Fraud Detection & Risk Analysis

This report presents SQL-based insights from a fraud detection and risk analysis project. The dataset includes customer details, transactional records, and associated risk scores. The goal is to identify fraud patterns and derive actionable insights.

1. Fraudulent vs Non-Fraudulent Transactions Count

This query calculates the total number of fraudulent and non-fraudulent transactions.

SQL Query & Output

SELECT is_fraud, COUNT(*) AS total_transactions FROM transactions GROUP BY is_fraud;

+----------+--------------------+

| is_fraud | total_transactions |

+----------+--------------------+

| 0 | 42 |

| 1 | 8 |

+----------+--------------------+

2 rows in set (0.01 sec)

2. Top 5 High-Risk Transactions (By Score)

Displays top 5 transactions with the highest risk scores to identify potential frauds.

SQL Query & Output

SELECT t.transaction_id, t.amount, r.risk_score, t.location, t.date FROM transactions t JOIN risk_scores r ON t.transaction_id = r.transaction_id ORDER BY r.risk_score DESC LIMIT 5;

Empty set (0.00 sec)

3. Fraud Cases by Region

Shows number of fraud cases per region to find high-risk geographical areas.

SQL Query & Output

SELECT c.region, COUNT(*) AS fraud_count FROM transactions t JOIN customers c ON t.customer_id = c.customer_id WHERE t.is_fraud = 1 GROUP BY c.region ORDER BY fraud_count DESC;

+--------+-------------+

| region | fraud_count |

+--------+-------------+

| South | 3 |

| West | 3 |

| North | 2 |

+--------+-------------+

3 rows in set (0.00 sec)

4. Fraud Percentage by Transaction Type

Finds which transaction types are most vulnerable to fraud by calculating fraud percentage.

SQL Query & Output

SELECT transaction_type, COUNT(*) AS total_txns, SUM(is_fraud) AS fraud_txns, ROUND(SUM(is_fraud) * 100.0 / COUNT(*), 2) AS fraud_percentage FROM transactions GROUP BY transaction_type;

+------------------+------------+------------+------------------+

| transaction_type | total_txns | fraud_txns | fraud_percentage |

+------------------+------------+------------+------------------+

| online | 17 | 0 | 0.00 |

| ATM | 14 | 5 | 35.71 |

| POS | 9 | 1 | 11.11 |

| offline | 10 | 2 | 20.00 |

+------------------+------------+------------+------------------+

4 rows in set (0.01 sec)

5. Daily Fraud Trends

Monitors fraudulent transaction activity over time to identify patterns.

SQL Query & Output

SELECT date, COUNT(*) AS total_txns, SUM(is_fraud) AS fraud_txns FROM transactions GROUP BY date ORDER BY date;

+------------+------------+------------+

| date | total_txns | fraud_txns |

+------------+------------+------------+

| 2024-01-02 | 2 | 1 |

| 2024-01-06 | 1 | 1 |

| 2024-01-11 | 2 | 1 |

| 2024-01-15 | 1 | 0 |

| 2024-01-16 | 1 | 0 |

| 2024-01-18 | 1 | 0 |

| 2024-01-28 | 1 | 0 |

| 2024-01-29 | 1 | 0 |

| 2024-01-30 | 2 | 0 |

| 2024-01-31 | 1 | 0 |

| 2024-02-01 | 2 | 0 |

| 2024-02-03 | 2 | 1 |

| 2024-02-05 | 1 | 0 |

| 2024-02-06 | 1 | 1 |

| 2024-02-18 | 2 | 0 |

| 2024-02-19 | 1 | 0 |

| 2024-02-21 | 2 | 1 |

| 2024-02-26 | 1 | 0 |

| 2024-03-02 | 2 | 1 |

| 2024-03-05 | 1 | 0 |

| 2024-03-13 | 1 | 0 |

| 2024-03-14 | 1 | 0 |

| 2024-03-15 | 1 | 0 |

| 2024-03-18 | 3 | 0 |

| 2024-03-25 | 2 | 0 |

| 2024-03-26 | 1 | 0 |

| 2024-03-27 | 1 | 0 |

| 2024-04-01 | 1 | 1 |

| 2024-04-03 | 1 | 0 |

| 2024-04-07 | 1 | 0 |

| 2024-04-08 | 1 | 0 |

| 2024-04-12 | 1 | 0 |

| 2024-04-17 | 1 | 0 |

| 2024-04-18 | 1 | 0 |

| 2024-04-20 | 2 | 0 |

| 2024-04-21 | 1 | 0 |

| 2024-04-22 | 1 | 0 |

| 2024-04-27 | 1 | 0 |

+------------+------------+------------+

38 rows in set (0.00 sec)

6. Average Risk Score for Fraud vs Non-Fraud

Compares the average risk scores between fraudulent and non-fraudulent transactions.

SQL Query & Output

SELECT t.is_fraud, ROUND(AVG(r.risk_score), 2) AS avg_risk_score FROM transactions t JOIN risk_scores r ON t.transaction_id = r.transaction_id GROUP BY t.is_fraud;

Empty set (0.00 sec)

7. Age-wise Fraudulent Transaction Count

Identifies which customer age groups report the most frauds.

SQL Query & Output

SELECT c.age, COUNT(*) AS fraud_txns FROM transactions t JOIN customers c ON t.customer_id = c.customer_id WHERE t.is_fraud = 1 GROUP BY c.age ORDER BY fraud_txns DESC LIMIT 10;

+------+------------+

| age | fraud_txns |

+------+------------+

| 31 | 2 |

| 39 | 2 |

| 18 | 2 |

| 65 | 1 |

| 61 | 1 |

+------+------------+

5 rows in set (0.00 sec)

8. Gender-based Fraud Distribution

Analyzes fraud statistics across genders to understand potential demographic influence.

SQL Query & Output

SELECT c.gender, COUNT(*) AS total_txns, SUM(t.is_fraud) AS fraud_txns FROM transactions t JOIN customers c ON t.customer_id = c.customer_id GROUP BY c.gender;

+--------+------------+------------+

| gender | total_txns | fraud_txns |

+--------+------------+------------+

| M | 35 | 5 |

| F | 15 | 3 |

+--------+------------+------------+

2 rows in set (0.00 sec)

9. Top 5 Locations with Most Fraud

Highlights locations with the highest frequency of fraudulent activity.

SQL Query & Output

SELECT location, COUNT(*) AS fraud_txns FROM transactions WHERE is_fraud = 1 GROUP BY location ORDER BY fraud_txns DESC LIMIT 5;

+----------+------------+

| location | fraud_txns |

+----------+------------+

| New York | 5 |

| Miami | 2 |

| Houston | 1 |

+----------+------------+

3 rows in set (0.00 sec)

10. Customers with the Highest Fraud Transactions

Identifies customers with the most number of fraudulent transactions.

SQL Query & Output

SELECT t.customer_id, COUNT(*) AS fraud_txns FROM transactions t WHERE is_fraud = 1 GROUP BY t.customer_id ORDER BY fraud_txns DESC LIMIT 5;

+-------------+------------+

| customer_id | fraud_txns |

+-------------+------------+

| C006 | 2 |

| C009 | 2 |

| C010 | 2 |

| C002 | 1 |

| C004 | 1 |

+-------------+------------+

5 rows in set (0.01 sec)

Summary of SQL Analysis

The SQL analysis provided a clear overview of the fraud landscape across regions, customer demographics, transaction types, and locations. We identified high-risk zones, user behaviors, and transaction patterns, which can be used by fraud analysts and data teams to improve monitoring systems, risk scoring models, and intervention strategies. This foundational analysis lays the groundwork for further predictive modeling and real-time fraud detection.