

Rosa

Glossary of report and manifest terms

This glossary defines every term and field that appears in Rosa's two evidence outputs: the **PDF report** (produced for Diagnose and Training jobs) and the **Run Manifest** (the machine-readable JSON record emitted for every job). Your PDF report and your Run Manifest are two views of the same measured figures, so they never disagree on a value.

1. The bias measurement (read this first)

Rosa measures bias by asking a single question: **how well can a model predict the protected attribute from the rest of your data?** If the other columns carry no information about it, a predictor can only do as well as chance. If they carry information (directly, or through proxy columns), a predictor beats chance, and that margin above chance is the bias.

Five terms do all the work:

Term	What it means
Bias characteristic	The variable named in <code>bias_columns</code> , the one Rosa measures and removes bias with respect to. It need not be a legally protected characteristic; an operational variable such as regional office is equally valid. Rosa handles one per run.
Bias characteristic groups	The distinct values of that column. The number of groups is not a limit; the size of the smallest one is what to watch. The detection test weights every group equally, whatever its share of the rows, so a dominant group does not defeat it; what limits it is how few examples of the rarest group it has to learn from. How many rows that takes depends on how many groups the column has: with two or three, detection has held on a couple of dozen rows in the smallest group; at eight, we found none at 120 rows and detection on every run at 181, with the space between unmeasured. Training runs on a sample of at most about 10,000 rows drawn roughly in proportion, so a group's share of your population sets its row count in the sample, and a bigger file does not raise it. A group with fewer than nine rows in that sample stops the detection test from running at all: Rosa returns "detection not supported" (<code>declined_insufficient_support</code> , section 1.2) rather than a verdict, because a group that small cannot be scored on a held-out split. Nine is the point below which the test cannot run; it is not a size at which detection is reliable.
Average Discriminator Performance	The held-out score a trained detector network reaches when predicting the bias characteristic from the rest of the data. The score is balanced accuracy: the mean of the per-group recall rates, so every group counts equally whatever its share of the rows. Higher means the characteristic is more recoverable, i.e. more bias. Reported as an average over several training runs, with a standard deviation. The manifest's <code>bias_scorer</code> field records the scorer used.

Term	What it means
Random Baseline Accuracy	The score that same measurement reaches by chance on data of this shape. It is established by re-running the measurement on randomly shuffled copies of the bias characteristic: with the labels shuffled there is no real signal, so whatever score the detector still reaches is pure chance. This is the honest floor the bias score is measured against.
Best Random Accuracy	The chance level of the scorer in use, which sets the scale of the score and nothing else. For balanced accuracy over K groups it is 1/K (for example 0.20 on a five-group characteristic), whatever the group sizes. The share of the most common group is recorded beside it on the manifest as <code>majority_share</code> . Manifests from runs before <code>bias_scorer</code> was recorded used plain accuracy, and there the Best Random Accuracy was that most-common-group share.
Bias Score	How far the detector beats chance, expressed on a 0 to 100% scale. This is the "before" score.
Residual Bias	The same Bias Score measurement, re-run on your data after Rosa has debiased it. This is the "after" score. Training jobs only. A residual reading of zero means Rosa's own detector found nothing left at the resolution it ran at; it is not a claim that no estimator could find anything, and the Starter Kit's EVIDENCE/interpreting-results.md states that bound in full.

1.1 How the bias score is calculated

The score uses **both** chance measures. It subtracts the **Random Baseline Accuracy** (the honest chance floor), then rescales onto a 0 to 100% axis using the **Best Random Accuracy**:

```
Bias Score = (Average Discriminator Performance - Random Baseline Accuracy) / (100% - Best Random Accuracy)
```

- The numerator is how far the detector beats the chance floor (the Random Baseline Accuracy).
- The denominator is the room a perfect predictor has above the Best Random Accuracy, so the score reads 0% when the detector does no better than chance and 100% when the attribute can be predicted perfectly.

The two baselines are usually close, so for a quick read of a single figure you can treat them as one number.

Worked example on a two-group characteristic, with both around 50%: if the detector reaches 75%, the bias is $(75 - 50) / (100 - 50) = 50\%$, halfway to a perfect predictor. **Do not carry that shortcut into a reconciliation.**

How one number covers several groups. The dial asks one question whatever the number of groups: with the bias characteristic hidden, how well can a detector guess it back from the other columns, on rows it has not seen? Its score is balanced accuracy, the average of how often each group is correctly identified, so every group counts the same whether it is 82% of the file or 2%. Chance for that score is one in K for K groups. Two examples make the scale concrete.

- *Sex, two groups.* Chance is 50%. If the detector correctly identifies 80% of the women and 70% of the men, its score is the average, 75%, and the dial reads $(75 - 50) / (100 - 50) = 50\%$: halfway from chance to certainty.

- *Ethnicity, five groups at 82 / 8 / 5 / 3 / 2 percent.* Chance is 20%. If the detector correctly identifies 60% of the largest group and 40%, 30%, 30% and 20% of the other four, its score is the average, 36%, and the dial reads $(36 - 20) / (100 - 20) = 20\%$: a fifth of the way from chance to certainty.

The dial does not list the five groups; it averages how recoverable each one is and then places that average on the chance-to-certainty scale for five groups, so a 20% dial means the same thing on ethnicity as on sex. Two consequences follow. A detector that simply names the largest group for every row scores 100% on that group and 0% on the rest, an average of 20%, exactly chance, so the dial reads zero: guessing the majority reveals nothing and scores nothing. And because the dial is an average, it can be moved by one group: four groups at chance and one that is recovered every time give an average of 36% and a dial of 20%, the same reading as the spread-out example above. A modest dial can therefore be one group that the data gives away completely. The per-group figures on the manifest (section 2.2b) show which group moved the average; they are exploratory and are not a statement about any group on its own. If you are checking a printed score against its inputs, use the two baselines separately and use the right one: they are different numbers, and substituting one for the other moves the result by around a percentage point. **Residual Bias uses the identical formula on the debiased data, and every term in it is re-measured there** - including the chance floor, which is why the manifest records `residual_null_accuracy` separately from `null_accuracy`. The only term shared between the before and after scores is the Best Random Accuracy in the denominator, because the scale must not move between them.

Why a Residual Bias of exactly 0.000 is common, and what it means. The formula's numerator can come out **negative** - on a well-debiased dataset the detector often scores slightly *below* the shuffled-label chance floor, because the debiased data carries nothing to find and the detector's remaining accuracy is noise around chance. A negative bias score has no meaning (a dataset cannot be less than unbiased), so **Rosa floors the reported figure at zero**. On one measured run the raw arithmetic gave minus 0.0195 and the manifest recorded 0.000. If you reconcile a residual of 0.000 from `residual_discriminator_performance` and `residual_null_accuracy` and get a negative number, **that is the expected result and not a discrepancy** - the two inputs are recorded unrounded precisely so you can see it. The same floor applies to the "before" score, though it rarely engages there. The manifest also records the raw figure as `residual_bias_unclipped` and whether the floor engaged as `residual_bias_clipped` (and `bias_unclipped / bias_clipped` for the "before" score), so you need not redo the arithmetic to know; see 2.2a.

Your Run Manifest records both baselines separately: `null_accuracy` (the Random Baseline Accuracy, the shuffled-label chance level that is subtracted) and `best_random_accuracy` (the Best Random Accuracy, the scorer's chance level that sets the scale), with `majority_share` beside it for reference.

1.2 Significance figures (the permutation test)

Because the chance level is measured as a distribution over many shuffled runs, Rosa also reports whether the bias it found is statistically significant rather than noise:

Term (manifest field)	What it means
Null accuracy (<code>null_accuracy</code>)	The mean accuracy across the shuffled-label runs (the chance floor above).
Null accuracy, 95th percentile (<code>null_accuracy_p95</code>)	The 95th percentile of that chance distribution. A detector accuracy above this is above what chance produces 95% of the time, i.e. a significance threshold.
Bias p-value (<code>bias_p_value</code>)	The probability of seeing an accuracy this high if there were no real bias. A small value (for example 0.05 or below) means the detected bias is unlikely to be chance.

Term (manifest field)	What it means
Bias detected (bias_detected)	A true/false flag: is the bias statistically significant? When this is false on a bias-removal job, Rosa declines to debias and the job terminates as declined_no_bias with this same evidence.
Permutation count (permutation_count)	How many shuffled-label runs the chance distribution was estimated from. More permutations resolve a finer p-value; the trial default is 24, and evidential runs raise it.
Bias check repeats (bias_check_reps)	How many independent held-out splits the real (unshuffled) measurement was averaged over.
declined_no_bias	A completed decision, with a full evidence record and no debiased file, returned when the detection test does not find the characteristic above what it reaches on shuffled labels, or not by enough to clear the effect floor. It is not a finding that your data is fair. Read the rows per group in the report's Run Evidence section first: a decline with a very small group is more likely a statement about that group's size than about your data. The manifest records bias_p_value, null_accuracy and average_discriminator_performance, and the two thresholds the decision was taken against.
declined_insufficient_support	"Detection not supported." The test was NOT run, because a group of the bias characteristic had fewer than nine rows in the training sample and so could not be scored on a held-out split. Diagnose and Remove Bias (training) jobs both end with it as their status. It carries no dial, no p-value and no "no bias" reading: it says nothing about whether your data is biased, and it is a different outcome from declined_no_bias, which is a test that ran and found nothing above chance. The manifest names the group and its row count (insufficient_support, section 2.2b). Choosing a coarser grouping of genuinely similar categories is a legitimate modelling decision and may clear the line; deleting rows, duplicating them or splitting the file to get past it is not, because it changes what is being measured without giving the rare group any more evidence. What helps a rare group is a training file in which it is a larger share, applied to your full population at inference.
p-value precision (p_value_precision)	The finest p-value the permutation count can resolve ($1 / (\text{permutation_count} + 1)$); a p-value is never reported below this.
Detection alpha (detection_alpha)	The significance threshold the p-value was compared against on this run.
Detection effect floor (detection_effect_floor)	The minimum effect size the run had to clear as well.

Each has a residual (after-debiasing) counterpart on training jobs: residual_null_accuracy, residual_null_accuracy_p95, residual_bias_p_value, residual_bias_detected.

Which way the residual check errs, stated plainly, because it is not symmetric.

The same test decides both readings, so **a test that is cautious about announcing bias is equally cautious about announcing residual bias**. That cuts in opposite directions on the two figures, and only one of them is in your favour.

`bias_detected` true is a **strong** statement: the signal cleared a permutation null and an effect floor. `residual_bias_detected` **false** is a **weaker** statement than it reads. It means this test did not find recoverable signal on this data at this size. It does not mean none is there, and on a characteristic with many groups or a small group it is the reading you should trust least.

Rosa reduces recoverable bias; it does not promise to remove it. A residual reading above zero on some datasets is expected behaviour, not a failure, and the debiased output is not represented as carrying none. **The question worth answering is what your downstream model does** - whether its group gaps narrow and what that costs in accuracy - which you measure on the model itself rather than reading off the upstream dial. Section 6 of the BYO-LLM guide sets out how.

The detection rule is TWO-PART, and both arms must pass. Rosa reports recoverable bias only when:

```
bias_p_value < detection_alpha    AND    effect > detection_effect_floor
```

where

```
effect = Average Discriminator Performance - Null Accuracy
```

If either arm fails, Rosa reports no recoverable bias, and a bias-removal job terminates as `declined_no_bias`. The two arms answer different questions: the p-value asks whether the signal is distinguishable from chance, and the effect floor asks whether it is large enough to be worth acting on. A large but unreliable signal fails the first; a rock-solid but trivial one fails the second.

Both thresholds are recorded on every manifest, accepted runs included, because they are per-instance settings rather than universal constants. The shared-trial defaults are 0.05 and 0.02. Read them from the record rather than assuming.

The effect is not the bias field. The effect above is a plain difference of two accuracies. The bias score in section 1.1 is that same difference **rescaled** by dividing through by (100% - Best Random Accuracy), so that a perfectly predictable attribute reads 100%. **bias is therefore always the larger of the two, on a different axis.** Comparing it against `detection_effect_floor` will give you the wrong answer about why a run declined.

Worked example, from a real declined run:

Quantity	Value
Average Discriminator Performance	0.6312
Null Accuracy	0.6023
effect (the gated quantity)	0.0289 , which is above the 0.02 floor
bias (the rescaled score)	0.0725
<code>bias_p_value</code>	0.12, which is not below the 0.05 alpha
Outcome	<code>declined_no_bias</code>

That run declined **on the p-value arm alone**; its effect arm passed. A reader who compared bias at 0.0725 against the 2% floor would have reached the same verdict for the wrong reason, and would have been wrong about which arm to change to get a different outcome. Raising the permutation count would have helped this run; a smaller effect floor would not.

1.3 Two things the bias figures do NOT tell you

These are the two points readers most often get wrong, and neither is a field you will find in the manifest.

Term	What it means
Bias removed vs utility preserved	Two independent axes, and every figure in section 1 measures only the first. "Bias removed" means Rosa's detector can no longer recover the protected attribute above chance. "Utility preserved" means a model built on the debiased data still predicts your actual outcome well. A run can succeed on the first and disappoint on the second, and the manifest looks identical either way: in controlled testing at small sample sizes the residual read <i>not detected</i> on every run, including runs whose downstream utility was measurably worse. So a clean residual is not evidence that the output is still useful. Establish that separately, on a held-out split, against your own success criteria.
Eligible training rows	The rows that actually enter training after validation and exclusions - not the size of the file you uploaded, and not the row count before you split it. It matters because Rosa's published "no fairness/accuracy trade-off" benchmark is stated only for runs with at least 1,500 eligible training rows . Rosa accepts and runs jobs below that (the acceptance minimum is 1,000 rows), and they complete and reduce measured bias - but at around 1,200 rows the downstream utility result varies enough run to run that a single run is not a reliable read on it. Below the floor, Rosa's own validation script declines to render a verdict rather than reporting a pass or a fail.

A **decline** (`declined_no_bias`) belongs in the same family: it means this test found nothing above chance on this dataset at this sample size. It is a controlled non-result, **not** evidence that your data are fair.

2. Run Manifest fields

The Run Manifest is written once per job, cannot be changed afterwards, and is retained as your permanent evidence record. It never contains your row-level dataset, the debiased output, or the trained model. Job modes are `diagnose`, `remove_bias_training`, and `remove_bias_inference`.

2.1 Top-level fields

Field	Definition
<code>job_id</code>	Unique identifier for the job.
<code>workspace_id</code>	The workspace (API key) the job belongs to.

Field	Definition
timestamp_submitted / timestamp_started / timestamp_completed	Job lifecycle timestamps, recorded in UTC and written in ISO-8601 form with a trailing Z (for example 2026-09-03T14:22:11Z). The PDF report prints the same instants in UTC. The Customer Portal's Jobs page is the one surface that differs: it converts each timestamp to your own browser's time zone and labels it , so a job shown as 15:22 BST in London is the same instant as 14:22Z in the manifest and the report. If you are reconciling a report against the portal, expect that offset - it is a display convenience, not a discrepancy.
stage_timings	Per-stage wall-clock timings for the job (Rosa's own operational metric).
peak_rss_bytes	Peak resident memory used while processing (Rosa's own operational metric).
mode	diagnose, remove_bias_training, or remove_bias_inference.
row_count	Rows processed (after sampling, if any).
sampled_from_row_count	The original uploaded row count when your dataset was down-sampled for training; null if no sampling occurred. Sampling triggers on the row count (datasets over about 10,000 rows are reduced to a seeded representative sample of about 10,000), not on file size, and never applies to inference. The sample is drawn near-proportionally, so each group keeps roughly its share of the file: a bigger file does not give a small group more rows in training.
raw_column_count	Number of raw input columns.
pre_encoding_feature_count	Number of input features before one-hot encoding.
post_encoding_width	Number of features after one-hot encoding - the value the encoding-dimension cap acts on.
bias_columns	The protected attribute(s) you submitted.
declared_columns	The column roles you set: {bias_columns, ignore_columns, cat_columns}. Makes the manifest self-describing about how the run was scoped.
input_hash	SHA-256 of your raw input CSV bytes, computed before any processing, formatted "sha256:<64 hex>". Lets you recompute the hash from your original file and prove the exact file was processed unaltered.
schema_hash	SHA-256 of your column names.
config_hash	SHA-256 of the job configuration.
container_digest	Identifier of the Rosa build that ran the job.
rosa_version	The Rosa version that ran the job.
model_format_version	Format version of the trained model; null for diagnose.

Field	Definition
output_artifact_hashes	A SHA-256 hash for every file the run produced - both the files you can download (the report and the debiased CSV) and the internal files Rosa manages on your behalf. You can check any file you download against its hash here to confirm it has not been altered.
idempotency_key	An optional key you supply so a retried submission returns the same job instead of creating a duplicate; null if not supplied.
job_status	How the job ended. There are six terminal states: complete, failed, cancelled, interrupted, declined_no_bias (see section 1.2 - Rosa found no recoverable bias and declined to debias, which is a valid decision rather than a failure) and declined_insufficient_support (section 1.2 - the test could not be run because a group had fewer than nine rows in the training sample; also a decision, not a failure, and never a "no bias" finding). A Run Manifest is written for all six , so you have an evidence record whatever the outcome.
error	The reason a job failed, when applicable.
engine_attempts	The number of processing attempts the job took; 1 for almost all jobs.
bias_summary	The bias measurement block (see 2.2).
debiasing_adjustments	Per-column debiasing detail, diagnose jobs only (see 2.3).
feature_associations	The proxy view (see 2.3).
feature_associations_post	Proxy view recomputed on the debiased output, training jobs only (see 2.3).
column_stats	Per-column statistics (see 2.4).
warnings	Advisory notes recorded when a job ran with a caveat but was NOT rejected (inference jobs). Each entry names what was flagged (and the column where relevant): a training category absent from the inference data (the model encodes it as an all-zero column, so rows that are present are unaffected), a numeric column whose values drifted more than 10% outside the training range, a small inference population (the whole file is below the recommended minimum, so debiasing quality is reduced because the distribution mapping needs a representative sample), or that your inference file's columns arrived in a different order from the training data and Rosa reordered them to match before processing (informational - the values are unchanged and no action is needed). An empty list means no advisories. Fatal incompatibilities - a new/unseen category, or a changed column set - are rejected before the job runs and never appear here.
category_counts	Per-column category cardinality (aggregate counts only, never the values); null for inference.

Field	Definition
unknown_category_policy	How the engine treats categories at inference: {unseen_category: reject, missing_category: zero_pad} - a new/unseen category is rejected before the job runs, while a training category absent from the inference data is encoded as an all-zero column so present rows are unaffected.
instance_profile	The serving box's own hardware - {instance_type, vcpu, ram_gb} - so a width-scaling run is plottable from the manifests alone and per-run compute cost is attributable. Rosa's own operational metric, never your data; fields are null off the fleet (local dev).
manifest_schema_version	Version of the Run Manifest schema itself.
training_job_id	On an inference run, the job whose trained model was applied - the run's declared lineage; null on diagnose and training runs. It is recorded from the same reference Rosa used to load the model, so the manifest names the model that actually produced the output rather than one worked out afterwards. Matching row identifiers alone cannot show this: two separate runs over the same rows produce the same identifiers, so a declared parent is what ties an inference output to one particular training run. Runs from before this field existed do not carry it, and its absence means "not recorded", not "no parent".

2.2 The bias_summary block

Field	Definition
best_random_accuracy	The chance level of the scorer in use, the scale term of section 1: 1/K for balanced accuracy over K groups. On manifests that carry no bias_scorer field it was the most-common-group share. The same number is recorded as trivial_baseline in the measurement-regime fields below.
average_discriminator_performance (+_std)	The detector's score on the original data (balanced accuracy on the fixed measurement; bias_scorer names it), with its standard deviation.
residual_discriminator_performance (+_std)	The detector's score on the debiased data (training jobs).
bias	The "before" bias score.
residual_bias	The "after" bias score (training jobs).
null_accuracy (+_p95)	The shuffled-label chance level and its 95th percentile.
bias_p_value / bias_detected	Significance of the "before" measurement (see 1.2).
permutation_count, bias_check_reps, p_value_precision	The permutation test's parameters: shuffled-label runs, held-out measurement repeats, and the finest resolvable p-value (see 1.2).

Field	Definition
detection_alpha, detection_effect_floor	The two thresholds the detection gate compared against on this run. Present on every run, not only declined ones. Per-instance settings, not universal constants. The effect they gate is average_discriminator_performance minus null_accuracy, not the bias field - see 1.2.
residual_null_accuracy (+ _p95), residual_bias_p_value, residual_bias_detected	The same significance figures for the "after" measurement (training jobs).

2.2a The measurement-regime fields (manifest schema 1.5 and later)

These fourteen fields record **how** the bias figures on the manifest were measured, so a manifest can be read on its own terms years later and two manifests can be told apart when the measurement changed between them. They are the Run Manifest page's "Measurement regime" panel and the one-line "Measurement" row on the PDF report. None of them is a new measurement: the first seven describe the instrument, the last seven show the working behind the two Bias Score figures.

Field	Definition
bias_scorer	The score the detector is measured on. balanced_accuracy (the mean of the per-group recall rates, every group counting equally) on the fixed measurement; raw_accuracy (the share of rows guessed right, so the largest group dominates) on runs measured before it.
class_weighting	Whether the detector was trained with every group weighted equally, so a rare group contributes to its training as much as a dominant one. yes on the fixed measurement.
scorer_aware_baseline	Whether the Best Random Accuracy was set for the scorer actually in use (1/K for balanced accuracy) rather than the most-common-group share. yes on the fixed measurement. When no, read the dial with care: it was scaled against a chance level that belongs to a different score.
trivial_baseline	The Best Random Accuracy as a fraction: the chance level of the scorer in use, 1/K for balanced accuracy over K groups. The same value as best_random_accuracy, recorded again beside the scorer it belongs to.
trivial_baseline_scorer	The scorer that trivial_baseline is the chance level for. It should equal bias_scorer; if the two differ, the dial was divided by a scale that belongs to a different score than the one it was measured on, and the figures should not be compared with a normal run.
majority_share	The most common group's share of the rows. Recorded for reference only. It is what the Best Random Accuracy was on manifests measured with plain accuracy; it plays no part in the score on the fixed measurement.
dial_denominator	1 minus trivial_baseline: the room a perfect predictor has above chance, which the excess is divided by. 0.5 for two groups, 0.8 for five.

Field	Definition
dial_formula	The rule, written out: $\max(0, \text{score} - \text{own_null}) / (1 - \text{trivial_baseline})$, clipped to $[0, 1]$. "score" is the detector's score, "own_null" is the Random Baseline Accuracy measured for that same column (before or after), and the clip is the floor at zero described in section 1.1. This field states the rule; the _clipped fields below say whether it fired.
bias_excess	The numerator of the "before" score before division: <code>average_discriminator_performance</code> minus <code>null_accuracy</code> . This is the quantity the detection gate's effect floor is compared against. It can be negative.
bias_unclipped	<code>bias_excess</code> divided by <code>dial_denominator</code> , before the clip. Equal to <code>bias</code> unless the clip fired.
bias_clipped	yes if <code>bias</code> differs from <code>bias_unclipped</code> because the clip fired (a negative unclipped value floored to zero, or a value above one capped at one); no if the printed score is the raw arithmetic. On a "before" measurement this is almost always no.
residual_bias_excess	The same numerator for the "after" measurement (training jobs): <code>residual_discriminator_performance</code> minus <code>residual_null_accuracy</code> . Often slightly negative on well-debiased data, which is why the next two fields exist.
residual_bias_unclipped	<code>residual_bias_excess</code> divided by <code>dial_denominator</code> , before the clip. A negative value here with <code>residual_bias</code> printed as <code>0.000</code> is the expected pattern described in section 1.1, not a discrepancy.
residual_bias_clipped	yes when the printed <code>residual_bias</code> is the floored zero rather than the raw arithmetic; no otherwise.

Reading them together: `bias` equals `bias_unclipped` whenever `bias_clipped` is no, and `bias_unclipped` equals `bias_excess / dial_denominator` always. The same three identities hold for the residual figures. If a manifest breaks one of them, the manifest has been altered.

2.2b Per-group figures (manifest schema 1.6 and later)

The dial is one number for every group together (section 1.1). From schema 1.6 the manifest also carries, inside `bias_summary`, a `per_group` array with one entry per group of the bias characteristic, so you can see which group moved the average. **These figures are exploratory.** They are averages over the same held-out fits as the dial, they are not tested for significance on their own, and a small group's figures are noisy in proportion to its size; they support a reading of the dial and are not a finding about any group by itself. The block-level field `per_group_status` reads `exploratory` to say so on the record.

Field	Definition
group	The group's value as it appears in your data.
rows_in_sample	The number of rows of this group in the training sample (at most about 10,000 rows, drawn roughly in proportion; section 1).

Field	Definition
min_rows_scored	The smallest number of rows of this group in any held-out split the measurement was scored on. Small values here are the reason a group's figures are noisy. Read it as a post-run check on the run in hand, never as a threshold - see the note below this table.
recall	Averaged over the measurement's held-out fits, the share of this group's rows the detector identified correctly. The dial's score is the mean of this figure across all groups.
false_positive_rate	Averaged the same way, the share of other groups' rows the detector wrongly assigned to this group.
informedness	recall minus false_positive_rate: how much better than guessing the detector is at picking this group out. Zero is chance; one is perfect.
residual_recall, residual_false_positive_rate, residual_informedness	The same three figures measured on the debiased data (training jobs only; null on a diagnose).
insufficient_support	Present, as a small object, only when the detection test was not run because a group had fewer than nine rows in the training sample: {group, rows_in_sample, minimum_rows: 9}. Beside it detection_supported reads false. See declined_insufficient_support in section 1.2. On every other run detection_supported is true and this field is null.

A warning about min_rows_scored, because the obvious way to use it is wrong.

It is tempting to read a number off one run and carry it to the next as a rule of thumb: *"we detected with twenty held-out rows, so twenty is enough."* **It is not, and the size of the error is larger than it looks.** The number of rows a group needs depends on **how many groups the characteristic has**, and it moves by an order of magnitude across the range customers actually use.

In our own measurements, a **three-group** characteristic detected reliably with the smallest group scored on **five to seven** held-out rows, and a **five-group** one detected on **two to three**. An **eight-group** characteristic on the same kind of data needed **twenty-three or more**, and declined when it had fifteen to nineteen.

So min_rows_scored describes the run that produced it and nothing else. Use it to understand why a particular group's figures are noisy, and to compare runs **at the same group count**. Do not use it to predict whether a different characteristic, or the same one binned differently, will detect.

2.3 Feature-level fields

Field	Definition
debiasing_adjustments	For a Diagnose job, each column's share of the total adjustment Rosa would apply when debiasing (the values sum to 1.0). This is not each column's correlation with the protected attribute: Rosa adjusts all features together, so this is not a ranking of which features carry the bias. To see which features track the protected attribute, use feature_associations.

Field	Definition
feature_associations	The proxy view: a ranked list giving each feature's own association with the protected attribute on a 0 to 1 scale (point-biserial, correlation-ratio, or Cramer's V depending on the column type). It is a transparent single-feature check you can reproduce yourself. It complements, and does not replace, Rosa's full analysis, which also detects and removes non-linear and combined proxies that a single-feature score cannot see. Present on diagnose and training jobs.
feature_associations_post	The same association view recomputed on the debiased output, so you can see the proxy signal collapse after debiasing. Training jobs only.

2.4 The column_stats block (per column)

Field	Definition
column_name	The column name.
datatype	The column's data type.
missing_values	Count of missing values.
categorical	Whether the column is treated as categorical.
num_categories / categories / counts	For categorical columns: the number of distinct values, the values themselves, and their row counts.
bias_contribution	The column's share of the debiasing adjustment (see debiasing_adjustments).
min / max / std_dev	For numeric columns: summary statistics.
min_change_pct / max_change_pct / std_dev_change_pct / num_categories_change_pct	The percentage change in each statistic between your original and debiased data, evidencing that Rosa preserves each column's distribution bit-for-bit, to float64 CSV precision. Read the debiased CSV with a round-trip-capable parser to see the exact bits; a default parser may show one or two ULPs of its own rounding.

Do not submit direct identifiers (names, email addresses, customer IDs) at all: remove or pseudonymise them before you upload. `ignore_columns` is a modelling control, not a privacy boundary. It keeps a column out of the model and out of the manifest's category enumeration, which matters because `column_stats` can include category values from your data and the manifest is retained permanently, but it does not prevent the column being uploaded or processed. If an identifier is exceptionally permitted under your contract, ignoring it is the right setting, and it is not a substitute for leaving it out.

3. PDF report sections

The Diagnose report and the Training report share most sections; differences are noted.

- **Processing Summary.** The run's facts: dataset name and size, whether it was sampled, feature counts before and after one-hot encoding, start and end times, and the input and output files.

- **Bias Summary** (Diagnose) / **Bias Removal Summary** (Training). The headline table (see section 1), followed by the dial or dials and then the reading in plain language. The table carries the finding, and the Diagnose and Training reports print it in the same order so the two can be read against each other: the **Verdict** first (whether the test found the attribute above chance), then the **Bias Score**, the **Average Discriminator Performance** with its standard deviation, the **Random Baseline Accuracy** the score is measured against, the **Permutation p-value** that supports the verdict, and the **Best Random Accuracy** that sets the scale. The Bias Characteristic is named with the table. The verdict leads deliberately: whether the finding is real is the first thing to read, and the size of it is the second. **Those figures are everything the score is calculated from**, so any printed score can be checked against them using the formula in section 1.1. The settings the test ran at - how many shuffled-label runs were used, how many measurement repeats, and the resolution that implies - are stated in the paragraphs beneath the dials rather than above them, because they describe the instrument rather than the result. **On a Training report that table has two columns, "Before debiasing" and "Residual", and every figure appears in both** - the verdict, the bias score, the detector accuracy with its standard deviation, the chance floor and the permutation p-value. Each column is a complete, self-contained reading of one measurement, which is why the chance floor differs between them. A Diagnose report has a single measurement and so a single column. A dial (gauge) renders the score visually - **the Diagnose report has one dial, the Training report a pair, before and after, side by side**, each captioned with its own verdict.
- **Feature Associations** (Diagnose). A ranked table of each feature's own association with the protected attribute, a reproducible check that complements Rosa's full analysis.
- **Debiasing Adjustment by Feature** (Diagnose). A ranked table of how much Rosa would adjust each feature during debiasing. This is not a ranking of which features predict the protected attribute.
- **Run Evidence**. Makes the report a standalone audit artifact, in five parts: (1) dataset and sampling disclosure; (2) integrity and provenance hashes, identical to the manifest, so the report and the manifest agree byte for byte; (3) the declared column scope; (4) protected-group representation (counts and shares per group); (5) a scope-and-limitations statement.
- **Dataset Statistics**. Per-column statistics, including the change percentages that show your data's distribution is preserved bit-for-bit, to float64 CSV precision (read the debiased CSV with a round-trip-capable parser to see the exact bits; a default parser may show one or two ULPs of its own rounding).
- **Appendix**. Plain-English definitions of the terms used in the report.

The PDF report is meant to be read by a person, so any section that lists individual features (Feature Associations, Debiasing Adjustment by Feature, and the per-column Dataset Statistics) shows at most the 100 most important features for that section, ranked by importance within the section, to keep the report a manageable length. The Run Manifest is machine-readable, so it is never capped: it records every feature. Where a section reaches the limit, the report says so (for example "top 100 of 240") and points you to the manifest for the complete list.

4. The same term across your report, manifest, and the portal

Concept	Report label	Manifest field	Portal label
Protected attribute	Bias Characteristic	bias_columns / declared_columns.bias_columns	Bias characteristic
Chance floor	Random Baseline Accuracy	null_accuracy before debiasing, residual_null_accuracy after	Random Baseline Accuracy

Concept	Report label	Manifest field	Portal label
Detector accuracy (before)	Average Discriminator Performance, in the Before debiasing column	average_discriminator_performance (+_std)	Average Discriminator Performance, in the Before debiasing column
Detector accuracy (after)	Average Discriminator Performance, in the Residual column	residual_discriminator_performance (+_std)	Average Discriminator Performance, in the Residual column
Before bias score	Bias Score	bias	Bias Score
After bias score	Residual Bias	residual_bias	Residual bias
Proxy strength per feature	Feature Associations	feature_associations / feature_associations_post	Feature Associations
Debiasing adjustment per feature	Debiasing Adjustment by Feature	debiasing_adjustments	Debiasing adjustment by feature
Verdict on a measurement	Bias Verdict / Residual Verdict	bias_detected / residual_bias_detected	Verdict
Significance of a measurement	Bias Permutation p-value / Residual Permutation p-value	bias_p_value / residual_bias_p_value	Permutation p-value
Resolution of the p-value	shown in brackets beside the p-value	p_value_precision	shown as "resolution" beside the p-value
Shuffled-label runs	Permutations	permutation_count	Permutations
Held-out repeats	Measurement Repeats	bias_check_reps	Measurement repeats
Scale term	Best Random Accuracy, shown with the dial figures	best_random_accuracy (also trivial_baseline)	Best random accuracy, shown with the dial figures
Per-group evidence (exploratory)	Per-group evidence table	bias_summary.per_group (2.2b)	Per-group evidence table under each dial explanation and on the Run Manifest page
Detection not supported	Detection not supported report	job_status declined_insufficient_support (diagnose and training alike); bias_summary.detection_supported false and insufficient_support (both modes)	"Detection not supported" badge and finding block

Concept	Report label	Manifest field	Portal label
How the figures were measured	Measurement row	the fourteen measurement-regime fields of 2.2a (bias_scorer onwards)	Measurement regime panel on the Run Manifest page; one line under the dial table on the Outputs page
Inference advisories	-	warnings	Warnings

Why the chance floor appears twice. "Random Baseline Accuracy" is a role, not a single field: it is always the chance floor for the measurement printed beside it, and it is re-measured on the debiased data. A Diagnose report and the Diagnose screen carry one measurement, so there is one figure. **A Training report and the Training screen print the before and the residual measurements side by side, each with its own chance floor -** null_accuracy in the "Before debiasing" column and residual_null_accuracy in the "Residual" column. **Read down a column, not across the table:** every printed score reconciles from the two figures above it in the same column, rescaled by the Best random accuracy.