IIT GUWAHATI

SBI Hackathon

Fraud Detection and Defaulter Localization Challenge

TEAM: IDATEN

Date: 18/05/2025

By: **Priyanshu Maurya**
Vishnu Singh
Aaditya Shahi
Anubhav Saha

Introduction

Problem Statement

Task 1: Detect Fraudulent Activity

Build a model to flag fraudulent transactions or accounts using history and profile/device data. Ensure interpretability. Handle adversarial mimicry and class imbalance.

Task 2: Infer Defaulters' Last Location

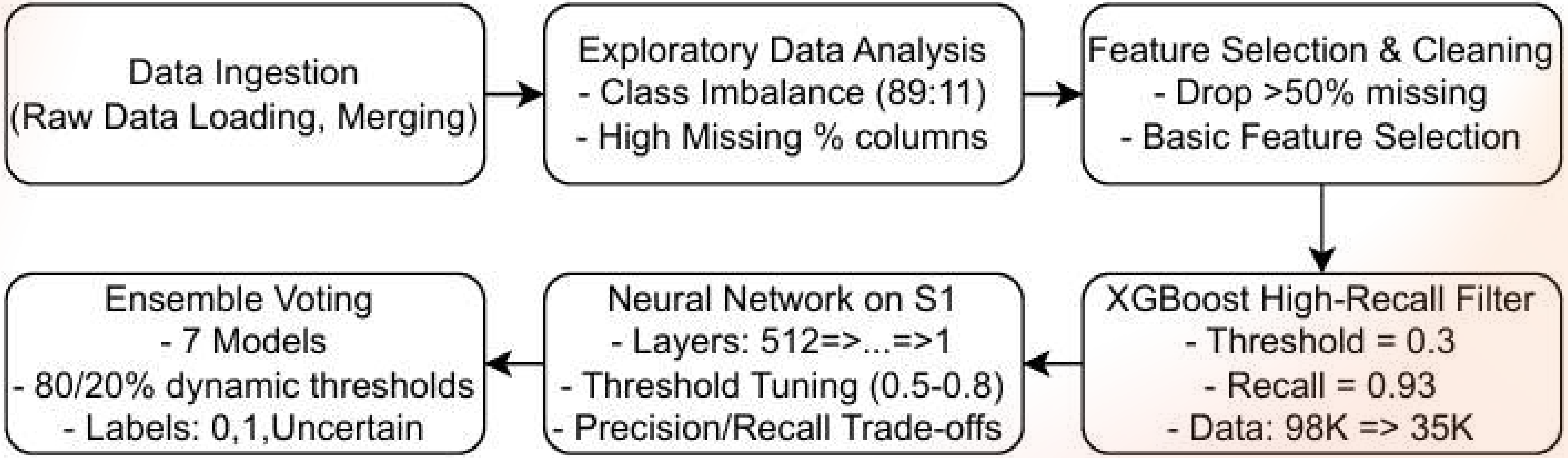
Estimate defaulters' last known location from device/tower connections. Predict final tower/region and explain how it was inferred.

Why we should solve this problem using ML?

We should use Machine Learning to solve these problems because it can analyze large, complex datasets, detect hidden patterns, adapt to evolving fraud tactics, and make accurate, real-time predictions—far beyond what manual methods can achieve.

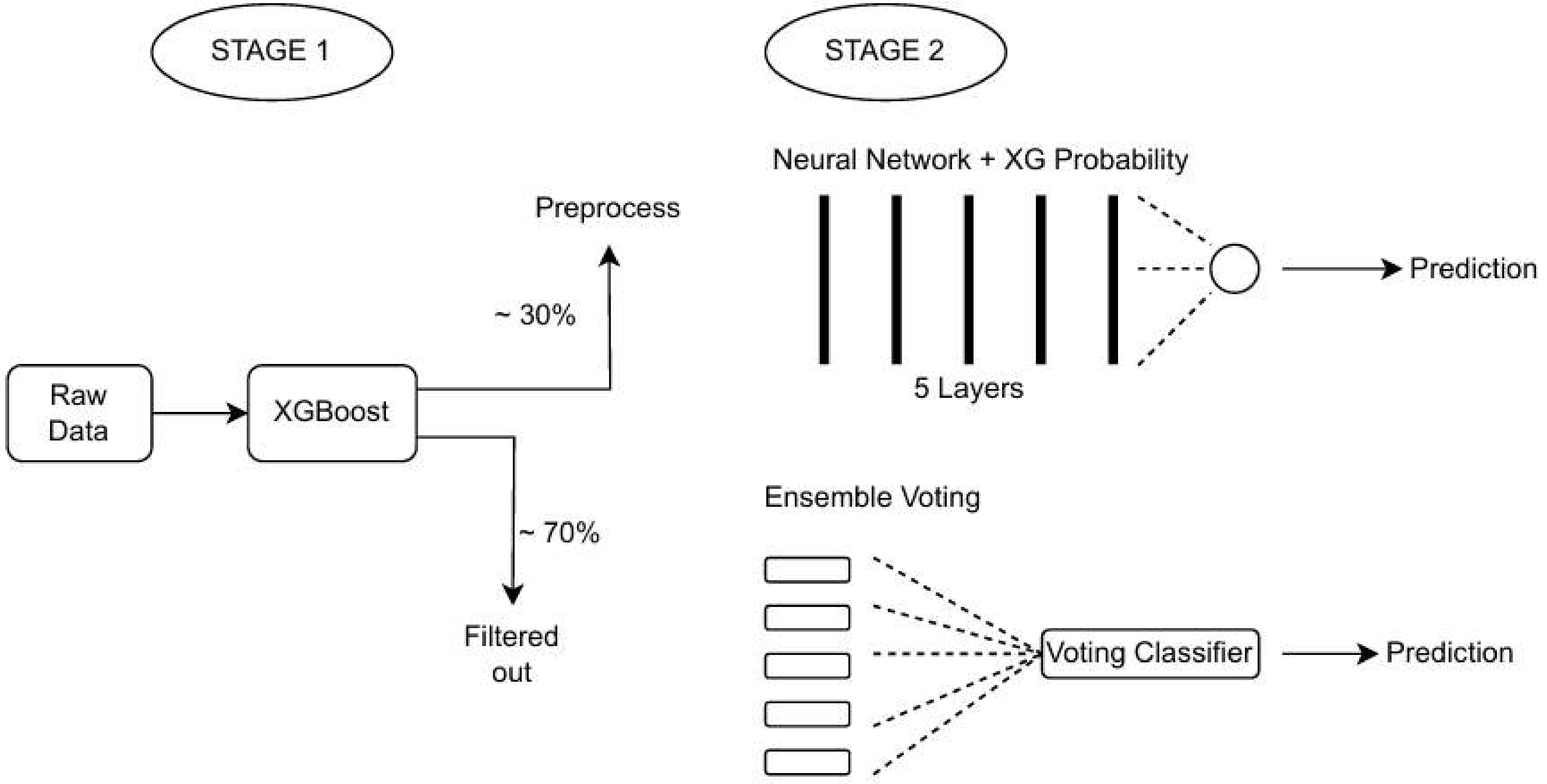
Workflow

Our prediction process follow this methodological workflow:

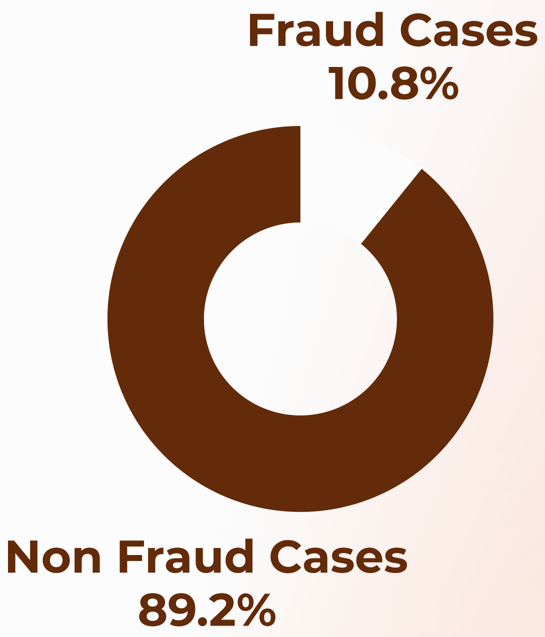


Workflow

We have divided the pipeline into 2 stages:



Dataset Overview



Number of rows: 327741
Number of columns: 139
Number of Fraud Cases: 35440
Number of Non-Fraud Cases: 292301
Number of Numerical Columns: 127
Number of Categorical Columns: 12
% of missing values: 8.9%

- Fraud cases are relatively rare, so imbalanced learning techniques will be needed.
- Categorical features likely represent device types, user profiles, or transaction categories.
- Numerical features may capture transaction amounts, frequencies, or timestamps.
- Potential presence of outliers in transaction-related columns.

- **Data Overview**

	ACCT_AGE	LIMIT	OUTS	ACCT_RESIDUAL_TENURE	LOAN_TENURE	INSTALAMT	SI_FLG	AGE	VINTAGE	KYC_SCR	...	CREDIT_HISTORY_LENGTH1	NO_OF_INQUIRIES1	INCOME_BAND1	AGREG_GROUP
0	1.613	1005500.0	494161.89	0.890	914	38513.0	Y	57.663	18.601	110.0	...	7yrs 6mon	0.0	G	#Total Xpress Credit
1	1.783	1005500.0	428072.24	0.720	914	38513.0	Y	57.833	18.771	110.0	...	7yrs 6mon	0.0	G	#Total Xpress Credit
2	1.698	1005500.0	461364.10	0.805	914	38513.0	Y	57.748	18.686	110.0	...	7yrs 6mon	0.0	G	#Total Xpress Credit
3	9.127	1005500.0	1204287.25	17.878	9862	12736.0	Y	52.302	14.039	110.0	...	10yrs 8mon	1.0	D	#Housing Loan
4	9.296	1005500.0	1203224.25	17.708	9862	12736.0	Y	52.472	14.209	110.0	...	10yrs 8mon	1.0	D	#Housing Loan

Data Cleaning Strategy

- **Missing Data Columns**

	Missing Values	Percentage
LAST_1_YR_RG4	303363	92.561809
LAST_3_YR_RG4	267761	81.698964
CUST_NO_OF_TIMES_NPA	215669	65.804706
FIRST_NPA_TENURE	215669	65.804706
LATEST_NPA_TENURE	215669	65.804706
NO_YRS_NPA	215669	65.804706
NO_ENQ	167955	51.246258
CRIFF_33	60540	18.471903
CRIFF_44	60380	18.423084
CRIFF_22	60174	18.360230
SIXMNTHAVGYTD	31064	9.478216
SIXMNTHAVGQTD	31064	9.478216
SIXMNTHSCR	31064	9.478216
SIXMNTHSDR	31064	9.478216
SIXMNTHOUTSTANGBAL	31064	9.478216
SIXMNTHAVGMTD	31064	9.478216
FIVEMNTHSCR	30826	9.405598
FIVEMNTHAVGMTD	30826	9.405598
FIVEMNTHOUTSTANGBAL	30826	9.405598
FIVEMNTHSDR	30826	9.405598

Missing Values Overview

- 117 columns have missing values.
- High-missing features include:
  - LAST_1_YR_RG4: 303k
  - LAST_3_YR_RG4: 267k
  - FIRST_NPA_TENURE, CUST_NO_OF_TIMES_NPA, LATEST_NPA_TENURE, NO_YRS_NPA: 215k each
  - NO_ENQ: 167k

Actions Taken

- Dropped 7 columns with over 50% missing values.

Key Points

- Data has high sparsity in key financial features.
- Requires robust handling to avoid bias and preserve model quality.

Miscellaneous Insights

- **AGREG_GROUP** and **PRODUCT_TYPE** represent the same info — one was dropped to avoid redundancy.
- **Unique ID** column removed as it doesn't contribute to model training.

Stage 1: Reducing Search Space via High-Recall Filtering (Xg Boost)

Approach:

- We leveraged the abundance of non-fraud samples to reduce our working dataset using XGBoost's ability to assign high recall to fraud cases while filtering many non-fraud cases.
- A lower threshold helps maximize recall (true fraud detection), allowing the model to flag more frauds with fewer missed cases.
- Chosen empirically based on cross-validation results.

Performance at Threshold 0.3:

Metric	Value	Interpretation
TP	9,905	True frauds correctly detected
FP	25,932	Non-frauds mistakenly flagged
TN	61,759	Non-frauds correctly identified
FN	727	Missed frauds

In Test Set:

- Total actual frauds: 9,905 + 727 = 10,632
- Fraud Miss Rate: 727 / 10,632 ≈ 6.8%
- Total actual non-frauds: 25,932 + 61,759 = 87,691
- Non-Fraud Filter Accuracy: 61,759 / 87,691 ≈ 70.4%

Filtered Subset for Further Modeling:

- 25,932 flagged non-frauds + 9,905 predicted frauds = 35,837
- Effectively reducing dataset size from 98,323 → 35,837 for expensive modeling.

Benefits of High-Recall Filtering Strategy

1. Improved Class Balance

- From initial imbalance of 89:11 → new ratio of 25,932 non-frauds : 9,905 frauds (2.6 : 1 ratio).
- Rebalanced Class Distribution:

○ Class 1 (Fraud): ~27.65%
○ Class 0 (Non-Fraud): ~72.35%

2. Easier Data Analysis

- With better balance and reduced volume, trends and tendencies in fraud-related features became more observable.
- Enabled identification of patterns that were previously masked by skewed class ratios.

3. Effective Model Training

- Allowed us to train and evaluate a variety of models on the filtered subset efficiently.
- Reduced sample size (from ~98K to ~35K) helped in:

○ Faster experimentation
○ Use of more compute-intensive models
○ Cleaner validation cycles

- This staged detection pipeline trades off minor fraud loss (6.8%) for a 70% reduction in evaluation volume, making it practical and impactful in real-world systems.

Feature Engineering

- Utilised binary indicators **(SI_FLG, LOCKER_HLDR_IND, UID_FLG, KYC_FLG, INB_FLG, EKYC_FLG)** into a single feature TOTAL_FLAGS to capture cumulative identity verification and service linkage behavior
- Created **TOTAL_FLAGS** by summing all flag features. It showed better correlation with target **(0.0800)** than individual flags

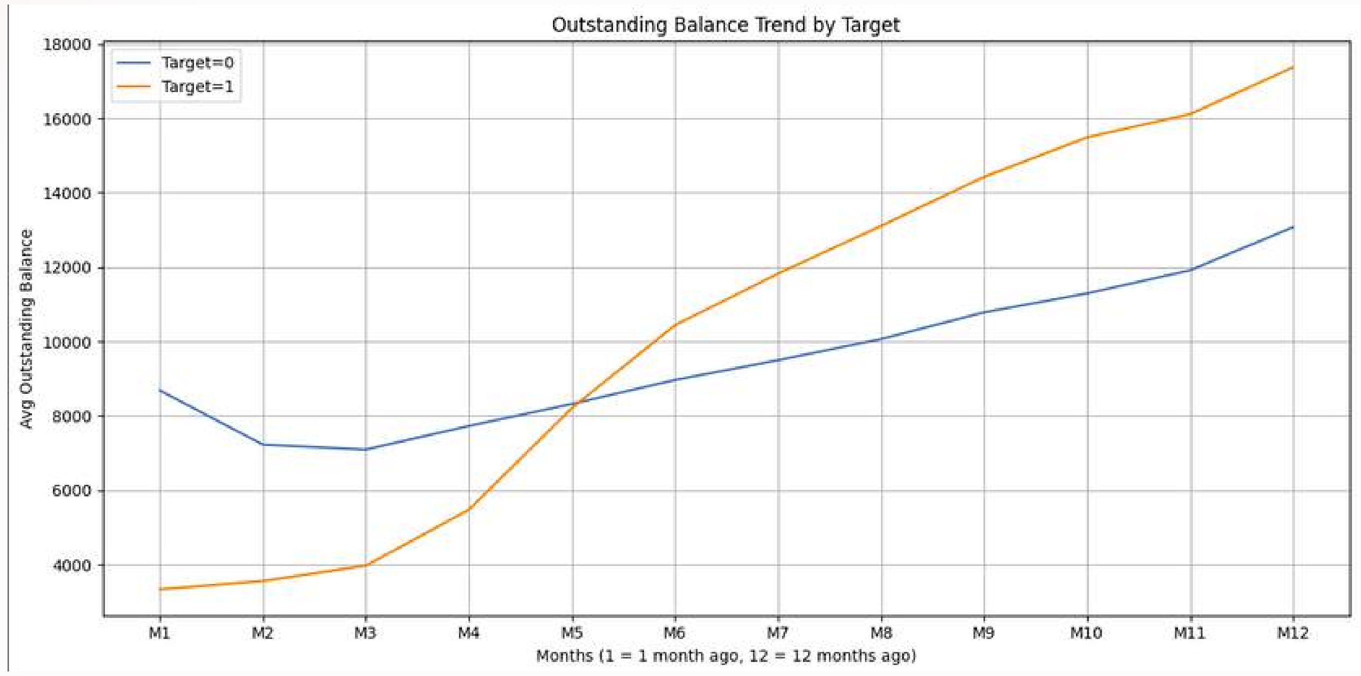
```
Correlation of each flag column with TARGET:
SI_FLG             0.058717
INB_FLG            0.058481
UID_FLG            0.048980
EKYC_FLG           0.028122
KYC_FLG            0.004896
LOCKER_HLDR_IND   -0.023120
```

- **Split-Averaged Balances**

Analyzing monthly balances revealed a trend shift in fraud cases. To capture this, we created:

- ***avg_balance_early:*** Mean of Month 1–5
- ***avg_balance_late:*** Mean of Month 6–12

These features help detect temporal balance anomalies.



Feature Engineering

Full processed dataset was narrowed down using high-recall filtering to a working subset (S1) of ~35,837 records and following observations were obtained:

- **Feature: LATEST_CR_DAYS**

Observation: In subset S1, a notable number of fraud cases had LATEST_CR_DAYS between 200 and 500.
Engineered Feature:
➤ **high_latest_cr_days** → Binary flag: 1 if 200 < LATEST_CR_DAYS < 500, else 0
➤ Helps capture credit inactivity pattern linked to fraud.

- **Feature: ALL_LON_MAX_IRAC**

Observation: Fraud cases were significantly higher when ALL_LON_MAX_IRAC ≠ 3.
Engineered Feature:
➤ non_irac_3_flag → Binary flag: 1 if ALL_LON_MAX_IRAC ≠ 3, else 0
➤ Reflects deviation from ideal loan classification associated with fraud.

- **Feature: LOAN_TENURE**

Observation: Higher fraud concentration in loans with tenure ≤ 1096 days (≈3 years).
Engineered Feature:
➤ **short_loan_tenure_flag** → Binary flag: 1 if tenure ≤ 1096, else 0
➤ Indicates risk-prone short-term loans often linked to fraud behavior.

Feature Engineering

Time-Series Feature Aggregation and Trend Extraction:
To better capture temporal financial behavior, we engineered statistical summaries from 12-month time-series features such as **SCR, SDR, OUTSTANGBAL, AVGMTD, AVGQTD, and AVGYTD**. For each of these, we computed:

- Mean – to capture average behavior over the year.
- Min/Max – to capture extremes in the trend.
- Standard Deviation – to measure variability.
- Slope – to quantify upward/downward trends using linear regression.
- First & Last Month Values – to detect shifts.
- Difference (Last - First) – to measure net change.

These aggregations help condense time-series data into meaningful features, enhancing model performance and interpretability.

Feature Conversion: Account Age & Credit History
 Transformed AVERAGE_ACCT_AGE1 and CREDIT_HISTORY_LENGTH1 to total months for better model compatibility.

Stage 2: Modelling Approach

S1 Subset Overview

- Total Samples: 35,837
- Training Set: 25,085 rows
- Testing Set: 10,752 rows
- Fraud Cases in Test Set: 2,972

- **Threshold-Wise Performance**

Threshold	True Positives (Frauds Caught)	False Positives (Mistaken Non-Frauds)
0.5	2,069 (~70%)	1,705
0.6	1,627	962
0.7	1,164	495
0.8	673	175

Model Architecture

- Type: Feed-Forward Neural Network (Binary Classification)
- Layers: 512 → 256 → 128 → 64 → 1
- Activations: ReLU
- Output: Sigmoid
- Regularization: Dropout after each hidden layer
- Loss Function: Binary Cross-Entropy
- Optimizer: Adam
- Additional Feature: XGBoost fraud probability appended

Insights:

- Threshold tuning allows precision-based filtering.
- At 0.8 threshold, only 175 false positives for 673 true frauds.
- Ideal for high-confidence flagging for manual verification or deeper models.

Ensemble Voting with Uncertainty Handling

Goal

Leverage model consensus to make high-confidence predictions for fraud detection, while isolating uncertain cases for further analysis.

Process:

1. Train all models on the training set
2. Generate probability predictions on the test set
3. Apply dynamic thresholding (per model):
   - Top 20% (≥80th percentile): Predict as Fraud (1)
   - Bottom 20% (≤20th percentile): Predict as Non-Fraud (0)
   - Middle 60%: Mark as Uncertain (2)

Model Ensemble Setup

Models Trained (7):

- XGBoost
- LightGBM
- CatBoost
- ExtraTrees
- Multi-Layer Perceptron (MLP)
- Logistic Regression
- Random Forest

Why Use Percentile Thresholds?

Model probability distributions vary significantly

Fixed thresholds (e.g., 0.5) aren't reliable across models

Percentile-based cutoffs standardize interpretation per model

Impact

- ~1,500 cases confidently labeled with high precision
- ~9,200 uncertain cases earmarked for deeper modeling
- Strikes a balance between accuracy and risk control
- Enables targeted manual review and downstream refinement

Modifications for Task-2

Objective: Identify the last known location of potential loan defaulters using device-tower interaction data.
Approach: A hybrid model combining Time Series Analysis and **Graph Neural Networks (GNNs)**.

Data Preprocessing

1. Identifying Relevant Accounts

- Filtered accounts where **TARGET = 1** (marked as defaulters).
- Mapped these accounts to their associated **device_ids.**

2 Generating Device-Tower Sequences

- For each device, created a time-ordered sequence of tower connections.
- Missing data was handled via forward/backward fill or removal, depending on context.

Graph-Based Representation

Graph Construction

- **Nodes:** device_ids and tower_ids.
- **Edges:** Created for each device-tower interaction with timestamp context.
- Built using NetworkX, later formatted for PyTorch Geometric (PyG).

Graph Objective: Capture spatial and relational behavior of defaulters through tower connectivity.

GNN Architecture

- Employed a 2-layer GCN (Graph Convolutional Network).
- First layer extracts initial relationships, second layer learns deeper spatial representations.
- Output node embeddings can represent location affinity and movement patterns.

Time Series Modeling:

Sequence-Based Prediction

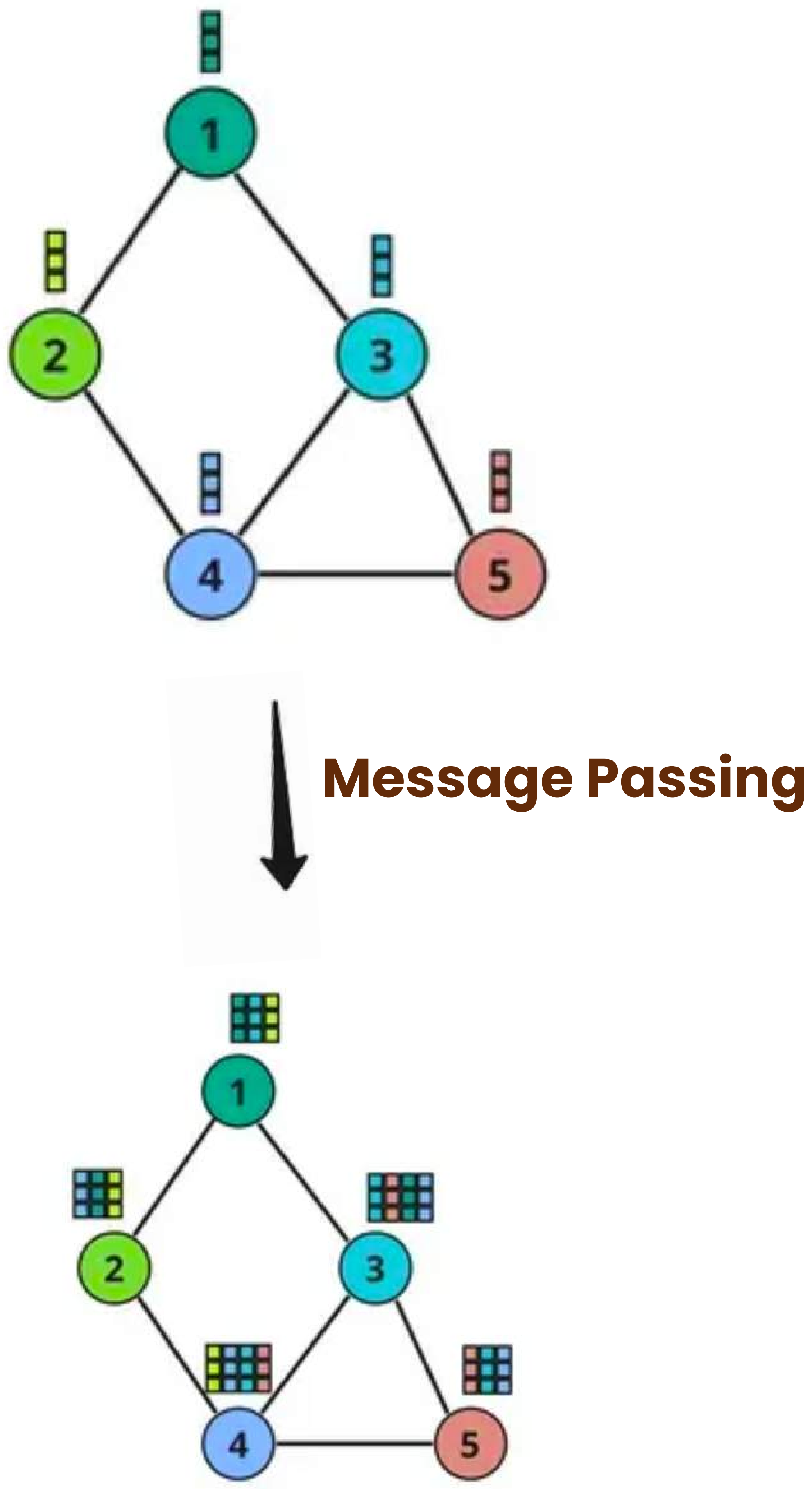
- Created sequences of towers visited by each defaulter.
- Applied foundational sequence models along with the last known tower as a heuristic location.

Why it works:

Devices tend to show recency bias – their most recent tower often reflects their last known physical presence.

Interpretability & Reasoning

- For every prediction, provided a trace of historical tower visits.
- Justified predictions using:
  - Recency of last tower visit.
  - Frequency of visits to a tower.
  - GNN-based spatial similarity with other devices/towers.



Benefits of High-Recall Filtering Strategy

1. Improved Class Balance

- From initial imbalance of 89:11 → new ratio of 25,932 non-frauds : 9,905 frauds (2.6 : 1 ratio).
- *Rebalanced Class Distribution:*
  - Class 1 (Fraud): ~27.65%
  - Class 0 (Non-Fraud): ~72.35%

2. Easier Data Analysis

- With better balance and reduced volume, trends and tendencies in fraud-related features became more observable.
- Enabled identification of patterns that were previously masked by skewed class ratios.

3. Effective Model Training

- Allowed us to train and evaluate a variety of models on the filtered subset efficiently.
- Reduced sample size (from ~98K to ~35K) helped in:
  - Faster experimentation
  - Use of more compute-intensive models
  - Cleaner validation cycles

    This staged detection pipeline trades off minor fraud loss (6.8%) for a *70% reduction in evaluation volume*, making it practical and impactful in real-world systems.

Stage 1: Reducing Search Space via High-Recall Filtering (Xg Boost)

Approach:

We leveraged the abundance of non-fraud samples to reduce our working dataset using XGBoost's ability to assign high recall to fraud cases while filtering many non-fraud cases.

- A lower threshold helps maximize recall (true fraud detection), allowing the model to flag more frauds with fewer missed cases.
- Chosen empirically based on cross-validation results.

Performance at Threshold 0.3

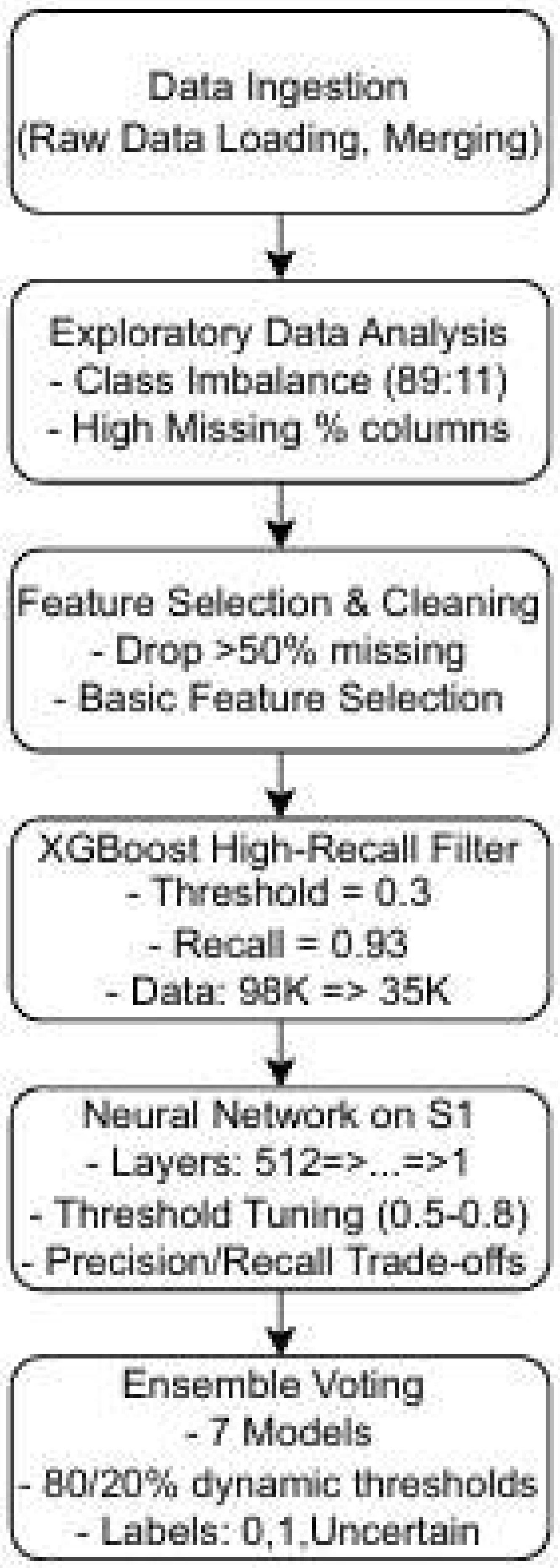
- **Recall:** 0.9316

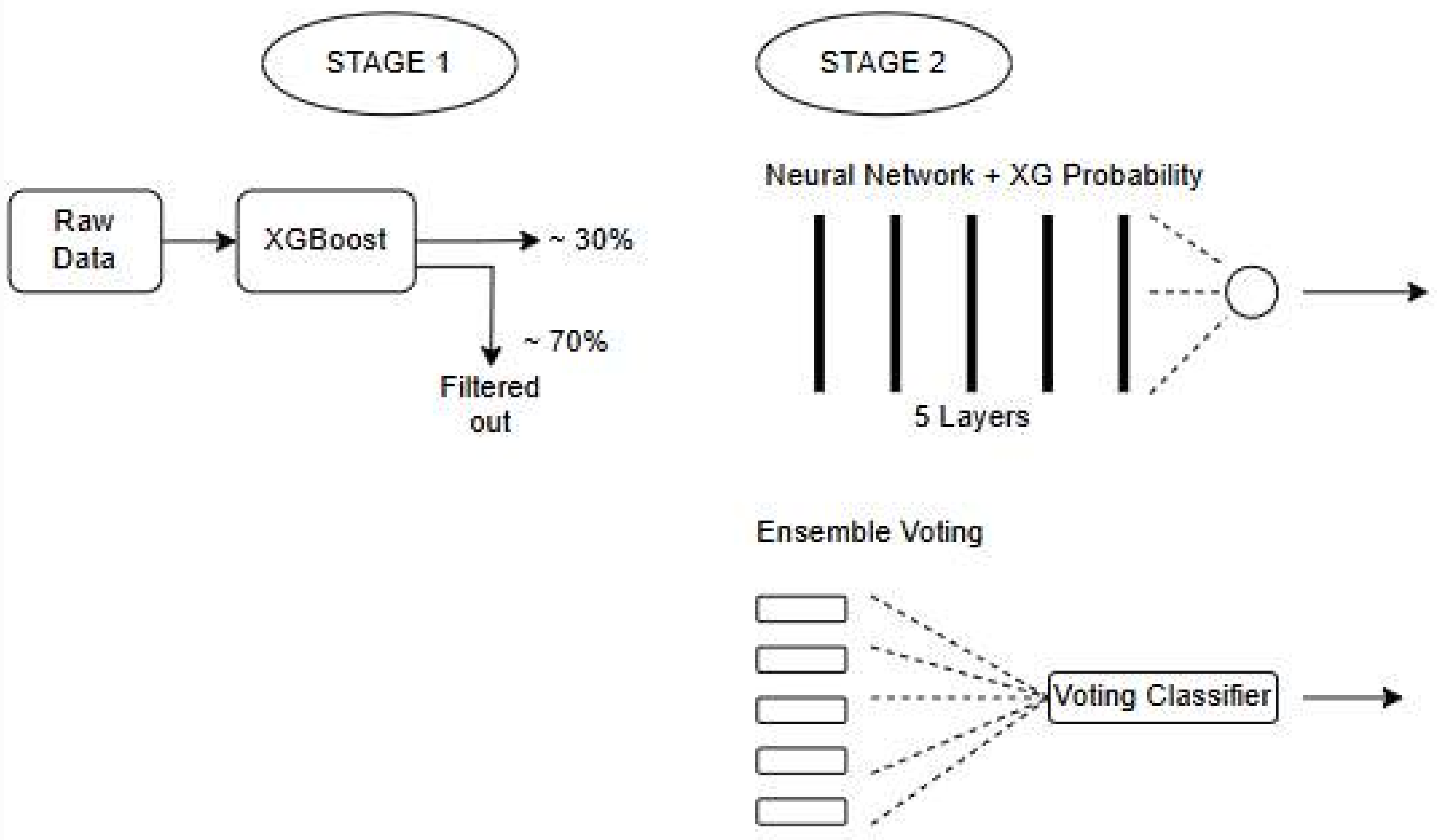
Metric	Value	Interpretation
TP	9,905	True frauds correctly det
FP	25,932	Non-frauds mistakenly fl
TN	61,759	Non-frauds correctly ider
FN	727	Missed frauds

- *Train/Test Split:*
    - Total samples: 327,741
    - Training set: 229,418
    - Test set: 98,323

- *Train/Test Split:*
    - Total samples: 327,741
    - Training set: 229,418
    - Test set: 98,323

- *In Test Set:*
    - Total actual frauds: 9,905 + 727 = 10,632
    - *Fraud Miss Rate:* 727 / 10,632 ≈ 6.8%
    - Total actual non-frauds: 25,932 + 61,759 = 87,691
    - *Non-Fraud Filter Accuracy:* 61,759 / 87,691 ≈ 70.4%
- *Filtered Subset for Further Modeling:*
    - 25,932 flagged non-frauds + 9,905 predicted frauds = 35,837
    - Effectively reducing dataset size from 98,323 → 35,837 for expensive modeling.


Data Ingestion
(Raw Data Loading, Merging)
Exploratory Data Analysis
- Class Imbalance (89:11)
- High Missing % columns
Feature Selection & Cleaning
- Drop >50% missing
- Basic Feature Selection
XGBoost High-Recall Filter
- Threshold = 0.3
- Recall = 0.93
- Data: 98K => 35K
Neural Network on S1
- Layers: 512=>...=>1
- Threshold Tuning (0.5-0.8)
- Precision/Recall Trade-offs
Ensemble Voting
- 7 Models
- 80/20% dynamic thresholds
- Labels: 0,1,Uncertain


STAGE 1
STAGE 2
Raw Data
XGBoost
~ 30%
~ 70%
Filtered out
Neural Network + XG Probability
5 Layers
Ensemble Voting
Voting Classifier

Conclusion and future work

Problems

- **Problem 1**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu

- **Problem 2**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu

- **Problem 3**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu

Solutions

- **Solution 1**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu

- **Solution 2**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu

- **Solution 3**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu

Market Size

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation

- **Total Available Market (TAM)**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud

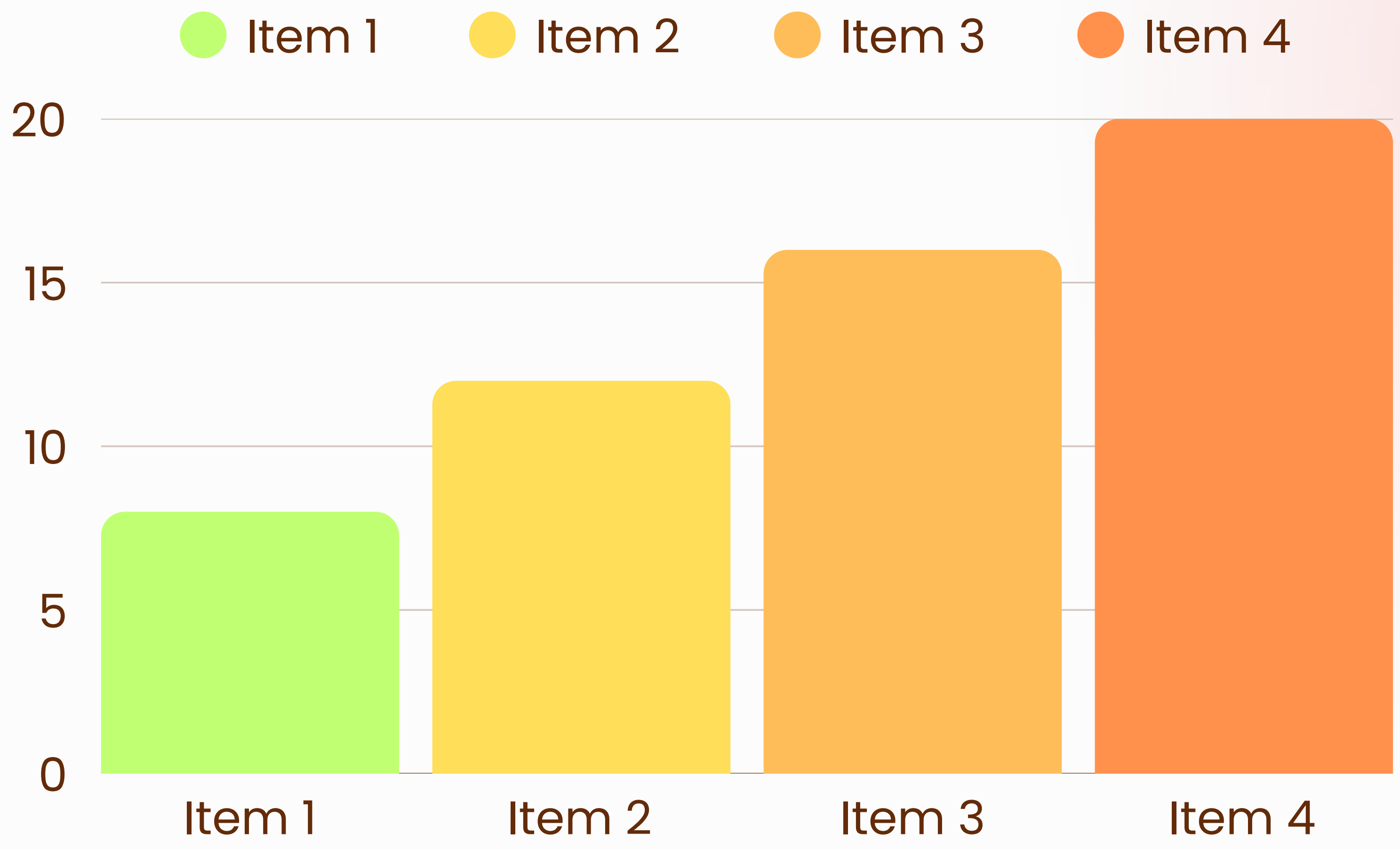
- **Serviceable Available Market (SAM)**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud

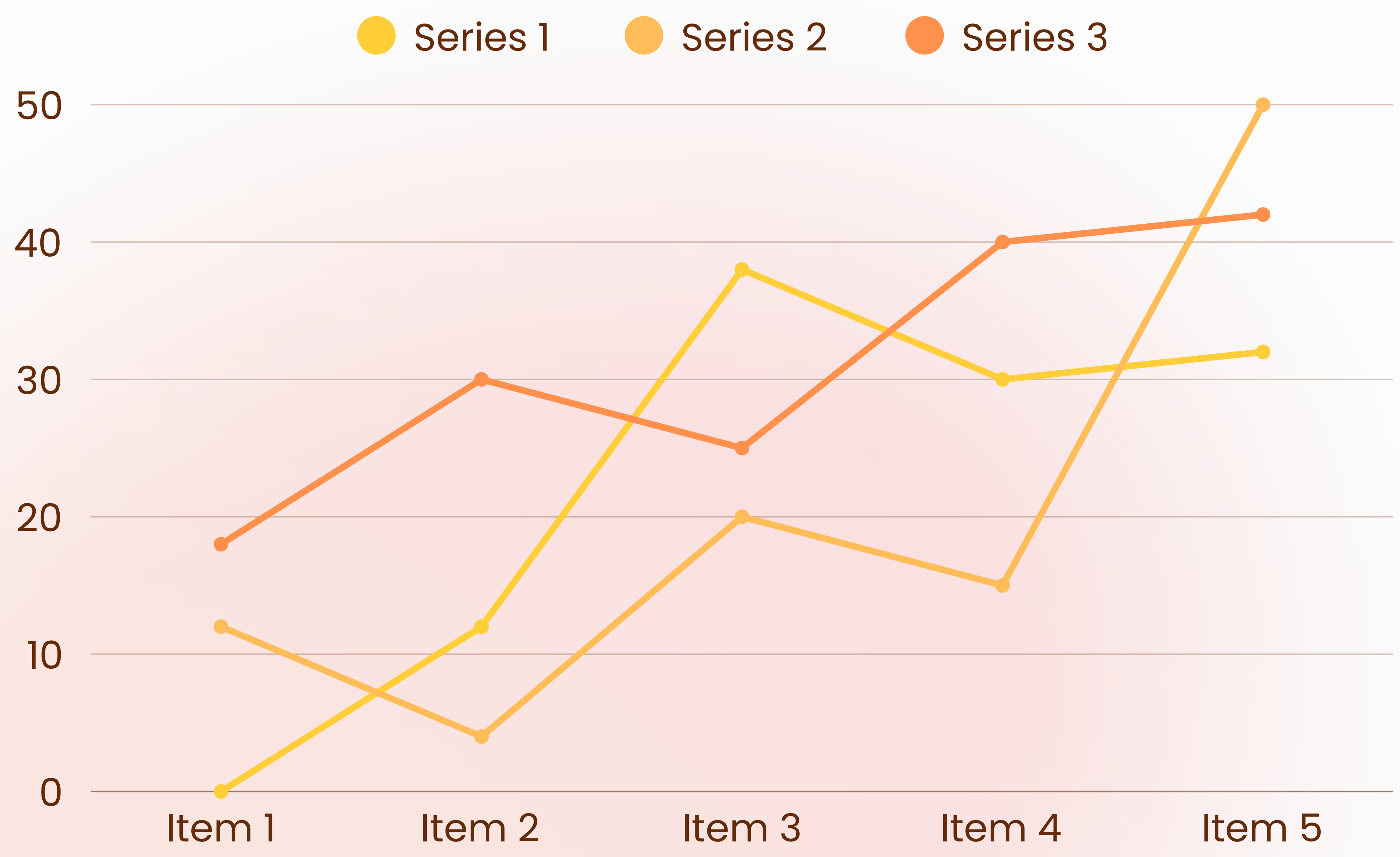
- **Serviceable Obtainable Market (SOM)**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud

Business Model



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur

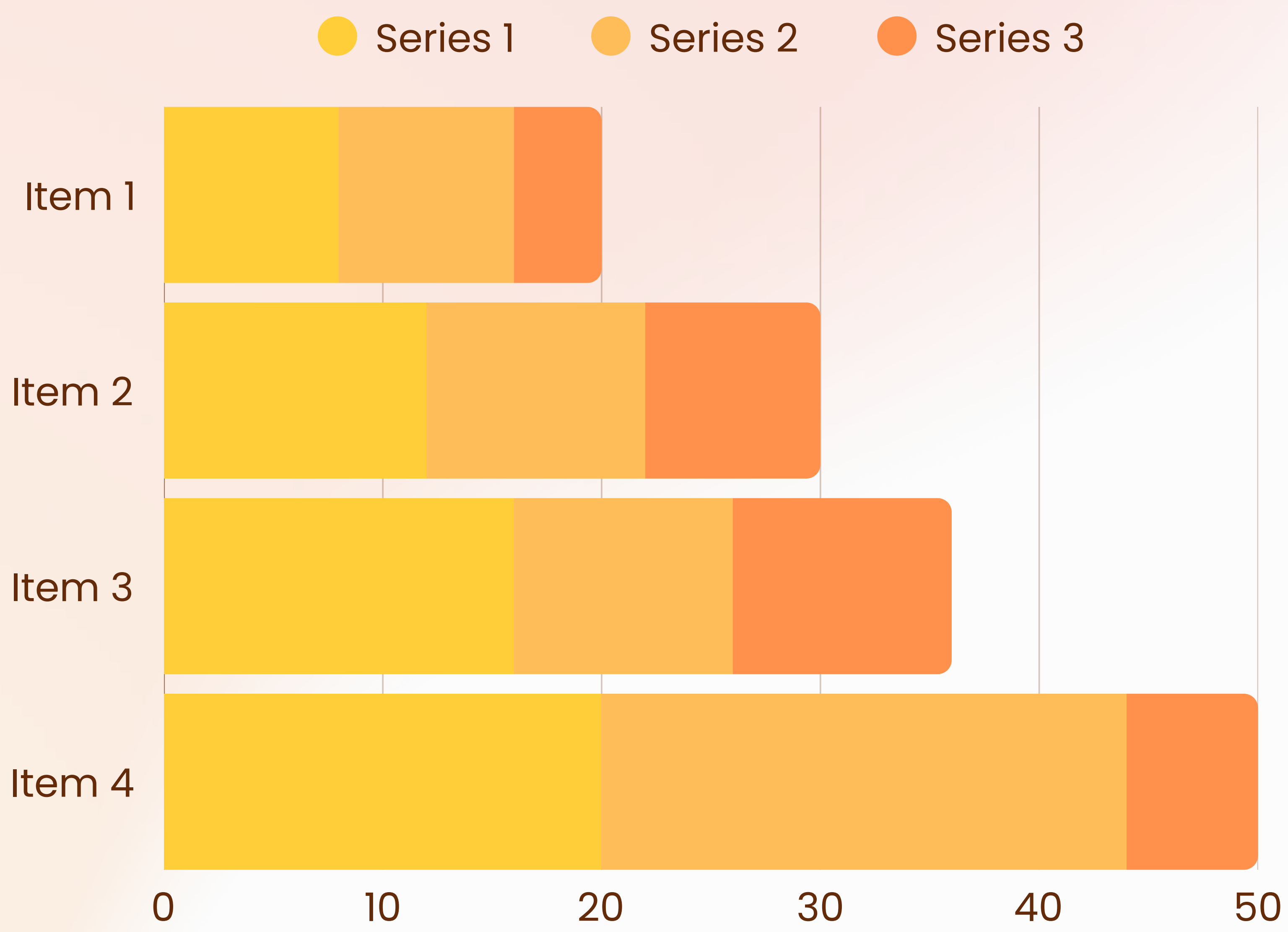


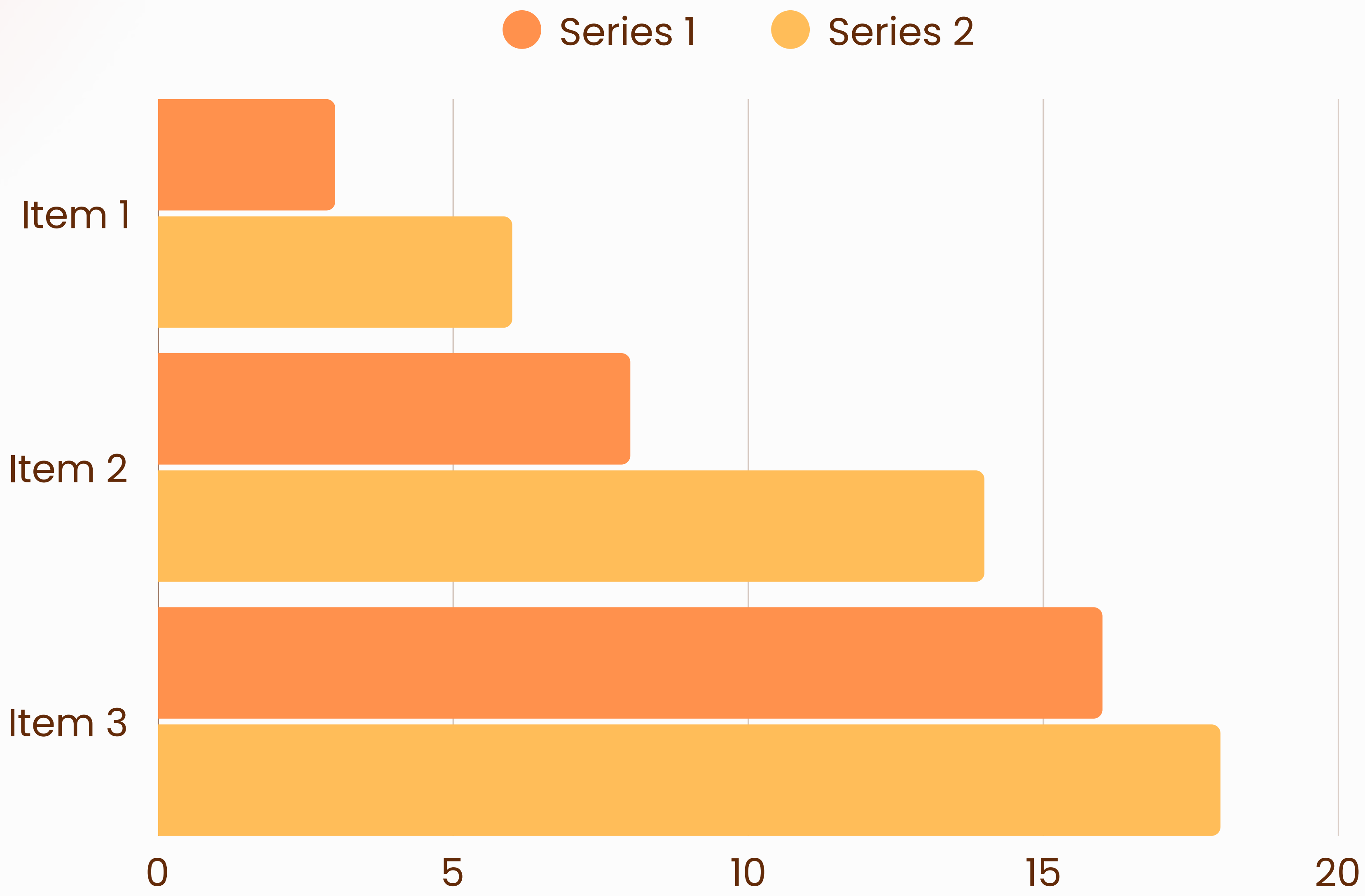
- Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

- Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu

Statictic





Statictic

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu

Testimonial

★★★★★

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris

★★★★★

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris

★★★★★

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris

Home About Contact

CHALLENGES IN DATA ANALYSIS

Overcoming Common Obstacles

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed at ipsum vitae lacus lobortis lacinia. Donec tristique arcu massa, at pharetra tortor feugiat non.

Read More