Using KNN to Classify Heart Disease Patients

Kin Wong
Calpine Information Analyst
1st year UCR MSE Student
kwong149@ucr.org

Abstract

In this paper, I present how to use a well know algorithm, K-Nearest Neighbor, to analyze heart disease data. The dataset is gathered from the UCI machine learning repository. The current trend of data science / machine learning is getting more involved in the medical field. This is a good chance to explore medical data through this dataset.

There are many medical datasets I could have picked from the UCI machine learning repository, but this particular dataset has less features than other datasets. It would be interesting to see how much meaningful analysis can be discovered with this limited amount of data.

Keywords

Supervised learning; K-Nearest Neighbors; Support Vector Machine; Grid search

1. Introduction

Machine Learning is often associated with self-driving car, predictive analysis, or other cool technology products. Medical field is also taking a big part in the field of machine learning. I have done some credit card / loan approval predictive analysis, but not medical data analysis. This is a good chance for me to explore how machine learning can be apply to medical field data and discover meaning knowledge.

The dataset can be applied with many different algorithms to train the dataset and build analytical model. Since I am working the project myself, I choose to build a K-Nearest Neighbor algorithm from scratch and compare the result with Sklearn's KNN algorithm, and Sklearn's SVM algorithm.

The dataset has 14 columns, 13 of them are attribute and last one is target variable. The 14 attribute variable contains the following medical conditions and metrics: gender, age, chest pain type, cholesterol level, blood sugar level, resting blood pressure, electrocardiographic results, maximum heart rate, angina condition when exercise, blood vessels condition, and exercise condition. Since these are relatively common metrics that can be obtained by patients' survey or clinical check-ups, implementing the technique in real life is not feasible. This is also the reason why machine learning plays a big part in the medical field.

The goal of this analysis is to find out how accuracy can a KNN machine learning model predict whether a patient have heart disease or not.

If model reach a high accuracy, future patient can utilize the metrics above to know the percentage of his or her have heart disease.

2. Related Work

More and more data are being implemented to the cloud, and this included medical data as well. The first paper I read is "Big Data and Cloud Computing - A Perfect Combination" written by Amit Verma. The title of the paper is intriguing given big data and cloud computing are the most used "buzz words" in the field of information technology. Verma begins by explaining the relationship between the two, and the simple understanding can be looked as the cloud manages the software and big data drives the business decision. There are many different cloud computing services, and the common three services are Infrastructure as a Service (IAAS), Platform as a Service (PAAS), and Software as a Service (SAAS). There are multiple benefits of practicing Big Data in Cloud. The first benefit is improved data analysis. Many companies prefer to perform analysis in the Cloud and have the in-house labor to focus on different issues. For example, a hospital can store their medical data up on the cloud and focus on its own medical issue. The second benefit is simplified infrastructure. The flexibility and scalability of cloudlet the company manage workload easily, and this led to the third point, which is lowering the cost. This would make more sense for hospital because it would be out of place to have a data storage in hospital. Not only the company can pay for what it used for, the company can process large-scale data without having to maintain computing-intensive hardware. The fourth benefit is security. Having the application and data hosted on the cloud is more secure than to have the data store in-house. This is due to the fact that Cloud companies invest a large amount of resources to ensure user's security is up to standard. The last point mentioned in the article is virtualization. Virtualized big data process applications can run seamlessly on the cloud. As applications become more dynamic, virtualization will become more prominent in the field of big data and cloud technology. Although there are many benefits to implementing cloud technology, companies need to take the long term cost and internet connection into account. These are the downside of cloud computing.

Hospital is often regarded as outdated in terms of information technology. It is importance to keep in tread in terms of technology if a company do not want to be left behind. The second article I read is about the 10 big trends in big data written by Cynthia Harvey. The first and most interesting point is open source big data application. Experts say that many enterprises will expand the use of Hadoop and NoSQL to speed data processing. In-memory is a technology that starts to emerge in the big data industry as the price of storage hardware decrease. Data stored in RAM-based such as SSD can read/write data much faster than traditional HDD. The third and fourth point is about machine learning and predictive analytics. As a branch of Artificial Intelligent, Machine Learning has become one of the top 10 strategic technology trends in recent years. ML is the core of many predictive analytics software and predictive analytics has become more and more powerful as ML algorithm gets more accurate. In the case of this project, I am trying

to predict if a patient has disease or not via a machine learning technique. Another way of enterprises using big data is by utilizing pre-built AI applications. With pre-built analytics tools, companies and start to use the application on the go. This would seems to be beneficial for hospital or medical companies the most. Assuming these companies mainly focuses are on medical related issue, they can get quick result from these pre-built AI application. ML can also improve cybersecurity by predict, or prevent attacks. With better security, the ransom attack in hospital could have been avoided. Harvey mentioned with data-gathering devices, the field will experience even faster data growth, which leads to the use of edge computing. Edge computing happens very close to the IoT devices. The IoT device can utilize edge computing to compute analysis on the fly and delete the old data afterward, saving the user or company time and storage. An example for medical field would be wearable device to measure patient's body and analysis on the go.

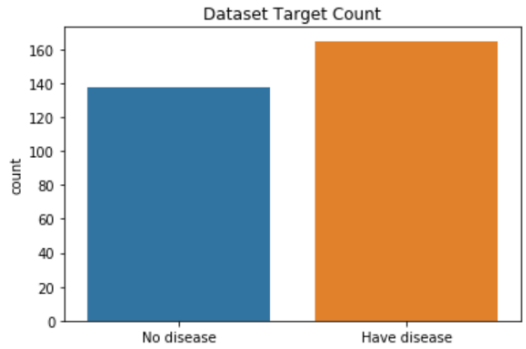
3. Data Exploration and Proposed Method

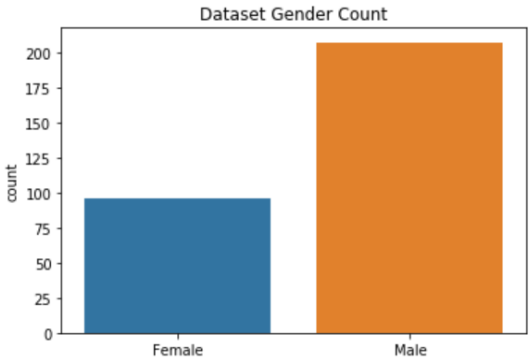
Analyze the dataset, prepare the data, and build model to train it.

3.1 Data Exploration

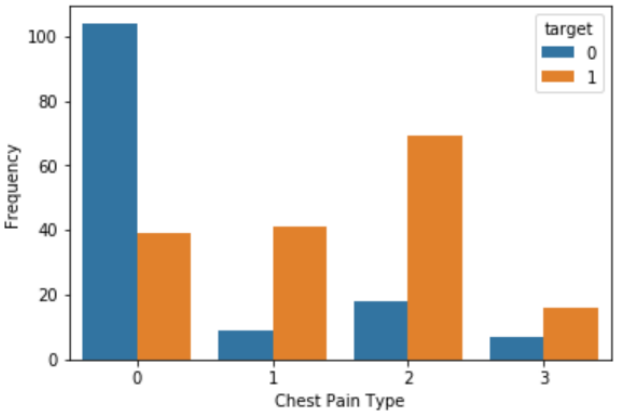
Since the dataset contains both categorical and numerical variables, I stated exploring the dataset via bar graph and cross tab graph. Since the dataset contains both patients with disease and no disease, I plot some histograms in regard to the target variable.

Although the distribution of patient with disease and no disease is relatively even, there are many more males in the dataset.

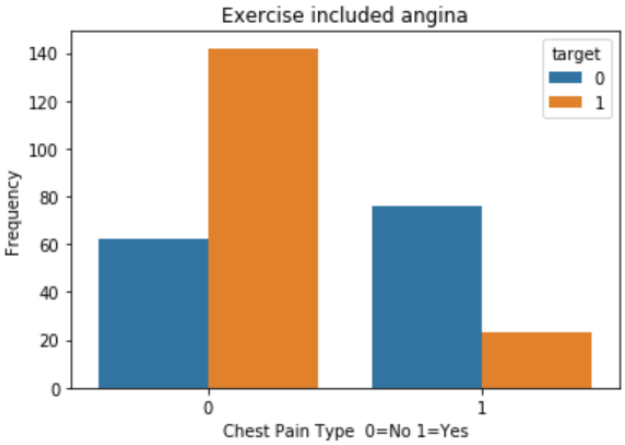




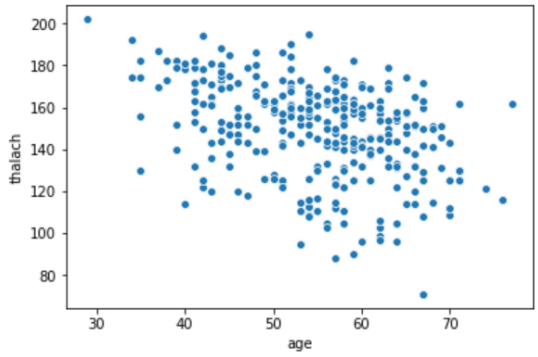
From the chest pain type crosstab plot, it shows the catagorical vairable cp (chest pain type) play a contributing role in determing where a patient can have heart disease. It looks like type 0 is less threathening than type 1 and 2, while there are not enough type 3 data to make a judgment.



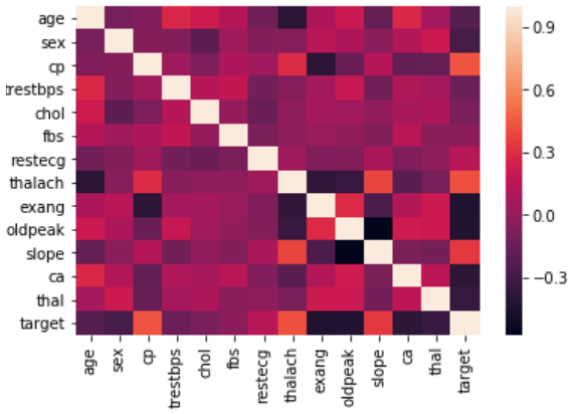
The plot here shows that patients who experience angina (chest pain) when exercising have a lower chance of having heart disease, which is really suprising to me. It also makes me wonder if this variable will have a negative effect on a machine learning model.



I look at the relationship with blood pressure and age, since many elderly people measure blood pressure as a daily routine. It seems like there is a slight negative corrlation in age and blood pressure, but the correlation is not strong.



The fina plot I made is a heatmap to show the relationship between every attributes. There are not many obivous relationship shown in the graph, and it is actually a good thing because I do not have to worry about the issue of linearity.



3.2 Data Preparation

I first drop the target variable from the dataset in order to feed the data for training.

Since the dataset contains multiple categorical variable, I use pandas get dummies function to concatenate all the dummy variables to the data frame. The function is highly convenient when creating dummy variable for multiple columns. I set `drop_first = True` to drop the baseline dummy variable, which decrease the number of columns the model will have to work with. This results in 21 columns excluding the target variable.

Unlike the MNIST dataset where all variables are pixels and equally weight, the numerical data have different units and range in the dataset. This make normalization a necessary procedure. According to Jaitley's article about normalization, "For machine learning, every dataset does not require normalization. It is required only when features have different ranges." This perfectly describe the situation between the MNIST dataset and this heart disease dataset. I followed this simply formula to normalize my whole data frame.

$$x_{new} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

The last step of data preparation is splitting up the data for training and testing. I used sklearn `train_test_split` and use a test size of 25%. Random state was set to 0 to keep the result repeatable. From the first row of training data, we can see all the nurmerical vairable is ranging from 0 to 1.

```
trainX[0]
```

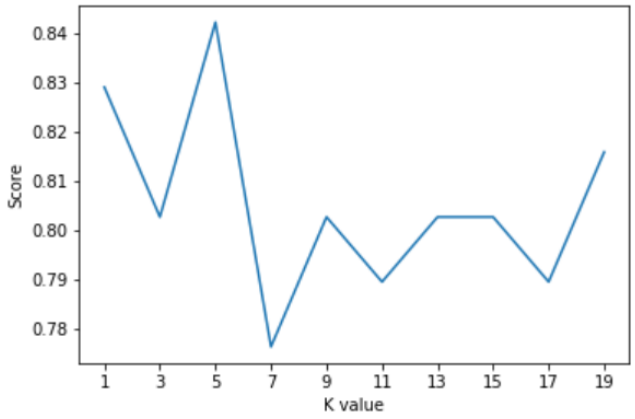
```
array([0.60416667, 1.        , 0.35849057, 0.22374429, 0.        ,
       0.        , 0.77862595, 0.51612903, 0.        , 1.        ,
       0.        , 0.        , 0.        , 1.        , 0.        ,
       1.        , 0.        , 0.        , 0.        , 0.        ,
       1.        ])
```

3.3 KNN Implementation

Although the gist of KNN algorithm is not too complicated, when implementing the algorithm from scratch I start from pseudo code. The pseudo code I came up with is as follow:

1. Have training, testing data, and k loaded.
2. Compare the Euclidean distance of the point and training data.
3. Sort the distance in a list.
4. Pick out the first k points.
5. Store the label of those k points in a list.
6. Get the most frequent label from the list
7. Assign the label to the point.
8. Repeat the same process for the whole test set.

Following the pseudo code, I made a Euclidean distance function and use it in the knn model function. I looped through an odd number list of k. The reason I used odd number is to avoid the situation of having same number of equal neighbors for a point. The result is as follow:



From the plot, k=5 produce the highest accuracy. Howver, this is a small dataset, so the random assignment of data in the train and test splitting process can affect the result.

To verify my KNN function is legitament, I run the same train and test data with sklearn knn function and got the same result.

```
# k=3 have the highest accuracy in both knn model, the accuracy is a
print('The k=5 has the highest accuracy of %f'% max(my_knn_result) )
print('Accuracy score of k=5 from the 2 lists are equal is', equal )
```

```
The k=5 has the highest accuracy of 0.842105
Accuracy score of k=5 from the 2 lists are equal is True
```

4. Evaluation with Cross Validation and Grid Search

As mentioned above, the optimal k value cannot be determined by the accuracy findings obtained from this model because of variability definition. To avoid overfitting I used the grid search function from sklearn. The optimal k value I got with 10-fold cross validation is 7. This contradicts the accurate vs k graph shown above, which show k=7 has a low accuracy.

```
GridSearchCV(cv=10, error_score='raise-deprecating',
       estimator=KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
           metric_params=None, n_jobs=None, n_neighbors=5, p=2,
           weights='uniform'),
       fit_params=None, iid='warn', n_jobs=None,
       param_grid={'n_neighbors': array([ 1,  3,  5,  7,  9, 11, 13, 15, 17, 19])},
       pre_dispatch='2*n_jobs', refit=True, return_train_score='warn',
       scoring=None, verbose=0)
```

```
gs_model.best_params_
```

```
{'n_neighbors': 7}
```

From the grid search process, I learned the importance of avoid overfitting and cross validation is needed to find the optimal parameter.

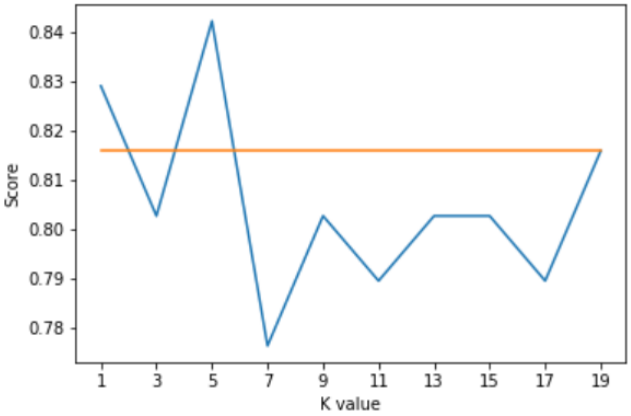
4.1 Comparison with SVM

I chose Support Vector Machine (SVM) as the method to be compared with since SVM one of the best methods to classify binary target variable. I used the same train and test set and got a score of 0.816.

```
svm = SVC(random_state = 0)
svm.fit(trainX, trainy)
score = svm.score(testX, testy)
score
```

```
Out[68]: 0.8157894736842105
```

When plot against the accuracy graph in knn, the result from SVM is almost in the middle. This shows the importance of choosing a optimal k value when building model.



5. Conclusion

KNN has a relatively straightforward algorithm but choosing k-value is a tricky part of the implementation. For example, I was surprised to see the optimal k value produced from cross-validation grid search is lowest accuracy for this train and test set.

On the other hand, SVM does not have much parameter to input when training a model, which can result in a more constituency output.

6. Reference

[1] Caner Dabakoglu. 2019. Heart Disease - Classifications (Machine Learning). (August 2019). Retrieved December 9, 2019

[2] Stephanie. 2018. Normalized Data / Normalization. (January 2018). Retrieved December 9, 2019 from https://www.statisticshowto.datasciencecentral.com/normalized/

[3] Urvashi Jaitley. 2019. Why Data Normalization is necessary for Machine Learning models. (April 2019). Retrieved December 9, 2019 from https://medium.com/@urvashilluniya/why-data-normalization-is-necessary-for-machine-learning-models-681b65a05029.

[4] Vineet Maheshwari. 2019. KNN ALGORITHM AND IMPLEMENTATION FROM SCRATCH. (January 2019). Retrieved December 9, 2019 from https://medium.com/datadriveninvestor/knn-algorithm-and-implementation-from-scratch-b9f9b739c28f

[5] Harvey, C. (2018, January 24). Big Data Trends. Retrieved from https://www.datamation.com/big-data/big-data-trends.html.

[6] Verma, A. (2018, July 21). Big Data and Cloud Computing – A Perfect Combination. Retrieved from https://www.whizlabs.com/blog/big-data-and-cloud-computing/.