ABSTRACT (DESCRIPTION)

The goal of this project was to use classification models to predict whether a credit card transaction was fraudulent or not. Using data obtained here, from Kaggle.com, features were analyzed to determine their impact and model predictive performance. Several learning models were evaluated and although the majority of models proved highly accurate, Decision Tree was determined to be best performing model with a high degree of interpretability.

DESIGN

The data, found here, classifies credit card transactions as either 'not fraud' – 0, or 'fraud' – 1. The data was exceptionally clean and was without any missing values – this made EDA extremely easy. Several models were evaluated by observing their baseline performance, before selecting the ideal model for further tuning and optimization.

DATA (FEATURE AND TARGET)

The data contains one million observations of credit card transactions, with 3 numerical and 4 categorical features, including:

- distance_from_home
- distance_from_last_transaction
- ratio_to_median_purchase
- repeat_retailer
- used_chip
- used_pin_number
- online_order

'Fraud" was the selected target, or the predictor variable.

ALGORITHMS

Feature Engineering

The dataset was remarkably clean to begin with, despite containing one million observations. Minimal feature engineering was required.

Modeling

k-Nearest Neighbors, Logistic Regression, Decision Tree, Random Forest, XGBoost classifiers were used to determine baseline performance, before selecting the best performing model and interpretability.

Model Evaluation and Selection

- The data set was split into 70/30 train, test (holdout). All learning models were trained on training data with 10-fold cross-validation.
- Model performance was evaluated on a variety scoring metrics, including recall, precision, F1, and ROC-AUC as well the analyses of confusion matrices.
- Decision Tree was considered to be the ideal model with high predictive performance and interpretability.
- Additional tuning and optimal hyperparameters were identified using GridSearchCV.

Scoring Evaluation (Test or Holdout:

Accuracy:	1.00
Precision:	1.00
Recall:	1.00
F1:	1.00
ROC-AUC:	1.00

TOOLS

- Python, Pandas, Numpy for data cleaning, transformation, and feature engineering
- Scikit-Learn for classification modeling, scoring, and evaluation
- Matplotlib and Seaborn for data visualization and plotting

COMMUNICATION

In addition to the slides and visuals presented, all work is available on my Github found here.