neural-networks

November 25, 2024

[1]:
```
import pandas as pd

from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn.model_selection import train_test_split, cross_val_score

# Create your first MLP in Keras
from keras.models import Sequential
from keras.layers import Dense
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:
```
data = pd.read_csv("/content/Alphabets_data.csv")
data
```

[2]:

	letter	xbox	ybox	width	height	onpix	xbar	ybar	x2bar	y2bar	\
0	T	2	8	3	5	1	8	13	0	6	
1	I	5	12	3	7	2	10	5	5	4	
2	D	4	11	6	8	6	10	6	2	6	
3	N	7	11	6	6	3	5	9	4	6	
4	G	2	1	3	1	1	8	6	6	6	
…	…	…	…	…	…	…	…	…	…	…	…
19995	D	2	2	3	3	2	7	7	7	6	
19996	C	7	10	8	8	4	4	8	6	9	
19997	T	6	9	6	7	5	6	11	3	7	
19998	S	2	3	4	2	1	8	7	2	6	
19999	A	4	9	6	6	2	9	5	3	1	

	xybar	x2ybar	xy2bar	xedge	xedgey	yedge	yedgex
0	6	10	8	0	8	0	8
1	13	3	9	2	8	4	10
2	10	3	7	3	7	3	9
3	4	4	10	6	10	2	8
4	6	5	9	1	7	5	10
…	…	…	…	…	…	…	…
19995	6	6	4	2	8	3	7
19996	12	9	13	2	9	3	7

```
19997     11       9       5       2      12       2       4
19998     10       6       8       1       9       5       8
19999      8       1       8       2       7       2       8

[20000 rows x 17 columns]
```

[5]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20000 entries, 0 to 19999
Data columns (total 17 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   letter  20000 non-null  object
 1   xbox    20000 non-null  int64
 2   ybox    20000 non-null  int64
 3   width   20000 non-null  int64
 4   height  20000 non-null  int64
 5   onpix   20000 non-null  int64
 6   xbar    20000 non-null  int64
 7   ybar    20000 non-null  int64
 8   x2bar   20000 non-null  int64
 9   y2bar   20000 non-null  int64
 10  xybar   20000 non-null  int64
 11  x2ybar  20000 non-null  int64
 12  xy2bar  20000 non-null  int64
 13  xedge   20000 non-null  int64
 14  xedgey  20000 non-null  int64
 15  yedge   20000 non-null  int64
 16  yedgex  20000 non-null  int64
dtypes: int64(16), object(1)
memory usage: 2.6+ MB
```

[7]:
```
data.corr(numeric_only=True)
```

[7]:
```
            xbox      ybox     width    height     onpix      xbar      ybar  \
xbox    1.000000  0.757793  0.851514  0.672764  0.619097 -0.032595  0.045545
ybox    0.757793  1.000000  0.671912  0.823207  0.555067  0.045690 -0.040925
width   0.851514  0.671912  1.000000  0.660215  0.765716  0.061959  0.024832
height  0.672764  0.823207  0.660215  1.000000  0.644366  0.042844 -0.020072
onpix   0.619097  0.555067  0.765716  0.644366  1.000000  0.139159 -0.028822
xbar   -0.032595  0.045690  0.061959  0.042844  0.139159  1.000000 -0.356580
ybar    0.045545 -0.040925  0.024832 -0.020072 -0.028822 -0.356580  1.000000
x2bar   0.014306 -0.025019 -0.098611  0.082383 -0.011985 -0.053306 -0.118625
y2bar   0.052086  0.096478  0.057074  0.059032 -0.065557 -0.122851 -0.049658
xybar   0.148056  0.159954  0.115018  0.012458 -0.069776  0.085963  0.178318
x2ybar  0.035464 -0.054648  0.011694 -0.011991 -0.072941 -0.341957  0.600397
```

```
xy2bar -0.046333 -0.007568 -0.045009  0.026386 -0.038858 -0.032115 -0.271649
xedge   0.489155  0.274431  0.557251  0.265243  0.627507  0.144325 -0.036722
xedgey  0.098180 -0.001336  0.045658  0.025359  0.017649 -0.253339  0.555060
yedge   0.273504  0.230883  0.260285  0.297545  0.492653  0.127056 -0.078008
yedgex -0.105147 -0.042741 -0.118273 -0.018853 -0.062969  0.248816 -0.207900

           x2bar     y2bar     xybar    x2ybar    xy2bar     xedge    xedgey  \
xbox    0.014306  0.052086  0.148056  0.035464 -0.046333  0.489155  0.098180
ybox   -0.025019  0.096478  0.159954 -0.054648 -0.007568  0.274431 -0.001336
width  -0.098611  0.057074  0.115018  0.011694 -0.045009  0.557251  0.045658
height  0.082383  0.059032  0.012458 -0.011991  0.026386  0.265243  0.025359
onpix  -0.011985 -0.065557 -0.069776 -0.072941 -0.038858  0.627507  0.017649
xbar   -0.053306 -0.122851  0.085963 -0.341957 -0.032115  0.144325 -0.253339
ybar   -0.118625 -0.049658  0.178318  0.600397 -0.271649 -0.036722  0.555060
x2bar   1.000000 -0.188431 -0.317780  0.042545  0.082020  0.142132 -0.084820
y2bar  -0.188431  1.000000  0.132000 -0.060116  0.119048 -0.384018 -0.052545
xybar  -0.317780  0.132000  1.000000  0.057988 -0.106759 -0.175676  0.029419
x2ybar  0.042545 -0.060116  0.057988  1.000000  0.063214  0.053566  0.527239
xy2bar  0.082020  0.119048 -0.106759  0.063214  1.000000 -0.008753 -0.184927
xedge   0.142132 -0.384018 -0.175676  0.053566 -0.008753  1.000000  0.002849
xedgey -0.084820 -0.052545  0.029419  0.527239 -0.184927  0.002849  1.000000
yedge   0.006546  0.277540 -0.087019 -0.226251  0.049695  0.108411 -0.064402
yedgex  0.182902 -0.061335 -0.114223 -0.236518  0.245808 -0.049789 -0.187591

           yedge    yedgex
xbox    0.273504 -0.105147
ybox    0.230883 -0.042741
width   0.260285 -0.118273
height  0.297545 -0.018853
onpix   0.492653 -0.062969
xbar    0.127056  0.248816
ybar   -0.078008 -0.207900
x2bar   0.006546  0.182902
y2bar   0.277540 -0.061335
xybar  -0.087019 -0.114223
x2ybar -0.226251 -0.236518
xy2bar  0.049695  0.245808
xedge   0.108411 -0.049789
xedgey -0.064402 -0.187591
yedge   1.000000  0.143588
yedgex  0.143588  1.000000
```

[8]:
```
data.isnull().sum()
```

[8]:
```
letter    0
xbox      0
ybox      0
```

```
width     0
height    0
onpix     0
xbar      0
ybar      0
x2bar     0
y2bar     0
xybar     0
x2ybar    0
xy2bar    0
xedge     0
xedgey    0
yedge     0
yedgex    0
dtype: int64
```

[9]:
```
duplicates = data.duplicated()
duplicates.value_counts()
```

[9]:
```
False    18668
True      1332
Name: count, dtype: int64
```

[11]:
```
data1 = data.drop_duplicates(keep=False)
```

[12]:
```
duplicates = data1.duplicated()
duplicates.value_counts()
```

[12]:
```
False    17823
Name: count, dtype: int64
```

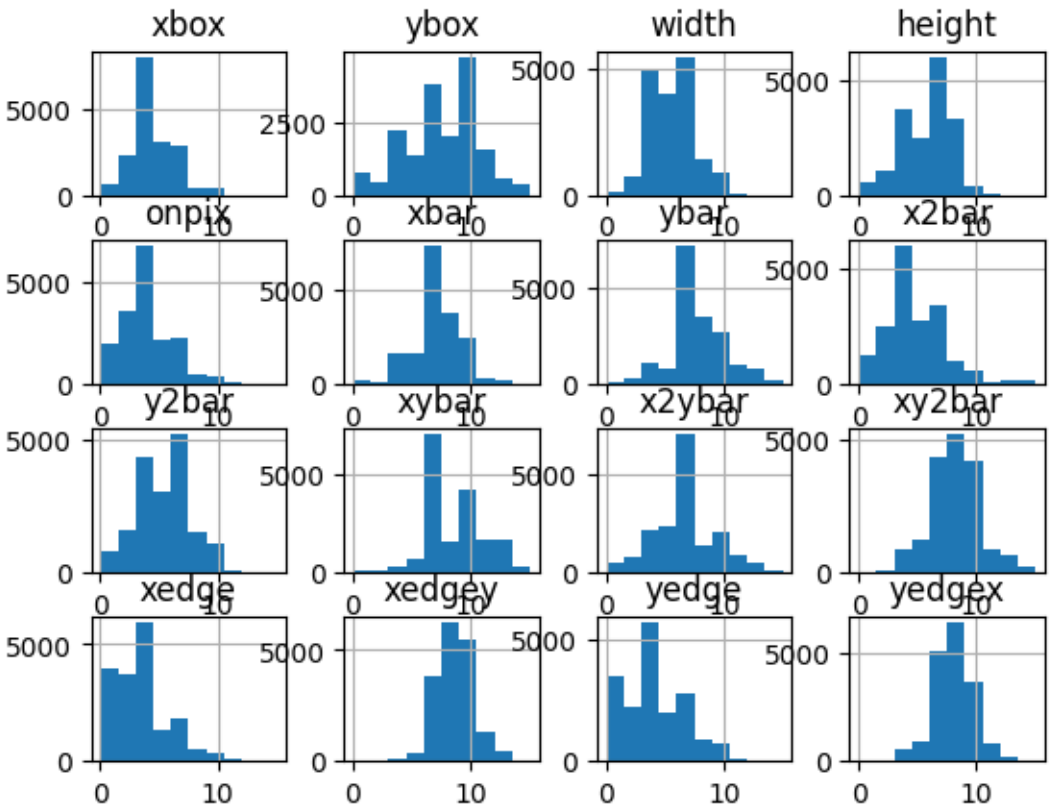
[13]:
```
data1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 17823 entries, 0 to 19999
Data columns (total 17 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   letter  17823 non-null  object
 1   xbox    17823 non-null  int64
 2   ybox    17823 non-null  int64
 3   width   17823 non-null  int64
 4   height  17823 non-null  int64
 5   onpix   17823 non-null  int64
 6   xbar    17823 non-null  int64
 7   ybar    17823 non-null  int64
 8   x2bar   17823 non-null  int64
```

```
 9   y2bar   17823 non-null  int64
 10  xybar   17823 non-null  int64
 11  x2ybar  17823 non-null  int64
 12  xy2bar  17823 non-null  int64
 13  xedge   17823 non-null  int64
 14  xedgey  17823 non-null  int64
 15  yedge   17823 non-null  int64
 16  yedgex  17823 non-null  int64
dtypes: int64(16), object(1)
memory usage: 2.4+ MB
```

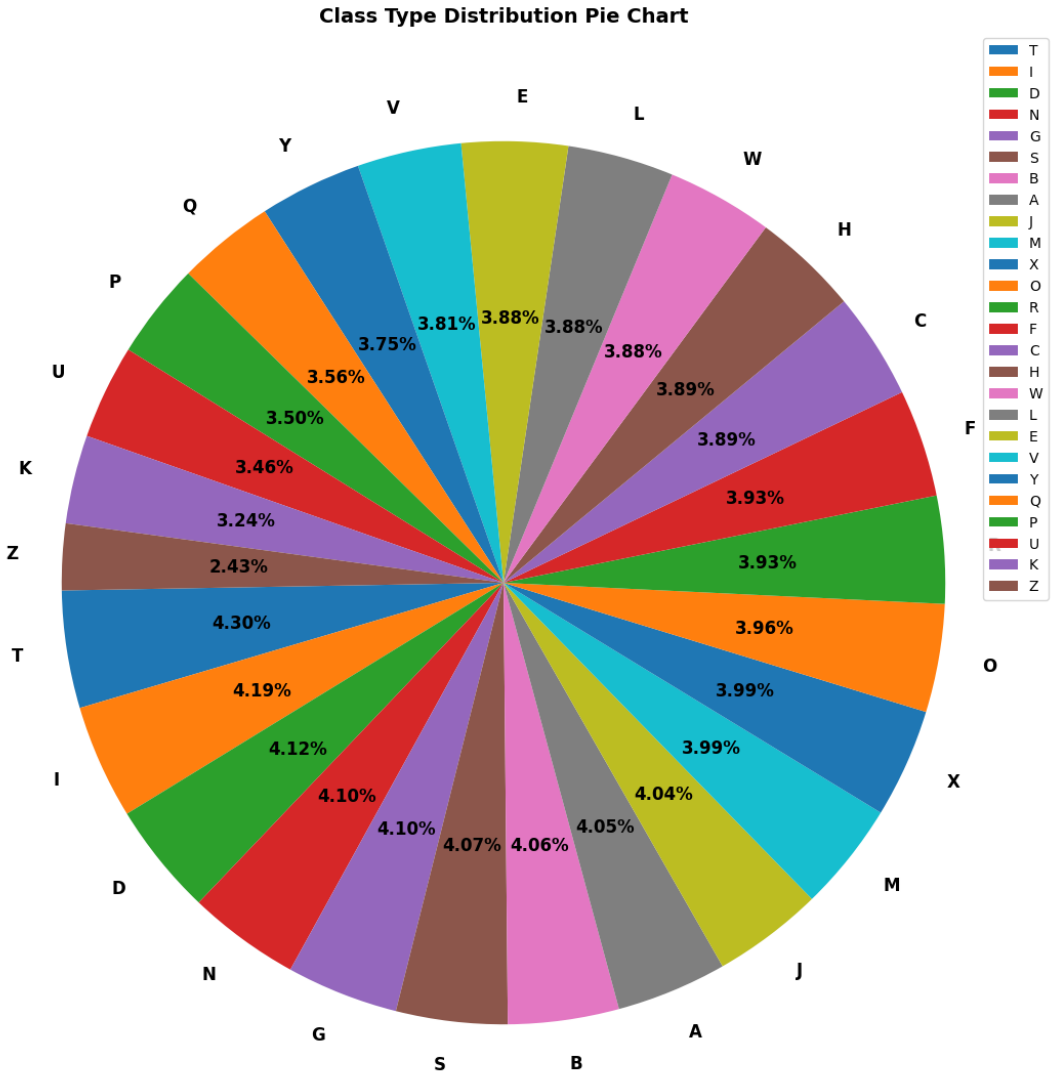
[14]:
```
data2= data1.copy()
```

[15]:
```
data2.hist()
plt.rcParams.update({'figure.figsize':(25,12), 'figure.dpi':100})
```

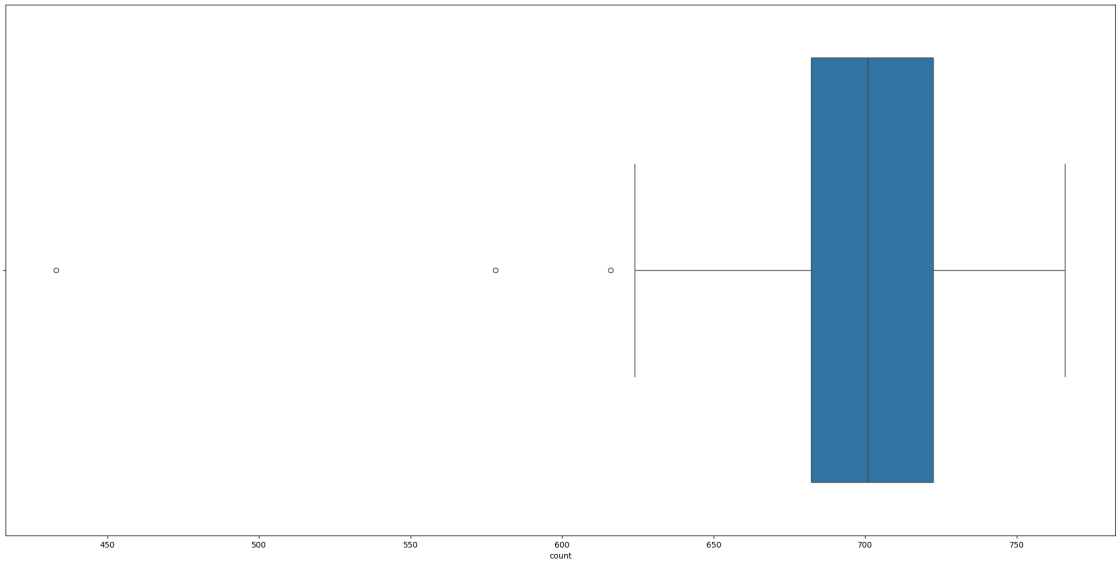


[16]:
```
plt.figure(figsize=(14, 14))
plt.pie(data2['letter'].value_counts(), labels=data2.letter.unique(),
        autopct='%.2f%%', startangle=181, textprops={'size': 'large',␣
 ↪'fontweight': 'bold', 'color': 'black'})
plt.legend(loc='upper right')
```

```
plt.title("Class Type Distribution Pie Chart", fontsize=14, fontweight='bold')
plt.show()
```



[17]:
```
df = data2.letter.value_counts()
sns.boxplot(x=df)
plt.show()
```

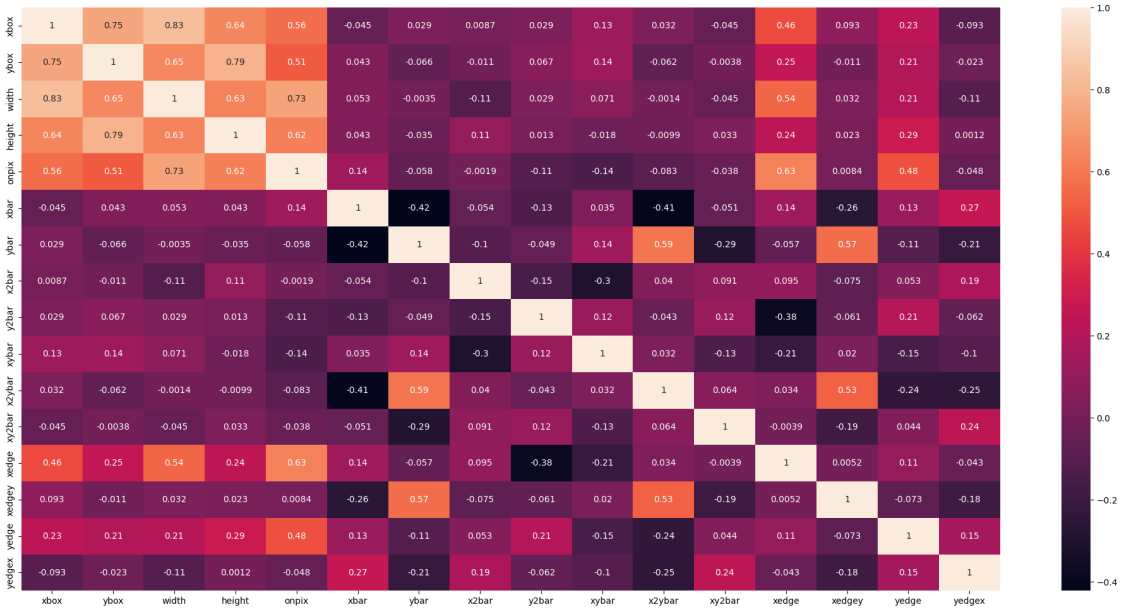


[19]:
```
correlation = data2.select_dtypes(include=np.number).corr()
correlation.style.background_gradient(cmap='BrBG')
```

[19]:
```
<pandas.io.formats.style.Styler at 0x7be620790b50>
```

[20]:
```
data2 = np.random.randint(low=1,
                          high=100,
                          size=(10, 10))
# plotting the heatmap
hm = sns.heatmap(data=correlation,
                 annot=True)
plt.rcParams.update({'figure.figsize':(10,8), 'figure.dpi':100})

# displaying the plotted heatmap
plt.show()
```



[21]:
```
from sklearn import preprocessing

# how to understand word labels.
label_encoder = preprocessing.LabelEncoder()

# Encode labels in column 'letters'.
data1['letter']= label_encoder.fit_transform(data1['letter'])

data1['letter'].unique()
```

```
<ipython-input-21-8732cd2a233d>:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data1['letter']= label_encoder.fit_transform(data1['letter'])
```

[21]:
```
array([19,  8,  3, 13,  6, 18,  1,  0,  9, 12, 23, 14, 17,  5,  2,  7, 22,
       11,  4, 21, 24, 16, 15, 20, 10, 25])
```

[22]:
```
x = data1.drop('letter',
               axis = 1)
y = data1['letter']
```

[23]:
```
x
```

[23]:
```
       xbox  ybox  width  height  onpix  xbar  ybar  x2bar  y2bar  xybar  \
0         2     8      3       5      1     8    13      0      6      6
1         5    12      3       7      2    10     5      5      4     13
2         4    11      6       8      6    10     6      2      6     10
3         7    11      6       6      3     5     9      4      6      4
4         2     1      3       1      1     8     6      6      6      6
…       …     …      …       …      …     …     …      …      …      …
19994     5     8      7       7      7     7     9      4      8      7
19995     2     2      3       3      2     7     7      7      6      6
19996     7    10      8       8      4     4     8      6      9     12
19997     6     9      6       7      5     6    11      3      7     11
19999     4     9      6       6      2     9     5      3      1      8

       x2ybar  xy2bar  xedge  xedgey  yedge  yedgex
0          10       8      0       8      0       8
1           3       9      2       8      4      10
2           3       7      3       7      3       9
3           4      10      6      10      2       8
4           5       9      1       7      5      10
…         …     …      …      …     …      …
19994       7       8      3      10      8       6
19995       6       4      2       8      3       7
19996       9      13      2       9      3       7
19997       9       5      2      12      2       4
19999       1       8      2       7      2       8

[17823 rows x 16 columns]
```

[24]:
```
y
```

[24]:
```
0        19
1         8
2         3
3        13
4         6
         ..
19994    19
19995     3
19996     2
19997    19
19999     0
Name: letter, Length: 17823, dtype: int64
```

[25]:
```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.20)
```

[26]:
```
model = Sequential()
model.add(Dense(128, input_dim=16, activation='relu'))
model.add(Dense(32, activation='relu'))
model.add(Dense(26, activation='softmax'))
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

[27]:
```
model.compile(loss='sparse_categorical_crossentropy', optimizer='adam',␣
↪metrics=['accuracy'])
```

[28]:
```
history=model.fit(x_train,y_train, validation_split=0.33, epochs=100,␣
↪batch_size= 128 )
```

```
Epoch 1/100
75/75             2s 5ms/step -
accuracy: 0.0886 - loss: 3.4697 - val_accuracy: 0.2346 - val_loss: 2.6565
Epoch 2/100
75/75             0s 3ms/step -
accuracy: 0.3198 - loss: 2.3936 - val_accuracy: 0.5161 - val_loss: 1.8203
Epoch 3/100
75/75             0s 3ms/step -
accuracy: 0.5515 - loss: 1.6615 - val_accuracy: 0.5986 - val_loss: 1.4521
Epoch 4/100
75/75             0s 3ms/step -
accuracy: 0.6300 - loss: 1.3609 - val_accuracy: 0.6458 - val_loss: 1.2812
Epoch 5/100
75/75             0s 3ms/step -
accuracy: 0.6701 - loss: 1.2203 - val_accuracy: 0.6864 - val_loss: 1.1562
Epoch 6/100
75/75             0s 3ms/step -
accuracy: 0.6986 - loss: 1.1146 - val_accuracy: 0.7133 - val_loss: 1.0689
Epoch 7/100
75/75             0s 3ms/step -
accuracy: 0.7251 - loss: 1.0096 - val_accuracy: 0.7159 - val_loss: 1.0166
Epoch 8/100
75/75             0s 3ms/step -
accuracy: 0.7316 - loss: 0.9584 - val_accuracy: 0.7331 - val_loss: 0.9413
Epoch 9/100
75/75             0s 3ms/step -
accuracy: 0.7577 - loss: 0.8926 - val_accuracy: 0.7418 - val_loss: 0.9150
Epoch 10/100
75/75             0s 3ms/step -
accuracy: 0.7614 - loss: 0.8408 - val_accuracy: 0.7486 - val_loss: 0.8731
Epoch 11/100
```

```
75/75            0s 3ms/step -
accuracy: 0.7702 - loss: 0.8100 - val_accuracy: 0.7563 - val_loss: 0.8342
Epoch 12/100
75/75            0s 3ms/step -
accuracy: 0.7737 - loss: 0.7956 - val_accuracy: 0.7743 - val_loss: 0.8000
Epoch 13/100
75/75            0s 3ms/step -
accuracy: 0.7878 - loss: 0.7568 - val_accuracy: 0.7818 - val_loss: 0.7813
Epoch 14/100
75/75            0s 3ms/step -
accuracy: 0.7883 - loss: 0.7413 - val_accuracy: 0.7860 - val_loss: 0.7465
Epoch 15/100
75/75            0s 3ms/step -
accuracy: 0.7967 - loss: 0.7137 - val_accuracy: 0.7879 - val_loss: 0.7367
Epoch 16/100
75/75            0s 3ms/step -
accuracy: 0.8083 - loss: 0.6776 - val_accuracy: 0.7928 - val_loss: 0.7201
Epoch 17/100
75/75            0s 3ms/step -
accuracy: 0.8096 - loss: 0.6640 - val_accuracy: 0.7952 - val_loss: 0.6963
Epoch 18/100
75/75            0s 3ms/step -
accuracy: 0.8226 - loss: 0.6442 - val_accuracy: 0.8130 - val_loss: 0.6621
Epoch 19/100
75/75            0s 3ms/step -
accuracy: 0.8229 - loss: 0.6280 - val_accuracy: 0.8098 - val_loss: 0.6614
Epoch 20/100
75/75            0s 3ms/step -
accuracy: 0.8177 - loss: 0.6321 - val_accuracy: 0.8124 - val_loss: 0.6480
Epoch 21/100
75/75            0s 4ms/step -
accuracy: 0.8333 - loss: 0.6004 - val_accuracy: 0.8147 - val_loss: 0.6303
Epoch 22/100
75/75            0s 3ms/step -
accuracy: 0.8333 - loss: 0.5822 - val_accuracy: 0.8185 - val_loss: 0.6339
Epoch 23/100
75/75            1s 6ms/step -
accuracy: 0.8365 - loss: 0.5645 - val_accuracy: 0.8179 - val_loss: 0.6197
Epoch 24/100
75/75            0s 4ms/step -
accuracy: 0.8383 - loss: 0.5664 - val_accuracy: 0.8255 - val_loss: 0.5976
Epoch 25/100
75/75            0s 5ms/step -
accuracy: 0.8486 - loss: 0.5378 - val_accuracy: 0.8351 - val_loss: 0.5807
Epoch 26/100
75/75            1s 8ms/step -
accuracy: 0.8468 - loss: 0.5273 - val_accuracy: 0.8264 - val_loss: 0.5953
Epoch 27/100
```

```
75/75            1s 6ms/step -
accuracy: 0.8484 - loss: 0.5256 - val_accuracy: 0.8319 - val_loss: 0.5914
Epoch 28/100
75/75            1s 6ms/step -
accuracy: 0.8485 - loss: 0.5208 - val_accuracy: 0.8383 - val_loss: 0.5488
Epoch 29/100
75/75            1s 6ms/step -
accuracy: 0.8497 - loss: 0.5077 - val_accuracy: 0.8455 - val_loss: 0.5555
Epoch 30/100
75/75            0s 5ms/step -
accuracy: 0.8602 - loss: 0.4932 - val_accuracy: 0.8389 - val_loss: 0.5591
Epoch 31/100
75/75            0s 3ms/step -
accuracy: 0.8579 - loss: 0.4871 - val_accuracy: 0.8430 - val_loss: 0.5330
Epoch 32/100
75/75            0s 3ms/step -
accuracy: 0.8611 - loss: 0.4761 - val_accuracy: 0.8513 - val_loss: 0.5082
Epoch 33/100
75/75            0s 3ms/step -
accuracy: 0.8567 - loss: 0.4778 - val_accuracy: 0.8479 - val_loss: 0.5130
Epoch 34/100
75/75            0s 3ms/step -
accuracy: 0.8731 - loss: 0.4421 - val_accuracy: 0.8483 - val_loss: 0.5012
Epoch 35/100
75/75            0s 3ms/step -
accuracy: 0.8713 - loss: 0.4596 - val_accuracy: 0.8587 - val_loss: 0.4872
Epoch 36/100
75/75            0s 3ms/step -
accuracy: 0.8734 - loss: 0.4333 - val_accuracy: 0.8538 - val_loss: 0.4844
Epoch 37/100
75/75            0s 3ms/step -
accuracy: 0.8724 - loss: 0.4272 - val_accuracy: 0.8498 - val_loss: 0.4945
Epoch 38/100
75/75            0s 3ms/step -
accuracy: 0.8773 - loss: 0.4283 - val_accuracy: 0.8551 - val_loss: 0.4923
Epoch 39/100
75/75            0s 3ms/step -
accuracy: 0.8711 - loss: 0.4355 - val_accuracy: 0.8561 - val_loss: 0.4776
Epoch 40/100
75/75            0s 3ms/step -
accuracy: 0.8795 - loss: 0.3983 - val_accuracy: 0.8627 - val_loss: 0.4581
Epoch 41/100
75/75            0s 3ms/step -
accuracy: 0.8839 - loss: 0.3914 - val_accuracy: 0.8657 - val_loss: 0.4475
Epoch 42/100
75/75            0s 3ms/step -
accuracy: 0.8878 - loss: 0.3816 - val_accuracy: 0.8638 - val_loss: 0.4657
Epoch 43/100
```

```
75/75            0s 3ms/step -
accuracy: 0.8863 - loss: 0.3919 - val_accuracy: 0.8685 - val_loss: 0.4388
Epoch 44/100
75/75            0s 3ms/step -
accuracy: 0.8883 - loss: 0.3815 - val_accuracy: 0.8691 - val_loss: 0.4423
Epoch 45/100
75/75            0s 3ms/step -
accuracy: 0.8881 - loss: 0.3756 - val_accuracy: 0.8672 - val_loss: 0.4406
Epoch 46/100
75/75            0s 3ms/step -
accuracy: 0.9012 - loss: 0.3639 - val_accuracy: 0.8693 - val_loss: 0.4424
Epoch 47/100
75/75            0s 3ms/step -
accuracy: 0.8917 - loss: 0.3711 - val_accuracy: 0.8719 - val_loss: 0.4227
Epoch 48/100
75/75            0s 4ms/step -
accuracy: 0.8932 - loss: 0.3627 - val_accuracy: 0.8674 - val_loss: 0.4329
Epoch 49/100
75/75            0s 4ms/step -
accuracy: 0.8879 - loss: 0.3696 - val_accuracy: 0.8787 - val_loss: 0.4084
Epoch 50/100
75/75            1s 4ms/step -
accuracy: 0.8937 - loss: 0.3514 - val_accuracy: 0.8759 - val_loss: 0.4096
Epoch 51/100
75/75            1s 6ms/step -
accuracy: 0.8965 - loss: 0.3441 - val_accuracy: 0.8780 - val_loss: 0.4083
Epoch 52/100
75/75            1s 4ms/step -
accuracy: 0.8987 - loss: 0.3457 - val_accuracy: 0.8729 - val_loss: 0.4286
Epoch 53/100
75/75            0s 3ms/step -
accuracy: 0.8981 - loss: 0.3345 - val_accuracy: 0.8802 - val_loss: 0.3912
Epoch 54/100
75/75            0s 3ms/step -
accuracy: 0.9027 - loss: 0.3292 - val_accuracy: 0.8789 - val_loss: 0.3945
Epoch 55/100
75/75            0s 4ms/step -
accuracy: 0.9015 - loss: 0.3247 - val_accuracy: 0.8787 - val_loss: 0.3927
Epoch 56/100
75/75            0s 3ms/step -
accuracy: 0.9131 - loss: 0.3026 - val_accuracy: 0.8808 - val_loss: 0.3968
Epoch 57/100
75/75            1s 7ms/step -
accuracy: 0.9045 - loss: 0.3263 - val_accuracy: 0.8789 - val_loss: 0.3871
Epoch 58/100
75/75            0s 3ms/step -
accuracy: 0.9079 - loss: 0.3063 - val_accuracy: 0.8816 - val_loss: 0.3862
Epoch 59/100
```

```
75/75            1s 8ms/step -
accuracy: 0.9133 - loss: 0.3127 - val_accuracy: 0.8840 - val_loss: 0.3808
Epoch 60/100
75/75            1s 7ms/step -
accuracy: 0.9074 - loss: 0.3120 - val_accuracy: 0.8780 - val_loss: 0.3923
Epoch 61/100
75/75            0s 5ms/step -
accuracy: 0.9098 - loss: 0.2943 - val_accuracy: 0.8859 - val_loss: 0.3725
Epoch 62/100
75/75            1s 6ms/step -
accuracy: 0.9110 - loss: 0.3058 - val_accuracy: 0.8821 - val_loss: 0.3783
Epoch 63/100
75/75            0s 4ms/step -
accuracy: 0.9083 - loss: 0.3009 - val_accuracy: 0.8901 - val_loss: 0.3716
Epoch 64/100
75/75            1s 5ms/step -
accuracy: 0.9071 - loss: 0.2981 - val_accuracy: 0.8802 - val_loss: 0.3799
Epoch 65/100
75/75            1s 6ms/step -
accuracy: 0.9145 - loss: 0.2974 - val_accuracy: 0.8948 - val_loss: 0.3610
Epoch 66/100
75/75            1s 5ms/step -
accuracy: 0.9167 - loss: 0.2742 - val_accuracy: 0.8829 - val_loss: 0.3776
Epoch 67/100
75/75            1s 7ms/step -
accuracy: 0.9129 - loss: 0.2835 - val_accuracy: 0.8853 - val_loss: 0.3650
Epoch 68/100
75/75            1s 7ms/step -
accuracy: 0.9193 - loss: 0.2740 - val_accuracy: 0.8874 - val_loss: 0.3698
Epoch 69/100
75/75            1s 9ms/step -
accuracy: 0.9166 - loss: 0.2807 - val_accuracy: 0.8838 - val_loss: 0.3697
Epoch 70/100
75/75            1s 7ms/step -
accuracy: 0.9218 - loss: 0.2618 - val_accuracy: 0.8889 - val_loss: 0.3585
Epoch 71/100
75/75            1s 7ms/step -
accuracy: 0.9189 - loss: 0.2684 - val_accuracy: 0.8921 - val_loss: 0.3518
Epoch 72/100
75/75            1s 8ms/step -
accuracy: 0.9237 - loss: 0.2625 - val_accuracy: 0.8933 - val_loss: 0.3489
Epoch 73/100
75/75            1s 6ms/step -
accuracy: 0.9248 - loss: 0.2605 - val_accuracy: 0.8878 - val_loss: 0.3730
Epoch 74/100
75/75            0s 3ms/step -
accuracy: 0.9185 - loss: 0.2590 - val_accuracy: 0.8867 - val_loss: 0.3715
Epoch 75/100
```

```
75/75            0s 3ms/step -
accuracy: 0.9138 - loss: 0.2706 - val_accuracy: 0.8999 - val_loss: 0.3417
Epoch 76/100
75/75            0s 3ms/step -
accuracy: 0.9212 - loss: 0.2603 - val_accuracy: 0.8929 - val_loss: 0.3472
Epoch 77/100
75/75            0s 3ms/step -
accuracy: 0.9271 - loss: 0.2430 - val_accuracy: 0.8965 - val_loss: 0.3397
Epoch 78/100
75/75            0s 3ms/step -
accuracy: 0.9241 - loss: 0.2471 - val_accuracy: 0.8946 - val_loss: 0.3380
Epoch 79/100
75/75            0s 5ms/step -
accuracy: 0.9267 - loss: 0.2408 - val_accuracy: 0.8931 - val_loss: 0.3402
Epoch 80/100
75/75            1s 10ms/step -
accuracy: 0.9288 - loss: 0.2335 - val_accuracy: 0.8893 - val_loss: 0.3636
Epoch 81/100
75/75            1s 9ms/step -
accuracy: 0.9320 - loss: 0.2357 - val_accuracy: 0.8957 - val_loss: 0.3510
Epoch 82/100
75/75            1s 8ms/step -
accuracy: 0.9231 - loss: 0.2485 - val_accuracy: 0.8942 - val_loss: 0.3352
Epoch 83/100
75/75            2s 11ms/step -
accuracy: 0.9321 - loss: 0.2359 - val_accuracy: 0.8984 - val_loss: 0.3327
Epoch 84/100
75/75            1s 10ms/step -
accuracy: 0.9331 - loss: 0.2262 - val_accuracy: 0.8923 - val_loss: 0.3415
Epoch 85/100
75/75            1s 7ms/step -
accuracy: 0.9297 - loss: 0.2377 - val_accuracy: 0.8959 - val_loss: 0.3309
Epoch 86/100
75/75            1s 6ms/step -
accuracy: 0.9229 - loss: 0.2477 - val_accuracy: 0.8974 - val_loss: 0.3276
Epoch 87/100
75/75            0s 4ms/step -
accuracy: 0.9289 - loss: 0.2298 - val_accuracy: 0.8963 - val_loss: 0.3429
Epoch 88/100
75/75            0s 3ms/step -
accuracy: 0.9329 - loss: 0.2254 - val_accuracy: 0.9014 - val_loss: 0.3169
Epoch 89/100
75/75            0s 3ms/step -
accuracy: 0.9314 - loss: 0.2232 - val_accuracy: 0.8950 - val_loss: 0.3286
Epoch 90/100
75/75            0s 3ms/step -
accuracy: 0.9336 - loss: 0.2199 - val_accuracy: 0.8978 - val_loss: 0.3258
Epoch 91/100
```

```
75/75             0s 3ms/step -
accuracy: 0.9288 - loss: 0.2275 - val_accuracy: 0.8978 - val_loss: 0.3372
Epoch 92/100
75/75             0s 3ms/step -
accuracy: 0.9335 - loss: 0.2218 - val_accuracy: 0.9012 - val_loss: 0.3204
Epoch 93/100
75/75             0s 3ms/step -
accuracy: 0.9394 - loss: 0.2098 - val_accuracy: 0.9003 - val_loss: 0.3193
Epoch 94/100
75/75             0s 3ms/step -
accuracy: 0.9348 - loss: 0.2258 - val_accuracy: 0.8931 - val_loss: 0.3345
Epoch 95/100
75/75             0s 3ms/step -
accuracy: 0.9304 - loss: 0.2197 - val_accuracy: 0.8980 - val_loss: 0.3285
Epoch 96/100
75/75             0s 3ms/step -
accuracy: 0.9315 - loss: 0.2169 - val_accuracy: 0.8995 - val_loss: 0.3248
Epoch 97/100
75/75             0s 3ms/step -
accuracy: 0.9313 - loss: 0.2167 - val_accuracy: 0.9086 - val_loss: 0.3098
Epoch 98/100
75/75             0s 4ms/step -
accuracy: 0.9363 - loss: 0.2087 - val_accuracy: 0.8959 - val_loss: 0.3266
Epoch 99/100
75/75             0s 3ms/step -
accuracy: 0.9343 - loss: 0.2144 - val_accuracy: 0.9065 - val_loss: 0.2998
Epoch 100/100
75/75             0s 3ms/step -
accuracy: 0.9412 - loss: 0.1941 - val_accuracy: 0.8980 - val_loss: 0.3212
```

[29]:
```
scores = model.evaluate(x, y)
print("%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))
```

```
557/557             1s 1ms/step -
accuracy: 0.9186 - loss: 0.2621
compile_metrics: 91.84%
```

[30]:
```
from sklearn.metrics import classification_report, confusion_matrix

# Define expected unique labels (based on your previous output)
expected_labels = ['T', 'I', 'D', 'N', 'G', 'S', 'B', 'A', 'J', 'M', 'X', 'O',␣
 ↪'R', 'F', 'C', 'H', 'W', 'L', 'P', 'E', 'V', 'Y', 'Q', 'U', 'K', 'Z']

# Evaluate the model
y_pred_prob = model.predict(x_test)
y_pred_encoded = np.argmax(y_pred_prob, axis=1)
```

```
# Convert predicted labels back to original letters using inverse_transform
y_test_pred = label_encoder.inverse_transform(y_test)
y_pred_letters = label_encoder.inverse_transform(y_pred_encoded)
```

```
112/112             0s 2ms/step
```

[31]:
```
report = classification_report(y_test_pred, y_pred_letters,␣
 ↪target_names=expected_labels)

# Print classification report
print("Classification Report:")
print(report)
```

```
Classification Report:
              precision    recall  f1-score   support

           T       0.94      0.95      0.94       140
           I       0.93      0.89      0.91       129
           D       0.94      0.87      0.90       138
           N       0.90      0.79      0.84       142
           G       0.78      0.83      0.80       129
           S       0.95      0.85      0.90       149
           B       0.75      0.86      0.80       125
           A       0.81      0.83      0.82       152
           J       0.90      0.84      0.87        73
           M       0.88      0.96      0.92       114
           X       0.90      0.92      0.91       132
           O       0.87      0.94      0.91       127
           R       0.85      0.98      0.91       135
           F       0.96      0.88      0.92       133
           C       0.88      0.86      0.87       164
           H       0.94      0.90      0.92       157
           W       0.94      0.95      0.94       173
           L       0.89      0.92      0.90       143
           P       0.87      0.92      0.90       157
           E       0.92      0.96      0.94       129
           V       0.87      0.95      0.91       153
           Y       0.95      0.96      0.95       128
           Q       0.97      0.93      0.95       147
           U       0.90      0.89      0.90       117
           K       0.98      0.92      0.95       142
           Z       0.97      0.81      0.88       137

    accuracy                           0.90      3565
   macro avg       0.90      0.90      0.90      3565
weighted avg       0.90      0.90      0.90      3565
```

[32]:
```
model = Sequential()
model.add(Dense(128, activation='relu'))
model.add(Dense(64, activation='relu'))
model.add(Dense(32, activation='relu'))
model.add(Dense(len(label_encoder.classes_), activation='softmax'))

# Compile the model
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy',␣
 ↪metrics=['accuracy'])

# Train the model
model.fit(x_train,y_train, epochs=50, batch_size=32, validation_split=0.2)
```

```
Epoch 1/50
357/357             6s 11ms/step -
accuracy: 0.2466 - loss: 2.7189 - val_accuracy: 0.6280 - val_loss: 1.3616
Epoch 2/50
357/357             3s 5ms/step -
accuracy: 0.6493 - loss: 1.2684 - val_accuracy: 0.6701 - val_loss: 1.1147
Epoch 3/50
357/357             3s 5ms/step -
accuracy: 0.7172 - loss: 0.9907 - val_accuracy: 0.7051 - val_loss: 0.9362
Epoch 4/50
357/357             1s 2ms/step -
accuracy: 0.7569 - loss: 0.8457 - val_accuracy: 0.7542 - val_loss: 0.7985
Epoch 5/50
357/357             1s 2ms/step -
accuracy: 0.7752 - loss: 0.7418 - val_accuracy: 0.7700 - val_loss: 0.7231
Epoch 6/50
357/357             1s 2ms/step -
accuracy: 0.7994 - loss: 0.6789 - val_accuracy: 0.8040 - val_loss: 0.6397
Epoch 7/50
357/357             1s 2ms/step -
accuracy: 0.8130 - loss: 0.6095 - val_accuracy: 0.8012 - val_loss: 0.6300
Epoch 8/50
357/357             1s 2ms/step -
accuracy: 0.8358 - loss: 0.5385 - val_accuracy: 0.8170 - val_loss: 0.5705
Epoch 9/50
357/357             1s 4ms/step -
accuracy: 0.8421 - loss: 0.5145 - val_accuracy: 0.8215 - val_loss: 0.5574
Epoch 10/50
357/357             1s 4ms/step -
accuracy: 0.8514 - loss: 0.4645 - val_accuracy: 0.8415 - val_loss: 0.5198
Epoch 11/50
357/357             1s 3ms/step -
accuracy: 0.8662 - loss: 0.4455 - val_accuracy: 0.8538 - val_loss: 0.4617
Epoch 12/50
```

```
357/357             1s 2ms/step -
accuracy: 0.8693 - loss: 0.4196 - val_accuracy: 0.8520 - val_loss: 0.4578
Epoch 13/50
357/357             1s 2ms/step -
accuracy: 0.8823 - loss: 0.3794 - val_accuracy: 0.8661 - val_loss: 0.4232
Epoch 14/50
357/357             1s 2ms/step -
accuracy: 0.8823 - loss: 0.3777 - val_accuracy: 0.8640 - val_loss: 0.4093
Epoch 15/50
357/357             1s 2ms/step -
accuracy: 0.8916 - loss: 0.3568 - val_accuracy: 0.8657 - val_loss: 0.4245
Epoch 16/50
357/357             1s 2ms/step -
accuracy: 0.9006 - loss: 0.3187 - val_accuracy: 0.8836 - val_loss: 0.3704
Epoch 17/50
357/357             1s 2ms/step -
accuracy: 0.9082 - loss: 0.2990 - val_accuracy: 0.8692 - val_loss: 0.3937
Epoch 18/50
357/357             1s 2ms/step -
accuracy: 0.9101 - loss: 0.2907 - val_accuracy: 0.8917 - val_loss: 0.3334
Epoch 19/50
357/357             1s 2ms/step -
accuracy: 0.9152 - loss: 0.2615 - val_accuracy: 0.8948 - val_loss: 0.3323
Epoch 20/50
357/357             1s 2ms/step -
accuracy: 0.9145 - loss: 0.2643 - val_accuracy: 0.8899 - val_loss: 0.3492
Epoch 21/50
357/357             1s 4ms/step -
accuracy: 0.9174 - loss: 0.2522 - val_accuracy: 0.8734 - val_loss: 0.3618
Epoch 22/50
357/357             1s 4ms/step -
accuracy: 0.9248 - loss: 0.2409 - val_accuracy: 0.8973 - val_loss: 0.3215
Epoch 23/50
357/357             2s 2ms/step -
accuracy: 0.9264 - loss: 0.2268 - val_accuracy: 0.8994 - val_loss: 0.3065
Epoch 24/50
357/357             1s 2ms/step -
accuracy: 0.9253 - loss: 0.2311 - val_accuracy: 0.8931 - val_loss: 0.3156
Epoch 25/50
357/357             1s 2ms/step -
accuracy: 0.9292 - loss: 0.2183 - val_accuracy: 0.8938 - val_loss: 0.3359
Epoch 26/50
357/357             1s 2ms/step -
accuracy: 0.9342 - loss: 0.2042 - val_accuracy: 0.9071 - val_loss: 0.2782
Epoch 27/50
357/357             1s 2ms/step -
accuracy: 0.9270 - loss: 0.2091 - val_accuracy: 0.8941 - val_loss: 0.3263
Epoch 28/50
```

```
357/357             1s 2ms/step -
accuracy: 0.9285 - loss: 0.2171 - val_accuracy: 0.9165 - val_loss: 0.2855
Epoch 29/50
357/357             1s 2ms/step -
accuracy: 0.9320 - loss: 0.1989 - val_accuracy: 0.9092 - val_loss: 0.2712
Epoch 30/50
357/357             1s 2ms/step -
accuracy: 0.9391 - loss: 0.1855 - val_accuracy: 0.9022 - val_loss: 0.3023
Epoch 31/50
357/357             1s 2ms/step -
accuracy: 0.9357 - loss: 0.1888 - val_accuracy: 0.9148 - val_loss: 0.2608
Epoch 32/50
357/357             2s 4ms/step -
accuracy: 0.9409 - loss: 0.1766 - val_accuracy: 0.9109 - val_loss: 0.2763
Epoch 33/50
357/357             2s 2ms/step -
accuracy: 0.9454 - loss: 0.1713 - val_accuracy: 0.9225 - val_loss: 0.2364
Epoch 34/50
357/357             1s 2ms/step -
accuracy: 0.9492 - loss: 0.1568 - val_accuracy: 0.9102 - val_loss: 0.2730
Epoch 35/50
357/357             1s 2ms/step -
accuracy: 0.9399 - loss: 0.1746 - val_accuracy: 0.9158 - val_loss: 0.2441
Epoch 36/50
357/357             1s 2ms/step -
accuracy: 0.9460 - loss: 0.1537 - val_accuracy: 0.9043 - val_loss: 0.2726
Epoch 37/50
357/357             1s 2ms/step -
accuracy: 0.9441 - loss: 0.1593 - val_accuracy: 0.9194 - val_loss: 0.2503
Epoch 38/50
357/357             1s 2ms/step -
accuracy: 0.9512 - loss: 0.1476 - val_accuracy: 0.9208 - val_loss: 0.2303
Epoch 39/50
357/357             1s 2ms/step -
accuracy: 0.9464 - loss: 0.1576 - val_accuracy: 0.9116 - val_loss: 0.2924
Epoch 40/50
357/357             1s 2ms/step -
accuracy: 0.9491 - loss: 0.1457 - val_accuracy: 0.9183 - val_loss: 0.2478
Epoch 41/50
357/357             1s 2ms/step -
accuracy: 0.9567 - loss: 0.1385 - val_accuracy: 0.9134 - val_loss: 0.2493
Epoch 42/50
357/357             2s 4ms/step -
accuracy: 0.9491 - loss: 0.1527 - val_accuracy: 0.9197 - val_loss: 0.2439
Epoch 43/50
357/357             2s 2ms/step -
accuracy: 0.9506 - loss: 0.1381 - val_accuracy: 0.9194 - val_loss: 0.2358
Epoch 44/50
```

```
357/357             1s 2ms/step -
accuracy: 0.9541 - loss: 0.1300 - val_accuracy: 0.9225 - val_loss: 0.2430
Epoch 45/50
357/357             1s 2ms/step -
accuracy: 0.9512 - loss: 0.1426 - val_accuracy: 0.9274 - val_loss: 0.2199
Epoch 46/50
357/357             1s 2ms/step -
accuracy: 0.9572 - loss: 0.1240 - val_accuracy: 0.9316 - val_loss: 0.2266
Epoch 47/50
357/357             1s 2ms/step -
accuracy: 0.9521 - loss: 0.1349 - val_accuracy: 0.9173 - val_loss: 0.2578
Epoch 48/50
357/357             1s 2ms/step -
accuracy: 0.9538 - loss: 0.1399 - val_accuracy: 0.9218 - val_loss: 0.2358
Epoch 49/50
357/357             1s 2ms/step -
accuracy: 0.9584 - loss: 0.1193 - val_accuracy: 0.9243 - val_loss: 0.2434
Epoch 50/50
357/357             1s 2ms/step -
accuracy: 0.9612 - loss: 0.1131 - val_accuracy: 0.9197 - val_loss: 0.2600
```

[32]:
```
<keras.src.callbacks.history.History at 0x7be622632e00>
```

[33]:
```
# Define expected unique labels (based on your previous output)
expected_labels = ['T', 'I', 'D', 'N', 'G', 'S', 'B', 'A', 'J', 'M', 'X', 'O',␣
 ↪'R', 'F', 'C', 'H', 'W', 'L', 'P', 'E', 'V', 'Y', 'Q', 'U', 'K', 'Z']

# Evaluate the model
y_pred_prob = model.predict(x_test)
y_pred_encoded = np.argmax(y_pred_prob, axis=1)

# Convert predicted labels back to original letters using inverse_transform
y_test_pred = label_encoder.inverse_transform(y_test)
y_pred_letters = label_encoder.inverse_transform(y_pred_encoded)
```

```
112/112             0s 2ms/step
```

[ ]:
```
```

[ ]:
```
# Generate classification report
report = classification_report(y_test_pred, y_pred_letters,␣
 ↪target_names=expected_labels)

# Print classification report
print("Classification Report:")
print(report)
```

[37]:
```
!pip install scikeras
```

```
Collecting scikeras
  Downloading scikeras-0.13.0-py3-none-any.whl.metadata (3.1 kB)
Requirement already satisfied: keras>=3.2.0 in /usr/local/lib/python3.10/dist-
packages (from scikeras) (3.5.0)
Requirement already satisfied: scikit-learn>=1.4.2 in
/usr/local/lib/python3.10/dist-packages (from scikeras) (1.5.2)
Requirement already satisfied: absl-py in /usr/local/lib/python3.10/dist-
packages (from keras>=3.2.0->scikeras) (1.4.0)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages
(from keras>=3.2.0->scikeras) (1.26.4)
Requirement already satisfied: rich in /usr/local/lib/python3.10/dist-packages
(from keras>=3.2.0->scikeras) (13.9.4)
Requirement already satisfied: namex in /usr/local/lib/python3.10/dist-packages
(from keras>=3.2.0->scikeras) (0.0.8)
Requirement already satisfied: h5py in /usr/local/lib/python3.10/dist-packages
(from keras>=3.2.0->scikeras) (3.12.1)
Requirement already satisfied: optree in /usr/local/lib/python3.10/dist-packages
(from keras>=3.2.0->scikeras) (0.13.1)
Requirement already satisfied: ml-dtypes in /usr/local/lib/python3.10/dist-
packages (from keras>=3.2.0->scikeras) (0.4.1)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-
packages (from keras>=3.2.0->scikeras) (24.2)
Requirement already satisfied: scipy>=1.6.0 in /usr/local/lib/python3.10/dist-
packages (from scikit-learn>=1.4.2->scikeras) (1.13.1)
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.10/dist-
packages (from scikit-learn>=1.4.2->scikeras) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in
/usr/local/lib/python3.10/dist-packages (from scikit-learn>=1.4.2->scikeras)
(3.5.0)
Requirement already satisfied: typing-extensions>=4.5.0 in
/usr/local/lib/python3.10/dist-packages (from optree->keras>=3.2.0->scikeras)
(4.12.2)
Requirement already satisfied: markdown-it-py>=2.2.0 in
/usr/local/lib/python3.10/dist-packages (from rich->keras>=3.2.0->scikeras)
(3.0.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in
/usr/local/lib/python3.10/dist-packages (from rich->keras>=3.2.0->scikeras)
(2.18.0)
Requirement already satisfied: mdurl~=0.1 in /usr/local/lib/python3.10/dist-
packages (from markdown-it-py>=2.2.0->rich->keras>=3.2.0->scikeras) (0.1.2)
Downloading scikeras-0.13.0-py3-none-any.whl (26 kB)
Installing collected packages: scikeras
Successfully installed scikeras-0.13.0
```

[38]:
```
#from tensorflow.keras.wrappers.scikit_learn import KerasClassifier  # Remove␣
 ↪this line
from scikeras.wrappers import KerasClassifier  # Import from scikeras instead
from sklearn.metrics import accuracy_score, make_scorer
from sklearn.model_selection import train_test_split, GridSearchCV
```

[40]:
```
def create_model(optimizer='adam', activation='relu', neurons=64,␣
 ↪hidden_layers=1):
    model = Sequential()
    model.add(Dense(neurons, activation=activation, input_shape=(x_train.
 ↪shape[1],)))
    model.add(Dense(32, activation='relu'))
    for _ in range(hidden_layers - 1):
        model.add(Dense(neurons, activation=activation))

    model.add(Dense(26, activation='softmax'))

    model.compile(optimizer=optimizer, loss='sparse_categorical_crossentropy',␣
 ↪metrics=['accuracy'])
    return model

# Create KerasClassifier based on the create_model function
model = KerasClassifier(build_fn=create_model, verbose=0)

# Define hyperparameters grid for GridSearchCV
param_grid = {
    'model__optimizer': ['adam', 'rmsprop'],
    'model__activation': ['relu', 'tanh'],
    'model__neurons': [32, 64, 128],
    'model__hidden_layers': [1, 2]
}

# Define accuracy as the scoring metric for GridSearchCV
scoring = {'accuracy': make_scorer(accuracy_score)}

# Perform GridSearchCV for hyperparameter tuning
grid_search = GridSearchCV(estimator=model, param_grid=param_grid,␣
 ↪scoring=scoring, refit='accuracy', cv=3)
grid_result = grid_search.fit(x_train, y_train)
```

```
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
```

```
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
```

```
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
```

```
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
```

```
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
```

```
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
```

```
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
```

```
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
```

```
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
```

```
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
```

```
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/scikeras/wrappers.py:925: UserWarning:
``build_fn`` will be renamed to ``model`` in a future release, at which point
use of ``build_fn`` will raise an Error instead.
  X, y = self._initialize(X, y)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

[41]:
```
# Print best hyperparameters and corresponding accuracy
print("Best Accuracy: {:.4f}".format(grid_result.best_score_))
print("Best Parameters:", grid_result.best_params_)

# Evaluate the best model on the test set
best_model = grid_result.best_estimator_
y_pred = best_model.predict(x_test)
test_accuracy = accuracy_score(y_test, y_pred)
print("Test Accuracy: {:.4f}".format(test_accuracy))
```

```
Best Accuracy: 0.6725
Best Parameters: {'model__activation': 'tanh', 'model__hidden_layers': 2,
'model__neurons': 128, 'model__optimizer': 'adam'}
Test Accuracy: 0.7086
```

[42]:
```
print(classification_report(y_test, y_pred))
```

```
              precision    recall  f1-score   support

           0       0.94      0.84      0.89       140
           1       0.48      0.84      0.61       129
```

2	0.75	0.66	0.70	138
3	0.67	0.71	0.69	142
4	0.57	0.64	0.60	129
5	0.71	0.74	0.73	149
6	0.48	0.48	0.48	125
7	0.75	0.51	0.61	152
8	0.76	0.73	0.74	73
9	0.84	0.77	0.80	114
10	0.54	0.81	0.65	132
11	0.86	0.80	0.83	127
12	0.69	0.92	0.79	135
13	0.93	0.59	0.72	133
14	0.72	0.68	0.70	164
15	0.78	0.82	0.80	157
16	0.87	0.55	0.67	173
17	0.54	0.77	0.64	143
18	0.55	0.26	0.35	157
19	0.73	0.79	0.76	129
20	0.89	0.85	0.87	153
21	0.84	0.88	0.86	128
22	0.73	0.90	0.81	147
23	0.73	0.54	0.62	117
24	0.86	0.75	0.80	142
25	0.64	0.66	0.65	137
accuracy			0.71	3565
macro avg	0.73	0.71	0.71	3565
weighted avg	0.73	0.71	0.71	3565

[ ]: