

# The Tool Learned to Say Not Yet

A note on why Danish Politics Data had to become more honest about partial years, hidden UI bias, and what it still cannot claim yet.

2026.03.30 04:48 DENNIS HEDEGREEN BUILD V1.1

[HTTPS://HEDEGREENRESEARCH.COM/ARTICLES/THE-TOOL-LEARNED-TO-SAY-NOT-YET/](https://hedegreenresearch.com/articles/the-tool-learned-to-say-not-yet/)

---

Earlier pieces around Danish Politics Data were about two different things.

First, getting the tool into public view.

Then, making the language around it more honest.

This piece is about something else.

The instrument itself had to grow up.

The first version of Danish Politics Data had a simpler life.

It could read a fixed historical frame.

Five elections.

A known set of factors.

A more stable relationship between the data and what the interface offered back.

Then Denmark had another election.

That should have been the easy part.

Add 2026.

Pull the result in.

Move on.

It was not the easy part.

Because 2026 did not arrive as a clean finished year.

The election result existed.

The wider municipality-level factor layer did not all exist yet in the same honest shape.

That changed the problem.

The question stopped being:

“can the tool read another election?”

It became:

“can the tool tell the truth about what kind of year this actually is?”

That is a better question.

It is also a more inconvenient one.

It is easy to build a small public data tool that looks smooth.

It is much harder to build one that knows when smoothness would be a lie.

If 2026 only has part of the surrounding factor system ready, then a good tool has three easy ways to fail.

It can expose dead factor choices because those choices existed for older years.

It can quietly keep parties visible even when they do not really belong to the selected year.

And it can create hidden framing through defaults.

That last part matters more than it sounds.

An interface can introduce bias without touching the underlying numbers.

If the tool opens with one party already selected, that choice starts reading before the user has done anything.

If a new year introduces a new party but the interface treats that party as silently unselected, the omission becomes a disguised negative choice.

Nothing in the math changed.

The framing did.

That is still bias.

So the tool had to become stricter.

Not more dramatic.

More exact.

An older year can still honestly expose a broad valid surface.

»

DANISH POLITICS DATA

## Is there a pattern?

Pick one or more factors, one or more parties, and an election year. Then find out.

STEP 1 - WHICH ELECTION YEAR?

2011

STEP 2 - WHAT FACTORS ARE AVAILABLE FOR THAT YEAR?

Population Education Income Employment Welfare Crime Cars Age 65+ Turnout Immigration share Population density Unemployment

STEP 3 - PICK A PARTY

☐ All parties

A. Soc. Dems B. Soc. Liberals C. Conservatives F. Socialist PP I. Liberal Alliance Independents K. Christian Dems O. Danish PP V. Venstre Ø. Red-Green

STEP 4 - HIGHLIGHT A SPECIFIC MUNICIPALITY? (OPTIONAL)

- none -

Show me what the data reveals →

RESULT

Showing highest correlation across selected factors and parties. Use the full correlation table to inspect all results.

An older year can still support a broader valid surface.

But 2026 cannot.

Not yet.

»

DANISH POLITICS DATA

## Is there a pattern?

Pick one or more factors, one or more parties, and an election year. Then find out.

STEP 1 - WHICH ELECTION YEAR?

2026

STEP 2 - WHAT FACTORS ARE AVAILABLE FOR THAT YEAR?

Population Cars Immigration share Population density

STEP 3 - PICK A PARTY

☒ All parties

STEP 4 - HIGHLIGHT A SPECIFIC MUNICIPALITY? (OPTIONAL)

- none -

2026 municipality vote share comes from an official VALG fine-count aggregation. The wider municipality indicator refresh is still partial, so some factor combinations remain unavailable.

Show me what the data reveals →

RESULT

Results are ranked by correlation strength (absolute value). Positive = more votes where factor is higher. Negative = more votes where factor is lower.

2026 is narrower on purpose. The tool now hides what it cannot support yet.

That narrower surface is the point.

The tool now hides what it cannot support yet instead of faking a complete year.

To make that possible, the structure underneath had to change too.

The app no longer improvises directly on mixed raw tables.

Instead, the new election gets translated into a municipality-safe public layer first, and the factor layer decides what a given year can honestly support before the interface offers it back.

That means 2026 can exist publicly without pretending to be a fully refreshed normal year.

It also means:

- factors only appear when they survive the chosen year's coverage
- parties only appear when they have real presence in that year's data
- All parties is the neutral starting state
- newly relevant parties are added positively when a year switch brings them into scope

That sounds like housekeeping.

It is not.

It is the difference between a tool that calculates and a tool that knows what it is doing.

The deeper change is underneath the screen.

Danish Politics Data is no longer just a front-end sitting on top of a pile of tables.

It now has a clearer shape under itself.

A bridge layer.

A normalized public factor layer.

A firmer split between what is internal, what is reduced for public use, and what should stay out until the source path is good enough.

That last part matters too.

One factor was not forced through.

Divorces still exists as an idea inside the tool, but the current local source path was not good enough, so the honest move was to leave it out.

That is not a weakness.

That is the point.

A public data instrument should not get credit for looking complete.

It should get credit for refusing false completeness.

And the stricter surface is not there to make the tool smaller.

It is there to make the surviving results more trustworthy.



The stricter surface is there to make the surviving results more trustworthy.

Most interface mistakes in this class of tool are small enough to sound harmless.

A factor shown too early.

A party left in from a zero-only placeholder row.

A default state that looks neutral but is not.

An unavailable comparison still exposed because hiding it would require more work.

None of those errors is dramatic on their own.

Together they change how the tool gets read.

That is why this pass matters.

Not because the tool is finished.

It is not.

Not because 2026 is now fully normal.

It is not.

But because the tool is starting to know what it cannot honestly claim yet.

That is a real upgrade.

Not only in the code.

In the instrument.

Small public tools often fail in one of two ways.

Either they stay too thin to be trusted.

Or they try to look more complete than they are.

The better path is narrower.

Make the thing stronger.

Add more structure underneath it.

Let the public surface become more selective, not less.

That is what changed here.

The tool got better because it learned to say:

not this year,

not this factor,

not from this source,

not yet.

That is not hesitation.

That is what makes it an instrument.

Danish Politics Data — open the tool

Source code and data on GitHub (<https://github.com/DennisHedegreen/danish-politics-data>).

— Dennis Hedegreen

---

RELATION MEMORY

MATURES FROM [A Tool Can Be Right and Wrong at the Same Time](#) · Turns the correction into a tool behavior: the interface should say not yet when evidence is not strong enough.

RETURNS TO [A Pattern Is Not an Explanation](#) · Returns to the first method boundary with a stronger refusal layer.

---

CANONICAL URL: [HTTPS://HEDEGREENRESEARCH.COM/ARTICLES/THE-TOOL-LEARNED-TO-SAY-NOT-YET/](https://hedegreenresearch.com/articles/the-tool-learned-to-say-not-yet/)  
PDF VIEW GENERATED: 2026-07-21T21:17:20.466Z