Dataset Representativeness and Variability

Whose data did we test on, and how far does the number travel?

Fabio Casati

What is the test dataset telling us

Variability

The system performs differently across clients, domains and languages. There is no single true number.

Representativeness

Our data may be one language, one customer, or synthetic. It may not represent the population.

The score is a fact about the test set. How far does it travel?

The system performs differently in each domain

Running example: one AI support assistant, sold to several clients, answering finance, legal and medical questions.

Correctness on a 1–5 scale, per domain. In this simulation we know these true scores; in real life you never do.

Same system, three levels of quality. Every number we report is a mix of these.

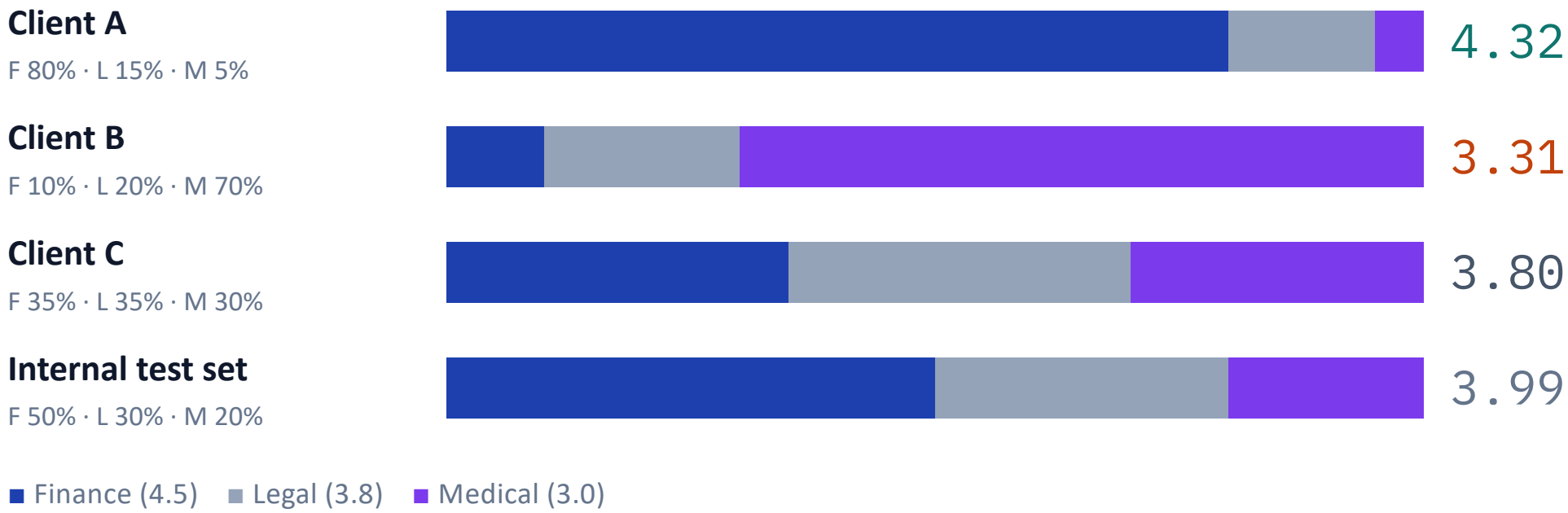
Variability

Clients and domains really differ.

The spread is a property of the world, not of the measurement.

How we model variability

The system performs differently by domain: finance 4.5, legal 3.8, medical 3.0. Each client sends a different mix.



Client's score = its mix × the system's per-domain scores. Same system, different numbers.

Same system, three clients

True correctness each client experiences, 1–5 scale. Spread across clients: 1.01 points.

"Our system scores 4.0" is true for nobody in particular.

Playground: one system, many clients

1. Step 1: read the internal scorecard and decide whether to ship.
2. Step 2: deploy to the three clients. Who is happy, who is not?
3. Drag a client's mix toward medical. Watch its score fall while the system stays the same.

Open the Domain Variance lab

It also performs differently across languages

Correctness on a 1–5 scale, per language. Same numbers as the Sampling Bias lab.

Domain is one axis. Language is another. Each one splits the average again.

Some differences we can test. Some we do not even see.

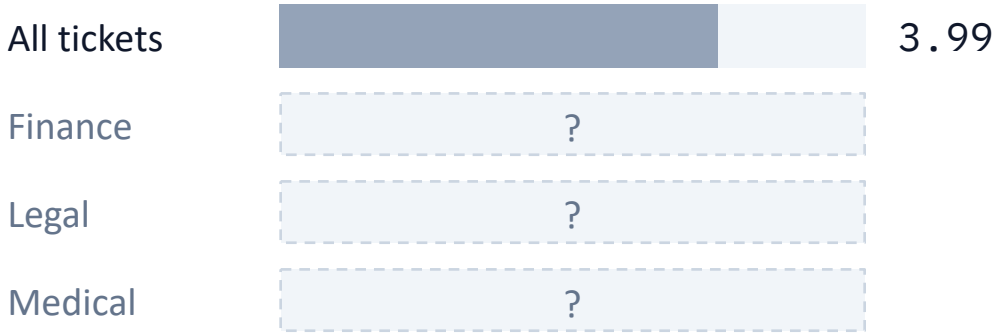
Known: language

We suspect it, and every ticket has a language field.

Easy to test: slice by the field, compare.

Latent: domain

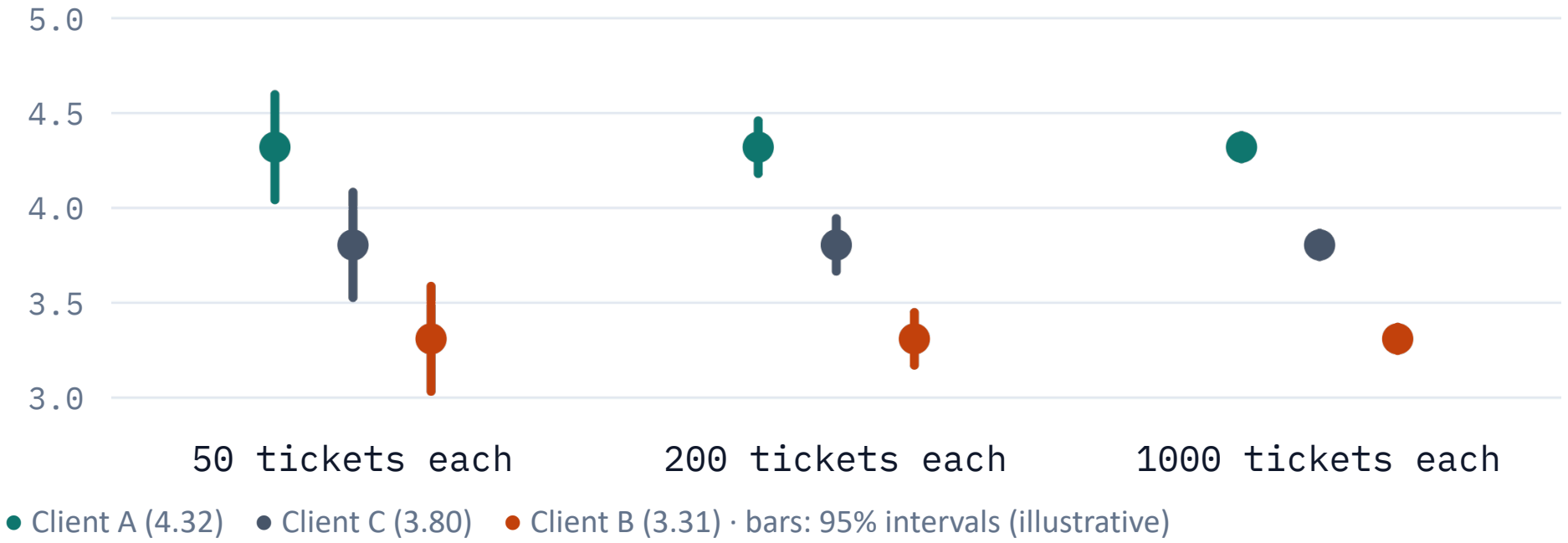
Nobody suspected it, and no ticket says "medical".



Hidden inside the average until someone looks.

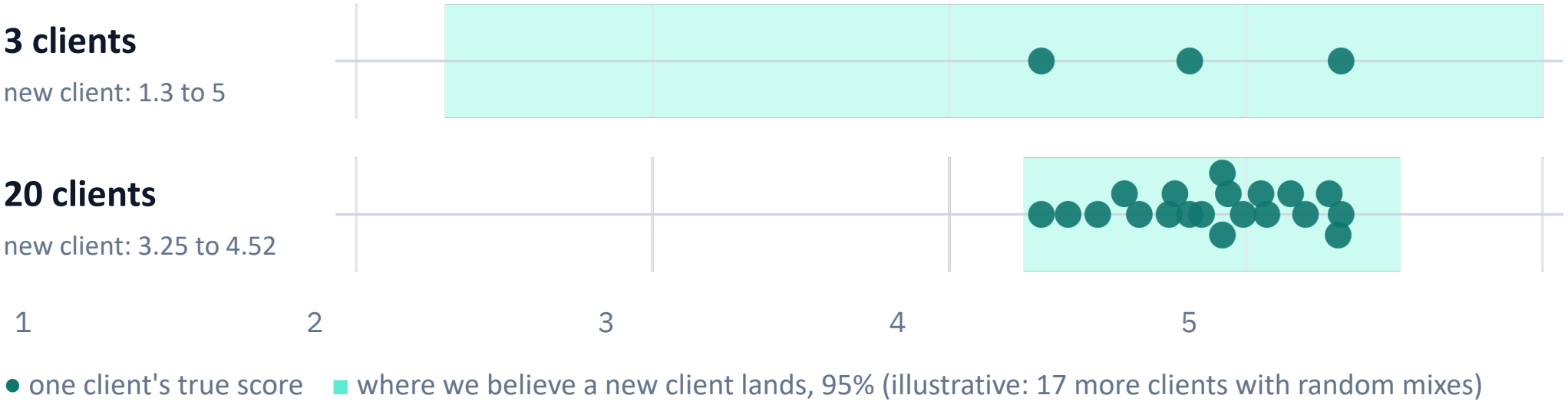
You can only test the variability you thought of, or the variability your data has a column for.

More tickets per client do not close the gap



More tickets sharpen each client's number. They do not move the clients closer.

More clients: each client is one data point

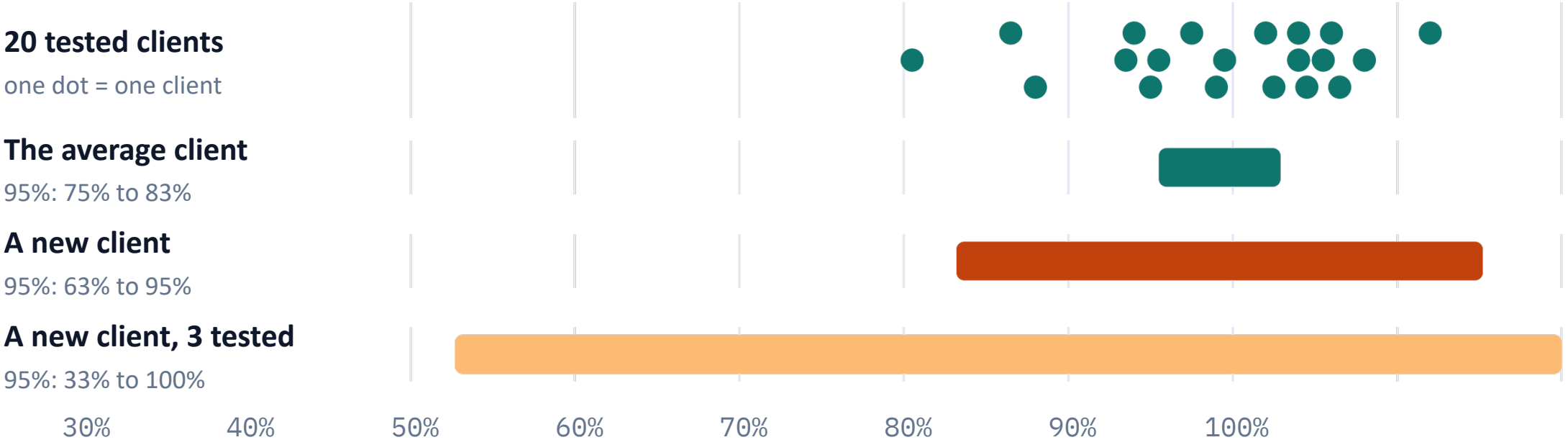


For "what will a new client see?", 10,000 tickets from 3 clients are **3 data points**.

More tickets sharpen each client's number. More clients show how far a new one can land.

After testing N clients: what we believe about a new one

Metric switch: from the 1–5 score to accuracy, the share of tickets answered correctly, as in the sample-size lesson. 20 clients, 200 tickets each (simulated).



P(a new client's accuracy > 70%) ≈ **88%**

The average is known to a few points. A new client can land anywhere in a 30-point range.

How to build the belief about a new client

1

Score each client

Accuracy from its own tickets, with its own interval

2

Estimate the spread

Mean and spread across clients, minus the ticket noise

3

Predict a new client

Mean ± about 2 × spread, wider when few clients were tested

Report two numbers: the average client, and the range for a new one.

Playground: what will the next client see?

1. Calculator, tab *A new client*: the 20 clients of the slide. Read the two beliefs: the average client, and a new one.

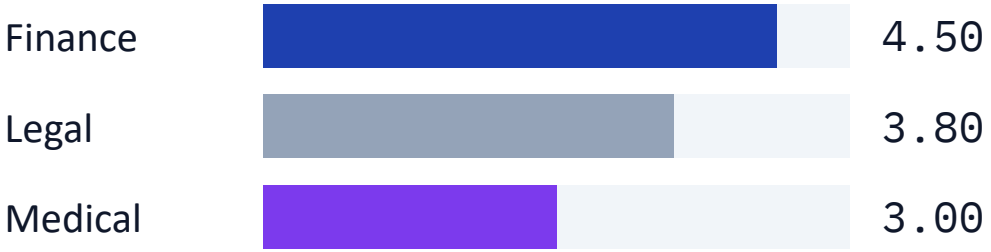
2. Click *Only the first 3 clients*, then *Same 20, 2,000 tickets each*. Which one changes the range for a new client?

3. Paste your own clients, one per line: name, tickets, correct.

Open the calculator: A new client

What does close the gap: a system that adapts

Before



Clients range **3.31–4.32**

spread 1.01

After adapting the system

Clients range **4.07–4.42**

spread 0.35

Illustrative: domain-specific prompts and retrieval lift medical 3.0 → 4.0 and legal 3.8 → 4.1. Client mixes unchanged.

The spread is a property of the system. Only changing the system shrinks it.

Implications of variability

One number hides the spread

No single score describes every client

More data does not help

Intervals shrink, the gaps between clients stay

Latent slices are invisible

No column, no breakdown, no warning

A new client is a prediction

Its score depends on its mix, which you may not know yet

Report a range across clients, not just an average, and expect slices you have not found yet.

How to measure variability: the axes you know

1

Slice

By client, language, product: every axis your data has a column for

2

Estimate each slice

With its own interval; small slices get wide ones

3

Report the spread

Between slices; with many clients, the range a new one will land in

Two kinds of uncertainty: within a slice (shrinks with data) and between slices (does not).

How to measure variability: the axes you don't know

1

Read the failures

Low-scored answers: what do they have in common?

2

Tag every case

Cluster, or let an LLM label topic and intent

3

Confirm on fresh data

Search many slices and some differences are luck

Discover slices on one sample, confirm them on another.

Playground: find the hidden slice

1. Dashboard by client and channel: Client B is lower. Why? No column says.
2. Read 12 failures. What do they have in common? Then tag every ticket with a noisy LLM and slice by topic.
3. Search 20 slices, then confirm on a fresh sample. Which slice survives?

Open the Hidden Slices lab

How to mitigate variability

Adapt the system

Per-domain prompts, retrieval, routing, client examples

Evaluate per client

Onboarding check on each new client's own data

Monitor per slice

Track each client and domain after launch, not just the total

Promote latent slices

Once confirmed: a column, a floor in the sample, a line in the report

Only the system can close the gap. The evaluation tells you where.

Representativeness

The data we test on is not the population we will face. The score is precise, and answers a narrower question.

Three common ways our data is not the population

Only one language

Usually English. Every other language: no estimate at all.

Only one customer

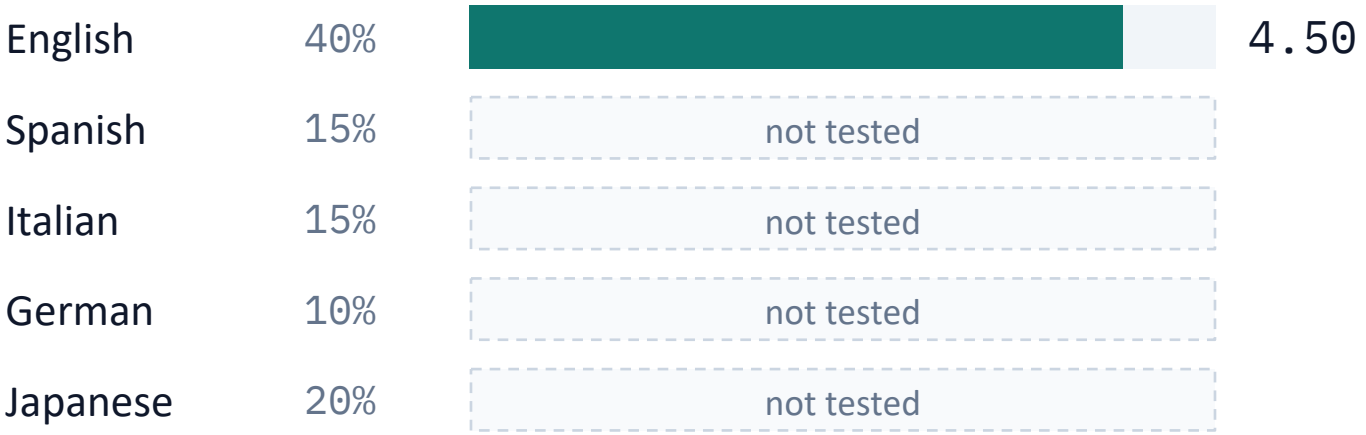
Its mix, its documents, its jargon. One customer is one data point.

Synthetic data

Generated requests: cleaner and simpler than real ones.

Often all three at once: generated tickets, in English, modelled on the pilot customer.

Only one language



We report

4.50

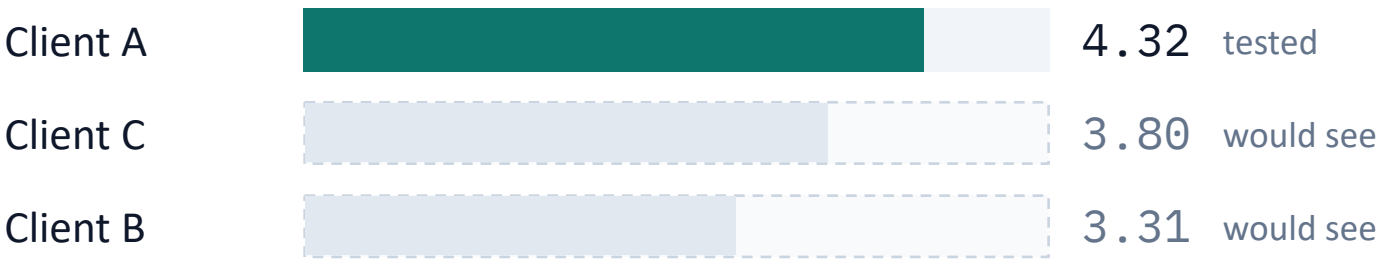
Users get

3.85

Percent: share of production traffic. Same numbers as the Sampling Bias lab. "Users get" uses the true per-language scores, which we could not see.

We have a precise number for English, and no number at all for 60% of the traffic.

Only one customer



Client A's specifics

Its mix: 80% finance

Its documents and policies

Its jargon and ticket templates

Its setup of the product

And for "what will the next customer see?", one customer is **one data point**.

A pilot customer's score is a fact about that customer, not about the product.

Playground: one language, one customer

1. Sampling Bias lab: the test set starts English only. Read the **Correctness** row, our numbers, and the languages with no estimate.

2. Reveal production, then Fix it. The fix is cases in every language, not more English cases.

3. One customer: Domain Variance lab, step 2. If you had tested only Client A, what would B and C see?

Sampling Bias lab · Domain Variance lab

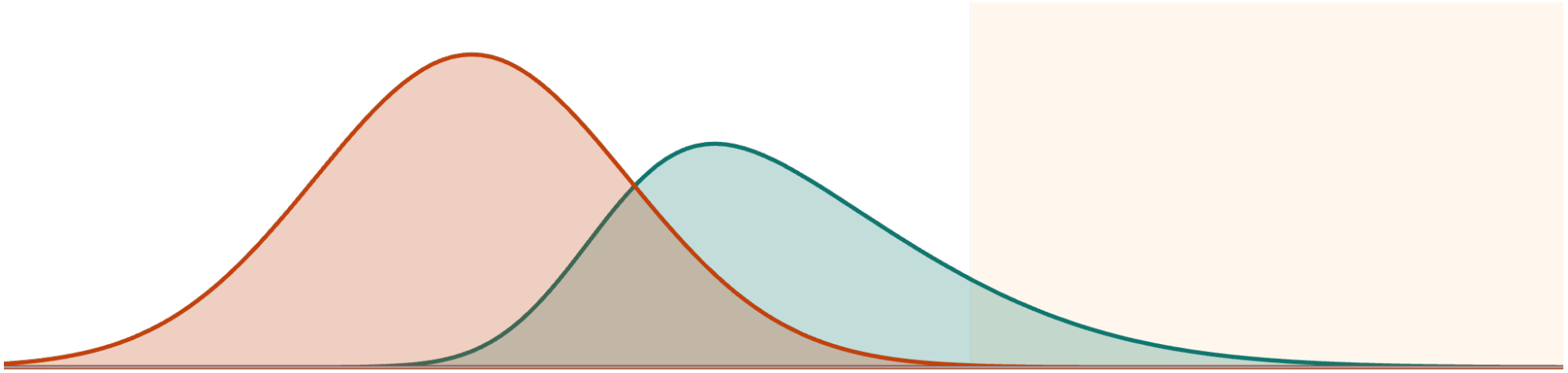
Synthetic data: never quite like real data

	Synthetic	Real
Wording	Clean, complete sentences	Typos, fragments, jargon, other languages
Intent	One clear question	Several questions, or an unclear one
Context	Self-contained	"As I said in my last ticket…"
Mix	What the prompt designer imagined	What users actually send
Style	One model's habits	Thousands of people's habits

A synthetic test set is a non-representative test set, almost by construction.

Synthetic data covers a different mix

■ **Synthetic requests** ■ **Real requests** Shaded: real cases synthetic rarely covers



clean, one clear question how messy a request is → typos, several intents, missing context

Illustrative shapes, not measured data.

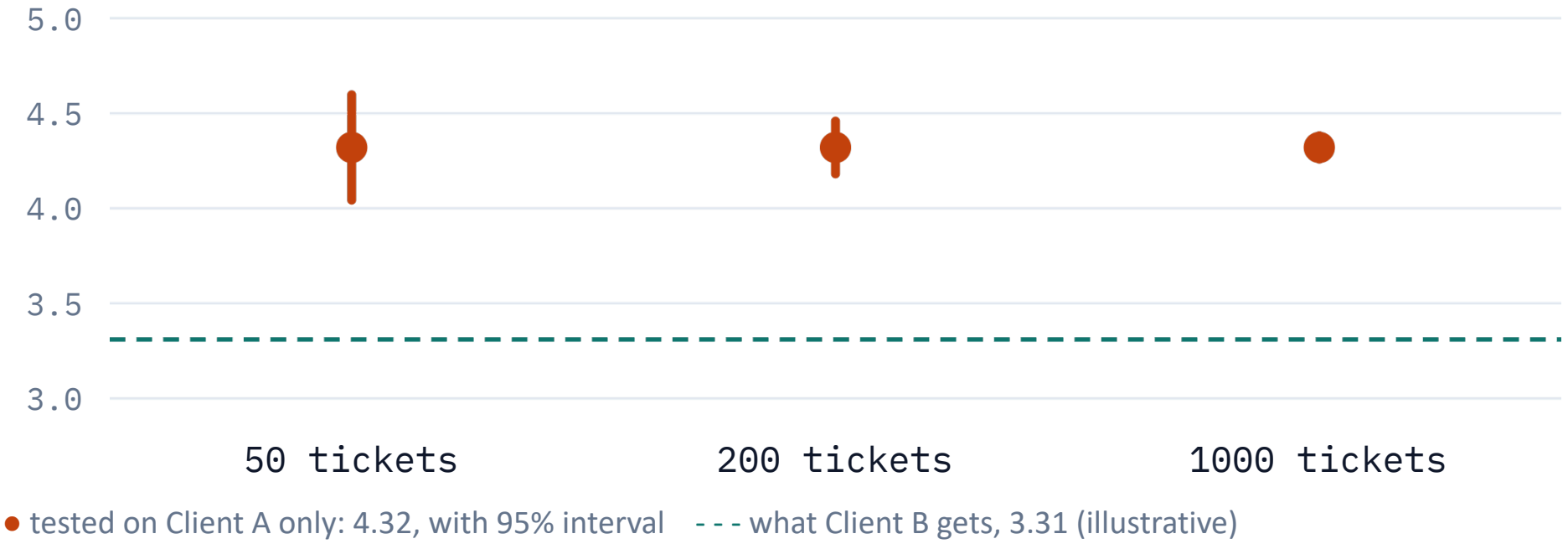
The gap between synthetic and real has a size and a direction you cannot know without real data.

Playground: synthetic vs. real

1. Score 300 synthetic cases, then 10,000. The interval narrows. Does the score move?
2. Add 200 real cases, then reveal the real population. Which belief was right about the level?
3. Seed the synthetic cases from real requests. How much of the gap is left?

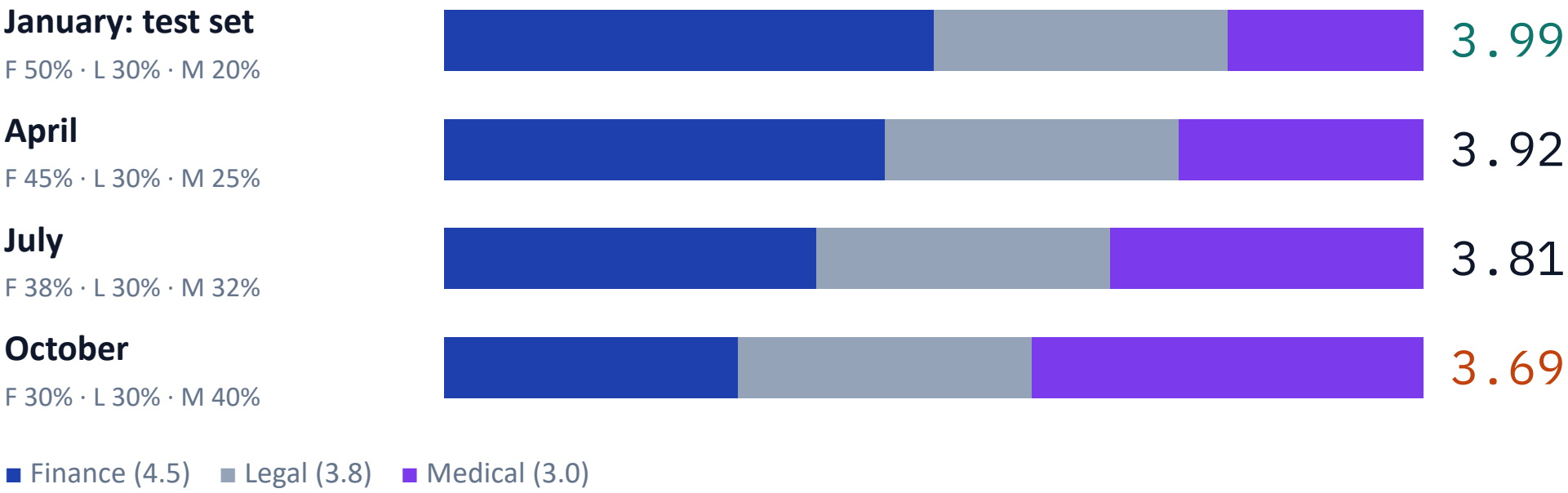
Open the Synthetic vs. Real lab

More data of the same kind: the same wrong answer



More English tickets, more Client A tickets, more synthetic tickets: the interval narrows around the wrong value.

The population moves: a test set describes the past



Production mix each quarter, as more health-tech clients sign up. Same system, same per-domain scores. Illustrative.

Playground: Temporal Drift lab: a year of drift and a sudden shock. When would monitoring catch it?

A test set represents traffic when it was collected. Refresh it from recent traffic, and monitor.

Scope the claim to the data

Our data has…	We can claim	We cannot claim
Only English tickets	Quality on English tickets	Quality for Japanese users, or overall
Only Client A	Quality for Client A	Quality for the next customer
Only synthetic tickets	Quality on generated requests	Quality on real requests
Only January tickets	Quality on January's traffic	Quality today

Every score holds for the population the data represents. Say which one.

Implications of non-representative data

Precise and wrong

The interval narrows around our data, not around the population

More data does not help

More of the same language, customer or generator repeats the error

Gaps leave no trace

Untested languages, other customers, the messy real tail

A score answers: how good is the system on this data? Pilots and synthetic sets are usually optimistic.

How to measure representativeness

1

Describe the population

Languages, customers, request types, from real traffic or the sales plan

2

Compare with our data

What is covered, what is thin, what is missing; synthetic or real

3

Check on a real sample

Another language, a second customer, real requests: score a small set

You can only see a mismatch by looking at data from the population. Without it, the gap is unknown, not zero.

How to mitigate non-representative data

Start from real traffic

Sample real requests; seed any synthetic cases from them

Cover what is missing

Every target language; many customers rather than many tickets from one

Synthetic for gaps only

Rare or new slices, adversarial probes; never the headline

Scope and label

State the population; report synthetic and real separately

The fix is representative data, not more data. Synthetic data is for coverage; real data is for the claim.

Exercise: your system, your data

1

List the axes

Where may your system vary? Clients, languages, domains, channels, time

2

Known or latent?

Is there a column for it, or would you have to read and tag?

3

What the data covers

For each axis: which values your test data has, and how many cases

4

Scope the headline

Rewrite your current headline claim so it says whose data

Template

“__% correct on **whose** requests __, in **which** languages __, **real or synthetic** __, collected **when** __. Not yet tested: __.”

10 minutes, in pairs. Then: which missing row would change the headline most?

Two questions for any test set

Whose data?

Report the range across clients, domains and languages. Close the gap by adapting the system.

Does it represent the population?

One language, one customer, synthetic, old? Scope the claim, and check on real data from outside.

Next lesson: how to select and sample the data so the numbers carry the claim.