Final Report

Zaher Khateeb - MLOps 24

We aim to build two robust machine learning pipelines to address the client's business problems: (without changing the model type or hyperparameters)

1. An improved model for predicting the success of marketing campaigns for term deposits using the bank marketing dataset.
2. An enhanced model for assessing credit risk using the German credit risk dataset.

Objectives:

- Develop an automatic feature selection step to improve model performance.
- Measure improvement using predefined ROC-AUC metric.

Customer Consumption:

- The marketing model will provide insights and predictions to optimize campaign strategies, targeting the most likely customers for term deposits.
- The credit risk model will offer more accurate risk assessments to inform lending decisions and reduce default rates.

Personnel

Y-Data:

- Project Lead: Zaher

Metrics

Qualitative Objectives:

- Enhance the efficiency of marketing campaigns.
- Improve the accuracy of credit risk assessments.

Quantifiable Metrics:

1. Marketing Model:
   - Baseline ROC-AUC: 0.89
   - Target improvement: Increase ROC-AUC by 2% (0.91)
2. Credit Risk Model:
   - Baseline ROC-AUC: 0.78
   - Target improvement: Increase ROC-AUC by 2% (0.80)

Analyses Execution

1. Iterating over the `analyses_to_run` list.
2. Checking if each analysis is defined in the `analysis_methods` dictionary.
3. For each valid analysis, iterating over the `trained_pipelines`.

4. Executing the specified analysis method on each pipeline.

Analyses Methods

Three specific types of analyses are defined in the `ModelImprover` class:

1. **Uncertainty Analysis**

   - **Purpose**: To understand the confidence level of the model's predictions.
   - **Method**: Uses a baseline ensemble Monte Carlo method to calculate the uncertainty of the model's predictions.

2. **Feature Importance Analysis**

   - **Purpose**: To determine the importance of different features used by the model.
   - **Method**: Uses SHAP values to plot feature importance and SHAP summary plots, and selects features based on their SHAP values.

3. **Feature Performance Analysis**

   - **Purpose**: To analyze the performance of individual features in contributing to the model's predictions.
   - **Method**: Assesses how changes in feature values affect model accuracy or other performance metrics, identifying weaknesses in the model's use of certain features.

Utility Classes and Methods

1. **Uncertainty**

   - **Methods**:
     - `baseline_ensemble_monte_carlo`: Calculates the uncertainty of the model's predictions using ensemble Monte Carlo simulations.

2. **Explainability**

   - **Methods**:
     - `plot_feature_importance`: Plots the importance of each feature.
     - `plot_shap_summary`: Creates a SHAP summary plot.
     - `select_features_based_on_shap`: Selects features based on their SHAP values.

3. **FeaturePerformanceWeaknessAnalyzer**

   - **Methods**:
     - `analyze_feature_performance`: Analyzes the performance of individual features.
     - `plot_metric_drops`: Plots the performance drops for vulnerable features.

Data and Model Parameters

Each analysis method receives a comprehensive set of parameters:

- **pipeline_name**: The name of the pipeline.
- **pipeline**: The pipeline object.

- **model**: The trained model.
- **all_feature_names**: List of all feature names.
- **X_train_processed**: Processed training dataset.
- **X_test_processed**: Processed test dataset.
- **X_test**: Original test dataset.
- **y_train**: Training labels.
- **y_test**: Test labels.
- **auc**: AUC score of the model.