

DevifyX Assignment

Customer Churn Prediction

Logistic Regression & XGBoost

Assignment Deadline: 7 Days

Submission Form: <https://forms.gle/UEUafAtyPUEm2ZjbA>

Objective

The goal of this assignment is to design and implement a machine learning solution to predict customer churn using both Logistic Regression and XGBoost algorithms. You will work with a real-world dataset, perform exploratory data analysis, preprocess the data, build predictive models, and evaluate their performance. This assignment will assess your ability to apply ML techniques, interpret results, and communicate findings effectively.

Core Features

- 1. Data Exploration:** Load the provided customer dataset and perform exploratory data analysis (EDA) to understand the distribution, correlations, and potential issues in the data.
- 2. Data Preprocessing:** Handle missing values, encode categorical variables, and scale numerical features as appropriate.
- 3. Feature Engineering:** Create at least two new features that could help improve model performance.
- 4. Model Implementation:** Implement Logistic Regression and XGBoost classifiers for churn prediction.
- 5. Model Evaluation:** Evaluate models using appropriate metrics such as accuracy, precision, recall, F1-score, and ROC-AUC.
- 6. Hyperparameter Tuning:** Perform hyperparameter tuning for both models using grid search or randomized search.
- 7. Visualization:** Visualize key findings from EDA, feature importance, and ROC curves for both models.

Bonus Features

- Deploy the best-performing model as a simple REST API using Flask or FastAPI.

- Implement cross-validation and compare results with train-test split.
- Provide SHAP or LIME-based interpretation of model predictions.
- Create an interactive dashboard (e.g., using Streamlit) to display results.

Technical Requirements

- Use Python 3.7 or above.
- Use standard ML libraries: `pandas`, `numpy`, `scikit-learn`, `xgboost`, `matplotlib`, `seaborn`.
- Code should be modular and well-documented.
- Use Jupyter Notebook or Python scripts for your solution.
- Include a `requirements.txt` file with all dependencies.
- Use Git for version control.

Deliverables

- Source code (Jupyter Notebook or Python scripts).
- A brief report (max 3 pages) summarizing your approach, results, and insights.
- `requirements.txt` file.
- (Optional) API code or dashboard code (for bonus features).

Use of AI Tools

You are **permitted and encouraged** to use AI-based coding tools such as **GitHub Copilot**, **ChatGPT**, or similar platforms to assist with code generation, debugging, and documentation. However, the final submission should reflect your own understanding and structure.

Submission

- Push your code and report to a public GitHub repository.
- Submit the GitHub repository link via the submission form: <https://forms.gle/UEUafAtyPUEm2Zjba>

Evaluation Criteria

- **Correctness:** Does the solution meet the requirements and solve the problem?
- **Code Quality:** Is the code clean, modular, and well-documented?
- **Analysis:** Quality of EDA, feature engineering, and interpretation of results.
- **Model Performance:** Effectiveness of models and justification of choices.
- **Presentation:** Clarity and completeness of the report and visualizations.
- **Bonus:** Implementation of any bonus features.