Workshop #1:
Kaggle Systems Engineering Analysis
Loan Default Prediction

Team #9

Juan Esteban Avila Trujillo - 20251020054
Juan Jose León Gomez - 20212020055
Juan Pablo Diaz Ricaurte - 20222020076
Miguel Angel Hernandez Medina - 20222020035

September 27, 2025

Contents

1. Introduction

The following article aims at the integral **Systems Engineering** study of the Kaggle Competition's loan default prediction system. The system is examined from a holistic perspective, identifying its key elements, the relationships between the data, chaotic aspects, and the emerging dynamics that influence the model's behavior. Through this analysis, we explore sensitivity factors and potential sources of chaos, such as unpredictable data variations and feedback effects.

2. Competition Overview

The Kaggle competition **"Loan Default Prediction"**, sponsored by researchers from Imperial College London, aims to address two main tasks simultaneously or jointly:

- **Predict Default:** Determine the probability that a loan will fail.
- **Predict Loss:** Estimate the magnitude (severity) of the financial loss if the default occurs.

Unlike traditional finance-based approaches to this problem, which make a binary distinction between good or bad counterparties, the goal is to anticipate and incorporate both default and the severity of the resulting losses. By doing so, a bridge is built between traditional banking, which seeks to reduce the consumption of economic capital, and an asset management perspective, where risk is optimized for the financial investor. This approach confirms that the competition is not only about classification but also about regression to estimate the monetary or percentage value of the loss.

The dataset presents a complex machine learning challenge with a strong focus on **Loss Regression**. This dataset contains over 200,000 observations and nearly 800 features. The data has been pre-processed through:

- **Standardization** (A scaling process that transforms features to have a mean of 0 and a standard deviation of 1.)
- **De-trended** (Elimination of temporal trends or patterns (e.g., constant loan growth over the years) from the dataset.)
- **Anonymization** (In order to conceal the identity of individuals and the exact nature of the transactions).

This is a Multivariate Regression project that requires sophisticated handling of large and missing data to predict the severity of financial loss, with the objective of going beyond a binary classification that only answers whether the loan "defaults or not defaults".

3. Systems Analysis Report

3.1. Systemic Analysis. The competition can be understood as a **supervised prediction system**, where the goal is to transform historical loan information into predictive decisions about the risk of default.

3.1.1 The Inputs

- **Training set (`train.csv`)**
  - Contains historical loan records.
  - Contains: features + target variable (*default*).
  - Includes:
    - Predictor variables: financial, personal, and credit characteristics.
    - Target label: `default` = 1 (default occurred) or 0 (default did not occur).
  - **Relationships:** Used to train prediction models, analyzed through EDA (Exploratory Data Analysis), subjected to preprocessing and transformation, and allows internal validation of model quality.
- **Test set (`test.csv`)**
  - Includes the same characteristics as the training set.
  - Does not include the target variable (which is what must be predicted).
  - **Relationships:** Processed the same way as `train.csv` to maintain consistency, and passed to the trained model to obtain predictions.

3.1.2 The Outputs

- **Prediction per loan (modeled output)**: Expected result: 0 (no default) or 1 (default) for each entry in the test set.
- **Submission file**: CSV with ID + prediction.

3.1.3 Limitations

These are the operational, technical, and rule limitations that frame the system's behavior:

- **Data**: Prohibited from using unapproved external data. Anonymized variables, without clear descriptions (complicates direct interpretation). Possible class imbalance (fewer default cases than non-default).
- **Daily submission limit (20)**: Prevents massive iterations, forcing planning and prioritization of tests.
- **Evaluation**: **F1-score** is used, not *accuracy*.

3.2. Problem Complexity.

- **High dimensionality of the dataset**: The number of predictor variables can be high (tens or more). Many of the variables are anonymized, hindering interpretation and intelligent feature selection. Requires advanced feature selection or dimensionality reduction techniques (e.g., PCA, regularization).
- **Class imbalance**: Default cases are usually a minority ($< 20\%$ in many financial datasets). A naive model that predicts everything as "no default" can have high accuracy but a terrible F1-score, affecting the result. This necessitates the use of strategies such as:
  - Class reweighting
  - Oversampling (e.g., SMOTE)
  - Fine-tuning the decision threshold
- **Sensitive preprocessing**: Missing values, variable scaling, categorical encoding, outliers... all impact performance. Small differences in how data is treated can lead to large changes in the final result.

3.3. System Sensitivity.

Sensitive Point	**Possible Impact on Results**
Feature selection	Irrelevant or redundant variables can introduce noise or collinearity
Model type (trees, NN)	Different algorithms respond differently to imbalance and noise
Missing value imputation	Different decisions (mean, median, elimination) can significantly alter the model
Classification threshold	A poorly calibrated threshold directly affects the F1 metric (e.g., high recall and low precision)
Validation technique	Poorly stratified validation can produce erroneous performance estimates
Evaluation metric (F1)	Strongly penalizes false negatives and positives; models with good accuracy may have poor F1

3.3.1 Critical System Constraints

- **Daily submission limit (20):** Prevents massive iterations, requires planning and prioritizing tests.
- **Lack of feature knowledge:** Hinders the application of business logic. The analysis must be purely statistical.

- **Prohibition of external data:** Limits the incorporation of macroeconomic variables, real default history, etc.

4. Chaos and Complexity Theory

The loan default prediction competition presents a series of elements that can be linked to **chaos and complexity theory**:

4.1. Non-linear Processes. In loan applicant data, the effects of certain variables are not simply proportional: for instance, a small change in the applicant's income or history can trigger a much larger change in the probability of default. This is typical of **non-linear systems**, where the output (default risk) does not scale linearly with the input.

Furthermore, the interaction between variables — such as income, employment, loan amount, interest rate — can generate compounded effects: the combination of several seemingly moderate conditions can lead to high risk, while each condition separately would not.

4.2. Feedback. There are feedback phenomena that can make the system complex and difficult to predict accurately. For example, if many people in a geographical area start defaulting, the financial institution may tighten criteria or raise rates for that area, which in turn can increase the financial strain on new applicants and thus increase defaults. This is a "system → borrower → system" feedback loop.

Also, when the model's predictors (e.g., credit history) are adjusted based on the sector's past behavior, the system evolves: what was once a good predictor ceases to be one when borrowers and institutions change their behavior to adapt.

4.3. Randomness and Unpredictability (Chaos) in Human and Economic Behavior. A competition like this must deal with human and macroeconomic factors that are difficult to model: job changes, unexpected crises, pandemics, personal decisions, etc. These factors generate a **"sensitivity to initial conditions"**: two borrowers with very similar profiles may end up with very different outcomes due to a small difference in their circumstances (e.g., losing their job one month earlier). This is one of the elements of chaos: sometimes a small trigger produces a large effect.

Also, the distribution of defaults is often heavily imbalanced, and because there are so many external factors, the model may fail to generalize if it does not capture these unexpected variations.

4.4. Emergence and Network Effect. Although the data in this competition comes from individual borrowers, we can imagine that in the real financial world, defaults can generate chain effects (e.g., banks tightening policies, local economic friction, etc.). While this data may not be directly in the *dataset*, the idea is that in a financial system, what happens collectively is more than the sum of the individuals.

In this sense, although the challenge focuses on predicting at the individual level, there is a broader context, and models may not fully capture these network and emergence effects.

In summary, although the competition initially appears to be a standard classification problem (default-yes or default-no?), there is a lot of **hidden complexity** internally: variables that interact non-linearly, feedback effects between borrowers/financial institution, random elements that escape control, and system conditions that change over time. All this makes the outcome not entirely predictable and requires models to include good *feature-engineering*, robust techniques, and consideration that patterns may shift.

5. Visual Representation

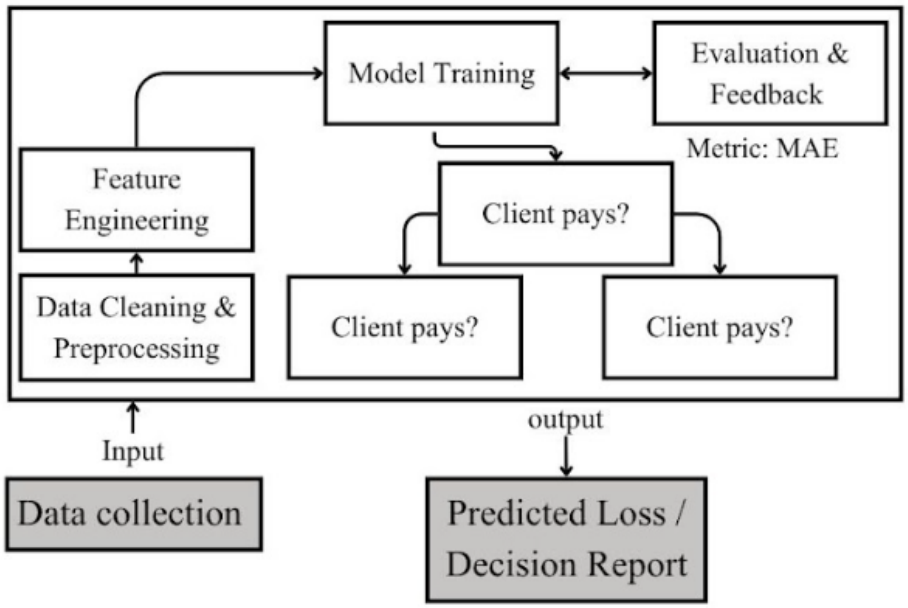

Figure 1: Diagram of structure competition

6. Conclusion

The systemic analysis showed that the difficulty of predicting loan delinquency functions as a **complex system**, driven by data uncertainty, variable dependencies, and non-linear behavior. Identifying the main elements (including data ingestion, preprocessing, feature engineering, modeling, evaluation, and feedback) made it possible to clearly understand how information moves and changes throughout the process. The study of sensitivity and chaotic influences highlighted the relevance of robust preprocessing, constant feedback, and model validation.