Chapter 1 – Introduction to Loan Default Risk Analysis

1.1 Introduction

Loan Default Risk Analysis is a specialized area of financial risk management that focuses on identifying, measuring, and interpreting the likelihood that borrowers will fail to repay their loans according to the agreed repayment schedule. Credit is a foundational mechanism in modern economies—individuals and businesses borrow funds to meet personal needs, expand operations, smooth cash flow mismatches, and achieve growth objectives. In return, financial institutions earn interest and other charges as compensation for providing capital and absorbing uncertainty.

However, credit comes with inherent risk. Borrowers may experience financial instability due to job loss, rising household expenses, excessive debt obligations, economic shocks, health issues, or poor financial discipline. When the borrower fails to meet repayment obligations, the loan may become delinquent and eventually default. Such failures are significant for lenders because they lead to financial losses, reduce portfolio quality, and require costly recovery processes.

Loan default risk analysis is therefore not merely an academic exercise; it is a practical business function used to maintain portfolio health and protect lending institutions from excessive risk concentration. It supports decision-making in areas such as underwriting policy design, loan pricing, customer segmentation, and credit monitoring.

The dataset used for this project contains loan-level and borrower-level variables such as loan amount, term, interest rate, income, debt-to-income ratio, revolving credit utilization, delinquencies, credit inquiries, and repayment failure status. Since these variables are widely used in real credit underwriting frameworks, the dataset offers strong potential for building a realistic and professional default risk analysis project.

1.2 Meaning of Lending in Financial Institutions

Financial institutions engage in lending as a core revenue-generating activity. Lending involves the transfer of funds from the lender (bank/NBFC/financial platform) to the borrower under an agreement that specifies:

- the loan amount (principal),
- the interest rate (cost of borrowing),
- repayment duration (term),
- repayment schedule (installments),
- conditions and penalties (late fees, restructuring rules, etc.)

Lending acts as a balance between two competing requirements:

1. **Profitability:** lending must generate interest income and portfolio returns
2. **Risk Control:** lending must protect the institution from excessive defaults

This makes credit management a discipline that demands both quantitative analysis and business understanding.

1.3 What is Loan Default? (Definition and Practical Meaning)

A loan default occurs when a borrower fails to make payments as required under the loan agreement and is unable to bring the account back to normal status. Default is typically preceded by delinquency stages (late payments), but default is a more severe condition indicating high probability of loss unless recovery is possible.

Default can occur due to many reasons such as:

- inability to pay (income shocks, expenses)
- unwillingness to pay (poor discipline)
- over-leverage (too much debt)
- poor credit selection by lender
- higher loan burden due to high interest and installments

In this dataset, repayment failure is captured by the target variable:

- **repay_fail = 1:** repayment failure occurred
- **repay_fail = 0:** repayment failure did not occur

This binary structure allows a clear comparative study between two groups: failed repayments vs successful repayments.

1.4 Why Default Risk Matters (Business Importance)

Loan default affects institutions across multiple dimensions:

(A) Financial Impact

Default reduces interest income and may lead to principal loss. When default rates rise, lenders must provision funds (reserve money for potential losses), directly reducing profitability.

(B) Portfolio Quality

Loan default reduces the overall health of the credit portfolio. A rising default rate is often a warning sign of poor underwriting or worsening borrower conditions.

(C) Operational Impact

Defaulted loans increase collections activity, legal costs, customer support workload, restructuring cases, and recovery processes.

(D) Regulatory and Strategic Impact

Financial institutions may face regulatory requirements regarding provisioning, capital adequacy, and stress testing. High default risk impacts lending strategy and long-term sustainability.

Thus, default risk analysis helps answer questions such as:

- Which borrower groups are riskier?
- Which loan products show higher repayment failure?
- Which features increase repayment stress?
- What segments should be monitored more closely?

1.5 Core Framework of Default Risk (Three-Pillar Understanding)

A standard way to understand default is through a borrower-based risk framework:

(A) Ability to Pay (Affordability)

This refers to whether the borrower has sufficient income and manageable obligations to repay.

Key dataset indicators:

- annual_inc (income)
- dti (debt to income ratio)
- installment (monthly repayment burden)

(B) Willingness to Pay (Behaviour and Discipline)

This refers to borrower's credit discipline and historical behavior.

Key dataset indicators:

- delinq_2yrs
- pub_rec
- inq_last_6mths
- revol_util

(C) Loan Design and Pricing Burden

Loan features determine repayment structure and stress levels.

Key dataset indicators:

- term
- int_rate
- loan_amnt

This dataset is well-aligned with these pillars, which makes it suitable for high-quality default risk analysis.

1.6 Importance of a Dataset-Relevant Approach

Not all default datasets contain the same variables. Some datasets focus on collateral and mortgages (LTV, property value), while others focus on borrower credit behaviour and unsecured lending.

This dataset belongs more to the second category—loan default in a lending portfolio with borrower credit behaviour indicators. Therefore, analysis will focus on:

- affordability patterns (income vs DTI vs installment)
- behavioural risk (delinquencies, inquiries, utilization)
- loan product structure (term, purpose, interest rate)
- segmentation based default rate analysis

Chapter 2 – Fundamental Concepts of Loan Default Risk

2.1 Credit Risk

Credit risk is the possibility that a borrower may fail to meet repayment obligations in full and on time. It is a primary risk faced by banks and lending companies and is influenced by borrower behaviour, loan product structure, and macroeconomic conditions.

Credit risk is commonly addressed through:

- underwriting screening
- pricing decisions
- credit limits
- monitoring and early warning systems

2.2 Default vs Delinquency

Delinquency

Delinquency refers to late payment behavior where the borrower misses payments for a short duration such as 30 days, 60 days, or 90 days past due.

Default

Default is a more permanent repayment failure where repayment is unlikely without restructuring or legal recovery.

In portfolio analytics, delinquency is often treated as a leading indicator of default.

2.3 Risk-Based Pricing

Risk-based pricing is a practice where interest rates are adjusted according to borrower risk profile. This means:

- Low-risk borrowers → lower interest rate
- High-risk borrowers → higher interest rate

In this dataset, int_rate acts as the pricing factor. This column is valuable because it reflects how the lender priced risk at the time of loan issuance.

2.4 Installment Burden and Repayment Stress

Installment is the periodic payment (monthly payment) required from the borrower. Even if two borrowers have the same income, higher installment burden increases stress and reduces affordability.

The dataset includes:

- installment (monthly burden)
- term (duration over which burden exists)

2.5 Debt-to-Income Ratio (DTI)

DTI measures borrower debt obligations relative to income. It indicates how much of borrower income is already tied to debt.

A simplified interpretation:

- Low DTI → more disposable income → safer
- High DTI → tight budget → risky

In this dataset:

- dti serves as a direct affordability metric.

2.6 Loan Purpose and Intent-Based Risk

Borrowers take loans for different purposes such as:

- debt consolidation
- credit card repayment
- home improvement
- education
- small business use

Some purposes may be riskier due to:

- unstable income situations
- low return on borrowed funds
- inability to generate additional cashflow

In this dataset:

- purpose captures borrower intent.

2.7 Credit Discipline Indicators

Credit discipline refers to how responsibly the borrower handles credit.

This dataset contains strong discipline indicators:

Delinquencies (delinq_2yrs)

Number of delinquency events in recent years.

Public Records (pub_rec)

Legal credit events such as bankruptcies or serious records.

Credit Inquiries (inq_last_6mths)

Higher inquiries indicate increased credit demand or financial need.

2.8 Revolving Balance and Revolving Utilization

Revolving credit refers to credit that can be reused, such as credit cards. Two key indicators are:

Revolving Balance (revol_bal)

Total revolving debt currently owed.

Revolving Utilization (revol_util)

Percentage of revolving credit limit being used.

High revolving utilization is often a sign of:

- credit dependency
- financial stress
- lack of liquidity buffer

2.9 Credit History Length

Borrowers with longer credit history have more data available for risk evaluation. Short histories may indicate uncertainty.

The dataset includes:

- earliest_cr_line (first credit line date)
- total_acc (total accounts across history)

Chapter 3 – Dataset Data Dictionary

3.1 Introduction

A data dictionary provides a standardized explanation of dataset variables. In credit risk projects, it is critical to interpret each feature correctly because the same term may hold different meaning across lending institutions and datasets.

This chapter defines all variables used for the final analysis and clarifies their business importance.

3.2 Target Variable

repay_fail

Definition: Binary indicator capturing repayment failure outcome.
Values:

- 0 → No repayment failure
- 1 → Repayment failure
  Importance: This is the dependent outcome variable used to measure default risk distribution and segment-based comparisons.

3.3 Exposure and Principal Variables

loan_amnt

Definition: Total loan principal amount.
Importance: Higher loan amounts increase financial exposure and repayment burden.

funded_amnt

Definition: Amount funded/disbursed by lender.
Importance: Represents actual exposure deployed by institution.

funded_amnt_inv

Definition: Amount funded by investors.
Importance: Useful to compare investor funding impact on performance.

3.4 Loan Structure and Pricing Variables

term

Definition: Loan repayment duration in months.
Importance: Longer term loans may carry different risk due to long uncertainty horizon.

int_rate

Definition: Interest rate charged on loan.
Importance: Represents pricing burden + risk premium.

installment

Definition: Monthly payment amount.
Importance: Strong affordability indicator.

3.5 Borrower Stability Variables

emp_length

Definition: Employment length (years).
Importance: Proxy for stability of cash flows.

home_ownership

Definition: Borrower home ownership status.
Importance: Stability and lifecycle indicator.

verification_status

Definition: Indicates whether borrower income was verified.
Importance: Underwriting confidence indicator.

3.6 Affordability Variables

annual_inc

Definition: Annual borrower income.
Importance: Determines repayment capacity.

dti

Definition: Debt-to-income ratio.
Importance: Measures repayment stress.

3.7 Behaviour and Credit Activity Variables

delinq_2yrs

Definition: Delinquencies in recent years.
Importance: Past payment issues indicate risk.

inq_last_6mths

Definition: Inquiries in last 6 months.
Importance: May indicate active borrowing needs.

open_acc

Definition: Open credit accounts.
Importance: Represents current active credit exposure.

pub_rec

Definition: Public record count.
Importance: Severe negative credit indicator.

revol_bal

Definition: Revolving debt balance.
Importance: Represents existing debt burden.

revol_util

Definition: Revolving utilization percentage.
Importance: Credit stress measure.

total_acc

Definition: Total credit accounts across history.
Importance: Credit maturity / exposure.

3.8 Credit History Timeline Variables

earliest_cr_line

Definition: Date of earliest credit line.
Importance: Allows estimation of credit history length.

mths_since_last_delinq

Definition: Months since last delinquency (or no delinquency).
Importance: Recency of risk behaviour.

3.9 Loan Timing Variable

issue_d

Definition: Loan issue date.
Importance: Supports time trend analysis and cohort monitoring.

3.10 Geographic and Purpose Variables

purpose

Definition: Stated reason for loan.
Importance: Purpose-based risk segmentation.

addr_state

Definition: Borrower state.
Importance: Geographic segmentation of default risk.

Chapter 4 – Business Problem Definition and Project Scope

4.1 Business Background

Financial institutions aim to grow loan portfolios while maintaining stable repayment performance. The major challenge is that borrower repayment depends on multiple interacting factors such as income stability, debt levels, credit discipline, and loan product design.

Without proper risk analysis, institutions may face:

- high default concentration
- declining profitability
- operational overload
- poor risk governance

Thus, analyzing portfolio behaviour using risk factors is critical for identifying borrower segments requiring monitoring and risk control.

4.2 Business Problem Statement

The business problem addressed in this project is:

To analyze borrower and loan characteristics to identify patterns associated with repayment failure and to build a segmented understanding of loan default risk across the portfolio.

The dataset provides multiple borrower-level and loan-level attributes that enable a portfolio-based risk study.

4.3 Project Objectives

The objectives of this project are:

1. Clean and preprocess the dataset to ensure accurate analysis
2. Study the default distribution across key affordability variables
3. Evaluate credit behaviour indicators associated with repayment failure
4. Perform SQL-driven segmentation analysis to generate business insights
5. Validate insights through statistical exploration and visualization
6. Present findings through a professional dashboard and interpretation

4.4 Scope of the Study

Included Scope

- Default rate calculation at portfolio level
- Default segmentation by:
    - term, interest rate, purpose

    - income and DTI
    - credit behaviour and revolving utilization
- State-wise and purpose-wise risk profiling
- High risk segment identification

Excluded Scope

- Loss given default (LGD) and recovery analysis
- Detailed repayment timeline evaluation
- External economic factor integration

4.5 Deliverables

- Clean dataset ready for SQL analysis
- SQL scripts for portfolio KPIs and segmented insights
- EDA graphs with business interpretation
- Risk dashboard summarizing portfolio behaviour
- Recommendations and risk mitigation insights

Chapter 5 – Dataset Overview and Data Quality Audit

5.1 Dataset Overview

The dataset used for this project represents loan-level records containing information related to borrower affordability, loan design, credit behaviour, and repayment outcome. Each row corresponds to a single loan issued to a borrower and includes the default outcome captured through repayment failure.

This dataset resembles real credit underwriting information because it contains loan structure details (interest rate, loan amount, term) and borrower behavioural variables (delinquencies, inquiries, revolving utilization).

5.2 Dataset Size and Structure

After Python-based preprocessing, the dataset contains:

- **Total records:** 38,475
- **Total columns:** 24

This size is sufficient for segmentation analysis and extraction of stable default patterns across multiple borrower and loan categories.

5.3 Target Variable Distribution (repay_fail)

The dataset includes a binary target variable repay_fail, which indicates repayment failure events. Understanding the distribution of this target is essential before performing deeper analysis because it determines the relative proportion of default and non-default cases and influences interpretation of segment default rates.

Portfolio analysis will therefore include:

- total loans count
- total repayment failures
- failure percentage (default rate)

5.4 Data Types and Variable Structure

The dataset includes:

Numerical Variables

- loan amounts, income, DTI
- interest rate and installment burden
- revolving balance, utilization, account counts
- delinquency and inquiry counts

Categorical Variables

- purpose, state, verification category, home ownership status

Date Variables

- issue date
- earliest credit line date

This combination supports both descriptive risk profiling and statistical evidence-based conclusions.

5.5 Missing Values and Data Completeness

The dataset was audited for missing values and inconsistencies. Key findings include:

- Most columns contain complete data with negligible missingness
- Some columns initially had limited missing values, such as employment length and revolving utilization
- A high missingness variable (mths_since_last_delinq) is common in credit datasets because many borrowers do not have delinquency history

Rather than treating this high missingness as a purely negative data quality issue, it may represent an informative condition: absence of delinquency records. Hence, this column should be interpreted carefully during analysis.

5.6 Data Cleaning Observations and Improvements

Key cleaning actions performed before SQL analysis include:

- Removal of leakage variables representing post-loan repayment outcomes
- Conversion of loan term from text format to numeric months
- Standardization of employment length into numeric years
- Conversion of revolving utilization percentage into numeric form
- Treatment of invalid interest rate values (0%) as missing
- Conversion of date fields into standard datetime format
- Drop of minimal missing rows in essential numeric fields

These steps ensure the dataset is both analytically sound and ready for SQL-driven segmentation.

5.7 Conclusion of Data Quality Audit

The dataset is well-suited for loan default risk analysis due to the presence of:

- A clear binary default outcome
- Strong affordability and burden variables (income, DTI, installment)
- Strong behavioural credit indicators (delinquencies, inquiries, utilization)
- Minimal missing values after preprocessing
- Clean variable structure enabling segmentation analysis and dashboard creation.

CHAPTER 6: DATA CLEANING AND PREPROCESSING

6.1 Purpose of Data Cleaning in Loan Default Risk Analysis

In loan default risk analysis, the quality of insights depends heavily on the quality of data. Since borrower characteristics and loan attributes are often recorded across multiple systems and time periods, datasets can contain:

- Missing values
- Incorrect formats (e.g., percentages stored as text)
- Inconsistent categories
- Outlier or invalid entries (e.g., interest rate = 0)
- Leakage variables (post-loan repayment information)

Therefore, before beginning SQL-based segmentation and exploratory analysis, the dataset was cleaned and transformed in Python to ensure that all variables represent valid and meaningful borrower-level information.

The overall objective of this chapter is to document how the dataset was cleaned, standardized, and prepared for downstream analysis.

6.3 Target Variable Understanding (repay_fail)

The dataset contains the target variable:

repay_fail

- **0** = Non-default (successful repayment)
- **1** = Default (repayment failure)

Using value_counts() and percentage distribution:

- **32,651 loans (≈84.85%)** were non-default cases
- **5,829 loans (≈15.15%)** were default cases

This reflects a dataset with **moderate class imbalance**, which is realistic for credit risk portfolios (defaults are usually lower than non-defaults).
This distribution makes the dataset suitable for comparing borrower characteristics between default vs non-default groups.

```
In [9]: df["repay_fail"].value_counts()
Out[9]: 0    32651
        1     5829
        Name: repay_fail, dtype: int64

In [10]: df["repay_fail"].value_counts(normalize=True) * 100
Out[10]: 0    84.851871
         1    15.148129
         Name: repay_fail, dtype: float64
```

6.4 Missing Values Audit

A missing value audit was performed using a tabular summary (missing count + missing percentage).
This step helped identify which columns required either:

- imputation
- transformation
- removal
- or special handling

The results showed:

Column	Missing Count	Missing %	Interpretation
next_pymnt_d	35097	~91.21%	Too much missing → non-reliable
mths_since_last_delinq	24363	~63.31%	Missing likely means “no delinquency history”
emp_length	993	~2.58%	manageable missing
last_pymnt_d	71	very low	minor missing
revol_util	59	very low	minor missing
annual_inc, installment, etc	very low	negligible	

Important conclusion:
Variables like **next_pymnt_d** were extremely incomplete and therefore not useful for reliable portfolio-level analysis.

	missing_count	missing_percent
next_pymnt_d	35097	91.208420
mths_since_last_delinq	24363	63.313410
emp_length	993	2.580561
last_pymnt_d	71	0.184511
revol_util	59	0.153326
revol_bal	4	0.010395
last_credit_pull_d	3	0.007796
annual_inc	2	0.005198
installment	1	0.002599
last_pymnt_amnt	1	0.002599
funded_amnt_inv	1	0.002599
open_acc	1	0.002599
total_rec_int	1	0.002599
total_rec_prncp	1	0.002599
total_pymnt_inv	1	0.002599
total_pymnt	1	0.002599
total_acc	1	0.002599
funded_amnt	1	0.002599
loan_amnt	1	0.002599
delinq_2yrs	1	0.002599

6.5 Format Fixing and Data Type Standardization

A major portion of cleaning involved fixing columns stored in incorrect formats.

6.5.1 Cleaning term

The variable term was originally stored as text values such as:

- “36 months”
- “60 months”

This was converted into an integer form:

- 36
- 60

This is important because numeric term enables grouping, sorting, and performing statistical analysis correctly.

```
In [15]: df["term"].value_counts()
```

```
Out[15]: 36 months    28593
         60 months     9887
         Name: term, dtype: int64
```

```
In [16]: df["term"] = df["term"].str.replace(" months", "", regex=False).astype(int)
```

6.5.2 Cleaning emp_length

The emp_length column contained text categories:

- “10+ years”
- “< 1 year”
- “1 year”
- “2 years”, etc.

To standardize this:

- “10+ years” → 10
- “< 1 year” → 0
- Remove the words “year” and “years”
- Convert into numeric form using pd.to_numeric(errors="coerce")

After cleaning, emp_length became a numeric variable representing approximate years of employment, which is meaningful in credit analysis since longer job stability usually correlates with lower default risk.

```
In [17]: df["emp_length"].value_counts(dropna=False)
```

```
Out[17]: 10+ years    8465
         < 1 year     4565
         2 years      4292
         3 years      3939
         4 years      3314
         1 year       3254
         5 years      3171
         6 years      2144
         7 years      1702
         8 years      1445
         9 years      1196
         NaN           993
         Name: emp_length, dtype: int64
```

```
In [18]: df["emp_length"] = df["emp_length"].str.strip()

         df["emp_length"] = df["emp_length"].replace({
             "10+ years": "10",
             "< 1 year": "0"
         })

         df["emp_length"] = df["emp_length"].str.replace(" years", "", regex=False)
         df["emp_length"] = df["emp_length"].str.replace(" year", "", regex=False)

         df["emp_length"] = pd.to_numeric(df["emp_length"], errors="coerce")
```

6.5.3 Handling invalid interest rate entries

The interest rate column int_rate contained **one invalid value = 0**, which is unrealistic for a loan dataset.

To correct this:

- int_rate == 0 was treated as missing (NaN)
- later imputed appropriately

Insight:
A 0% interest rate is typically either:

- a data entry error
- or missing record stored incorrectly

```
In [19]: (df["int_rate"] == 0).sum()
Out[19]: 1

In [20]: df.loc[df["int_rate"] == 0, "int_rate"] = np.nan
```

6.5.4 Cleaning revol_util

The revolving utilization revol_util was stored in mixed format such as:

- “0.00%”
- “21.30%”
- “99.90%”
- “0%”

This column was transformed by:

- removing “%”
- converting to numeric

This created a clean numeric variable representing credit utilization, which is one of the strongest indicators of credit stress and default risk.

```
In [21]: df["revol_util"].head(10)
Out[21]: 0    “0.00%”
         1     21.30%
         2     99.90%
         3     47.20%
         4         0%
         5         0%
         6     13.60%
         7     47.70%
         8     70.80%
         9     98.70%
         Name: revol_util, dtype: object

In [22]: df["revol_util"] = df["revol_util"].astype(str).str.replace("%", "", regex=False)
         df["revol_util"] = pd.to_numeric(df["revol_util"], errors="coerce")
```

6.5.5 Converting date columns into datetime

Date-like fields were converted into datetime format:

- issue_d
- earliest_cr_line

- last_pymnt_d
- next_pymnt_d
- last_credit_pull_d

This enables time-based exploration such as:

- loan issue period trends
- credit history duration estimation
- repayment timeline insights

However, these columns were later evaluated for leakage risk and relevance.

In [23]:
```
df["issue_d"] = pd.to_datetime(df["issue_d"], format="%b-%y", errors="coerce")
df["earliest_cr_line"] = pd.to_datetime(df["earliest_cr_line"], format="%b-%y", errors="coerce")
df["last_pymnt_d"] = pd.to_datetime(df["last_pymnt_d"], format="%b-%y", errors="coerce")
df["next_pymnt_d"] = pd.to_datetime(df["next_pymnt_d"], format="%b-%y", errors="coerce")
df["last_credit_pull_d"] = pd.to_datetime(df["last_credit_pull_d"], format="%b-%y", errors="coerce")
```

6.6 Removal of Leakage Columns

One of the most important steps in default risk analysis is eliminating **leakage variables**.

Leakage variables

These are columns that contain information available **after repayment begins** or after default outcomes occur.

Examples in this dataset included:

- loan_status
- total_pymnt
- total_pymnt_inv
- total_rec_prncp
- total_rec_int
- last_pymnt_d
- last_pymnt_amnt
- next_pymnt_d
- last_credit_pull_d

Why they were removed:
If these variables are used during analysis, they can falsely inflate conclusions because they indirectly contain repayment outcomes.

✅ **Removing post-loan (leakage) columns**

Some columns contain information that becomes available only after the loan is issued and repayments begin. These variables can leak the outcome and will not be used for default driver analysis.

In [25]:
```
leakage_cols = [
    "loan_status",
    "total_pymnt", "total_pymnt_inv",
    "total_rec_prncp", "total_rec_int",
    "last_pymnt_d", "last_pymnt_amnt",
    "next_pymnt_d",
    "last_credit_pull_d"
]

df = df.drop(columns=leakage_cols, errors="ignore")
```

6.7 Dropping Identifiers (Non-Analytical Fields)

The dataset contained columns like:

- id
- member_id
- zip_code
- "1" (extra unnamed index-like column)

These were removed because identifiers:

- do not provide business insight
- do not explain default risk drivers
- increase noise in analysis

This ensures the analysis remains borrower-centric and interpretable.

```
In [26]: id_cols = ["id", "member_id", "zip_code","1"]
         df = df.drop(columns=id_cols, errors="ignore")

In [27]: df["mths_since_last_delinq"] = df["mths_since_last_delinq"].fillna(-1)

In [28]: df = df.dropna(subset=["loan_amnt", "funded_amnt", "funded_amnt_inv", "installment",
                                "annual_inc", "dti", "delinq_2yrs", "inq_last_6mths",
                                "open_acc", "pub_rec", "revol_bal"])
```

6.8 Special Handling of mths_since_last_delinq

The variable mths_since_last_delinq had heavy missingness.
Instead of dropping this column, it was treated carefully.

✅ The missing values were replaced with **-1**, meaning:

"No delinquency history reported"

This was a meaningful business encoding because:

- missing here is not random
- it represents a borrower who never had delinquency

```
In [27]: df["mths_since_last_delinq"] = df["mths_since_last_delinq"].fillna(-1)
```

6.9 Handling Remaining Missing Values

6.9.1 Row removal for critical financial variables

Rows containing missing values in critical columns were removed using dropna(subset=...) for fields such as:

- loan_amnt
- funded_amnt
- funded_amnt_inv
- installment
- annual_inc

- dti
- delinq_2yrs
- inq_last_6mths
- open_acc
- pub_rec
- revol_bal

This ensures that affordability, credit behavior, and loan terms remain reliable for analysis.

```
In [28]: df = df.dropna(subset=["loan_amnt", "funded_amnt", "funded_amnt_inv", "installment",
                                "annual_inc", "dti", "delinq_2yrs", "inq_last_6mths",
                                "open_acc", "pub_rec", "revol_bal"])
```

```
In [29]: df.shape
Out[29]: (38475, 24)
```

6.9.2 Median imputation for small missingness

For columns with small missing percentage, median was used:

- emp_length
- revol_util
- int_rate

Why median:
Median is robust to outliers and prevents distortion due to skewed distributions.

Important clarification:
Median imputation here does **not reduce analysis quality**, because:

- missing values are very low in count
- values are continuous numeric
- imputation is only applied after leakage variables were removed
- and later analysis is portfolio-level, not predictive ML deployment

```
In [32]: df["emp_length"] = df["emp_length"].fillna(df["emp_length"].median())
         df["revol_util"] = df["revol_util"].fillna(df["revol_util"].median())
         df["int_rate"] = df["int_rate"].fillna(df["int_rate"].median())
```

6.10 Standardization of Categorical Text Columns

Text-based columns were standardized using:

- conversion to lowercase
- stripping extra spaces

Columns standardized:

- home_ownership
- verification_status
- purpose

- addr_state

This avoids category duplication such as:

- “Rent” vs “rent”
- “Not Verified” vs “not verified”

```
In [33]: cat_cols = ["home_ownership", "verification_status", "purpose", "addr_state"]

         for col in cat_cols:
             df[col] = df[col].astype(str).str.strip().str.lower()
```

```
In [34]: df.isna().sum().sort_values(ascending=False).head(15)
```

```
Out[34]: loan_amnt                0
         funded_amnt              0
         total_acc                0
         revol_util               0
         revol_bal                0
         pub_rec                  0
         open_acc                 0
         mths_since_last_delinq   0
         inq_last_6mths           0
         earliest_cr_line         0
         delinq_2yrs              0
         dti                      0
         addr_state               0
         purpose                  0
         issue_d                  0
         dtype: int64
```

6.11 Final Dataset Quality Check After Cleaning

After cleaning, missing values were again re-checked:

- df.isna().sum() confirmed no significant missingness remained
- df.info() verified correct data types
- df.head() confirmed structured, consistent rows

Final dataset included:

- **38,475 rows**
- **24 clean analytical columns**

Missing values after cleaning were reduced to near-zero, making the dataset suitable for:

SQL risk segmentation
EDA comparisons (default vs non-default)
Statistical testing
Regression-based interpretability
Power BI dashboard reporting

```
In [34]: df.isna().sum().sort_values(ascending=False).head(15)
```

```
Out[34]: loan_amnt                0
         funded_amnt              0
         total_acc                0
         revol_util               0
         revol_bal                0
         pub_rec                  0
         open_acc                 0
         mths_since_last_delinq   0
         inq_last_6mths           0
         earliest_cr_line         0
         delinq_2yrs              0
         dti                      0
         addr_state               0
         purpose                  0
         issue_d                  0
         dtype: int64
```

```
In [35]: df.head()
```

Out[35]:

	loan_amnt	funded_amnt	funded_amnt_inv	term	int_rate	installment	emp_length	home_ownership	annual_inc	verification_status	...	delinq_2yrs	earliest
0	0.0	0.0	0.0	36	11.99	0.00	0.0	rent	0.0	not verified	...	0.0	20
1	2500.0	2500.0	2500.0	36	13.98	85.42	4.0	rent	20004.0	not verified	...	0.0	20
2	5000.0	5000.0	5000.0	36	15.95	175.67	4.0	rent	59000.0	not verified	...	0.0	19
3	7000.0	7000.0	7000.0	36	9.91	225.58	10.0	mortgage	53796.0	not verified	...	3.0	19
4	2000.0	2000.0	2000.0	36	5.42	60.32	10.0	rent	30000.0	not verified	...	0.0	19

5 rows × 24 columns

6.12 Final Output of Python Cleaning Phase

The Python cleaning phase successfully delivered:

1. A cleaned dataset with consistent formats
2. Removal of leakage variables for unbiased risk driver analysis
3. Reduced missing values through valid imputation strategies
4. Proper numerical and categorical formats for SQL + visualization
5. A stable foundation for analysis phases such as:
    - SQL segmentation
    - EDA exploration
    - Hypothesis testing
    - Dashboarding

Thus, Python preprocessing acted as the primary foundation step that ensured all further insights remained valid and business-interpretable.

CHAPTER 7: SQL ANALYSIS AND SEGMENTATION INSIGHTS (MySQL)

7.1 Why SQL Was Used

SQL was used to perform structured portfolio analysis by:

- grouping borrower profiles
- computing default rates across segments
- building risk flags and bucket-wise insights
- replicating real-world business analyst workflows

This phase was aligned with the business objective:
"Build a segmented understanding of loan default risk across the portfolio."

7.2 Portfolio Summary Metrics

Initial portfolio-level KPIs were computed:

- Total Loans
- Total Defaults
- Default Rate %
- Average Interest Rate
- Total Funded Amount

```sql
-- ============================================
-- 1. PORTFOLIO OVERVIEW METRICS
-- ============================================
SELECT
    COUNT(*) AS total_loans,
    SUM(repay_fail) AS total_defaults,
    ROUND(SUM(repay_fail) * 100.0 / COUNT(*), 2) AS default_rate_percent
FROM loan_default_cleaned;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

total_loans	total_defaults	default_rate_percent
38474	5826	15.14

```sql
SELECT
    ROUND(AVG(funded_amnt), 2) AS avg_loan_amount,
    ROUND(AVG(int_rate), 2) AS avg_interest_rate,
    ROUND(AVG(dti), 2) AS avg_dti,
    ROUND(AVG(annual_inc), 2) AS avg_annual_income
FROM loan_default_cleaned;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

avg_loan_amount	avg_interest_rate	avg_dti	avg_annual_income
10832.75	12.16	13.38	68998.27

7.3 Default Rate by Loan Term

Default rate was compared across **36 months vs 60 months** loans.

✅ Findings showed:

- **60-month loans carry much higher default risk** compared to 36-month loans.

```sql
-- ============================================
-- 2. DEFAULT RATE BY LOAN TERM
-- ============================================
SELECT
    term,
    COUNT(*) AS total_loans,
    SUM(repay_fail) AS total_defaults,
    ROUND(SUM(repay_fail) * 100.0 / COUNT(*), 2) AS default_rate_percent
FROM loan_default_cleaned
GROUP BY term
ORDER BY default rate percent DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

term	total_loans	total_defaults	default_rate_percent
60	9885	2307	23.34
36	28589	3519	12.31

7.4 Default Rate by Purpose

Loan purpose segmentation highlighted that some categories show consistently higher risk.

Strong insight:

- **small_business loans** had the highest default rate
- educational and medical were also elevated compared to average

```sql
SELECT
    purpose,
    COUNT(*) AS total_loans,
    SUM(repay_fail) AS total_defaults,
    ROUND(SUM(repay_fail) * 100.0 / COUNT(*), 2) AS default_rate_percent
FROM loan_default_cleaned
GROUP BY purpose
ORDER BY default_rate_percent DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

purpose	total_loans	total_defaults	default_rate_percent
small_business	1808	503	27.82
educational	386	82	21.24
renewable_energy	91	16	17.58
medical	675	117	17.33
other	3948	678	17.17
moving	562	93	16.55
house	387	63	16.28
debt_consolidation	17917	2781	15.52
vacation	360	53	14.72
home_improvement	2899	384	13.25
credit_card	4973	576	11.58
car	1481	163	11.01
wedding	909	100	11.00
major_purchase	2078	217	10.44

7.5 Default Rate by Home Ownership

Home ownership was examined to understand stability and default patterns.

Key insight:

- rent + other ownership types showed slightly higher default rates
- mortgage tends to be comparatively stable

```sql
-- ============================================
-- 4. DEFAULT RATE BY HOME OWNERSHIP
-- ============================================
SELECT
    home_ownership,
    COUNT(*) AS total_loans,
    SUM(repay_fail) AS total_defaults,
    ROUND(SUM(repay_fail) * 100.0 / COUNT(*), 2) AS default_rate_percent
FROM loan_default_cleaned
GROUP BY home_ownership
ORDER BY default_rate_percent DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

home_ownership	total_loans	total_defaults	default_rate_percent
none	4	1	25.00
other	125	29	23.20
rent	18250	2887	15.82
own	2958	461	15.58
mortgage	17137	2448	14.28

7.6 Bucketing-Based Segmentation (SQL-driven)

7.6.1 Income Buckets vs Default Rate

Borrowers were bucketed into income segments.

Key insight:

- default rate declines as income increases
- lowest income bucket (<25k) shows highest default rate

```sql
SELECT
    CASE
        WHEN annual_inc < 25000 THEN '<25k'
        WHEN annual_inc < 50000 THEN '25k-50k'
        WHEN annual_inc < 75000 THEN '50k-75k'
        WHEN annual_inc < 100000 THEN '75k-100k'
        WHEN annual_inc < 150000 THEN '100k-150k'
        ELSE '150k+'
    END AS income_bucket,
    COUNT(*) AS total_loans,
    SUM(repay_fail) AS total_defaults,
    ROUND(SUM(repay_fail) * 100.0 / COUNT(*), 2) AS default_rate_percent
FROM loan_default_cleaned
GROUP BY income_bucket
ORDER BY default_rate_percent DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

income_bucket	total_loans	total_defaults	default_rate_percent
<25k	2360	477	20.21
25k-50k	12016	2058	17.13
50k-75k	11766	1772	15.06
150k+	1791	230	12.84
75k-100k	6316	810	12.82
100k-150k	4225	479	11.34

7.6.2 DTI Buckets vs Default Rate

Affordability stress bucket analysis was performed.

Key insight:

- DTI bucket **20–30** shows high risk
- DTI = 0 / not reported also had elevated risk

```sql
SELECT
    CASE
        WHEN dti = 0 THEN 'DTI = 0 / Not Reported'
        WHEN dti < 10 THEN '0-10'
        WHEN dti < 20 THEN '10-20'
        WHEN dti < 30 THEN '20-30'
        WHEN dti < 40 THEN '30-40'
        ELSE '40+'
    END AS dti_bucket,
    COUNT(*) AS total_loans,
    SUM(repay_fail) AS total_defaults,
    ROUND(SUM(repay_fail) * 100.0 / COUNT(*), 2) AS default_rate_percent
FROM loan_default_cleaned
GROUP BY dti_bucket
ORDER BY default_rate_percent DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

dti_bucket	total_loans	total_defaults	default_rate_percent
DTI = 0 / Not Reported	190	38	20.00
20-30	7441	1276	17.15
10-20	18322	2870	15.66
0-10	12521	1642	13.11

7.6.3 Interest Rate Buckets vs Default Rate

Interest rate gradients were one of the clearest risk indicators.

Key insight:

- default rate rises sharply as interest rate bucket increases
- highest risk observed for **20%+**

```
43  SELECT
44      CASE
45          WHEN int_rate < 7 THEN '0-7'
46          WHEN int_rate < 10 THEN '7-10'
47          WHEN int_rate < 13 THEN '10-13'
48          WHEN int_rate < 16 THEN '13-16'
49          WHEN int_rate < 20 THEN '16-20'
50          ELSE '20+'
51      END AS int_rate_bucket,
52      COUNT(*) AS total_loans,
53      SUM(repay_fail) AS total_defaults,
54      ROUND(SUM(repay_fail) * 100.0 / COUNT(*), 2) AS default_rate_percent
55  FROM loan_default_cleaned
56  GROUP BY int_rate_bucket
57      ORDER BY default_rate_percent DESC;
58
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

int_rate_bucket	total_loans	total_defaults	default_rate_percent
20+	847	295	34.83
16-20	5158	1350	26.17
13-16	9576	1792	18.71
10-13	11724	1635	13.95
7-10	7666	595	7.76
0-7	3503	159	4.54

7.6.4 Revolving Utilization Buckets vs Default Rate

Revolving utilization was also bucketed.

Key insight:

- **90%+ utilization is a strong stress indicator**

- default rate rises steadily with higher utilization

```sql
SELECT
    CASE
        WHEN revol_util < 10 THEN '0-10'
        WHEN revol_util < 30 THEN '10-30'
        WHEN revol_util < 50 THEN '30-50'
        WHEN revol_util < 70 THEN '50-70'
        WHEN revol_util < 90 THEN '70-90'
        ELSE '90+'
    END AS revol_util_bucket,
    COUNT(*) AS total_loans,
    SUM(repay_fail) AS total_defaults,
    ROUND(SUM(repay_fail) * 100.0 / COUNT(*), 2) AS default_rate_percent
FROM loan_default_cleaned
GROUP BY revol_util_bucket
  ORDER BY default_rate_percent DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

revol_util_bucket	total_loans	total_defaults	default_rate_percent
90+	3171	689	21.73
70-90	7579	1423	18.78
50-70	8300	1334	16.07
30-50	8240	1153	13.99
10-30	6904	759	10.99
0-10	4280	468	10.93

7.7 Multi-dimensional Risk Deep Dive (2-variable Segmentation)

To replicate real-world credit portfolio analytics, multi-variable segmentation was created.

7.7.1 Term × Interest Rate Bucket

This showed how pricing and loan duration interact.

Key insight:

- **60M + high interest rates** produced the worst default outcomes.

```sql

SELECT
    term,
    CASE
        WHEN int_rate < 7 THEN '0-7'
        WHEN int_rate BETWEEN 7 AND 10 THEN '7-10'
        WHEN int_rate BETWEEN 10 AND 13 THEN '10-13'
        WHEN int_rate BETWEEN 13 AND 16 THEN '13-16'
        WHEN int_rate BETWEEN 16 AND 20 THEN '16-20'
        ELSE '20+'
    END AS int_rate_bucket,
    COUNT(*) AS total_loans,
    SUM(repay_fail) AS total_defaults,
    ROUND(SUM(repay_fail) * 100.0 / COUNT(*), 2) AS default_rate_percent
FROM loan_default_cleaned
GROUP BY term, int_rate_bucket
ORDER BY default_rate_percent DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

term	int_rate_bucket	total_loans	total_defaults	default_rate_percent
60	20+	749	264	35.25
36	20+	98	31	31.63
60	16-20	3168	897	28.31
36	16-20	1879	430	22.88
60	13-16	2688	605	22.51
60	10-13	2513	456	18.15
36	13-16	6999	1210	17.29
36	10-13	8986	1150	12.80
60	7-10	633	74	11.69
60	0-7	134	11	8.21
36	7-10	7258	550	7.58
36	0-7	3369	148	4.39

7.7.2 Purpose × Interest Rate Bucket

This showed how category-level risk changes under different interest rates.

Key insight:

- small business + high interest buckets are extremely risky
- debt consolidation also shows high exposure due to volume

```sql
SELECT
    purpose,
    CASE
        WHEN int_rate < 7 THEN '0-7'
        WHEN int_rate < 10 THEN '7-10'
        WHEN int_rate < 13 THEN '10-13'
        WHEN int_rate < 16 THEN '13-16'
        WHEN int_rate < 20 THEN '16-20'
        ELSE '20+'
    END AS int_rate_bucket,
    COUNT(*) AS total_loans,
    SUM(repay_fail) AS total_defaults,
    ROUND(SUM(repay_fail) * 100.0 / COUNT(*), 2) AS default_rate_percent
FROM loan_default_cleaned
GROUP BY purpose, int_rate_bucket
HAVING COUNT(*) >= 200
ORDER BY default_rate_percent DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

purpose	int_rate_bucket	total_loans	total_defaults	default_rate_percent
debt_consolidation	20+	489	177	36.20
small_business	16-20	365	131	35.89
other	16-20	457	137	29.98
small_business	13-16	471	138	29.30
small_business	10-13	497	141	28.37
debt_consolidation	16-20	2804	735	26.21
other	13-16	990	224	22.63
credit_card	16-20	560	123	21.96
home_improvement	16-20	317	64	20.19
home_improvement	13-16	626	125	19.97
debt_consolidation	13-16	4757	860	18.08
major_purchase	13-16	441	75	17.01
other	10-13	1261	192	15.23
credit_card	13-16	1174	175	14.91
home_improvement	10-13	875	126	14.40
car	10-13	420	59	14.05
small_business	7-10	274	36	13.14

7.8 Risk Flag Segmentation (Custom Rule-Based Segments)

A business-style risk flag was created using combined rules like:

- 60M and 20%+
- small_business and 16%+
- revol_util 90%+
- low income
- high DTI

These groups were compared by:

- total loans
- total defaults
- default rate

```sql
-- ============================================
-- 10. RULE-BASED RISK SEGMENTATION (RISK FLAG)
-- ============================================


SELECT
    CASE
        WHEN term = 60 AND int_rate >= 20 THEN 'High Risk (60M + 20%+)'
        WHEN purpose = 'small_business' AND int_rate >= 16 THEN 'High Risk (SB + 16%+)'
        WHEN revol_util >= 90 THEN 'High Risk (Revol Util 90%+)'
        WHEN annual_inc < 25000 THEN 'Moderate Risk (Low Income)'
        WHEN dti >= 20 THEN 'Moderate Risk (High DTI)'
        ELSE 'Low Risk / Standard'
    END AS risk_flag,
    COUNT(*) AS total_loans,
    SUM(repay_fail) AS total_defaults,
    ROUND(SUM(repay_fail) * 100.0 / COUNT(*), 2) AS default_rate_percent
FROM loan_default_cleaned
GROUP BY risk_flag
ORDER BY default_rate_percent DESC;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

risk_flag	total_loans	total_defaults	default_rate_percent
High Risk (SB + 16%+)	382	139	36.39
High Risk (60M + 20%+)	749	264	35.25
High Risk (Revol Util 90%+)	2903	600	20.67
Moderate Risk (Low Income)	2153	424	19.69
Moderate Risk (High DTI)	6124	992	16.20
Low Risk / Standard	26163	3407	13.02

```sql
-- ==========================================
-- 11. DEFAULT CONTRIBUTION (%) BY SEGMENT
-- ==========================================

SELECT
    CASE
        WHEN purpose = 'small_business' AND int_rate >= 16 THEN 'High Risk (SB + 16%+)'
        WHEN term = 60 AND int_rate >= 20 THEN 'High Risk (60M + 20%+)'
        WHEN revol_util >= 90 THEN 'High Risk (Revol Util 90%+)'
        WHEN annual_inc < 25000 THEN 'Moderate Risk (Low Income)'
        WHEN dti >= 20 THEN 'Moderate Risk (High DTI)'
        ELSE 'Low Risk / Standard'
    END AS risk_flag,
    SUM(repay_fail) AS total_defaults,
    ROUND(SUM(repay_fail) * 100.0 / (SELECT SUM(repay_fail) FROM loan_default_cleaned), 2) AS percen
FROM loan_default_cleaned
GROUP BY risk_flag
ORDER BY total_defaults DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

risk_flag	total_defaults	percent_of_all_defaults
Low Risk / Standard	3407	58.48
Moderate Risk (High DTI)	992	17.03
High Risk (Revol Util 90%+)	600	10.30
Moderate Risk (Low Income)	424	7.28
High Risk (60M + 20%+)	227	3.90
High Risk (SB + 16%+)	176	3.02

CHAPTER 8: EXPLORATORY DATA ANALYSIS (EDA) – INSIGHTS & STATISTICS

8.1 Objective of EDA

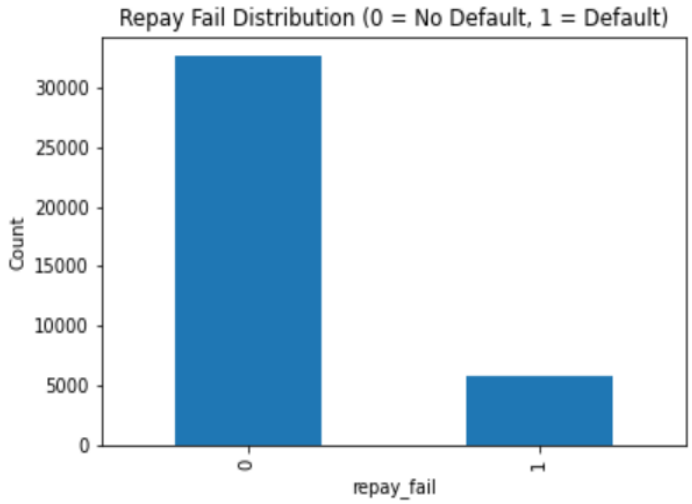
EDA was conducted to validate SQL patterns using statistical exploration and visualization.
It also helped in identifying key drivers associated with repayment failure.

8.2 Target Variable Distribution

A bar chart was used to observe default vs non-default counts.

Key insight:

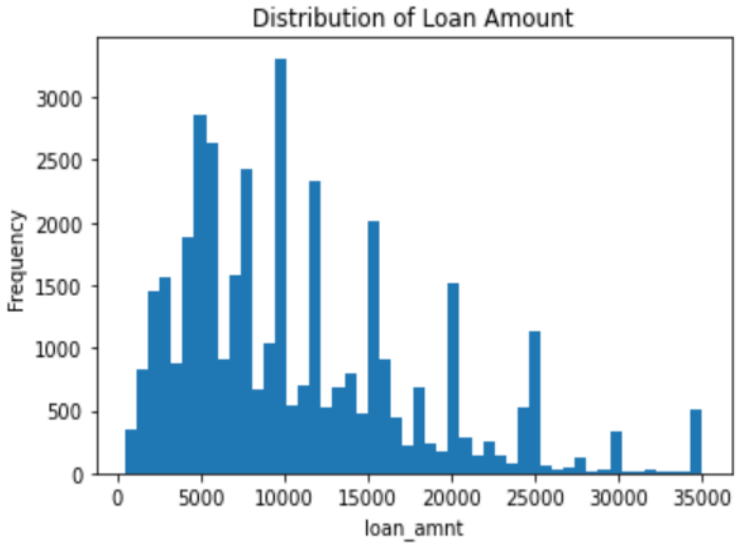
- the dataset is imbalanced, but still has sufficient default cases (~15%)



8.3 Univariate Analysis

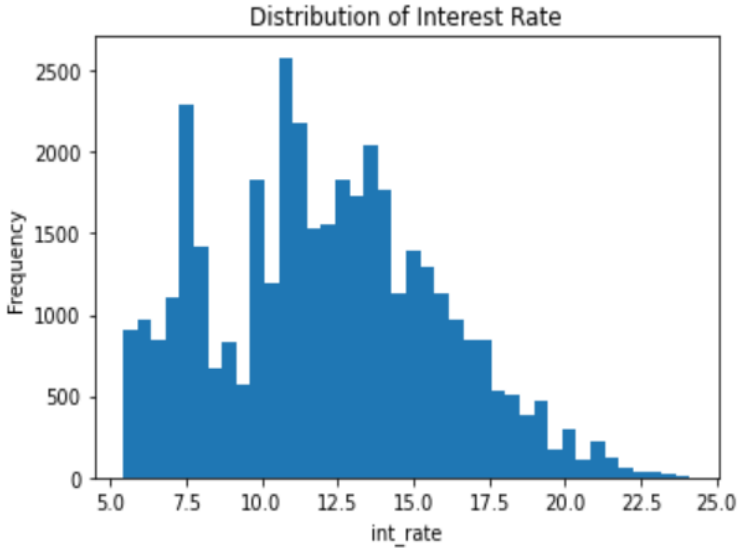
8.3.1 Loan Amount Distribution

Most loans are concentrated in small to mid-amounts. Large loans are fewer.



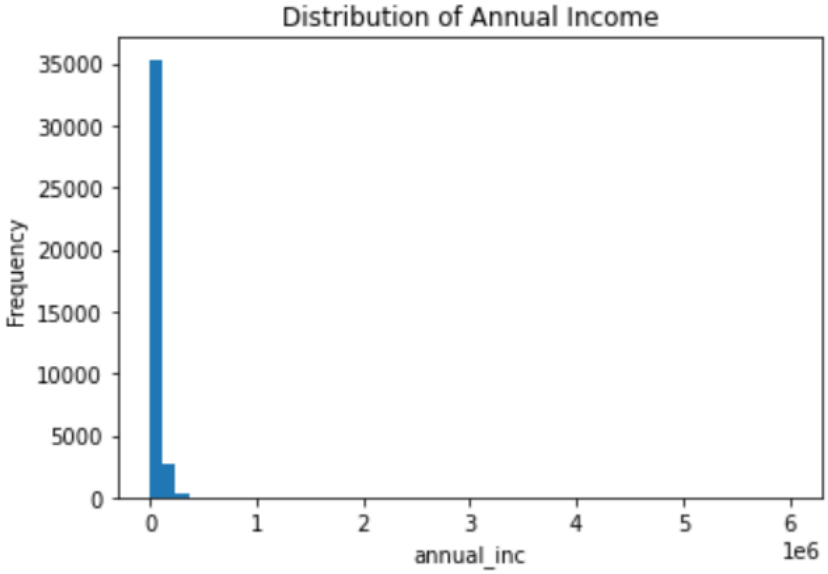
8.3.2 Interest Rate Distribution

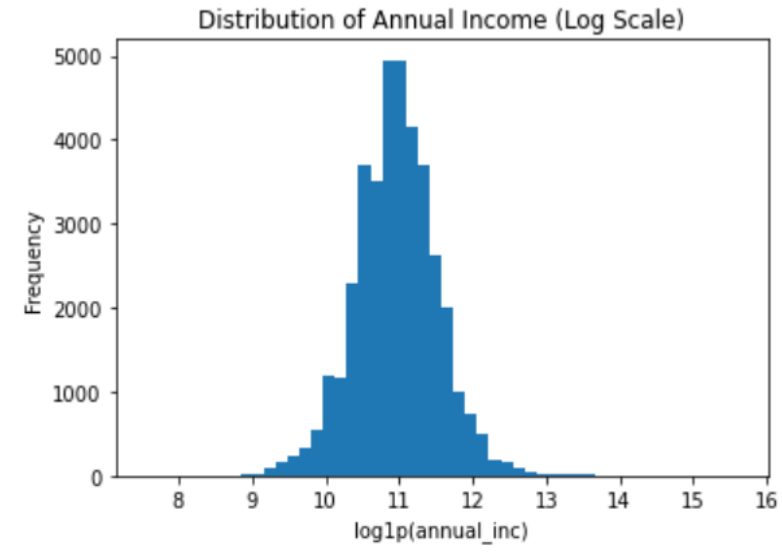
Interest rates cluster around mid values, with tail toward high-risk borrowers.



8.3.3 Annual Income Distribution (Raw + Log)

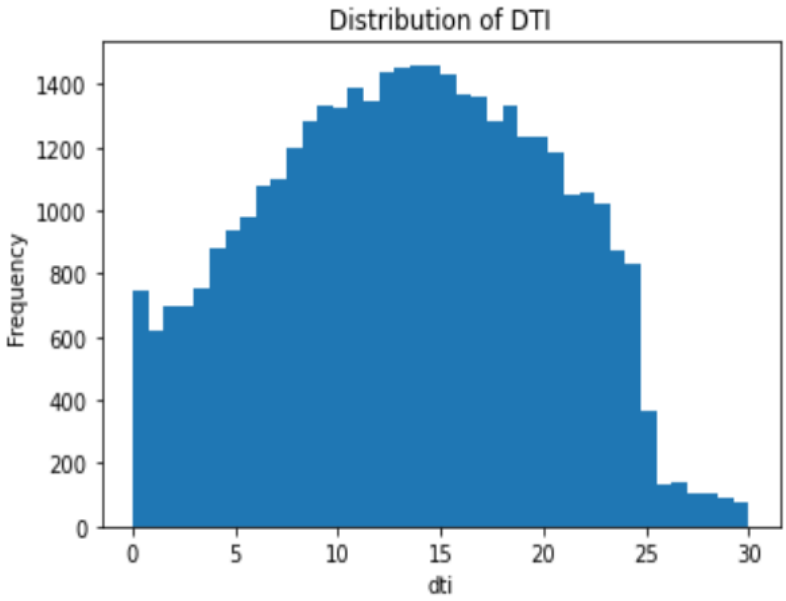
Income is highly skewed and log transformation improves interpretability.





8.3.4 DTI Distribution

DTI is spread between 0–30, showing affordability differences.



8.4 Default vs Non-default Comparison (Means)

Group-wise averages were compared between repay_fail groups.

Major insight: Defaults show:

- higher interest rate
- higher revolving utilization
- slightly higher loan amount
- lower income
- higher DTI

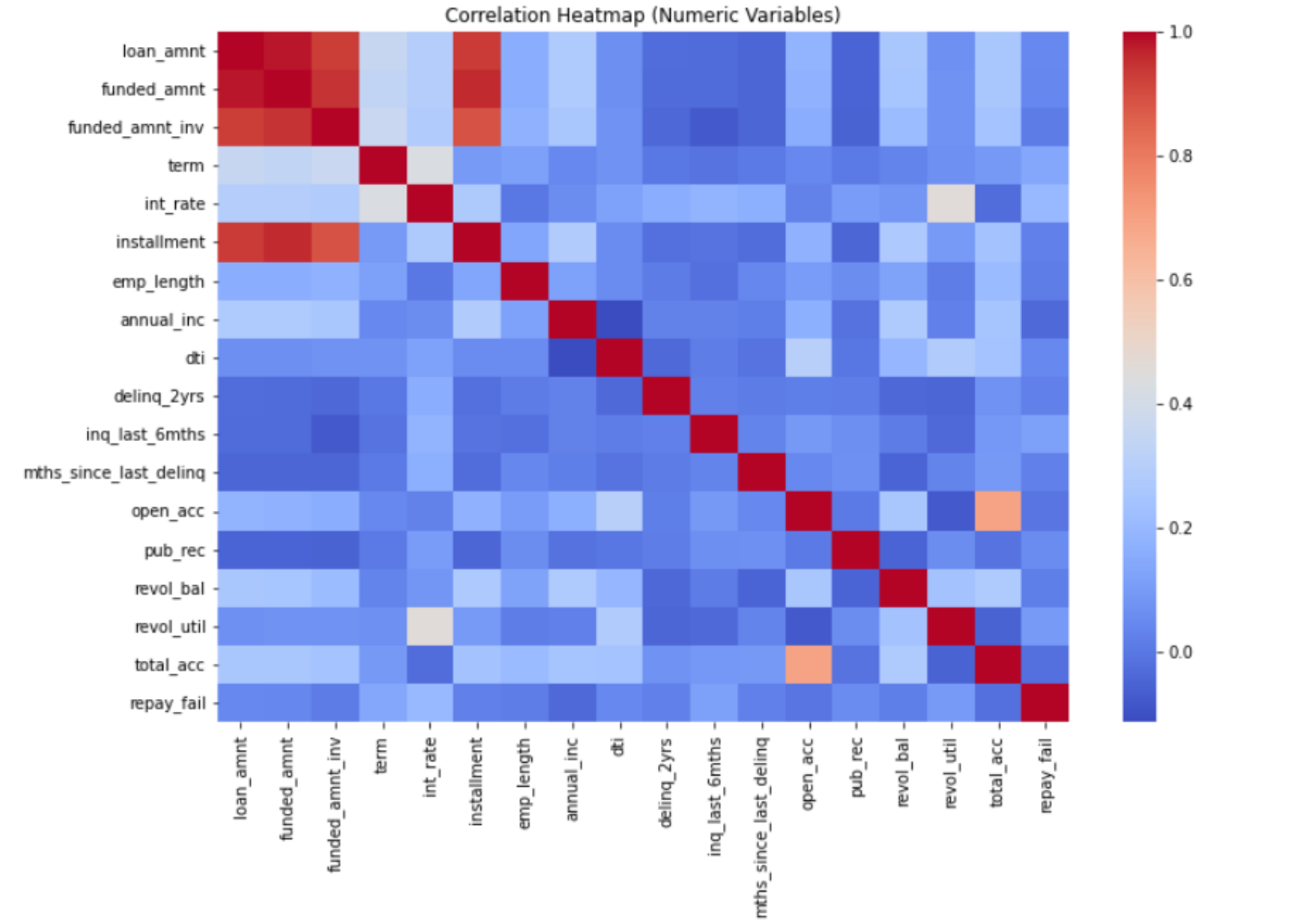
repay_fail	loan_amnt	int_rate	installment	annual_inc	dti	revol_util
0	10962.767245	11.847119	321.349152	70029.919787	13.255933	47.978027
1	11840.353587	13.927896	333.501819	63217.092909	14.053232	55.480614

8.5 Correlation Analysis

A correlation heatmap was created for numeric variables.

Key insight:

- int_rate showed the strongest correlation with repay_fail
- term + inquiries + revolving utilization also aligned with default risk



8.6 Hypothesis Testing

T-tests were performed to statistically validate differences between defaulters and non-defaulters.

✅ Key insight:
Key financial variables show statistically significant differences, strengthening confidence in observed patterns.

	feature	mean_default	mean_non_default	p_value
2	int_rate	13.927896	11.847119	0.000000e+00
6	revol_util	55.480614	47.978027	1.007082e-77
0	loan_amnt	11840.353587	10962.767245	7.680945e-17
5	dti	14.053232	13.255933	7.834032e-17
1	funded_amnt	11501.630621	10713.391326	8.642947e-15
4	annual_inc	63217.092909	70029.919787	1.071906e-13
3	installment	333.501819	321.349152	4.368535e-05

8.7 Logistic Regression (Statistical Validation)

A logistic regression model was fitted as a statistical interpretability step.

Key insight:

- interest rate is the strongest driver
- annual income reduces risk
- term increases risk
- revolving utilization contributes positively to default likelihood

```
Optimization terminated successfully.
         Current function value: 0.401292
         Iterations 6
                           Logit Regression Results
==============================================================================
Dep. Variable:              repay_fail   No. Observations:                38474
Model:                           Logit   Df Residuals:                    38467
Method:                            MLE   Df Model:                            6
Date:                 Tue, 20 Jan 2026   Pseudo R-squ.:                 0.05617
Time:                         16:32:56   Log-Likelihood:                -15439.
converged:                       True   LL-Null:                       -16358.
Covariance Type:            nonrobust   LLR p-value:                     0.000
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const          -3.8488      0.074    -52.286      0.000      -3.993      -3.705
int_rate        0.1362      0.005     26.496      0.000       0.126       0.146
dti             0.0037      0.002      1.601      0.109      -0.001       0.008
revol_util      0.0015      0.001      2.494      0.013       0.000       0.003
annual_inc  -4.757e-06   4.37e-07    -10.873      0.000   -5.61e-06    -3.9e-06
term            0.0137      0.001      9.513      0.000       0.011       0.017
installment    -0.0001   8.02e-05     -1.534      0.125      -0.000    3.42e-05
==============================================================================
```

	feature	coef	odds_ratio	p_value
1	int_rate	0.136242	1.145959	1.074406e-154
5	term	0.013727	1.013822	1.861065e-21
2	dti	0.003695	1.003702	1.093377e-01
3	revol_util	0.001502	1.001503	1.264889e-02
4	annual_inc	-0.000005	0.999995	1.545017e-27
6	installment	-0.000123	0.999877	1.251321e-01
0	const	-3.848775	0.021306	0.000000e+00

8.8 Final EDA Summary

This EDA confirms that default is driven by a combination of:

- pricing risk (interest rate)
- affordability (income + DTI)
- credit behaviour stress (revol_util, inquiries)
- loan structure (term)

CHAPTER 9: POWER BI DASHBOARDING AND BUSINESS STORYTELLING

9.1 Objective of Dashboarding

The Power BI dashboard was created to convert insights into an interactive business report, enabling stakeholders to:

- monitor portfolio risk
- compare segments using slicers
- identify high-risk borrower groups
- observe where most defaults come from

9.2 Dashboard Page 1: Portfolio Overview

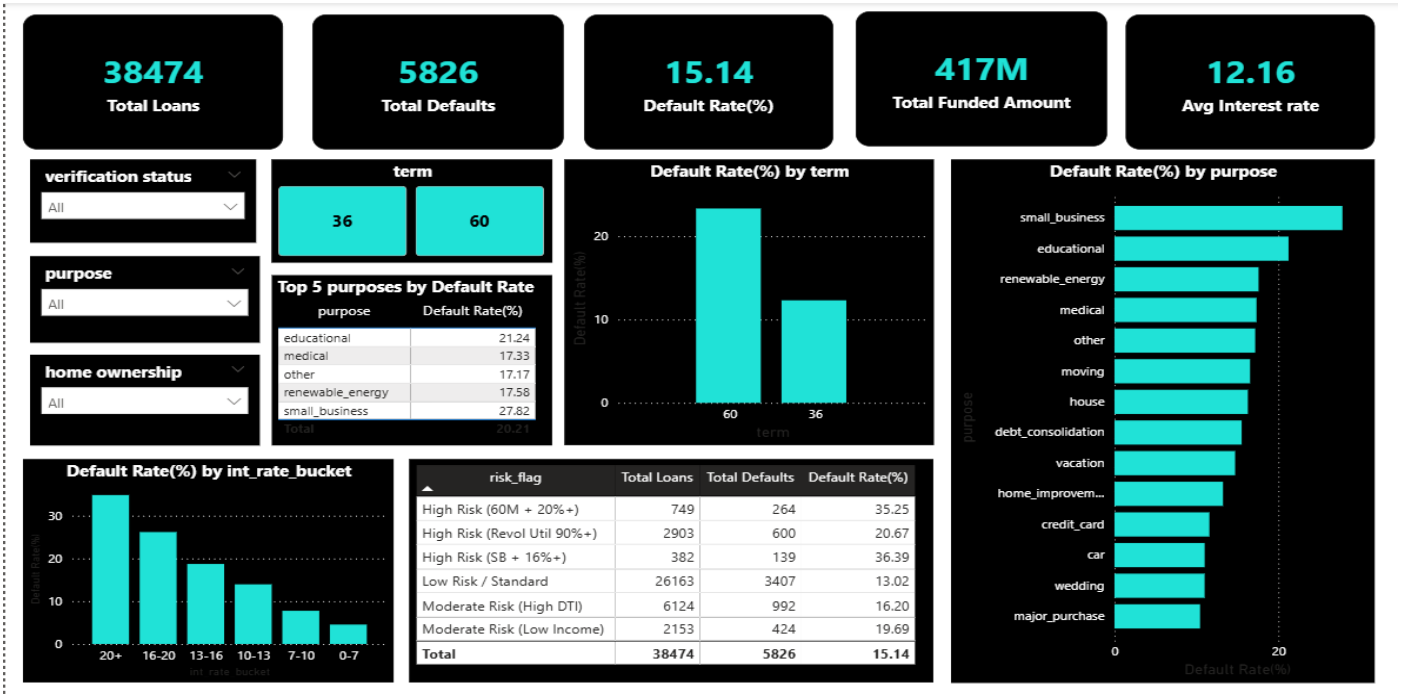
This page summarizes the portfolio through KPIs and top-level risk slices:

Included visuals:

- Total Loans (card)
- Total Defaults (card)
- Default Rate % (card)
- Total Funded Amount (card)
- Avg Interest Rate (card)

Risk charts:

- Default Rate by Term
- Default Rate by Purpose
- Default Rate by Interest Rate Bucket
- Risk Flag Table

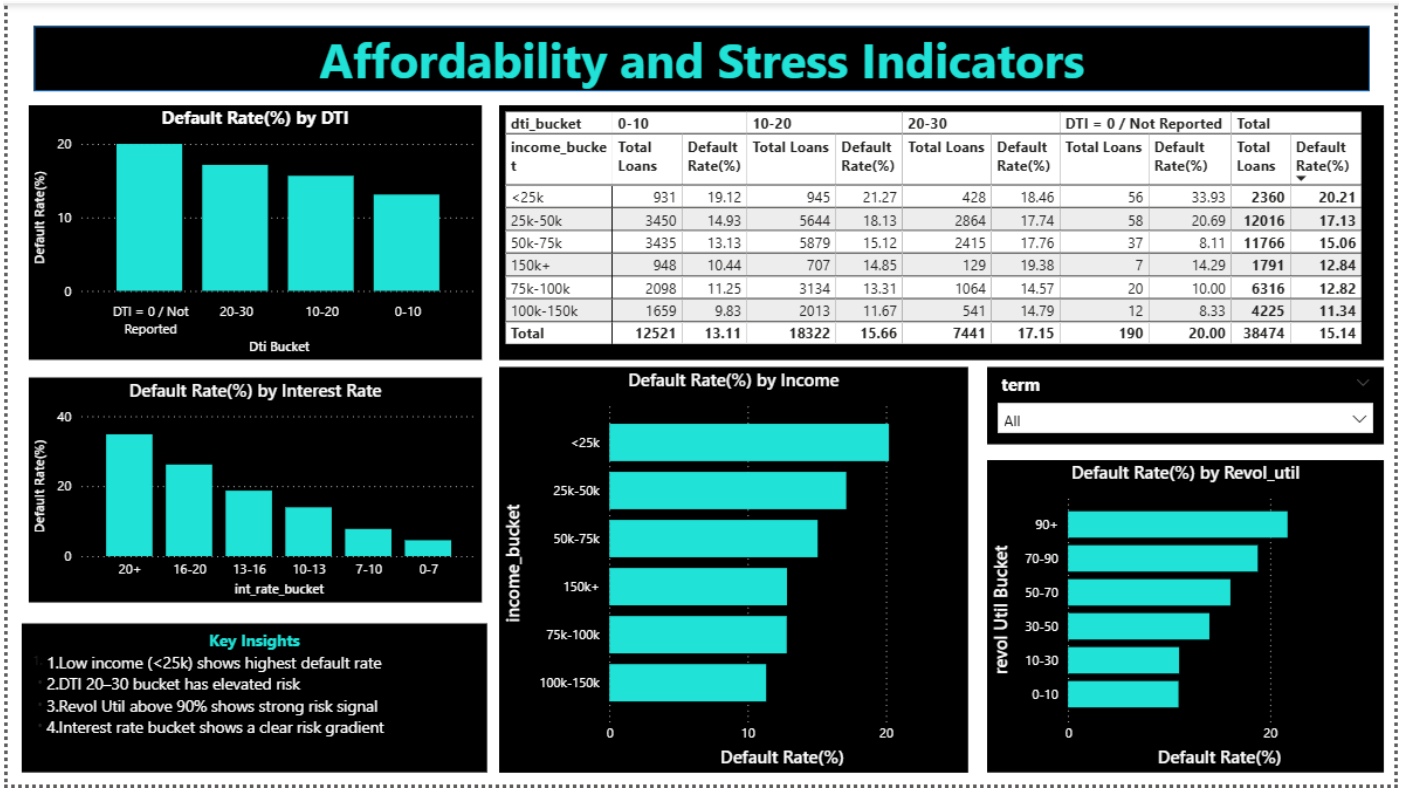


9.3 Dashboard Page 2: Affordability & Stress Indicators

This page focuses on affordability-related risk factors:

Visuals:

- Default Rate by Income Bucket
- Default Rate by DTI Bucket
- Default Rate by Revol Util Bucket
- Default Rate by Interest Rate Bucket
- Matrix (Income × DTI showing loans + default rates)



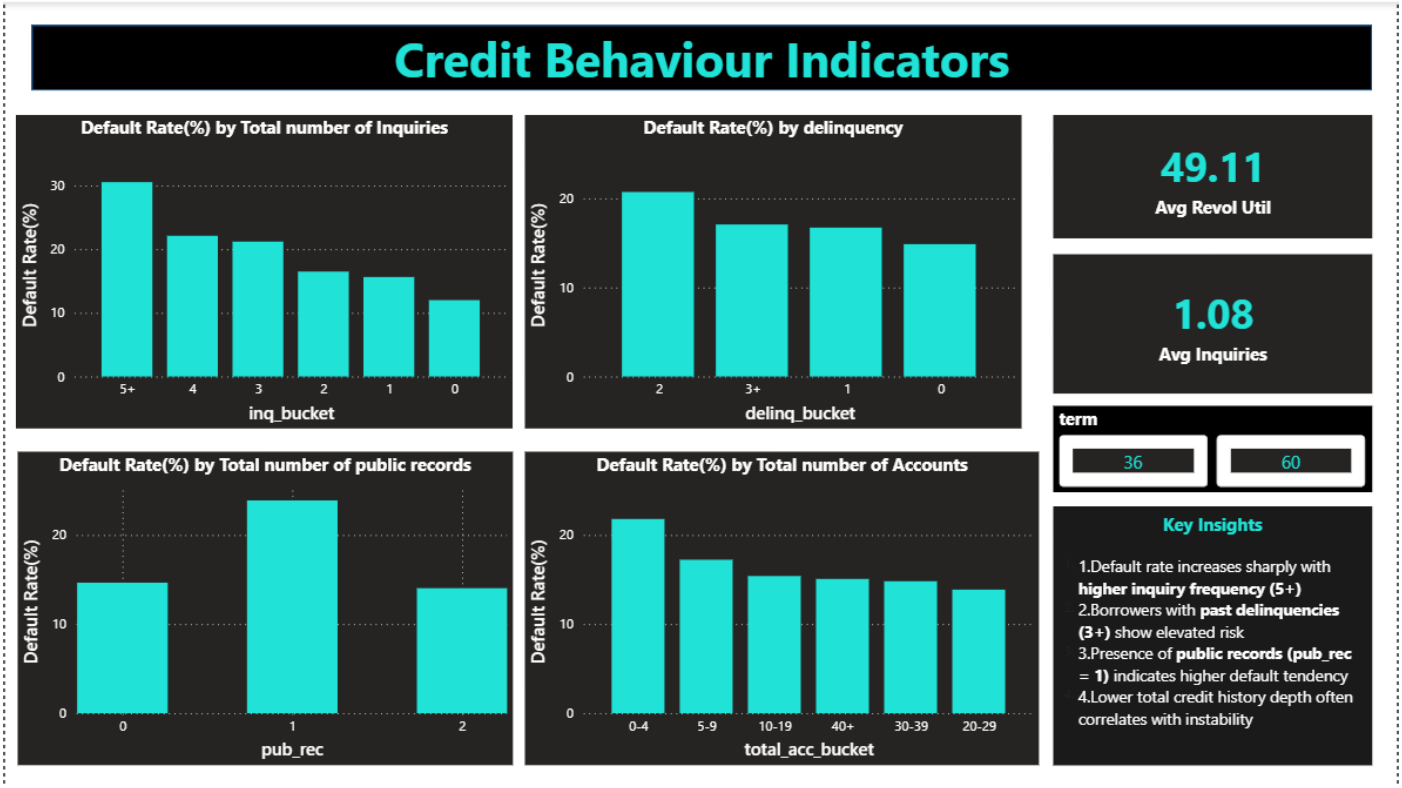
dti_bucket	0-10		10-20		20-30		DTI = 0 / Not Reported		Total	
income_bucket	Total Loans	Default Rate(%)	Total Loans	Default Rate(%)	Total Loans	Default Rate(%)	Total Loans	Default Rate(%)	Total Loans	Default Rate(%)
<25k	931	19.12	945	21.27	428	18.46	56	33.93	**2360**	**20.21**
25k-50k	3450	14.93	5644	18.13	2864	17.74	58	20.69	**12016**	**17.13**
50k-75k	3435	13.13	5879	15.12	2415	17.76	37	8.11	**11766**	**15.06**
150k+	948	10.44	707	14.85	129	19.38	7	14.29	**1791**	**12.84**
75k-100k	2098	11.25	3134	13.31	1064	14.57	20	10.00	**6316**	**12.82**
100k-150k	1659	9.83	2013	11.67	541	14.79	12	8.33	**4225**	**11.34**
Total	**12521**	**13.11**	**18322**	**15.66**	**7441**	**17.15**	**190**	**20.00**	**38474**	**15.14**

9.4 Dashboard Page 3: Credit Behaviour Indicators

This page evaluates borrower credit activity and risk behaviour signals:

Visuals:

- Default Rate by Inquiry Bucket
- Default Rate by Delinquency Bucket
- Default Rate by Public Records (pub_rec)
- Default Rate by Total Accounts Bucket
- Avg Revol Util card
- Key Insights box



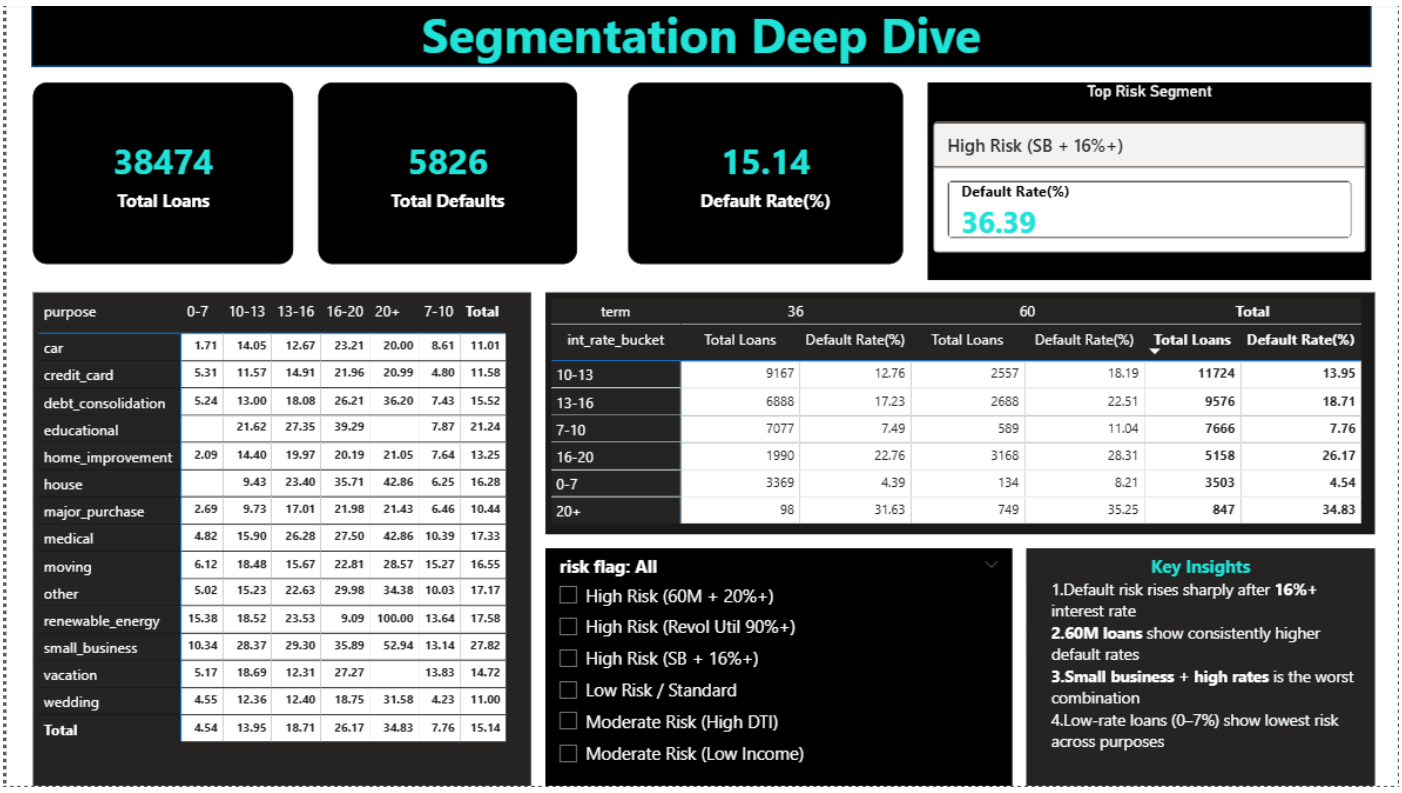
9.5 Dashboard Page 4: Segmentation Deep Dive

This page provides deeper portfolio segmentation using matrices:

Matrix 1:
Term × Interest Rate Bucket → Total Loans + Default Rate %

Matrix 2:
Purpose × Interest Rate Bucket → Default Rate %

Slicer:
Risk flag



purpose	0-7	10-13	13-16	16-20	20+	7-10	Total
car	1.71	14.05	12.67	23.21	20.00	8.61	11.01
credit_card	5.31	11.57	14.91	21.96	20.99	4.80	11.58
debt_consolidation	5.24	13.00	18.08	26.21	36.20	7.43	15.52
educational		21.62	27.35	39.29		7.87	21.24
home_improvement	2.09	14.40	19.97	20.19	21.05	7.64	13.25
house		9.43	23.40	35.71	42.86	6.25	16.28
major_purchase	2.69	9.73	17.01	21.98	21.43	6.46	10.44
medical	4.82	15.90	26.28	27.50	42.86	10.39	17.33
moving	6.12	18.48	15.67	22.81	28.57	15.27	16.55
other	5.02	15.23	22.63	29.98	34.38	10.03	17.17
renewable_energy	15.38	18.52	23.53	9.09	100.00	13.64	17.58
small_business	10.34	28.37	29.30	35.89	52.94	13.14	27.82
vacation	5.17	18.69	12.31	27.27		13.83	14.72
wedding	4.55	12.36	12.40	18.75	31.58	4.23	11.00
Total	4.54	13.95	18.71	26.17	34.83	7.76	15.14

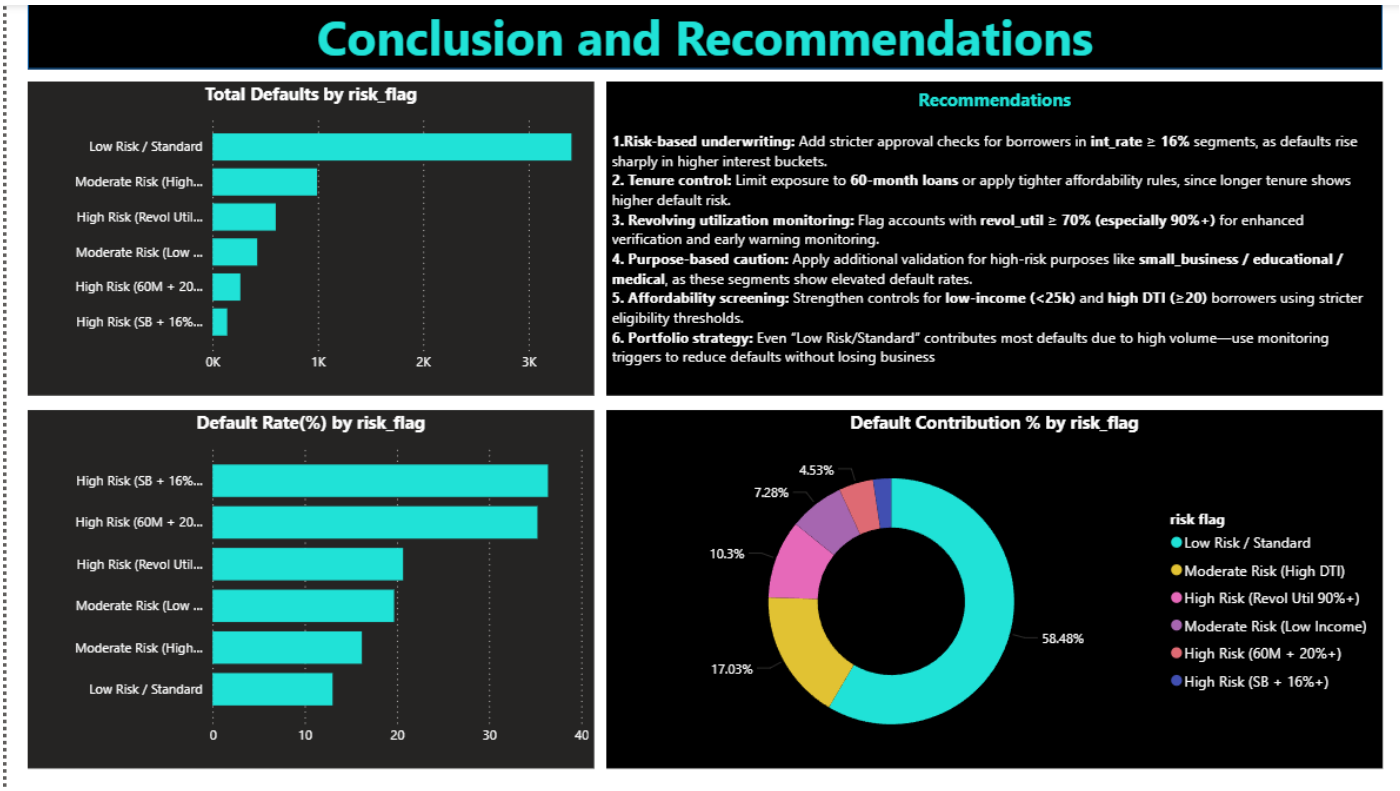
term	36		60		Total	
int_rate_bucket	Total Loans	Default Rate(%)	Total Loans	Default Rate(%)	Total Loans	Default Rate(%)
10-13	9167	12.76	2557	18.19	11724	13.95
13-16	6888	17.23	2688	22.51	9576	18.71
7-10	7077	7.49	589	11.04	7666	7.76
16-20	1990	22.76	3168	28.31	5158	26.17
0-7	3369	4.39	134	8.21	3503	4.54
20+	98	31.63	749	35.25	847	34.83

9.6 Dashboard Page 5: Final Findings & Recommendations

This page summarizes outcomes in an executive format.

Visuals:

- Total Defaults by Risk Flag
- Default Rate by Risk Flag
- Default Contribution % by Risk Flag (donut)
- Recommendations text box



FINAL NOTE

This dashboard bridges analytical findings with business action, enabling non-technical stakeholders to quickly identify high-risk segments, monitor risk distribution, and prioritize risk-control strategies based on both default rate and default contribution.