Fraud Analytics - PaySim Dataset

Dataset: Synthetic Financial Datasets for Fraud Detection Source: https://www.kaggle.com/datasets/ealaxi/paysim1 Tool Used: PostgreSQL Date: August 2025

Prepared By: Kushagra Sudhir

**Table of Contents**

1. Project Overview
2. Dataset Overview
3. Transaction Summary
4. Fraud Detection Overview
5. Transaction Type Risk Analysis
6. Time-Series Fraud Trends
7. Fraud Type Breakdown
8. Account-Level Risk Profiling
9. Key Insights & Recommendations
10. Appendix – SQL Query Collection

**1. Project Overview**

**Objective:**

This project focuses on identifying, analyzing, and profiling fraudulent behavior in a mobile money transaction dataset (PaySim). The goal is to:

* Detect fraud patterns
* Understand fraudster behavior
* Identify high-risk accounts and transaction types
* Build a foundation for real-time fraud monitoring using SQL

**Problem Statement:**

Fraudulent transactions in digital financial ecosystems are difficult to detect due to their rarity and the complex behaviors fraudsters adopt. By analyzing a realistic synthetic dataset (PaySim), this project aims to:

* Uncover fraud mechanisms
* Use SQL queries for pattern recognition
* Translate findings into business-driven fraud prevention strategies

**Tools & Techniques:**

* **PostgreSQL** for all analysis
* **SQL aggregations, filters, groupings, window functions**
* Rolling metrics, time-based analysis, and account profiling
* No Python, ML, or external tools were used — SQL only

**Outcomes:**

* Actionable insights into fraud hotspots, patterns, and risk indicators
* A complete SQL-powered fraud detection framework
* Ready-to-deploy logic for real-time alerts and risk scoring

**2. Dataset Overview**

The dataset used for this analysis is the PaySim Synthetic Financial Transactions Dataset, simulating mobile money transactions over a 31-day period (Step 1 to Step 744).

The dataset, includes:

* **Total Rows:** 6,362,620 transactions
* **Fraud Cases:** 11,425 transactions (0.18%)

**Columns:**

* ***Step***:
  + Hour of the simulation (1–744)
* **Type**:
  + Transaction type (CASH\_IN, CASH\_OUT, DEBIT, PAYMENT, TRANSFER)
* ***Amount***:
  + Transaction amount in local currency
* ***Nameorig***:
  + Unique ID of sender
* ***Oldbalanceorg***:
  + Sender’s balance before transaction
* ***Newbalanceorig***:
  + Sender’s balance after transaction
* ***Namedest***:
  + Unique ID of recipient
* **Oldbalancedest:**
  + Recipient’s balance before transaction
* **Newbalancedest:**
  + Recipient’s balance after transaction
* **Isfraud:**
  + 1 if fraud, 0 otherwise
* **Isflaggedfraud:**
  + 1 if flagged by system (never occurs in this dataset)

**3. Transaction Summary**

This section provides an overview of the volume and value of transactions in the PaySim dataset.

**Total Transactions and Value**

|  |  |
| --- | --- |
| **Metric** | **Value** |
| Total Transactions | 6,362,620 |
| Total Transaction Value | 2,308,232,917 |

The ecosystem processes over **6.3 million transactions**, totaling more than **$2.3 billion** in synthetic value.

**Transaction Type Breakdown**

|  |  |  |  |
| --- | --- | --- | --- |
| **Transaction Type** | **Count** | **Total Amount** | **Average Amount** |
| **CASH\_OUT** | 2,233,310 | 756,066,839 | 338.42 |
| **PAYMENT** | 2,158,716 | 542,360,084 | 251.22 |
| **CASH\_IN** | 1,399,220 | 509,287,942 | 364.04 |
| **TRANSFER** | 533,228 | 491,527,128 | 921.61 |
| **DEBIT** | 4146 | 9,991,025 | 921.61 |

**Insights**

* CASH\_OUT and PAYMENT dominate transaction volume
* TRANSFERs, though fewer in count, have a significantly higher average value — a potential red flag when profiling fraud

**4. Fraud Detection Overview**

Fraudulent activity is rare but impactful in the PaySim ecosystem. Let’s break it down.

**Fraud vs Legitimate Transactions**

|  |  |  |  |
| --- | --- | --- | --- |
| **isFraud** | **Count** | **Total Amount** | **Average Amount** |
| **0** | 6,351,195 | 2,298,313,504 | 361.98 |
| **1** | 11,425 | 9,919,413 | 868.19 |

* **Fraud Rate:** 0.18% of all transactions
* **Fraud Volume:** ~$9.9 million

**Fraud Rate by Transaction Type**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Total Txns** | **Fraud Txns** | **Fraud Rate (%)** |
| **TRANSFER** | 532,909 | 5,483 | 1.0293% |
| **CASH\_OUT** | 2,233,088 | 5,942 | 0.2661% |
| **Others** | 3,596,623 | 0 | 0.0000% |

**Insights**

* Fraud is entirely concentrated in TRANSFER and CASH\_OUT types
* These transactions **move money out** of the system — aligning with real-world fraud behavior

**5. Transaction Type Risk Analysis**

Here we explore which transaction types are more vulnerable to fraud by analyzing the fraud rate and average value per fraud.

**Fraud Rate and Impact by Type**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Transaction Type** | **Total Txns** | **Fraud Txns** | **Fraud Rate (%)** | **Total Fraud Value** | **Avg Fraud Amount** |
| **TRANSFER** | 532,909 | 5,483 | 1.03% | 5,242,234 | 956.02 |
| **CASH\_OUT** | 2,233,088 | 5,942 | 0.27% | 4,677,179 | 786.77 |
| **Others** | 3,596,623 | 0 | 0.00% | 0 | 0.00 |

**Insights**

1. **Overall Retention is Low**

* **TRANSFERs** are the **riskiest transaction type**, with over **1% fraud rate**
* **CASH\_OUT** transactions have a lower fraud rate but **still critical**, especially since they remove funds
* All other types (PAYMENT, DEBIT, CASH\_IN) show no fraud cases.

**High-Value Fraud Focus**

|  |  |  |
| --- | --- | --- |
| **Range** | **Fraud Count** | **Total Fraud Value** |
| **< $1000** | 5,344 | 3,408,527 |
| **$1000–$10,000** | 5,762 | 5,229,779 |
| **> $10,000** | 319 | 1,281,107 |

* Most fraud occurs in the **$1K–$10K** range, but the largest individual frauds occur above $10K.
* **Value-focused attacks** — not volume-driven

**6. Time-Series Fraud Trends**

This section analyzes fraud activity over time (using the step column, where 1 step = 1 hour)

**Fraud Activity by Time (Hourly Steps)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Total Txns** | **Fraud Txns** | **Fraud Value** | **Avg Fraud** |
| **1** | 8,228 | 1 | 4,518.69 | 4,518.69 |
| **2** | 8,247 | 5 | 3,984.40 | 796.88 |
| **3** | 8,623 | 2 | 6,118.16 | 3,059.08 |
| **4** | 8,264 | 6 | 5,235.69 | 872.61 |
| **5** | 8,667 | 4 | 4,086.78 | 1,021.69 |
| **6** | 8,598 | 3 | 5,166.34 | 1,722.11 |

(Only first 6 steps shown for brevity)

**Rolling Trend Insight**

* Fraud activity is non-linear — it occurs in bursts, with some hours showing spikes and others showing none

**Cumulative Rolling Fraud**

|  |  |
| --- | --- |
| **Step** | **Cumulative Fraud Value** |
| **1** | 4,518.69 |
| **2** | 8,503.09 |
| **3** | 14,621.25 |
| **4** | 19,856.94 |
| **5** | 23,943.72 |
| **6** | 29,110.06 |

* **Rolling metrics** help detect sudden fraud surges
* A real-time system could flag **hourly deviations** from expected behavior and trigger alerts.

**7. Fraud Type Breakdown**

This section dives into how each transaction type contributes to fraud activity — in both volume and impact.

**Fraud Distribution by Transaction Type**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Fraud Count** | **% Delayed** | **Avg Fraud Amount)** |
| **CASH\_OUT** | 5,942 | 4,677,179 | 786.77 |
| **TRANSFER** | 5,483 | 5,242,234 | 956.02 |
| **Others** | 0 | 0 | 0 |

* **TRANSFER** transactions account for the highest total fraud value, despite fewer cases than CASH\_OUT
* Average fraud size is larger for TRANSFERs, reinforcing that these are preferred by high-impact fraudsters

**Highest Single Fraud per Type**

|  |  |
| --- | --- |
| **Transaction Type** | **Max Fraud Amount** |
| **TRANSFER** | 92,445,520 |
| **CASH\_OUT** | 10,000,000 |
| **Others** | 0 |

* The single largest fraud (₹92.4M) occurred via a TRANSFER.
* These outliers signal targeted attacks, likely on high-balance accounts.

**8. Account-Level Risk Profiling**

This section identifies the **top originators and recipients** involved in fraud — crucial for blocking or further investigation

**Top 10 Fraud-Originating Accounts**

|  |  |  |
| --- | --- | --- |
| **Account ID** | **Fraud Count** | **Total Fraud Amount** |
| **C1357854955** | 2 | 183,911.35 |
| **C423292172** | 1 | 182,467.94 |
| **C1554739534** | 1 | 182,072.53 |
| **C1605780810** | 1 | 181,986.01 |
| **C2027667467** | 1 | 181,731.94 |
| **C1769892456** | 1 | 181,630.00 |
| **C1305482379** | 1 | 181,624.00 |
| **C1085352751** | 1 | 181,580.00 |
| **C758502930** | 1 | 181,389.00 |
| **C386189968** | 1 | 181,356.00 |

**Top 10 Fraud-Originating Accounts**

|  |  |  |
| --- | --- | --- |
| **Account ID** | **Fraud Received** | **Count** |
| **C843724798** | 3,774,180.00 | 1 |
| **C1101042610** | 792,684.00 | 1 |
| **C1442557236** | 799,367.00 | 1 |
| **C1230668816** | 34,393,948.00 | 1 |
| **C1217230360** | 34,775,788.00 | 1 |
| **C847222869** | 35,750,656.00 | 1 |
| **C1957752997** | 48,165,352.00 | 1 |
| **C1854746760** | 49,754,392.00 | 1 |
| **C113620708** | 61,231,020.00 | 1 |
| **C414350180** | 68,167,000.00 | 1 |

* Fraud originates from a few high-volume senders.
* But many frauds are routed to unique, one-time receivers — suggesting money mules or burner accounts.
* Recipient accounts have massive spikes (e.g., $68M into a single account) — a red flag for real-time AML (Anti-Money Laundering) alerts.

**9. Money Mule Chain Analysis**

Objective is to Identify suspicious two-hop laundering patterns:

1. A large TRANSFER to an intermediate account (mule)
2. A CASH\_OUT from that mule within 6 hours

This pattern mimics real-world laundering flows, where funds are passed through intermediaries to obscure their origin.

**Top Detected Chains**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Source Account** | **Mule Account** | **Cashout Dest** | **Transfer Step** | **Cashout Step** | **Time Gap (hrs)** | **Transfer Amount** | **Cashout Amount** | **Leakage Amount** | **Leakage %** | **Fraud Flags** | **Source Account** |
| C1923160510 | C636092700 | C377734662 | 305 | 307 | 2 | 1,278,572.50 | 217,273.86 | 1,061,298.64 | 83.01% | 0 / 0 | C1923160510 |
| C992346365 | C53804285 | C1241891991 | 210 | 211 | 1 | 596,675.00 | 100,626.36 | 496,048.62 | 83.14% | 0 / 0 | C992346365 |
| C1024612194 | C1479919685 | C312616124 | 206 | 210 | 4 | 567,514.00 | 264,722.28 | 302,791.72 | 53.35% | 0 / 0 | C1024612194 |
| C11465170 | C1883483535 | C792212230 | 139 | 139 | 0 | 456,477.88 | 247,241.30 | 209,236.58 | 45.84% | 0 / 0 | C11465170 |
| C1483406592 | C2013691076 | C1228316980 | 129 | 132 | 3 | 255,345.27 | 433,722.00 | -178,376.73 | -69.86% | 0 / 0 | C1483406592 |

* Large leakages (40–83%) in most chains suggest funds are siphoned off before cashout — possibly split among multiple mule accounts.
* Negative leakage cases (e.g., C1483406592 → C2013691076) may indicate value consolidation from other sources before cashout, a potential sign of coordinated laundering.
* Short time gaps (0–4 hours) reinforce the suspicion of pre-planned movement rather than organic transfers.
* None of these top chains are flagged as fraud in the dataset — indicating gaps in rule-based detection  
  Implement link analysis to trace these mules further down the chain.
* Apply velocity rules (large inbound transfer followed by large cashout within hours) as an immediate fraud risk flag.

**10. Key Insights & Recommendations**

**Key Fraud Insights**

1. **Concentrated Risk**
   * Fraud occurs only in TRANSFER and CASH\_OUT transactions.
   * 0.18% of transactions are fraudulent but represent ~$9.9M in value.
2. **High-Value Targeting**
   * Fraudsters prefer amounts between $1K–$10K, but outliers above $10M show targeted account takeovers.
3. **Temporal Fraud Bursts**
   * Fraud spikes irregularly across hours — a strong case for rolling-window monitoring.
4. **Suspicious Account Patterns**
   * Many frauds end up in unique recipient accounts, indicating mule networks.

**Fraud Prevention Recommendations**

* **Real-Time Transaction Scoring**  
  Implement rules for:
  + High-value TRANSFER or CASH\_OUT
  + Sudden balance drops
  + Multiple transactions within short periods
* **Recipient Account Monitoring**  
  Flag accounts receiving **unusually large single transfers**.
* **Rolling Fraud Detection**  
  Calculate fraud metrics for **last X hours** to catch bursts.
* **Behavioral Profiling**  
  Monitor sender-recipient relationships; flag new connections for large sums.

**Appendix: SQL Queries Used**

**Query 1:**

SELECT type, COUNT(\*) AS total\_txns,

SUM(amount) AS total\_amount,

ROUND(AVG(amount),2) AS avg\_amount

FROM ps

GROUP BY type

ORDER BY total\_amount DESC;

**Query 2:**

SELECT isFraud, COUNT(\*) AS total\_txns,

SUM(amount) AS total\_amount,

ROUND(AVG(amount),2) AS avg\_amount

FROM ps

GROUP BY isFraud;

**Query 3:**

SELECT type, COUNT(\*) AS total\_txns,

SUM(CASE WHEN isFraud=1 THEN 1 ELSE 0 END) AS fraud\_txns,

ROUND(SUM(CASE WHEN isFraud=1 THEN 1 ELSE 0 END)::NUMERIC\*100/COUNT(\*),4) AS fraud\_rate

FROM ps

GROUP BY type

ORDER BY fraud\_rate DESC;

**Query 4:**

SELECT CASE

WHEN amount < 1000 THEN 'Below $1000'

WHEN amount BETWEEN 1000 AND 10000 THEN '$1000–$10000'

ELSE 'Above $10000'

END AS amount\_range,

COUNT(\*) AS fraud\_count,

SUM(amount) AS total\_fraud\_value

FROM ps

WHERE isFraud = 1

GROUP BY amount\_range

ORDER BY total\_fraud\_value DESC;

**Query 5:**

SELECT step, COUNT(\*) AS total\_txns,

SUM(CASE WHEN isFraud=1 THEN 1 ELSE 0 END) AS fraud\_txns,

SUM(CASE WHEN isFraud=1 THEN amount ELSE 0 END) AS fraud\_value,

ROUND(AVG(CASE WHEN isFraud=1 THEN amount END),2) AS avg\_fraud

FROM ps

GROUP BY step

ORDER BY step;

**Query 6:**

SELECT step,

SUM(CASE WHEN isFraud=1 THEN amount ELSE 0 END)

OVER (ORDER BY step ROWS BETWEEN 5 PRECEDING AND CURRENT ROW) AS rolling\_fraud\_value

FROM ps

ORDER BY step;

**Query 7:**

SELECT type, MAX(amount) AS max\_fraud\_amount

FROM ps

WHERE isFraud = 1

GROUP BY type;

**Query 8:**  
SELECT nameDest, SUM(amount) AS fraud\_received, COUNT(\*) AS txn\_count

FROM ps

WHERE isFraud = 1

GROUP BY nameDest

ORDER BY fraud\_received DESC

LIMIT 10;  
  
**Query 9:**  
SELECT type,

SUM(CASE WHEN isFraud=1 THEN 1 ELSE 0 END) AS fraud\_count,

SUM(CASE WHEN isFraud=1 THEN amount ELSE 0 END) AS total\_fraud\_value,

ROUND(AVG(CASE WHEN isFraud=1 THEN amount END),2) AS avg\_fraud\_amount

FROM ps

GROUP BY type;

**Query 10:**

WITH

-- First leg: suspicious inbound TRANSFER to an intermediate account (the mule)

t1 AS (

SELECT

step AS step\_t1,

nameorig AS src\_acct,

namedest AS mule\_acct,

amount AS t1\_amount,

isfraud AS t1\_isfraud

FROM ps

WHERE type = 'TRANSFER'

),

-- Second leg: CASH\_OUT from that same intermediate account, shortly after

t2 AS (

SELECT

step AS step\_t2,

nameorig AS mule\_acct,

namedest AS cashout\_dest,

amount AS t2\_amount,

isfraud AS t2\_isfraud

FROM ps

WHERE type = 'CASH\_OUT'

),

-- Pair legs where the mule account matches and the cashout happens within 6 hours

paired AS (

SELECT

t1.src\_acct,

t1.mule\_acct,

t2.cashout\_dest,

t1.step\_t1,

t2.step\_t2,

(t2.step\_t2 - t1.step\_t1) AS delta\_hours,

t1.t1\_amount,

t2.t2\_amount,

t1.t1\_isfraud,

t2.t2\_isfraud,

-- how much value “leaks” between legs (fees, splits, skimming)

(t1.t1\_amount - t2.t2\_amount) AS leakage\_amount,

CASE

WHEN t1.t1\_amount = 0 THEN NULL

ELSE ROUND(((t1.t1\_amount - t2.t2\_amount) / t1.t1\_amount::numeric) \* 100, 2)

END AS leakage\_pct

FROM t1

JOIN t2

ON t1.mule\_acct = t2.mule\_acct

AND t2.step\_t2 >= t1.step\_t1

AND t2.step\_t2 <= t1.step\_t1 + 6 -- window: 6 hours

),

-- Rank multiple cashouts that follow the same inbound transfer; keep the first cashout by time

dedup AS (

SELECT

\*,

ROW\_NUMBER() OVER (

PARTITION BY src\_acct, mule\_acct, step\_t1

ORDER BY step\_t2 ASC

) AS rn

FROM paired

)

-- Final: detailed chains + a rollup per mule

SELECT

src\_acct,

mule\_acct,

cashout\_dest,

step\_t1,

step\_t2,

delta\_hours,

t1\_amount,

t2\_amount,

leakage\_amount,

leakage\_pct,

t1\_isfraud,

t2\_isfraud

FROM dedup

WHERE rn = 1

ORDER BY t1\_amount DESC

LIMIT 200;