Predicting the Chance of Heart Attack with a Machine Learning Approach – Supervised Learning

Lanting Zhu[1] and Guillermo Goldsztein[#]

[1] Centennial High School, USA
[#] Advisor

ABSTRACT

Machine learning is a multidisciplinary field combining statistics, computer science and artificial intelligence. This research finds a way to use machine learning to predict the chance of heart attack based on information about the patient. There are 13 features collected about each patient which are age, sex, cholesterol, chest pain type, maximum heart rate achieved, resting blood pressure, resting electrocardiographic results, fasting blood sugar, exercise-induced angina, previous peak, slope, number of major blood vessels, and thalassemia. The information of all the patients is put into a dataset. The dataset is split into two sets, one for training and another for validation. A computer model using a supervised learning algorithm is developed and trained to predict the chance of heart attack. During training, the training set is used for training the model, while the validation set is used for evaluating the accuracy of the model.

Introduction

Machine learning is a field of artificial intelligence that is an intersection of statistics and computer science[1]. It can also be described as the process of solving an issue by developing a predictive model and training it using a dataset[2]. The dataset is a collection of examples that are like units. In this research, a unit is the collection of information about each patient. There are two main types of machine learning approaches: supervised and unsupervised. In supervised learning, training data is labeled so a pattern can be deduced from the data. In unsupervised learning, inferences are drawn from unlabeled data. Supervised learning is one of the most commonly used types of machine learning[1], and it can be used in a wide range of areas including image classification, disease prediction, fraud detection, and risk assessment. This research develops a computer model using a supervised learning algorithm to predict the chance of heart attack. The research uses two types of data: features and labels. The features are the inputs while the labels are the outputs. The research has 13 features which are information collected about each patient and a label which is the chance of heart attack. Algorithms are trained to identify patterns in data to make the best decisions and predictions. It tries to mimic the way humans learn and improves in accuracy with data and experience.

Supervised Learning and Dataset

Heart attack, which is caused by the sudden occurrence of coronary thrombosis, usually results in partial heart muscle death[3]. They happen when your heart muscle cannot get enough oxygen and nutrients to function. There are many factors as to why they happen such as age, lifestyle, genetics, and different medical conditions.

       A computer model is developed using a supervised learning algorithm to predict the chance of heart attack. The dataset consists of information about 303 patients, and there are 13 features collected about each patient. Table 1 gives a sample of the dataset with the information about 6 patients and the "output", chance of

heart attack. In Table 1, columns 1-13 each represent a feature. The information is used to predict the chance of heart attack for each patient.

The abbreviations in the heading rows of Table 1 are explained below[4]:

1. **age** refers to the age of the patient.
2. **sex** refers to the gender of the patient. Value 0 means female, value 1 means male.
3. **chol** refers to cholesterol in mg/dl fetched via a BMI sensor.
4. **cp** refers to the type of chest pain. Value 0 means the patient has typical angina, value 1 means the patient has atypical angina, value 2 means the patient has non-anginal pain, value 3 means the patient has asymptomatic.
5. **thalach** refers to the maximum heart rate achieved.
6. **trtbps** refers to resting blood pressure in mmHg.
7. **restecg** refers to the patient's resting electrocardiogram. Value 0 means the result is normal, value 1 means the ST-T wave is abnormal, value 2 means showing probable or definite left ventricular hypertrophy.
8. **fbs** refers to fasting blood sugar. Value 0 means the fasting blood sugar is < 120 mg/dl, value 1 means the fasting blood sugar is > 120 mg/dl.
9. **exng** refers to exercise-induced angina. Value 0 means no, value 1 means yes.
10. **oldpeak** refers to exercise-induced ST depression compared to rest.
11. **slp** refers to the slope of the ST segment at the exercise's peak. Value 0 indicates a downsloping segment, value 1 indicates a flat segment, value 2 indicates an upward segment.
12. **caa** refers to the amount of major vessels ranging from 0 – 3.
13. **thal** refers to a blood disorder called thalassemia. Value 1 means certain parts of the heart do not have blood flow, value 2 means the blood flow is normal, value 3 means a blood flow is detected but abnormal.

Table 1. Heart attack dataset and prediction sample.

age	sex	chol	cp	thalach	trtbps	restecg	fbs	exng	oldpeak	slp	caa	thal	output
63	1	233	3	150	145	0	1	0	2.3	0	0	1	**1**
37	1	250	2	187	130	1	0	0	3.5	0	0	2	**1**
41	0	204	1	172	130	0	0	0	1.4	2	0	2	**1**
56	1	236	1	178	120	1	0	0	0.8	2	0	2	**1**
57	0	354	0	163	120	1	0	1	0.6	2	0	2	**1**
67	1	286	0	108	0	160	0	1	1.5	1	3	2	**0**
62	0	268	0	160	0	140	0	0	3.6	0	2	2	**0**

Dataset obtained from Kaggle data repository that is used to build machine learning models [4].

Table 1 lists 13 features which are common factors of heart attacks. The last column "output" describes the chance of a heart attack with 0 being less chance of a heart attack and 1 being a greater chance of a heart attack.

This research uses a supervised learning approach, which is the machine learning technique that uses labeled datasets, to try to predict the chance of heart attack. The datasets are designed to train algorithms to classify data so the algorithm can make predictions. The dataset is split into features and labels. The label for this dataset is the "output", the chances of having a heart attack, and the features are the text in the top row from "age" to "thal".

Methods

Binary Classification Problem

The label for this dataset has two possible values, 0 or 1. This is known as a binary classification problem. Binary classification classifies elements of a dataset into two groups, called classes. The model for a binary classification problem will take in data from the features and output a value of 0 or 1. This is denoted as $\hat{y}$ and if $\hat{y}$ is < 0.5 it is in category 0 and if $\hat{y}$ is > 0.5 it is in category 1.

Logistic Regression

In order to develop a model for binary classification problems, logistic regression is used. Logistic regression is one of the most popular and widely used machine learning algorithms under the category of supervised learning.

To make sure that the model's prediction output is either 0 or 1, an activation function, commonly known as a sigmoid function, is applied. The sigmoid function is a mathematical function that is used in logistic regression to map the predicted values to probabilities. By using this, it makes it easier to sum the error of every patient, in turn allowing us to move the prediction line in small steps until the minimum summation error is as low as possible. The equation of the sigmoid function can be found here:

Equation 1: Equation for the sigmoid function:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

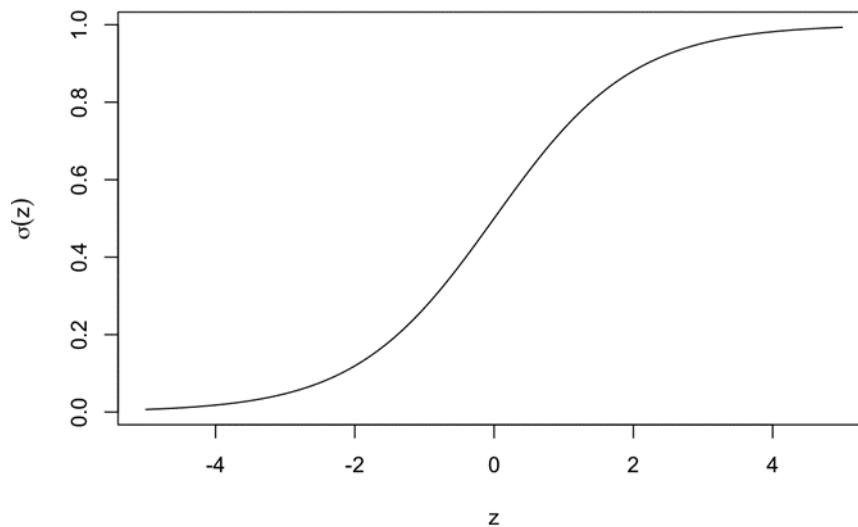A graph showing the sigmoid function is illustrated in Figure 1:

Figure 1. A graph showing the sigmoid function

The primary properties of a sigmoid function are:

a. For all values of x, $0 < \sigma(x) < 1$.
b. It is an increasing function and as x increases, $\sigma(x)$ increases. Vice versa, as x decreases, $\sigma(x)$ decreases. Vice versa, as x decreases, $\sigma(x)$ decreases.
c. $\sigma(0) = 0.5$.

The $\sigma(x)$ values are only between 0 and 1 because in a binary classification problem there can only be two outputs, 0 and 1. There are different ranges for the x values depending on the features of the dataset. They are denoted as $x_1, x_2, \ldots, x_{13}$ since there are 13 features in this data set. x_1 corresponds to the age, x_2 corresponds to the sex, etc. $\hat{y}$ is the predicted output that is defined by the following equation:

Equation 2: Equation for the logistic regression:
$$\hat{y} = \sigma(w_1 x_1 + w_2 x_2 + \ldots + w_{13} x_{13} + b)).$$

Binary Cross Entropy Loss

Binary cross entropy is among the most common loss functions for training binary classifiers. It can be written using NumPy in the following line:

return -(y * np.log(yhat) + (1 - y) * np.log(1 - yhat)).mean()

To better understand binary cross entropy, assume the label of an example dataset is *y*. Remember that *y* can only be 0 or 1 and assume the value of *y* is known. $\hat{y}$ is the predicted value. The binary cross entropy loss equation is as follows:

Equation 3: Equation for the binary cross entropy loss:

$$BCE(\hat{y}, y) = -y \log(\hat{y}) - (1 - y) \log(1 - \hat{y})$$

The primary properties of a binary cross entropy error are:

a. $BCE(\hat{y}, y) \geq 0$.
b. The closer $\hat{y}$ is to y, the smaller $BCE(\hat{y}, y)$ is.
c. If $\hat{y} = y$, then $BCE(\hat{y}, y) = 0$.

$BCE(\hat{y}, y)$ is a measure of the error the model makes in predicting the label of the example. In simpler terms $BCE(\hat{y}, y)$ is just the measure of the difference between $\hat{y}$ and y.

The error in mean binary cross entropy errors depend on the parameters $w_1, w_2, \ldots, w_{13}, b$. If the parameters were changed, the error would also change. These parameters are selected to make the mean binary cross entropy error as small as possible.

Model Training

In order to teach the computer to make accurate predictions, training is used. In model training, scientists try to work to fit the best biases to an algorithm to minimize the loss function over the prediction range. The dataset used is split into two sets, one for training and another for validation and testing. The following is the code for the training:

model = 0

```
model = Sequential()

model.add(Dense(2, activation='relu'))

model.add(Dense(1, activation='sigmoid'))

model.compile(loss='binary_crossentropy', metrics = ['accuracy'])

model.fit(X_train_scaled, y_train,epochs = 400, verbose=0, validation_data = (X_val_scaled,y_val))

J_list = model.history.history['loss']

plt.plot(J_list)
```

The accuracy of the model is 0.901639 meaning around 90% of diagnosis made with the algorithm is correct. During the training the "epochs" value, as seen in the code above, was adjusted to get the highest accuracy possible. "epochs" is the number of passes the training dataset through the algorithm.

Results

Having an epoch of 40 gave the most accurate result of 0.901639 or around 90% accuracy in predictions. Table 2 shows the evaluation of the machine learning model using training data after 400 iterations.

"Loss" represents the summation of errors of the model. It measures how well the model is doing. meaning the loss will be high if the errors are high and vice versa. The loss was calculated using cross entropy which will be covered in more detail later in the article. Accuracy measures how well the model predicts. This is done by comparing the model's predictions with the correct values from the training set. The accuracy is measured in percentages. An example would be if there were five test cases. If the model correctly predicts the output of four out of the five cases, the accuracy would be 80% or 0.8.

Table 2. Evaluation of the machine learning model

	loss	accuracy	val_loss	val_accuracy
0	1.242390	0.260331	1.015842	0.278689
1	1.205138	0.322314	0.991623	0.393443
2	1.178557	0.351240	0.970021	0.393443
3	1.154705	0.363636	0.949447	0.393443
4	1.131240	0.367769	0.929241	0.393443
…				
395	0.359711	0.818182	0.270695	0.901639
396	0.359628	0.818182	0.270855	0.901639
397	0.359425	0.818182	0.270829	0.901639
398	0.359608	0.818182	0.271165	0.901639
399	0.359560	0.818182	0.271213	0.901639

This machine learning evaluation is run on Kaggle [4].

Overfitting is a big problem that should be avoided when training models. Overfitting occurs when the model fits the training data well but cannot make an accurate prediction for the unseen data. That makes the model not fit for real-world use. To prevent this, the data is split into two separate sets, one for training and another for validation and testing. The training set is used to train the model and the validation set is used to

measure the model's performance. The loss and accuracy as seen in the first two columns are measured using the training data. The "val_loss" and "val_accuracy" which stand for validation loss and validation accuracy respectively are calculated from the validation set. In other words, the "loss" and "accuracy" show how well the model fits the training data, while the validation loss and validation accuracy show how well the model fits new data.

Discussion

The purpose of this research is to find a way to make more accurate predictions of the chance of heart attacks in patients. Through the use of machine learning, this goal was achieved. By applying supervised learning and logistic regression techniques, the model was developed. Then, in order to improve the accuracy of the model, training was applied. The final model had an accuracy of 90% meaning one out of ten diagnoses is incorrect. This makes the model fairly accurate. The accuracy can be improved by providing more data on different cases and/or by adding more features. This would allow the model to better respond to a variety of different patient situations.

Conclusion

Machine learning can be used in many different fields including the medical field to help make accurate predictions. The model created can predict 90% accurately if a patient has a high chance of heart attack. The problem of predicting the chance of heart attack was solved using supervised learning where data is labeled and split into two categories, training data and validation data. The model performs well in both the training dataset and validation dataset with an accuracy of 0.818182 ad 0.9061639, respectively when epoch equals 40. Thus, the model should work well when it is applied to any new patients to predict their chance of a heart attack.

Acknowledgments

References

[1] Müller, A. & Guido, S. (2016). Introduction to Machine Learning with Python. O'Reilly Media, Inc.

[2] Burkov, A. (2019). The Hundred-Page Machine Learning Book. Andriy Burkov.

[3] Chowdhury, Muhammad E. H., Alzoubi, K., Khandakar, A., Khallifa, R., Abouhasera, R., Koubaa, S., Ahmed, R., & Hasan, M. (2019) Wearable Real-Time Heart Attack Detection and Warning System to Reduce Road Accidents. Sensors. https://doi.org/10.3390/s19122780

[4] Kaggle. https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset?select=heart.csv