NOVASYNE

Analytics Discovery Workbench

User Guide

Import a table, let the workbench work out what is in it, then profile it, test it, model it and write it up — with every refusal explained.

Version 1.0 August 2026

Novasyne · Signal analysis for clinical and translational research

Contents

Before you start

The Analytics Discovery Workbench is a browser-based statistical bench for tabular research data. You give it a table — one row per observation, one column per measurement — and it works out what each column is, then lets you profile it, plot it, test it, model it and write up what you did.

It is deliberately domain-agnostic. Biomedical, chemical, environmental and financial data are equal citizens: column roles and units are read from the data rather than assumed from a field.

This guide is task-oriented. Each chapter answers a question you might arrive with — *how do I compare my arms?*, *how do I get a limit of detection?*, *how do I stop a model cheating?* — and shows the screens you will actually see. Appendix A is the control-by-control reference if you would rather look something up than read forward.

What you need

The application	Running and reachable in a browser. See the README shipped with the package for installation.
Your data	A CSV or TXT file. One row per observation, one column per measurement, column names in the first row.
An API key	Only for the Methods writer and the AI assistant. Everything else runs offline.

> **SET SECRET_KEY BEFORE YOU START USING IT IN EARNEST**
>
> Left unset, the application generates a new signing key every time it starts, so restarting it invalidates the page already open in your browser. The next thing you click reports that the security token is no longer valid and asks you to reload — nothing is lost, but it is avoidable. Set `SECRET_KEY` in `.env` once and restarts stop mattering. The README says how.

The four datasets in this guide

No single file exercises the whole bench, so this guide uses four. Every figure names the one it came from.

Dataset	Shape	Used for
`multimodal_cohort.csv`	360 × 35	The spine of the guide. 120 subjects × 3 visits × 3 arms, spanning blood, cortisol, wearable, EEG and self-report.
`dpv_voltammograms_long.csv`	649 × 3	Curve fitting — a differential-pulse voltammogram with a peak to fit.
`calibration_standards.csv`	27 × 4	Calibration — nine standards in triplicate, with saturation at the top end.
`daily_rentals.csv`	731 × 11	Changepoints, periodicity and forecasting — two years of daily counts.

> **WHY THE COHORT FILE MATTERS**
>
> `multimodal_cohort.csv` has effects planted in three tiers: some strong enough to detect at this sample size, some real but underpowered, and some with no effect at all. That last tier is the point. A bench that finds something wherever it looks is not telling you anything, and a guide that only ever shows a significant result teaches you to expect one.

The workspace

What you see before any data arrives

Open the workbench and it is empty on purpose. Nine tabs of controls over nothing would be worse than one instruction, so the panes stay hidden until a file arrives.

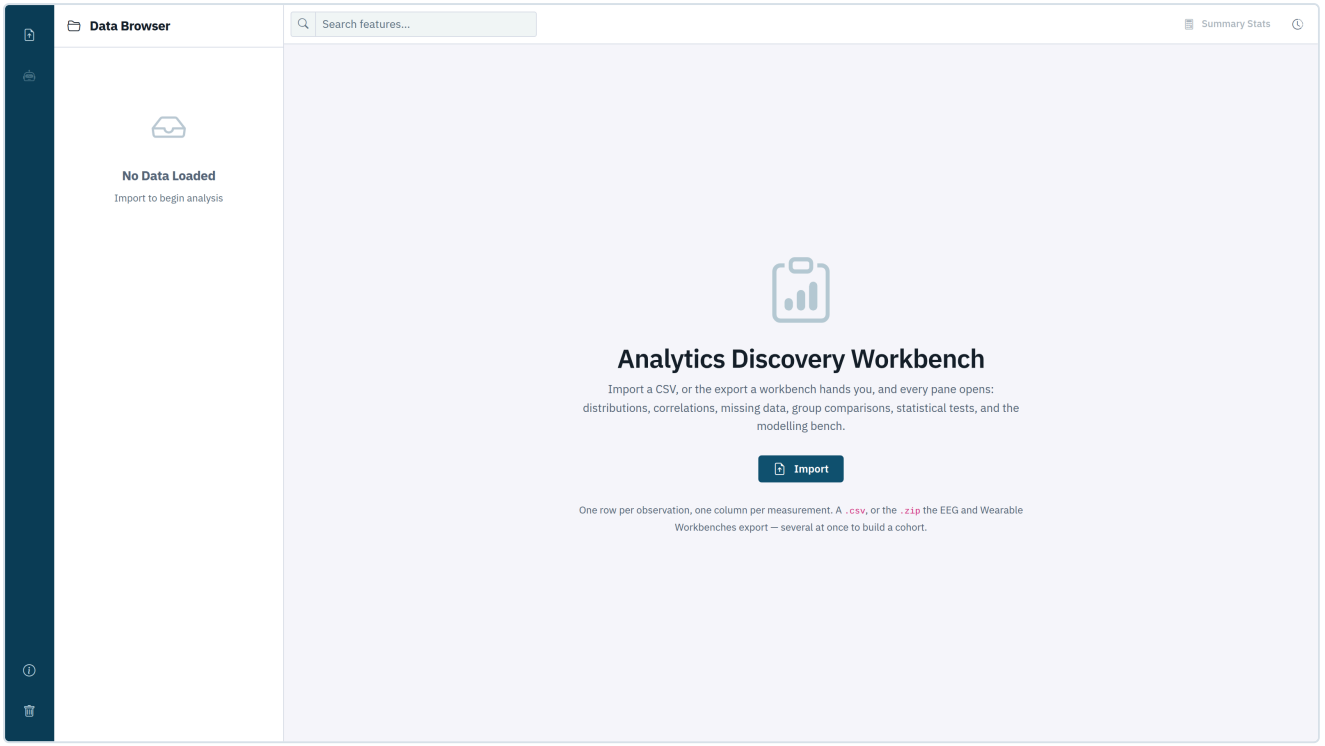

Figure 1 — The workbench on first open. The only action available is Import.

Three regions matter, and they stay put for the rest of your session:

The rail, far left, is fixed: Import, the AI assistant, About, and Clear Session at the bottom. Clear Session discards the loaded data and the analysis history — it is the only destructive control in the application.

The Data Browser, the wider left column, describes whatever is currently loaded: warnings first, then the file name and its shape, then every column grouped by domain.

The main area holds the nine tabs and the toolbar above them.

Once a file is loaded

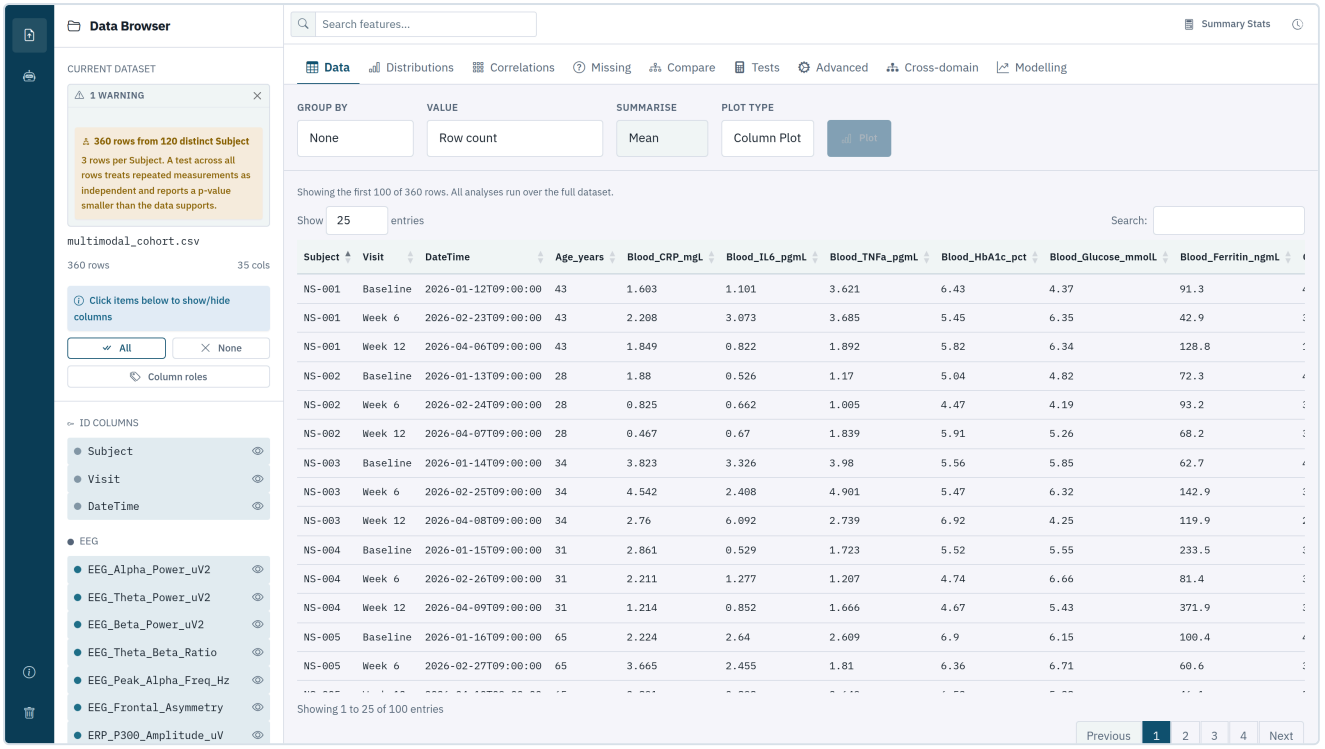

Figure 2 — The workspace with the cohort file loaded. Nine tabs, all enabled.

The tabs run left to right roughly in the order you would use them: look at the data, look at its shape, compare, test, then model.

Tab	What it answers
Data	What is actually in the file, and a quick plot off any two columns
Distributions	What shape is this measurement, and does it differ by group
Correlations	What moves with what
Missing	What is absent, and whether the absence has a pattern
Compare	How do my groups differ, with intervals
Tests	Is that difference more than noise
Advanced	Components, periodicity, changepoints, trajectories
Cross-domain	How do whole domains relate, and how do I combine them
Modelling	Fit, predict, cluster, curves, calibration, forecasting

Importing your data

The Import dialog

Click **Import** on the rail. You can drop a file onto the dialog or browse for it.

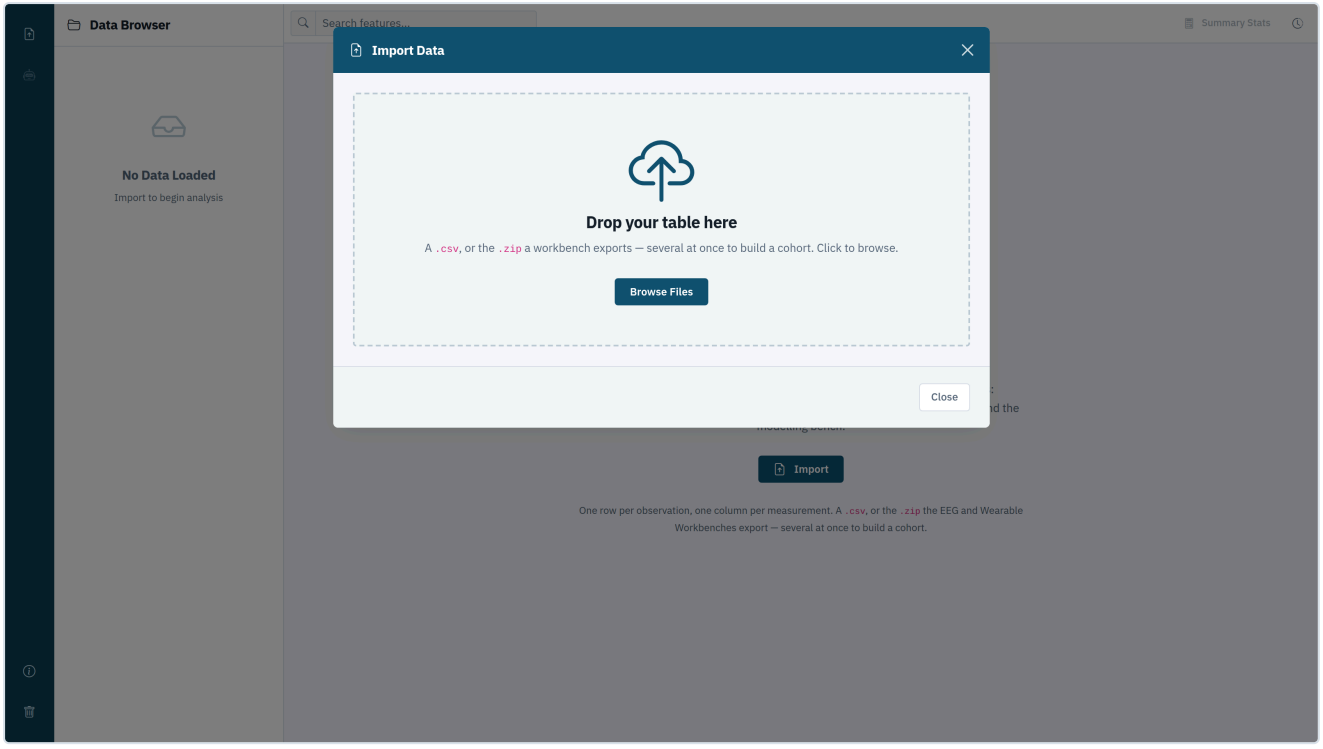

Figure 3 — The Import dialog accepts a drop or a file picker.

The workbench reads the first row as column names, and reads the units in those names with them — `Blood_CRP_mgL` is understood to be a CRP measurement in mg/L, and every axis label and Methods sentence downstream will say so. You do not have to name columns this way, but you get better output if you do.

What a good table looks like

One row per observation. If a subject was measured at three visits, that is three rows, not one row with three columns.

One column per measurement. Keep identifiers, factors and measurements in separate columns.

Real blanks for missing values. Leave the cell empty. Do not use `-999`, `N/A` or `0` to mean "not measured" — the workbench looks for sentinel values like these and will warn you, but it cannot always tell a real zero from a stand-in.

Import confirms what it read

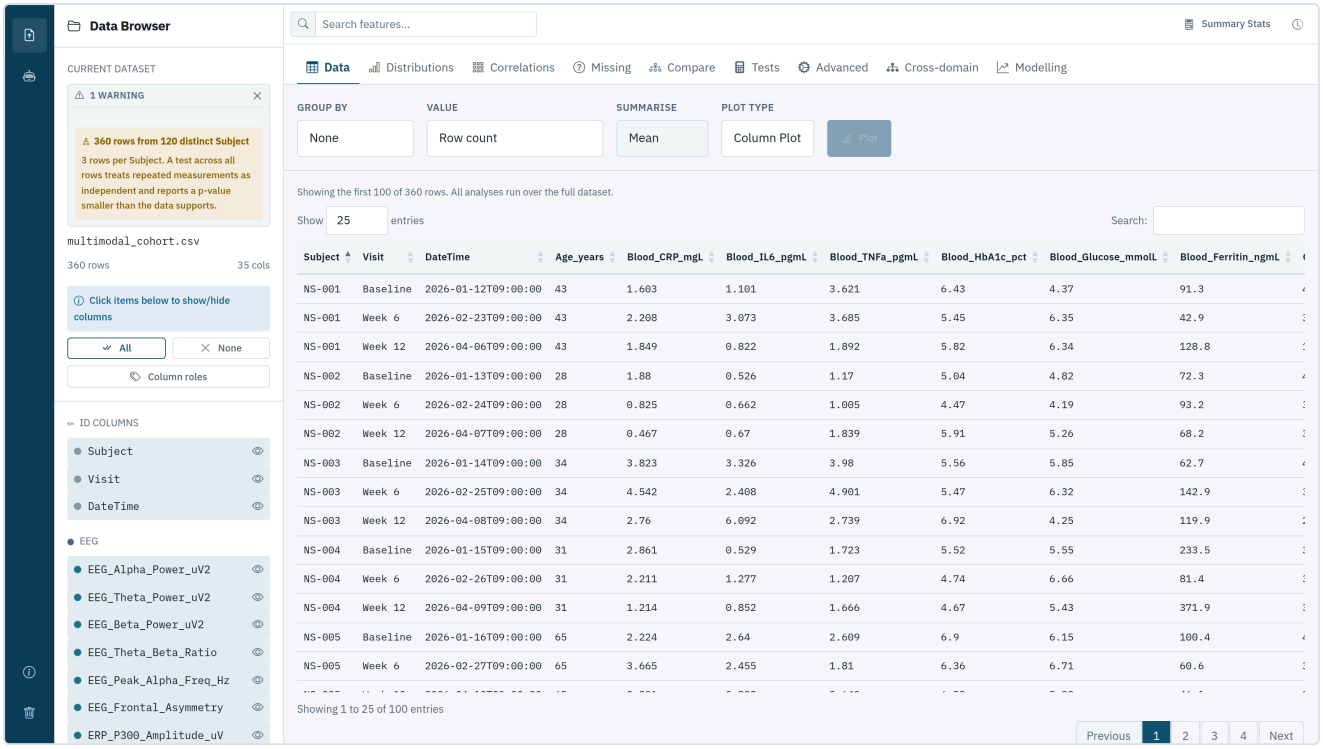

Figure 4 — Import reports the shape it read before you close the dialog.

Check this number against what you expect. A row count that is one short usually means a stray header; a column count that is one long usually means a trailing comma.

Importing an export from another workbench

The EEG and Wearable Workbenches write a `.zip` holding a table and a manifest — a `_analytics.json` beside it. Drop that zip straight in. You can also drop the two files loose; they pair by name.

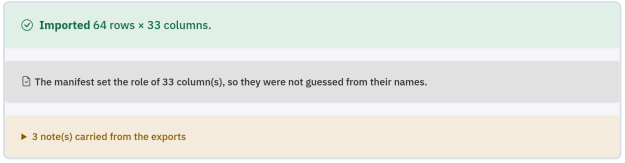

Figure 5 — A single export states its shape and where its roles came from.

The manifest is doing real work. It carries the units, the roles and the upstream preprocessing, so the columns are not guessed at from their names — and the summary says how many roles came from it. Where this matters most is a column whose name reads like something else. A per-channel EEG export has 64 distinct channel names in 64 rows, which looks exactly like an identifier to anything working from the data alone; only the manifest says `Channel` is a grouping factor. Without it that export is inert.

The upstream preprocessing goes into the analysis history with the table and the workbench that produced it, so the Methods section can say what happened before the file reached here.

Several files at once

Select or drop more than one and they are combined into a cohort. What the import does with them depends on what they are, and it says which it chose.

Figure 6 — What the import decided, and what it cost.

The tables are	What happens
The same kind of row, no two sharing a key	**Stacked** — one row after another
The same kind of row, sharing keys, from **different** workbenches	**Joined** on the key columns
The same kind of row, sharing keys, from the **same** workbench	Refused — stacking would duplicate, joining would collide
Different kinds of row, or different key columns	Refused, naming both

The figure above is a join: one subject's EEG and wearable recordings of the same visit, side by side on one row. Columns present in both are compared before merging — where they agree, one copy is kept; where they disagree, both survive under the name of the workbench that wrote them. `Session_Start` is the usual case, and it should differ: the two recordings of one visit did not start at the same moment, and keeping only one would attach the wrong clock to half the row.

> **A CROSS-DEVICE COHORT IS MOSTLY EMPTY, AND THE IMPORT SAYS SO**
>
> Two instruments measuring one subject produce almost no columns in common — in the figure, 61 of 68 come from one table only. That is not a fault in the import; it is what cross-device data looks like, and it is why the warning states how many columns every table carries. Those are the ones a test across the cohort can actually use.

When a batch is refused

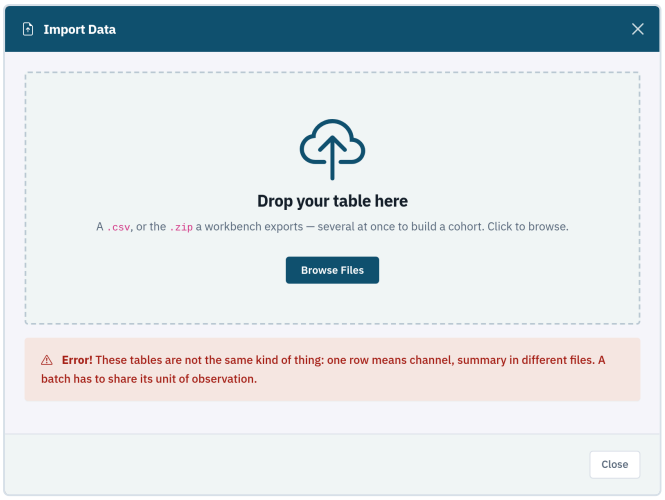

Figure 7 — A batch that cannot be combined is refused, and says why.

A refusal names both sides of the problem. Mixing granularities is the common one — a whole-recording export and a per-channel export are not the same kind of row, and stacking them would put 64 channels beside a subject as though they were comparable observations.

If the files carry no manifest at all, they can still be stacked when their column sets match exactly. When they do not, the import asks for the zip rather than guessing at an alignment.

What the profiler tells you

Import is not just a load. The workbench profiles the file and tells you what it found — and some of what it finds changes whether a number can be trusted, so it appears beside the data rather than in a log.

The Data Browser

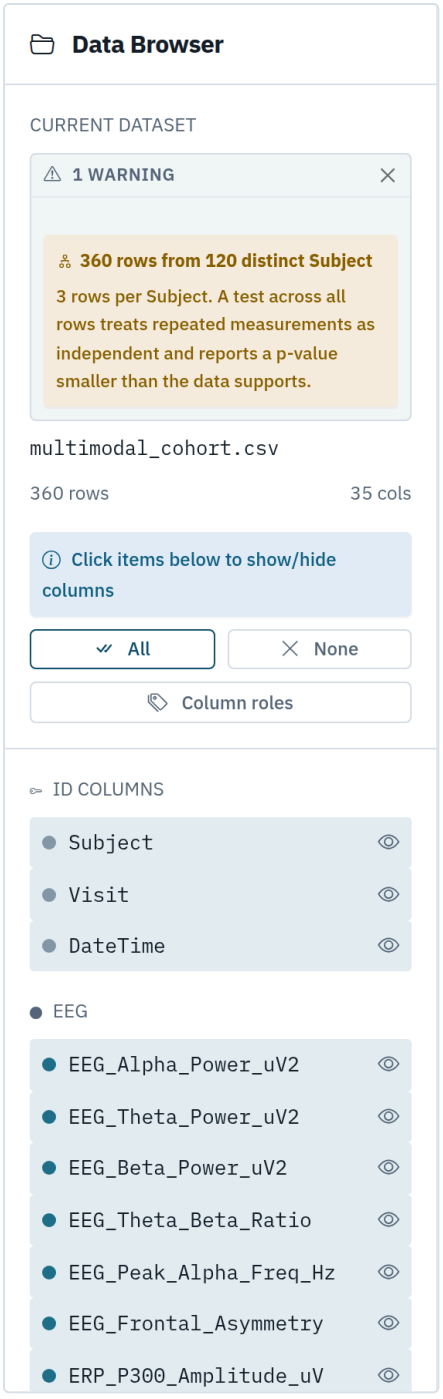

Figure 8 — The Data Browser after import: warnings, shape, then columns by domain.

Columns are grouped by the domain the profiler inferred from their names — ID, EEG, Cardiac, Blood, and so on. Click any column to hide it from the analyses; click again to bring it back. **All** and **None** act on the whole list.

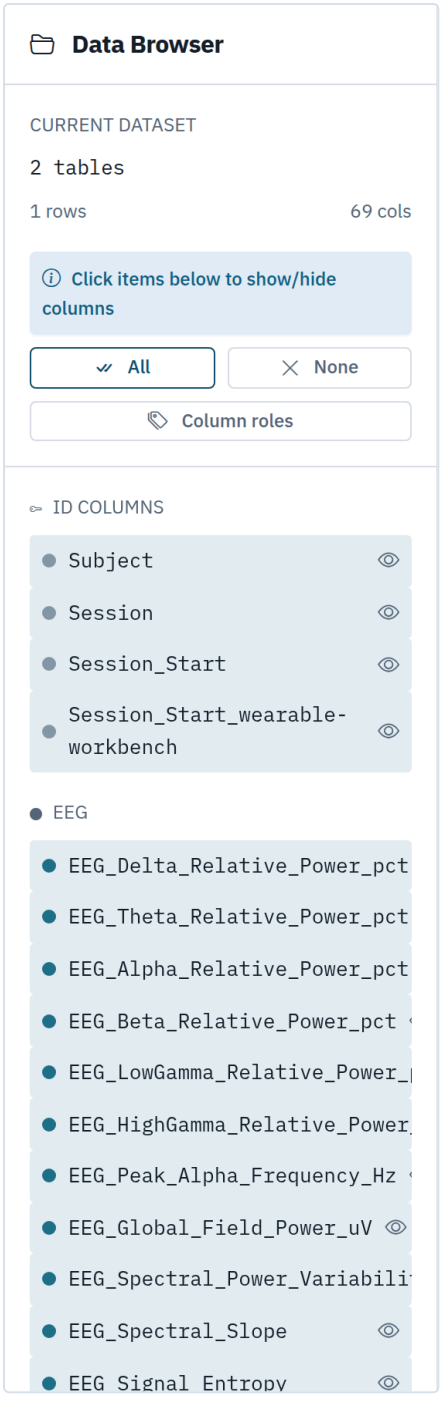

Figure 9 — Both instruments in one column list, separated by modality.

After a cross-workbench join the list is the clearest view of what you actually have: which modalities are present, and how many columns each contributes.

Warnings you should read

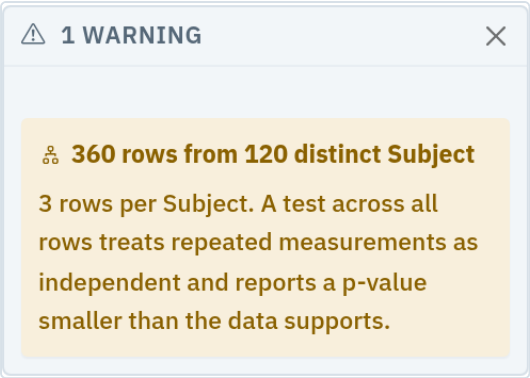

Figure 10 — The repeated-measures warning, raised at import.

This is the most consequential thing the workbench will tell you, so it is worth reading slowly. The file has 360 rows, but they come from 120 subjects measured three times each. A test run across all 360 rows treats those three measurements as three independent people. It will report a p-value smaller than the data supports.

The warning is not a blocker. It is telling you that later on you will have to choose a policy, and Chapter 10 is where you choose it.

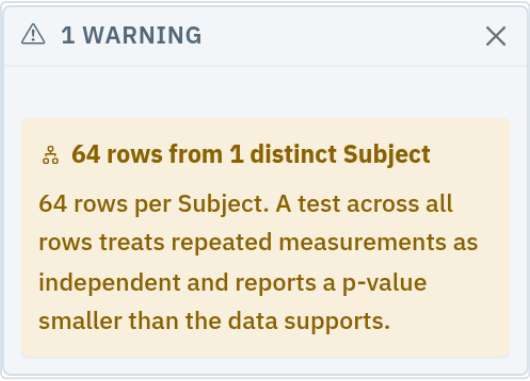

Figure 11 — Sixty-four rows, one subject: not sixty-four observations.

The same warning is raised when a file holds **one** subject measured many times, which is what every export from the EEG and Wearable Workbenches is: 64 channels, or 96 windows, or a row per day for five months, all from one person. The wording differs because the remedy does — there is no timepoint to filter to and nothing to average across. What the file needs is the rest of the cohort, imported alongside it.

> **DISMISSING IS NOT FIXING**
>
> The close button on the warnings panel hides it so the column list has room. It does not change the data or the analyses. The row policy in the Tests tab is what actually addresses this.

Column roles

Every column is assigned a role: identifier, time, group, covariate or measurement. Those roles decide which pickers a column appears in — you cannot group by a measurement, and you cannot test an identifier.

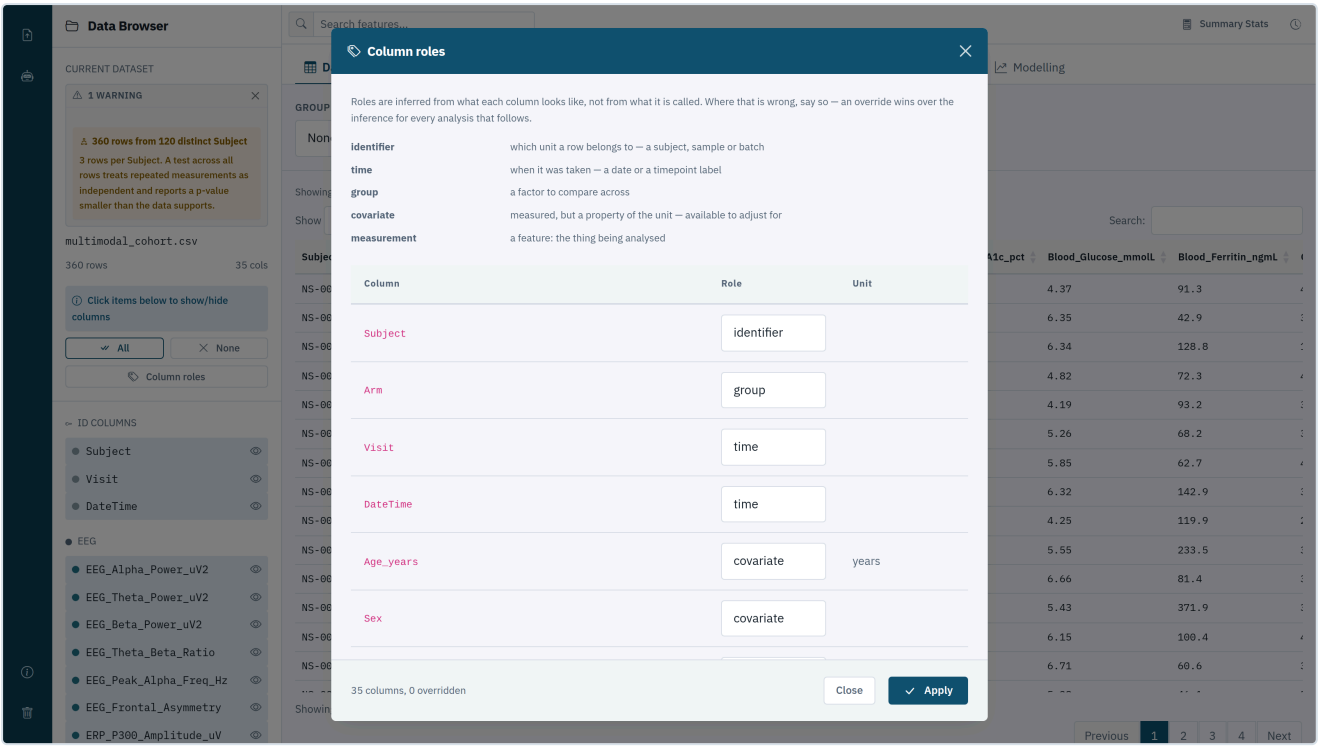

Figure 12 — Every inferred role is visible, and every one can be overridden.

The inference is good but not clairvoyant. An ordinal 1–10 score is genuinely both a measurement and a factor, and the workbench has to pick one. If it picked wrong, set it here.

A file that came with a manifest has already answered this. Roles declared by the workbench that wrote the table are applied on import rather than inferred, and the import summary says how many. You can still override them — a declared role is a starting point, not a lock.

Summary statistics

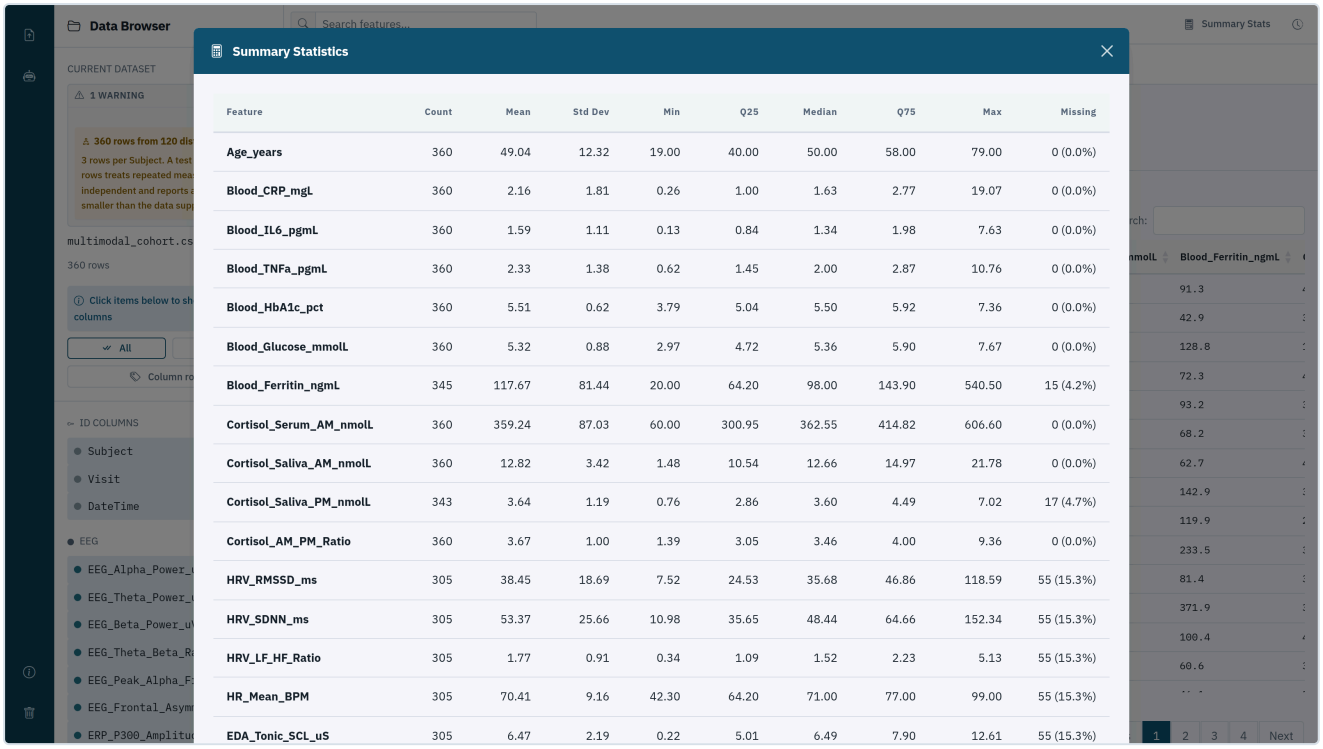

Figure 13 — Summary statistics across every column at once.

Worth a look before anything else: a mean far from the median tells you a measurement is skewed, and a minimum of exactly zero on something that cannot be zero tells you a sentinel value survived import.

Looking at your data

The Data tab

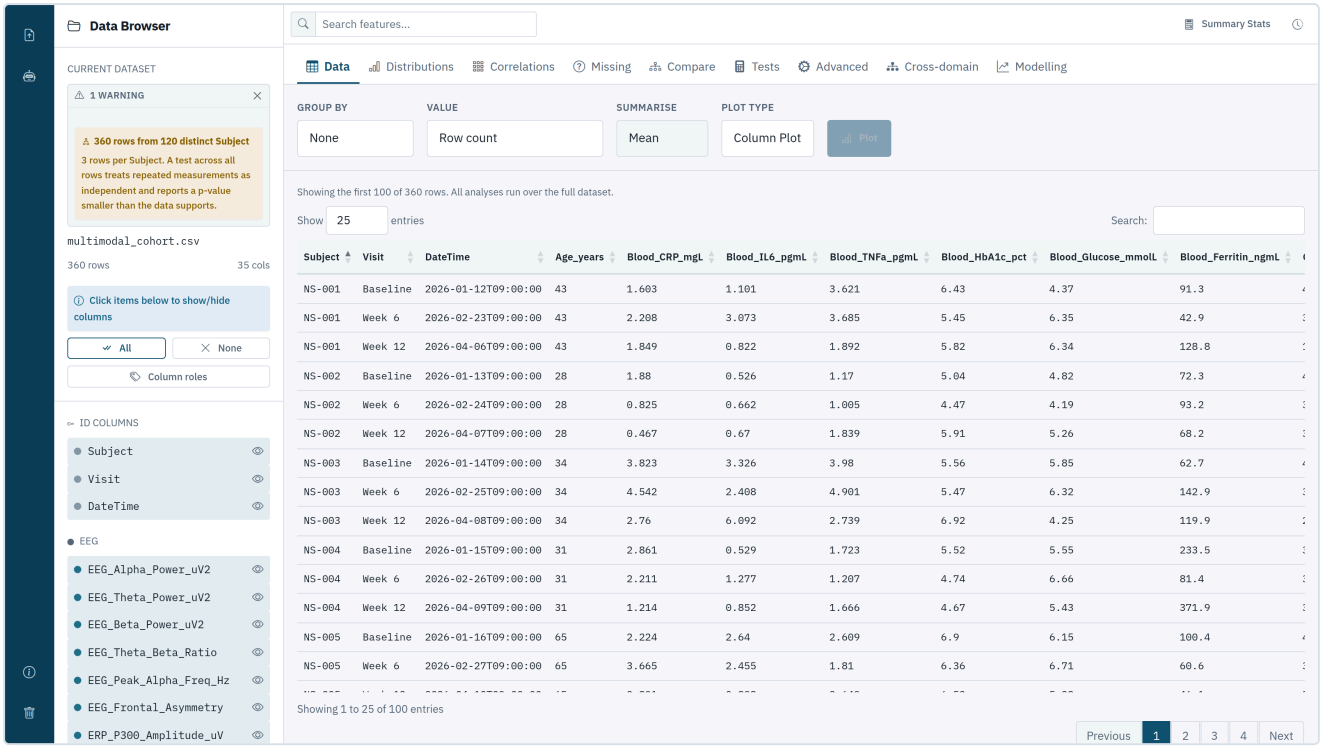

Figure 14 — The Data tab: the table as read, with the controls for a quick plot above it.

The table shows what was actually imported, paginated and searchable. Hiding a column in the Data Browser removes it here too.

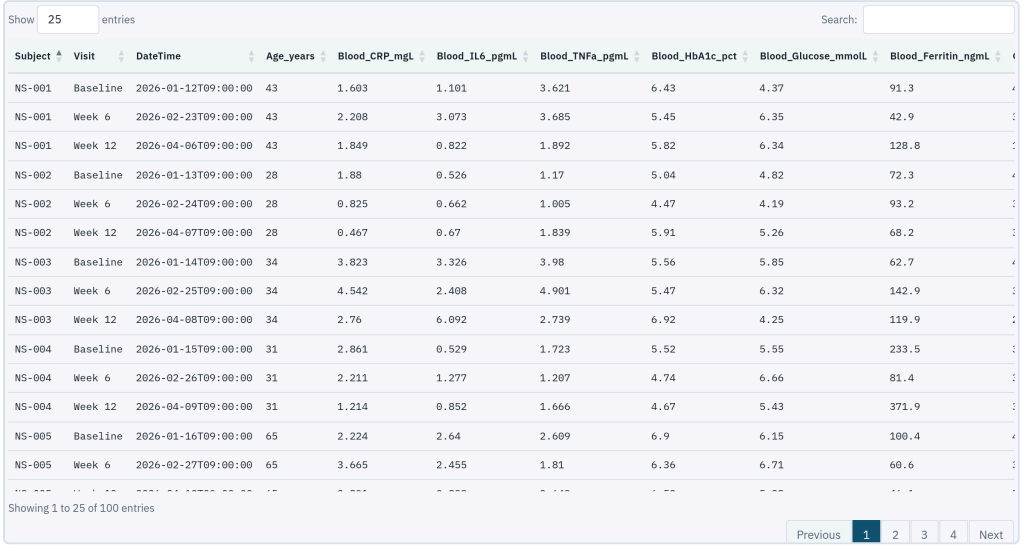

Figure 15 — The preview table. Sort by any column header.

Plotting straight off the preview

The row of controls above the table builds a chart from any two columns without leaving the tab. Pick a **Group By**, a **Value**, how to **Summarise** it, and a plot type.

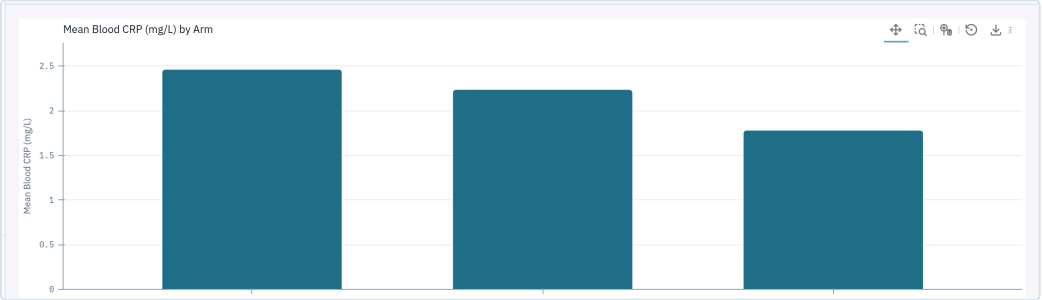

Figure 16 — Mean CRP by arm, as a column plot.

Five plot types share these controls — column, bar, line, histogram and box. Switching type re-draws the same pairing, so you can find the right view quickly.

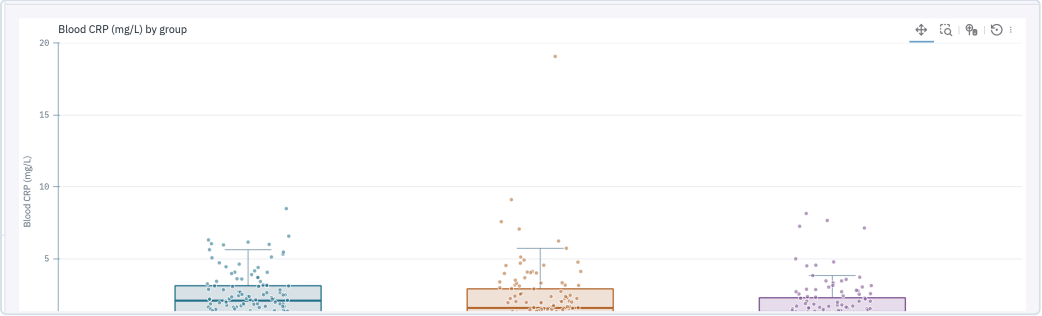

Figure 17 — The same pairing as a box plot, which shows the spread the column plot hid.

Clear **Group By** and the distribution plots take over: a histogram or box plot of the value column on its own.

> **THIS IS THE FASTEST WAY TO SANITY-CHECK AN IMPORT**
>
> Plot the measurement you know best against the factor you know best. If it looks wrong here, the problem is in the file, not in the analysis.

Distributions

The Distributions tab asks one question properly: what shape is this measurement?

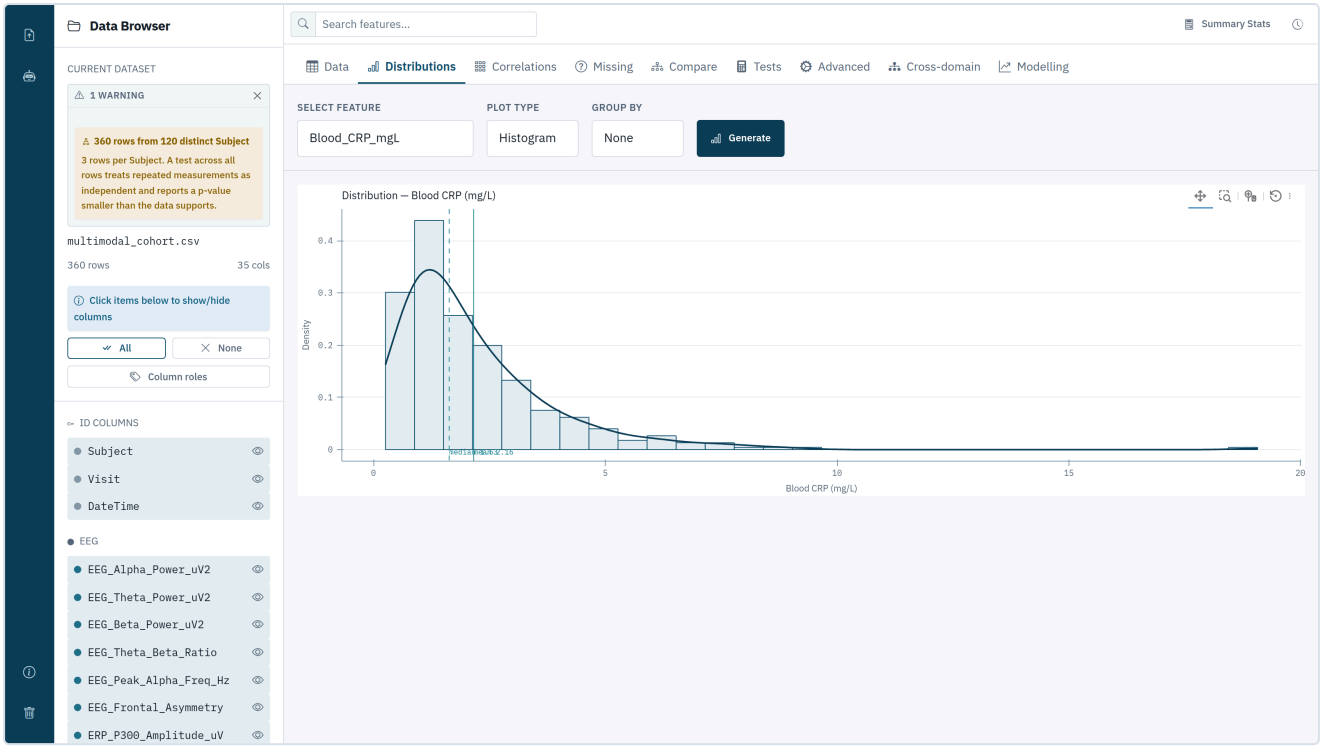

Figure 18 — CRP as a histogram — the long right tail of a log-normal marker.

Four plot types are available, and each answers something different:

Histogram

The shape itself. Look for skew, bimodality and a pile-up at a boundary.

Box plot

The quartiles and the outliers, compactly. Best when comparing groups.

Violin plot

The shape *and* the quartiles. Use it when a box plot hides something — two humps look identical to a box plot and obvious in a violin.

Q-Q plot

Whether the measurement is normal. Points on the line mean normal; a curve means skewed; an S-shape means heavy tails.

Set **Group By** and the plot splits.

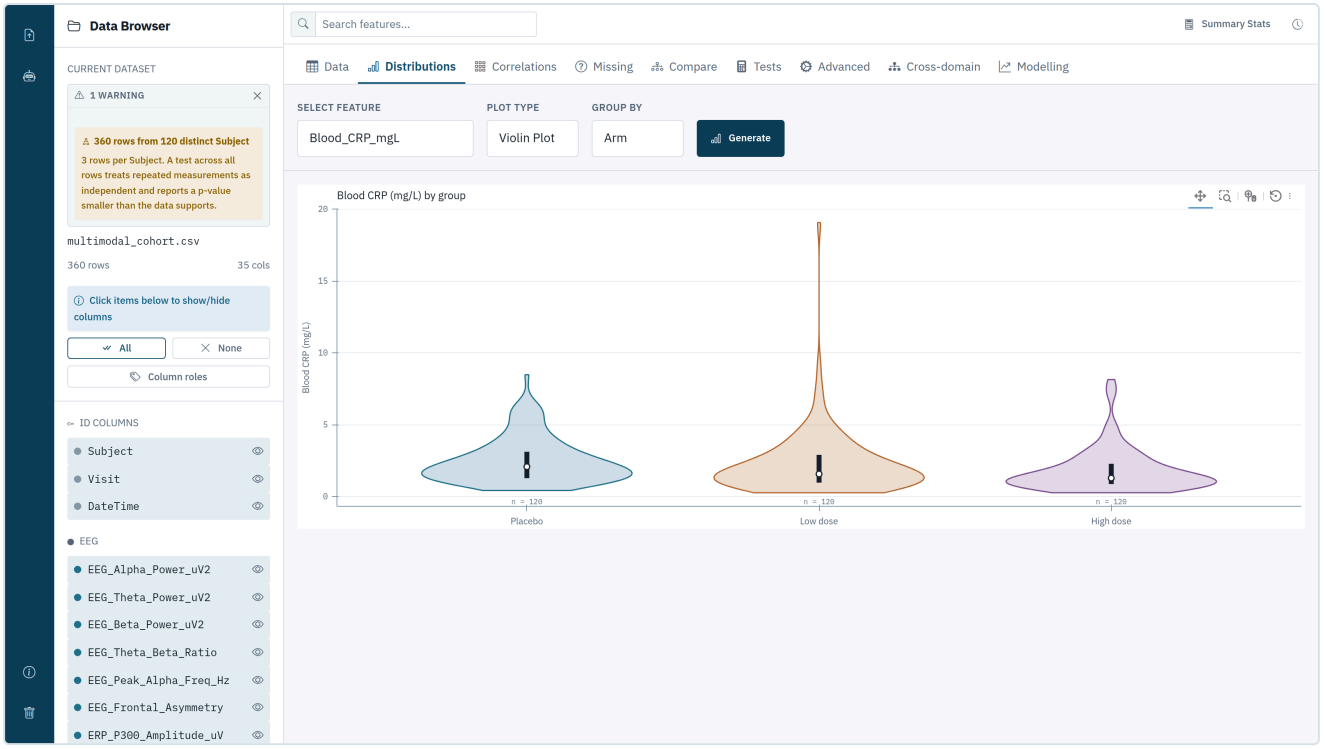

Figure 19 — The same marker split by arm.

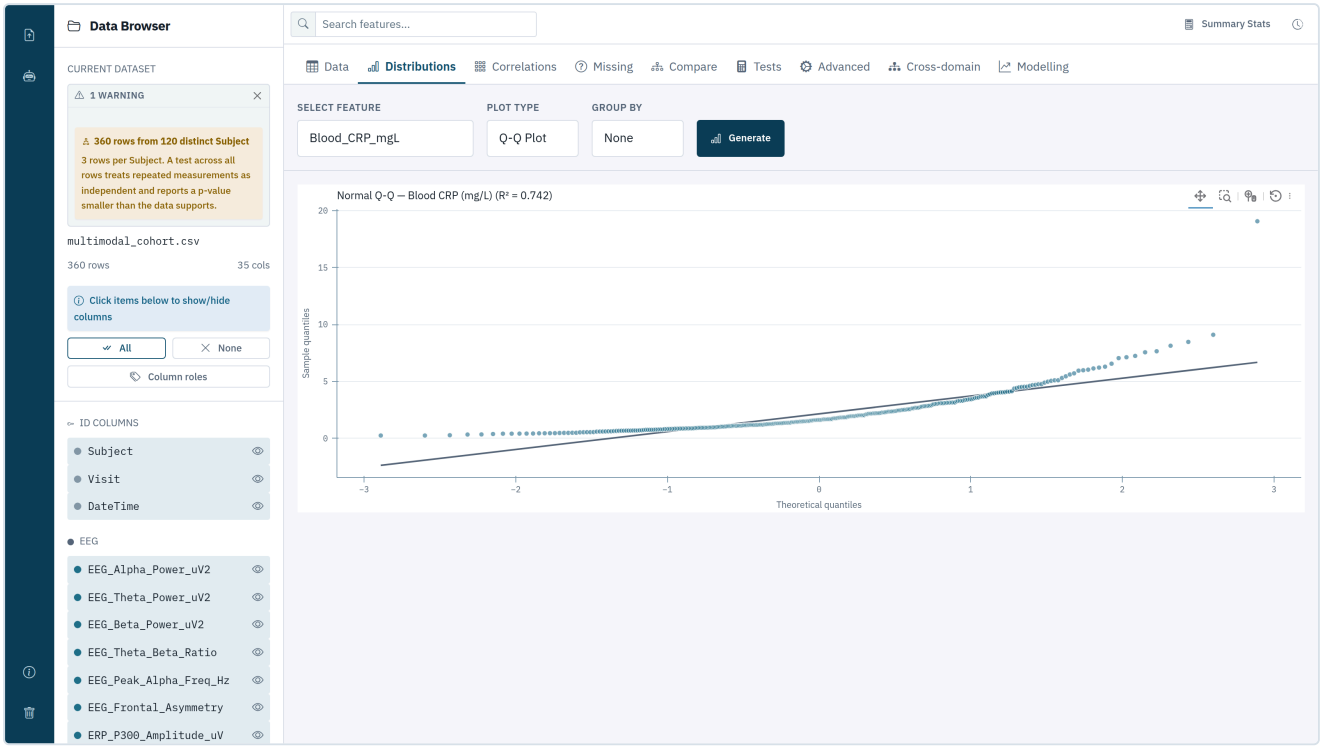

Figure 20 — A Q-Q plot answers the normality question directly, rather than by eye on a histogram.

This matters for the next chapter but one: if the Q-Q plot curves away from the line, prefer the rank-based tests — Mann-Whitney and Kruskal-Wallis — over the t-test and ANOVA.

Correlations

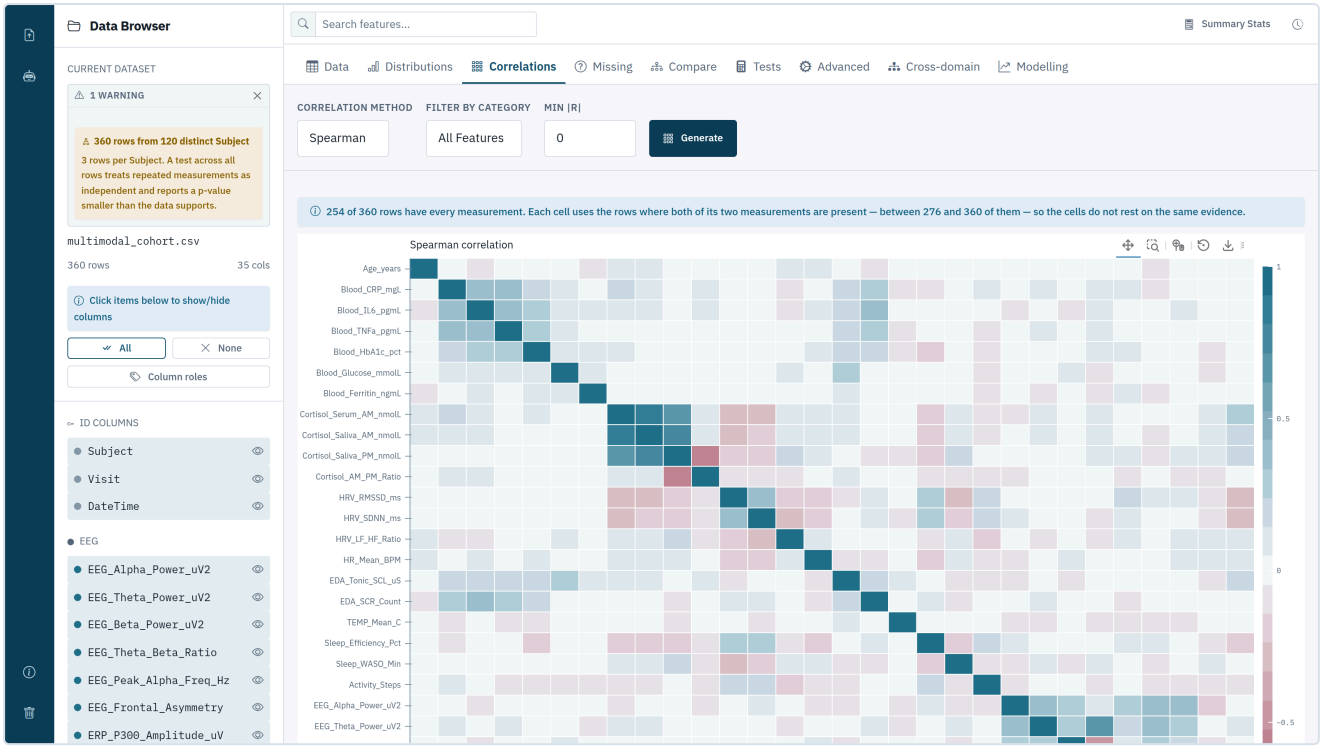

Figure 21 — Spearman correlation across every measurement in the file.

Three methods are offered. **Pearson** measures straight-line association and assumes roughly normal data. **Spearman** works on ranks, so it survives skew and outliers — the safer default for biological measurements. **Kendall** is also rank-based and more conservative on small samples.

With 30 measurements the matrix is 435 pairs, which is too many to read. Two controls cut it down: **Filter By Category** restricts to one domain, and **Min |r|** hides everything weaker than a threshold.

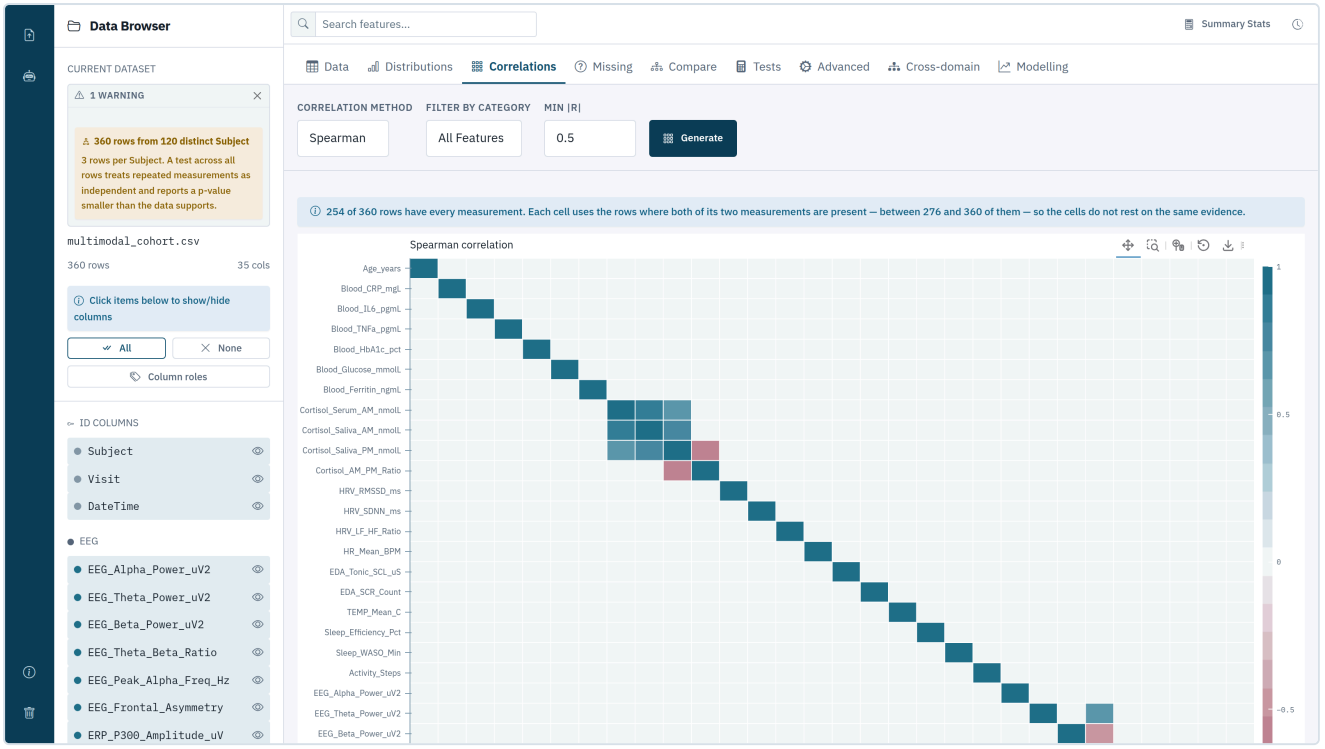

Figure 22 — The same matrix filtered to |r| ≥ 0.5.

A CORRELATION MATRIX IS A HYPOTHESIS GENERATOR, NOT A RESULT

435 pairs at p < 0.05 will produce about 22 significant correlations by chance alone. Use this to decide what to test, then test it properly with correction — see Chapter 11.

Missing data

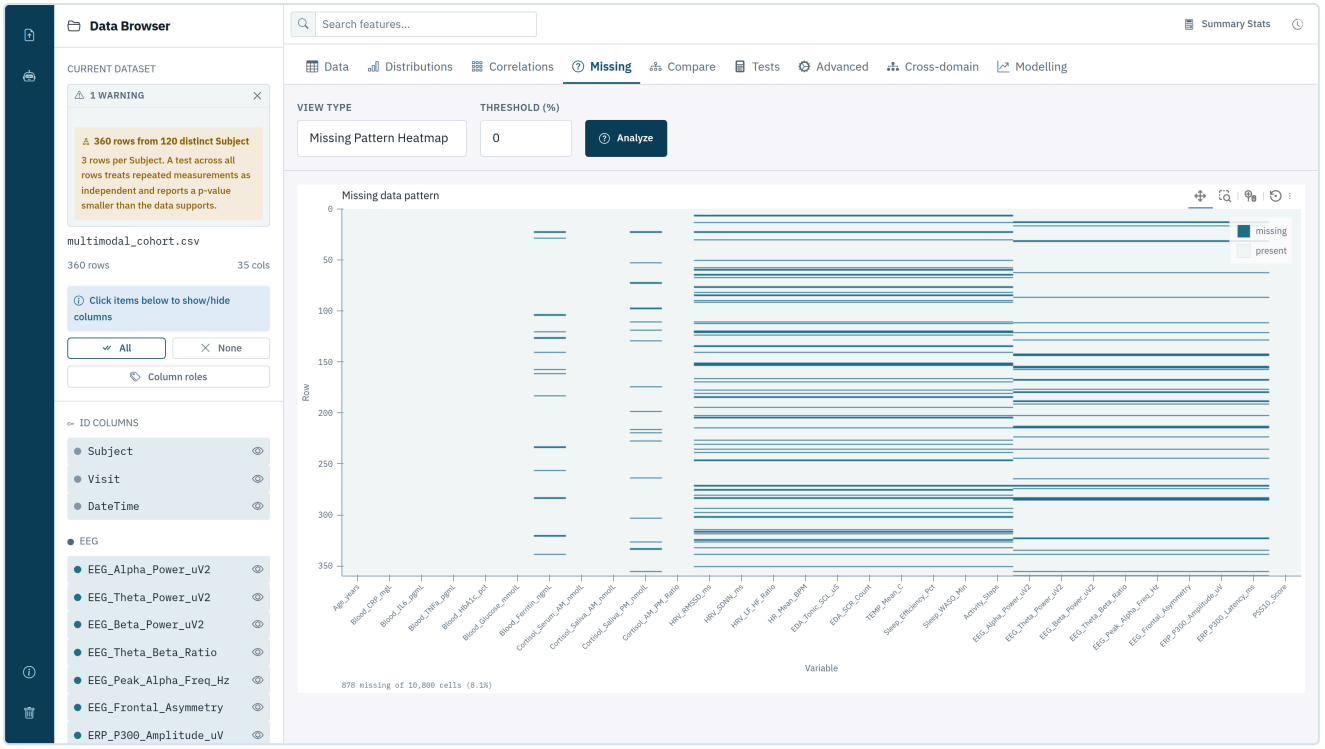

Figure 23 — The missing-data heatmap. Each row is an observation, each column a measurement.

The heatmap is the view worth learning to read, because it shows *pattern*, not just quantity. In this cohort the absences come in vertical blocks: when a wearable was not worn, every wearable column for that visit is blank at once, and the same for a missed EEG session.

That is structured missingness, and it matters. An analysis that assumes each column goes missing independently is wrong about this file — and any imputation that fills columns one at a time will invent correlations that were never measured.

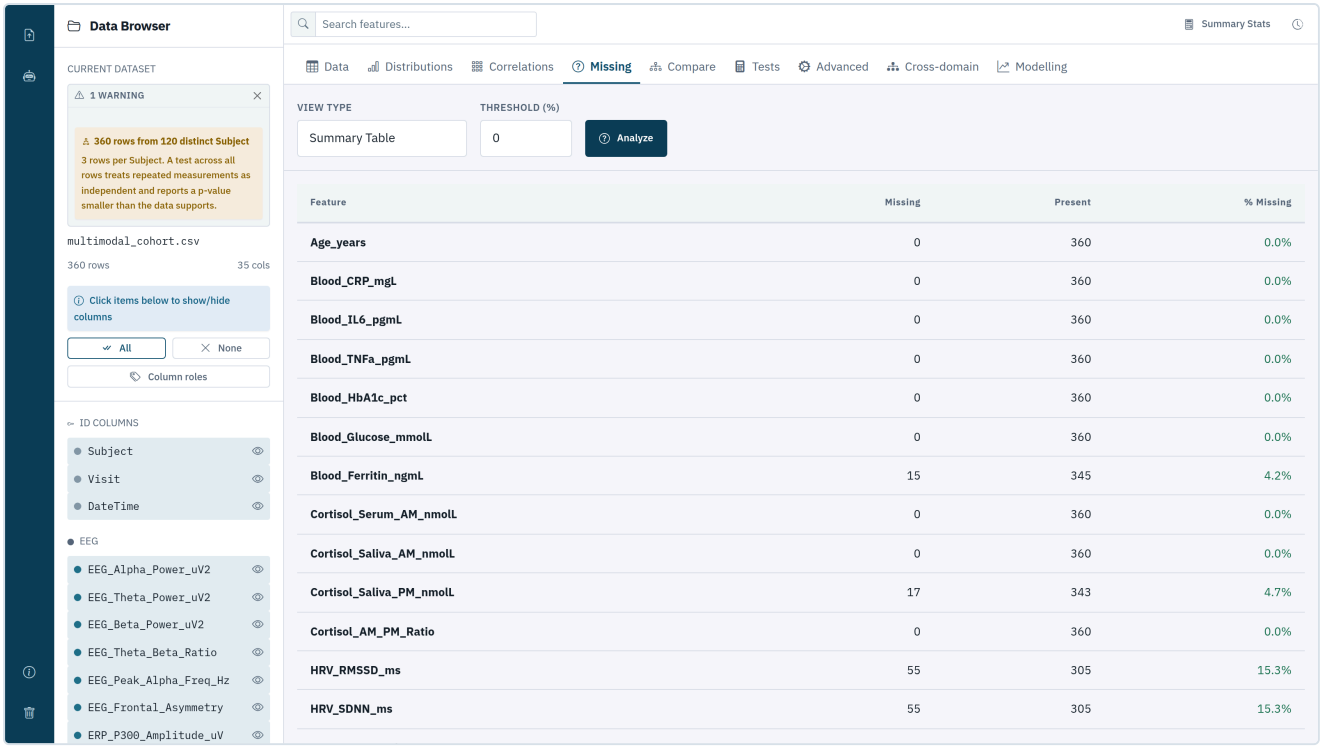

Figure 24 — The same absences as a summary table, with counts and percentages.

Switch **View Type** to **Summary Table** for the numbers, or **Missing Counts by Feature** for a ranked bar chart. The **Threshold (%)** box hides columns below a given level of missingness.

Comparing and testing

Comparing groups

The Compare tab draws the difference between groups with the uncertainty attached.

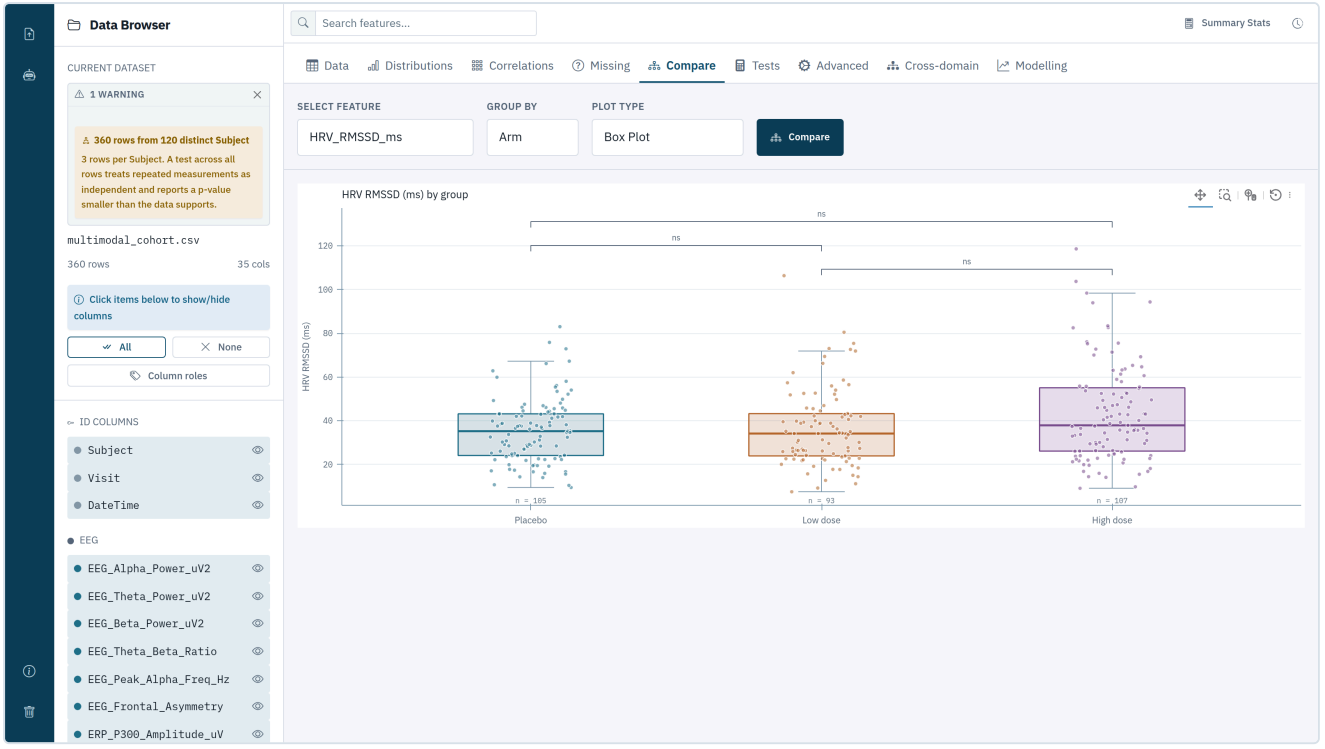

Figure 25 — HRV RMSSD across three arms, with pairwise significance brackets.

The brackets above the boxes mark pairwise comparisons that reached significance. They are a visual aid, not the test — run the test in the next chapter and report that.

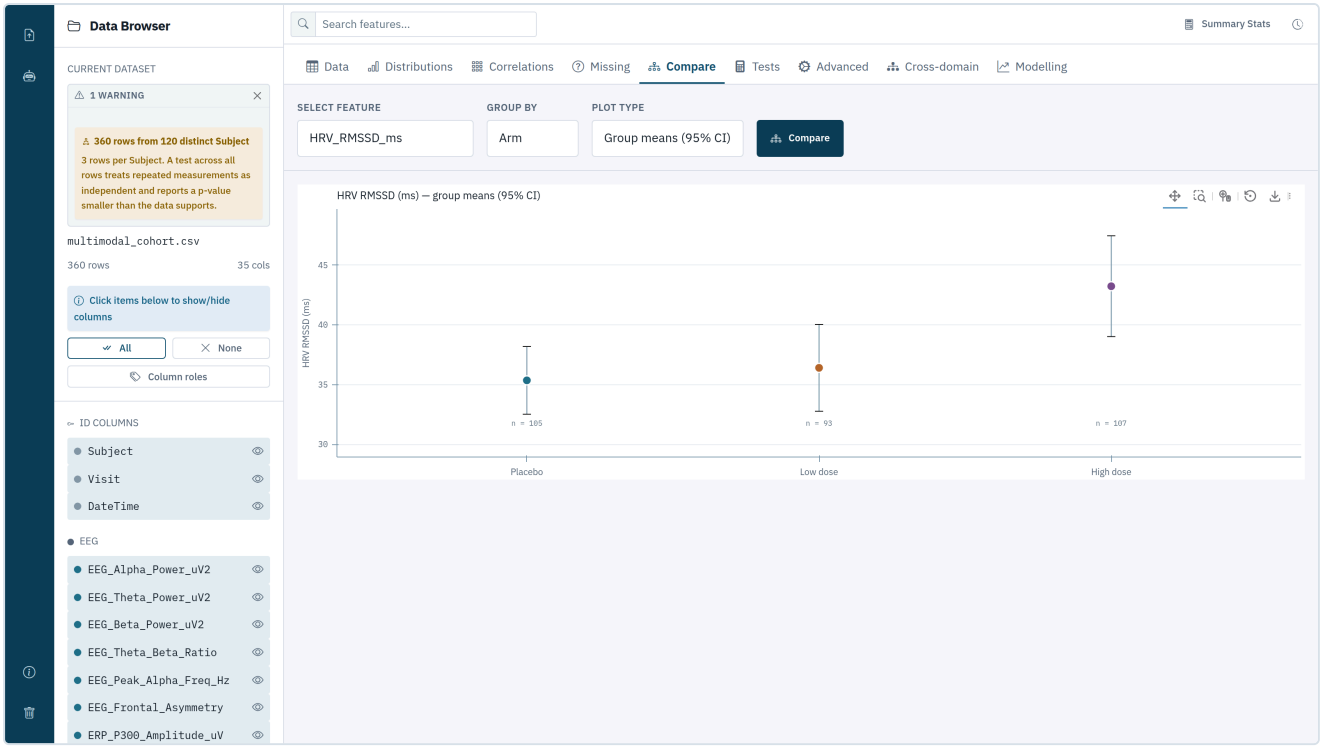

Figure 26 — The same comparison as group means with 95% confidence intervals.

The means-with-intervals view is usually the more honest figure for a paper. It shows the estimate and its precision, rather than inviting the reader to eyeball overlapping boxes.

Running a statistical test

Pick a measurement, a grouping column and a test.

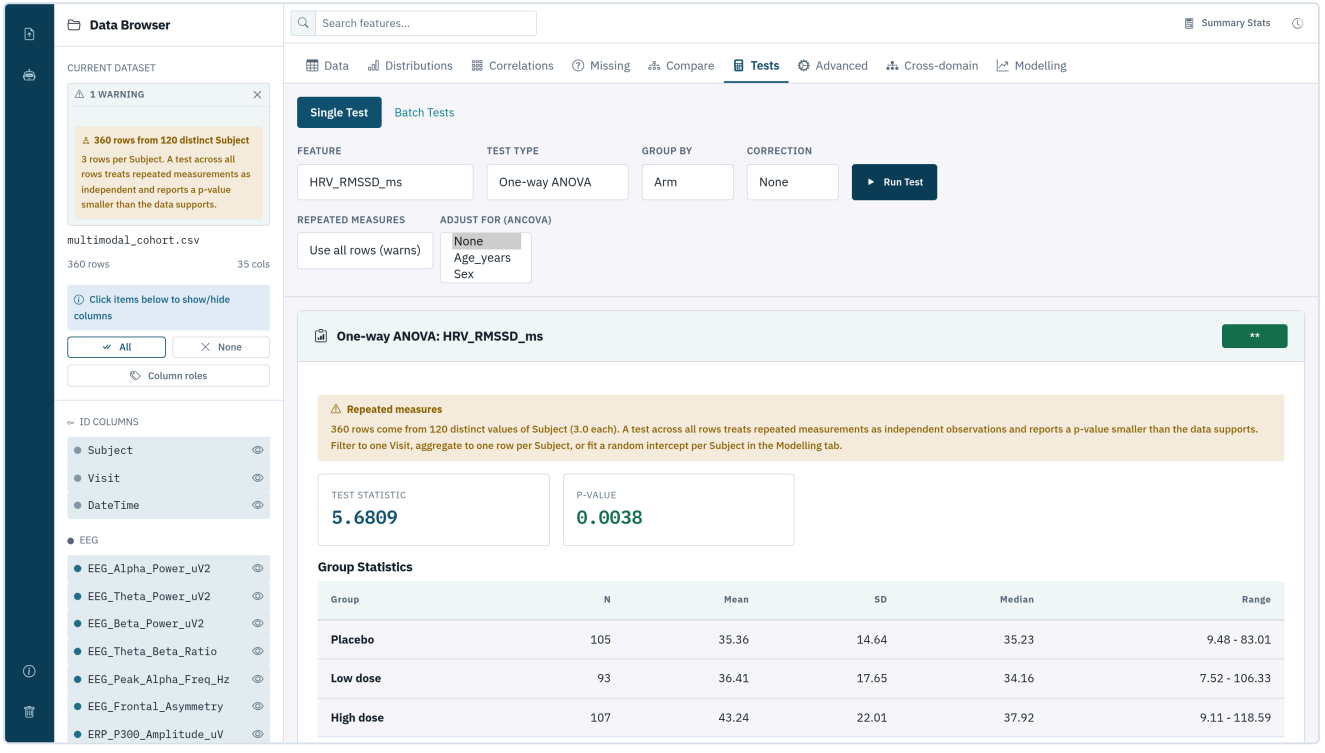

Figure 27 — A one-way ANOVA across three arms.

Six tests are available, in matched parametric and rank-based pairs:

Comparing	Parametric	Rank-based
Two independent groups	Independent t-test	Mann-Whitney U
Two paired measurements	Paired t-test	Wilcoxon
Three or more groups	One-way ANOVA	Kruskal-Wallis

The result reports the test statistic, the p-value, and per-group statistics — n, mean, SD, median and range — so you can see the groups the p-value came from.

The row policy

Because this file has repeated measures, the workbench will not let the point pass silently.

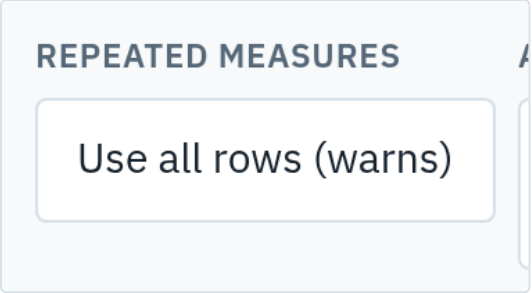

Figure 28 — The row policy appears whenever repeated measures were detected.

You have three choices, and the guide recommends being deliberate about which:

Use all rows (warns). Runs the test on all 360 rows and keeps the warning attached to the result. Defensible only as a first look.

Average per unit. Collapses to one row per subject before testing. The sample size becomes 120, which is the honest n. This is the simplest correct answer.

One timepoint only. Filters to a single visit. Correct, and the right choice when the question is genuinely about that visit.

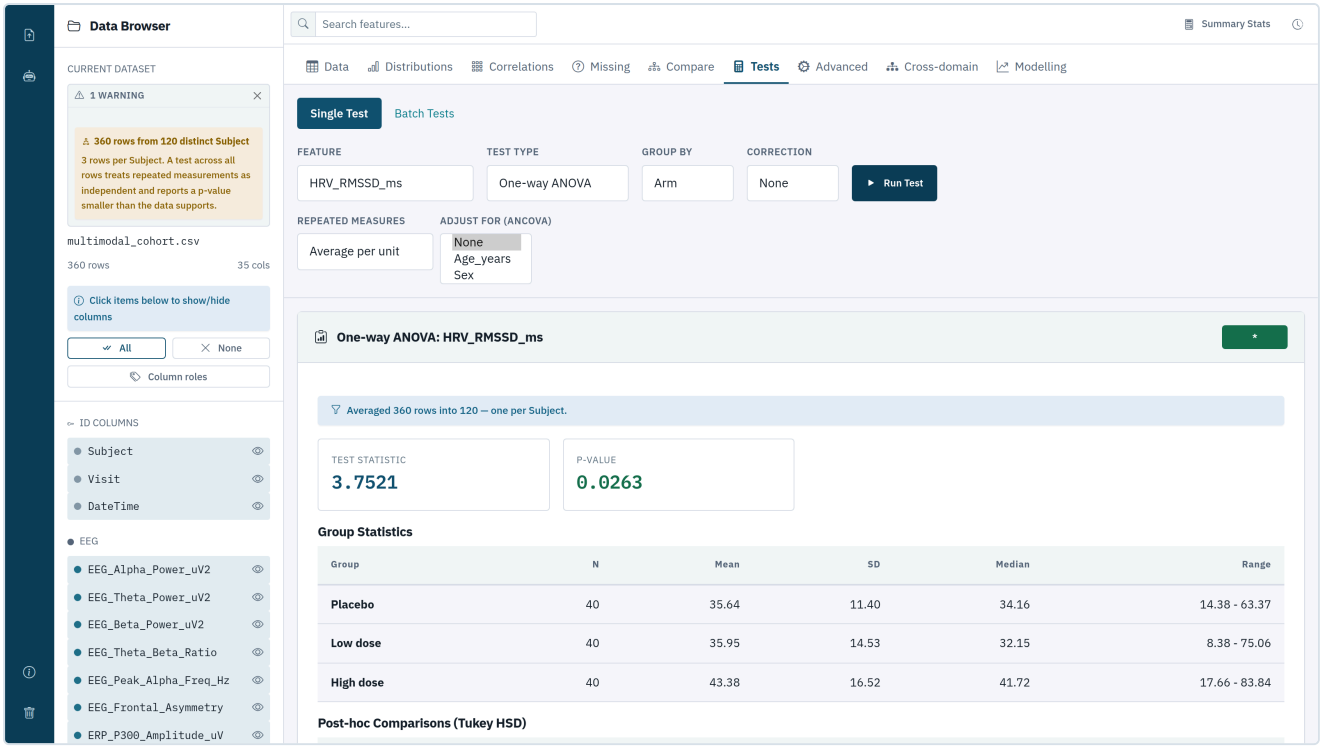

Figure 29 — The same test after aggregating to one row per subject.

Compare the two results. The aggregated test is the one you can report.

> **THE FOURTH OPTION IS A MODEL**
>
> The warning also mentions fitting a random intercept per subject. That keeps every row and still counts the evidence correctly — see Chapter 16.

Screening every measurement at once

Batch mode runs the same test across every measurement in the file, with multiple-testing correction.

Before the working example, a useful failure. Here the test type is an independent t-test and the grouping column has three arms:

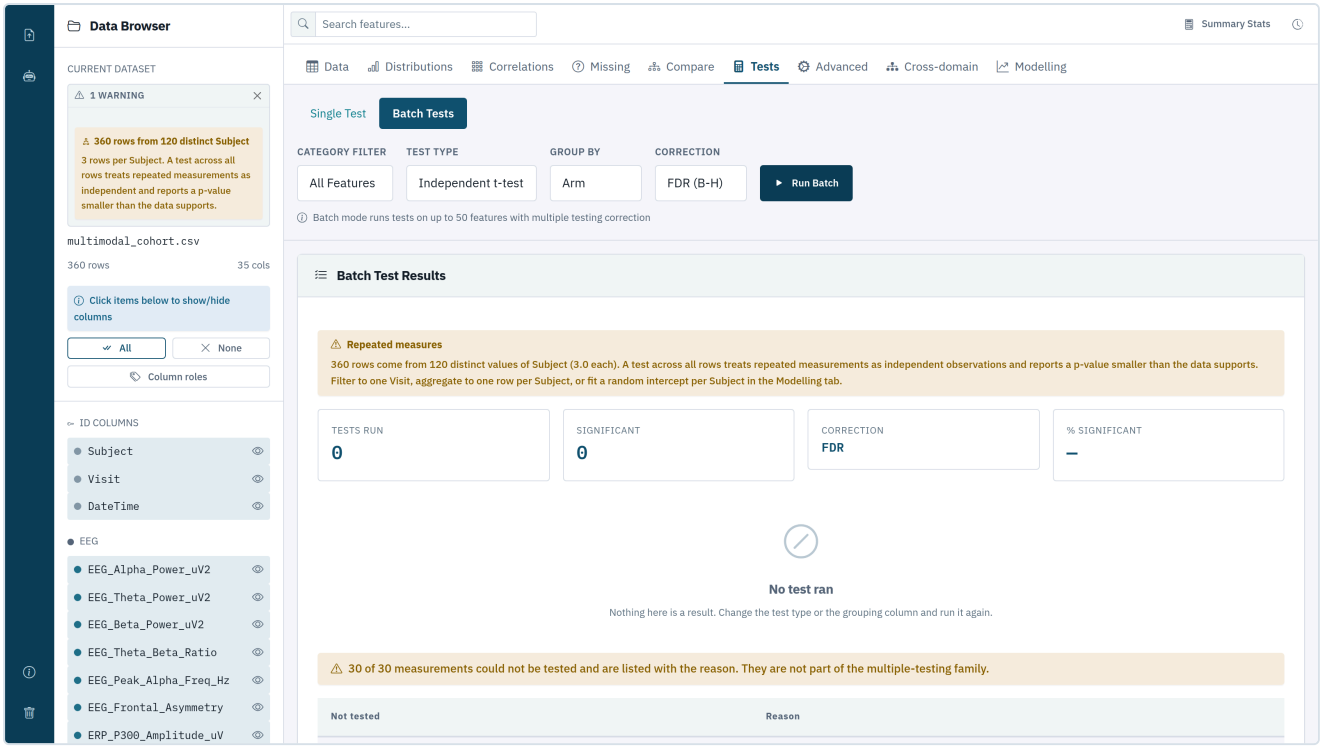

Figure 30 — A two-sample test against a three-level factor. Nothing runs, and every refusal is listed with its reason.

Nothing ran, and the workbench says so plainly — 30 of 30 measurements could not be tested, each with a reason, and they are explicitly excluded from the multiple-testing family. This is the behaviour to expect throughout: a refusal with an explanation, never a number that looks fine and is not.

Switch to a test that accepts three groups and it runs:

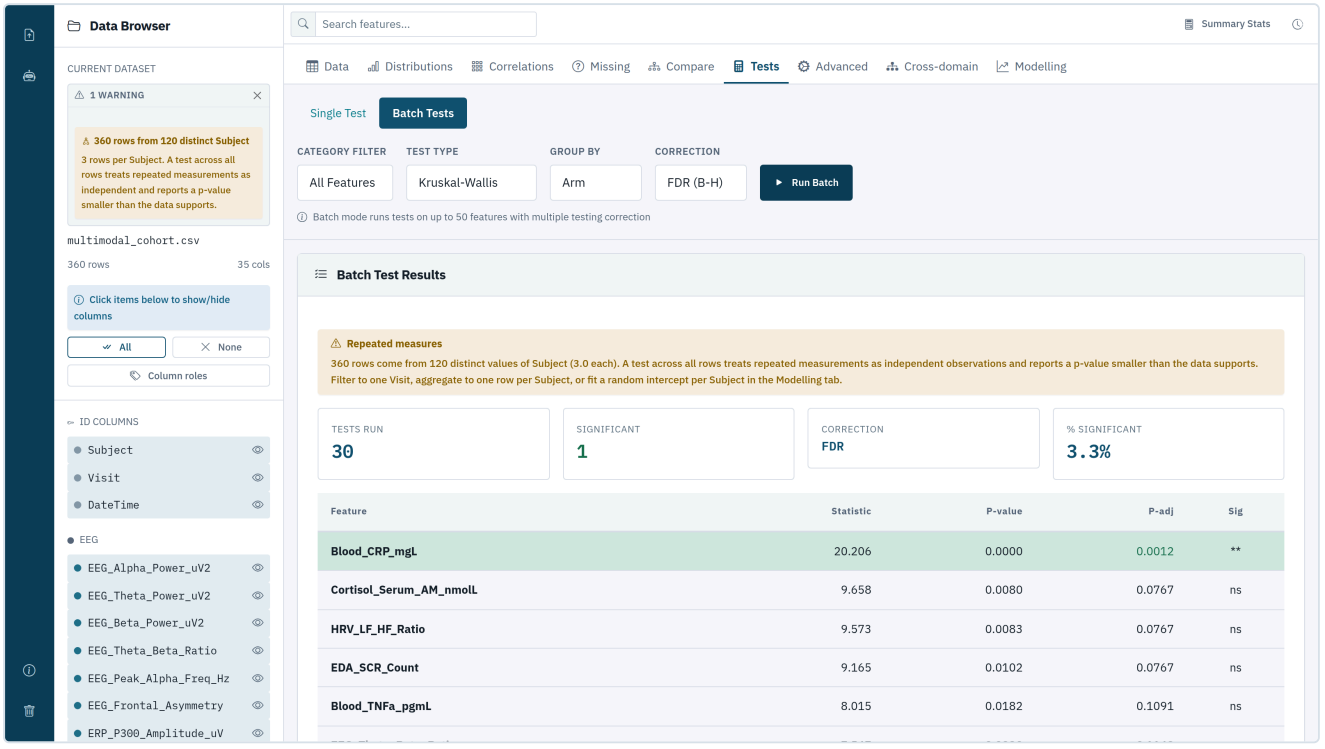

Figure 31 — Every measurement screened with Kruskal-Wallis and FDR correction.

Four corrections are offered. **None** is for when you have exactly one pre-registered hypothesis. **Bonferroni** and **Holm** control the chance of any false positive at all — strict, and appropriate for confirmatory work. **FDR (Benjamini-Hochberg)** controls the expected *proportion* of false positives among your hits, which is the usual choice for discovery.

> **BATCH MODE IS A SCREEN, NOT A STUDY**
>
> A biomarker that survives FDR here is a candidate worth a designed study. It is not a finding. The repeated-measures warning applies to every row of the batch table as well.

Structure in the data

Principal components

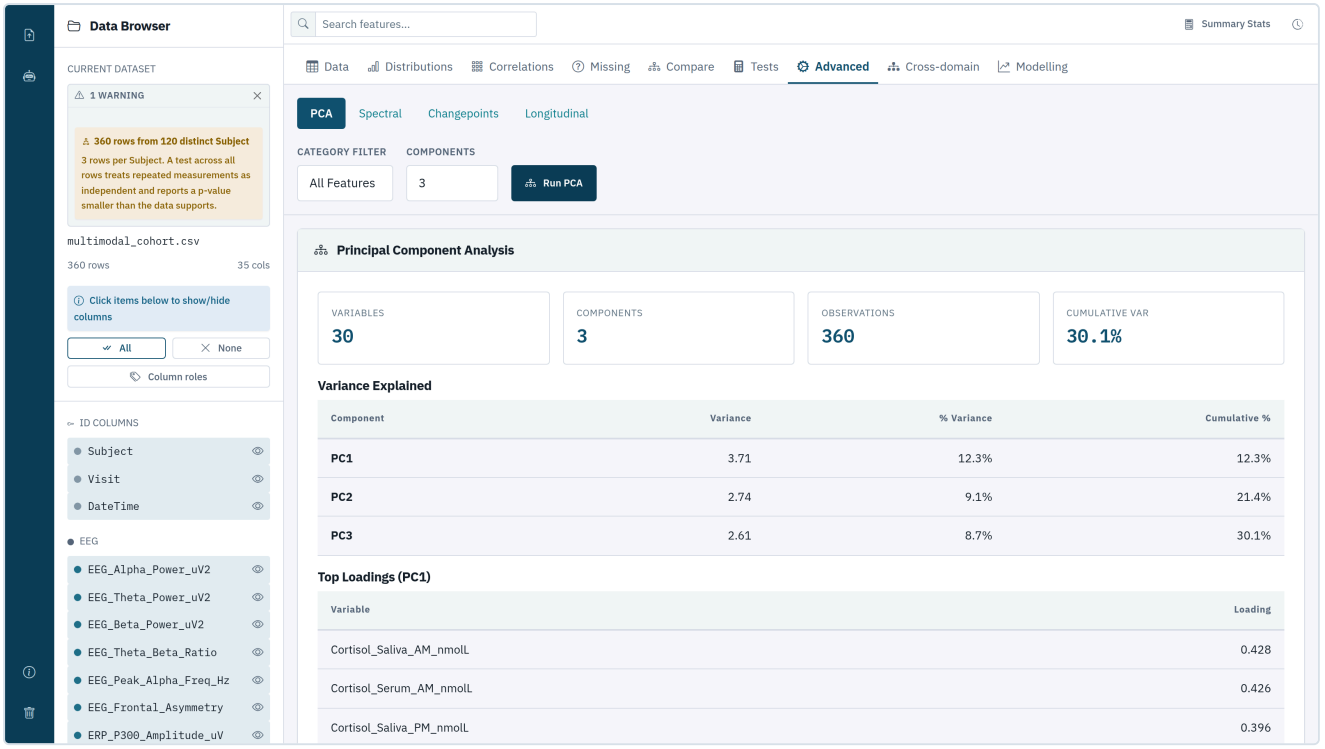

Figure 32 — Principal components over the measurement panel.

PCA answers "how many distinct things am I really measuring?" A panel of 30 measurements with three underlying drivers will put most of its variance in the first three components.

Read the explained-variance figures before the scatter. If PC1 and PC2 together account for a small fraction, the two-dimensional picture is not the data.

Change over time

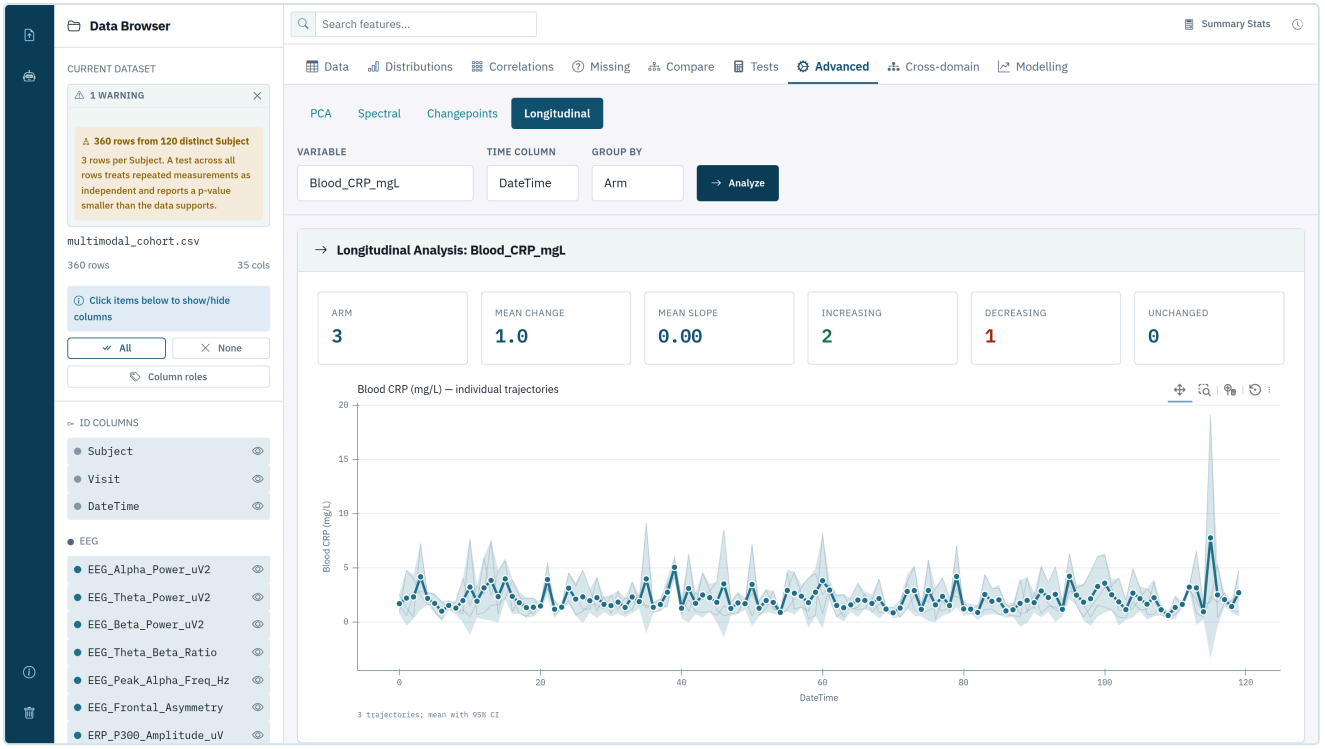

Figure 33 — Individual trajectories with the group means over them.

The longitudinal view needs a variable, a time column and optionally a grouping column. It reports how many subjects increased, decreased and stayed flat, alongside the mean change and slope.

> **CHOOSE THE TIME COLUMN CAREFULLY**
>
> A label column like `Visit` holds text, so the workbench can only take the order the rows arrive in. A real date or a numeric time gives it something to sort on. If your visit labels are not in file order, use the date column.

Changepoints and periodicity

These two need a genuine sequence, so the figures here use the daily rentals series rather than the cohort.

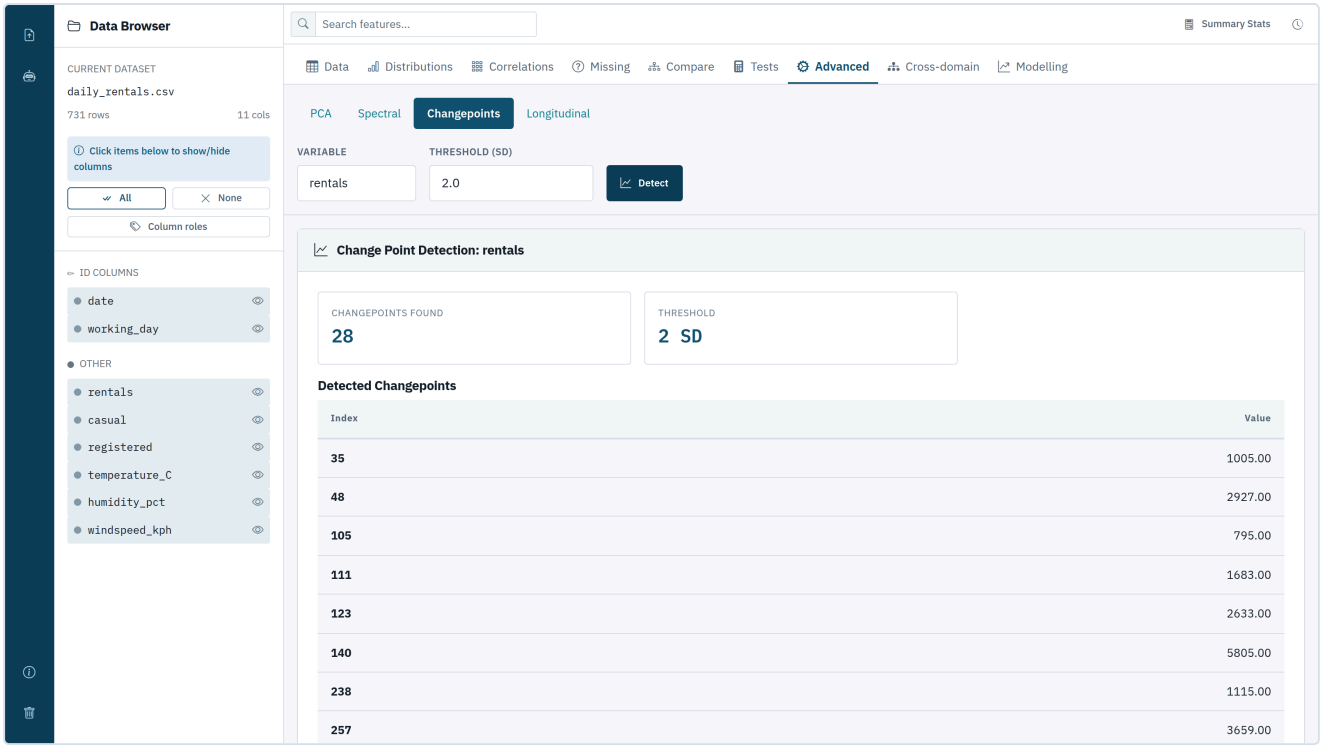

Figure 34 — Where the level of a series shifts.

Changepoint detection finds where a series changes level. The **Threshold** control sets how large a shift has to be before it counts — raise it if you are getting changepoints everywhere.

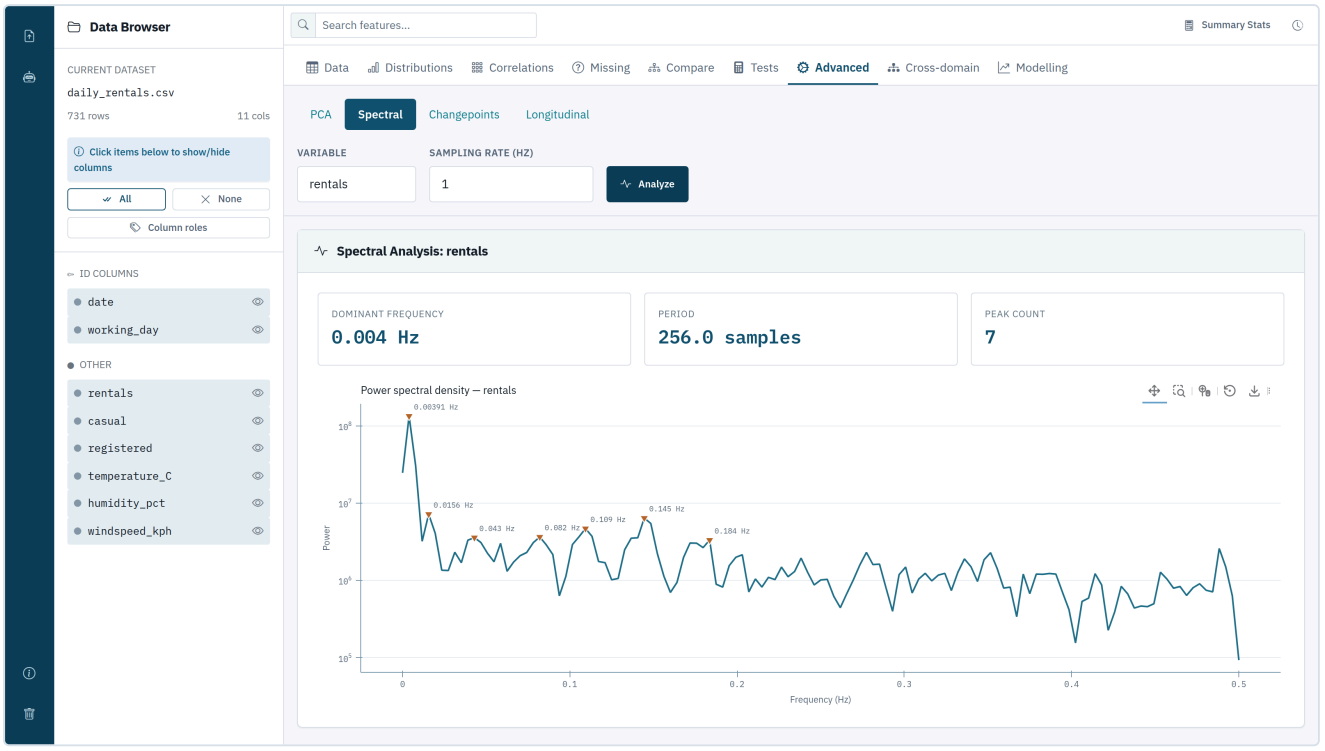

Figure 35 — The periodicity hiding in a daily count.

Spectral analysis finds repeating structure. Set the **Sampling Rate** to the number of observations per unit time — for daily data that is 1, and a peak at a period of 7 is the working week.

Cross-domain

The Cross-domain tab is for data that spans more than one kind of measurement.

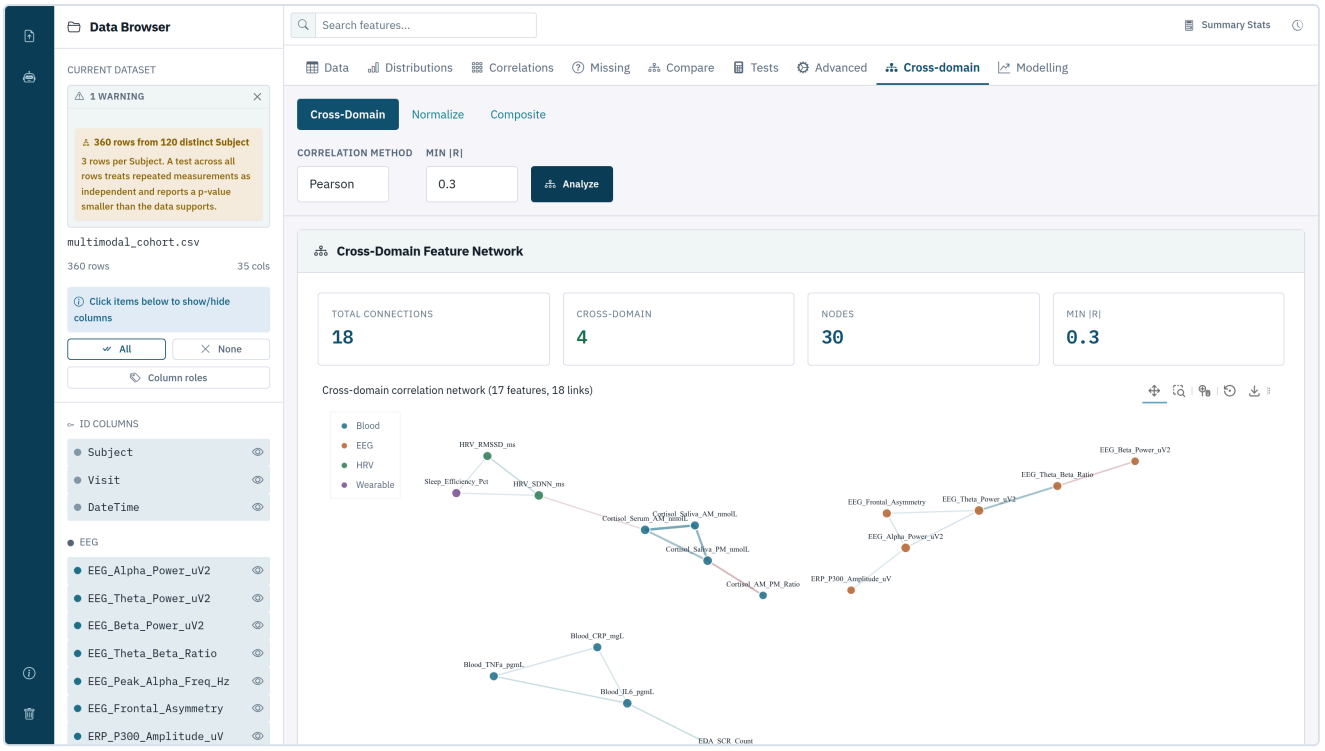

Figure 36 — Which domains move together, as a network.

Each node is a measurement and each edge a correlation above your threshold, laid out so that domains which move together cluster. It answers a question a correlation matrix cannot: is there structure *between* modalities, or does each domain only correlate with itself?

How a column reaches a domain. The workbench reads the head of the column name first — `HRV_RMSSD_ms` is HRV, `EEG_Alpha_Relative_Power_pct` is EEG — and falls back to the rest of the name only if the head says nothing. The most specific match wins, so `HRV_` is never taken by the `HR` that belongs to Cardiac. Columns that describe the recording rather than the subject — coverage, window length, sampling rate, the number of preprocessing steps — are deliberately left out; they are context, not biomarkers.

Domain	Covers
EEG	EEG and ERP measures, band powers, P300
HRV	Heart-rate variability
Cardiac	HR, pulse rate, IBI, BVP, ECG, PPG
Wearable	EDA, motion, sleep, activity, temperature, respiration, SpO_2
Blood, Saliva, Urine, CSF, Sweat, Hair	Assay panels, by matrix
Scales	Questionnaire instruments

A SINGLE-MODALITY FILE NEEDS TO SAY WHAT IT IS

Cross-domain analysis needs at least two domains to separate, so a file that matches only one is not given a domain at all — the workbench will not conclude that a table is biomedical on one family's evidence. A file that arrives with a manifest from the EEG or Wearable Workbench does not have to be guessed at: it says which workbench wrote it, and is categorised on that basis even when everything in it is EEG.

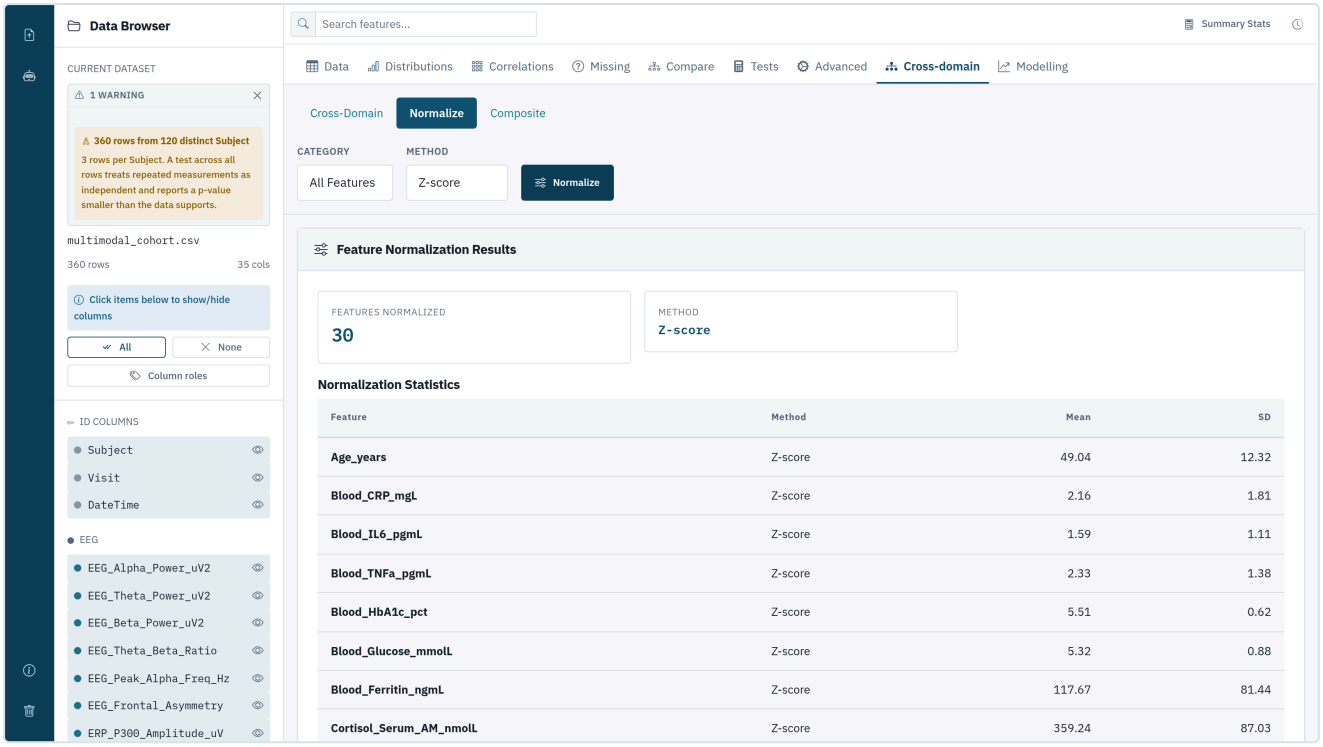

Figure 37 — Putting domains on one scale.

Normalisation exists because domains do not share units — a cortisol in nmol/L and an EEG power in µV² cannot be averaged as they stand. **Z-score** centres each column and scales it to unit variance. **Min-Max** maps to [0, 1]. **Robust (IQR)** uses the median and interquartile range, so outliers do not drag the scale.

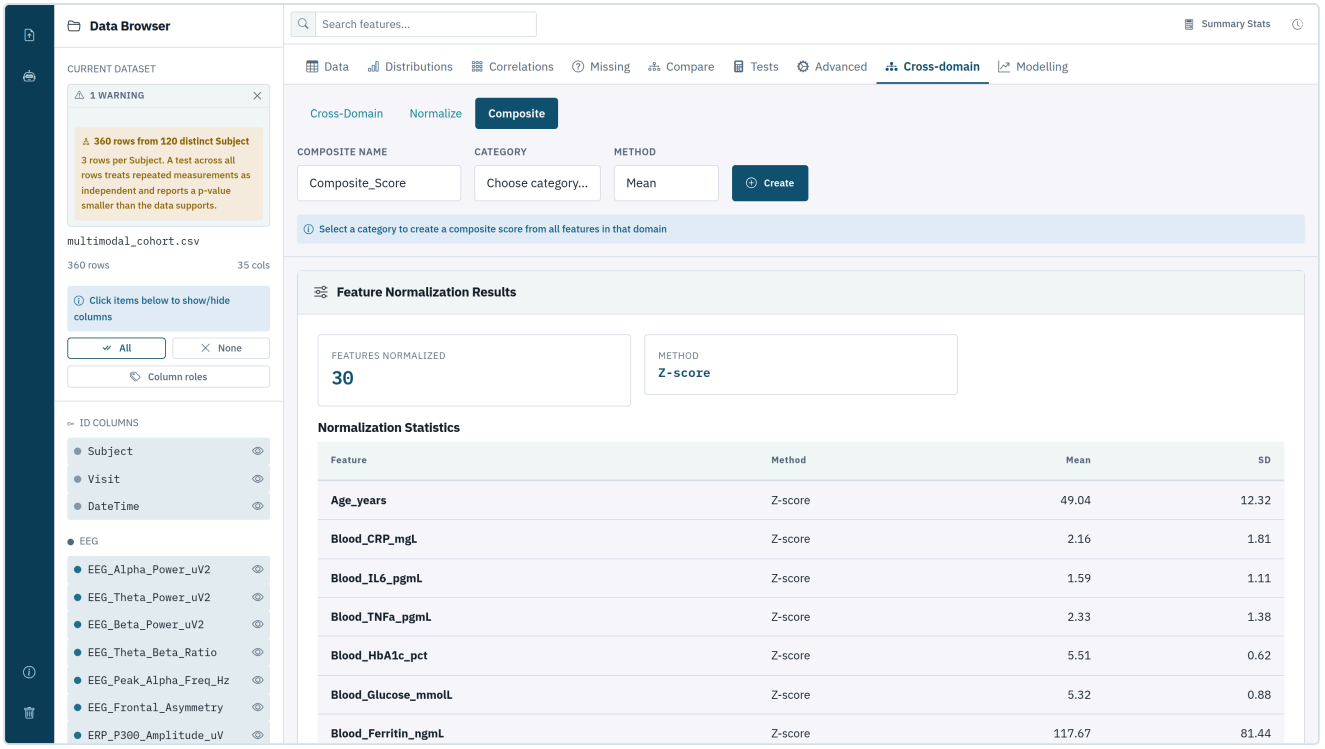

Figure 38 — Building a composite index from a domain.

A composite collapses several measurements into one index — an inflammation score from the blood panel, say. Normalise first, or the column with the largest numbers will dominate the result. The composite is added to your dataset and can be used anywhere a measurement can.

Modelling

The Modelling tab has seven benches behind one row of buttons: Fit, Predict, Cluster, Curve, Calibrate, Forecast and Compare.

Fitting a model

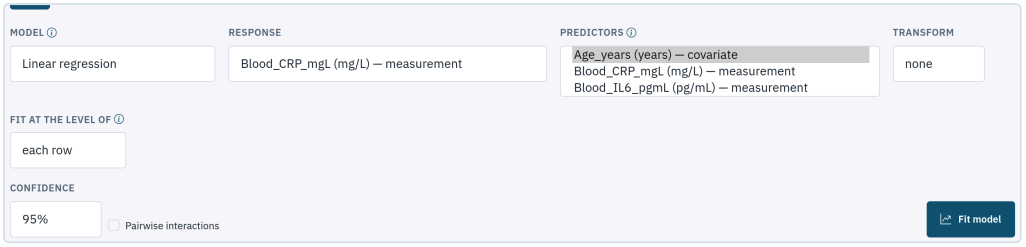

Figure 39 — The Fit bench: a response, some predictors, and a model family.

Twelve model families sit behind one selector — linear, polynomial, weighted, robust, ridge, lasso, elastic net, PLS, quantile, logistic, mixed effects and two-way ANOVA. The controls that appear change with the family: choose ridge and you get a penalty box, choose polynomial and you get an order.

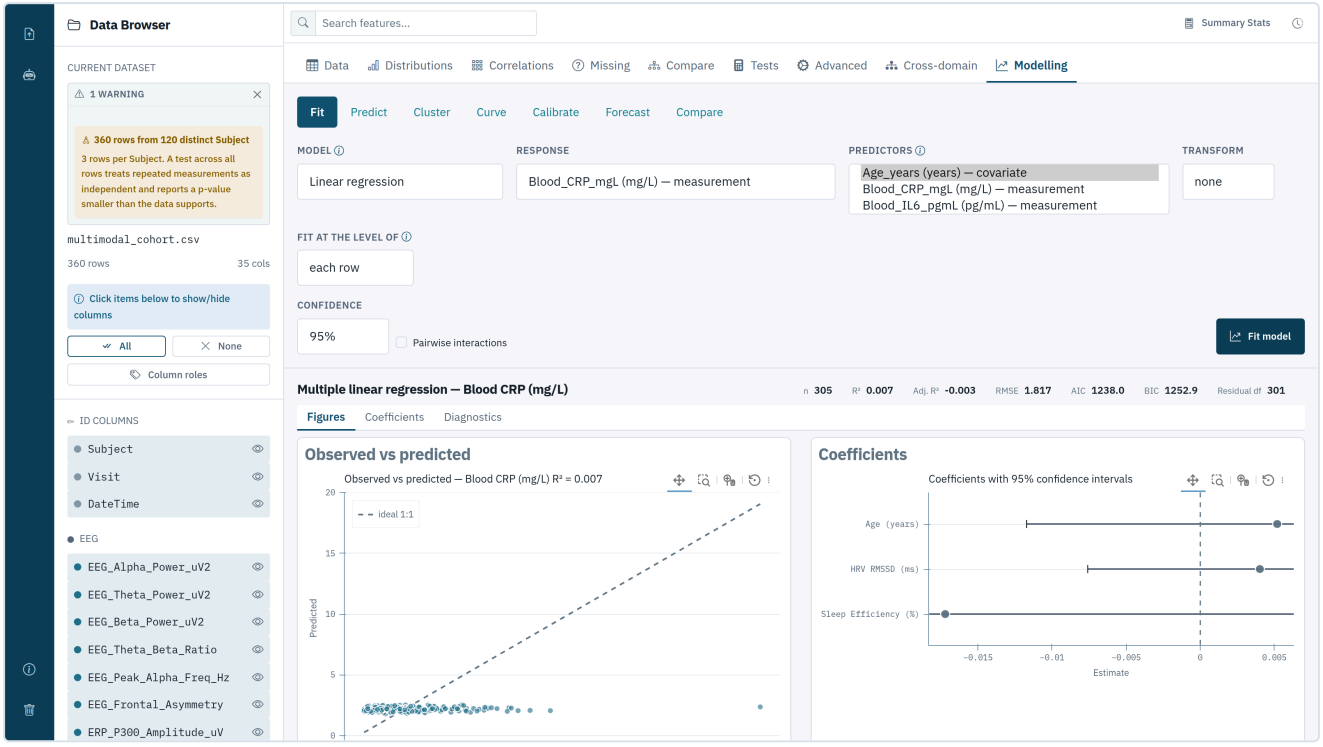

Figure 40 — A multiple linear regression over three predictors.

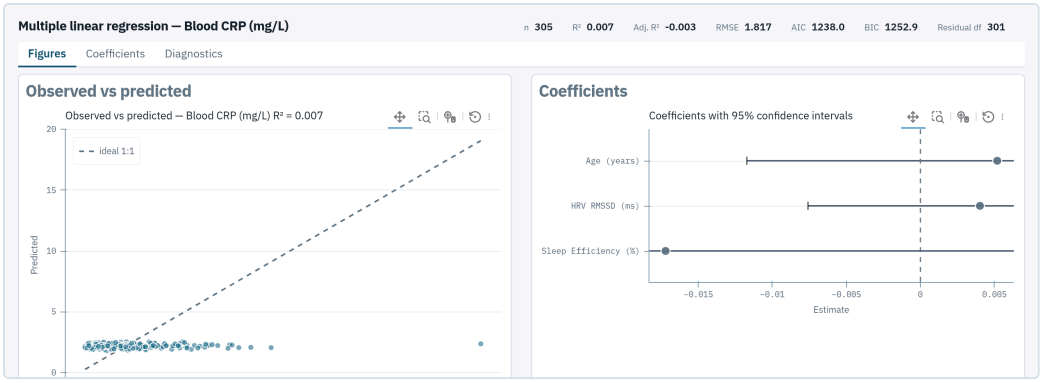

Figure 41 — The coefficient table, with confidence intervals and diagnostics.

Every term reports an estimate, a standard error, a confidence interval and whether that interval crosses zero. The Diagnostics tab beside the coefficients carries the residual plots — check them before believing the coefficients.

Answering the repeated-measures problem properly

Chapter 10 aggregated to one row per subject, which works but throws away the within-subject information. The alternative keeps every row and models the structure instead: **Mixed effects (random intercept)**, with `Subject` as the random group.

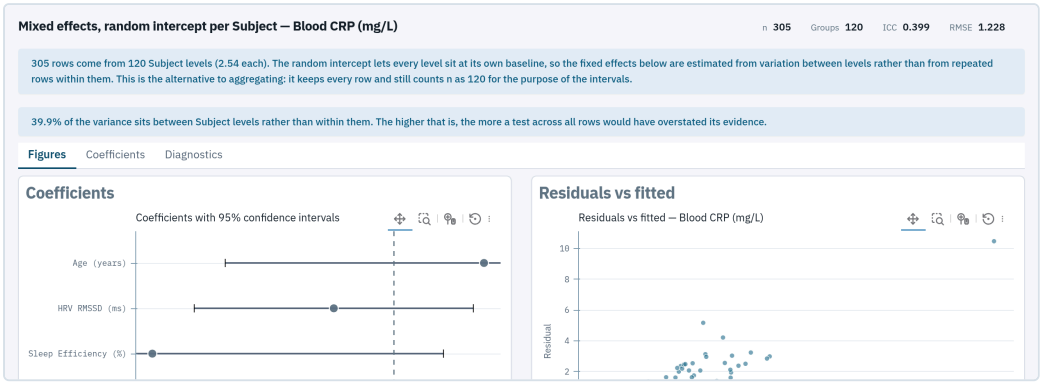

Figure 42 — A random intercept per subject: every row kept, and the evidence counted correctly.

Two numbers here are worth understanding. **Groups** is 120 — the effective sample size for the intervals, not the 305 rows. And the **ICC** of 0.399 says that 40% of the variance sits between subjects rather than within them. The higher that number, the more a test across all rows would have overstated its evidence.

Clustering

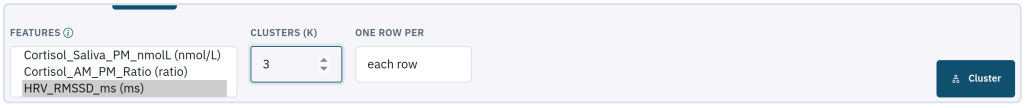

Figure 43 — k-means over the autonomic panel.

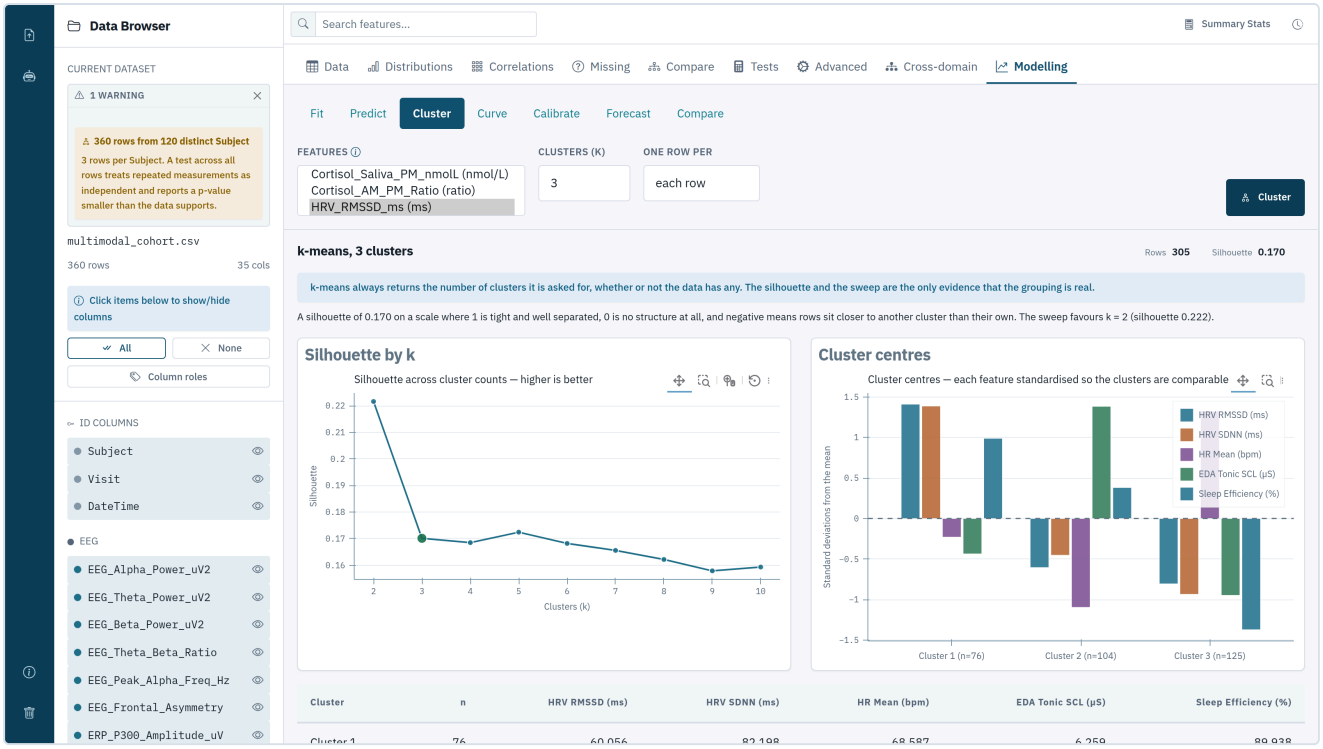

Figure 44 — Three clusters, with the silhouette score that says how separated they are.

Choose the features and a **k**. The silhouette score reports how well separated the clusters are: near 1 is clean separation, near 0 means the boundaries are arbitrary, and negative means points are closer to a neighbouring cluster than their own.

> **K-MEANS ALWAYS RETURNS K CLUSTERS**
>
> It will partition noise as happily as it partitions structure. The silhouette score is what tells you which of those two happened.

Prediction and cross-validation

The Predict bench fits a learner and scores it out of sample. Seven are available: ridge as a baseline, then k-nearest neighbours, decision tree, random forest, gradient boosting, support vector machine and PLS.

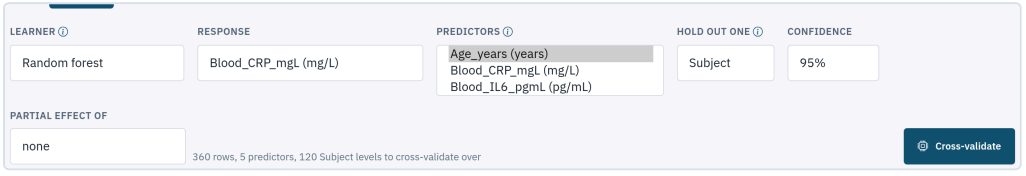

Figure 45 — A random forest with folds grouped by subject.

The **Fold grouping** control is the one that matters most and is easiest to skip. Left as random folds, two visits from the same subject can land in different folds — the model sees that subject in training and is then scored on them in testing. The score that comes back is inflated, and the model looks better than it is.

Set it to `Subject` and cross-validation holds out one subject at a time.

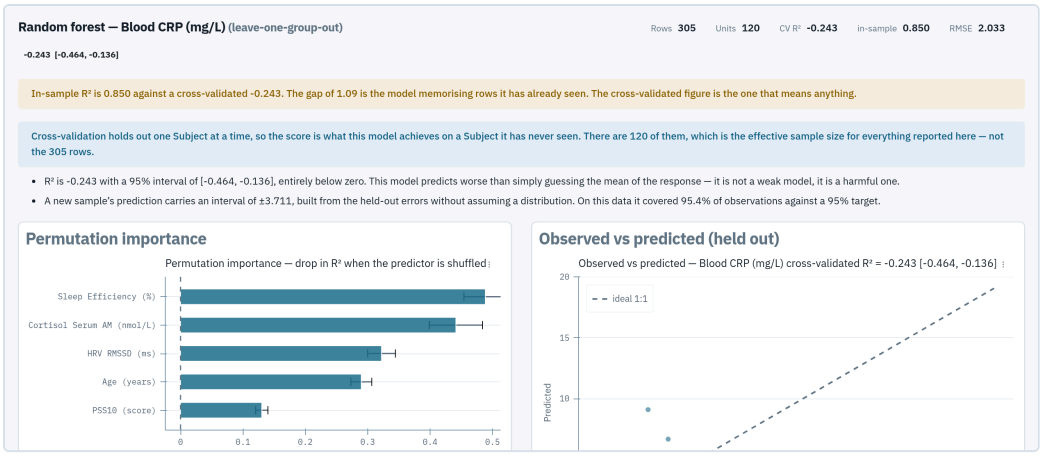

Figure 46 — In-sample 0.850 against cross-validated −0.243. The gap is the model memorising rows it has already seen.

This figure is the reason the control exists. In-sample R^2 is 0.850, which looks excellent. Cross-validated R^2 is −0.243, with an interval entirely below zero — the model predicts worse than guessing the mean. The workbench states it directly: it is not a weak model, it is a harmful one.

The effective sample size is reported as 120 subjects, not 305 rows, and the permutation importance panel shows which predictors the model leaned on.

> **ALWAYS READ THE CROSS-VALIDATED FIGURE**
>
> In-sample R^2 measures how well a model memorised. Cross-validated R^2 measures whether it learned anything. When they disagree, the second one is the answer.

Fitting a curve

The Curve bench fits 21 named nonlinear shapes — isotherms, kinetics, dose response, and peak shapes. The figures here use a voltammogram.

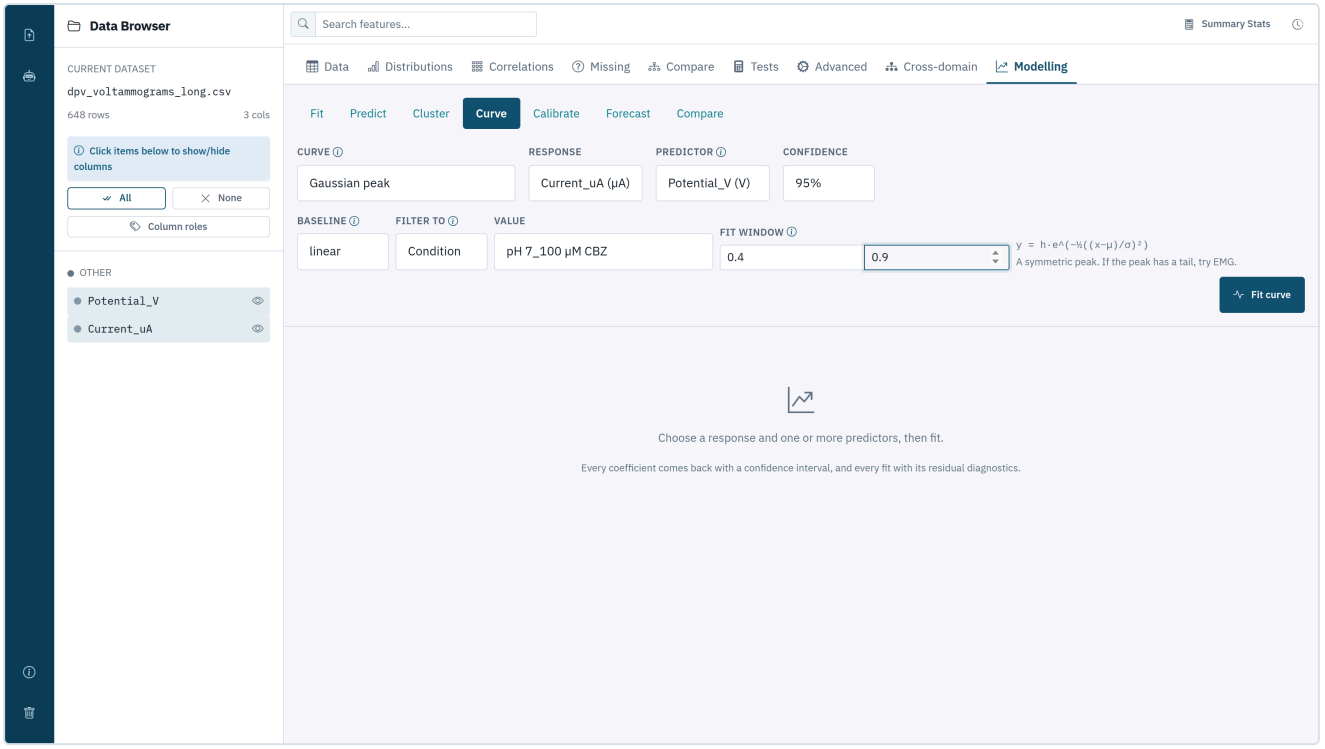

Figure 47 — Fitting a Gaussian peak on a linear baseline, restricted to a potential window.

Four controls shape the fit. **Baseline** subtracts a constant or linear background before fitting, which is what you want for a peak on a sloping baseline. **Filter** restricts to one experimental condition. **X min** and **X max** set the window, so a peak can be fitted without the rest of the sweep pulling on it.

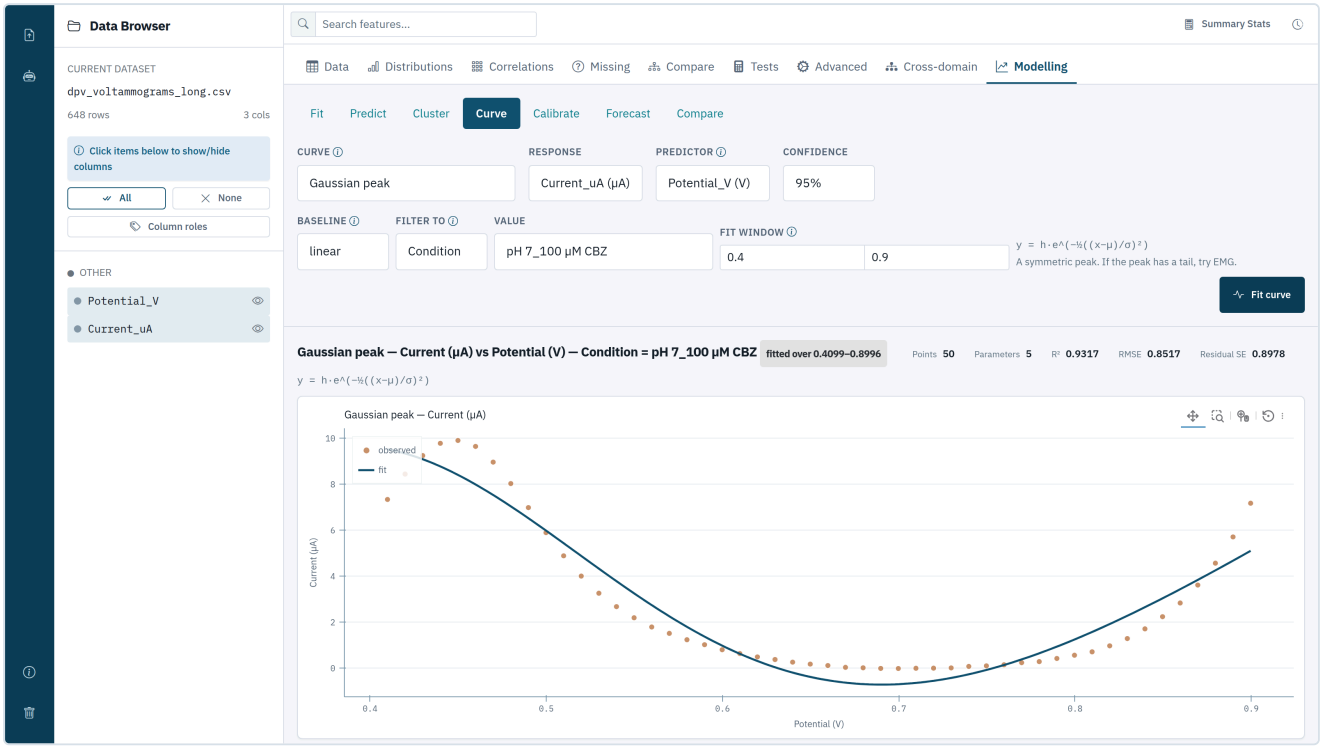

Figure 48 — The fitted peak over the observed sweep.

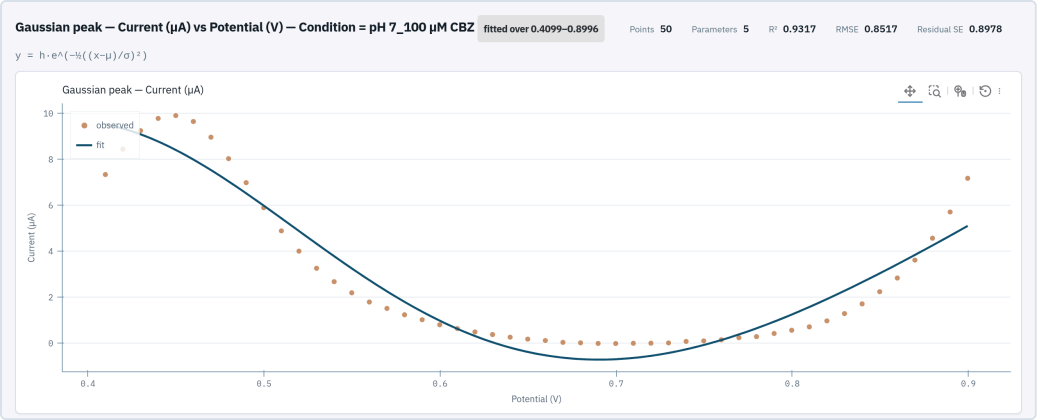

Figure 49 — What the fit reports: parameters with intervals, and goodness of fit.

Every parameter comes back with a confidence interval. A parameter whose interval spans an implausible range is telling you the model is not identified by this data, however good the R^2 looks.

Calibration

The Calibrate bench turns a series of standards into a calibration curve, a limit of detection and a limit of quantitation.

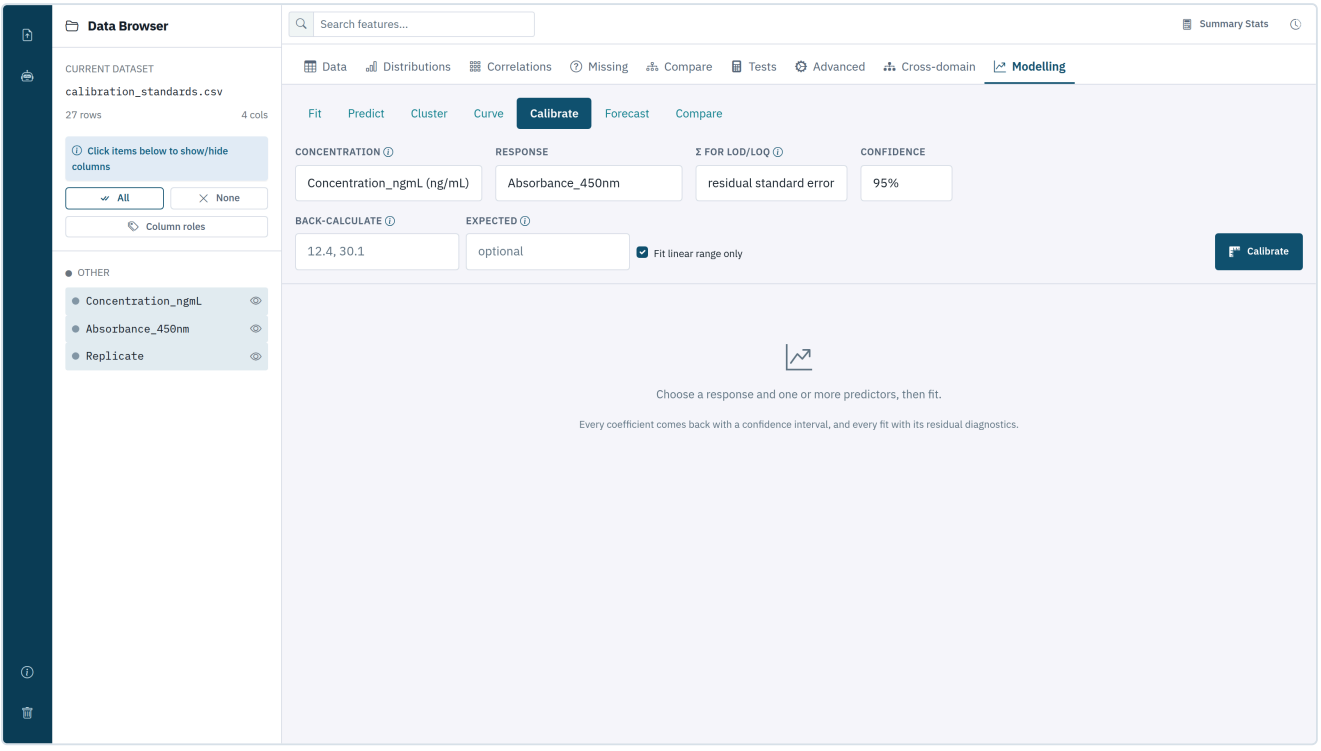

Figure 50 — Calibration needs a known concentration and the response measured at it, in two columns.

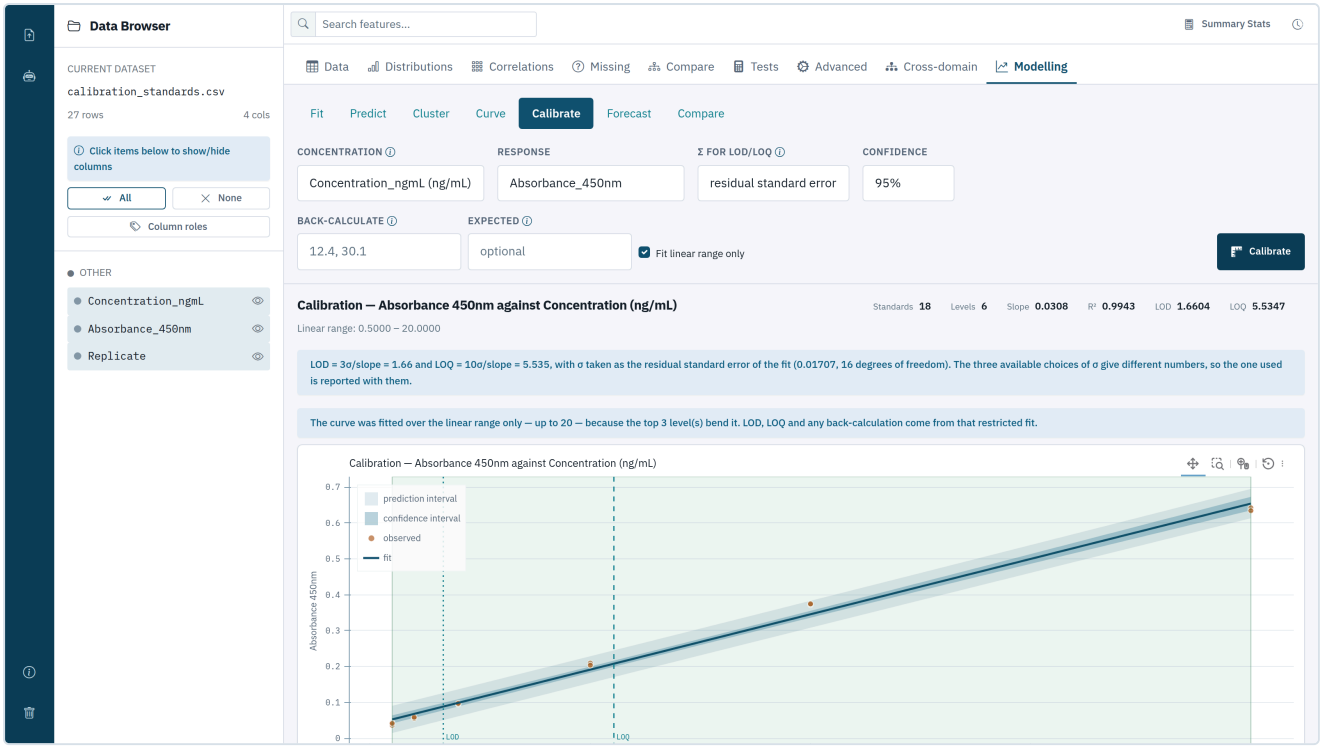

Figure 51 — The fitted curve, restricted to its linear range, with LOD and LOQ marked.

Figure 52 — LOD, LOQ, and the note explaining which levels were dropped.

Three things happen here that are worth calling out, because each is a decision the workbench makes explicit rather than silently.

The linear range is found, not assumed. The standards run to 200 ng/mL, but the top three levels bend the line. The fit is restricted to 0.5–20 ng/mL and the message says so: *including them bends the line, which is saturation, not noise.* LOD, LOQ and any back-calculation come from the restricted fit.

σ is declared. LOD is 3σ/slope and LOQ is 10σ/slope, but σ has three reasonable definitions — the residual standard error of the fit, the standard deviation of blank replicates, or the standard deviation of the lowest standard. They give different numbers, so the one used is reported alongside them. Choose it with the **σ for LOD/LOQ** selector.

Back-calculation knows about the LOQ. Enter sample responses in **Back-calculate** and they are turned into concentrations, with intervals from Fieller's theorem — which widen away from the centre of the standards, as they should.

Figure 53 — Three responses turned back into concentrations, with one flagged below the LOQ.

One of the three falls below the LOQ, and the workbench says to report it as `< LOQ` rather than as a number. Add **Expected** concentrations in the same order to get recovery reported against them.

Forecasting

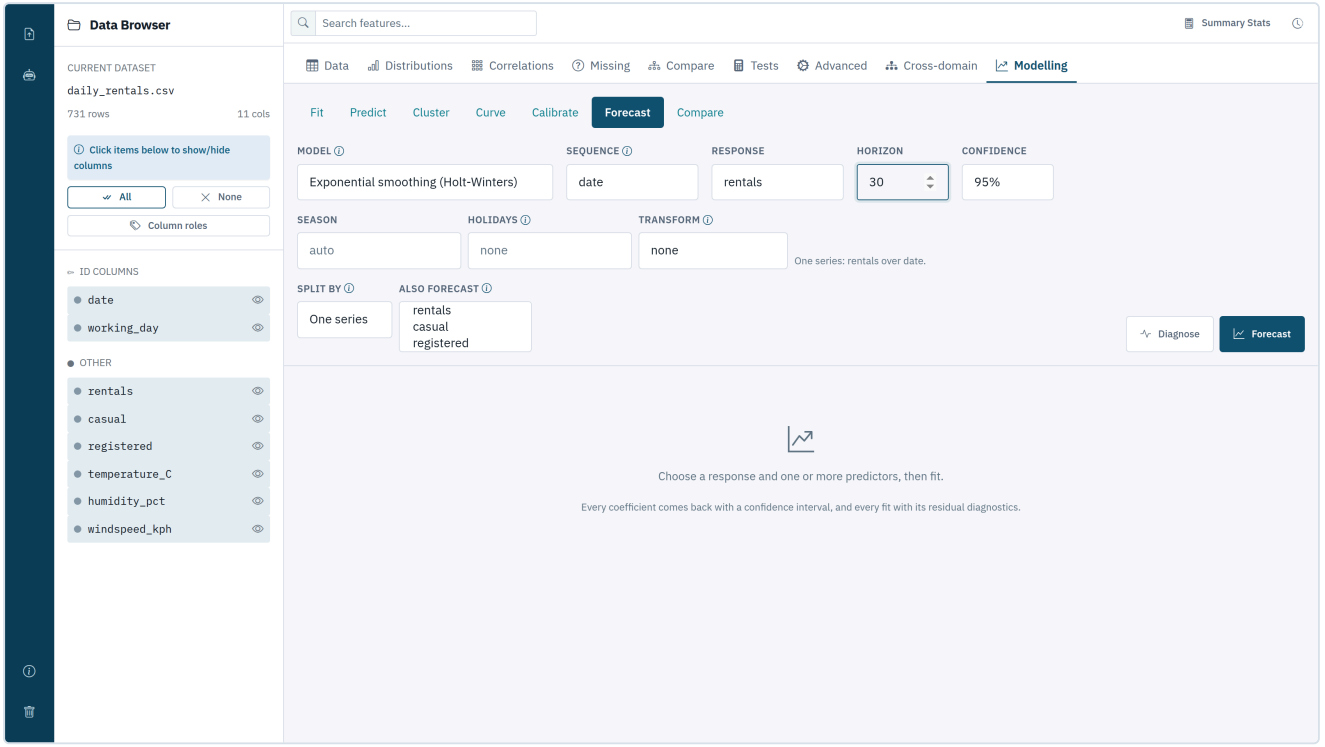

Figure 54 — A forecast needs a sequence column and a response.

Eleven models are available, from naive baselines through exponential smoothing, ARIMA and SARIMA to a calendar model with Fourier seasonality and holidays.

Diagnose before you forecast. It reports what the series actually looks like — trend, seasonality, stationarity — so you can choose a model on evidence rather than habit.

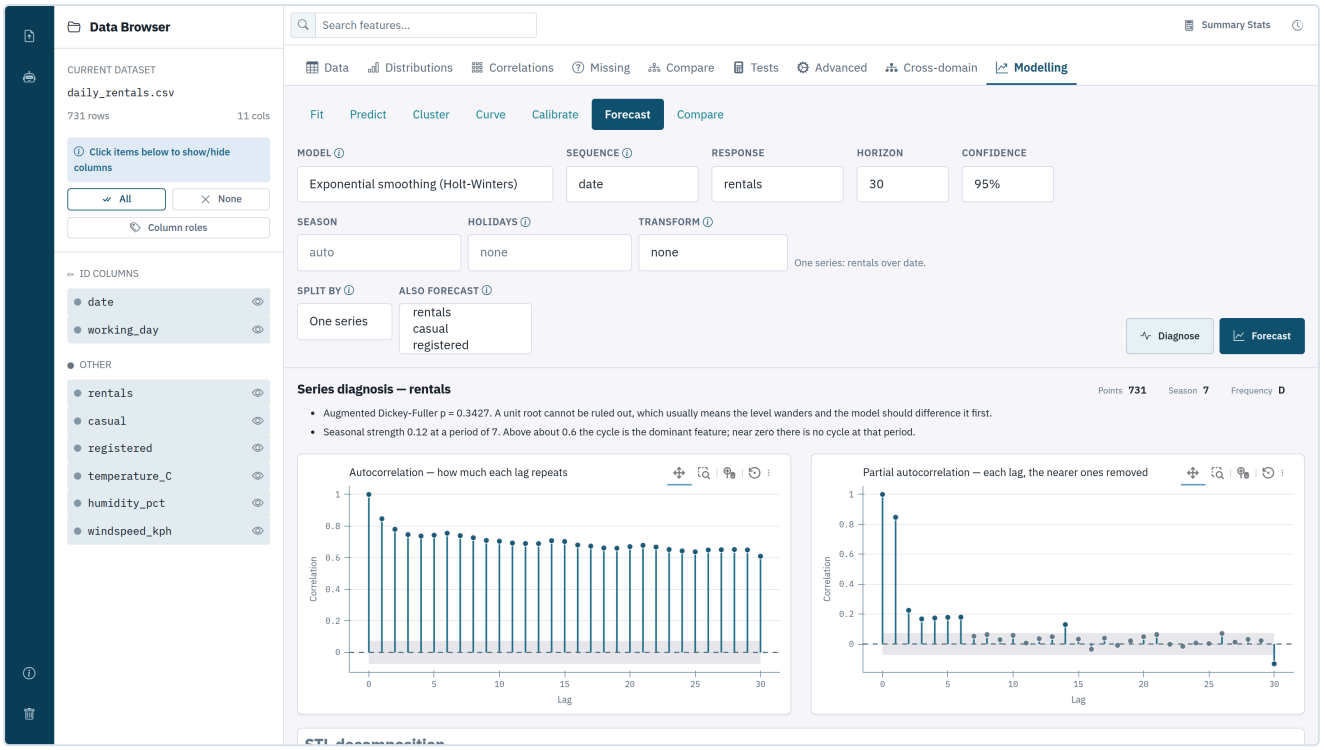

Figure 55 — What the series looks like before a model is chosen.

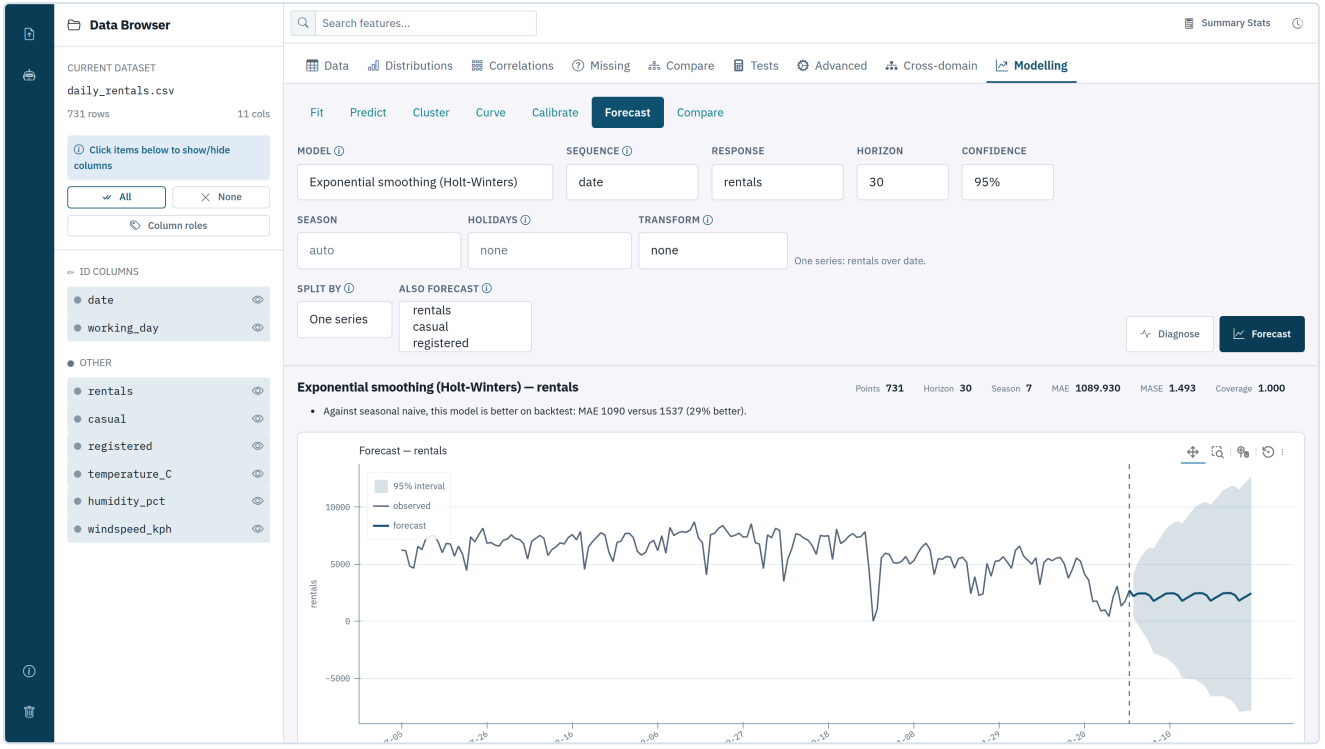

Figure 56 — Thirty days ahead, with a 95% interval.

Figure 57 — The backtest behind the forecast.

The number to read is **MASE** — mean absolute scaled error. Below 1 means the model beats a seasonal naive forecast; at or above 1 means it does not, and you should use the naive one. Here the model is 29% better than seasonal naive on backtest, which earns it.

> **AN INTERVAL THAT GOES NEGATIVE ON A COUNT IS TELLING YOU SOMETHING**
>
> Prediction intervals are computed on the scale you fitted on. If the lower bound of a count forecast falls below zero, either the horizon is too long for the data, or the response wants a log transform first.

Comparing models

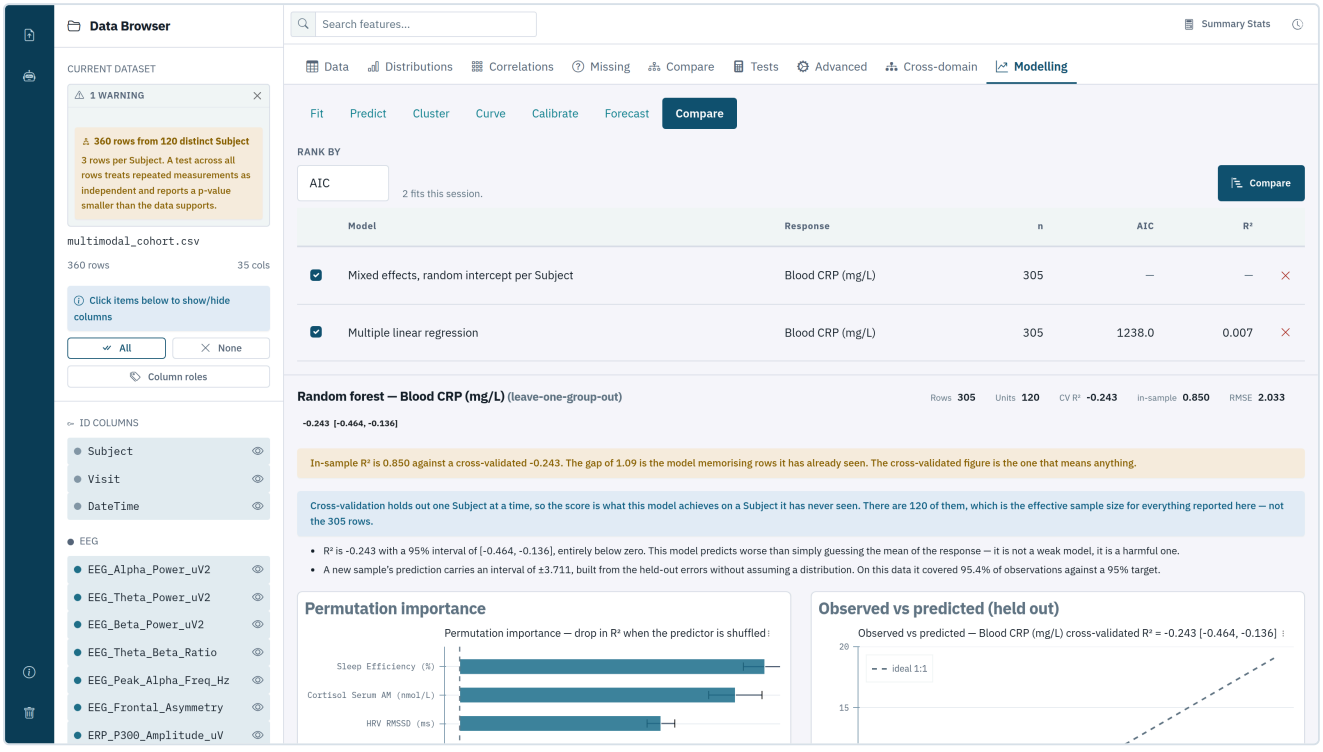

Figure 58 — Every model fitted this session, ranked side by side.

Each fit is added to a registry as you run it. Compare ranks them by AIC, BIC, adjusted R^2, R^2 or RMSE.

ONLY COMPARE MODELS OF THE SAME RESPONSE ON THE SAME ROWS

AIC across different responses, or different row counts, is not a comparison. The table shows n for each row so you can check before you rank.

Writing it up

Analysis history

Every analysis is recorded as you run it, with its parameters.

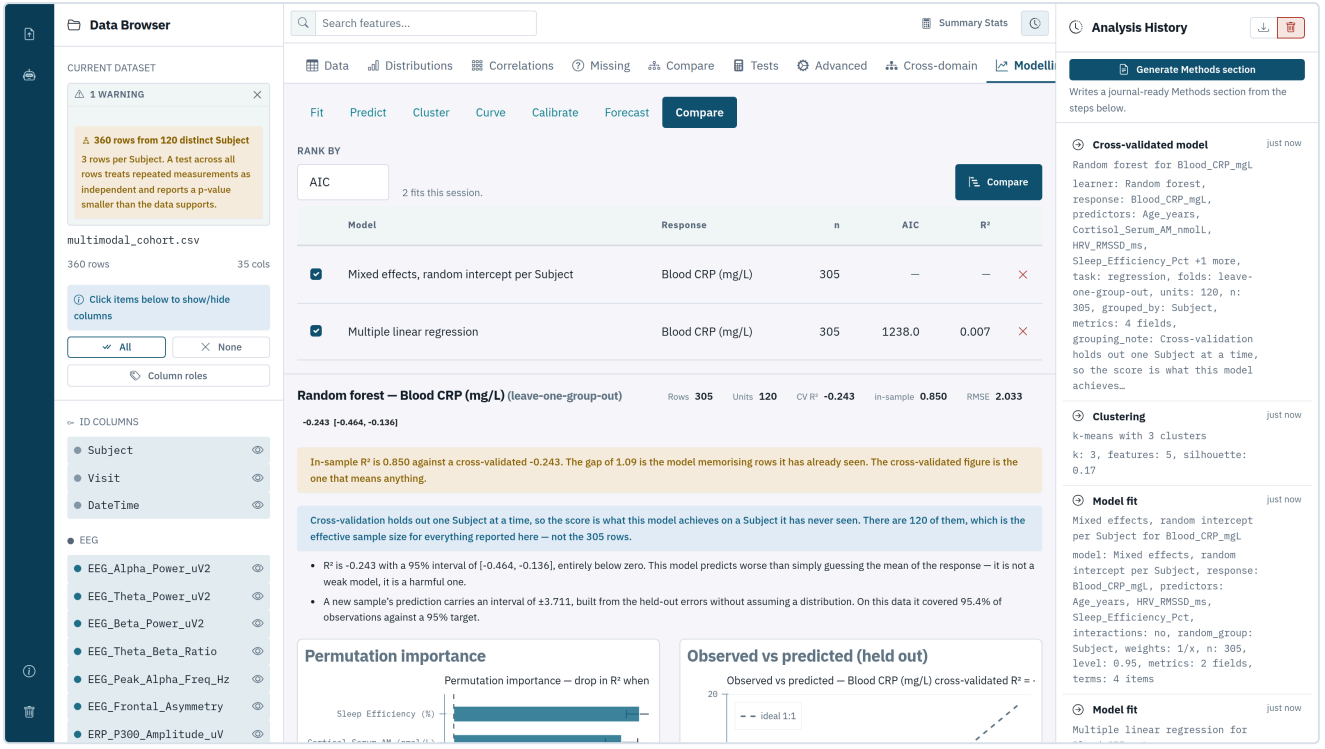

Figure 59 — The analysis history, in the order things were run.

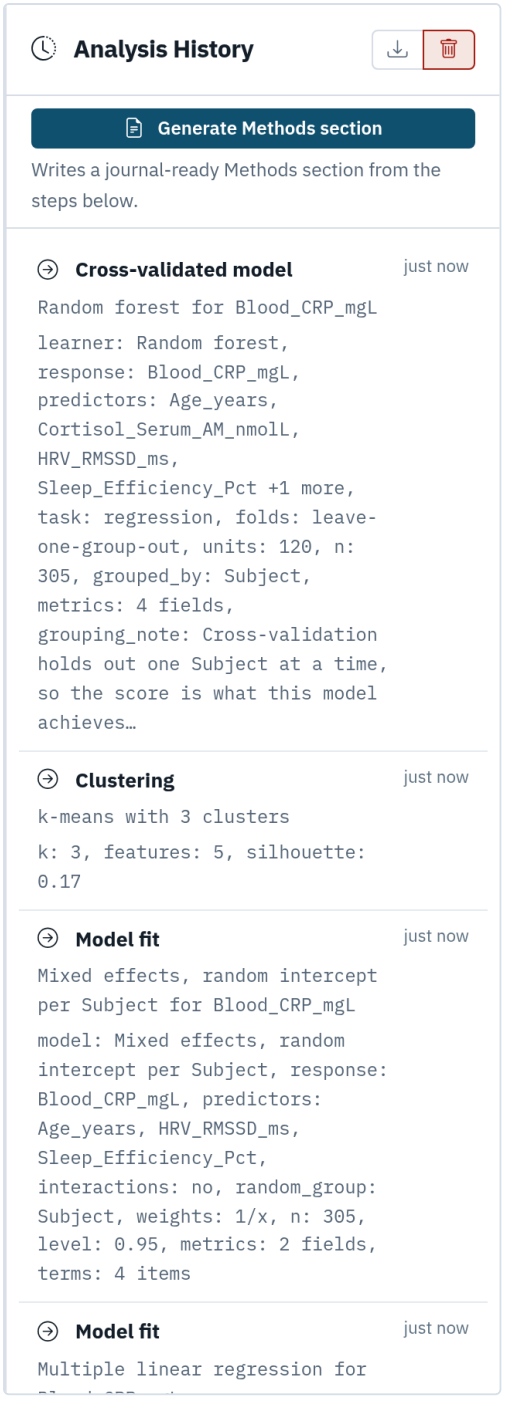

Figure 60 — Each entry records what was run and with which settings.

Open it with the clock icon in the toolbar. You can download the history as a record of the session, or clear it.

This is what makes a session reproducible, and it is what the Methods writer reads.

The Methods section

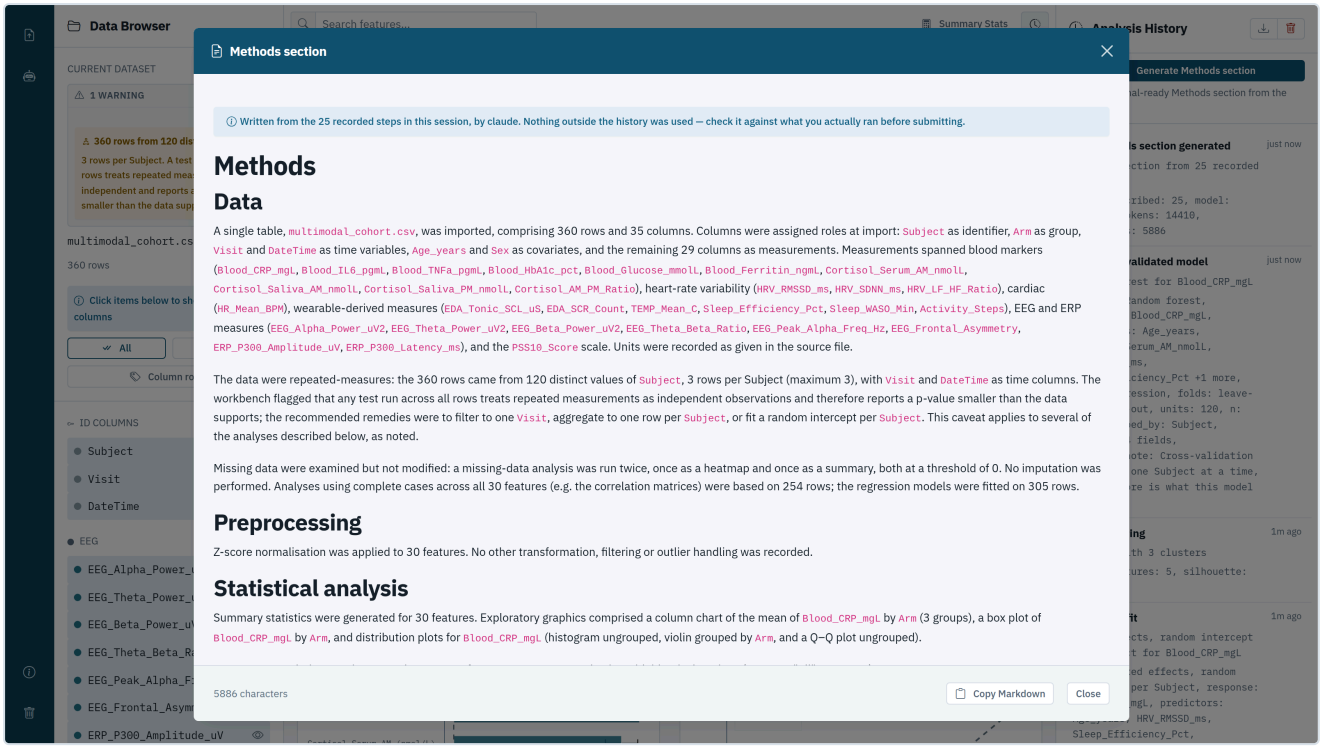

Figure 61 — A journal-ready Methods section, written from the recorded steps.

Generate Methods section turns the history into prose: what the data was, how it was preprocessed, which tests were run with which policies, and which models were fitted. Copy it as Markdown with the button at the bottom.

Figure 62 — The provenance line states how many steps were used and by which model.

> **CHECK IT BEFORE YOU SUBMIT IT**
>
> The Methods writer uses nothing outside the recorded history — it cannot describe a step you did not run in the workbench, and it will not invent one. It can still misdescribe what you meant. Read it against what you actually did.

Where a table arrived from the EEG or Wearable Workbench, the preprocessing done **there** is in the history too, tagged with the table and the workbench it came from. The provenance therefore does not stop at this application's front door: a filter applied to the signal before the measures were derived is part of the record.

The AI research assistant

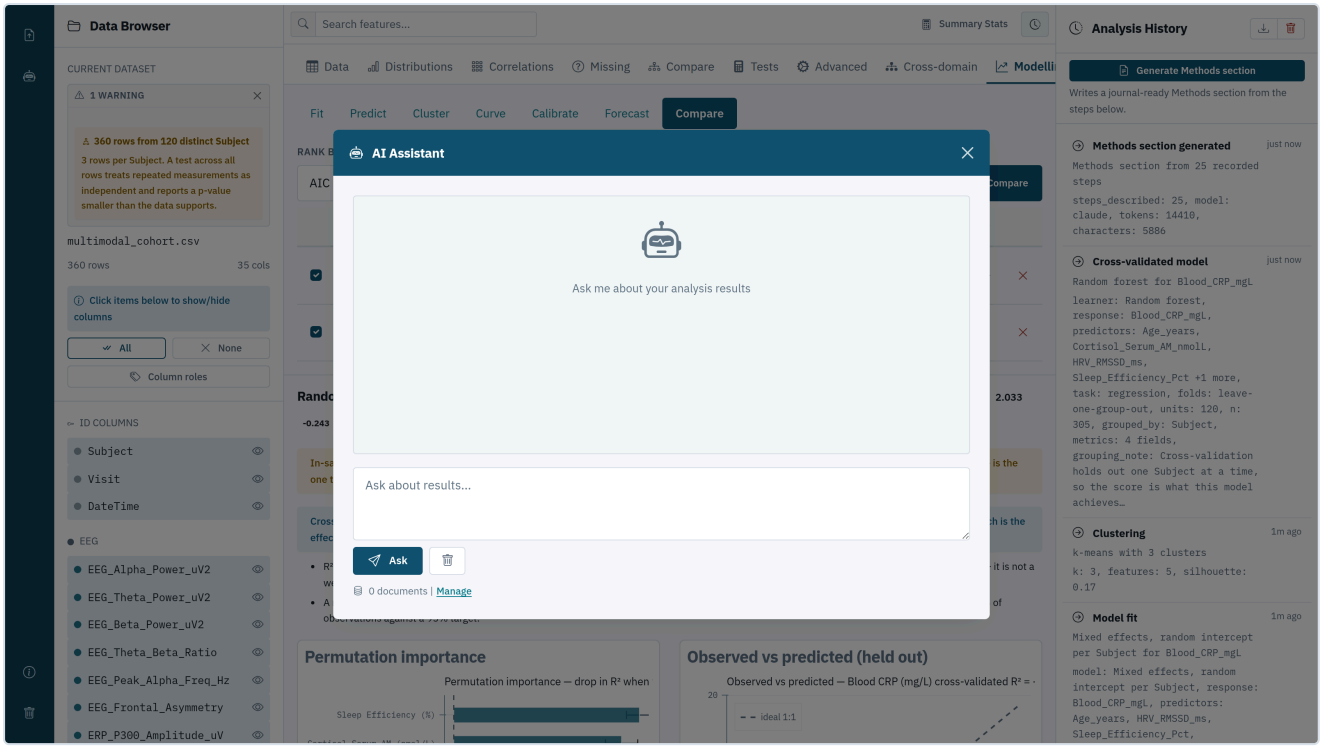

Figure 63 — The assistant, before a question is asked.

The assistant answers questions about the dataset you have loaded, grounded in the numbers it has been given rather than in general knowledge.

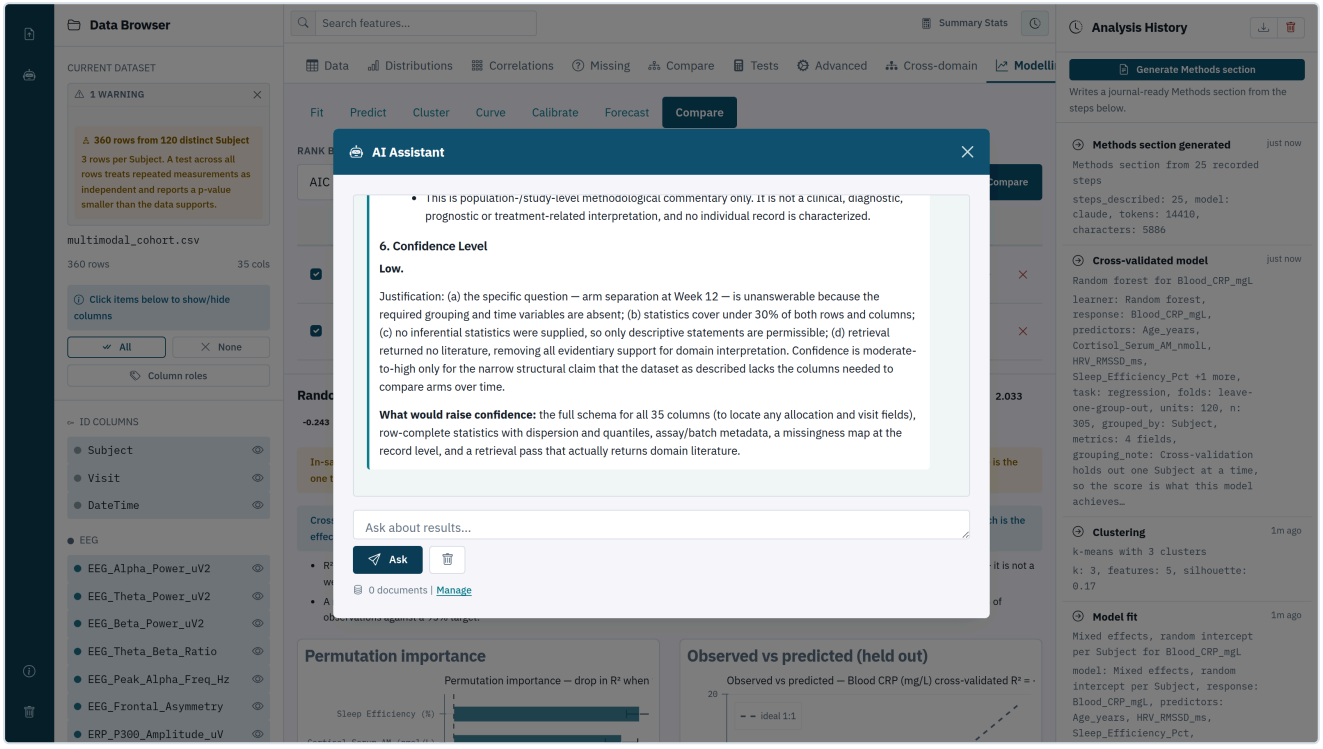

Figure 64 — An answer grounded in the loaded dataset.

It is at its best on questions of the form *what does this show, and what does it not establish?* You can also upload your own literature and have retrieval draw on it.

> **IT IS AN ASSISTANT, NOT AN AUTHOR**
>
> Ask it about the data you have loaded. Treat anything it says about the wider literature the way you would treat a colleague's recollection — as a pointer to check, not a citation.

Appendix A — Control reference

The rail and toolbar

Control	Effect
Import	Opens the import dialog
AI Assistant	Opens the assistant; disabled until data is loaded
About	Version and build information
Clear Session	Discards the data and the analysis history
Search features	Filters the column list
Summary Stats	Descriptive statistics for every column
History (clock)	Opens the analysis history sidebar

Data

Control	Effect
Group By	The categorical column to group by; clear it for distribution plots
Value	The measurement to summarise
Summarise	Mean, median, sum, count, min, max
Plot Type	Column, bar, line, histogram, box

Distributions

Control	Effect
Select Feature	The measurement to plot
Plot Type	Histogram, box, violin, Q-Q
Group By	Splits the plot by a factor

Correlations

Control	Effect
Correlation Method	Pearson, Spearman, Kendall
Filter By Category	Restrict to one domain
Min |r|	Hide weaker associations

Missing

Control	Effect
View Type	Pattern heatmap, counts by feature, summary table
Threshold (%)	Hide columns below this level of missingness

Compare

Control	Effect
Select Feature	The measurement to compare
Group By	The factor defining the groups
Plot Type	Box, violin, group means with 95% CI

Tests

Control	Effect
Feature	The measurement to test
Test Type	t-test, paired t-test, Mann-Whitney, Wilcoxon, ANOVA, Kruskal-Wallis
Group By	The factor defining the groups
Correction	None, Bonferroni, FDR, Holm
Repeated measures	All rows, average per unit, one timepoint
Adjust for (ANCOVA)	Covariates to adjust the comparison for
Batch: Category Filter	Restrict the screen to one domain

Advanced

Mode	Needs	Reports
PCA	Numeric columns	Components, explained variance, loadings
Spectral	A sequence and a sampling rate	Dominant periods
Changepoints	A sequence and a threshold	Where the level shifts
Longitudinal	A variable, a time column, optionally a group	Trajectories, mean change, slope

Cross-domain

Mode	Effect
Cross-domain	Correlation network across domains
Normalize	Z-score, min-max or robust scaling
Composite	Combines a domain into one index by mean, weighted sum or PC1

Modelling

Bench	Fits
Fit	Twelve model families, from linear to mixed effects
Predict	Seven cross-validated learners, with optional fold grouping
Cluster	k-means, with a silhouette score
Curve	21 named nonlinear curves, with baseline and window
Calibrate	Calibration curve, linear range, LOD/LOQ, back-calculation
Forecast	Eleven forecasting models, with backtest and diagnosis
Compare	Ranks the session's fits by AIC, BIC, adjusted R^2, R^2 or RMSE

Appendix B — What the workbench refuses to do

The workbench declines analyses that its data cannot support, and says why. A refusal is working as designed. These are the ones you are most likely to meet.

A two-sample test against three or more groups. An independent t-test or Mann-Whitney compares exactly two groups. Use ANOVA or Kruskal-Wallis. Every refused measurement is listed with its reason and excluded from the multiple-testing family.

A calibration curve with fewer than four standards, or fewer than three distinct levels. A slope through two levels is a line through two points.

A calibration whose top levels bend the line. Not refused, but restricted: the fit is confined to the linear range and the message says how many levels were dropped.

A back-calculated concentration below the LOQ. Reported as `< LOQ` rather than as a number.

A test across repeated measures. Not refused, but warned about at import and again on every result, with the three policies offered inline.

A grouping column with too many levels. Grouping by something with hundreds of values produces a chart no one can read. Identifier columns are still offered where they belong — as a random-effects group, and as a cross-validation fold grouping.

A model whose parameters are not identified. The fit returns, and the confidence intervals will be enormous. That is the refusal.

Appendix C — Datasets in this guide

Dataset	Where it came from
`multimodal_cohort.csv`	Synthetic, shipped with the workbench in `SampleData/`. 120 subjects × 3 visits × 3 arms. Effects are planted in three tiers — strong, underpowered, and null — so that a correct analysis and an over-eager one give visibly different answers.
`dpv_voltammograms_long.csv`	Differential-pulse voltammograms, shipped in `SampleData/`. Nine conditions across three pH values.
`calibration_standards.csv`	A synthetic ELISA-style standard curve created for this guide: nine levels in triplicate, 0.5–200 ng/mL, with detector saturation above 20 ng/mL.
`daily_rentals.csv`	Two years of daily counts with weekly seasonality and weather covariates, shipped in `SampleData/open/`.

None of these are patient data. If you reuse the open datasets beyond testing, check their licence and attribution requirements first.