Group names: Joseph Barr, Linh Ta, and Sophia Wang

11 May 2021

Modeling Cardiovascular Disease

Description:

Cardiovascular disease is a term that classifies many health conditions involving the cardiovascular system. These health conditions include heart attacks, heart disease, strokes (ischemic and hemorrhagic), heart failure, arrhythmia, and heart valve problems. Cardiovascular disease is the leading cause of deaths in the United States for most demographic groups. According to the CDC website, “One person dies every 36 seconds in the United States from cardiovascular disease.” In this project it is our goal to be able to predict the presence or absence of cardiovascular disease in patients based on their health, demographics, and physical activities. In order to accomplish this goal, many machine learning algorithms were used to predict the presence or lack thereof of cardiovascular disease.

The dataset that is being analyzed comes from a Data Science student at Ryerson University in Toronto, Ontario. The data was gathered and recorded at the time of the medical examination of the patients. The dataset records the medical data of 70,000 patients. The medical records are 12 variables which are our predictors in this study. These predictors are divided into three categories. The first category are the objective predictors. This is factual information that is not subject to testing or opinions and in the dataset are age, height, weight, and gender. The second category contains the examination predictors. The examination predictors are those that were gathered in the medical examination and are called ap_high, ap_low, cholesterol, and glucose. And lastly, the third category are the subjective predictors. These predictors are information that the patients filled which says whether they smoke, drink alcohol, or are

physically active. The variables are titled, smoke, alcohol, and physical activity. The response variable in this study is cardiovascular disease and is binary. The variable being binary means that if cardiovascular disease is present in a patient it will be recorded as a 1 and if it is not found then a 0 is recorded.

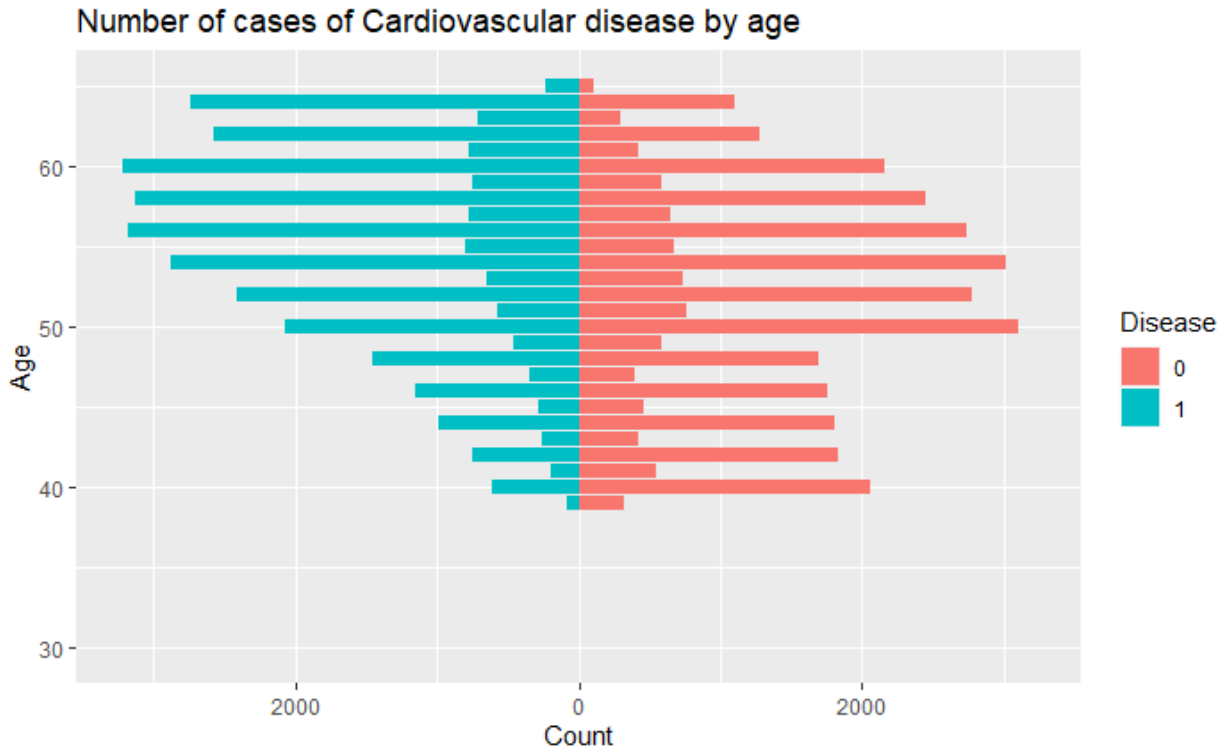
Method of approach:

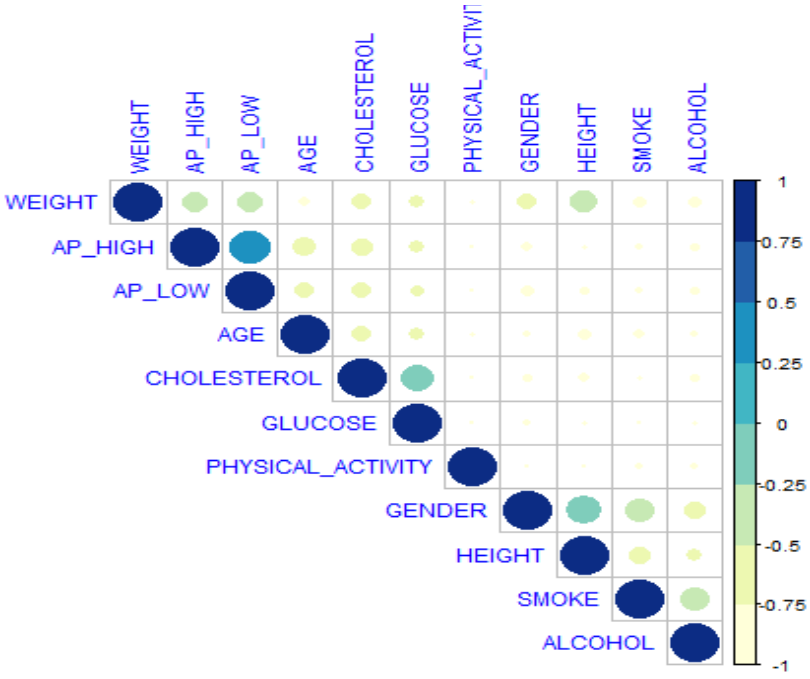
The investigation approach taken was to further understand some simple statistics using a data summary and visualizations. This allowed for any patterns or interesting statistics that appeared to be subject to analysis. After the initial investigation, four different machine learning algorithms were used to model the data. The methods that were used were Logistic Regression, Random Forest, K Nearest Neighbors, and Support Vector Machine. These techniques were used to predict whether or not cardiovascular disease was present or not within patients. The model that produced the lowest error rate would be the model that best predicts cardiovascular disease.

Data exploration:

The dataset contains 12 integer variables, therefore, we change some variables to the correct data type. More particularly, we change Gender, Cholesterol, Glucose, Smoke, Alcohol, Physical_activity and Cardio_disease to factors. The summary of the data shows us all the levels in those factor features and gives us the minimum, 1st quarter, median, mean, 3rd quarter and maximum of other numerical variables.

```
      AGE          GENDER         HEIGHT          WEIGHT          AP_HIGH          AP_LOW       CHOLESTEROL GLUCOSE   SMOKE
 Min.   :30.00   1:44795   Min.   : 55.0   Min.   : 11.00   Min.   : 60.0   Min.   : 40.00   1:51582     1:58474   0:62730
 1st Qu.:48.00   2:23988   1st Qu.:159.0   1st Qu.: 65.00   1st Qu.:120.0   1st Qu.: 80.00   2: 9315     2: 5074   1: 6053
 Median :54.00             Median :165.0   Median : 72.00   Median :120.0   Median : 80.00   3: 7886     3: 5235
 Mean   :53.33             Mean   :164.4   Mean   : 74.12   Mean   :126.6   Mean   : 81.38
 3rd Qu.:58.00             3rd Qu.:170.0   3rd Qu.: 82.00   3rd Qu.:140.0   3rd Qu.: 90.00
 Max.   :65.00             Max.   :250.0   Max.   :200.00   Max.   :240.0   Max.   :190.00
 ALCOHOL    PHYSICAL_ACTIVITY CARDIO_DISEASE
 0:65094    0:13525           0:34742
 1: 3689    1:55258           1:34041
```





As we can see in the population pyramid above, older people from around 50 to 60 will more likely be diagnosed with Cardiovascular disease. Whereas, there are no records below 39,

which reveals that people below 39 are unlikely to have Cardiovascular disease's symptoms or the disease.

1. **Logistic regression:**

We use logistic regression to gain initial insights of the dataset. We split the data into a 70/30 training/testing test set and then use the glm(general linear model) function to fit a logistic regression model. The target of the generalized linear model is CARDIO_VASCULAR and all other variables are features.

The summary of the logistic regression model gives out all p values smaller than 5%, therefore, it is reasonable to count on the significance of each predictors. We can say that all predictors influence whether a person will have Cardiovascular disease or not and they should be included in all the models. The accuracy given out by the model is 72.89%.

2. **Support vector machine**

For Support Vector Machine, we split the data into a 70/15/15 training/dev/test set and trained 3 models: a model with type='C-classification' and kernel='linear', a model with type='C-classification', and a model with just the default parameters.

On the dev set, the model with type='C-classification' specified performed best with an accuracy of 74.09%. The model with type='C-classification' and kernel='linear' had an accuracy of 73.08%, and the model with default parameters had accuracy of 74.04%. However, on the test set, the model with type='C-classification' had 72.49% accuracy. The other models also performed worse on the test set, with the 2nd place model with default parameters having 72.43% accuracy.

!	0	1
0	4066	1763
1	1083	3406

model: 'C-classification'

t	0	1
0	4193	1651
1	1027	3446

model: default

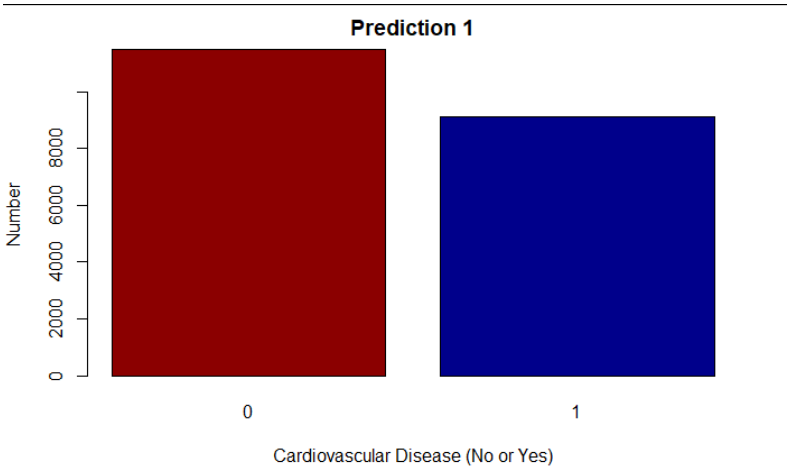
	0	1
0	4290	1847
1	930	3250

model: 'C-classification' and kernel='linear'

3. KNN - K nearest neighbor:

K nearest neighbor utilizes a 70% training subset and a 30% testing subset of the data. However, prior to subsetting the data the dataset was normalized. After the data was normalized a subset of the data was formed by randomly sampling from the newly normalized dataset. After the training and testing subsets were instantiated, the model was run with k equalling the square root of the number of predictors (11). The KNN model provided a prediction accuracy of 72.21%.

	cardio_test_cat	
cardio_test_prediction	0	1
0	8076	3422
1	2312	6825

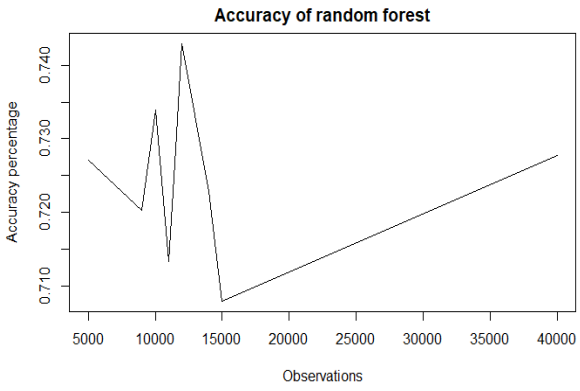


4. Random Forest:

The Cardiovascular disease dataset is too large, therefore, if we use the whole dataset, the model will be overfitted, which increases the error rate and decreases the accuracy. Consequently, we subset the dataset. We randomly pick a number of observations without replacement, starting from 5000 observations, and use a for loop to continue increasing the number of observations until the rate error stops decreasing. During each iteration, the random forest is implemented with the data split into 70% for training and 30% for testing (features test

and target test). After the model is fitted, we apply the model to the feature test and calculate the error rate and the accuracy.

Test_rate_error <dbl>	Accuracy <dbl>	Obs <dbl>
0.2728486	0.7271514	5000
0.2797332	0.7202668	9000
0.2660887	0.7339113	10000
0.2867536	0.7132464	11000
0.2713043	0.7286957	11500
0.2570158	0.7429842	12000
0.2772089	0.7227911	14000
0.2920649	0.7079351	15000
0.2722727	0.7277273	40000



We can see from the table and the graphic that at 12000 observations, the accuracy of random forest peaks. From 14000 to 15000 and 40000 observations, the accuracy does not change much and it is lower than the accuracy at 12000 observations. From this model, we can see that the most effective number of observations we should use for random forest in this dataset is 12000 observations with accuracy of 74.29%.

Method Comparison:

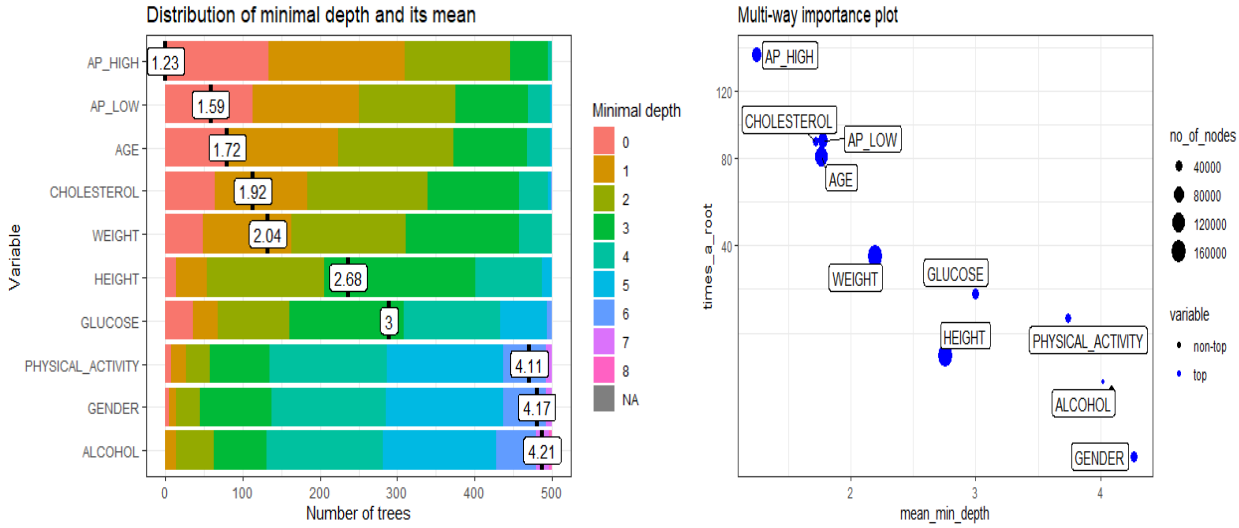
Method	Accuracy
Logistic Regression	72.89%
Support vector machine	72.43%
Random forest	74.29%
K nearest neighbor	72.21%

For the four methods of approach, the prediction accuracy was able to be determined. The method that produced the lowest error rate was Random Forest with an accuracy of 74.29%. The remainder of the models all fell within less than 1% difference of each other showing that none of them are distinguishably better than the others.

After comparing the results gathered here to the results of other individuals who have used this dataset, the general consensus is Logistic Regression produces the highest accuracy of 73%. However, the Random Forest method that we implemented produces a higher accuracy of 74.29%.

Final Approach:

Random Forest, which resulted in 74.29% of accuracy, is the most efficient model amongst all 4 models. In addition, we construct the random forest model with a default of 500 decision trees. Random forest will grow the tree and add additional randomness to the model. It searches for the best feature among a random subset of features, which will result in a wide diversity that usually results in a better model.



Results:

Based on the graphs and information above, AP_HIGH, CHOLESTEROL and AGE are the most 3 important features to look at when predicting the presence of Cardiovascular disease. The least 3 important features that we have to pay attention to are GENDER, ALCOHOL and PHYSICAL_ACTIVITY. Even though they are the least 3 important predictors, they still have

their own weights since they are all significant predictors. Therefore, those parameters still influence the classification process of Cardiovascular disease.

Conclusion:

In conclusion, blood pressure, cholesterol level and age are the most important factors when predicting Cardiovascular Disease in patients. Therefore, in order to prevent Cardiovascular Disease it is crucial to monitor an individual's blood pressure and cholesterol levels. Age is an objective predictor and thus cannot be altered by the patient. However, due to blood pressure and cholesterol being a medical predictor, these values can be changed by the patient. The best course of action for changing these predictor's values is by living a healthy lifestyle and monitoring the amount of saturated fat in the food you are eating.

References:

"Heart Disease Facts." *Centers for Disease Control and Prevention*, Centers for Disease Control and Prevention, 8 Sept. 2020, www.cdc.gov/heartdisease/facts.htm.

James, Gareth, et al. *An Introduction to Statistical Learning: with Applications in R*. Springer, 2021.

V, David. "Cardiovascular Diseases Dataset (Clean)." *Kaggle*, 14 Mar. 2020, www.kaggle.com/aiaiaidavid/cardio-data-dv13032020/code.

"What Is Cardiovascular Disease?" Www.heart.org, www.heart.org/en/health-topics/consumer-healthcare/what-is-cardiovascular-disease.

R Code:

1. Data exploration:

```r
heart[c("GENDER", "CHOLESTEROL", "GLUCOSE", "SMOKE", "ALCOHOL", "PHYSICAL_ACTIVITY", "CARDIO_DISEASE")] <- lapply(heart[c("GENDER", "CHOLESTEROL", "GLUCOSE", "SMOKE", "ALCOHOL",
"PHYSICAL_ACTIVITY", "CARDIO_DISEASE")], factor)
summary (heart)
df <- data.frame(unique(heart$AGE)[order(unique(heart$AGE))])
df$Age <- df$unique.heart.AGE..order.unique.heart.AGE...
df = df %>% select(-c(unique.heart.AGE..order.unique.heart.AGE...))
df$Disease = replicate(nrow(df), factor("1"))
count <- numeric()
i = 1
for (ele in unique(heart$AGE)[order(unique(heart$AGE))]) {
  count[i]=nrow(heart %>% filter(AGE == ele, CARDIO_DISEASE == 1))
  i = i + 1
}
df$Count = count
ex <- data.frame(unique(heart$AGE)[order(unique(heart$AGE))])
ex$Age <- ex$unique.heart.AGE..order.unique.heart.AGE...
ex = ex %>% select(-c(unique.heart.AGE..order.unique.heart.AGE...))
ex$Disease = replicate(nrow(df), factor("0"))
count.0 <- numeric()
i = 1
for (ele in unique(heart$AGE)[order(unique(heart$AGE))]) {
  count.0[i]=nrow(heart %>% filter(AGE == ele, CARDIO_DISEASE == 0))
  i = i + 1
}
ex$Count = count.0
full <- rbind(ex, df)
Full

library(ggplot2)
ggplot(full, aes(x = Age, fill = Disease,
  y = ifelse(test = Disease == "1", yes = -Count, no = Count))) +
  geom_bar(stat = "identity") +
  scale_y_continuous(labels = abs, limits = max(full$Count)*c(-1,1)) +
  labs(title = "Number of cases of Cardiovascular disease by age", x = "Age", y = "Count") +
  coord_flip()
```

```r
library(corrplot)
library(RColorBrewer)
library(tidyverse)
correlation <- cor(cardio_normal)

png("file2.png", width = 350, height = 400)

corrplot(correlation, type = "upper", order = "hclust", col=brewer.pal(n=8, name = "YlGnBu"), tl.cex = 1, tl.col = "Blue")
```

2. Logistic regression

```r
n=nrow(cardio)
train=sample(1:n,0.7*n)
test=-train

cardio.glm <- glm(cardio[train,]$CARDIO_DISEASE ~ ., data = cardio[train,], family =
"binomial")
summary(cardio.glm)
p = predict(cardio.glm, cardio[test,], type="response")
cardio.absent = rep(0,length(p))
cardio.absent[p > 0.5] = 1
table(cardio.absent, cardio[test,]$CARDIO_DISEASE)
mean(cardio.absent == cardio[test,]$CARDIO_DISEASE)
```

3. Support vector machine

```
set.seed(1)
n=nrow(cardio)
train=sample(1:n,0.7*n)
remain=setdiff(1:n,train)
dev = sample(remain,size = 0.5*length(remain))
test = setdiff(remain,dev)
cardio1.svm <- svm(cardio[train,]$CARDIO_DISEASE ~ ., data = cardio[train,], type =
'C-classification',kernel='linear')
cardio2.svm <- svm(cardio[train,]$CARDIO_DISEASE ~ ., data = cardio[train,], type =
'C-classification')
cardio3.svm <- svm(cardio[train,]$CARDIO_DISEASE ~ ., data = cardio[train,])
p1 = predict(cardio1.svm, cardio[dev,],type="response")
cm1 = table(p1,cardio[dev,]$CARDIO_DISEASE)
p2 = predict(cardio2.svm, cardio[dev,],type="response")
cm2 = table(p2,cardio[dev,]$CARDIO_DISEASE)
p3 = predict(cardio3.svm, cardio[dev,],type="response")
cardio3.absent = rep(0,length(p3))
cardio3.absent[p3 > 0.5] = 1
cm3 = table(cardio3.absent, cardio[dev,]$CARDIO_DISEASE)
accuracy <- function(x){sum(diag(x)/sum(rowSums(x)))*100}
accuracy(cm1)
accuracy(cm2)
accuracy(cm3)
```

4. Random forest:

```
```{r}
obs = c(5000, 9000, 10000, 11000, 11500, 12000, 14000, 15000, 40000)
error_df<-data.frame(matrix(ncol = 3, nrow = length(obs)))
colnames(error_df) <- c('Test_rate_error',"Accuracy",'Obs')

i = 1
for (ele in obs) {
  s = sample(nrow(heart), ele)
  heart.set = heart[s,]

  split_index <- as.vector(createDataPartition(heart.set$CARDIO_DISEASE, p = 0.7, list = F))
  training = heart.set[split_index, ]
  features_test = heart.set[-split_index, !(colnames(heart.set) %in% c('CARDIO_DISEASE'))]
  target_test = heart.set[-split_index, "CARDIO_DISEASE"]
  rf_train <- randomForest(CARDIO_DISEASE~., data= training, mtry = 3)
  preds.heart <- predict(rf_train, newdata = features_test)
  mean.rate.error <- mean(preds.heart!=target_test)
  error_df[i,"Obs"] = ele
  error_df[i, "Test_rate_error"] = mean.rate.error
  error_df[i, "Accuracy"] = 1-mean.rate.error
  i = i+1
}
Error_df
plot(error_df$Obs, error_df$Accuracy, xlab = "Observations", ylab = "Accuracy percentage", main = "Accuracy of random forest", type = "l")
```
```

5. K Nearest Neighbor

```
```{r}
normalize <- function(x){
  return ((x-min(x)) / (max(x) - min(x)))
}

accuracy <- function(x){
  sum(diag(x) / (sum(rowSums(x)))) * 100
}
cardio_normal <- as.data.frame(lapply(Cardio[1:11], normalize))
summary(cardio_normal)
randomSample <- sample(1:nrow(cardio_normal), 0.7 * nrow(cardio_normal), replace = FALSE)
cardio_train <- cardio_normal[randomSample,]
cardio_test <- cardio_normal[-randomSample,]
cardio_subset <- Cardio[c("AGE", "GENDER", "HEIGHT", "WEIGHT", "AP_HIGH", "AP_LOW", "CHOLESTEROL", "GLUCOSE", "SMOKE", "ALCOHOL", "PHYSICAL_ACTIVITY", "CARDIO_DISEASE")]
cardio_train_cat <- Cardio_subset[randomSample,12]
cardio_test_cat <- Cardio_subset[-randomSample,12]

library(class)
cardio_test_prediction1 <- knn(train = cardio_train, test = cardio_test, cl =  cardio_train_cat, k = 265)
plot(cardio_test_prediction1, xlab = "Cardiovascular Disease (No or Yes)", ylab = "Number", col = c("dark red", "dark blue"), main = "Prediction 1")
tab <- table(cardio_test_prediction1,cardio_test_cat)

accuracy(tab)
table(cardio_test_prediction1,cardio_test_cat)
```
```