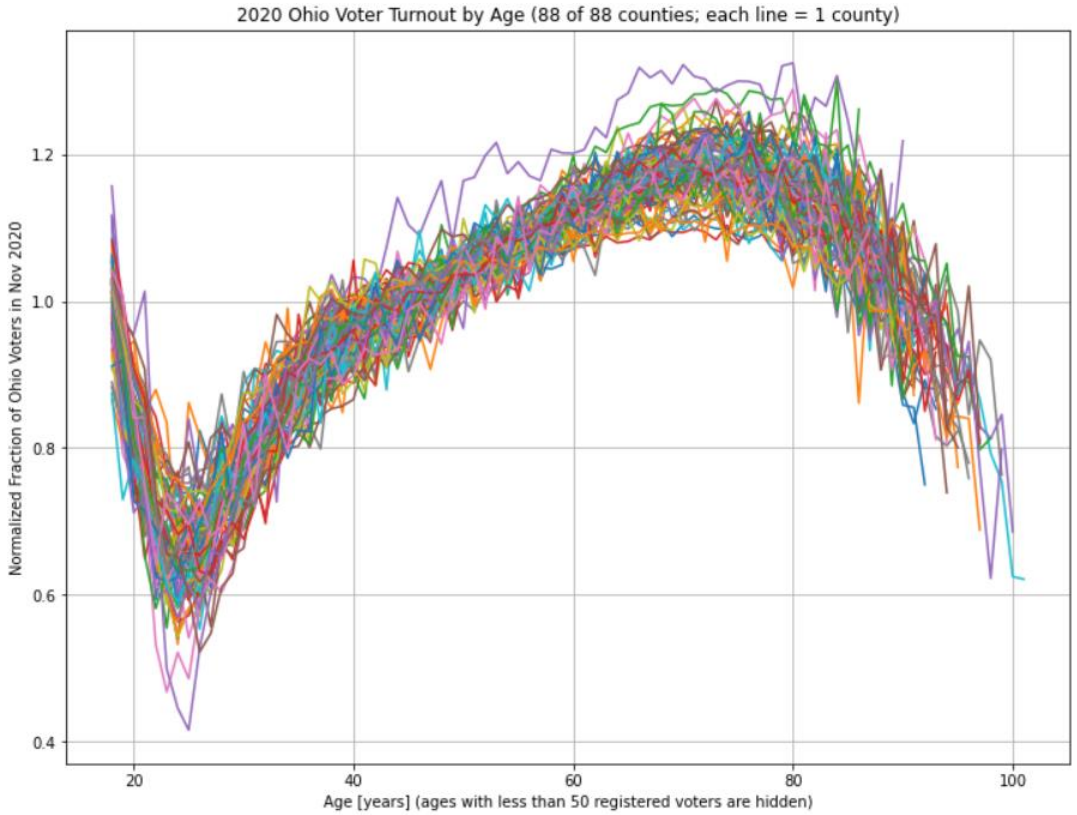
But, there is *no need for polynomial fit if original normalized turnout curve is used directly!*

RStudio Notebook: Ohio-Key-Polynomials.html, Section 6, polyfit5-1.png, polyfit6-1.png, polyfit7-1.png

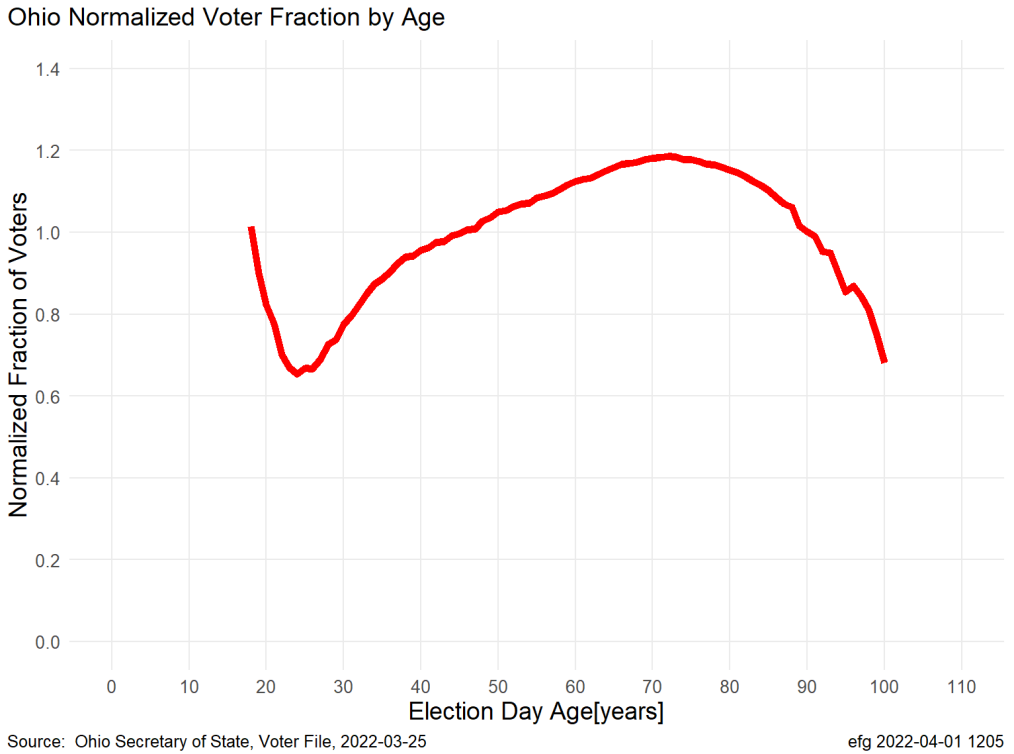
County vs State Normalized “Key” Curves

All curves are *highly* correlated but not identical

All Counties



Ohio Statewide



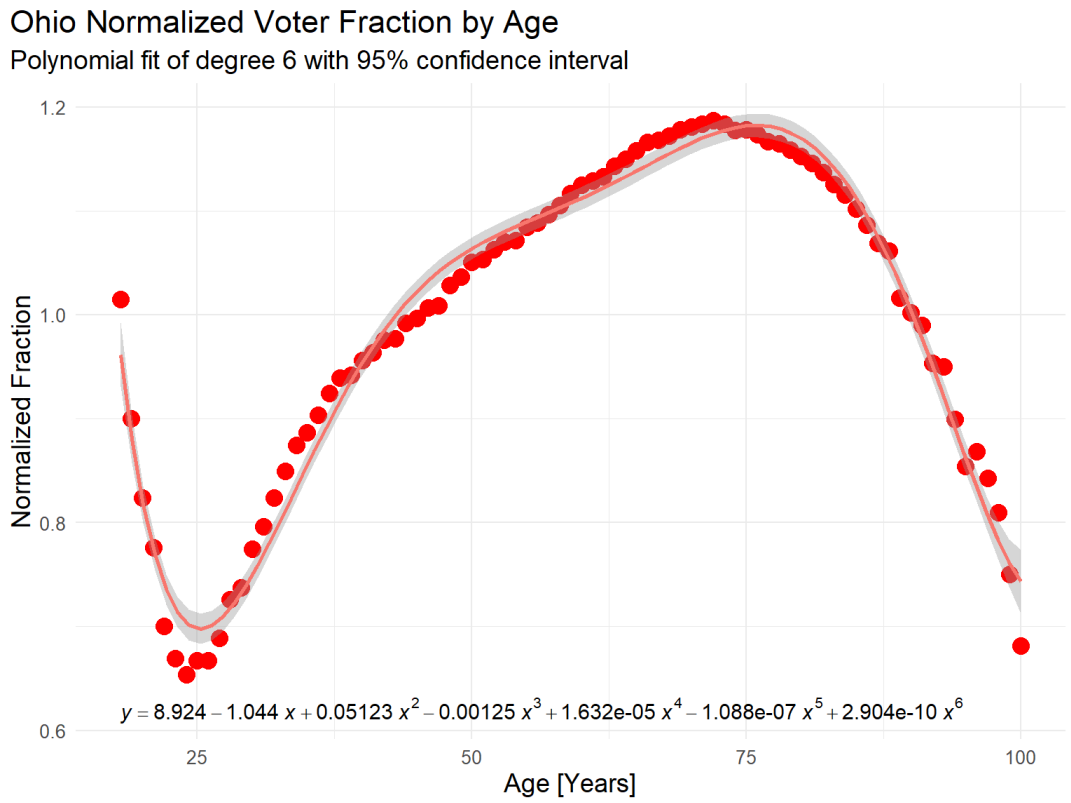
Scaling is different between plots.

Plots show raw turnout values without using polynomial fit.

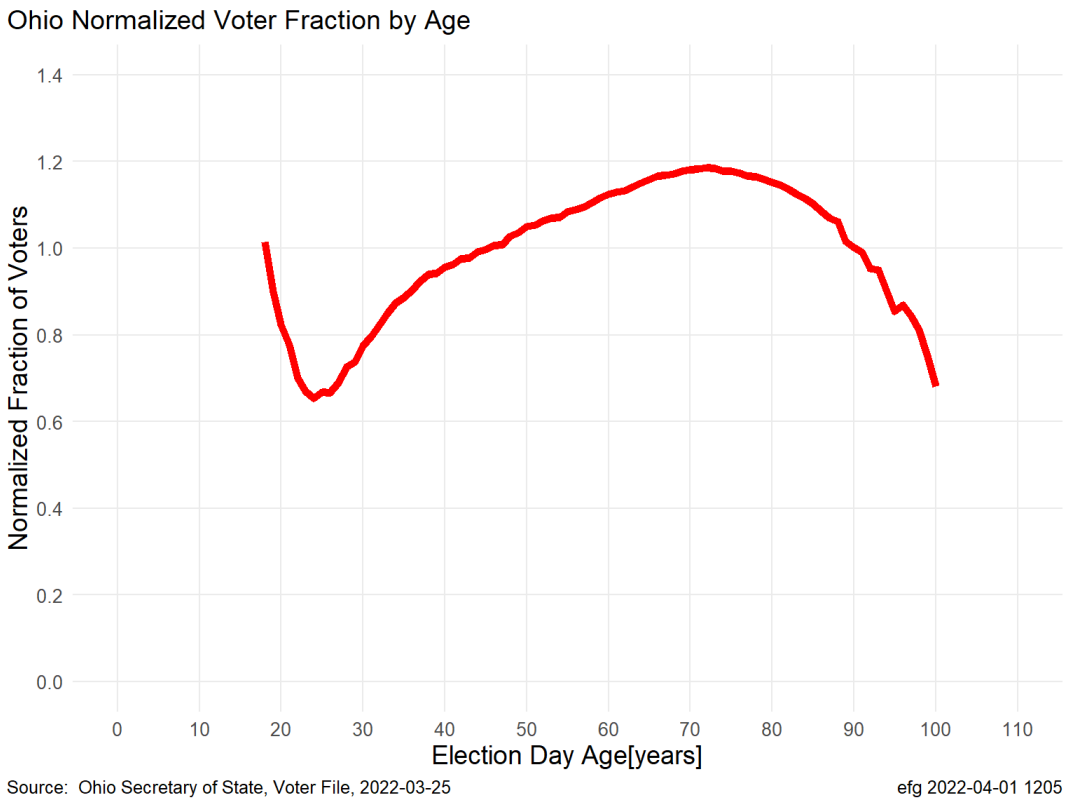
Ohio-Analysis-3-All-Counties.html; Ohio-Key-Polynomials.html, Section 3, OhioNormalized-1.png

Ohio Turnout "Key"

Key: 6th Degree Polynomial Fit to Data (**7 numbers**)



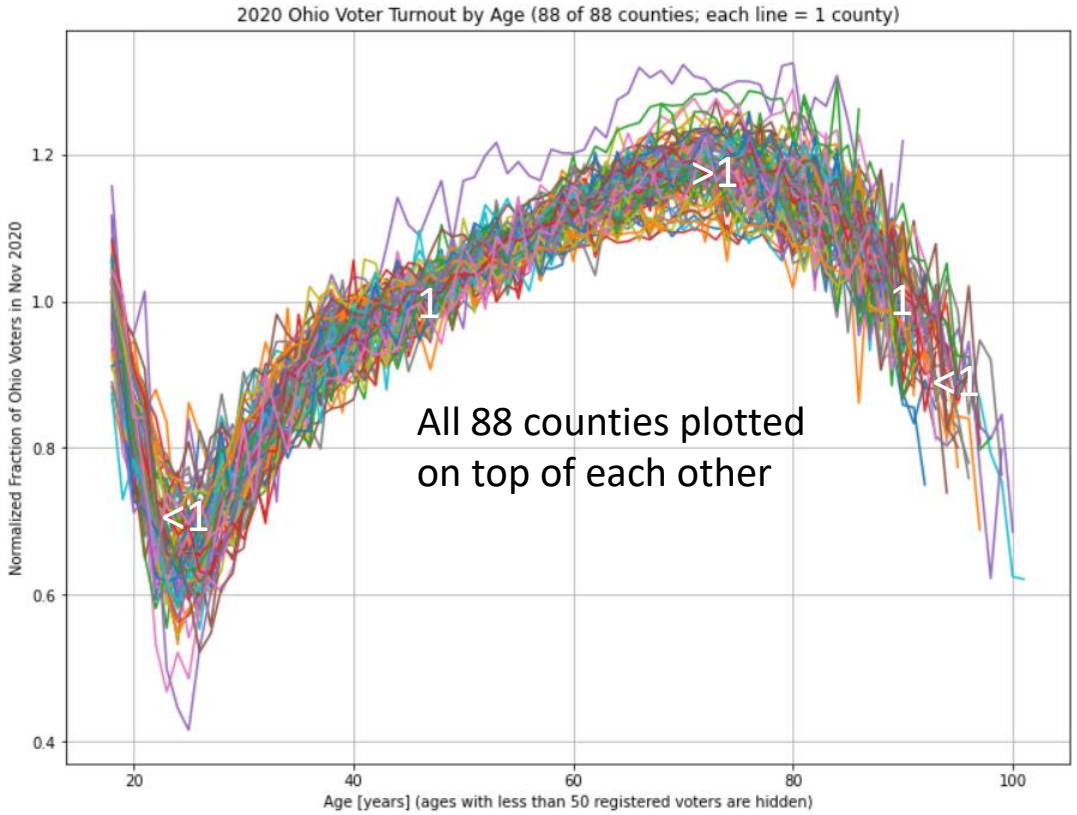
Key: Based on Yearly Data (**83 numbers**)



Scaling is different between plots but both are based on the same data

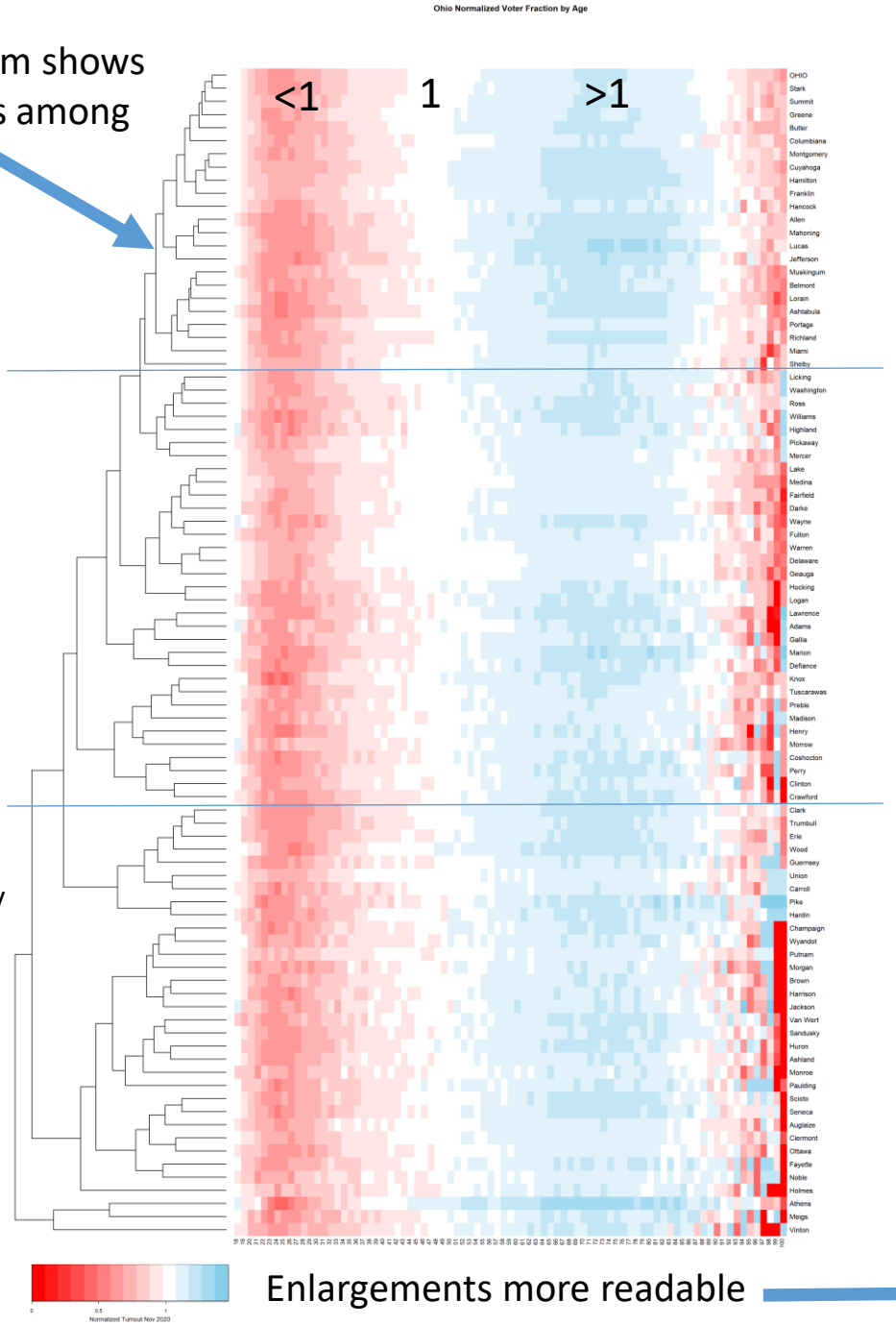
County Normalized Curves

All Counties



Ohio-Analysis-3-All-Counties.html;
Ohio-Key-Heatmap.html, heatmap-1.png

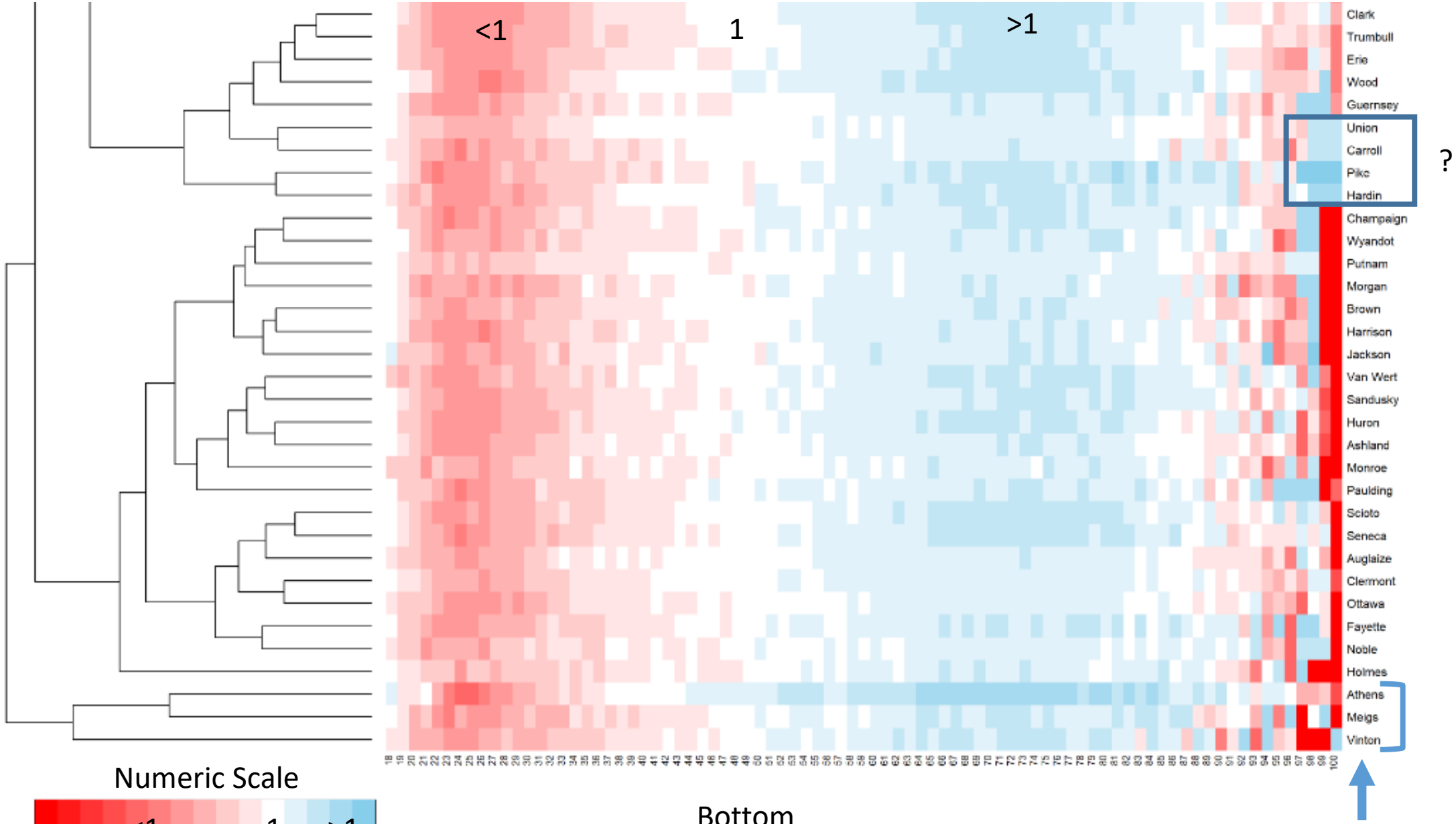
dendrogram shows similarities among counties



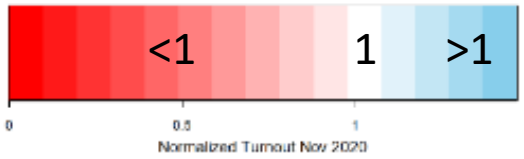
"Heatmap"

One row per county. Ordered by similarity (but ordering is not unique).

Enlargements more readable



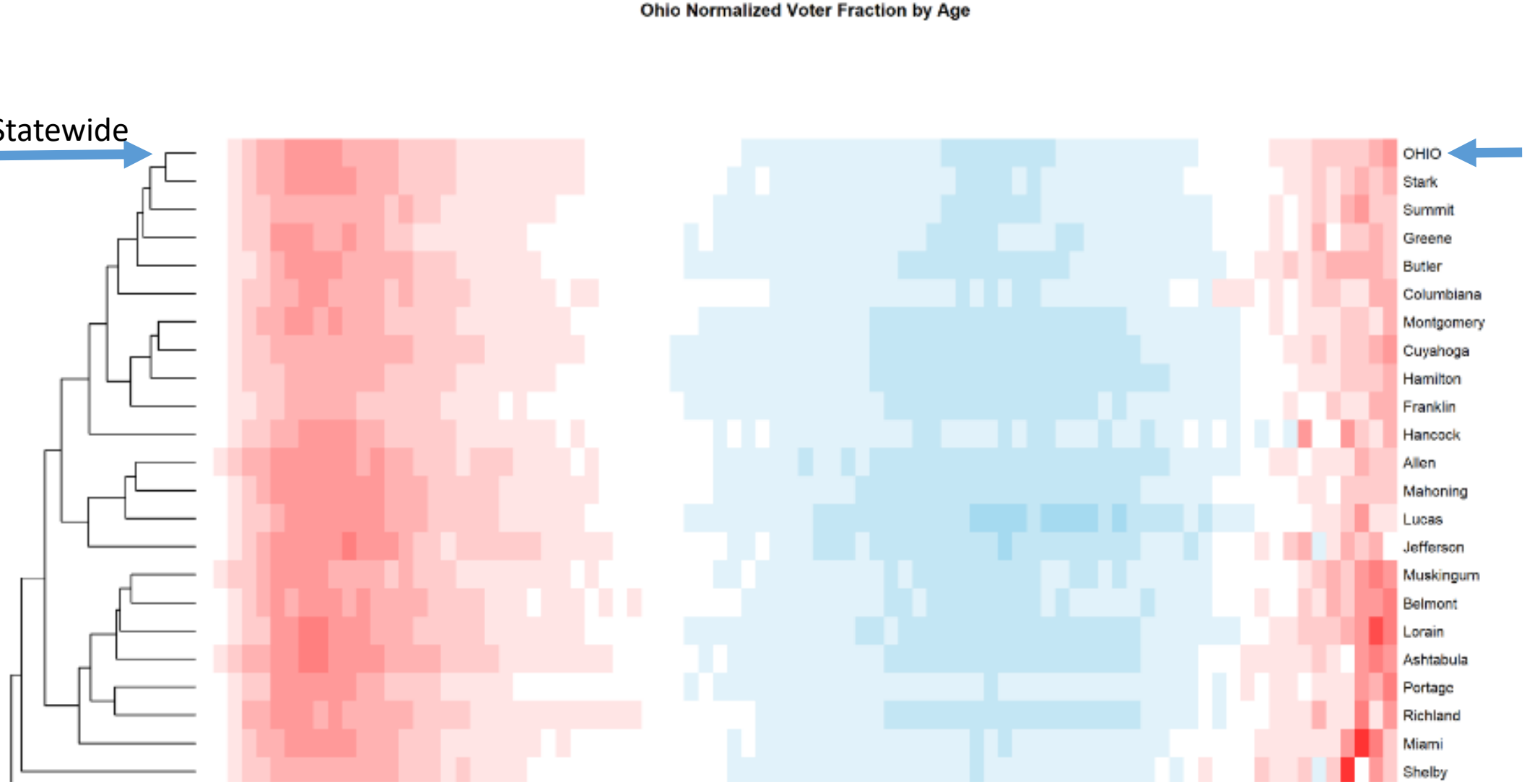
Numeric Scale



Bottom
With age scale

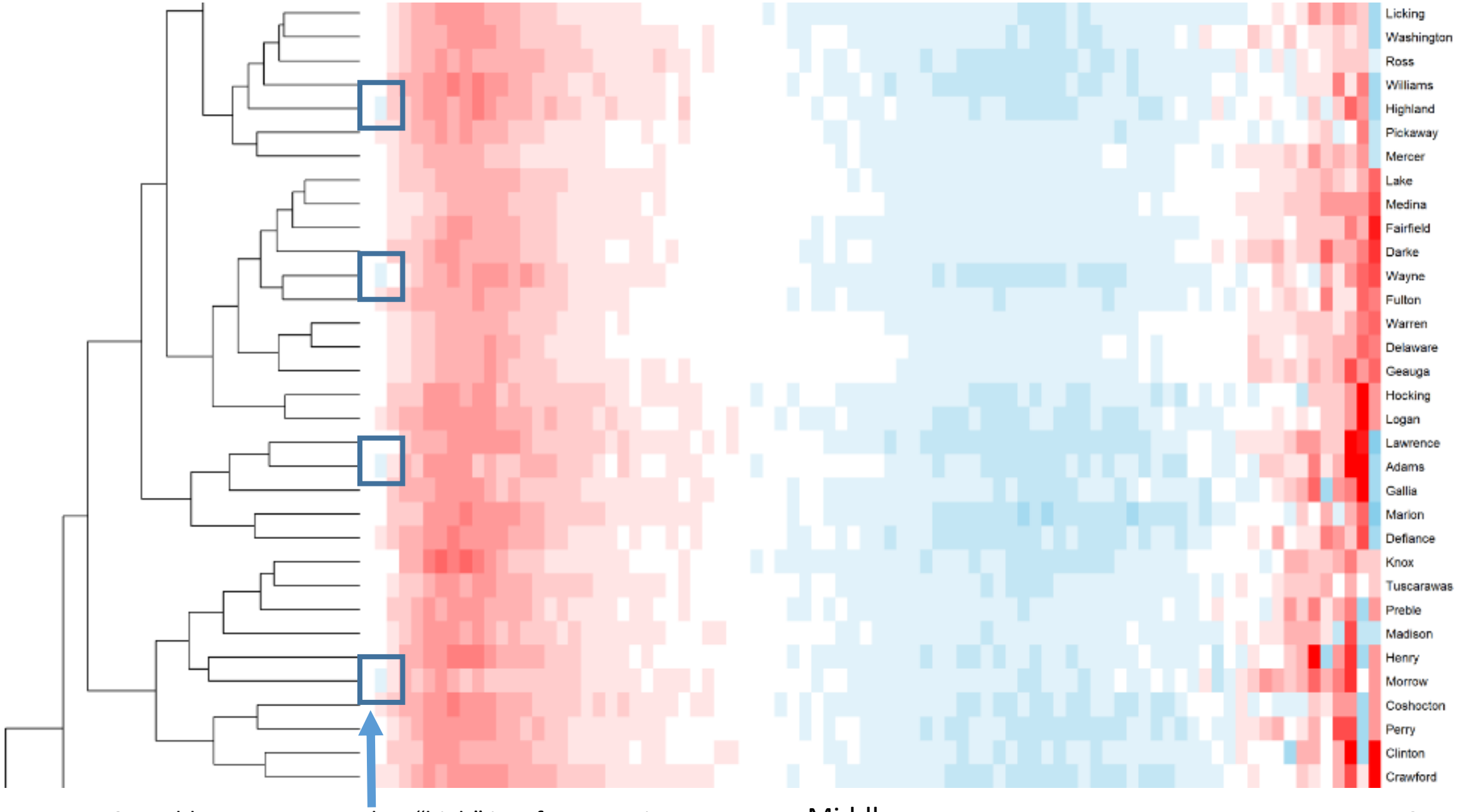
"Most different" counties:
Athens, Meigs, Vinton

Ohio Statewide



Top
with overall OHIO row

Stark County profile is most similar to Ohio statewide profile



18-yr old turnout somewhat “high” in a few counties

Middle