UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY

UCCD 2063 Artificial Intelligence Techniques Group Assignment

June 2024

Machine Learning Model for Cardiovascular Risk

Student Name:	**Lee Zong Hao**	**Lim Phai Yuan**	**Ng Yi Jian**
Student ID:	**2206561**	**2206149**	**2207029**
Contribution	**40%**	**30%**	**30%**
Signature			

Contents

1.0 Introduction

1.1 Background

Cardiovascular diseases (CVD) are one of the most common causes of mortality globally. 17.9 million deaths are recorded annually for CVD. CVD consists of various types including coronary artery disease, strokes, and hypertension. With changes of lifestyle and environments, the occurrence of CVD continues to rise, especially among the low and middle-income country which covers over 3 quarters of CVD deaths. The key risk factors that significantly affect the cardiovascular risk are poor diet, physical inactivity, excessive alcohol consumption and smoking.

Preventative measures in clinical guidelines suggested regular screening for cardiovascular abnormality. However, the traditional method is not efficient and cost friendly. With the implementation of a data-driven prediction model, the detection of cardiovascular risk can be carried out faster in the meanwhile saving costs. Machine learning models can learn the patterns in all the factors and determine the cardiovascular risk.

This research aims to resolve the challenges of determining cardiovascular risk by utilizing machine learning techniques, with the goal of improvising early detection techniques and early prevention of disease progression.

1.2 Objectives

- **Predict Cardiovascular Risk**

  Develop a data-driven machine learning model that can accurately predict an individual’s cardiovascular risk.

- **Identify relationship between various risk factors**

  Explore the correlation between the risk factors of cardiovascular disease such as gender, age, height, weight, diets and lifestyle.

- **Implement multiple machine learning algorithms**

  Apply multiple machine learning algorithms to predict individual’s cardiovascular risk

- **Evaluate models performance**

  Fine-tune models and evaluate model performance using metrics such as accuracy, precision, recall, f1-score and confusion matrix to identify the best model for predicting cardiovascular risk.

2.0 Methods

2.1 Dataset Description

In the dataset (dataset.csv) used to feed the machine learning models contain 2100 records and 18 columns in which of 17 features and 1 target variable. The 17 features include demographic data, diet habits, lifestyle and income group and the target variable is cardiovascular risk (Low, Medium, High). The prediction task for this research will be classification.

2.2 Data exploration and visualization

Dataset is checked for missing value, imbalance and analyzed using various graphs such as pair plot, heatmap, box plot, scatter plot and more to further understand the relationship between features, identifying outliers and visualize the distribution.

2.3 Data pre-processing

Dataset is split into train set and test set with a 70:30 ratio. Since the dataset has inbalance in class, we resampled the training set using undersampling method to balance the dataset. The training set is split again into X and y where X contains all the features and y contains only the target variable. A pipeline is created to impute missing value and preprocess features in the training set. In the pipeline, missing value is filled with median for numerical data and mode for categorical data using Simple Imputer. Numerical data is then preprocessed using Standard Scaler while categorical data is preprocessed using One Hot Encoder before merging both data back into one.

2.4 Model Selection

Logistic Regression

A linear model which uses logistic functions to predict the class of a given input. Logistic functions is used to calculate the probability of an input belonging to a class. This model outputs a number between 0 and 1 which is then controlled by a threshold before classifying the input into their class.

Decision Tree Classifier

A non-linear algorithm which can be used in both regression and classification scenario. Decision tree continuously branching data based on the conditions which will end up like a tree structure. Each node in the tree represents a decision based on a feature. This model can cause overfitting easily

Random Forest Classifier

An ensemble learning model which merges multiple decision tree. By creating multiple sets of decision stress with different parameters from each split, the model will then use the majority voting method to make a final prediction. This model solves the problem of decision tree being too easy to be overfitting.

Support Vector Classification

SVC uses hyperplanes to classify inputs into respective classes. SVC finds the maximum margin of the hyperplanes between data points and the classes. This model can work on linear and non-linear data as it has functions that can project data into a higher dimension.

2.5 Model training and validation

Models are trained with the training set and evaluated using cross-fold validation in which it is set to 10-fold. The mean accuracy of each model is calculated and shown to make comparisons between the 4 models chosen. 3 out of 4 will be shortlisted for further tuning.

2.6 Model tuning and testing

Fine tuning is carried out by using GridsearchCV in which it will iterate all the given hyperparameter to find out the best combinations for each model. However, in this research, after the first fine tuning is carried out, feature selection is implemented for another round of fine tuning. The feature selection method used is Variance Threshold in which it calculates the correlation of each feature and removes the features that are under the threshold set to reduce noises in the datasets and retain the features that are important. After comparing both the result of fine-tuning with and without feature selection, the best model will be used to evaluate in the test sets. The metrics used for the evaluation include accuracy, precision, recall, f1-score and confusion matrix.

3.0 Result and Discussion

3.1 Summary of Training and Testing Results

Training Result

Model	Accuracy
Logistic Regression (Short-listed)	0.9679
Decision Tree (Short-listed)	0.9528
Random Forest	0.9392
Support Vector Classifier (Short-listed)	0.9519

Table 3.1.0 training Set Accuracy Result

Logistic Regression

Criteria		Default / Training result	Fine Tuning Best Parameter	
			With threshold	Without Threshold
Hyperparameter	Threshold		0.1	
	Penalty	12	l2	l2
	C	1	100	100
Accuracy		0.9679	0.9848	0.9823

Table 3.1.1 Logistic Regression performance during training and fine-tuning

Decision Tree

Criteria		Fine Tuning Best Parameter

		Default / Training result	With threshold	Without Threshold
Hyperparameter	Threshold		0.2	
	Criterion	gini	entropy	log_loss
	Max-depth	none	none	none
	Min_split	2	5	5
Accuracy		0.9528	0.9730	0.9688

Table 3.1.2 Decision Tree performance during training and fine-tuning

Support Vector Classifier

Criteria		Default / Training result	Fine Tuning Best Parameter	
			With threshold	Without Threshold
Hyperparameter	Threshold		0.2	
	Kernel	rbf	Linear	rbf
	C	1	100	100
	Gamma	Scale	Scale	0.01
Accuracy		0.9519	0.9865	0.9857

Table 3.1.3 Support Vector Classifier performance during training and fine-tuning

Evaluation Result

SVM with feature selection

Model	**SVM with feature selection**
Criteria	
Accuracy (%)	99.05
Recall (%)	99.05
Precision (%)	99.05
F1-score (%)	99.05

Table 3.1.4 Evaluation results on test sets

		Predicted		
Actual		**High**	**Medium**	**Low**
	High	292	0	0
	Medium	2	168	1
	Low	0	3	164

Table 3.1.5 Confusion Matrix on Performance evaluation of SVM with feature selection

3.2 In-depth Analysis of the Prediction Performance and Errors.

1. **Logistic Regression**: Based on table 3.1.1, the model has shown it has consistent cross-validation scores, which indicates it has a minimal variance across the folds. It also achieved a mean score of 96.79% during training, which is the highest among all models used. For performance with fine tuning, it shows 98.23% accuracy, 1.46% improvement. But after feature selection using Variance Threshold, it increases to 98.48% accuracy, which is another 0.25%

improvement. This has indicated that fine tuning and feature selection helps in increasing the performance of this model.

2. **Decision Tree**: Based on table 3.1.2, the model has achieved a mean score of 95.28% during training. For testing set after fine tuning, it shows 96.88% accuracy, increases 1.6%. But after feature selection using Variance Threshold, it increases to 97.30% accuracy, which is an improvement of 0.43%. This has indicated that fine tuning and feature selection helps in increasing the performance of this model.

3. **Support Vector Classifier**: Based on table 3.1.3, the model has achived a mean score of 95.19% during training. For the result with fine tuning, it shows 98.57% accuracy, a 3.55% improvement. But after feature selection using Variance Threshold, it increases to 98.65% accuracy, which is the highest score achieved compared to all models. Fine-tuning and feature selection has greatly improved the performance of SVM.

4. **Evaluation using test set**: Based on table 3.1.4, the result of performance of SVM on test sets is evaluated with accuracy, recall, precision and f1-score. SVM has 99.05% across the matrix which means the models is consistent and very accurate. Table 3.1.5 shows the confusion matrix which also shows how this model performs in classifying the test set.

3.3 Performance Comparing between Models

Model	**Training accuracy score**	**Score after fine tuning**	**Score after fine tuning with Variance Threshold**
Logistic Regression	96.79	98.23	98.48
Decision tree	95.28	96.88	97.30
Support Vector Classifier	95.19	98.57	98.65

Table 3.3.1 Accuracy Performance on Models

Based on table 3.3.1 , we can see that during training set, Logistic Regression has the highest score but in test score while using GridSearchCV and GridSearchCV combined with threshold feature selection, SVM holds the highest score. From here we can see that with further fine tuning SVM has the most consistent improvement, this might be due to the fact that SVM uses regularization to avoid overfitting in general, and it is also good at generalizing unseen data if properly tuned. In comparison, Decision Tree model is the least accurate among the 3 model. This might be caused by lack of pruning since SVM and Logistic Regression already have built in regularization mechanisms for complex datasets. It might be also because of unwanted noise in the data. Decision Tree cannot handle noise as well as SVM and Logistic Regression.

3.4 Strengths and Weaknesses

1.Logistic Regression

Strength: It works well when classes are either in linear form or multiclass. It is also easily implemented. Its output is in probability form, hence making it easy to understand relationships among inputs and outputs.

Weakness: It is very sensitive to outliers. Meaning if it is not tuned properly, it may not function as well and underperform. This can be shown when variance threshold is not implemented, the test score isn’t improved as much.

2. Decision Tree

Strength: Easy to visualize and understand. A clear visual representation is provided hence users will interpret it easily. The tree like process further contributes to understanding the process and relationship better. It can also process both numerical and categorical data, making it more flexible than Logistic Regression.

Weakness: If not pruned properly, it might overfit the data easily. Can be seen as it has the lowest test score among all the methods used in all processes.

3. Support Vector Classifier

Strength: This method is best when there are multiple features in the dataset, making it suitable for complex datasets. Hence it is best suited to detect cardiovascular diseases in this project, as shown above it has the highest accuracy out of the 3 chosen models. It also performs well when there are high dimensional datasets that are needed to be handled.

Weakness: It is more computationally intensive in comparison to other models due to its ability to process complex datasets. It is also very heavily dependent on its parameters. Which makes tuning challenging. Poorly chosen parameters will lead to poor performance.

3.5 Feature Importance

Using the Variance Threshold method, a new pipeline is created specifically for feature selection. With a threshold of 0.01, the accuracy of the test scores has been greatly improved on, especially for Logistic Regression. By comparing with and without feature selection for test scores, Logistic Regression has improved 1.19%, Decision Tree has improved 0.l5% and SVM has improved 0.88%.

4.0 Conclusion

This project has successfully developed and demonstrated 3 machine learning models to predict cardiovascular disease. The 3 models are Logistic Regression, Decision Tree and Support Vector Machine. To improve upon the training accuracy, some countermeasures and tuning has been done including setting up hyperparameters and feature selection. These methods have proven to be effective as shown when the test scores are being improved upon these methods. Other than that, separating the sets into numerical and categorical, creating pipeline, encoding labels also contributed to this success.

Among these 3 models, Support Vector Machine has proven to be the most accurate out of the 3, after GridSearchCV and feature selection. The score is then followed by Logistic Regression and Decision Tree. There are several possible reasons for this case. For Decision Tree, it might be due to too much noise, leading to overfitting. As for Logistic Regression, it might is also good but with the correct tuning for SVM, SVM simply outperforms Logistic Regression that has a more complex dataset. It is more capable in looking for optimal decision boundaries. SVM is also very good in handling non-linear relationships data for maximum efficiency. Hence with using the methods in this project, SVM is recommended for use in predicting cardiovascular diseases.