**Assignment for Week 2 – KNN**

Given the heart disease dataset, which has 14 variables (13 input variables and 1 output variable)

**Description of input variables:**

1. age: Age of the patient in years.
2. sex: Sex of the patient (1 = male, 0 = female).
3. cp: Chest pain type (1 = typical angina, 2 = atypical angina, 3 = non-anginal pain, 4 = asymptomatic).
4. trestbps: Resting blood pressure (in mm Hg).
5. chol: Serum cholesterol level (in mg/dl).
6. cigs: Number of cigarettes smoked per day.
7. years: Number of years of smoking.
8. fbs: Fasting blood sugar (1 = fasting blood sugar > 120 mg/dl, 0 = otherwise).
9. famhist: Family history of heart disease (1 = yes, 0 = no).
10. restecg: Resting electrocardiographic results (0 = normal, 1 = having ST-T wave abnormality, 2 = showing probable/definite left ventricular hypertrophy).
11. thalach: Maximum heart rate achieved.
12. exang: Exercise-induced angina (1 = yes, 0 = no).
13. thal: Thalassemia (3 = normal, 6 = fixed defect, 7 = reversible defect).

**Target Variable:**

1. num: Target variable (0 = no heart disease, 1,2,3,4 are levels of heart disease ).

**Objectives:** Predict heart disease in patients.

1. Get to know your data, start by data exploration. Summarized your finding.
2. Divide the data into training set and test set randomly with ratio 80:20. Make prediction based on 1-nearest neighbor. What is the error rate of this approach? Report your results in a confusion matrix.
3. Use different values for K, what is the optimal value of K from your experiments? Report the error rate of the optimal K value and its confusion matrix. Is there any improvement (by how much) over 1-nearest neighbor?
4. Is there anything else you can do to improve your model? If yes, demonstrate your approach. (Hint: there is always something that you can try, unless your accuracy score is 100%)

**Solution:**

1. Task 1: Get to know your data and start by data exploration. Summarize your findings.

Approach:

* Load all the required libraries and data set
* Look at the data
* Analyze the data by checking the data type, statistics, no of categorical and numerical features
* Univariate analysis to understand the distribution (histograms, barplots and box plots) of input features
* Bivariate analysis to understand the correlations of the features

Summary:

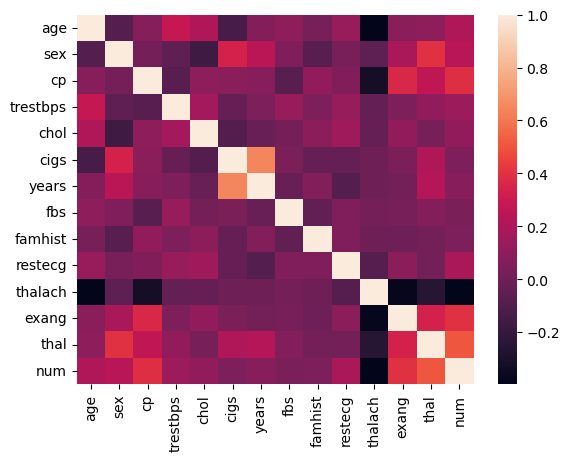
* The Given data set has 14 features, our targert feature is 'num' and the other features can be considered as input features
* The heart disease data set has 282 datapoints without any missing values
* Based on statistical analysis:
  + the people with age group of 29 to 77
  + trestbps in the range of 126 to 170
  + no of cigs range from 0 to 75
  + years of smoking 0 to 54
  + thalach range from 88 to 202
* Based on the number of unique values in each feacture
  + Numerical Features: are age, trestbps, chol, cigs, years, thalach
  + Categorical Features: sex, cp, fbs, famhist, restecg, exang, thal, num
* based on the univariate analysis : the features like chol, cigs and years are not normally distributed and they have skewness. features such as trestbps, chol and cigs have skewness and outliers

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

* categorical variables are not equally distributed particularly fbps, restcg, thal

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  | |

* Based on correlation matrix, No features are majorly correlated and hence these can not be removed

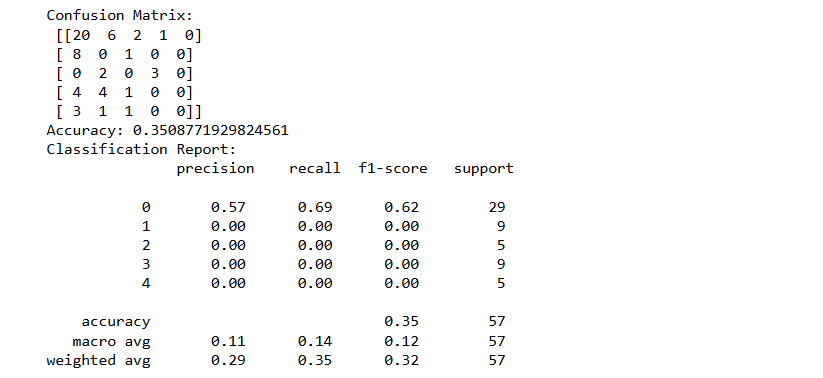


1. Task 2: Divide the data into training set and test set randomly with ratio 80:20. Make prediction based on 1-nearest neighbor. What is the error rate of this approach? Report your results in a confusion matrix.

Approach:

* Segregated the target column ‘num’ as y and remaining all input features as ‘X’
* Split the data set in to training and test data set (80, 20% split)
* Developed a knn classifier using that gives the prediction based on single nearest neighbour

Summary and Results:

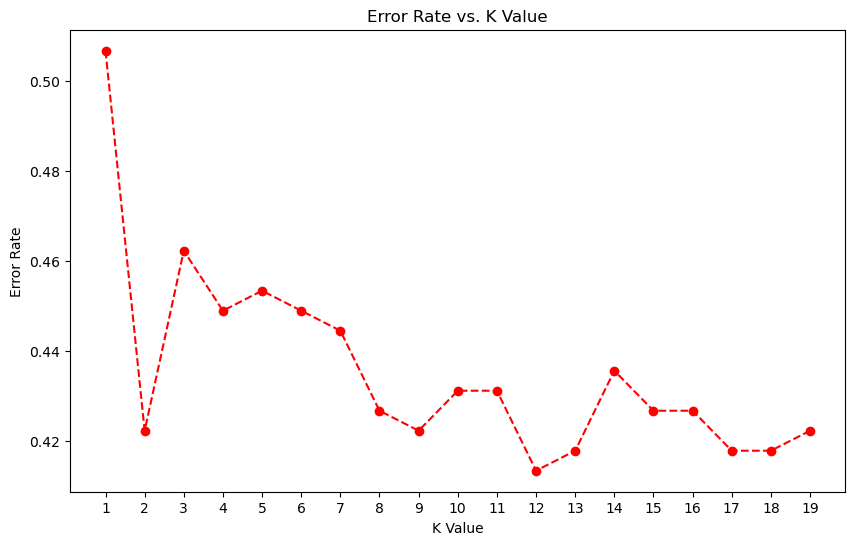


* The knn classifier with 1 nearest neighbour doesn’t give the best accuracy score of 0.35 as it is not a good model and need to be improved. We will be increasing the k-value

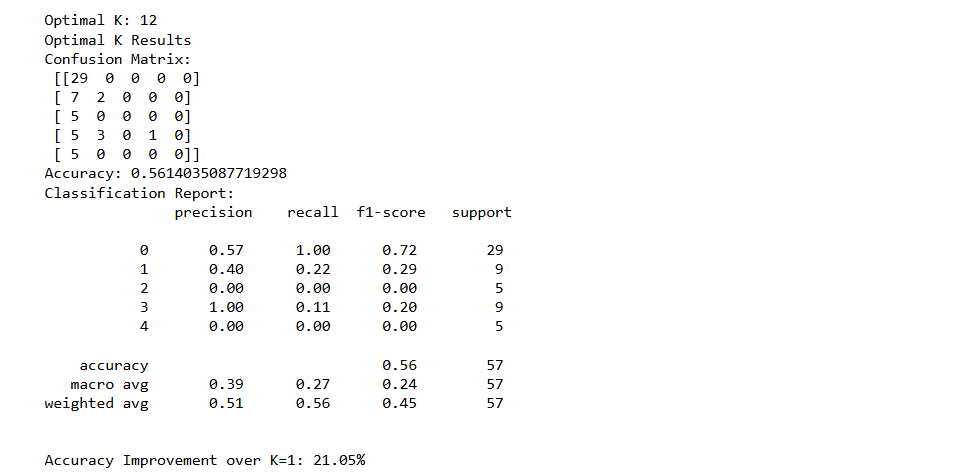
1. Task 3: Use different values for K, what is the optimal value of K from your experiments? Report the error rate of the optimal K value and its confusion matrix. Is there any improvement (by how much) over 1-nearest neighbor?

Approach:

* + Iteratively increased the k value from 1 to 20 and obseving the error rate.
  + A plot to understand the trend of error rate w.r.t k values was plotted
  + The optimal ‘k’ value is selected bsaed on the point at which the error rate is low



Based on the trend the point with lowest error rate is at k = 12, and thus a knn classifier model with k = 12 is developed and is used to predict its performance on the test data.

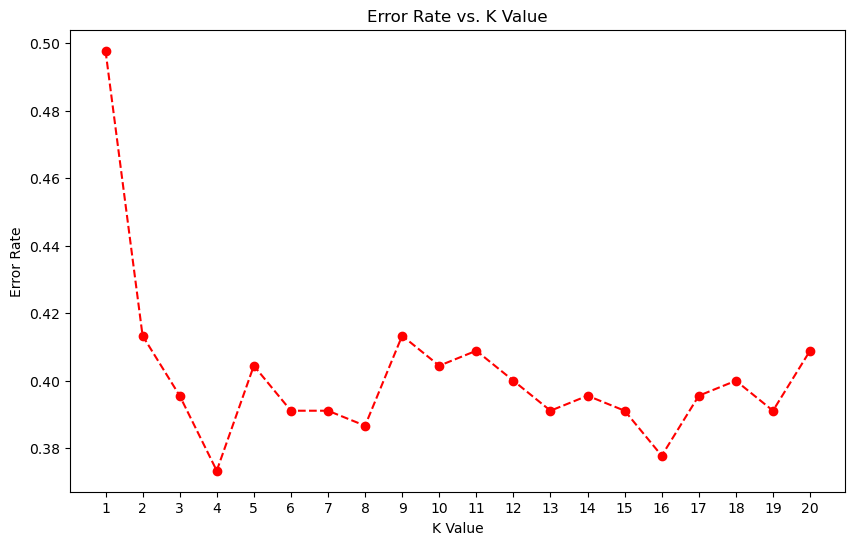


* + The accuracy is improved from 0.35 to 0.561. clearly, the accuracy improved by increasing the value of ‘k’
  + The accuracy is improved almost by 21%

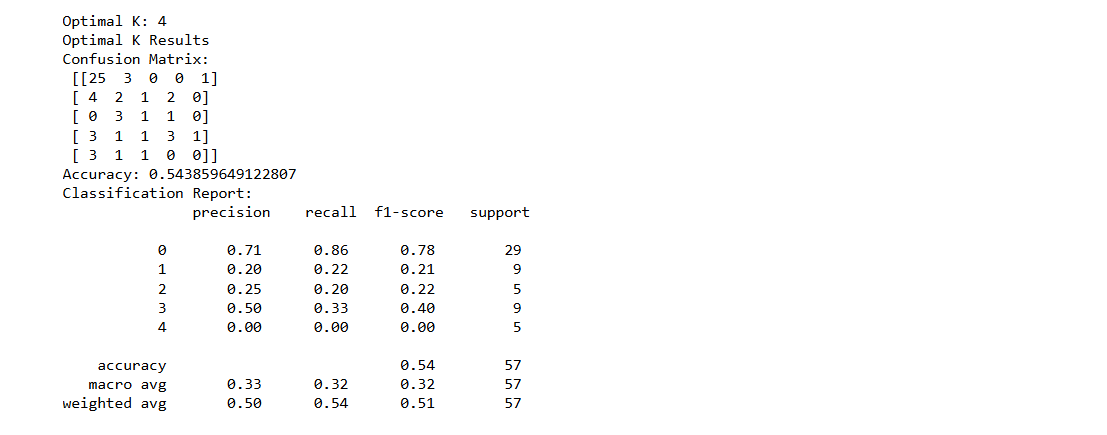
1. Task 4: Is there anything else you can do to improve your model? If yes, demonstrate your approach. (Hint: there is always something that you can try, unless your accuracy score is 100%)

Approach:

* We checked the effect of normalizing the data using a ‘standard scaler’ on performance of model.



* The optimal value of k changed from 12 to 4, but accuracy reduced from the 0.56 to 0.54.



We can try other techniques to improve the models performance

* Do feature engineering and dimentionality reduction using PCA
* Use different distance metrics such as eucledian, manhanttan etc
* Using weighted knn techniques
* Using some other classification algorithms like logistic regresison, SVM classifier etc. also we ca try ensemble Classification algorithms such as random forest, XGBoost and Neural network.