< CIND 820 Final Report >

< The Effectiveness of Logistic Regression and Random Forest Models to Predict Cardiovascular Disease >

By: Sang Won Baek

Student ID: 501144120

Supervisor: Uzair Ahmad, Ph.D

Date: July 25, 2022

GitHub: https://github.com/swb1113/CIND820/blob/main/CIND820_Final_Results.ipynb

Ryerson University

Table of Contents

Introduction ..... Error! Bookmark not defined.
Revised Literature Review ..... Error! Bookmark not defined.
Methodology **4**
Dataset 4
Methodology and Approach 4
Analysis **6**
Attribute Descriptions 6
Exploratory Data Analysis and Visualization 7
Data Preprocessing 12
Logisitic Regression Model using Stratified K(5) Fold Validation 14
Random Forest Model using Stratified K(5) Fold Validation 15
Results **16**
Results of the Logistic Regression Model 16
Results of the Random Forest Model 19
Conclusion and Next Steps **22**
References **23**

Introduction

Cardiovascular diseases (CVD) are the leading cause of death globally, resulting in an estimated 17.8 million deaths per year (Kaptoge et al., 2019). The United Nations Sustainable Development goals highlighted the need to drastically reduce incidence of premature mortality from non-communicable diseases by a third by the year 2030 (Kaptoge et al., 2019). An important step towards achieving the goal set out by the United Nations is to reduce incidence of CVD worldwide. Therefore, the main focus of the capstone project is to answer the question of how practical supervised learning techniques can be in aiding the diagnoses of adverse heart disease events.

The source code for the capstone project can be viewed on GitHub at https://github.com/swb1113/CIND820/blob/main/CIND820_Final_Results.ipynb

Revised Literature Review

Several published research papers outlined a prediction model to predict cardiovascular disease status in patients. A publication in the Institute of Electrical and Electronic Engineers utilized a dataset obtained from the Cleveland, Hungarian, Switzerland, Long Beach VA heart disease database from the UCI machine learning repository to train their prediction model (K. G. Dinesh, K. Arumugaraj, K. D. Santhosh, & V. Mareeswari, 2018). The programming language R was used to perform logistic regression, naïve bayes, random forest, support vector machine, and gradient boosting (K. G. Dinesh et al., 2018). The results of the paper displayed that the logistic regression model resulted in the highest overall accuracy of 0.865, while the support vector machine model resulted in the lowest overall accuracy of 0.798. The publication concluded that the R software gives an immediate mechanism for users to utilize machine learning algorithms for forecasting CVD, but future research should include different ensemble methods of the algorithms to improve performance (K. G. Dinesh et al., 2018).

Furthermore, some studies utilized the use of auto machine learning (AutoML) libraries to predict CVD risk. A paper published in 2019 discovered that through AutoML libraries, biomedical researchers without in depth knowledge on machine learning algorithms could quickly build practical machine learning classifiers to predict cardiovascular disease risk (Padmanabhan, Yuan, Chada, & Nguyen, 2019). Manually created machine learning classifiers were compared to the AutoML classifiers using the Heart UCI dataset which contained 76 attributes (only 12 were used) and 303 records (Padmanabhan et al., 2019). The paper concluded that the AutoML libraries took one hour to build classifiers that outperformed the manually created classifiers that took one month to build (Padmanabhan et al., 2019). Overall, the research team supported the adoption of AutoML in the healthcare domain and aimed to break stereotypes that only trained experts can use machine learning algorithms (Padmanabhan et al., 2019).

The current literature presents a wide scope of interest in the utilization of machine learning algorithms in clinical healthcare settings. Unlike many previous publications that focus

on a wide berth of machine learning algorithms, this project will narrow its focus on logistic regression and decision tree-based (random forest) models. This project also employs the use of simpler supervised learning methods opposed to more complex methods. The use of simpler supervised learning methods ensures that heavy computational power is not needed to recreate this project. Overall, while the current literature presents a solid basis to answer whether machine learning algorithms can be used to aid in the diagnosis of cardiovascular disease, the presented capstone project will insightfully add on to the ever-expanding literature.

Methodology

Dataset:

The dataset used for this capstone project is titled “Heart Failure Prediction Dataset” and can be found at https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction. The dataset is a compilation of 5 different publicly available datasets (from the UCI Machine Learning Repository) containing common features (fedesoriano, 2021). In total, the dataset is comprised of 12 attributes (patient medical features) and 918 observations (fedesoriano, 2021).

Methodology and Approach:

1. Import Kaggle dataset and Libraries:

   Using the Kaggle API, the Heart Failure Prediction Dataset was downloaded onto a personal google drive. The google Colab notebook (Jupyter notebook) was connected to the google drive to gain access to the dataset. Multiple libraries were imported to aid in the analysis of the dataset and to design the machine learning models. Imported libraries used for exploratory data analysis and visualization included NumPy, pandas, matplotlib, seaborn, and plotly express. The logistic regression and random forest models were designed and evaluated using the sklearn library.

2. Exploratory Data Analysis and Visualization:

   Involved a general overlook of the dataset, examining the total number of attributes (12) and observations (918). Checked the data type of the attributes and the number of missing data. The descriptive statistics of the numerical attributes were analyzed along with the description of the categorical attributes. The target variable (heart disease) was thoroughly analyzed in terms of its correlation to other attributes. Data visualization techniques such as correlation plots and multiple pairwise bivariate distributions were carried out to emphasize certain correlating factors and discover patterns in the data. Furthermore, histograms were plotted to visually display the spread of categorical attributes. Lastly, boxplots were plotted to analyze the outliers present in the dataset.

3. Data Preprocessing and Cleaning:

   Several encoding methods were needed to prepare the dataset for the modelling step. One-hot encoding was performed to prepare the data for logistic regression. One-hot encoding transforms categorical data (with no ordinal relationships) into numerical data (0 or 1) for the machine learning model to use. For example, it splits the “Sex” attribute into two columns “Sex_F” and “Sex_M”, then assigns 0 or 1 under the new columns. On the other hand, Label encoding was performed to prepare the data for the random forest model. Label encoding assigns a number for each value in the categorical attribute (does not split attributes). Label encoding was performed to use less disk space (as random forest models can use categorical attributes). Lastly, the heart disease variable was designated as the target variable.

4. Logistic Regression Model:

   A logistic regression model was built using the sklearn library to predict the incidence of heart disease based on the discussed dataset. Stratified K fold validation was performed to evaluate the model as the binary target variable is unbalanced.

5. Random Forest Model:

   A random forest model was built using the sklearn library to predict the incidence of heart disease based on the discussed dataset. Stratified K fold validation was performed to evaluate the model as the binary target variable is unbalanced.

6. Model Evaluation:

   Multiple evaluation metrics were employed to compare the performance of the models. Metrics such as accuracy, precision, recall, f1-score, ROC curve, precision-recall curve, and feature importance were viewed to test the performance of the models.

Analysis

The exploratory analysis and the data visualization portion were performed using python alongside imported libraries NumPy, pandas, matplotlib, seaborn, and plotly express

Attribute Descriptions:

Attribute	Definition	DType
Age	Age of the patient (years)	int64
Sex	Sex of the patient (M: Male, F: Female)	object
ChestPainType	Chest pain type (TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic)	object
RestingBP	Resting blood pressure (mm Hg)	int64
Cholesterol	Serum cholesterol (mm/dl)	int64
FastingBS	Fasting blood sugar (1: if FastingBS > 120 mg/dl, 0: otherwise)	int64
RestingECG	Resting electrocardiogram results (Normal: Normal, ST: having ST-T wave abnormality, LVH: showing probable or definite left ventricular hypertrophy by Estes' criteria)	object
MaxHR	Maximum heart rate achieved (Numeric value between 60 and 202)	int64
ExerciseAngina	Exercise-induced angina (Y: Yes, N: No)	object
Oldpeak	Oldpeak = ST (Numeric value measured in depression)	float64
ST_Slope	The slope of the peak exercise ST segment (Up: upsloping, Flat: flat, Down: down sloping)	object
HeartDisease	Target class (1: Heart Disease, 0: Normal)	int64

Exploratory Data Analysis and Visualization:

The descriptive statistics of the numerical attributes and the description of categorical attributes:

```
df.describe().T
```

	count	mean	std	min	25%	50%	75%	max
Age	918.0	53.510893	9.432617	28.0	47.00	54.0	60.0	77.0
RestingBP	918.0	132.396514	18.514154	0.0	120.00	130.0	140.0	200.0
Cholesterol	918.0	198.799564	109.384145	0.0	173.25	223.0	267.0	603.0
FastingBS	918.0	0.233115	0.423046	0.0	0.00	0.0	0.0	1.0
MaxHR	918.0	136.809368	25.460334	60.0	120.00	138.0	156.0	202.0
Oldpeak	918.0	0.887364	1.066570	-2.6	0.00	0.6	1.5	6.2
HeartDisease	918.0	0.553377	0.497414	0.0	0.00	1.0	1.0	1.0

Age, RestingBP, Cholesterol, and MaxHR may be close to normal distribution. However, more exploratory analysis is needed to confirm.

```
df.describe(include=object).T
```

	count	unique	top	freq
Sex	918	2	M	725
ChestPainType	918	4	ASY	496
RestingECG	918	3	Normal	552
ExerciseAngina	918	2	N	547
ST_Slope	918	3	Flat	460

At initial glance, most of the patients in the dataset are male (725 out of 918) and most patient's FastingBS is not over 120 mg/dl (704 out of 918).

Checking for null or missing values:

```
df.isnull().sum()
```

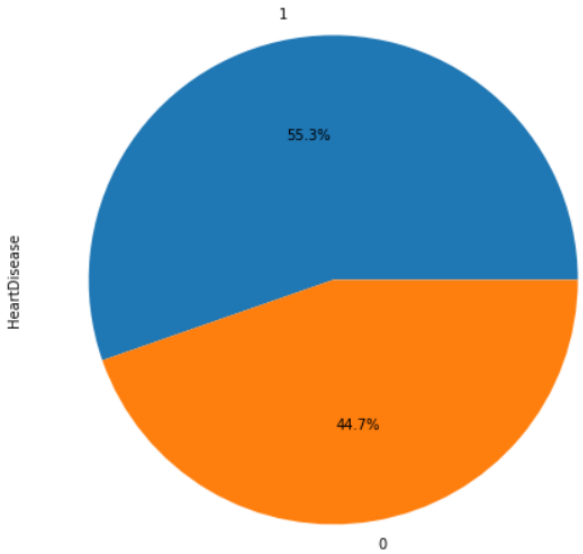
```
Age               0
Sex               0
ChestPainType     0
RestingBP         0
Cholesterol       0
FastingBS         0
RestingECG        0
MaxHR             0
ExerciseAngina    0
Oldpeak           0
ST_Slope          0
HeartDisease      0
dtype: int64
```

There are no missing or null values in the dataset.

HeartDisease (Target) Variable Exploratory Analysis:

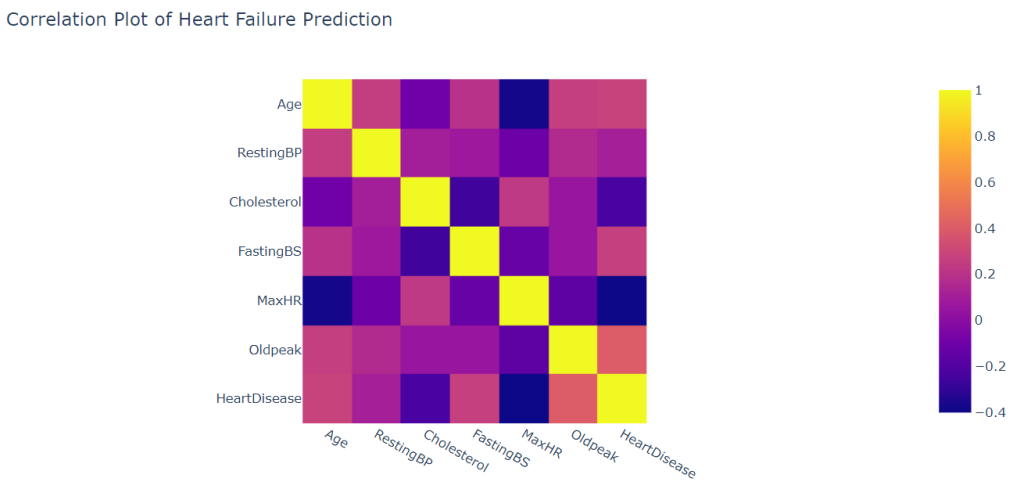
Heart Disease pie chart:

```
1    508
0    410
Name: HeartDisease, dtype: int64
```



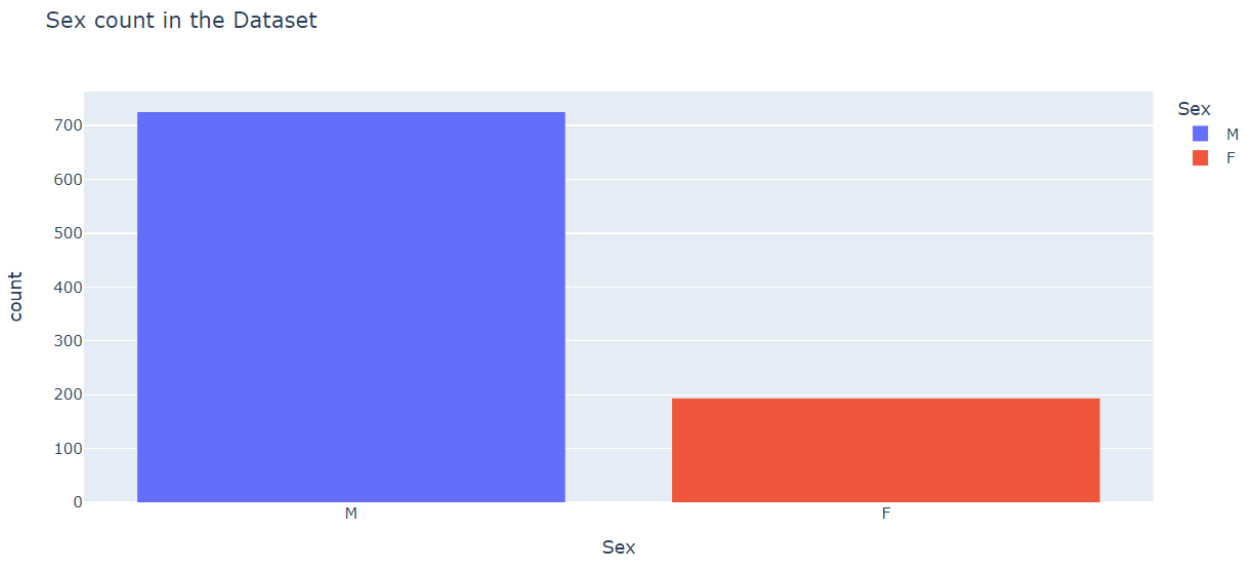
The dataset contains 508 (55.3%) of patients with heart disease and 410 (44.7%) of patients with no heart disease.

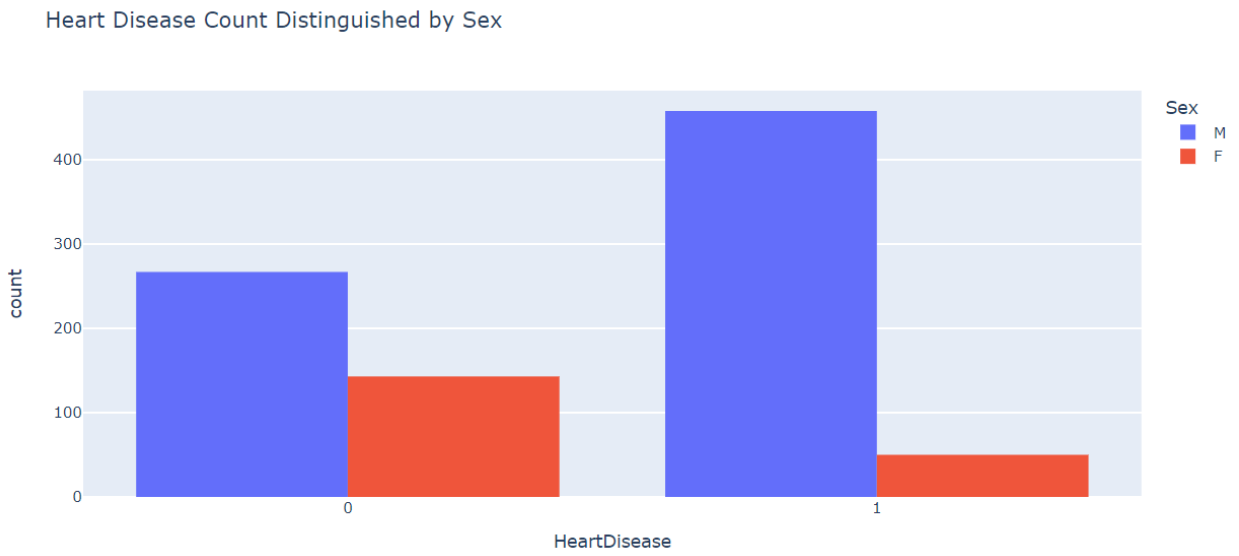
Heart Disease Correlation plot:

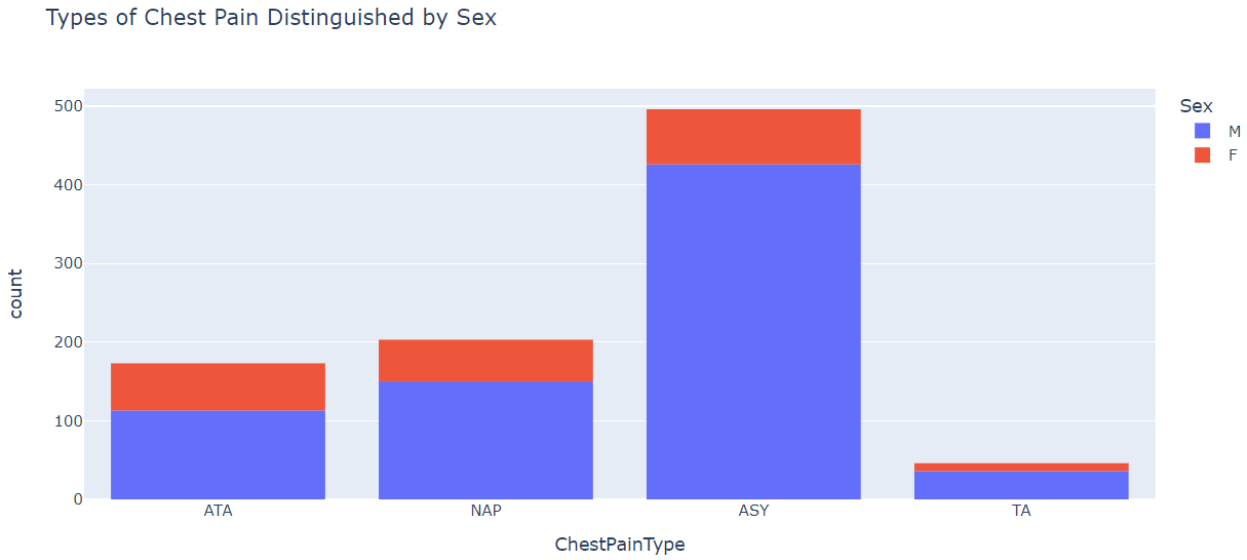


The Correlation plot indicates that heart disease has a negative correlation with max heartrate (-0.4) and a (weaker) negative correlation with cholesterol (-0.23). On the other hand, heart disease has a positive correlation with Oldpeak (0.4), age (0.28), FastingBS (0.27), and RestingBP (0.1).

Heart Disease in relation to sex:







In relation to heart disease and sex, the plots indicate that a larger proportion of males in the dataset experience heart disease opposed to females. There are 63.2% (458/725) of males in the dataset that experience heart disease opposed to 25.9% (50/193) of females that experience heart disease.

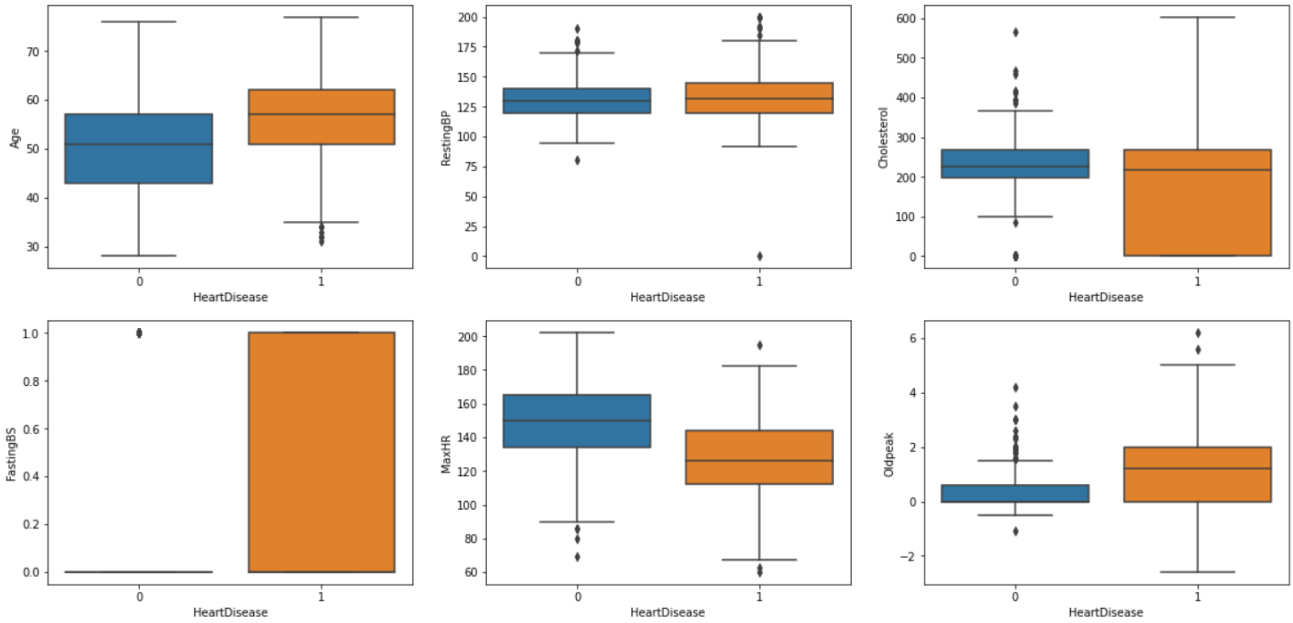
This discrepancy may be caused by factors outside of “sex”. For instance, figure 8 highlights that 41.2% ((113 + 36 + 150)/725) of males in the dataset experience chest pain (typical angina, atypical angina, or non-anginal pain) opposed to 63.7% ((60 + 10 + 53)/193) of females experiencing chest pain.

The plots indicate that while males are more likely to experience heart disease, they are less likely to experience (or report) chest pain than females. This may indicate that males are more likely to experience heart disease due to reluctance to report or ignore chest pain which may further develop into more complex heart disease. Therefore, there may be an under-reporting of male chest pain as males may tend to ignore their chest pains in contrast to females. However, more research and analysis must be performed to confirm the correlation between chest pain, heart disease and sex.

Multiple Pairwise Bivariate Distributions:

The plot shows us that regardless of any other numeric attribute (age, RestingBP, Cholesterol, FastingBS, MaxHR, and Oldpeak), a patient is more prone to heart disease if they have a lower MaxHR (around below 150).

Outlier Analysis:



Most of the outliers tend to be concentrated in Oldpeak and cholesterol with patients with no heart disease. When looking at the dataset as a whole, the outliers only compose a small percentage of the total data points.

Data Preprocessing:

There is no need to deal with missing data as the dataset contains no missing/null data. However, the data must be preprocessed to be prepared for the logistic regression and random forest models.

One-Hot Encoding for Logistic Regression Model:

```
df_lr = pd.get_dummies(df, drop_first=False)
print(df_lr.shape)
df_lr.head()
```

```
(918, 21)
```

	Age	RestingBP	Cholesterol	FastingBS	MaxHR	Oldpeak	HeartDisease	Sex_F	Sex_M	ChestPainType_ASY	...
0	40	140	289	0	172	0.0	0	0	1	0	...
1	49	160	180	0	156	1.0	1	1	0	0	...
2	37	130	283	0	98	0.0	0	0	1	0	...
3	48	138	214	0	108	1.5	1	1	0	1	...
4	54	150	195	0	122	0.0	0	0	1	0	...

5 rows × 21 columns

The one-hot encoding resulted in transforming the original 12 columns into 21 columns. Each categorical values were converted into a new categorical column and a binary value of 1 or 0 were assigned to those new columns. Furthermore, one-hot encoding was chosen as the categorical data had no ordinal value. Therefore, this process converted the categorical data into numerical (integer) data to ensure better prediction from the logistic regression model.

Label Encoding for Random Forest Model:

```
df_tr = df.apply(LabelEncoder().fit_transform)
df_tr.head()
```

	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina	Oldpeak	ST_Slope	HeartDisease
0	12	1	1	41	147	0	1	98	0	10	2	0
1	21	0	2	55	40	0	1	82	0	20	1	1
2	9	1	1	31	141	0	2	25	0	10	2	0
3	20	0	0	39	72	0	1	34	1	25	1	1
4	26	1	2	49	53	0	1	48	0	10	2	0

Label encoding was performed to prepare the data for the random forest model. Each value in the categorical columns were assigned a number. As random forest models are compatible with categorical values, label encoding was performed to use less disk space.

Setting HeartDisease as target attribute:

```
[ ]  target="HeartDisease"
     y=df[target].values
```

Separating the target variable and independent variables for both models:

```
[ ]  # Seperating target variable and independant variables for logistic regression
     feature_col_lr=df_lr.columns.to_list()
     feature_col_lr.remove(target)
```

```
[ ]  # Seperating target variable and independant variables for random forest
     feature_col_tr=df_tr.columns.to_list()
     feature_col_tr.remove(target)
```

The last step of data preprocessing involved setting the y (target) variable as the HeartDisease attribute and to separate the target variable and the independent variables for the logistic regression and random forest models.

Logistic Regression Model using Stratified K (5) Fold Validation:

```
[37] acc_log=[]

     # Using Stratified K fold validation to evaluate model
     kf = StratifiedKFold(n_splits=5)

     for fold , (trn,val) in enumerate(kf.split(X=df_lr,y=y)):

       # Splitting training and validation data
       X_train = df_lr.loc[trn, feature_col_lr]
       y_train = df_lr.loc[trn, target]

       X_valid = df_lr.loc[val, feature_col_lr]
       y_valid = df_lr.loc[val, target]

       # Using Min Max Scaler to scale the data
       scaler=MinMaxScaler()
       X_train = scaler.fit_transform(X_train)
       X_valid = scaler.transform(X_valid)

       # Fitting logistic regression model
       clf = LogisticRegression()
       clf.fit(X_train,y_train)

       # prediction
       y_pred = clf.predict(X_valid)

       # printing the results (classification report, accuracy)
       print(f"FOLD: {fold+1} ")
       print(classification_report(y_valid,y_pred))

       acc = roc_auc_score(y_valid,y_pred)
       acc_log.append(acc)

       print(f"Accuracy for Fold {fold+1} : {acc}\n")
       pass
```

In preparation for the logistic regression model, min max scaler was utilized to properly normalize the data. Stratified 5-fold validation was used to evaluate the model as the binary target variable was unbalanced (55.3% heart disease vs 44.7% normal).

Random Forest Model using Stratified K (5) Fold Validation:

```
[41] acc_RF=[]

     # Using Stratified K fold validation to evaluate model
     kf=StratifiedKFold(n_splits=5)

     for fold , (trn_,val_) in enumerate(kf.split(X=df_tr,y=y)):

         # Splitting training and validation data
         X_train=df_tr.loc[trn_,feature_col_tr]
         y_train=df_tr.loc[trn_,target]

         X_valid=df_tr.loc[val_,feature_col_tr]
         y_valid=df_tr.loc[val_,target]

         # Fitting random forest classifier model
         clf_2=RandomForestClassifier(n_estimators=200,criterion="entropy")
         clf_2.fit(X_train,y_train)

         # Prediction
         y_pred=clf_2.predict(X_valid)

         # Printing results (classification report, accuracy)
         print(f"FOLD: {fold+1} ")
         print(classification_report(y_valid,y_pred))

         acc=roc_auc_score(y_valid,y_pred)
         acc_RF.append(acc)

         print(f"Accuracy for Fold {fold+1} : {acc}\n")
```

Stratified 5-fold validation was used to evaluate the model due to the same reason as the logistic regression where the binary target variable was unbalanced (55.3% heart disease vs 44.7% normal).

Results

RESULTS OF THE LOGISTIC REGRESSION MODEL:

Classification Report and Accuracy:

FOLD: 1

	precision	recall	f1-score	support
0	0.82	0.93	0.87	82
1	0.93	0.83	0.88	102
accuracy			0.88	184
macro avg	0.88	0.88	0.87	184
weighted avg	0.88	0.88	0.88	184

Accuracy for Fold 1 : 0.8800813008130083

FOLD: 2

	precision	recall	f1-score	support
0	0.79	0.90	0.84	82
1	0.91	0.80	0.85	102
accuracy			0.85	184
macro avg	0.85	0.85	0.85	184
weighted avg	0.86	0.85	0.85	184

Accuracy for Fold 2 : 0.8531802965088474

FOLD: 3

	precision	recall	f1-score	support
0	0.96	0.65	0.77	82
1	0.78	0.98	0.87	102
accuracy			0.83	184
macro avg	0.87	0.81	0.82	184
weighted avg	0.86	0.83	0.82	184

Accuracy for Fold 3 : 0.8133668101386896

FOLD: 4

	precision	recall	f1-score	support
0	0.88	0.80	0.84	82
1	0.85	0.91	0.88	101
accuracy			0.86	183
macro avg	0.87	0.86	0.86	183
weighted avg	0.86	0.86	0.86	183

Accuracy for Fold 4 : 0.8578845689446993

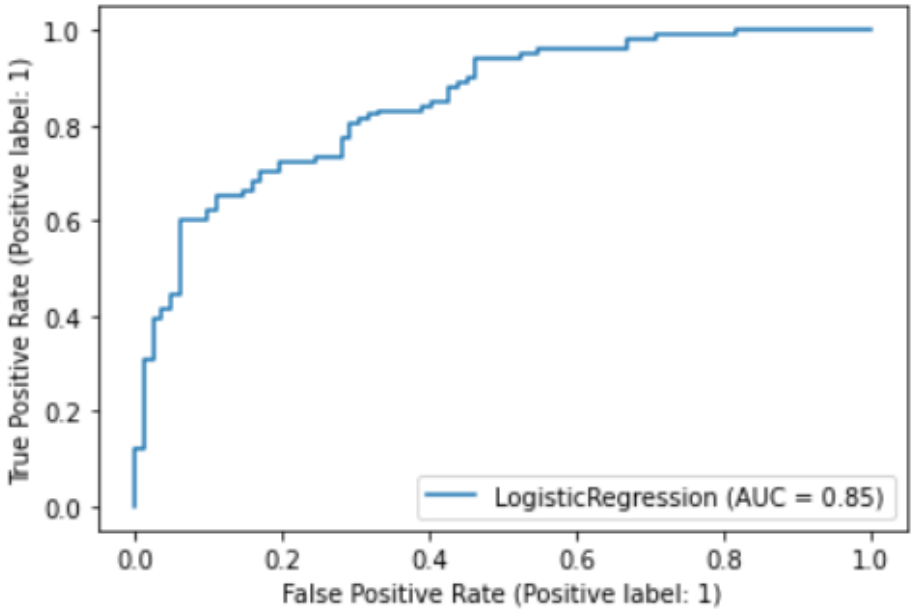
FOLD: 5

	precision	recall	f1-score	support
0	0.69	0.76	0.72	82
1	0.78	0.72	0.75	101
accuracy			0.74	183
macro avg	0.74	0.74	0.74	183
weighted avg	0.74	0.74	0.74	183

Accuracy for Fold 5 : 0.7394349191016663

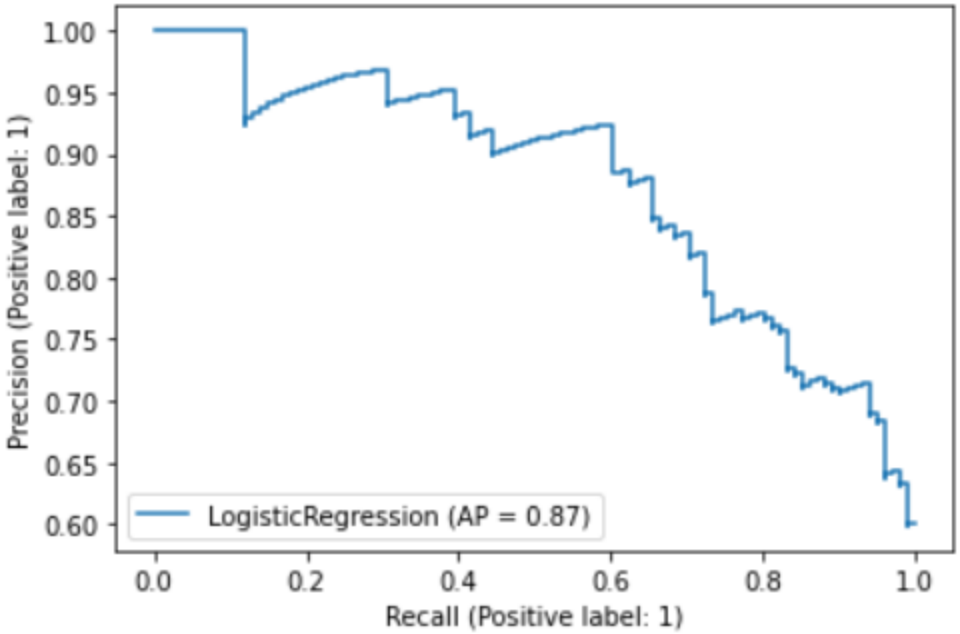
The average accuracy across all 5 folds were 82.88%. Fold 1 had the highest accuracy with 88.01% and fold 5 had the lowest accuracy at 73.94%. In terms of precision (averaged across all 5 folds), out of all the patients that the model predicted to have heart disease, 85% actually did. Therefore, the high precision indicates a relatively low false positive rate. In terms of recall (averaged across all 5 folds), out of all the patients that actually had heart disease, the model correctly predicted this outcome for 84.8% of those patients. Therefore, the high recall indicates a relatively low false negative rate.

ROC (Receiver Operator Characteristic) Curve:



The ROC Curve measures the trade off between sensitivity (true positive rate) and specificity (false positive rate). The AUC (Area Under the Curve) is a general measure of the model's predictive accuracy. The higher (closer to 1) the AUC is, the better the predictive accuracy of the model. The logistic regression model has an AUC of 0.85. Therefore, the logistic regression model has relatively high predictive accuracy.

Precision-Recall Curve:



The Precision-Recall Curve measures the trade off between precision and recall at different thresholds. The AP (average precision)/AUC is a general measure of the model's precision and recall scores. A high AP (close to 1) represents the model having a high precision and high recall score. High precision indicates a low false positive rate, while high recall relates to low false negative rates. The logistic regression model has an AP of 0.87. Therefore, the logistic regression model has a relatively high precision and high recall score.

Logistic Regression Coefficients:

	coefficients
FastingBS	5.106545
ST_Slope_Flat	4.254713
Oldpeak	3.217847
ChestPainType_ASY	2.647163
Sex_M	1.875171
ExerciseAngina_Y	1.529154
RestingBP	1.404335
Age	1.300578
ChestPainType_TA	1.123128
RestingECG_ST	1.039103
ST_Slope_Down	1.025241
RestingECG_Normal	1.008790
RestingECG_LVH	0.954028
ChestPainType_NAP	0.735112
ExerciseAngina_N	0.653987
Sex_F	0.533310
ChestPainType_ATA	0.457569
MaxHR	0.368157
ST_Slope_Up	0.229258
Cholesterol	0.140794

The coefficients were obtained through exponentiating the log odd coefficients. The Coefficients indicate that FastingBS (5.1), ST_Slope_Flat (4.3), Oldpeak (3.2), and ChestPainType_ASY (2.6) are the attributes to most influence the diagnosis of heart disease.

RESULTS OF THE RANDOM FOREST MODEL:

Classification Report and Accuracy:

FOLD: 1

	precision	recall	f1-score	support
0	0.85	0.91	0.88	82
1	0.93	0.87	0.90	102
accuracy			0.89	184
macro avg	0.89	0.89	0.89	184
weighted avg	0.89	0.89	0.89	184

Accuracy for Fold 1 : 0.8935915829746534

FOLD: 2

	precision	recall	f1-score	support
0	0.79	0.91	0.85	82
1	0.92	0.80	0.86	102
accuracy			0.85	184
macro avg	0.86	0.86	0.85	184
weighted avg	0.86	0.85	0.85	184

Accuracy for Fold 2 : 0.8592778574844573

FOLD: 3

	precision	recall	f1-score	support
0	0.98	0.63	0.77	82
1	0.77	0.99	0.87	102
accuracy			0.83	184
macro avg	0.88	0.81	0.82	184
weighted avg	0.86	0.83	0.82	184

Accuracy for Fold 3 : 0.8121712099473937

FOLD: 4

	precision	recall	f1-score	support
0	0.87	0.74	0.80	82
1	0.81	0.91	0.86	101
accuracy			0.84	183
macro avg	0.84	0.83	0.83	183
weighted avg	0.84	0.84	0.83	183

Accuracy for Fold 4 : 0.8273967640666505

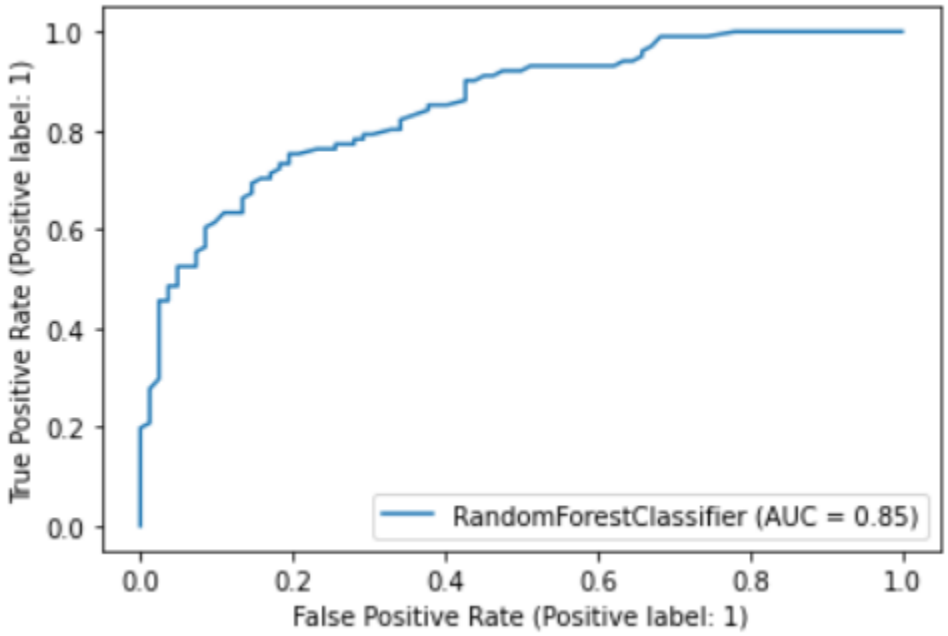
FOLD: 5

	precision	recall	f1-score	support
0	0.71	0.80	0.75	82
1	0.82	0.73	0.77	101
accuracy			0.77	183
macro avg	0.77	0.77	0.76	183
weighted avg	0.77	0.77	0.77	183

Accuracy for Fold 5 : 0.7687756580536103

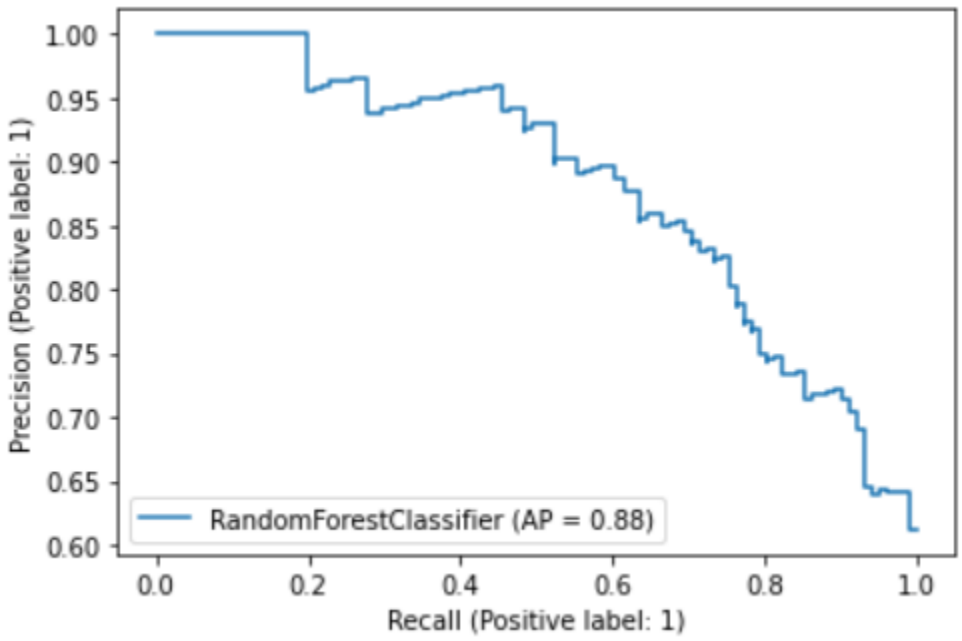
The average accuracy across all 5 folds were 83.23%. Fold 1 had the highest accuracy with 89.36% and fold 5 had the lowest accuracy at 76.88%. In terms of precision (averaged across all 5 folds), out of all the patients that the model predicted to have heart disease, 85% actually did. Therefore, the high precision indicates a relatively low false positive rate. In terms of recall (averaged across all 5 folds), out of all the patients that actually had heart disease, the model correctly predicted this outcome for 86% of those patients. Therefore, the high recall indicates a relatively low false negative rate.

ROC (Receiver Operator Characteristic) Curve:



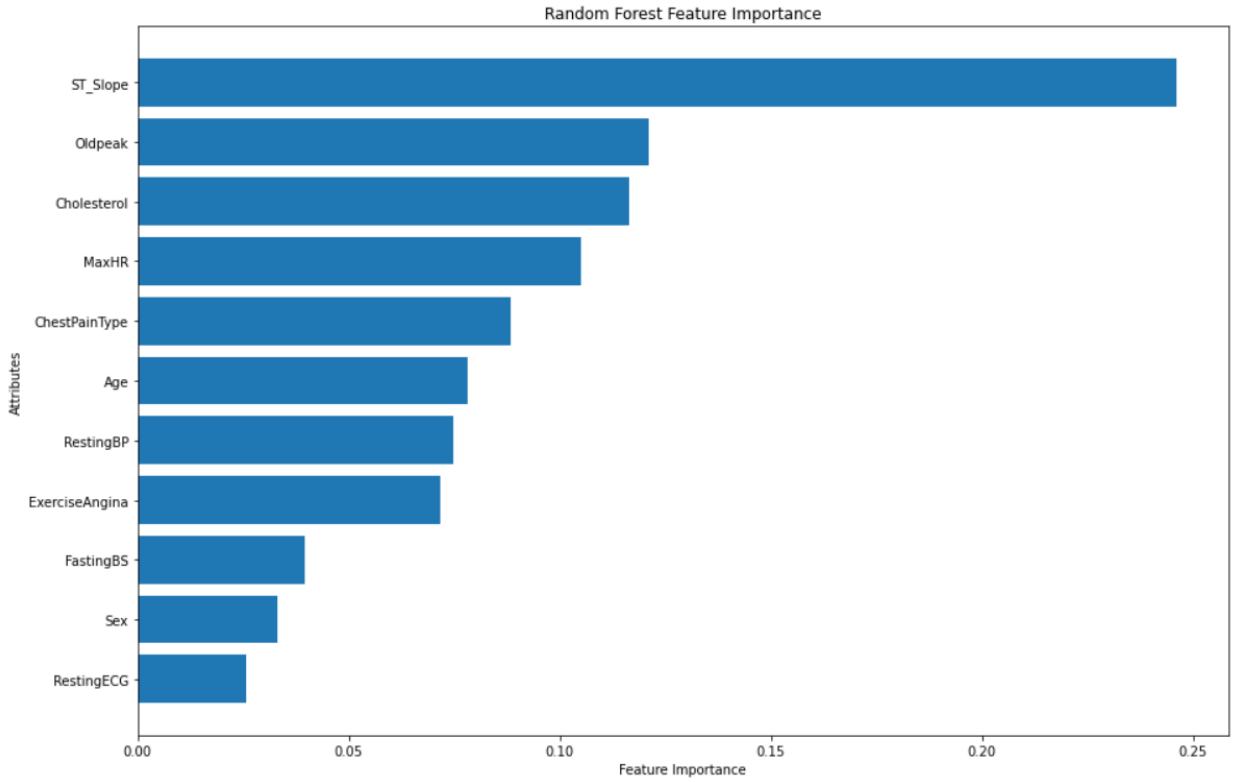
As mentioned, the ROC Curve measures the trade off between sensitivity (true positive rate) and specificity (false positive rate) and the AUC (Area Under the Curve) is a general measure of the model's predictive accuracy. The random forest model has an AUC of 0.85. Therefore, the model has relatively high predictive accuracy.

Precision-Recall Curve:



As mentioned, the Precision-Recall Curve measures the trade off between precision and recall at different thresholds and the AP (average precision)/AUC is a general measure of the model's precision and recall scores. High precision indicates a low false positive rate, while high recall relates to low false negative rates. The Random Forest model has an AP of 0.88. Therefore, the model has a relatively high precision and high recall score.

Random Forest Feature Importance:



The random forest model shows that ST_Slope, Oldpeak, and Cholesterol are the top three most important features when predicting the patient's heart disease status.

COMPARISON BETWEEN LOGISTIC REGRESSION MODEL AND RANDOM FOREST MODEL:

Both the Models had very similar performance for most evaluation metrics. Both models had very similar average accuracy at 82.88% for logistic regression and 83.23% for random forest. Both models had precision rates of 85%. In terms of recall, the random forest model had a slightly better performance of 86% compared to the logistic regression model which had 84.8%.

In terms of the ROC and Precision-Recall curve, both of the models had similar metrics all around. Both the models have AUC scores of 0.85 in regard to the ROC curve, which indicates a relatively high predictive accuracy of the models. The precision-recall curve indicates that the random forest model has an AP of 0.88, while the logistic regression model has an AP of 0.87. therefore, both the models also have a relatively high precision and recall score.

Lastly, when comparing the logistic regression coefficients and the random forest feature importance. Both models agree that ST_Slope and Oldpeak are highly relevant in the diagnosis of Heart disease. The logistic regression model had ST_Slope_Flat as a coefficient of 4.3 and Oldpeak as a 3.2. Likewise, the random forest model pointed out ST_Slope and Oldpeak as the most important features when predicting the patient's heart disease status.

Conclusion and Next Steps

Overall, both the binary classifiers performed very similarly as both the models had an average accuracy of around 83%. In terms of precision and recall, both the models had a precision of 85% (indicating a low false positive rate), while the random forest model had a recall of 86% and the logistic regression had a recall of 85% (indicating a low false negative rate). The AUC and AP scores also showcase the similarity in the logistic regression and random forest models, both models achieved an AUC score of 0.85, and an AP score of 0.87 and 0.88 respectively.

There were several shortcomings in regard to the methodology of the capstone project. For instance, feature engineering would have benefitted the project in removing less important attributes when training the models. Furthermore, as the evaluation metrics are so similar across both models, it may benefit from increasing the number of folds, experimenting with other aspects of the model to improve model performance or to compare both the models with a different tree-based model (such as decision tree classifier or XGBoost).

In conclusion, this capstone project has shown that cardiovascular disease prevalence can be predicted with the use of simple machine learning algorithms such as logistic regression and random forest models. The continuation of this field of research is pivotal towards introducing the benefits of machine learning algorithms in the use of healthcare industries. The next steps of research should include increasing the complexity and optimization of the algorithms to increase their efficiency and efficacy. The use of live data would also benefit to further test the real-world effectiveness of the models.

References

fedesoriano. (September 2021). Heart Failure Prediction Dataset. Retrieved [May 15, 2022] from https://www.kaggle.com/fedesoriano/heart-failure-prediction.

K. G. Dinesh, K. Arumugaraj, K. D. Santhosh, & V. Mareeswari. (2018). Prediction of cardiovascular disease using machine learning algorithms. Paper presented at the - *2018 International Conference on Current Trends Towards Converging Technologies (ICCTCT),* 1-7. doi:10.1109/ICCTCT.2018.8550857

Kaptoge, S., Pennells, L., De Bacquer, D., Cooney, M. T., Kavousi, M., Stevens, G., . . . Altay, S. (2019). World health organization cardiovascular disease risk charts: Revised models to estimate risk in 21 global regions. *The Lancet Global Health, 7*(10), e1332-e1345.

Padmanabhan, M., Yuan, P., Chada, G., & Nguyen, H. V. (2019). Physician-friendly machine learning: A case study with cardiovascular disease risk prediction. *Journal of Clinical Medicine, 8*(7), 1050.