

BBM407 - FUZZY LOGIC

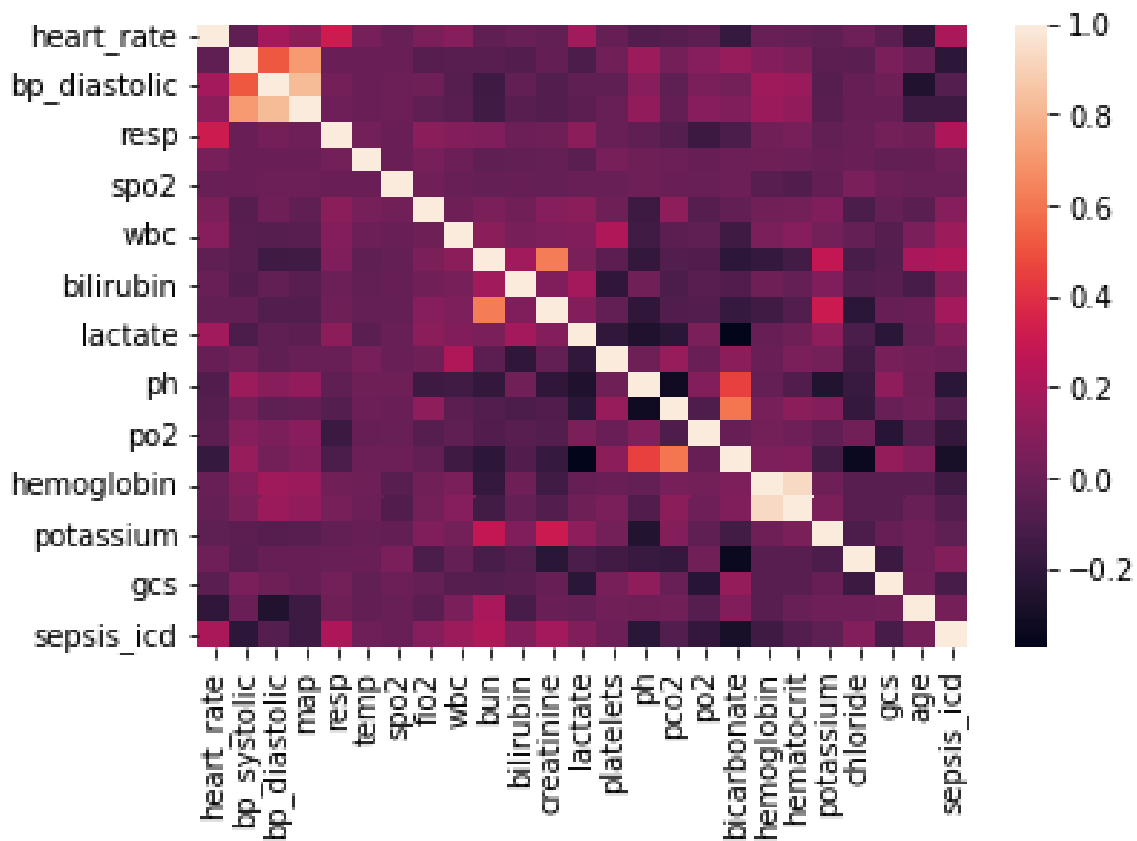
FINAL PROJECT REPORT

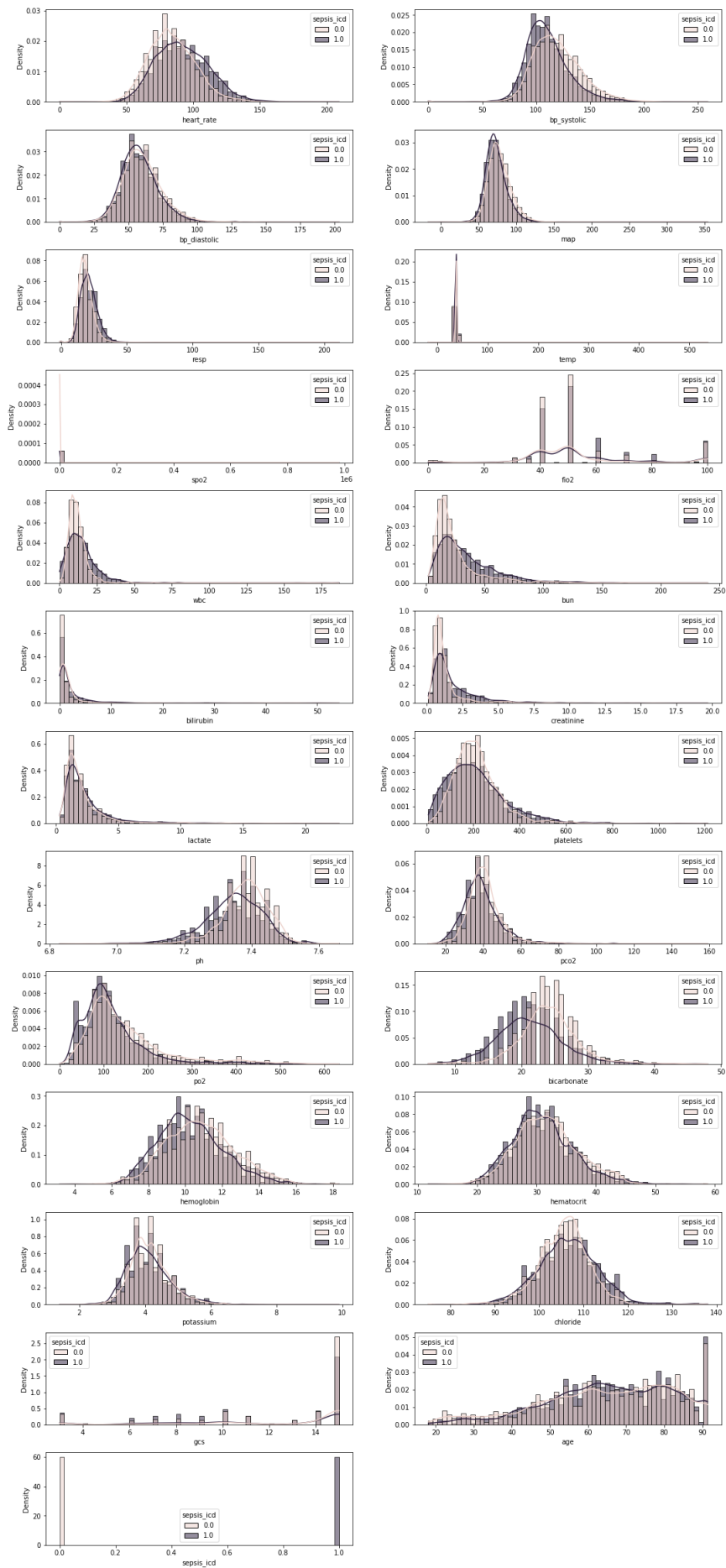
FEYZA ÖZKAN – 2200356815

FINAL PROJECT REPORT

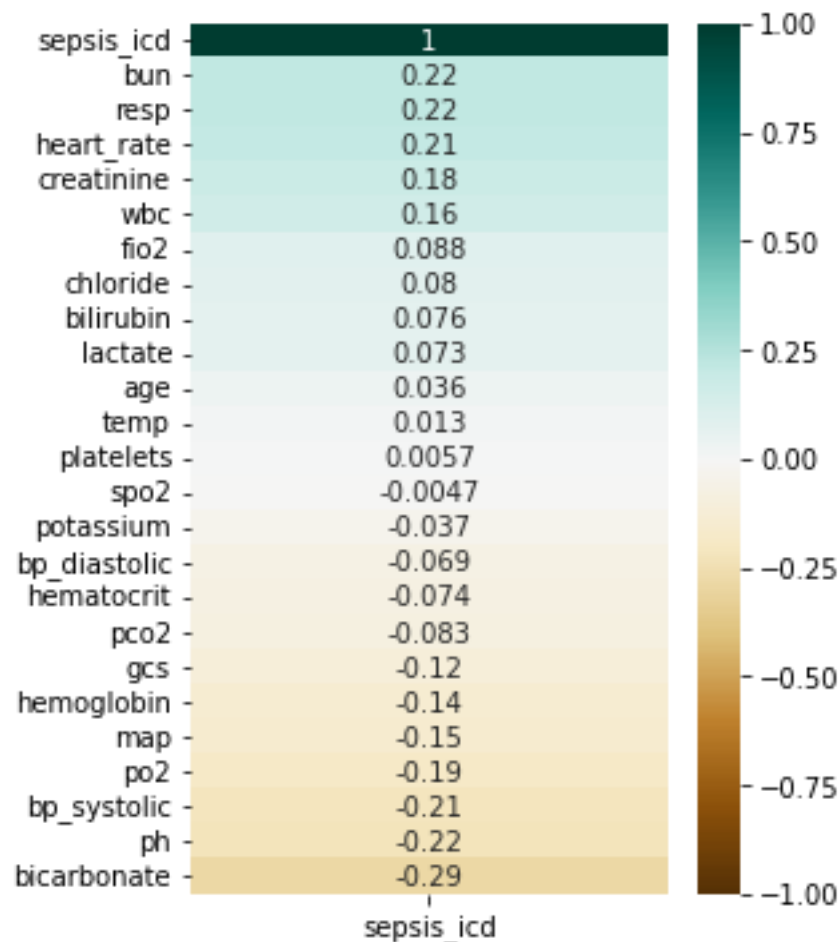
FEYZA ÖZKAN – 2200356815

This part was done in the dataUnderstindg.py file and the codes are the same with some parts of midterm project.





Features Correlating with sepsis_icd



According to table above , I have used bun,resp, heart_rate , creatinine , lactate, bicarbonate as an inputs and sepsis_icd as an output.

After deciding most correlating features, i have created new dataset by selecting instances from dataset "combined.csv", and I put those instances to "selected_data.csv" file. After this step , I have used "selected_data.csv" file in my project.

"selected_data.csv" file;

- shape is (860,7)
- 413 instances have sepsis_icd = 0
- 447 instances have sepsis_icd = 1

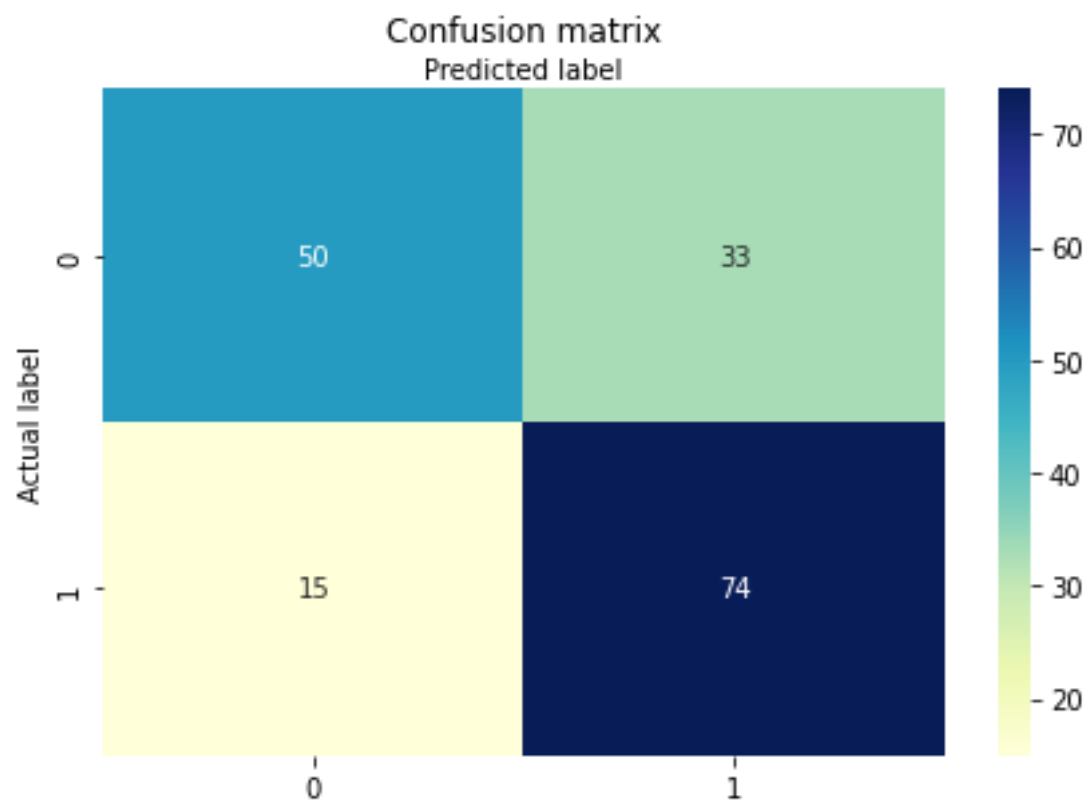
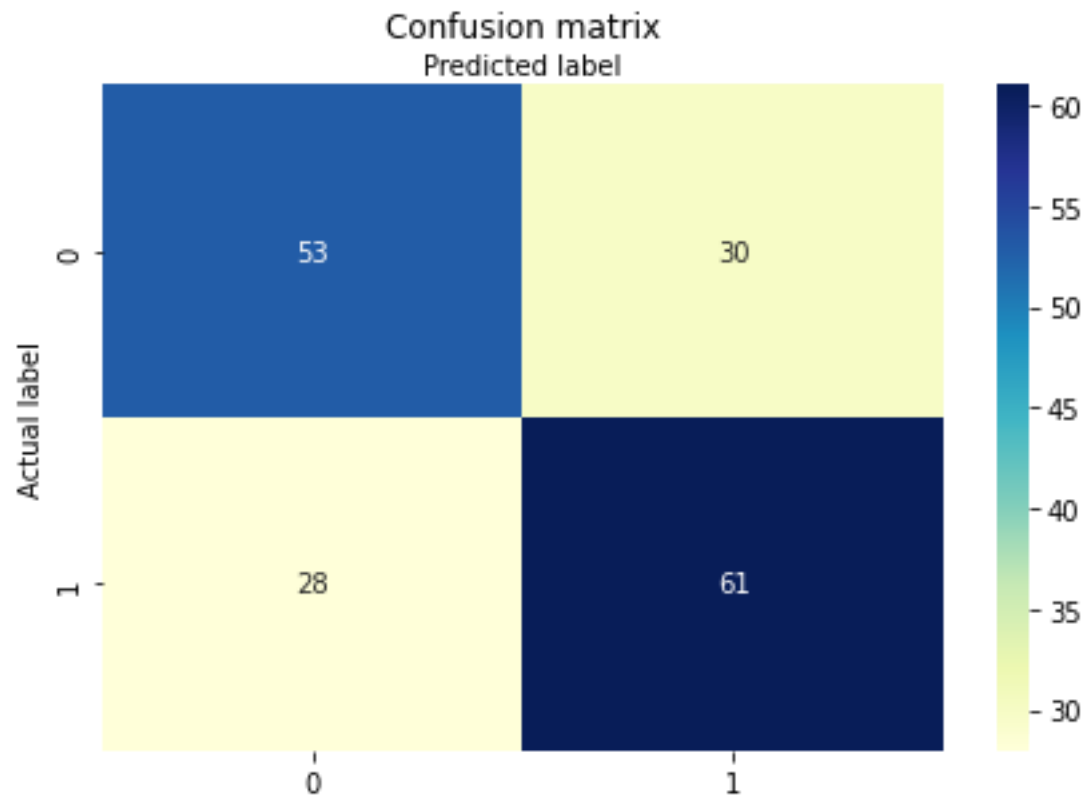
EVALUATION MODEL PERFORMANCE :

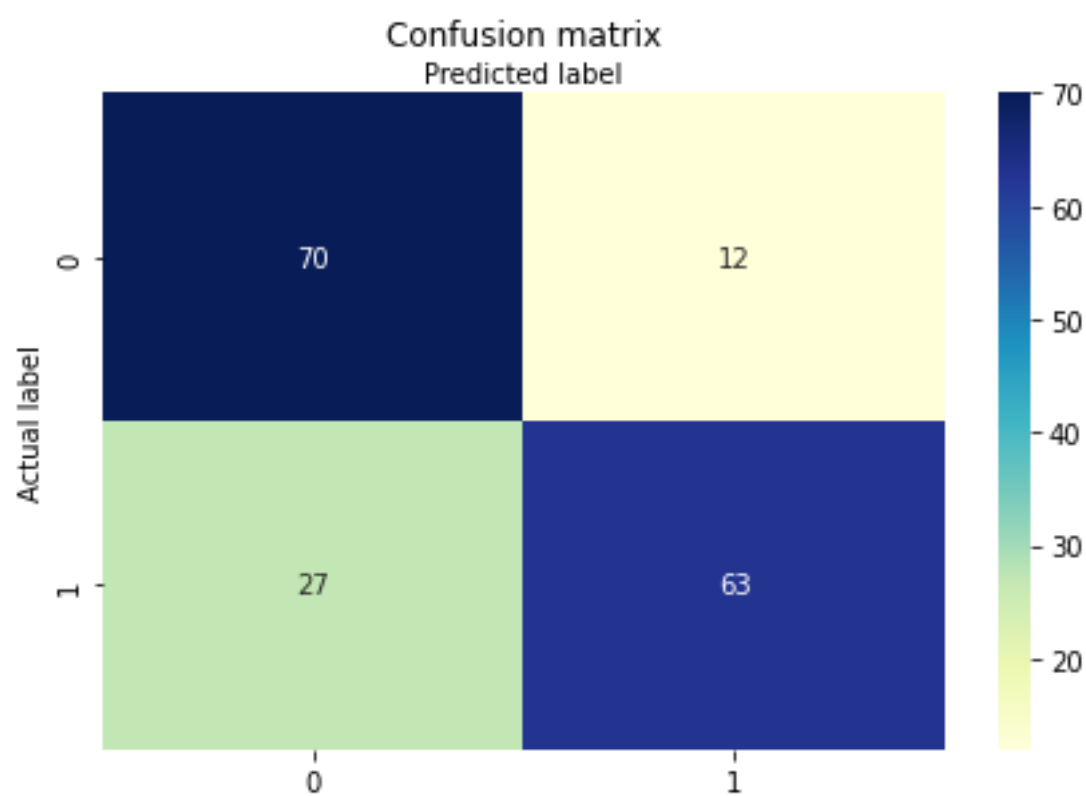
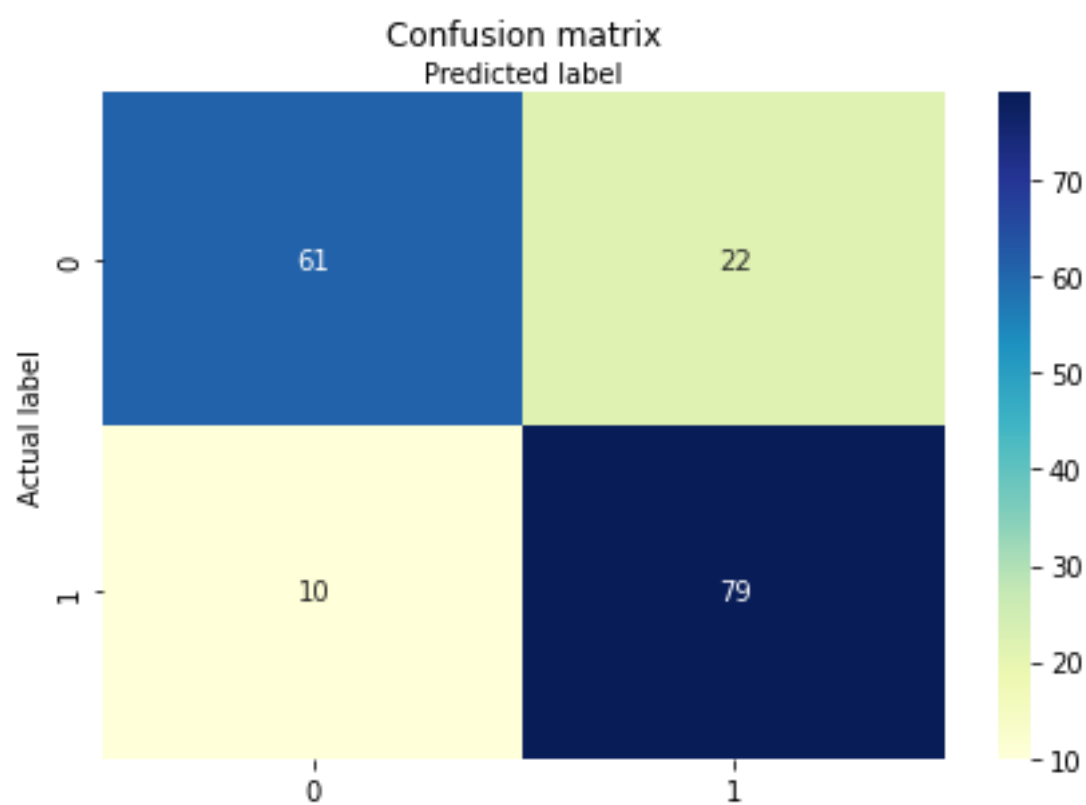
This part was done in ANN.py file. I have generated my ann model with the helps of [1] and [2].

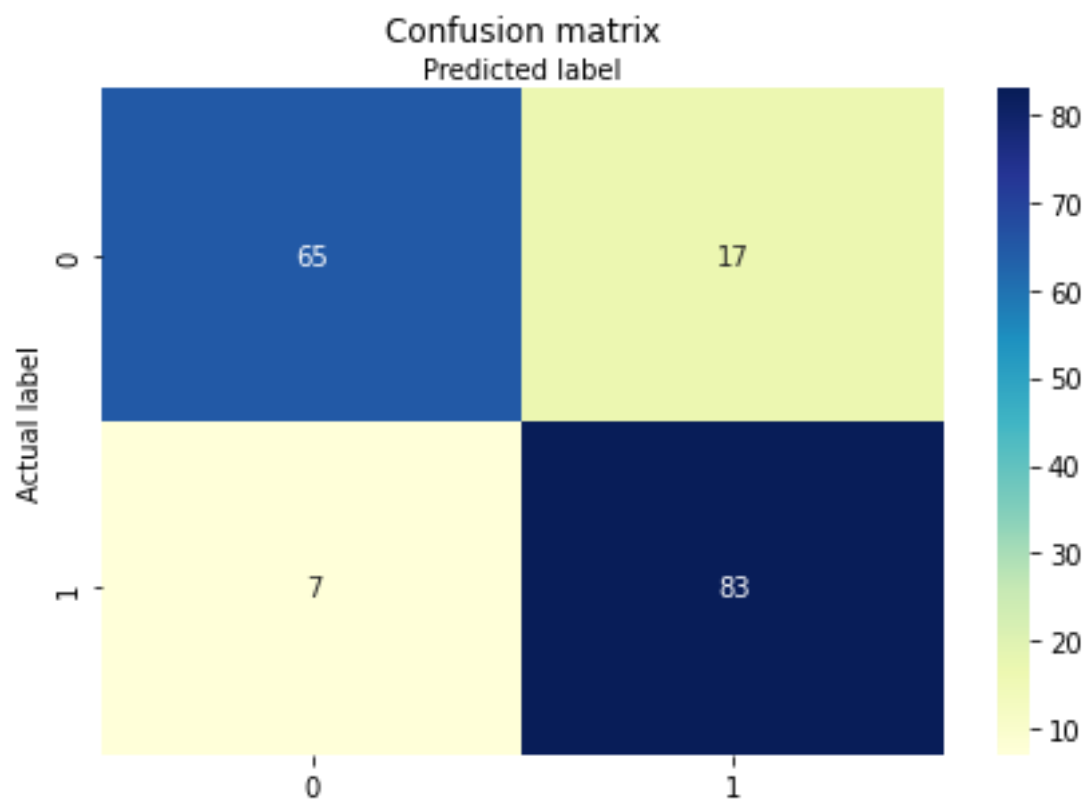
The size of training data is 688.

The size of testing data is 172.

CONFUSION MATRICES FOR 5 FOLDS







SCORES

1: 0.67	acc: 0.66	pre: 0.67	r: 0.68
2: 0.75	acc: 0.72	pre: 0.69	r: 0.83
3: 0.83	acc: 0.81	pre: 0.78	r: 0.88
4: 0.76	acc: 0.77	pre: 0.84	r: 0.7
5: 0.87	acc: 0.86	pre: 0.83	r: 0.92

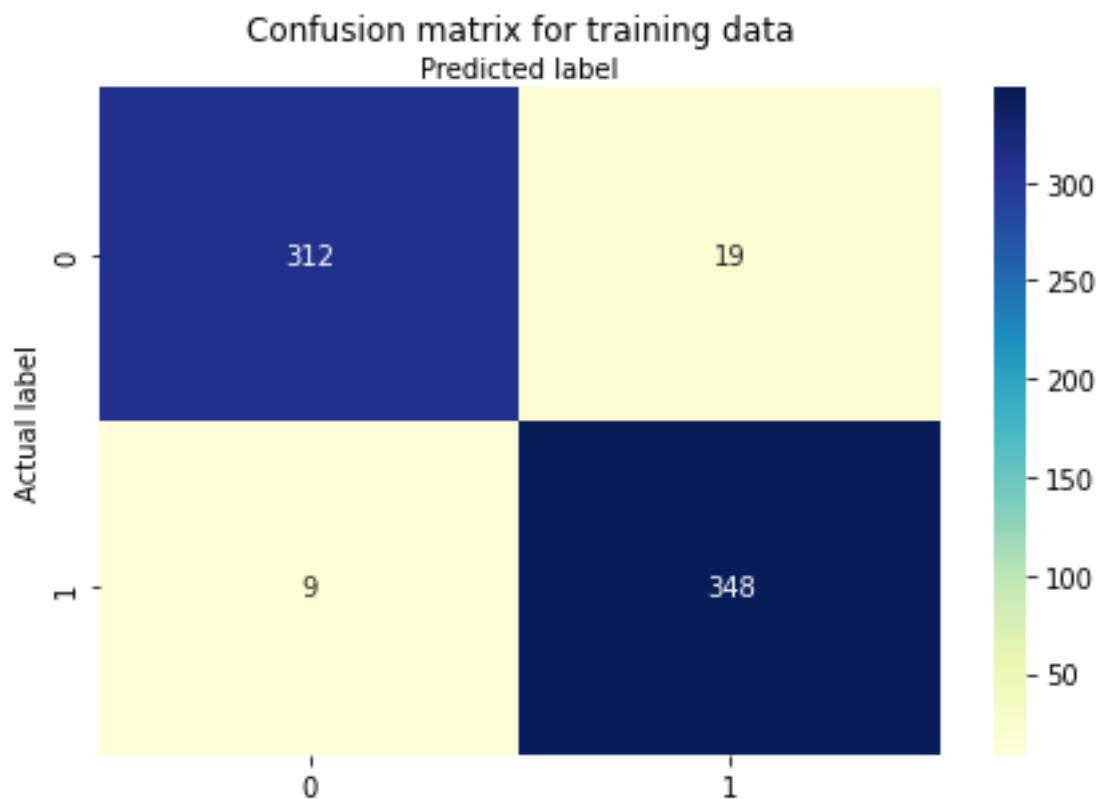
and average values

p average:	0.7628193346396362
r average:	0.8053433208489389
f1 average:	0.780355868025041
acc average:	0.7662790697674419

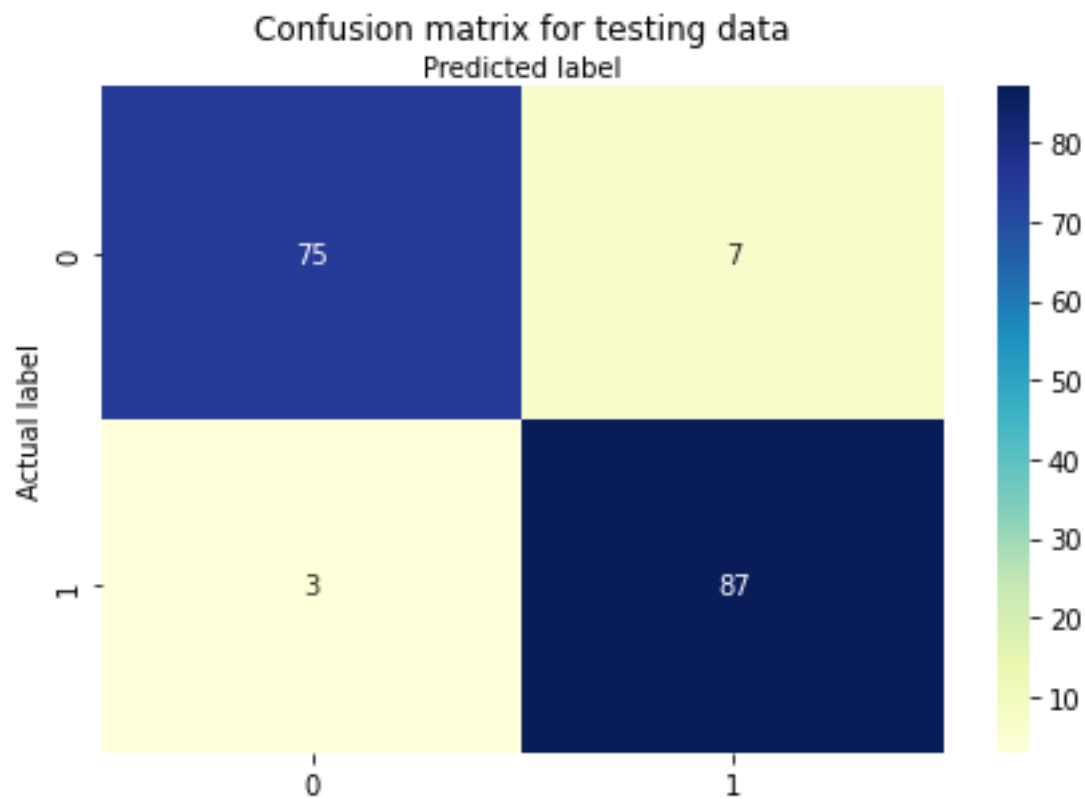
FINAL EVALUATION:

Using ANFIS training methods(used 3 membership for each membership functions) , I got new predicted values for sepsis_icd([4],[5]). I put actual and predicted y values into csv file("y_actual_pred.csv"). Then in finalEval.py file , I have created new dataframe from csv file and I have created new column named predicted_ that consists of 0s and 1s (0s for predicted values less than 0.5 and 1s for predicted values greater or equal 0.5).

Finally , I have created confusion matrices for both training and testing data. I've evaluated the f1 ,p, r, acc scores.



```
f1: 0.9613259668508287
acc: 0.9593023255813954
p: 0.9482288828337875
r: 0.9747899159663865
```



```
f1: 0.9456521739130436  
acc: 0.9418604651162791  
p: 0.925531914893617  
r: 0.9666666666666667
```

As you see, anfis model has good performance on both training and testing data. Also, it's scores are better than my ann model.

REFERENCES

- [1]<https://www.analyticsvidhya.com/blog/2021/10/implementing-artificial-neural-networkclassification-in-python-from-scratch/>
- [2]<https://github.com/bayysp/PredictiveModelANN>
- [3]<https://www.datacamp.com/tutorial/understanding-logistic-regression-python>
- [4]<https://www.youtube.com/watch?v=cBSDIHJ-RYM>
- [5]<https://www.mathworks.com/help/fuzzy/neuro-adaptive-learning-and-anfis.html>