

Research & Analysis / Trend Analysis

Identify exactly when a trend started shifting and what might have caused it.

Difficulty: Advanced

Model: GPT-4 / Claude / Gemini

Use Case: Root Cause Analysis, Incident Investigation, Performance Reviews

Updated: May 2026

Why This Prompt Exists

Something changed. But when? And what caused it? Most teams guess — and usually guess wrong.

You get:

- blaming the wrong product launch for a metric change
- missing the actual change point because you're looking at monthly averages
- celebrating a "recovery" that started before your intervention
- debating causes without first establishing timing
- wasting weeks investigating the wrong period

But change points reveal causation:

- sudden drop/rise: find what happened that day or week
- gradual shift: look for process changes, not single events
- variance change: consistency shifted, not average
- trend reversal: growth turned to decline (or vice versa)
- step change: new baseline established permanently

Without change point detection, you investigate blind.

This prompt finds exactly when your data started behaving differently.

The Prompt

Assume the role of a forensic data analyst who finds change points.

Your task is to identify when a metric's behavior changed and what might explain it.

Generate:

1. CHANGE POINT LOCATION

- Most likely change date(s)
- Confidence (High/Medium/Low)
- Type of change (level shift / trend change / variance change)

2. BEFORE VS. AFTER COMPARISON

- Average before change point: [value]
- Average after change point: [value]
- Difference: [absolute and percentage]
- Statistical significance of difference

3. CHANGE CHARACTERISTICS

- Sudden (overnight) vs. gradual (over weeks)
- Permanent (new baseline) vs. temporary (reverted)
- Magnitude (small, medium, large relative to normal variation)

4. POTENTIAL CAUSES (based on timing)

- Internal changes (launches, process changes, team changes)
- External events (competitor moves, market shifts, seasonality)
- Data issues (tracking changes, bugs, measurement errors)

5. INVESTIGATION PRIORITIES

- Most likely cause to check first
- Evidence needed to confirm
- Who to ask

INPUTS:

Time-series data (daily or weekly recommended):

[PASTE VALUES WITH DATES]

Metric name and description:

[E.G., "Support ticket volume – daily"]

Known events in time range (optional):

[E.G., "Product launch April 3, pricing change May 15"]

Business context:

[E.G., "SaaS company, 24/7 support"]

RULES:

- Daily data is better than weekly; weekly better than monthly
- Multiple change points can exist in the same series
- Distinguish between change point and outlier (one bad day vs. sustained shift)
- Flag when change point coincides with known events (correlation, not causation yet)

How To Use It

- Use daily data for change point detection – monthly averages hide the timing.

- List all known events (launches, changes, incidents) before running — then match timing.
- Check for multiple change points — sometimes things shift more than once.
- Distinguish between a change point and an outlier (one bad day vs. new normal).
- Use the “investigation priorities” to assign who investigates what.

Example Input

Time-series data:

“Daily active users: Week 1: 10.2k, 10.1k, 10.3k, 10.2k, 10.1k, 10.0k, 10.2k. Week 2: 10.1k, 10.0k, 9.8k, 9.5k, 9.3k, 9.1k, 8.9k. Week 3: 8.8k, 8.7k, 8.6k, 8.5k, 8.4k, 8.3k, 8.2k.”

Metric name:

Daily Active Users (DAU)

Known events:

“Server migration attempted on day 8 of this series”

Business context:

Mobile gaming app

Why It Works

Most root cause analysis starts with brainstorming causes before establishing timing — which leads to wild goose chases.

This framework improves outcomes by forcing:

- change point location (when, exactly)
- before/after comparison (magnitude of change)
- change characteristics (sudden vs. gradual, permanent vs. temporary)
- potential cause matching (timeline alignment)
- investigation priorities (where to look first)

Great change point detection doesn't guess causes — it tells you when to look so you can find the real cause.

Build Better AI Systems

Subscribe for advanced prompt engineering, AI coding tools, debugging frameworks, and practical strategies for developers and engineers.

Carefully engineered prompts for people doing real work.

Share this:

- [Share on Facebook \(Opens in new window\) Facebook](#)
- [Share on X \(Opens in new window\) X](#)

See also [Forecast Confidence Assessor](#)