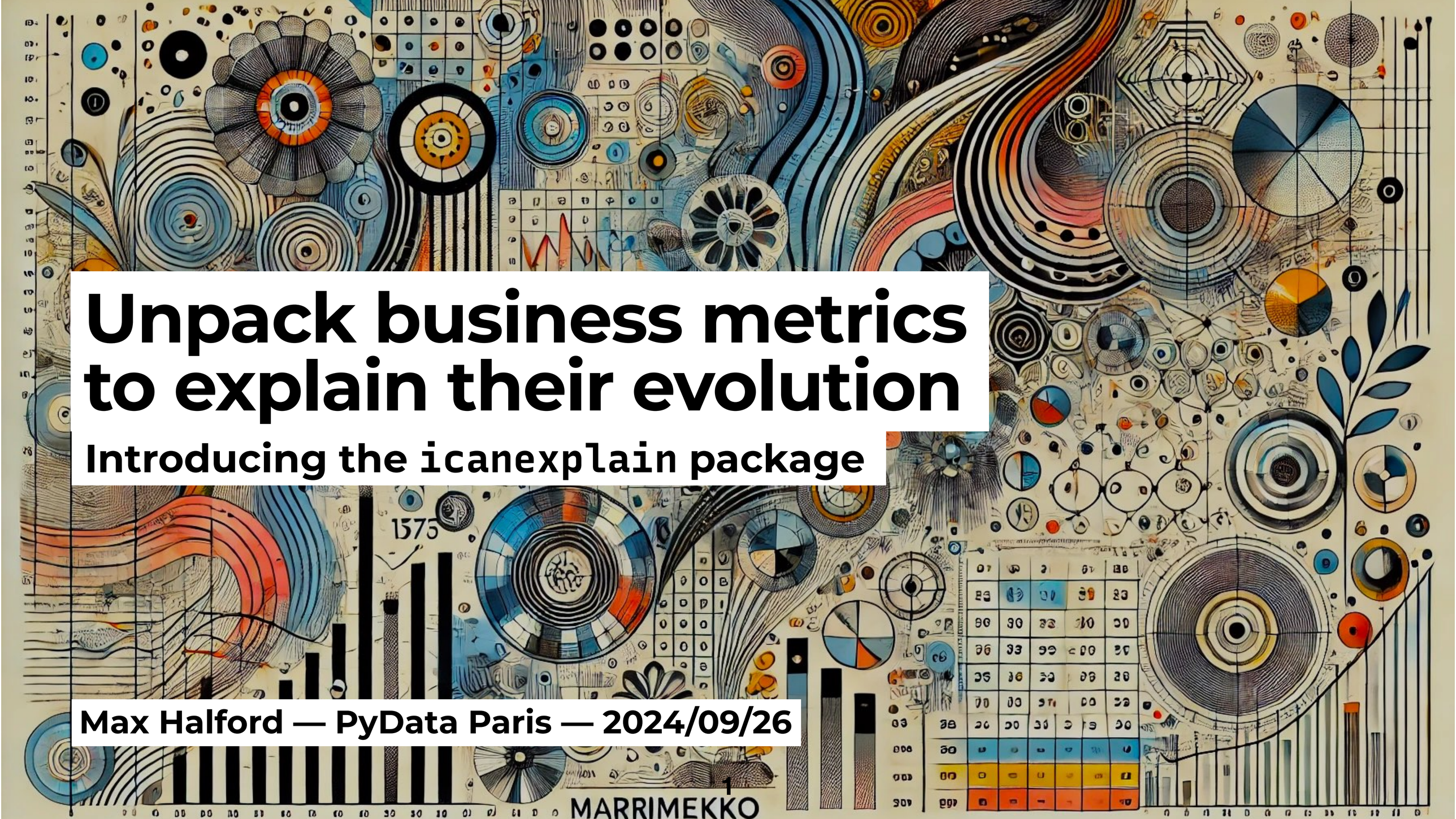


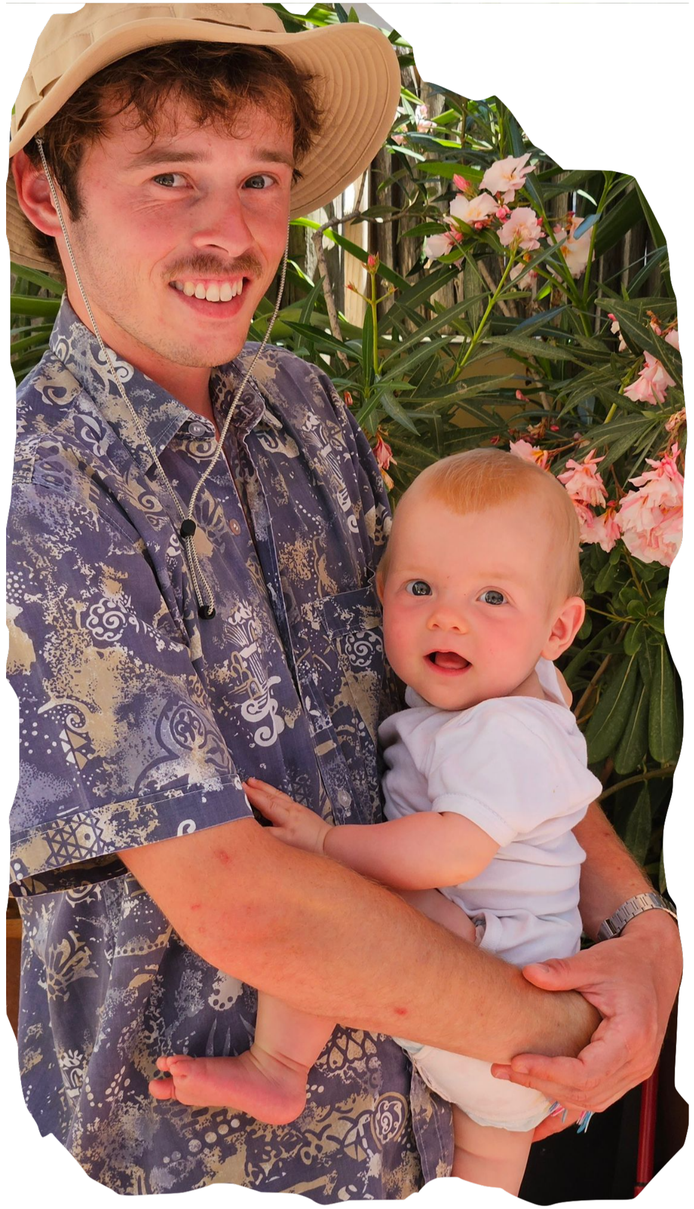
Unpack business metrics to explain their evolution

Introducing the `icanexplain` package

Max Halford — PyData Paris — 2024/09/26

Good morning 👋

- Carbonfact's Head of Data
- PhD in machine learning
- Kaggle Master
- Co-creator of River
- Second PyData talk :)
- Based in the *Landes*
- Papa of Olivia



Nobody asks questions...

Boss: hey @data, what's out current ARR?

Data: here you go, we made a dashboard

Boss: ok thx. ARR is up week on week, nice

Boss: nice dashboard btw

Data: thanks, we made it with dbt and ...

Boss: ok thx bye, see you next week

… until things go wrong

Boss: yo @data, why is ARR down week on week?

Data: not sure, there are many inputs

Boss: can you please dig?

Data: spend/customer went down

Data: number of customers is down too

Boss: how much does each impact the ARR?

Data: not sure, let me get back to you

Analytics engineering is hard

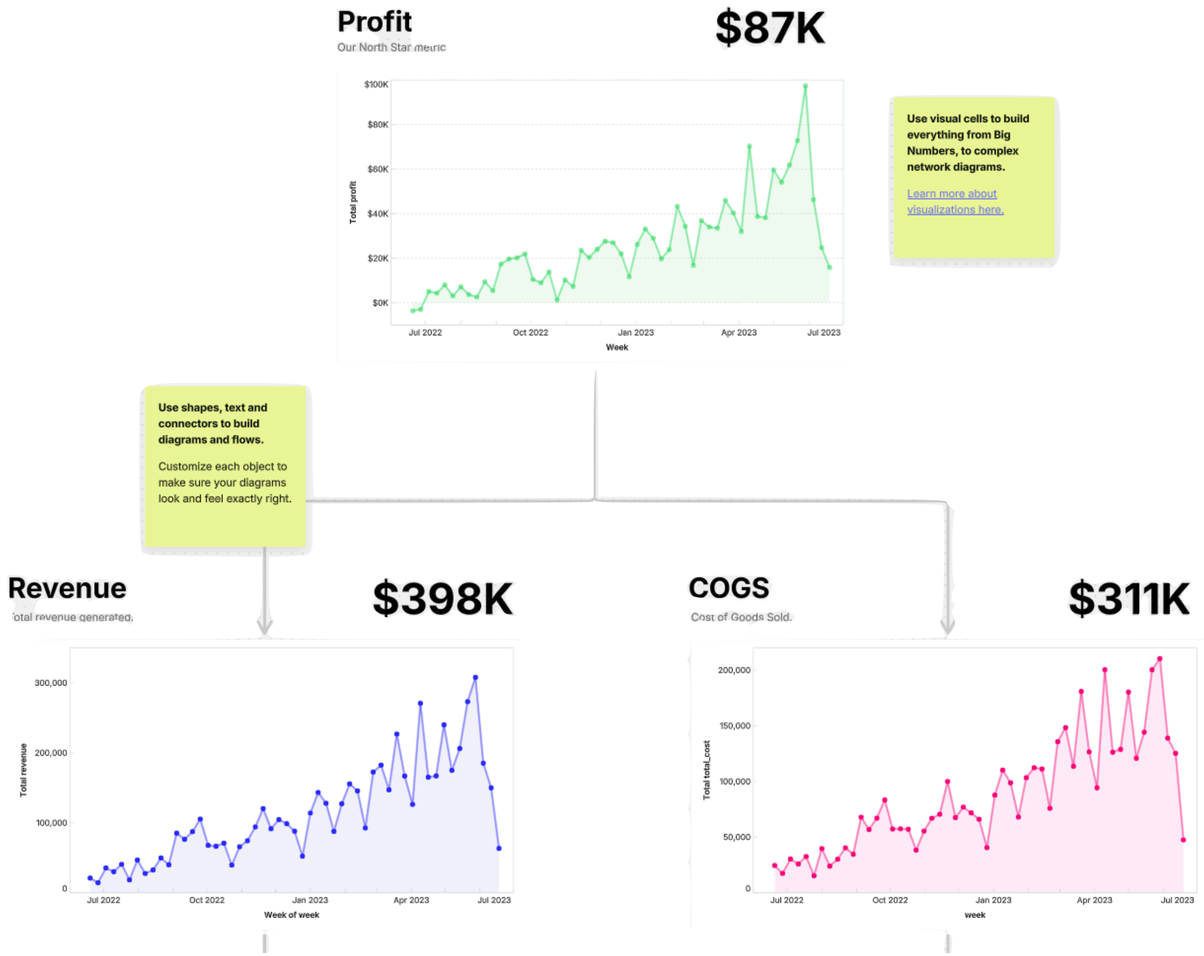
- Reporting top-line figures is not enough
- It’s fun and games as long as numbers are good
- Stakeholders want a breakdown
- Finding answers is time consuming
- Handwavy explanations break trust

Examples at Carbonfact

- Why did the average product footprint go down?
- Which product categories contributed the most?
- Why did my overall emissions go up?
- What is the impact of product mass?

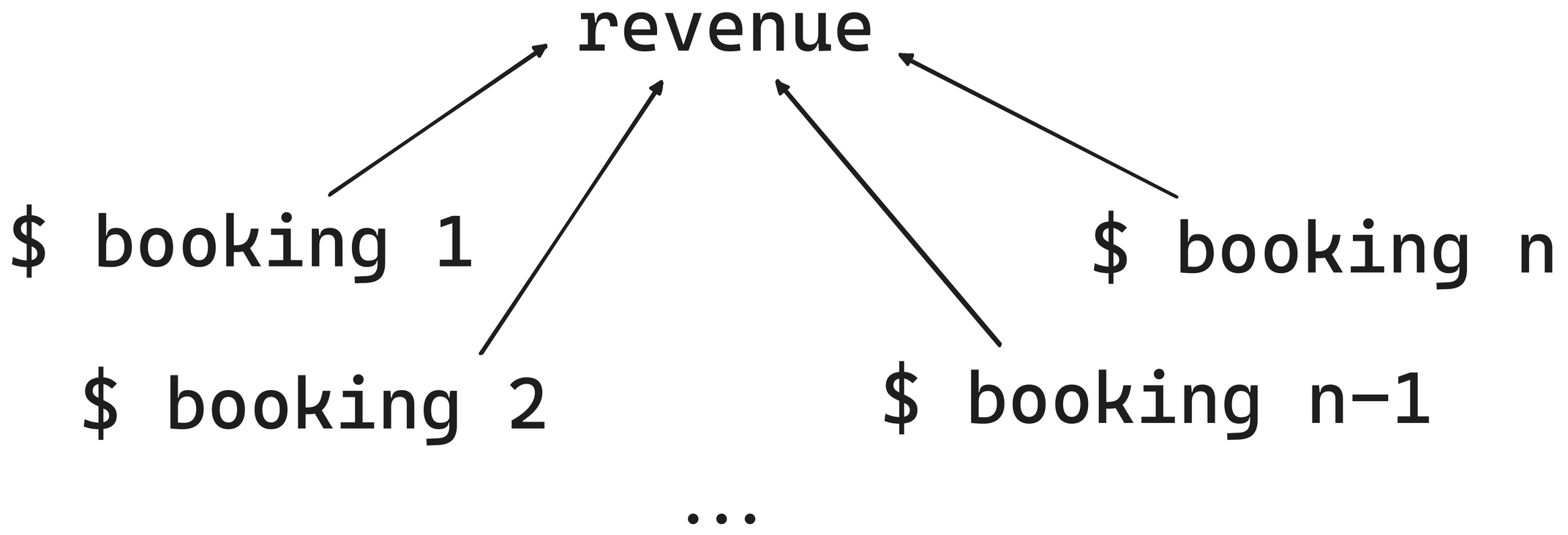
It’s often about time 🤔

Metric trees are a good start

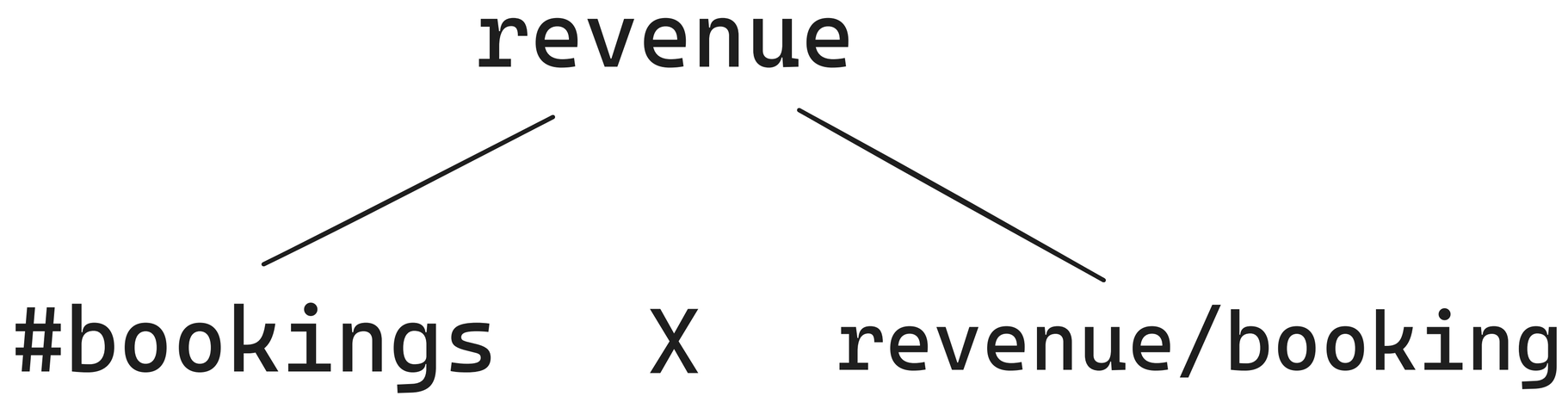


See An Introduction to Metric Trees from count.co

Revenue at Airbnb example



Revenue at Airbnb example



Revenue at Airbnb example

year	bookings	revenue_per_booking
2019	1000	$200
2020	1000	$220 (+$20)
2021	1500 (+500)	$220
2022	1700 (+200)	$225 (+$5)

Revenue at Airbnb example

year	bookings	revenue_per_booking	revenue	growth
2019	1000	$200	$200,000	
2020	1000	$220	$220,000	$20,000
2021	1500	$220	$330,000	$110,000
2022	1700	$225	$382,500	$52,500


WELL, WE CAN DO BETTER THAN
THAT.

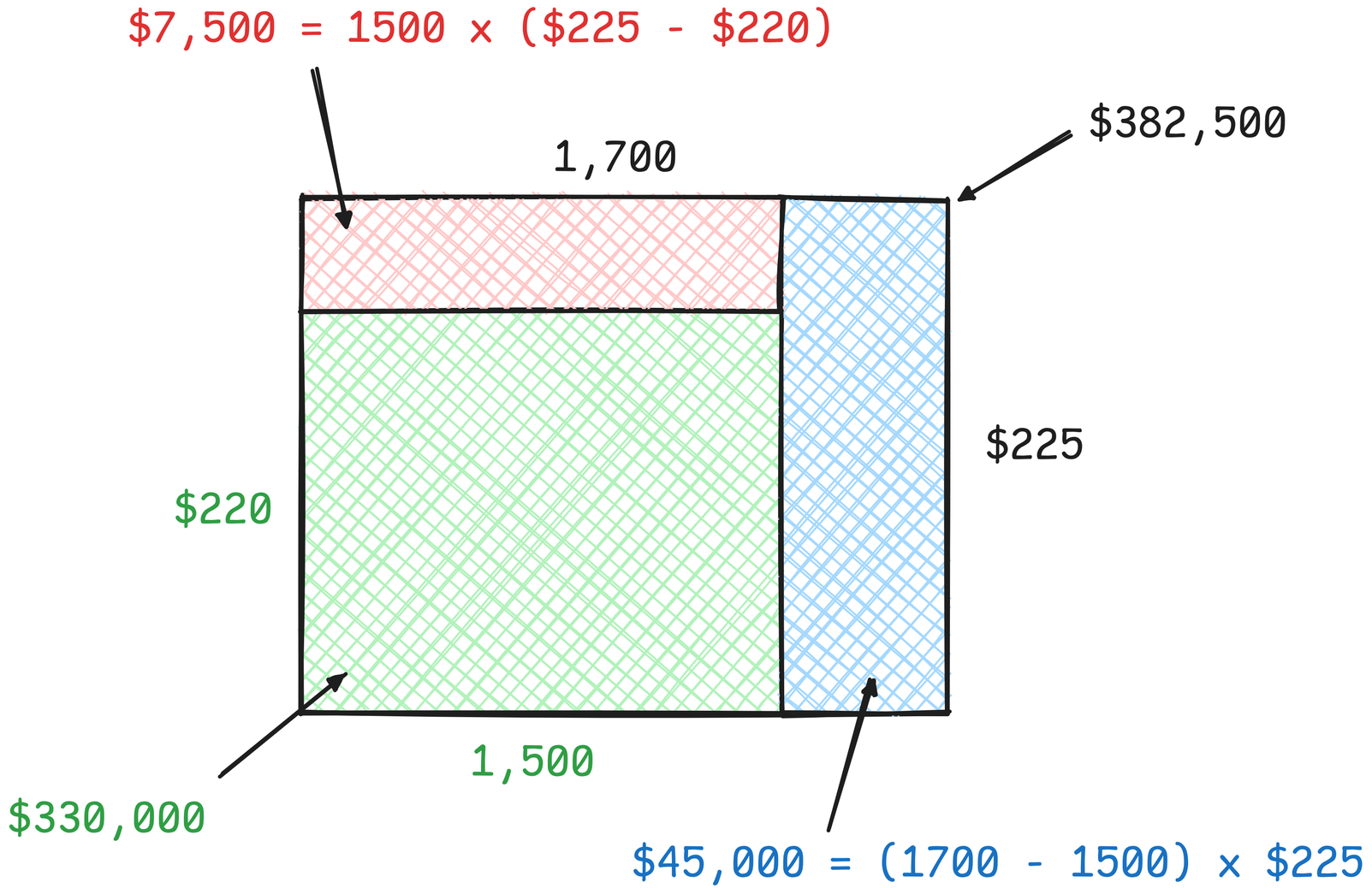
```
pip install icanexplain
```

Revenue at Airbnb example

```
import icanexplain as ice

explainer = ice.SumExplainer(
    fact='revenue_per_booking',
    period='year',
    count='bookings'
)

explanation = explainer(revenue)
```

year	inner	mix
2020	$20,000	$0
2021	$0	$110,000
2022	$7,500	$45,000

Revenue at Airbnb example



2022 = 2021 + inner + mix

Many directions to explore

- Support more formulas
- Different ways to break down a formula
- Drill-down by dimension(s)

Enough reasons to make a package :)

Example: average product footprint

year	category	footprint	units
2021	PANTS	43kgCO2e	279
2022	JACKET	49kgCO2e	1008
2023	PANTS	37kgCO2e	118
...	...	...	...

Example: average product footprint

year	average	diff
2021	21.95kgCO2e	
2022	21.71kgCO2e	-0.24kgCO2e
2023	22.74kgCO2e	+1.03kgCO2e

Example: average product footprint

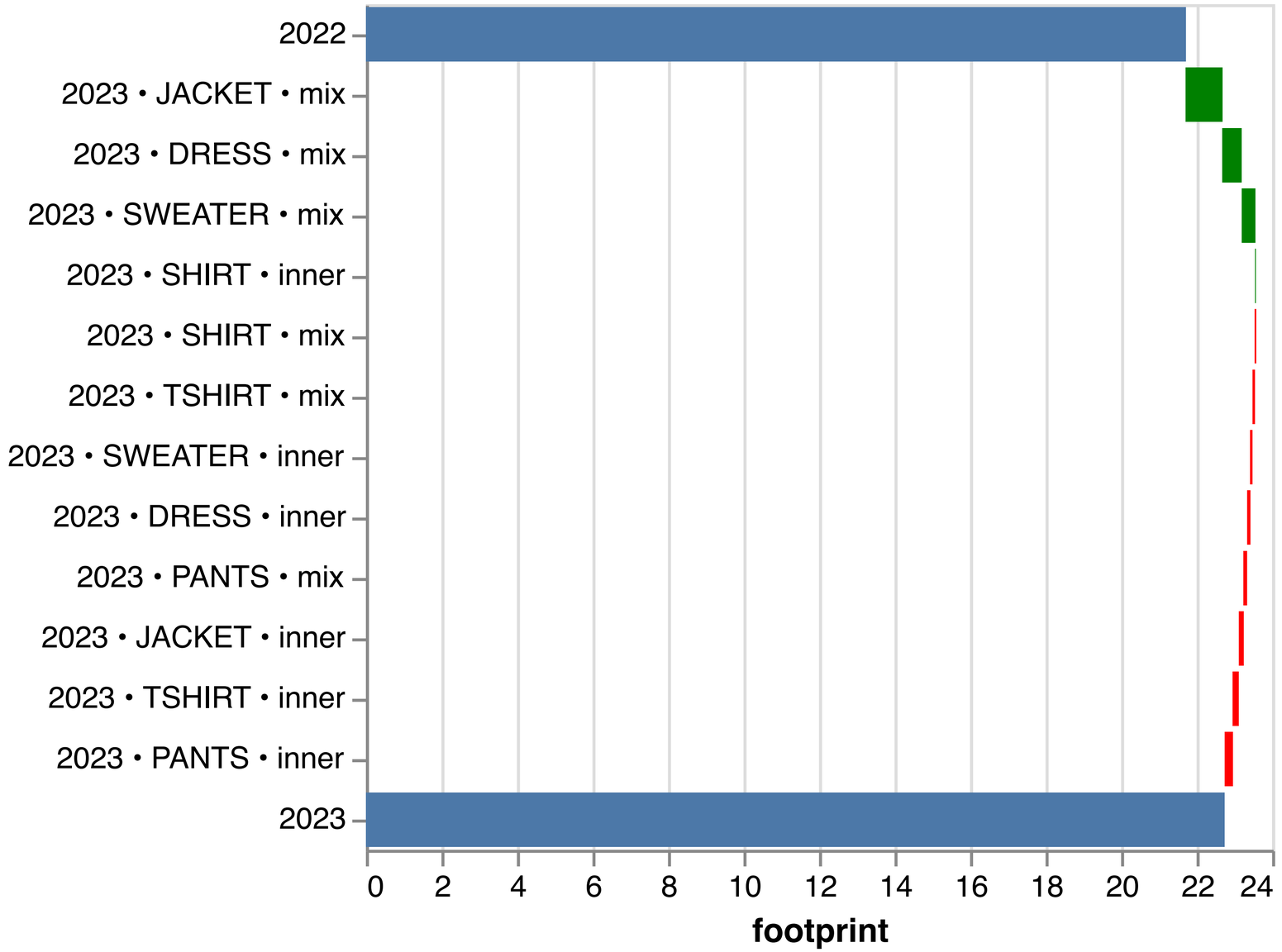
```python
import icanexplain as ice

explainer = ice.MeanExplainer(
    fact='footprint',
    count='units',
    period='year',
    group='category',
)

explanation = explainer(products)
```

year	category	inner	mix
2022	DRESS	+0.047	-0.136
	JACKET	-0.165	-0.690
	PANTS	+0.614	+0.202
	SHIRT	-0.017	+0.004
	SWEATER	-0.394	-0.009
	TSHIRT	+0.084	+0.301
2023	DRESS	-0.081	+0.512
	JACKET	-0.129	+0.969
	PANTS	-0.218	-0.088
	SHIRT	+0.017	-0.034
	SWEATER	-0.062	+0.359
	TSHIRT	-0.162	-0.056

kgCO2e

Waterfall chart

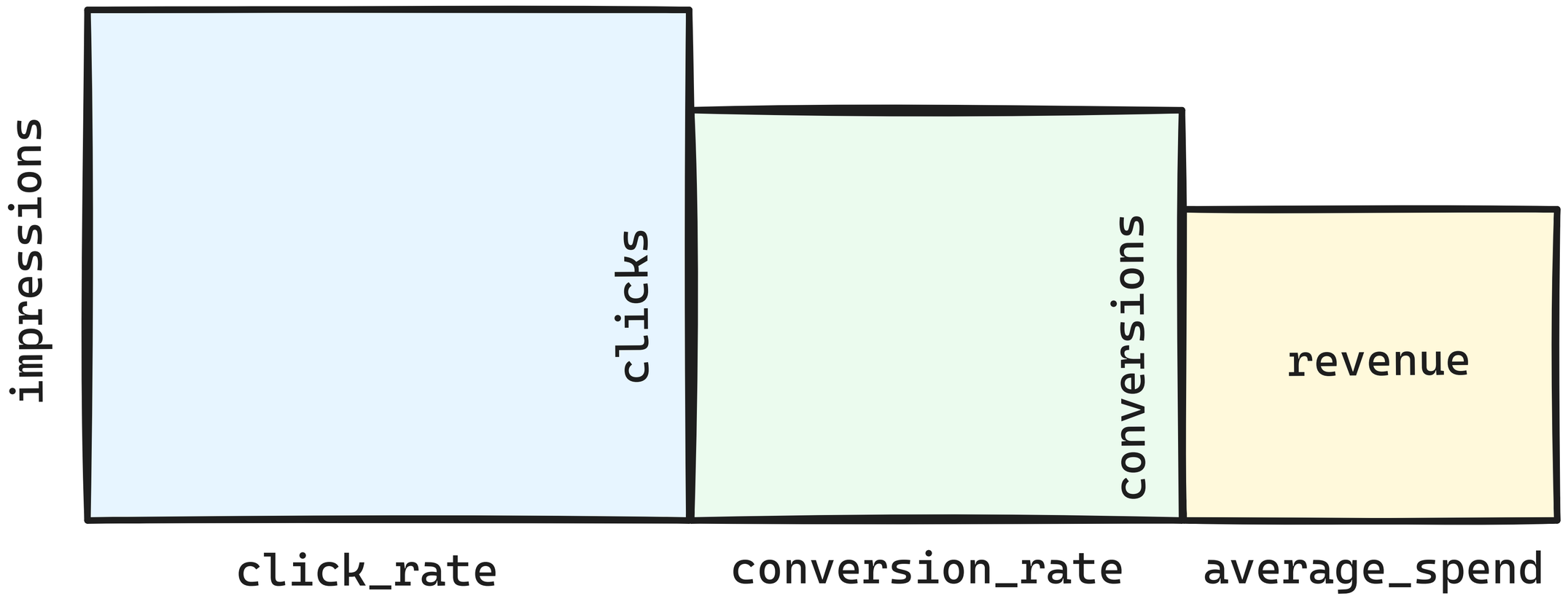
```
explainer.plot(explanation)
```



Example: website funnel

month	group	impressions	clicks	conversions	revenue
2018-01	A	1000	150	120	$8,600
2018-01	B	2000	150	150	$9,400
2018-01	C	2500	250	125	$10,750
2019-01	A	1000	120	160	$9,055
2019-01	B	2150	200	145	$8,739
2019-01	C	2000	400	166	$10,147
2018-02	A	1000	150	120	$500
2018-02	B	2000	150	150	$11,400
2018-02	C	2500	250	125	$8,750
2019-02	A	1000	120	160	$50,000
2019-02	B	2150	200	145	$10,739
2019-02	C	2000	400	166	$12,147

Example: website funnel



Example: website funnel

year	impressions	click_rate	conversion_rate	average_spend	revenue
2018	10,050	11.14%	60.71%	$72.65	$49,400
2019	11,800	20.02%	54.25%	$78.65	$100,827

Example: website funnel

year	group	impressions	clicks	conversions	spend
2019	A	+$21,233	+$29,619	-$13,753	+$12,855
2019	B	+$1,560	+$1,814	-$4,067	-$629
2019	C	-$3,900	+$12,480	-$2,184	-$3,602

Why I believe in this method

- The math is simple
- Outputs are easy to unit test (they sum up to the total difference)
- Method is easy to audit if stakeholders get nosey
- The approach is generic and can be used in different fields

Extra motivation 💪

aldbs commented on 3 Dec 2023

Thank you Max for sharing this KPI decomposition technic ! I truly think the premise you have framed at the beginning of your article in one of the most important challenge that data analysts have to face with when interacting with business teams. Since I've read your article, I have been implementing this technic and it gave me pretty interesti[illegible] triggered valuable conversations with business teams.

martindaniel4 commented on 10 Aug 2023

Fantastic writeup Max! I hope it will be useful for your readers in their daily analytical work. Here are a few comments based on my experience at Airbnb:

- I found those frameworks useful when decomposing "growth rates" such as year over year values. The maths stay the same but the business question becomes what drove the **acceleration** between two points in time.
- You alluded to it in the MECE section, but of course you can use multiple dimensions to decompose. For instance if that made sense, you could combine new / returning visitors as well as claim type.

DEC 13, 2023

SHAO Zhifei • 12:50 PM

Interesting, thank you for sharing this. I started on this about 3 or 4 years back, revised the formulas several times and finally got to this. It has been widely usd inside Grab analytics team.

Yeah agree why this is also not more popular, and yet the formula seems so intuitive

Alex Stenlake <alex.stenl... Sat, 7 Sept, 09:44

to me

Hey mate,

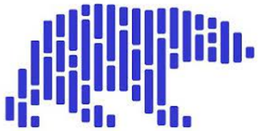
I'm Alex Stenlake, greetings from sunny Australia. Reaching out to let you know I really liked some of your older posts. I've kind of taken your initial work, generalised all of it and turned it into an initial python package. I've also taken the work and turned it into a kind of automated "key driver" analysis. Currently in the process of writing up the blog, where there'll be full backlinking to your work (and some other people who helped it click for me). I assume you won't have any objections, but given the fact that you literally left it hanging saying "it would be cool if someone did this..." I figured it was worth checking you didn't have your own work in the pipeline!

`icanexplain` — implementation

- Implemented with Ibis
- Works with different backends (Polars, pandas, SQL...)
- Wanted to use Narwhals, but no SQL support (yet!)
- ~400 lines of code

```
>>> ibis.to_sql(explanation)
```

```
SELECT
  *
FROM (
  SELECT
    "t11"."year",
    "t11"."category",
    "t11"."count_lag" * (
      "t11"."mean" - "t11"."mean_lag"
    ) AS "inner",
    (
      "t11"."count" - "t11"."count_lag"
    ) * "t11"."mean" AS "mix"
  FROM (
    SELECT
      "t10"."category",
      "t10"."year",
      "t10"."mean",
      "t10"."count",
      LAG("t10"."mean", CAST(1 AS TINYINT)) OVER (PARTITION BY "t10"."category" ORDER BY
"t10"."year" ASC ROWS BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING) AS "mean_lag",
      LAG("t10"."count", CAST(1 AS TINYINT)) OVER (PARTITION BY "t10"."category" ORDER BY
"t10"."year" ASC ROWS BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING) AS "count_lag"
    FROM (
```



github.com/carbonfact/icanexplain

`icanexplain` — future steps

- Make sure many people know about it
- Support arbitrary metric formulas
- Connect with metric trees, somehow
- Interactivity (*à la* Facets/Rill)
- Anomaly detection
- Publish a paper

If you’re interested

maxhalford.github.io/blog/kpi-evolution-decomposition

maxhalford.github.io/blog/funnel-decomposition

observablehq.com/@martinda/decomposing-metrics

medium.com/@shaozhifei/metric-decomposition-formula-to-understand-metric-trend-e693b7a4c8cf

We’re hiring

careers.carbonfact.com

maxhalford25@gmail.com