# DSC 478 Project Proposal

## Project Type (Data Analysis)

## Team Member: Jonesh Shrestha

## Overview

This project aims to perform a **comprehensive analysis of US healthcare insurance claims data to detect patterns of Fraud, Waste, and Abuse (FWA) among providers**.
Key objectives include:

1. Exploring data characteristics to identify FWA indicators, such as unusual claim volumes or diagnosis patterns.
2. Applying unsupervised methods to uncover anomalies.
3. Building and optimizing supervised predictive models to classify fraudulent providers.
4. Drawing actionable insights for insurance auditing.

Methods and approaches will follow the Knowledge Discovery in Databases (KDD) process using Python tools: Preprocessing with Pandas/NumPy (merging, imputation, feature engineering like claim aggregates and age derivation); EDA with Matplotlib/Seaborn (statistics, visualizations); Unsupervised learning via Scikit-learn (e.g., KMeans clustering); Supervised learning (e.g., Random Forest and XGBoost classifiers with hyperparameter tuning via GridSearchCV and evaluation using cross-validation). Data-driven choices will be justified, e.g., selecting features based on mutual information scores and addressing class imbalance with SMOTE or SMOTE-NC.

Expected outcomes: Identification of key FWA predictors (e.g., high reimbursements linked to chronic conditions) and model performance metrics (e.g., >80% ROC-AUC). Deliverables: Executive summary (1-2 pages), main report (up to 5 pages) detailing KDD application and findings, Jupyter notebooks (HTML/IPYNB) with code, and a 5-minute demo video showcasing analysis runs and insights.

## Data Set(s)

I will use the "HEALTHCARE PROVIDER FRAUD DETECTION ANALYSIS" dataset from Kaggle. This is a synthetic dataset inspired by real US Medicare claims, consisting of relational, temporal, and transactional data across four main CSV files:

- **Beneficiarydata**: This data contains beneficiary KYC details.
  138,556 rows, 25 columns (e.g., demographics like DOB/Gender/Race, 11 chronic condition flags, annual reimbursement amounts).
- **Inpatientdata**: This data provides insights about the claims filed for those patients who are admitted to the hospitals.

40,474 rows, 30 columns (e.g., claim dates, 10 diagnosis codes, 6 procedure codes, physician IDs, deductible/reimbursement amounts).

- **Outpatientdata**: This data provides details about the claims filed for those patients who visit hospitals and are not admitted to them.
  517,737 rows, 27 columns (similar to inpatient, with outpatient-specific details).
- **Train**: This data contains ProviderID and PotentialFraud label.
  5,410 rows, 2 columns (Provider ID, PotentialFraud binary label).

Total ~700K rows; features are mixed (numerical, categorical, binary). The dataset's complexity (high-cardinality codes and merges on IDs such as BeneID, ClaimID, and Provider) supports broad ML tasks, including supervised classification on fraud labels and unsupervised anomaly detection on claims.

## Tasks and Methods

- **Data Preparation and Pre-processing**: Combine inpatient and outpatient claim files, then merge with beneficiary and provider label data using BeneID and Provider keys. Handle missing values through imputation for categorical codes (e.g., replacing null ICD/Procedure codes with 'UNKNOWN') and fillna for numeric fields like reimbursement amounts. Encode categorical variables using target or frequency encoding (due to high-cardinality diagnosis/procedure codes). Normalize continuous numerical features using StandardScaler. Perform feature engineering such as computing claim duration, average reimbursement per provider, and mortality indicators (Is_Dead from DOD). Ensure data integrity through quality checks (e.g., duplicate removal and schema validation).

- **Exploratory Data Analysis**: Compute descriptive statistics (df.describe() for reimbursement amounts) and visualize patterns using Seaborn (e.g., boxplots for reimbursements by fraud label, heatmaps for correlations among chronic conditions). Identify class imbalance and regional patterns (e.g., fraud rates by state or provider type).

- **Unsupervised Learning**: Perform clustering with KMeans (n_clusters=5–10) on aggregated provider-level features (e.g., total claim count, average claim amount, and reimbursement ratios). Apply Isolation Forest for anomaly detection. Evaluate clusters using silhouette scores and PCA-based visualizations; compare results with alternative algorithms (e.g., DBSCAN) and hyperparameter variations.

- **Supervised Learning**: Frame the task as a binary classification problem for fraud detection. Before model training, perform feature selection and extraction (e.g., RFE, mutual information) to identify the most informative variables. Use Logistic Regression as a baseline and train ensemble models such as Random Forest and XGBoost. Optimize key hyperparameters (e.g., n_estimators, max_depth) via GridSearchCV and evaluate

using Stratified K-Fold cross-validation. Report model performance with F1-score, ROC-AUC, and confusion matrix, addressing class imbalance appropriately.

## Timeline Work Distribution

The project will be completed over 4 weeks as an individual effort, culminating in submission by the deadline:

- **Week 1 (Data Prep)**: Download, merge, and clean all four CSV files; perform initial feature engineering (claim aggregates, age derivation); conduct data quality validation.
- **Week 2 (EDA & Unsupervised Learning)**: Conduct exploratory data analysis with visualizations; perform clustering (KMeans, DBSCAN) and anomaly detection (Isolation Forest); evaluate using silhouette scores and PCA visualizations.
- **Week 3 (Supervised Learning & Optimization)**: Implement feature selection; train baseline and ensemble models (Logistic Regression, Random Forest, XGBoost); optimize hyperparameters via GridSearchCV; perform cross-validation and comparative evaluation.
- **Week 4 (Report & Demo)**: Compile analysis findings; write executive summary and main report; finalize Jupyter notebooks with documentation; create 5-minute demo video with voice commentary.