GFX Threshold Deviation Dashboard - Master Functional & Business Logic Documentation

Overview

This Streamlit-based dashboard enables end-to-end deviation-based trade alert analysis, grouped by thresholds across currencies. It combines data sourcing, threshold management, alert impact analysis, deviation bucket summaries, and interactive UI behavior with detailed business logic for precise operational handling.

1. Data Sourcing

Trade data and exception data are sourced from APIs and processed in the following way: a. Trade Data (UAT and PROD): - Downloaded using API endpoints, authenticated via OAuth. - URL pattern (one file per API call): Format: ProductType_LegalEntity_SourceSystem_StartDate_EndDate.gz - For each product type, legal entity, and source system, data is downloaded using parallel threads. - Only records from UAT where `trade_id` exists in corresponding PROD file are retained. - 'alert_description' is used to exclude "out of scope" trades (case-insensitive substring check). b. Exception Data: - Retrieved using a separate API call. - Start date: user-selected - End date: current date

2. Threshold File (User Upload)

User uploads a file with the following columns: - LegalEntity - CCY - Original_Group - Original_Threshold - Proposed_Group - Proposed_Threshold Derived Columns: - Adjusted_Group (default = Proposed_Group) - Adjusted_Threshold (default = Proposed_Threshold) - These two columns are editable by the user in the UI. User Selection: - Group-wise thresholding - Currency-wise thresholding - Based on this choice, the threshold view will adapt dynamically in the dashboard. Note: When saving Adjusted thresholds, Ops-style rounding is applied (e.g., 0.43 → 0.40).

3. Threshold Application Logic (Impact Analysis)

- For each trade: - Extract CCY1 and CCY2 from ccypair. - Determine threshold using: a. LegalEntity and currency match b. If match not found, check for fallback "ALL" in LegalEntity column c. If currency still missing, use default threshold for group 1 - Final threshold used = max(threshold for CCY1, threshold for CCY2) - Tie-breaker: If both thresholds are equal, assign evaluation to the alphabetically first currency. Example: If CAD threshold = 0.60 (G2) and AUD threshold = 0.40 (G1), final = CAD G2, 0.60.

4. Deviation Bucket Distribution Summaries

a. Groupwise Summary: - Buckets are dynamically created using ascending Adjusted_Threshold values. - Final bucket is last threshold → inf. - Trade is mapped to both CCY1 and CCY2 in bucket count. - Pivot table: Deviation_Bucket vs. Currency (count of alerts). - Rows where bucket lower bound > max threshold are highlighted. Example highlight: If CAD max threshold is 0.60 and a bucket is [0.65, 0.80), highlight it since 0.65 > 0.60. b. Expanded Bucket Summary: - For alerts where deviationpercent > max(Adjusted_Threshold). - Fixed bin size (default 0.5, configurable in settings), from max threshold → max deviation rounded. - Uses dual currency mapping and highlighting.

5. UI Structure and Interactive Behavior

- Streamlit Dashboard (Single Page App) - Sidebar contains: - Product Type selector - Legal Entity, Source System - Start and End Dates - Group-wise / Currency-wise toggle - File upload for Threshold - Main page contains: - Trade Data Summary - Groupwise Deviation Bucket Summary (AgGrid, scrollable, downloadable) -

Expanded Bucket Summary - Alert Drill-down View (based on bucket selection) - Threshold Table with editable adjusted columns - Severity mapping legend: Info: Deviation < 0.2 Low: 0.2 ≤ Deviation < 0.4 Medium: 0.4 ≤ Deviation < 0.6 High: Deviation ≥ 0.6

6. Drill-Down Interaction

- On selecting any row in the groupwise summary table: - Extract lower and upper range of bucket. - Filter alerts_df where deviationpercent falls in selected bucket range. - Show filtered alert records below in a second table. - Provide option to download filtered records.

7. Output and Export Options

- Download buttons for: - Filtered alerts - Groupwise and expanded summaries - Threshold tables - Audit log of threshold changes (Original → Adjusted) Edge Cases Handled: ---------- - Missing trade_ids in UAT vs PROD. - Fallback to "ALL" and default thresholds. - Tie-breaker rule for equal thresholds. - Inf parsing in bin labels. - Zero-only columns auto-dropped in summaries. - Multiple threshold views based on toggle. - Outliers can be excluded from recalibration but shown in separate view.

8. Detailed Business Logic

A. Definitions

- **Trade**: Minimum fields include Trade ID, Currency Pair (e.g., CADAUD), Notional, Trade/COB Date, and any exception flags. - **Currency Group**: Group (G1..Gn) that a currency belongs to, each with a numeric threshold (e.g., G1=0.40, G2=0.60). - **Threshold Types**: Original (prod), Proposed (from analysis), Adjusted (user-edited for impact analysis). - **Deviation**: Non-negative value showing how far a trade's metric deviates from its group threshold. - **Bucket**: Closed-open interval [L, U) for deviation values (e.g., [0.0, 0.2), [0.2, 0.4), ...).

B. Inputs

1) Trade dataset from UAT/PROD APIs for selected filters. 2) Threshold file mapping group → threshold per currency (Original, Proposed, Adjusted). 3) User selections for threshold type, filters, and edits within the UI grid.

C. Currency Group Assignment Rule

For each trade pair XY (e.g., CADAUD): 1) Look up group thresholds for X and Y from active threshold set. 2) Compare thresholds; assign to currency with higher value. 3) Tie-breaker: If equal, assign to alphabetically first currency. 4) Persist both currencies for distribution calculations.

Pair	CAD Group/Thr	AUD Group/Thr	Assigned Group
CADAUD	G2 / 0.60	G1 / 0.40	CAD → G2 (0.60)

D. Deviation Computation

Formula: **Deviation = max(0, trade.metric** – **Thr_g)** If metric is already deviation-like, use directly against Thr_g.

E. Bucketing Rules

1) Bucket edges from 0.0 to ≥ 99th percentile of deviations (UI-configurable width, default 0.2). 2) Place deviation into [L, U) bucket. 3) Highlight if L ≥ MaxThreshold(group).

F. Distribution & Counting

- Evaluation: Trade counts towards alerts for its assigned group only. - Distribution: Trade counts in deviation distribution for both currencies.

G. Impact Analysis (on Adjusted Edits)

1) Recompute evaluation group. 2) Recalculate deviation. 3) Re-bucket deviations. 4) Recount alerts and regenerate summaries. 5) Update Δ metrics vs Original/Proposed.

H. Alert Categorization (Severity Mapping)

Severity	Deviation Condition	Usage
Info	< 0.2	Monitor only
Low	0.2 ≤ d < 0.4	Triage later
Medium	0.4 ≤ d < 0.6	Actionable if persistent
High	≥ 0.6	Immediate review

I. Rounding Policy

Ops rounds thresholds when setting them; dashboard mirrors this only when saving Adjusted thresholds.

J. Edge Cases & Tie-Breakers

- Equal thresholds: use alphabetically first currency. - Missing threshold: fallback to Proposed → Original → default; else flag in Integrity panel. - Clamp zero/negative values to ≥ 0. - Outliers: toggle exclusion from recalibration; shown separately.

K. Key KPIs

- Total Alerts by severity/group/currency. - Δ Alerts vs Original/Proposed (% change). - Bucket distributions with highlights beyond MaxThreshold. - Top Drivers and Outlier counts.

L. Reference Pseudocode

```
for trade in trades: thr_x = get_threshold(ccy1) thr_y = get_threshold(ccy2) if thr_x is None or
thr_y is None: flag_integrity(trade) continue eval_thr = max(thr_x, thr_y) eval_ccy = ccy1 if thr_x
>= thr_y else ccy2 deviation = max(0, trade.metric - eval_thr) bucket = place_in_bucket(deviation)
counts.eval[eval_ccy_group][bucket] += 1 dist[ccy1][bucket] += 1 dist[ccy2][bucket] += 1
```