**Simple Anomaly Ops (Unrelated/Ops)**

This process documents a basic unsupervised outlier detection system using simple, clear thresholds.

**Phase 1: Data Preparation and Anomaly Injection**

1. **Data Simulation:** Generate simulated operational data (CPU\_Usage\_Pct, Disk\_IO\_Rate) representing normal system activity.
2. **Anomaly Injection:** Deliberately inject clear outliers that represent a 'Ghost Process' (e.g., high CPU but extremely low Disk I/O).

**Phase 2: Anomaly Detection and Rule Definition**

1. **Detection Method:** Define a simple, custom **rule-based outlier detection system** (simplifying from Isolation Forest).
2. **Threshold Definition:** Establish clear, explainable thresholds for abnormality:
   * CPU Anomaly Threshold: $\text{CPU Usage} > 85\%$
   * I/O Anomaly Threshold: $\text{Disk I/O Rate} < 100$
3. **Flagging Logic:** Combine the rules using boolean logic ($\text{CPU Anomaly} \text{ AND } \text{I/O Anomaly}$) to flag an event as a "Ghost Process."

**Phase 3: Business Action and Visualization**

1. **Alert Generation:** Filter the dataset to isolate all flagged anomalies.
2. **Visualization:** Create a scatter plot of CPU vs. Disk I/O to visually demonstrate how the flagged anomalies sit outside the **normal operating cluster**.
3. **Actionable Output:** Document the resulting **cost savings** (simulated) and the immediate action: alert IT Ops to kill the processes associated with the anomalous resource usage.