**Overview:**

This report is on our Empowering Diabetes Management project during this phase, our focus was on developing a classification model.

**Data Preprocessing:**

We conducted an Exploratory Data Analysis (EDA) on the diabetes\_012\_health\_indicators\_BRFSS2015 dataset, revealing that it contained only numeric values with no missing data. Therefore, no further preprocessing was necessary.

**Classification Model Development:**

The initial step was to split the dataset into training, validation, and test sets, utilizing the 'train\_test\_split' module from the sklearn library. The dataset was partitioned into 80% for training, 10% for validation, and 10% for testing.

**Stratified** **Sampling**:

To ensure an even distribution of labels in each set, we employed the 'stratify' attribute. This method preserved the label distribution, making the training set more representative of the original data.

**Model Development:**

We addressed a significant class imbalance issue in our dataset by employing multiple classification algorithms, including Random Forest, XGBoost, and LightGBM. Each approach involved a process of hyperparameter tuning, feature importance analysis, and model performance evaluation on the validation set.

**Random Forest Classifier:**

* Grid search cross-validation in 5 folds to identify optimal hyperparameters.
* Best parameters resulted in weighted ROC AUC score.
* Feature importance analysis indicated strengths and weaknesses.
* Model performance on the validation set was better for non-diabetic cases.

**XGBoost Classifier:**

* Similar to Random Forest, we conducted a grid search for optimal parameters.
* Feature importance analysis provided valuable insights.
* Model performance showed limitations in handling the three-class classification problem.

**LightGBM:**

* Due to class imbalance, we explored boosting methods, transitioning from XGBoost to LightGBM.
* Grid search for best weighted ROC AUC score.
* Custom weight experimentation to address class imbalance.
* Model performance analysis revealed limitations similar to XGBoost.

**Resampling and Random Forest:**

* Given the previous results, we revisited the same methods with resampling techniques, such as under-sampling class 0 and over-sampling class 1.
* The resampled training data led to improved results with a weighted ROC AUC score of 0.89.
* However, the recall score for class 0 remained a concern.

**Resampling and XGBoost:**

* We assessed the performance of XGBoost on the resampled training data.
* The results showed that Random Forest outperformed XGBoost in this context.

**Fitbit:**

* The Fitbit API allows developers to access health and fitness data such as movement, sleep, heart rate from Fitbit devices. Access to this data is granted by creating a registered application through a Fitbit developer account, following OAuth 2.0 authorization. The API provides specific endpoints for different types of data, and the responses are in JSON format.

**Next Phase steps:**

* To achieve better classifier performance, we plan to explore hyperparameter tuning and other methods like support vector machines or neural networks in future work.
* We plan to implement SMOTE to address data imbalance and will compare its performance with previous resampling methods. This comparative analysis aims to assess the effectiveness of SMOTE in improving classifier performance.
* We will explore automatic data collection through a wearable device (Fitbit)