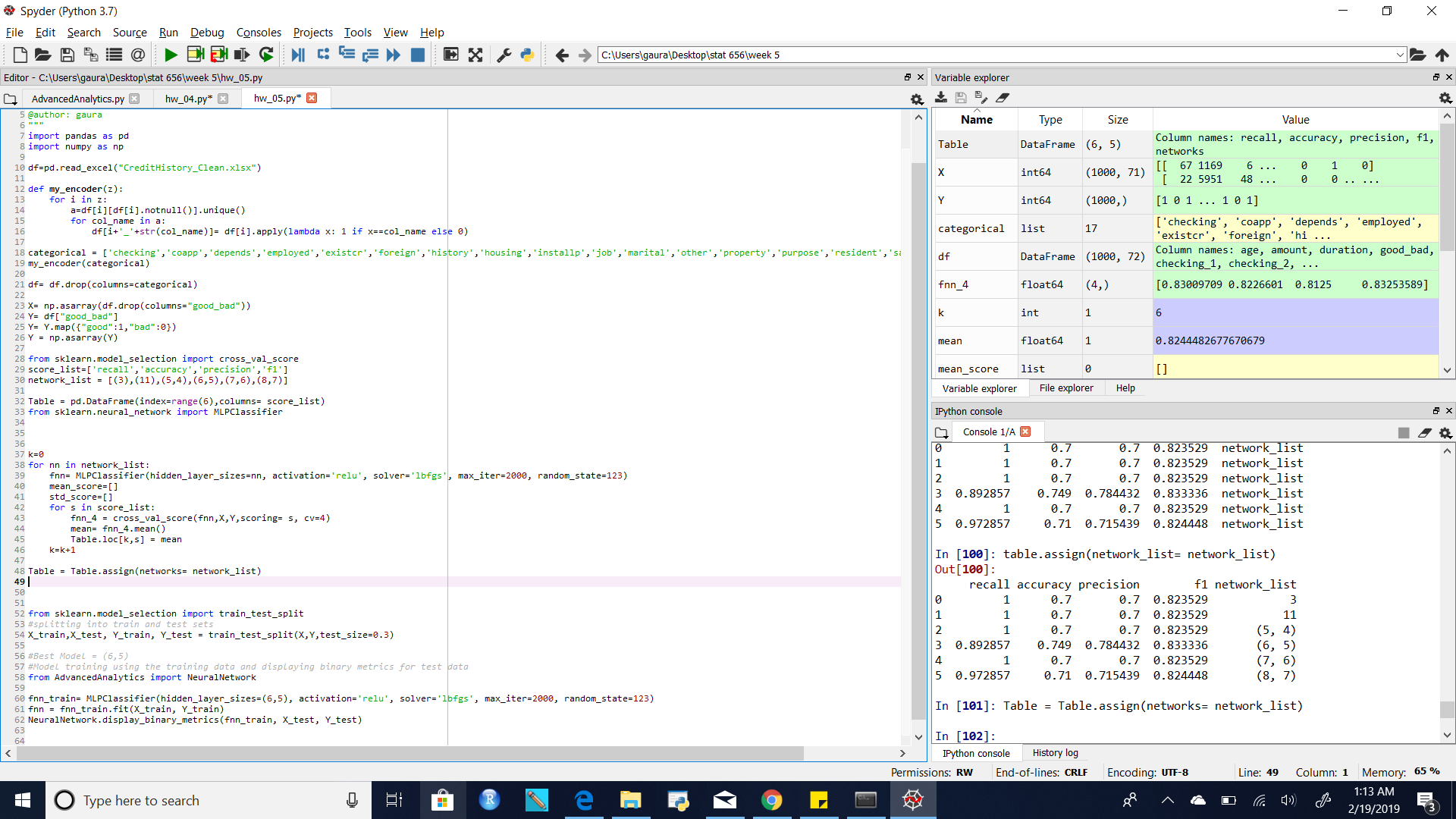
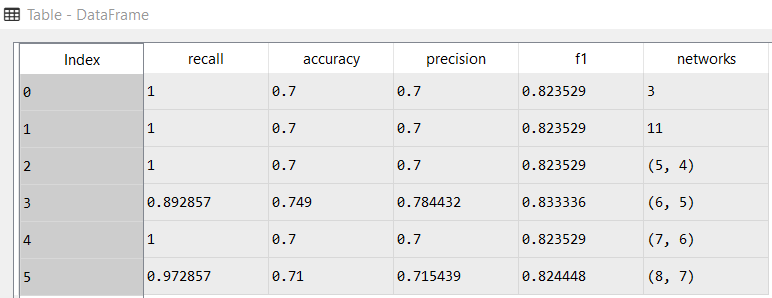
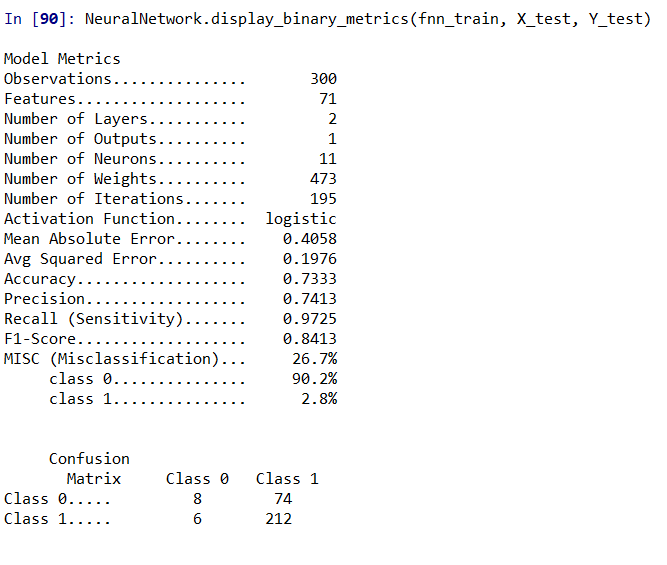
1. **Python code**



1. Table for Accuracy, precision, recall and f1 Score



1. Best Network is (6,5) because it has the best f1 score as well as precision and accuracy.
2. Splitting into train and test data and again calculating the metrics and fitting the neural network on the training set and calculating its predictions on test set and tabulating the confusion matrix and other metrics as follows:



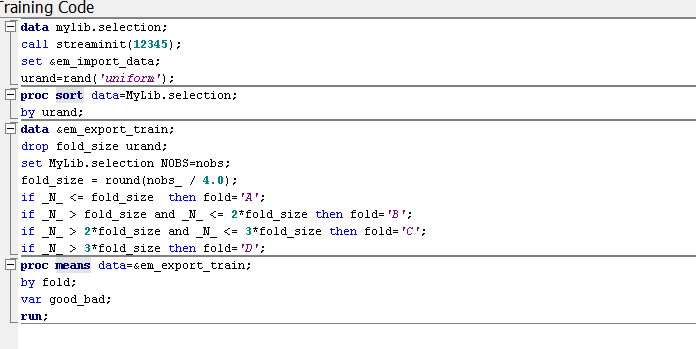
**SAS**

1. SAS\_diagram

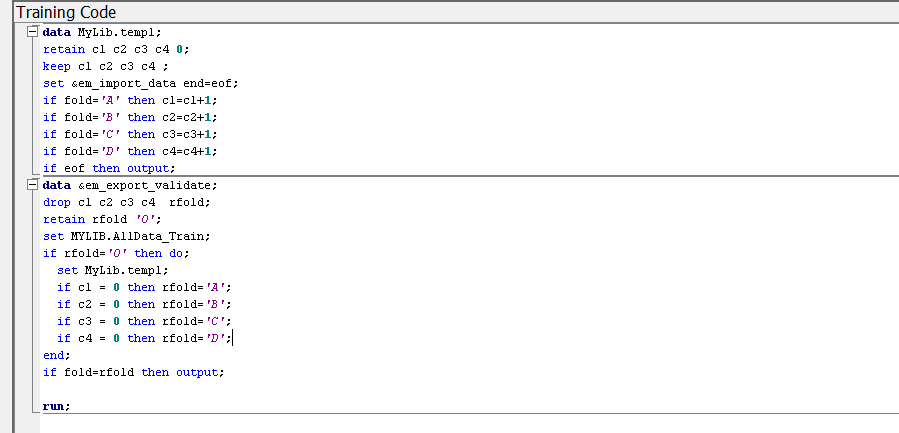
A screenshot of a cell phone

Description automatically generated

1. Creating the folds



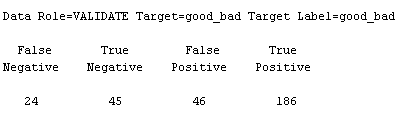
1. Code for cross validation



1. Table of metrics for all the Network configurations

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Network configuration | Accuracy | Recall | Precision | F1 score |
| (3) | 0.79 | 0.84 | 0.85 | 0.85 |
| (11) | 0.75 | 0.83 | 0.82 | 0.83 |
| (5,4) | 0.76 | 0.87 | 0.80 | 0.83 |
| (6,5) | 0.73 | 0.84 | 0.79 | 0.81 |
| (7,6) | 0.80 | 0.86 | 0.81 | 0.86 |
| (8,7) | 0.72 | 0.84 | 0.78 | 0.80 |

1. Best network configuration out of all these is (7,6) as it has the highest f1 score and acuuracy as well as comparable recall and precision.
2. Table for test set after training on 70% data and testing on the remaining 30% and using the network (7,6)



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Network configuration | Accuracy | Recall | Precision | F1 score |
| (3) | 0.77 | 0.89 | 0.85 | 0.87 |

1. Neural network Diagram

A close up of a map

Description automatically generated