60 fits failed out of a total of 80.

The score on these train-test partitions for these parameters will be set to nan.

If these failures are not expected, you can try to debug them by setting error\_score='raise'.

Below are more details about the failures:

--------------------------------------------------------------------------------

10 fits failed with the following error:

Traceback (most recent call last):

File "c:\Users\Glonk\anaconda3\lib\site-packages\sklearn\model\_selection\\_validation.py", line 680, in \_fit\_and\_score

estimator.fit(X\_train, y\_train, \*\*fit\_params)

File "c:\Users\Glonk\anaconda3\lib\site-packages\sklearn\neural\_network\\_multilayer\_perceptron.py", line 752, in fit

return self.\_fit(X, y, incremental=False)

File "c:\Users\Glonk\anaconda3\lib\site-packages\sklearn\neural\_network\\_multilayer\_perceptron.py", line 427, in \_fit

self.\_fit\_stochastic(

File "c:\Users\Glonk\anaconda3\lib\site-packages\sklearn\neural\_network\\_multilayer\_perceptron.py", line 635, in \_fit\_stochastic

batch\_loss, coef\_grads, intercept\_grads = self.\_backprop(

File "c:\Users\Glonk\anaconda3\lib\site-packages\sklearn\neural\_network\\_multilayer\_perceptron.py", line 307, in \_backprop

inplace\_derivative = DERIVATIVES[self.activation]

KeyError: 'softmax'

--------------------------------------------------------------------------------

10 fits failed with the following error:

Traceback (most recent call last):

File "c:\Users\Glonk\anaconda3\lib\site-packages\sklearn\model\_selection\\_validation.py", line 680, in \_fit\_and\_score

...

c:\Users\Glonk\anaconda3\lib\site-packages\sklearn\model\_selection\\_search.py:969: UserWarning: One or more of the test scores are non-finite: [ nan nan nan nan nan nan

0.107 0.107 0.80366667 0.76466667 nan nan

nan nan nan nan]

warnings.warn(

Best: 0.803667 using {'activation': 'tanh', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'adam'}

nan (nan) with: {'activation': 'softmax', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'adam'}

nan (nan) with: {'activation': 'softmax', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'sgd'}

nan (nan) with: {'activation': 'softplus', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'adam'}

nan (nan) with: {'activation': 'softplus', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'sgd'}

nan (nan) with: {'activation': 'softsign', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'adam'}

nan (nan) with: {'activation': 'softsign', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'sgd'}

0.107000 (0.000667) with: {'activation': 'relu', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'adam'}

0.107000 (0.000667) with: {'activation': 'relu', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'sgd'}

0.803667 (0.016411) with: {'activation': 'tanh', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'adam'}

0.764667 (0.014583) with: {'activation': 'tanh', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'sgd'}

nan (nan) with: {'activation': 'sigmoid', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'adam'}

nan (nan) with: {'activation': 'sigmoid', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'sgd'}

nan (nan) with: {'activation': 'hard\_sigmoid', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'adam'}

nan (nan) with: {'activation': 'hard\_sigmoid', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'sgd'}

nan (nan) with: {'activation': 'linear', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'adam'}

nan (nan) with: {'activation': 'linear', 'alpha': 0.006579332246575682, 'hidden\_layer\_sizes': range(64, 100), 'learning\_rate\_init': 0.00031992671377973844, 'max\_iter': 1800, 'n\_iter\_no\_change': 25, 'random\_state': 1, 'solver': 'sgd'}