Summary experiment

Taliya Shitreet 314855099

Renana Rimon 207616830

Classify Head Stroke

According to the World Health Organization (WHO) stroke is the 2nd leading cause of death globally, responsible for approximately 11% of total deaths. Our goal is to predict whether a patient is likely to get stroke based on the input parameters like gender, age, various diseases, and smoking status. Each row in the data provides relevant information about the patient.

Data set:

	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	Male	67.0	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
1	Male	80.0	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
2	Female	49.0	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
3	Female	79.0	1	0	Yes	Self-employed	Rural	174.12	24.0	never smoked	1
4	Male	81.0	0	0	Yes	Private	Urban	186.21	29.0	formerly smoked	1

Preprocessing

Data shape: (4981, 11)

11 columns: 10 features + 1 label.

4981 rows: 248 of class 1 and 4733 of class 0.

Due to the distribution of the 1 in relation to the 0, the model may learn biasedly, and even classify everything to 0.

In that’s reason during our preprocessing we added 1's examples to the label so it's divided like 4733 of class 1 and 4733 of class 0. Therefore, the data shape now is (9466, 11).

In addition we changed the data to numeric, and normalize it with standard scaling.

Modeling

We decided to split the data into 80% train, 20% test and split the train into validation and train data.

First part: Logistic Regression results:

We trained the data for 300 epochs and split it into batches of 300 rows size each.

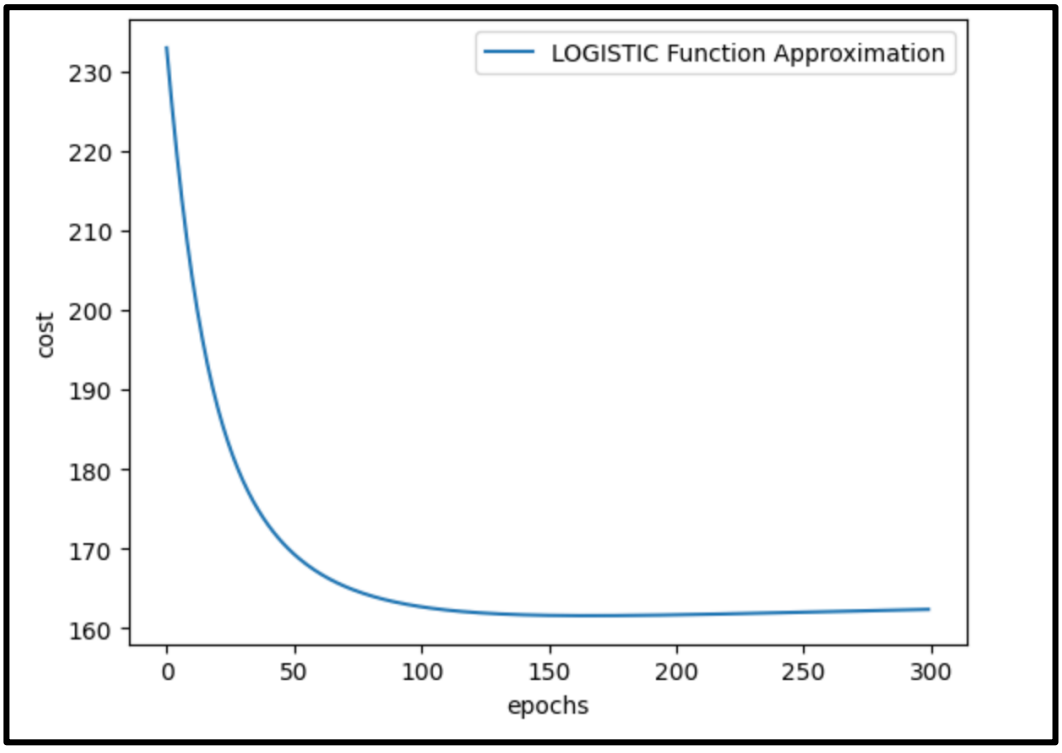
Functions:

- prediction: sigmoid
- cost: cross entropy
- update: gradient descent

Loss results:

We can see the decrease in cost in each epoch.

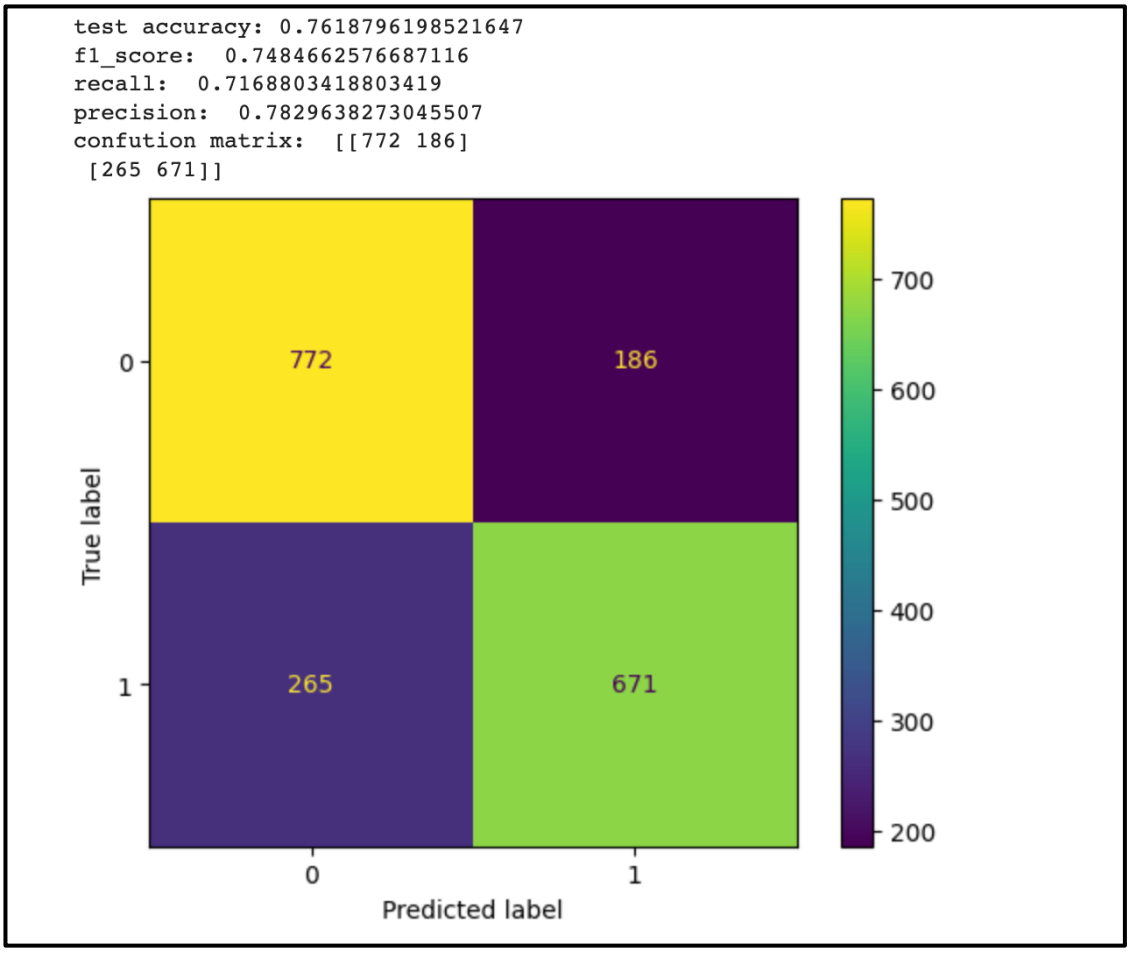
```
epoch 0, cost = 232.928
epoch 100, cost = 162.698
epoch 200, cost = 161.69
```



Accuracy results:

When we calculated the accuracy for the first time, we got 95% of accuracy, but F1-Score of 0.1% - very low! So, we realized that our model misses the prediction of almost most class 1 – this is the reason we decided to duplicate the 1's in the labeling column, as explained above.

Accuracy results after arranging the data:



Although the accuracy decreased (75%), the f-score, precision and recall increased significantly (74%)! and thus we know that most of the patients who will have a stroke will receive a positive answer and most patients who are not will receive a negative answer and that is the most important thing.

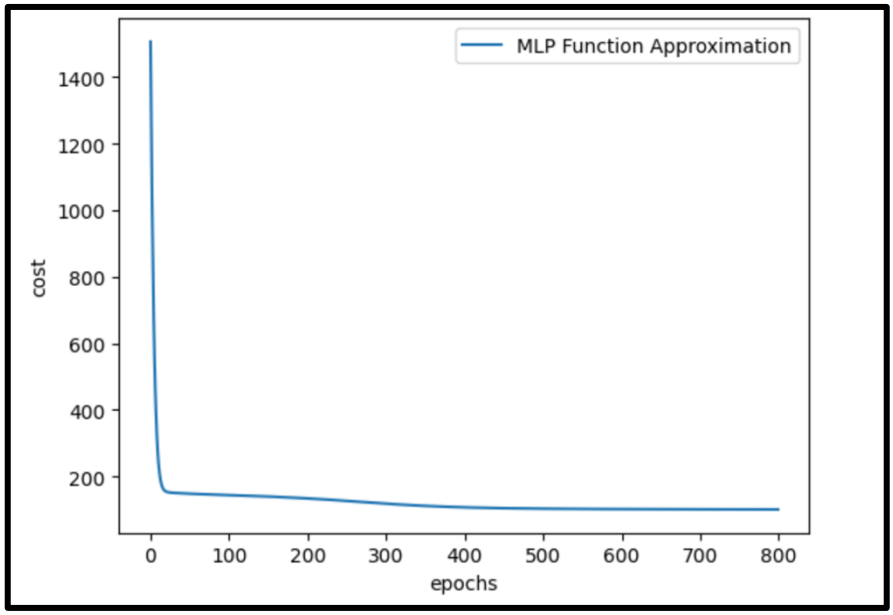
Second part – MLP results

First, we chose 10 neurons for one hidden layer and run for 800 epochs, on each epoch split the data into baches - 300 row each time.

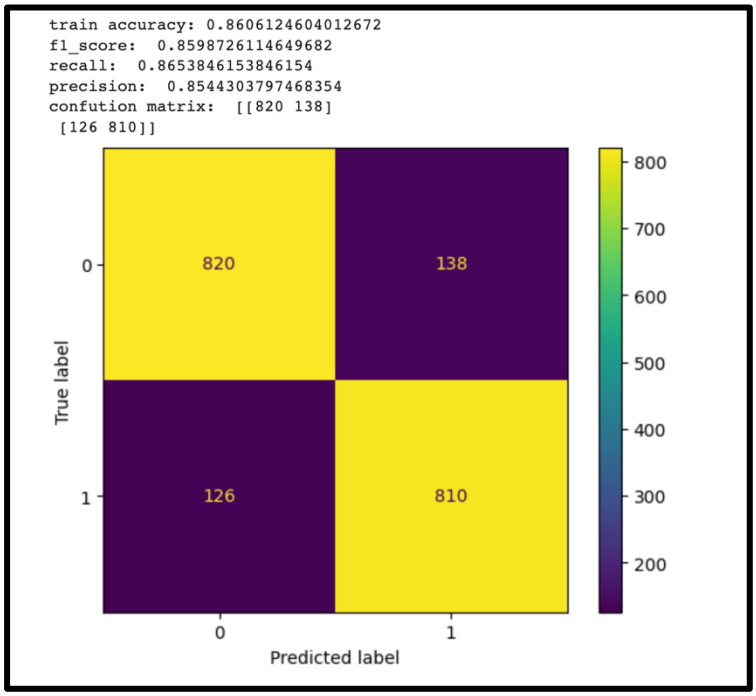
results: We can see decrease of the cost in compere to Logistic running.

161 vs 101 with one hidden layer.

```
epoch 0, cost = 1506.77
epoch 100, cost = 144.076
epoch 200, cost = 134.121
epoch 300, cost = 118.354
epoch 400, cost = 107.278
epoch 500, cost = 103.383
epoch 600, cost = 101.99
epoch 700, cost = 101.336
```




```
train accuracy: 0.8606124604012672
f1_score:  0.8598726114649682
recall:  0.8653846153846154
precision:  0.8544303797468354
confution matrix:  [[820 138]
 [126 810]]
```



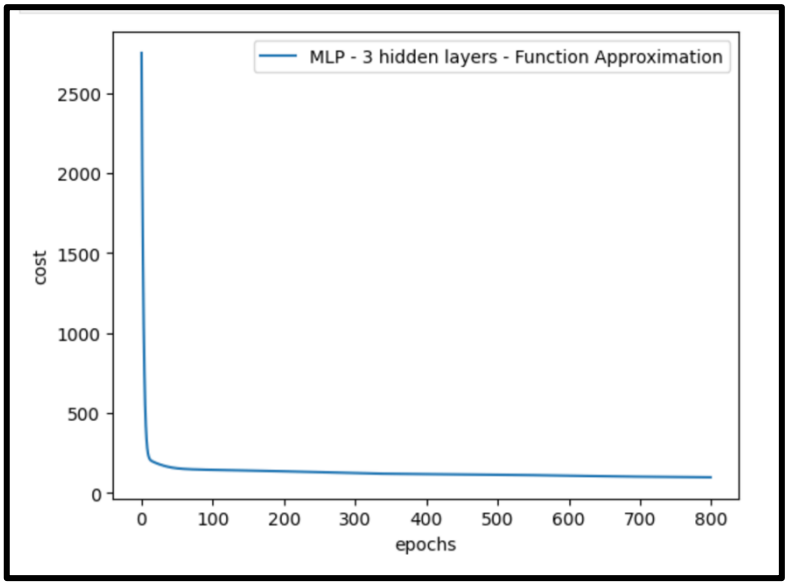
The results we got for this model are much better! There was a ten percent improvement!

Second, we chose to train the data with 3 different hidden layers-

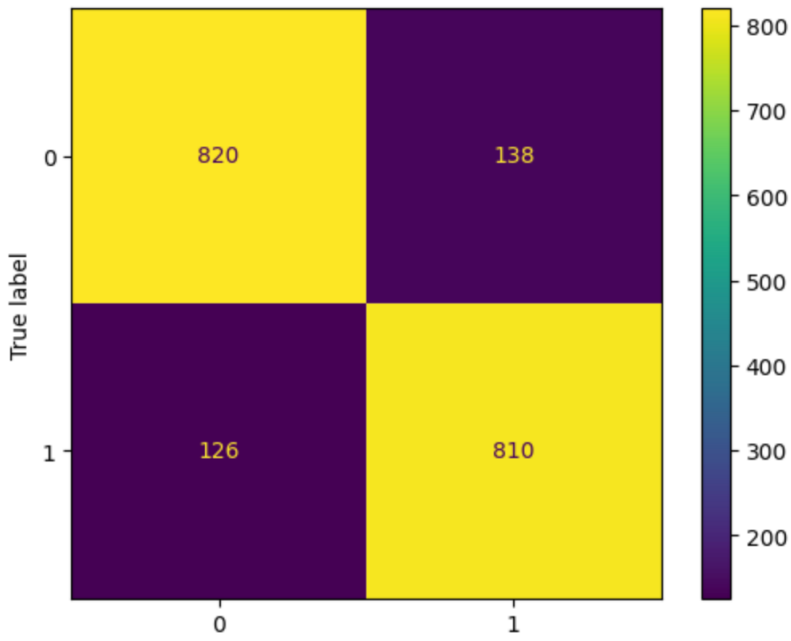
All consist of 10 neurons.

Results: we can say that there is no need to complicate our first MLP model because we get the same results by adding another hidden layer.

```
epoch 0, cost = 2751.95
epoch 100, cost = 143.35
epoch 200, cost = 134.441
epoch 300, cost = 122.551
epoch 400, cost = 116.963
epoch 500, cost = 112.994
epoch 600, cost = 107.081
epoch 700, cost = 100.93
```




```
train accuracy: 0.8606124604012672
f1_score:  0.8598726114649682
recall:  0.8653846153846154
precision:  0.8544303797468354
confusion matrix:  [[820 138]
 [126 810]]
```



In conclusion, the best model for our data is: MLP – logistic regression with one hidden layer. Although 3 layers gave the same results, we prefer as few layers as possible, to save place and time.