Tharana

Bopearachchi

Week 01

Advanced Exploratory Data Analysis

A comprehensive guide to EDA techniques for machine learning practitioners

{{mobile}}

tharanabope30@gmail.com

2011

About This Presentation

Tharana

Bopearachchi

Focus Area

Applied EDA for Customer Churn Prediction

Course Objectives

- Master advanced data exploration techniques specifically for churn analysis
- Learn to identify key indicators and patterns that predict customer attrition
- Develop actionable insights from complex customer datasets

Target Audience

Data scientists and analysts working with customer retention challenges across industries including telecommunications, subscription services, and e-commerce.

{{mobile}}

Approach

Hands-on learning with real-world datasets, practical visualization techniques, and industry-proven methodologies for translating EDA into business value.

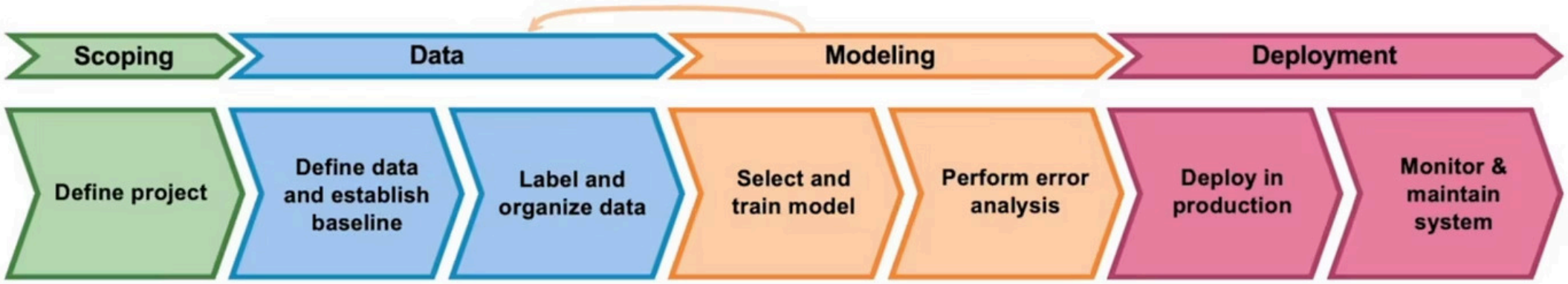
0011

tharanabope30@gmail.com

Tharana

Bopearachchi

ML Life Cycle



{{mobile}}

tharanabope30@gmail.com

Tharana Bopearachchi

ML Lifecycle Overview

CRISP-DM (1996)

Cross-Industry Standard Process for Data Mining

1. Business understanding
2. Data understanding
3. Data preparation
4. Modeling
5. Evaluation
6. Deployment

CRISP-ML(Q) (2020)

Extends CRISP-DM for modern ML with Quality focus

Adds critical phases:

- Monitoring
- Maintenance
- Quality assurance
- Governance

These frameworks provide structured yet iterative approaches to ML projects, emphasizing the cyclical nature of development and continuous improvement.

{{mobile}} tharanabope30@gmail.com

Phase 1: Business & Data Understanding

Define Business Objectives

Identify the specific business problem to solve and establish clear success criteria

Example: "Reduce customer churn by 15% within 6 months"

Establish Key Performance Indicators (KPIs)

Determine metrics to measure success in business terms

Example: Retention rate, Customer Lifetime Value (CLV), Cost per acquisition

Assess Constraints & Requirements

Identify technical, ethical, and regulatory limitations

Example: GDPR compliance, model explainability needs, deployment environment

This critical first phase aligns technical work with business value and prevents solving the wrong problem effectively.

Tharana Bopearachchi

Phase 2: Data Collection & Labeling

Data Source Selection

- Internal databases and warehouses
- External APIs and services
- Web scraping and public datasets
- Synthetic data generation

Labeling Workflows

- In-house labeling teams
- Crowdsourcing (e.g., Mechanical Turk)
- Specialized labeling services
- Semi-automated approaches

Sampling Strategies

- Random sampling
- Stratified sampling
- Time-based sampling

Cost-Quality Tradeoffs

- Label quality vs. quantity
- Active learning to prioritize examples
- Expert validation of critical cases

{{mobile}} tharanabope30@gmail.com

Tharana Bopearachchi

Phase 3: Data Cleaning & Exploration

Data Cleaning Techniques

- Handling missing values (imputation, deletion)
- Outlier detection and treatment
- Deduplication and consistency checks
- Type conversion and standardization
- Handling imbalanced classes

Exploratory Data Analysis (EDA)

- Descriptive statistics (mean, median, std)
- Distribution visualization (histograms, box plots)
- Correlation analysis (heatmaps)
- Time-series patterns
- Dimensionality reduction for visualization

EDA uncovers patterns, anomalies, and relationships in data that guide feature engineering and model selection decisions. Thorough cleaning prevents the "garbage in, garbage out" problem.

{{mobile}} tharanabope30@gmail.com

Tharana Bopearachchi

Phase 4: Feature Engineering & Modeling

Feature Engineering

- Feature creation (ratios, aggregations)
- Encoding categorical variables
- Scaling and normalization
- Dimensionality reduction
- Feature selection based on importance

Model Selection & Training

- Algorithm selection based on problem type
- Cross-validation strategies
- Hyperparameter tuning (grid/random search)
- Ensemble methods
- Transfer learning from pre-trained models

Feature engineering often has greater impact on model performance than algorithm selection. The best features capture domain knowledge and transform raw data into informative signals.

{{mobile}} tharanabope30@gmail.com

Tharana Bopearachchi

Phase 5: Model Evaluation & Validation

Classification Metrics

- Accuracy, Precision, Recall, F1-score
- ROC curves and AUC
- Confusion matrices
- Log loss and cross-entropy

Regression Metrics

- MSE, RMSE, MAE
- R-squared and adjusted R-squared
- Explained variance

Validation Strategies

- Train/validation/test splits
- K-fold cross-validation
- Stratified and time-series splits
- Bootstrap sampling

Bias-Variance Tradeoff

Balancing model complexity to avoid underfitting (high bias) and overfitting (high variance)

Regularization techniques (L1/L2, dropout, early stopping)

{{mobile}}

tharanabope30@gmail.com

Phase 6: Deployment & Monitoring

Deployment Strategies

- CI/CD pipelines for ML models
- Containerization (Docker, Kubernetes)
- API endpoints (REST, gRPC)
- Batch vs. real-time inference
- Edge deployment for low-latency

Monitoring Systems

- Performance metrics tracking
- Data drift detection
- Concept drift detection
- Resource utilization
- A/B testing infrastructure

Deployment is where models generate actual business value. Robust monitoring is essential to detect degradation before it impacts users and to inform retraining decisions.

{{mobile}} tharanabope30@gmail.com

Phase 7: Maintenance & Governance

Tharana

Bopearachchi

Model Versioning

T rack all model versions, parameters, and training data to enable rollback and reproducibility

Retraining Cycles

Establish regular and event-triggered retraining schedules based on performance degradation or data drift

Governance & Compliance

Maintain audit logs, model cards, and lineage tracking to ensure regulatory compliance and ethical use

{{mobile}}

Feedback Loops

Capture user feedback and outcomes to continuously improve models and identify new features

0011

Proper MLOps practices ensure models remain effective, compliant, and aligned with business goals throughout their lifecycle.

tharanabope30@gmail.com

What is Exploratory Data Analysis?

Tharana Bopearachchi

Exploratory Data Analysis (EDA) is a critical approach to analyzing datasets that combines statistical methods and visualization techniques to:

- Understand data structure and properties
- Detect patterns, anomalies, and relationships
- Test assumptions and formulate hypotheses
- Inform feature engineering and modeling decisions

EDA bridges the gap between raw data collection and sophisticated machine learning, ensuring quality inputs for predictive models.

{{mobile}}

tharanahope30@gmail.com

Tharana Bopearachchi

Why is EDA Important?

Data Validation

Confirms dataset quality and identifies issues before they impact model performance

Pattern Discovery

Surfaces hidden structures, distributions, and relationships that might not be apparent in raw data

Model Foundation

Creates a solid basis for feature engineering and algorithm selection decisions

Insight Generation

Provides business value through data-driven insights, independent of modeling outcomes

{{mobile}} tharanabope30@gmail.com

Types of Data

Tharana Bopearachchi

Qualitative (Categorical)

1 Nominal

Categories without inherent order (e.g., Country, Product Type)

Encoding: One-hot encoding

2 Ordinal

Categories with meaningful order (e.g., Education Level, Customer Satisfaction)

Encoding: Label encoding with ordered mapping

Quantitative (Numerical)

1 Discrete

Countable values (e.g., Number of Purchases, Children)

Treatment: May need special handling for zero-inflation

2 Continuous

Measurable values (e.g., Income, Age, Transaction Amount)

Treatment: Often requires normalization, binning

EDA Workflow Overview

Tharana Bopearachchi

Data Sourcing

Identify and access relevant data sources

Data Cleaning

Handle missing values, outliers, and inconsistencies

Univariate Analysis

Examine distribution of individual variables

Bivariate & Multivariate Analysis

Explore relationships between variables

{{mobile}}
tharanabope30@gmail.com

Feature Engineering

0011

Create transformed variables based on insights

Visualization

Create explanatory charts and graphs

Summary & Recommendations

Document findings and next steps

Data Sourcing

Tharana

Bopearachchi

Source Types

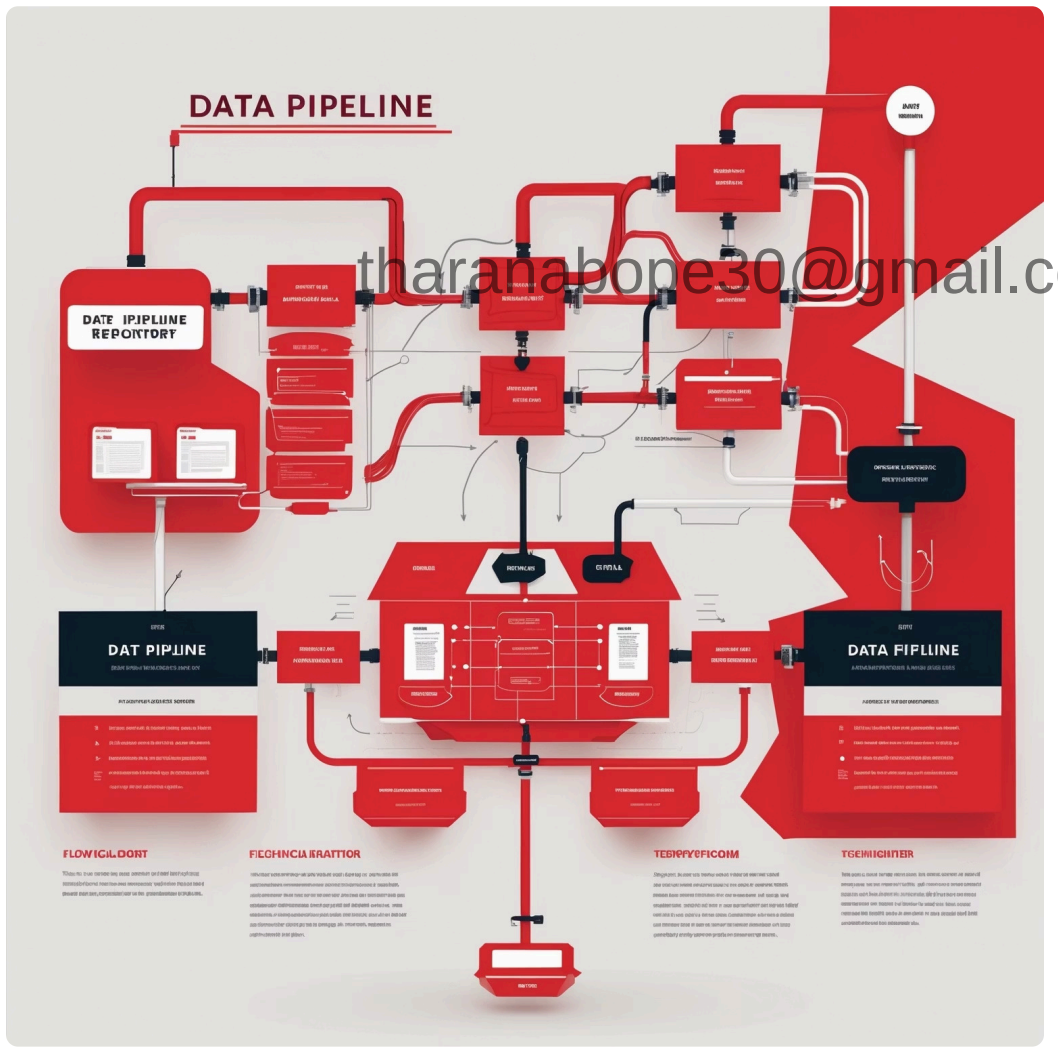
Internal	CRM, transaction logs, web analytics, customer support
External	Government datasets, APIs, purchased data, web scraping
Structured	Databases, CSV files, Excel spreadsheets
Unstructured	Text, images, audio, social media

{{mobile}}

0011

Documentation Needs

- Source origins and ownership
- Update frequency and timeliness
- Access methods and credentials
- Known quality issues or limitations
- Schema definitions and metadata
- Privacy and compliance considerations



Data Cleaning Overview

Bopearachchi

Missing Values

Identify gaps in data and apply appropriate imputation or removal strategies

Outliers

Detect extreme values that may represent errors or special cases requiring handling

{{mobile}}
tharanabope30@gmail.com

Invalid Formats

Correct inconsistent data types, formats, and units across the dataset

Duplicates

Identify and remove redundant records that could skew analysis

Questions to Ask During Cleaning

Data Sense Check

Does the data align with business understanding and domain knowledge?

- Are values within expected ranges?
- Do aggregates match known business metrics?
- Are relationships between variables logical?

Technical Validation

Is the data structurally sound and consistent?

- Are data types appropriate for each column?
- Is date/time formatting standardized?
- Are categorical variables consistently coded?
- Do summary statistics reveal potential issues?

Handling Missing Values

Tharana Bopearachchi

Detection Strategies

Missing data can be represented as NULL, NaN, empty strings, or special values like -999

```
# Python code for missing value detection
import pandas as pd
import numpy as np

# Check missing values
df.isna().sum()

# Visualize missing values
import missingno as msno
msno.matrix(df)
```

{{mobile}}

Treatment Options

1 Deletion

Drop rows or columns with high missingness (>30%)

2 Simple Imputation

Replace with mean/median (numerical) or mode (categorical)

3 Advanced Imputation

KNN, regression models, or MICE for preserving relationships

tharanabope30@gmail.com

4 Flagging

Create binary indicators to mark imputed values

Outlier Detection & Treatment

Tharana Bopearachchi

Detection Methods

1 Statistical

- Z-score: |z| > 3 standard deviations
- IQR method: x < Q1-1.5*IQR or x > Q3+1.5*IQR

2 Visual

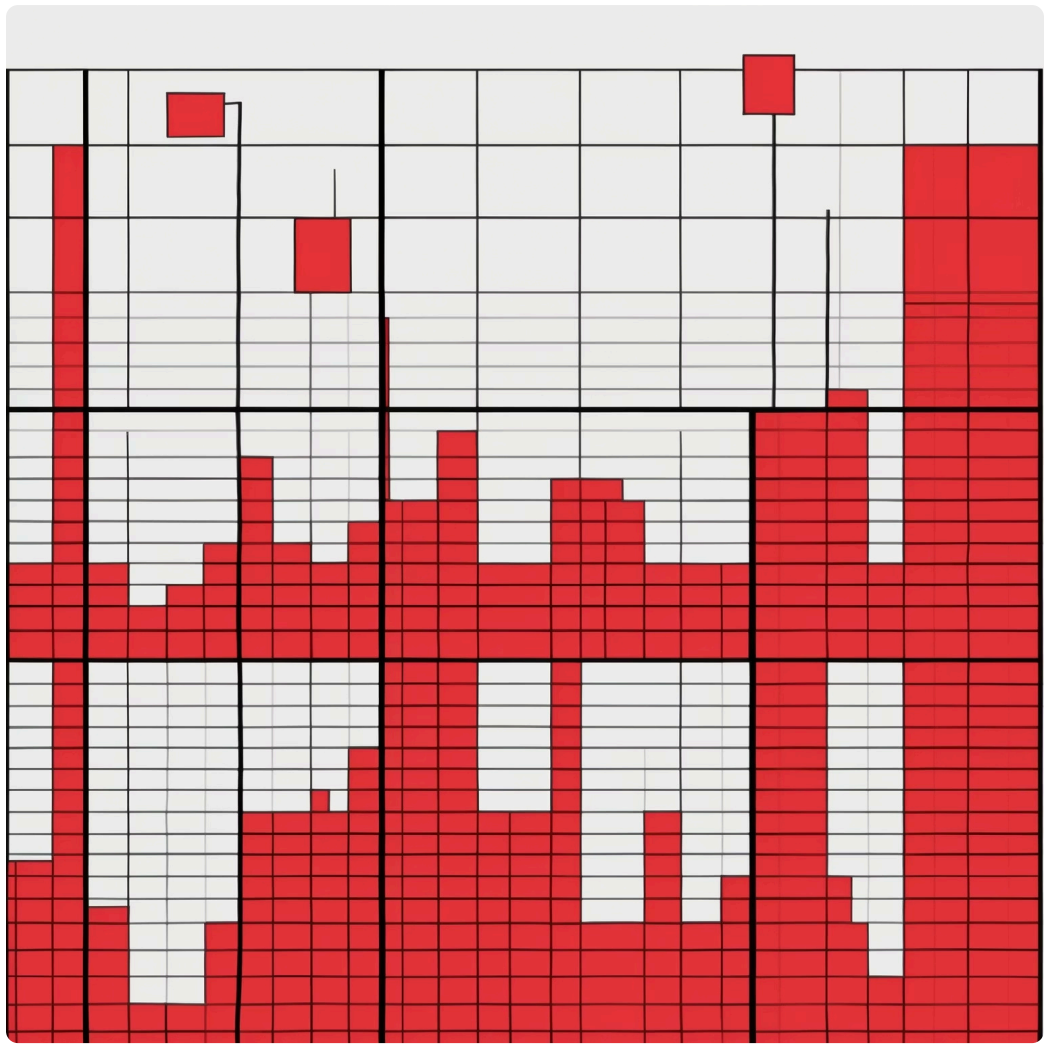
- Box plots
- Histograms
- Scatter plots

3 Machine Learning

- Isolation Forest
- DBSCAN clustering
- Local Outlier Factor

Treatment Strategies

Retain	Keep outliers if they represent valid special cases
Remove	Delete outliers if they're data errors
Transform	Apply log or other transformations to normalize distribution
Cap	Winsorize by replacing with percentile boundaries (e.g., 5th/95th)
Separate	Create special models for outlier segments



Handle Invalid Values

Common Invalid Data Types

- Negative values in strictly positive fields (e.g., age, quantity)
- Future dates in historical data
- Text in numeric fields
- Values outside physically possible ranges
- Inconsistent units (e.g., mixing meters and feet)
- Malformed identifiers or codes

Resolution Approaches

1 Data Type Conversion

Force appropriate types with validation

2 Range Enforcement

Clip values to valid domain-specific bounds

3 Standardization

Convert all values to consistent units/formats

4 Flagging

Mark suspicious values for review

Standardization / Feature Scaling

Tharana

Bopearachchi

Why Scale Features?

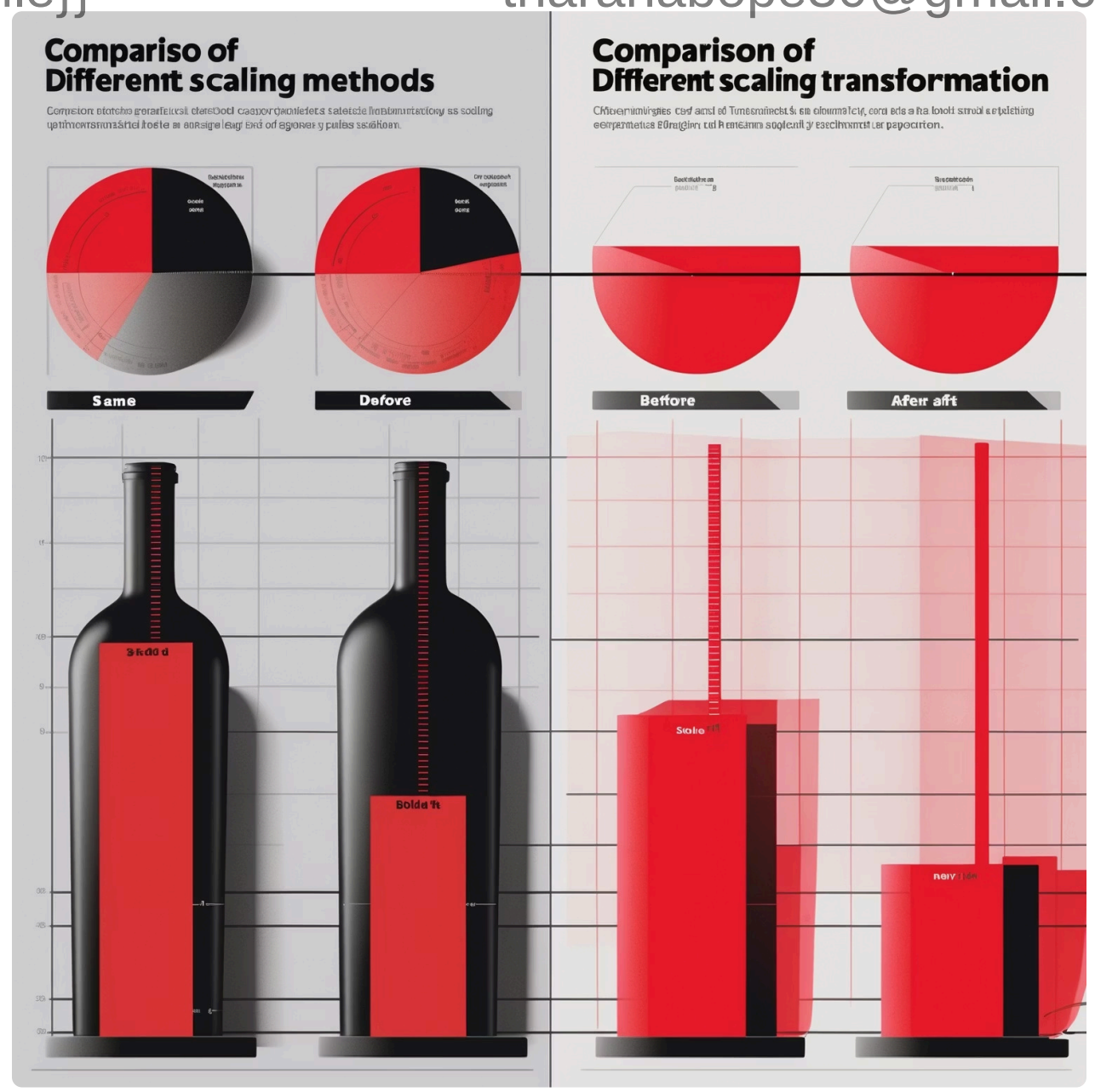
- Prevent features with large ranges from dominating
- Improve convergence speed for gradient-based algorithms
- Required for distance-based methods (KNN, k-means)
- Enables fair comparison between features
- Helps with regularization effectiveness

Scaling Methods

Min-Max Scaling	Scales values to [0,1] range x' = (x - min) / (max - min)
Standardization	Centers at mean=0, std=1 x' = (x - μ) / σ
Robust Scaling	Uses median and IQR instead of mean/std x' = (x - median) / IQR
Log Transform	Compresses wide ranges x' = log(x)

{{mobile}}

tharanabope30@gmail.com



Univariate Analysis

Tharana

Bopearachchi

Numerical Variables

- **Central Tendency:** Mean, median, mode
- **Spread:** Range, variance, standard deviation
- **Shape:** Skewness, kurtosis
- **Percentiles:** Quartiles, deciles, custom

Categorical Variables

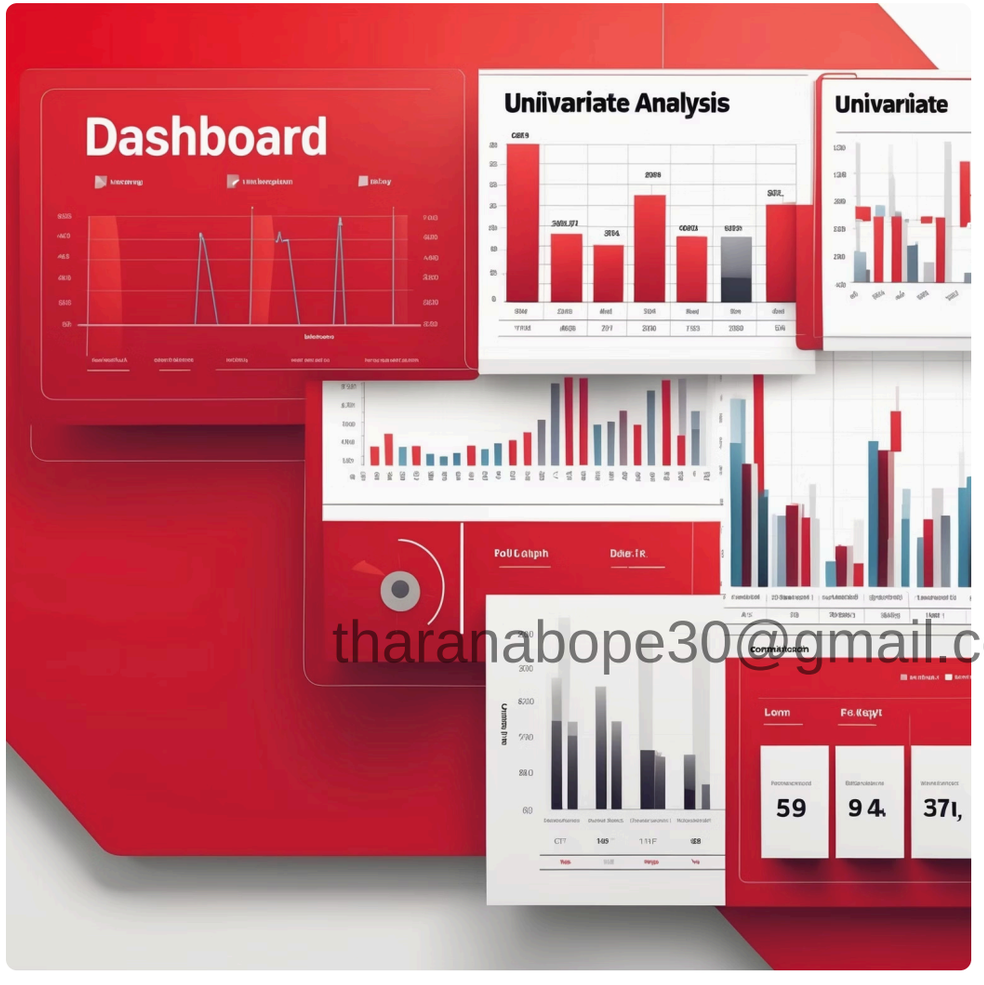
- **Frequency:** Count, proportion of each value
- **Cardinality:** Number of unique values
- **Distribution:** Balance or imbalance across categories

{{mobile}}

tharanabope30@gmail.com

0011

Visualization Tools



- **Numerical:** Histogram, density plot, boxplot, violin plot
- **Categorical:** Bar chart, pie chart, Pareto chart
- **Time Series:** Line chart, seasonal decomposition

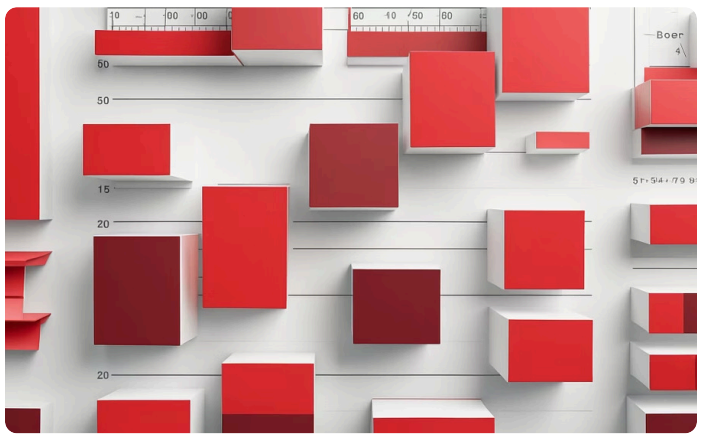
Tharana

Bivariate Analysis

Bopearachchi

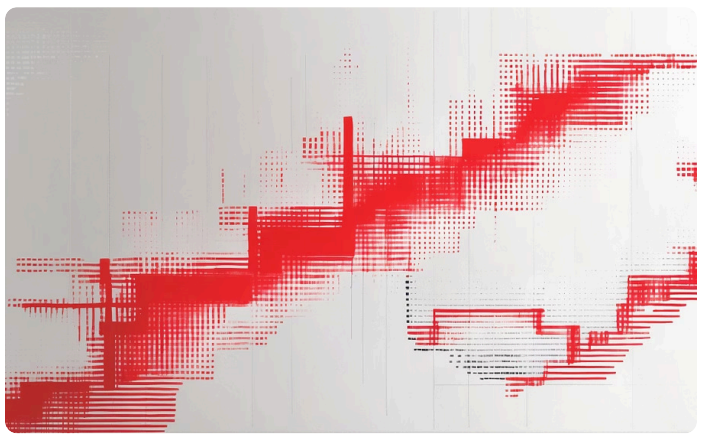
Numerical-Numerical

Scatter plots, correlation coefficients (Pearson, Spearman), and hexbin plots for high-density data



Numerical-Categorical

Box plots, violin plots, and bar charts with error bars to compare distributions across groups



Categorical-Categorical

Contingency tables, stacked bar charts, and heatmaps to visualize frequency distributions

{{mobile}}

tharanabope30@gmail.com

Bivariate analysis reveals relationships that aren't visible when examining variables in isolation, highlighting potential predictive power and interactions.

Multivariate Analysis

Tharana

Bopearachchi

Techniques

- **Correlation Matrix:** Identify relationships between all variable pairs
- **Partial Correlation:** Control for confounding variables
- **Dimensionality Reduction:** PCA, t-SNE, UMAP
- **Conditional Analysis:** Examine relationships within subgroups
- **ANOVA:** Compare means across multiple groups

Key Questions

- How do variables interact together?
- Are there complex relationships that simple correlations miss?
- What feature combinations best separate target classes?

Visualization Methods

- **Heatmaps:** Color-coded correlation matrices
- **Pair Plots:** Matrix of scatterplots for all variable combinations
- **Parallel Coordinates:** Visualize high-dimensional patterns
- **3D Plots:** Examine three variables simultaneously
- **Facet Grids:** Split visualization by categorical variables

Derived Metrics

Ratios

Create meaningful business metrics by combining variables (e.g., cost-to-income ratio, conversion rate, efficiency metrics)

Time-Based

Derive temporal insights with features like days_since_last_purchase, average_monthly_spend, or seasonal indicators

Binary Flags

Create indicator variables for important conditions (e.g., is_new_customer, has_returned_product, exceeds_threshold)

Aggregations

Summarize related data points (e.g., total_lifetime_value, frequency_count, variance_of_purchases)

Derived metrics often have more predictive power than raw variables because they encode domain knowledge and business logic.

Feature Binning

Tharana

Bopearachchi

Binning Methods

1 Equal Width

Divides range into equal-sized intervals

Example: Age 20-30, 30-40, 40-50, etc.

2 Equal Frequency

Creates bins with equal number of observations

Example: Income percentiles (0-25%, 25-50%, etc.)

3 Domain-Based

Uses business knowledge to create meaningful groups

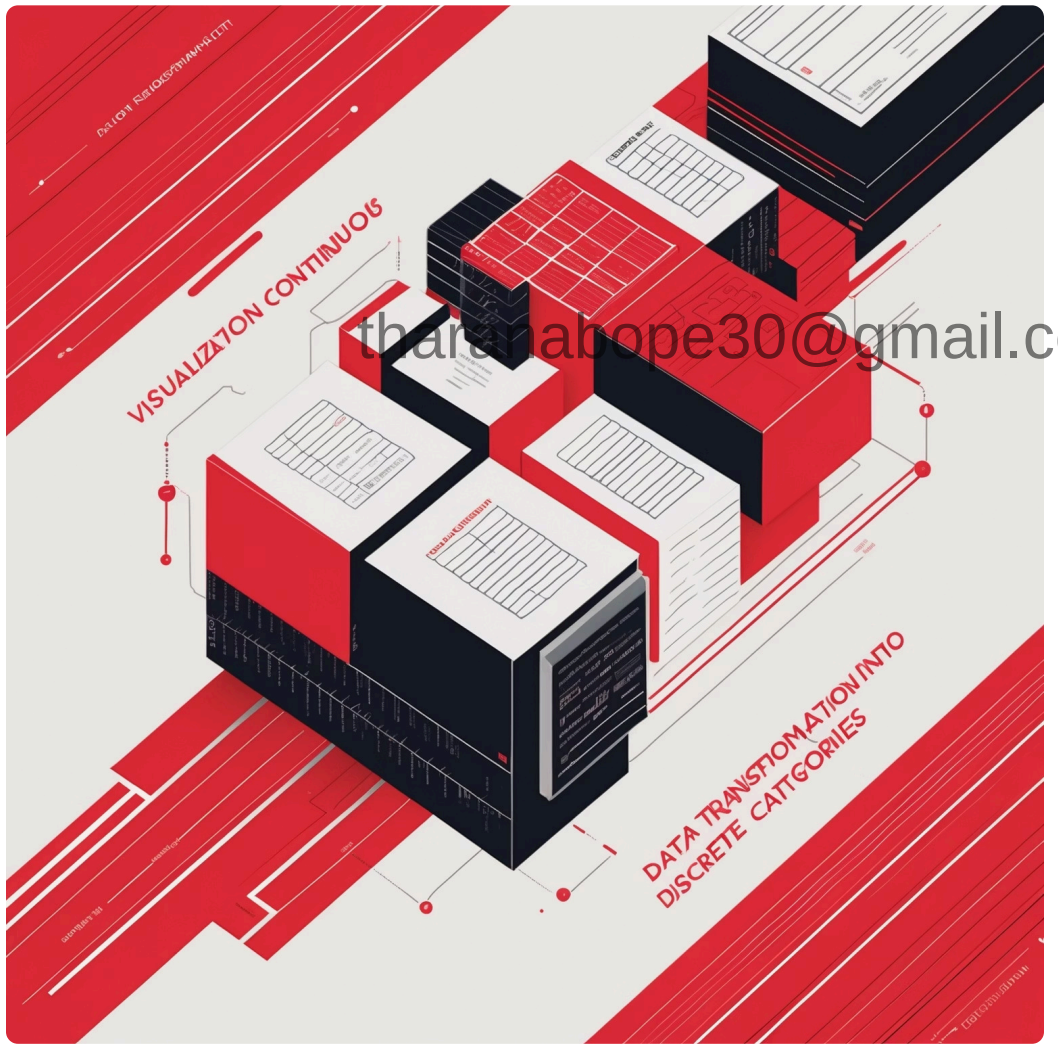
Example: Age groups (Child, Teen, Adult, Senior)

{{mobile}}

0011

Benefits of Binning

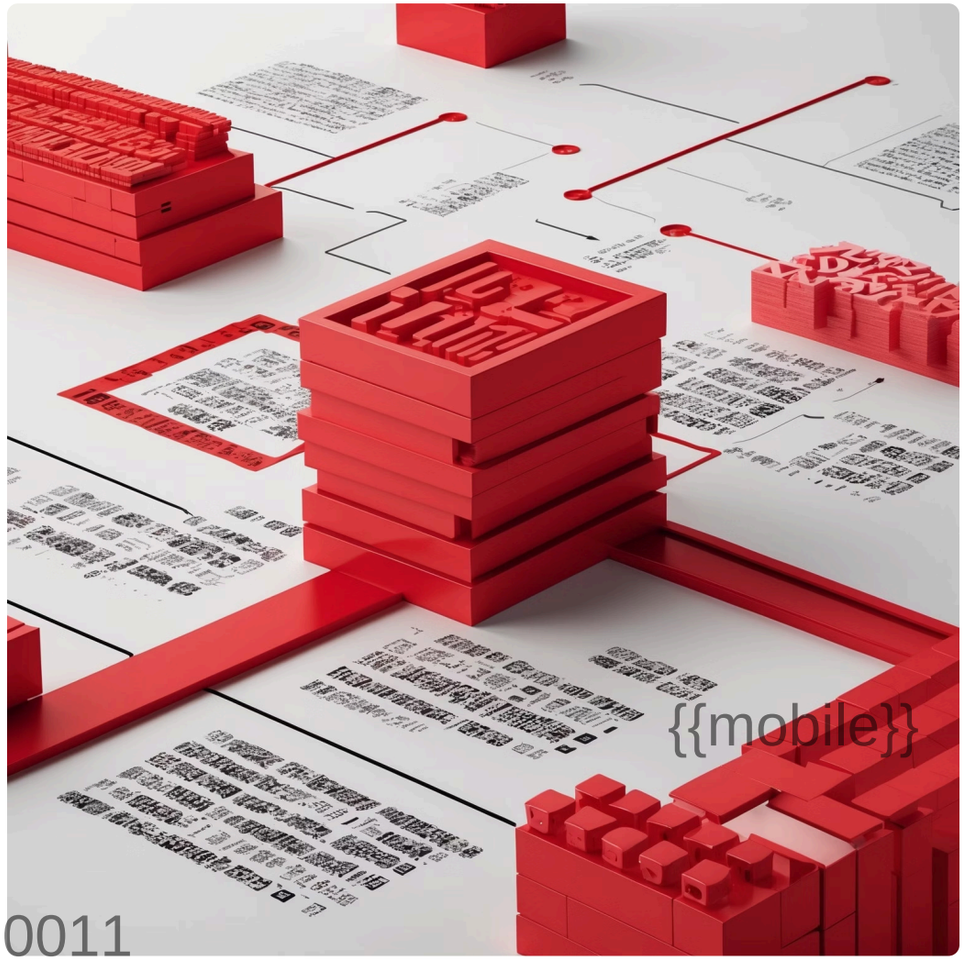
- Handles non-linear relationships
- Reduces impact of outliers
- Creates interpretable features
- Handles missing values (as separate bin)
- Can improve model performance for tree-based models



Feature Encoding

Tharana

Bopearachchi



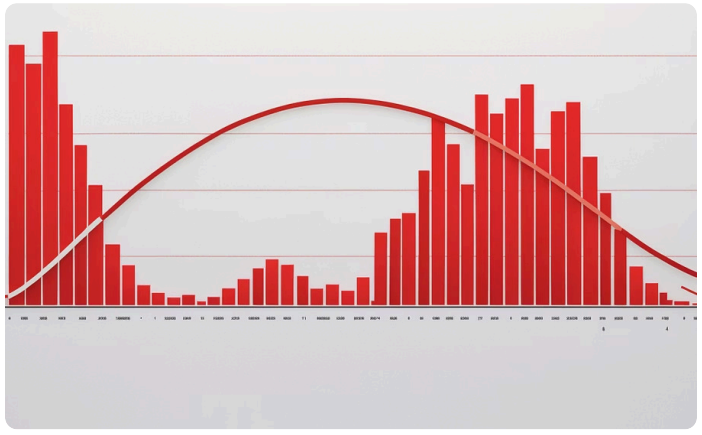
0011

Encoding Techniques

Label Encoding	Assigns integer to each category (0, 1, 2...) Best for: Ordinal data
One-Hot Encoding	Creates binary columns for each category Best for: Nominal data with few categories
Target Encoding	Replaces category with target mean Best for: High cardinality predictive features
Binary Encoding	Represents integers as binary code Best for: High cardinality with limited memory
Frequency Encoding	Replaces category with its frequency Best for: When frequency matters

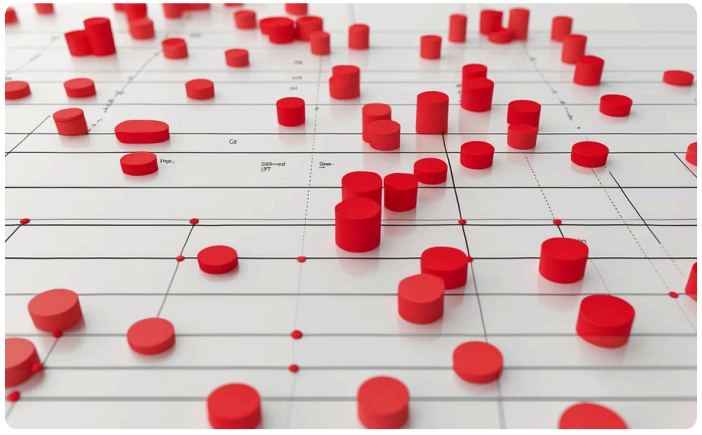
Visualization in EDA

Tharana Bopearachchi



Distribution Plots

Histograms, density plots, and box plots reveal data shape, central tendency, and spread



Relationship Plots

Scatter plots, pair plots, and correlation heatmaps visualize connections between variables

Temporal Plots

Line charts, area charts, and calendar heatmaps display patterns over time

Visualization Best Practices

{{mobile}} tharanabope30@gmail.com

Choose appropriate chart types for your data, maintain consistent color schemes, label axes clearly, and include explanatory titles. Focus on clarity and insight over visual complexity.

Numerical Analysis Tools

Tharana

Bopearachchi

Summary Statistics

```
# Python code for numerical analysis
import pandas as pd
import numpy as np

# Basic statistics
df.describe()

# Advanced statistics
from scipy import stats

# Normality test
stats.shapiro(df['numerical_column'])

# Correlation matrix
df.corr()
```

{{mobile}}

0011

Key Metrics to Examine

Central Tendency	Mean, Median, Mode
Dispersion	Standard Deviation, Variance, Range, IQR
Shape	Skewness, Kurtosis
Position	Percentiles, Quartiles
Relationship	Correlation, Covariance

Categorical Feature Profiling

Tharana

Bopearachchi

Analysis Techniques

1 Frequency Analysis

Count and percentage of each category value

2 Cardinality Assessment

Number of unique values and their distribution

3 Target Relationship

Average target value per category (e.g., churn rate by state)

{{mobile}}

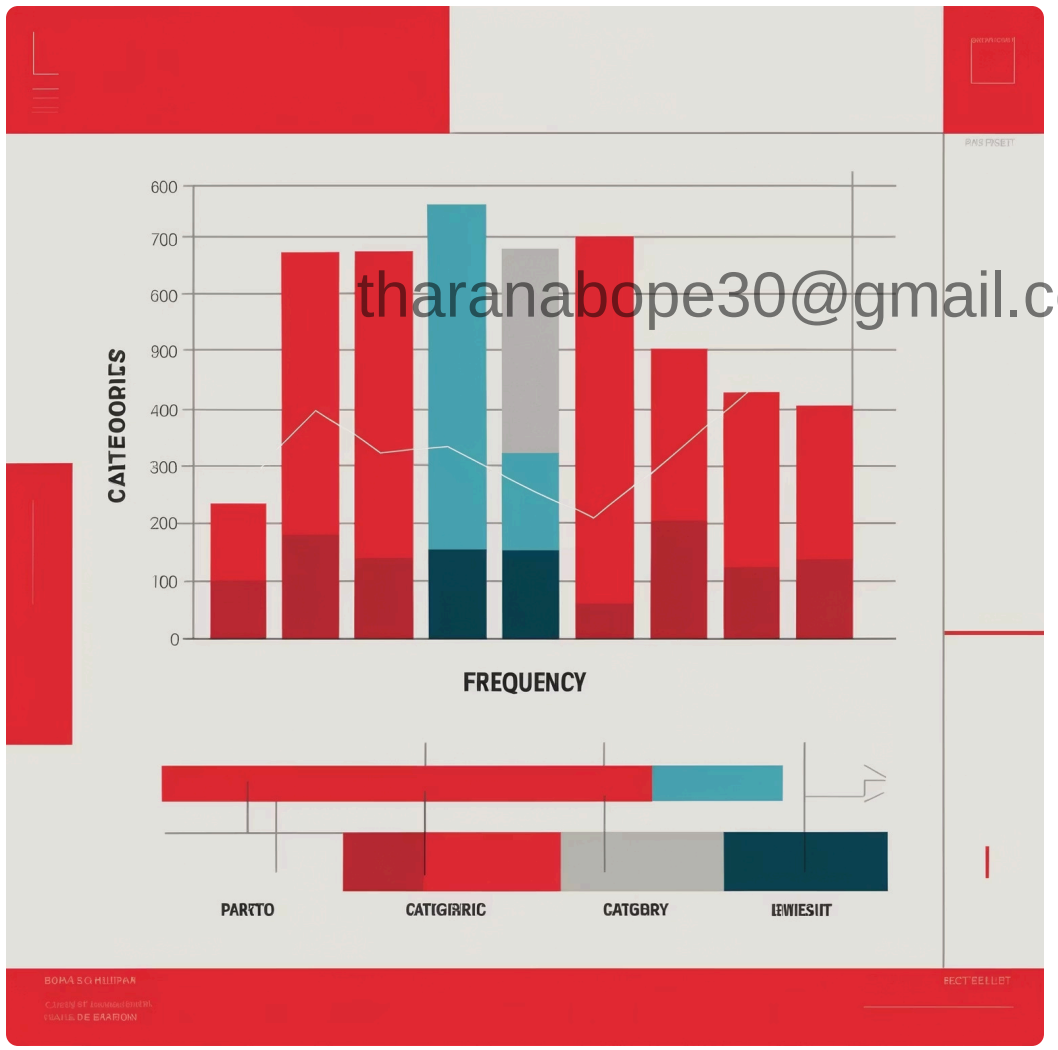
4 Missing Value Patterns

Percentage and distribution of nulls

0011

Handling Strategies

- **High Cardinality:** Group rare categories into "Other"
- **Sparse Categories:** Consider consolidating similar values
- **Hierarchical Categories:** Create broader groupings
- **Encoding Strategy:** Choose based on cardinality and predictive power



Tharana

Class Imbalance & Target Analysis

Bopearachchi

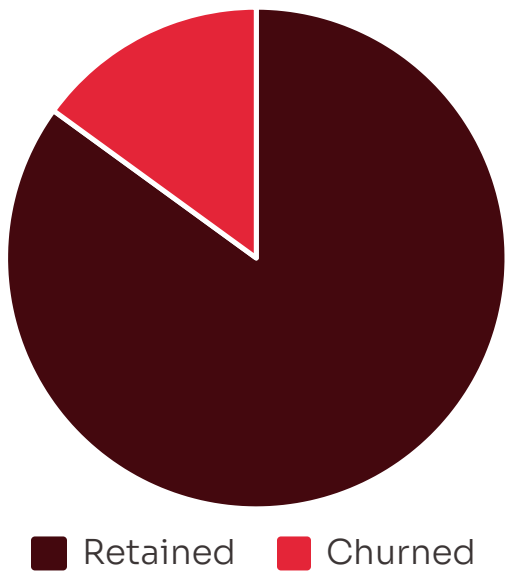
Target Variable Assessment

- Check distribution (balanced vs. imbalanced)
- Identify class proportions
- Examine target across key segments
- Analyze temporal trends in target

Implications of Imbalance

- Models may bias toward majority class
- Standard accuracy metrics can be misleading
- Rare events (fraud, churn) typically imbalanced
- Sampling strategies may be needed

{{mobile}}



A typical churn dataset might show significant class imbalance (e.g., 15% churn rate), requiring special handling during modeling.

tharanabope30@gmail.com

Tharana

Bopearachchi

EDA Use Cases

Customer Churn Prediction

Identify leading indicators and risk factors that signal potential customer departures before they happen

Fraud Detection

Discover unusual patterns and anomalies that may indicate fraudulent transactions or activities

Health Risk Assessment

Analyze patient data to predict disease risk and recommend preventive interventions

Recommendation Systems

Understand user preferences and item characteristics to create personalized suggestions

{{mobile}}

tharanabope30@gmail.com

Best Practices

Combine Visual & Statistical

Use both visualization and statistical tests to validate findings, as each approach catches different issues

Document Everything

Keep detailed notes on data sources, assumptions, transformations, and decisions to ensure reproducibility

Validate with Domain Experts

Consult with business stakeholders to ensure insights align with real-world understanding

Iterate Continuously

Revisit EDA as new data arrives or business conditions change, treating it as an ongoing process

Effective EDA is as much about scientific rigor and communication as it is about technical skills. Document your process, share insights clearly, and maintain healthy skepticism about initial findings.

Final Feature Summary

Tharana

Bopearachchi

Dataset Transformation Journey

Raw Data

Initial uncleaned dataset with issues

Cleaned Data

Missing values handled, outliers addressed

Transformed Data

Normalized, encoded, balanced features

{{mobile}}

Engineered Data

New features added, irrelevant ones removed

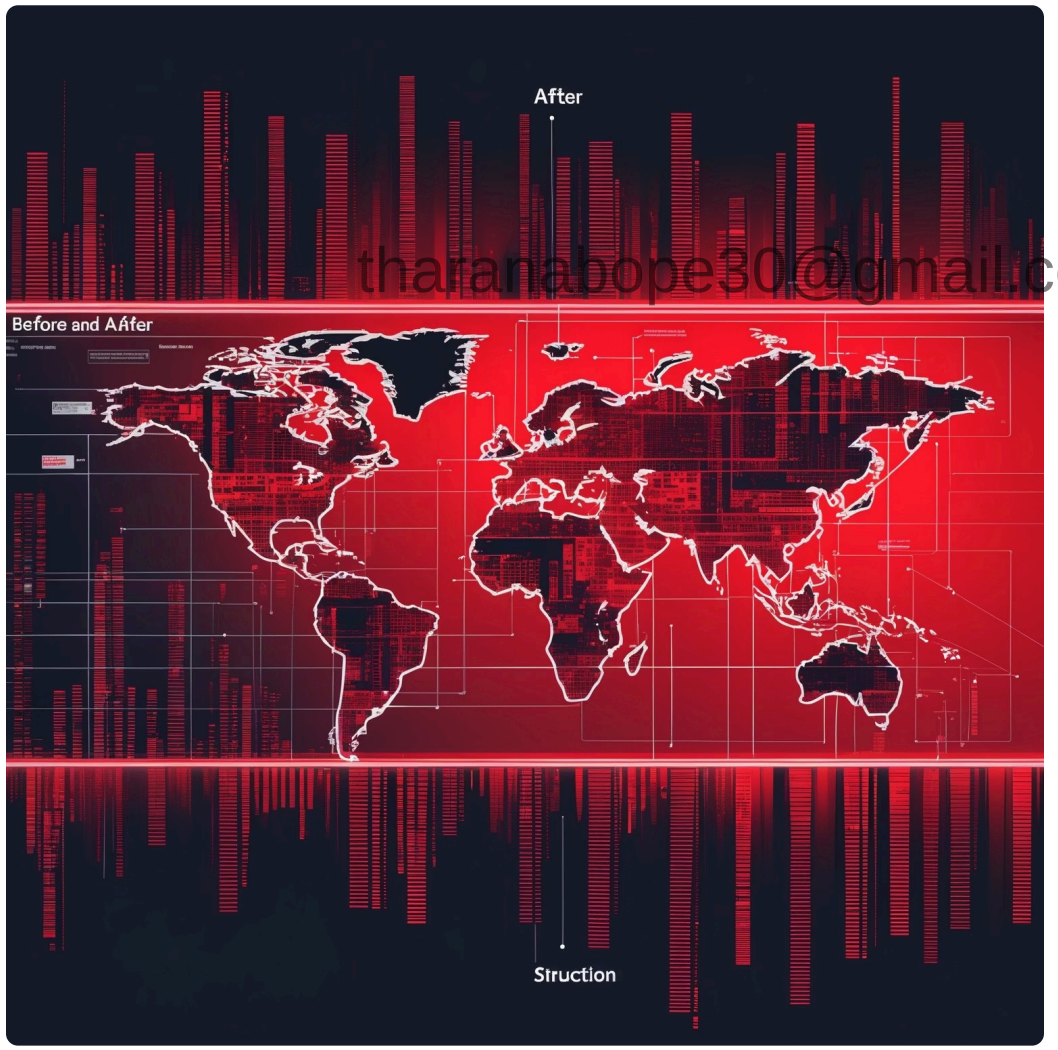
0011

Model-Ready Data

Final dataset prepared for algorithms

Documentation Checklist

- Feature dictionary with descriptions
- Transformation steps and rationale
- Statistical summaries before/after
- Key insights from EDA process
- Known limitations or caveats
- Expected feature importance



Strategic Insights

Tharana

Bopearachchi

Data-Driven Recommendations

EDA findings should directly inform business strategy and model development:

- Feature importance hypotheses based on correlation analysis
- Customer segments with highest churn risk identified
- Behavioral patterns that precede customer departure
- Product usage trends that correlate with retention

{{mobile}}

tharanabope30@gmail.com

Segment Focus Strategy

Based on EDA, prioritize retention efforts on:

- High-value customers showing early warning signs
- Segments with highest predicted churn probability
- Customers reaching critical lifecycle milestones
- Groups with declining engagement metrics