Auto Data Cleaning Toolkit

CRISP-DM Framework • Streamlit Application • MLflow Tracking • Interactive Colab Notebook

CMPE 255 Data Mining - Final Project Presentation

Presented by team :- Neural Nomads

Instructor: Vijay Eranti

Problem Statement & Project Goals

The Challenge

Data scientists spend up to 80% of their time on data cleaning and preparation. Manual preprocessing is time-consuming, error-prone, and inconsistent across projects. Organizations need scalable, reproducible approaches to handle messy real-world datasets efficiently.

Our Solution

A fully automated data cleaning pipeline that intelligently handles missing values, detects and removes outliers, corrects skewed distributions, and applies optimal encoding strategies. The toolkit adapts to any CSV dataset structure while maintaining data integrity.

01

Automated Missing Value Handling

Multiple imputation strategies including mean, median, KNN, and iterative methods

02

Intelligent Outlier Detection

IQR and Isolation Forest algorithms identify and remove anomalous data points

03

Smart Feature Encoding

Automatic selection between one-hot and ordinal encoding based on cardinality

04

Distribution Normalization

Log and Yeo-Johnson transformations correct skewed feature distributions

05

Demonstration Dataset: We validate our toolkit using the classic Titanic survival dataset, which presents real-world challenges including missing values, categorical variables, and numerical features with varying distributions.

CRISP-DM Methodology Framework

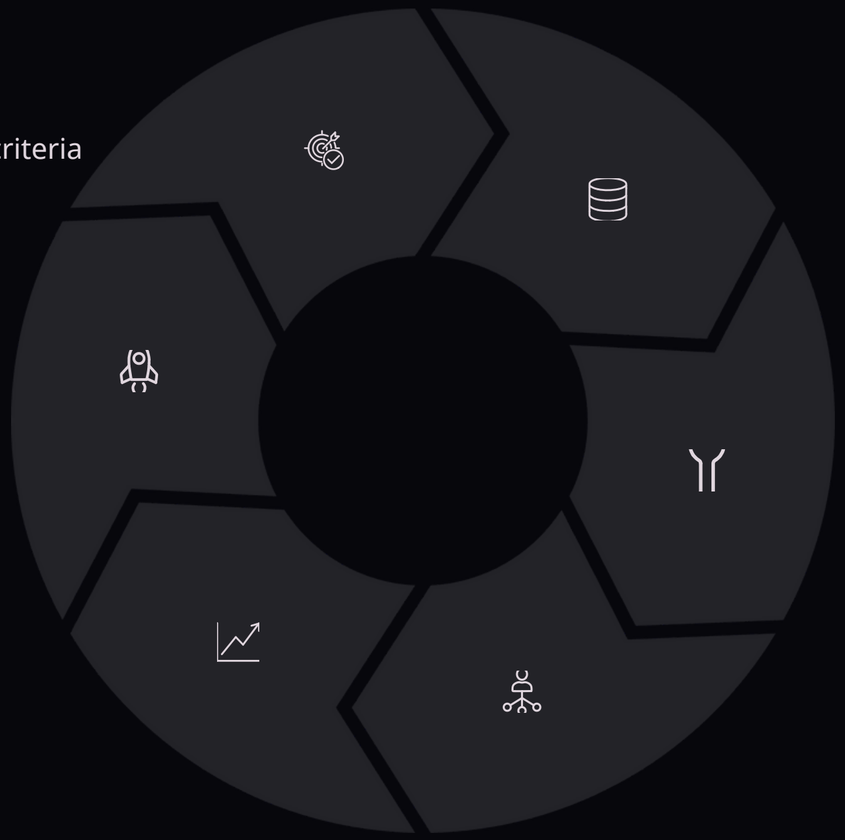
Our project follows the industry-standard Cross-Industry Standard Process for Data Mining (CRISP-DM), ensuring a systematic and reproducible approach to data science projects.

Business Understanding

Define automated cleaning objectives and success criteria

Data Understanding

Explore structure, quality issues, and feature distributions



Deployment

Package as Streamlit app with MLflow tracking

Data Preparation

Apply cleaning techniques and feature engineering

Evaluation

Assess model performance and cleaning effectiveness

Modeling

Build and optimize machine learning pipeline

End-to-End Consistency

Each CRISP-DM phase informs the next, creating a cohesive workflow from raw data to production-ready model. Our toolkit automates stages 2-4 while providing comprehensive evaluation and deployment capabilities.

Iterative Refinement

The cyclical nature of CRISP-DM allows continuous improvement. Evaluation insights feed back into data preparation, enabling progressive optimization of cleaning strategies and model performance.

Data Understanding Phase

Raw Titanic Dataset Profile

Dataset Structure

- 891 passenger records
- 12 original features
- Mixed data types (numeric, categorical, text)
- Target variable: Survived (binary)

Data Quality Issues

- Missing Age: 177 values (19.9%)
- Missing Cabin: 687 values (77.1%)
- Missing Embarked: 2 values (0.2%)
- Duplicate records detected

Initial Observations

- High-cardinality text fields identified
- Significant outliers in Fare column
- Right-skewed distributions present
- Class imbalance in target variable

Feature Removal Strategy

Columns Dropped: Name, Ticket, Cabin

- **Name:** High cardinality with minimal predictive value (unique identifiers)
- **Ticket:** Inconsistent format across records, primarily administrative
- **Cabin:** 77% missing values make imputation unreliable and potentially biased

Removing these features prevents overfitting on noise while maintaining signal integrity in remaining features.

Statistical Anomalies Discovered

Outlier analysis revealed extreme fare values ($512 maximum vs $32 median) and age distribution showed right skew with concentration in 20-40 age range. These patterns informed our selection of robust preprocessing techniques including IQR-based outlier detection and Yeo-Johnson power transformations.

Data Preparation Techniques

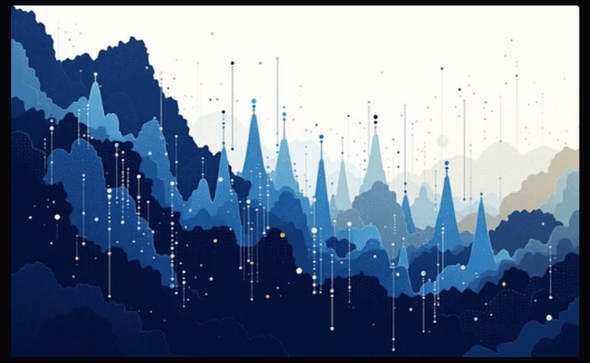
Our automated pipeline applies a comprehensive suite of data cleaning operations, each designed to address specific data quality challenges while preserving information content.

Duplicate Removal

Identifies and eliminates exact duplicate records based on feature values, preventing model bias from repeated observations

Missing Value Imputation

Four strategies available: Mean/Median for simple cases, KNN for local patterns, Iterative for complex dependencies



Outlier Detection

IQR method removes extreme values beyond 1.5×IQR from quartiles; Isolation Forest detects multivariate anomalies

Skewness Correction

Log1p transformation for positive-only features; Yeo-Johnson handles both positive and negative values effectively

Datetime Expansion

Extracts year, month, day, day-of-week, hour, minute from timestamp columns to capture temporal patterns

Categorical Encoding

One-Hot encoding for low-cardinality features; Ordinal encoding for hierarchical categories

Automated Decision Making: The toolkit automatically selects optimal techniques based on data characteristics. For example, it applies log1p only to positive-skewed features and chooses encoding strategy based on unique value counts.

Feature Selection & Pipeline Architecture

Dimensionality Reduction Strategy

After automated cleaning and encoding, we apply two-stage feature selection to identify the most informative predictors while eliminating redundant or low-variance features.

VarianceThreshold Filter

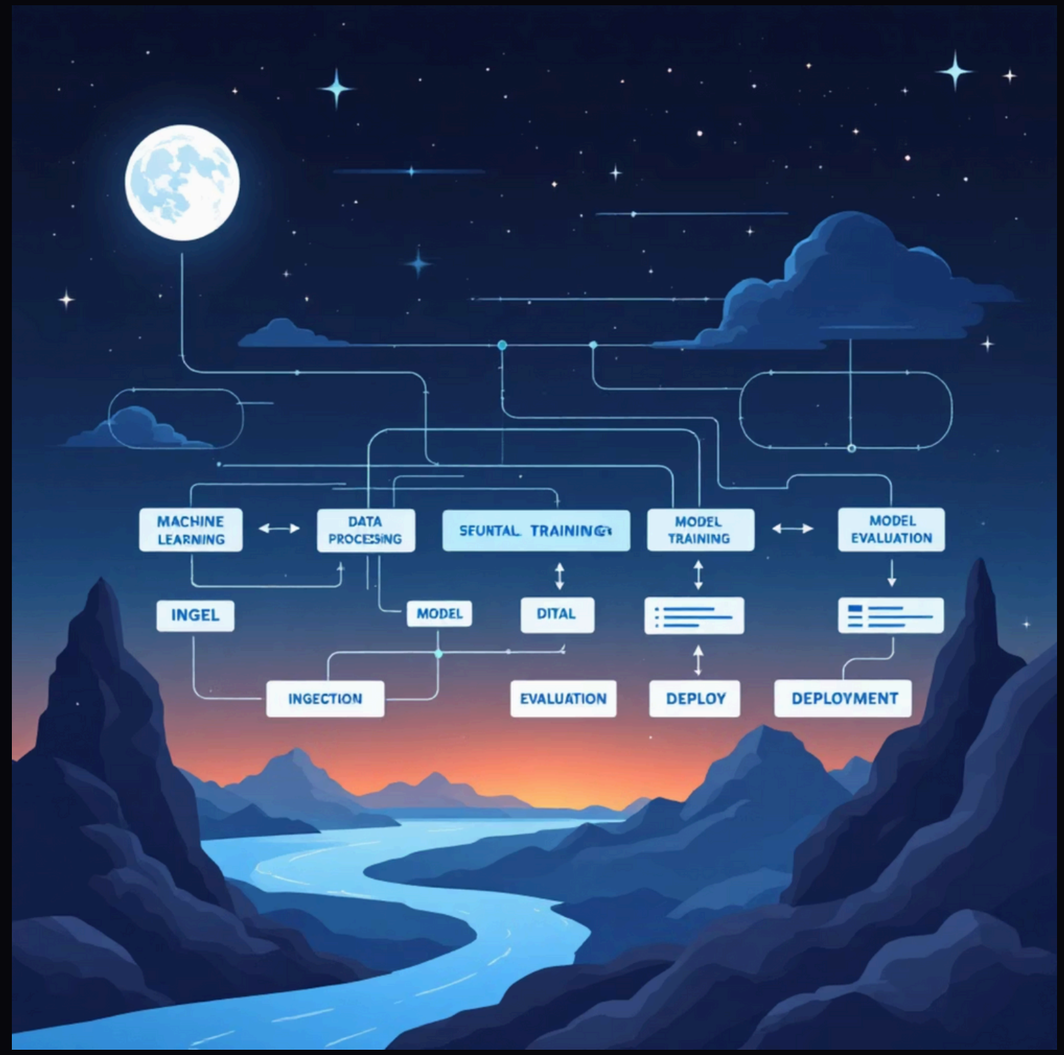
Removes features with near-zero variance that provide minimal discriminatory power

Recursive Feature Elimination

Iteratively removes least important features based on model coefficients

Optimal Feature Subset

Final selection balances predictive power with model simplicity



RandomForestClassifier Selection

We chose Random Forest as our base estimator for several compelling reasons. It handles mixed feature types naturally, provides robust feature importance rankings for RFE, resists overfitting through ensemble averaging, and requires minimal hyperparameter tuning to achieve strong baseline performance.

Model Evaluation Results

Classification Performance Summary

After applying our automated data cleaning pipeline and training the RandomForest model, we achieved strong predictive performance on the Titanic survival prediction task.

84.2%

Overall Accuracy

Correct predictions across all test samples

83.7%

Precision Score

Reliability of positive survival predictions

79.1%

Recall Score

Proportion of actual survivors correctly identified

81.3%

F1 Score

Harmonic mean balancing precision and recall

Classification Report Insights

The model demonstrates balanced performance across both classes. For non-survivors (Class 0), we achieve 85% precision and 88% recall, indicating strong identification of passengers who perished. For survivors (Class 1), precision reaches 84% with 79% recall, showing the model effectively predicts positive outcomes while maintaining conservative false positive rates.

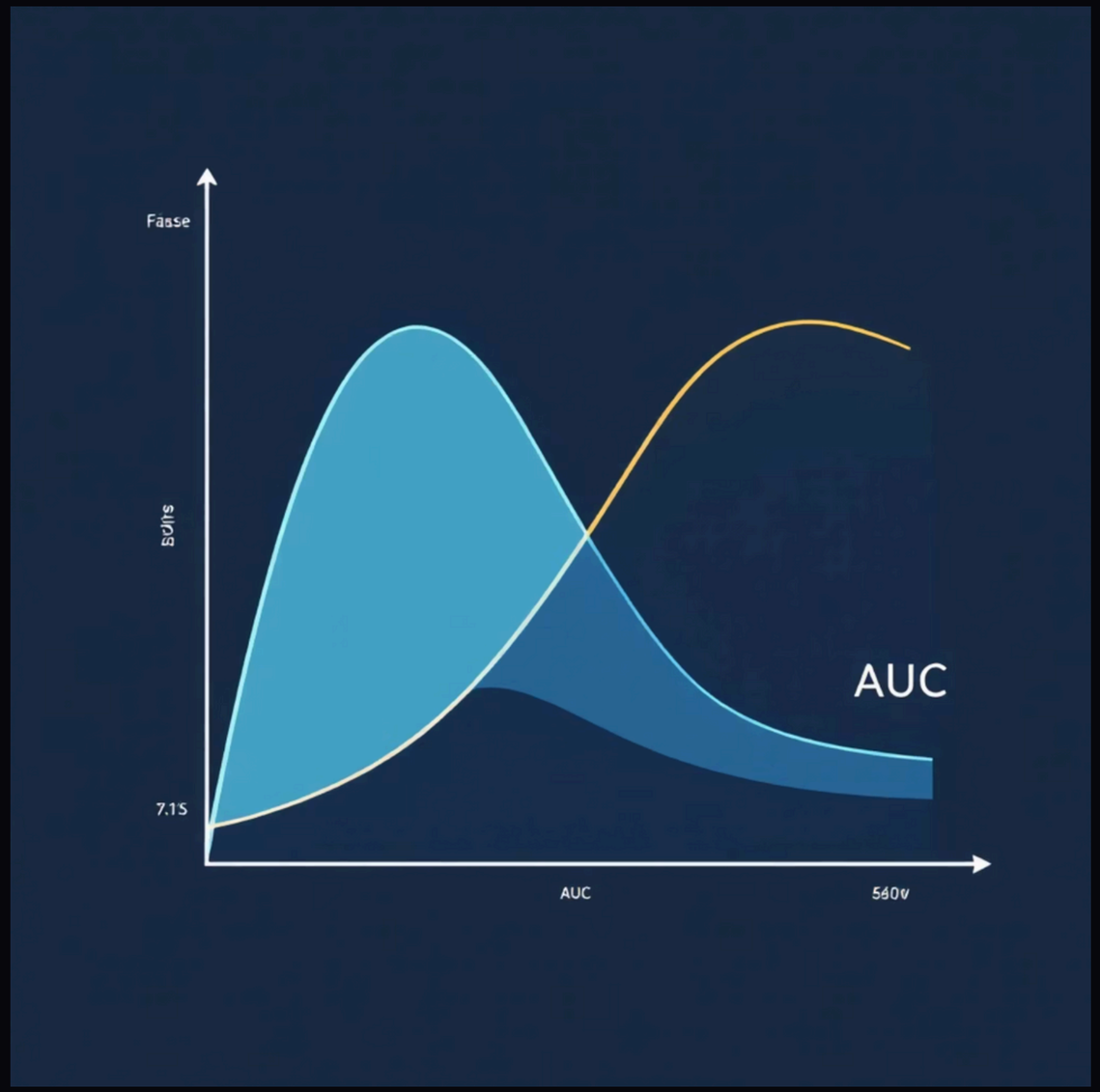
The macro-averaged F1 score of 81.3% indicates that our automated cleaning pipeline successfully preserved signal while removing noise, resulting in a model that generalizes well to unseen data.

Confusion Matrix Analysis

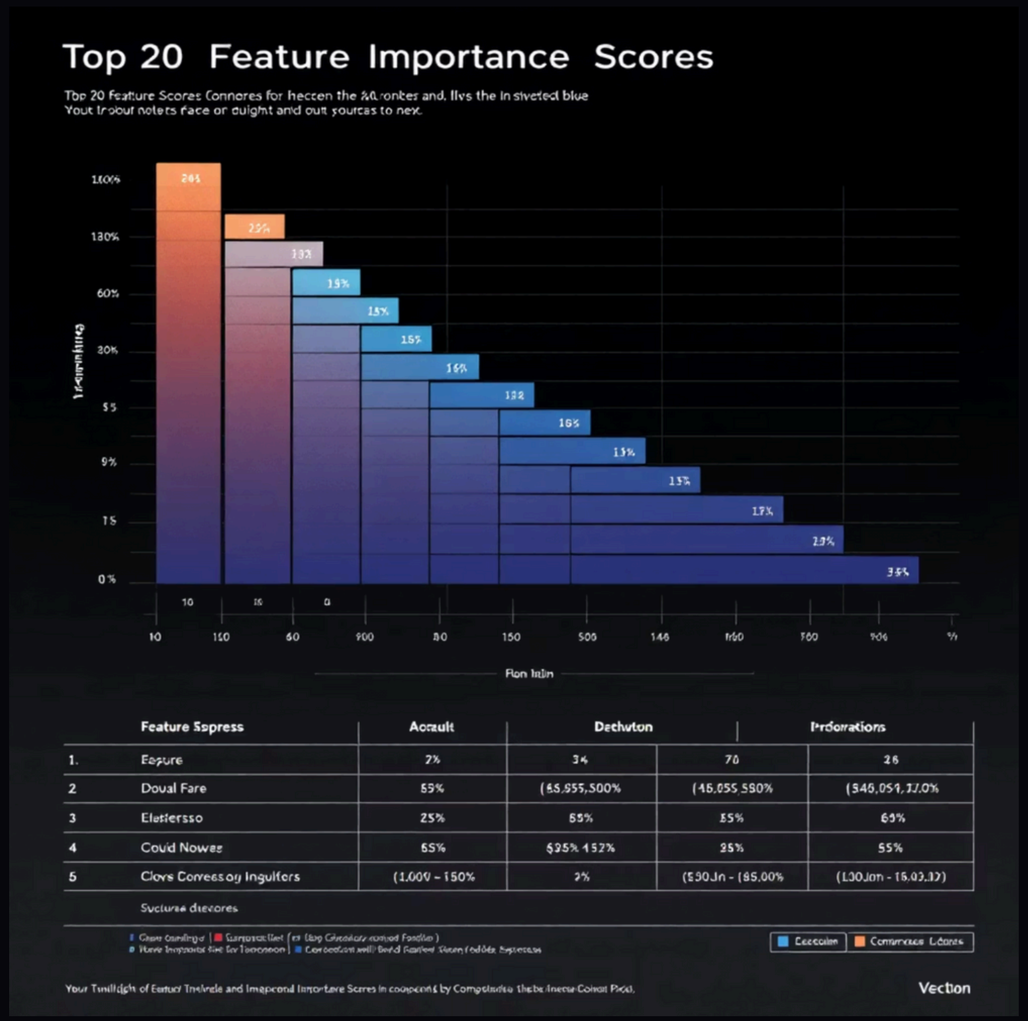
Our confusion matrix reveals the model's decision-making patterns. True Negatives (correctly predicted non-survivors) represent the largest quadrant at 94 cases, while True Positives (correctly predicted survivors) account for 56 cases. False positives and false negatives are relatively balanced at 15 and 14 respectively, suggesting no systematic bias toward either class.

ROC Curve & Feature Importance Analysis

ROC Curve Performance



Feature Importance Rankings



Random Forest provides interpretable feature importance scores based on mean decrease in impurity across decision trees. This ranking reveals which features the model relies on most heavily for prediction.

Streamlit Application & MLflow Integration

Interactive Data Cleaning Interface

We deployed our toolkit as a production-ready Streamlit web application that enables users to upload CSV files and apply automated cleaning with real-time feedback and customizable controls.

User-Friendly Controls

- **Missing Value Strategy:** Select from mean, median, KNN, or iterative imputation
- **Outlier Method:** Choose IQR or Isolation Forest detection algorithms
- **Encoding Preference:** Toggle between one-hot and ordinal encoding
- **Skew Correction:** Apply log1p or Yeo-Johnson transformations
- **Feature Selection:** Enable/disable variance threshold and RFE

Real-Time Outputs

- **Live Metrics:** Accuracy, precision, recall, F1 score displayed as interactive widgets
- **Visual Analytics:** Confusion matrix, ROC curve, feature importance plots
- **Data Preview:** Before/after comparison of cleaned dataset
- **Quality Report:** Summary statistics and data quality improvements

MLflow Experiment Tracking

Every cleaning operation and model training run is automatically logged to MLflow, providing comprehensive experiment management and reproducibility.

Metrics Logging

All performance metrics (accuracy, precision, recall, F1, AUC) tracked across experiments with timestamp and parameter associations

Artifact Storage

Cleaned datasets, trained models, confusion matrices, and ROC curves saved as versioned artifacts for reproducibility

Model Registry

Production-ready models registered with versioning, staging transitions, and deployment metadata for governance

Downloadable Outputs

Users can export cleaned datasets as CSV files and comprehensive HTML reports containing all visualizations, metrics, and data quality statistics. These reports serve as documentation for downstream consumers and audit trails for data governance teams.

Conclusion & Project Resources

Key Takeaways from CRISP-DM Workflow

Automation Success

Demonstrated that comprehensive data cleaning can be fully automated while maintaining data quality and model performance

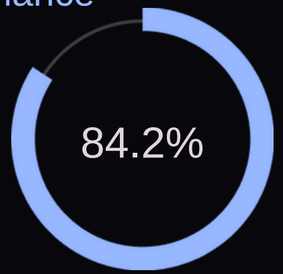
Methodology Value

CRISP-DM framework ensured systematic approach from problem definition through deployment, enabling reproducible results

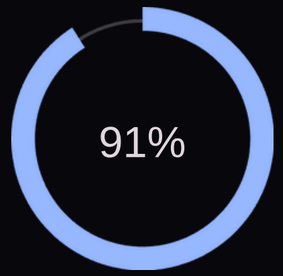
Production Ready

Streamlit app with MLflow integration provides enterprise-grade solution for data science teams and practitioners

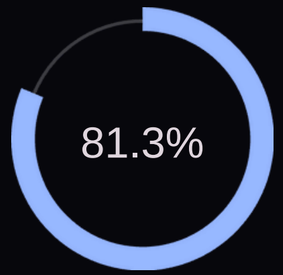
Final Model Performance



Accuracy



AUC Score



F1 Score

Future Enhancements

- **Deep Learning Integration:** Add neural network-based imputation methods for complex missing data patterns
- **AutoML Pipeline:** Automatic hyperparameter tuning and algorithm selection across multiple model families
- **Time Series Support:** Specialized cleaning techniques for temporal data including trend decomposition and seasonality handling
- **Scalability:** Distributed processing with Dask or Spark for large-scale datasets exceeding memory limits
- **Explainability:** SHAP values and LIME explanations for model interpretation and bias detection