ANN

February 8, 2021

[2]:
```
import pandas as pd
import numpy as np
import warnings
from sklearn.neural_network import MLPClassifier
from sklearn.model_selection import RepeatedStratifiedKFold
from sklearn.model_selection import GridSearchCV

warnings.filterwarnings('ignore')
```

1 Feature Selection Accuracy

[3]:
```
# Feature Selection Accuracy
```

[8]:
```
cv_strat = RepeatedStratifiedKFold(n_splits=5, n_repeats=4, random_state=42)
param = {'hidden_layer_sizes': [4, 8, 10, 12, 16, 20, 25], 'activation':␣
 ↪['identity', 'logistic', 'tanh', 'relu' ], 'solver':['lbfgs', 'sgd',␣
 ↪'adam'],'alpha':[0.0001, 0.001], 'learning_rate':['constant', 'adaptive',␣
 ↪'invscaling'], 'max_iter':[200, 500]}
```

[9]:
```
gs_ANN = GridSearchCV(estimator = MLPClassifier(), param_grid = param,␣
 ↪cv=cv_strat, scoring='accuracy')
gs_ANN.fit(X_, y_)
print(gs_ANN.best_params_)
print(gs_ANN.best_score_)
```

```
{'activation': 'identity', 'alpha': 0.001, 'hidden_layer_sizes': 25,
'learning_rate': 'constant', 'max_iter': 500, 'solver': 'lbfgs'}
0.7095588235294117
```

[7]:
```
results_ANN = pd.DataFrame(gs_ANN.cv_results_['params'])
results_ANN['test_score'] = gs_ANN.cv_results_['mean_test_score']
results_ANN
```

[7]:
```
    activation   alpha  hidden_layer_sizes learning_rate  max_iter solver  \
0         relu  0.0001                   4      constant       500  lbfgs
```

```
1         relu  0.0001                   4      constant      500    sgd
2         relu  0.0001                   4      constant      500   adam
3         relu  0.0001                   4      adaptive      500  lbfgs
4         relu  0.0001                   4      adaptive      500    sgd
..         …     …                 …            …       …    …
121       relu  0.0010                  25      adaptive      500    sgd
122       relu  0.0010                  25      adaptive      500   adam
123       relu  0.0010                  25    invscaling      500  lbfgs
124       relu  0.0010                  25    invscaling      500    sgd
125       relu  0.0010                  25    invscaling      500   adam

     test_score
0      0.298529
1      0.100735
2      0.233824
3      0.361029
4      0.094853
..          …
121    0.163971
122    0.498529
123    0.677941
124    0.033824
125    0.486029

[126 rows x 7 columns]
```

2 Full Dataset Accuracy

[12]:

```
data_all = pd.read_csv("../Dataset/leaf.csv", delimiter=",")
X_all, y_all = data_all.iloc[:, :-1], data_all.iloc[:, -1:]
num_features_all = len(X_all.columns)
```

[9]:

```
param = {'hidden_layer_sizes': [4, 8, 10, 12, 16, 20, 25], 'activation':␣
 ↪['relu' ], 'solver':['lbfgs', 'sgd', 'adam'],'alpha':[0.0001, 0.001],␣
 ↪'learning_rate':['constant', 'adaptive'], 'max_iter':[500]}
gs_ANN_all = GridSearchCV(estimator = MLPClassifier(), param_grid = param,␣
 ↪cv=cv_strat, scoring='accuracy')
gs_ANN_all.fit(X_all, y_all)
print(gs_ANN_all.best_params_)
print(gs_ANN_all.best_score_)
```

```
{'activation': 'relu', 'alpha': 0.001, 'hidden_layer_sizes': 25,
'learning_rate': 'adaptive', 'max_iter': 500, 'solver': 'lbfgs'}
0.7169117647058824
```

```
[11]: results_ANN_all= pd.DataFrame(gs_ANN_all.cv_results_['params'])
      results_ANN_all['test_score'] = gs_ANN_all.cv_results_['mean_test_score']
      results_ANN_all
```

```
[11]:    activation   alpha  hidden_layer_sizes learning_rate  max_iter solver  \
      0        relu  0.0001                   4      constant       500  lbfgs
      1        relu  0.0001                   4      constant       500    sgd
      2        relu  0.0001                   4      constant       500   adam
      3        relu  0.0001                   4      adaptive       500  lbfgs
      4        relu  0.0001                   4      adaptive       500    sgd
      ..        …     …                 …           …       …    …
      79       relu  0.0010                  25      constant       500    sgd
      80       relu  0.0010                  25      constant       500   adam
      81       relu  0.0010                  25      adaptive       500  lbfgs
      82       relu  0.0010                  25      adaptive       500    sgd
      83       relu  0.0010                  25      adaptive       500   adam

          test_score
      0     0.335294
      1     0.080882
      2     0.251471
      3     0.390441
      4     0.097059
      ..         …
      79    0.186765
      80    0.550735
      81    0.716912
      82    0.191176
      83    0.549265

      [84 rows x 7 columns]
```

```
[ ]:
```