SQL Project and explanation

Creating RFM matrix for customer segmentation

Applicable to all lines of business, especially B2C businesses such as Amazon, Myntra, Flipkart, Big basket, and retail stores such as Reliance etc.

Lead Trainer

Saiful Hoda, Exeter MBA (UK), B.Tech. in Computer Science
www.Impetustech.in
+91 93809 73378

Contents

1. Brief Background

There are customers... and there are CUSTOMERS. Knowing the difference between them can make or break an e-commerce business.

You must know who loves you so you can reciprocate and grow the relationship. You must know who is just getting to know you so you can invite them closer. And you should probably know what customers are just plain more trouble than they're worth and no matter how much your try and change them, and no matter how many times they hurt you, they just aren't going to become a faithful partner... err valuable customer.

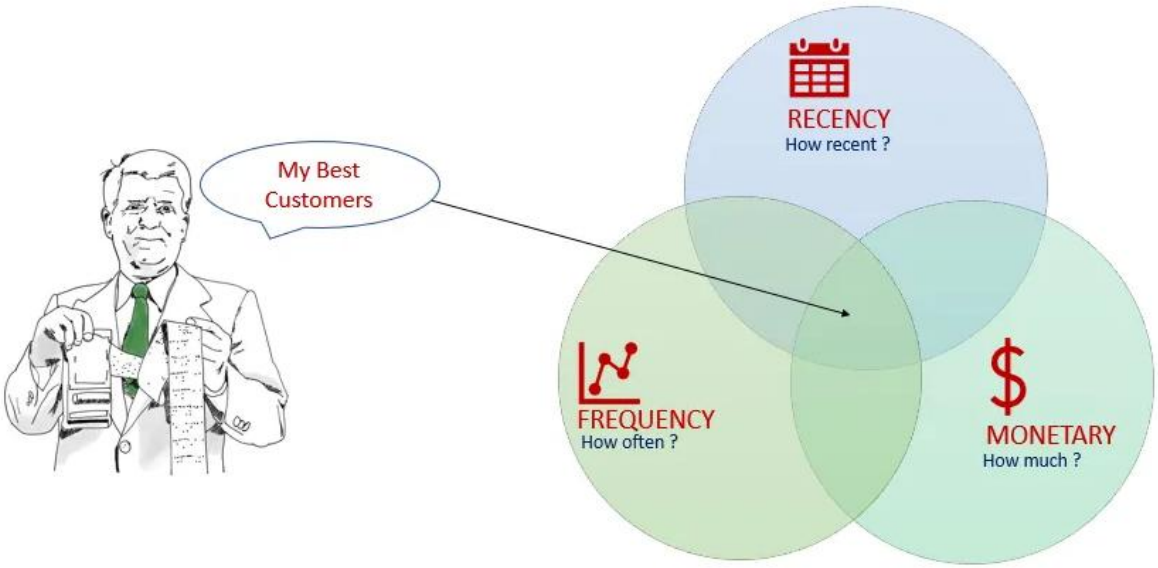
How do you segment customers by love signals? That's the topic of the day.

Introducing RFM Analysis

RFM (Recency, Frequency, Monetary) Analysis is a method of creating homogeneous customer segments based on purchasing behavior.

RFM buckets provide businesses a quick, easy, and well-rounded view of customer spending behaviour. It takes into account

1. **RECENCY** The most recent purchase to ensure the customer is active
2. **FREQUENCY** Number of times a customer transacts
3. **MONETARY** The total amount of money spent.



2. Understanding RFM Analysis

RFM is a marketing analysis tool used to identify a company's best customers by examining their shopping habits. It stands for:

- **Recency:** How recently did the customer make a purchase? (Lower value = more recent = better).

- **Frequency:** How often does the customer purchase? (Higher value = more frequent = better).
- **Monetary:** How much money does the customer spend? (Higher value = more money = better).

Each customer gets an R, F, and M score (typically from 1 to 5, where 5 is the highest/best). These scores are then combined to create an **RFM Segment** (e.g., "555" for the best customers, "111" for the least engaged).

3. Project Setup: Creating Hypothetical Transaction Data

```
DROP TABLE IF EXISTS customer_orders;
```

Create the hypothetical customer_orders table

```
CREATE TABLE customer_orders (
    order_id INT PRIMARY KEY AUTO_INCREMENT,
    customer_id INT NOT NULL,
    order_date DATE NOT NULL,
    order_amount DECIMAL(10, 2) NOT NULL
);

INSERT INTO customer_orders (customer_id, order_date, order_amount) VALUES
(101, '2025-06-10', 50.00),  -- Recent, moderate amount
(101, '2025-05-15', 75.00),  -- Frequent
(101, '2025-04-01', 100.00),
(102, '2025-06-08', 200.00), -- Recent, high amount
(103, '2025-03-20', 30.00),  -- Less recent, low amount
(103, '2025-01-10', 45.00),
(104, '2025-06-05', 120.00), -- Recent, moderate amount
(104, '2025-05-20', 80.00),
(104, '2025-05-01', 90.00),
(104, '2025-04-10', 150.00), -- Very Frequent
(105, '2025-02-28', 15.00),  -- Not recent, low amount
(106, '2025-06-12', 300.00), -- Very recent, very high amount
(107, '2025-05-01', 60.00),
(108, '2025-06-09', 90.00),
(108, '2025-06-01', 110.00),
(109, '2024-11-20', 25.00),  -- Oldest Recency
```

```
(110, '2025-06-11', 85.00),
(110, '2025-06-07', 40.00),
(110, '2025-06-03', 60.00),
(111, '2025-05-25', 180.00),
(112, '2025-06-06', 220.00);

-- Verify the inserted data
SELECT * FROM customer_orders;
```

4. Calculate R, F, M Values for Each Customer

Now, let's calculate the Recency, Frequency, and Monetary values for each unique customer. We use June 14, 2025, for consistency with this exercise to calculate recency.

```
SET @snapshot_date = '2025-06-14';
CREATE TEMPORARY TABLE RFM
(
   SELECT
      customer_id,
      DATEDIFF(@snapshot_date, MAX(order_date)) AS Recency,
      COUNT(DISTINCT order_id) AS Frequency,
      SUM(order_amount) AS Monetary
   FROM
      customer_orders
   GROUP BY
      customer_id
);

SELECT * FROM RFM;
```

5. Assign RFM Scores (1-5 Scale)

We assign a score of 1 through 5, for Recency, Frequency, and Monetary based on their distribution. For simplicity and clarity, we'll use **quantiles (like quintiles)**. This requires **MySQL 8.0+** for window functions like NTILE.

- **Recency:** Lower DATEDIFF means more recent, so a lower rank gets a higher score (e.g., score 5).
- **Frequency:** Higher COUNT means more frequent, so a higher rank gets a higher score (e.g., score 5).
- **Monetary:** Higher SUM means more spent, so a higher rank gets a higher score (e.g., score 5).

```
CREATE TEMPORARY TABLE RFM_Scores
(
SELECT
    customer_id,
    Recency,
    Frequency,
    Monetary,-- Assign Recency Score (lower days = higher score)
    NTILE(5) OVER (ORDER BY Recency DESC) AS R_Score,
-- Use DESC for Recency    (smaller days = better = higher score)
    NTILE(5) OVER (ORDER BY Frequency ASC) AS F_Score,
-- Use ASC for Frequency (smaller count = worse = lower score)
    NTILE(5) OVER (ORDER BY Monetary ASC) AS M_Score
-- Use ASC for Monetary (smaller amount = worse = lower score)
FROM
    RFM
ORDER BY
    customer_id);


    SELECT * FROM RFM_Scores;
```

6. Create RFM Segments

Finally, we'll combine the R, F, and M scores to create a single RFM segment string for each customer. Then, we can define a simple categorization based on these segments.

```
SELECT
    customer_id,
    Recency,
    Frequency,
    Monetary,
    R_Score,
    F_Score,
    M_Score,
    -- Concatenate scores to create the RFM Segment
    CONCAT(R_Score, F_Score, M_Score) AS RFM_Segment,
    -- Categorize customers based on their RFM Segment
    CASE
        WHEN CONCAT(R_Score, F_Score, M_Score) IN ('555', '545', '455', '554') THEN 'Champions' -- Most
valuable customers
        WHEN CONCAT(R_Score, F_Score, M_Score) IN ('544', '454', '445', '535', '355') THEN 'Loyal Customers'
```

```
        WHEN CONCAT(R_Score, F_Score, M_Score) IN ('551', '515', '155', '541', '145') THEN 'New/High-Value but Infrequent' -- Needs frequency boost

        WHEN CONCAT(R_Score, F_Score, M_Score) LIKE '5%' AND F_Score < 3 THEN 'New Customers (Potential)' -- Recently joined, low frequency

        WHEN CONCAT(R_Score, F_Score, M_Score) LIKE '_5_' OR CONCAT(R_Score, F_Score, M_Score) LIKE '__5' THEN 'High-Value/Frequent (Needs Recency)' -- High F/M, but not recent

        WHEN CONCAT(R_Score, F_Score, M_Score) IN ('333', '323', '233') THEN 'At Risk'

        WHEN CONCAT(R_Score, F_Score, M_Score) IN ('111', '112', '121', '211', '122', '212', '221') THEN 'Lost Customers' -- Least engaged

        ELSE 'Other' -- Catch-all for less common combinations

    END AS Customer_Segment

FROM

    RFM_Scores

ORDER BY

    customer_id;
```

7. Interpretation and Next Steps

Once you run the final query, you'll have an RFM matrix and segmentation for your hypothetical customers.

- **Champions (e.g., 555, 545):** Your best customers. Focus on retention, loyalty programs, and asking for referrals.
- **Loyal Customers (e.g., 544, 454):** Solid, consistent customers. Engage them with exclusive offers or new product previews.
- **New Customers (e.g., 5x1/2/3):** Recent, but maybe low frequency/monetary. Nurture them with onboarding materials, special first-time offers.
- **At Risk (e.g., 3x3, 2x3):** Becoming less recent, possibly declining frequency/monetary. Win-back campaigns, personalized discounts.
- **Lost Customers (e.g., 111, 112):** Haven't purchased recently, low frequency and monetary. Deep discounts, re-engagement campaigns, or even focus on acquisition.

8. DROP TEMPORARY TABLES

```
DROP TEMPORARY TABLE RFM;

DROP TEMPORARY TABLE RFM_Scores;
```

9. Next Steps for Your Project - DIY

1. **Visualize:** Import the final RFM data into a visualization tool (like Tableau, Power BI, Google Data Studio, or even Excel) to create charts and dashboards.
2. **Automate:** For ongoing analysis, you could wrap these queries into a **stored procedure** (as discussed before) that calculates and updates RFM scores on a scheduled basis.