Package 'AI4OfficialStats'

September 22, 2026

Type Package

Title Audit Statistical Fidelity of AI-Mediated Official Statistics

Version 0.1.0

Description Provides deterministic tools for auditing whether artificial intelligence systems preserve the numerical, semantic, contextual, temporal, geographic, unit, provenance, revision, transformation, and uncertainty properties of official statistics. Structured reference statistics and machine-generated claims can be compared using non-compensatory critical-error rules, weakest-link and geometric fidelity summaries, provenance graphs, and portable SHA-256 proof bundles. The package provides bounded connectors for official statistical services and an extensible HTTPS JSON API registry. Prompt perturbation, statistical red-team generation, minimal-pair tests, and benchmark data support reproducible evaluation of generative, retrieval-augmented, and agentic statistical systems. An embedded alignment layer maps claim-level controls to relevant activities of the Generic Statistical Business Process Model (GSBPM) 5.2, including Analyse, Disseminate, Evaluate, Quality Management, and Metadata Management. The GSBPM alignment follows United Nations Economic Commission for Europe (2025) ``Generic Statistical Business Process Model (GSBPM) version 5.2" <https://unece.org/statistics/gsbpm-v5.2>.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.1.0)

Imports digest, jsonlite, stats, utils

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

RoxygenNote 7.3.2

NeedsCompilation no

Author Hossein Hassani [aut],
Steve MacFeely [aut],
Leila Marvian Mashhad [aut, cre]

Maintainer Leila Marvian Mashhad <leila.marveian@gmail.com>

Repository CRAN

Date/Publication 2026-09-22 06:20:17 UTC

Contents

`AI4OfficialStats-package`

Audit Statistical Fidelity of AI-Mediated Official Statistics

---

Description

Tools for representing official-statistics references and AI-generated claims, auditing multiple dimensions of statistical fidelity, applying critical-error rules, creating proof bundles, tracing provenance, retrieving bounded official data from supported providers, and generating robustness and red-team tests.

Details

The package is model-provider neutral and performs no mandatory network calls. Live provider access occurs only through explicit fetch functions. It can be used with outputs from generative AI, retrieval-augmented generation, or agentic systems after those outputs are represented as `stat_claim` objects. It complements process-level official-statistics quality frameworks such as GSBPM 5.2 by providing an executable claim-level fidelity-control layer rather than a replacement statistical production model. The package also provides offline GSBPM mapping and quality-report helpers via `gsbpm52_map()`, `gsbpm_align()`, and `gsbpm_quality_report()`.

Author(s)

Hossein Hassani, Steve MacFeely, and Leila Marvian Mashhad

`as_stat_reference` *Convert Provider Data to an Official Statistical Reference*

Description

Converts one canonical provider row into a structured official-statistics reference while preserving the source row and retrieval provenance.

Usage

```
as_stat_reference(x, row = 1L, indicator = NULL, geo = NULL, time = NULL,
  unit = NULL, source = NULL, dataset = NULL, population = NULL,
  adjustment = NULL, frequency = NULL, methodology = NULL, revision = NULL,
  quality_flag = NULL, uncertainty = NULL, transformation = NULL)
```

Arguments

`x`	A `stat_provider_data` object.
`row`	Selected observation row.
`indicator`, `geo`, `time`, `unit`, `source`, `dataset`	Optional overrides for canonical fields.
`population`, `adjustment`, `frequency`, `methodology`, `revision`	Optional official-statistics context.
`quality_flag`	Optional quality flag override.
`uncertainty`	Optional uncertainty list.
`transformation`	Optional expected transformation.

Value

A `stat_reference` object.

audit_components *Component-Level Statistical Fidelity Audits*

Description

Compares individual semantic, numerical, temporal, geographic, unit, provenance, revision, and uncertainty dimensions of an AI-mediated claim against an official-statistics reference.

Usage

```
semantic_audit(reference, claim, aliases = NULL, minimum_similarity = 0.65)
numeric_audit(reference, claim, abs_tol = 1e-08, rel_tol = 1e-06)
temporal_audit(reference, claim)
geo_audit(reference, claim, aliases = NULL)
unit_audit(reference, claim)
source_audit(reference, claim, missing_score = 0.4)
revision_audit(reference, claim)
uncertainty_audit(reference, claim)
```

Arguments

reference	A stat_reference object.
claim	A stat_claim object.
aliases	Optional accepted aliases for the indicator or geography.
minimum_similarity	Indicator token-similarity threshold below which mismatch is critical.
abs_tol	Absolute numerical tolerance.
rel_tol	Relative numerical tolerance.
missing_score	Score assigned when source attribution is omitted.

Value

A stat_audit_component object containing score, status, critical flag, error code, expected value, observed value, and details.

Examples

```
ref <- stat_reference(10.4, "unemployment rate", "Spain", "2025",
                      "percent", "Eurostat")
clm <- stat_claim(10.4, "unemployment rate", "Spain", "2025",
                  "%", "Eurostat")
numeric_audit(ref, clm)
unit_audit(ref, clm)
```

audit_stat_ai *Audit and Summarize Statistical Fidelity*

Description

Runs the full ten-component audit and applies geometric composite scoring plus non-compensatory critical-error rules. The package also records the audit's intended alignment with relevant GSBPM 5.2 Analyse, Quality Management, and Metadata Management activities.

Usage

```
audit_stat_ai(reference, claim, indicator_aliases = NULL,
  geo_aliases = NULL, numeric_abs_tol = 1e-08,
  numeric_rel_tol = 1e-06, weights = NULL,
  pass_threshold = 0.90, weak_threshold = 0.75)

fidelity(components, weights = NULL, pass_threshold = 0.90,
  weak_threshold = 0.75)

audit_batch(references, claims, ids = NULL, ...)

weakest_link(x)
```

Arguments

reference	A stat_reference.
claim	A stat_claim.
indicator_aliases	Optional accepted indicator aliases.
geo_aliases	Optional accepted geography aliases.
numeric_abs_tol	Absolute numerical tolerance.
numeric_rel_tol	Relative numerical tolerance.
weights	Optional named non-negative component weights.
pass_threshold	Composite fidelity threshold for PASS.
weak_threshold	Minimum weakest-link fidelity for PASS.
components	Named list of component audit objects, or a complete audit.
x	Audit, fidelity result, or numeric score vector.
references	List of official-statistics reference objects.
claims	List of claim objects of the same length.
ids	Optional identifiers for batch pairs.
...	Additional arguments passed to audit_stat_ai().

Value

`audit_stat_ai()` returns a `stat_fidelity_audit` with ten component audits, the certification decision, and an embedded GSBPM 5.2 alignment record; `audit_batch()` returns a batch summary data frame; `fidelity()` returns summary scores; `weakest_link()` returns the lowest component score.

Examples

```
ref <- stat_reference(10.4, "unemployment rate", "Spain", "2025",
                      "percent", "Eurostat")
clm <- stat_claim(10.4, "unemployment rate", "Spain", "2025",
                  "%", "Eurostat")
aud <- audit_stat_ai(ref, clm)
aud
weakest_link(aud)
```

---

benchmark_tools *Official Statistics AI Benchmark Tools*

Description

Creates adversarial statistical minimal pairs and loads the offline OfficialStatAI-Bench starter cases.

Usage

```
statistical_minimal_pairs(reference)
benchmark_from_reference(reference)
officialstat_benchmark()
```

Arguments

`reference`	A `stat_reference` object.

Value

A data frame for minimal pairs or the packaged starter benchmark; `benchmark_from_reference()` returns a list.

Examples

```
ref <- stat_reference(5.1, "unemployment rate", "Exampleland", "2025", "percent", "NSO")
statistical_minimal_pairs(ref)
head(officialstat_benchmark())
```

context_audit *Audit Statistical Context and Revision Propagation*

Description

Audits dataset and methodological context and measures revision propagation lag, including right-censored lag when an AI system has not yet reflected a revision.

Usage

```
context_audit(reference, claim, missing_score = 0.5,
  methodology_min_similarity = 0.35)
revision_propagation_lag(official_revised_at, ai_reflected_at = NULL,
  observed_at = Sys.time())
```

Arguments

reference	A stat_reference object.
claim	A stat_claim object.
missing_score	Penalty for omitted required context.
methodology_min_similarity	Similarity threshold for methodology mismatch.
official_revised_at	Official revision date/time.
ai_reflected_at	Optional date/time when AI first reflected the revision.
observed_at	Observation date/time for a not-yet-reflected revision.

Value

context_audit() returns a component audit. revision_propagation_lag() returns a one-row data frame.

extract_stat_claims *Extract Candidate Statistical Claims from Text*

Description

Provides a conservative built-in numerical extractor and a parser interface for advanced NLP or LLM-based claim extraction without tying the package to any model vendor.

Usage

```
extract_stat_claims(text, hints = list(), parser = NULL)
```

Arguments

text	Character vector containing an AI answer.
hints	Optional named list of indicator, geography, time, unit, and source metadata.
parser	Optional external parser returning one or more structured claims.

Value

A list of stat_claim objects.

Examples

```
extract_stat_claims("The rate was 10.4% in 2025.",
  hints = list(indicator = "rate", geo = "Example", time = "2025"))
```

---

gsbpm52_map *Align AI4OfficialStats with GSBPM 5.2*

Description

Offline helpers that map AI4OfficialStats's AI-mediated claim-level quality controls to relevant phases, sub-processes and overarching activities in the Generic Statistical Business Process Model (GSBPM) version 5.2.

Usage

```
gsbpm52_map(relationship = c("all", "direct", "supporting"))

gsbpm_align(x)

gsbpm_quality_report(audit)
```

Arguments

relationship	Optional relationship filter: "all", "direct", or "supporting".
x	A fidelity component name, an exported function name, or a stat_fidelity_audit object.
audit	A stat_fidelity_audit returned by audit_stat_ai().

Details

The mapping is intended to make the relationship to GSBPM 5.2 explicit and auditable. It does not claim that AI4OfficialStats replaces the GSBPM or that using the package constitutes official UNECE certification of GSBPM conformance. The principal direct relationships are with Analyse 6.2 (Validate outputs), 6.5 (Finalise outputs), Disseminate 7.2/7.5, Evaluate 8.1/8.2, and the overarching Quality and Metadata Management activities.

Value

`gsbpm52_map()` and `gsbpm_align()` return data frames. `gsbpm_quality_report()` returns a `stat_gsbpm_report` object containing component-level evidence, the audit decision, and mapped GSBPM activities.

Examples

```
gsbpm52_map("direct")
gsbpm_align("semantic")
gsbpm_align("audit_stat_ai")
```

---

official_data_connectors

Official-Statistics API Connectors

Description

Build URLs and retrieve bounded data from Eurostat, the World Bank Indicators API, and the OECD Data Explorer SDMX API.

Usage

```
eurostat_url(dataset, filters = list(), lang = "EN")
fetch_eurostat(dataset, filters = list(), lang = "EN", cache = TRUE,
  cache_dir = NULL, max_age = 3600, timeout = 30, max_cells = 1e6)
worldbank_url(indicator, country = "all", start = NULL, end = NULL,
  source = NULL, page = 1L, per_page = 1000L, metadata = FALSE)
fetch_worldbank(indicator, country = "all", start = NULL, end = NULL,
  source = NULL, per_page = 1000L, include_metadata = TRUE, cache = TRUE,
  cache_dir = NULL, max_age = 3600, timeout = 30)
oecd_url(agency, dataset, selection = "all", version = NULL,
  start_period = NULL, end_period = NULL, labels = TRUE)
fetch_oecd(agency, dataset, selection = "all", version = NULL,
  start_period = NULL, end_period = NULL, labels = TRUE, cache = TRUE,
  cache_dir = NULL, max_age = 3600, timeout = 30)
```

Arguments

`dataset`	Provider dataset identifier.
`filters`	Named Eurostat filter list.
`lang`	Eurostat response language.
`cache`	Whether to cache downloads.
`cache_dir`	Optional cache directory; by default a temporary session directory is used.
`max_age`	Maximum cache age in seconds.
`timeout`	Download timeout in seconds.

max_cells	Maximum Eurostat JSON-stat cube size accepted.
indicator	World Bank indicator code.
country	World Bank country/economy selection.
start, end	Optional World Bank period bounds.
source	Optional World Bank source identifier.
page	World Bank API page.
per_page	World Bank API page size.
metadata	Build the World Bank indicator metadata endpoint.
include_metadata	Retrieve World Bank indicator metadata.
agency	OECD SDMX agency identifier.
selection	OECD SDMX dimension selection.
version	Optional OECD dataset structure version.
start_period, end_period	Optional OECD SDMX period bounds.
labels	Request OECD CSV labels together with codes.

Details

Network access occurs only after an explicit call to a fetch function. Tests use local fixtures rather than provider endpoints.

Value

URL builders return a character string. Fetchers return a `stat_provider_data` data frame.

Examples

```
eurostat_url("demo_pjan", list(geo = "ES", time = "2025"))
worldbank_url("SP.POP.TOTL", country = "FRA", start = 2020, end = 2025)
oecd_url("OECD.SDD.STES", "DSD_STES@DF_CLI", start_period = "2025")

wb <- fetch_worldbank("SP.POP.TOTL", country = "FRA", start = 2025, end = 2025)
```

---

proof_io *Write, Read, and Verify Statistical Proof Bundles*

Description

Serializes Proof-Carrying Statistical Answers as canonical JSON and verifies their SHA-256 payload fingerprint.

Usage

```
write_proof(proof, path, pretty = TRUE)
read_proof(path)
verify_proof(proof)
```

Arguments

proof	A AI4OfficialStats proof bundle or compatible proof list.
path	JSON file path.
pretty	Pretty-print JSON.

Value

Writing returns the path invisibly; reading returns a proof list; verification returns hash validity diagnostics.

---

provenance_tools *Provenance Graphs and Proof-Carrying Statistical Answers*

Description

Creates traceable relationships between source statistics, transformations, and AI claims, and packages audit evidence into a reproducible proof object.

Usage

```
claim_graph(reference, claim)
trace_claim(graph)
proof_bundle(audit)
certify_claim(audit, require_pass = FALSE)
```

Arguments

reference	A stat_reference.
claim	A stat_claim.
graph	A graph returned by claim_graph().
audit	A stat_fidelity_audit.
require_pass	If true, fail instead of issuing a proof for WARN/FAIL audits.

Value

A lightweight graph list, or a stat_proof carrying a SHA-256 fingerprint and audit status.

Examples

```
ref <- stat_reference(10.4, "unemployment rate", "Spain", "2025",
                      "percent", "Eurostat")
clm <- stat_claim(10.4, "unemployment rate", "Spain", "2025",
                  "%", "Eurostat")
aud <- audit_stat_ai(ref, clm)
g <- claim_graph(ref, clm)
trace_claim(g)
certify_claim(aud)
```

---

search_official_stats *Search multiple official-statistics catalogues*

Description

Runs explicit API/catalogue searches across supported official providers and combines ranked metadata results without treating unavailable providers as ground-truth failures.

Usage

```
search_official_stats(query,
  providers = c("worldbank", "ons", "unsd_sdg"), limit = 10L,
  strict = FALSE, cache = TRUE, cache_dir = NULL, max_age = 3600,
  timeout = 30)
```

Arguments

query	Free-text search term.
providers	Subset of "worldbank", "ons", and "unsd_sdg".
limit	Maximum matches retained per provider.
strict	Whether a provider failure should stop the whole search.
cache	Whether responses should be cached.
cache_dir	Optional user-selected cache directory.
max_age	Maximum cache age in seconds.
timeout	Request timeout in seconds.

Value

A stat_search_results data frame. When strict = FALSE, provider errors are preserved in the provider_errors attribute.

Examples

```
search_official_stats(
  "unemployment",
  limit = 5
)
```

search_worldbank *Additional official-statistics API connectors and search tools*

Description

Adds metadata discovery and bounded retrieval for official World Bank, UNSD SDG, UK ONS, and U.S. BLS services. The IMF helper builds URLs for the current IMF SDMX 2.1 and 3.0 services while leaving dataset-specific resource paths to the caller.

Usage

```
search_worldbank(query, source = NULL, limit = 20L, per_page = 1000L,
  max_pages = 20L, cache = TRUE, cache_dir = NULL, max_age = 86400,
  timeout = 30)

ons_search_url(query, content_type = "dataset", limit = 20L, offset = 0L,
  filters = list())

search_ons(query, content_type = "dataset", limit = 20L, offset = 0L,
  filters = list(), cache = TRUE, cache_dir = NULL, max_age = 3600,
  timeout = 30)

unsd_sdg_series_url(all_releases = FALSE)

unsd_sdg_data_url(series_code, page = 1L, page_size = 1000L,
  release_code = NULL)

search_unsd_sdg(query, limit = 20L, all_releases = FALSE, cache = TRUE,
  cache_dir = NULL, max_age = 86400, timeout = 30)

fetch_unsd_sdg(series_code, page = 1L, page_size = 1000L, max_pages = 1L,
  release_code = NULL, cache = TRUE, cache_dir = NULL, max_age = 3600,
  timeout = 30)

bls_url(series_id, latest = FALSE)

fetch_bls(series_id, latest = FALSE, cache = TRUE, cache_dir = NULL,
  max_age = 3600, timeout = 30)

imf_api_url(path, version = c("3.0", "2.1"), query = list())
```

Arguments

query	Free-text search term.
source	Optional World Bank source/database identifier.
limit	Maximum returned matches.

per_page, max_pages	Safety bounds for catalogue or data pagination.
cache	Whether responses should be cached.
cache_dir	Optional user-selected cache directory.
max_age	Maximum cache age in seconds.
timeout	Request timeout in seconds.
content_type	ONS content type, for example "dataset".
offset	ONS search offset.
filters	Additional named ONS search parameters.
all_releases	Whether the UNSD SDG catalogue should include all releases.
series_code	UNSD SDG series code.
page, page_size	UNSD SDG pagination controls.
release_code	Optional UNSD SDG release identifier.
series_id	BLS time-series identifier.
latest	Whether to request the latest BLS observation only.
path	IMF SDMX resource path relative to the selected version endpoint.
version	IMF SDMX service version.

Value

URL builders return character URLs. Search functions return stat_search_results. Data retrieval functions return stat_provider_data.

Examples

```
ons_search_url("unemployment", content_type = "dataset")
unsd_sdg_series_url()
unsd_sdg_data_url("SI_COV_SOCINS", page_size = 100)
bls_url("LNS14000000", latest = TRUE)
imf_api_url("dataflow/IMF/all/latest", version = "3.0")


search_worldbank("unemployment rate")
search_ons("unemployment")
search_unsd_sdg("social protection")
fetch_bls("LNS14000000", latest = TRUE)
```

stat_api_key *Extensible REST API support and session-only API keys*

Description

Provides explicit GET-only access to additional official JSON APIs without scraping HTML. Custom registrations are session-local and never store secrets. API keys can be supplied directly or through environment variables named STATFIDELITY_<PROVIDER>_API_KEY.

Usage

```
stat_api_key(provider, key = NULL, required = FALSE)

set_stat_api_key(provider, key)

clear_stat_api_key(provider)

api_get_json(base_url, path = NULL, query = list(), provider = "custom",
  api_key = NULL, api_key_name = NULL, cache = TRUE, cache_dir = NULL,
  max_age = 3600, timeout = 30)

register_stat_api(name, base_url, search_path = NULL, query_param = "q",
  api_key_name = NULL, provider_label = name, overwrite = FALSE)

official_stat_providers(include_custom = TRUE)

fetch_stat_api(name, path, params = list(), parser = NULL, api_key = NULL,
  cache = TRUE, cache_dir = NULL, max_age = 3600, timeout = 30)

search_stat_api(name, query, params = list(), parser = NULL, api_key = NULL,
  cache = TRUE, cache_dir = NULL, max_age = 3600, timeout = 30)
```

Arguments

provider	Provider name used for provenance and API-key lookup.
key	Optional API key.
required	If TRUE, error when no key is available.
base_url	HTTPS base URL of a JSON REST API.
path	Endpoint path relative to the base URL.
query, params	Named lists of URL query parameters.
api_key	Optional API key supplied directly for one request.
api_key_name	Query-parameter name expected by an API for the key.
cache	Whether responses should be cached.
cache_dir	Optional user-selected cache directory.

max_age	Maximum cache age in seconds.
timeout	Request timeout in seconds.
name	Session-local provider identifier.
search_path	Optional search endpoint.
query_param	Name of the search-term parameter.
provider_label	Human-readable provider name.
overwrite	Whether to replace an existing custom registration.
include_custom	Whether to include custom providers in the registry table.
parser	Optional user function that receives parsed JSON.

Value

API-key helpers return a key or invisibly return the environment-variable name. `api_get_json()` returns a `stat_api_response`. Registered fetch and search functions return parser output or a best-effort data frame.

Examples

```
official_stat_providers()

register_stat_api(
  "example_nso",
  base_url = "https://api.example.gov/v1",
  search_path = "search",
  query_param = "q"
)
```

---

stat_claim *Construct an AI-Mediated Statistical Claim*

Description

Creates a structured representation of a statistical claim produced or mediated by an AI system.

Usage

```
stat_claim(value = NA_real_, indicator = NA_character_, geo = NA_character_,
  time = NA_character_, unit = NA_character_, source = NA_character_,
  dataset = NULL, population = NULL, adjustment = NULL, frequency = NULL,
  methodology = NULL, revision = NULL, quality_flag = NULL,
  uncertainty = NULL, transformation = NULL, answer = NULL,
  metadata = list())
```

Arguments

value	Numeric value stated by the AI system.
indicator	Statistical indicator or concept name.
geo	Geography label or code.
time	Reference period.
unit	Statistical unit.
source	Source attributed by the AI system.
dataset	Optional dataset identifier.
population	Optional population definition.
adjustment	Optional adjustment label.
frequency	Optional frequency.
methodology	Optional methodology note or label.
revision	Optional revision/version information.
quality_flag	Optional quality or provisional flag.
uncertainty	Optional named list describing uncertainty.
transformation	Optional named list describing a derivation.
answer	Optional original AI answer text.
metadata	Additional named metadata.

Value

An object of class `stat_claim`.

Examples

```
clm <- stat_claim(10.4, "unemployment rate", "Spain", "2025",
                  "%", "Eurostat")
clm
```

---

stat_reference *Construct an Official-Statistics Reference Object*

Description

Creates a structured canonical representation of one official statistic used as ground truth for fidelity auditing.

Usage

```
stat_reference(value, indicator, geo, time, unit, source,
  dataset = NULL, population = NULL, adjustment = NULL,
  frequency = NULL, methodology = NULL, revision = NULL,
  quality_flag = NULL, uncertainty = NULL, transformation = NULL,
  retrieved_at = Sys.time(), source_url = NULL, version = NULL,
  metadata = list())
```

Arguments

value	Numeric official value.
indicator	Statistical indicator or concept name.
geo	Geography label or code.
time	Reference period.
unit	Statistical unit.
source	Official source/provider.
dataset	Optional dataset identifier.
population	Optional population definition.
adjustment	Optional adjustment.
frequency	Optional frequency.
methodology	Optional methodology note or label.
revision	Optional revision/version status.
quality_flag	Optional quality or provisional flag.
uncertainty	Optional named list describing uncertainty.
transformation	Optional named list describing an expected derivation.
retrieved_at	Retrieval timestamp.
source_url	Optional canonical source URL.
version	Optional source version.
metadata	Additional named metadata.

Value

An object of class `stat_reference`.

Examples

```
ref <- stat_reference(10.4, "unemployment rate", "Spain", "2025",
                      "percent", "Eurostat")
ref
```

---

stress_tools *Prompt Robustness, Statistical Red-Teaming, and Model Stability*

Description

Utilities for testing prompt sensitivity, generating official-statistics failure traps, and measuring cross-model answer stability.

Usage

```
perturb_prompt(prompt, n = 8L, include_original = TRUE)
redteam_stats(reference)
model_stability(results, item_col = "item", model_col = "model",
  answer_col = "answer")
```

Arguments

prompt	A single user prompt.
n	Maximum number of variants.
include_original	Include the original prompt.
reference	A stat_reference.
results	Data frame of benchmark results.
item_col	Item identifier column.
model_col	Model/system identifier column.
answer_col	Answer or canonical answer-key column.

Value

Prompt variants, a red-team data frame, or a stability summary list.

Examples

```
perturb_prompt("What is the unemployment rate in Spain in 2025?", n = 3)
ref <- stat_reference(10.4, "unemployment rate", "Spain", "2025",
                      "percent", "Eurostat")
head(redteam_stats(ref))
```

---

transformation_audit *Audit a Derived Statistical Transformation*

Description

Checks whether the AI used the required statistical operation and, when possible, whether the derived result is numerically correct. Supported built-in operations include difference, percentage-point change, percent/percentage change, ratio, share, sum, and mean.

Usage

```
transformation_audit(reference, claim, abs_tol = 1e-08, rel_tol = 1e-06)
```

Arguments

reference	A stat_reference with optional transformation.
claim	A stat_claim with optional transformation.
abs_tol	Absolute result tolerance.
rel_tol	Relative result tolerance.

Value

A stat_audit_component.

Examples

```
ref <- stat_reference(-0.7, "rate change", "A", "2024-2025",
  "percentage points", "Provider",
  transformation = list(operation = "percentage point change",
                        inputs = c(11.1, 10.4), result = -0.7))
clm <- stat_claim(-6.31, "rate change", "A", "2024-2025",
  "percent", "Provider",
  transformation = list(operation = "percent change",
                        inputs = c(11.1, 10.4), result = -6.31))
transformation_audit(ref, clm)
```

Index