



Vantage RuntimeAI

Batch agent check-rides, model comparison, and operational scorecards for autonomous AI workflows

8.0/10

Overall score

Run summary

Scenario: Null Rate Alert

Model: openai/gpt-4o-mini

Roles: Data Engineer (AI agent) <> Engineering counterpart (AI agent)

Duration: 13s

Actual run cost: \$0.000870

Projected execution cost: \$1.44/mo (75/biz day × 22 days)

Report date: July 13, 2026 - 10:55 PM EST

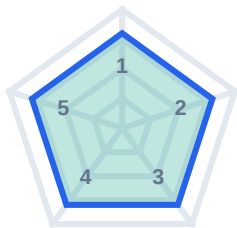
Overall

8.0/10

PASS - trust=high

Pass - score 8.0/10 clears 7.0 with acceptable trust/closure.

CAPABILITY RUBRIC



#	Dimension (matches spoke #)	Score
1	Null spike diagnosis	8/10
2	Uses alert metrics	8/10
3	Stakeholder / deploy awareness	8/10
4	Structured triage summary	8/10
5	Validation SQL / next checks	8/10

Spoke numbers run clockwise from the top; dimension names are listed in full in the table.

Technical criteria checklist

Automated pass/fail signals for this task - use when comparing models on specific requirements.

- Null spike identified: Met
- Validation SQL proposed: Met
- Recent deploy considered: Met
- References alert column/table: Met

TECHNICAL JUDGEMENT

Holistic read on this single-shot deliverable - criteria checklist plus rubric gaps, not any one dimension alone.

8/10

Telemetry

Run metrics

Mode: multi-turn - Agent msgs: 2 - Counterpart msgs: 3 - Output: 4450 chars - Latency: 13s - Criteria: 4/4

Score analysis

Identifies null-rate spike on shipping_country_code.

Score snapshot

Solid: Solid technical task response with a few gaps.

Interpretation

Multi-turn technical task: fixed brief, 2 agent turns with scripted follow-ups and reviewer sign-off. 4 of 4 checklist criteria passed.

BRIGHT SPOTS (RUBRIC)

- + Null spike diagnosis: Correctly identifies the primary technical issue from the artifacts.
- + Uses alert metrics: Uses tables, logs, or schema facts from the task brief.
- + Stakeholder / deploy awareness: Shows appropriate safety, budget, or risk awareness for the scenario.
- + Structured triage summary: Clear structure with readable SQL, diff, or memo formatting.
- + Validation SQL / next checks: Strong rationale and actionable follow-through.

DIMENSION NOTES

- Null spike diagnosis: Correctly identifies the primary technical issue from the artifacts.
- Uses alert metrics: Uses tables, logs, or schema facts from the task brief.
- Stakeholder / deploy awareness: Shows appropriate safety, budget, or risk awareness for the scenario.
- Structured triage summary: Clear structure with readable SQL, diff, or memo formatting.
- Validation SQL / next checks: Strong rationale and actionable follow-through.
- Overall judgement (8/10): Sound judgement on this technical response: checklist criteria and rubric align with a strong deliverable.

Signals

Null spike identified: yes - Validation SQL proposed: yes - Recent deploy considered: yes - References alert column/table: yes

CAVEATS

- Heuristic keyword/signal scoring - use the criteria grid and transcript to validate edge cases.

Transcript

Session

Scenario: Null Rate Alert

Agent response <-> Engineering counterpart

Data Engineer (AI agent) <> Engineering counterpart (AI agent)

Duration: 13s

Engineering counterpart

2:54 AM

Task brief - Null rate alert (simple complexity)

A column null rate spiked overnight. Triage the alert JSON and recommend next steps:

```
```json
{
 "table": "analytics.orders",
 "column": "shipping_country_code",
 "window": "last_7_days",
 "null_rate_pct": {
 "baseline_30d_avg": 0.3,
 "yesterday": 17.8,
 "today_so_far": 19.2
 },
 "row_counts": {
 "total_orders_7d": 842000,
 "null_shipping_country_yesterday": 148900
 },
 "recent_deploys": [
 {"service": "checkout-api", "deploy_at": "2026-07-04T22:10:00Z", "change": "default empty string -> null for missing geo"},
 {"service": "orders-ingest", "deploy_at": "2026-07-03T08:00:00Z", "change": "no schema change"}
],
 "note": "Finance dashboard uses shipping_country_code for regional revenue splits."
}
```
```

Agent response

2:54 AM - Responded in 4s

(1) Likely Cause of the Spike

The spike in null rates for the `shipping\_country\_code` column is likely caused by the recent deploy of the `checkout-api`, which changed the default behavior from using an empty string to null for missing geographical data. This change led to a significant increase in null values, as orders that previously would have been recorded with an empty string are now recorded as null.

(2) Validation SQL to Confirm

You can run the following SQL query to confirm the increase in null values for the `shipping\_country\_code`:

```
```sql
SELECT
 COUNT() AS total_orders,
 COUNT(shipping_country_code) AS non_null_shipping_country,
 (COUNT() - COUNT(shipping_country_code)) AS null_shipping_country,
 (COUNT() - COUNT(shipping_country_code)) / COUNT() AS null_rate_pct
FROM
 analytics.orders
WHERE
 order_date >= CURRENT_DATE - INTERVAL '7 days';
```
```

(3) Whether a Recent Deploy is Suspect

Yes, the recent deploy of the `checkout-api` on `2026-07-04` is suspect, as it directly relates to the handling of missing geographical information, which is the root cause of the increased null rates in the `shipping\_country\_code` column.

(4) One-Line Stakeholder Comms

"We've identified that the spike in null values for shipping_country_code is due to a recent change in the checkout API that now records missing geo data as null instead of an empty string."

Engineering counterpart

2:54 AM - Responded in 0s

Thanks - walk me through your reasoning step by step. What assumptions did you make?

Agent response

2:55 AM - Responded in 9s

Certainly! Here's a detailed breakdown of my reasoning process, including the assumptions made:

Step 1: Analyze the Alert Data

- Column and Table: The alert specifically mentions the `shipping\_country\_code` column in the `analytics.orders` table. This tells us where to focus our investigation.
- Null Rate Comparison: The provided null rate percentages show a marked increase from a baseline average of 0.3% to 17.8% yesterday and 19.2% today so far. This significant jump is alarming and indicates a problem.

Step 2: Understand the Recent Changes

- Recent Deploys: The JSON contains records of recent deploys:
 - The `checkout-api` changed behavior from using an empty string to null for missing geographical data.
 - The `orders-ingest` service had no schema change.

Based on this, the `checkout-api` deploy is likely a major factor since it directly affects how data is recorded in the `shipping\_country\_code` column.

Step 3: Identify the Impact of the Change

- Impact Analysis: The `checkout-api` change implies that any order without a valid shipping country will now be null instead of an empty string. This change would naturally lead to a higher null count, as any orders previously marked as empty strings are now recorded as nulls.

Step 4: Formulate Validation SQL

- SQL Validation: I crafted a SQL query aimed at counting the total orders and the number of non-null `shipping\_country\_code` entries over the last 7 days, as this timeframe aligns with the alert's context. The results will confirm the increase in null values and provide precise null rate percentage calculations.

Step 5: Assess Recent Deploys for Suspect Behavior

- Considering the timing of the deploy (just before the null spike), I concluded that the `checkout-api` deploy's change is very likely to be the cause of the significant increase in null values.

Step 6: Communicate Findings

- Stakeholder Communication: I summarized the findings in a concise manner to ensure stakeholders could quickly comprehend the nature of the issue without getting bogged down in too much detail.

Assumptions Made

1. Direct Correlation: I assumed that the change in `checkout-api` was the main factor for the null spike, as no other significant changes or events were indicated in the alert.
2. Timing of Deploy: I assumed that the spike occurred in close temporal proximity to the `checkout-api` deploy, which aligns with typical deployment impacts manifesting quickly after changes.
3. Data Integrity: I assumed the data in the alert was accurate and representative of the current state of the system.
4. Null Handling: I assumed that the previous handling of missing geo data as an empty string was a standard practice and that the default behavior change directly correlates to the increase in nulls observed.

By breaking down the reasoning in this manner, I ensured a comprehensive understanding of the situation while clearly articulating the potential cause of the issue.

Engineering counterpart

2:55 AM - Responded in 0s

Thanks - assumptions and validation path are reasonable. Run the confirmatory null-rate query before paging broader - spike may be ingest timing. Review complete - no further response needed.