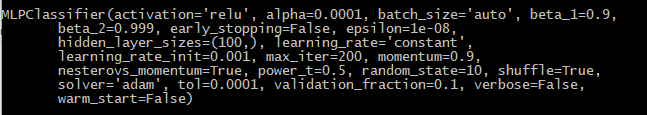
**Task 4. Predictive Modeling Using Neural Networks (5.5 marks)**

***1. Build a Neural Network model using the default setting. Answer the following:***

a. What are the parameters used? Explain your decision. What is the optimal network architecture?

First model was built on default parameters. **Such as**

b. How many iterations are needed to train this network?

Since the model was built and tested on the default **MLPClassifier** settings, ***200 epochs (iterations)*** was used to train this network

c. Do you see any sign of over-fitting?

There is a **major overfitting** issue in this model. As seen in the picture below, there is a significant accuracy difference of around 0.22. When the train accuracy gives distinct difference compared to the test accuracy, it is most likely saying, the learning algorithm has failed to generalise on the data inputs and has found relationships between input and target variable, outputting high training accuracy, however, when it came to test data (which does not display its target values), it will perform poorly.



d. Did the training process converge and resulted in the best model?

As this is the first model produced with default hyperparameters, we cannot conclude this is the best model. However, from what we can see from the Train / Test accuracy values, this model is poorly performing, due to overfitting.

e. What is classification accuracy on training and test datasets?

The classification accuracy on training and test datasets are;

**+** **Train accuracy: 0.8808219178082192**

**+ Test Accuracy: 0.6624203821656051**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| **0** | 0.54 | 0.50 | 0.52 | 115 |
| **1** | 0.72 | 0.76 | 0.74 | 199 |
| **avg / total** | 0.66 | 0.66 | 0.66 | 314 |

***2. Refine this network by tuning it with GridSearchCV. Report the trained model, same as Task 4.1***

a. What are the parameters used? Explain your decision. What is the optimal network architecture?

Using the GridSearchCV library, **“max\_iter(max iterations or epoch)”**, **“hidden\_layer\_sizes (number of neurons in each hidden layer)”** and **“alpha (regularsation parameter)”** was tuned to optimse the neural network’s model.

Its optimal values were listed below (also shown in photo below)

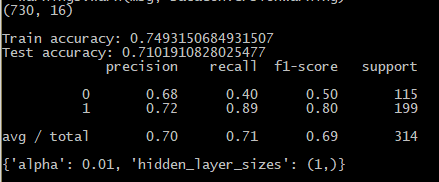
**+ max\_iter = 300+**

**+ hidden\_layer\_sizes = 1**

**+ alpha = 0.01**

It is analysed that, less complex models (smaller feature sets) tend to generalise on this dataset, therefore, hidden layer sizes were chosen as 1, which it was preformed best at.

‘Alpha’ is a learning rate for gradient descent algorithm (process of going back and forward and modifying the weights). Larger the alpha, bigger steps and faster it trains and vice versa. GridSearhCV has returned 0.01 as the optimal value for ‘alpha’



b. How many iterations are needed to train this network?

As shown in the picture above, approximately **300 epochs** (iterations) are required to optimally train this network.

c. Do you see any sign of over-fitting?

As seen in the picture below, there is a slight sign of over-fitting.



d. Did the training process converge and resulted in the best model?

Compared to the previous model (picture below), there is a remarkable improvement in the test accuracy. Even though Train accuracy has ‘reduced’, big value difference between train and test accuracy signifies overfitting of the models. Therefore, current model tuned with GridSearchCV is significantly better compared to the default model.



e. What is classification accuracy on training and test datasets?

The classification accuracy on training and test datasets are;

**+** **Train accuracy: 0.7493150684931507**

**+ Test Accuracy: 0.7101910828025477**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| **0** | 0.68 | 0.40 | 0.50 | 115 |
| **1** | 0.72 | 0.89 | 0.80 | 199 |
| **avg / total** | 0.70 | 0.71 | 0.69 | 314 |

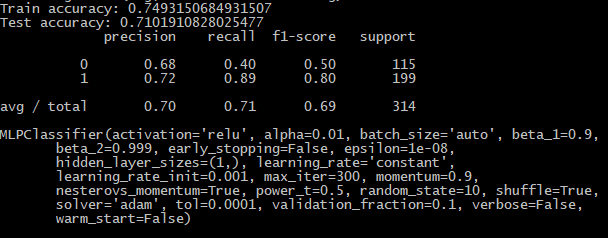
***3. Build another Neural Network model with inputs selected from RFE with regression (use the best model generated in Task 3) and selection with decision tree (use the best model from Task 2). Answer the following:***

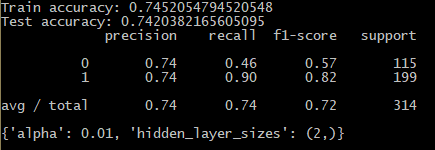
a. Did feature selection help here? Any change in the network architecture? What inputs are being used as the network input?

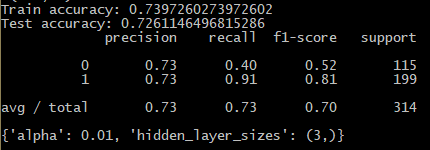
In the model tuned with GridSearchCV, there was a big improvement from the default model with a slight sign of overfitting. Using RFE, the decision tree’s feature selection did help slightly, however, using RFE with regression displayed better results.

Overall, using Recursive Feature Elimination has improved accuracy notably and most importantly, RFE with Logistic Regression shows close to no overfitting with well improved accuracy.

The difference in neural network architecture in all three models was ‘hidden\_layer\_sizes’ or number of inputs in the hidden node. GridSearchCV showed optimal performance with at 1, RFE with Logistic Regression at 2, and RFE Feature selection with Decision tree at 3.

 **GridSearchCV only**

**Logistic Regression RFE**

**Decision Tree RFE**

b. What is classification accuracy on training and test datasets? Is there any

improvement in the outcome?

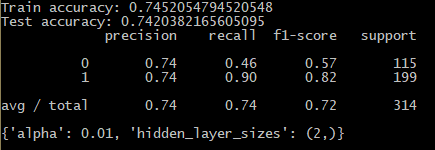
As explained above, there was a notable improvement on accuracy compared to the last model tuned with GridSearchCV with a difference of 0.03, and compared to the default model, a significant difference of 0.08.

c. How many iterations are now needed to train this network?

800 iterations are now required to train the network.

d. Do you see any sign of over-fitting?

As seen in the picture below, the accuracy score displays no overfitting signs from this model.



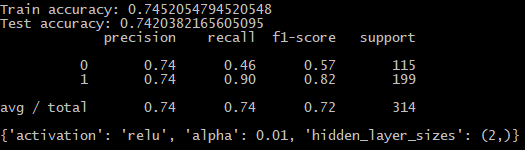
e. Did the training process converge and resulted in the best model?

Combining GridSearchCV and dimensionality reduction with recursive feature elimination using logistic regression has significantly improved its accuracy and removed overfitting. Therefore, it can be concluded that this model is the best model out of previous methods.

f. Finally, see whether the change in network architecture can further improve the performance, use GridSearchCV to tune the network. Report if there was any improvement.

Best performing model was attempted to tune ‘activation’ using the parameter ‘relu(current version)’, ‘identity’, ‘tanh’ and ‘logistic’ for further optimization.

The model’s accuracy showed no improvements



***4. Using the comparison methods, which of the models (i.e one with selected variables and another with all variables) appears to be better? From the better model, can you identify which students to target? Can you provide some descriptive summary of those students?***

Using the accuracy comparison method (first picture below), it is seen that GridSearchCV refined logarithmic regression with selected variables using Recursive Feature Elimination applied to the Neural networks classifier worked the best.

From this model, using the ‘feature importance’ function, it has displayed in descending order of the input variables that were deemed the most important during the decision process, it is seen that two input variable, ‘failures’ and ‘absences’ were the only variables that were used in the classifier and ‘failures’ with the importance of 0.87 and ‘absences’ with the importance of 0.13. From this feature importance analysis and this optimal classification model, we can conclude students with **previous** **failure history** and also students with **high absence history** are most likely to receive an overall grade of **fail**.

