MCP SERVER | NO CODE | CLOUD HOSTED

T-Test Statistics Engine MCP

Calculate guaranteed p-values from raw data.

T-Test Statistics Engine provides mathematically guaranteed t-tests for your AI client. Stop relying on language models to calculate p-values; this MCP runs exact Student's, Welch's, and Paired t-tests locally using a robust statistical engine. Get precise, deterministic results every time you need to test data significance.

A+ Quality Score 100/100

statistics | data-science | mathematics | hypothesis-testing | deterministic-math | p-value

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/t-test-statistics-engine

SCAN TO VIEW

The connectivity layer between AI and the world's software.

Vinkius sits between AI and every application. All communication passes through Vinkius Cloud via the Model Context Protocol (MCP) — with governance, observability, and security at every layer.

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the connectivity layer where AI connects to the software your business already runs. We handle the hosting, the security, the credentials, the uptime — you get agents that actually do things.

We operate the world's largest managed MCP catalog. Major SaaS platforms, CRMs, databases, and cloud providers — running, monitored, production-ready. This MCP server is hosted and maintained by the Vinkius Cloud for AI Agents.

> *The agent doesn't manage credentials, doesn't manage uptime, doesn't manage security. Vinkius does.*
>
> — Architecture principle

Four Pillars of the Vinkius Runtime

01 — Security by design

Credentials stay encrypted at rest via AES-256. The AI agent never touches raw keys — they're injected into a sandboxed V8 isolate at runtime. Actions are logged, and connections have an emergency kill switch.

02 — Built on MCP Fusion

This MCP server was built with **MCP Fusion**, the open-source framework (Apache 2.0) that powers the entire Vinkius catalog. Schema-as-firewall strips undeclared fields, compiled PII redaction runs at zero overhead, and cryptographic lockfiles produce git-diffable audit trails.

03 — Deterministic observability

Eight immutable metrics per endpoint: request volume, p95 latency, error rate, active connections, cost attribution. A live payload feed logs every tool call with mutation detection.

04 — Autonomous operations

Servers are deployed, monitored, and patched autonomously. New capabilities and security patches ship weekly. Zero-downtime deployments ensure continuous availability across all managed MCP servers.

AES-256	Ed25519	24h TTL	V8 Isolate
Encryption at rest	PKI vault signatures	Ephemeral session keys	Sandboxed execution

One Token. Instant Access.

Every MCP server on Vinkius is accessed through a **Connection Token**. Tokens are generated in the cloud dashboard and produce a unique MCP endpoint URL. Paste this URL into any MCP-compatible client — no SDK required.

A single token can serve **multiple AI clients simultaneously**, or you can issue separate tokens per client for granular access control. Each token tracks its own request count, last activity timestamp, and can be individually enabled or revoked.

MCP ENDPOINT

```
https://edge.vinkius.com/{token}/mcp
```

Claude · Cursor · VS Code · Windsurf · Grok · Gemini

Security Is the Architecture

Security in Vinkius is not a feature — it's the foundation of the runtime. The gateway enforces multiple independent protection layers between AI agents and third-party APIs.

01 — Ed25519 PKI Vault

Every workspace has an Ed25519 Master Key. Session keys are generated ephemerally (24h TTL) and signed by the Master Key. Credentials never leave the vault boundary.

02 — V8 Isolate Sandboxing

Tool code runs inside isolated-vm V8 isolates with 64 MB memory caps and per-request timeouts. No filesystem access, no network access except through the SSRF-guarded fetch bridge.

03 — SSRF Guard

All outbound HTTP requests are DNS-resolved and validated before execution. Private IP ranges (10.x, 172.16-31.x, 192.168.x, AWS metadata 169.254.x) are blocked at the network layer.

04 — DLP & PII Redaction

A ResponseGuard pipeline intercepts every tool response. Configurable redaction patterns strip sensitive fields (emails, SSNs, card numbers) before data reaches the AI agent.

05 — Cryptographic Audit Trail

Every request is signed into a SHA-256 hash chain with Ed25519 signatures. Events form a tamper-proof, SIEM-exportable forensic record.

06 — Honeytoken Trap System

Phantom credentials are injected into isolated environments. If a honeytoken is used outside Vinkius infrastructure, the server is quarantined instantly.

Emergency Kill Switch

EU AI Act Art. 14(1) Compliant

The kill switch is an **emergency halt** mechanism — not a simple toggle. When triggered, it executes three actions atomically:

01 — Server deactivated

The MCP server is immediately taken offline across the entire cluster.

02 — All tokens revoked

Every connection token is invalidated. Total lockout — reconnection blocked until new tokens are issued.

03 — WebSocket connections killed

Active connections terminated via Redis pubsub broadcast. Propagates to every runtime node in the cluster.

Full Visibility. Zero Guesswork.

The Vinkius cloud dashboard includes a full MCP Governance suite — real-time analytics and security controls for production AI operations.

Control Plane

KPI dashboard with request volume, latency, success rate, token consumption, and AI-generated operational briefings.

Agent Activity

Which AI clients are connecting, how often, and what they're doing — real-time session tracking.

FinOps

Cost tracking per tool, payload compression savings, budget optimization signals, and consumption trends.

Tool Health

Slowest and most error-prone tools, with actionable root-cause insights and performance baselines.

Firewall & DLP

PII redaction activity, sensitive data protection counters, and security event timeline.

Incident Log

Error trends, failure rates, status-code breakdowns, and forensic audit trail access.

> Get started at **cloud.vinkius.com** — connect your AI agent in under 60 seconds.

T-Test Statistics Engine MCP

1 tools available
Cloud-hosted on Vinkius

When you’re working with real data—like A/B testing conversion rates or medical readings—you can't afford for your AI agent to guess the math. Language models are great at talking about statistics, but they fail spectacularly when it comes to calculation.

This MCP solves that problem by bringing deterministic computation into your workflow. Instead of asking your agent to calculate a p-value and hoping for the best, you route the data through this engine. It handles all complex math—including Student's t-tests, Welch's t-tests, and Paired t-tests—using a reliable local statistical library. Your AI client extracts the raw numbers and sends them here; we guarantee the mathematically correct t-score, degrees of freedom, and p-value back to you.

This means your analysis is based on solid computation, not educated guesswork. You'll know exactly whether or not to reject the null hypothesis at alpha=0.05 without needing a second pair of eyes. Connecting this MCP via Vinkius gives all your compatible AI clients access to statistical rigor, making your data-driven decisions trustworthy.

Core Capabilities

01 — Determine Statistical Significance

The tool calculates the p-value and t-score to tell you if observed differences between datasets are statistically meaningful.

02 — Compare Independent Datasets

You can run a Student's t-test to see if two separate groups, like conversion rates for Variant A and Variant B, differ significantly.

03 — Analyze Paired Measurements

The engine processes paired data, such as blood pressure readings taken before and after a treatment, to find meaningful changes.

04 — Validate Against a Target Mean

Check if a single dataset's average deviates from a known benchmark or target value using a one-sample t-test.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/t-test-statistics-engine** — connect your AI agent in three steps.

01 Your AI client identifies the data points and the type of test needed (e.g., paired, independent).

02 It sends the raw dataset to this MCP for deterministic calculation.

03 You receive a clean output containing the precise t-score, degrees of freedom, and statistically guaranteed p-value.

The bottom line is that you get accurate statistical results without having to write complex Python code or worry about LLM math errors.

Built For

This MCP is built for data scientists, quantitative researchers, and product analysts who treat their numbers like gospel. If your job requires proof—whether it's proving a marketing lift or validating scientific results—you need this reliability.

Data Scientist

Runs A/B tests and compares model performance across different user groups to determine if observed differences are real.

Product Analyst

Checks if a new feature's adoption rate is statistically significant compared to the old version, informing build decisions.

Biostatistician

Analyzes clinical trial results from pre- and post-treatment measurements to validate drug efficacy.

What Changes When You Connect

01 Eliminates math hallucination. You get deterministic, CPU-guaranteed p-values instead of relying on an LLM's best guess for statistical significance.

02 Supports the full suite of necessary tests: run independent comparisons (like comparing two ad campaign groups), paired measurements (pre/post data), and one-sample checks against a target mean.

03 Keeps your data private. The complex math runs locally, meaning sensitive company or research data never leaves your environment when using this MCP.

04 Automates interpretation. After calculating the metrics, the tool automatically tells your agent whether to reject the null hypothesis at the standard alpha=0.05 level.

05 Direct integration for deep workflows. Connects directly through Vinkius, letting any compatible AI client use statistical rigor in natural conversation or code execution.

Real-World Applications

Comparing two marketing variants

A product analyst wants to know if the new checkout flow (Variant B) truly increases conversions compared to the old one (Variant A). They use calculate_t_test on both datasets, receiving a clear p-value. Since p < 0.05, they confirm that Variant B is statistically better and proceed with the rollout.

Checking batch quality control

A manufacturing engineer needs to verify if an entire run of product weights is consistent with the 500g standard. A one-sample t-test runs against the target, confirming that the average weight is not significantly different from spec.

Validating clinical trial results

A biostatistician has blood pressure readings taken before and after a new medication. Running a paired t-test shows a strong, significant drop in average readings, allowing them to confidently conclude the treatment was effective.

Patterns to Avoid

Asking the AI to compute math

✗ AVOID

Prompt: 'What's the p-value if I compare these two lists of numbers?' The LLM provides a plausible number, but it is mathematically incorrect because it can't run true statistical engines.

✓ INSTEAD

Use calculate_t_test. Feed the data into this MCP via your agent to force deterministic calculation and get an accurate result.

Ignoring test type differences

✗ AVOID

Treating pre-treatment scores and post-treatment scores as two separate, unrelated groups when they are actually dependent measurements. This leads to invalid conclusions.

✓ INSTEAD

Use calculate_t_test for paired t-tests. The tool is built to handle the dependency between related data points.

Using generic code snippets

✗ AVOID

Trying to implement complex statistical formulas in general Python code that might miss edge cases or assume incorrect distributions.

✓ INSTEAD

Rely on this MCP. It wraps a robust, dedicated statistical engine, giving you tested reliability for all three major t-test types.

The Right Fit

Use this MCP if your workflow requires absolute mathematical certainty in hypothesis testing. Specifically, if you must compare two groups (independent or paired), or test a sample against a known mean, use calculate_t_test.

Don't use it if you just need general data summaries, like calculating simple averages or generating basic visualizations; those tools are fine for that. Also, don't confuse statistical testing with predictive modeling; this tool only determines significance on *existing* numbers. If your goal is to predict future outcomes based on historical trends, look into dedicated time-series analysis MCPs instead. This engine is pure measurement validation.

The Problem: Statistical Conclusions Based on Guesswork

Today, running a simple A/B test often means copy-pasting data into an AI prompt and asking it to 'figure out the significance.' You get a number back—a p-value or t-score—but you can't verify how that number was generated. The result might look correct, but if the model hallucinates even one variable, your entire multi-million dollar product launch decision rests on a lie.

With this MCP, your agent doesn't just talk about math; it *runs* math. You feed the raw data into calculate_t_test, and you get back verifiable, deterministic results calculated by a dedicated statistical engine. Your conclusion is now trustworthy.

T-Test Statistics Engine MCP: Guaranteed Precision

The manual steps that disappear are the need to write complex boilerplate code, checking for edge cases in Python distributions, or manually verifying the formula used for paired versus independent samples. You don't have to worry about which statistical version you're using.

Now, your agent simply asks the question, and this MCP provides the validated answer—whether it's a clear rejection of the null hypothesis or confirmation that nothing changed.

T-Test Statistics Engine: 1 Tool

Use these tools to perform precise calculations for independent, paired, or one-sample t-tests on your data.

#	TOOL	DESCRIPTION
01	`calculate_t_test`	Runs precise t-tests (independent, paired, one-sample) on data to calculate statistical significance without guessing.

See It in Action

Real prompts you can use once this MCP is connected to your AI agent through Vinkius Cloud.

U Run an independent t-test to see if the conversion rates for Variant A and Variant B are significantly different.

The t-score is 2.45 and the p-value is 0.018. Since p < 0.05, there is a statistically significant difference between the two variants.

U Do a paired t-test on these pre-treatment and post-treatment blood pressure readings.

The paired t-test gives a p-value of 0.002. We reject the null hypothesis — the treatment had a statistically significant effect on blood pressure.

U Perform a one-sample t-test to check if this batch's mean weight differs from the target of 500g.

The calculated p-value is 0.34. We fail to reject the null hypothesis — the batch weight is not significantly different from the 500g target.

Frequently Asked Questions

01 **Does T-Test Statistics Engine MCP handle A/B testing?**

Yes, you use calculate_t_test for this. You simply feed in the conversion data from Variant A and Variant B as two separate groups to determine if their performance difference is statistically significant.

02 **Can I run a paired t-test with T-Test Statistics Engine MCP?**

Yes, calculate_t_test supports paired tests. This is crucial for measuring change over time, like comparing pre- and post-intervention measurements on the same subject.

03 **Is this better than using a standard Python library?**

It's designed to be easier for your agent to use. While it uses robust engines under the hood, you interact with reliable tools that guarantee calculation without needing to manage complex code dependencies.

04 **What kind of data does calculate_t_test accept?**

It accepts numerical datasets—any numbers representing measurements (e.g., rates, counts, scores). It's designed for continuous measurement metrics.

Go Live in 60 Seconds

Get your connection token from **cloud.vinkius.com**, then paste the endpoint URL into any MCP-compatible client.

YOUR MCP ENDPOINT

```
https://edge.vinkius.com/[TOKEN]/mcp
```

CLIENT	WHERE TO CONFIGURE
Claude AI	Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint
Cursor	Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint
VS Code	Ctrl/Cmd+Shift+P → "MCP: Add Server" → add `"t-test-statistics-engine": { "url": "…" }`
Windsurf	MCP Settings → `mcp_settings.json` → Add endpoint URL
ChatGPT	Settings → Tools & plugins → Add MCP server → Paste endpoint
Gemini	Extensions → Add MCP Server → Paste endpoint URL

ASK AN AI ABOUT THIS

Let your preferred AI explain this MCP server

- Ask ChatGPT
- Ask Claude
- Ask Perplexity
- Ask Gemini
- Ask Grok

READY TO CONNECT

T-Test Statistics Engine is live on Vinkius Cloud.

Get your connection token, paste it into your AI agent, and start building. No SDK. No deployment. Just results.

Start at cloud.vinkius.com →

vinkius.com · support@vinkius.com

INDEPENDENT PLATFORM DISCLAIMER

Vinkius is an independent platform and is not affiliated with, endorsed by, sponsored by, verified by, or otherwise authorized by T-Test Statistics Engine. All third-party trademarks, logos, and brand names are the property of their respective owners. Their use in this document is strictly for informational purposes to identify service compatibility and interoperability.

DOCUMENT INFORMATION

Generated	June 2026
MCP Server	T-Test Statistics Engine MCP
Server ID	`019e38f7-0ad7-73ac-8c34-a91d5780f4fd`
Platform	Vinkius Cloud for AI Agents
Endpoint	`https://edge.vinkius.com/{token}/mcp`

LICENSE & USAGE

This document is generated automatically by the Vinkius PDF Engine. Content reflects the MCP server configuration at the time of generation and may change as updates are deployed. For the most current information, visit vinkius.com/mcp/t-test-statistics-engine.