Title: Credit Card Fraud Detection Project Documentation

1. Problem Statement:

The objective of this project is to develop a credit card fraud detection system that can identify and flag potentially fraudulent credit card transactions in real-time. Credit card fraud is a prevalent issue, and it is crucial to have a robust system in place to protect both the financial institutions and their customers from unauthorized transactions.

2. Design Thinking Process:

The project followed the design thinking process, which involves empathizing with the users (fraud analysts and customers), defining the problem, ideating solutions, prototyping the system, and testing it iteratively.

3. Phases of Development:

a. Data Collection:

- The project started by collecting historical credit card transaction data, which included both genuine and fraudulent transactions.

b. Data Preprocessing:

- Data Cleaning: Removed duplicates, missing values, and irrelevant features.
- Feature Engineering: Created new features to improve model performance.
- Data Split: Divided the dataset into training, validation, and testing sets.

c. Model Selection:

- Chose machine learning algorithms suitable for anomaly detection, such as Isolation Forest, One-Class SVM, and Autoencoders.

d. Model Training:

- Trained the selected models on the training data while tuning hyperparameters.
- Used cross-validation to prevent overfitting and validate model performance.

e. Model Evaluation:

- Evaluated models using metrics such as Precision, Recall, F1-Score, ROC-AUC, and confusion matrix.
- Conducted a cost-benefit analysis to assess the impact of false positives and false negatives.

f. Deployment:

- Deployed the best-performing model to a real-time credit card fraud detection system.
- Integrated the system with the bank's transaction processing system.

g. Monitoring and Maintenance:

- Set up continuous monitoring to detect model degradation.
- Implemented an update mechanism to retrain the model with new data.

4. Dataset Description:

- The dataset used in this project contains credit card transaction records with a label indicating whether each transaction is fraudulent or genuine.
- Features include transaction amount, timestamp, and various anonymized features for transaction details.

5. Data Preprocessing:

- Removed duplicates and missing values.
- Scaled and standardized numerical features.
- Created additional features like transaction hour and day of the week.

- Addressed class imbalance through oversampling (SMOTE) and undersampling techniques.

6. **Model Training Process:**
    - Selected Isolation Forest as the final model due to its effectiveness in anomaly detection.
    - Tuned hyperparameters like the contamination rate and the number of estimators.
    - Utilized cross-validation for model selection and performance assessment.
7. **Choice of Machine Learning Algorithm and Evaluation Metrics:**
    - Chose Isolation Forest due to its ability to identify anomalies effectively.
    - Selected evaluation metrics:
        - Precision, Recall, and F1-Score to assess the model's ability to detect fraud while minimizing false positives.
        - ROC-AUC to measure overall model performance.
        - Confusion matrix to visualize true positives, true negatives, false positives, and false negatives.
8. **Conclusion:**

    The credit card fraud detection project successfully developed a system to identify potentially fraudulent transactions in real-time, protecting both the financial institution and its customers. It employed Isolation Forest as the primary machine learning algorithm, with a focus on optimizing precision and recall to minimize false positives and negatives.

```
# Import necessary libraries

import pandas as pd

import numpy as np

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler

from sklearn.ensemble import IsolationForest

from sklearn.metrics import precision_score, recall_score, f1_score, roc_auc_score,
confusion_matrix

from imblearn.over_sampling import SMOTE


# Load the credit card transaction dataset

data = pd.read_csv("credit_card_data.csv")  # Replace with the actual dataset file


# Problem Statement

# Define the problem statement

print("Problem Statement: Detecting credit card fraud transactions.")


# Design Thinking Process
# No code required for this section. It's a description of the design thinking process.

# Phases of Development
# Data Preprocessing
# Remove duplicates and missing values
data.drop_duplicates(inplace=True)
data.dropna(inplace=True)

# Feature Engineering
data['transaction_hour'] = pd.to_datetime(data['timestamp']).dt.hour
data['day_of_week'] = pd.to_datetime(data['timestamp']).dt.dayofweek

# Split the dataset into features and labels
X = data.drop(columns=['fraudulent'])
y = data['fraudulent']

# Split data into training, validation, and testing sets
X_train, X_temp, y_train, y_temp = train_test_split(X, y, test_size=0.3, random_state=42)
X_val, X_test, y_val, y_test = train_test_split(X_temp, y_temp, test_size=0.5, random_state=42)

# Model Selection
# Initialize the Isolation Forest model
model = IsolationForest(contamination=0.01, random_state=42)

# Model Training
model.fit(X_train)

# Model Evaluation
# Make predictions on the validation set
y_val_pred = model.predict(X_val)
```

```
# Convert model predictions to binary (0: Genuine, 1: Fraudulent)
y_val_pred[y_val_pred == 1] = 0
y_val_pred[y_val_pred == -1] = 1

# Evaluation Metrics
precision = precision_score(y_val, y_val_pred)
recall = recall_score(y_val, y_val_pred)
f1 = f1_score(y_val, y_val_pred)
roc_auc = roc_auc_score(y_val, y_val_pred)
conf_matrix = confusion_matrix(y_val, y_val_pred)

print("Evaluation Metrics:")
print(f"Precision: {precision:.2f}")
print(f"Recall: {recall:.2f}")
print(f"F1 Score: {f1:.2f}")
print(f"ROC-AUC Score: {roc_auc:.2f}")
print("Confusion Matrix:")
print(conf_matrix)

# Choice of Machine Learning Algorithm and Evaluation Metrics
# The Isolation Forest algorithm was chosen for its effectiveness in anomaly detection.
# Evaluation metrics include Precision, Recall, F1 Score, ROC-AUC, and a Confusion Matrix.
```