Automated Data Visua for Policymaking.

Session 9

Autumn 2025

Resources.

Places to find, and share

Resource 1: **my site:**
www.richarddavies.io/data-science

Resource 2: **chart library.**
www.richarddavies.io/library

Resource 3: **course Google sheet.**
Google sheet. Link.

Resource 4: **course DropBox.**
DropBox. Link.

Resource 5: **Playfair Prize**
www.playfairprize.com

Google Sheet

DropBox

Nearly there...

Three more weeks…

- W1. Your Live Site. HTML, CSS, JavaScript. GitHub and embedding.
- W2. Data debating – chart design. Raw files, editing data. String and date functions.
- W3. The Language of Data. Why visualisation is vital in policy. Manipulating data with Python.
- W4. Data. Collection, storage and biases. “Tidy” data storage. Data access. APIs. Scraping with Python.
- W5. Programming. Control structures. Loops and conditionals. Combining APIs with loops in Python.
- W6. Break. No classes. No homework. Tip: Complete all your portfolio work and start on project.
- W7. Maps. Cartography history. Projections. Base maps. Putting data on maps.
- W8. Big data. Challenges with scale. A big data cookbook.
- W9. Advanced analytics. Creating visualisations that help investigate relationships.
- W10. Machine Learning. History. Regressions and classification. Clustering and dimensionality reduction.
- W11. Large Language Models. Gathering, cleaning, visualising data. Auditing output.

9. Deeper analysis.

Visualisations to investigate relationships

Lecture 9.

- 9.1 Motivation: understanding things
- 9.2 Advanced Analysis: charts that suggest causes
- 9.3 Complex Analysis: inside the black box
- 9.4 Interactive analysis: letting users decide
- 9.5 Practical 9: Advanced charts, Interactives

9.1 Motivation.

Understanding things

RERUM CAUSAS COGNOSCERE

The causes of things.

The LSE's motto

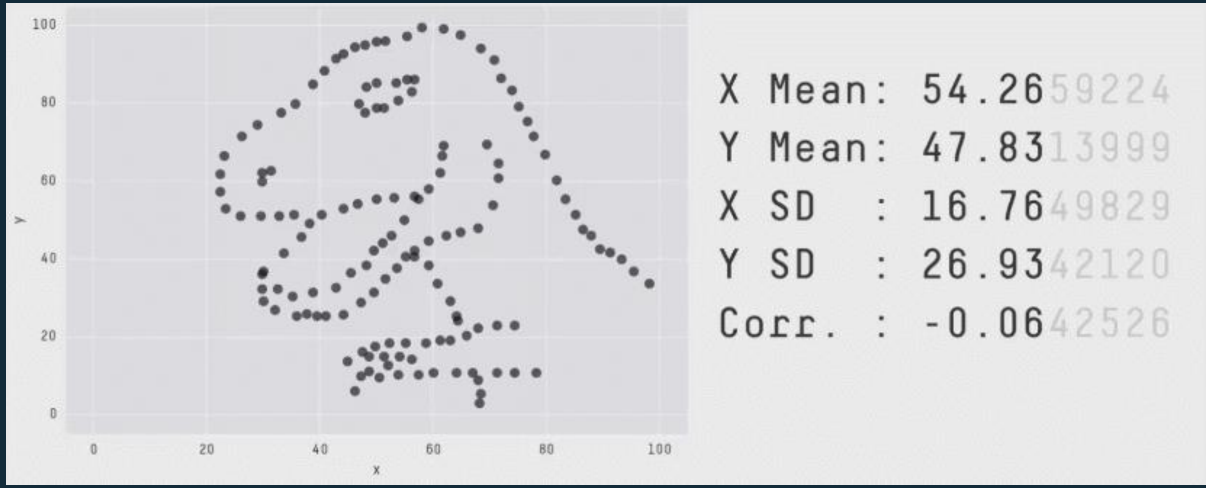
Felix qui potuit rerum cognoscere causas

Edwin Cannan, Professor of Political Economy, took this from Book 2 of Virgil's *Georgics*.

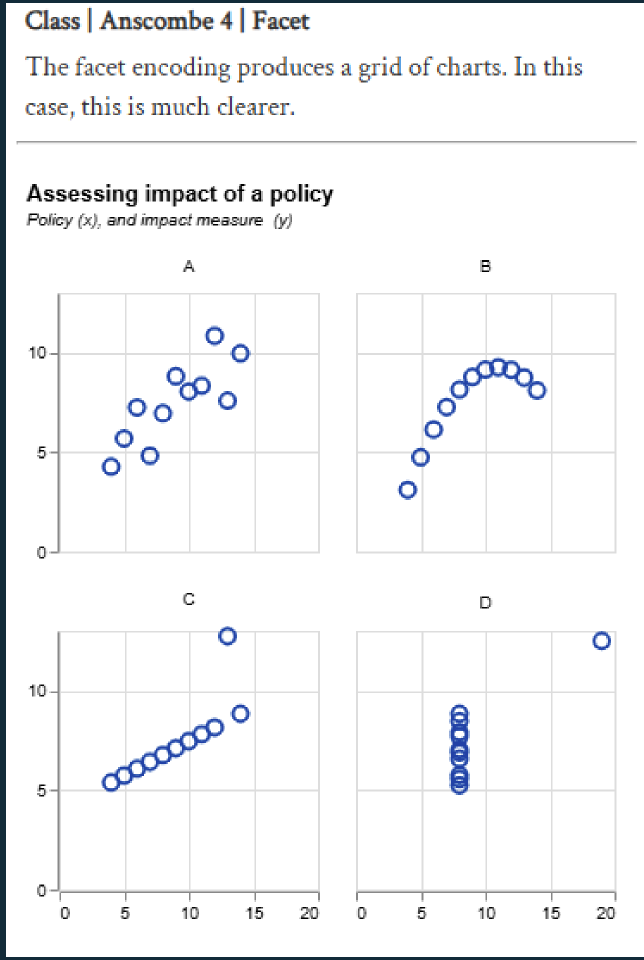
History of the LSE motto, and coat of arms: https://blogs.lse.ac.uk/lsehistory/2017/06/20/cheerful-nonsense-with-brains-behind-it-devising-the-lse-coat-of-arms/

Re-cap: Why visualisation is vital 1.

The hidden T-rex; Datasaurus, Autodesk, Albert Cairo, 2017



https://www.research.autodesk.com/publications/same-stats-different-graphs/



Anscombe:

"make both calculations and graphs. Both sorts of output should be studied; each will contribute to understanding"

Anscombe, F. J. "Graphs in Statistical Analysis." The American Statistician 27, no. 1 (1973): 17–21. https://doi.org/10.2307/2682899.

Our conclusion

You cannot make a responsible policy prescription without visualisation.

Why visualisation is vital 2.

Most types of causal inference rely on charts

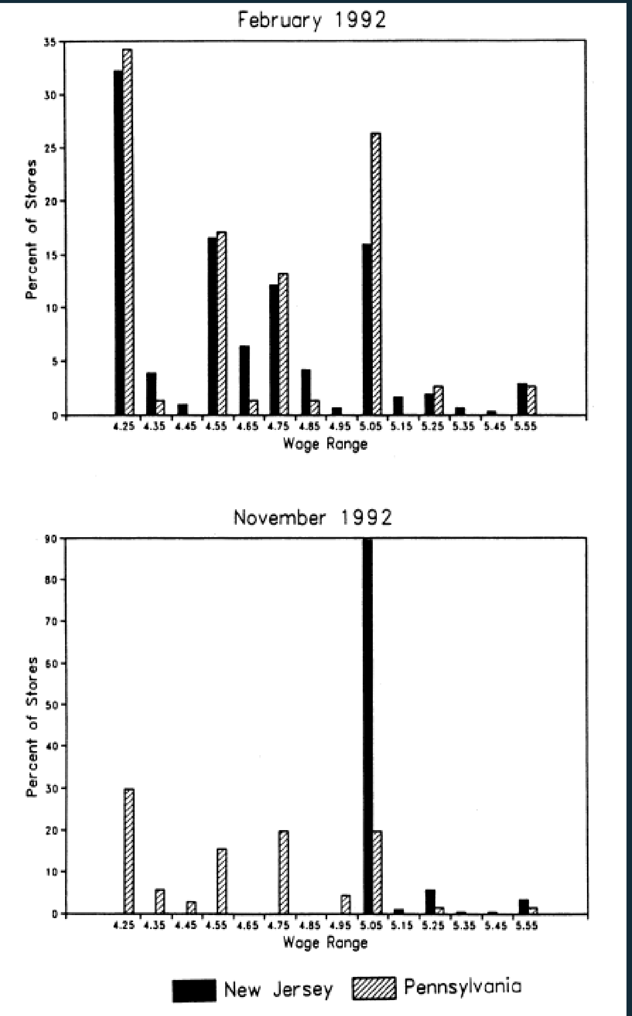


FIGURE 1. DISTRIBUTION OF STARTING WAGE RATES

Card and Kruger (1992)
https://davidcard.berkeley.edu/papers/njmin-aer.pdf

Two neighbouring states

NJ brings in min wage

Employment rises

Huge impact on policy

My assertion

Causal inference very often relies on charts and images as the vital step in convincing the reader.

Three levels of analysis.

Pros, cons and trade-offs.

- Level 1: Basic.
  - Line, Bar, Area, Pie.
    - Pros: simple, easy to understand. Work well in 2D.
    - Cons: hard to draw out relationships.

- Level 2: Advanced.
  - Scatter, bubble, distributions, de-trended (including univariate regression) shock analysis, Diff-in-Diff.
    - Pros: allow us to move towards causation, establish relationships. 3D is possible in 2D image by using bubble sizes etc.
    - Cons: harder to explain.

- Level 3: Complex.
  - Multivariate Regression, Principal Component Analysis, ML techniques.
    - Pros: fully interrogate data and establish significant patterns.
    - Cons: a ‘black box’ - often impossible to chart / visualise.

Level four: interactivity.

Get the user to do their own analysis

- Interactive analysis
  - We are seeking to convince the user that they understand the data.
  - We want to lead them to a policy conclusion.
  - In the new world of embedded live charts, we can allow the user to “play” with the data themselves.
    - Pros: allows the user to engage with the data.
    - Cons: more complex charts to code.

9.2 Advanced analysis.

Going beyond the bar.

Advanced charts.

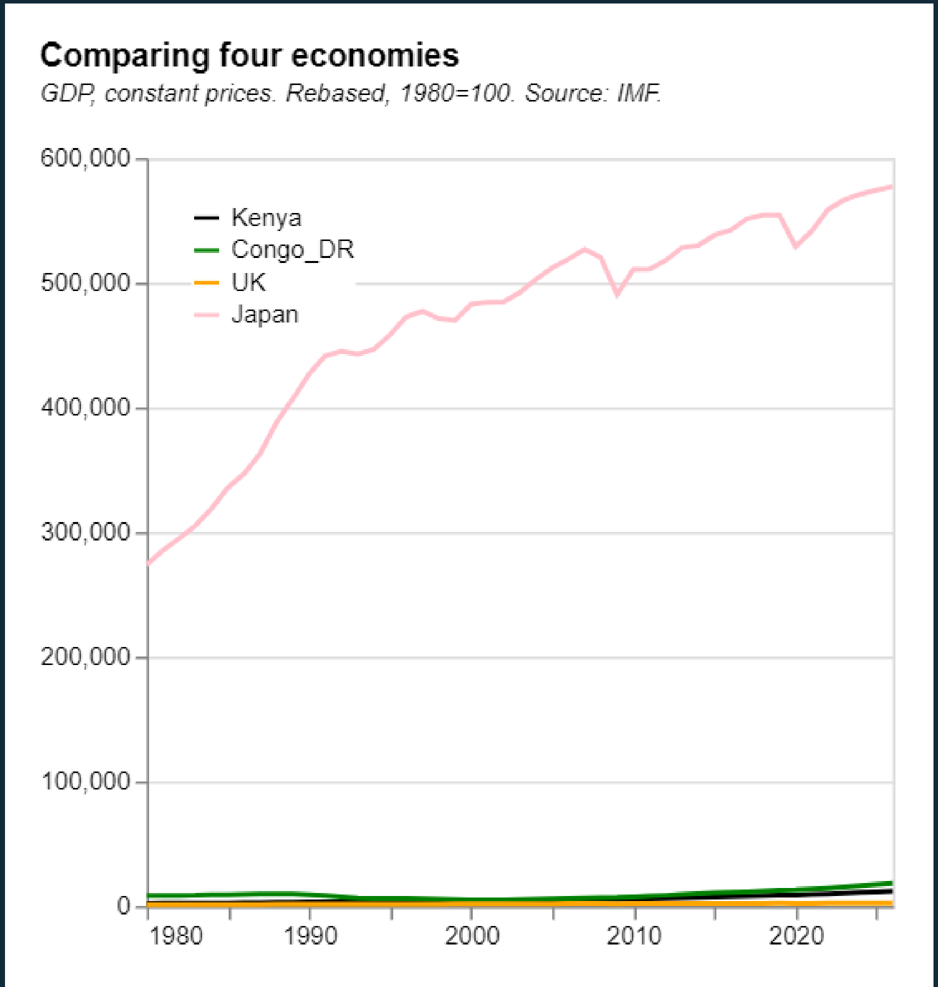
When to use them

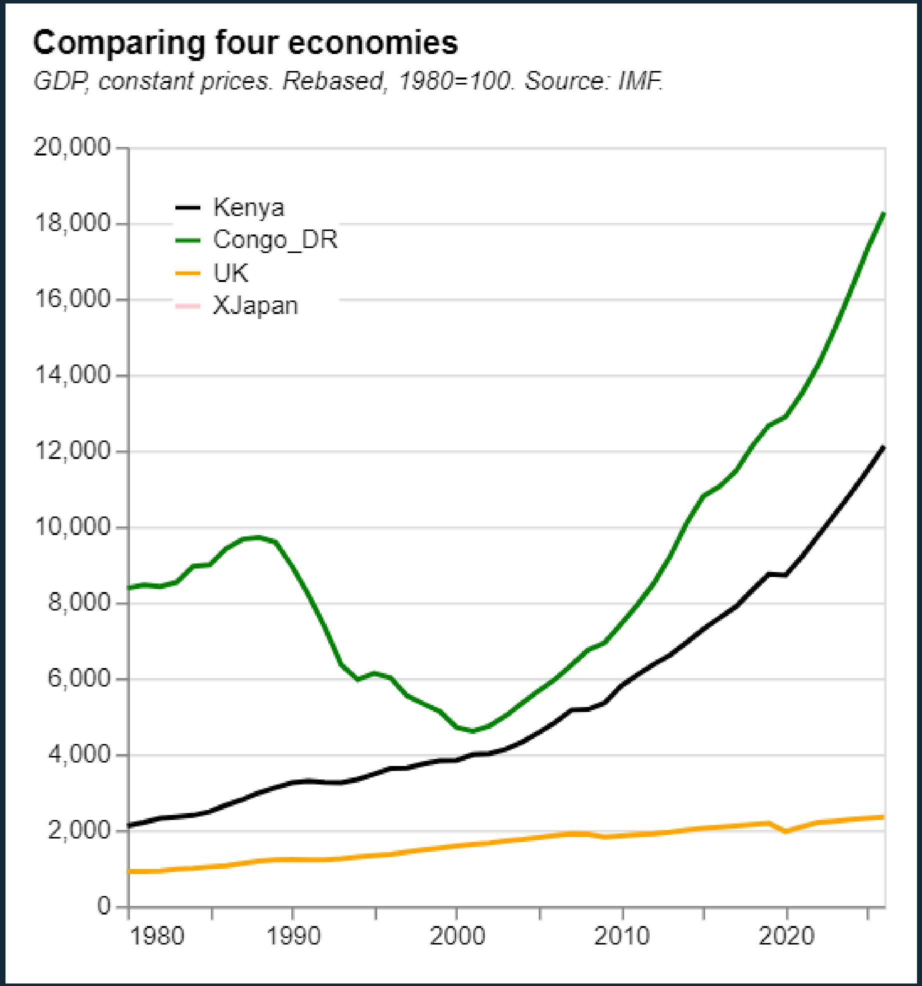
Why might we want to go beyond the bar chart, line chart?

- Accuracy. Removing noise / volatility.
- Comparability. Making series comparable.
- Clarity. Helping the reader: drawing attention to key findings.
- Causality. Establishing impact.
- Uncertainty. To communicate that we are not sure about future data.

Rebasing.

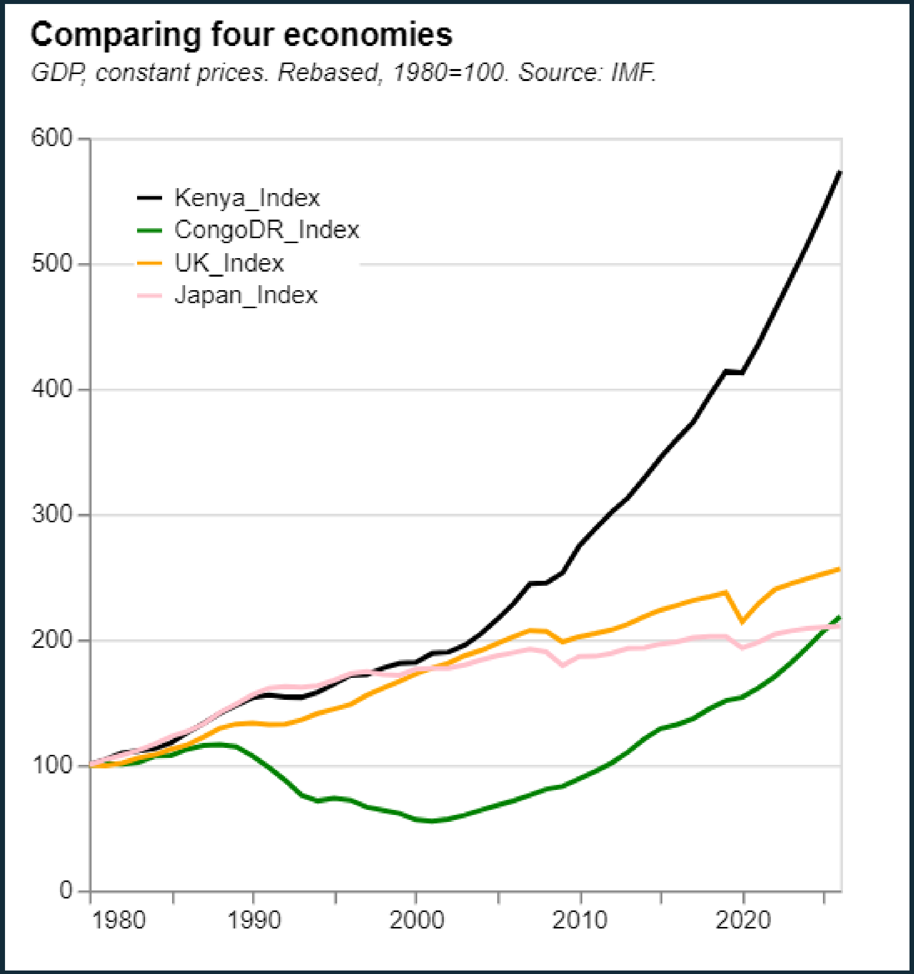
Problem: sometimes the same data series takes very different values across units of observation





Rebasing.

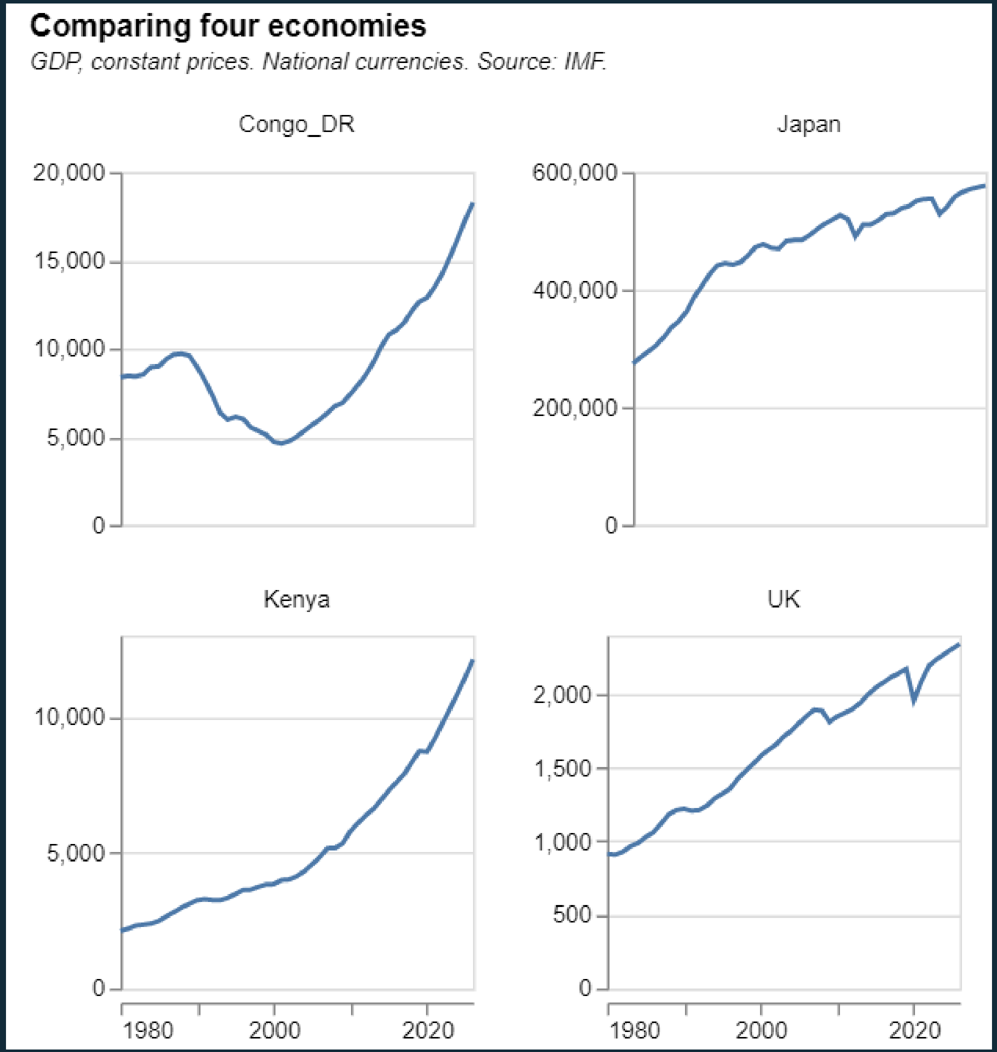
Solution: rebase so that the data start at the same point (often = 100).
Alternatively, rebase to a point that has some context as an anchor (2008, say).



Vega code.

Rebasing - alternative.

There is often no single correct way to visualise data. Here, a Trellis chart does a similar thing…



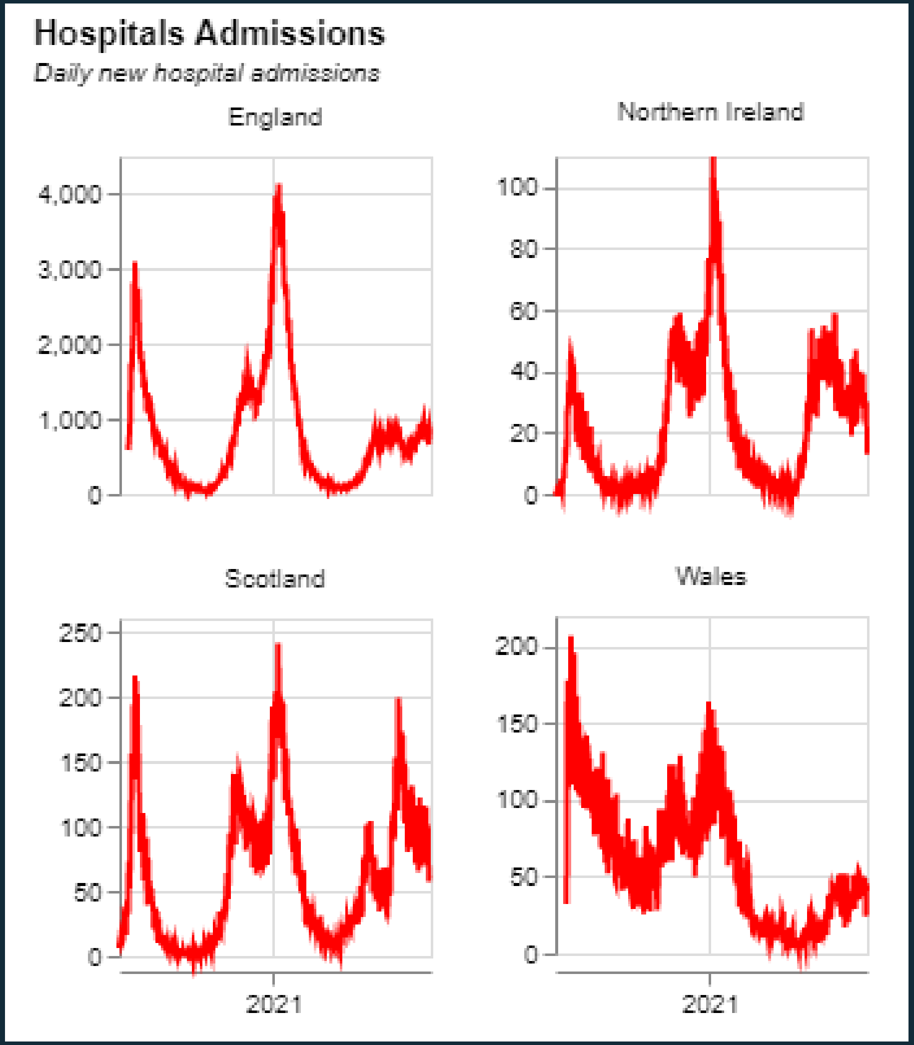
[Vega code].

Trellis 1.

Using a trellis to compare data series. The same y axis.

Trellis 2.

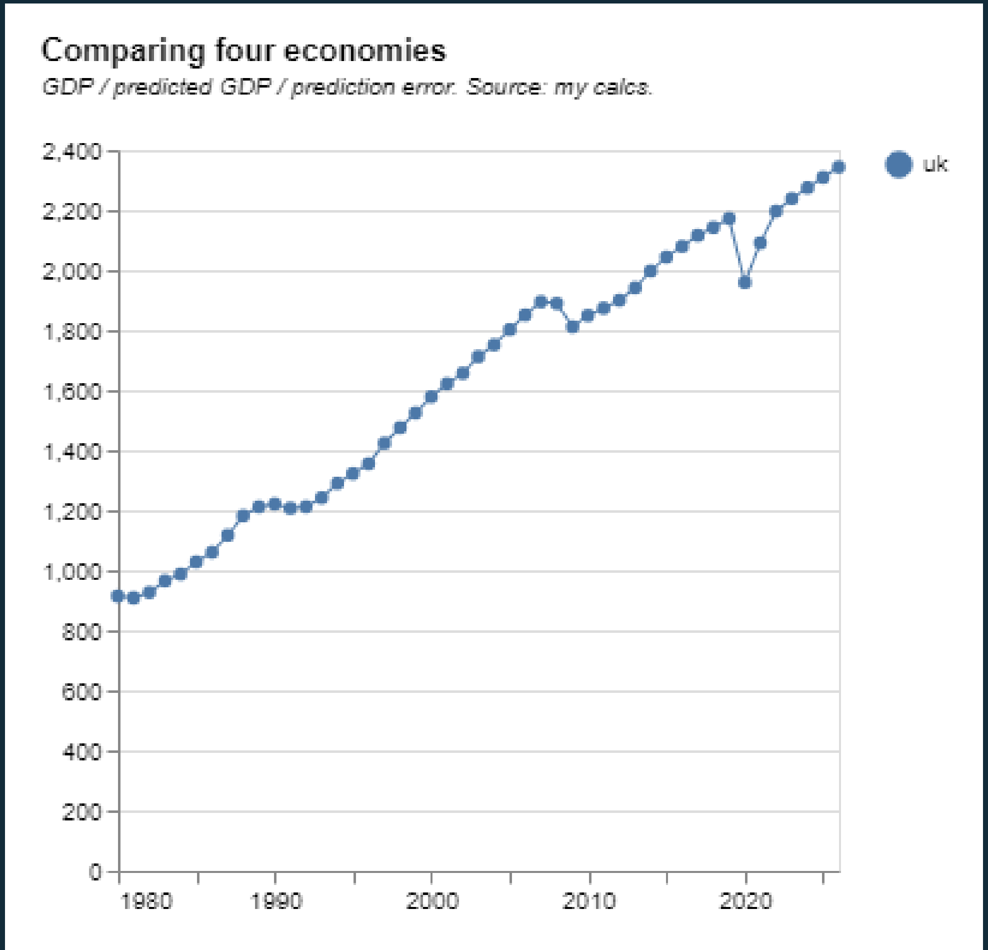
Comparing data series with independent y axes.



Vega code.

Removing a time trend.

Comparing volatility, or business cycles across countries.



Removing a time trend.

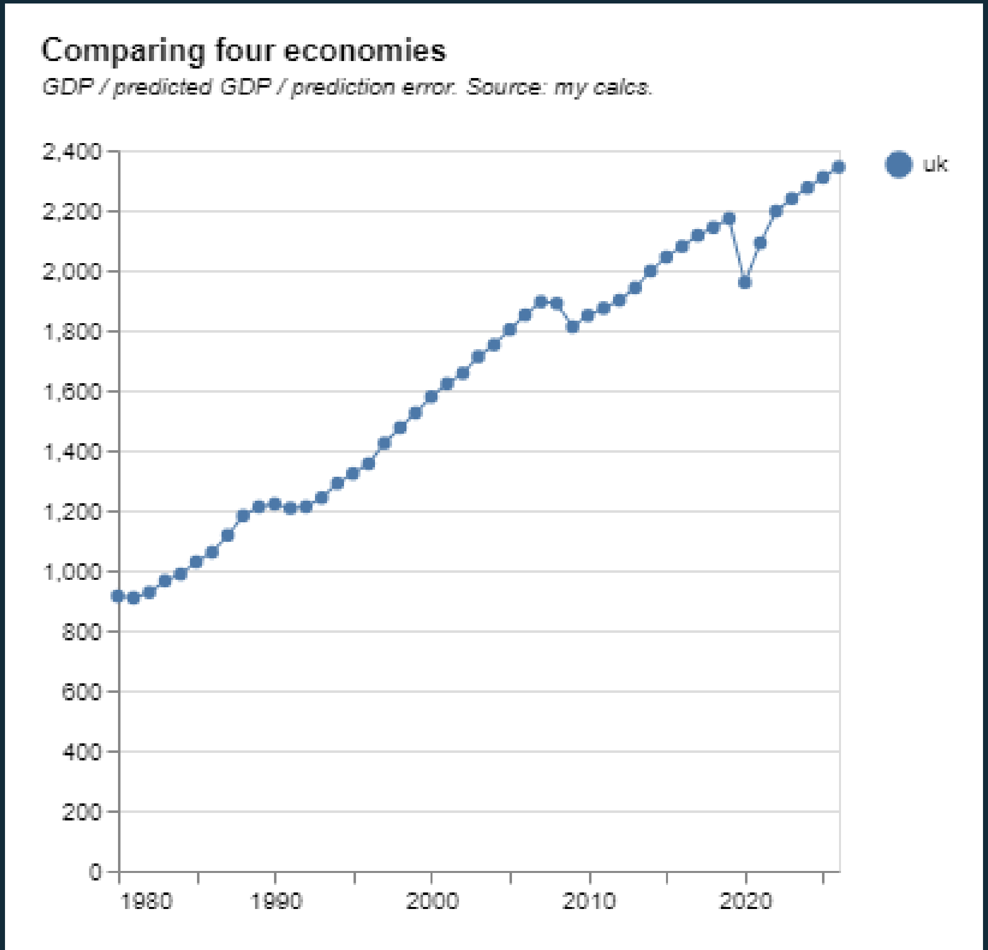
Using a simple OLS regression.

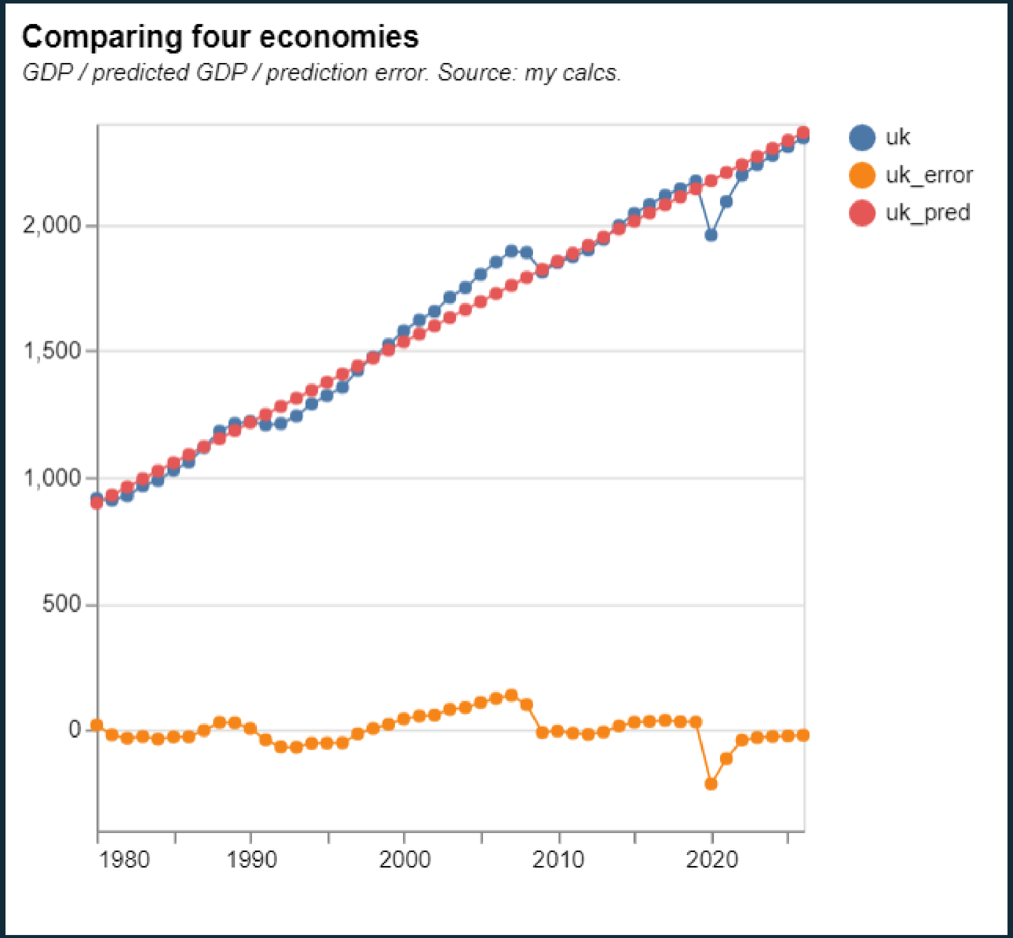
```
***RICHARD DAVIES
***DATA SCIENCE 2022
***REMOVING TIME TREND FROM DATA:

//Open data
import delimited "C:\Users\hi19329\Documents\GitHub\RDeconomist.github.io\data\fourCountriesGDP.csv", encoding(ISO-8859-9)

//For each country: regress, predict, calc errors
foreach i in uk kenya japan congo_dr{
regress `i' year
predict `i'_pred
gen `i'_error = `i'-`i'_pred
}

//Export the data:
export delimited using "C:\Users\hi19329\Documents\GitHub\RDeconomist.github.io\data\fourCountriesGDP_trends.csv", replace
```

Code for this Stata Do file. Here

Removing a time trend.

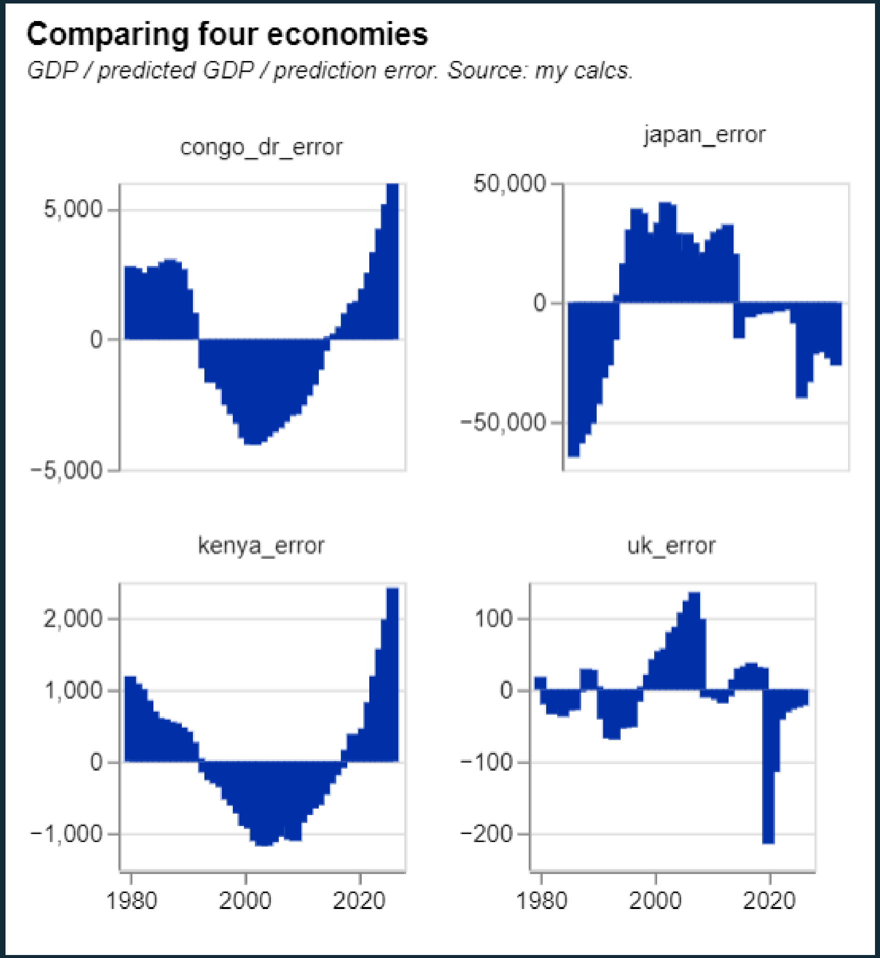
Using a simple OLS regression.

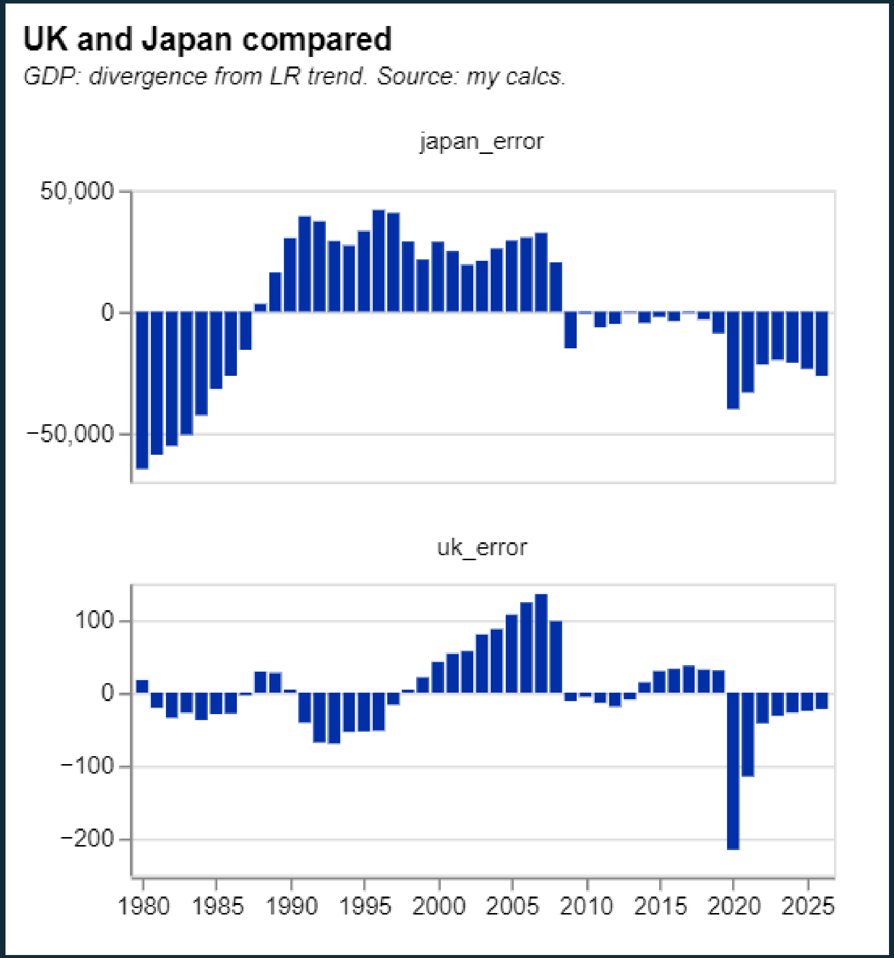




Removing a time trend.

Comparing the deviations from an underlying trend.

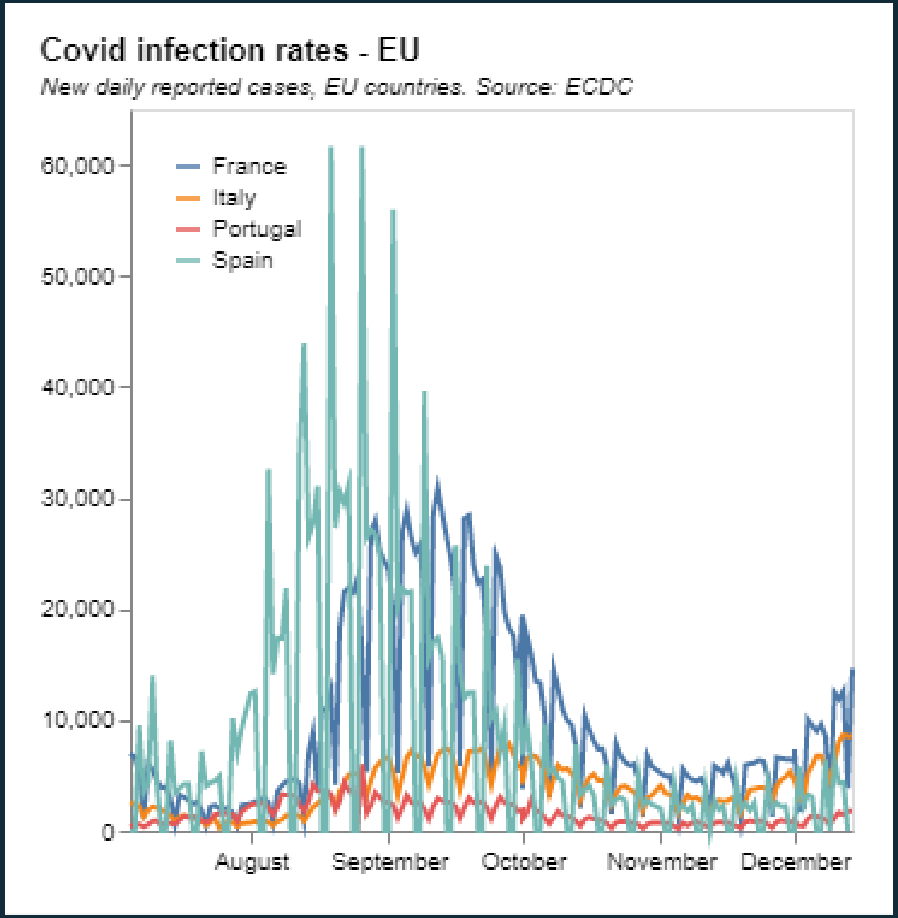


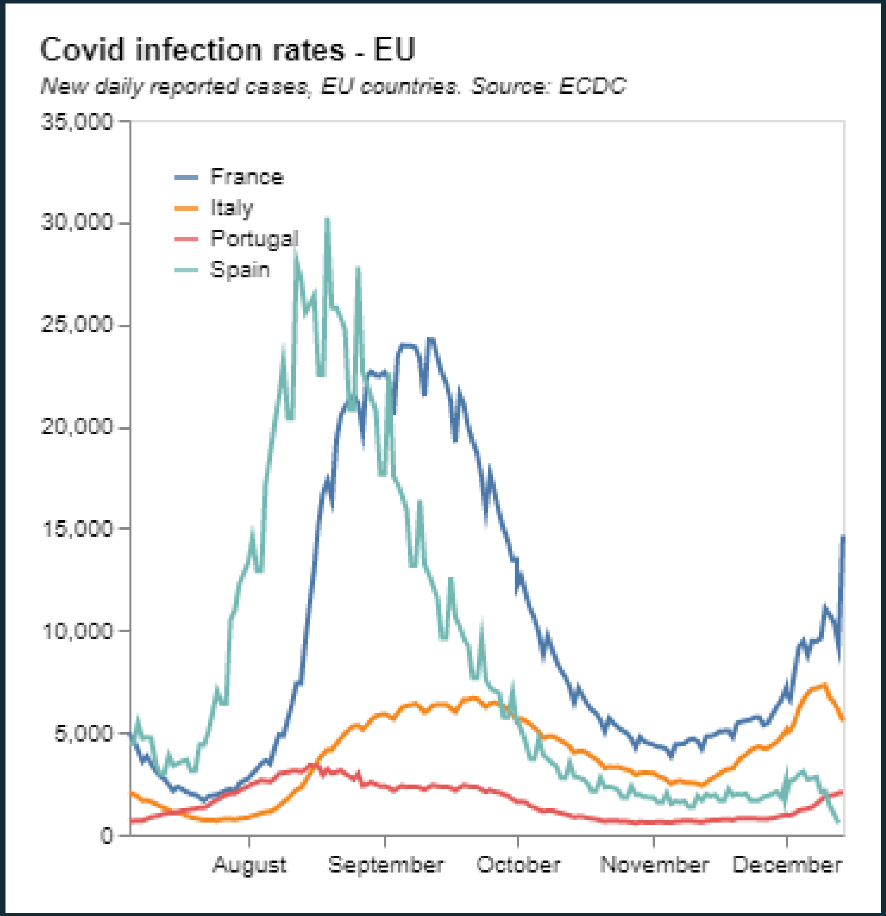


Code for these Vega charts: Here

Filtering and smoothing.

Using a moving average (MA)

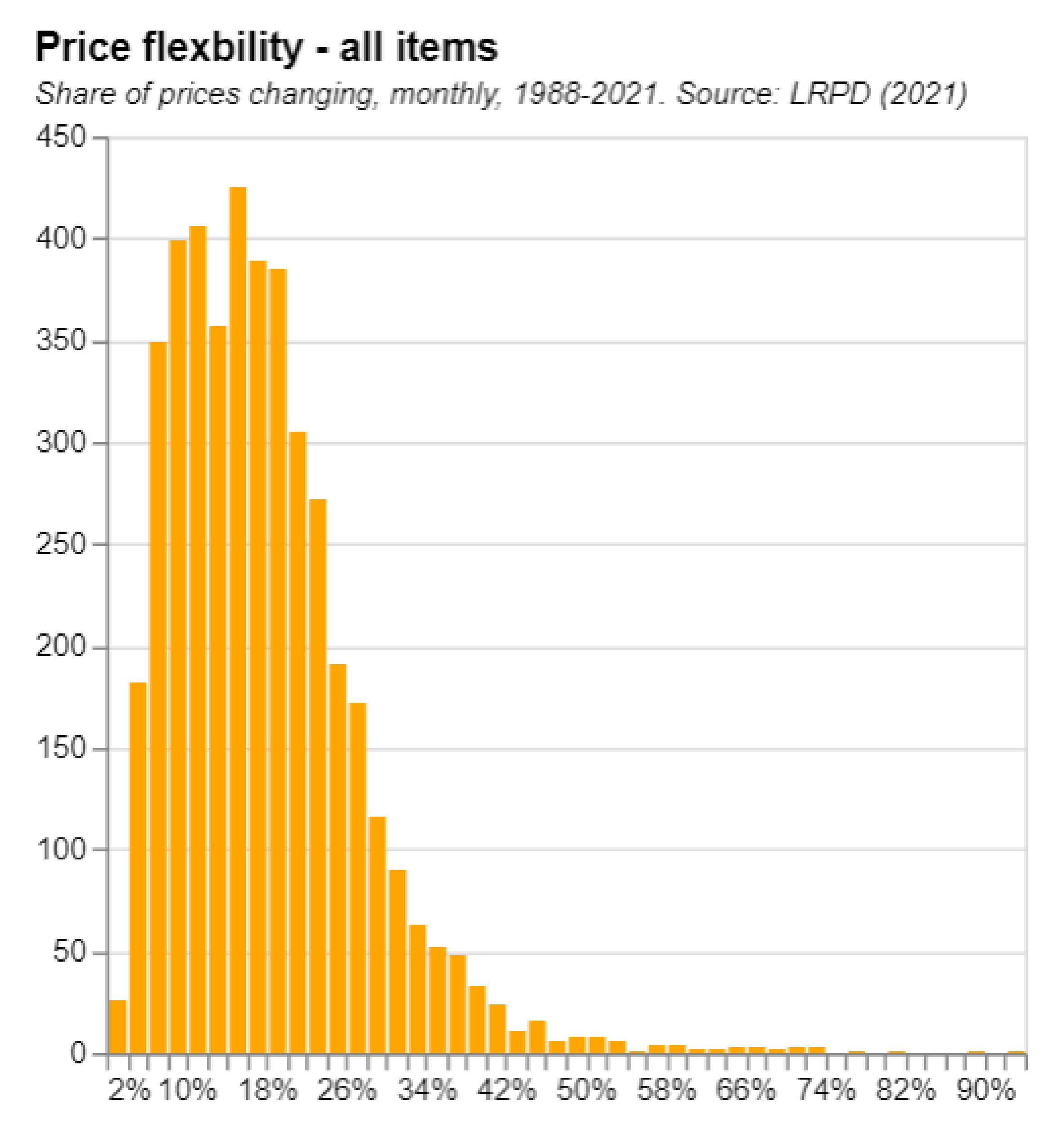




Vega code

Distributions 1.

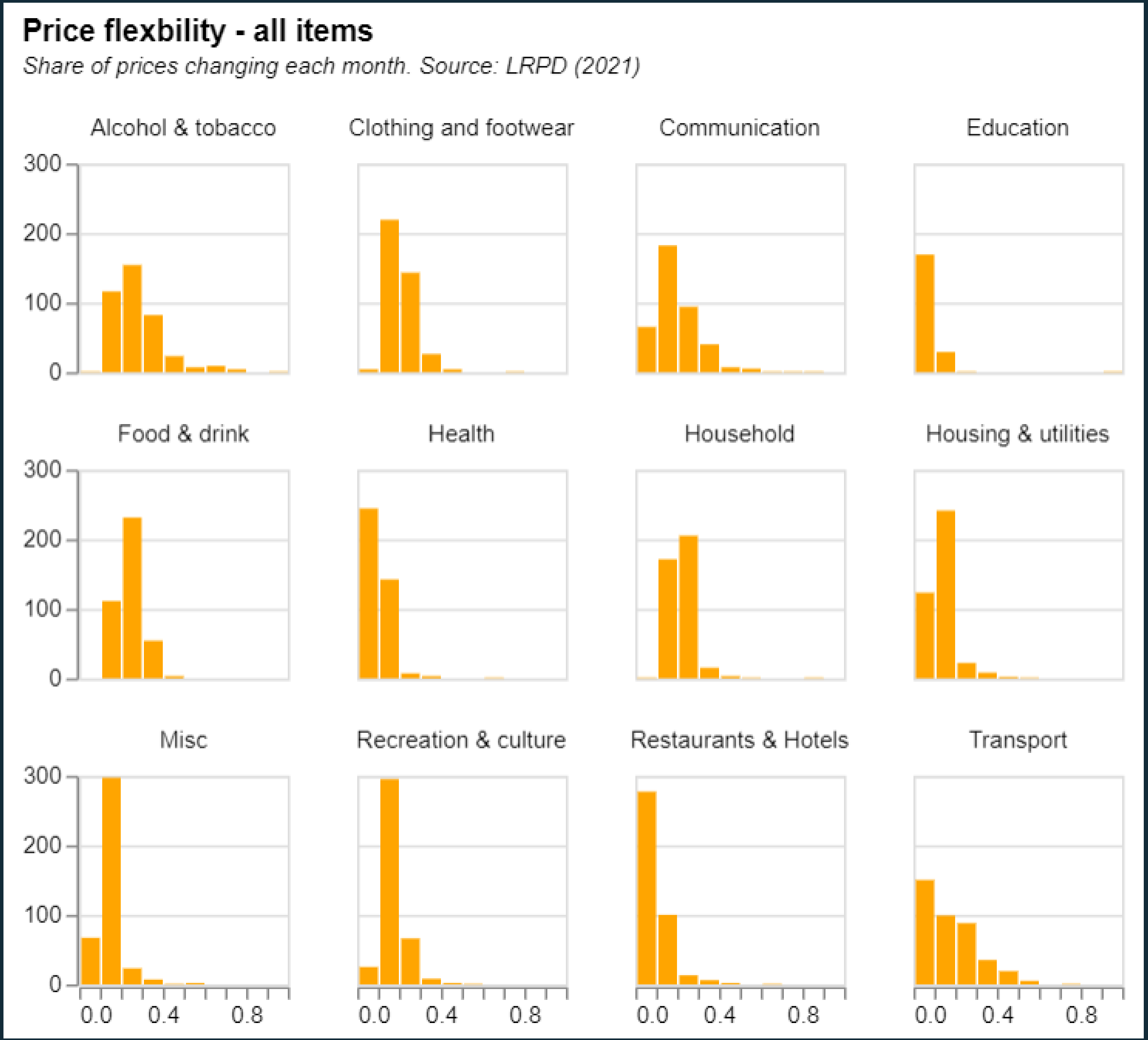
Simple histogram



[Code]

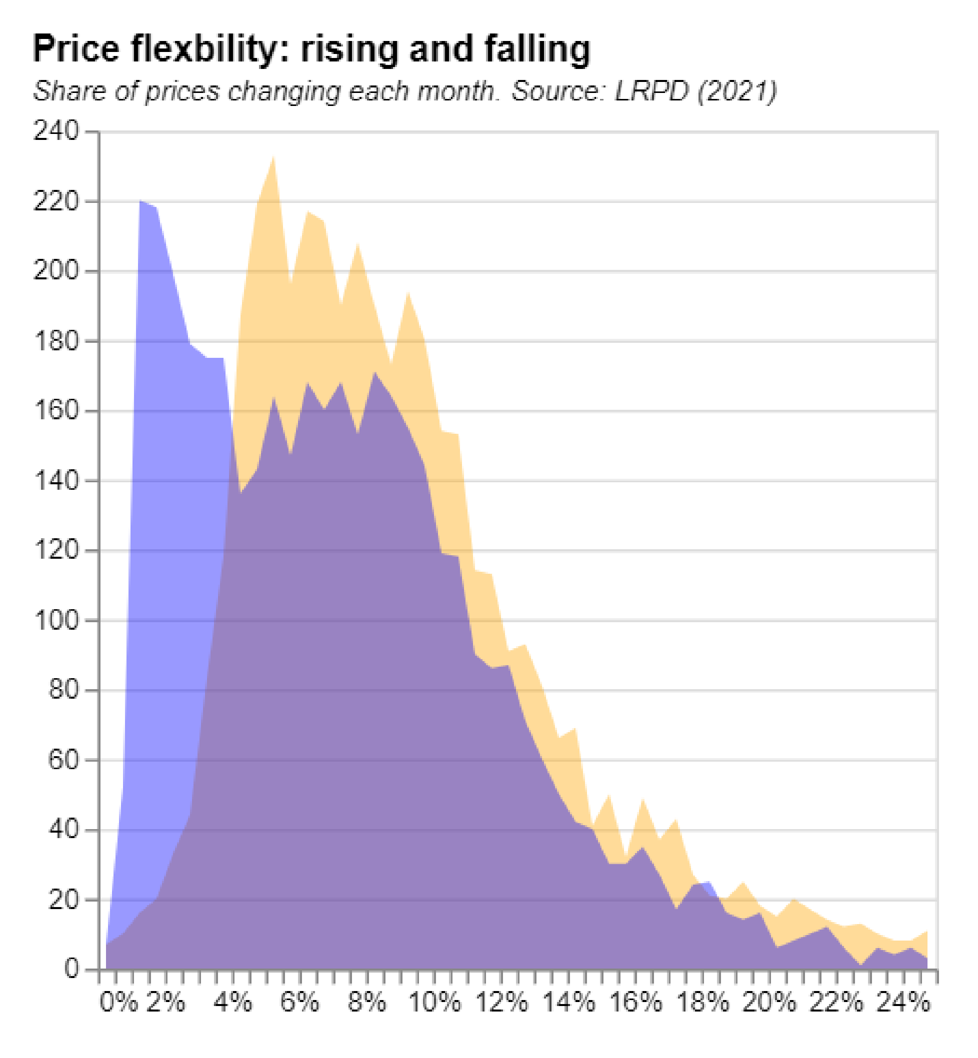
Distributions 2.

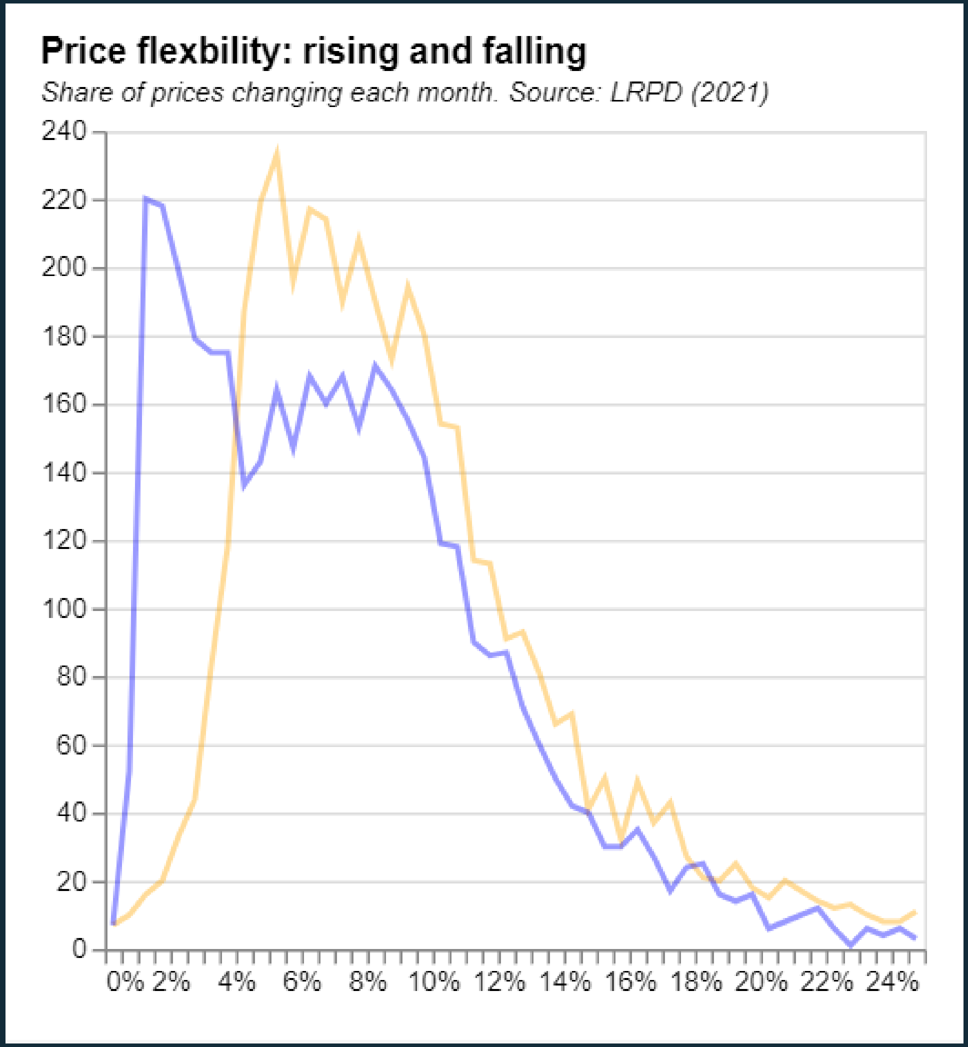
Comparing distributions with a trellis



Distributions 3.

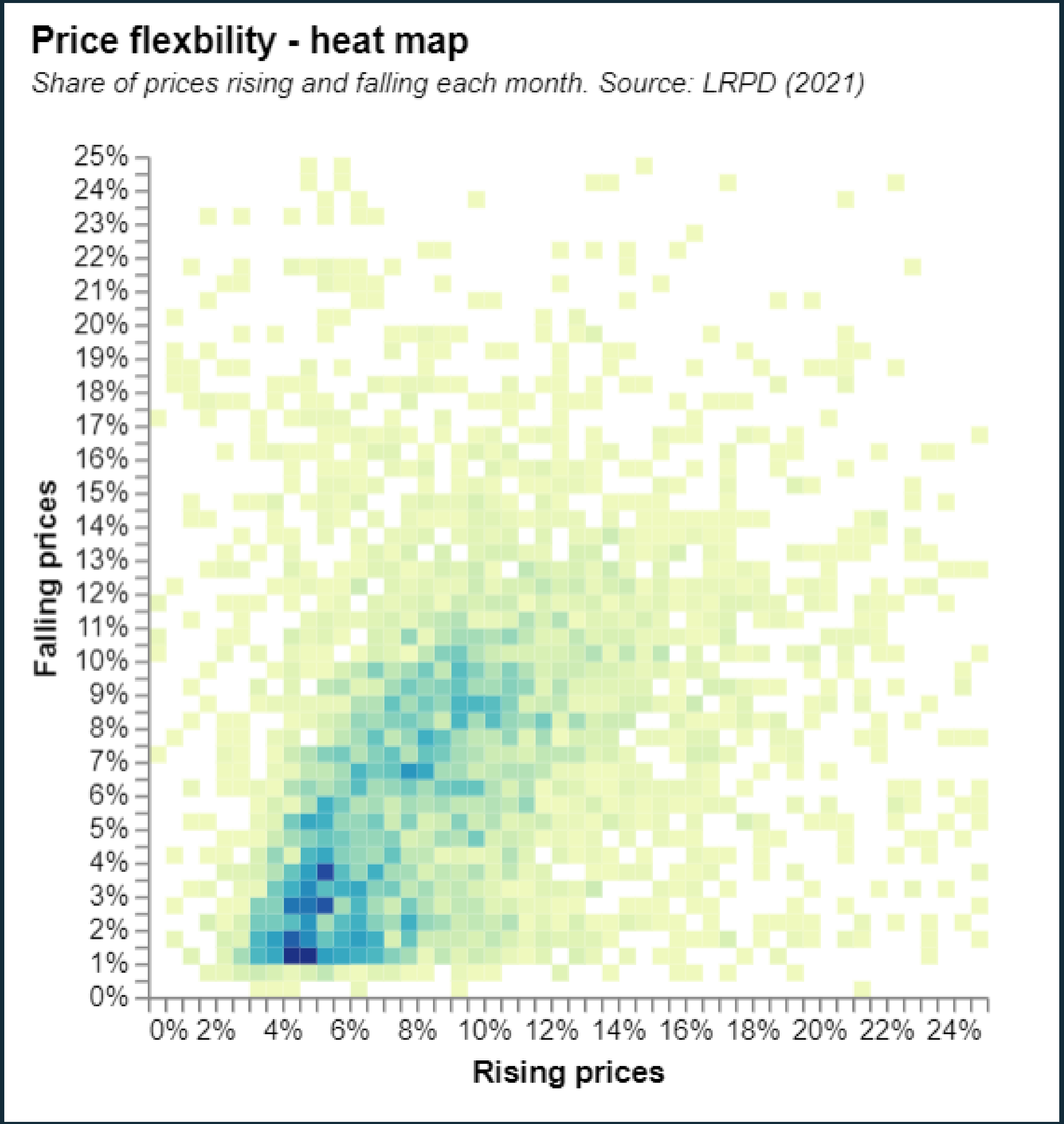
Estimating a distribution and comparing on the same chart.





Distributions 4.

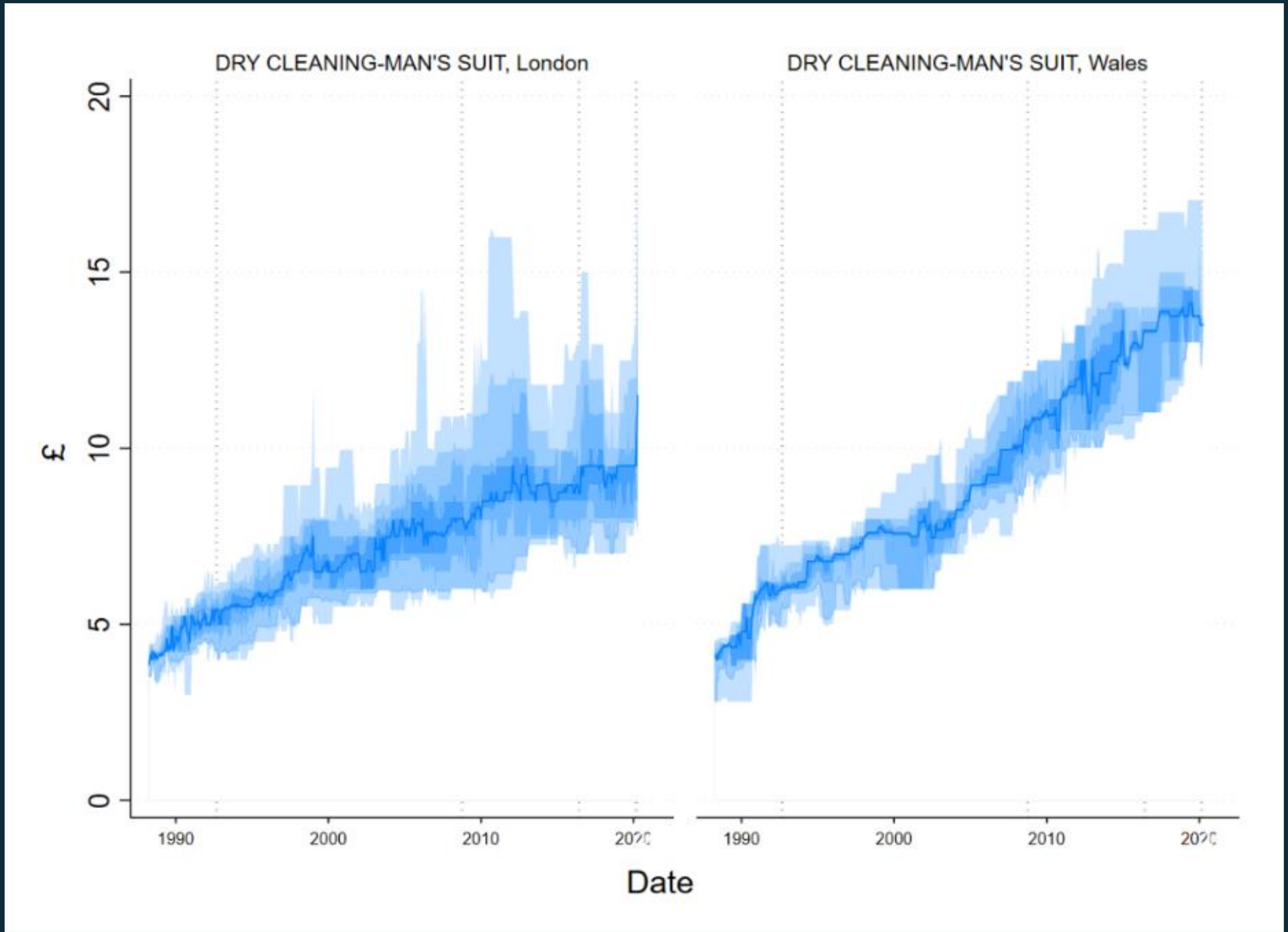
Joint distributions – using a heat map.



Code

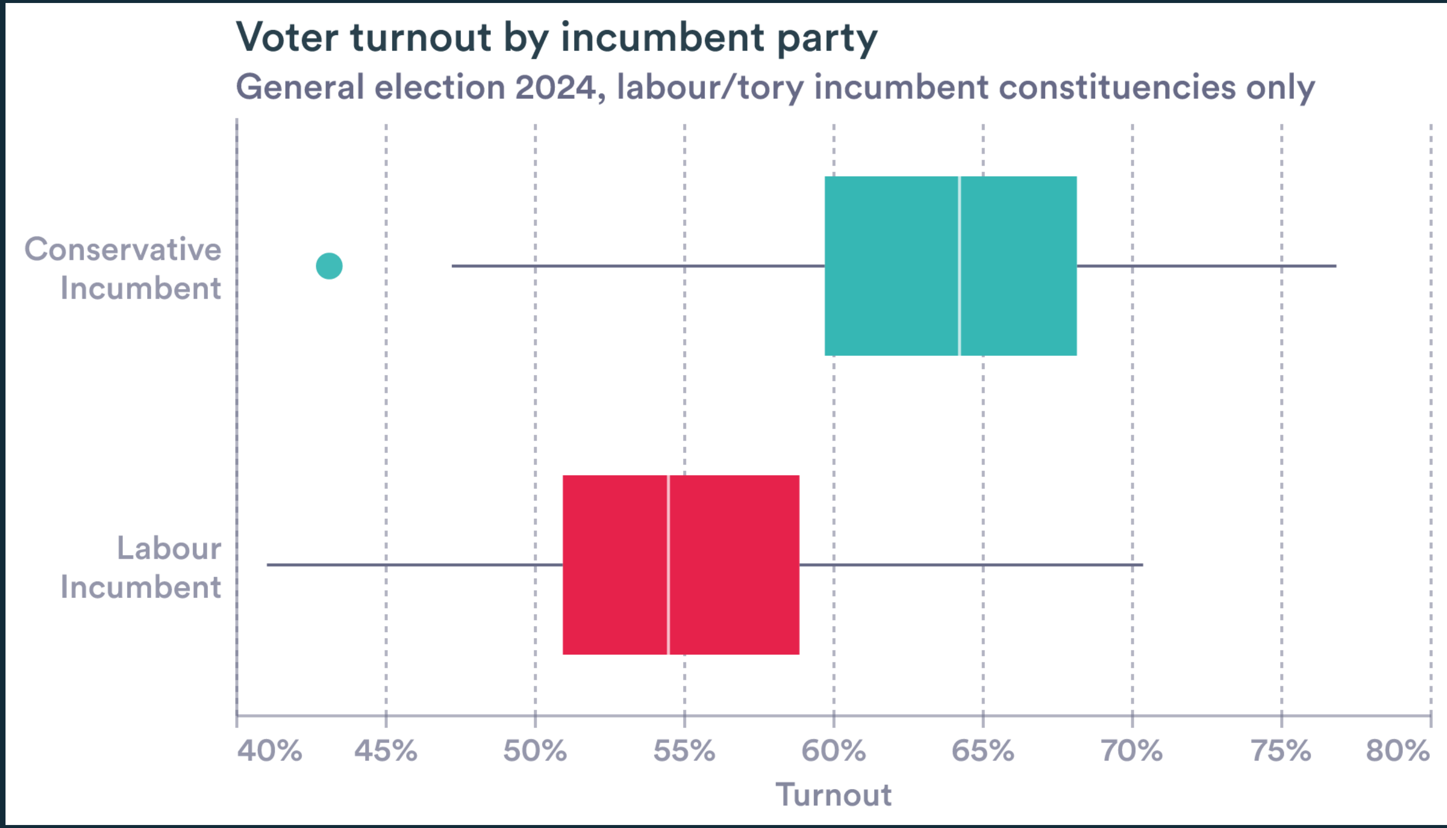
Distributions 5.

Distributions over time, using a swathe or area chart.



Distributions 6.

Comparing central tendency and spread, using a box and whisker plot (boxplot).



Code (fig 7)

'Tukey' box plot shows interquartile range (Q1-Q3) as box, with 1.5x IQR added to Q1 & Q3 as whiskers, and outliers as points. Median shown at box centre line.

Causality.

Why have things changed?

Causal inference 1.

The big idea, and a reading list

What does it mean for one phenomena to *cause* another?

Causation requires:

- X happens before Y (temporal precedence)
- When X changes, Y changes (covariation)
- No other explanation for the relationship (ruling out alternatives)
- A mechanism linking X to Y (how does it work)

Correlation: Countries with more economists have higher GDP
Causation: Does hiring economists boost GDP? Or do rich countries hire more economists?

Reading for deeper understanding of causation:

- Casual Inference: the Mixtape – Cunningham
- Causal Forests and Treatment Heterogeneity – Davis and Heller
- Causal Inference: What if? – Hernán and Robins

Causal inference 2.

Why correlation ≠ causation, and the counterfactual problem

Correlation is NOT causation.

- Reverse causality
- Confounding variables (interaction effect of third variable)
- Spurious correlation (coincidence)

Establishing causality is more difficult. We need a counterfactual.
"What would have happened if we hadn't implemented the policy?"

Fundamental problem: We can never observe the same person/place both with and without treatment at the same time.

Solution: Use clever research designs to create approximate counterfactuals.

Example: Does job training increase earnings?

- We observe trainees earn £5,000 more after training
- We can't observe what they would have earned without training

Maybe they were more motivated and would have earned more anyway. Maybe everyone's earning rose.

Causal inference 3.

Some of the key techniques to understand

Gold standard: Randomised controlled trials (RCTs)

- Randomly select treatment and control groups and observe differences.
- Key: Groups should be identical on average if they are truly randomly selected
- For many questions RCTs are practically (or ethically) impossible.

Quasi-experimental designs (when RCTs aren't possible).

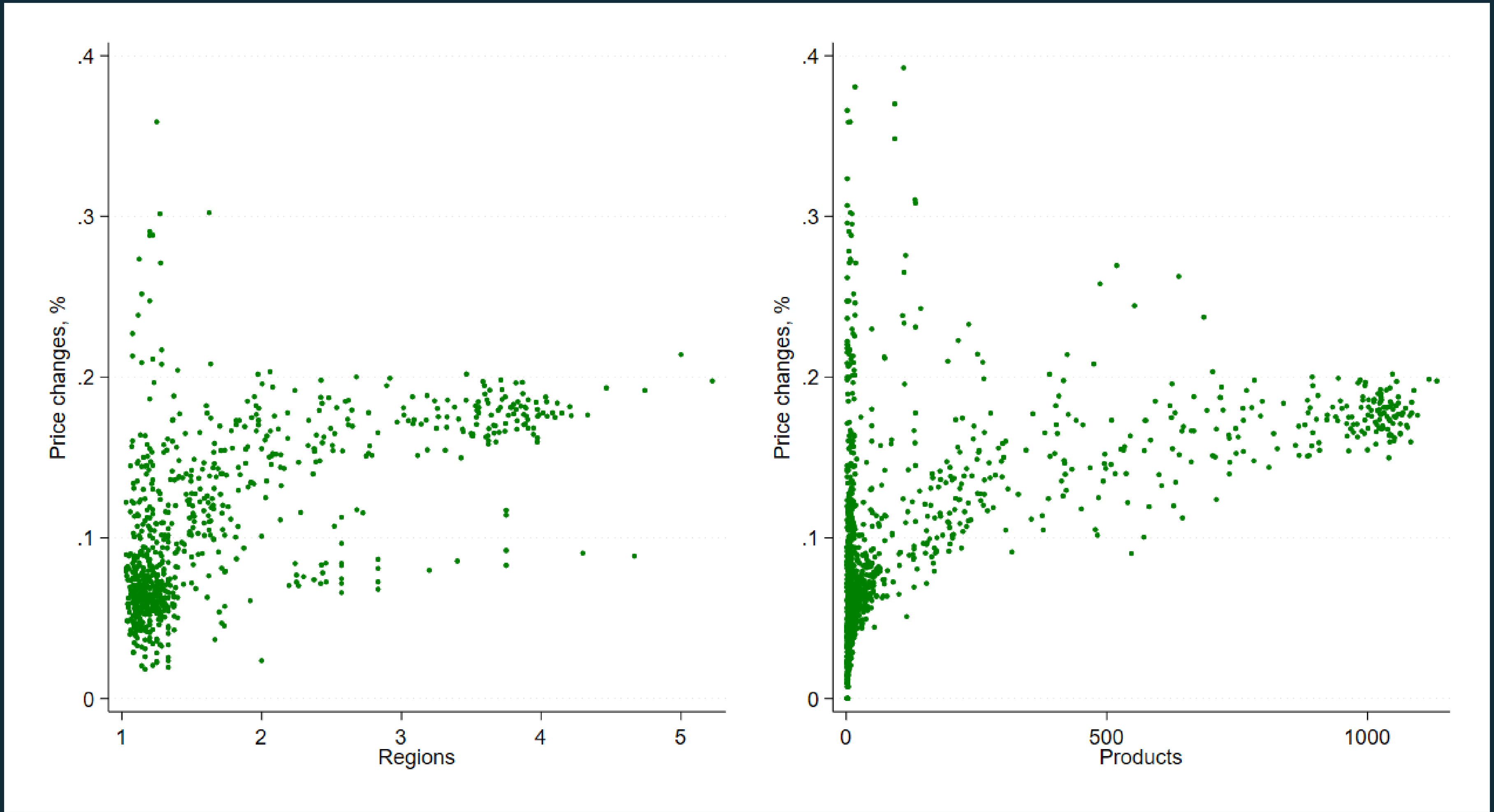
1. Matching / propensity score methods
   - Find 'control units' similar to treated units on observable characteristics
   - *Example*: Compare cities that adopted bike lanes to similar cities that didn't
2. Instrumental variables
3. Synthetic controls
4. Event studies
5. Difference-in-difference
6. Regression discontinuity

None of these designs are perfect.

Threats to causality should be considered in each design.

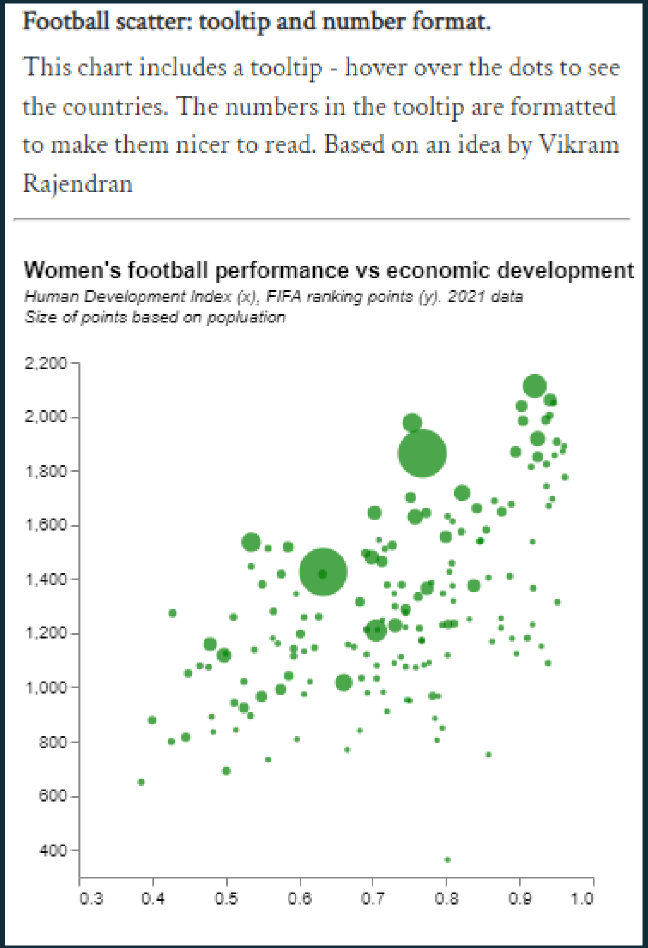
Correlation.

Isn’t causation. But it instructive.



Correlation – many variables.

Bubble charts / or facets of scatter charts



Code

Event studies.

Creating a window around an event

The idea

- Use data before and after an event to assess its impact

The problem

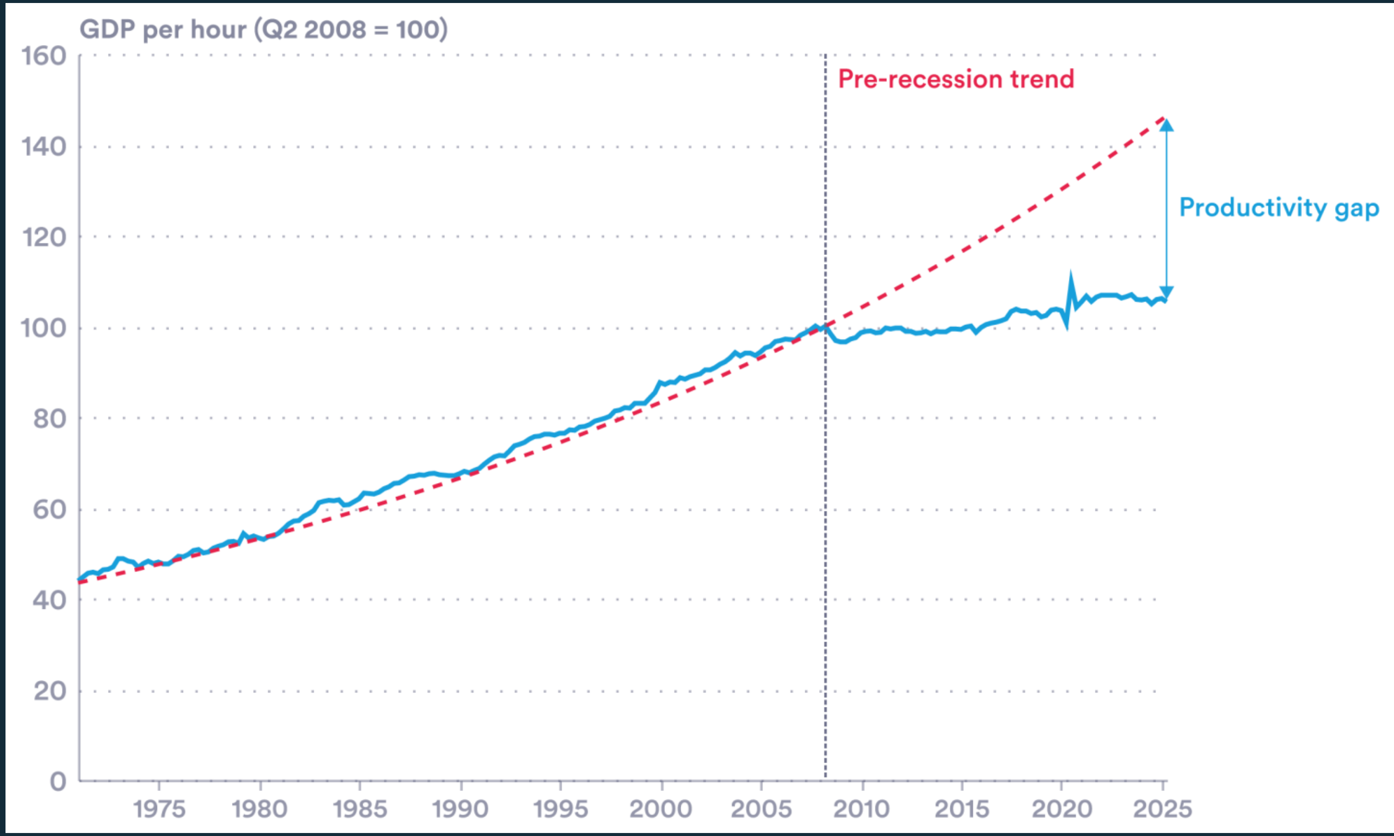
- Was the change going to happen anyway?
- Could your study be like a superstition? Lucky football socks example.

Overcoming this

- Try to establish some pre-trend.
- Use this to predict a counterfactual future

Event study.

Examples: UK business lending / UK productivity



Before the 2008 Financial Crisis, UK annual productivity growth averaged 2.27%

Since the financial crisis, it has averaged 0.45% annual growth.

In 2025, UK productivity was 28% lower than if it had continued growing at pre-GFC rates.

DiD - theory.

Theory and conditions to check

The idea

- DiD exploits natural experiments where:
  - Some units receive treatment at a specific time
  - Other units do not receive treatment
  - We observe both groups before and after treatment

Treatment effect = (Change in treated group) – (change in control group)

The problem

- Critical assumption: both groups would have followed parallel trends over time.
- If trends weren't parallel before treatment, they probably won't be after, and our estimate is biased.
- (Other issues, e.g. anticipation, spillovers)

Overcoming this

- Visual trends check pre-treatment & formal pre-treatment tests
- Check against anticipatory effects

DiD – John Snow.

Study of Cholera - again

Region Supplier	Death Rates 1849	Death Rates 1854
Non-Lambeth Only (Dirty)	134.9	146.6
Lambeth + Others (Mix Dirty and Clean)	130.1	84.9

Note: Death rates are deaths per 10,000 for the 1851 population, from Snow (1855).

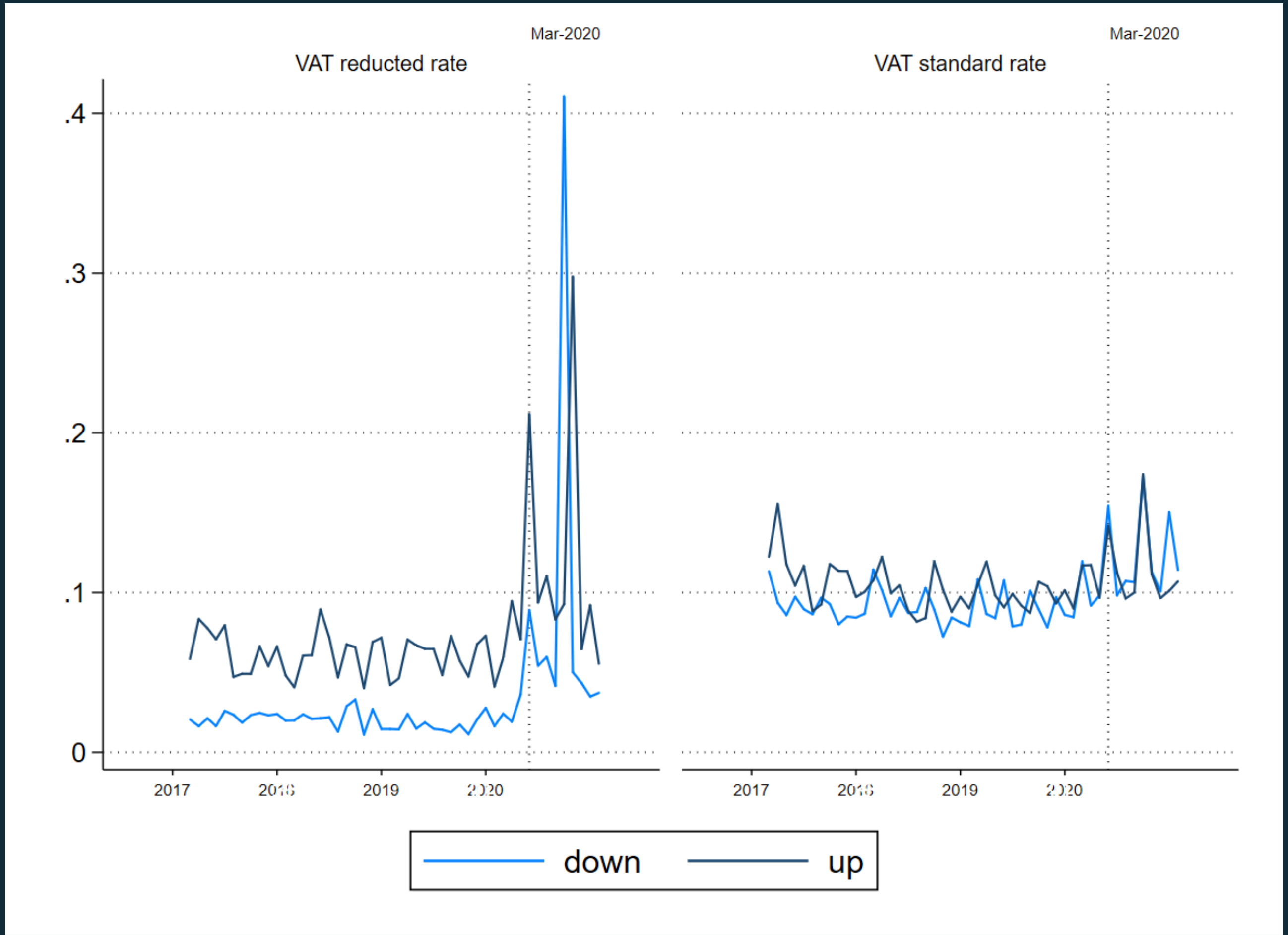
Coleman, Thomas. 2019. "Causality in the Time of Cholera: John Snow as a Prototype for Causal Inference." *Available at SSRN 3262234*.

https://johnsnow.matrix.msu.edu/documentUploads/15-78-52/15-78-52-22-1855-MCC2.pdf

https://theeffectbook.net/ch-DifferenceinDifference.html

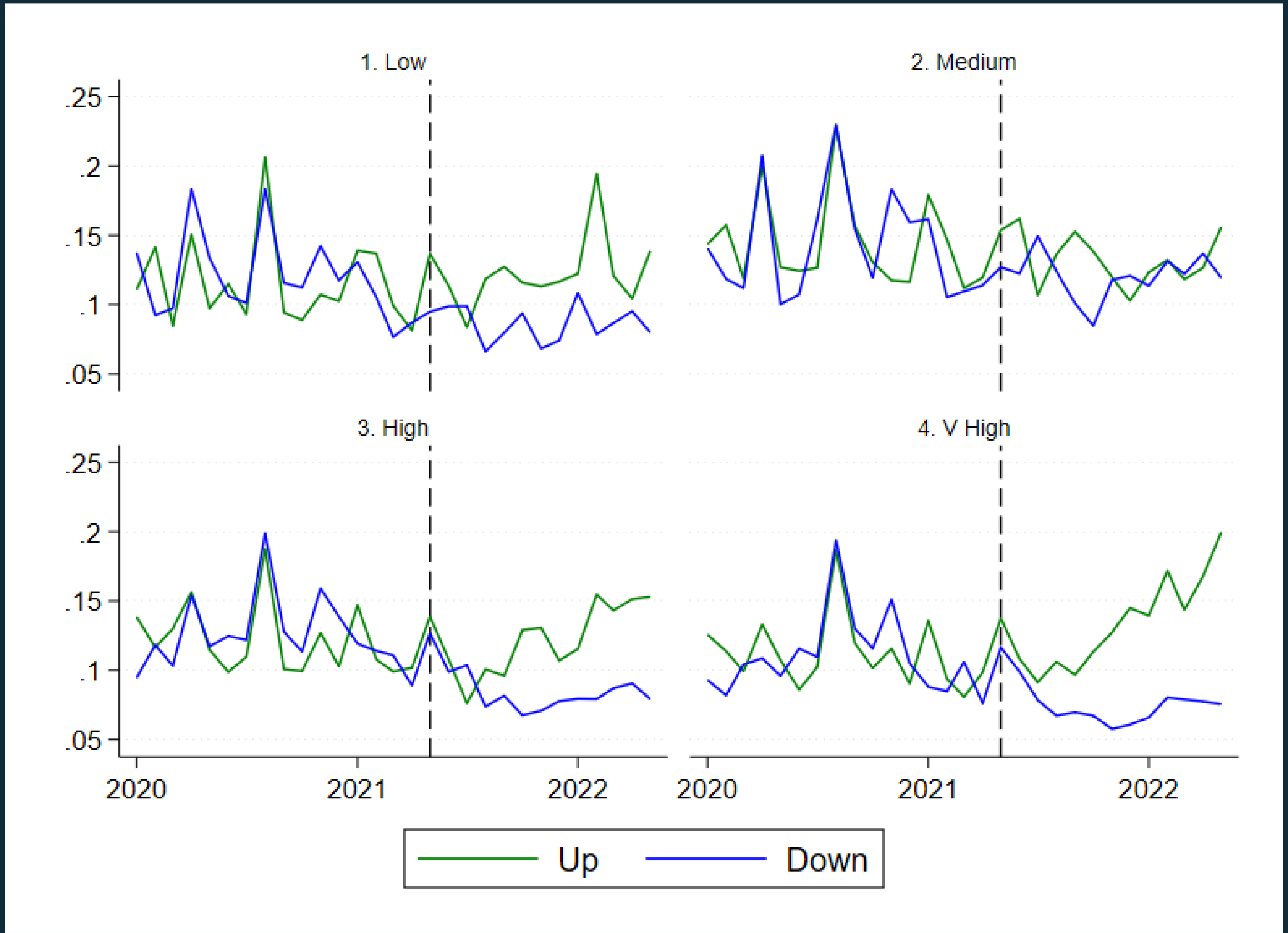
Diff-in-Diff.

Charts motivated by DiD techniques



Diff-in-Diff.

Charts motivated by DiD techniques



Notes: Figures show the proportion of prices, analysed at the firm-good-region level, which are rising or falling compared to their previous monthly value. Consumer items are split by EU exposure with low (0-40%), medium (40%-60%), high (60%-80%) and very high (80%+) groupings. Vertical lines are drawn for May 2021.

Discontinuity.

Regression discontinuity theory: exploiting arbitrary cutoffs

The idea

Treatment assigned based on a threshold in a continuous variable.
Units just above vs just below cutoff are nearly identical – except for treatment status.

Example: schools with >30 students must split into two classes

- 29 students = one large class (control), 31 students = two small classes (treatment)

Compare outcomes for schools near 30-student threshold.

The problem

- Key assumptions: No manipulation; only treatment changes.
- Cannot generalise far from threshold

When RD works

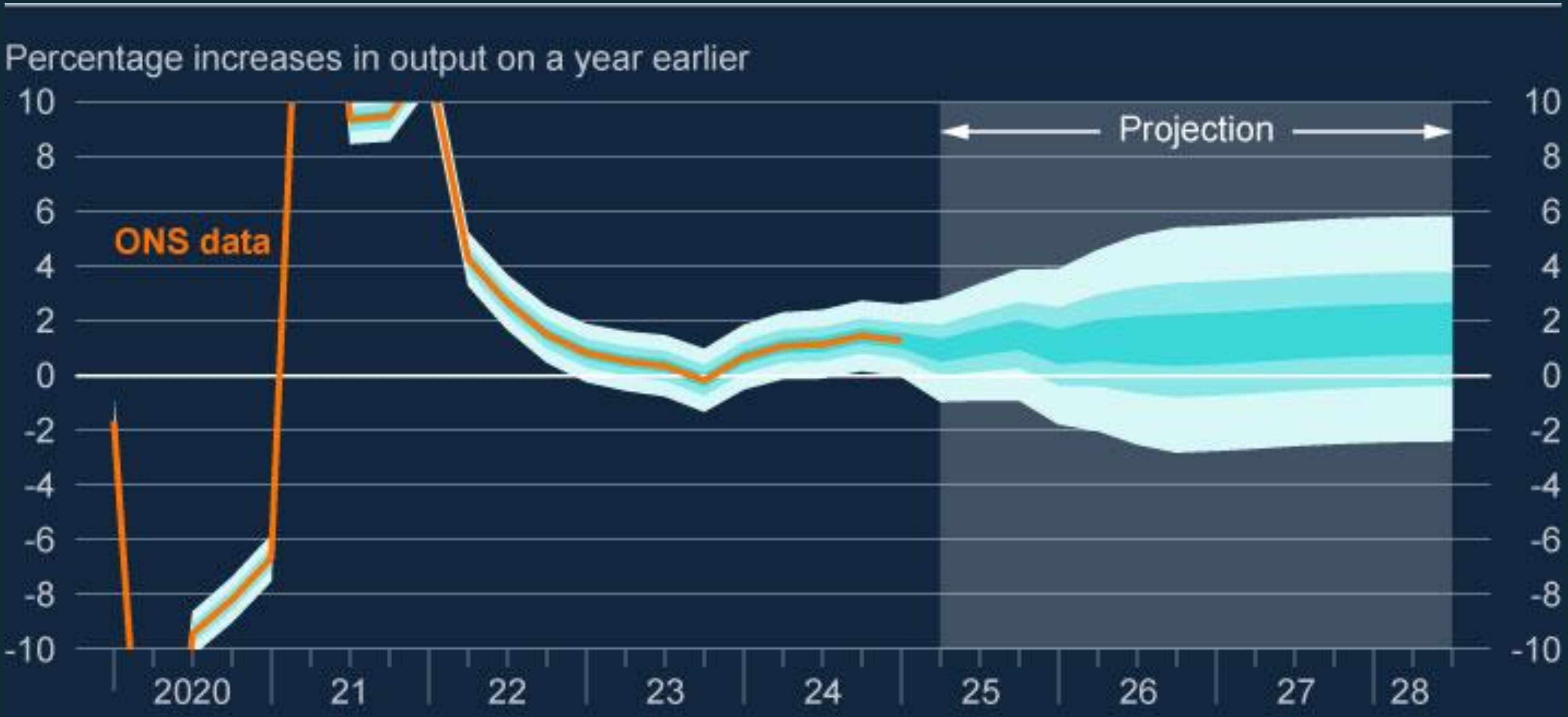
- Clear cutoff rule (e.g. age limits, test scores, geographic boundaries, firm size regulations)

Uncertainty.

Being modest about our analysis

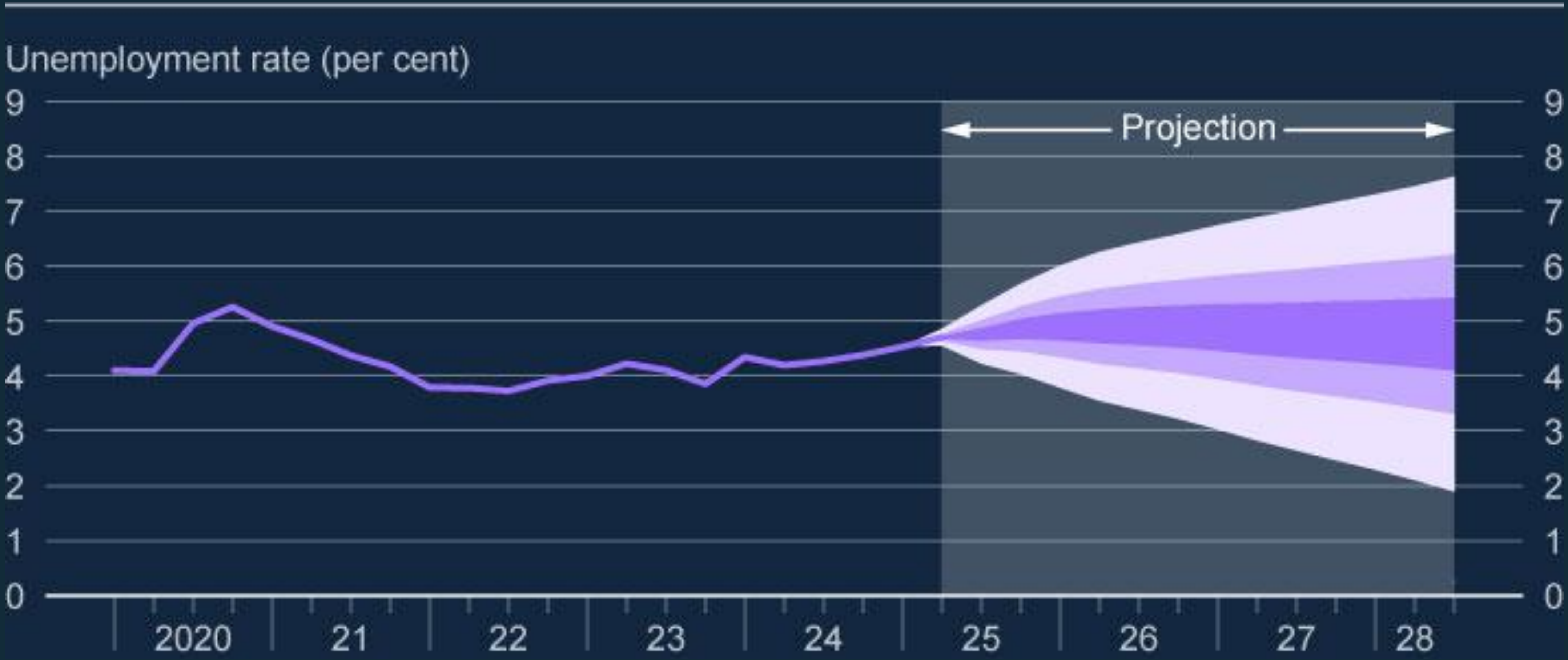
Scenarios.

How robust are results to changes in assumptions?



Scenarios.

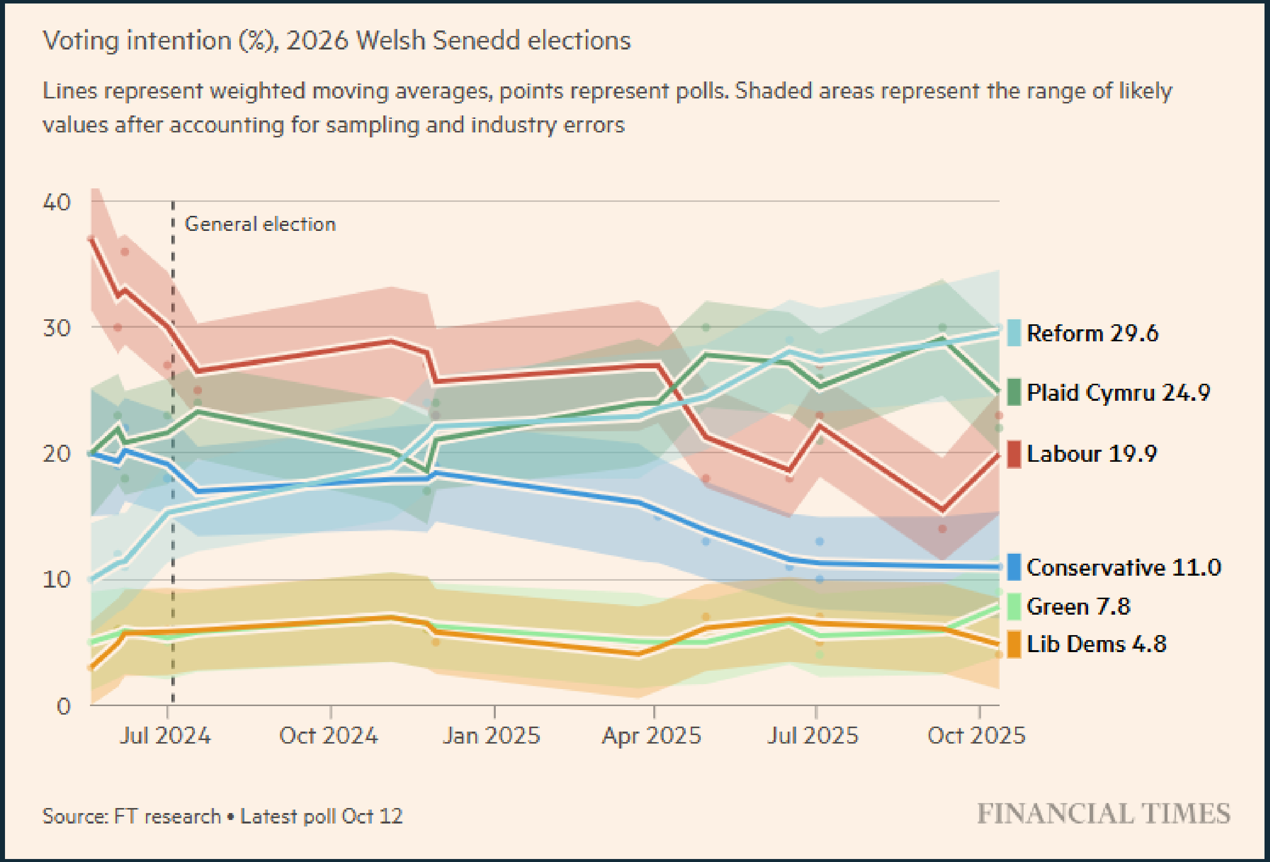
How robust are results to changes in assumptions?



https://www.bankofengland.co.uk/monetary-policy-report/2025/august-2025

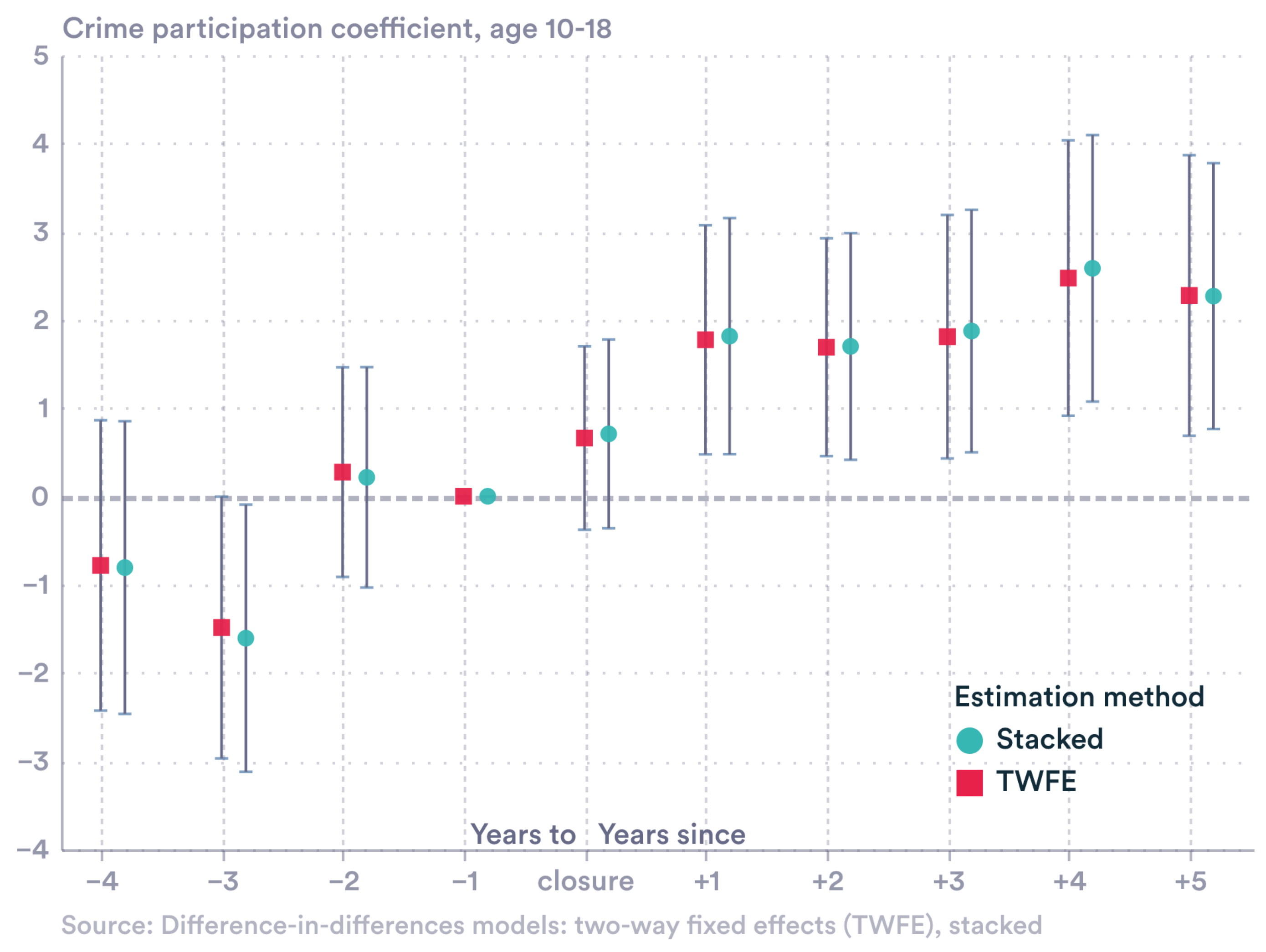
Error swathes.

Adding margins of error to line charts



Error bands.

Confidence interval, using error bands.



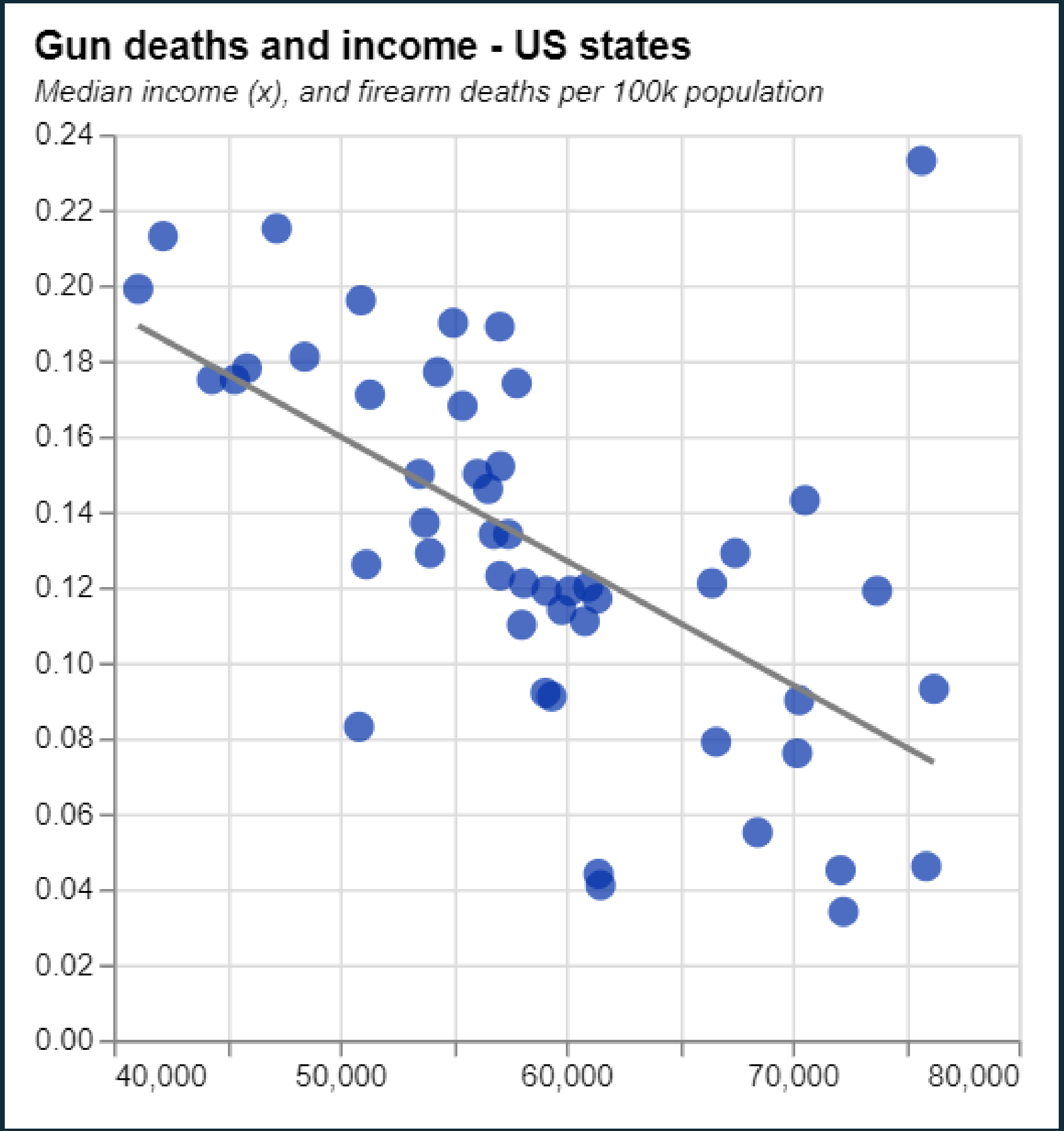
Code (fig 1)

9.3 Complex analysis.

Powerful tools, or black boxes?

Regression.

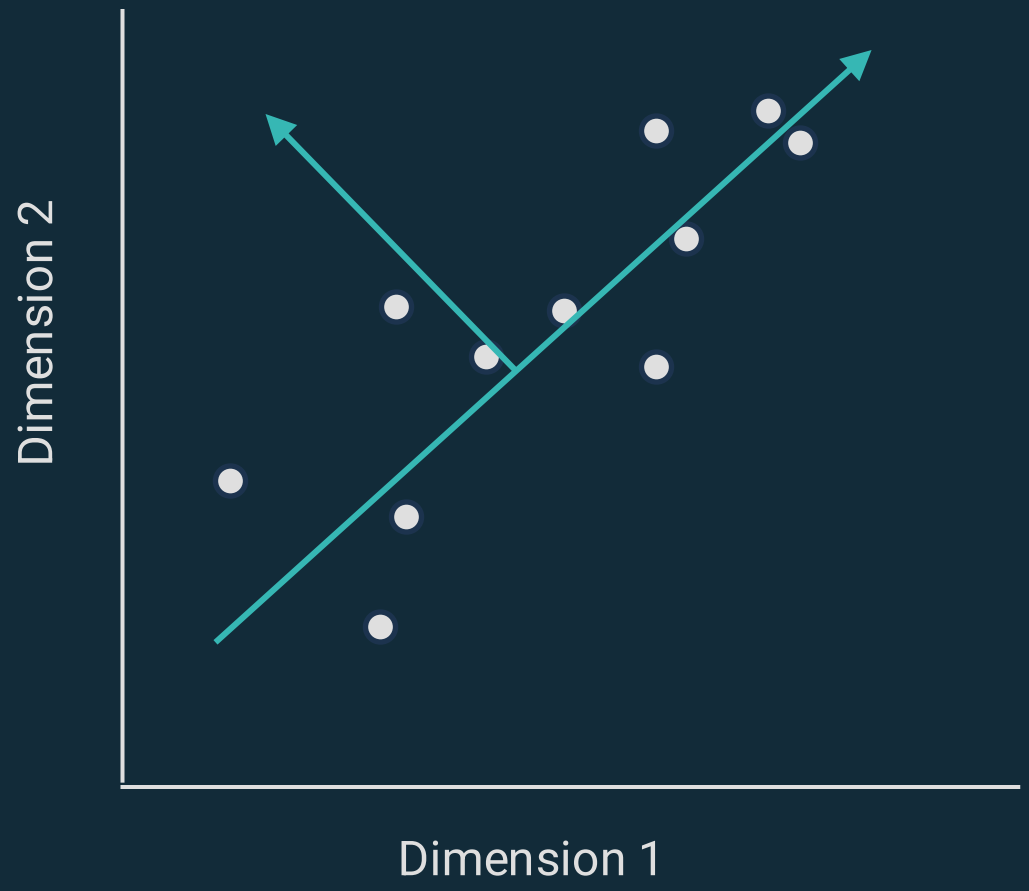
Adding lines of best fit to charts.



[Code]

Principal Component Analysis.

Dimensionality reduction



High dimension data is rich data.

It gives us a fuller picture of our subject.

But computational complexity can increase exponentially with dimensionality.

Principal component analysis transforms data into a new coordinate system, typically with fewer dimensions, to reduce complexity.

These new bases – the principal components – are chosen such that they preserve as much variation in the source as possible.

Embeddings.

Semantic representations

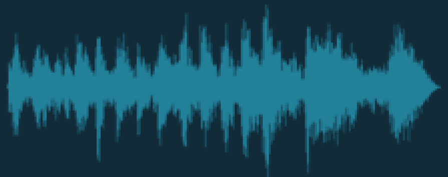
Text, audio, and images are very high dimension data.

To treat them as data, we need a way of representing their meaning numerically.

Embeddings are vector representations of an input's semantic meaning.

"Bread"

[0.016758177429437637, -
0.018055250868201256, -
0.000592194730415940 3, -
0.014877422712743282, -
0.014903363771736622,
0.002294197212904691 7, -
0.021557345986366272, -
0.02664187178015709, -
0.011440180242061615, -
0.03317911550402641,
0.01030524168163538,
0.02825024165213108, -
0.03330882266163826, -



[0.012685369700193405,
0.020688308402895927,
0.017471568658947945,
0.007646243087947369,
0.006939338520169258, -
0.025383710861206055,
0.007477623410522938, -
0.010888924822211266, -
0.026265719905495644,
0.00058814132353290 92, -
0.002655756194144487 4,
0.02905442565 67955, -
0.025682037696242332, -

[-0.010454405099153519,
0.04573477804660797,
0.002081801416352391 2, -
0.009124905802309513,
0.003043256467208266 3, -
0.029806727543473244, -
0.023943960666656494, -
0.003301049815490842, -
0.0017218638677150 01,
0.004740800242871046, -
0.003250788198783993 7,
0.015500017441809177,
0.018612992018461227

Embeddings.

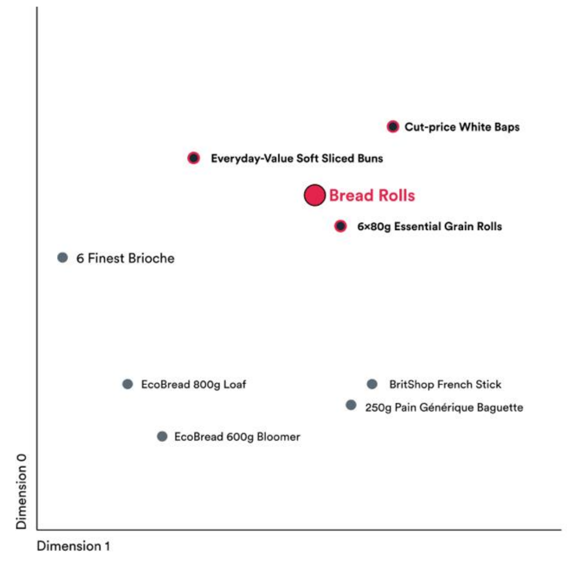
Comparisons

Generating embeddings allows us to compare the semantic meanings of data.

In our research, we use them as a first estimate of the similarity between products:

Figure 5: Forming a candidate shortlist with embeddings

A toy example

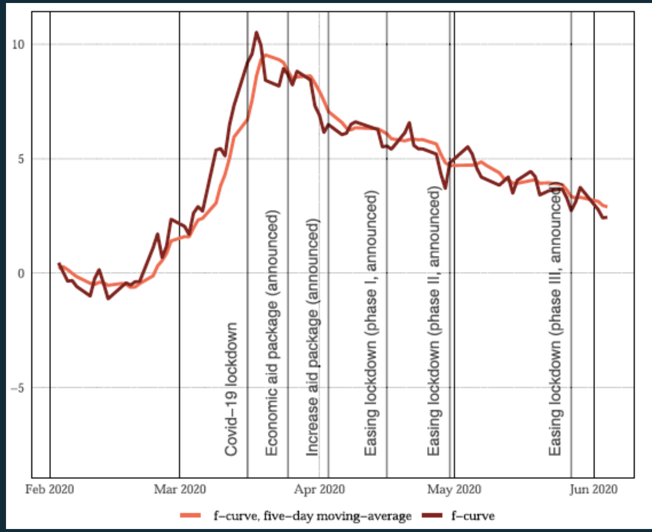


Notes: This shows a highly simplified representation of our use of embeddings to generate a product shortlist for a CPI item (Bread Rolls). Source: Authors

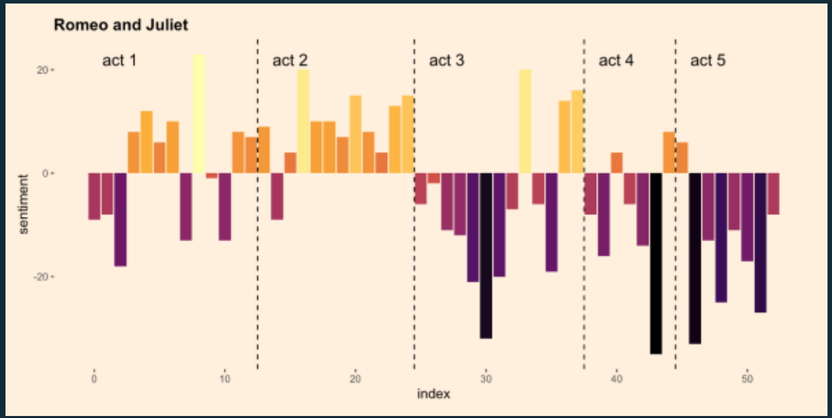
Faster, bigger, cheaper:
how AI can improve UK price data

Richard Davies, Finn McEvoy

Text as Data – Sentiment Analysis



Tracking sentiment on the Swiss economy. [Source](#).



Tracking the sentiment of Romeo and Juliet. [Source](#).

- Also called "opinion mining": using machine learning to systematically quantify the affective state (positive, negative, or neutral) of a piece of text.
- Common uses: customer reviews/brand monitoring, social media posts, news pieces on a given topic (e.g. financial markets.)
- Tools: Python packages NLTK, Texblob, VADER

More to follow.

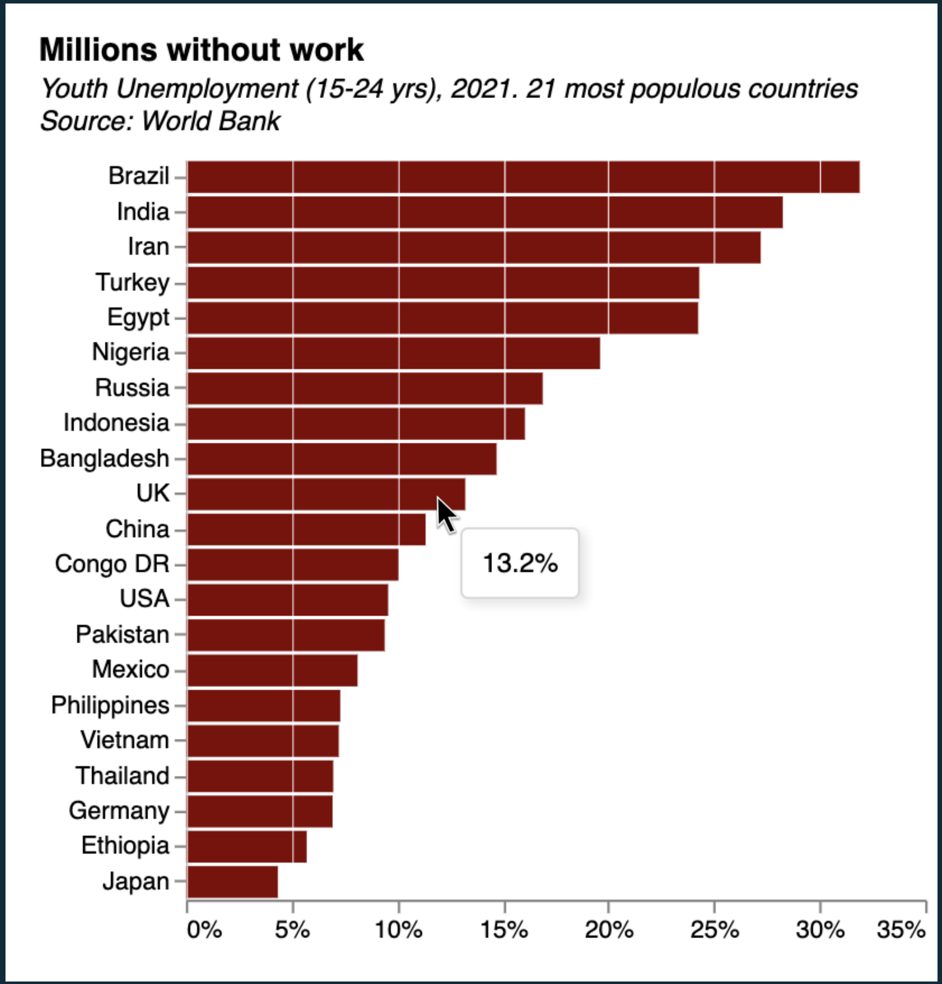
We will use these tools in our two AI weeks…

9.4 Interactive charts.

Letting the user find their own answers.

1 Tooltip.

Simple tooltip, single field



Add tooltip channel to encodings

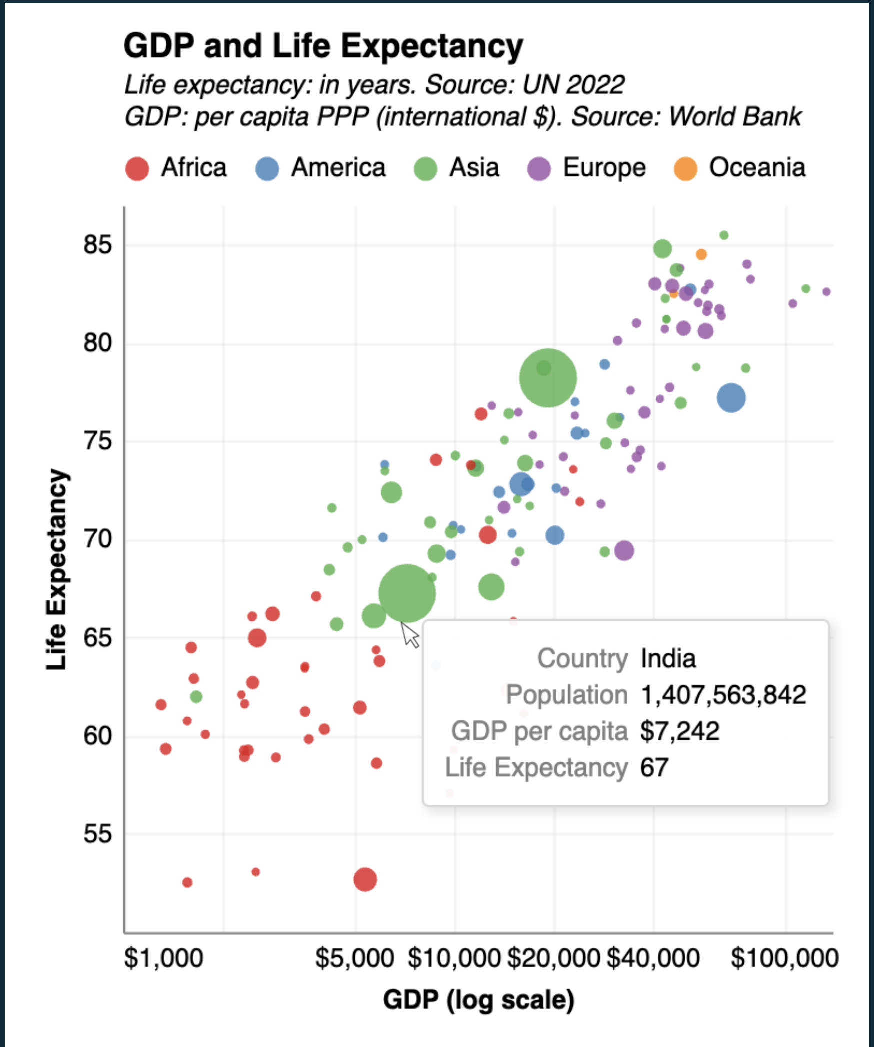
```
"encoding": {
  /// x, y
  "tooltip": {
          "field": "unempRate",
          "format": ".1%”
     },
}
```

Single object in tooltip will show values for one data field.

Automatically appears when hovering over encoded datapoints.

2 Tooltip.

Simple tooltip, multiple fields



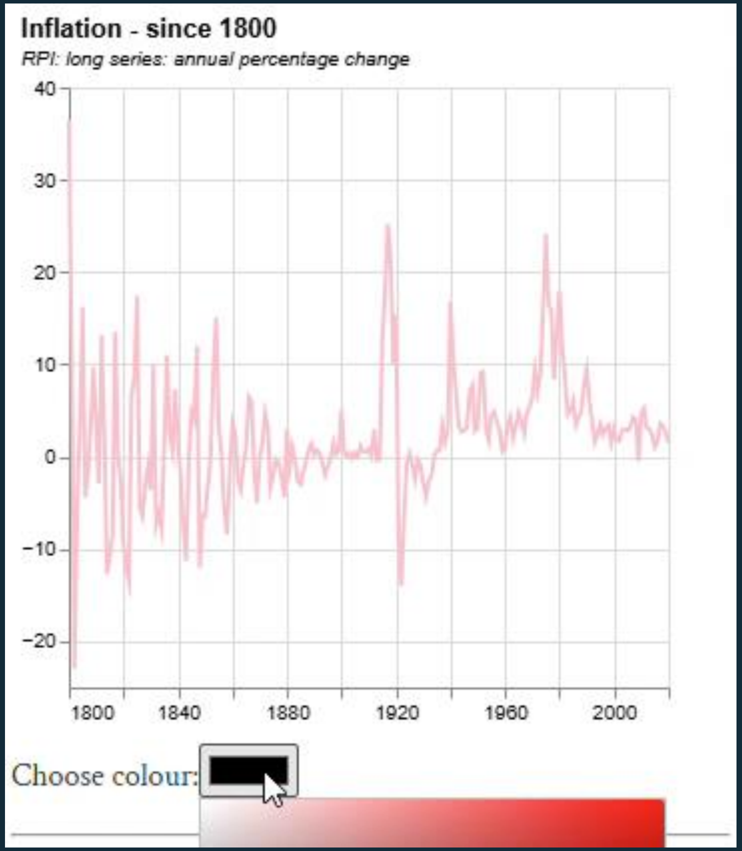
Add tooltip channel to encodings

```
"encoding": {
    /// x, y, color,
    "tooltip": [
        {"field": "Country"},
        {"field": "Population", "format": ",d"},
        {
            "field": "GDP per capita (PPP)",
            "title": "GDP per capita",
            "format": "$,d"
        },
        {"field": "Life Expectancy", "format": "d"}
    ]
}
```

Pass an array (list) of objects to show multiple fields in the tooltip, each corresponding to a data field.

3 User input.

Picking a colour



A regular line chart

With a user-selected parameter

A name for that parameter

A starting value

A way to choose that parameter

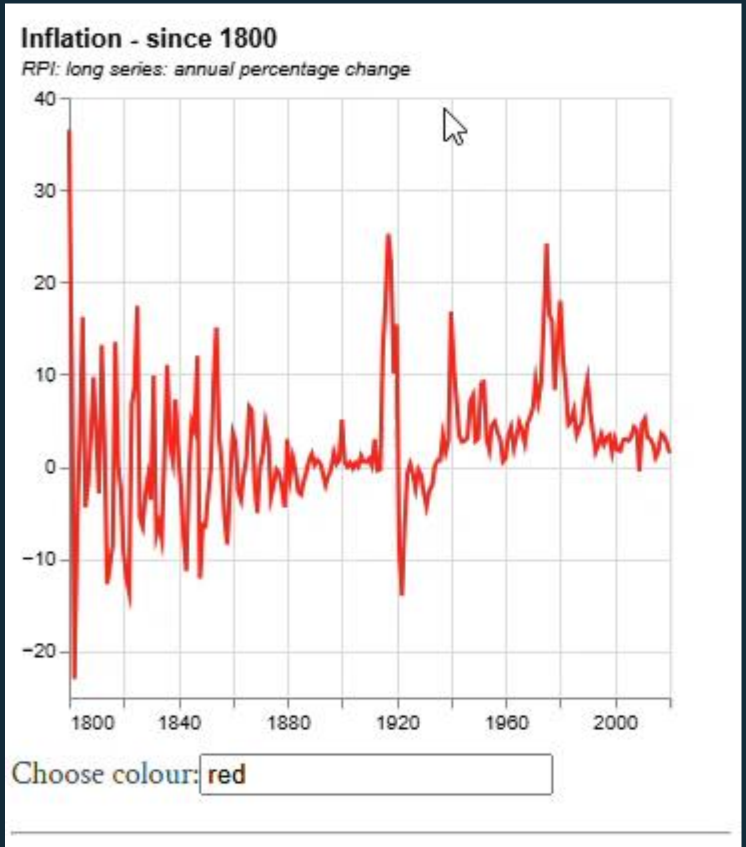
And some text for the page

And a color encoding that reflects the user's selection

```
"params": [{
        "name": “colourPick",
        "value": “red",
        "bind":{
                "input": “color",
                "name": "Choose colour:”
        }
}],

"mark":{
   "type": "line",
   "color": {
                "expr": “colourPick”
        },
   "strokeWidth": 2
},
```

4 User input.

Picking a colour – another way



Exactly the same as above

This time using a text box

Try it out:

red, magenta, RGB(10, 30, 150)

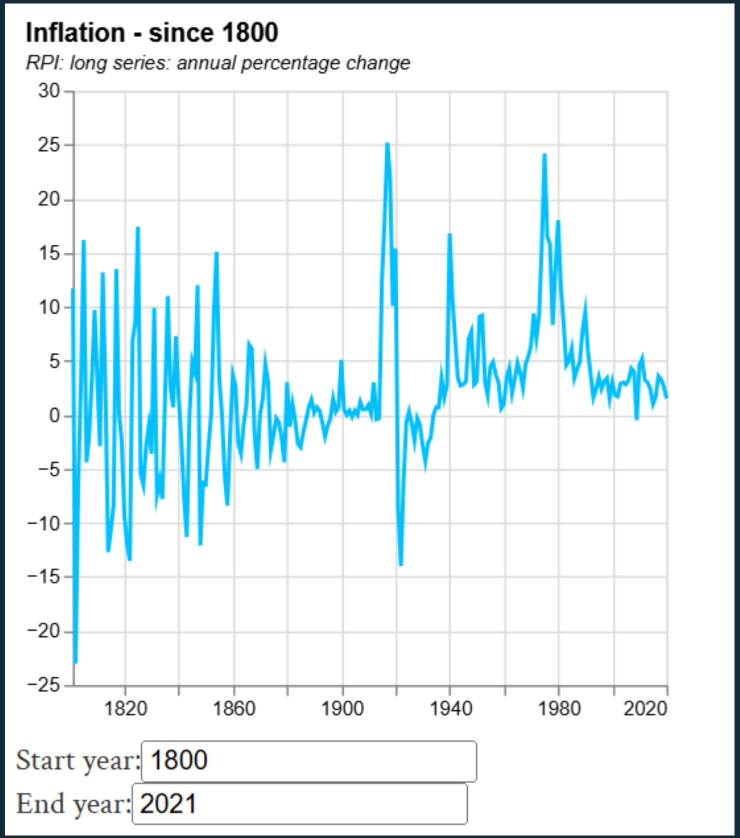
```
"params": [{
        "name": “myColour",
        "value": “red",
        "bind":{
                "input": “text",
                "name": "Choose colour:”
        }
}],

"mark":{
   "type": "line",
   "color":{
                "expr": “myColour”
        },
   "strokeWidth": 2
},
```

TIP: you can create more powerful search inputs using regular expressions (regex). Example [Vega-Lite] & [Altair]

5 User filtering.

Step by step

Step 1 – user gives some input

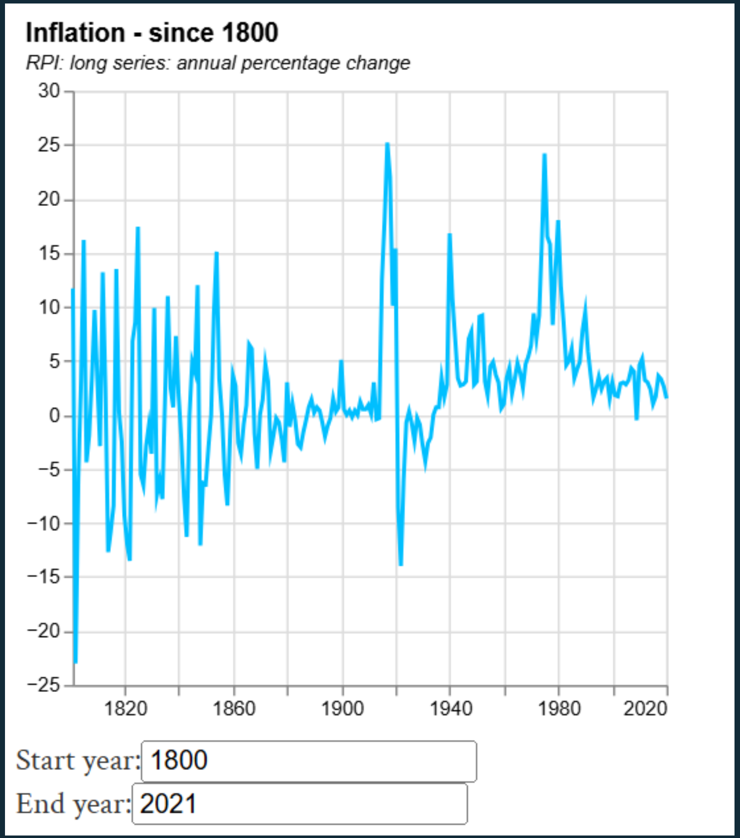



```
"params": [
    {
        "name": "minYear",
        "value": 1800,
        "bind": {
            "input": "text",
            "name": "Start year:"
        }
    },
    {
        "name": "maxYear",
        "value": 2021,
        "bind": {
            "input": "text",
            "name": "End year:"
        }
    }
],
```

5 User filtering.

Step by step

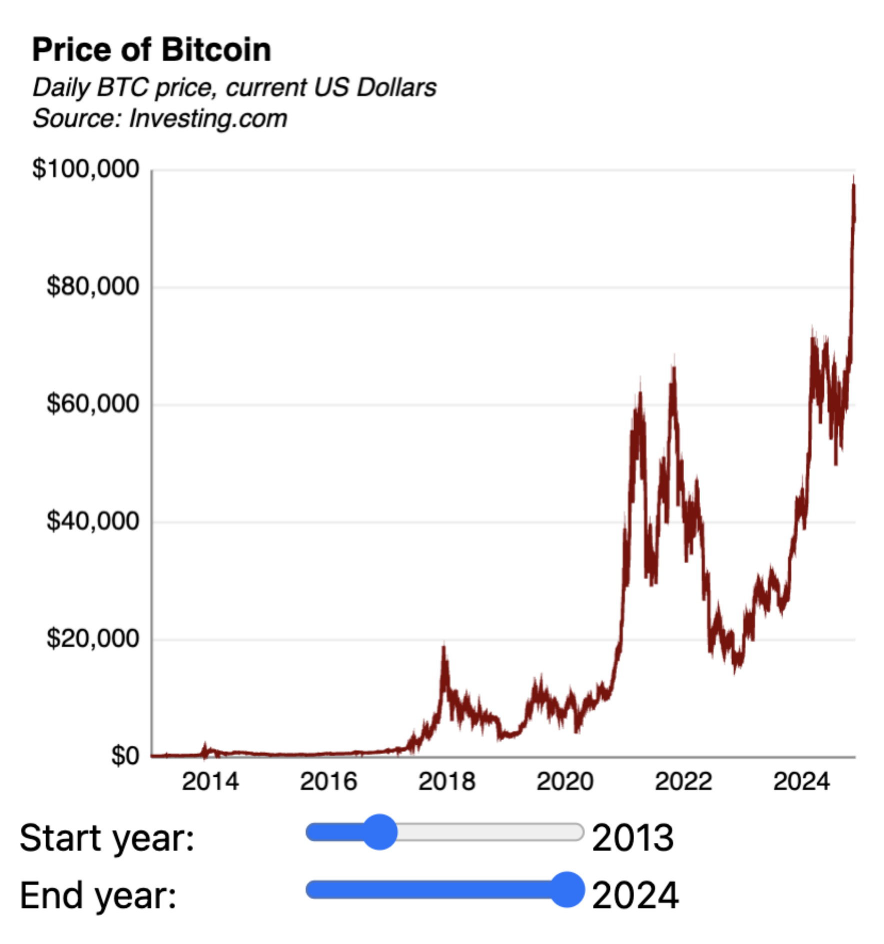
Step 2 – the user changes the chart




```
"transform": [
        {"filter": "datum.year > minYear"},
        {"filter": "datum.year < maxYear"}
],
```

6 User filtering.

Filtering time with range sliders



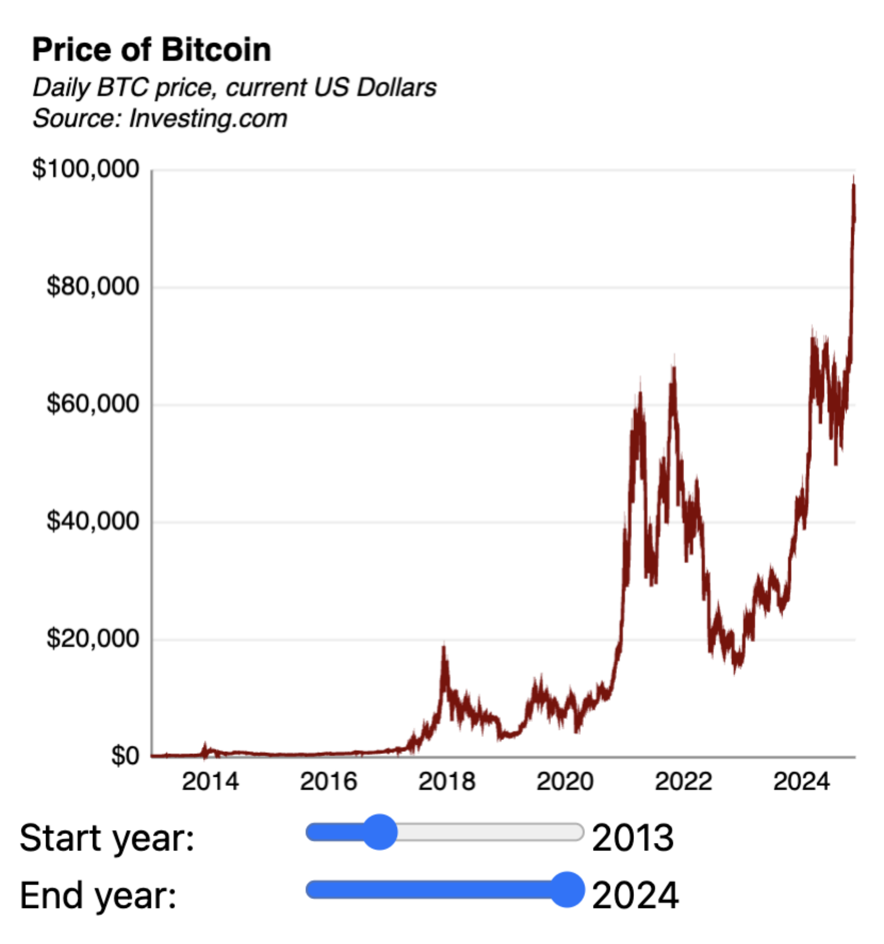
Using ‘range’ instead of ‘text’ input

```
"params": [
    {
        "name": ”minYear",
        "value": 2013,
        "bind": {
            "input": "range",
            "min": 2010,
            "max": 2024,
            "step": 1,
            "name": ”Start year:"
        }
    },
    /// + maxYear
],
```

Bind a range input to our parameter, specifying max/min values with a step.

6 User filtering.

Filtering time with range sliders



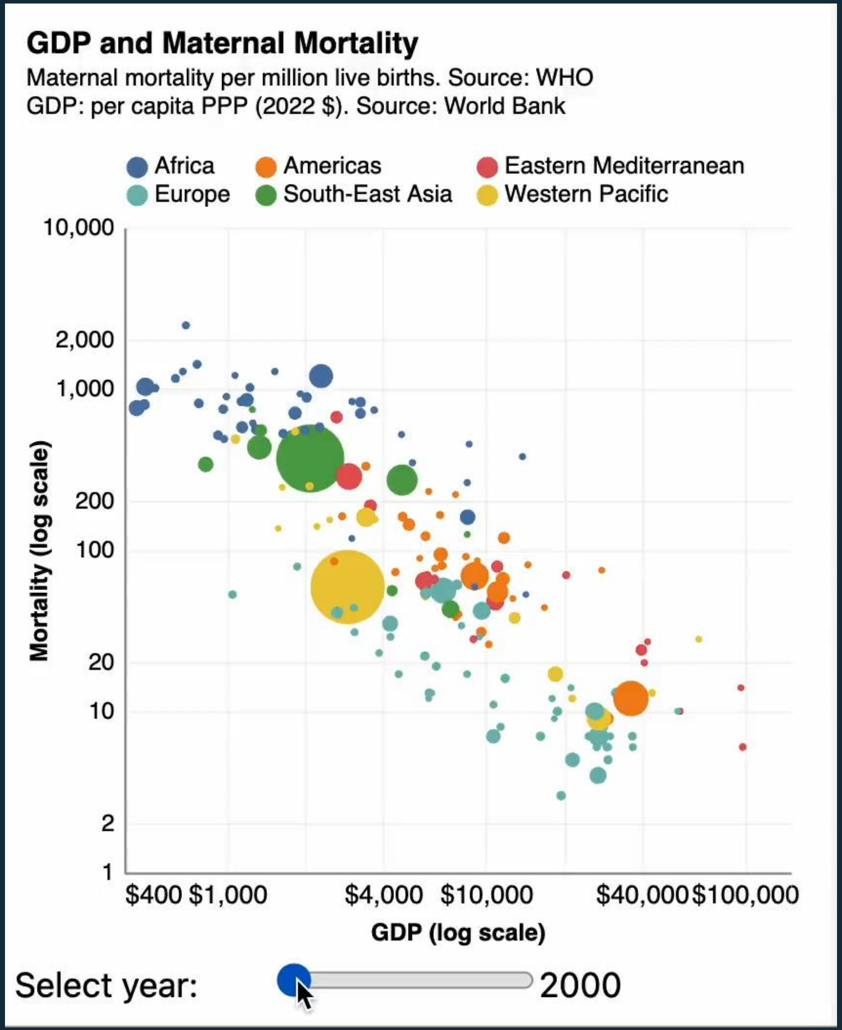
Filter based on user selection.

```
"transform": [
        {
                "calculate": "year(datum.Date)",
                "as": "year”
        },
        {"filter": "datum.year >= minYear"},
        {"filter": "datum.year <= maxYear"}
],
```

‘Date’ is in yyyy-mm-dd format, so to filter by year we must first calculate a new ‘year’ variable.

6 User filtering.

Filtering time with range sliders – selecting a year

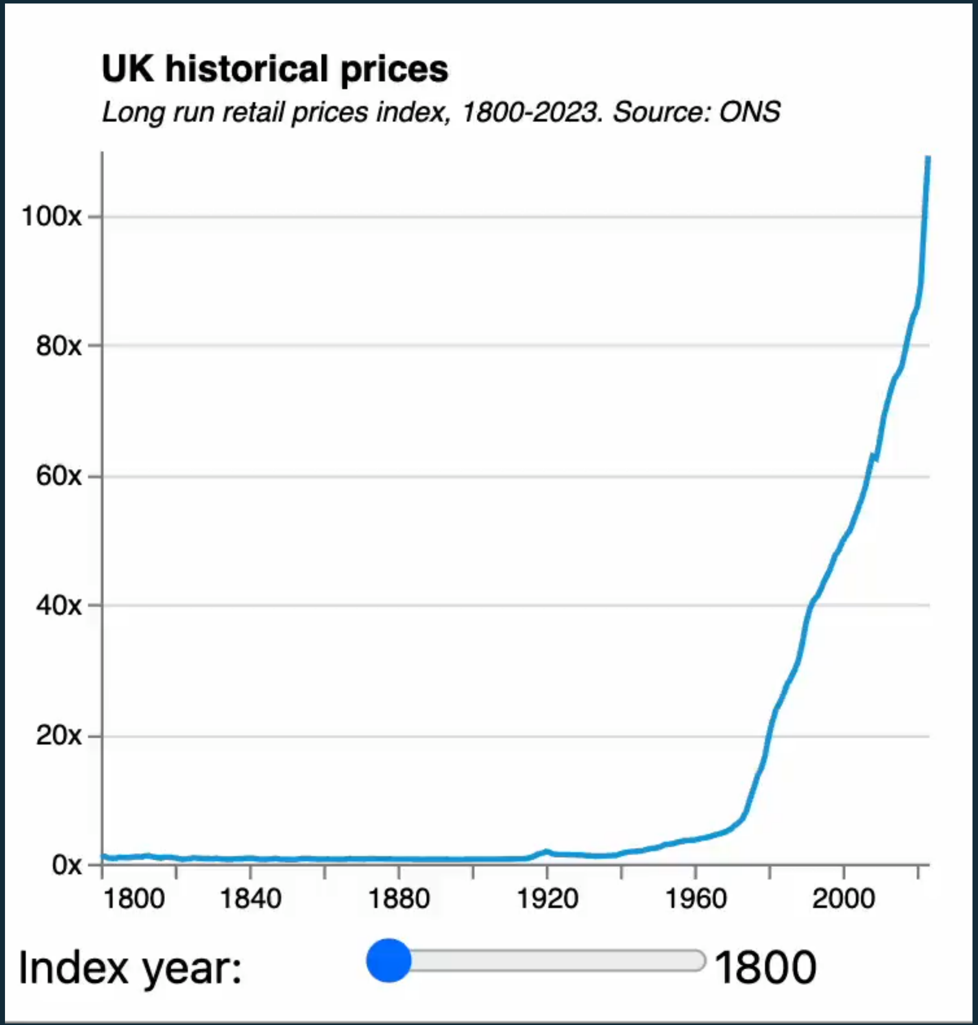



```
"params": [
    {
        "name": "yearSelector",
        "value": 2000,
        "bind": {
            "input": "range",
            "min": 2000,
            "max": 2017,
            "step": 1,
            "name": "Select year:"
        }
    }
],

"transform": [
    {"filter": "datum.Year == yearSelector"}
]
```

Filtering for a single year is just as easy.

7 User indexing.

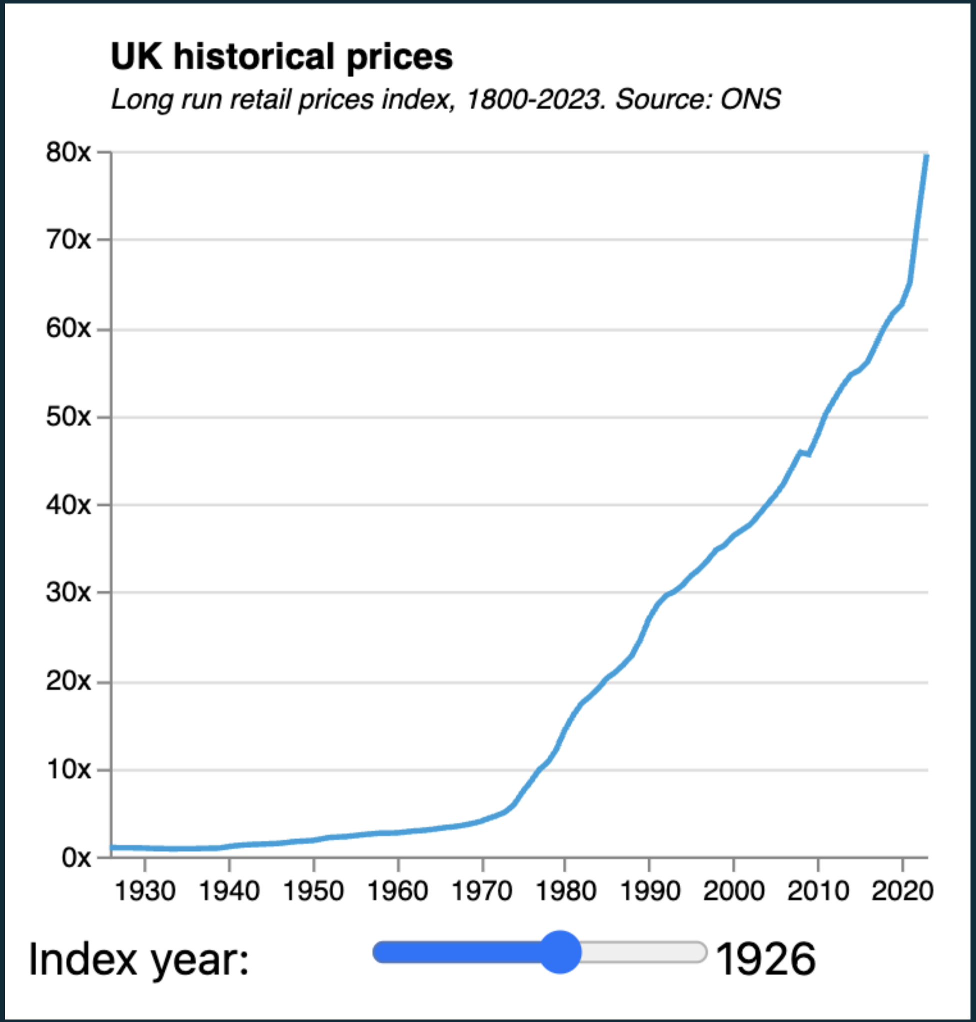
Interactive re-indexing from a range slider



With some extra transforms we can use a slider to recalculate a starting index from the selected year.

7 User indexing.

Interactive re-indexing from a range slider



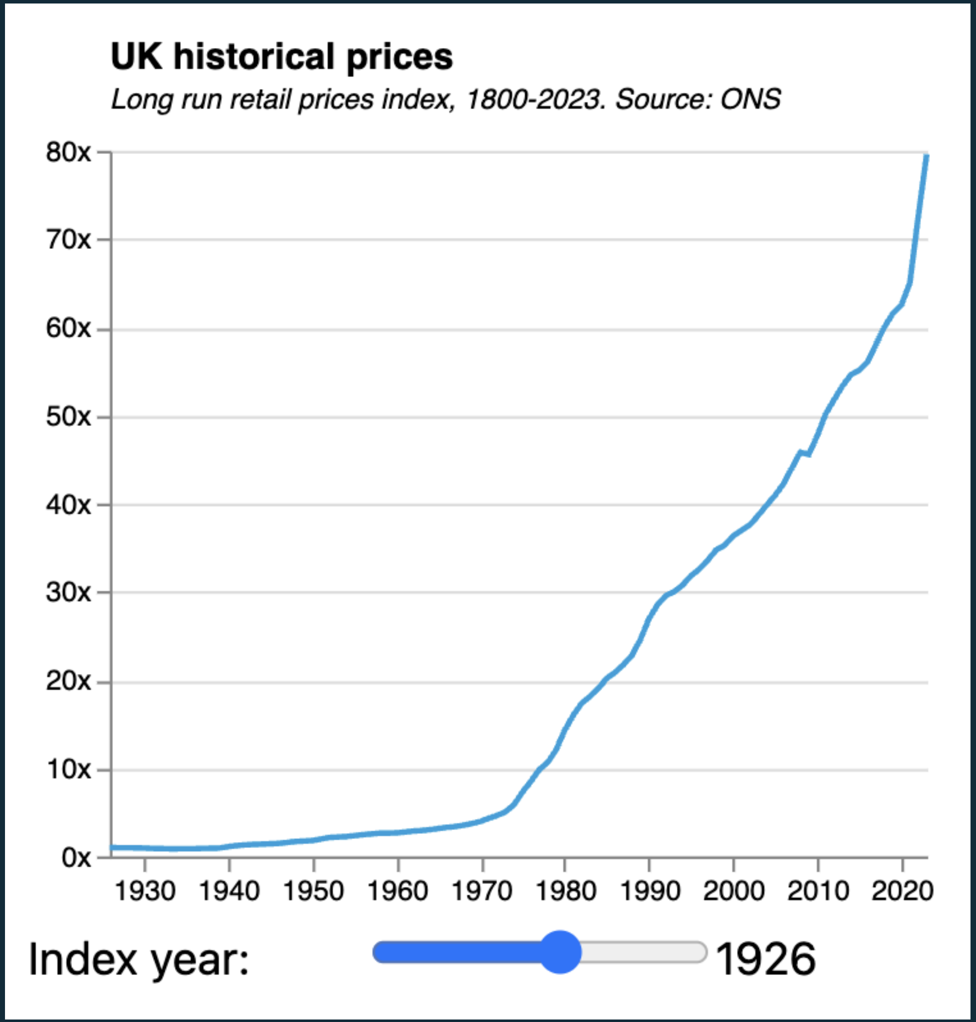
Start with the same parameter setup

```
"params": [
    {
        "name": "startYear",
        "value": 1974,
        "bind": {
            "input": "range",
            "min": 1800,
            "max": 2022,
            "step": 1,
            "name": "Index year:”
        }
    }
]
```

7 User indexing.

Interactive re-indexing from a range slider

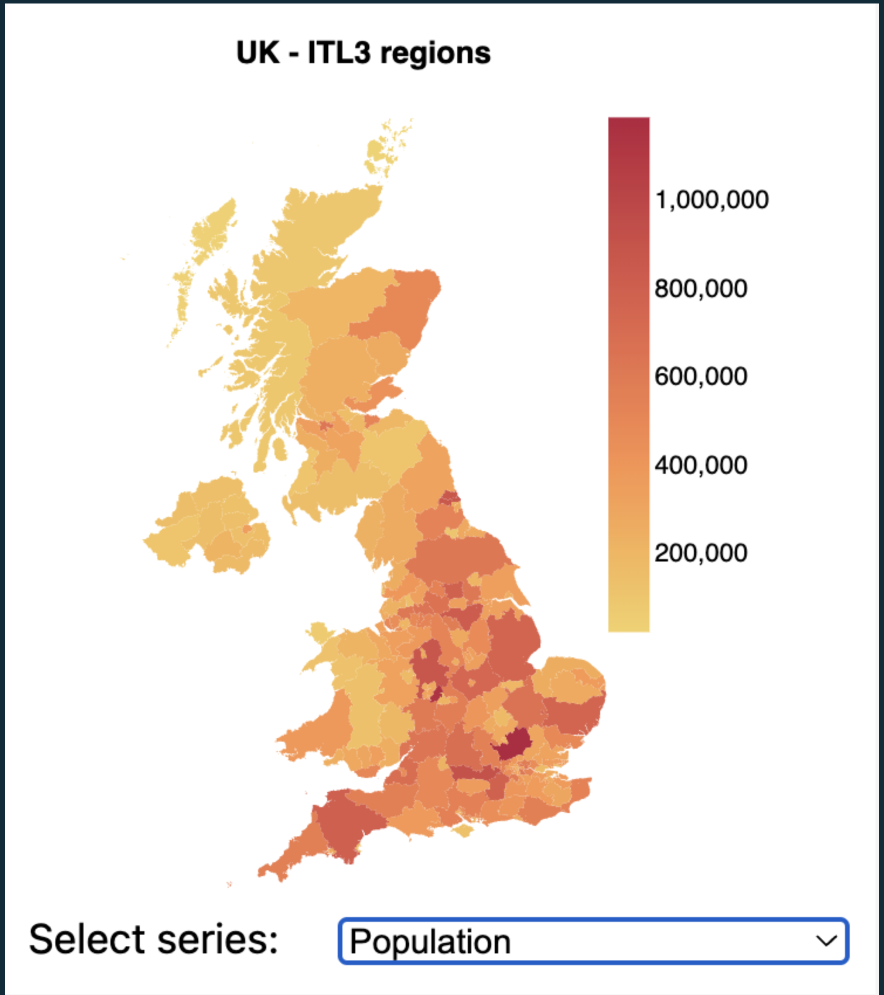
1. Filter for after the selected date
2. Use a 'window' transform to find the first value
3. Calculate new variable with change from each value

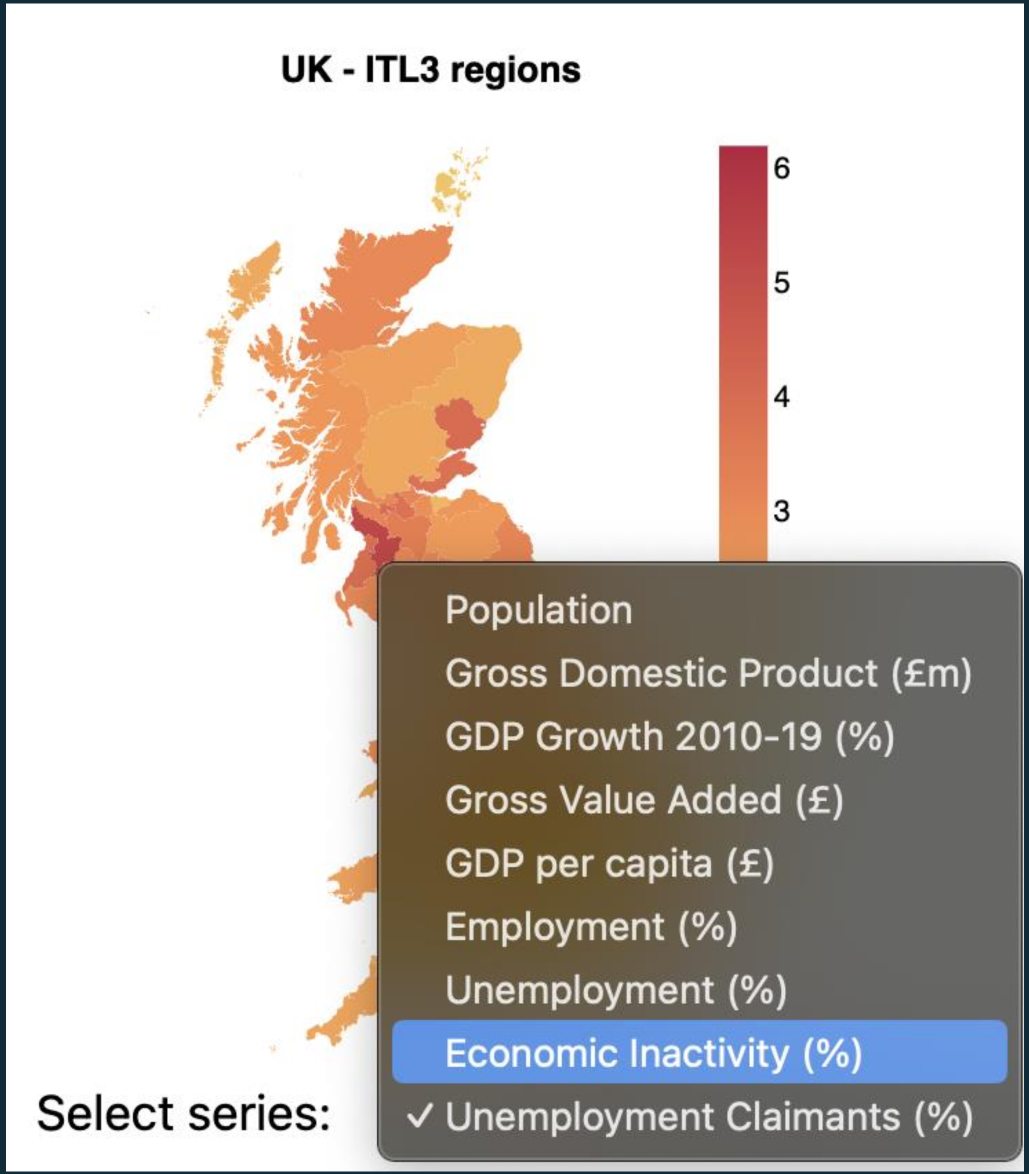



```
"transform": [
    {"filter": "datum.year >= startYear"},

    {"window": [
        {
            "op": "first_value",
            "field": "value",
            "as": "value_first”
        }
        ],
        "frame": [null, null],
        "sort": [{"field": "_index"}]
    },

    {"calculate": "datum.value /
        datum.value_first * 100",
        "as": "value_live”
    }
],
```

8 User selection.

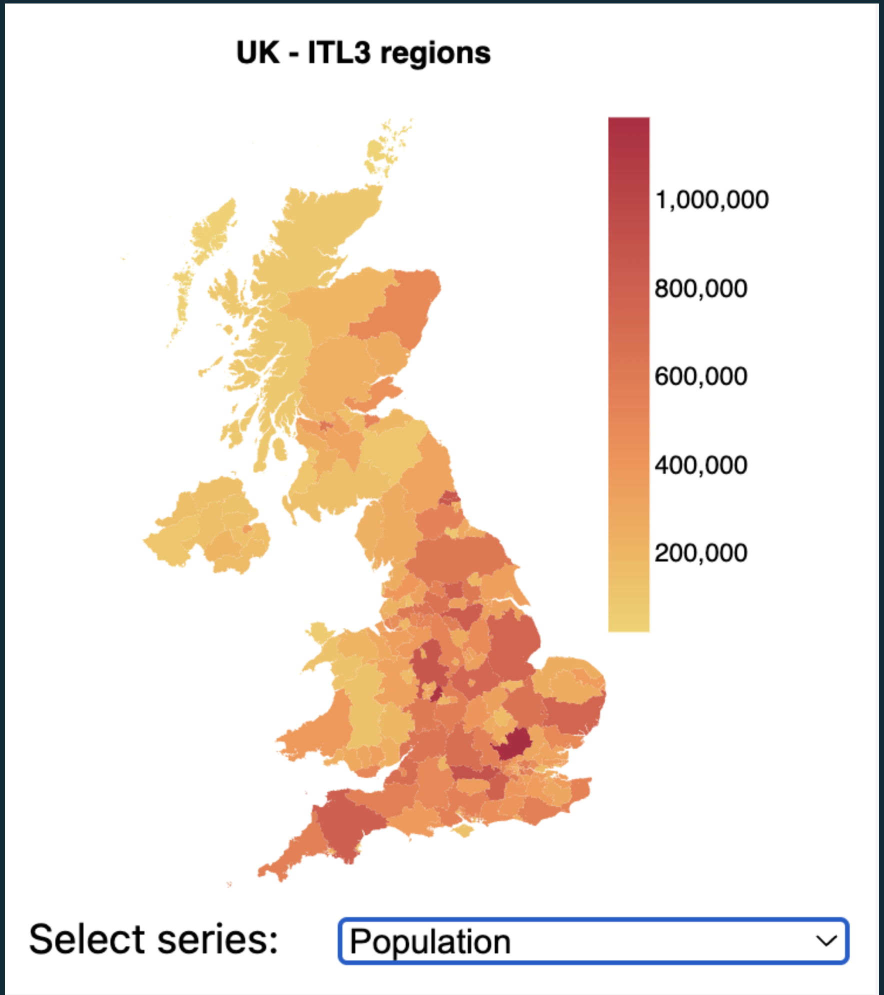
Selecting a series from dropdown





8 User selection.

Selecting a series from dropdown

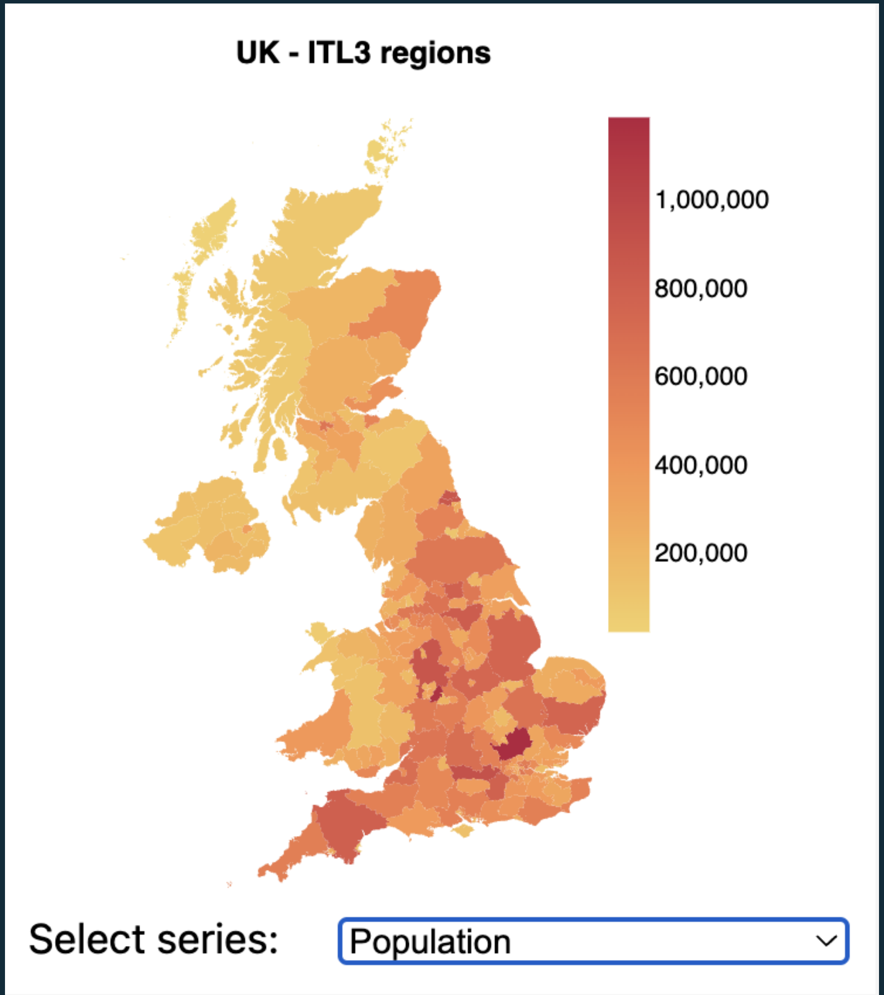


1. Add selection parameter, with options matching variable names

```
"params": [
    {
        "name": ”SelectParam",
        "value": "Population",
        "bind": {
            "input": "select",
            "options": [
                "Population",
                "Gross Domestic Product (£m)",
                "GDP Growth 2010-19 (%)",
                "Gross Value Added (£)",
                "GDP per capita (£)",
                "Employment (%)",
                "Unemployment (%)",
                "Economic Inactivity (%)",
                "Unemployment Claimants (%)”
            ],
            "name": "Select series:”
        }
    }
],
```

8 User selection.

Selecting a series from dropdown



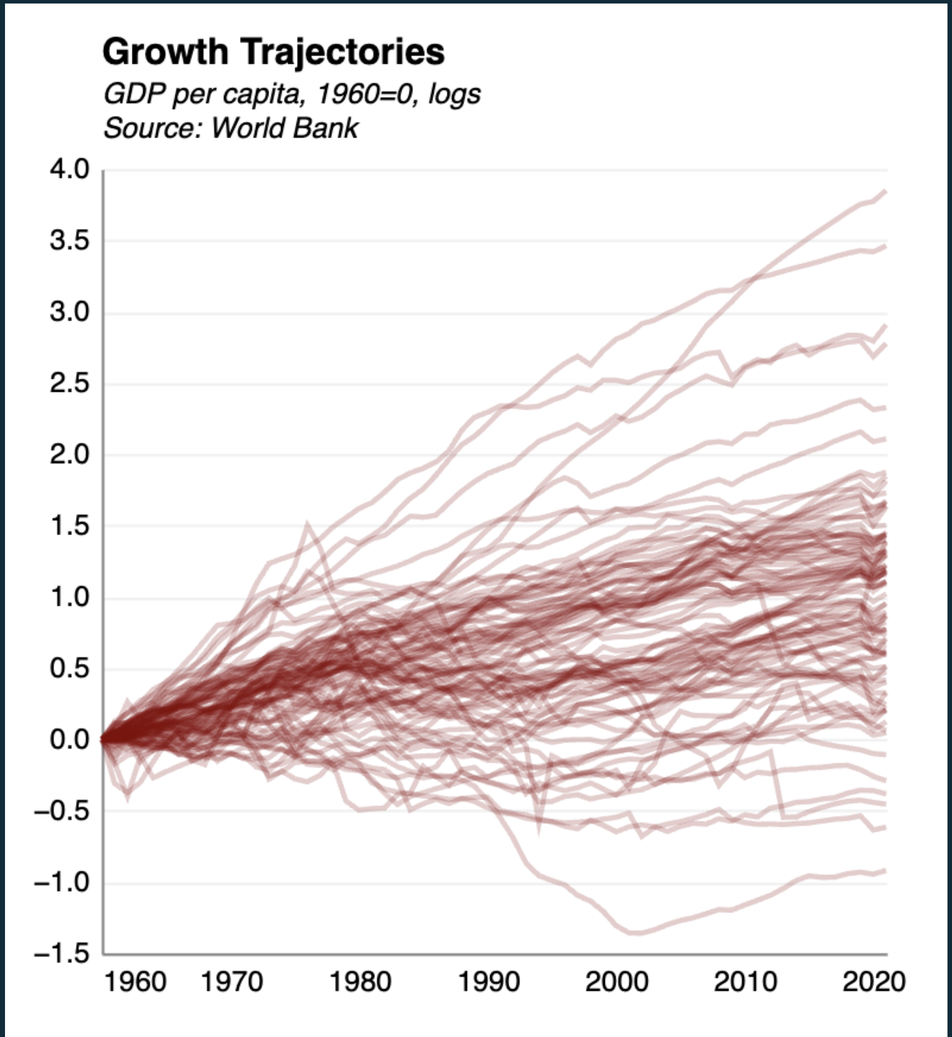
2. Calculate a new variable with values from the selected column.

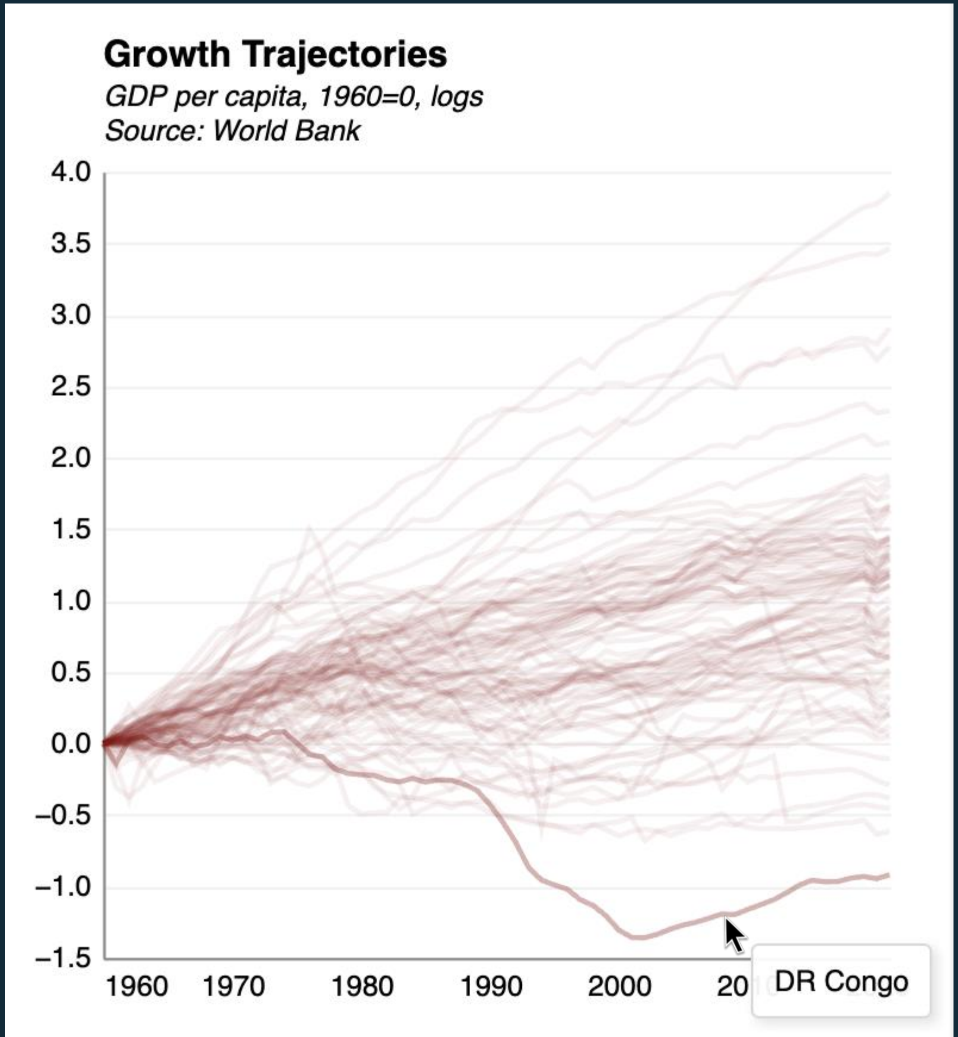
3. Assign new responsive variable to the colour encoding.

```
"transform": [
        {
                "calculate": "datum[SelectParam]",
                "as":"varSelected”
        }
],


"encoding":{
        "color":{
                "field":"varSelected",
                "type": "quantitative",
                "scale":{"scheme":"goldred”},
        }
},
```

9 Highlight.

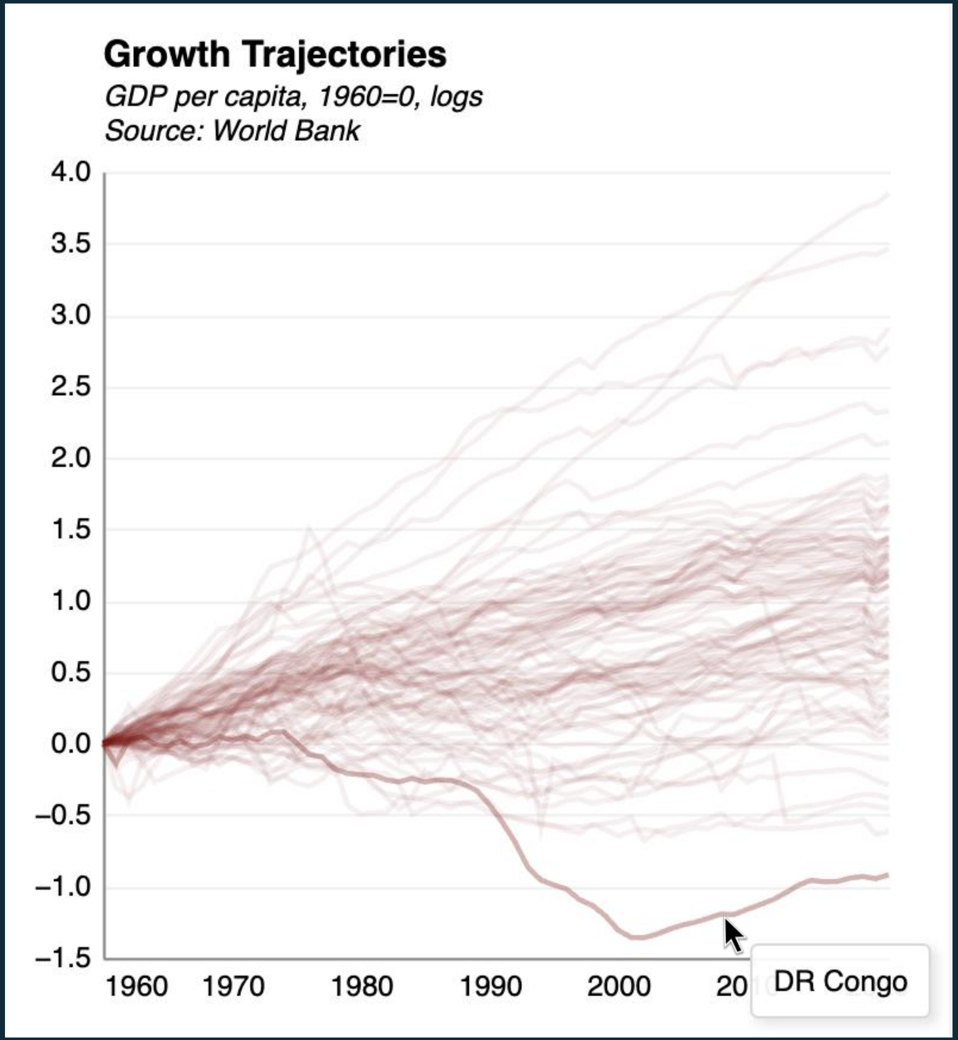
Highlighting on mouse hover





9 Highlight.

Highlighting on mouse hover



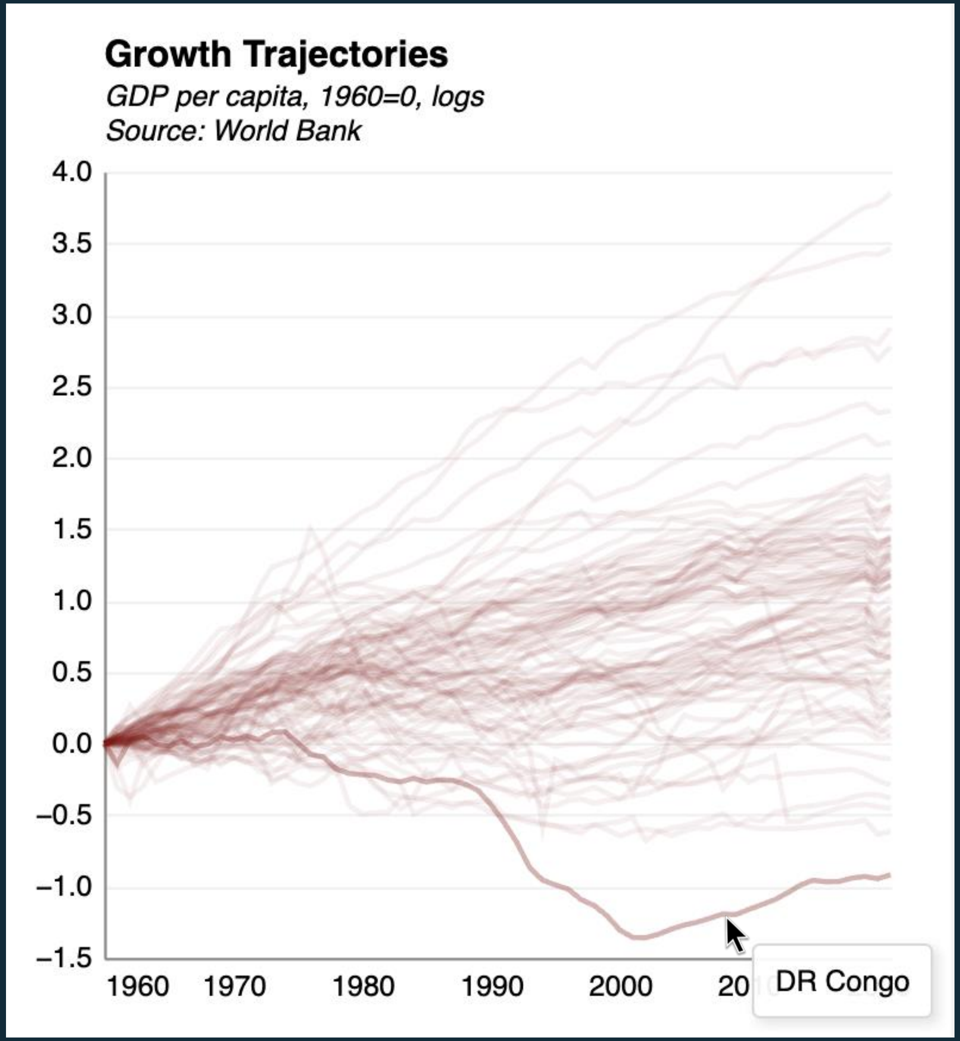
Layer with actual lines and hidden transparent lines (easier to trigger selection)

```
"layer": [
    {
        "params": [{
            "name": "hover",

            "select": {
                "type": "point",
                "fields": ["Country"],
                "on": "mouseover”
            }
        }],
        "mark": {
            "type": "line",
            "strokeWidth": 8,
            "stroke": "transparent"}
    },

    {"mark": "line"}
]
```

9 Highlight.

Highlighting on mouse hover

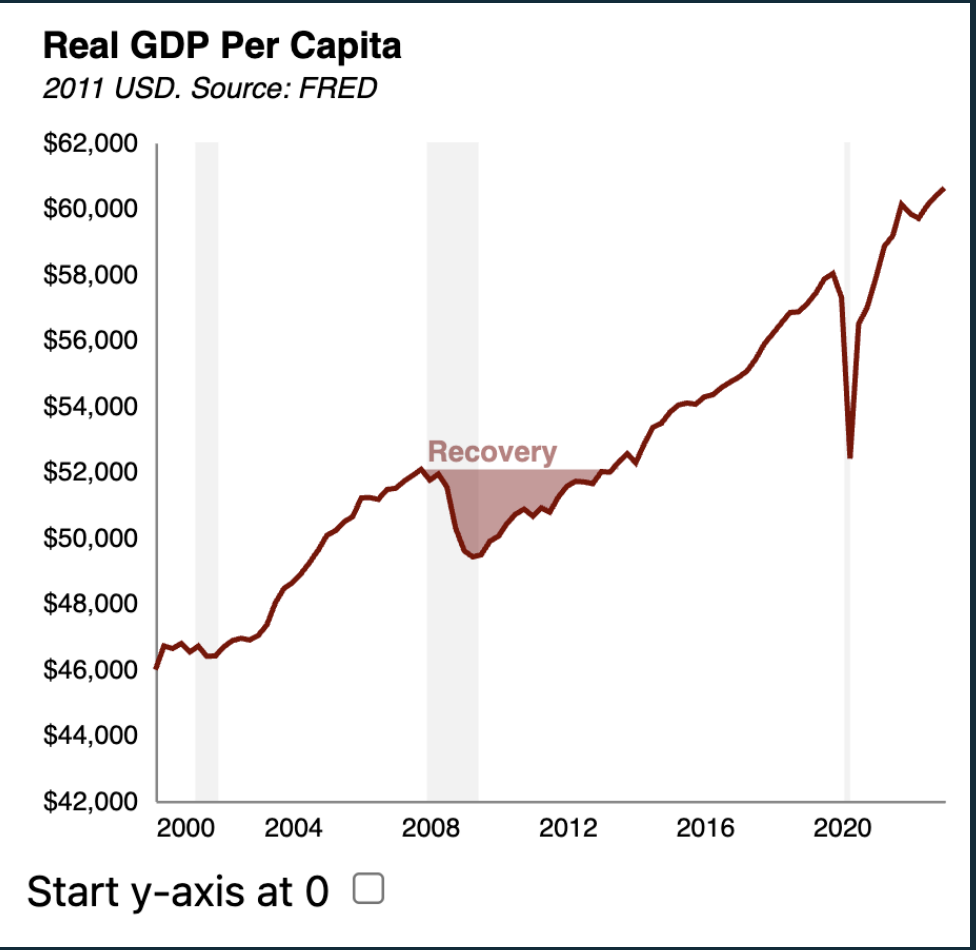
Conditional opacity based on our hover parameter

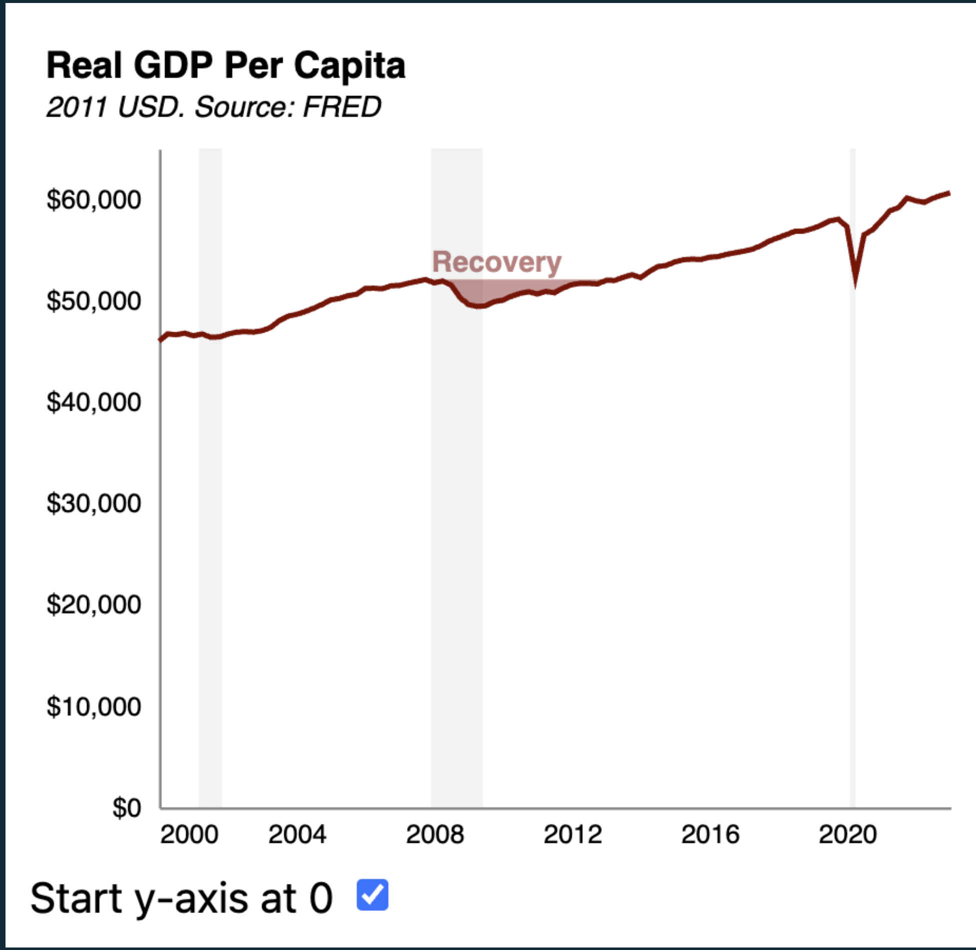



```
"encoding": {
…
      "opacity": {
            "condition":{
                  "param": "hover",
                  "value": 1
            },
            "value": 0.2
      }
}
```

10 User checkbox.

Checkbox options

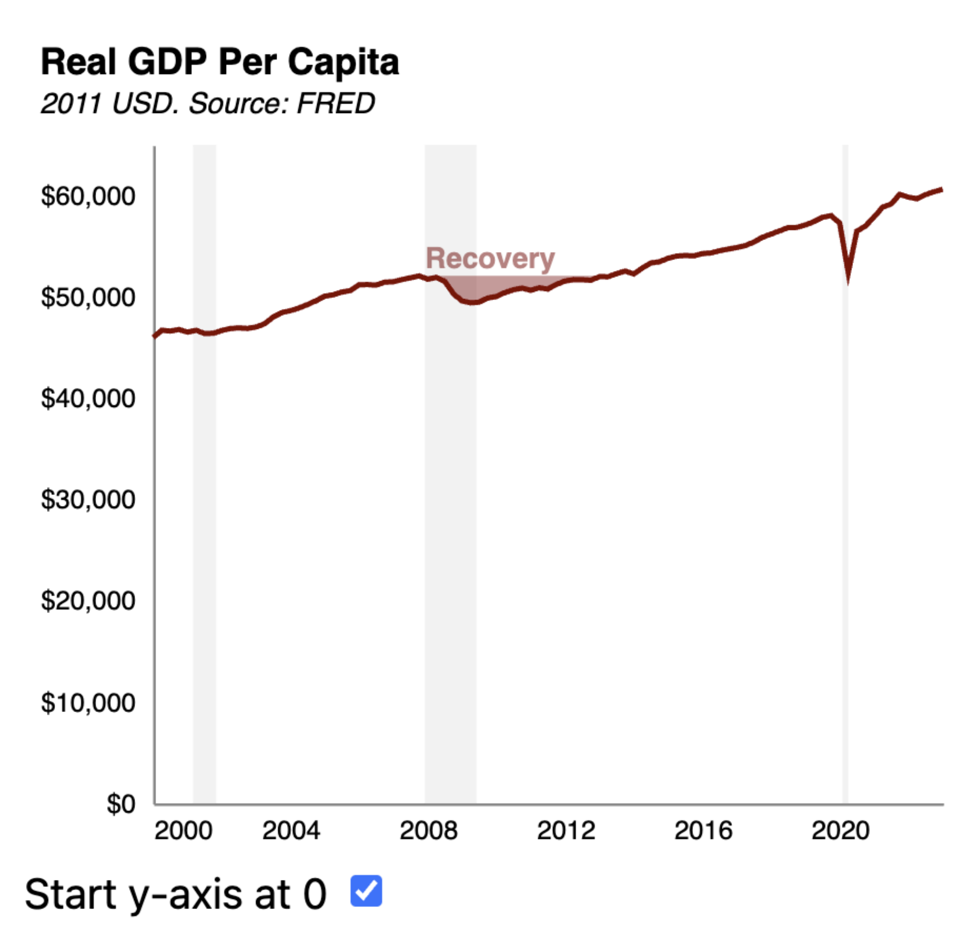




10 User checkbox.

Checkbox options

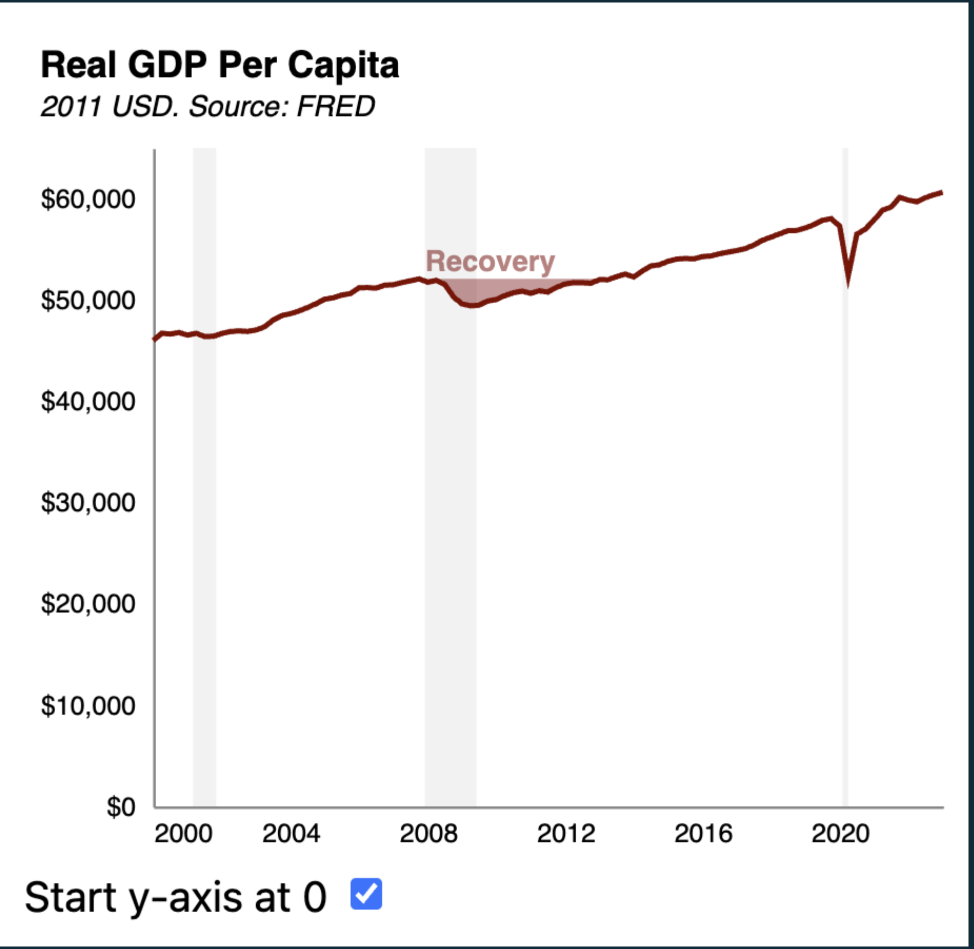
1. Add parameter with input set to checkbox




```
"params": [
    {
        "name": "toggleZero",
        "value": false,
        "bind":
            "input": "checkbox",
            "name": "Start y-axis at 0”
        }
    },
]
```

10 User checkbox.

Checkbox options



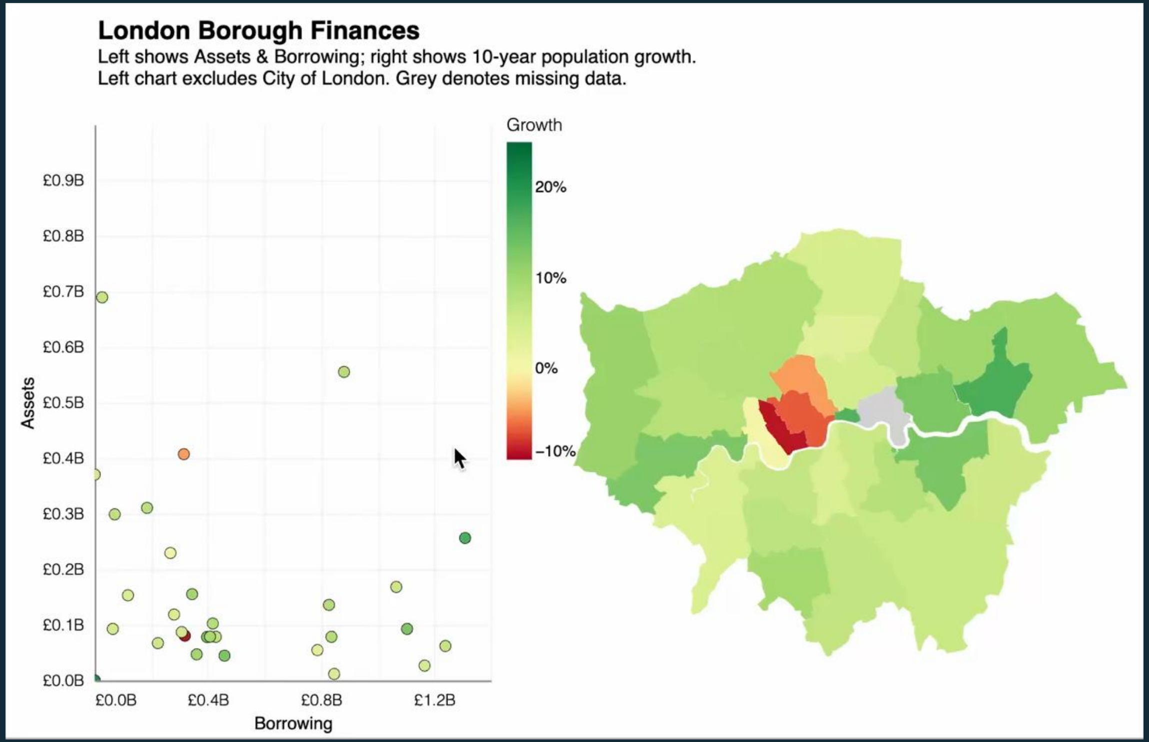

```
"encoding": {
…
        "y": {
                "field": "Value",
                "type": "quantitative",
                "scale": {
                        "domainMin": {
                                "expr": "toggleZero ? 0 : 42000"
                        }
        }
}
```

2. On the y-axis scale, add a conditional domain minimum.

- If toggle is true (checked), then domain starts from 0
- If toggle is false (unchecked), then domain starts from 42000.

11 Selection.

Selection across charts




```
"params": [{
        "name": ”point_hover",
        "select": {
                "type": "point",
                "fields": [”Name"],
                "on": "mouseover”}
}],
… /// Add conditional encodings with param
```

Connect concatenated charts using interactivity.

1. Add a point selection parameter set on mouseover.

Left point chart:

- Conditional opacity encoding on points
- Text labels layer with transform filter to include only selected area

Right map:

- Conditional opacity and strokeWidth (border)

[Code]()

Break.

Then: portfolios, projects, practical.

9.5 Practical.

Interactive visualisations