

Research & Analysis / Trend Analysis

Identify leading indicators that predict future changes in your key metrics before they happen.

Difficulty: Advanced

Model: GPT-4 / Claude / Gemini

Use Case: Proactive Monitoring, Risk Management, Business Intelligence

Updated: May 2026

Why This Prompt Exists

Most dashboards show what already happened — by the time you see a problem, it's too late to prevent it.

You get:

- reacting to churn after customers already left
- discovering a trend 3 months after it started
- no ability to test whether an intervention worked until the next report
- firefighting instead of prevention
- surprise at outcomes that were predictable from earlier signals

But leading indicators exist:

- user behavior: support tickets, login frequency, feature usage → predict retention
- sales pipeline: demos booked, proposals sent → predict revenue
- operational: inventory levels, supplier lead times → predict stockouts
- quality: error rates, test failures → predict outages
- financial: payment delays, credit usage → predict defaults

Without early warnings, you're always late.

This prompt identifies which metrics predict your most important outcomes and how to monitor them.

The Prompt

Assume the role of a business intelligence architect who builds early warning systems.

Your task is to identify leading indicators that predict your key metric.

Generate:

1. TARGET METRIC (the lagging indicator you care about)
 - Metric name
 - Typical lag (how late you learn about changes)

2. CANDIDATE LEADING INDICATORS (ranked by predictive power)
 - Indicator name
 - Lead time (how far ahead it signals)
 - Correlation strength with target
 - Direction of relationship (higher indicator = higher/lower target)

3. THRESHOLD & ALERT LOGIC
 - Normal range for each indicator
 - Warning threshold (something's changing)
 - Critical threshold (action required)
 - Example: "When support tickets exceed 500/day, retention drops 2%"

next month"

4. MONITORING RECOMMENDATIONS

- Dashboard to build
- Update frequency (daily/weekly/monthly)
- Who should review

5. ACTION PROTOCOL

- If warning threshold crossed → investigate
- If critical threshold crossed → escalate + intervene

INPUTS:

Target metric (what you want to predict):

[E.G., "Monthly customer churn"]

Available data (what you can measure):

[E.G., "Support tickets, NPS scores, feature usage, billing history"]

Historical relationship (if known):

[E.G., "We think support tickets predict churn, but not sure"]

Business context:

[E.G., "B2B SaaS, enterprise customers"]

RULES:

- Correlation does not guarantee causation – test before acting
- Lead time must be useful (1 week lead time for quarterly planning is useless)

- Distinguish between leading indicator (predicts) and driver (causes)
- Flag when no good leading indicators exist (some things are unpredictable)
- Recommend A/B tests to validate causal relationships

How To Use It

- Start with the metrics you care about most (revenue, churn, engagement) and work backward.
- Look for metrics that move before your target — ideally 2-4 weeks earlier.
- Test candidate leading indicators by seeing if they predict future target values in historical data.
- Build a separate dashboard for leading indicators — don't bury them with lagging metrics.
- Set up automated alerts for threshold crossings, but require human investigation before action.

Example Input

Target metric:

“Monthly customer churn rate (SaaS)”

Available data:

“Daily login frequency, support ticket volume, feature usage (analytics module), NPS survey responses, payment failures”

Historical relationship (if known):

“Customers who stop logging in often churn within 30 days”

Business context:

“B2B SaaS, average customer lifetime 18 months”

Why It Works

Most businesses track what's easy to measure (lagging indicators) instead of what's useful to predict (leading indicators).

This framework improves outcomes by forcing:

- target metric clarity (what you actually care about)
- candidate ranking (not all predictors are equal)
- threshold logic (when to worry)
- monitoring recommendations (dashboard + frequency)
- action protocol (what to do when alerted)

Great early warning systems don't just predict the future — they give you time to change it.

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 [Change Point Detector](#)