Exploratory Data Analysis Report (H2)

Your Name

2025-12-25

Contents

1. Introduction

This report presents an Exploratory Data Analysis (EDA) for H2 of the prepared loan dataset used for graph-based default prediction.

Objectives:

- Understand feature distributions and correlations
- Assess predictive signal with respect to the target (Target_Y)
- Inspect temporal trends across the modeling window
- Validate assumptions used later in graph construction (group sizes)

All analyses are performed only on training-period data unless explicitly stated.

1.1 Dependencies and Data Loading

This section loads final, cleaned artifacts produced by the data pipeline. The structure mirrors the validation scripts to ensure consistency.

```
# --- DEPENDENCIES ---
source("00_config.R")
library(data.table)
library(lubridate)
library(ggplot2)
library(corrplot)

# --- LOAD PROCESSED ARTIFACTS ---
cat("--- LOADING ALL SAVED FILES FOR EDA ---\n")
```

```
## --- LOADING ALL SAVED FILES FOR EDA ---
```

```
files_to_load <- list(
  final_features = file.path(SAVE_DIR, "final_features_raw_clean.rds"),
  final_with_targets = file.path(SAVE_DIR, "final_data_base_with_targets.rds"),
  train_targets = file.path(SAVE_DIR, "train_targets.rds")
)

loaded_data_eda <- list()
all_files_loaded_eda <- TRUE
for (name in names(files_to_load)) {
  file_path <- files_to_load[[name]]

  if (file.exists(file_path)) {
    loaded_data_eda[[name]] <- readRDS(file_path)
    # Ensure all loaded objects are data.tables
    setDT(loaded_data_eda[[name]])
    cat(sprintf("%s loaded successfully\n", name))
  } else {
    cat(sprintf("%s NOT FOUND at %s\n", name, file_path))
    all_files_loaded_eda <- FALSE
  }
}
```

```
## final_features loaded successfully
## final_with_targets loaded successfully
## train_targets loaded successfully
```

```
if (!all_files_loaded_eda) {
  stop("Critical data files are missing for EDA. Halting report generation.")
}

# --- PREPARE TRAINING SUBSET ---
# We analyze correlations ONLY on the training set to respect the temporal split.
TRAIN_SPLIT_DATE <- as.integer(format(ymd(paste0(START_PERIOD_TRAIN, "01")) %m+% months(17), "%Y%m"))

if (!is.null(loaded_data_eda$final_with_targets) && !is.null(loaded_data_eda$train_targets)) {
  train_data_for_eda <- loaded_data_eda$final_with_targets[
```

```
    Monthly_Reporting_Period %in% unique(loaded_data_eda$train_targets$Snapshot_Date)
  ]
  train_data_for_eda[, Target_Y := as.factor(Target_Y)]
} else {
  warning("Not all data required for EDA (final_with_targets, train_targets) could be loaded.")
  train_data_for_eda <- NULL
}
```

2. Feature Correlation Analysis

To assess the stability of the feature space for subsequent **Graph Neural Network (GNN)** construction, we examined pairwise correlations among continuous predictors. The goal is to identify redundancy and potential sources of instability in gradient-based learning.

Observation 1: Original vs. Current Unpaid Balance

The correlation matrix reveals near-perfect linear dependence ($\rho = 0.91$) between **orig_upb** (Original Unpaid Balance) and **current_upb** (Current Unpaid Balance).

Rationalization. This relationship is structurally expected rather than coincidental. The current unpaid balance is a deterministic function of the original balance, interest rate, and loan age via the amortization schedule. Given that the observation window (2012–2014) captures the early life of these mortgages, when principal reduction is minimal, the outstanding balance remains closely tied to the original balance. Including both variables would introduce redundant information and may destabilize gradient-based optimization during model training. Consequently, feature selection is required to remove one of these variables.

Observation 2: Interest Rate and Remaining Loan Term

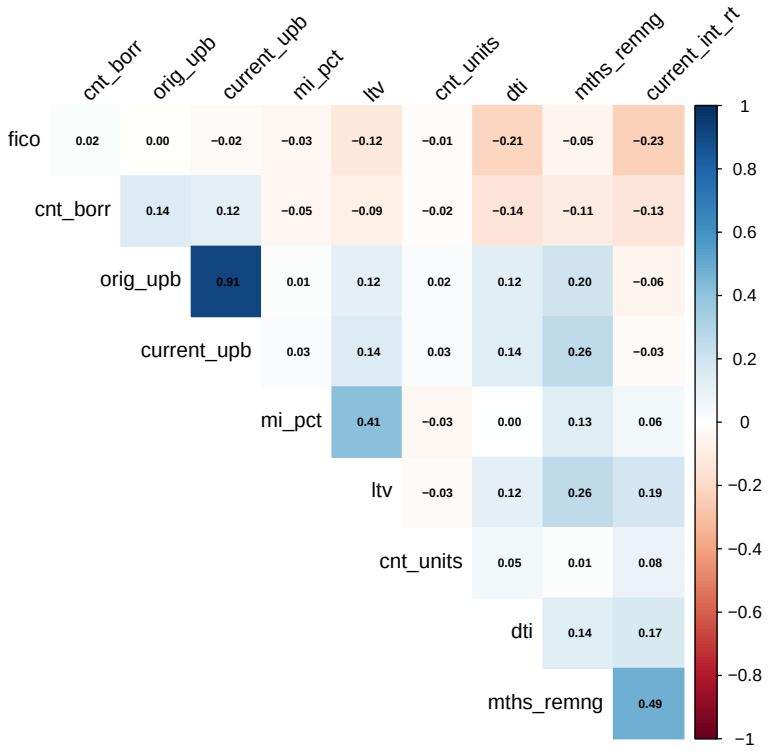
A moderate positive correlation ($\rho = 0.49$) is observed between **current_int_rt** (Current Interest Rate) and **mths_remng** (Months Remaining).

Rationalization. This pattern reflects the term structure of lending products. Loans with longer maturities typically carry higher interest rates to compensate lenders for increased duration risk and exposure to inflation uncertainty over extended horizons. The observed correlation therefore aligns with economic theory and does not indicate problematic multicollinearity.

```
if (!is.null(train_data_for_eda)) {
  # Select only numeric columns for correlation
  numeric_cols_for_corr <- intersect(ALL_NUMERIC_COLS_FINAL, names(train_data_for_eda))
  numeric_data_subset <- train_data_for_eda[, ..numeric_cols_for_corr]

  cat("Calculating correlation matrix...\n")
  cor_matrix <- cor(numeric_data_subset, use = "pairwise.complete.obs")

  cat("\nCorrelation Matrix (first 10x10, rounded to 2 decimal places):\n")
  print(round(head(cor_matrix, 10), 2))

  cat("\nGenerating correlation plot...\n")
  corrplot(cor_matrix, method = "color", type = "upper", order = "hclust",
           tl.col = "black", tl.srt = 45, diag = FALSE,
           addCoef.col = "black", number.cex = 0.6)
```

```
} else {
  cat("Skipping correlation analysis as training data is not available.\n")
}
```

```
## Calculating correlation matrix...
##
## Correlation Matrix (first 10x10, rounded to 2 decimal places):
##                 fico mi_pct cnt_units   dti   ltv cnt_borr orig_upb current_upb
## fico            1.00  -0.03     -0.01 -0.21 -0.12     0.02     0.00       -0.02
## mi_pct         -0.03   1.00     -0.03  0.00  0.41    -0.05     0.01        0.03
## cnt_units      -0.01  -0.03      1.00  0.05 -0.03    -0.02     0.02        0.03
## dti            -0.21   0.00      0.05  1.00  0.12    -0.14     0.12        0.14
## ltv            -0.12   0.41     -0.03  0.12  1.00    -0.09     0.12        0.14
## cnt_borr        0.02  -0.05     -0.02 -0.14 -0.09     1.00     0.14        0.12
## orig_upb        0.00   0.01      0.02  0.12  0.12     0.14     1.00        0.91
## current_upb    -0.02   0.03      0.03  0.14  0.14     0.12     0.91        1.00
## mths_remng     -0.05   0.13      0.01  0.14  0.26    -0.11     0.20        0.26
## current_int_rt -0.23   0.06      0.08  0.17  0.19    -0.13    -0.06       -0.03
##                mths_remng current_int_rt
## fico                -0.05          -0.23
## mi_pct               0.13           0.06
## cnt_units            0.01           0.08
## dti                  0.14           0.17
## ltv                  0.26           0.19
## cnt_borr            -0.11          -0.13
## orig_upb             0.20          -0.06
## current_upb          0.26          -0.03
## mths_remng           1.00           0.49
## current_int_rt       0.49           1.00
##
## Generating correlation plot...
```



3. Univariate Predictive Power

Evaluated the discriminatory power of standard credit risk drivers against the binary target **Target_Y** (Default).

Observation 1: FICO Score

FICO scores demonstrate the strongest separation between classes. The distribution for defaulters exhibits a clear downward shift, with a median score of approximately 720, compared to a median of approximately 770 for non-defaulters.

Rationalisation. This finding supports the Credit History Hypothesis, according to which past payment behaviour is the most reliable indicator of future delinquency. The pronounced separation between classes confirms the high signal quality of credit bureau information in the H2 sample and validates its inclusion as a core predictive feature.

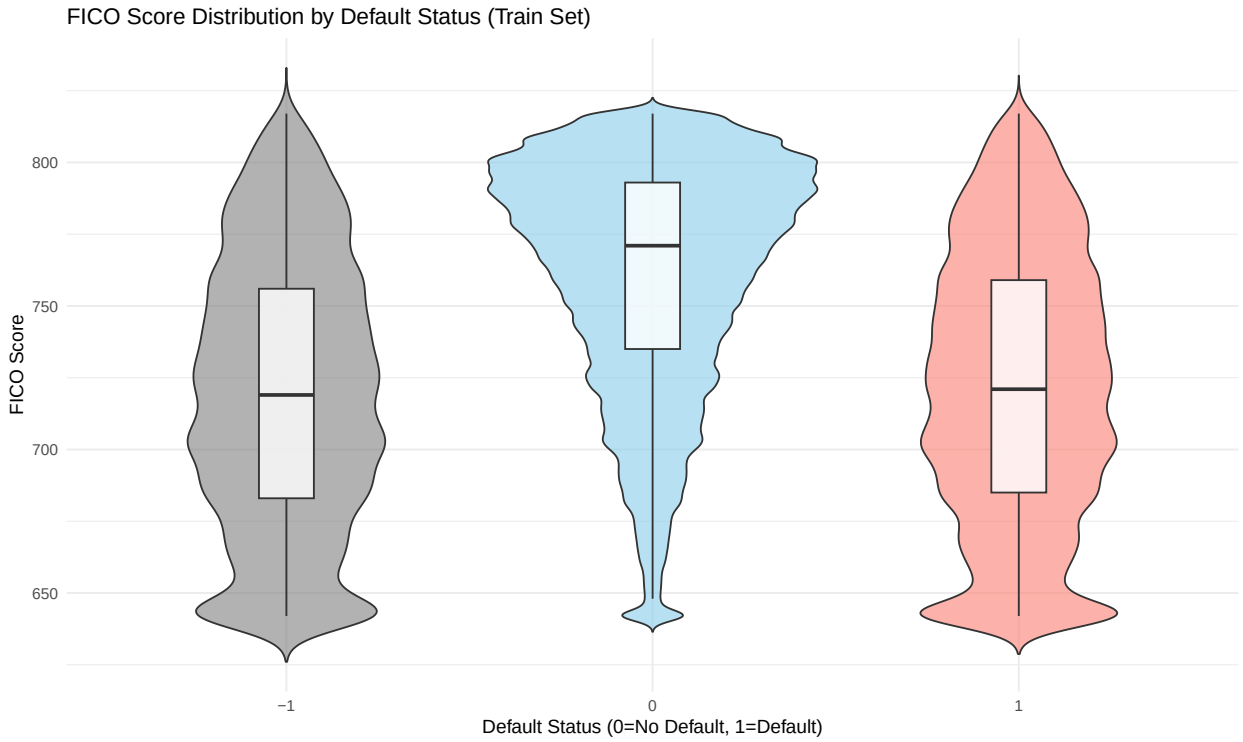
Observation 2: Loan-to-Value Ratio (LTV)

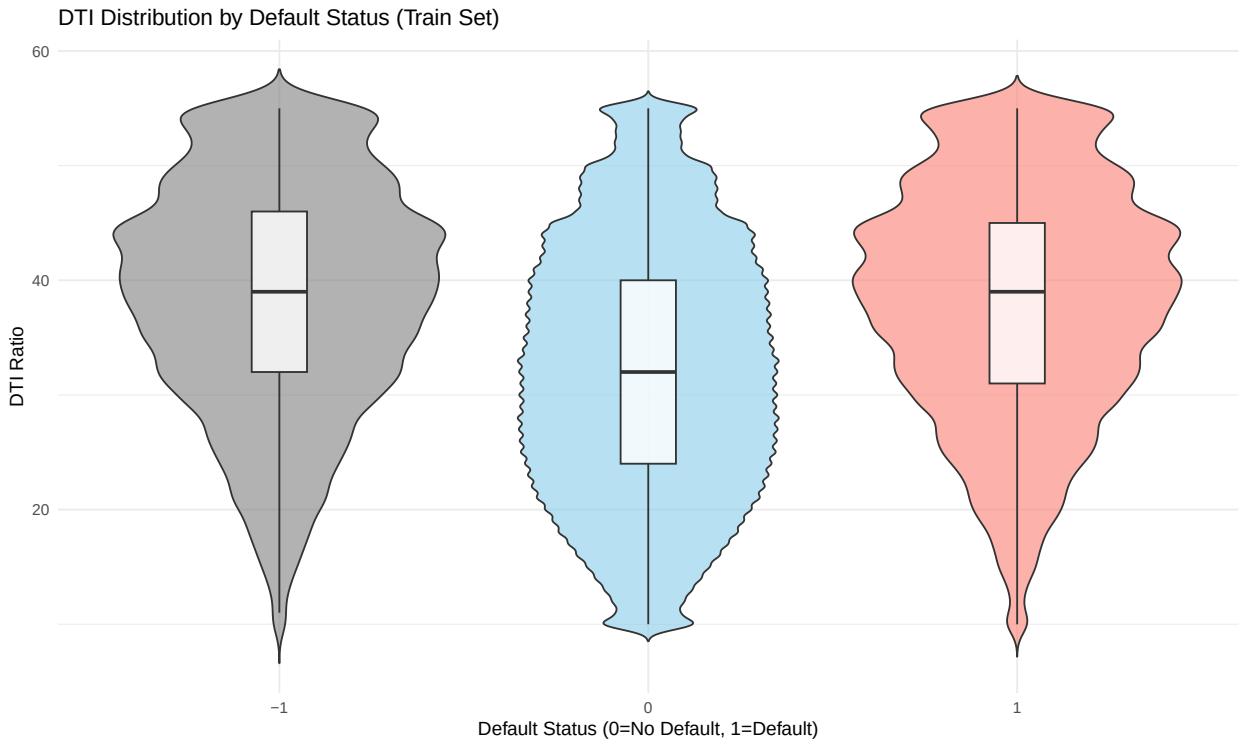
Loan-to-Value ratios show negligible separation between defaulters and non-defaulters. The corresponding boxplots display nearly identical interquartile ranges, indicating limited discriminatory power.

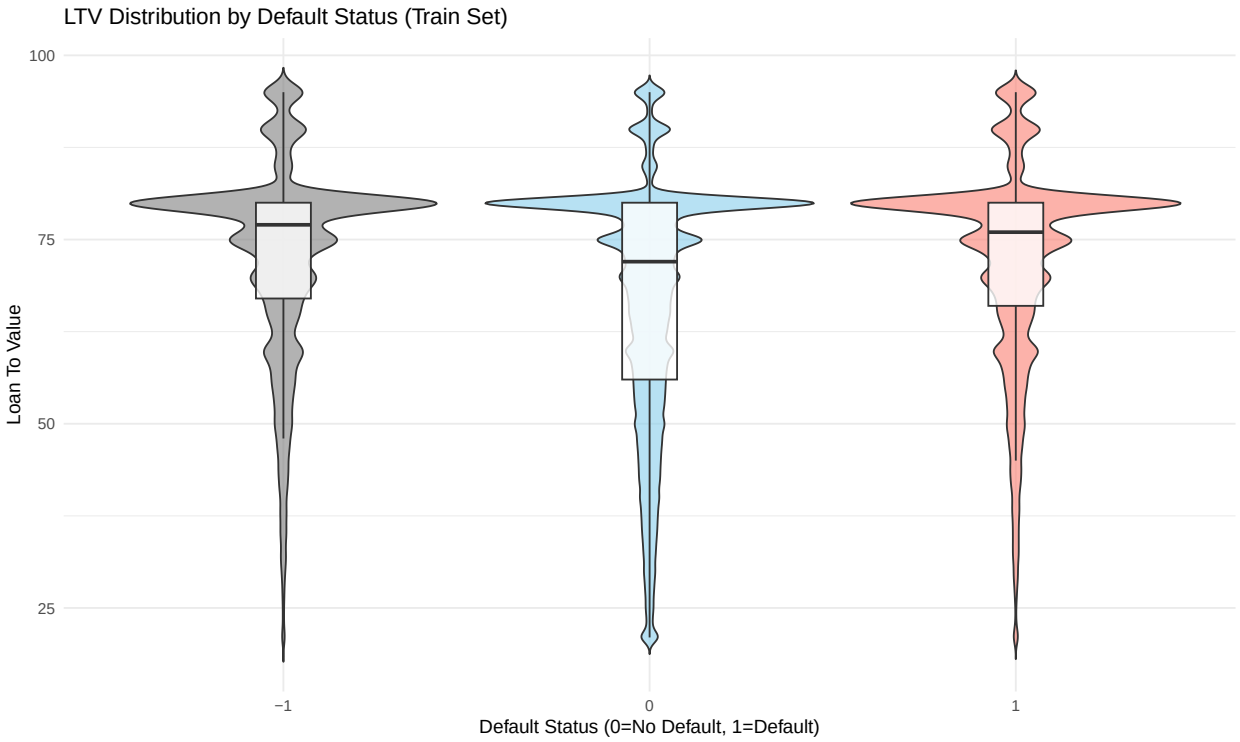
Rationalisation. The absence of a strong signal is attributable to underwriting standardisation. Government-Sponsored Enterprise loans are typically subject to strict LTV caps at origination, commonly around 80 percent, in order to qualify for purchase. This truncation compresses the LTV distribution and removes much of its variance, limiting its usefulness as a discriminator within this dataset despite its theoretical relevance in credit risk modelling.

```r
if (!is.null(train_data_for_eda)) {
  cat("Generating box plots for key features vs. Target_Y...\n")

  plot_data <- train_data_for_eda[!is.na(Target_Y)]
  plot_data[, Target_Y := as.factor(Target_Y)]

  # FICO
  plot_fico <- ggplot(plot_data, aes(x = Target_Y, y = fico, fill = Target_Y)) +
    geom_violin(trim = FALSE, alpha = 0.6) +
    geom_boxplot(width = 0.15, fill = "white", alpha = 0.8, outlier.shape = NA) +
    labs(title = "FICO Score Distribution by Default Status (Train Set)",
         x = "Default Status (0=No Default, 1=Default)", y = "FICO Score") +
    scale_fill_manual(values = c("0" = "skyblue", "1" = "salmon")) +
    theme_minimal() +
    theme(legend.position = "none")
  print(plot_fico)

  # DTI
  plot_dti <- ggplot(plot_data, aes(x = Target_Y, y = dti, fill = Target_Y)) +
    geom_violin(trim = FALSE, alpha = 0.6) +
    geom_boxplot(width = 0.15, fill = "white", alpha = 0.8, outlier.shape = NA) +
    labs(title = "DTI Distribution by Default Status (Train Set)",
         x = "Default Status (0=No Default, 1=Default)", y = "DTI Ratio") +
    scale_fill_manual(values = c("0" = "skyblue", "1" = "salmon")) +
    theme_minimal() +
    theme(legend.position = "none")
  print(plot_dti)

  # LTV
  plot_oltv <- ggplot(plot_data, aes(x = Target_Y, y = ltv, fill = Target_Y)) +
    geom_violin(trim = FALSE, alpha = 0.6) +
    geom_boxplot(width = 0.15, fill = "white", alpha = 0.8, outlier.shape = NA) +
    labs(title = "LTV Distribution by Default Status (Train Set)",
         x = "Default Status (0=No Default, 1=Default)", y = "Loan To Value") +
    scale_fill_manual(values = c("0" = "skyblue", "1" = "salmon")) +
    theme_minimal() +
    theme(legend.position = "none")
  print(plot_oltv)

} else {
  cat("Skipping predictive power plots as training data is not available.\n")
}
```

```
## Generating box plots for key features vs. Target_Y...
```


FICO Score Distribution by Default Status (Train Set)
FICO Score
800
750
700
650
−1
0
1
Default Status (0=No Default, 1=Default)


DTI Distribution by Default Status (Train Set)
DTI Ratio
60
40
20
−1
0
1
Default Status (0=No Default, 1=Default)



4. Temporal Dynamics (2011 - 2014)

Time-series analysis was conducted to detect potential concept drift and to validate the temporal train–test split strategy. A counterintuitive relationship was observed between interest rates and default rates at the aggregate level.

Observation

The data exhibits a strong negative ecological correlation between the average interest rate and the observed default rate. Over the analysis window, the average interest rate decreases monotonically from approximately 4.78 percent to 4.76 percent, while the observed default rate increases from roughly 2.0 percent to 2.8 percent.

Rationalisation. This pattern reflects a temporal confounding artifact rather than a causal relationship. Two independent processes evolve simultaneously over the observation window.

Cohort maturation. As the loan portfolio progresses from 2012 to 2014, loans naturally age. Default is a time-dependent event, and early-life loans rarely default. As a result, the aggregate hazard rate increases as more loans enter higher-risk stages of their life cycle.

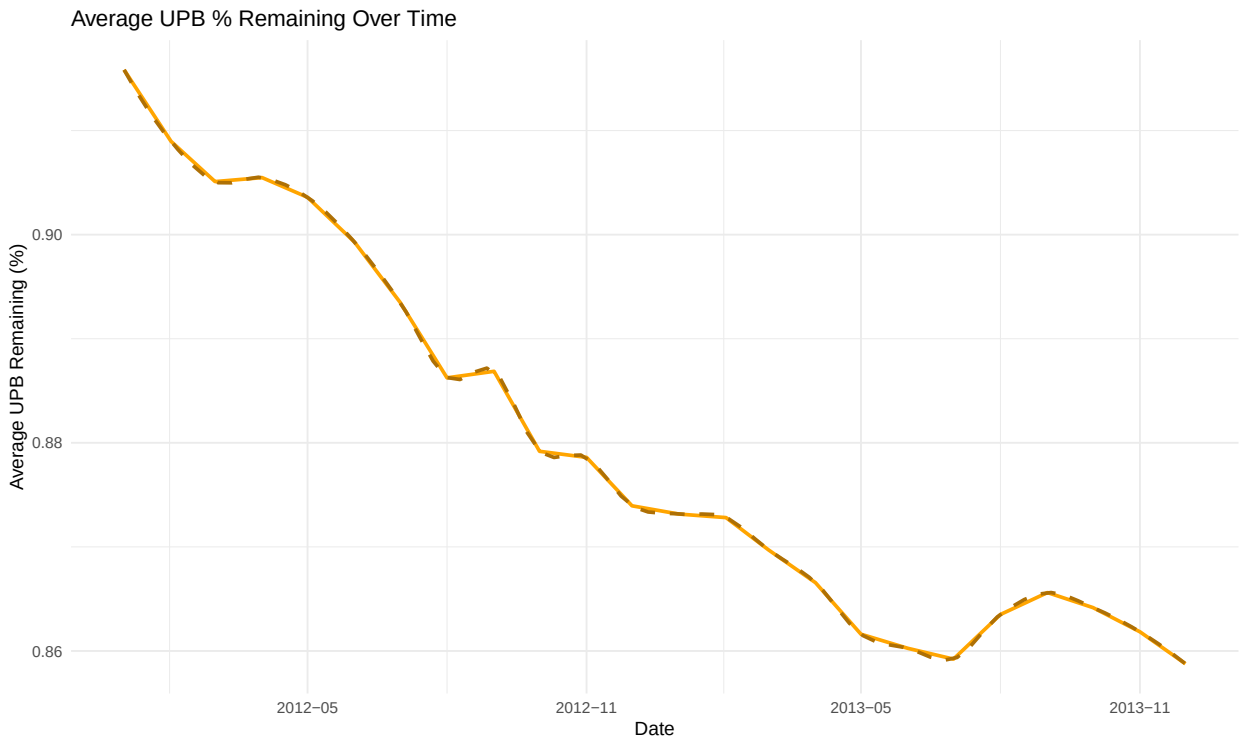
Macroeconomic trends. Over the same period, prevailing market interest rates decline modestly due to broader macroeconomic conditions. This reduction is exogenous to borrower-level default risk.

The observed negative correlation therefore arises because later time periods contain both lower interest rates and more mature, risk-prone loans. The implication for modeling is critical. The learning algorithm must distinguish temporal position and loan age effects from genuine causal drivers. Lower interest rates should not be interpreted as increasing default risk; instead, time and loan maturation must be explicitly modeled to avoid spurious inference.

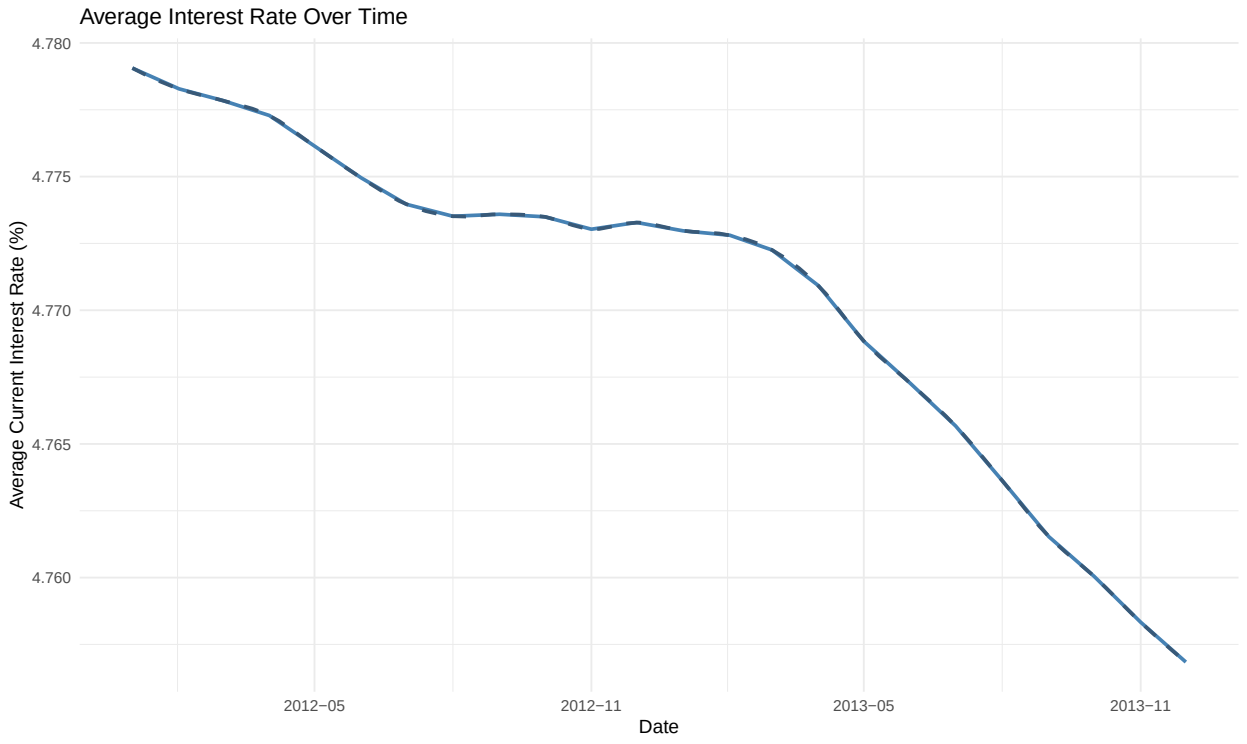
```r
library(colorspace)

FINAL_DATA_WITH_TARGETS_FILE <- file.path(SAVE_DIR, "final_data_base_with_targets.rds")

# --- Data Preparation ---
cat("--- Loading final data with targets for time-series EDA ---\n")
```

```
## --- Loading final data with targets for time-series EDA ---
```

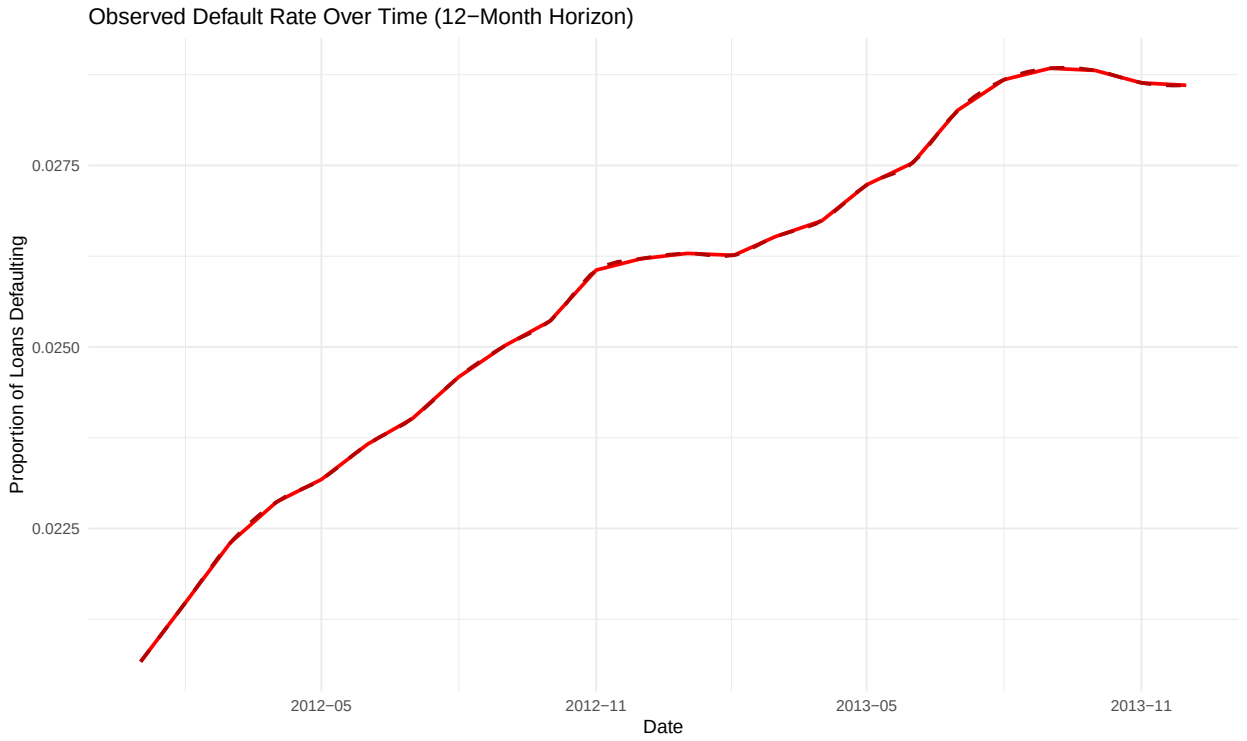
```r
if (!file.exists(FINAL_DATA_WITH_TARGETS_FILE)) {
  stop("ERROR: Cannot find the data file with targets. Run 03_prep_split_scale.R first.")
}

final_data <- readRDS(FINAL_DATA_WITH_TARGETS_FILE)
setDT(final_data)

final_data[, Date := ymd(paste0(Monthly_Reporting_Period, "01"))]

# Aggregation by Month
time_trends <- final_data[!is.na(Target_Y), .(
  avg_upb_pct        = mean(upb_pct_remaining, na.rm = TRUE),
  avg_int_rate       = mean(current_int_rt, na.rm = TRUE),
  avg_mths_remaining = mean(mths_remng, na.rm = TRUE),
  avg_loan_age       = mean(clean_loan_age, na.rm = TRUE),
  default_rate       = mean(Target_Y == 1, na.rm = TRUE)
), by = Date]

# --- Plotting Function ---
plot_ts <- function(df, yvar, color, title, ylab) {
  ggplot(df, aes(x = Date, y = get(yvar))) +
    geom_line(color = color, size = 1) +
    geom_smooth(method = "loess", span = 0.2, se = FALSE,
                color = darken(color, 0.3), linetype = "dashed") +
    labs(title = title, x = "Date", y = ylab) +
    scale_x_date(date_breaks = "6 months", date_labels = "%Y-%m") +
    theme_minimal()
}

# PLOTS
# --- 1. Average Unpaid Balance (UPB) ---
plot_ts(time_trends, "avg_upb_pct", "orange",
        "Average UPB % Remaining Over Time",
        "Average UPB Remaining (%)")
```



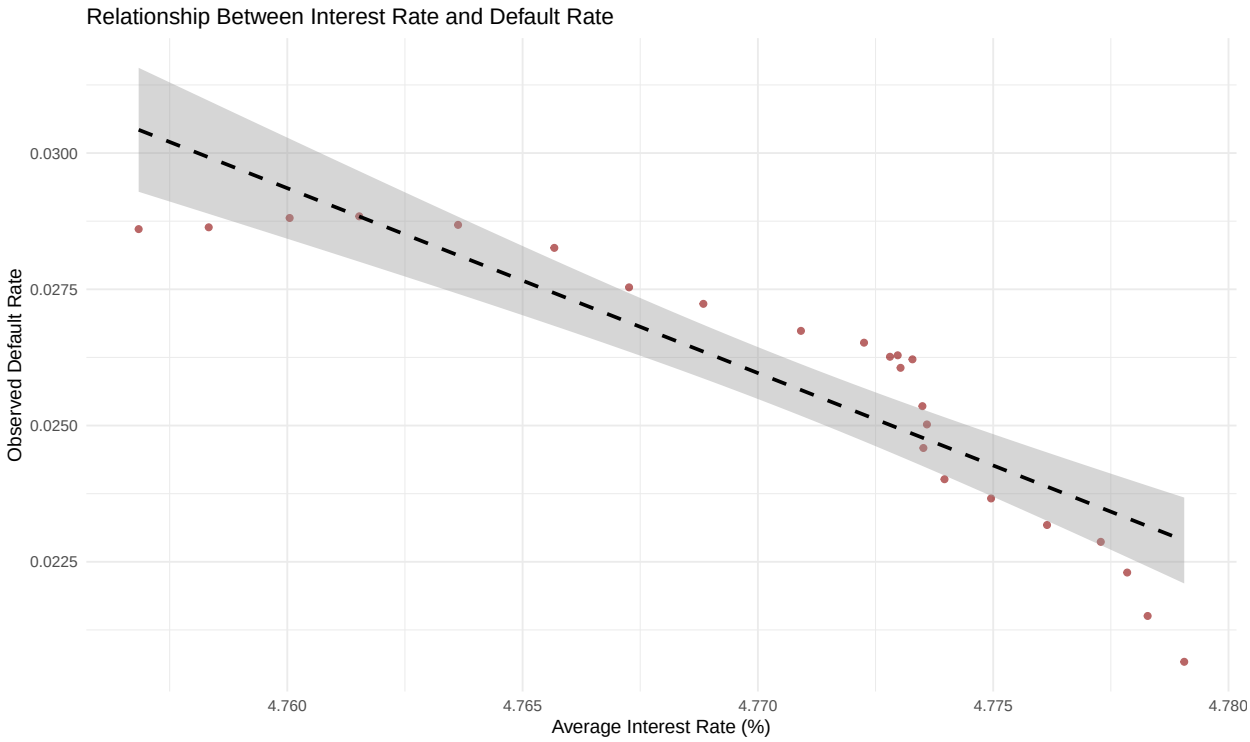

```
# --- 2. Average Interest Rate ---
plot_ts(time_trends, "avg_int_rate", "steelblue",
        "Average Interest Rate Over Time",
        "Average Current Interest Rate (%)")
```



```r
# --- 3. Default Rate ---
plot_ts(time_trends, "default_rate", "red",
        "Observed Default Rate Over Time (12-Month Horizon)",
        "Proportion of Loans Defaulting")
```




```r
# --- 4. Correlation between Interest Rate & Default Rate ---
ggplot(time_trends, aes(x = avg_int_rate, y = default_rate)) +
  geom_point(color = "darkred", alpha = 0.6) +
  geom_smooth(method = "lm", color = "black", linetype = "dashed") +
  labs(title = "Relationship Between Interest Rate and Default Rate",
       x = "Average Interest Rate (%)",
       y = "Observed Default Rate") +
  theme_minimal()
```



5. Implications for the H2 Methodology

The exploratory analysis reinforces the methodological choices adopted in the H2 framework and supports the earlier critique of the linear specification used by Zandi et al., particularly in relation to Equation 16 in their paper. The empirical patterns observed in the data indicate that simple linear relationships are insufficient to capture the structural and temporal complexity of mortgage default dynamics.

Class Imbalance

The observed default rate fluctuates between approximately 2.0 percent and 3.0 percent across the reporting period. This confirms that the dataset is highly imbalanced, with non-default observations dominating the target distribution. In such settings, standard loss functions tend to bias model training toward the majority class, leading to poor minority-class recall.

This empirical imbalance directly motivates the use of the Focal Loss function in the H2 methodology. By down-weighting easy, correctly classified non-default cases and emphasizing harder, misclassified default observations, Focal Loss mitigates majority-class dominance and improves the model's ability to learn rare default events.

Graph Topology and Neighborhood Structure

Group density diagnostics for the proposed spatial identifiers indicate that the graph construction strategy is well supported by the data. The median group size for `Geo_Key` is approximately 2,190 loans, while the median group size for `Lender_Key` is approximately 6,497 loans.

These group sizes are sufficiently large to support stable K-nearest neighbor aggregation within the Spatial Graph Attention Network (GAT) layer. Adequate neighborhood density ensures that node embeddings

are informed by meaningful peer information rather than sparse or noisy connections, thereby improving relational learning and reducing variance in message passing.

```
cat("\n--- Running K-NN Group Size Analysis ---\n")
```

```
##
## --- Running K-NN Group Size Analysis ---
```

```
if (!"final_with_targets" %in% names(loaded_data_eda)) {
  cat("  - WARNING: 'final_with_targets' not found in loaded EDA data. Skipping.\n")
} else {

  eda_data <- loaded_data_eda$final_with_targets

  # We only need one snapshot of the data to see the groups.
  # Use the first training snapshot as a representative sample.
  if (!exists("START_PERIOD_TRAIN")) {
    cat("  - WARNING: 'START_PERIOD_TRAIN' variable not found. Skipping snapshot filter.\n")
    first_snapshot_data <- eda_data
  } else {
    first_snapshot_data <- eda_data[Monthly_Reporting_Period == START_PERIOD_TRAIN]
  }

  if (nrow(first_snapshot_data) == 0) {
    cat("  - WARNING: No data found for the first snapshot. Skipping.\n")
  } else {

    # 1. Analyze Geo_Key groups
    geo_group_sizes <- first_snapshot_data[, .N, by = Geo_Key]
    cat("\n--- Geo_Key Group Size Stats ---\n")
    print(summary(geo_group_sizes$N))

    # 2. Analyze Lender_Key groups
    lender_group_sizes <- first_snapshot_data[, .N, by = Lender_Key]
    cat("\n--- Lender_Key Group Size Stats ---\n")
    print(summary(lender_group_sizes$N))

    # 4. Clean up
    rm(first_snapshot_data, geo_group_sizes, lender_group_sizes)
  }
}
```

```
##
## --- Geo_Key Group Size Stats ---
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     171    1310    2190    2497    3338    8885
##
## --- Lender_Key Group Size Stats ---
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     217    2342    6497   12487   13781   58868
```