COE379L - Project 01 Report

Methodology for Data Preparation

To prepare the data, I read the breast cancer csv file into a pandas dataframe to visualize the content I was working with inside my Jupyter notebook. Afterwards, I examined the data frames' shape, size, and data types to ensure a comprehensive understanding of the data structure. I wanted to read the column descriptions along with the appropriate data types to infer the logical inputs that should be present. For each column I called the 'unique' function to see all possible values an entry in this column might take, and found unusable inputs for some columns (i.e. NaNs, ?, and * symbols). Duplicate rows were identified and removed to maintain data integrity. Then, missing and invalid values were treated using mode-based univariate imputation, as most of the data was categorical. Data visualization was then used via matplotlib and seaborn libraries. After analysis of data, multivariate imputation was considered, but it was difficult to determine possible correlations between different variables due to ignorance to the subject matter. Categorical variables were then transformed through one-hot encoding to make them suitable to develop machine learning models for K-Nearest Neighbor, K-Nearest Neighbor using Grid search CV, and linear classification. Lastly, the dataset used in the models was split into 70% training and 30% test sets while ensuring reproducibility and maintaining the proportion of each class in the dependent variable by stratifying.

Insights from Preprocessing

There were numerous insights gained from data preparation. For instance, univariate analysis visualizations provided an understanding of the distribution of each feature, helping to identify potential outliers or skewed distributions. Particularly, it was noted that the tumor size distribution was skewed right. Additionally, the analysis revealed a class imbalance for the breast cancer recurrence group, which was critical for understanding potential bias in the model. There were nearly twice as many people who did not exhibit recurrence events compared to people who did exhibit recurrence events. Because of this, the models that were trained might have been more biased in predicting no recurrence, negatively affecting their own recallability performance.

Procedures used to train the Models

The model training procedure involved selecting three classification algorithms: K-Nearest Neighbor (KNN), KNN with Grid Search CV for hyperparameter tuning of *k*, and Linear Classification. Each model was fit on the same training data. For the model involving cross validation, scoring was set to 'recall' and the parameter

grid would take on values from 1 to 20. A standard of 5 folds was also used. The model chose a value of the hyperparameter *k* that optimizes recall since it would be better to improve our detection of false negatives–in a realistic healthcare setting, improving detections of false negatives will remediate issues that undiscovered/underlying health issues may cause to an individual.

Model Performance and ability to predict Breast Cancer Recurrence

The performance of the models was evaluated using multiple metrics: accuracy, recall, precision, F1-score. Also, the optimal hyperparameter value $k = 1$ for the KNN with Grid Search CV model.

1) Accuracy: KNN with 3 neighbors achieved 60% on test data, while KNN with 1 neighbor and SGD showed 58% and 41% respectively. These moderate to low scores indicate limited overall predictive power.
2) Recall: SGD classifier demonstrated the highest recall of 75% on test data, outperforming even the KNN model utilizing cross validation. The train accuracy for KNN with CV was 99%, yet its test accuracy was 36%, making it indicative of overfitting issues.
3) Precision: All models showed low precision on test data (32-35%), indicating a high rate of false positives.
4) F1-score: Low for all models on test data (33%-45%), with SGD achieving the highest.

Overall, the best performing model was the SGD classifier, as despite its poor accuracy, it had the best recall rate on the test data.

Confidence in Model

Given the performance metrics of the three models, and the importance of the recall metric in relation to our dataset, the only model I would have slight confidence in is the SGD classifier. Further work is needed to improve performance for the KNN models. Recall could have been improved for KNN models by changing the decision threshold from its default value of 0.5 to a value like 0.3.