Banking Analytics Dashboard DAX Measures and Columns

Prepared for Banking Data Analysis Competition

May 18, 2025

Created by Grok 3, xAI

Contents

1 Overview

This document consolidates all DAX Measures and Calculated Columns for a Power BI dashboard analyzing banking data (Customers, Accounts, Transactions, Loans, Cards, SupportCalls). Each section corresponds to a table, detailing the purpose, DAX code, and Power BI visualization for each measure and column. The goal is to provide a comprehensive, competition-ready set of analytics to highlight customer behavior, financial performance, and operational efficiency.

2 Customers

2.1 Calculated Columns

2.1.1 Full Name

- **Purpose**: Combines FirstName and LastName for clear customer identification.
- **DAX Code**:

```
Full Name = CONCATENATE(Customers[FirstName] & " ", Customers[
   LastName])
```

- **Visualization**: Table Visual to display customer details.

2.1.2 Tenure (Years)

- **Purpose**: Measures the duration of a customer's relationship with the bank in years.
- **DAX Code**:

```
Tenure (Years) = DATEDIFF(Customers[JoinDate], TODAY(), YEAR)
```

- **Visualization**: Bar Chart to compare tenure across customers.

2.1.3 Total Accounts Per Customer

- **Purpose**: Counts the number of accounts per customer.
- **DAX Code**:

```
Total Accounts Per Customer =
COUNTROWS(FILTER(Accounts, Accounts[CustomerID] = Customers[
   CustomerID]))
```

- **Visualization**: Table Visual to identify active customers.

2.1.4 *Total Loans Per Customer*

- **Purpose**: Counts the number of loans per customer.
- **DAX Code**:

```
Total Loans Per Customer =
COUNTROWS(FILTER(Loans, Loans[CustomerID] = Customers[CustomerID
   ]))
```

- **Visualization**: Stacked Bar Chart with Total Accounts for product analysis.

2.1.5 *Customer Activity Tier*

- **Purpose**: Classifies customers (High, Medium, Low) based on transaction and support call activity.
- **DAX Code**:

```
Customer Activity Tier =
VAR TransactionCount =
    CALCULATE(
        COUNTROWS(Transactions),
        FILTER(Accounts, Accounts[CustomerID] = Customers[
           CustomerID])
    )
VAR SupportCallCount =
    CALCULATE(
        COUNTROWS(SupportCalls),
        SupportCalls[CustomerID] = Customers[CustomerID]
    )
RETURN
SWITCH(
    TRUE(),
    TransactionCount > 10 && SupportCallCount <= 2, "High",
    TransactionCount > 5 || SupportCallCount <= 5, "Medium",
    "Low"
)
```

- **Visualization**: Pie Chart for tier distribution.

2.1.6 *Customer Risk Level*

- **Purpose**: Classifies customers based on loan and unresolved support call risks.
- **DAX Code**:

```
Customer Risk Level =
VAR LoanCount =
    CALCULATE(
        COUNTROWS(Loans),
        Loans[CustomerID] = Customers[CustomerID],
```

```
        Loans[InterestRate] > 8
    )
VAR SupportCallCount =
    CALCULATE(
        COUNTROWS(SupportCalls),
        SupportCalls[CustomerID] = Customers[CustomerID],
        SupportCalls[Resolved] = "No"
    )
RETURN
SWITCH(
    TRUE(),
    LoanCount > 1 || SupportCallCount > 3, "High",
    LoanCount = 1 || SupportCallCount > 1, "Medium",
    "Low"
)
```

- **Visualization**: Stacked Bar Chart with state (from Address).

2.2 Measures

2.2.1 Total Customers

- **Purpose**: Total number of customers.
- **DAX Code**:

```
Total Customers = COUNTROWS(Customers)
```

- **Visualization**: Card Visual in Overview page.

2.2.2 New Customers This Year

- **Purpose**: Number of new customers in the current year.
- **DAX Code**:

```
New Customers This Year =
CALCULATE(
    COUNTROWS(Customers),
    YEAR(Customers[JoinDate]) = YEAR(TODAY())
)
```

- **Visualization**: Line Chart to track growth.

2.2.3 Average Tenure (Months)

- **Purpose**: Average customer tenure in months.
- **DAX Code**:

```
Average Tenure (Months) =
AVERAGEX(Customers, DATEDIFF(Customers[JoinDate], TODAY(), MONTH
   ))
```

- **Visualization**: KPI Card.

2.2.4 Average Tenure (Years)

- **Purpose**: Average customer tenure in years.
- **DAX Code**:

```
Average Tenure (Years) =
AVERAGEX(Customers, DATEDIFF(Customers[JoinDate], TODAY(), YEAR)
    )
```

- **Visualization**: KPI Card.

2.2.5 Customer Profitability Score

- **Purpose**: Measures each customer's value based on accounts, loans, and transactions.
- **DAX Code**:

```
Customer Profitability Score =
SUMX(
    Customers,
    CALCULATE(
        SUM(Accounts[Balance]) * 0.5 +
        SUM(Loans[LoanAmount]) * 0.3 +
        SUM(Transactions[Amount]) * 0.2
    )
)
```

- **Visualization**: Table Visual for top 10 customers with a state Slicer.

3 Accounts

3.1 Calculated Columns

3.1.1 Balance Category

- **Purpose**: Classifies accounts by balance (Low, Medium, High).
- **DAX Code**:

```
Balance Category =
SWITCH(
    TRUE(),
    Accounts[Balance] <= 10000, "Low",
    Accounts[Balance] <= 50000, "Medium",
    "High"
)
```

- **Visualization**: Pie Chart for type distribution.

3.1.2 Account Age (Years)

- **Purpose**: Measures account age in years.
- **DAX Code**:

```
Account Age (Years) =
IF(
    ISBLANK(Accounts[CreatedDate]),
    BLANK(),
    DATEDIFF(Accounts[CreatedDate], TODAY(), YEAR)
)
```

- **Visualization**: Bar Chart for age analysis.

3.2 Measures

3.2.1 Total Accounts

- **Purpose**: Total number of accounts.
- **DAX Code**:

```
Total Accounts = COUNTROWS(Accounts)
```

- **Visualization**: Card Visual.

3.2.2 Total Balance

- **Purpose**: Total balance across all accounts.
- **DAX Code**:

```
Total Balance = SUM(Accounts[Balance])
```

- **Visualization**: Card Visual with currency formatting.

3.2.3 Average Account Balance

- **Purpose**: Average balance per account.
- **DAX Code**:

```
Average Account Balance =
AVERAGEX(FILTER(Accounts, Accounts[Balance] > 0), Accounts[
   Balance])
```

- **Visualization**: Bar Chart comparing by AccountType.

4 Transactions

4.1 Calculated Columns

4.1.1 Transaction Category

- **Purpose**: Classifies transactions by amount (Small, Medium, Large).
- **DAX Code**:

```
Transaction Category =
SWITCH(
    TRUE(),
    Transactions[Amount] <= 1000, "Small",
    Transactions[Amount] <= 5000, "Medium",
    "Large"
)
```

- **Visualization**: Donut Chart for type distribution.

4.1.2 Is Active Account

- **Purpose**: Identifies if an account is active based on recent transactions.
- **DAX Code**:

```
Is Active Account =
IF(
    CALCULATE(
        COUNTROWS(Transactions),
        Transactions[AccountID] = Accounts[AccountID],
        Transactions[TransactionDate] >= TODAY() - 180
    ) > 0,
    "Active",
    "Inactive"
)
```

- **Visualization**: Stacked Bar Chart with AccountType.

4.2 Measures

4.2.1 Num of Transactions

- **Purpose**: Total number of transactions.
- **DAX Code**:

```
Num of Transactions = COUNTROWS(Transactions)
```

- **Visualization**: Card Visual.

4.2.2 *Avg Transaction Amount*

- **Purpose**: Average transaction amount.
- **DAX Code**:

```
Avg Transaction Amount =
AVERAGEX(FILTER(Transactions, Transactions[Amount] > 0),
   Transactions[Amount])
```

- **Visualization**: Bar Chart comparing by TransactionType.

4.2.3 *Total Transaction Amount*

- **Purpose**: Total value of transactions.
- **DAX Code**:

```
Total Transaction Amount = SUM(Transactions[Amount])
```

- **Visualization**: Area Chart for monthly volume.

4.2.4 *Year-over-Year Transaction Growth*

- **Purpose**: Measures annual transaction value growth.
- **DAX Code**:

```
YoY Transaction Growth =
VAR CurrentYear = YEAR(TODAY())
VAR PrevYear = CurrentYear - 1
VAR CurrentYearAmount =
    CALCULATE(
        SUM(Transactions[Amount]),
        YEAR(Transactions[TransactionDate]) = CurrentYear
    )
VAR PrevYearAmount =
    CALCULATE(
        SUM(Transactions[Amount]),
        YEAR(Transactions[TransactionDate]) = PrevYear
    )
RETURN
DIVIDE(CurrentYearAmount - PrevYearAmount, PrevYearAmount, 0)
```

- **Visualization**: KPI Card with a 10% target.

5 Loans

5.1 Calculated Columns

5.1.1 *Loan Duration (Years)*

- **Purpose**: Measures loan duration in years.

- **DAX Code**:

```
Loan Duration (Years) =
DATEDIFF(Loans[LoanStartDate], Loans[LoanEndDate], YEAR)
```

- **Visualization**: Bar Chart for duration by LoanType.

5.1.2 Loan Growth Rate

- **Purpose**: Measures annual loan amount growth.
- **DAX Code**:

```
Loan Growth Rate =
VAR CurrentYear = YEAR(TODAY())
VAR PrevYear = CurrentYear - 1
VAR CurrentYearAmount =
    CALCULATE(
        SUM(Loans[LoanAmount]),
        YEAR(Loans[LoanStartDate]) = CurrentYear
    )
VAR PrevYearAmount =
    CALCULATE(
        SUM(Loans[LoanAmount]),
        YEAR(Loans[LoanStartDate]) = PrevYear
    )
RETURN
DIVIDE(CurrentYearAmount - PrevYearAmount, PrevYearAmount, 0)
```

- **Visualization**: KPI Card.

5.1.3 Loan Status

- **Purpose**: Identifies if a loan is active or completed.
- **DAX Code**:

```
Loan Status =
IF(Loans[LoanEndDate] > TODAY(), "Active", "Completed")
```

- **Visualization**: Pie Chart for status distribution.

5.2 Measures

5.2.1 Num of Loans

- **Purpose**: Total number of active loans.
- **DAX Code**:

```
Num of Loans =
CALCULATE(COUNTROWS(Loans), Loans[LoanEndDate] > TODAY())
```

- **Visualization**: Card Visual.

5.2.2 *Total Loan Amount*

- **Purpose**: Total value of loans.
- **DAX Code**:

```
Total Loan Amount = SUM(Loans[LoanAmount])
```

- **Visualization**: Card Visual.

5.2.3 *Avg Interest Rate*

- **Purpose**: Average interest rate for active loans.
- **DAX Code**:

```
Avg Interest Rate =
AVERAGEX(FILTER(Loans, Loans[LoanEndDate] > TODAY()), Loans[
   InterestRate])
```

- **Visualization**: Bar Chart comparing by LoanType.

5.2.4 *Avg Loan Period (Years)*

- **Purpose**: Average loan duration.
- **DAX Code**:

```
Avg Loan Period (Years) =
AVERAGEX(Loans, DATEDIFF(Loans[LoanStartDate], Loans[LoanEndDate
   ], YEAR))
```

- **Visualization**: KPI Card.

5.2.5 *Risky Loans Indicator*

- **Purpose**: Percentage of high-risk loans.
- **DAX Code**:

```
Risky Loans Indicator =
VAR HighRiskThreshold = 8
VAR LargeLoanThreshold = 300000
RETURN
CALCULATE(
    COUNTROWS(Loans),
    FILTER(
        Loans,
        Loans[InterestRate] > HighRiskThreshold &&
        Loans[LoanAmount] > LargeLoanThreshold &&
        Loans[LoanEndDate] > TODAY()
    )
) / COUNTROWS(Loans) * 100
```

- **Visualization**: Gauge Chart with a 20% target.

6 Cards

6.1 Calculated Columns

6.1.1 Card Status

- **Purpose**: Identifies if a card is active or expired.
- **DAX Code**:

```
Card Status =
IF(Cards[ExpirationDate] > TODAY(), "Active", "Expired")
```

- **Visualization**: Pie Chart for status distribution.

6.1.2 Card Age (Years)

- **Purpose**: Measures card age from issuance date.
- **DAX Code**:

```
Card Age (Years) =
IF(
    ISBLANK(Cards[IssuedDate]),
    BLANK(),
    DATEDIFF(Cards[IssuedDate], TODAY(), YEAR)
)
```

- **Visualization**: Bar Chart for age analysis.

6.2 Measures

6.2.1 Num of Cards

- **Purpose**: Total number of active cards.
- **DAX Code**:

```
Num of Cards =
CALCULATE(COUNTROWS(Cards), Cards[ExpirationDate] > TODAY())
```

- **Visualization**: Card Visual.

6.2.2 Avg Card Period (Years)

- **Purpose**: Average card validity period.
- **DAX Code**:

```
Avg Card Period (Years) =
AVERAGEX(Cards, DATEDIFF(Cards[IssuedDate], Cards[ExpirationDate
   ], YEAR))
```

- **Visualization**: KPI Card.

7 SupportCalls

7.1 Calculated Columns

7.1.1 Resolution Status

- **Purpose**: Formats resolution status (Resolved/Unresolved).
- **DAX Code**:

```
Resolution Status =
IF(SupportCalls[Resolved] = "Yes", "Resolved", "Unresolved")
```

- **Visualization**: Pie Chart for status distribution.

7.2 Measures

7.2.1 Total Calls

- **Purpose**: Total number of support calls.
- **DAX Code**:

```
Total Calls = COUNTROWS(SupportCalls)
```

- **Visualization**: Card Visual.

7.2.2 Resolved Calls

- **Purpose**: Number of resolved calls.
- **DAX Code**:

```
Resolved Calls =
CALCULATE(COUNTROWS(SupportCalls), SupportCalls[Resolved] = "Yes
    ")
```

- **Visualization**: Card Visual.

7.2.3 Resolved Rate

- **Purpose**: Percentage of resolved calls.
- **DAX Code**:

```
Resolved Rate =
DIVIDE([Resolved Calls], [Total Calls], 0)
```

- **Visualization**: Gauge Chart with a 95% target.

7.2.4 *Unresolved Calls*

- **Purpose**: Number of unresolved calls.
- **DAX Code**:

```
Unresolved Calls =
CALCULATE(COUNTROWS(SupportCalls), SupportCalls[Resolved] = "No"
    )
```

- **Visualization**: Card Visual.

7.2.5 *Support Calls per Active Account*

- **Purpose**: Number of calls per active account.
- **DAX Code**:

```
Support Calls per Active Account =
DIVIDE(
    COUNTROWS(SupportCalls),
    CALCULATE(
        COUNTROWS(Accounts),
        FILTER(
            Transactions,
            Transactions[AccountID] = Accounts[AccountID],
            Transactions[TransactionDate] >= TODAY() - 180
        )
    ),
    0
)
```

- **Visualization**: Gauge Chart with a 0.5 target.

7.2.6 *Support Calls Efficiency Score*

- **Purpose**: Measures support team efficiency.
- **DAX Code**:

```
Support Calls Efficiency Score =
VAR ResolutionRate = [Resolved Rate]
VAR AvgCallsPerCustomer =
    DIVIDE(
        COUNTROWS(SupportCalls),
        DISTINCTCOUNT(SupportCalls[CustomerID]),
        0
    )
RETURN
ResolutionRate * 0.7 - (AvgCallsPerCustomer / 10) * 0.3
```

- **Visualization**: KPI Card with a 90% target.

8 Dashboard Design Recommendations

- **Structure**:
  - Page 1: Overview (Total Customers, Total Balance, Total Loan Amount, Resolved Rate, Num of Cards).
  - Page 2: Customers (Customer Profitability Score, Customer Risk Level, Customer Activity Tier).
  - Page 3: Accounts & Transactions (Balance Category, YoY Transaction Growth, Transaction Category).
  - Page 4: Loans & Cards (Risky Loans Indicator, Avg Interest Rate, Card Status).
  - Page 5: Support (Support Calls Efficiency Score, Resolved Rate, Support Calls per Active Account).
- **Interactivity**: Add Slicers for JoinDate, TransactionDate, LoanType, IssueType, and State (from Address). Use Drillthrough for customer details (CustomerID).
- **Visual Appeal**: Use a blue-and-white banking Theme with icons (dollar for Total Balance, phone for Total Calls). Apply Conditional Formatting (e.g., green for High Balance Category).
- **Storytelling**: Start with a problem (e.g., high Unresolved Calls), use Resolved Rate in a Gauge Chart, and propose a solution (e.g., automate Account Access Issues).