Fluid: Explorable, Transparent Data Visualisation

Roly Perera

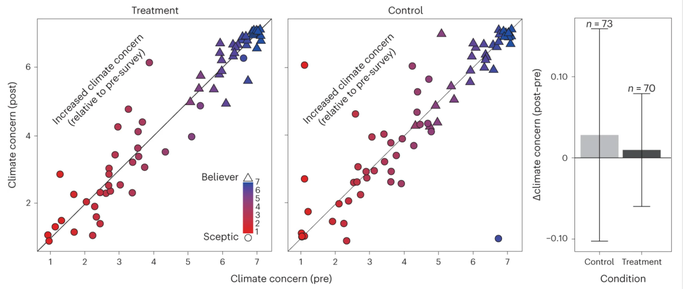
Institute of Computing for Climate Science, University of Cambridge
School of Computer Science, University of Bristol

From opaque snapshots to “open outputs”

research papers

Fig. 2: Distributions of climate beliefs before and after participating in the climate market.

From: Participating in a climate prediction market increases concern about global warming



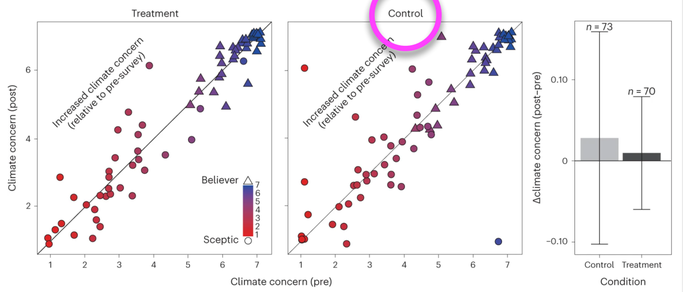
Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

From opaque snapshots to “open outputs”

research papers

Fig. 2: Distributions of climate beliefs before and after participating in the climate market.

From: Participating in a climate prediction market increases concern about global warming



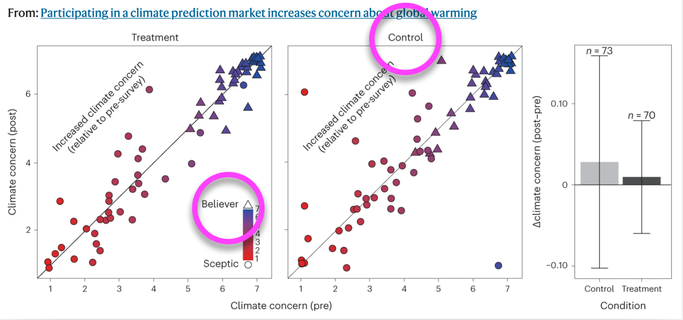
Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

From opaque snapshots to “open outputs”

research papers

Fig. 2: Distributions of climate beliefs before and after participating in the climate market.

From: Participating in a climate prediction market increases concern about global warming



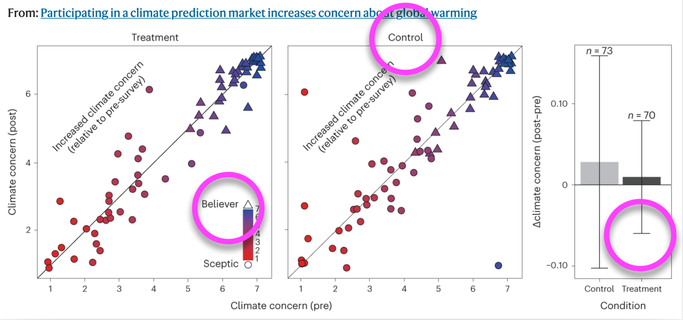
Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

From opaque snapshots to “open outputs”

research papers

Fig. 2: Distributions of climate beliefs before and after participating in the climate market.

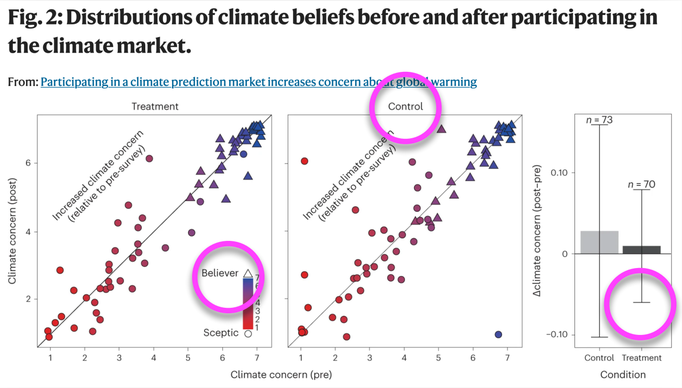
From: Participating in a climate prediction market increases concern about global warming

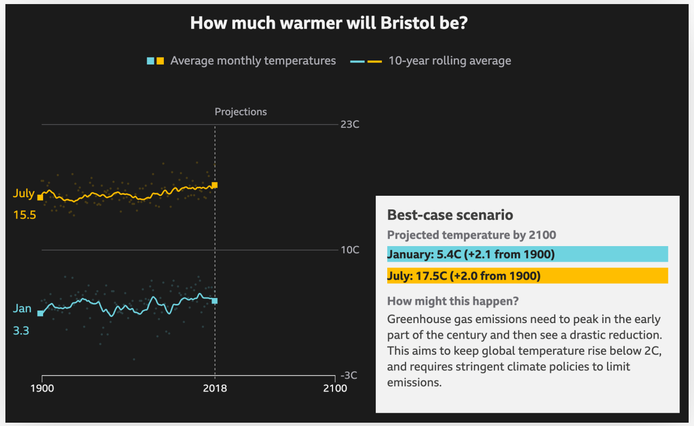


Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

From opaque snapshots to “open outputs”

research papers



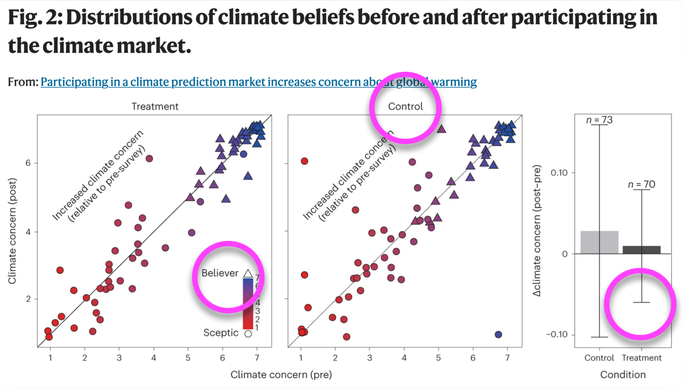


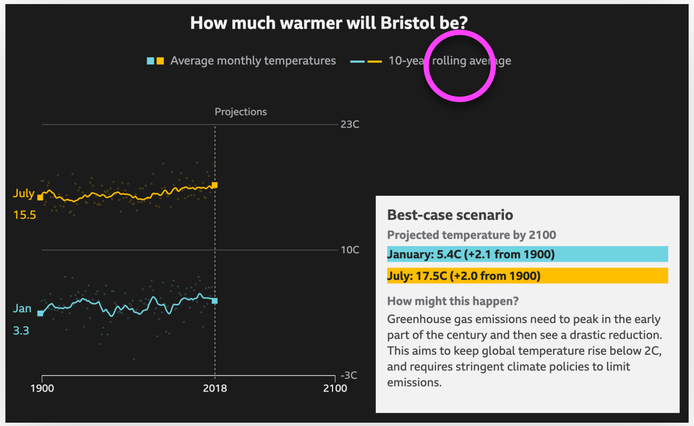
visual journalism

Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

From opaque snapshots to “open outputs”

research papers



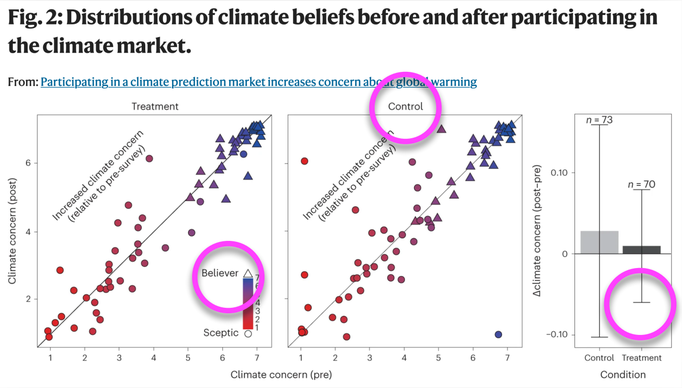


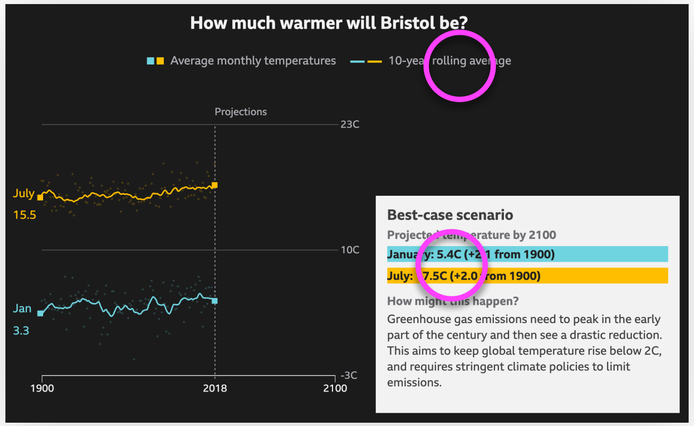
visual journalism

Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

From opaque snapshots to “open outputs”

research papers



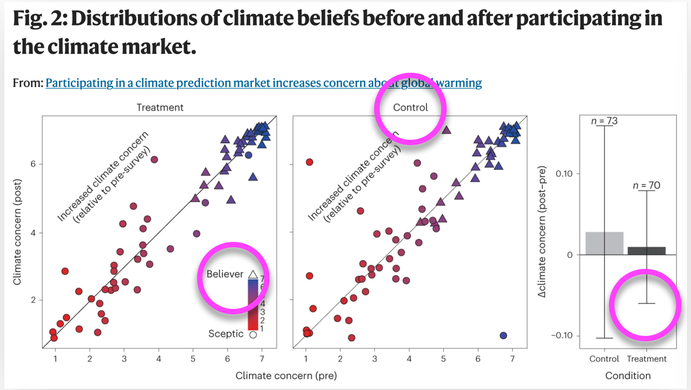


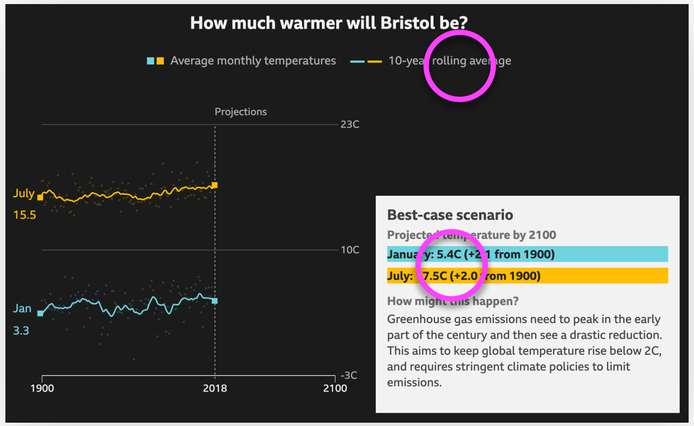
visual journalism

Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

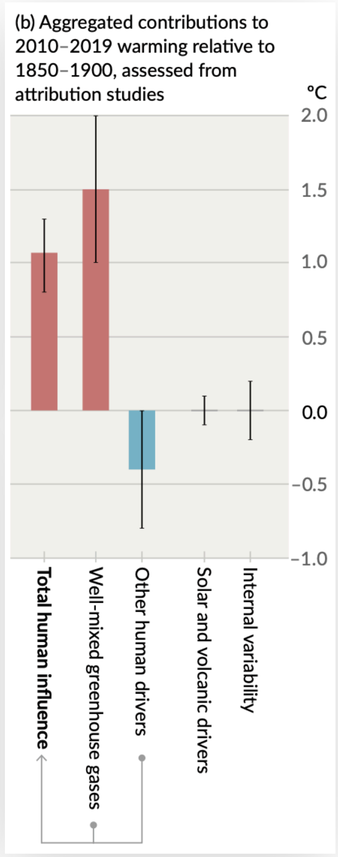
From opaque snapshots to “open outputs”

research papers





visual journalism

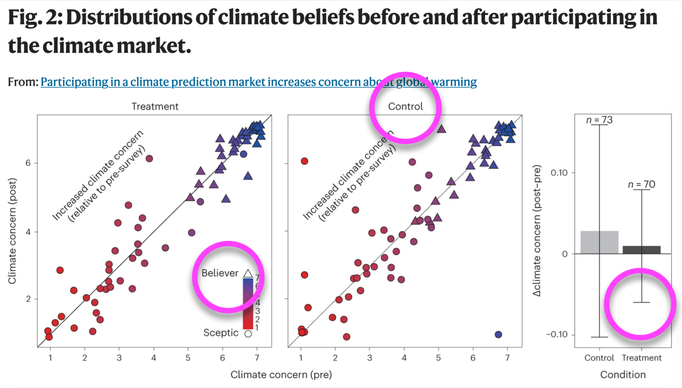


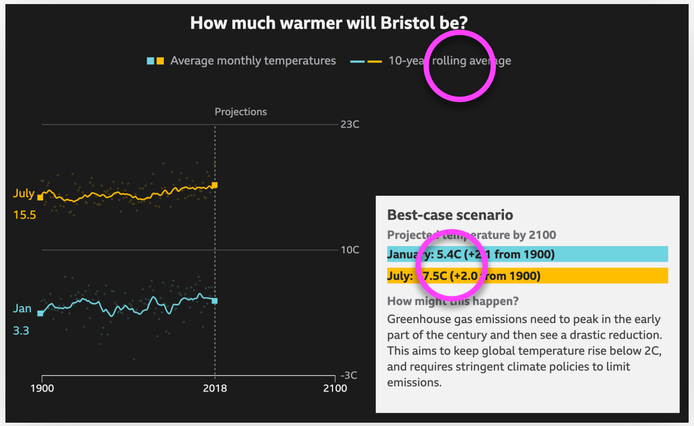
IPCC reports

Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

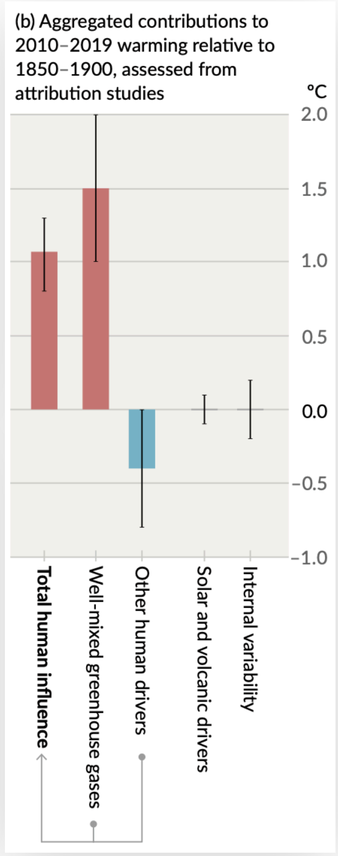
From opaque snapshots to “open outputs”

research papers

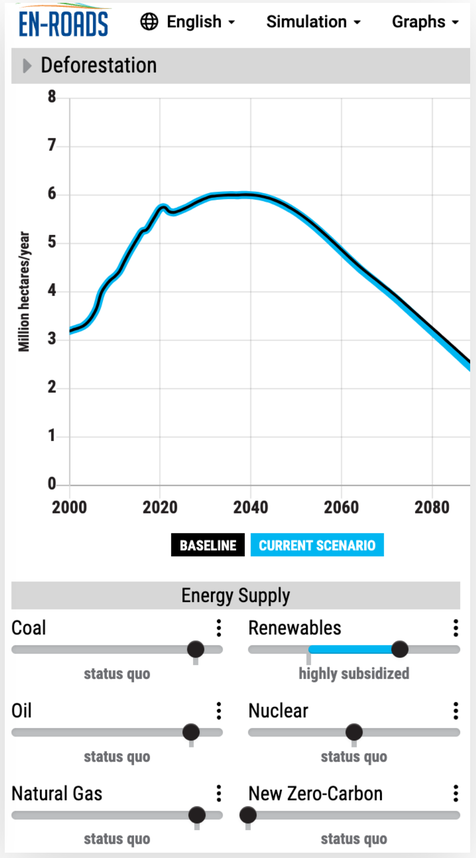




visual journalism



IPCC reports

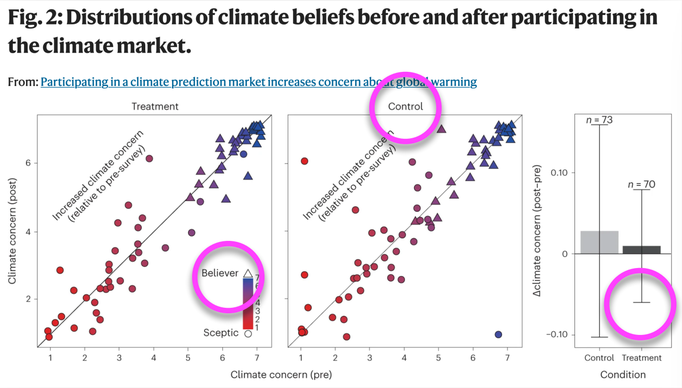


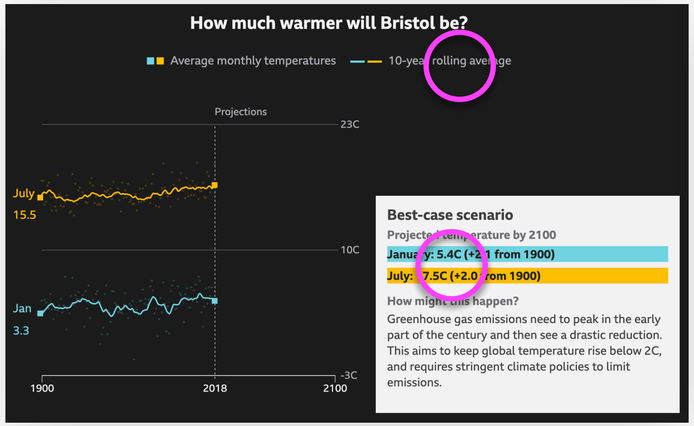
interactive simulation

Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

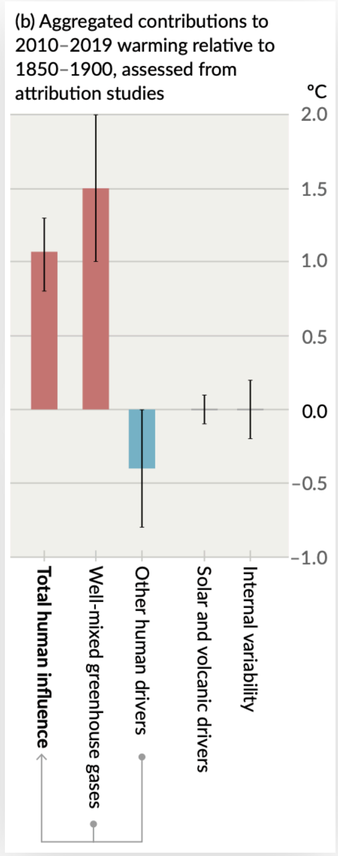
From opaque snapshots to “open outputs”

research papers

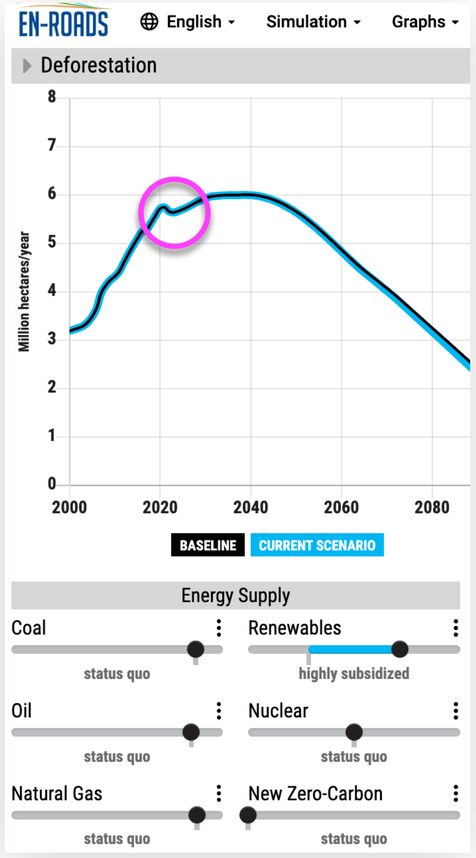




visual journalism



IPCC reports

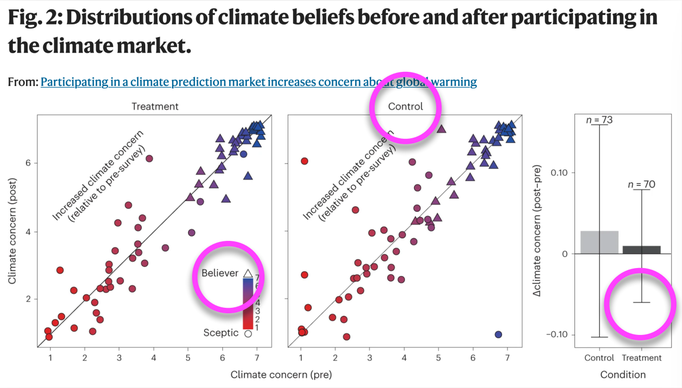


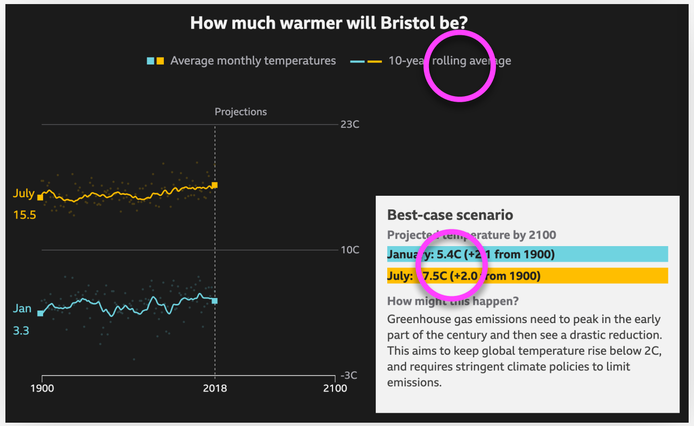
interactive simulation

Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

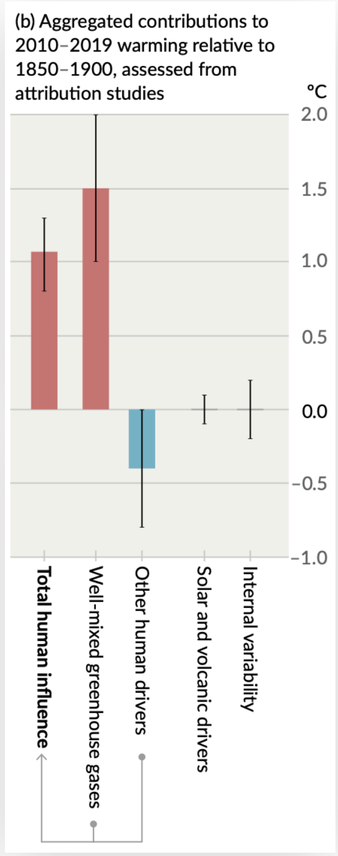
From opaque snapshots to “open outputs”

research papers

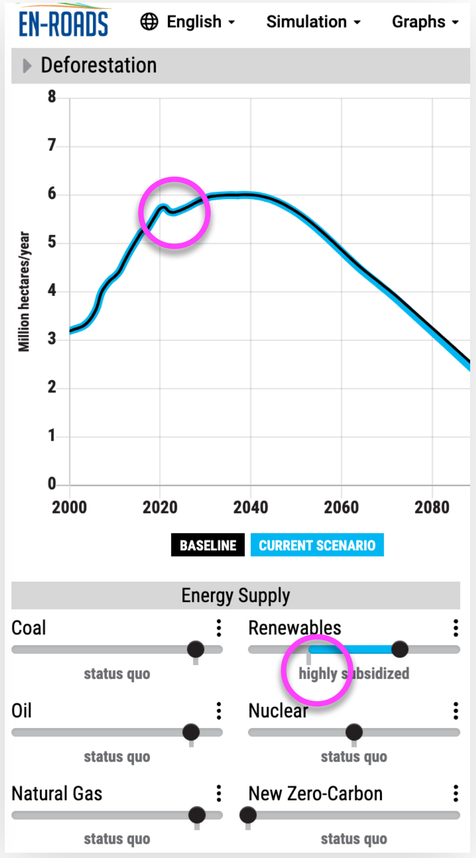




visual journalism



IPCC reports

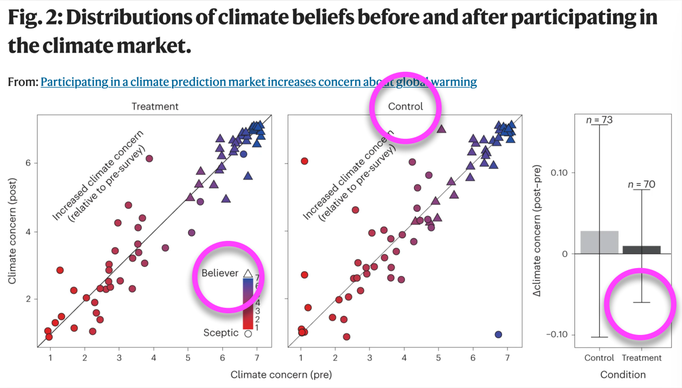


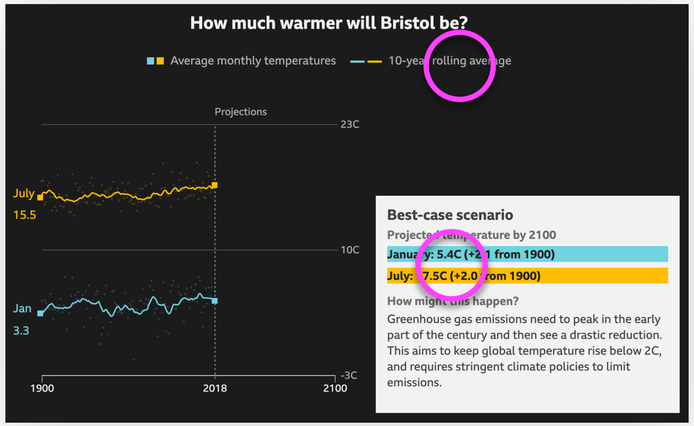
interactive simulation

Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

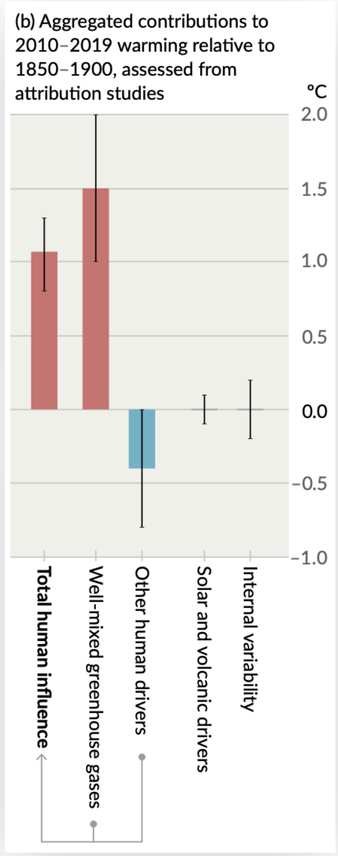
From opaque snapshots to “open outputs”

research papers

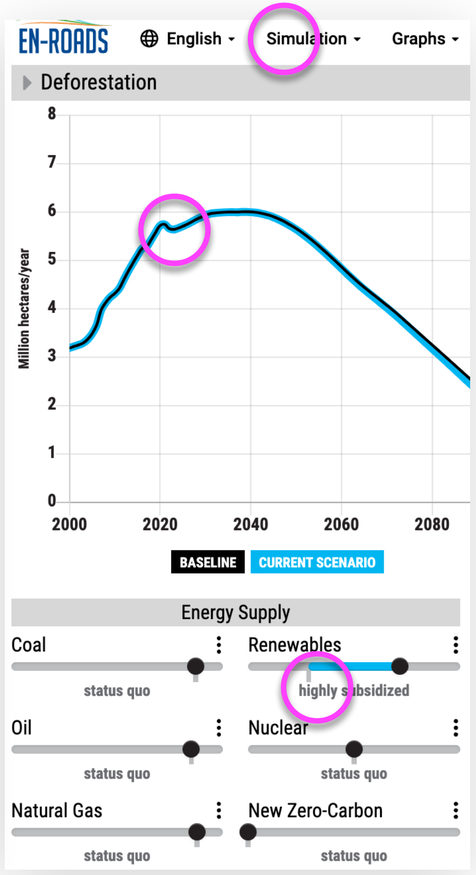




visual journalism



IPCC reports

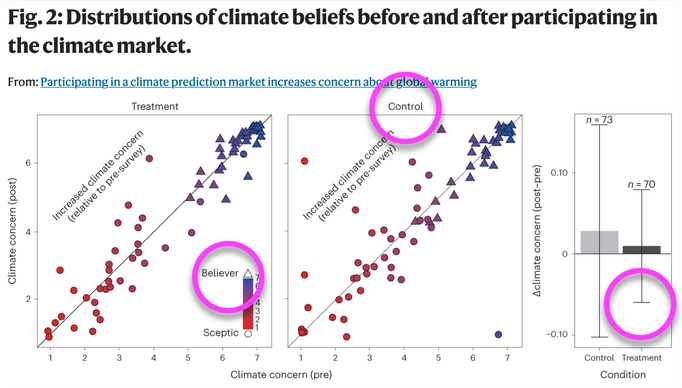


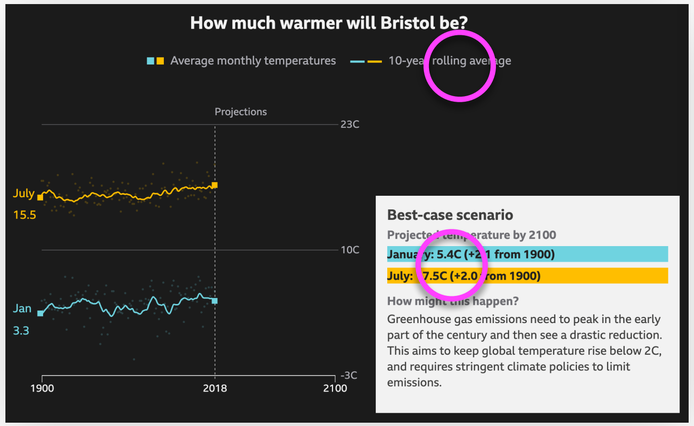
interactive simulation

Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

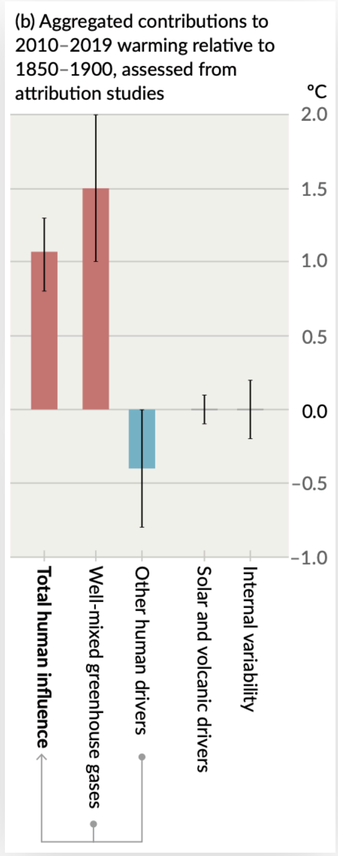
From opaque snapshots to “open outputs”

research papers

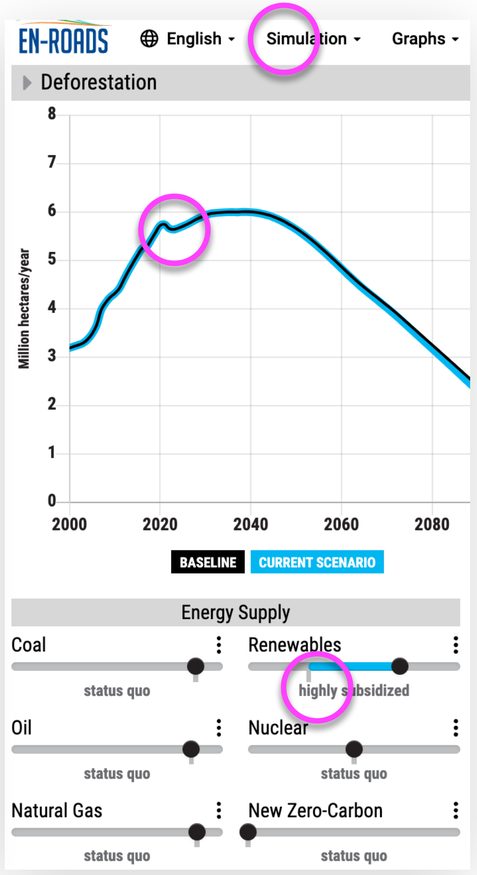




visual journalism



IPCC reports



interactive simulation

Today’s research outputs are **opaque**: disconnected from the data and computations used to produce them

What can we do to make these artifacts more **transparent and self-explanatory**?

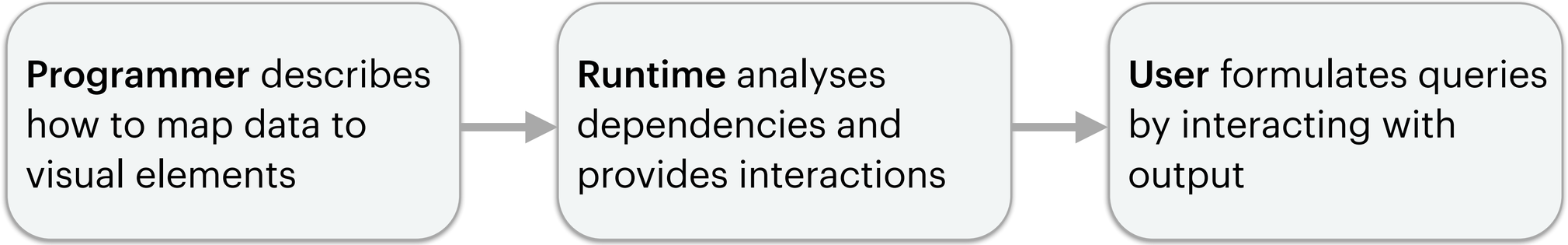
Demo: non-renewable energy charts

Demo: non-renewable energy charts

```
let totalFor year country rows =
   let [ row ] = [ row | row ← rows, row.year == year, row.country == country ]
   in row.nuclearOut + row.gasOut + row.coalOut + row.petrolOut;
let stack year = [ { y: country, z: totalFor year country nonRenewables }
                 | country ← ["BRA", "EGY", "IND", "JPN"] ]
in BarChart {
   caption: "Non-renewable output",
   size: { width: 275, height: 185 },
   stackedBars: [ { x: numToStr year, bars: stack year }
                | year ← [2014..2018] ]
}
```

Demo: non-renewable energy charts

```
let totalFor year country rows =
   let [ row ] = [ row | row ← rows, row.year == year, row.country == country ]
   in row.nuclearOut + row.gasOut + row.coalOut + row.petrolOut;
let stack year = [ { y: country, z: totalFor year country nonRenewables }
                 | country ← ["BRA", "EGY", "IND", "JPN"] ]
in BarChart {
   caption: "Non-renewable output",
   size: { width: 275, height: 185 },
   stackedBars: [ { x: numToStr year, bars: stack year }
                | year ← [2014..2018] ]
}
```



Demo: convolution

Demo: convolution

```
let zero n = const n;
    wrap n n_max = ((n - 1) `mod` n_max) + 1;
    extend n = min (max n 1);

let convolve image kernel method =
    let ((m, n), (i, j)) = (dims image, dims kernel);
        (half_i, half_j) = (i `quot` 2, j `quot` 2);
        area = i * j
    in  [| let weightedSum = sum [
              image!(x, y) * kernel!(i' + 1, j' + 1)
              | (i', j') ← range (0, 0) (i - 1, j - 1),
                let x = method (m' + i' - half_i) m,
                let y = method (n' + j' - half_j) n,
                x ⩾ 1, x ⩽ m, y ⩾ 1, y ⩽ n
           ] in weightedSum `quot` area
           | (m', n') in (m, n) |];
```

Demo: convolution

```
let zero n = const n;
    wrap n n_max = ((n - 1) `mod` n_max) + 1;
    extend n = min (max n 1);

let convolve image kernel method =
    let ((m, n), (i, j)) = (dims image, dims kernel);
        (half_i, half_j) = (i `quot` 2, j `quot` 2);
        area = i * j
    in  [| let weightedSum = sum [
             image!(x, y) * kernel!(i' + 1, j' + 1)
           | (i', j') ← range (0, 0) (i - 1, j - 1),
             let x = method (m' + i' - half_i) m,
             let y = method (n' + j' - half_j) n,
             x ⩾ 1, x ⩽ m, y ⩾ 1, y ⩽ n
         ] in weightedSum `quot` area
         | (m', n') in (m, n) |];
```

Programmer implements convolution in a conventional way

Runtime provides interactions that reveal behaviour of convolution

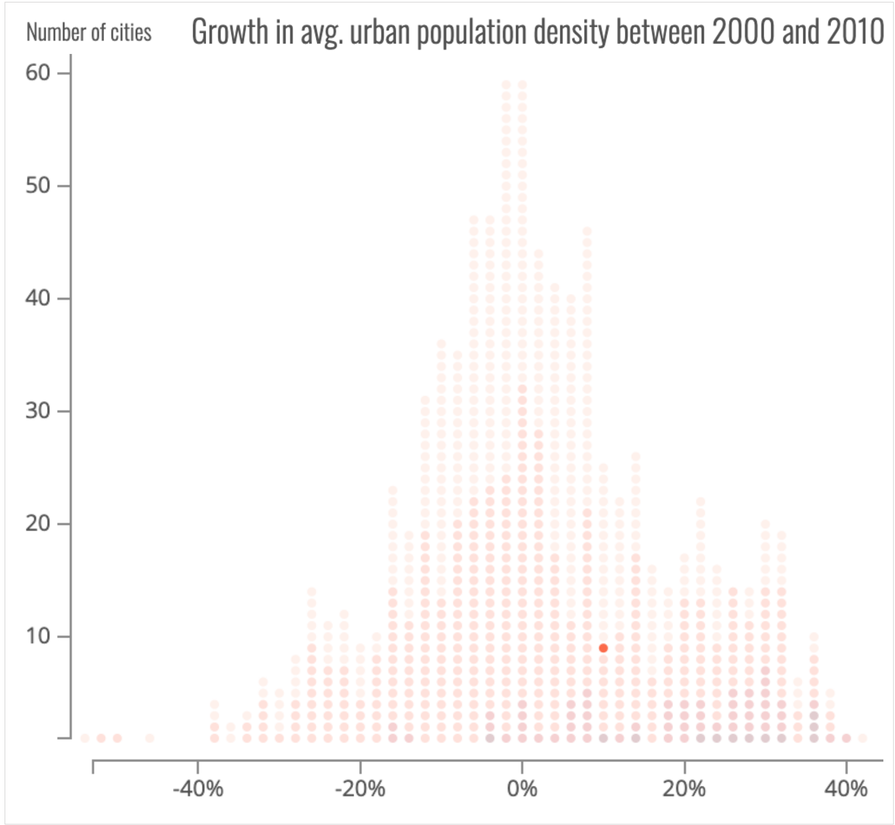
User formulates hypotheses and tests them through various interactions

Demo: moving average

```
let nthPad n xs =
      nth (min (max n 0) (length xs - 1)) xs;
    movingAvg ys window =
      [ sum [ nthPad n ys | n ← [ i - window .. i + window ] ] / (1 + 2 * window)
      | i ← [ 0 .. length ys - 1 ] ];
    movingAvg' rs window =
      zipWith
         (fun x y → {x: x, y: y})
         (map (fun r → r.x) rs)
         (movingAvg (map (fun r → r.y) rs) window);
let points =
      [ { x: r.year, y: r.emissions } | r ← methane, r.type == "Agriculture" ]
in LineChart {
   tickLabels: { x: Rotated, y: Default },
   size: { width: 330, height: 285 },
   caption: "SSP5-8.5 projected methane emissions (Agriculture)",
   plots: [ LinePlot { name: "Moving average", points: movingAvg' points 1 },
            LinePlot { name: "Original curve", points: points } ]
}
```

Next steps

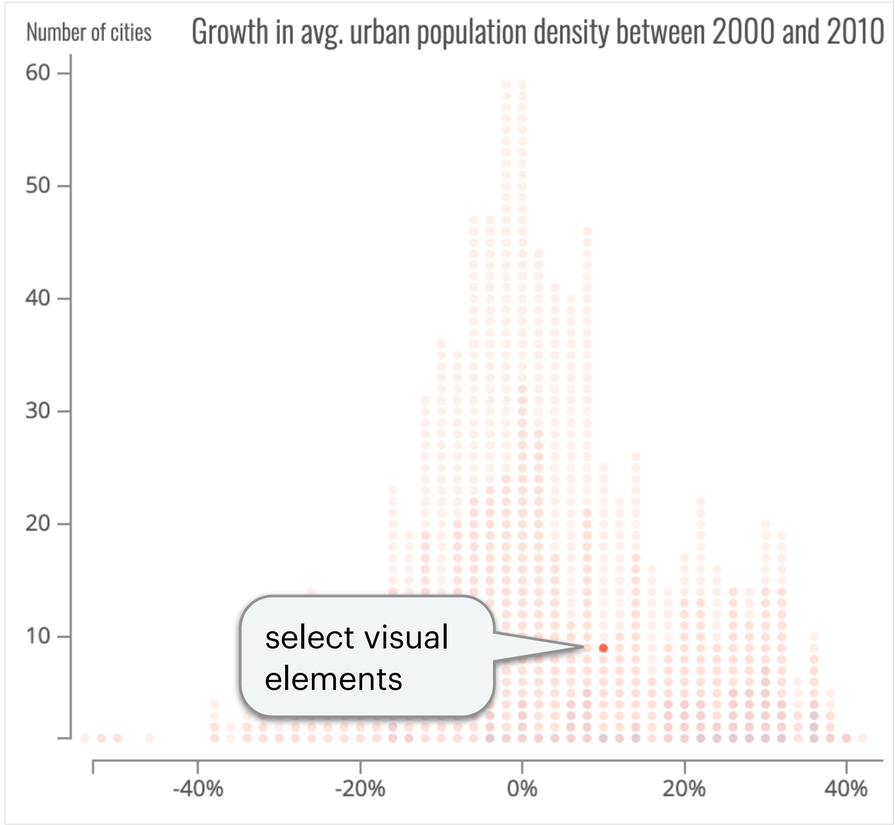
Enrich outputs with
computational explanations
(how, not just what)



Urbanization in East Asia between 2000 and 2010
Nadieh Bremer, Marlieke Ranzijn (http://nbremer.github.io/urbanization, 2015)

Next steps

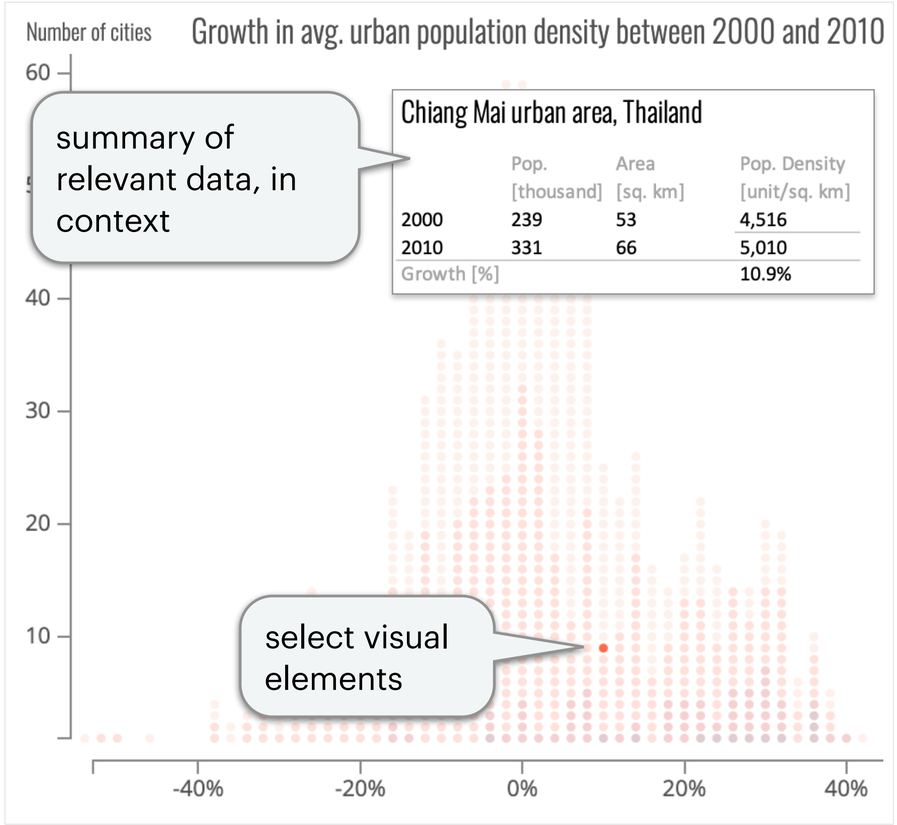
Enrich outputs with
computational explanations
(how, not just what)



Urbanization in East Asia between 2000 and 2010
Nadieh Bremer, Marlieke Ranzijn (http://nbremer.github.io/urbanization, 2015)

Next steps

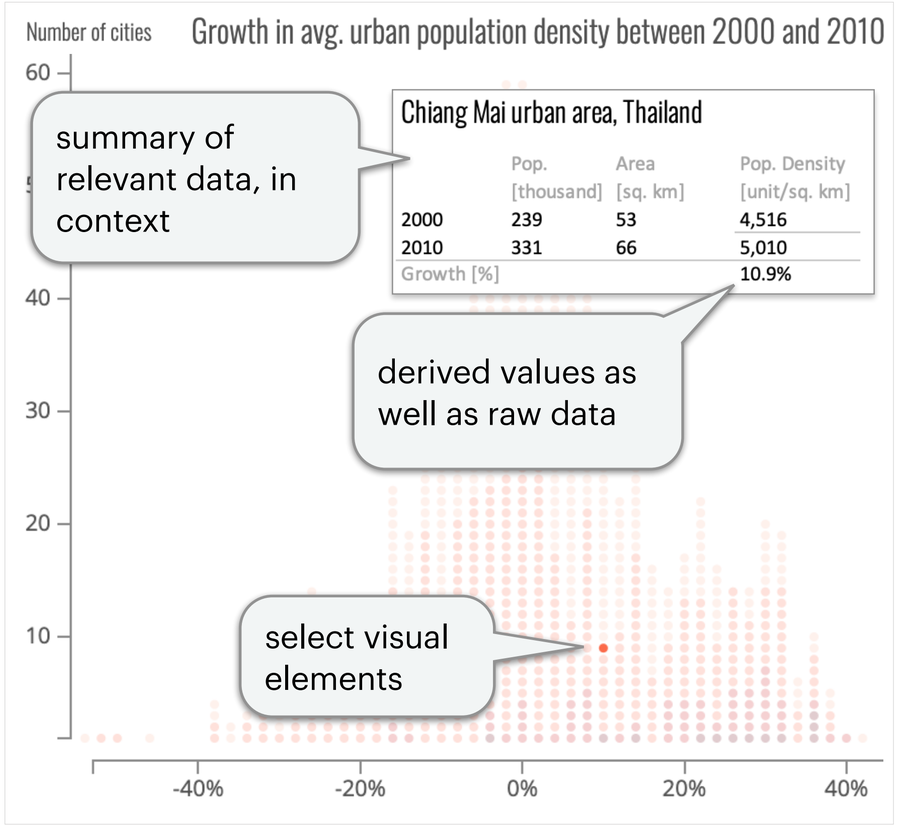
Enrich outputs with
computational explanations
(how, not just what)



Urbanization in East Asia between 2000 and 2010
Nadieh Bremer, Marlieke Ranzijn (http://nbremer.github.io/urbanization, 2015)

Next steps

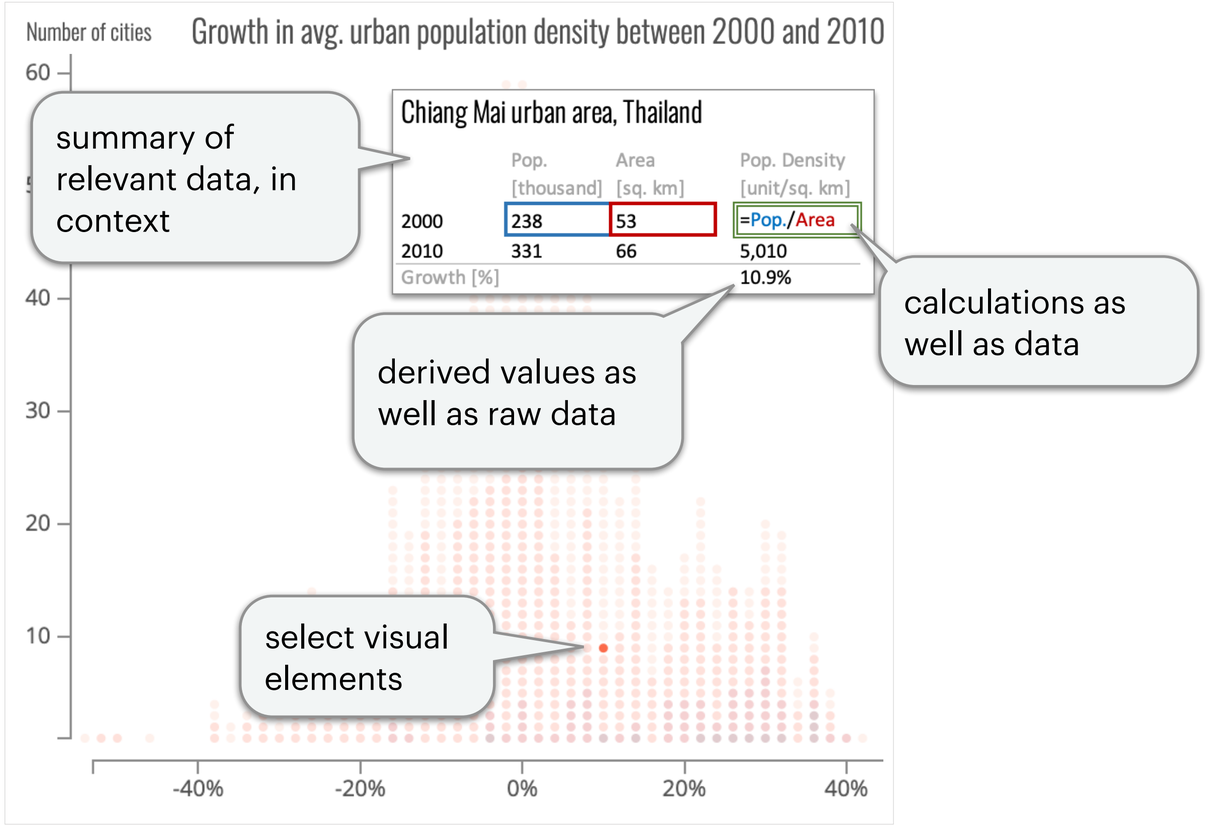
Enrich outputs with
computational explanations
(how, not just what)



Urbanization in East Asia between 2000 and 2010
Nadieh Bremer, Marlieke Ranzijn (http://nbremer.github.io/urbanization, 2015)

Next steps

Enrich outputs with
computational explanations
(how, not just what)



Urbanization in East Asia between 2000 and 2010
Nadieh Bremer, Marlieke Ranzijn (http://nbremer.github.io/urbanization, 2015)

thanks!

Contributors

Joe Bond [1]
Haofei Chen [4]
Colin Crawford [4]
Cristina David [1]
Thomas Frith [2]
Harleen Gulati [1]
Hana Iza Kim [2]
Minh Nguyen [1]
Dominic Orchard [5, 2]
Roly Perera [2, 1]
Tomas Petricek [3]
Achintya Rao [6, 2]
Meng Wang [1]

[1]University of Bristol
[2]University of Cambridge
[3]Charles University
[4]University of Edinburgh
[5]University of Kent
[6]University of West of England

https://f.luid.org
https://github.com/explorable-viz/fluid