UNIVERSITI TUNKU ABDUL RAHMAN

Faculty of Information and Communication Technology

Bachelor of Computer Science

UCCD 2063 Artificial Intelligence Techniques

Assignment 1

Group 39

Student Name	Wong Dong Hao	Yong Yi Kang	Siah Wei Hao
Student ID	2300937	2301506	2301895
Contribution	35%	35%	30%
Signature	Hao	Yong	Siah

Chapter 1: Introduction

Background and Objectives

As society and technologies advanced rapidly through these years, the lifestyle of the public has changed significantly and became unhealthier. People with unhealthy lifestyles and bad habits such as unhealthy diets, lack of exercise, and smoking, face a higher risk of having cardiovascular diseases.

But on the positive side, the growing field of machine learning has enabled us to predict a wide range of outcomes, including health risks. Data scientists collected data from all over the world to train models and evaluate the best models to predict specific disease, including cardiovascular. With the trained model, people can now have a straightforward number of how high or low risk that they will have cardiovascular disease.

So, the objective of this this assignment focuses on predicting cardiovascular risk of one, according to the dataset provided. The dataset contains a total of 17 features such as demographic, lifestyle, health-related attributes and the risk of cardiovascular of the person. In this assignment, we will utilize and train machine learning models to predict risk of cardiovascular and enhance the prediction accuracy. By undergo data exploration, data preprocessing, model selection, model training and validation, model tuning and testing. We aim to improve prediction accuracy and gain insights into the factors contributing to cardiovascular disease risk.

Chapter 2 : Method

Description of the Dataset

The dataset consists of 18 features and 2100 records. The features are categorized as follows:

- **Categorical Features:** Gender, family history, alcohol consumption, junk food consumption, snack frequency, smoking status, transportation method, TV watching, discipline, cardiovascular risk.
- **Numerical Features**: Age, height (height), weight, vegetable intake per day, meal frequency per day, exercise frequency, water intake, income.

The dataset includes:

- Three features with floating-point values.
- Five features with integer values.
- Ten features with object (string) types.

The dataset is clean and well-structured, with no missing values, so no additional data cleaning is required.

Data Exploration and Visualization

Numerical Data Analysis

1. **Age:**
   - The distribution of ages shows a concentration of younger individuals. This suggests that the dataset may have a younger demographic, which could impact the analysis of cardiovascular risk.
2. **Other Numerical Features:**
   - Other numerical features (e.g., height, weight, exercise frequency) exhibit a more normal distribution, with values spread across their respective ranges.

Categorical Data Analysis

1. **Cardiovascular Risk:**
   - The distribution of cardiovascular risk categories shows a high count for the "high" risk category compared to "low" and "medium" categories. This indicates an imbalance in the risk levels represented in the dataset.
2. **Gender:**
   - The gender distribution appears balanced, with a roughly equal number of male and female entries.
3. **Family History:**
   - There is a significantly higher count of individuals with a family history of cardiovascular issues ("yes") compared to those without ("no"). This suggests that family history is a prevalent factor in this dataset.
4. **Alcohol Consumption:**
   - The distribution of alcohol consumption is imbalanced, with "low" consumption being much higher compared to other levels (“none’, "moderate" and "high").

5. **Junk Food Consumption:**

    - The "yes" category for junk food consumption is notably higher than the "no" category, indicating a greater prevalence of junk food consumption among the individuals in the dataset.

6. **Snack Consumption:**

    - The frequency of snack consumption shows a much higher count for "sometimes" compared to “always”, "rarely" and "never," suggesting that snack consumption is a common behaviour.

7. **Smoking:**

    - The dataset shows a higher count of smokers ("yes") compared to non-smokers ("no"), highlighting a significant prevalence of smoking in the sample.

8. **Transportation:**

    - Public transportation, particularly taking the bus, is the most common mode of transport among individuals in the dataset.

9. **TV Watching:**

    - The "often" category for TV watching is very low compared to "rarely" and "moderate," indicating that frequent TV watching is less common.

10. **Discipline:**

    - The distribution shows a higher count of individuals who are "no" for discipline-related questions compared to those who are "yes." This suggests that lack of discipline is more prevalent in the dataset.

Data Pre-processing

1. **Splitting the Dataset:**
   - The dataset is divided into training and test sets to facilitate model evaluation. The split is typically 70% for training and 30% for testing.
   - For reproducibility, the random generator is seeded with random state = 42, ensuring consistent splits across different runs.
2. **Feature and Target Assignment:**
   - The feature set (x) consists of all columns except the target variable.
   - The target variable (y) is the Cardiovascular risk feature.
3. **Numerical and Categorical Preprocessing:**
   - **Numerical Features:** Standardization is applied, scaling features to have a mean of 0 and a standard deviation of 1.
   - **Categorical Features:** One-hot encoding is used to convert categorical data into a format suitable for machine learning models.
4. **Combining Data:**
   - After preprocessing, the numerical and categorical features are combined into a single dataset.
   - The target variable y_train is converted to a numpy array format to facilitate model training.

This approach ensures that the data is properly formatted and standardized for subsequent modelling tasks.

Model Selection

In this assignment, according to the dataset cardiovascular risk is categorized into three class: low, medium and high. So, predicting cardiovascular risk in this assignment involves multiclass classification. Model selection is crucial since we aim to evaluate the most suitable model to predict the cardiovascular risk without overfitting or underfitting the dataset. Due to this reason, we considered several machine learning models with unique characteristics that can handle different types of dataset and objectives.

1 **SGD Classifier (Stochastic Gradient Descent Classifier)**: SGD Classifier is a type of optimization algorithm that updates the parameters of the model based on the gradient of the loss function computed concerning a small subset of the training data, rather than the entire dataset at once.

Reason

- SGD Classifier is highly efficient for large datasets in machine learning, which allows it to train faster than other models such as Logistic regression.
- SGD Classifier have several setting and tuning parameters available for making it a more flexible model for various situations. For example, we can set and tune the learning rate and regularization to prevent overfitting while training the model.

2 **Logistic Regression:** Logistic Regression is a widely used linear model for binary classification problems. It can also use to handle multiclass classification through selecting different solver.

Reason

- Logistic Regression is one of the most popular and simple models compare to others. A simpler model will have a lower chance for overfitting to the datasets compare to a complex model.
- By selecting different solver of the model that can handle multiclass classification such as lbfgs, newton-cg and liblinear that support one-vs-rest, logistic regression can also be a suitable model for predicting the cardiovascular risk in this assignment.

3 **Random Forest Classifier:** Random Forest is an ensemble learning method that builds multiple decision trees and combines their predictions to improve accuracy and robustness. Each tree in the forest is trained on a random subset of the training data and a random subset of features, which helps in reducing overfitting and variance.

Reason

- This model can handle complex and non-linear features in the datasets, this makes it suitable for this assignment since we have 9 categorical data in the dataset.
- Besides, random forest classifier is known for handling imbalanced datasets. Since the distribution of each feature in our dataset is quite imbalanced. So we are thinking about this model can predict well on this dataset.
- Random forest is also a easy tuning model, by setting the n estimators, max depth, max leaf and others. All the hyperparameters in random forest are easy to understand as the name tells all.

Summary of model selection:

- **SGD Classifier** is efficient and flexible, particularly suited for large datasets.
- **Logistic Regression** offers a simple, reliable, and interpretable model, ideal for situations where explainability is important.
- **Random Forest Classifier** can handle complex, non-linear relationships and imbalanced datasets, while providing insights into feature importance.

Model Training and Validation

In this section, we performed model training and validation using the three models and predict the result. The result is then to use for calculating performance metrics. Cross validation is used on train set for a less overfitting predict result.

1. **Model Fitting on Train set:**
    - Each model is trained using the train set

2. **Prediction using Train and Test sets:**
    - After training the model, it is used to predict both the train and test sets for performance metrics calculations.

3. **Apply 5-fold cross validation and shuffle on Train Set:**
    - Cross validation is applied on training set to reduce overfitting of the model, shuffle is applied during cross validation to randomly reordering the data.

4. **Calculate Performance Metrics for Train Set:**
    - The performance of models is measured using metrics such as accuracy, precision, recall, F1-score and AUC(Area Under the Curve).
    - For the training set, cross-validation-based metrics (like precision, recall, F1-score) are used, which give a better estimate than simple training accuracy.

Model Tuning and Testing

In this section, we performed model tuning using GridSearchCV with train set to identify the best hyperparameters for each model to enhance predictive accuracy and mitigate overfitting. Lastly, each model with the best hyperparameters will be use in final testing and predict the test set.

Hyperparameter Tuning and validating with GridSearch

1. **SGD Classifier**
    - **Loss Functions**
        1. **Squared error:** Useful for regression tasks, but less commonly used for classification.
        2. **Log loss:** Standard for classification tasks.
        3. **Modified huber:** A robust loss function for classification that blends hinge loss and squared loss.
    - **Eta0 :** Learning rate. Lower values ensure gradual learning; higher values make learning faster but can lead to overshooting**.**
    - **Alpha:** Controls the regularization strength. Small values (0.01) allow less regularization, while higher values (1) increase it.
    - **Max iteration :** Number of iterations before convergence. Values around 2000-4000 are good for large datasets.
    - **Tol :** Tolerance for the stopping criterion. Lower values make the optimization stricter.
    - **Penalty :** L1, L2, and elasticnet provide regularization, with elasticnet combining both L1 and L2.

2. **Logistic Regression**
   - **Solver**
     1. **liblinear:** Good for small datasets and L1 regularization.
     2. **lbfgs** and **newton-cg:** Suitable for larger datasets and support multiclass classification.
   - **C:** Inverse regularization strength. Smaller values make the model more regularized (simpler), and larger values allow the model to fit the data more closely.
   - **Max Iteration**

3. **Random Forest Classifier**
   - **N Estimators:** Number of decision trees in the forest. More trees generally improve accuracy but increase computation time.
   - **Max depth:** Limits the depth of each tree. Controlling this helps to reduce overfitting.
   - **Min sample split** and **Min sample leaf:** Control how decision trees split, impacting overfitting.

Summary of model tuning

After performing the grid search, the best hyperparameters for each model:

SGD Classifier: {**alpha:** 0.01, **class weight:** balanced, **eta0**: 0.01, **loss**: modified huber, **max iter:** 1000, **penalty:** L1**, tol:** 0.0001}

Logistic Regression: {**C:** 1.0, **class weight:** balanced, **max iter:** 1000, **solver**: newton-cg}

Random Forest Classifier: {**class weight:** balanced, **max depth**: 20, **min samples leaf:** 4, **Min_samples_split:**10, **n_estimators:**300}

Model Testing

After getting the best hyperparameters for each model, we evaluated their performance on the test set. We categorize the performance metrics as follows

Accuracy: The overall percentage of correct predictions.

Precision: The ability of the model to correctly identify positive samples.

Recall: The ability of the model to identify all relevant instances.

F1-Score: A weighted harmonic mean of precision and recall.

AUC (Area Under the Curve): The area under the ROC curve for multi-class classification.

Additionally, confusion matrices, PR graph and ROC graph were plotted for both train set and test set.

Chapter 3: Result and Discussion

Summary of Train set and Test set for each model

SGD Classifier	**Accuracy**	**Precision**	**Recall**	**F1 Score**	**AUC**
Train Set (Cross validation Prediction)	0.9830	0.9804	0.9806	0.9804	0.9897
Test Set (Standard Predict)	0.9730	0.9688	0.9674	0.9681	0.9811

Logistic Regression	**Accuracy**	**Precision**	**Recall**	**F1 Score**	**AUC**
Train Set (Cross validation Prediction)	0.9694	0.9646	0.9672	0.9655	0.9983
Test Set (Standard Predict)	0.9587	0.9514	0.9502	0.9505	0.9972

Random Forest Classifier	**Accuracy**	**Precision**	**Recall**	**F1 Score**	**AUC**
Train Set (Cross validation Prediction)	0.9401	0.9303	0.9340	0.9320	0.9908
Test Set (Standard Predict)	0.9397	0.9277	0.9310	0.9292	0.9900

In-depth analysis of performance and errors of the models

SGD Classifier

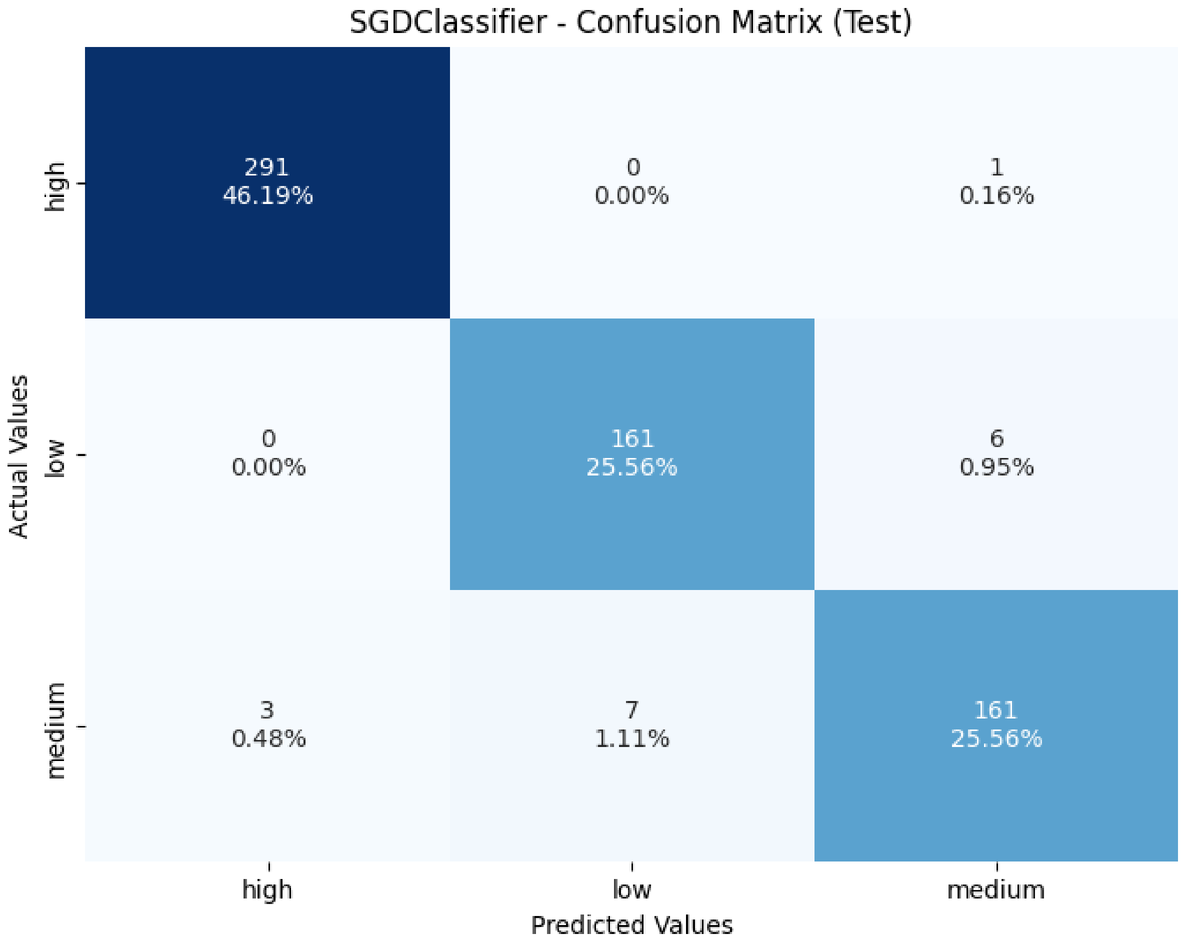
- **Performance metrics**

  The F1 score of SGD Classifier is the highest but also having the largest gap of AUC value on both train and test sets at the same time between the tree models, showing the potential of overfitting.

- **Error Analysis**

  i) Precision and recall values are very close to each other, suggesting balanced performance. The lower AUC on the test set compared to training set could be due to slightly more challenging test data or potential overfitting on the training set.

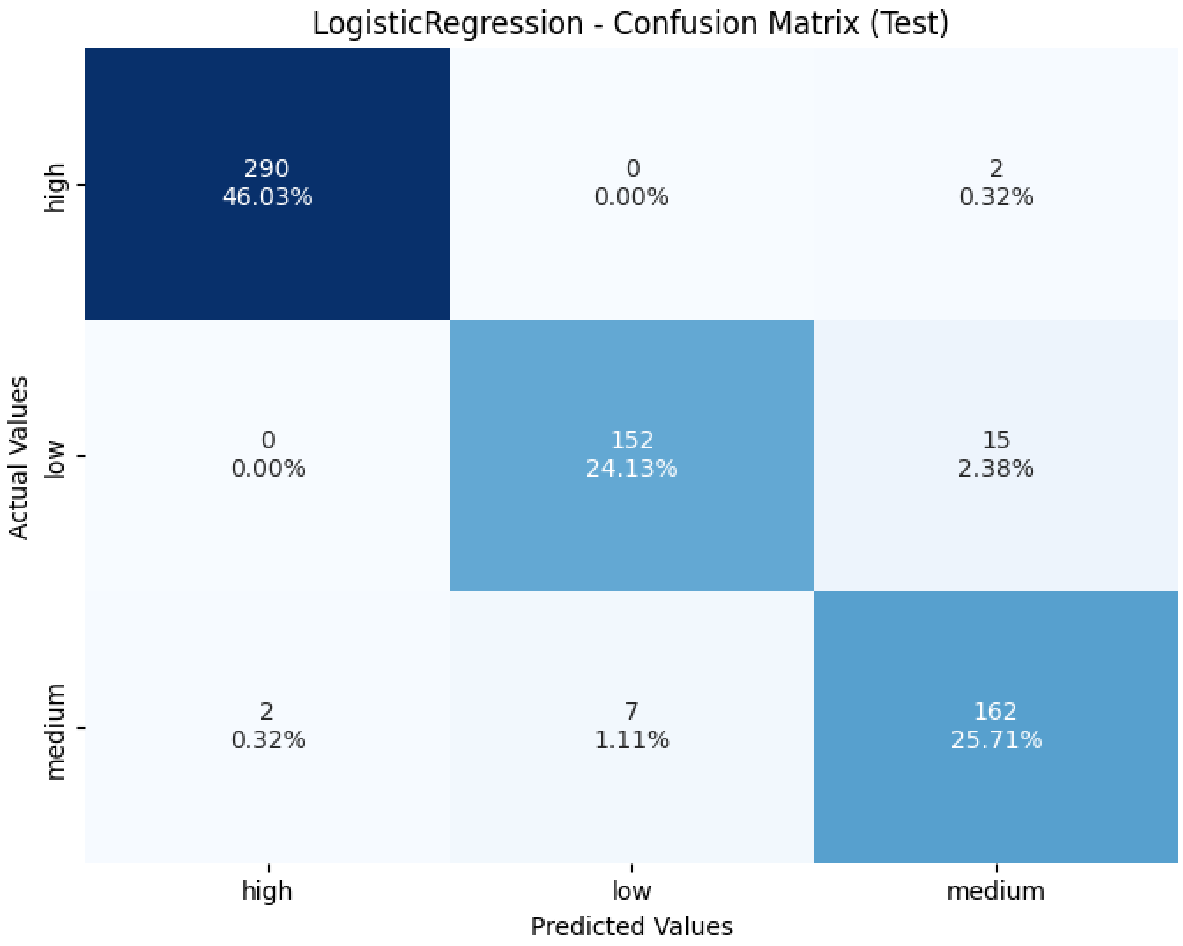
  ii) According to the confusion matrices that is plotted, both train set, and test set have more mistakes in predicting the low and medium risk of cardiovascular. One of the reasons maybe the unbalanced distribution of the cardiovascular risk in the dataset.



Logistic Regression

- **Performance metrics**
  Logistic regression also shows good performance on both train and test sets, with a balanced and small gap of F1 score and AUC value on both train and test sets. Showing a moderate performance of the model.

- **Error Analysis**
  The lower F1 score of Logistic Regression states that maybe the model performs poorly in an imbalanced dataset. As according to the confusion matrix of Logistic Regression, it predicted the medium risk wrongly way more than two other classes.



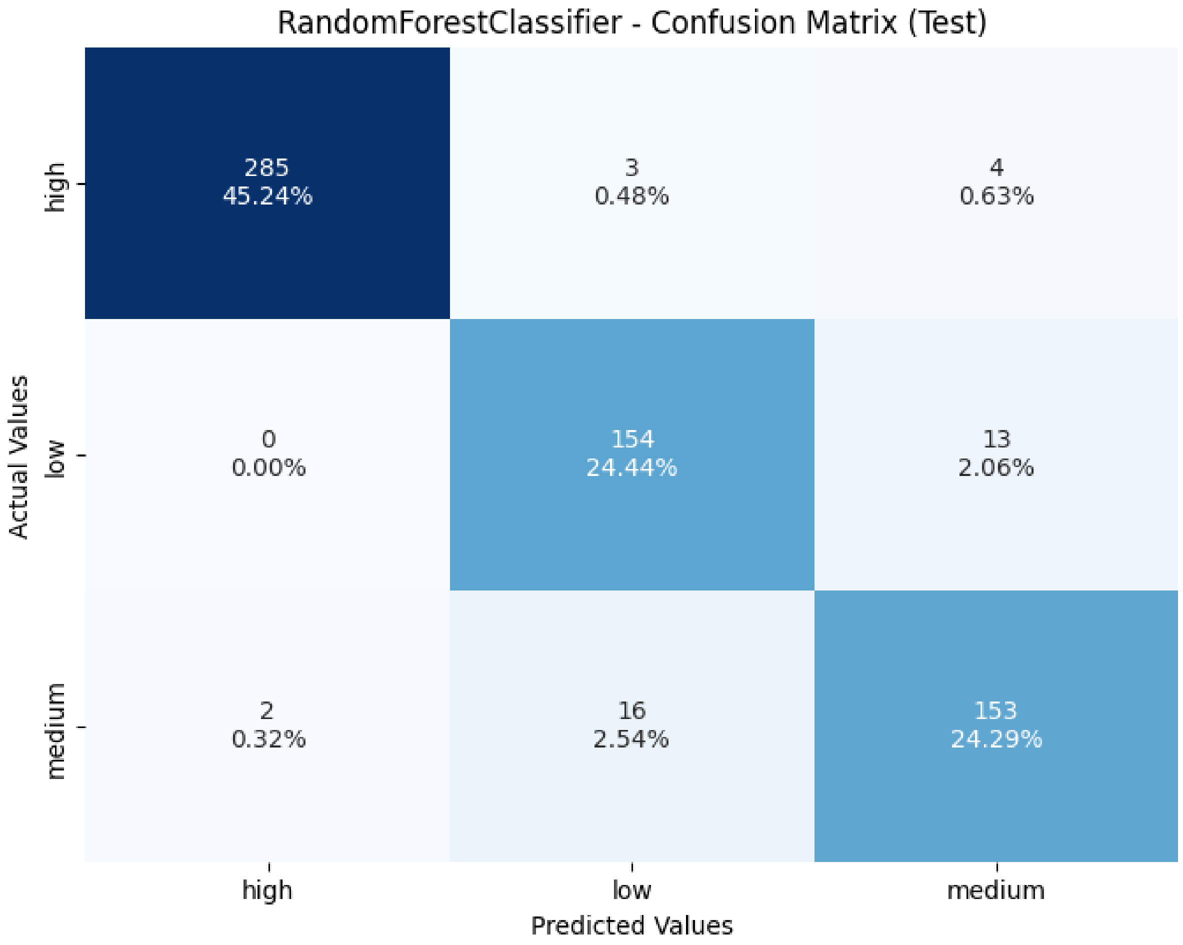
Random Forest Classifier

- **Performance metrics**
  1. The Random Forest Classifier performs comparably well on both train and test sets. It has a slightly lower accuracy and AUC compared to the other models but still shows strong performance.

  2. On the other hand, Random Forest Classifier have the smallest gap of F1 score and AUC value on both train and test sets, indicates that Random Forest Classifier maybe a more flexible and not overfitting model than other two models.

- **Error Analysis**
  1. Random Forest Classifier have the most error in prediction according to the confusion matrix, a higher error prediction in medium and low risk than two other models.

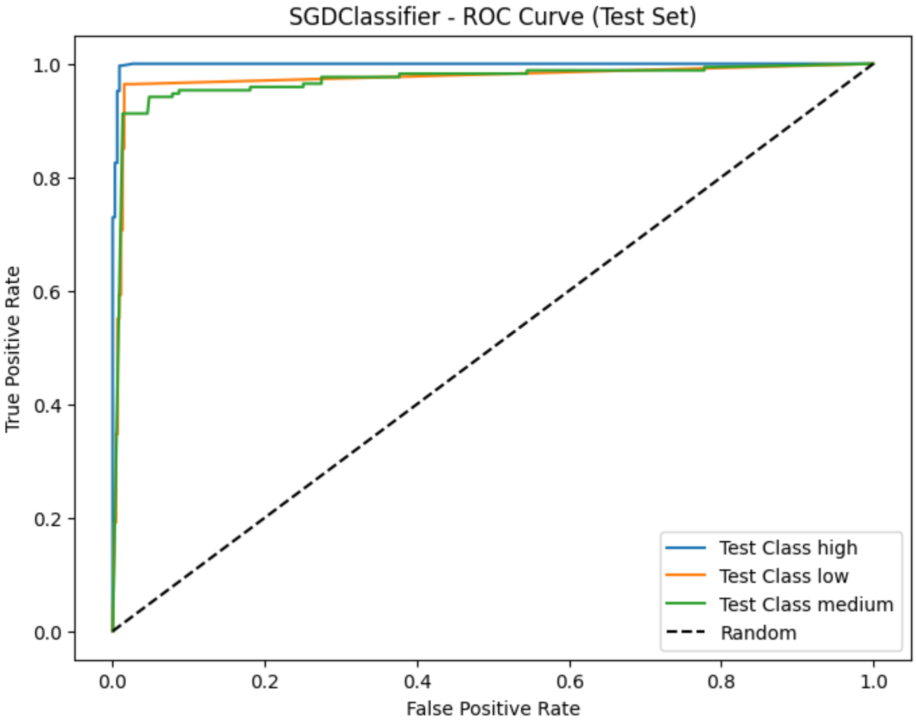
  2. Although Random Forest Classifier have the lowest performance metrics, but the small drop in AUC and F1 score is not significant and suggests good generalization with slight variations in the data.



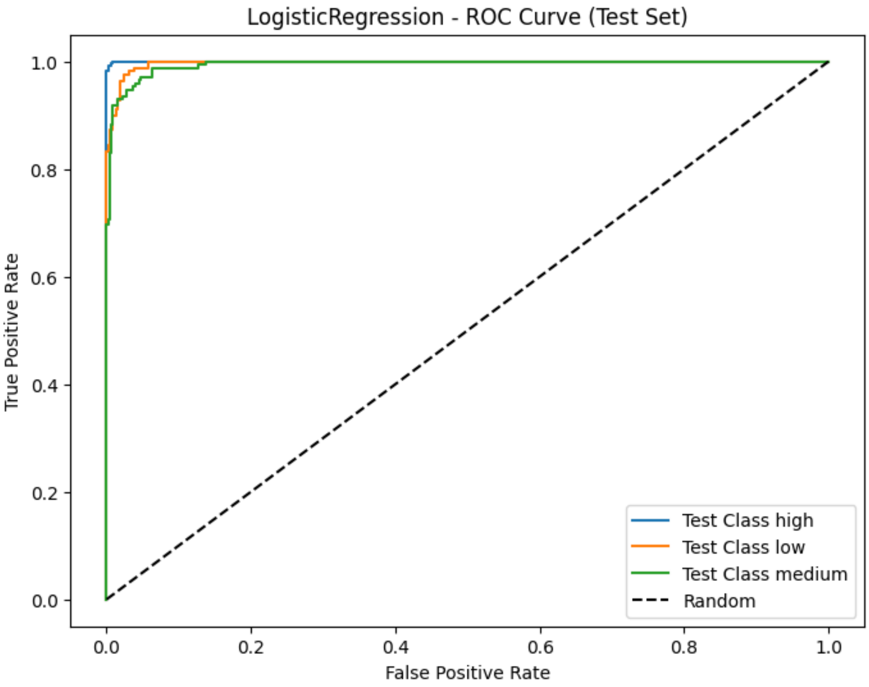
Comparing each model performance in table and figures

Performance Metrics	**F1 Score (Train)**	**F1 Score (Test)**	**AUC (Train)**	**AUC (Test)**
SGD Classifier	0.9804	0.9681	0.9897	0.9811
Logistic Regression	0.9655	0.9505	0.9983	0.9972
Random Forest Classifier	0.9320	0.9292	0.9908	0.9900

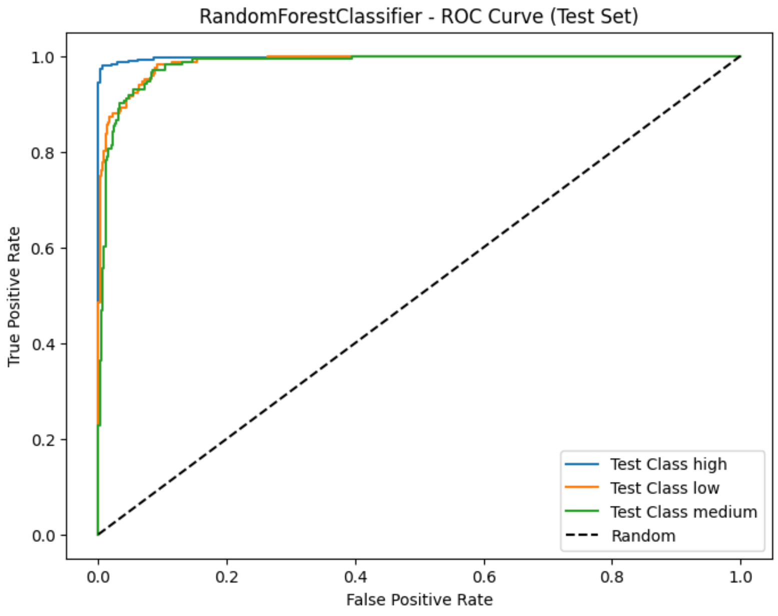
ROC Curves Comparison



From the ROC curves, we can see that although the curves are towards the top left corner, the lines are not smooth and presented to be a little zic-zac. This may be the model is overfitting, so the curves are going close to the noise, showing the curves are not smooth.



The Logistic Regression ROC curve performs better than the GSD Classifier by having a smoother curve and also towards the top left corner. Although the low and medium risk curve seems a little bit affected by the noise, overall is still a good performance.



For the Random Forest Classifier, it has the smoothest curve between three models ROC curve. Additionally, the low and medium risk curves show the less affected by noise compared to two other models. To make it a more flexible and maybe a more well-trained model.

Summary of comparison of performance metrices and ROC curves

Overall, as the comparison above, SGD Classifier has the largest gap of F1 score and AUC value between train set and test set. The ROC curve also showing it may be affected by noise, indicating it may be overfitting

On the other hand, Logistic Regression has the most moderate and balanced performance either on performance metrics or ROC curve.

Lastly, although Random Forest Classifier has the lowest performance among the three models, it has the smallest gap of F1 score and AUC value between train set and test set. Additionally, with a smoothest and least noise affected ROC curve. Indicating that it is also performing well and has a lower chance of overfitting.

Strength and Weakness of three models

Model	**Strength**	**Weakness**
SGD Classifier	• Fast and scalable for large datasets • Efficient for linear data	• Requires feature scaling • Sensitive to hyperparameters • Not suited for non-linear data
Logistic Regression	• Easy to interpret • Less sensitive to hyperparameters	• Assumes linear relationship • Struggles with large datasets
Random Forest Classifier	• Handles non-linearity well • Less prone to overfitting • Feature importance insights	• Computationally expensive • Hard to interpret • Can be biased towards majority class

Conclusion

- **Data explore and preprocessing**
  1. For the dataset provided, we look through it to check for distribution and any missing values in the dataset.
  2. After preprocessing, we split the dataset into train and test sets. Furthermore, categorize it into numerical and categorical features.
  3. We utilize standardization and one hot encoding to process the numerical and categorical data respectively.
  4. Lastly, we combine the processed data and check for the data shape

- **Model Training and validation**
  1. SGD Classifier, Logistic Regression and Random Forest Classifier is chosen.
  2. Each model is trained with the train set, and it is trained in k fold cross validation.
  3. Trained model is used to predict for validation.

- **Model tuning and testing**
  1. GridSearchCV is used in this step, we set some combinations of hyperparameters for each model to find the best hyperparameters.
  2. During grid search, model is tuning and validating using train set
  3. After that, each model with best hyperparameters is used to test and predict the test set

- **Important features while predicting**
- **Model performance comparison**
  1. After testing, we try to compare each model according to the performance metrics and graph plotted.
  2. We try to analyse the performance, errors, pros and cons, and characteristics of each model.

Findings

We find that **SGD Classifier** have the highest performance, but it may be overfitting according to the metrics calculated.

Logistic Regression emerged as the most balanced model, offering strong generalization, interpretability, and performance across various metrics. It provides a probabilistic output, which makes it a good choice for practical implementation.

Random Forest Classifier is highly robust and handles complex, non-linear data well, but at the cost of increased computational resources and reduced interpretability. It also take the longest time while training and validating.