INM427 Neural Computing - Individual Project

Sahan Chowdhury

This project compares two NECO methods Multilayer perceptron (mlp) and Support vector machines (SVM) on a pumpkin seed classification dataset

Dataset link : https://www.muratkoklu.com/datasets/

```
#libraries required

import random
import time

import pandas as pd
import numpy as np
import pickle

import scipy
from scipy.stats import boxcox

import matplotlib.pyplot as plt
import seaborn as sns
from pandas.plotting import radviz

from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix, roc_curve, auc
from sklearn.preprocessing import MinMaxScaler
from sklearn.svm import SVC
from sklearn.metrics import average_precision_score
from sklearn.metrics import precision_recall_curve

import torch
import torch.nn as nn
import torch.optim as optim
import torch.nn.functional as F

import skorch
from skorch import NeuralNetClassifier
from skorch.callbacks import EarlyStopping, EpochScoring


#Reproducibility
#Reproduciblity using the random module
random.seed(1)
#Setting random seed for pytorch library
torch.manual_seed(1)
#Ensuring reproducility for numpy arrays
np.random.seed(1)
```

```
#Importing the dataset from an excel file into pandas dataframe
df = pd.read_excel('Pumpkin_Seeds_Dataset.xlsx', sheet_name=0)
#The dataset was stored in a excel file and sheet name was 0
#Displaying the data in a pandas dataframe called 'df'
df
```

```
       Area  Perimeter  Major_Axis_Length  Minor_Axis_Length
Convex_Area  \
0     56276    888.242           326.1485           220.2388
56831
1     76631   1068.146           417.1932           234.2289
77280
2     71623   1082.987           435.8328           211.0457
72663
3     66458    992.051           381.5638           222.5322
67118
4     66107    998.146           383.8883           220.4545
67117
...     ...        ...                ...                ...
...
2495  79637   1224.710           533.1513           190.4367
80381
2496  69647   1084.318           462.9416           191.8210
70216
2497  87994   1210.314           507.2200           222.1872
88702
2498  80011   1182.947           501.9065           204.7531
80902
2499  84934   1159.933           462.8951           234.5597
85781

      Equiv_Diameter  Eccentricity  Solidity  Extent  Roundness  \
0           267.6805        0.7376    0.9902  0.7453     0.8963
1           312.3614        0.8275    0.9916  0.7151     0.8440
2           301.9822        0.8749    0.9857  0.7400     0.7674
3           290.8899        0.8123    0.9902  0.7396     0.8486
4           290.1207        0.8187    0.9850  0.6752     0.8338
...              ...           ...       ...     ...        ...
2495        318.4289        0.9340    0.9907  0.4888     0.6672
2496        297.7874        0.9101    0.9919  0.6002     0.7444
2497        334.7199        0.8990    0.9920  0.7643     0.7549
2498        319.1758        0.9130    0.9890  0.7374     0.7185
2499        328.8485        0.8621    0.9901  0.7360     0.7933

      Aspect_Ration  Compactness          Class
0            1.4809       0.8207     Çerçevelik
1            1.7811       0.7487     Çerçevelik
2            2.0651       0.6929     Çerçevelik
3            1.7146       0.7624     Çerçevelik
4            1.7413       0.7557     Çerçevelik
```

```
...              ...          ...             ...
2495         2.7996       0.5973  Ürgüp Sivrisi
2496         2.4134       0.6433  Ürgüp Sivrisi
2497         2.2828       0.6599  Ürgüp Sivrisi
2498         2.4513       0.6359  Ürgüp Sivrisi
2499         1.9735       0.7104  Ürgüp Sivrisi

[2500 rows x 13 columns]
```

```
#Reference [1]:  https://www.freecodecamp.org/news/dataframe-to-csv-how-to-save-pandas-dataframes-by-exporting/
# Saving the pumpkin seed data to a CSV file
df.to_csv('Pumpkin_Seeds_Dataset.csv', index=False)

#Displaying all the column names
df.columns.values
```

```
array(['Area', 'Perimeter', 'Major_Axis_Length', 'Minor_Axis_Length',
       'Convex_Area', 'Equiv_Diameter', 'Eccentricity', 'Solidity',
       'Extent', 'Roundness', 'Aspect_Ration', 'Compactness',
'Class'],
      dtype=object)
```

The dataset contains 2500 rows and 13 columns and its features are Area, Perimeter, Major_Axis_Length, Minor_Axis_Length, Convex_Area Equiv_Diameter, Eccentricity, Solidity, Extent, Roundness, Aspect_Ration, compactness, Class

```
#Obtaining the different classes of pumpkin seeds
df['Class'].unique()
```

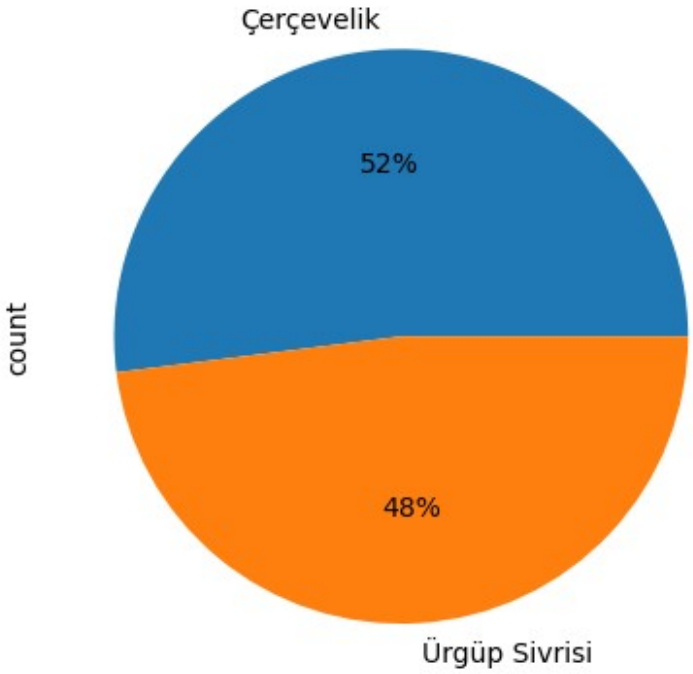
```
array(['Çerçevelik', 'Ürgüp Sivrisi'], dtype=object)
```

```
#Counts of each pumpkin seed class
df['Class'].value_counts()
```

```
Class
Çerçevelik       1300
Ürgüp Sivrisi    1200
Name: count, dtype: int64
```

```
#Visualising class dominance on pie chart
df['Class'].value_counts().plot.pie(autopct='%1.0f%%')
```

```
<Axes: ylabel='count'>
```




```
print("Çerçevelik is",52/48,'Greater than Ürgüp Sivrisi ')
```

```
Çerçevelik is 1.0833333333333333 Greater than Ürgüp Sivrisi
```

The dataset contains 2 unique classes of pumpkin seed types which are 'Çerçevelik' and 'Ürgüp Sivrisi'. 'Çerçevelik' has a count of 1300 and 'Ürgüp Sivrisi' has a count of 1200. 'Çerçevelik' is the dominant class

Data Preprocessing

The 2 types of of pumpkin seed which are Çerçevelik and Ürgüp Sivrisi, we will encode Çerçevelik as 1 and Ürgüp Sivrisi as 0

```
#Reference [2] : https://stackoverflow.com/questions/73371405/remap-values-in-pandas-column-with-a-dict-none-if-keyerror
#Converting into binary class of 1 and 0 corresponding to Çerçevelik and Ürgüp Sivrisi
df['Class']= df['Class'].replace({'Çerçevelik': 1, 'Ürgüp Sivrisi': 0})


#Printing data types for the variables in the dataset to gain insight into structure and characterstics
df.dtypes
```

```
Area                 int64
Perimeter          float64
```

```
Major_Axis_Length    float64
Minor_Axis_Length    float64
Convex_Area            int64
Equiv_Diameter       float64
Eccentricity         float64
Solidity             float64
Extent               float64
Roundness            float64
Aspect_Ration        float64
Compactness          float64
Class                  int64
dtype: object
```

There is difference amongst some variable types and all will be converted to float64 to ensure consistency as well precision for when training our models

```
#Converting all the variables into float for consistency. Also float
is more precise and is preffered for activation functions and
normalisation later on.
df = df.astype('float64')
df.dtypes
```

```
Area                 float64
Perimeter            float64
Major_Axis_Length    float64
Minor_Axis_Length    float64
Convex_Area          float64
Equiv_Diameter       float64
Eccentricity         float64
Solidity             float64
Extent               float64
Roundness            float64
Aspect_Ration        float64
Compactness          float64
Class                float64
dtype: object
```

```
df.describe()
```

```
                Area    Perimeter  Major_Axis_Length
Minor_Axis_Length  \
count    2500.000000  2500.000000        2500.000000
2500.000000
mean    80658.220800  1130.279015         456.601840
225.794921
std     13664.510228   109.256418          56.235704
23.297245
min     47939.000000   868.485000         320.844600
152.171800
25%     70765.000000  1048.829750         414.957850
```

```
211.245925
50%     79076.000000  1123.672000          449.496600
224.703100
75%     89757.500000  1203.340500          492.737650
240.672875
max    136574.000000  1559.450000          661.911300
305.818000

         Convex_Area  Equiv_Diameter  Eccentricity     Solidity
Extent  \
count    2500.000000     2500.000000   2500.000000  2500.000000
2500.000000
mean    81508.084400      319.334230      0.860879     0.989492
0.693205
std     13764.092788       26.891920      0.045167     0.003494
0.060914
min     48366.000000      247.058400      0.492100     0.918600
0.468000
25%     71512.000000      300.167975      0.831700     0.988300
0.658900
50%     79872.000000      317.305350      0.863700     0.990300
0.713050
75%     90797.750000      338.057375      0.897025     0.991500
0.740225
max    138384.000000      417.002900      0.948100     0.994400
0.829600

         Roundness  Aspect_Ration  Compactness      Class
count  2500.000000    2500.000000  2500.000000  2500.0000
mean      0.791533       2.041702     0.704121     0.5200
std       0.055924       0.315997     0.053067     0.4997
min       0.554600       1.148700     0.560800     0.0000
25%       0.751900       1.801050     0.663475     0.0000
50%       0.797750       1.984200     0.707700     1.0000
75%       0.834325       2.262075     0.743500     1.0000
max       0.939600       3.144400     0.904900     1.0000
```

The pumpkin seeds have an average area of 80,658.

```
# Checking for missing values
df.isnull().sum()

Area                0
Perimeter           0
Major_Axis_Length   0
Minor_Axis_Length   0
Convex_Area         0
Equiv_Diameter      0
Eccentricity        0
```

```
Solidity              0
Extent                0
Roundness             0
Aspect_Ration         0
Compactness           0
Class                 0
dtype: int64
```

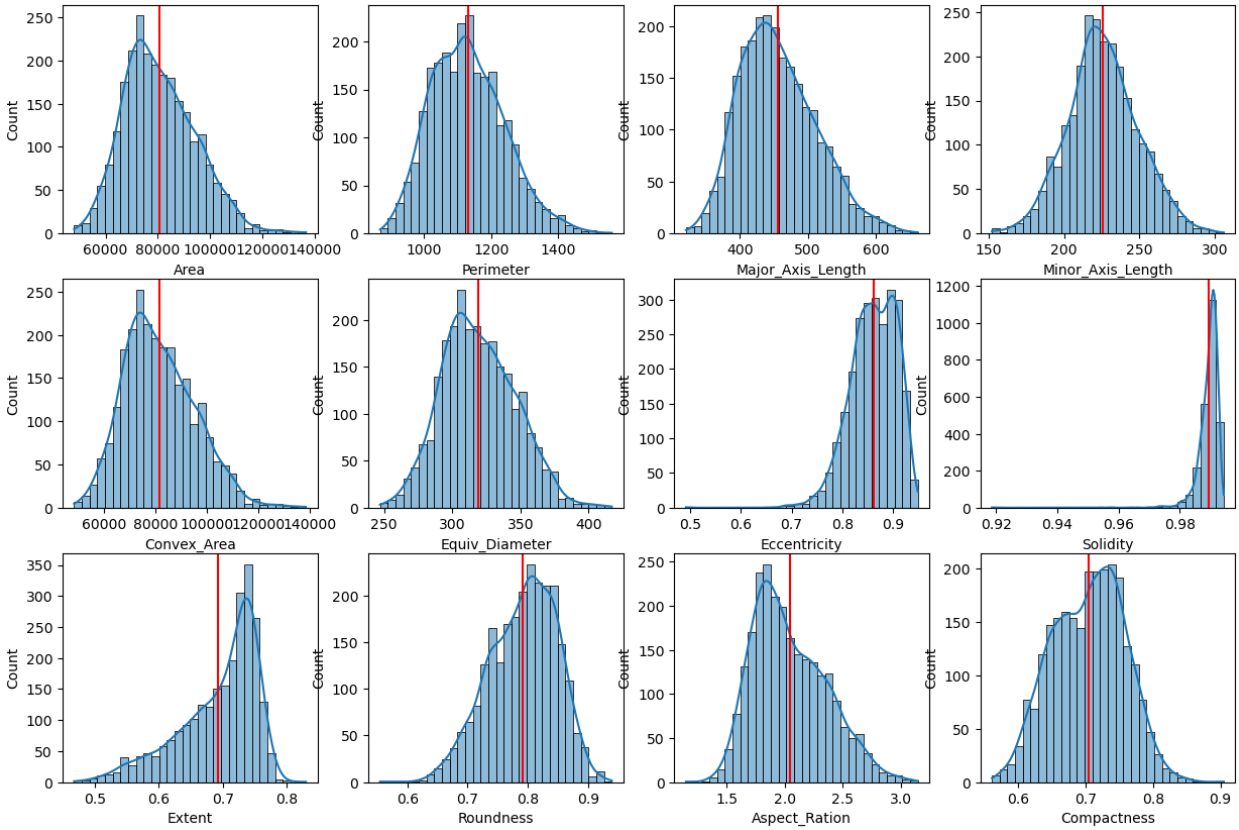
There are no missing values

```
#Creating a copy of the orginal dataframe (df) in which we will make
changes and used later on
data = df.copy()

#Creating a target variable which will contain all the features
excluding the target column ('class')
feature_columns = df.drop(columns=['Class']).columns
feature_columns
```

```
Index(['Area', 'Perimeter', 'Major_Axis_Length', 'Minor_Axis_Length',
       'Convex_Area', 'Equiv_Diameter', 'Eccentricity', 'Solidity',
'Extent',
       'Roundness', 'Aspect_Ration', 'Compactness'],
      dtype='object')
```

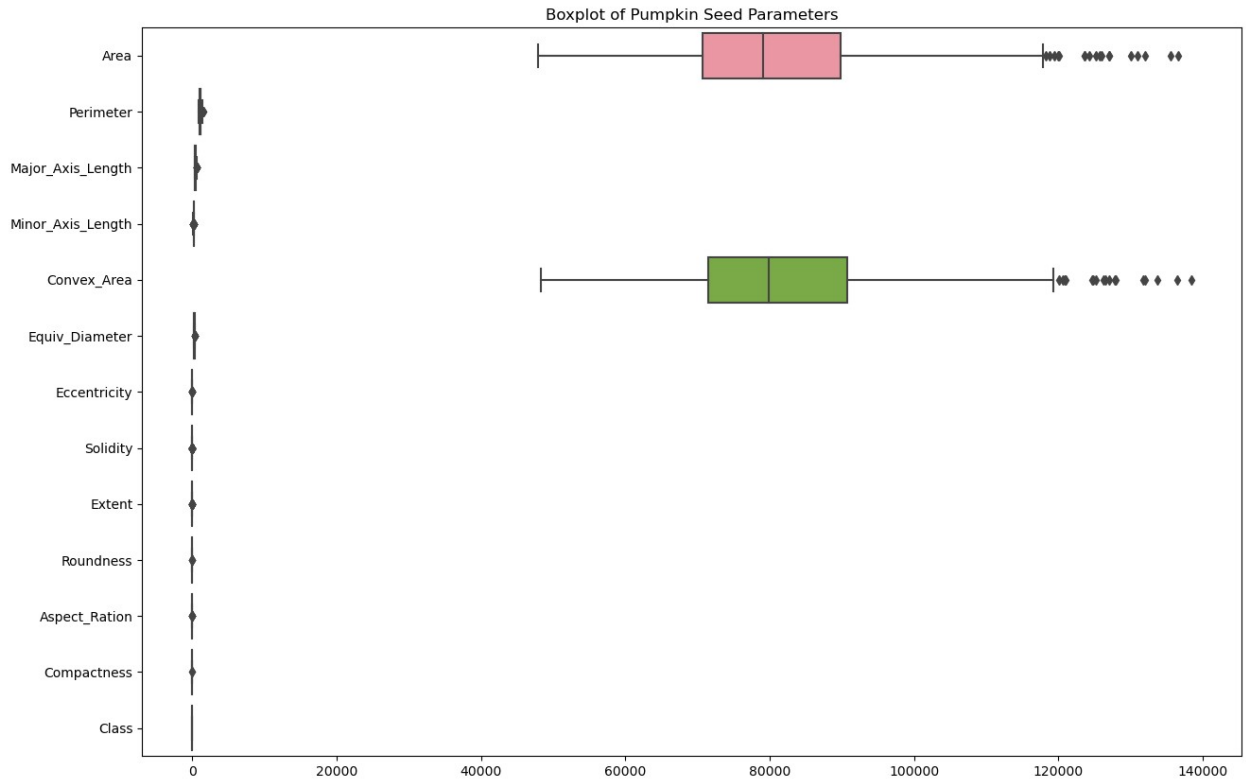
Data Visualisation

```
#Reference [3]: https://github.com/vighnesh32/Neural-Computing-
Project/blob/main/project.ipynb

#Creating subplots for each variable in the dataframe to visualise the
distribution

#12 features (3 rows and 4 columns)
fig, ax = plt.subplots(3, 4, figsize=(15, 10))
# Flattening the axes array to a 1D array
axes_list = ax.flatten()
#Creating a for loop to loop through each feature
for feature_column, subplot in zip(df.columns, axes_list):
    #Using seaborn librabry to plot graphs for every feature
    ax = sns.histplot(df[feature_column], bins=30, kde=True,
ax=subplot)
    #Creating a mean line for every subplot
    ax.axvline(df[feature_column].mean(), color='red')
plt.show()
```



Majority of the features have no skew however 'Eccentricity' and 'solidity' are left skwewed

```
#Boxplots to visualise distribution and outliers
plt.figure(figsize=(15, 10))
sns.boxplot(df, orient = 'h')
plt.title('Boxplot of Pumpkin Seed Parameters')
plt.show()
```



Due to vast range its nessesary to normalise the data to visualise box plots more clearly, Therfore we will use z scores to detect outliers

```
#Detecting outliers using z scores
# Setting thresholds for outlier detection for each feature
thresholds = {'Area': 3.5,'Perimeter': 3.0,'Major_Axis_Length':
6,'Minor_Axis_Length': 3.5,'Eccentricity': 3.5,'Convex_Area':
3.5,'Equiv_Diameter': 3.5,'Extent': 3.5,'Roundness':
3.5,'Aspect_Ration': 3.5,'Solidity': 3,'Compactness': 3,}

#creating a variable to store all the outliers
outliers = {}

# For loop to iterate over every variable and its threshold z score to
detect outliers
for column, threshold in thresholds.items():
    # Calculate Z-scores for the current feature by obtaining total of
the feature minus the mean and then divided by the standard deviation
of the feature
    z_scores = (data[column] - data[column].mean()) /
data[column].std()

    #If absolute value is greater than the set threshold of the
feature it stores in the outlier list
    outliers[column] = data[np.abs(z_scores) > threshold]
```

```python
# Display outliers for each feature
for column, outlier_data in outliers.items():
    print(f"Outliers for '{column}':")
    print(outlier_data)
```

```
Outliers for 'Area':
          Area  Perimeter  Major_Axis_Length  Minor_Axis_Length
Convex_Area  \
1583  130913.0   1490.954           632.2535           264.1584
131934.0
1854  132035.0   1520.525           640.1907           265.3734
133706.0
1991  135455.0   1451.905           580.8759           297.7952
136373.0
2045  136574.0   1559.450           661.9113           267.3850
138384.0
2373  130071.0   1491.946           621.7082           267.3021
131713.0

      Equiv_Diameter  Eccentricity  Solidity  Extent  Roundness  \
1583        408.2690        0.9085    0.9923  0.5472     0.7401
1854        410.0149        0.9100    0.9875  0.5606     0.7176
1991        415.2911        0.8586    0.9933  0.6928     0.8075
2045        417.0029        0.9148    0.9869  0.5313     0.7057
2373        406.9540        0.9029    0.9875  0.5943     0.7343

      Aspect_Ration  Compactness  Class
1583         2.3935       0.6457    0.0
1854         2.4124       0.6405    0.0
1991         1.9506       0.7149    0.0
2045         2.4755       0.6300    0.0
2373         2.3259       0.6546    0.0
Outliers for 'Perimeter':
          Area  Perimeter  Major_Axis_Length  Minor_Axis_Length
Convex_Area  \
1301  118751.0   1468.224           629.7230           240.9782
120036.0
1583  130913.0   1490.954           632.2535           264.1584
131934.0
1654  125214.0   1465.654           623.0155           256.4077
126196.0
1708  126963.0   1476.738           625.3347           259.2516
127781.0
1854  132035.0   1520.525           640.1907           265.3734
133706.0
1880  120014.0   1492.183           648.9984           236.2167
121003.0
2045  136574.0   1559.450           661.9113           267.3850
```

```
138384.0
2373  130071.0   1491.946           621.7082           267.3021
131713.0

      Equiv_Diameter  Eccentricity  Solidity  Extent  Roundness  \
1301        388.8425        0.9239    0.9893  0.7440     0.6922
1583        408.2690        0.9085    0.9923  0.5472     0.7401
1654        399.2836        0.9114    0.9922  0.7464     0.7325
1708        402.0626        0.9100    0.9936  0.6485     0.7316
1854        410.0149        0.9100    0.9875  0.5606     0.7176
1880        390.9048        0.9314    0.9918  0.5495     0.6773
2045        417.0029        0.9148    0.9869  0.5313     0.7057
2373        406.9540        0.9029    0.9875  0.5943     0.7343

      Aspect_Ration  Compactness  Class
1301         2.6132       0.6175    0.0
1583         2.3935       0.6457    0.0
1654         2.4298       0.6409    0.0
1708         2.4121       0.6430    0.0
1854         2.4124       0.6405    0.0
1880         2.7475       0.6023    0.0
2045         2.4755       0.6300    0.0
2373         2.3259       0.6546    0.0
Outliers for 'Major_Axis_Length':
Empty DataFrame
Columns: [Area, Perimeter, Major_Axis_Length, Minor_Axis_Length,
Convex_Area, Equiv_Diameter, Eccentricity, Solidity, Extent,
Roundness, Aspect_Ration, Compactness, Class]
Index: []
Outliers for 'Minor_Axis_Length':
Empty DataFrame
Columns: [Area, Perimeter, Major_Axis_Length, Minor_Axis_Length,
Convex_Area, Equiv_Diameter, Eccentricity, Solidity, Extent,
Roundness, Aspect_Ration, Compactness, Class]
Index: []
Outliers for 'Eccentricity':
         Area  Perimeter  Major_Axis_Length  Minor_Axis_Length
Convex_Area  \
303   93178.0   1124.779           405.6369           293.4700
93976.0
632   77905.0   1038.548           371.9293           267.6285
78770.0
653   69132.0    961.531           342.3836           257.6274
69616.0
1028  92432.0   1124.035           402.6564           292.9598
93267.0
1079  97634.0   1157.568           416.0763           300.5777
98478.0
1133  83188.0   1101.598           383.7482           278.4777
```

```
84738.0
1194  66204.0   1006.743           320.8446           279.3001
67295.0

      Equiv_Diameter  Eccentricity  Solidity  Extent  Roundness  \
303         344.4385        0.6903    0.9915  0.7755     0.9255
632         314.9472        0.6944    0.9890  0.7455     0.9077
653         296.6843        0.6586    0.9930  0.7391     0.9396
1028        343.0570        0.6860    0.9910  0.7606     0.9193
1079        352.5783        0.6915    0.9914  0.7190     0.9156
1133        325.4508        0.6880    0.9817  0.7140     0.8614
1194        290.3335        0.4921    0.9838  0.6315     0.8208

      Aspect_Ration  Compactness  Class
303          1.3822       0.8491    1.0
632          1.3897       0.8468    1.0
653          1.3290       0.8665    1.0
1028         1.3744       0.8520    1.0
1079         1.3843       0.8474    1.0
1133         1.3780       0.8481    1.0
1194         1.1487       0.9049    1.0
Outliers for 'Convex_Area':
          Area  Perimeter  Major_Axis_Length  Minor_Axis_Length
Convex_Area  \
1583  130913.0   1490.954           632.2535           264.1584
131934.0
1854  132035.0   1520.525           640.1907           265.3734
133706.0
1991  135455.0   1451.905           580.8759           297.7952
136373.0
2045  136574.0   1559.450           661.9113           267.3850
138384.0
2373  130071.0   1491.946           621.7082           267.3021
131713.0

      Equiv_Diameter  Eccentricity  Solidity  Extent  Roundness  \
1583        408.2690        0.9085    0.9923  0.5472     0.7401
1854        410.0149        0.9100    0.9875  0.5606     0.7176
1991        415.2911        0.8586    0.9933  0.6928     0.8075
2045        417.0029        0.9148    0.9869  0.5313     0.7057
2373        406.9540        0.9029    0.9875  0.5943     0.7343

      Aspect_Ration  Compactness  Class
1583         2.3935       0.6457    0.0
1854         2.4124       0.6405    0.0
1991         1.9506       0.7149    0.0
2045         2.4755       0.6300    0.0
2373         2.3259       0.6546    0.0
Outliers for 'Equiv_Diameter':
          Area  Perimeter  Major_Axis_Length  Minor_Axis_Length
```

```
Convex_Area  \
1991  135455.0   1451.905           580.8759           297.7952
136373.0
2045  136574.0   1559.450           661.9113           267.3850
138384.0

      Equiv_Diameter  Eccentricity  Solidity  Extent  Roundness  \
1991        415.2911        0.8586    0.9933  0.6928     0.8075
2045        417.0029        0.9148    0.9869  0.5313     0.7057

      Aspect_Ration  Compactness  Class
1991         1.9506       0.7149    0.0
2045         2.4755       0.6300    0.0
Outliers for 'Extent':
         Area  Perimeter  Major_Axis_Length  Minor_Axis_Length
Convex_Area  \
1823  65691.0   1122.062           492.8912            170.134
66383.0
2061  90597.0   1352.215           605.5829            192.588
91557.0

      Equiv_Diameter  Eccentricity  Solidity  Extent  Roundness  \
1823        289.2065        0.9385    0.9896  0.4680     0.6557
2061        339.6346        0.9481    0.9895  0.4695     0.6226

      Aspect_Ration  Compactness  Class
1823         2.8971       0.5868    0.0
2061         3.1444       0.5608    0.0
Outliers for 'Roundness':
         Area  Perimeter  Major_Axis_Length  Minor_Axis_Length
Convex_Area  \
1376  55045.0   1116.756           398.7896            177.7192
56734.0
1725  65353.0   1187.425           507.2473            177.0789
71144.0

      Equiv_Diameter  Eccentricity  Solidity  Extent  Roundness  \
1376        264.7366        0.8952    0.9702  0.7954     0.5546
1725        288.4615        0.9371    0.9186  0.5411     0.5825

      Aspect_Ration  Compactness  Class
1376         2.2439       0.6639    0.0
1725         2.8645       0.5687    0.0
Outliers for 'Aspect_Ration':
Empty DataFrame
Columns: [Area, Perimeter, Major_Axis_Length, Minor_Axis_Length,
Convex_Area, Equiv_Diameter, Eccentricity, Solidity, Extent,
Roundness, Aspect_Ration, Compactness, Class]
Index: []
Outliers for 'Solidity':
```

```
         Area  Perimeter  Major_Axis_Length  Minor_Axis_Length
Convex_Area  \
48    67135.0   1031.821           368.1006           238.7819
68784.0
108   80920.0   1171.651           397.4495           262.3499
82943.0
128   64073.0   1033.250           405.1633           207.1816
67150.0
144   62584.0   1017.196           345.0958           233.8076
64781.0
154   67629.0   1043.160           394.4112           219.7025
69144.0
169   78571.0   1144.291           422.1389           239.1871
80655.0
187   90239.0   1298.350           538.0211           217.9732
92604.0
229   58347.0    976.686           350.2990           213.4355
60534.0
497   61589.0    989.151           389.2343           203.1030
62939.0
517   95914.0   1203.477           468.4100           269.2576
98178.0
561   75967.0   1118.878           424.7328           232.8153
78044.0
577   65694.0   1039.506           348.0645           243.9780
67364.0
617   68156.0   1064.093           387.2451           225.5328
69811.0
777   61567.0   1092.549           383.9640           207.7470
64254.0
840   95508.0   1224.636           446.7743           279.6244
98182.0
937   62615.0   1102.542           392.9788           206.4167
64422.0
1019  81137.0   1126.696           426.4425           244.0338
83064.0
1091  80288.0   1193.410           496.2107           207.5564
82781.0
1198  91652.0   1195.961           446.3604           262.7367
93654.0
1295  64276.0   1002.303           366.1481           226.6391
65696.0
1376  55045.0   1116.756           398.7896           177.7192
56734.0
1379  56899.0    997.835           405.4002           181.2477
58404.0
1470  84036.0   1201.495           500.2555           217.8894
86362.0
1725  65353.0   1187.425           507.2473           177.0789
```

```
71144.0
1807  49171.0    938.006           379.6055           166.5137
50268.0
2272  74143.0   1192.175           448.3368           212.2402
75809.0
2280  74525.0   1153.205           478.0779           200.0458
76573.0
2447  64783.0   1069.990           394.0979           214.5254
66488.0
2480  72140.0   1178.246           516.2589           184.5502
75406.0
```

	Equiv_Diameter	Eccentricity	Solidity	Extent	Roundness	\
48	292.3678	0.7611	0.9760	0.6867	0.7924	
108	320.9837	0.7512	0.9756	0.7258	0.7407	
128	285.6226	0.8594	0.9542	0.6974	0.7542	
144	282.2843	0.7355	0.9661	0.7357	0.7601	
154	293.4415	0.8305	0.9781	0.6970	0.7810	
169	316.2905	0.8240	0.9742	0.6673	0.7540	
187	338.9629	0.9143	0.9745	0.5071	0.6727	
229	272.5614	0.7929	0.9639	0.6359	0.7686	
497	280.0313	0.8531	0.9786	0.6249	0.7910	
517	349.4589	0.8183	0.9769	0.7389	0.8322	
561	311.0051	0.8364	0.9734	0.7621	0.7626	
577	289.2131	0.7132	0.9752	0.6930	0.7640	
617	294.5826	0.8129	0.9763	0.6553	0.7564	
777	279.9813	0.8410	0.9582	0.7029	0.6482	
840	348.7185	0.7799	0.9728	0.7724	0.8003	
937	282.3542	0.8509	0.9720	0.6337	0.6473	
1019	321.4138	0.8201	0.9768	0.7301	0.8032	
1091	319.7278	0.9083	0.9699	0.6049	0.7084	
1198	341.6064	0.8084	0.9786	0.7352	0.8052	
1295	286.0747	0.7854	0.9784	0.7404	0.8040	
1376	264.7366	0.8952	0.9702	0.7954	0.5546	
1379	269.1581	0.8945	0.9742	0.7651	0.7181	
1470	327.1054	0.9002	0.9731	0.7620	0.7315	
1725	288.4615	0.9371	0.9186	0.5411	0.5825	
1807	250.2128	0.8987	0.9782	0.6304	0.7023	
2272	307.2488	0.8809	0.9780	0.6325	0.6555	
2280	308.0392	0.9082	0.9733	0.5578	0.7042	
2447	287.2008	0.8389	0.9744	0.7678	0.7111	
2480	303.0701	0.9339	0.9567	0.5163	0.6530	

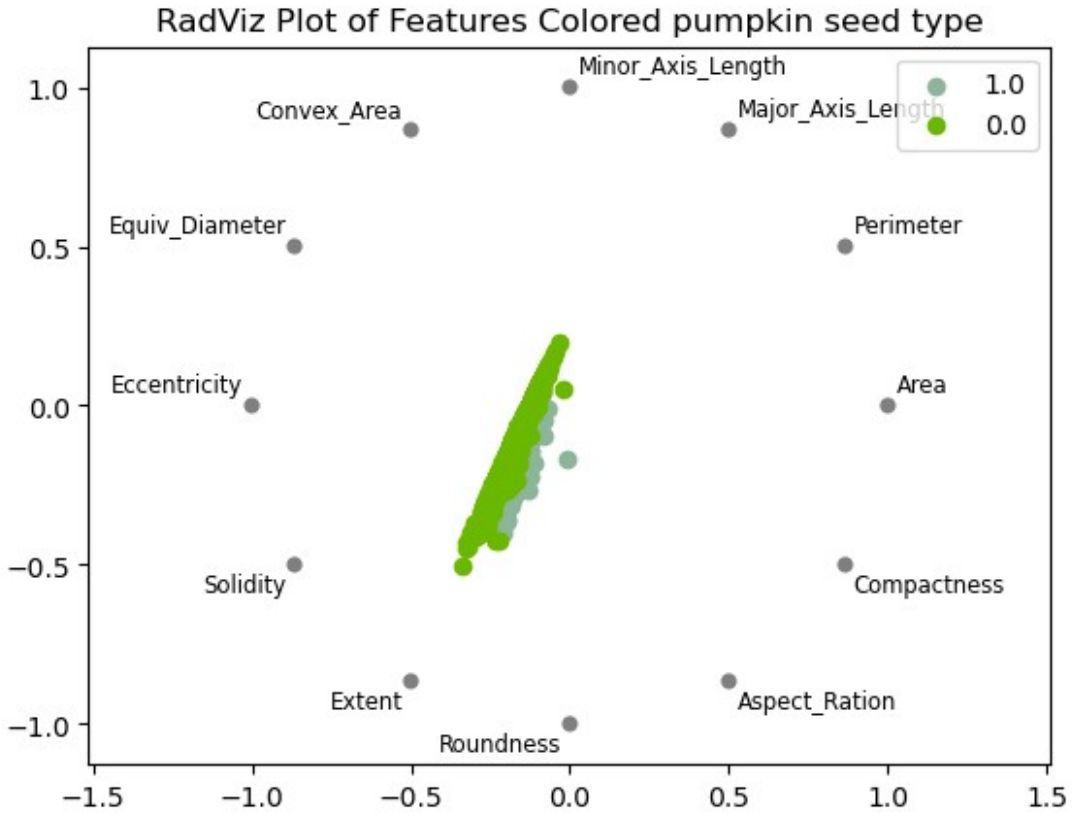
	Aspect_Ration	Compactness	Class
48	1.5416	0.7943	1.0
108	1.5150	0.8076	1.0
128	1.9556	0.7050	1.0
144	1.4760	0.8180	1.0
154	1.7952	0.7440	1.0

```
169          1.7649       0.7493    1.0
187          2.4683       0.6300    1.0
229          1.6412       0.7781    1.0
497          1.9164       0.7194    1.0
517          1.7396       0.7461    1.0
561          1.8243       0.7322    1.0
577          1.4266       0.8309    1.0
617          1.7170       0.7607    1.0
777          1.8482       0.7292    1.0
840          1.5978       0.7805    1.0
937          1.9038       0.7185    1.0
1019         1.7475       0.7537    1.0
1091         2.3907       0.6443    1.0
1198         1.6989       0.7653    1.0
1295         1.6156       0.7813    1.0
1376         2.2439       0.6639    0.0
1379         2.2367       0.6639    0.0
1470         2.2959       0.6539    0.0
1725         2.8645       0.5687    0.0
1807         2.2797       0.6591    0.0
2272         2.1124       0.6853    0.0
2280         2.3898       0.6443    0.0
2447         1.8371       0.7288    0.0
2480         2.7974       0.5871    0.0
Outliers for 'Compactness':
         Area  Perimeter  Major_Axis_Length  Minor_Axis_Length
Convex_Area  \
653   69132.0    961.531           342.3836           257.6274
69616.0
1194  66204.0   1006.743           320.8446           279.3001
67295.0

      Equiv_Diameter  Eccentricity  Solidity  Extent  Roundness  \
653         296.6843        0.6586    0.9930  0.7391     0.9396
1194        290.3335        0.4921    0.9838  0.6315     0.8208

      Aspect_Ration  Compactness  Class
653          1.3290       0.8665    1.0
1194         1.1487       0.9049    1.0
```

```
#Reference
[4]:https://pandas.pydata.org/pandas-docs/stable/user_guide/visualizat
ion.html
#Radviz plot to understand relationshi between features and class
plt.title('RadViz Plot of Features Colored pumpkin seed type')
radviz(df,'Class', cmap='bwr')
```

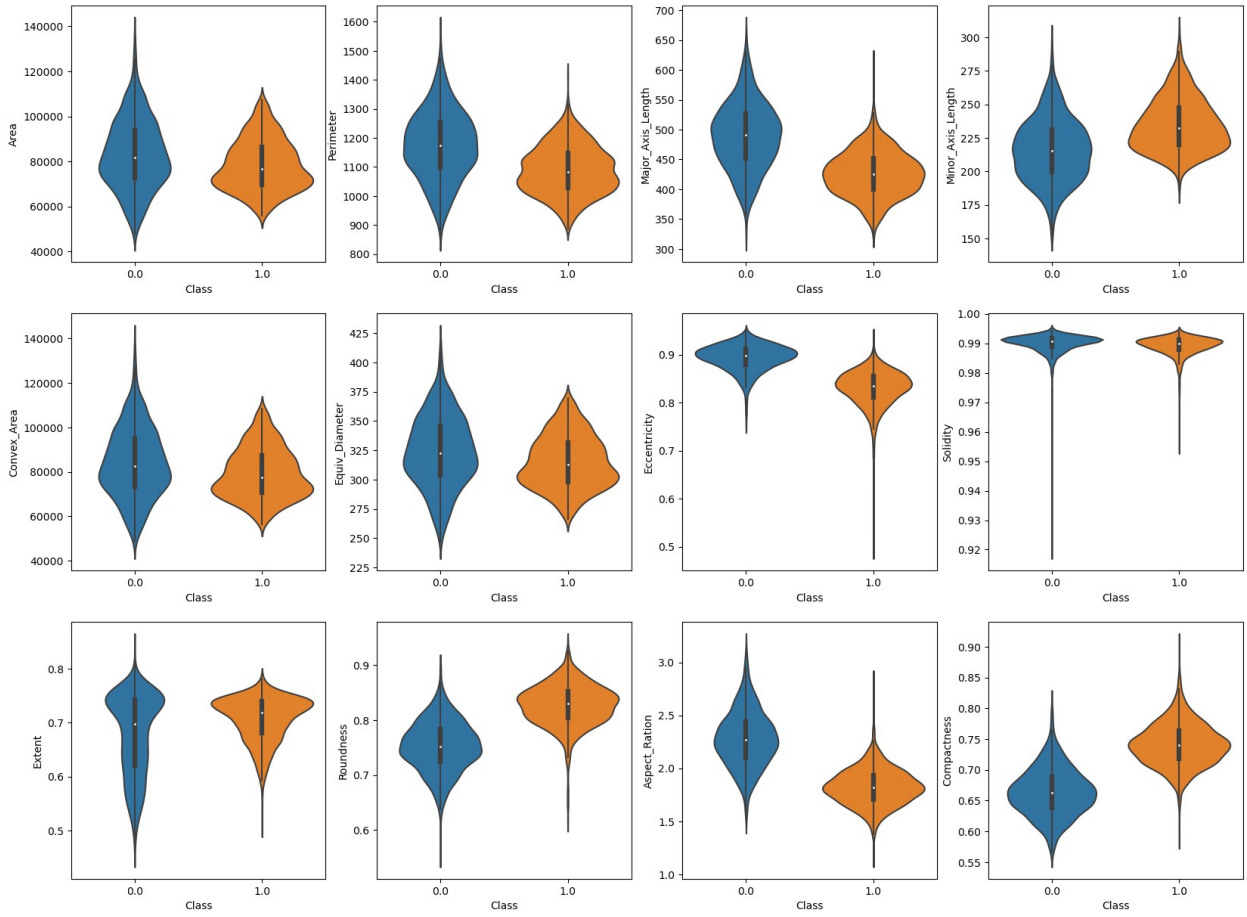
```
/Users/sahan/anaconda3/lib/python3.11/site-packages/pandas/plotting/
_matplotlib/misc.py:185: UserWarning: No data for colormapping
```

```
provided via 'c'. Parameters 'cmap' will be ignored
  ax.scatter(

<Axes: title={'center': 'RadViz Plot of Features Colored pumpkin seed
type'}>
```




```
#Reference [5] : https://www.geeksforgeeks.org/violinplot-using-
seaborn-in-python/
#In order to see the distribution of data better amongst the
differerent classes violin plots will be created
#Violin plots of class against each feature

#Subplots organised in 3x4
fig, axes = plt.subplots(3, 4, figsize=(20, 15))

# Violin plots for each feature
sns.violinplot(x='Class', y='Area', data=df, ax=axes[0, 0])
sns.violinplot(x='Class', y='Perimeter', data=df, ax=axes[0, 1])
sns.violinplot(x='Class', y='Major_Axis_Length', data=df, ax=axes[0,
2])
sns.violinplot(x='Class', y='Minor_Axis_Length', data=df, ax=axes[0,
3])
sns.violinplot(x='Class', y='Convex_Area', data=df, ax=axes[1, 0])
sns.violinplot(x='Class', y='Equiv_Diameter', data=df, ax=axes[1, 1])
```

```
sns.violinplot(x='Class', y='Eccentricity', data=df, ax=axes[1, 2])
sns.violinplot(x='Class', y='Solidity', data=df, ax=axes[1, 3])
sns.violinplot(x='Class', y='Extent', data=df, ax=axes[2, 0])
sns.violinplot(x='Class', y='Roundness', data=df, ax=axes[2, 1])
sns.violinplot(x='Class', y='Aspect_Ration', data=df, ax=axes[2, 2])
sns.violinplot(x='Class', y='Compactness', data=df, ax=axes[2, 3])
plt.show()
```



some variables were skewed, in order to prevent sknewness influencing the perfermance Box-Cox transformaton will be applied

```
#Reference [6]: https://www.geeksforgeeks.org/python-pandas-dataframe-skew/
#Observing skewness for each variable
data.skew()
```

```
Area                0.495999
Perimeter           0.414539
Major_Axis_Length   0.502980
```

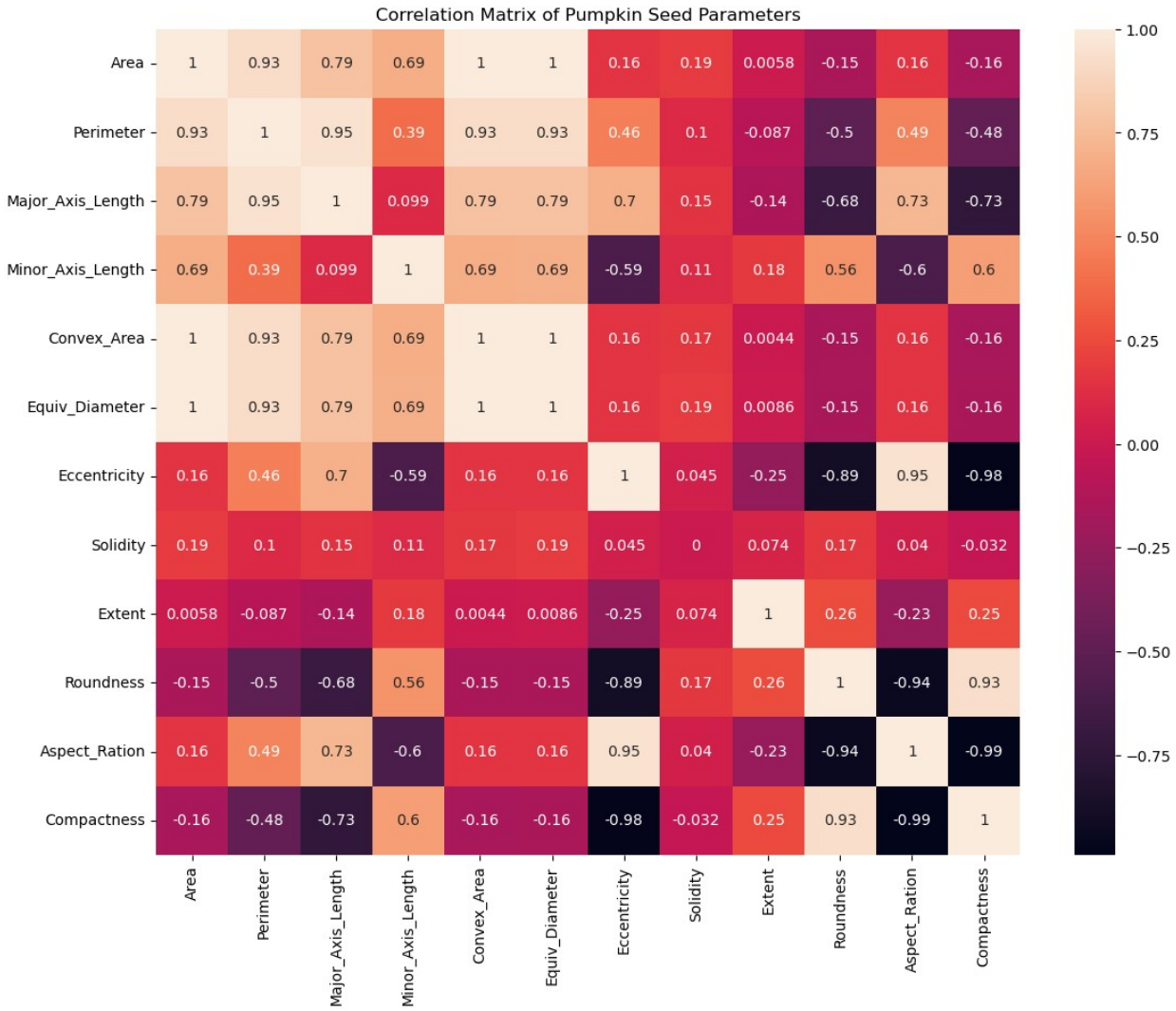
```
Minor_Axis_Length    0.104303
Convex_Area          0.494016
Equiv_Diameter       0.271868
Eccentricity        -0.748623
Solidity            -5.691009
Extent              -1.026568
Roundness           -0.372687
Aspect_Ration        0.548231
Compactness         -0.062377
Class               -0.080112
dtype: float64
```

It is clear to see that 'Solidity' and 'Extent' are very skewed, therfore box cox will only be applied to these variables only

```
#Applying Box-Cox using the box-cox function
# Columns with high skewness are solidity and extent
data['Solidity'], _ = boxcox(data['Solidity'] + 1)
# Adding 1 to handle zero and negative values
data['Extent'], _ = boxcox(data['Extent'] + 1)
#Displaying the skewness after box-cox transformation
data.skew()
```

```
Area                 0.495999
Perimeter            0.414539
Major_Axis_Length    0.502980
Minor_Axis_Length    0.104303
Convex_Area          0.494016
Equiv_Diameter       0.271868
Eccentricity        -0.748623
Solidity            -0.135784
Extent              -0.186925
Roundness           -0.372687
Aspect_Ration        0.548231
Compactness         -0.062377
Class               -0.080112
dtype: float64
```

```
#Correlation matrix to understand relatationship between variables
plt.figure(figsize=(13,10))
sns.heatmap(data[feature_columns].corr(), annot=True)
plt.title('Correlation Matrix of Pumpkin Seed Parameters')
plt.show()
```



Data Splitting

```
#Creating an X set Y set and separating the target variable ('class')
from the predictors
X = data.drop(columns=['Class'])
y = data['Class']

# Set random seed for reproducibility
random.seed(1)
np.random.seed(1)

# Splitting the data into training set, validation set and test set
#60% for training, 20% for validation and 20% for test
X_train, X_temp, y_train, y_temp = train_test_split(X, y,
test_size=0.4, stratify=y, random_state=10)
X_val, X_test, y_val, y_test = train_test_split(X_temp, y_temp,
test_size=0.5, stratify=y_temp, random_state=10)
```

```
#Size of all x sets
X_train.shape, X_test.shape, X_val.shape,
```

```
((1500, 12), (500, 12), (500, 12))
```

```
#Size of all Y sets
y_train.shape, y_test.shape, y_val.shape,
```

```
((1500,), (500,), (500,))
```

```
#Saving test sets to csv
X_test.to_csv('X_test.csv', index=False)

y_test.to_csv('y_test.csv', index=False, header=['Class'])

#Converting data into numpy arrays to ensure compatibility
X_train, y_train = np.array(X_train), np.array(y_train)
X_val, y_val = np.array(X_val), np.array(y_val)
X_test, y_test = np.array(X_test), np.array(y_test)

#Changing data into pytorch tensors ready for neural network training
X_train = torch.from_numpy(X_train).float()
y_train = torch.from_numpy(y_train).long()
X_val = torch.from_numpy(X_val).float()
y_val = torch.from_numpy(y_val).long()
X_test = torch.from_numpy(X_test).float()
y_test = torch.from_numpy(y_test).long()
```

Multilayer Perceptron (MLP)

```
#Reference [7] : Tutorial Lab (7)
#Reference [3]: https://github.com/vighnesh32/Neural-Computing-
Project/blob/main/project.ipynb

#Creating the base MLP architecture

# Defining the base Multi-Layer Perceptron (MLP) model architecture

class MLP(nn.Module):
    # Initialize the model
    def __init__(self, input_size=len(X.columns), hidden=50,
output_size=2, dropout=0.2, activation=F.relu):
        # Call the constructor of the parent class
        super(MLP, self).__init__()
        #dropout layer
        self.dropout = nn.Dropout(dropout)
```

```
        #Setting the activation function
        self.activation = activation
        #creating first Hidden layer
        self.fc1 = nn.Linear(input_size, hidden)
        #creating second Hidden layer
        self.fc2 = nn.Linear(hidden, hidden)
        #creating third Hidden Layer
        self.fc3 = nn.Linear(hidden, hidden)
        #Output layer
        self.output = nn.Linear(hidden, num_classes)

    #Forward Pass
    def forward(self, x):
        #Applying activation function to the first layer
        x = self.activation(self.fc1(x))
        #dropout for regularization
        x = self.dropout(x)
        #Activation function to the second layer
        x = self.activation(self.fc2(x))
        #dropout for regularization
        x = self.dropout(x)
        #Activation function to the third layer
        x = self.activation(self.fc3(x))
        #dropout for regularization
        x = self.dropout(x)
        #softmax activation
        x = F.softmax(self.output(x), dim=-1)

        return x

#The MLP model has three hidden layers (fc1, fc2, fc3) and each hidden
layer has 50 neurons as base
#Dropout is applied after each hidden layer for regularization and
dropout rate is 20%
#ReLU activation function is used
#Softmax activation function is used for the output layer to produce
the final output probabilities.

#Reference [8]: Tutorial lab (4)
#Reference [3]: https://github.com/vighnesh32/Neural-Computing-
Project/blob/main/project.ipynb

#Defining early stopback when it detects no further improvement
early_stopping_callback = EarlyStopping(monitor='valid_loss',
patience=40, lower_is_better=True)
#Definifn conditions for epoch scoring callback
epoch_scoring_callback = EpochScoring(scoring='accuracy',
lower_is_better=False)
#2 classes for classification
num_classes = 2
```

```python
# Neural network classifier
mlp_classifier = NeuralNetClassifier(
    MLP,
    # Max training epoch
    max_epochs=50,
    # Adam Optimiser
    optimizer=optim.Adam,
    # Learning rate set 0.01 as default
    optimizer__lr=0.01,
    # Weight decay for regularisation set to 1e-4
    optimizer__weight_decay=1e-4,
    # Cross entropy loss function
    criterion=nn.CrossEntropyLoss(),
    # Callbacks
    callbacks=[epoch_scoring_callback, early_stopping_callback]
)

# Fitting the model
mlp_classifier.fit(X_train, y_train)
```

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.4800	nan	0.4800	nan	0.1097
2	0.4800	nan	0.4800	nan	0.0201
3	0.4800	nan	0.4800	nan	0.0258
4	0.4800	nan	0.4800	nan	0.0289
5	0.4800	nan	0.4800	nan	0.0232
6	0.4800	nan	0.4800	nan	0.0221
7	0.4800	nan	0.4800	nan	0.0239
8	0.4800	nan	0.4800	nan	0.0287
9	0.4800	nan	0.4800	nan	0.0290
10	0.4800	nan	0.4800	nan	0.0283
11	0.4800	nan	0.4800	nan	0.0257
12	0.4800	nan	0.4800	nan	0.0195
13	0.4800	nan	0.4800	nan	0.0268
14	0.4800	nan	0.4800	nan	0.0240
15	0.4800	nan	0.4800	nan	0.0247
16	0.4800	nan	0.4800	nan	0.0195
17	0.4800	nan	0.4800	nan	0.0234
18	0.4800	nan	0.4800	nan	0.0231
19	0.4800	nan	0.4800	nan	0.0223
20	0.4800	nan	0.4800	nan	0.0239
21	0.4800	nan	0.4800	nan	0.0238
22	0.4800	nan	0.4800	nan	0.0275
23	0.4800	nan	0.4800	nan	0.0267
24	0.4800	nan	0.4800	nan	0.0227
25	0.4800	nan	0.4800	nan	0.0198

```
     26      0.4800            nan       0.4800            nan  0.0206
     27      0.4800            nan       0.4800            nan  0.0257
     28      0.4800            nan       0.4800            nan  0.0237
     29      0.4800            nan       0.4800            nan  0.0211
     30      0.4800            nan       0.4800            nan  0.0214
     31      0.4800            nan       0.4800            nan  0.0221
     32      0.4800            nan       0.4800            nan  0.0266
     33      0.4800            nan       0.4800            nan  0.0214
     34      0.4800            nan       0.4800            nan  0.0256
     35      0.4800            nan       0.4800            nan  0.0290
     36      0.4800            nan       0.4800            nan  0.0298
     37      0.4800            nan       0.4800            nan  0.0262
     38      0.4800            nan       0.4800            nan  0.0268
     39      0.4800            nan       0.4800            nan  0.0257
Stopping since valid_loss has not improved in the last 40 epochs.
```

```
<class 'skorch.classifier.NeuralNetClassifier'>[initialized](
  module_=MLP(
    (dropout): Dropout(p=0.2, inplace=False)
    (fc1): Linear(in_features=12, out_features=50, bias=True)
    (fc2): Linear(in_features=50, out_features=50, bias=True)
    (fc3): Linear(in_features=50, out_features=50, bias=True)
    (output): Linear(in_features=50, out_features=2, bias=True)
  ),
)
```

```
# Training accuracy of the base model
print("Training Accuracy")
accuracy_score(y_train, mlp_classifier.predict(X_train)) * 100
```

```
Training Accuracy
```

```
48.0
```

```
#validation set accuracy of the base model
print("Validation Accuracy")
accuracy_score(y_val, mlp_classifier.predict(X_val))*100
```

```
Validation Accuracy
```

```
48.0
```

Reconstructing the model but normalising the data. The accuracy was poor therefore we will retry with scaling.

```
#Reference [9]:
https://scikit-learn.org/stable/modules/generated/sklearn.preprocessin
g.MinMaxScaler.html
#Scaling
```

```
#Columns to scale
cols_to_scale = ['Area', 'Perimeter', 'Major_Axis_Length',
'Minor_Axis_Length',
                 'Convex_Area', 'Equiv_Diameter', 'Eccentricity',
'Solidity',
                 'Extent', 'Roundness', 'Aspect_Ration',
'Compactness',]


scaler = MinMaxScaler()
# Scaling the feature columns of the dataframe
data[cols_to_scale] = scaler.fit_transform(data[cols_to_scale])
```

Splitting the new scaled data

```
# Separating inputs and outputs for scaled data
X_scaled = data.drop('Class', axis=1)
y_scaled = data['Class']



# Splitting the data into training, validation, and test sets
X_train_scaled, X_temp_scaled, y_train_scaled, y_temp_scaled =
train_test_split(X_scaled, y_scaled, test_size=0.4, stratify=y_scaled,
random_state=10)
X_val_scaled, X_test_scaled, y_val_scaled, y_test_scaled =
train_test_split(X_temp_scaled, y_temp_scaled, test_size=0.5,
stratify=y_temp_scaled, random_state=10)

# Converting the scaled data to numpy arrays
X_train_scaled, y_train_scaled = np.array(X_train_scaled),
np.array(y_train_scaled)
X_val_scaled, y_val_scaled = np.array(X_val_scaled),
np.array(y_val_scaled)
X_test_scaled, y_test_scaled = np.array(X_test_scaled),
np.array(y_test_scaled)

#converitng scaled data to tensors from nupy arrays
X_train_scaled = torch.from_numpy(X_train_scaled).float()
y_train_scaled = torch.from_numpy(y_train_scaled).long()
X_val_scaled = torch.from_numpy(X_val_scaled).float()
y_val_scaled = torch.from_numpy(y_val_scaled).long()
X_test_scaled = torch.from_numpy(X_test_scaled).float()
y_test_scaled = torch.from_numpy(y_test_scaled).long()

# Reference [10] : https://blog.hubspot.com/website/python-
pickle#:~:text=To%20use%20pickle%20in%20Python,stored%20correctly
%20for%20later%20access.
# Saving the scaled test sets as a pickle file to load in the test
```

```
notebook

# X_test_scaled
with open('X_test_scaled.pkl', 'wb') as f:
    pickle.dump(X_test_scaled, f)

# y_test_scaled
with open('y_test_scaled.pkl', 'wb') as f:
    pickle.dump(y_test_scaled, f)

# Scaled MLP model
# Defining the scaled Multi-Layer Perceptron (MLP) model architecture
class MLP_scaled(nn.Module):
    def __init__(self, input_size=len(X_scaled.columns), hidden=50,
output_size=2, dropout=0.5, activation=F.relu):
        super(MLP_scaled, self).__init__()
        self.dropout = nn.Dropout(dropout)
        self.activation = activation
        self.fc1 = nn.Linear(input_size, hidden)
        self.fc2 = nn.Linear(hidden, hidden)
        self.fc3 = nn.Linear(hidden, hidden)
        self.output = nn.Linear(hidden, output_size)

    def forward(self, x):
        x = self.activation(self.fc1(x))
        x = self.dropout(x)
        x = self.activation(self.fc2(x))
        x = self.dropout(x)
        x = self.activation(self.fc3(x))
        x = self.dropout(x)
        x = F.softmax(self.output(x), dim=-1)
        return x

#Scaled data modelling
net_scaled = NeuralNetClassifier(
    MLP_scaled,
    max_epochs=100,
    optimizer=optim.Adam,
    optimizer__lr=0.01,
    optimizer__weight_decay=1e-4,
    criterion=nn.CrossEntropyLoss(),
    callbacks=[epoch_scoring_callback,early_stopping_callback])

net_scaled.fit(X_train_scaled, y_train_scaled)
```

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.8767	0.6751	0.8767	0.5846	0.0453
2	0.8667	0.5433	0.8667	0.4402	0.0379
3	0.8733	0.4651	0.8733	0.4343	0.0453

4	0.8700	0.4654	0.8700	0.4330	0.0501
5	0.8567	0.4631	0.8567	0.4518	0.0548
6	0.8533	0.4718	0.8533	0.4479	0.0409
7	0.8767	0.4685	0.8767	0.4310	0.0519
8	0.8733	0.4606	0.8733	0.4316	0.0606
9	0.8700	0.4646	0.8700	0.4368	0.0447
10	0.8800	0.4479	0.8800	0.4308	0.0700
11	0.8733	0.4543	0.8733	0.4320	0.0445
12	0.8733	0.4455	0.8733	0.4340	0.0408
13	0.8767	0.4506	0.8767	0.4321	0.0344
14	0.8700	0.4511	0.8700	0.4342	0.0322
15	0.8733	0.4470	0.8733	0.4333	0.0315
16	0.8733	0.4473	0.8733	0.4340	0.0316
17	0.8733	0.4414	0.8733	0.4349	0.0322
18	0.8733	0.4463	0.8733	0.4321	0.0333
19	0.8733	0.4551	0.8733	0.4347	0.0329
20	0.8700	0.4516	0.8700	0.4325	0.0306
21	0.8800	0.4511	0.8800	0.4309	0.0351
22	0.8500	0.4477	0.8500	0.4543	0.0329
23	0.8567	0.4633	0.8567	0.4518	0.0340
24	0.8800	0.4496	0.8800	0.4372	0.0391
25	0.8800	0.4483	0.8800	0.4318	0.0384
26	0.8800	0.4480	0.8800	0.4345	0.0354
27	0.8667	0.4494	0.8667	0.4369	0.0302
28	0.8733	0.4523	0.8733	0.4314	0.0390
29	0.8800	0.4438	0.8800	0.4283	0.0353
30	0.8733	0.4319	0.8733	0.4337	0.0343
31	0.8800	0.4385	0.8800	0.4302	0.0348
32	0.8800	0.4333	0.8800	0.4315	0.0333
33	0.8733	0.4496	0.8733	0.4308	0.0304
34	0.8800	0.4482	0.8800	0.4304	0.0333
35	0.8767	0.4320	0.8767	0.4370	0.0363
36	0.8767	0.4520	0.8767	0.4327	0.0361
37	0.8733	0.4497	0.8733	0.4334	0.0361
38	0.8533	0.4569	0.8533	0.4531	0.0345
39	0.8800	0.4476	0.8800	0.4312	0.0345
40	0.8767	0.4453	0.8767	0.4339	0.0325
41	0.8767	0.4460	0.8767	0.4311	0.0325
42	0.8700	0.4404	0.8700	0.4363	0.0305
43	0.8833	0.4391	0.8833	0.4312	0.0331
44	0.8767	0.4420	0.8767	0.4344	0.0383
45	0.8767	0.4409	0.8767	0.4303	0.0364
46	0.8733	0.4439	0.8733	0.4337	0.0370
47	0.8733	0.4483	0.8733	0.4319	0.0378
48	0.8733	0.4464	0.8733	0.4332	0.0374
49	0.8767	0.4390	0.8767	0.4342	0.0367
50	0.8767	0.4447	0.8767	0.4344	0.0349
51	0.8633	0.4416	0.8633	0.4427	0.0319
52	0.8733	0.4391	0.8733	0.4362	0.0310

```
     53      0.8767        0.4373       0.8767        0.4314  0.0325
     54      0.8800        0.4412       0.8800        0.4379  0.0315
     55      0.8233        0.4575       0.8233        0.4807  0.0315
     56      0.8733        0.4564       0.8733        0.4320  0.0310
     57      0.8733        0.4416       0.8733        0.4317  0.0307
     58      0.8800        0.4293       0.8800        0.4320  0.0330
     59      0.8733        0.4387       0.8733        0.4321  0.0315
     60      0.8767        0.4374       0.8767        0.4306  0.0313
     61      0.8733        0.4413       0.8733        0.4343  0.0327
     62      0.8700        0.4401       0.8700        0.4365  0.0317
     63      0.8800        0.4352       0.8800        0.4334  0.0322
     64      0.8800        0.4362       0.8800        0.4322  0.0304
     65      0.8733        0.4413       0.8733        0.4350  0.0319
     66      0.8767        0.4313       0.8767        0.4328  0.0314
     67      0.8800        0.4348       0.8800        0.4316  0.0329
     68      0.8733        0.4306       0.8733        0.4340  0.0357
Stopping since valid_loss has not improved in the last 40 epochs.
```

```
<class 'skorch.classifier.NeuralNetClassifier'>[initialized](
  module_=MLP_scaled(
    (dropout): Dropout(p=0.5, inplace=False)
    (fc1): Linear(in_features=12, out_features=50, bias=True)
    (fc2): Linear(in_features=50, out_features=50, bias=True)
    (fc3): Linear(in_features=50, out_features=50, bias=True)
    (output): Linear(in_features=50, out_features=2, bias=True)
  ),
)
```

```
# Training set accuracy of the scaled model
print("Training Accuracy")
accuracy_score(y_train_scaled, net_scaled.predict(X_train_scaled))*100
```

```
Training Accuracy
```

```
88.0
```

```
# Validation set accuracy of the scaled model
print("Validation Accuracy")
accuracy_score(y_val_scaled, net_scaled.predict(X_val_scaled))*100
```

```
Validation Accuracy
```

```
90.0
```

The accuracy of the model has improved significantly after normalisation scaling of the data. Next gridsearch will be carried out with the scaled data to find the optimal hyperparameters for MLP to ensure the highest accuracy

```
# Reference [11] : https://pieriantraining.com/gridsearchcv-with-scikit-learn-and-python/

# Starting timer
start_time = time.time()

# Gridsearch criteria
mlp_grid = {
    'lr': [0.0001, 0.01, 0.1],
    'max_epochs': [10, 50, 100],
    'optimizer':[optim.Adam, optim.SGD],
    'batch_size':[32,64]
}

mlp_scaled = NeuralNetClassifier(
    MLP_scaled,
    max_epochs=10,
    optimizer=optim.Adam,
    criterion=nn.CrossEntropyLoss(),
    optimizer__lr=0.01,
    optimizer__weight_decay=1e-4,
    callbacks=[epoch_scoring_callback, early_stopping_callback]
)

# GridSearchCV object
grid_search_scaled = GridSearchCV(mlp_scaled, mlp_grid, scoring='accuracy', cv=2, verbose=1)

# Fitting grid search for scaled data
grid_search_scaled.fit(X_train_scaled, y_train_scaled)

# Stop timer
end_time = time.time()

# Output best parameters
print("Best parameters found for scaled data: ", grid_search_scaled.best_params_)

# Time taken
print(f"Time taken for grid search: {end_time - start_time:.2f} seconds")
```

```
Fitting 2 folds for each of 36 candidates, totalling 72 fits
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.7800        0.6811       0.7800        0.6075  0.0428
      2      0.8267        0.5505       0.8267        0.4869  0.0406
      3      0.7533        0.5213       0.7533        0.5584  0.0437
      4      0.8067        0.4948       0.8067        0.4934  0.0434
      5      0.8067        0.4741       0.8067        0.4885  0.0441
```

```
      6      0.8200        0.4737       0.8200        0.4874  0.0430
      7      0.8133        0.4610       0.8133        0.4854  0.0471
      8      0.8267        0.4757       0.8267        0.4757  0.0448
      9      0.8267        0.4806       0.8267        0.4841  0.0422
     10      0.8133        0.4491       0.8133        0.4858  0.0414
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.7000        0.6880       0.7000        0.6397  0.0388
      2      0.9000        0.5685       0.9000        0.4146  0.0423
      3      0.9067        0.5129       0.9067        0.4104  0.0422
      4      0.9067        0.4736       0.9067        0.4051  0.0410
      5      0.9000        0.4665       0.9000        0.4065  0.0380
      6      0.8867        0.4500       0.8867        0.4213  0.0369
      7      0.8600        0.4588       0.8600        0.4501  0.0376
      8      0.9000        0.4620       0.9000        0.4038  0.0349
      9      0.8800        0.4562       0.8800        0.4328  0.0335
     10      0.8867        0.4638       0.8867        0.4211  0.0381
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.4800        0.6946       0.4800        0.6934  0.0276
      2      0.4800        0.6953       0.4800        0.6931  0.0284
      3      0.4800        0.6951       0.4800        0.6928  0.0254
      4      0.4800        0.6951       0.4800        0.6926  0.0286
      5      0.4800        0.6938       0.4800        0.6924  0.0304
      6      0.4800        0.6931       0.4800        0.6921  0.0294
      7      0.4800        0.6950       0.4800        0.6920  0.0298
      8      0.4800        0.6936       0.4800        0.6918  0.0298
      9      0.4800        0.6933       0.4800        0.6916  0.0313
     10      0.4800        0.6917       0.4800        0.6914  0.0298
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5200        0.6915       0.5200        0.6899  0.0284
      2      0.5200        0.6923       0.5200        0.6898  0.0302
      3      0.5200        0.6913       0.5200        0.6897  0.0267
      4      0.5200        0.6922       0.5200        0.6896  0.0294
      5      0.5200        0.6900       0.5200        0.6894  0.0281
      6      0.5200        0.6897       0.5200        0.6893  0.0295
      7      0.5200        0.6914       0.5200        0.6892  0.0293
      8      0.5200        0.6918       0.5200        0.6891  0.0284
      9      0.5200        0.6897       0.5200        0.6890  0.0307
     10      0.5200        0.6874       0.5200        0.6888  0.0307
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.8267        0.6687       0.8267        0.5440  0.0375
      2      0.8333        0.5218       0.8333        0.4749  0.0355
      3      0.8133        0.4861       0.8133        0.4841  0.0358
      4      0.8267        0.4885       0.8267        0.4802  0.0397
      5      0.8333        0.4828       0.8333        0.4783  0.0387
      6      0.8067        0.4768       0.8067        0.4835  0.0370
```

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
7	0.8067	0.4835	0.8067	0.4842	0.0358
8	0.8067	0.4785	0.8067	0.4801	0.0356
9	0.8200	0.4683	0.8200	0.4819	0.0377
10	0.8267	0.4623	0.8267	0.4806	0.0353
11	0.8200	0.4551	0.8200	0.4839	0.0377
12	0.7867	0.5176	0.7867	0.5178	0.0353
13	0.8267	0.4747	0.8267	0.4854	0.0371
14	0.8333	0.4604	0.8333	0.4795	0.0357
15	0.8267	0.4722	0.8267	0.4774	0.0386
16	0.8133	0.4756	0.8133	0.4958	0.0347
17	0.8200	0.4651	0.8200	0.4723	0.0366
18	0.8333	0.4494	0.8333	0.4799	0.0346
19	0.8333	0.4634	0.8333	0.4759	0.0366
20	0.8000	0.4633	0.8000	0.5036	0.0344
21	0.8267	0.4755	0.8267	0.4870	0.0369
22	0.8267	0.4756	0.8267	0.4866	0.0349
23	0.8200	0.4619	0.8200	0.4828	0.0334
24	0.8333	0.4636	0.8333	0.4742	0.0320
25	0.8133	0.4776	0.8133	0.5020	0.0342
26	0.8333	0.4579	0.8333	0.4844	0.0346
27	0.8200	0.4691	0.8200	0.4857	0.0353
28	0.8200	0.4615	0.8200	0.4836	0.0350
29	0.8267	0.4509	0.8267	0.4856	0.0333
30	0.8333	0.4626	0.8333	0.4782	0.0342
31	0.8200	0.4470	0.8200	0.4896	0.0335
32	0.8067	0.4536	0.8067	0.4814	0.0343
33	0.8267	0.4579	0.8267	0.4844	0.0349
34	0.8333	0.4614	0.8333	0.4778	0.0317
35	0.8267	0.4424	0.8267	0.4865	0.0357
36	0.8200	0.4670	0.8200	0.4911	0.0351
37	0.8133	0.4715	0.8133	0.4912	0.0353
38	0.8267	0.4476	0.8267	0.4851	0.0345
39	0.8000	0.4593	0.8000	0.5043	0.0347
40	0.8133	0.4918	0.8133	0.4972	0.0346
41	0.8067	0.4819	0.8067	0.4949	0.0346
42	0.8467	0.4669	0.8467	0.4762	0.0330
43	0.8200	0.4693	0.8200	0.4863	0.0333
44	0.8267	0.4777	0.8267	0.4838	0.0341
45	0.8267	0.4515	0.8267	0.4845	0.0334
46	0.8133	0.4669	0.8133	0.5001	0.0332
47	0.8267	0.4621	0.8267	0.4807	0.0343
48	0.8200	0.4522	0.8200	0.4832	0.0339
49	0.8333	0.4567	0.8333	0.4801	0.0320
50	0.8333	0.4497	0.8333	0.4797	0.0342
epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.8733	0.6887	0.8733	0.6317	0.0326
2	0.8933	0.5832	0.8933	0.4225	0.0337
3	0.9067	0.4972	0.9067	0.4033	0.0333

```
      4      0.9133        0.4525       0.9133        0.4030  0.0349
      5      0.9133        0.4706       0.9133        0.4011  0.0338
      6      0.9000        0.4575       0.9000        0.4091  0.0341
      7      0.8733        0.4423       0.8733        0.4259  0.0350
      8      0.9067        0.4663       0.9067        0.4066  0.0328
      9      0.9067        0.4857       0.9067        0.4049  0.0321
     10      0.9067        0.4675       0.9067        0.4045  0.0326
     11      0.9067        0.4425       0.9067        0.4020  0.0333
     12      0.8600        0.4661       0.8600        0.4520  0.0326
     13      0.9000        0.4707       0.9000        0.4125  0.0343
     14      0.8733        0.4715       0.8733        0.4361  0.0339
     15      0.9133        0.4568       0.9133        0.3979  0.0346
     16      0.8200        0.4754       0.8200        0.4937  0.0327
     17      0.8667        0.4775       0.8667        0.4401  0.0326
     18      0.8667        0.4902       0.8667        0.4495  0.0318
     19      0.8800        0.4703       0.8800        0.4316  0.0338
     20      0.8933        0.5090       0.8933        0.4185  0.0343
     21      0.8800        0.4772       0.8800        0.4344  0.0346
     22      0.9000        0.4563       0.9000        0.4068  0.0296
     23      0.9133        0.4487       0.9133        0.3989  0.0334
     24      0.8800        0.4400       0.8800        0.4305  0.0309
     25      0.8867        0.4661       0.8867        0.4245  0.0319
     26      0.9067        0.4595       0.9067        0.4039  0.0334
     27      0.9067        0.4511       0.9067        0.4054  0.0339
     28      0.9067        0.4474       0.9067        0.4076  0.0338
     29      0.9067        0.4520       0.9067        0.4053  0.0301
     30      0.9000        0.4658       0.9000        0.4045  0.0333
     31      0.8933        0.4536       0.8933        0.4176  0.0339
     32      0.9067        0.4737       0.9067        0.4056  0.0338
     33      0.8800        0.4544       0.8800        0.4303  0.0320
     34      0.9067        0.4492       0.9067        0.4074  0.0329
     35      0.8733        0.4382       0.8733        0.4337  0.0339
     36      0.9000        0.4389       0.9000        0.4071  0.0339
     37      0.9000        0.4327       0.9000        0.4056  0.0315
     38      0.8800        0.4457       0.8800        0.4256  0.0301
     39      0.9000        0.4334       0.9000        0.4100  0.0328
     40      0.8933        0.4358       0.8933        0.4169  0.0356
     41      0.8933        0.4420       0.8933        0.4084  0.0325
     42      0.8800        0.4488       0.8800        0.4313  0.0340
     43      0.8800        0.4332       0.8800        0.4327  0.0327
     44      0.8933        0.4414       0.8933        0.4167  0.0327
     45      0.9000        0.4321       0.9000        0.4075  0.0327
     46      0.9000        0.4355       0.9000        0.4116  0.0337
     47      0.9133        0.4449       0.9133        0.4011  0.0334
     48      0.8667        0.4625       0.8667        0.4442  0.0358
     49      0.9067        0.4355       0.9067        0.4033  0.0332
     50      0.9067        0.4400       0.9067        0.4031  0.0343
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.3067        0.6949       0.3067        0.6937  0.0269
```

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
2	0.3867	0.6926	0.3867	0.6937	0.0239
3	0.5133	0.6945	0.5133	0.6936	0.0250
4	0.5200	0.6929	0.5200	0.6935	0.0270
5	0.5200	0.6937	0.5200	0.6935	0.0245
6	0.5200	0.6932	0.5200	0.6934	0.0259
7	0.5200	0.6913	0.5200	0.6933	0.0274
8	0.5200	0.6944	0.5200	0.6933	0.0265
9	0.5200	0.6933	0.5200	0.6932	0.0262
10	0.5200	0.6924	0.5200	0.6932	0.0249
11	0.5200	0.6944	0.5200	0.6931	0.0258
12	0.5200	0.6914	0.5200	0.6931	0.0273
13	0.5200	0.6935	0.5200	0.6931	0.0264
14	0.5200	0.6921	0.5200	0.6930	0.0264
15	0.5200	0.6928	0.5200	0.6930	0.0271
16	0.5200	0.6932	0.5200	0.6929	0.0291
17	0.5200	0.6925	0.5200	0.6929	0.0282
18	0.5200	0.6909	0.5200	0.6929	0.0277
19	0.5200	0.6902	0.5200	0.6928	0.0257
20	0.5200	0.6926	0.5200	0.6928	0.0252
21	0.5200	0.6922	0.5200	0.6928	0.0296
22	0.5200	0.6919	0.5200	0.6927	0.0315
23	0.5200	0.6910	0.5200	0.6927	0.0316
24	0.5200	0.6924	0.5200	0.6927	0.0303
25	0.5200	0.6934	0.5200	0.6926	0.0277
26	0.5200	0.6909	0.5200	0.6926	0.0286
27	0.5200	0.6933	0.5200	0.6926	0.0292
28	0.5200	0.6939	0.5200	0.6925	0.0269
29	0.5200	0.6922	0.5200	0.6925	0.0267
30	0.5200	0.6927	0.5200	0.6925	0.0262
31	0.5200	0.6918	0.5200	0.6924	0.0288
32	0.5200	0.6919	0.5200	0.6924	0.0290
33	0.5200	0.6931	0.5200	0.6924	0.0274
34	0.5200	0.6930	0.5200	0.6923	0.0284
35	0.5200	0.6932	0.5200	0.6923	0.0282
36	0.5200	0.6915	0.5200	0.6923	0.0282
37	0.5200	0.6915	0.5200	0.6922	0.0294
38	0.5200	0.6915	0.5200	0.6922	0.0252
39	0.5200	0.6922	0.5200	0.6922	0.0283
40	0.5200	0.6923	0.5200	0.6921	0.0276
41	0.5200	0.6923	0.5200	0.6921	0.0287
42	0.5200	0.6923	0.5200	0.6921	0.0290
43	0.5200	0.6936	0.5200	0.6920	0.0265
44	0.5200	0.6934	0.5200	0.6920	0.0264
45	0.5200	0.6915	0.5200	0.6920	0.0261
46	0.5200	0.6925	0.5200	0.6920	0.0288
47	0.5200	0.6924	0.5200	0.6919	0.0288
48	0.5200	0.6925	0.5200	0.6919	0.0269
49	0.5200	0.6915	0.5200	0.6919	0.0280
50	0.5200	0.6909	0.5200	0.6918	0.0293

1	0.4800	0.6981	0.4800	0.6968	0.0275
2	0.4800	0.6960	0.4800	0.6965	0.0262
3	0.4800	0.6971	0.4800	0.6963	0.0274
4	0.4800	0.6971	0.4800	0.6960	0.0300
5	0.4800	0.6963	0.4800	0.6958	0.0274
6	0.4800	0.6959	0.4800	0.6956	0.0291
7	0.4800	0.6976	0.4800	0.6954	0.0291
8	0.4800	0.6961	0.4800	0.6952	0.0274
9	0.4800	0.6953	0.4800	0.6951	0.0303
10	0.4800	0.6953	0.4800	0.6949	0.0310
11	0.4800	0.6963	0.4800	0.6947	0.0263
12	0.4800	0.6947	0.4800	0.6946	0.0273
13	0.4800	0.6952	0.4800	0.6944	0.0292
14	0.4800	0.6961	0.4800	0.6943	0.0260
15	0.4800	0.6949	0.4800	0.6942	0.0274
16	0.4800	0.6923	0.4800	0.6941	0.0275
17	0.4800	0.6946	0.4800	0.6940	0.0261
18	0.4800	0.6939	0.4800	0.6938	0.0269
19	0.4800	0.6949	0.4800	0.6937	0.0269
20	0.4800	0.6919	0.4800	0.6936	0.0271
21	0.4867	0.6958	0.4867	0.6935	0.0269
22	0.4867	0.6940	0.4867	0.6934	0.0257
23	0.4933	0.6927	0.4933	0.6933	0.0264
24	0.5067	0.6937	0.5067	0.6932	0.0283
25	0.4800	0.6911	0.4800	0.6931	0.0270
26	0.5067	0.6941	0.5067	0.6930	0.0264
27	0.5200	0.6935	0.5200	0.6929	0.0276
28	0.5267	0.6925	0.5267	0.6929	0.0280
29	0.5200	0.6919	0.5200	0.6928	0.0282
30	0.5200	0.6921	0.5200	0.6927	0.0284
31	0.5200	0.6934	0.5200	0.6926	0.0294
32	0.5200	0.6913	0.5200	0.6925	0.0242
33	0.5200	0.6933	0.5200	0.6925	0.0253
34	0.5200	0.6937	0.5200	0.6924	0.0293
35	0.5200	0.6940	0.5200	0.6923	0.0277
36	0.5200	0.6930	0.5200	0.6923	0.0262
37	0.5200	0.6923	0.5200	0.6922	0.0288
38	0.5200	0.6915	0.5200	0.6921	0.0257
39	0.5200	0.6924	0.5200	0.6920	0.0279
40	0.5200	0.6924	0.5200	0.6920	0.0276
41	0.5200	0.6922	0.5200	0.6919	0.0261
42	0.5200	0.6919	0.5200	0.6919	0.0268
43	0.5200	0.6907	0.5200	0.6918	0.0276
44	0.5200	0.6922	0.5200	0.6917	0.0274
45	0.5200	0.6918	0.5200	0.6917	0.0286
46	0.5200	0.6911	0.5200	0.6916	0.0281
47	0.5200	0.6901	0.5200	0.6915	0.0269
48	0.5200	0.6941	0.5200	0.6915	0.0288

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.7800	0.6895	0.7800	0.6349	0.0351
2	0.8267	0.5619	0.8267	0.4768	0.0347
3	0.8133	0.4936	0.8133	0.4877	0.0337
4	0.8133	0.4848	0.8133	0.4890	0.0350
5	0.8267	0.4734	0.8267	0.4763	0.0356
6	0.8400	0.4731	0.8400	0.4696	0.0331
7	0.7867	0.4694	0.7867	0.5051	0.0345
8	0.8133	0.5023	0.8133	0.4934	0.0334
9	0.8133	0.4634	0.8133	0.4900	0.0341
10	0.8200	0.4472	0.8200	0.4932	0.0299
11	0.8200	0.4925	0.8200	0.4795	0.0325
12	0.8333	0.4480	0.8333	0.4773	0.0336
13	0.8400	0.4638	0.8400	0.4665	0.0335
14	0.8133	0.4574	0.8133	0.4922	0.0351
15	0.8333	0.4715	0.8333	0.4741	0.0340
16	0.8333	0.4589	0.8333	0.4762	0.0323
17	0.8200	0.4562	0.8200	0.4777	0.0338
18	0.8133	0.4493	0.8133	0.4791	0.0334
19	0.8467	0.4429	0.8467	0.4682	0.0335
20	0.8400	0.4408	0.8400	0.4772	0.0348
21	0.8200	0.4609	0.8200	0.4854	0.0332
22	0.8400	0.4786	0.8400	0.4687	0.0337
23	0.8333	0.4675	0.8333	0.4792	0.0329
24	0.8400	0.4510	0.8400	0.4730	0.0326
25	0.8333	0.4720	0.8333	0.4719	0.0349
26	0.8533	0.4591	0.8533	0.4625	0.0342
27	0.8533	0.4434	0.8533	0.4655	0.0334
28	0.8267	0.4514	0.8267	0.4788	0.0345
29	0.8133	0.4463	0.8133	0.4926	0.0334
30	0.8400	0.4583	0.8400	0.4767	0.0346
31	0.8267	0.4779	0.8267	0.4813	0.0335
32	0.8400	0.4756	0.8400	0.4781	0.0333
33	0.8400	0.4503	0.8400	0.4716	0.0340
34	0.8333	0.4556	0.8333	0.4771	0.0340
35	0.8267	0.4609	0.8267	0.4797	0.0361
36	0.8133	0.4598	0.8133	0.4992	0.0358
37	0.8400	0.4568	0.8400	0.4711	0.0370
38	0.8400	0.4409	0.8400	0.4733	0.0340
39	0.8333	0.4487	0.8333	0.4766	0.0361
40	0.8267	0.4589	0.8267	0.4724	0.0341
41	0.8133	0.4572	0.8133	0.4825	0.0326
42	0.8333	0.4619	0.8333	0.4723	0.0351
43	0.8333	0.4427	0.8333	0.4665	0.0347
44	0.8333	0.4444	0.8333	0.4648	0.0340
45	0.8333	0.4393	0.8333	0.4793	0.0348
49	0.5200	0.6912	0.5200	0.6914	0.0264
50	0.5200	0.6921	0.5200	0.6913	0.0271

46	0.8333	0.4449	0.8333	0.4694	0.0342
47	0.8333	0.4450	0.8333	0.4756	0.0340
48	0.8400	0.4577	0.8400	0.4754	0.0340
49	0.8400	0.4447	0.8400	0.4687	0.0344
50	0.8267	0.4557	0.8267	0.4756	0.0352
51	0.8400	0.4563	0.8400	0.4683	0.0318
52	0.8400	0.4561	0.8400	0.4615	0.0343
53	0.8267	0.4591	0.8267	0.4886	0.0340
54	0.8467	0.4456	0.8467	0.4640	0.0350
55	0.8133	0.4458	0.8133	0.4903	0.0351
56	0.8267	0.4769	0.8267	0.4756	0.0355
57	0.8400	0.4415	0.8400	0.4687	0.0345
58	0.8467	0.4435	0.8467	0.4651	0.0335
59	0.8400	0.4437	0.8400	0.4704	0.0342
60	0.8467	0.4478	0.8467	0.4581	0.0353
61	0.8467	0.4504	0.8467	0.4708	0.0348
62	0.8400	0.4390	0.8400	0.4592	0.0338
63	0.8333	0.4319	0.8333	0.4732	0.0342
64	0.8467	0.4408	0.8467	0.4662	0.0341
65	0.8400	0.4387	0.8400	0.4672	0.0350
66	0.8467	0.4440	0.8467	0.4684	0.0351
67	0.8467	0.4359	0.8467	0.4585	0.0350
68	0.8467	0.4489	0.8467	0.4605	0.0322
69	0.8333	0.4368	0.8333	0.4749	0.0352
70	0.8400	0.4420	0.8400	0.4748	0.0353
71	0.8533	0.4432	0.8533	0.4610	0.0346
72	0.8400	0.4324	0.8400	0.4699	0.0346
73	0.8533	0.4516	0.8533	0.4543	0.0332
74	0.8467	0.4412	0.8467	0.4658	0.0315
75	0.8467	0.4575	0.8467	0.4623	0.0330
76	0.8467	0.4486	0.8467	0.4615	0.0332
77	0.8333	0.4537	0.8333	0.4735	0.0339
78	0.8600	0.4539	0.8600	0.4553	0.0328
79	0.8533	0.4556	0.8533	0.4532	0.0320
80	0.8533	0.4470	0.8533	0.4535	0.0318
81	0.8400	0.4438	0.8400	0.4648	0.0333
82	0.8533	0.4445	0.8533	0.4482	0.0352
83	0.8600	0.4381	0.8600	0.4517	0.0337
84	0.8533	0.4389	0.8533	0.4469	0.0358
85	0.8533	0.4496	0.8533	0.4486	0.0343
86	0.8600	0.4369	0.8600	0.4433	0.0333
87	0.8467	0.4349	0.8467	0.4598	0.0339
88	0.8600	0.4311	0.8600	0.4462	0.0350
89	0.8733	0.4421	0.8733	0.4431	0.0345
90	0.8533	0.4387	0.8533	0.4585	0.0298
91	0.8467	0.4409	0.8467	0.4577	0.0344
92	0.8533	0.4521	0.8533	0.4452	0.0369
93	0.8467	0.4426	0.8467	0.4666	0.0350
94	0.8667	0.4413	0.8667	0.4415	0.0359

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
95	0.8600	0.4425	0.8600	0.4564	0.0361
96	0.8400	0.4409	0.8400	0.4720	0.0356
97	0.8533	0.4637	0.8533	0.4540	0.0375
98	0.8400	0.4553	0.8400	0.4707	0.0396
99	0.8533	0.4420	0.8533	0.4557	0.0438
100	0.8600	0.4575	0.8600	0.4553	0.0422
epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.8867	0.6819	0.8867	0.5782	0.0367
2	0.8600	0.5353	0.8600	0.4478	0.0418
3	0.8800	0.4944	0.8800	0.4249	0.0356
4	0.9133	0.4512	0.9133	0.4037	0.0399
5	0.8667	0.4578	0.8667	0.4377	0.0397
6	0.8667	0.4796	0.8667	0.4385	0.0375
7	0.8733	0.4645	0.8733	0.4367	0.0367
8	0.8800	0.4476	0.8800	0.4225	0.0436
9	0.8867	0.4669	0.8867	0.4258	0.0431
10	0.8933	0.4547	0.8933	0.4155	0.0408
11	0.9133	0.4550	0.9133	0.3996	0.0431
12	0.8800	0.4603	0.8800	0.4325	0.0397
13	0.8933	0.4755	0.8933	0.4132	0.0394
14	0.8867	0.4727	0.8867	0.4230	0.0389
15	0.8933	0.4579	0.8933	0.4158	0.0402
16	0.8867	0.4420	0.8867	0.4093	0.0382
17	0.8933	0.4474	0.8933	0.4201	0.0376
18	0.8867	0.4330	0.8867	0.4121	0.0381
19	0.8867	0.4587	0.8867	0.4256	0.0381
20	0.8600	0.4549	0.8600	0.4489	0.0412
21	0.9000	0.4740	0.9000	0.4078	0.0390
22	0.8933	0.4624	0.8933	0.4204	0.0380
23	0.9200	0.4377	0.9200	0.3999	0.0354
24	0.8933	0.4523	0.8933	0.4091	0.0340
25	0.8867	0.4392	0.8867	0.4146	0.0360
26	0.8867	0.4463	0.8867	0.4221	0.0346
27	0.9133	0.4444	0.9133	0.4016	0.0315
28	0.8600	0.4491	0.8600	0.4492	0.0345
29	0.8867	0.4624	0.8867	0.4210	0.0356
30	0.9067	0.4424	0.9067	0.4075	0.0365
31	0.9133	0.4288	0.9133	0.3997	0.0345
32	0.8867	0.4568	0.8867	0.4234	0.0351
33	0.8933	0.4374	0.8933	0.4127	0.0350
34	0.8933	0.4363	0.8933	0.4209	0.0345
35	0.9067	0.4430	0.9067	0.4032	0.0314
36	0.9000	0.4566	0.9000	0.4124	0.0343
37	0.9067	0.4527	0.9067	0.3984	0.0352
38	0.8933	0.4453	0.8933	0.4101	0.0333
39	0.9067	0.4395	0.9067	0.4032	0.0345
40	0.8800	0.4552	0.8800	0.4326	0.0353
41	0.8800	0.4556	0.8800	0.4294	0.0353

```
     42      0.8867        0.4570       0.8867        0.4268  0.0354
     43      0.8933        0.4660       0.8933        0.4042  0.0372
     44      0.9133        0.4499       0.9133        0.3964  0.0329
     45      0.8867        0.4631       0.8867        0.4249  0.0336
     46      0.9000        0.4383       0.9000        0.4103  0.0358
     47      0.8733        0.4486       0.8733        0.4373  0.0352
     48      0.8933        0.4422       0.8933        0.4131  0.0329
     49      0.8600        0.4317       0.8600        0.4463  0.0350
     50      0.9067        0.4402       0.9067        0.4034  0.0343
     51      0.8733        0.4603       0.8733        0.4371  0.0346
     52      0.9000        0.4417       0.9000        0.4064  0.0325
     53      0.8933        0.4463       0.8933        0.4173  0.0343
     54      0.9000        0.4393       0.9000        0.4059  0.0353
     55      0.9067        0.4432       0.9067        0.4011  0.0362
     56      0.8867        0.4395       0.8867        0.4218  0.0352
     57      0.8933        0.4384       0.8933        0.4018  0.0352
     58      0.8867        0.4344       0.8867        0.4261  0.0329
     59      0.8800        0.4330       0.8800        0.4075  0.0340
     60      0.8933        0.4665       0.8933        0.4088  0.0350
     61      0.8867        0.4358       0.8867        0.4258  0.0313
     62      0.9000        0.4301       0.9000        0.4116  0.0344
     63      0.8800        0.4379       0.8800        0.4306  0.0347
     64      0.9133        0.4215       0.9133        0.4013  0.0345
     65      0.9067        0.4386       0.9067        0.4027  0.0355
     66      0.9133        0.4356       0.9133        0.4029  0.0336
     67      0.9133        0.4567       0.9133        0.3990  0.0359
     68      0.8867        0.4326       0.8867        0.4234  0.0347
     69      0.9000        0.4417       0.9000        0.4055  0.0362
     70      0.8867        0.4366       0.8867        0.4227  0.0312
     71      0.9000        0.4320       0.9000        0.4048  0.0351
     72      0.8933        0.4383       0.8933        0.4169  0.0344
     73      0.8733        0.4395       0.8733        0.4357  0.0359
     74      0.9067        0.4433       0.9067        0.4014  0.0344
     75      0.9067        0.4324       0.9067        0.4039  0.0351

     76      0.8867        0.4384       0.8867        0.4220  0.0354
     77      0.8933        0.4310       0.8933        0.4160  0.0352
     78      0.9000        0.4277       0.9000        0.4107  0.0320
     79      0.9067        0.4290       0.9067        0.4059  0.0333
     80      0.9000        0.4305       0.9000        0.4118  0.0332
     81      0.9000        0.4325       0.9000        0.4104  0.0350
     82      0.8867        0.4343       0.8867        0.4286  0.0356
     83      0.8933        0.4365       0.8933        0.4202  0.0346
Stopping since valid_loss has not improved in the last 40 epochs.
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5200        0.6940       0.5200        0.6911  0.0265
      2      0.5200        0.6899       0.5200        0.6910  0.0244
      3      0.5200        0.6917       0.5200        0.6909  0.0242
      4      0.5200        0.6927       0.5200        0.6908  0.0275
```

5	0.5200	0.6942	0.5200	0.6907	0.0281
6	0.5200	0.6936	0.5200	0.6906	0.0269
7	0.5200	0.6906	0.5200	0.6905	0.0269
8	0.5200	0.6916	0.5200	0.6904	0.0261
9	0.5200	0.6918	0.5200	0.6903	0.0254
10	0.5200	0.6920	0.5200	0.6902	0.0268
11	0.5200	0.6910	0.5200	0.6902	0.0270
12	0.5200	0.6927	0.5200	0.6901	0.0265
13	0.5200	0.6902	0.5200	0.6899	0.0274
14	0.5200	0.6920	0.5200	0.6898	0.0286
15	0.5200	0.6916	0.5200	0.6897	0.0286
16	0.5200	0.6913	0.5200	0.6897	0.0269
17	0.5200	0.6925	0.5200	0.6895	0.0262
18	0.5200	0.6923	0.5200	0.6894	0.0254
19	0.5200	0.6903	0.5200	0.6893	0.0289
20	0.5200	0.6895	0.5200	0.6892	0.0290
21	0.5200	0.6885	0.5200	0.6891	0.0295
22	0.5200	0.6917	0.5200	0.6890	0.0303
23	0.5200	0.6900	0.5200	0.6889	0.0298
24	0.5200	0.6930	0.5200	0.6888	0.0264
25	0.5200	0.6907	0.5200	0.6887	0.0288
26	0.5200	0.6901	0.5200	0.6886	0.0267
27	0.5200	0.6909	0.5200	0.6886	0.0287
28	0.5200	0.6910	0.5200	0.6885	0.0279
29	0.5200	0.6892	0.5200	0.6884	0.0291
30	0.5200	0.6924	0.5200	0.6883	0.0293
31	0.5200	0.6894	0.5200	0.6882	0.0277
32	0.5200	0.6910	0.5200	0.6881	0.0275
33	0.5200	0.6903	0.5200	0.6881	0.0281
34	0.5200	0.6900	0.5200	0.6880	0.0280
35	0.5200	0.6896	0.5200	0.6879	0.0256
36	0.5200	0.6874	0.5200	0.6877	0.0289
37	0.5200	0.6910	0.5200	0.6877	0.0274
38	0.5200	0.6914	0.5200	0.6876	0.0297
39	0.5200	0.6906	0.5200	0.6875	0.0257
40	0.5200	0.6887	0.5200	0.6874	0.0303
41	0.5200	0.6894	0.5200	0.6873	0.0294
42	0.5200	0.6889	0.5200	0.6872	0.0287
43	0.5200	0.6890	0.5200	0.6871	0.0288
44	0.5200	0.6902	0.5200	0.6870	0.0282
45	0.5200	0.6898	0.5200	0.6869	0.0269
46	0.5200	0.6885	0.5200	0.6868	0.0292
47	0.5200	0.6904	0.5200	0.6867	0.0276
48	0.5200	0.6898	0.5200	0.6866	0.0282
49	0.5200	0.6886	0.5200	0.6865	0.0277
50	0.5200	0.6909	0.5200	0.6864	0.0287
51	0.5200	0.6891	0.5200	0.6863	0.0278
52	0.5200	0.6901	0.5200	0.6862	0.0288
53	0.5200	0.6897	0.5200	0.6861	0.0290

54	0.5200	0.6886	0.5200	0.6859	0.0282
55	0.5200	0.6898	0.5200	0.6858	0.0285
56	0.5200	0.6884	0.5200	0.6857	0.0300
57	0.5200	0.6892	0.5200	0.6856	0.0266
58	0.5200	0.6875	0.5200	0.6854	0.0277
59	0.5200	0.6883	0.5200	0.6853	0.0282
60	0.5200	0.6908	0.5200	0.6852	0.0299
61	0.5200	0.6876	0.5200	0.6850	0.0285
62	0.5200	0.6877	0.5200	0.6849	0.0297
63	0.5200	0.6833	0.5200	0.6846	0.0292
64	0.5200	0.6889	0.5200	0.6845	0.0279
65	0.5200	0.6883	0.5200	0.6844	0.0285
66	0.5200	0.6889	0.5200	0.6842	0.0273
67	0.5200	0.6859	0.5200	0.6841	0.0285
68	0.5200	0.6873	0.5200	0.6839	0.0287
69	0.5200	0.6901	0.5200	0.6838	0.0263
70	0.5200	0.6879	0.5200	0.6836	0.0287
71	0.5200	0.6855	0.5200	0.6834	0.0280
72	0.5200	0.6866	0.5200	0.6832	0.0271
73	0.5200	0.6849	0.5200	0.6829	0.0275
74	0.5200	0.6884	0.5200	0.6827	0.0281
75	0.5200	0.6857	0.5200	0.6825	0.0264
76	0.5200	0.6858	0.5200	0.6822	0.0268
77	0.5200	0.6861	0.5200	0.6820	0.0280
78	0.5267	0.6853	0.5267	0.6818	0.0283
79	0.5333	0.6830	0.5333	0.6815	0.0308
80	0.5333	0.6872	0.5333	0.6813	0.0294
81	0.5400	0.6853	0.5400	0.6811	0.0271
82	0.5467	0.6832	0.5467	0.6808	0.0265
83	0.5533	0.6858	0.5533	0.6805	0.0279
84	0.5600	0.6841	0.5600	0.6802	0.0315
85	0.5733	0.6839	0.5733	0.6799	0.0285
86	0.5800	0.6828	0.5800	0.6796	0.0277
87	0.5800	0.6863	0.5800	0.6793	0.0286
88	0.5800	0.6838	0.5800	0.6790	0.0280
89	0.5800	0.6847	0.5800	0.6786	0.0288
90	0.6000	0.6873	0.6000	0.6783	0.0272
91	0.6067	0.6852	0.6067	0.6780	0.0288
92	0.6200	0.6806	0.6200	0.6776	0.0271
93	0.6200	0.6838	0.6200	0.6772	0.0298
94	0.6267	0.6819	0.6267	0.6768	0.0308
95	0.6333	0.6806	0.6333	0.6763	0.0255
96	0.6400	0.6818	0.6400	0.6760	0.0267
97	0.6533	0.6829	0.6533	0.6755	0.0305
98	0.6600	0.6826	0.6600	0.6751	0.0294
99	0.6667	0.6835	0.6667	0.6747	0.0291
100	0.6800	0.6795	0.6800	0.6742	0.0276

epoch	accuracy	train_loss	valid_acc	valid_loss	dur

1	0.5200	0.6936	0.5200	0.6926	0.0277
2	0.5200	0.6923	0.5200	0.6925	0.0253
3	0.5200	0.6919	0.5200	0.6924	0.0270
4	0.5200	0.6925	0.5200	0.6922	0.0257
5	0.5200	0.6932	0.5200	0.6921	0.0272
6	0.5200	0.6924	0.5200	0.6920	0.0291
7	0.5200	0.6928	0.5200	0.6919	0.0268
8	0.5200	0.6931	0.5200	0.6918	0.0264
9	0.5200	0.6912	0.5200	0.6917	0.0293
10	0.5200	0.6900	0.5200	0.6916	0.0286
11	0.5200	0.6905	0.5200	0.6915	0.0270
12	0.5200	0.6907	0.5200	0.6914	0.0281
13	0.5200	0.6919	0.5200	0.6913	0.0283
14	0.5200	0.6913	0.5200	0.6912	0.0261
15	0.5200	0.6919	0.5200	0.6911	0.0286
16	0.5200	0.6919	0.5200	0.6910	0.0277
17	0.5200	0.6911	0.5200	0.6909	0.0268
18	0.5200	0.6910	0.5200	0.6908	0.0287
19	0.5200	0.6916	0.5200	0.6907	0.0289
20	0.5200	0.6918	0.5200	0.6906	0.0291
21	0.5200	0.6892	0.5200	0.6905	0.0272
22	0.5200	0.6913	0.5200	0.6904	0.0276
23	0.5200	0.6910	0.5200	0.6903	0.0278
24	0.5200	0.6913	0.5200	0.6902	0.0297
25	0.5200	0.6916	0.5200	0.6901	0.0287
26	0.5200	0.6909	0.5200	0.6900	0.0280
27	0.5200	0.6890	0.5200	0.6898	0.0283
28	0.5200	0.6901	0.5200	0.6897	0.0295
29	0.5200	0.6909	0.5200	0.6896	0.0274
30	0.5200	0.6899	0.5200	0.6894	0.0262
31	0.5200	0.6895	0.5200	0.6893	0.0277
32	0.5200	0.6897	0.5200	0.6892	0.0285
33	0.5200	0.6913	0.5200	0.6890	0.0280
34	0.5200	0.6897	0.5200	0.6889	0.0285
35	0.5200	0.6881	0.5200	0.6887	0.0284
36	0.5200	0.6886	0.5200	0.6886	0.0278
37	0.5200	0.6891	0.5200	0.6885	0.0275
38	0.5200	0.6898	0.5200	0.6883	0.0292
39	0.5200	0.6908	0.5200	0.6881	0.0312
40	0.5200	0.6888	0.5200	0.6880	0.0297
41	0.5200	0.6880	0.5200	0.6878	0.0291
42	0.5200	0.6887	0.5200	0.6876	0.0317
43	0.5200	0.6897	0.5200	0.6875	0.0291
44	0.5200	0.6889	0.5200	0.6873	0.0285
45	0.5200	0.6886	0.5200	0.6872	0.0297
46	0.5200	0.6910	0.5200	0.6870	0.0294
47	0.5200	0.6855	0.5200	0.6868	0.0291
48	0.5200	0.6872	0.5200	0.6866	0.0283
49	0.5200	0.6887	0.5200	0.6864	0.0292

50	0.5200	0.6901	0.5200	0.6863	0.0288
51	0.5200	0.6882	0.5200	0.6861	0.0296
52	0.5200	0.6893	0.5200	0.6859	0.0277
53	0.5200	0.6868	0.5200	0.6857	0.0260
54	0.5200	0.6850	0.5200	0.6855	0.0282
55	0.5200	0.6875	0.5200	0.6852	0.0284
56	0.5200	0.6871	0.5200	0.6850	0.0303
57	0.5200	0.6875	0.5200	0.6848	0.0300
58	0.5200	0.6870	0.5200	0.6845	0.0291
59	0.5200	0.6866	0.5200	0.6843	0.0296
60	0.5200	0.6870	0.5200	0.6841	0.0285
61	0.5200	0.6898	0.5200	0.6839	0.0274
62	0.5200	0.6867	0.5200	0.6837	0.0254
63	0.5200	0.6875	0.5200	0.6835	0.0270
64	0.5200	0.6856	0.5200	0.6832	0.0279
65	0.5200	0.6857	0.5200	0.6829	0.0288
66	0.5200	0.6871	0.5200	0.6827	0.0278
67	0.5200	0.6838	0.5200	0.6824	0.0302
68	0.5200	0.6861	0.5200	0.6821	0.0288
69	0.5200	0.6858	0.5200	0.6818	0.0257
70	0.5200	0.6857	0.5200	0.6815	0.0270
71	0.5200	0.6814	0.5200	0.6811	0.0321
72	0.5200	0.6855	0.5200	0.6808	0.0287
73	0.5200	0.6829	0.5200	0.6804	0.0283
74	0.5200	0.6850	0.5200	0.6801	0.0279
75	0.5200	0.6847	0.5200	0.6798	0.0278
76	0.5467	0.6821	0.5467	0.6794	0.0275
77	0.5733	0.6834	0.5733	0.6790	0.0277
78	0.5733	0.6860	0.5733	0.6787	0.0288
79	0.5800	0.6823	0.5800	0.6782	0.0280
80	0.5800	0.6855	0.5800	0.6779	0.0289
81	0.6000	0.6815	0.6000	0.6774	0.0298
82	0.6067	0.6851	0.6067	0.6771	0.0282
83	0.6067	0.6821	0.6067	0.6766	0.0288
84	0.6067	0.6843	0.6067	0.6762	0.0282
85	0.6067	0.6844	0.6067	0.6758	0.0268
86	0.6067	0.6819	0.6067	0.6753	0.0262
87	0.6133	0.6831	0.6133	0.6749	0.0295
88	0.6133	0.6783	0.6133	0.6745	0.0291
89	0.6133	0.6849	0.6133	0.6741	0.0293
90	0.6200	0.6799	0.6200	0.6737	0.0287
91	0.6333	0.6808	0.6333	0.6732	0.0291
92	0.6467	0.6798	0.6467	0.6727	0.0300
93	0.6533	0.6791	0.6533	0.6721	0.0298
94	0.6600	0.6766	0.6600	0.6715	0.0328
95	0.6667	0.6747	0.6667	0.6709	0.0281
96	0.6800	0.6781	0.6800	0.6703	0.0287
97	0.6800	0.6777	0.6800	0.6698	0.0268
98	0.6867	0.6791	0.6867	0.6693	0.0286

```
    99      0.6867        0.6784       0.6867        0.6687  0.0260
   100      0.7067        0.6766       0.7067        0.6680  0.0272
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
     1      0.7867        0.6785       0.7867        0.6068  0.0362
     2      0.8400        0.5592       0.8400        0.4838  0.0351
     3      0.8133        0.5077       0.8133        0.4861  0.0391
     4      0.7933        0.4766       0.7933        0.4974  0.0367
     5      0.8067        0.4756       0.8067        0.4975  0.0373
     6      0.8133        0.5053       0.8133        0.4936  0.0349
     7      0.8200        0.4678       0.8200        0.4961  0.0346
     8      0.8267        0.4675       0.8267        0.4755  0.0351
     9      0.8133        0.4628       0.8133        0.4919  0.0346
    10      0.8133        0.4752       0.8133        0.4975  0.0325
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
     1      0.8667        0.6843       0.8667        0.6301  0.0329
     2      0.9000        0.5696       0.9000        0.4159  0.0354
     3      0.9000        0.4802       0.9000        0.4067  0.0359
     4      0.9000        0.4688       0.9000        0.4111  0.0363
     5      0.8867        0.4586       0.8867        0.4286  0.0343
     6      0.8867        0.4739       0.8867        0.4190  0.0351
     7      0.9000        0.4905       0.9000        0.4024  0.0357
     8      0.8933        0.4511       0.8933        0.4101  0.0346
     9      0.8800        0.4404       0.8800        0.4207  0.0358
    10      0.8867        0.4412       0.8867        0.4233  0.0364
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
     1      0.4800        0.6969       0.4800        0.6962  0.0257
     2      0.4800        0.6970       0.4800        0.6959  0.0266
     3      0.4800        0.6930       0.4800        0.6956  0.0255
     4      0.4800        0.6954       0.4800        0.6953  0.0260
     5      0.4800        0.6949       0.4800        0.6951  0.0271
     6      0.4800        0.6924       0.4800        0.6949  0.0272
     7      0.4800        0.6951       0.4800        0.6946  0.0324
     8      0.4800        0.6950       0.4800        0.6944  0.0367
     9      0.4800        0.6960       0.4800        0.6942  0.0302
    10      0.4800        0.6954       0.4800        0.6941  0.0283
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
     1      0.4800        0.6930       0.4800        0.6935  0.0418
     2      0.4800        0.6931       0.4800        0.6933  0.0450
     3      0.4800        0.6929       0.4800        0.6931  0.0435
     4      0.4800        0.6924       0.4800        0.6929  0.0372
     5      0.4800        0.6920       0.4800        0.6927  0.0470
     6      0.4800        0.6938       0.4800        0.6926  0.0434
     7      0.4933        0.6935       0.4933        0.6924  0.0357
     8      0.5067        0.6937       0.5067        0.6922  0.0347
     9      0.5867        0.6947       0.5867        0.6921  0.0335
```

```
     10      0.7400        0.6941       0.7400        0.6919  0.0311
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5333        0.6897       0.5333        0.6641  0.0345
      2      0.8400        0.6086       0.8400        0.4913  0.0352
      3      0.8067        0.4963       0.8067        0.4879  0.0362
      4      0.8200        0.4792       0.8200        0.4874  0.0359
      5      0.8067        0.4775       0.8067        0.4939  0.0352
      6      0.8133        0.4977       0.8133        0.4830  0.0360
      7      0.8333        0.4638       0.8333        0.4827  0.0373
      8      0.8067        0.4629       0.8067        0.4924  0.0350
      9      0.8467        0.4748       0.8467        0.4760  0.0366
     10      0.8000        0.4661       0.8000        0.5078  0.0363
     11      0.8333        0.4678       0.8333        0.4821  0.0370
     12      0.8333        0.4600       0.8333        0.4819  0.0376
     13      0.8267        0.4889       0.8267        0.4758  0.0350
     14      0.8067        0.4768       0.8067        0.5007  0.0377
     15      0.8267        0.4634       0.8267        0.4797  0.0365
     16      0.8333        0.4791       0.8333        0.4756  0.0354
     17      0.8333        0.4415       0.8333        0.4717  0.0357
     18      0.8333        0.4535       0.8333        0.4823  0.0355
     19      0.8333        0.4652       0.8333        0.4760  0.0346
     20      0.8400        0.4557       0.8400        0.4767  0.0351
     21      0.8333        0.4501       0.8333        0.4760  0.0352
     22      0.8133        0.4567       0.8133        0.5020  0.0369
     23      0.8400        0.4647       0.8400        0.4698  0.0363
     24      0.8333        0.4562       0.8333        0.4744  0.0379
     25      0.8133        0.4733       0.8133        0.4957  0.0339
     26      0.8067        0.4502       0.8067        0.4944  0.0342
     27      0.8400        0.4588       0.8400        0.4693  0.0363
     28      0.8200        0.4487       0.8200        0.4834  0.0331
     29      0.8400        0.4891       0.8400        0.4738  0.0343
     30      0.8267        0.4603       0.8267        0.4850  0.0355
     31      0.7667        0.4638       0.7667        0.5389  0.0337
     32      0.8267        0.4747       0.8267        0.4825  0.0370
     33      0.8200        0.4713       0.8200        0.4807  0.0358
     34      0.8200        0.4619       0.8200        0.4885  0.0357
     35      0.8333        0.4494       0.8333        0.4791  0.0367
     36      0.8400        0.4508       0.8400        0.4744  0.0346
     37      0.8200        0.4442       0.8200        0.4823  0.0351
     38      0.8200        0.4565       0.8200        0.4822  0.0348
     39      0.8333        0.4507       0.8333        0.4739  0.0353
     40      0.8267        0.4460       0.8267        0.4711  0.0319
     41      0.8333        0.4441       0.8333        0.4736  0.0390
     42      0.8133        0.4468       0.8133        0.4904  0.0337
     43      0.7867        0.4758       0.7867        0.5208  0.0334
     44      0.8267        0.4876       0.8267        0.4882  0.0344

     45      0.8400        0.4525       0.8400        0.4802  0.0369
     46      0.8267        0.4673       0.8267        0.4743  0.0363
```

```
     47      0.8467        0.4604       0.8467        0.4687  0.0382
     48      0.8333        0.4432       0.8333        0.4747  0.0339
     49      0.8333        0.4514       0.8333        0.4756  0.0349
     50      0.8400        0.4502       0.8400        0.4597  0.0360
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.6267        0.6757       0.6267        0.6026  0.0341
      2      0.9067        0.5539       0.9067        0.4132  0.0341
      3      0.8733        0.4704       0.8733        0.4368  0.0349
      4      0.8867        0.4769       0.8867        0.4219  0.0348
      5      0.8933        0.4514       0.8933        0.4175  0.0371
      6      0.8467        0.4970       0.8467        0.4663  0.0325
      7      0.8933        0.4746       0.8933        0.4204  0.0325
      8      0.9000        0.4499       0.9000        0.4093  0.0345
      9      0.8933        0.4764       0.8933        0.4076  0.0365
     10      0.8867        0.4481       0.8867        0.4218  0.0347
     11      0.9133        0.4571       0.9133        0.4007  0.0319
     12      0.8867        0.4561       0.8867        0.4202  0.0335
     13      0.8867        0.4508       0.8867        0.4311  0.0342
     14      0.9000        0.4805       0.9000        0.4053  0.0346
     15      0.8933        0.4469       0.8933        0.4090  0.0373
     16      0.8800        0.4560       0.8800        0.4318  0.0351
     17      0.9000        0.4649       0.9000        0.4073  0.0336
     18      0.8933        0.4383       0.8933        0.4094  0.0322
     19      0.8933        0.4489       0.8933        0.4105  0.0326
     20      0.9000        0.4737       0.9000        0.4084  0.0340
     21      0.9000        0.4447       0.9000        0.4029  0.0326
     22      0.9067        0.4415       0.9067        0.4018  0.0357
     23      0.9000        0.4308       0.9000        0.4034  0.0561
     24      0.8933        0.4483       0.8933        0.4151  0.0392
     25      0.9000        0.4379       0.9000        0.4104  0.0426
     26      0.9133        0.4509       0.9133        0.4018  0.0410
     27      0.9200        0.4577       0.9200        0.4008  0.0369
     28      0.8867        0.4624       0.8867        0.4270  0.0391
     29      0.8867        0.4620       0.8867        0.4230  0.0410
     30      0.9000        0.4490       0.9000        0.4097  0.0384
     31      0.9000        0.4418       0.9000        0.4134  0.0393
     32      0.8867        0.4762       0.8867        0.4283  0.0390
     33      0.9000        0.4827       0.9000        0.4125  0.0391
     34      0.9000        0.4681       0.9000        0.4099  0.0386
     35      0.8933        0.4602       0.8933        0.4151  0.0373
     36      0.8933        0.4336       0.8933        0.4144  0.0425
     37      0.8933        0.4706       0.8933        0.4189  0.0499
     38      0.9000        0.4343       0.9000        0.4012  0.0443
     39      0.8933        0.4499       0.8933        0.4199  0.0606
     40      0.8933        0.4567       0.8933        0.4085  0.0689
     41      0.9067        0.4626       0.9067        0.4048  0.0662
     42      0.8867        0.4543       0.8867        0.4169  0.0560
     43      0.8933        0.4587       0.8933        0.4188  0.0527
```

```
     44      0.8867        0.4476       0.8867        0.4232  0.1854
     45      0.9067        0.4446       0.9067        0.4063  0.1378
     46      0.9000        0.4509       0.9000        0.4099  0.0503
     47      0.8867        0.4597       0.8867        0.4285  0.0615
     48      0.9067        0.4437       0.9067        0.4064  0.0547
     49      0.9000        0.4586       0.9000        0.4144  0.0729
     50      0.8867        0.4492       0.8867        0.4195  0.0558
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.4800        0.6942       0.4800        0.6940  0.0363
      2      0.4800        0.6923       0.4800        0.6938  0.0414
      3      0.4800        0.6943       0.4800        0.6937  0.0397
      4      0.4800        0.6964       0.4800        0.6936  0.0633
      5      0.4800        0.6939       0.4800        0.6934  0.0445
      6      0.4800        0.6925       0.4800        0.6933  0.1161
      7      0.5133        0.6942       0.5133        0.6931  0.1681
      8      0.5200        0.6948       0.5200        0.6930  0.0436
      9      0.5200        0.6927       0.5200        0.6929  0.0406
     10      0.5200        0.6931       0.5200        0.6928  0.0467
     11      0.5200        0.6918       0.5200        0.6928  0.0454
     12      0.5200        0.6945       0.5200        0.6927  0.0391
     13      0.5200        0.6943       0.5200        0.6926  0.0534
     14      0.5200        0.6930       0.5200        0.6925  0.0397
     15      0.5200        0.6943       0.5200        0.6924  0.0381
     16      0.5200        0.6951       0.5200        0.6923  0.0399
     17      0.5200        0.6906       0.5200        0.6922  0.0397
     18      0.5200        0.6931       0.5200        0.6921  0.0393
     19      0.5200        0.6943       0.5200        0.6921  0.0423
     20      0.5200        0.6914       0.5200        0.6920  0.0438
     21      0.5200        0.6936       0.5200        0.6919  0.4594
     22      0.5200        0.6911       0.5200        0.6919  0.1257
     23      0.5200        0.6930       0.5200        0.6918  0.1338
     24      0.5200        0.6921       0.5200        0.6918  0.1032
     25      0.5200        0.6932       0.5200        0.6917  0.0844
     26      0.5200        0.6907       0.5200        0.6916  0.0621
     27      0.5200        0.6921       0.5200        0.6915  0.0426
     28      0.5200        0.6919       0.5200        0.6915  0.0377
     29      0.5200        0.6917       0.5200        0.6914  0.0381
     30      0.5200        0.6927       0.5200        0.6914  0.0535
     31      0.5200        0.6937       0.5200        0.6913  0.0438
     32      0.5200        0.6934       0.5200        0.6913  0.0507
     33      0.5200        0.6907       0.5200        0.6912  0.0481
     34      0.5200        0.6909       0.5200        0.6912  0.0569
     35      0.5200        0.6926       0.5200        0.6911  0.0409
     36      0.5200        0.6941       0.5200        0.6911  0.0317
     37      0.5200        0.6932       0.5200        0.6910  0.0372
     38      0.5200        0.6916       0.5200        0.6910  0.0353
     39      0.5200        0.6913       0.5200        0.6910  0.0331
     40      0.5200        0.6918       0.5200        0.6909  0.0337
```

41	0.5200	0.6923	0.5200	0.6909	0.0329
42	0.5200	0.6909	0.5200	0.6908	0.0423
43	0.5200	0.6909	0.5200	0.6908	0.0716
44	0.5200	0.6909	0.5200	0.6907	0.0421
45	0.5200	0.6916	0.5200	0.6907	0.0384
46	0.5200	0.6920	0.5200	0.6907	0.0402
47	0.5200	0.6912	0.5200	0.6906	0.0415
48	0.5200	0.6915	0.5200	0.6906	0.0332
49	0.5200	0.6905	0.5200	0.6905	0.0351
50	0.5200	0.6915	0.5200	0.6905	0.0318

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.4800	0.6943	0.4800	0.6947	0.0305
2	0.4800	0.6952	0.4800	0.6944	0.0311
3	0.4800	0.6935	0.4800	0.6942	0.0314
4	0.4800	0.6948	0.4800	0.6939	0.0400
5	0.4800	0.6954	0.4800	0.6937	0.0353
6	0.4800	0.6948	0.4800	0.6935	0.0306
7	0.4800	0.6928	0.4800	0.6933	0.0289
8	0.4800	0.6914	0.4800	0.6931	0.0291
9	0.4800	0.6923	0.4800	0.6929	0.0295
10	0.4800	0.6926	0.4800	0.6928	0.0266
11	0.4800	0.6939	0.4800	0.6926	0.0314
12	0.4733	0.6930	0.4733	0.6925	0.0279
13	0.5267	0.6946	0.5267	0.6923	0.0260
14	0.6267	0.6933	0.6267	0.6921	0.0272
15	0.6933	0.6917	0.6933	0.6920	0.0285
16	0.7133	0.6921	0.7133	0.6919	0.0294
17	0.6533	0.6916	0.6533	0.6918	0.0308
18	0.5933	0.6929	0.5933	0.6916	0.0293
19	0.5400	0.6908	0.5400	0.6915	0.0306
20	0.5267	0.6910	0.5267	0.6914	0.0303
21	0.5200	0.6916	0.5200	0.6913	0.0293
22	0.5200	0.6929	0.5200	0.6912	0.0294
23	0.5200	0.6923	0.5200	0.6912	0.0272
24	0.5200	0.6920	0.5200	0.6910	0.0289
25	0.5200	0.6897	0.5200	0.6909	0.0291
26	0.5200	0.6897	0.5200	0.6908	0.0277
27	0.5200	0.6919	0.5200	0.6908	0.0265
28	0.5200	0.6924	0.5200	0.6907	0.0267
29	0.5200	0.6921	0.5200	0.6906	0.0270
30	0.5200	0.6911	0.5200	0.6905	0.0351
31	0.5200	0.6927	0.5200	0.6904	0.0327
32	0.5200	0.6922	0.5200	0.6904	0.0315
33	0.5200	0.6906	0.5200	0.6903	0.0304
34	0.5200	0.6892	0.5200	0.6902	0.0291
35	0.5200	0.6919	0.5200	0.6901	0.0287
36	0.5200	0.6923	0.5200	0.6900	0.0271
37	0.5200	0.6901	0.5200	0.6900	0.0290

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
38	0.5200	0.6906	0.5200	0.6899	0.0344
39	0.5200	0.6918	0.5200	0.6898	0.0281
40	0.5200	0.6895	0.5200	0.6897	0.0290
41	0.5200	0.6903	0.5200	0.6896	0.0319
42	0.5200	0.6916	0.5200	0.6895	0.0357
43	0.5200	0.6916	0.5200	0.6894	0.0382
44	0.5200	0.6895	0.5200	0.6893	0.0298
45	0.5200	0.6909	0.5200	0.6893	0.0281
46	0.5200	0.6902	0.5200	0.6891	0.0309
47	0.5200	0.6908	0.5200	0.6891	0.0353
48	0.5200	0.6906	0.5200	0.6889	0.0321
49	0.5200	0.6903	0.5200	0.6889	0.0299
50	0.5200	0.6905	0.5200	0.6887	0.0274
1	0.7933	0.6865	0.7933	0.6149	0.0352
2	0.8200	0.5497	0.8200	0.4756	0.0381
3	0.8200	0.4881	0.8200	0.4751	0.0381
4	0.8200	0.4821	0.8200	0.4758	0.0498
5	0.8267	0.4799	0.8267	0.4907	0.0371
6	0.8000	0.4781	0.8000	0.4810	0.0338
7	0.8200	0.4574	0.8200	0.4824	0.0331
8	0.8200	0.4717	0.8200	0.4900	0.0356
9	0.8133	0.4605	0.8133	0.4880	0.0351
10	0.8333	0.4620	0.8333	0.4755	0.0337
11	0.8200	0.4637	0.8200	0.4883	0.0344
12	0.8267	0.4703	0.8267	0.4850	0.0345
13	0.8000	0.4584	0.8000	0.5088	0.0330
14	0.8200	0.5031	0.8200	0.4948	0.0341
15	0.8200	0.4646	0.8200	0.4911	0.0348
16	0.8200	0.4724	0.8200	0.4824	0.0331
17	0.8133	0.4598	0.8133	0.4938	0.0347
18	0.8333	0.4858	0.8333	0.4713	0.0336
19	0.7933	0.4757	0.7933	0.5150	0.0333
20	0.8133	0.4720	0.8133	0.4841	0.0341
21	0.8133	0.4545	0.8133	0.4800	0.0354
22	0.8467	0.4596	0.8467	0.4749	0.0343
23	0.8333	0.4522	0.8333	0.4759	0.0329
24	0.8533	0.4504	0.8533	0.4624	0.0333
25	0.8333	0.4468	0.8333	0.4783	0.0342
26	0.8400	0.4570	0.8400	0.4751	0.0381
27	0.8200	0.4661	0.8200	0.4817	0.0372
28	0.8400	0.4575	0.8400	0.4704	0.0361
29	0.8400	0.4731	0.8400	0.4742	0.0343
30	0.8400	0.4563	0.8400	0.4705	0.0354
31	0.8200	0.4695	0.8200	0.4865	0.0335
32	0.8333	0.4628	0.8333	0.4734	0.0359
33	0.8467	0.4553	0.8467	0.4644	0.0351
34	0.8400	0.4503	0.8400	0.4665	0.0318

35	0.8267	0.4497	0.8267	0.4729	0.0344
36	0.8333	0.4530	0.8333	0.4621	0.0336
37	0.8467	0.4353	0.8467	0.4672	0.0342
38	0.8267	0.4710	0.8267	0.4761	0.0356
39	0.8333	0.4746	0.8333	0.4734	0.0348
40	0.8267	0.4543	0.8267	0.4834	0.0380
41	0.8400	0.4445	0.8400	0.4692	0.0606
42	0.8333	0.4626	0.8333	0.4809	0.0560
43	0.8333	0.4465	0.8333	0.4819	0.0672
44	0.8267	0.4570	0.8267	0.4659	0.0771
45	0.8333	0.4368	0.8333	0.4751	0.0627
46	0.8467	0.4498	0.8467	0.4644	0.0477
47	0.8333	0.4519	0.8333	0.4649	0.0528
48	0.8400	0.4484	0.8400	0.4750	0.0757
49	0.8400	0.4451	0.8400	0.4800	0.0761
50	0.8467	0.4527	0.8467	0.4606	0.0877
51	0.8400	0.4520	0.8400	0.4751	0.0758
52	0.8333	0.4521	0.8333	0.4753	0.0529
53	0.8067	0.4581	0.8067	0.4870	0.0588
54	0.8400	0.4637	0.8400	0.4774	0.0448
55	0.8467	0.4424	0.8467	0.4648	0.0442
56	0.8467	0.4323	0.8467	0.4581	0.0619
57	0.8400	0.4438	0.8400	0.4710	0.0637
58	0.8467	0.4509	0.8467	0.4551	0.0616
59	0.8333	0.4382	0.8333	0.4732	0.0595
60	0.8333	0.4301	0.8333	0.4690	0.0545
61	0.8333	0.4412	0.8333	0.4710	0.0530
62	0.8467	0.4443	0.8467	0.4635	0.0519
63	0.8533	0.4312	0.8533	0.4631	0.0509
64	0.8400	0.4398	0.8400	0.4750	0.0441
65	0.8533	0.4572	0.8533	0.4558	0.0632
66	0.8400	0.4424	0.8400	0.4621	0.0486
67	0.8400	0.4408	0.8400	0.4713	0.0522
68	0.8400	0.4399	0.8400	0.4626	0.0603
69	0.8333	0.4480	0.8333	0.4790	0.0895
70	0.8333	0.4588	0.8333	0.4790	0.0667
71	0.8333	0.4480	0.8333	0.4770	0.0623
72	0.8467	0.4546	0.8467	0.4624	0.0679
73	0.8400	0.4518	0.8400	0.4652	0.0701
74	0.8333	0.4416	0.8333	0.4742	0.0727
75	0.8333	0.4464	0.8333	0.4613	0.0720
76	0.8333	0.4464	0.8333	0.4702	0.0941
77	0.8400	0.4426	0.8400	0.4725	0.0528
78	0.8400	0.4482	0.8400	0.4756	0.0600
79	0.8333	0.4482	0.8333	0.4685	0.0632
80	0.8400	0.4401	0.8400	0.4676	0.1166
81	0.8333	0.4508	0.8333	0.4591	0.1052
82	0.8400	0.4382	0.8400	0.4672	0.0516
83	0.8333	0.4397	0.8333	0.4723	0.0498

84	0.8467	0.4565	0.8467	0.4596	0.0589
85	0.8467	0.4416	0.8467	0.4595	0.0438
86	0.8467	0.4420	0.8467	0.4652	0.0404
87	0.8667	0.4434	0.8667	0.4453	0.0354
88	0.8400	0.4413	0.8400	0.4687	0.0348
89	0.8600	0.4477	0.8600	0.4421	0.0350
90	0.8467	0.4457	0.8467	0.4636	0.0371
91	0.8400	0.4571	0.8400	0.4567	0.0459
92	0.8533	0.4483	0.8533	0.4471	0.0377
93	0.8400	0.4473	0.8400	0.4671	0.0364
94	0.8400	0.4361	0.8400	0.4680	0.0390
95	0.8333	0.4504	0.8333	0.4737	0.0430
96	0.8533	0.4544	0.8533	0.4615	0.0495
97	0.8400	0.4555	0.8400	0.4785	0.0431
98	0.8533	0.4608	0.8533	0.4631	0.0399
99	0.8467	0.4641	0.8467	0.4603	0.0441
100	0.8400	0.4476	0.8400	0.4727	0.0386

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.8467	0.6844	0.8467	0.6153	0.0367
2	0.9000	0.5697	0.9000	0.4151	0.0414
3	0.9000	0.5005	0.9000	0.4063	0.0397
4	0.9000	0.4766	0.9000	0.4077	0.0392
5	0.9000	0.4623	0.9000	0.4046	0.0391
6	0.8933	0.4568	0.8933	0.4183	0.0406
7	0.8800	0.4633	0.8800	0.4365	0.0410
8	0.8800	0.4586	0.8800	0.4269	0.0395
9	0.8600	0.4641	0.8600	0.4474	0.0435
10	0.9000	0.4607	0.9000	0.4147	0.0450
11	0.8600	0.4647	0.8600	0.4461	0.0417
12	0.8867	0.4625	0.8867	0.4201	0.0412
13	0.9133	0.4455	0.9133	0.4010	0.0459
14	0.8867	0.4609	0.8867	0.4116	0.0412
15	0.8867	0.4485	0.8867	0.4196	0.0495
16	0.8800	0.4647	0.8800	0.4284	0.0378
17	0.8867	0.4466	0.8867	0.4234	0.0447
18	0.9267	0.4410	0.9267	0.3936	0.0618
19	0.9000	0.4411	0.9000	0.4093	0.0599
20	0.8867	0.4518	0.8867	0.4216	0.0532
21	0.8867	0.4466	0.8867	0.4222	0.0707
22	0.9200	0.4605	0.9200	0.3977	0.0457
23	0.9067	0.4359	0.9067	0.3976	0.0423
24	0.8933	0.4876	0.8933	0.4139	0.0422
25	0.8933	0.4383	0.8933	0.4085	0.0457
26	0.8867	0.4584	0.8867	0.4307	0.0462
27	0.8867	0.4654	0.8867	0.4155	0.0466
28	0.8800	0.4556	0.8800	0.4303	0.0357
29	0.9067	0.4528	0.9067	0.4021	0.0354
30	0.8933	0.4398	0.8933	0.4153	0.0417

```
     31      0.8800        0.4339       0.8800        0.4338  0.0482
     32      0.8867        0.4404       0.8867        0.4156  0.0364
     33      0.9067        0.4580       0.9067        0.4061  0.0390
     34      0.8867        0.4445       0.8867        0.4220  0.0347
     35      0.9000        0.4486       0.9000        0.4033  0.0469
     36      0.8800        0.4383       0.8800        0.4310  0.0431
     37      0.8933        0.4394       0.8933        0.4148  0.0617
     38      0.8867        0.4479       0.8867        0.4188  0.0783
     39      0.8933        0.4561       0.8933        0.4198  0.0372
     40      0.9133        0.4471       0.9133        0.3982  0.0357
     41      0.9067        0.4525       0.9067        0.4035  0.0365
     42      0.9067        0.4380       0.9067        0.4067  0.0522
     43      0.8867        0.4311       0.8867        0.4266  0.0386
     44      0.9200        0.4436       0.9200        0.3987  0.0412
     45      0.8867        0.4331       0.8867        0.4211  0.0450
     46      0.8933        0.4359       0.8933        0.4090  0.0414
     47      0.8867        0.4396       0.8867        0.4326  0.0388
     48      0.9067        0.4484       0.9067        0.4070  0.0395
     49      0.8867        0.4533       0.8867        0.4340  0.0406
     50      0.9133        0.4572       0.9133        0.4015  0.0398
     51      0.8867        0.4389       0.8867        0.4298  0.0390
     52      0.9133        0.4379       0.9133        0.3974  0.0374
     53      0.9000        0.4374       0.9000        0.4046  0.0418
     54      0.9133        0.4450       0.9133        0.4033  0.0416
     55      0.9067        0.4378       0.9067        0.4022  0.0453
     56      0.8867        0.4446       0.8867        0.4244  0.0373
     57      0.8933        0.4582       0.8933        0.4180  0.0404
Stopping since valid_loss has not improved in the last 40 epochs.
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.4800        0.6939       0.4800        0.6944  0.0247
      2      0.4800        0.6948       0.4800        0.6942  0.0269
      3      0.4800        0.6958       0.4800        0.6940  0.0288

      4      0.4800        0.6931       0.4800        0.6939  0.0272
      5      0.4800        0.6938       0.4800        0.6937  0.0290
      6      0.4800        0.6919       0.4800        0.6936  0.0268
      7      0.4800        0.6930       0.4800        0.6934  0.0265
      8      0.4800        0.6933       0.4800        0.6933  0.0260
      9      0.4867        0.6926       0.4867        0.6931  0.0269
     10      0.5000        0.6926       0.5000        0.6929  0.0261
     11      0.5200        0.6919       0.5200        0.6928  0.0279
     12      0.5733        0.6943       0.5733        0.6927  0.0282
     13      0.5867        0.6925       0.5867        0.6925  0.0293
     14      0.6200        0.6918       0.6200        0.6924  0.0256
     15      0.6267        0.6932       0.6267        0.6923  0.0276
     16      0.5733        0.6927       0.5733        0.6922  0.0284
     17      0.5267        0.6931       0.5267        0.6921  0.0281
     18      0.5200        0.6932       0.5200        0.6920  0.0272
     19      0.5200        0.6935       0.5200        0.6918  0.0263
```

20	0.5200	0.6925	0.5200	0.6917	0.0278
21	0.5200	0.6907	0.5200	0.6916	0.0267
22	0.5200	0.6926	0.5200	0.6915	0.0278
23	0.5200	0.6917	0.5200	0.6915	0.0259
24	0.5200	0.6905	0.5200	0.6914	0.0278
25	0.5200	0.6922	0.5200	0.6913	0.0254
26	0.5200	0.6914	0.5200	0.6912	0.0283
27	0.5200	0.6930	0.5200	0.6911	0.0390
28	0.5200	0.6919	0.5200	0.6910	0.0315
29	0.5200	0.6925	0.5200	0.6909	0.0327
30	0.5200	0.6916	0.5200	0.6908	0.0316
31	0.5200	0.6904	0.5200	0.6907	0.0289
32	0.5200	0.6901	0.5200	0.6906	0.0280
33	0.5200	0.6912	0.5200	0.6905	0.0288
34	0.5200	0.6915	0.5200	0.6904	0.0246
35	0.5200	0.6914	0.5200	0.6903	0.0286
36	0.5200	0.6914	0.5200	0.6902	0.0292
37	0.5200	0.6912	0.5200	0.6902	0.0302
38	0.5200	0.6912	0.5200	0.6901	0.0307
39	0.5200	0.6907	0.5200	0.6900	0.0367
40	0.5200	0.6929	0.5200	0.6899	0.0366
41	0.5200	0.6896	0.5200	0.6898	0.0300
42	0.5200	0.6916	0.5200	0.6897	0.0298
43	0.5200	0.6920	0.5200	0.6897	0.0294
44	0.5200	0.6914	0.5200	0.6896	0.0371
45	0.5200	0.6908	0.5200	0.6895	0.0309
46	0.5200	0.6883	0.5200	0.6894	0.0338
47	0.5200	0.6887	0.5200	0.6893	0.0384
48	0.5200	0.6902	0.5200	0.6892	0.0378
49	0.5200	0.6909	0.5200	0.6891	0.0312
50	0.5200	0.6883	0.5200	0.6889	0.0312
51	0.5200	0.6862	0.5200	0.6888	0.0403
52	0.5200	0.6887	0.5200	0.6887	0.0285
53	0.5200	0.6893	0.5200	0.6886	0.0298
54	0.5200	0.6904	0.5200	0.6885	0.0289
55	0.5200	0.6906	0.5200	0.6884	0.0300
56	0.5200	0.6919	0.5200	0.6883	0.0292
57	0.5200	0.6885	0.5200	0.6882	0.0297
58	0.5200	0.6873	0.5200	0.6881	0.0279
59	0.5200	0.6880	0.5200	0.6880	0.0292
60	0.5200	0.6885	0.5200	0.6878	0.0270
61	0.5200	0.6899	0.5200	0.6877	0.0297
62	0.5200	0.6878	0.5200	0.6875	0.0279
63	0.5200	0.6911	0.5200	0.6874	0.0363
64	0.5200	0.6870	0.5200	0.6873	0.0365
65	0.5200	0.6891	0.5200	0.6872	0.0336
66	0.5200	0.6893	0.5200	0.6870	0.0339
67	0.5200	0.6868	0.5200	0.6869	0.0633
68	0.5200	0.6874	0.5200	0.6868	0.0674

69	0.5200	0.6872	0.5200	0.6866	0.0459
70	0.5200	0.6863	0.5200	0.6865	0.0743
71	0.5200	0.6888	0.5200	0.6863	0.0576
72	0.5200	0.6854	0.5200	0.6862	0.0521
73	0.5200	0.6874	0.5200	0.6860	0.0679
74	0.5200	0.6881	0.5200	0.6859	0.0352
75	0.5200	0.6848	0.5200	0.6856	0.1001
76	0.5200	0.6867	0.5200	0.6854	0.0774
77	0.5200	0.6887	0.5200	0.6853	0.0568
78	0.5200	0.6899	0.5200	0.6851	0.0658
79	0.5200	0.6852	0.5200	0.6849	0.0658
80	0.5200	0.6866	0.5200	0.6847	0.0793
81	0.5200	0.6868	0.5200	0.6846	0.1093
82	0.5200	0.6837	0.5200	0.6843	0.0702
83	0.5200	0.6869	0.5200	0.6841	0.1108
84	0.5200	0.6851	0.5200	0.6838	0.1019
85	0.5200	0.6846	0.5200	0.6836	0.0836
86	0.5200	0.6860	0.5200	0.6833	0.0750
87	0.5200	0.6867	0.5200	0.6831	0.0432
88	0.5200	0.6833	0.5200	0.6828	0.0435
89	0.5200	0.6856	0.5200	0.6825	0.0476
90	0.5200	0.6822	0.5200	0.6822	0.0450
91	0.5200	0.6862	0.5200	0.6820	0.0460
92	0.5200	0.6835	0.5200	0.6816	0.0441
93	0.5200	0.6846	0.5200	0.6813	0.0390
94	0.5200	0.6810	0.5200	0.6810	0.0472
95	0.5200	0.6839	0.5200	0.6807	0.0642
96	0.5200	0.6854	0.5200	0.6804	0.0375
97	0.5200	0.6837	0.5200	0.6801	0.0601
98	0.5200	0.6804	0.5200	0.6797	0.1363
99	0.5200	0.6835	0.5200	0.6793	0.0755
100	0.5200	0.6835	0.5200	0.6790	0.0802

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.5200	0.6938	0.5200	0.6936	0.0396
2	0.5200	0.6936	0.5200	0.6935	0.0385
3	0.5200	0.6927	0.5200	0.6935	0.0450
4	0.5200	0.6917	0.5200	0.6934	0.0661
5	0.5200	0.6930	0.5200	0.6934	0.0526
6	0.5200	0.6925	0.5200	0.6933	0.0452
7	0.5200	0.6940	0.5200	0.6933	0.0495
8	0.5200	0.6921	0.5200	0.6933	0.0629
9	0.5200	0.6937	0.5200	0.6932	0.0974
10	0.5200	0.6920	0.5200	0.6932	0.0610
11	0.5200	0.6938	0.5200	0.6931	0.0442
12	0.5200	0.6928	0.5200	0.6931	0.0597
13	0.5200	0.6919	0.5200	0.6931	0.0378
14	0.5200	0.6934	0.5200	0.6930	0.0470
15	0.5200	0.6919	0.5200	0.6930	0.0421

16	0.5200	0.6922	0.5200	0.6930	0.0447
17	0.5200	0.6930	0.5200	0.6929	0.0470
18	0.5200	0.6926	0.5200	0.6929	0.0440
19	0.5200	0.6920	0.5200	0.6929	0.0403
20	0.5200	0.6928	0.5200	0.6928	0.0452
21	0.5200	0.6916	0.5200	0.6928	0.0501
22	0.5200	0.6935	0.5200	0.6927	0.0588
23	0.5200	0.6934	0.5200	0.6927	0.0712
24	0.5200	0.6931	0.5200	0.6927	0.0409
25	0.5200	0.6929	0.5200	0.6927	0.0440
26	0.5200	0.6922	0.5200	0.6926	0.0464
27	0.5200	0.6930	0.5200	0.6926	0.0361
28	0.5200	0.6928	0.5200	0.6926	0.0328
29	0.5200	0.6925	0.5200	0.6925	0.0366
30	0.5200	0.6932	0.5200	0.6925	0.0381
31	0.5200	0.6924	0.5200	0.6924	0.0308
32	0.5200	0.6922	0.5200	0.6924	0.0322
33	0.5200	0.6924	0.5200	0.6924	0.0395
34	0.5200	0.6924	0.5200	0.6924	0.0315
35	0.5200	0.6937	0.5200	0.6923	0.0296
36	0.5200	0.6917	0.5200	0.6923	0.0290
37	0.5200	0.6919	0.5200	0.6923	0.0292
38	0.5200	0.6928	0.5200	0.6922	0.0294
39	0.5200	0.6916	0.5200	0.6922	0.0436
40	0.5200	0.6918	0.5200	0.6922	0.0345
41	0.5200	0.6916	0.5200	0.6921	0.0302
42	0.5200	0.6927	0.5200	0.6921	0.0262
43	0.5200	0.6915	0.5200	0.6921	0.0280
44	0.5200	0.6912	0.5200	0.6920	0.0252
45	0.5200	0.6921	0.5200	0.6920	0.0276
46	0.5200	0.6908	0.5200	0.6919	0.0287
47	0.5200	0.6915	0.5200	0.6919	0.0266
48	0.5200	0.6913	0.5200	0.6918	0.0265
49	0.5200	0.6918	0.5200	0.6918	0.0284
50	0.5200	0.6930	0.5200	0.6918	0.0293
51	0.5200	0.6914	0.5200	0.6917	0.0296
52	0.5200	0.6905	0.5200	0.6917	0.0288
53	0.5200	0.6932	0.5200	0.6916	0.0355
54	0.5200	0.6914	0.5200	0.6916	0.0287
55	0.5200	0.6919	0.5200	0.6915	0.0331
56	0.5200	0.6914	0.5200	0.6915	0.0315
57	0.5200	0.6917	0.5200	0.6914	0.0320
58	0.5200	0.6913	0.5200	0.6914	0.0316
59	0.5200	0.6901	0.5200	0.6913	0.0311
60	0.5200	0.6911	0.5200	0.6913	0.0267
61	0.5200	0.6908	0.5200	0.6912	0.0322
62	0.5200	0.6909	0.5200	0.6911	0.0320
63	0.5200	0.6906	0.5200	0.6911	0.0320
64	0.5200	0.6906	0.5200	0.6910	0.0300

```
     65      0.5200        0.6910       0.5200        0.6909  0.0316
     66      0.5200        0.6911       0.5200        0.6908  0.0307
     67      0.5200        0.6921       0.5200        0.6907  0.0281
     68      0.5200        0.6894       0.5200        0.6906  0.0314
     69      0.5200        0.6906       0.5200        0.6905  0.0308
     70      0.5200        0.6904       0.5200        0.6904  0.0301
     71      0.5200        0.6901       0.5200        0.6903  0.0297
     72      0.5200        0.6903       0.5200        0.6902  0.0278
     73      0.5200        0.6911       0.5200        0.6901  0.0292
     74      0.5200        0.6907       0.5200        0.6900  0.0280
     75      0.5200        0.6911       0.5200        0.6899  0.0321
     76      0.5200        0.6903       0.5200        0.6897  0.0543
     77      0.5200        0.6902       0.5200        0.6896  0.0465
     78      0.5200        0.6897       0.5200        0.6895  0.0477
     79      0.5200        0.6905       0.5200        0.6893  0.0353
     80      0.5200        0.6896       0.5200        0.6892  0.0353
     81      0.5200        0.6892       0.5200        0.6891  0.0332
     82      0.5200        0.6896       0.5200        0.6890  0.0371
     83      0.5200        0.6904       0.5200        0.6889  0.0353
     84      0.5200        0.6888       0.5200        0.6887  0.0315
     85      0.5200        0.6914       0.5200        0.6886  0.0335
     86      0.5200        0.6895       0.5200        0.6884  0.0343
     87      0.5200        0.6886       0.5200        0.6882  0.0354
     88      0.5200        0.6890       0.5200        0.6881  0.0370
     89      0.5200        0.6886       0.5200        0.6880  0.0356
     90      0.5200        0.6886       0.5200        0.6878  0.0373
     91      0.5200        0.6922       0.5200        0.6877  0.0376
     92      0.5200        0.6890       0.5200        0.6875  0.0368
     93      0.5200        0.6891       0.5200        0.6873  0.0391
     94      0.5200        0.6880       0.5200        0.6872  0.0363
     95      0.5200        0.6887       0.5200        0.6870  0.0373
     96      0.5200        0.6889       0.5200        0.6869  0.0372
     97      0.5200        0.6892       0.5200        0.6867  0.0350
     98      0.5200        0.6882       0.5200        0.6865  0.0326
     99      0.5200        0.6867       0.5200        0.6864  0.0325
    100      0.5200        0.6880       0.5200        0.6862  0.0358
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.7667        0.6850       0.7667        0.6433  0.0451
      2      0.8200        0.5764       0.8200        0.4918  0.0442
      3      0.8333        0.5048       0.8333        0.4779  0.0546

      4      0.8133        0.4955       0.8133        0.4793  0.0734
      5      0.8133        0.4790       0.8133        0.4839  0.0470
      6      0.8133        0.4746       0.8133        0.4898  0.0642
      7      0.8067        0.4901       0.8067        0.5069  0.0453
      8      0.8133        0.4893       0.8133        0.4883  0.0442
      9      0.8333        0.4634       0.8333        0.4726  0.0424
     10      0.8067        0.4567       0.8067        0.4918  0.0431
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
```

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.8867	0.6827	0.8867	0.5950	0.0383
2	0.9133	0.5606	0.9133	0.4119	0.0531
3	0.8733	0.4719	0.8733	0.4376	0.0551
4	0.9000	0.4635	0.9000	0.4101	0.0538
5	0.8733	0.4604	0.8733	0.4397	0.1115
6	0.8533	0.4683	0.8533	0.4565	0.0650
7	0.8800	0.4598	0.8800	0.4197	0.0706
8	0.9000	0.4255	0.9000	0.4049	0.0848
9	0.9000	0.4419	0.9000	0.4083	0.0914
10	0.9133	0.4522	0.9133	0.4002	0.0669

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.4800	0.6957	0.4800	0.6956	0.0372
2	0.4800	0.6956	0.4800	0.6955	0.0911
3	0.4800	0.6952	0.4800	0.6953	0.0711
4	0.4800	0.6948	0.4800	0.6952	0.0474
5	0.4800	0.6949	0.4800	0.6950	0.0534
6	0.4800	0.6930	0.4800	0.6949	0.0607
7	0.4800	0.6939	0.4800	0.6948	0.0547
8	0.4800	0.6939	0.4800	0.6947	0.0578
9	0.4800	0.6949	0.4800	0.6945	0.0720
10	0.4800	0.6942	0.4800	0.6944	0.0432

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.4800	0.6964	0.4800	0.6947	0.4256
2	0.4800	0.6969	0.4800	0.6945	0.1462
3	0.4800	0.6961	0.4800	0.6942	0.1601
4	0.4800	0.6935	0.4800	0.6940	0.0595
5	0.4733	0.6943	0.4733	0.6938	0.0611
6	0.4267	0.6950	0.4267	0.6936	0.0785
7	0.4267	0.6950	0.4267	0.6934	0.0744
8	0.4800	0.6932	0.4800	0.6932	0.1026
9	0.5800	0.6944	0.5800	0.6930	0.0921
10	0.5200	0.6937	0.5200	0.6928	0.0394

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.8333	0.6816	0.8333	0.6058	0.0521
2	0.8200	0.5715	0.8200	0.4796	0.0574
3	0.8067	0.4853	0.8067	0.4841	0.0542
4	0.8133	0.4766	0.8133	0.4910	0.0652
5	0.8067	0.4790	0.8067	0.4909	0.0525
6	0.8267	0.4766	0.8267	0.4915	0.0376
7	0.8067	0.4623	0.8067	0.4942	0.0380
8	0.8333	0.4767	0.8333	0.4842	0.0380
9	0.8133	0.4661	0.8133	0.4856	0.0346
10	0.8133	0.4692	0.8133	0.4863	0.0360
11	0.8200	0.4672	0.8200	0.4786	0.0358
12	0.8267	0.4746	0.8267	0.4803	0.0497

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
13	0.8133	0.4536	0.8133	0.4987	0.0473
14	0.8133	0.4895	0.8133	0.4988	0.0605
15	0.8267	0.4596	0.8267	0.4853	0.0467
16	0.8333	0.4753	0.8333	0.4809	0.0520
17	0.8333	0.4626	0.8333	0.4843	0.0510
18	0.8267	0.4613	0.8267	0.4832	0.0631
19	0.8333	0.4657	0.8333	0.4719	0.0677
20	0.8400	0.4514	0.8400	0.4735	0.0547
21	0.8133	0.4647	0.8133	0.4944	0.0495
22	0.8333	0.4781	0.8333	0.4797	0.0478
23	0.8200	0.4426	0.8200	0.4882	0.0485
24	0.8333	0.4476	0.8333	0.4755	0.0435
25	0.8067	0.4555	0.8067	0.4946	0.0410
26	0.8200	0.4628	0.8200	0.4867	0.0372
27	0.8200	0.4594	0.8200	0.4862	0.0367
28	0.8333	0.4824	0.8333	0.4763	0.0359
29	0.8267	0.4685	0.8267	0.4850	0.0364
30	0.8267	0.4635	0.8267	0.4855	0.0326
31	0.8267	0.4571	0.8267	0.4854	0.0364
32	0.8333	0.4537	0.8333	0.4811	0.0357
33	0.8133	0.4442	0.8133	0.4838	0.0370
34	0.8133	0.4614	0.8133	0.4970	0.0371
35	0.8267	0.4712	0.8267	0.4806	0.0385
36	0.8133	0.4644	0.8133	0.4890	0.0362
37	0.8333	0.4527	0.8333	0.4792	0.0376
38	0.8200	0.4678	0.8200	0.4928	0.0367
39	0.8200	0.4468	0.8200	0.4771	0.0342
40	0.8267	0.4581	0.8267	0.4850	0.0402
41	0.8333	0.4664	0.8333	0.4750	0.0387
42	0.8200	0.4662	0.8200	0.4898	0.0391
43	0.8267	0.4564	0.8267	0.4772	0.0364
44	0.8133	0.4579	0.8133	0.4910	0.0374
45	0.8400	0.4636	0.8400	0.4723	0.0379
46	0.8400	0.4445	0.8400	0.4764	0.0398
47	0.8400	0.4410	0.8400	0.4753	0.0407
48	0.8200	0.4530	0.8200	0.4756	0.0420
49	0.8267	0.4577	0.8267	0.4748	0.0425
50	0.8267	0.4476	0.8267	0.4730	0.0380
epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.8133	0.6830	0.8133	0.5926	0.0353
2	0.9067	0.5564	0.9067	0.4184	0.0361
3	0.9133	0.4822	0.9133	0.4016	0.0345
4	0.9000	0.4796	0.9000	0.4137	0.0350
5	0.9000	0.4642	0.9000	0.4084	0.0393
6	0.9067	0.4704	0.9067	0.4068	0.0406
7	0.8933	0.4749	0.8933	0.4237	0.0407
8	0.9067	0.4648	0.9067	0.4003	0.0412

```
      9      0.9000        0.4538       0.9000        0.4075  0.0382
     10      0.8867        0.4457       0.8867        0.4251  0.0371

     11      0.8800        0.4732       0.8800        0.4187  0.0399
     12      0.9000        0.4534       0.9000        0.4037  0.0377
     13      0.9000        0.4651       0.9000        0.4056  0.0378
     14      0.8933        0.4510       0.8933        0.4130  0.0357
     15      0.8933        0.4435       0.8933        0.4106  0.0387
     16      0.9000        0.4691       0.9000        0.4073  0.0390
     17      0.9133        0.4397       0.9133        0.4040  0.0336
     18      0.8467        0.4634       0.8467        0.4642  0.0356
     19      0.9200        0.4484       0.9200        0.4019  0.0396
     20      0.8867        0.4418       0.8867        0.4274  0.0486
     21      0.8867        0.4487       0.8867        0.4240  0.0516
     22      0.8867        0.4544       0.8867        0.4262  0.0567
     23      0.8867        0.4614       0.8867        0.4226  0.0507
     24      0.9133        0.4464       0.9133        0.3984  0.0465
     25      0.9200        0.4627       0.9200        0.3961  0.0490
     26      0.9133        0.4473       0.9133        0.3983  0.0474
     27      0.9000        0.4460       0.9000        0.4162  0.0337
     28      0.9000        0.4681       0.9000        0.4145  0.0440
     29      0.9067        0.4639       0.9067        0.4028  0.0466
     30      0.9000        0.4545       0.9000        0.4116  0.0497
     31      0.8867        0.4387       0.8867        0.4261  0.0474
     32      0.9133        0.4418       0.9133        0.3987  0.0426
     33      0.8867        0.4443       0.8867        0.4259  0.0384
     34      0.9000        0.4401       0.9000        0.4067  0.0385
     35      0.9000        0.4414       0.9000        0.4027  0.0392
     36      0.9000        0.4367       0.9000        0.4022  0.0345
     37      0.9133        0.4445       0.9133        0.4025  0.0353
     38      0.8867        0.4474       0.8867        0.4287  0.0364
     39      0.9133        0.4480       0.9133        0.3987  0.0355
     40      0.8867        0.4324       0.8867        0.4217  0.0360
     41      0.9000        0.4417       0.9000        0.4068  0.0359
     42      0.8933        0.4466       0.8933        0.4173  0.0348
     43      0.8733        0.4460       0.8733        0.4320  0.0374
     44      0.9067        0.4477       0.9067        0.4051  0.0375
     45      0.9067        0.4487       0.9067        0.4106  0.0375
     46      0.8867        0.4256       0.8867        0.4171  0.0356
     47      0.8933        0.4348       0.8933        0.4174  0.0341
     48      0.9000        0.4521       0.9000        0.4063  0.0369
     49      0.9000        0.4516       0.9000        0.4014  0.0377
     50      0.9000        0.4367       0.9000        0.3985  0.0373
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5200        0.6946       0.5200        0.6932  0.0280
      2      0.5200        0.6925       0.5200        0.6931  0.0274
      3      0.5200        0.6943       0.5200        0.6931  0.0256
      4      0.5200        0.6926       0.5200        0.6930  0.0289
      5      0.5200        0.6939       0.5200        0.6930  0.0285
```

```
      6      0.5200        0.6935       0.5200        0.6929  0.0301
      7      0.5200        0.6937       0.5200        0.6929  0.0329
      8      0.5200        0.6933       0.5200        0.6929  0.0316
      9      0.5200        0.6945       0.5200        0.6928  0.0304
     10      0.5200        0.6921       0.5200        0.6928  0.0275
     11      0.5200        0.6925       0.5200        0.6927  0.0288
     12      0.5200        0.6938       0.5200        0.6927  0.0299
     13      0.5200        0.6928       0.5200        0.6927  0.0312
     14      0.5200        0.6922       0.5200        0.6926  0.0325
     15      0.5200        0.6921       0.5200        0.6926  0.0312
     16      0.5200        0.6932       0.5200        0.6926  0.0304
     17      0.5200        0.6936       0.5200        0.6925  0.0316
     18      0.5200        0.6921       0.5200        0.6925  0.0280
     19      0.5200        0.6923       0.5200        0.6925  0.0262
     20      0.5200        0.6926       0.5200        0.6924  0.0275
     21      0.5200        0.6916       0.5200        0.6924  0.0282
     22      0.5200        0.6919       0.5200        0.6924  0.0285
     23      0.5200        0.6928       0.5200        0.6924  0.0302
     24      0.5200        0.6916       0.5200        0.6923  0.0328
     25      0.5200        0.6931       0.5200        0.6923  0.0341
     26      0.5200        0.6933       0.5200        0.6923  0.0343
     27      0.5200        0.6915       0.5200        0.6922  0.0399
     28      0.5200        0.6932       0.5200        0.6922  0.0412
     29      0.5200        0.6916       0.5200        0.6922  0.0560
     30      0.5200        0.6917       0.5200        0.6921  0.0428
     31      0.5200        0.6932       0.5200        0.6921  0.0375
     32      0.5200        0.6929       0.5200        0.6921  0.0378
     33      0.5200        0.6926       0.5200        0.6921  0.0388
     34      0.5200        0.6927       0.5200        0.6920  0.0384
     35      0.5200        0.6918       0.5200        0.6920  0.0344
     36      0.5200        0.6922       0.5200        0.6919  0.0460
     37      0.5200        0.6931       0.5200        0.6919  0.0366
     38      0.5200        0.6909       0.5200        0.6919  0.0337
     39      0.5200        0.6919       0.5200        0.6918  0.0376
     40      0.5200        0.6921       0.5200        0.6918  0.0362
     41      0.5200        0.6914       0.5200        0.6918  0.0367
     42      0.5200        0.6917       0.5200        0.6917  0.0313
     43      0.5200        0.6908       0.5200        0.6917  0.0290
     44      0.5200        0.6921       0.5200        0.6917  0.0333
     45      0.5200        0.6927       0.5200        0.6916  0.0306
     46      0.5200        0.6907       0.5200        0.6916  0.0321
     47      0.5200        0.6903       0.5200        0.6915  0.0357
     48      0.5200        0.6918       0.5200        0.6915  0.0357
     49      0.5200        0.6926       0.5200        0.6915  0.0374
     50      0.5200        0.6913       0.5200        0.6914  0.0349
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5200        0.6922       0.5200        0.6934  0.0320
      2      0.5200        0.6919       0.5200        0.6933  0.0377
      3      0.5200        0.6927       0.5200        0.6933  0.0354
```

```
      4      0.5200        0.6931       0.5200        0.6932  0.0365
      5      0.5200        0.6944       0.5200        0.6932  0.0322
      6      0.5200        0.6951       0.5200        0.6931  0.0384
      7      0.5200        0.6916       0.5200        0.6931  0.0322
      8      0.5200        0.6958       0.5200        0.6931  0.0342
      9      0.5200        0.6966       0.5200        0.6930  0.0362
     10      0.5200        0.6926       0.5200        0.6930  0.0323
     11      0.5200        0.6930       0.5200        0.6929  0.0344
     12      0.5200        0.6921       0.5200        0.6929  0.0531
     13      0.5200        0.6926       0.5200        0.6929  0.0569
     14      0.5200        0.6923       0.5200        0.6928  0.0520

     15      0.5200        0.6948       0.5200        0.6928  0.0616
     16      0.5200        0.6924       0.5200        0.6928  0.0907
     17      0.5200        0.6916       0.5200        0.6927  0.0902
     18      0.5200        0.6921       0.5200        0.6927  0.0774
     19      0.5200        0.6924       0.5200        0.6927  0.0806
     20      0.5200        0.6936       0.5200        0.6926  0.0689
     21      0.5200        0.6932       0.5200        0.6926  0.0675
     22      0.5200        0.6931       0.5200        0.6925  0.1236
     23      0.5200        0.6939       0.5200        0.6925  0.1677
     24      0.5200        0.6927       0.5200        0.6925  0.0920
     25      0.5200        0.6927       0.5200        0.6925  0.0832
     26      0.5200        0.6933       0.5200        0.6924  0.0660
     27      0.5200        0.6930       0.5200        0.6924  0.0810
     28      0.5200        0.6930       0.5200        0.6924  0.1147
     29      0.5200        0.6934       0.5200        0.6924  0.1497
     30      0.5200        0.6908       0.5200        0.6923  0.0739
     31      0.5200        0.6950       0.5200        0.6923  0.0818
     32      0.5200        0.6920       0.5200        0.6923  0.0824
     33      0.5200        0.6930       0.5200        0.6922  0.1155
     34      0.5200        0.6928       0.5200        0.6922  0.0699
     35      0.5200        0.6952       0.5200        0.6922  0.0904
     36      0.5200        0.6934       0.5200        0.6922  0.0782
     37      0.5200        0.6907       0.5200        0.6921  0.0760
     38      0.5200        0.6928       0.5200        0.6921  0.0484
     39      0.5200        0.6906       0.5200        0.6921  0.0520
     40      0.5200        0.6917       0.5200        0.6920  0.0615
     41      0.5200        0.6937       0.5200        0.6920  0.0404
     42      0.5200        0.6924       0.5200        0.6920  0.0408
     43      0.5200        0.6923       0.5200        0.6919  0.0647
     44      0.5200        0.6925       0.5200        0.6919  0.0564
     45      0.5200        0.6926       0.5200        0.6919  0.0597
     46      0.5200        0.6928       0.5200        0.6918  0.0462
     47      0.5200        0.6930       0.5200        0.6918  0.0392
     48      0.5200        0.6908       0.5200        0.6917  0.0388
     49      0.5200        0.6938       0.5200        0.6917  0.0442
     50      0.5200        0.6923       0.5200        0.6917  0.0504
  epoch    accuracy    train_loss    valid_acc    valid_loss      dur
-------  ----------  ------------  -----------  ------------  ------
```

1	0.8133	0.6709	0.8133	0.5582	0.0507
2	0.8267	0.5266	0.8267	0.4788	0.0527
3	0.7933	0.4781	0.7933	0.5088	0.0488
4	0.8200	0.4851	0.8200	0.4882	0.0566
5	0.8133	0.4683	0.8133	0.4856	0.0580
6	0.8333	0.4676	0.8333	0.4761	0.0521
7	0.8200	0.4743	0.8200	0.4824	0.0515
8	0.8133	0.4934	0.8133	0.4856	0.0498
9	0.8067	0.4705	0.8067	0.4947	0.0516
10	0.8133	0.4602	0.8133	0.4875	0.0565
11	0.8200	0.4817	0.8200	0.4899	0.0487
12	0.8200	0.4642	0.8200	0.4875	0.0530
13	0.8200	0.4552	0.8200	0.4826	0.0407
14	0.8133	0.4652	0.8133	0.4850	0.0401
15	0.8333	0.4752	0.8333	0.4771	0.0397
16	0.8267	0.4746	0.8267	0.4787	0.0387
17	0.8267	0.4574	0.8267	0.4836	0.0401
18	0.8333	0.4759	0.8333	0.4759	0.0388
19	0.8267	0.4580	0.8267	0.4762	0.0390
20	0.8267	0.4503	0.8267	0.4804	0.0407
21	0.8133	0.4543	0.8133	0.4923	0.0401
22	0.8133	0.4784	0.8133	0.4884	0.0471
23	0.8333	0.4581	0.8333	0.4795	0.0505
24	0.8400	0.4666	0.8400	0.4736	0.0538
25	0.8200	0.4602	0.8200	0.4850	0.0433
26	0.8200	0.4678	0.8200	0.4831	0.0403
27	0.8267	0.4956	0.8267	0.4760	0.0359
28	0.8333	0.4791	0.8333	0.4774	0.0386
29	0.8200	0.4752	0.8200	0.4845	0.0383
30	0.8200	0.4482	0.8200	0.4845	0.0404
31	0.8200	0.4549	0.8200	0.4804	0.0400
32	0.8200	0.4488	0.8200	0.4821	0.0363
33	0.8333	0.4505	0.8333	0.4758	0.0419
34	0.8267	0.4552	0.8267	0.4872	0.0375
35	0.8133	0.4554	0.8133	0.4942	0.0394
36	0.8133	0.4493	0.8133	0.4844	0.0482
37	0.8267	0.4459	0.8267	0.4859	0.0454
38	0.8133	0.4664	0.8133	0.4982	0.0424
39	0.8000	0.4709	0.8000	0.4975	0.0392
40	0.8400	0.4694	0.8400	0.4778	0.0473
41	0.8267	0.4437	0.8267	0.4802	0.0429
42	0.8200	0.4632	0.8200	0.4840	0.0410
43	0.8133	0.4486	0.8133	0.4868	0.0405
44	0.8200	0.4540	0.8200	0.4845	0.0412
45	0.8333	0.4460	0.8333	0.4792	0.0404
46	0.8400	0.4537	0.8400	0.4801	0.0474
47	0.8333	0.4514	0.8333	0.4766	0.0493
48	0.8400	0.4577	0.8400	0.4703	0.0452
49	0.8333	0.4532	0.8333	0.4802	0.0432

50	0.8400	0.4544	0.8400	0.4743	0.0485
51	0.8400	0.4787	0.8400	0.4685	0.0475
52	0.8400	0.4673	0.8400	0.4734	0.0410
53	0.8400	0.4499	0.8400	0.4763	0.0382
54	0.8133	0.4548	0.8133	0.4922	0.0380
55	0.8200	0.4681	0.8200	0.4818	0.0391
56	0.8200	0.4539	0.8200	0.4847	0.0387
57	0.8267	0.4688	0.8267	0.4727	0.0391
58	0.8133	0.4532	0.8133	0.4912	0.0426
59	0.8333	0.4666	0.8333	0.4763	0.0407
60	0.8267	0.4529	0.8267	0.4862	0.0373
61	0.8467	0.4463	0.8467	0.4714	0.0399
62	0.8267	0.4536	0.8267	0.4820	0.0498
63	0.8400	0.4572	0.8400	0.4723	0.0457
64	0.8467	0.4618	0.8467	0.4682	0.0434
65	0.8267	0.4487	0.8267	0.4757	0.0421
66	0.8333	0.4502	0.8333	0.4711	0.0438
67	0.8333	0.4469	0.8333	0.4747	0.0415
68	0.8400	0.4499	0.8400	0.4687	0.0503
69	0.8333	0.4579	0.8333	0.4708	0.0459
70	0.8333	0.4388	0.8333	0.4694	0.0436
71	0.8333	0.4450	0.8333	0.4735	0.0415
72	0.8333	0.4545	0.8333	0.4676	0.0404
73	0.8400	0.4409	0.8400	0.4708	0.0348
74	0.8400	0.4457	0.8400	0.4747	0.0380
75	0.8333	0.4574	0.8333	0.4755	0.0378
76	0.8133	0.4427	0.8133	0.4962	0.0449
77	0.8333	0.4522	0.8333	0.4772	0.0454
78	0.8400	0.4442	0.8400	0.4709	0.0514
79	0.8133	0.4433	0.8133	0.4810	0.0594
80	0.8267	0.4498	0.8267	0.4716	0.0655
81	0.8467	0.4528	0.8467	0.4745	0.0597
82	0.8333	0.4398	0.8333	0.4760	0.0630
83	0.8333	0.4520	0.8333	0.4763	0.0534
84	0.8333	0.4390	0.8333	0.4753	0.0554
85	0.8400	0.4397	0.8400	0.4661	0.0496
86	0.8333	0.4472	0.8333	0.4770	0.0431
87	0.8333	0.4626	0.8333	0.4797	0.0434
88	0.8200	0.4477	0.8200	0.4871	0.0464
89	0.8200	0.4458	0.8200	0.4810	0.0600
90	0.8267	0.4810	0.8267	0.4797	0.0508
91	0.8267	0.4432	0.8267	0.4731	0.0526
92	0.8333	0.4574	0.8333	0.4721	0.0470
93	0.8400	0.4563	0.8400	0.4621	0.0481
94	0.8400	0.4468	0.8400	0.4624	0.0468
95	0.8467	0.4638	0.8467	0.4664	0.0456
96	0.8333	0.4686	0.8333	0.4711	0.0434
97	0.8400	0.4537	0.8400	0.4643	0.0486
98	0.8467	0.4508	0.8467	0.4568	0.0544

```
    99      0.8400        0.4512       0.8400        0.4729  0.0597
   100      0.8333        0.4503       0.8333        0.4755  0.0651
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
     1      0.7000        0.6749       0.7000        0.5992  0.0521
     2      0.9133        0.5493       0.9133        0.4099  0.0486
     3      0.9133        0.4979       0.9133        0.4044  0.0519
     4      0.8600        0.4997       0.8600        0.4503  0.0519
     5      0.9133        0.4754       0.9133        0.4015  0.0574
     6      0.9133        0.4631       0.9133        0.4012  0.0679
     7      0.9000        0.4434       0.9000        0.4087  0.0570
     8      0.9133        0.4692       0.9133        0.3978  0.0579
     9      0.9133        0.4518       0.9133        0.3957  0.0615
    10      0.9067        0.4459       0.9067        0.3982  0.0514
    11      0.8933        0.4513       0.8933        0.4192  0.0537
    12      0.8600        0.4579       0.8600        0.4468  0.0538
    13      0.8933        0.4811       0.8933        0.4217  0.0501
    14      0.9200        0.4864       0.9200        0.3982  0.0518
    15      0.8867        0.4530       0.8867        0.4249  0.0516
    16      0.9000        0.4374       0.9000        0.4087  0.0456
    17      0.8733        0.4659       0.8733        0.4368  0.0483
    18      0.9200        0.4636       0.9200        0.4012  0.0568
    19      0.9067        0.4713       0.9067        0.4040  0.0467
    20      0.9000        0.4547       0.9000        0.4049  0.0552
    21      0.9133        0.4560       0.9133        0.3991  0.0457
    22      0.9000        0.4458       0.9000        0.4082  0.0505
    23      0.9133        0.4469       0.9133        0.4000  0.0419
    24      0.8867        0.4365       0.8867        0.4181  0.0595
    25      0.8867        0.4578       0.8867        0.4306  0.0464
    26      0.9133        0.4617       0.9133        0.3996  0.0430
    27      0.8667        0.4642       0.8667        0.4444  0.0430
    28      0.9000        0.4446       0.9000        0.4057  0.0625
    29      0.8800        0.4542       0.8800        0.4333  0.0889
    30      0.9133        0.4507       0.9133        0.4012  0.0644
    31      0.8933        0.4577       0.8933        0.4156  0.0755
    32      0.8867        0.4645       0.8867        0.4245  0.0754
    33      0.9000        0.4414       0.9000        0.4079  0.0635
    34      0.9200        0.4432       0.9200        0.3956  0.0755
    35      0.8867        0.4317       0.8867        0.4263  0.0825
    36      0.9067        0.4497       0.9067        0.4045  0.0594
    37      0.9067        0.4580       0.9067        0.4058  0.0688
    38      0.9000        0.4452       0.9000        0.4073  0.0492
    39      0.8933        0.4551       0.8933        0.4168  0.0634
    40      0.8800        0.4461       0.8800        0.4331  0.0528
    41      0.8867        0.4749       0.8867        0.4234  0.0795
    42      0.8867        0.4636       0.8867        0.4260  0.0550
    43      0.8933        0.4536       0.8933        0.4154  0.0743
    44      0.9000        0.4406       0.9000        0.4068  0.0778
    45      0.8800        0.4378       0.8800        0.4190  0.0731
```

```
     46      0.9067        0.4399       0.9067        0.3991  0.0620
     47      0.8867        0.4452       0.8867        0.4176  0.0664
     48      0.9067        0.4493       0.9067        0.4018  0.0672
     49      0.8867        0.4422       0.8867        0.4190  0.0676
     50      0.9200        0.4596       0.9200        0.3960  0.0655
     51      0.9000        0.4298       0.9000        0.4044  0.0669
     52      0.9267        0.4350       0.9267        0.3949  0.0633
     53      0.9067        0.4313       0.9067        0.4025  0.0632
     54      0.9067        0.4470       0.9067        0.4002  0.0587
     55      0.8933        0.4379       0.8933        0.4161  0.0583
     56      0.9000        0.4401       0.9000        0.4004  0.0549
     57      0.9067        0.4377       0.9067        0.4103  0.0527
     58      0.9133        0.4567       0.9133        0.3976  0.0597
     59      0.9133        0.4330       0.9133        0.4032  0.0463
     60      0.8867        0.4382       0.8867        0.4282  0.0534
     61      0.9067        0.4357       0.9067        0.4037  0.0549
     62      0.9067        0.4313       0.9067        0.4025  0.0589
     63      0.9133        0.4323       0.9133        0.3992  0.0539
     64      0.8933        0.4376       0.8933        0.4178  0.0518
     65      0.8867        0.4416       0.8867        0.4294  0.0566
     66      0.9133        0.4347       0.9133        0.4013  0.0582
     67      0.8933        0.4481       0.8933        0.4119  0.0667
     68      0.9000        0.4365       0.9000        0.4089  0.0521
     69      0.9000        0.4285       0.9000        0.4134  0.0539
     70      0.9133        0.4415       0.9133        0.3993  0.0498
     71      0.8867        0.4433       0.8867        0.4268  0.0582
     72      0.9000        0.4516       0.9000        0.4114  0.0651
     73      0.8867        0.4519       0.8867        0.4171  0.0596
     74      0.8933        0.4306       0.8933        0.4096  0.0699
     75      0.9133        0.4289       0.9133        0.3966  0.0626
     76      0.8933        0.4287       0.8933        0.4171  0.0599
     77      0.8867        0.4336       0.8867        0.4261  0.0612
     78      0.9067        0.4523       0.9067        0.4026  0.0533
     79      0.9067        0.4386       0.9067        0.4056  0.0505
     80      0.8933        0.4360       0.8933        0.4195  0.0538
     81      0.9000        0.4285       0.9000        0.4081  0.0496
     82      0.8933        0.4309       0.8933        0.4176  0.0454
     83      0.8933        0.4475       0.8933        0.4198  0.0425
     84      0.9067        0.4409       0.9067        0.4025  0.0434
     85      0.8733        0.4486       0.8733        0.4339  0.0433

     86      0.8933        0.4395       0.8933        0.4210  0.0367
     87      0.8933        0.4503       0.8933        0.4147  0.0379
     88      0.9067        0.4356       0.9067        0.4022  0.0411
     89      0.9133        0.4535       0.9133        0.4008  0.0381
     90      0.8933        0.4399       0.8933        0.4148  0.0355
     91      0.9067        0.4300       0.9067        0.4067  0.0366
Stopping since valid_loss has not improved in the last 40 epochs.
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
```

1	0.5200	0.6911	0.5200	0.6923	0.0297
2	0.5200	0.6933	0.5200	0.6923	0.0341
3	0.5200	0.6916	0.5200	0.6922	0.0374
4	0.5200	0.6914	0.5200	0.6922	0.0423
5	0.5200	0.6924	0.5200	0.6921	0.0413
6	0.5200	0.6928	0.5200	0.6921	0.0388
7	0.5200	0.6935	0.5200	0.6920	0.0373
8	0.5200	0.6911	0.5200	0.6920	0.0396
9	0.5200	0.6909	0.5200	0.6919	0.0541
10	0.5200	0.6914	0.5200	0.6919	0.0425
11	0.5200	0.6920	0.5200	0.6918	0.0343
12	0.5200	0.6924	0.5200	0.6918	0.0539
13	0.5200	0.6902	0.5200	0.6917	0.0374
14	0.5200	0.6918	0.5200	0.6917	0.0420
15	0.5200	0.6908	0.5200	0.6916	0.0370
16	0.5200	0.6910	0.5200	0.6916	0.0399
17	0.5200	0.6930	0.5200	0.6915	0.0386
18	0.5200	0.6919	0.5200	0.6915	0.0436
19	0.5200	0.6922	0.5200	0.6914	0.0428
20	0.5200	0.6921	0.5200	0.6914	0.0383
21	0.5200	0.6906	0.5200	0.6913	0.0372
22	0.5200	0.6923	0.5200	0.6913	0.0357
23	0.5200	0.6911	0.5200	0.6912	0.0351
24	0.5200	0.6920	0.5200	0.6911	0.0392
25	0.5200	0.6910	0.5200	0.6911	0.0390
26	0.5200	0.6907	0.5200	0.6911	0.0367
27	0.5200	0.6920	0.5200	0.6910	0.0433
28	0.5200	0.6915	0.5200	0.6910	0.0383
29	0.5200	0.6905	0.5200	0.6909	0.0379
30	0.5200	0.6914	0.5200	0.6909	0.0342
31	0.5200	0.6923	0.5200	0.6908	0.0364
32	0.5200	0.6897	0.5200	0.6907	0.0344
33	0.5200	0.6920	0.5200	0.6907	0.0383
34	0.5200	0.6908	0.5200	0.6906	0.0356
35	0.5200	0.6904	0.5200	0.6906	0.0331
36	0.5200	0.6904	0.5200	0.6905	0.0361
37	0.5200	0.6927	0.5200	0.6905	0.0409
38	0.5200	0.6897	0.5200	0.6904	0.0337
39	0.5200	0.6907	0.5200	0.6903	0.0360
40	0.5200	0.6895	0.5200	0.6902	0.0363
41	0.5200	0.6903	0.5200	0.6901	0.0387
42	0.5200	0.6903	0.5200	0.6901	0.0699
43	0.5200	0.6891	0.5200	0.6900	0.0441
44	0.5200	0.6900	0.5200	0.6899	0.0342
45	0.5200	0.6891	0.5200	0.6899	0.0364
46	0.5200	0.6894	0.5200	0.6898	0.0396
47	0.5200	0.6920	0.5200	0.6897	0.0387
48	0.5200	0.6899	0.5200	0.6896	0.0355
49	0.5200	0.6901	0.5200	0.6896	0.0366

50	0.5200	0.6893	0.5200	0.6895	0.0320
51	0.5200	0.6891	0.5200	0.6894	0.0297
52	0.5200	0.6889	0.5200	0.6893	0.0314
53	0.5200	0.6913	0.5200	0.6892	0.0299
54	0.5200	0.6888	0.5200	0.6891	0.0310
55	0.5200	0.6884	0.5200	0.6891	0.0320
56	0.5200	0.6902	0.5200	0.6890	0.0321
57	0.5200	0.6880	0.5200	0.6889	0.0321
58	0.5200	0.6885	0.5200	0.6888	0.0546
59	0.5200	0.6896	0.5200	0.6887	0.0347
60	0.5200	0.6894	0.5200	0.6886	0.0389
61	0.5200	0.6902	0.5200	0.6885	0.0363
62	0.5200	0.6886	0.5200	0.6884	0.0378
63	0.5200	0.6885	0.5200	0.6883	0.0364
64	0.5200	0.6876	0.5200	0.6881	0.0354
65	0.5200	0.6890	0.5200	0.6880	0.0366
66	0.5200	0.6870	0.5200	0.6879	0.0381
67	0.5200	0.6878	0.5200	0.6878	0.0382
68	0.5200	0.6889	0.5200	0.6876	0.0345
69	0.5200	0.6890	0.5200	0.6875	0.0369
70	0.5200	0.6874	0.5200	0.6874	0.0378
71	0.5200	0.6899	0.5200	0.6872	0.0360
72	0.5200	0.6879	0.5200	0.6871	0.0348
73	0.5200	0.6866	0.5200	0.6870	0.0374
74	0.5200	0.6886	0.5200	0.6868	0.0329
75	0.5200	0.6871	0.5200	0.6866	0.0340
76	0.5200	0.6888	0.5200	0.6864	0.0327
77	0.5200	0.6876	0.5200	0.6863	0.0387
78	0.5200	0.6877	0.5200	0.6861	0.0386
79	0.5200	0.6893	0.5200	0.6859	0.0361
80	0.5200	0.6858	0.5200	0.6857	0.0353
81	0.5200	0.6867	0.5200	0.6855	0.0373
82	0.5200	0.6896	0.5200	0.6853	0.0355
83	0.5200	0.6844	0.5200	0.6851	0.0380
84	0.5200	0.6857	0.5200	0.6849	0.0376
85	0.5200	0.6863	0.5200	0.6847	0.0362
86	0.5200	0.6873	0.5200	0.6845	0.0346
87	0.5200	0.6883	0.5200	0.6844	0.0319
88	0.5200	0.6860	0.5200	0.6842	0.0349
89	0.5200	0.6840	0.5200	0.6839	0.0346
90	0.5200	0.6871	0.5200	0.6837	0.0354
91	0.5200	0.6860	0.5200	0.6835	0.0376
92	0.5200	0.6846	0.5200	0.6832	0.0343
93	0.5200	0.6840	0.5200	0.6829	0.0357
94	0.5200	0.6848	0.5200	0.6826	0.0361
95	0.5200	0.6845	0.5200	0.6824	0.0323
96	0.5200	0.6875	0.5200	0.6822	0.0358
97	0.5200	0.6848	0.5200	0.6819	0.0331
98	0.5200	0.6826	0.5200	0.6816	0.0327

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.5200	0.6923	0.5200	0.6938	0.0294
2	0.5200	0.6942	0.5200	0.6937	0.0290
3	0.5200	0.6931	0.5200	0.6936	0.0300
4	0.5200	0.6943	0.5200	0.6936	0.0296
5	0.5200	0.6938	0.5200	0.6935	0.0302
6	0.5200	0.6944	0.5200	0.6934	0.0304
7	0.5200	0.6932	0.5200	0.6933	0.0308
8	0.5200	0.6930	0.5200	0.6933	0.0322
9	0.5200	0.6914	0.5200	0.6932	0.0325
10	0.5200	0.6920	0.5200	0.6932	0.0301
11	0.5200	0.6911	0.5200	0.6931	0.0316
12	0.5200	0.6935	0.5200	0.6931	0.0300
13	0.5200	0.6932	0.5200	0.6930	0.0305
14	0.5200	0.6923	0.5200	0.6929	0.0315
15	0.5200	0.6922	0.5200	0.6929	0.0306
16	0.5200	0.6929	0.5200	0.6928	0.0316
17	0.5200	0.6923	0.5200	0.6928	0.0293
18	0.5200	0.6936	0.5200	0.6927	0.0301
19	0.5200	0.6903	0.5200	0.6927	0.0309
20	0.5200	0.6919	0.5200	0.6926	0.0327
21	0.5200	0.6914	0.5200	0.6926	0.0313
22	0.5200	0.6938	0.5200	0.6925	0.0284
23	0.5200	0.6941	0.5200	0.6925	0.0275
24	0.5200	0.6932	0.5200	0.6924	0.0301
25	0.5200	0.6938	0.5200	0.6924	0.0310
26	0.5200	0.6920	0.5200	0.6924	0.0311
27	0.5200	0.6907	0.5200	0.6923	0.0302
28	0.5200	0.6928	0.5200	0.6923	0.0310
29	0.5200	0.6937	0.5200	0.6922	0.0311
30	0.5200	0.6932	0.5200	0.6922	0.0279
31	0.5200	0.6918	0.5200	0.6921	0.0307
32	0.5200	0.6941	0.5200	0.6921	0.0309
33	0.5200	0.6914	0.5200	0.6920	0.0301
34	0.5200	0.6917	0.5200	0.6920	0.0269
35	0.5200	0.6938	0.5200	0.6919	0.0340
36	0.5200	0.6925	0.5200	0.6919	0.0309
37	0.5200	0.6915	0.5200	0.6918	0.0318
38	0.5200	0.6926	0.5200	0.6918	0.0444
39	0.5200	0.6919	0.5200	0.6918	0.1015
40	0.5200	0.6933	0.5200	0.6917	0.0942
41	0.5200	0.6924	0.5200	0.6917	0.0375
42	0.5200	0.6925	0.5200	0.6916	0.0310
43	0.5200	0.6921	0.5200	0.6916	0.0312
44	0.5200	0.6895	0.5200	0.6915	0.0318
45	0.5200	0.6923	0.5200	0.6915	0.0337

46	0.5200	0.6905	0.5200	0.6914	0.0302
47	0.5200	0.6908	0.5200	0.6914	0.0281
48	0.5200	0.6929	0.5200	0.6913	0.0383
49	0.5200	0.6915	0.5200	0.6913	0.0382
50	0.5200	0.6925	0.5200	0.6912	0.0332
51	0.5200	0.6901	0.5200	0.6912	0.0311
52	0.5200	0.6921	0.5200	0.6911	0.0339
53	0.5200	0.6895	0.5200	0.6910	0.0313
54	0.5200	0.6908	0.5200	0.6910	0.0325
55	0.5200	0.6911	0.5200	0.6910	0.0314
56	0.5200	0.6912	0.5200	0.6909	0.0287
57	0.5200	0.6913	0.5200	0.6909	0.0305
58	0.5200	0.6923	0.5200	0.6908	0.0305
59	0.5200	0.6918	0.5200	0.6908	0.0310
60	0.5200	0.6915	0.5200	0.6908	0.0317
61	0.5200	0.6901	0.5200	0.6907	0.0316
62	0.5200	0.6902	0.5200	0.6907	0.0320
63	0.5200	0.6907	0.5200	0.6906	0.0309
64	0.5200	0.6928	0.5200	0.6905	0.0304
65	0.5200	0.6898	0.5200	0.6905	0.0311
66	0.5200	0.6906	0.5200	0.6904	0.0300
67	0.5200	0.6896	0.5200	0.6903	0.0297
68	0.5200	0.6910	0.5200	0.6903	0.0313
69	0.5200	0.6896	0.5200	0.6902	0.0312
70	0.5200	0.6910	0.5200	0.6902	0.0297
71	0.5200	0.6924	0.5200	0.6901	0.0325
72	0.5200	0.6917	0.5200	0.6900	0.0306
73	0.5200	0.6893	0.5200	0.6900	0.0322
74	0.5200	0.6916	0.5200	0.6899	0.0294
75	0.5200	0.6879	0.5200	0.6898	0.0312
76	0.5200	0.6913	0.5200	0.6897	0.0307
77	0.5200	0.6899	0.5200	0.6896	0.0281
78	0.5200	0.6899	0.5200	0.6896	0.0327
79	0.5200	0.6893	0.5200	0.6895	0.0299
80	0.5200	0.6892	0.5200	0.6894	0.0308
81	0.5200	0.6902	0.5200	0.6893	0.0289
82	0.5200	0.6882	0.5200	0.6892	0.0308
83	0.5200	0.6895	0.5200	0.6891	0.0316
84	0.5200	0.6906	0.5200	0.6890	0.0300
85	0.5200	0.6888	0.5200	0.6889	0.0308
86	0.5200	0.6891	0.5200	0.6889	0.0309
87	0.5200	0.6896	0.5200	0.6887	0.0292
88	0.5200	0.6885	0.5200	0.6886	0.0319
89	0.5200	0.6887	0.5200	0.6886	0.0312
90	0.5200	0.6887	0.5200	0.6884	0.0323
91	0.5200	0.6887	0.5200	0.6883	0.0338
92	0.5200	0.6899	0.5200	0.6882	0.0268
93	0.5200	0.6895	0.5200	0.6881	0.0299
94	0.5200	0.6890	0.5200	0.6880	0.0325

```
     95      0.5200        0.6884       0.5200        0.6878  0.0286
     96      0.5200        0.6890       0.5200        0.6877  0.0322
     97      0.5200        0.6893       0.5200        0.6875  0.0299

     98      0.5200        0.6877       0.5200        0.6874  0.0320
     99      0.5200        0.6893       0.5200        0.6873  0.0302
    100      0.5200        0.6883       0.5200        0.6871  0.0328
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.7667        0.6776       0.7667        0.6102  0.0283
      2      0.8400        0.5599       0.8400        0.4789  0.0252
      3      0.8267        0.4746       0.8267        0.4808  0.0255
      4      0.8133        0.4679       0.8133        0.4833  0.0274
      5      0.8133        0.4678       0.8133        0.4879  0.0278
      6      0.8200        0.4669       0.8200        0.4871  0.0249
      7      0.8333        0.4573       0.8333        0.4788  0.0266
      8      0.8333        0.4457       0.8333        0.4833  0.0242
      9      0.8000        0.4713       0.8000        0.4973  0.0238
     10      0.8067        0.4871       0.8067        0.4905  0.0251
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.7333        0.6909       0.7333        0.6529  0.0248
      2      0.9000        0.6256       0.9000        0.4602  0.0249
      3      0.9133        0.5190       0.9133        0.4091  0.0254
      4      0.8600        0.4698       0.8600        0.4455  0.0259
      5      0.9067        0.4618       0.9067        0.4033  0.0270
      6      0.8200        0.4599       0.8200        0.4845  0.0241
      7      0.9000        0.4618       0.9000        0.4094  0.0247
      8      0.8867        0.4496       0.8867        0.4220  0.0246
      9      0.8733        0.4521       0.8733        0.4378  0.0249
     10      0.8867        0.4573       0.8867        0.4146  0.0258
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5200        0.6928       0.5200        0.6914  0.0207
      2      0.5200        0.6921       0.5200        0.6914  0.0199
      3      0.5200        0.6927       0.5200        0.6914  0.0243
      4      0.5200        0.6931       0.5200        0.6913  0.0226
      5      0.5200        0.6912       0.5200        0.6913  0.0199
      6      0.5200        0.6916       0.5200        0.6913  0.0234
      7      0.5200        0.6917       0.5200        0.6912  0.0228
      8      0.5200        0.6907       0.5200        0.6912  0.0243
      9      0.5200        0.6918       0.5200        0.6911  0.0201
     10      0.5200        0.6925       0.5200        0.6911  0.0217
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.4800        0.6968       0.4800        0.6950  0.0233
      2      0.4800        0.6985       0.4800        0.6949  0.0214
      3      0.4800        0.6959       0.4800        0.6947  0.0223
      4      0.4800        0.6937       0.4800        0.6946  0.0235
      5      0.4800        0.6956       0.4800        0.6944  0.0202
```

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.8067	0.6696	0.8067	0.5716	0.0267
2	0.8133	0.5534	0.8133	0.4881	0.0255
3	0.8600	0.4812	0.8600	0.4662	0.0264
4	0.8333	0.4753	0.8333	0.4749	0.0269
5	0.8267	0.4561	0.8267	0.4776	0.0289
6	0.8200	0.4670	0.8200	0.4898	0.0275
7	0.8200	0.4637	0.8200	0.4885	0.0271
8	0.8333	0.4460	0.8333	0.4819	0.0284
9	0.8267	0.4717	0.8267	0.4726	0.0290
10	0.8400	0.4680	0.8400	0.4791	0.0264
11	0.8333	0.4416	0.8333	0.4698	0.0278
12	0.8333	0.4507	0.8333	0.4785	0.0279
13	0.8267	0.4443	0.8267	0.4696	0.0278
14	0.8200	0.4637	0.8200	0.4778	0.0260
15	0.8200	0.4677	0.8200	0.4777	0.0260
16	0.8333	0.4480	0.8333	0.4764	0.0238
17	0.8200	0.4537	0.8200	0.4930	0.0286
18	0.8333	0.4715	0.8333	0.4805	0.0260
19	0.8267	0.4636	0.8267	0.4736	0.0276
20	0.8267	0.4481	0.8267	0.4755	0.0274
21	0.8333	0.4476	0.8333	0.4735	0.0287
22	0.8267	0.4396	0.8267	0.4759	0.0263
23	0.8333	0.4455	0.8333	0.4840	0.0285
24	0.8333	0.4583	0.8333	0.4724	0.0275
25	0.8333	0.4366	0.8333	0.4783	0.0253
26	0.8333	0.4435	0.8333	0.4763	0.0280
27	0.8467	0.4516	0.8467	0.4727	0.0276
28	0.8333	0.4477	0.8333	0.4749	0.0255
29	0.8333	0.4413	0.8333	0.4813	0.0271
30	0.8400	0.4452	0.8400	0.4718	0.0284
31	0.8333	0.4512	0.8333	0.4805	0.0287
32	0.8333	0.4567	0.8333	0.4788	0.0290
33	0.8200	0.4418	0.8200	0.4682	0.0283
34	0.8333	0.4589	0.8333	0.4783	0.0275
35	0.8400	0.4427	0.8400	0.4669	0.0274
36	0.8467	0.4422	0.8467	0.4648	0.0289
37	0.8467	0.4436	0.8467	0.4628	0.0285
38	0.8267	0.4397	0.8267	0.4778	0.0257
39	0.8267	0.4619	0.8267	0.4678	0.0241
40	0.8000	0.4413	0.8000	0.4982	0.0273
41	0.8400	0.4425	0.8400	0.4663	0.0277
42	0.8333	0.4657	0.8333	0.4702	0.0289

```
    43      0.8333        0.4535       0.8333        0.4756  0.0305
    44      0.8400        0.4498       0.8400        0.4611  0.0293
    45      0.8267        0.4391       0.8267        0.4832  0.0278
    46      0.8333        0.4618       0.8333        0.4756  0.0277
    47      0.8067        0.4449       0.8067        0.4926  0.0305
    48      0.8333        0.4594       0.8333        0.4721  0.0302
    49      0.8267        0.4574       0.8267        0.4757  0.0309
    50      0.8467        0.4529       0.8467        0.4635  0.0305
 epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
     1      0.9067        0.6809       0.9067        0.5907  0.0268
     2      0.9067        0.5909       0.9067        0.4370  0.0252

     3      0.8600        0.5259       0.8600        0.4555  0.0276
     4      0.8933        0.4842       0.8933        0.4183  0.0328
     5      0.8933        0.4492       0.8933        0.4108  0.0290
     6      0.9067        0.4672       0.9067        0.4008  0.0314
     7      0.8867        0.4396       0.8867        0.4228  0.0275
     8      0.8800        0.4546       0.8800        0.4328  0.0283
     9      0.9000        0.4675       0.9000        0.4100  0.0292
    10      0.8600        0.4656       0.8600        0.4465  0.0272
    11      0.9200        0.4580       0.9200        0.3938  0.0258
    12      0.8933        0.4438       0.8933        0.4175  0.0279
    13      0.9200        0.4378       0.9200        0.3942  0.0271
    14      0.8933        0.4412       0.8933        0.4159  0.0269
    15      0.8867        0.4517       0.8867        0.4143  0.0264
    16      0.8933        0.4447       0.8933        0.4117  0.0264
    17      0.8933        0.4459       0.8933        0.4099  0.0282
    18      0.8867        0.4421       0.8867        0.4196  0.0242
    19      0.8867        0.4494       0.8867        0.4218  0.0274
    20      0.8933        0.4405       0.8933        0.4094  0.0303
    21      0.8867        0.4419       0.8867        0.4272  0.0292
    22      0.9200        0.4666       0.9200        0.3954  0.0312
    23      0.9000        0.4417       0.9000        0.4064  0.0294
    24      0.9067        0.4522       0.9067        0.4001  0.0307
    25      0.8600        0.4477       0.8600        0.4458  0.0286
    26      0.9200        0.4541       0.9200        0.3965  0.0299
    27      0.9067        0.4508       0.9067        0.4076  0.0252
    28      0.8933        0.4303       0.8933        0.4034  0.0271
    29      0.9067        0.4539       0.9067        0.4046  0.0275
    30      0.8800        0.4378       0.8800        0.4311  0.0309
    31      0.8867        0.4384       0.8867        0.4269  0.0284
    32      0.9067        0.4396       0.9067        0.4021  0.0297
    33      0.8933        0.4343       0.8933        0.4172  0.0309
    34      0.8933        0.4274       0.8933        0.4135  0.0314
    35      0.8933        0.4448       0.8933        0.4084  0.0312
    36      0.8867        0.4299       0.8867        0.4169  0.0319
    37      0.9000        0.4409       0.9000        0.4133  0.0285
    38      0.9000        0.4670       0.9000        0.4093  0.0294
    39      0.8800        0.4368       0.8800        0.4266  0.0290
```

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.5200	0.6943	0.5200	0.6938	0.0245
2	0.5200	0.6930	0.5200	0.6937	0.0207
3	0.5200	0.6918	0.5200	0.6937	0.0246
4	0.5200	0.6919	0.5200	0.6937	0.0230
5	0.5200	0.6922	0.5200	0.6937	0.0281
6	0.5200	0.6914	0.5200	0.6936	0.0298
7	0.5200	0.6921	0.5200	0.6936	0.0281
8	0.5200	0.6921	0.5200	0.6936	0.0295
9	0.5200	0.6926	0.5200	0.6935	0.0314
10	0.5200	0.6933	0.5200	0.6935	0.0442
11	0.5200	0.6938	0.5200	0.6935	0.0369
12	0.5200	0.6918	0.5200	0.6934	0.0361
13	0.5200	0.6919	0.5200	0.6934	0.0274
14	0.5200	0.6915	0.5200	0.6934	0.0294
15	0.5200	0.6930	0.5200	0.6934	0.1642
16	0.5200	0.6944	0.5200	0.6933	0.0613
17	0.5200	0.6940	0.5200	0.6933	0.0313
18	0.5200	0.6923	0.5200	0.6933	0.0761
19	0.5200	0.6916	0.5200	0.6932	0.0521
20	0.5200	0.6921	0.5200	0.6932	0.0282
21	0.5200	0.6943	0.5200	0.6932	0.0260
22	0.5200	0.6923	0.5200	0.6932	0.0371
23	0.5200	0.6929	0.5200	0.6931	0.0244
24	0.5200	0.6928	0.5200	0.6931	0.0279
25	0.5200	0.6926	0.5200	0.6931	0.0274
26	0.5200	0.6935	0.5200	0.6930	0.0274
27	0.5200	0.6920	0.5200	0.6930	0.0285
28	0.5200	0.6932	0.5200	0.6930	0.0250
29	0.5200	0.6912	0.5200	0.6930	0.0275
30	0.5200	0.6932	0.5200	0.6929	0.0287
31	0.5200	0.6927	0.5200	0.6929	0.0292
32	0.5200	0.6917	0.5200	0.6929	0.0283
33	0.5200	0.6916	0.5200	0.6928	0.0293
34	0.5200	0.6923	0.5200	0.6928	0.0267
35	0.5200	0.6926	0.5200	0.6928	0.0252
36	0.5200	0.6914	0.5200	0.6928	0.0261
37	0.5200	0.6923	0.5200	0.6928	0.0264
38	0.5200	0.6923	0.5200	0.6928	0.0264
39	0.5200	0.6923	0.5200	0.6928	0.0264
40	0.9200	0.4426	0.9200	0.3938	0.0284
41	0.8933	0.4351	0.8933	0.4160	0.0288
42	0.9200	0.4326	0.9200	0.3965	0.0314
43	0.9200	0.4462	0.9200	0.3947	0.0321
44	0.9067	0.4375	0.9067	0.4020	0.0359
45	0.9000	0.4290	0.9000	0.3987	0.0374
46	0.8867	0.4414	0.8867	0.4186	0.0328
47	0.8867	0.4449	0.8867	0.4236	0.0346
48	0.8867	0.4404	0.8867	0.4167	0.0298
49	0.9200	0.4427	0.9200	0.3968	0.0306
50	0.9000	0.4315	0.9000	0.4082	0.0305

```
     37      0.5200        0.6926       0.5200        0.6928  0.0288
     38      0.5200        0.6912       0.5200        0.6927  0.0311
     39      0.5200        0.6921       0.5200        0.6927  0.0273
     40      0.5200        0.6929       0.5200        0.6927  0.0311
     41      0.5200        0.6919       0.5200        0.6926  0.0256
     42      0.5200        0.6904       0.5200        0.6926  0.0308
     43      0.5200        0.6914       0.5200        0.6926  0.0297
     44      0.5200        0.6920       0.5200        0.6926  0.0259
     45      0.5200        0.6934       0.5200        0.6925  0.1377
     46      0.5200        0.6926       0.5200        0.6925  0.0518
     47      0.5200        0.6916       0.5200        0.6925  0.0349
     48      0.5200        0.6923       0.5200        0.6925  0.0360
     49      0.5200        0.6930       0.5200        0.6924  0.0315
     50      0.5200        0.6930       0.5200        0.6924  0.0282
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.4800        0.6995       0.4800        0.6989  0.0286
      2      0.4800        0.6994       0.4800        0.6986  0.0319
      3      0.4800        0.6993       0.4800        0.6983  0.0283
      4      0.4800        0.6981       0.4800        0.6980  0.0381
      5      0.4800        0.6990       0.4800        0.6978  0.0287
      6      0.4800        0.6984       0.4800        0.6975  0.0404

      7      0.4800        0.6962       0.4800        0.6973  0.0347
      8      0.4800        0.6975       0.4800        0.6970  0.0402
      9      0.4800        0.6974       0.4800        0.6968  0.0354
     10      0.4800        0.6961       0.4800        0.6965  0.0602
     11      0.4800        0.6971       0.4800        0.6963  0.0839
     12      0.4800        0.6970       0.4800        0.6961  0.0485
     13      0.4800        0.6946       0.4800        0.6959  0.0360
     14      0.4800        0.6950       0.4800        0.6956  0.0366
     15      0.4800        0.6964       0.4800        0.6954  0.0641
     16      0.4800        0.6955       0.4800        0.6952  0.0479
     17      0.4800        0.6954       0.4800        0.6951  0.0359
     18      0.4800        0.6956       0.4800        0.6949  0.0370
     19      0.4800        0.6959       0.4800        0.6947  0.0522
     20      0.4800        0.6956       0.4800        0.6946  0.0266
     21      0.4800        0.6945       0.4800        0.6944  0.0236
     22      0.4800        0.6954       0.4800        0.6942  0.0292
     23      0.4800        0.6947       0.4800        0.6941  0.0249
     24      0.4800        0.6942       0.4800        0.6939  0.0245
     25      0.4800        0.6948       0.4800        0.6937  0.0253
     26      0.4800        0.6931       0.4800        0.6935  0.0200
     27      0.4800        0.6945       0.4800        0.6934  0.0220
     28      0.4800        0.6927       0.4800        0.6932  0.0204
     29      0.4800        0.6935       0.4800        0.6931  0.0279
     30      0.4800        0.6941       0.4800        0.6929  0.0255
     31      0.4800        0.6928       0.4800        0.6928  0.0216
     32      0.4800        0.6931       0.4800        0.6926  0.0209
     33      0.4800        0.6953       0.4800        0.6925  0.0271
```

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
34	0.4800	0.6927	0.4800	0.6924	0.0234
35	0.4800	0.6920	0.4800	0.6922	0.0259
36	0.4800	0.6925	0.4800	0.6921	0.0268
37	0.4800	0.6923	0.4800	0.6920	0.0219
38	0.4800	0.6938	0.4800	0.6919	0.0304
39	0.4800	0.6906	0.4800	0.6917	0.0288
40	0.4800	0.6913	0.4800	0.6916	0.0225
41	0.4800	0.6953	0.4800	0.6915	0.0229
42	0.4800	0.6923	0.4800	0.6914	0.0224
43	0.4800	0.6933	0.4800	0.6913	0.0259
44	0.4800	0.6934	0.4800	0.6912	0.0284
45	0.4800	0.6918	0.4800	0.6910	0.0266
46	0.4800	0.6932	0.4800	0.6910	0.0270
47	0.4800	0.6930	0.4800	0.6909	0.0225
48	0.4800	0.6909	0.4800	0.6907	0.0251
49	0.4933	0.6923	0.4933	0.6906	0.0237
50	0.5000	0.6908	0.5000	0.6905	0.0277
epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.5933	0.6865	0.5933	0.6539	0.0306
2	0.8000	0.6314	0.8000	0.5305	0.0248
3	0.8267	0.5260	0.8267	0.4873	0.0245
4	0.8000	0.4886	0.8000	0.4996	0.0214
5	0.8067	0.4783	0.8067	0.4934	0.0205
6	0.8400	0.4721	0.8400	0.4751	0.0206
7	0.8133	0.4769	0.8133	0.4921	0.0200
8	0.8133	0.4617	0.8133	0.4907	0.0225
9	0.8133	0.4562	0.8133	0.4895	0.0235
10	0.8067	0.4673	0.8067	0.4898	0.0249
11	0.8133	0.4662	0.8133	0.4921	0.0227
12	0.8267	0.4733	0.8267	0.4878	0.0223
13	0.8333	0.4707	0.8333	0.4778	0.0203
14	0.8067	0.4649	0.8067	0.4958	0.0187
15	0.8200	0.4587	0.8200	0.4781	0.0186
16	0.8200	0.4543	0.8200	0.4797	0.0189
17	0.8200	0.4539	0.8200	0.4870	0.0230
18	0.8133	0.4465	0.8133	0.4872	0.0370
19	0.8333	0.4484	0.8333	0.4804	0.0356
20	0.8067	0.4636	0.8067	0.4907	0.0222
21	0.8267	0.4684	0.8267	0.4745	0.0212
22	0.8133	0.4430	0.8133	0.4859	0.0211
23	0.8267	0.4519	0.8267	0.4789	0.0199
24	0.8267	0.4628	0.8267	0.4806	0.0197
25	0.8333	0.4490	0.8333	0.4766	0.0214
26	0.8333	0.4797	0.8333	0.4747	0.0262
27	0.8200	0.4539	0.8200	0.4811	0.0224
28	0.8200	0.4638	0.8200	0.4851	0.0229
29	0.8200	0.4537	0.8200	0.4931	0.0220
30	0.8333	0.4696	0.8333	0.4771	0.0205

31	0.8467	0.4575	0.8467	0.4708	0.0202
32	0.8267	0.4489	0.8267	0.4764	0.0206
33	0.8467	0.4536	0.8467	0.4721	0.0195
34	0.8200	0.4654	0.8200	0.4916	0.0227
35	0.8333	0.4826	0.8333	0.4804	0.0204
36	0.8200	0.4529	0.8200	0.4801	0.0252
37	0.8333	0.4532	0.8333	0.4780	0.0253
38	0.8200	0.4710	0.8200	0.4906	0.0237
39	0.8333	0.4638	0.8333	0.4789	0.0214
40	0.8267	0.4571	0.8267	0.4780	0.0201
41	0.8200	0.4492	0.8200	0.4818	0.0189
42	0.8333	0.4561	0.8333	0.4752	0.0204
43	0.8200	0.4510	0.8200	0.4819	0.0198
44	0.8333	0.4412	0.8333	0.4778	0.0231
45	0.8333	0.4416	0.8333	0.4770	0.0235
46	0.8333	0.4448	0.8333	0.4710	0.0236
47	0.8333	0.4404	0.8333	0.4759	0.0217
48	0.8400	0.4440	0.8400	0.4682	0.0199
49	0.8067	0.4483	0.8067	0.5045	0.0202
50	0.8400	0.4683	0.8400	0.4754	0.0198
51	0.8333	0.4592	0.8333	0.4688	0.0182
52	0.8400	0.4420	0.8400	0.4744	0.0220
53	0.8267	0.4459	0.8267	0.4751	0.0238
54	0.8333	0.4490	0.8333	0.4728	0.0217
55	0.8267	0.4380	0.8267	0.4737	0.0211
56	0.8400	0.4468	0.8400	0.4693	0.0213
57	0.8200	0.4429	0.8200	0.4847	0.0188
58	0.8333	0.4487	0.8333	0.4771	0.0197
59	0.8200	0.4405	0.8200	0.4813	0.0180
60	0.8333	0.4403	0.8333	0.4763	0.0184
61	0.8400	0.4359	0.8400	0.4781	0.0218
62	0.8467	0.4445	0.8467	0.4682	0.0222
63	0.8267	0.4428	0.8267	0.4883	0.0216
64	0.8200	0.4484	0.8200	0.4948	0.0223
65	0.8333	0.4509	0.8333	0.4799	0.0206
66	0.8200	0.4580	0.8200	0.4854	0.0201
67	0.8267	0.4397	0.8267	0.4692	0.0199
68	0.8333	0.4513	0.8333	0.4792	0.0188
69	0.8333	0.4475	0.8333	0.4751	0.0202
70	0.8333	0.4602	0.8333	0.4748	0.0227
71	0.8333	0.4505	0.8333	0.4695	0.0252
72	0.8400	0.4627	0.8400	0.4696	0.0232
73	0.8400	0.4599	0.8400	0.4744	0.0246
74	0.8400	0.4525	0.8400	0.4734	0.0225
75	0.8267	0.4497	0.8267	0.4841	0.0203
76	0.8333	0.4805	0.8333	0.4709	0.0200
77	0.8400	0.4589	0.8400	0.4671	0.0230
78	0.8400	0.4547	0.8400	0.4751	0.0253
79	0.8400	0.4455	0.8400	0.4695	0.0253

80	0.8400	0.4370	0.8400	0.4744	0.0273
81	0.8467	0.4514	0.8467	0.4626	0.0262
82	0.8133	0.4458	0.8133	0.4904	0.0261
83	0.8267	0.4588	0.8267	0.4768	0.0237
84	0.8333	0.4431	0.8333	0.4761	0.0211
85	0.8400	0.4499	0.8400	0.4717	0.0199
86	0.8333	0.4463	0.8333	0.4726	0.0204
87	0.8400	0.4553	0.8400	0.4769	0.0212
88	0.8267	0.4583	0.8267	0.4839	0.0217
89	0.8400	0.4445	0.8400	0.4735	0.0248
90	0.8333	0.4406	0.8333	0.4726	0.0234
91	0.8333	0.4433	0.8333	0.4752	0.0238
92	0.8533	0.4500	0.8533	0.4659	0.0298
93	0.8400	0.4383	0.8400	0.4736	0.0266
94	0.8533	0.4427	0.8533	0.4663	0.0252
95	0.8400	0.4457	0.8400	0.4736	0.0229
96	0.8467	0.4598	0.8467	0.4636	0.0288
97	0.8400	0.4451	0.8400	0.4661	0.0332
98	0.8400	0.4390	0.8400	0.4736	0.0336
99	0.8267	0.4542	0.8267	0.4717	0.0338
100	0.8467	0.4574	0.8467	0.4626	0.0330

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.5867	0.6889	0.5867	0.6586	0.0370
2	0.8867	0.6310	0.8867	0.4815	0.0325
3	0.8600	0.5244	0.8600	0.4476	0.0288
4	0.9133	0.4761	0.9133	0.4000	0.0362
5	0.8933	0.4579	0.8933	0.4189	0.0310
6	0.8800	0.4597	0.8800	0.4322	0.0307
7	0.9000	0.4690	0.9000	0.4086	0.0333
8	0.9000	0.4443	0.9000	0.4055	0.0399
9	0.8867	0.4576	0.8867	0.4212	0.0481
10	0.8933	0.4567	0.8933	0.4097	0.0379
11	0.9067	0.4496	0.9067	0.4078	0.0345
12	0.9200	0.4391	0.9200	0.3994	0.0358
13	0.9000	0.4357	0.9000	0.4048	0.0328
14	0.9133	0.4602	0.9133	0.4026	0.0395
15	0.9133	0.4441	0.9133	0.4040	0.0317
16	0.8933	0.4349	0.8933	0.4135	0.0331
17	0.8867	0.4493	0.8867	0.4276	0.0340
18	0.8867	0.4328	0.8867	0.4275	0.0355
19	0.9133	0.4486	0.9133	0.4022	0.0333
20	0.8933	0.4481	0.8933	0.4092	0.0317
21	0.9133	0.4390	0.9133	0.4002	0.0332
22	0.9067	0.4518	0.9067	0.4048	0.0320
23	0.9200	0.4522	0.9200	0.3951	0.0343
24	0.8933	0.4441	0.8933	0.4129	0.0320
25	0.8800	0.4514	0.8800	0.4213	0.0307
26	0.9067	0.4447	0.9067	0.4034	0.0325

27	0.9067	0.4414	0.9067	0.4012	0.0323
28	0.8933	0.4849	0.8933	0.4097	0.0354
29	0.9000	0.4578	0.9000	0.4070	0.0301
30	0.8800	0.4502	0.8800	0.4389	0.0303
31	0.9067	0.4412	0.9067	0.4050	0.0350
32	0.8933	0.4383	0.8933	0.4159	0.0329
33	0.9067	0.4498	0.9067	0.4003	0.0317
34	0.9133	0.4330	0.9133	0.3953	0.0395
35	0.9000	0.4302	0.9000	0.4127	0.0301
36	0.9000	0.4454	0.9000	0.4053	0.0290
37	0.8867	0.4466	0.8867	0.4286	0.0289
38	0.9000	0.4413	0.9000	0.4045	0.0323
39	0.8933	0.4579	0.8933	0.4135	0.0329
40	0.9133	0.4494	0.9133	0.3972	0.0310
41	0.8867	0.4451	0.8867	0.4266	0.0299
42	0.8933	0.4453	0.8933	0.4150	0.0282
43	0.9067	0.4378	0.9067	0.4053	0.0281
44	0.9067	0.4511	0.9067	0.4047	0.0263
45	0.9000	0.4668	0.9000	0.4127	0.0284
46	0.8867	0.4671	0.8867	0.4297	0.0260
47	0.9133	0.4405	0.9133	0.3968	0.0260
48	0.9133	0.4440	0.9133	0.4014	0.0258
49	0.9067	0.4346	0.9067	0.4058	0.0254
50	0.9000	0.4537	0.9000	0.4071	0.0222
51	0.8867	0.4404	0.8867	0.4255	0.0230
52	0.9133	0.4402	0.9133	0.3950	0.0206
53	0.9133	0.4407	0.9133	0.4034	0.0253
54	0.9067	0.4294	0.9067	0.3977	0.0234
55	0.9000	0.4343	0.9000	0.3991	0.0248
56	0.8867	0.4351	0.8867	0.4178	0.0244
57	0.9000	0.4614	0.9000	0.4081	0.0257
58	0.8867	0.4385	0.8867	0.4221	0.0260
59	0.9067	0.4478	0.9067	0.4079	0.0261
60	0.8867	0.4421	0.8867	0.4273	0.0270
61	0.9067	0.4472	0.9067	0.4081	0.0271
62	0.8933	0.4413	0.8933	0.4188	0.0292
63	0.8933	0.4621	0.8933	0.4175	0.0304
64	0.8933	0.4287	0.8933	0.4207	0.0277
65	0.9133	0.4452	0.9133	0.3949	0.0259
66	0.9200	0.4336	0.9200	0.3919	0.0241
67	0.9133	0.4400	0.9133	0.3951	0.0237
68	0.8867	0.4350	0.8867	0.4197	0.0243
69	0.9067	0.4359	0.9067	0.4029	0.0267
70	0.9200	0.4476	0.9200	0.3962	0.0246
71	0.8933	0.4428	0.8933	0.4124	0.0237
72	0.8867	0.4451	0.8867	0.4231	0.0245
73	0.9133	0.4517	0.9133	0.3961	0.0240
74	0.9067	0.4441	0.9067	0.3995	0.0240

75	0.9200	0.4392	0.9200	0.3937	0.0206
76	0.9200	0.4392	0.9200	0.3958	0.0220
77	0.9067	0.4436	0.9067	0.4018	0.0203
78	0.9067	0.4323	0.9067	0.4019	0.0221
79	0.9133	0.4348	0.9133	0.3961	0.0235
80	0.9000	0.4283	0.9000	0.4024	0.0245
81	0.9000	0.4265	0.9000	0.4085	0.0235
82	0.8867	0.4361	0.8867	0.4268	0.0226
83	0.8933	0.4362	0.8933	0.4167	0.0230
84	0.8600	0.4508	0.8600	0.4518	0.0243
85	0.9200	0.4480	0.9200	0.3964	0.0261
86	0.9000	0.4625	0.9000	0.4128	0.0247
87	0.8933	0.4498	0.8933	0.4082	0.0228
88	0.9000	0.4565	0.9000	0.4057	0.0213
89	0.9000	0.4378	0.9000	0.4056	0.0244
90	0.8933	0.4349	0.8933	0.4202	0.0227
91	0.8867	0.4395	0.8867	0.4178	0.0229
92	0.9067	0.4330	0.9067	0.4053	0.0221
93	0.9067	0.4372	0.9067	0.4050	0.0200
94	0.9067	0.4285	0.9067	0.4014	0.0206
95	0.9133	0.4276	0.9133	0.3973	0.0222
96	0.9133	0.4254	0.9133	0.4004	0.0214
97	0.9000	0.4308	0.9000	0.4028	0.0225
98	0.9000	0.4335	0.9000	0.4038	0.0226
99	0.8933	0.4245	0.8933	0.4092	0.0224
100	0.9000	0.4482	0.9000	0.4157	0.0211

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.4800	0.6975	0.4800	0.6975	0.0198
2	0.4800	0.6963	0.4800	0.6973	0.0180
3	0.4800	0.6954	0.4800	0.6972	0.0200
4	0.4800	0.6968	0.4800	0.6971	0.0183
5	0.4800	0.6951	0.4800	0.6969	0.0209
6	0.4800	0.6960	0.4800	0.6968	0.0188
7	0.4800	0.6973	0.4800	0.6967	0.0181
8	0.4800	0.6969	0.4800	0.6966	0.0181
9	0.4800	0.6959	0.4800	0.6964	0.0178
10	0.4800	0.6940	0.4800	0.6963	0.0183
11	0.4800	0.6959	0.4800	0.6962	0.0201
12	0.4800	0.6958	0.4800	0.6961	0.0175
13	0.4800	0.6956	0.4800	0.6960	0.0176
14	0.4800	0.6948	0.4800	0.6959	0.0179
15	0.4800	0.6948	0.4800	0.6958	0.0166
16	0.4800	0.6942	0.4800	0.6957	0.0161
17	0.4800	0.6945	0.4800	0.6956	0.0185
18	0.4800	0.6963	0.4800	0.6955	0.0179
19	0.4800	0.6957	0.4800	0.6954	0.0202
20	0.4800	0.6954	0.4800	0.6953	0.0195
21	0.4800	0.6924	0.4800	0.6952	0.0199
22	0.4800	0.6948	0.4800	0.6951	0.0204

23	0.4800	0.6927	0.4800	0.6950	0.0200
24	0.4800	0.6943	0.4800	0.6949	0.0186
25	0.4800	0.6953	0.4800	0.6948	0.0191
26	0.4800	0.6951	0.4800	0.6947	0.0181
27	0.4800	0.6927	0.4800	0.6947	0.0202
28	0.4800	0.6931	0.4800	0.6946	0.0201
29	0.4800	0.6940	0.4800	0.6945	0.0197
30	0.4800	0.6928	0.4800	0.6944	0.0165
31	0.4800	0.6934	0.4800	0.6944	0.0162
32	0.4800	0.6937	0.4800	0.6943	0.0184
33	0.4800	0.6928	0.4800	0.6942	0.0164
34	0.4800	0.6941	0.4800	0.6942	0.0171
35	0.4800	0.6939	0.4800	0.6941	0.0175
36	0.4800	0.6929	0.4800	0.6940	0.0169
37	0.4800	0.6942	0.4800	0.6939	0.0168
38	0.4800	0.6932	0.4800	0.6939	0.0161
39	0.4800	0.6963	0.4800	0.6938	0.0158
40	0.4800	0.6943	0.4800	0.6938	0.0171
41	0.4800	0.6924	0.4800	0.6937	0.0161
42	0.4800	0.6942	0.4800	0.6937	0.0162
43	0.4800	0.6932	0.4800	0.6936	0.0179
44	0.4733	0.6931	0.4733	0.6935	0.0167
45	0.4667	0.6922	0.4667	0.6935	0.0168
46	0.4667	0.6935	0.4667	0.6934	0.0174
47	0.4667	0.6922	0.4667	0.6934	0.0163
48	0.4800	0.6941	0.4800	0.6933	0.0161
49	0.4867	0.6913	0.4867	0.6933	0.0168
50	0.4933	0.6928	0.4933	0.6932	0.0179
51	0.5000	0.6906	0.5000	0.6932	0.0163
52	0.5200	0.6924	0.5200	0.6931	0.0167
53	0.5400	0.6944	0.5400	0.6930	0.0161
54	0.5267	0.6930	0.5267	0.6930	0.0170
55	0.5133	0.6923	0.5133	0.6929	0.0167
56	0.5267	0.6931	0.5267	0.6929	0.0170
57	0.5267	0.6951	0.5267	0.6928	0.0160
58	0.5133	0.6941	0.5133	0.6928	0.0166
59	0.5200	0.6915	0.5200	0.6928	0.0167
60	0.5200	0.6926	0.5200	0.6927	0.0159
61	0.5133	0.6920	0.5133	0.6927	0.0165
62	0.5200	0.6929	0.5200	0.6926	0.0175
63	0.5200	0.6914	0.5200	0.6926	0.0163
64	0.5200	0.6924	0.5200	0.6925	0.0142
65	0.5200	0.6927	0.5200	0.6925	0.0147
66	0.5200	0.6918	0.5200	0.6924	0.0138
67	0.5200	0.6933	0.5200	0.6924	0.0144
68	0.5200	0.6906	0.5200	0.6923	0.0145
69	0.5200	0.6912	0.5200	0.6923	0.0134
70	0.5200	0.6904	0.5200	0.6922	0.0138
71	0.5200	0.6915	0.5200	0.6922	0.0144
72	0.5200	0.6900	0.5200	0.6921	0.0139

73	0.5200	0.6946	0.5200	0.6921	0.0148
74	0.5200	0.6914	0.5200	0.6920	0.0184
75	0.5200	0.6921	0.5200	0.6920	0.0206
76	0.5200	0.6906	0.5200	0.6919	0.0209
77	0.5200	0.6893	0.5200	0.6919	0.0194
78	0.5200	0.6896	0.5200	0.6918	0.0176
79	0.5200	0.6905	0.5200	0.6918	0.0253
80	0.5200	0.6911	0.5200	0.6917	0.0224
81	0.5200	0.6925	0.5200	0.6917	0.0202
82	0.5200	0.6929	0.5200	0.6917	0.0204
83	0.5200	0.6897	0.5200	0.6916	0.0198
84	0.5200	0.6922	0.5200	0.6916	0.0236
85	0.5200	0.6912	0.5200	0.6915	0.0927
86	0.5200	0.6918	0.5200	0.6915	0.0300
87	0.5200	0.6906	0.5200	0.6914	0.0355
88	0.5200	0.6908	0.5200	0.6914	0.0267
89	0.5200	0.6914	0.5200	0.6913	0.0263
90	0.5200	0.6926	0.5200	0.6913	0.0226
91	0.5200	0.6924	0.5200	0.6912	0.0250
92	0.5200	0.6928	0.5200	0.6912	0.0226
93	0.5200	0.6912	0.5200	0.6911	0.0224
94	0.5200	0.6905	0.5200	0.6911	0.0225
95	0.5200	0.6899	0.5200	0.6910	0.0219
96	0.5200	0.6913	0.5200	0.6909	0.0211
97	0.5200	0.6916	0.5200	0.6909	0.0218
98	0.5200	0.6924	0.5200	0.6909	0.0194
99	0.5200	0.6897	0.5200	0.6908	0.0176
100	0.5200	0.6925	0.5200	0.6908	0.0183

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.5200	0.6923	0.5200	0.6933	0.0178
2	0.5200	0.6921	0.5200	0.6933	0.0183
3	0.5200	0.6951	0.5200	0.6932	0.0170
4	0.5200	0.6932	0.5200	0.6931	0.0173
5	0.5200	0.6955	0.5200	0.6931	0.0148
6	0.5200	0.6951	0.5200	0.6930	0.0146
7	0.5200	0.6954	0.5200	0.6929	0.0142
8	0.5200	0.6932	0.5200	0.6929	0.0141
9	0.5200	0.6948	0.5200	0.6928	0.0169
10	0.5200	0.6908	0.5200	0.6927	0.0177
11	0.5200	0.6936	0.5200	0.6927	0.0173
12	0.5200	0.6947	0.5200	0.6926	0.0170
13	0.5200	0.6954	0.5200	0.6925	0.0159
14	0.5200	0.6918	0.5200	0.6924	0.0169
15	0.5200	0.6910	0.5200	0.6924	0.0170
16	0.5200	0.6938	0.5200	0.6923	0.0185
17	0.5200	0.6954	0.5200	0.6922	0.0179
18	0.5200	0.6928	0.5200	0.6921	0.0158
19	0.5200	0.6965	0.5200	0.6921	0.0179

20	0.5200	0.6944	0.5200	0.6920	0.0168
21	0.5200	0.6946	0.5200	0.6920	0.0153
22	0.5200	0.6934	0.5200	0.6919	0.0168
23	0.5200	0.6929	0.5200	0.6918	0.0144
24	0.5200	0.6923	0.5200	0.6918	0.0171
25	0.5200	0.6932	0.5200	0.6917	0.0145
26	0.5200	0.6939	0.5200	0.6916	0.0154
27	0.5200	0.6934	0.5200	0.6916	0.0147
28	0.5200	0.6928	0.5200	0.6915	0.0174
29	0.5200	0.6939	0.5200	0.6915	0.0170
30	0.5200	0.6933	0.5200	0.6914	0.0171
31	0.5200	0.6927	0.5200	0.6913	0.0177
32	0.5200	0.6917	0.5200	0.6913	0.0153
33	0.5200	0.6935	0.5200	0.6912	0.0158
34	0.5200	0.6906	0.5200	0.6912	0.0150
35	0.5200	0.6923	0.5200	0.6911	0.0148
36	0.5200	0.6918	0.5200	0.6910	0.0154
37	0.5200	0.6936	0.5200	0.6910	0.0166
38	0.5200	0.6934	0.5200	0.6909	0.0141
39	0.5200	0.6923	0.5200	0.6908	0.0163
40	0.5200	0.6901	0.5200	0.6908	0.0156
41	0.5200	0.6911	0.5200	0.6907	0.0144
42	0.5200	0.6905	0.5200	0.6906	0.0181
43	0.5200	0.6930	0.5200	0.6906	0.0213
44	0.5200	0.6901	0.5200	0.6905	0.0180
45	0.5200	0.6922	0.5200	0.6905	0.0176
46	0.5200	0.6928	0.5200	0.6904	0.0155
47	0.5200	0.6935	0.5200	0.6903	0.0151
48	0.5200	0.6910	0.5200	0.6903	0.0175
49	0.5200	0.6903	0.5200	0.6902	0.0170
50	0.5200	0.6899	0.5200	0.6902	0.0191
51	0.5200	0.6936	0.5200	0.6901	0.0154
52	0.5200	0.6930	0.5200	0.6900	0.0179
53	0.5200	0.6892	0.5200	0.6900	0.0153
54	0.5200	0.6916	0.5200	0.6899	0.0169
55	0.5200	0.6919	0.5200	0.6899	0.0173
56	0.5200	0.6934	0.5200	0.6898	0.0161
57	0.5200	0.6916	0.5200	0.6897	0.0190
58	0.5200	0.6910	0.5200	0.6897	0.0182
59	0.5200	0.6926	0.5200	0.6896	0.0155
60	0.5200	0.6906	0.5200	0.6896	0.0161
61	0.5200	0.6916	0.5200	0.6895	0.0182
62	0.5200	0.6910	0.5200	0.6894	0.0187
63	0.5200	0.6909	0.5200	0.6894	0.0177
64	0.5200	0.6914	0.5200	0.6893	0.0187
65	0.5200	0.6897	0.5200	0.6893	0.0187
66	0.5200	0.6887	0.5200	0.6892	0.0171
67	0.5200	0.6902	0.5200	0.6891	0.0176
68	0.5200	0.6904	0.5200	0.6890	0.0176

```
     69      0.5200        0.6899       0.5200        0.6890  0.0169
     70      0.5200        0.6906       0.5200        0.6889  0.0273
     71      0.5200        0.6919       0.5200        0.6889  0.0216
     72      0.5200        0.6918       0.5200        0.6888  0.0207
     73      0.5200        0.6920       0.5200        0.6888  0.0219
     74      0.5200        0.6904       0.5200        0.6887  0.0236
     75      0.5200        0.6917       0.5200        0.6886  0.0188
     76      0.5200        0.6936       0.5200        0.6886  0.0211
     77      0.5200        0.6900       0.5200        0.6885  0.0242
     78      0.5200        0.6896       0.5200        0.6885  0.0371
     79      0.5200        0.6886       0.5200        0.6884  0.0244

     80      0.5200        0.6902       0.5200        0.6883  0.0224
     81      0.5200        0.6900       0.5200        0.6882  0.0213
     82      0.5200        0.6933       0.5200        0.6882  0.0228
     83      0.5200        0.6900       0.5200        0.6881  0.0223
     84      0.5200        0.6905       0.5200        0.6881  0.0186
     85      0.5200        0.6895       0.5200        0.6880  0.0192
     86      0.5200        0.6903       0.5200        0.6879  0.0198
     87      0.5200        0.6907       0.5200        0.6878  0.0217
     88      0.5200        0.6887       0.5200        0.6878  0.0201
     89      0.5200        0.6902       0.5200        0.6877  0.0207
     90      0.5200        0.6927       0.5200        0.6877  0.0200
     91      0.5200        0.6909       0.5200        0.6876  0.0197
     92      0.5200        0.6897       0.5200        0.6875  0.0200
     93      0.5200        0.6893       0.5200        0.6875  0.0184
     94      0.5200        0.6929       0.5200        0.6874  0.0191
     95      0.5200        0.6897       0.5200        0.6873  0.0188
     96      0.5200        0.6917       0.5200        0.6873  0.0199
     97      0.5200        0.6894       0.5200        0.6872  0.0187
     98      0.5200        0.6880       0.5200        0.6871  0.0189
     99      0.5200        0.6902       0.5200        0.6871  0.0367
    100      0.5200        0.6903       0.5200        0.6870  0.0620
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.7933        0.6859       0.7933        0.6439  0.1160
      2      0.8200        0.6063       0.8200        0.4916  0.0388
      3      0.8200        0.5104       0.8200        0.4782  0.0464
      4      0.8200        0.4803       0.8200        0.4826  0.0435
      5      0.8333        0.4666       0.8333        0.4784  0.0366
      6      0.8200        0.4672       0.8200        0.4802  0.0285
      7      0.8200        0.4653       0.8200        0.4834  0.0293
      8      0.8467        0.4645       0.8467        0.4653  0.0331
      9      0.8400        0.4673       0.8400        0.4761  0.0304
     10      0.8267        0.4786       0.8267        0.4863  0.0300
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5933        0.6853       0.5933        0.6383  0.0310
      2      0.9000        0.6023       0.9000        0.4450  0.0322
      3      0.9133        0.4781       0.9133        0.3993  0.0314
```

```
      4      0.9067        0.4574       0.9067        0.4043  0.0382
      5      0.9067        0.4522       0.9067        0.4043  0.0328
      6      0.8533        0.4462       0.8533        0.4563  0.0300
      7      0.9067        0.4496       0.9067        0.4017  0.0333
      8      0.8867        0.4587       0.8867        0.4274  0.0361
      9      0.8933        0.4438       0.8933        0.4143  0.0325
     10      0.8867        0.4668       0.8867        0.4283  0.0353
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5200        0.6925       0.5200        0.6930  0.0223
      2      0.5200        0.6941       0.5200        0.6929  0.0196
      3      0.5200        0.6917       0.5200        0.6929  0.1375
      4      0.5200        0.6916       0.5200        0.6928  0.0644
      5      0.5200        0.6917       0.5200        0.6928  0.0325
      6      0.5200        0.6922       0.5200        0.6928  0.0509
      7      0.5200        0.6932       0.5200        0.6927  0.0237
      8      0.5200        0.6926       0.5200        0.6927  0.0292
      9      0.5200        0.6926       0.5200        0.6926  0.0260
     10      0.5200        0.6917       0.5200        0.6926  0.0247
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.4800        0.6965       0.4800        0.6953  0.0248
      2      0.4800        0.6974       0.4800        0.6952  0.0271
      3      0.4800        0.6971       0.4800        0.6950  0.0364
      4      0.4800        0.6963       0.4800        0.6948  0.0294
      5      0.4800        0.6962       0.4800        0.6947  0.0477
      6      0.4800        0.6961       0.4800        0.6945  0.0351
      7      0.4800        0.6966       0.4800        0.6944  0.0293
      8      0.4800        0.6954       0.4800        0.6942  0.0443
      9      0.4800        0.6958       0.4800        0.6940  0.0330
     10      0.4800        0.6949       0.4800        0.6939  0.0280
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.7733        0.6859       0.7733        0.6321  0.0349
      2      0.8133        0.5868       0.8133        0.4938  0.0349
      3      0.8067        0.5045       0.8067        0.4926  0.0319
      4      0.8267        0.4714       0.8267        0.4788  0.0339
      5      0.8400        0.4557       0.8400        0.4774  0.0334
      6      0.8067        0.4748       0.8067        0.4937  0.0353
      7      0.8333        0.4757       0.8333        0.4808  0.0345
      8      0.8200        0.4621       0.8200        0.4860  0.0290
      9      0.8267        0.4601       0.8267        0.4767  0.0397
     10      0.8133        0.4511       0.8133        0.4855  0.0333
     11      0.8133        0.4667       0.8133        0.4846  0.0340
     12      0.8333        0.4570       0.8333        0.4720  0.0387
     13      0.8267        0.4551       0.8267        0.4747  0.0313
     14      0.8467        0.4573       0.8467        0.4596  0.0233
     15      0.8333        0.4589       0.8333        0.4740  0.0231
     16      0.8333        0.4438       0.8333        0.4709  0.0267
```

17	0.8333	0.4694	0.8333	0.4781	0.0277
18	0.8400	0.4532	0.8400	0.4689	0.0258
19	0.8333	0.4576	0.8333	0.4747	0.0346
20	0.8333	0.4568	0.8333	0.4776	0.0283
21	0.8400	0.4676	0.8400	0.4708	0.0314
22	0.8133	0.4615	0.8133	0.4973	0.0255
23	0.8333	0.4521	0.8333	0.4770	0.0297
24	0.8333	0.4450	0.8333	0.4733	0.0281
25	0.8400	0.4557	0.8400	0.4749	0.0271
26	0.8467	0.4604	0.8467	0.4671	0.0266
27	0.8467	0.4508	0.8467	0.4668	0.0286
28	0.8400	0.4411	0.8400	0.4689	0.0271
29	0.8400	0.4552	0.8400	0.4705	0.0303
30	0.8333	0.4519	0.8333	0.4746	0.0265
31	0.8400	0.4436	0.8400	0.4646	0.0249
32	0.8333	0.4394	0.8333	0.4646	0.0232
33	0.8400	0.4586	0.8400	0.4727	0.0249
34	0.8467	0.4499	0.8467	0.4687	0.0218
35	0.8267	0.4507	0.8267	0.4776	0.0254
36	0.8333	0.4506	0.8333	0.4745	0.0241
37	0.8333	0.4518	0.8333	0.4704	0.0212
38	0.8333	0.4557	0.8333	0.4682	0.0229
39	0.8467	0.4281	0.8467	0.4649	0.0223
40	0.8400	0.4541	0.8400	0.4697	0.0240
41	0.8533	0.4555	0.8533	0.4635	0.0246
42	0.8333	0.4488	0.8333	0.4707	0.0339
43	0.8467	0.4667	0.8467	0.4687	0.1323
44	0.8267	0.4644	0.8267	0.4832	0.0433
45	0.8200	0.4503	0.8200	0.4726	0.0288
46	0.8333	0.4454	0.8333	0.4695	0.0246
47	0.8400	0.4745	0.8400	0.4618	0.0230
48	0.8400	0.4688	0.8400	0.4757	0.0227
49	0.8467	0.4501	0.8467	0.4647	0.0232
50	0.8333	0.4381	0.8333	0.4640	0.0218

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.8867	0.6883	0.8867	0.6591	0.0212
2	0.8133	0.6493	0.8133	0.5413	0.0242
3	0.8933	0.5442	0.8933	0.4245	0.0246
4	0.9133	0.4730	0.9133	0.4066	0.0224
5	0.8800	0.4631	0.8800	0.4309	0.0198
6	0.8867	0.4495	0.8867	0.4339	0.0204
7	0.9067	0.4670	0.9067	0.3982	0.0206
8	0.8867	0.4489	0.8867	0.4186	0.0188
9	0.9000	0.4605	0.9000	0.4052	0.0207
10	0.8933	0.4386	0.8933	0.4173	0.0203
11	0.9133	0.4415	0.9133	0.3977	0.0230
12	0.8867	0.4654	0.8867	0.4230	0.0217
13	0.9067	0.4605	0.9067	0.3996	0.0227

14	0.8867	0.4356	0.8867	0.4177	0.0234
15	0.9000	0.4811	0.9000	0.4079	0.0239
16	0.9200	0.4401	0.9200	0.3990	0.0257
17	0.9000	0.4587	0.9000	0.4139	0.0267
18	0.8933	0.4517	0.8933	0.4097	0.0223
19	0.8867	0.4409	0.8867	0.4158	0.0217
20	0.9000	0.4520	0.9000	0.4101	0.0219
21	0.9000	0.4365	0.9000	0.4064	0.0224
22	0.9067	0.4458	0.9067	0.4070	0.0216
23	0.8867	0.4298	0.8867	0.4168	0.0214
24	0.8933	0.4407	0.8933	0.4095	0.0242
25	0.9133	0.4510	0.9133	0.3995	0.0214
26	0.8867	0.4545	0.8867	0.4251	0.0217
27	0.8800	0.4406	0.8800	0.4167	0.0222
28	0.8867	0.4523	0.8867	0.4270	0.0224
29	0.9000	0.4516	0.9000	0.4112	0.0234
30	0.9000	0.4551	0.9000	0.4027	0.0240
31	0.8800	0.4447	0.8800	0.4317	0.0232
32	0.9067	0.4486	0.9067	0.4078	0.0241
33	0.8533	0.4593	0.8533	0.4554	0.0234
34	0.8933	0.4450	0.8933	0.4177	0.0220
35	0.8933	0.4496	0.8933	0.4202	0.0231
36	0.8867	0.4522	0.8867	0.4273	0.0201
37	0.9000	0.4491	0.9000	0.4155	0.0211
38	0.9000	0.4576	0.9000	0.4112	0.0194
39	0.8800	0.4435	0.8800	0.4350	0.0212
40	0.9000	0.4666	0.9000	0.4113	0.0210
41	0.9133	0.4608	0.9133	0.3984	0.0224
42	0.9000	0.4488	0.9000	0.4108	0.0220
43	0.9067	0.4720	0.9067	0.4053	0.0210
44	0.9133	0.4592	0.9133	0.3994	0.0218
45	0.8867	0.4630	0.8867	0.4262	0.0204
46	0.9067	0.4624	0.9067	0.4061	0.0224
47	0.8867	0.4483	0.8867	0.4222	0.0191
48	0.8733	0.4417	0.8733	0.4316	0.0214
49	0.8867	0.4699	0.8867	0.4306	0.0203
50	0.9000	0.4736	0.9000	0.4050	0.0222

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.4800	0.6942	0.4800	0.6938	0.0171
2	0.4800	0.6935	0.4800	0.6937	0.0150
3	0.4800	0.6942	0.4800	0.6936	0.0154
4	0.4800	0.6946	0.4800	0.6936	0.0168
5	0.4800	0.6952	0.4800	0.6935	0.0140
6	0.4800	0.6929	0.4800	0.6934	0.0142
7	0.4800	0.6932	0.4800	0.6933	0.0139
8	0.4800	0.6957	0.4800	0.6933	0.0161
9	0.4800	0.6919	0.4800	0.6932	0.0152
10	0.4800	0.6942	0.4800	0.6931	0.0152

```
     11      0.4800        0.6925       0.4800        0.6931  0.0147
     12      0.4800        0.6926       0.4800        0.6930  0.0164
     13      0.4800        0.6942       0.4800        0.6929  0.0151
     14      0.4800        0.6945       0.4800        0.6929  0.0163
     15      0.4800        0.6931       0.4800        0.6928  0.0159
     16      0.4733        0.6937       0.4733        0.6927  0.0187
     17      0.4400        0.6932       0.4400        0.6927  0.0150
     18      0.5133        0.6930       0.5133        0.6926  0.0186
     19      0.5733        0.6927       0.5733        0.6925  0.0220
     20      0.6533        0.6914       0.6533        0.6925  0.0335
     21      0.7000        0.6928       0.7000        0.6924  0.0258
     22      0.6733        0.6939       0.6733        0.6924  0.0205
     23      0.6133        0.6935       0.6133        0.6923  0.0191
     24      0.6133        0.6933       0.6133        0.6922  0.0229
     25      0.6200        0.6927       0.6200        0.6922  0.0226
     26      0.5733        0.6931       0.5733        0.6921  0.0215
     27      0.5600        0.6909       0.5600        0.6921  0.0180
     28      0.5533        0.6918       0.5533        0.6920  0.0188
     29      0.5467        0.6930       0.5467        0.6919  0.0204
     30      0.5467        0.6928       0.5467        0.6919  0.0205
     31      0.5400        0.6920       0.5400        0.6918  0.0198
     32      0.5333        0.6919       0.5333        0.6917  0.0191
     33      0.5267        0.6935       0.5267        0.6917  0.0209
     34      0.5267        0.6923       0.5267        0.6916  0.0247
     35      0.5267        0.6923       0.5267        0.6916  0.0200
     36      0.5200        0.6927       0.5200        0.6915  0.0219
     37      0.5200        0.6928       0.5200        0.6915  0.0278
     38      0.5200        0.6933       0.5200        0.6914  0.0208

     39      0.5200        0.6904       0.5200        0.6914  0.0293
     40      0.5200        0.6925       0.5200        0.6913  0.0220
     41      0.5200        0.6915       0.5200        0.6912  0.0198
     42      0.5200        0.6925       0.5200        0.6912  0.0232
     43      0.5200        0.6922       0.5200        0.6911  0.0247
     44      0.5200        0.6926       0.5200        0.6911  0.0235
     45      0.5200        0.6925       0.5200        0.6910  0.0229
     46      0.5200        0.6923       0.5200        0.6910  0.0497
     47      0.5200        0.6927       0.5200        0.6909  0.0265
     48      0.5200        0.6904       0.5200        0.6908  0.0206
     49      0.5200        0.6915       0.5200        0.6908  0.0232
     50      0.5200        0.6906       0.5200        0.6907  0.0228
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.4800        0.6937       0.4800        0.6943  0.0224
      2      0.4800        0.6950       0.4800        0.6942  0.0204
      3      0.4800        0.6936       0.4800        0.6941  0.0185
      4      0.4800        0.6949       0.4800        0.6940  0.0218
      5      0.4800        0.6943       0.4800        0.6939  0.0246
      6      0.4800        0.6934       0.4800        0.6938  0.0210
      7      0.4800        0.6933       0.4800        0.6937  0.0223
```

```
      8      0.4800        0.6949       0.4800        0.6936  0.0211
      9      0.4800        0.6947       0.4800        0.6935  0.0179
     10      0.4800        0.6935       0.4800        0.6934  0.0200
     11      0.4800        0.6926       0.4800        0.6934  0.0181
     12      0.4800        0.6929       0.4800        0.6933  0.0174
     13      0.4800        0.6932       0.4800        0.6932  0.0181
     14      0.4800        0.6928       0.4800        0.6931  0.0174
     15      0.4800        0.6941       0.4800        0.6930  0.0170
     16      0.4800        0.6935       0.4800        0.6930  0.0167
     17      0.4800        0.6960       0.4800        0.6929  0.0163
     18      0.4800        0.6946       0.4800        0.6928  0.0164
     19      0.4800        0.6936       0.4800        0.6928  0.0169
     20      0.4800        0.6932       0.4800        0.6927  0.0159
     21      0.4800        0.6934       0.4800        0.6926  0.0145
     22      0.4800        0.6930       0.4800        0.6926  0.0244
     23      0.4800        0.6934       0.4800        0.6925  0.0559
     24      0.4800        0.6932       0.4800        0.6925  0.0223
     25      0.4800        0.6953       0.4800        0.6924  0.0347
     26      0.4800        0.6944       0.4800        0.6924  0.0278
     27      0.4800        0.6922       0.4800        0.6923  0.0408
     28      0.4800        0.6933       0.4800        0.6922  0.0215
     29      0.4800        0.6922       0.4800        0.6922  0.0252
     30      0.4800        0.6916       0.4800        0.6921  0.0241
     31      0.4800        0.6928       0.4800        0.6921  0.0232
     32      0.4800        0.6915       0.4800        0.6920  0.0229
     33      0.4800        0.6924       0.4800        0.6920  0.0224
     34      0.4867        0.6914       0.4867        0.6919  0.0270
     35      0.5067        0.6907       0.5067        0.6918  0.0266
     36      0.6067        0.6918       0.6067        0.6918  0.0403
     37      0.7733        0.6942       0.7733        0.6917  0.0408
     38      0.9000        0.6915       0.9000        0.6917  0.0346
     39      0.8733        0.6926       0.8733        0.6916  0.0345
     40      0.8733        0.6929       0.8733        0.6916  0.0232
     41      0.8400        0.6930       0.8400        0.6915  0.0215
     42      0.7667        0.6924       0.7667        0.6915  0.0270
     43      0.7200        0.6925       0.7200        0.6914  0.0225
     44      0.6600        0.6943       0.6600        0.6914  0.0206
     45      0.6067        0.6916       0.6067        0.6913  0.0193
     46      0.5867        0.6920       0.5867        0.6913  0.0196
     47      0.5467        0.6915       0.5467        0.6912  0.0169
     48      0.5200        0.6906       0.5200        0.6912  0.0168
     49      0.5200        0.6915       0.5200        0.6911  0.0164
     50      0.5200        0.6927       0.5200        0.6911  0.0170
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.7533        0.6854       0.7533        0.6421  0.0202
      2      0.8267        0.6033       0.8267        0.4949  0.0208
      3      0.8267        0.5036       0.8267        0.4846  0.0196
      4      0.8267        0.4751       0.8267        0.4873  0.0202
```

5	0.8000	0.4770	0.8000	0.4912	0.0196
6	0.8200	0.4922	0.8200	0.4866	0.0197
7	0.8267	0.4676	0.8267	0.4884	0.0198
8	0.8200	0.4819	0.8200	0.4910	0.0200
9	0.8267	0.4515	0.8267	0.4902	0.0279
10	0.8000	0.4655	0.8000	0.4911	0.0193
11	0.8133	0.4711	0.8133	0.4825	0.0182
12	0.8200	0.4555	0.8200	0.4809	0.0217
13	0.8333	0.4565	0.8333	0.4709	0.0204
14	0.8267	0.4553	0.8267	0.4850	0.0217
15	0.8267	0.4514	0.8267	0.4741	0.0251
16	0.8400	0.4493	0.8400	0.4707	0.0233
17	0.8400	0.4426	0.8400	0.4733	0.0245
18	0.8333	0.4451	0.8333	0.4755	0.0248
19	0.8267	0.4480	0.8267	0.4789	0.0245
20	0.8333	0.4517	0.8333	0.4746	0.0229
21	0.8333	0.4453	0.8333	0.4791	0.0209
22	0.8267	0.4649	0.8267	0.4805	0.0209
23	0.8333	0.4450	0.8333	0.4800	0.0221
24	0.8267	0.4687	0.8267	0.4813	0.0203
25	0.8267	0.4671	0.8267	0.4829	0.0212
26	0.8200	0.4749	0.8200	0.4870	0.0200
27	0.8400	0.4635	0.8400	0.4762	0.0227
28	0.8200	0.4864	0.8200	0.4777	0.0204
29	0.8200	0.4655	0.8200	0.4880	0.0206
30	0.8200	0.4660	0.8200	0.4791	0.0222
31	0.8267	0.4535	0.8267	0.4832	0.0219
32	0.8200	0.4465	0.8200	0.4793	0.0215
33	0.8200	0.4430	0.8200	0.4843	0.0209
34	0.8333	0.4559	0.8333	0.4764	0.0204
35	0.8067	0.4734	0.8067	0.4998	0.0219
36	0.8133	0.4524	0.8133	0.4844	0.0261
37	0.8267	0.4550	0.8267	0.4817	0.0215
38	0.8333	0.4449	0.8333	0.4782	0.0220
39	0.8467	0.4376	0.8467	0.4696	0.0210
40	0.8333	0.4509	0.8333	0.4799	0.0219
41	0.8333	0.4471	0.8333	0.4753	0.0216
42	0.8333	0.4465	0.8333	0.4758	0.0210
43	0.8467	0.4535	0.8467	0.4628	0.0252
44	0.8267	0.4430	0.8267	0.4750	0.0212
45	0.8267	0.4382	0.8267	0.4775	0.0196
46	0.8400	0.4551	0.8400	0.4768	0.0239
47	0.8267	0.4486	0.8267	0.4794	0.0219
48	0.8333	0.4460	0.8333	0.4790	0.0236
49	0.8333	0.4474	0.8333	0.4725	0.0252
50	0.8400	0.4460	0.8400	0.4582	0.0264
51	0.8333	0.4437	0.8333	0.4638	0.0273
52	0.8333	0.4464	0.8333	0.4691	0.0251
53	0.8333	0.4462	0.8333	0.4790	0.0261

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
54	0.8400	0.4497	0.8400	0.4685	0.0268
55	0.8333	0.4552	0.8333	0.4766	0.0274
56	0.8333	0.4537	0.8333	0.4788	0.0290
57	0.8333	0.4470	0.8333	0.4728	0.0297
58	0.8400	0.4412	0.8400	0.4714	0.0275
59	0.8200	0.4515	0.8200	0.4705	0.0291
60	0.8333	0.4483	0.8333	0.4707	0.0325
61	0.8133	0.4626	0.8133	0.4987	0.0319
62	0.8333	0.4776	0.8333	0.4804	0.0325
63	0.8200	0.4501	0.8200	0.4788	0.0305
64	0.8267	0.4647	0.8267	0.4740	0.0318
65	0.8400	0.4538	0.8400	0.4740	0.0324
66	0.8533	0.4321	0.8533	0.4619	0.0299
67	0.8400	0.4573	0.8400	0.4632	0.0300
68	0.8400	0.4425	0.8400	0.4708	0.0278
69	0.8533	0.4609	0.8533	0.4613	0.0285
70	0.8267	0.4623	0.8267	0.4662	0.0252
71	0.8333	0.4428	0.8333	0.4770	0.0227
72	0.8400	0.4507	0.8400	0.4731	0.0264
73	0.8267	0.4456	0.8267	0.4765	0.0241
74	0.8400	0.4387	0.8400	0.4776	0.0272
75	0.8333	0.4387	0.8333	0.4761	0.0242
76	0.8600	0.4575	0.8600	0.4608	0.0302
77	0.8267	0.4623	0.8267	0.4696	0.0329
78	0.8333	0.4481	0.8333	0.4679	0.0303
79	0.8400	0.4465	0.8400	0.4572	0.0309
80	0.8400	0.4433	0.8400	0.4555	0.0278
81	0.8467	0.4375	0.8467	0.4649	0.0261
82	0.8467	0.4562	0.8467	0.4675	0.0266
83	0.8400	0.4394	0.8400	0.4692	0.0261
84	0.8400	0.4362	0.8400	0.4687	0.0220
85	0.8400	0.4266	0.8400	0.4674	0.0236
86	0.8400	0.4439	0.8400	0.4692	0.0263
87	0.8400	0.4369	0.8400	0.4685	0.0246
88	0.8333	0.4369	0.8333	0.4705	0.0248
89	0.8267	0.4522	0.8267	0.4805	0.0254
90	0.8400	0.4471	0.8400	0.4715	0.0265
91	0.8400	0.4602	0.8400	0.4693	0.0285
92	0.8067	0.4680	0.8067	0.5050	0.0267
93	0.8400	0.4589	0.8400	0.4674	0.0276
94	0.8267	0.4473	0.8267	0.4795	0.0284
95	0.8267	0.4335	0.8267	0.4764	0.0280
96	0.8400	0.4484	0.8400	0.4704	0.0249
97	0.8333	0.4434	0.8333	0.4716	0.0220
98	0.8333	0.4483	0.8333	0.4693	0.0217
99	0.8333	0.4451	0.8333	0.4725	0.0213
100	0.8400	0.4331	0.8400	0.4709	0.0211

1	0.5600	0.6907	0.5600	0.6657	0.0219
2	0.9000	0.6499	0.9000	0.5169	0.0224
3	0.8933	0.5315	0.8933	0.4197	0.0217
4	0.9067	0.4781	0.9067	0.4014	0.0223
5	0.8800	0.4655	0.8800	0.4300	0.0223
6	0.9133	0.4719	0.9133	0.4065	0.0200
7	0.8933	0.4622	0.8933	0.4150	0.0208
8	0.9067	0.4541	0.9067	0.3982	0.0230
9	0.9067	0.4513	0.9067	0.4026	0.0219
10	0.8867	0.4508	0.8867	0.4222	0.0220
11	0.8200	0.4637	0.8200	0.4834	0.0218
12	0.9133	0.4655	0.9133	0.4019	0.0217
13	0.8733	0.4580	0.8733	0.4438	0.0214
14	0.9067	0.4404	0.9067	0.4027	0.0201
15	0.8933	0.4485	0.8933	0.4098	0.0209
16	0.8933	0.4753	0.8933	0.4147	0.0205
17	0.8600	0.4484	0.8600	0.4530	0.0210
18	0.9000	0.4399	0.9000	0.4072	0.0198
19	0.8933	0.4496	0.8933	0.4160	0.0246
20	0.9000	0.4504	0.9000	0.4069	0.0217
21	0.8800	0.4514	0.8800	0.4378	0.0243
22	0.9000	0.4597	0.9000	0.4061	0.0242
23	0.9000	0.4602	0.9000	0.4055	0.0252
24	0.8800	0.4433	0.8800	0.4311	0.0269
25	0.8933	0.4475	0.8933	0.4157	0.0234
26	0.8933	0.4385	0.8933	0.4152	0.0233
27	0.8867	0.4420	0.8867	0.4221	0.0207
28	0.8400	0.4596	0.8400	0.4680	0.0197
29	0.8933	0.4622	0.8933	0.4248	0.0207
30	0.8733	0.4667	0.8733	0.4372	0.0221
31	0.9200	0.4594	0.9200	0.3950	0.0241
32	0.8667	0.4564	0.8667	0.4443	0.0214
33	0.9000	0.4635	0.9000	0.4044	0.0238
34	0.8933	0.4642	0.8933	0.4162	0.0240
35	0.9133	0.4570	0.9133	0.3988	0.0243
36	0.8867	0.4504	0.8867	0.4249	0.0228
37	0.9000	0.4434	0.9000	0.4141	0.0227
38	0.9000	0.4494	0.9000	0.4042	0.0224
39	0.8867	0.4485	0.8867	0.4265	0.0223
40	0.9133	0.4541	0.9133	0.3990	0.0223
41	0.8933	0.4568	0.8933	0.4082	0.0203
42	0.8867	0.4341	0.8867	0.4295	0.0213
43	0.9067	0.4434	0.9067	0.4053	0.0229
44	0.9200	0.4595	0.9200	0.3987	0.0224
45	0.8933	0.4491	0.8933	0.4217	0.0213
46	0.8800	0.4460	0.8800	0.4341	0.0205
47	0.9000	0.4578	0.9000	0.4145	0.0204
48	0.8800	0.4534	0.8800	0.4163	0.0194
49	0.8867	0.4392	0.8867	0.4292	0.0207

```
     50      0.8867        0.4378       0.8867        0.4196  0.0215
     51      0.9067        0.4406       0.9067        0.4058  0.0201

     52      0.9067        0.4504       0.9067        0.4011  0.0196
     53      0.8867        0.4374       0.8867        0.4314  0.0208
     54      0.9000        0.4464       0.9000        0.4113  0.0192
     55      0.9067        0.4547       0.9067        0.4034  0.0198
     56      0.8867        0.4281       0.8867        0.4238  0.0213
     57      0.9000        0.4329       0.9000        0.4080  0.0190
     58      0.8600        0.4469       0.8600        0.4488  0.0207
     59      0.9200        0.4431       0.9200        0.3952  0.0203
     60      0.9067        0.4446       0.9067        0.3999  0.0203
     61      0.8867        0.4369       0.8867        0.4237  0.0205
     62      0.9000        0.4364       0.9000        0.4099  0.0193
     63      0.8933        0.4410       0.8933        0.4144  0.0200
     64      0.9067        0.4262       0.9067        0.4044  0.0176
     65      0.9133        0.4324       0.9133        0.4051  0.0174
     66      0.9133        0.4551       0.9133        0.4013  0.0177
     67      0.8867        0.4335       0.8867        0.4222  0.0175
     68      0.9000        0.4628       0.9000        0.4137  0.0194
     69      0.8733        0.4494       0.8733        0.4387  0.0198
     70      0.8867        0.4536       0.8867        0.4241  0.0177
Stopping since valid_loss has not improved in the last 40 epochs.
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5200        0.6927       0.5200        0.6927  0.0157
      2      0.5200        0.6930       0.5200        0.6927  0.0142
      3      0.5200        0.6923       0.5200        0.6927  0.0154
      4      0.5200        0.6917       0.5200        0.6926  0.0138
      5      0.5200        0.6935       0.5200        0.6926  0.0144
      6      0.5200        0.6921       0.5200        0.6926  0.0165
      7      0.5200        0.6916       0.5200        0.6926  0.0141
      8      0.5200        0.6931       0.5200        0.6925  0.0165
      9      0.5200        0.6921       0.5200        0.6925  0.0144
     10      0.5200        0.6920       0.5200        0.6925  0.0176
     11      0.5200        0.6933       0.5200        0.6925  0.0161
     12      0.5200        0.6923       0.5200        0.6925  0.0143
     13      0.5200        0.6941       0.5200        0.6924  0.0189
     14      0.5200        0.6929       0.5200        0.6924  0.0167
     15      0.5200        0.6922       0.5200        0.6924  0.0164
     16      0.5200        0.6928       0.5200        0.6924  0.0157
     17      0.5200        0.6914       0.5200        0.6924  0.0163
     18      0.5200        0.6934       0.5200        0.6923  0.0162
     19      0.5200        0.6913       0.5200        0.6923  0.0147
     20      0.5200        0.6920       0.5200        0.6923  0.0139
     21      0.5200        0.6920       0.5200        0.6923  0.0139
     22      0.5200        0.6914       0.5200        0.6922  0.0154
     23      0.5200        0.6915       0.5200        0.6922  0.0150
     24      0.5200        0.6934       0.5200        0.6922  0.0147
     25      0.5200        0.6923       0.5200        0.6922  0.0153
```

26	0.5200	0.6939	0.5200	0.6921	0.0154
27	0.5200	0.6930	0.5200	0.6921	0.0142
28	0.5200	0.6926	0.5200	0.6921	0.0166
29	0.5200	0.6914	0.5200	0.6921	0.0142
30	0.5200	0.6923	0.5200	0.6921	0.0159
31	0.5200	0.6922	0.5200	0.6921	0.0166
32	0.5200	0.6939	0.5200	0.6920	0.0154
33	0.5200	0.6929	0.5200	0.6920	0.0148
34	0.5200	0.6917	0.5200	0.6920	0.0157
35	0.5200	0.6922	0.5200	0.6920	0.0169
36	0.5200	0.6925	0.5200	0.6919	0.0194
37	0.5200	0.6929	0.5200	0.6919	0.0188
38	0.5200	0.6916	0.5200	0.6919	0.0166
39	0.5200	0.6918	0.5200	0.6919	0.0164
40	0.5200	0.6944	0.5200	0.6918	0.0169
41	0.5200	0.6918	0.5200	0.6918	0.0160
42	0.5200	0.6912	0.5200	0.6918	0.0165
43	0.5200	0.6920	0.5200	0.6918	0.0174
44	0.5200	0.6910	0.5200	0.6917	0.0162
45	0.5200	0.6924	0.5200	0.6917	0.0166
46	0.5200	0.6924	0.5200	0.6917	0.0165
47	0.5200	0.6905	0.5200	0.6917	0.0174
48	0.5200	0.6931	0.5200	0.6917	0.0179
49	0.5200	0.6918	0.5200	0.6916	0.0170
50	0.5200	0.6924	0.5200	0.6916	0.0176
51	0.5200	0.6928	0.5200	0.6916	0.0180
52	0.5200	0.6919	0.5200	0.6916	0.0163
53	0.5200	0.6916	0.5200	0.6915	0.0170
54	0.5200	0.6909	0.5200	0.6915	0.0166
55	0.5200	0.6922	0.5200	0.6915	0.0139
56	0.5200	0.6901	0.5200	0.6914	0.0173
57	0.5200	0.6905	0.5200	0.6914	0.0163
58	0.5200	0.6916	0.5200	0.6914	0.0161
59	0.5200	0.6920	0.5200	0.6914	0.0159
60	0.5200	0.6916	0.5200	0.6913	0.0149
61	0.5200	0.6911	0.5200	0.6913	0.0144
62	0.5200	0.6935	0.5200	0.6913	0.0154
63	0.5200	0.6912	0.5200	0.6913	0.0148
64	0.5200	0.6933	0.5200	0.6912	0.0166
65	0.5200	0.6930	0.5200	0.6912	0.0181
66	0.5200	0.6924	0.5200	0.6912	0.0166
67	0.5200	0.6925	0.5200	0.6912	0.0174
68	0.5200	0.6909	0.5200	0.6912	0.0176
69	0.5200	0.6911	0.5200	0.6911	0.0181
70	0.5200	0.6915	0.5200	0.6911	0.0185
71	0.5200	0.6918	0.5200	0.6911	0.0202
72	0.5200	0.6912	0.5200	0.6910	0.0167
73	0.5200	0.6916	0.5200	0.6910	0.0209
74	0.5200	0.6927	0.5200	0.6910	0.0193

75	0.5200	0.6903	0.5200	0.6910	0.0187
76	0.5200	0.6913	0.5200	0.6910	0.0202
77	0.5200	0.6914	0.5200	0.6909	0.0187
78	0.5200	0.6913	0.5200	0.6909	0.0196
79	0.5200	0.6913	0.5200	0.6909	0.0195
80	0.5200	0.6903	0.5200	0.6908	0.0195
81	0.5200	0.6911	0.5200	0.6908	0.0170
82	0.5200	0.6912	0.5200	0.6908	0.0191
83	0.5200	0.6909	0.5200	0.6908	0.0196
84	0.5200	0.6922	0.5200	0.6907	0.0173
85	0.5200	0.6915	0.5200	0.6907	0.0198
86	0.5200	0.6909	0.5200	0.6907	0.0185
87	0.5200	0.6919	0.5200	0.6907	0.0187
88	0.5200	0.6915	0.5200	0.6906	0.0188
89	0.5200	0.6913	0.5200	0.6906	0.0189
90	0.5200	0.6915	0.5200	0.6906	0.0181
91	0.5200	0.6916	0.5200	0.6906	0.0185
92	0.5200	0.6913	0.5200	0.6905	0.0172
93	0.5200	0.6903	0.5200	0.6905	0.0172
94	0.5200	0.6923	0.5200	0.6905	0.0148
95	0.5200	0.6919	0.5200	0.6905	0.0163
96	0.5200	0.6914	0.5200	0.6905	0.0141
97	0.5200	0.6922	0.5200	0.6904	0.0140
98	0.5200	0.6906	0.5200	0.6904	0.0163
99	0.5200	0.6914	0.5200	0.6904	0.0155
100	0.5200	0.6935	0.5200	0.6904	0.0180

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.5200	0.6930	0.5200	0.6934	0.0161
2	0.5200	0.6942	0.5200	0.6934	0.0179
3	0.5200	0.6934	0.5200	0.6933	0.0170
4	0.5200	0.6922	0.5200	0.6933	0.0185
5	0.5200	0.6920	0.5200	0.6933	0.0183
6	0.5200	0.6932	0.5200	0.6932	0.0198
7	0.5200	0.6926	0.5200	0.6932	0.0189
8	0.5200	0.6915	0.5200	0.6932	0.0171
9	0.5200	0.6923	0.5200	0.6931	0.0214
10	0.5200	0.6935	0.5200	0.6931	0.0190
11	0.5200	0.6940	0.5200	0.6931	0.0185
12	0.5200	0.6933	0.5200	0.6930	0.0170
13	0.5200	0.6923	0.5200	0.6930	0.0178
14	0.5200	0.6908	0.5200	0.6930	0.0208
15	0.5200	0.6930	0.5200	0.6930	0.0189
16	0.5200	0.6925	0.5200	0.6929	0.0187
17	0.5200	0.6929	0.5200	0.6929	0.0180
18	0.5200	0.6927	0.5200	0.6929	0.0174
19	0.5200	0.6935	0.5200	0.6928	0.0158
20	0.5200	0.6919	0.5200	0.6928	0.0185
21	0.5200	0.6933	0.5200	0.6927	0.0151

22	0.5200	0.6918	0.5200	0.6927	0.0176
23	0.5200	0.6912	0.5200	0.6927	0.0178
24	0.5200	0.6927	0.5200	0.6926	0.0170
25	0.5200	0.6925	0.5200	0.6926	0.0187
26	0.5200	0.6914	0.5200	0.6926	0.0175
27	0.5200	0.6918	0.5200	0.6926	0.0170
28	0.5200	0.6927	0.5200	0.6925	0.0169
29	0.5200	0.6925	0.5200	0.6925	0.0174
30	0.5200	0.6920	0.5200	0.6925	0.0154
31	0.5200	0.6931	0.5200	0.6924	0.0163
32	0.5200	0.6926	0.5200	0.6924	0.0173
33	0.5200	0.6924	0.5200	0.6924	0.0157
34	0.5200	0.6922	0.5200	0.6923	0.0176
35	0.5200	0.6951	0.5200	0.6923	0.0177
36	0.5200	0.6912	0.5200	0.6923	0.0175
37	0.5200	0.6912	0.5200	0.6922	0.0178
38	0.5200	0.6915	0.5200	0.6922	0.0175
39	0.5200	0.6921	0.5200	0.6922	0.0171
40	0.5200	0.6920	0.5200	0.6921	0.0169
41	0.5200	0.6935	0.5200	0.6921	0.0168
42	0.5200	0.6910	0.5200	0.6921	0.0167
43	0.5200	0.6918	0.5200	0.6920	0.0166
44	0.5200	0.6923	0.5200	0.6920	0.0160
45	0.5200	0.6936	0.5200	0.6920	0.0143
46	0.5200	0.6918	0.5200	0.6919	0.0152
47	0.5200	0.6921	0.5200	0.6919	0.0137
48	0.5200	0.6919	0.5200	0.6918	0.0135
49	0.5200	0.6925	0.5200	0.6918	0.0153
50	0.5200	0.6937	0.5200	0.6918	0.0152
51	0.5200	0.6914	0.5200	0.6918	0.0157
52	0.5200	0.6916	0.5200	0.6917	0.0157
53	0.5200	0.6912	0.5200	0.6917	0.0167
54	0.5200	0.6908	0.5200	0.6917	0.0163
55	0.5200	0.6921	0.5200	0.6916	0.0169
56	0.5200	0.6919	0.5200	0.6916	0.0182
57	0.5200	0.6920	0.5200	0.6915	0.0165
58	0.5200	0.6916	0.5200	0.6915	0.0164
59	0.5200	0.6913	0.5200	0.6915	0.0171
60	0.5200	0.6923	0.5200	0.6914	0.0169
61	0.5200	0.6932	0.5200	0.6914	0.0142
62	0.5200	0.6907	0.5200	0.6914	0.0157
63	0.5200	0.6918	0.5200	0.6913	0.0149
64	0.5200	0.6919	0.5200	0.6913	0.0147
65	0.5200	0.6913	0.5200	0.6913	0.0165
66	0.5200	0.6912	0.5200	0.6912	0.0160
67	0.5200	0.6924	0.5200	0.6912	0.0149
68	0.5200	0.6893	0.5200	0.6911	0.0160
69	0.5200	0.6925	0.5200	0.6911	0.0155
70	0.5200	0.6911	0.5200	0.6911	0.0162

```
     71      0.5200        0.6913       0.5200        0.6910  0.0164
     72      0.5200        0.6907       0.5200        0.6910  0.0155
     73      0.5200        0.6921       0.5200        0.6909  0.0142
     74      0.5200        0.6911       0.5200        0.6909  0.0147
     75      0.5200        0.6913       0.5200        0.6908  0.0146
     76      0.5200        0.6896       0.5200        0.6908  0.0139
     77      0.5200        0.6902       0.5200        0.6908  0.0152
     78      0.5200        0.6905       0.5200        0.6907  0.0139
     79      0.5200        0.6905       0.5200        0.6907  0.0147
     80      0.5200        0.6907       0.5200        0.6906  0.0146
     81      0.5200        0.6917       0.5200        0.6906  0.0140
     82      0.5200        0.6912       0.5200        0.6906  0.0150
     83      0.5200        0.6923       0.5200        0.6905  0.0154
     84      0.5200        0.6906       0.5200        0.6905  0.0145
     85      0.5200        0.6906       0.5200        0.6904  0.0151
     86      0.5200        0.6920       0.5200        0.6904  0.0156

     87      0.5200        0.6922       0.5200        0.6904  0.0148
     88      0.5200        0.6916       0.5200        0.6904  0.0151
     89      0.5200        0.6912       0.5200        0.6903  0.0156
     90      0.5200        0.6908       0.5200        0.6903  0.0156
     91      0.5200        0.6912       0.5200        0.6902  0.0163
     92      0.5200        0.6907       0.5200        0.6902  0.0145
     93      0.5200        0.6909       0.5200        0.6902  0.0164
     94      0.5200        0.6915       0.5200        0.6901  0.0166
     95      0.5200        0.6908       0.5200        0.6901  0.0151
     96      0.5200        0.6922       0.5200        0.6900  0.0164
     97      0.5200        0.6907       0.5200        0.6900  0.0167
     98      0.5200        0.6897       0.5200        0.6899  0.0169
     99      0.5200        0.6907       0.5200        0.6899  0.0150
    100      0.5200        0.6917       0.5200        0.6898  0.0163
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5933        0.6912       0.5933        0.6712  0.0216
      2      0.8200        0.6427       0.8200        0.5294  0.0182
      3      0.8133        0.5303       0.8133        0.4906  0.0168
      4      0.8000        0.4769       0.8000        0.4951  0.0170
      5      0.8267        0.4599       0.8267        0.4834  0.0186
      6      0.8067        0.4787       0.8067        0.5026  0.0186
      7      0.7867        0.4644       0.7867        0.5019  0.0195
      8      0.8200        0.4808       0.8200        0.4832  0.0204
      9      0.7800        0.4775       0.7800        0.5226  0.0214
     10      0.8467        0.4825       0.8467        0.4742  0.0216
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.7667        0.6912       0.7667        0.6592  0.0209
      2      0.8867        0.6313       0.8867        0.4828  0.0202
      3      0.8600        0.5472       0.8600        0.4416  0.0221
      4      0.9067        0.4737       0.9067        0.4038  0.0201
      5      0.8867        0.4437       0.8867        0.4270  0.0209
```

```
      6      0.9000        0.4945       0.9000        0.4134  0.0202
      7      0.8933        0.4735       0.8933        0.4221  0.0191
      8      0.8733        0.4557       0.8733        0.4358  0.0200
      9      0.9000        0.4544       0.9000        0.4034  0.0197
     10      0.8867        0.4433       0.8867        0.4192  0.0199
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5200        0.6932       0.5200        0.6937  0.0156
      2      0.5200        0.6932       0.5200        0.6936  0.0138
      3      0.5200        0.6931       0.5200        0.6936  0.0150
      4      0.5200        0.6910       0.5200        0.6936  0.0141
      5      0.5200        0.6945       0.5200        0.6936  0.0150
      6      0.5200        0.6925       0.5200        0.6936  0.0191
      7      0.5200        0.6933       0.5200        0.6935  0.0206
      8      0.5200        0.6927       0.5200        0.6935  0.0171
      9      0.5200        0.6930       0.5200        0.6935  0.0157
     10      0.5200        0.6932       0.5200        0.6935  0.0152
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.4800        0.6962       0.4800        0.6953  0.0162
      2      0.4800        0.6955       0.4800        0.6952  0.0150
      3      0.4800        0.6946       0.4800        0.6950  0.0166
      4      0.4800        0.6963       0.4800        0.6949  0.0149
      5      0.4800        0.6950       0.4800        0.6947  0.0164
      6      0.4800        0.6943       0.4800        0.6946  0.0146
      7      0.4800        0.6940       0.4800        0.6944  0.0161
      8      0.4800        0.6944       0.4800        0.6943  0.0152
      9      0.4800        0.6942       0.4800        0.6942  0.0151
     10      0.4800        0.6943       0.4800        0.6941  0.0166
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.8133        0.6877       0.8133        0.6516  0.0206
      2      0.8200        0.5952       0.8200        0.4894  0.0211
      3      0.8333        0.4923       0.8333        0.4770  0.0198
      4      0.8200        0.5031       0.8200        0.4916  0.0194
      5      0.7933        0.4629       0.7933        0.5028  0.0200
      6      0.8333        0.4792       0.8333        0.4752  0.0195
      7      0.8267        0.4775       0.8267        0.4811  0.0193
      8      0.8267        0.4765       0.8267        0.4823  0.0183
      9      0.8200        0.4675       0.8200        0.4785  0.0186
     10      0.8200        0.4742       0.8200        0.4852  0.0199
     11      0.8267        0.4581       0.8267        0.4759  0.0220
     12      0.8200        0.4486       0.8200        0.4882  0.0226
     13      0.8267        0.4517       0.8267        0.4711  0.0225
     14      0.8400        0.4538       0.8400        0.4730  0.0239
     15      0.8333        0.4571       0.8333        0.4737  0.0220
     16      0.8200        0.4881       0.8200        0.4819  0.0235
     17      0.8200        0.4609       0.8200        0.4811  0.0229
     18      0.8267        0.4545       0.8267        0.4777  0.0216
```

```
     19      0.8200        0.4647       0.8200        0.4639  0.0240
     20      0.8400        0.4478       0.8400        0.4590  0.0268
     21      0.8400        0.4577       0.8400        0.4613  0.0268
     22      0.8400        0.4621       0.8400        0.4772  0.0263
     23      0.8400        0.4706       0.8400        0.4670  0.0259
     24      0.8400        0.4697       0.8400        0.4588  0.0221
     25      0.8267        0.4537       0.8267        0.4725  0.0210
     26      0.8267        0.4550       0.8267        0.4772  0.0209
     27      0.8200        0.4403       0.8200        0.4701  0.0198
     28      0.8533        0.4429       0.8533        0.4644  0.0187
     29      0.8333        0.4494       0.8333        0.4731  0.0211
     30      0.8333        0.4495       0.8333        0.4813  0.0271
     31      0.8333        0.4808       0.8333        0.4704  0.0257
     32      0.8533        0.4414       0.8533        0.4678  0.0246
     33      0.8400        0.4517       0.8400        0.4752  0.0240
     34      0.8467        0.4562       0.8467        0.4642  0.0245
     35      0.8400        0.4479       0.8400        0.4668  0.0216
     36      0.8333        0.4511       0.8333        0.4685  0.0225
     37      0.8467        0.4546       0.8467        0.4576  0.0216
     38      0.8400        0.4421       0.8400        0.4643  0.0216
     39      0.8667        0.4519       0.8667        0.4515  0.0214
     40      0.8467        0.4547       0.8467        0.4662  0.0227
     41      0.8400        0.4425       0.8400        0.4632  0.0211

     42      0.8400        0.4503       0.8400        0.4674  0.0213
     43      0.8200        0.4597       0.8200        0.4823  0.0220
     44      0.8333        0.4519       0.8333        0.4709  0.0183
     45      0.8533        0.4605       0.8533        0.4584  0.0202
     46      0.8400        0.4567       0.8400        0.4749  0.0191
     47      0.8467        0.4455       0.8467        0.4617  0.0194
     48      0.8333        0.4527       0.8333        0.4728  0.0244
     49      0.8400        0.4441       0.8400        0.4725  0.0207
     50      0.8533        0.4406       0.8533        0.4607  0.0219
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5200        0.6922       0.5200        0.6754  0.0215
      2      0.8933        0.6622       0.8933        0.5408  0.0222
      3      0.9200        0.5523       0.9200        0.4128  0.0226
      4      0.9067        0.4769       0.9067        0.4046  0.0233
      5      0.8400        0.4980       0.8400        0.4653  0.0211
      6      0.9067        0.4730       0.9067        0.3998  0.0228
      7      0.8467        0.4808       0.8467        0.4579  0.0221
      8      0.8800        0.4709       0.8800        0.4272  0.0214
      9      0.8933        0.4606       0.8933        0.4161  0.0216
     10      0.8933        0.4464       0.8933        0.4153  0.0251
     11      0.9133        0.4564       0.9133        0.3991  0.0249
     12      0.8867        0.4629       0.8867        0.4254  0.0229
     13      0.8933        0.4515       0.8933        0.4191  0.0222
     14      0.9067        0.4369       0.9067        0.4022  0.0222
     15      0.8733        0.4418       0.8733        0.4423  0.0207
```

16	0.9067	0.4806	0.9067	0.4063	0.0220
17	0.8800	0.4629	0.8800	0.4286	0.0204
18	0.9067	0.4563	0.9067	0.3975	0.0222
19	0.8933	0.4522	0.8933	0.4170	0.0223
20	0.8933	0.4451	0.8933	0.4093	0.0203
21	0.8800	0.4443	0.8800	0.4389	0.0234
22	0.9133	0.4488	0.9133	0.4010	0.0245
23	0.8867	0.4581	0.8867	0.4233	0.0242
24	0.8800	0.4471	0.8800	0.4372	0.0215
25	0.8933	0.4455	0.8933	0.4090	0.0207
26	0.8867	0.4439	0.8867	0.4223	0.0213
27	0.8933	0.4304	0.8933	0.4134	0.0206
28	0.9000	0.4698	0.9000	0.4071	0.0209
29	0.9067	0.4428	0.9067	0.4069	0.0189
30	0.9067	0.4487	0.9067	0.4022	0.0200
31	0.8800	0.4515	0.8800	0.4327	0.0185
32	0.9133	0.4532	0.9133	0.4023	0.0205
33	0.8867	0.4518	0.8867	0.4248	0.0207
34	0.8933	0.4356	0.8933	0.4113	0.0204
35	0.8733	0.4517	0.8733	0.4381	0.0198
36	0.9067	0.4463	0.9067	0.4099	0.0211
37	0.8867	0.4418	0.8867	0.4220	0.0204
38	0.9067	0.4445	0.9067	0.4040	0.0209
39	0.8933	0.4375	0.8933	0.4166	0.0186
40	0.9000	0.4469	0.9000	0.4083	0.0200
41	0.9067	0.4475	0.9067	0.4014	0.0169
42	0.8867	0.4399	0.8867	0.4184	0.0190
43	0.8867	0.4342	0.8867	0.4146	0.0170
44	0.8800	0.4412	0.8800	0.4345	0.0194
45	0.8933	0.4467	0.8933	0.4096	0.0201
46	0.8867	0.4564	0.8867	0.4274	0.0189
47	0.9067	0.4389	0.9067	0.4001	0.0217
48	0.8800	0.4374	0.8800	0.4312	0.0201
49	0.9000	0.4499	0.9000	0.4045	0.0223
50	0.8867	0.4284	0.8867	0.4237	0.0189

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.5200	0.6947	0.5200	0.6933	0.0160
2	0.5200	0.6930	0.5200	0.6932	0.0173
3	0.5200	0.6926	0.5200	0.6931	0.0167
4	0.5200	0.6917	0.5200	0.6931	0.0167
5	0.5200	0.6948	0.5200	0.6930	0.0166
6	0.5200	0.6949	0.5200	0.6929	0.0175
7	0.5200	0.6925	0.5200	0.6929	0.0165
8	0.5200	0.6932	0.5200	0.6928	0.0169
9	0.5200	0.6917	0.5200	0.6928	0.0169
10	0.5200	0.6915	0.5200	0.6927	0.0158
11	0.5200	0.6926	0.5200	0.6926	0.0156
12	0.5200	0.6936	0.5200	0.6925	0.0164

```
     13      0.5200        0.6930       0.5200        0.6925  0.0164
     14      0.5200        0.6924       0.5200        0.6924  0.0159
     15      0.5200        0.6929       0.5200        0.6924  0.0157
     16      0.5200        0.6916       0.5200        0.6923  0.0138
     17      0.5200        0.6950       0.5200        0.6922  0.0135
     18      0.5200        0.6943       0.5200        0.6922  0.0142
     19      0.5200        0.6942       0.5200        0.6921  0.0138
     20      0.5200        0.6921       0.5200        0.6921  0.0151
     21      0.5200        0.6916       0.5200        0.6920  0.0182
     22      0.5200        0.6912       0.5200        0.6919  0.0181
     23      0.5200        0.6942       0.5200        0.6919  0.0168
     24      0.5200        0.6923       0.5200        0.6918  0.0162
     25      0.5200        0.6914       0.5200        0.6917  0.0170
     26      0.5200        0.6932       0.5200        0.6917  0.0162
     27      0.5200        0.6923       0.5200        0.6916  0.0161
     28      0.5200        0.6930       0.5200        0.6916  0.0159
     29      0.5200        0.6922       0.5200        0.6915  0.0159
     30      0.5200        0.6946       0.5200        0.6915  0.0157
     31      0.5200        0.6918       0.5200        0.6914  0.0160
     32      0.5200        0.6931       0.5200        0.6914  0.0164
     33      0.5200        0.6913       0.5200        0.6913  0.0195
     34      0.5200        0.6912       0.5200        0.6912  0.0174
     35      0.5200        0.6930       0.5200        0.6912  0.0181
     36      0.5200        0.6935       0.5200        0.6912  0.0181
     37      0.5200        0.6899       0.5200        0.6911  0.0198
     38      0.5200        0.6936       0.5200        0.6910  0.0183
     39      0.5200        0.6912       0.5200        0.6910  0.0191
     40      0.5200        0.6906       0.5200        0.6909  0.0185
     41      0.5200        0.6922       0.5200        0.6908  0.0178
     42      0.5200        0.6937       0.5200        0.6908  0.0159
     43      0.5200        0.6903       0.5200        0.6907  0.0160
     44      0.5200        0.6916       0.5200        0.6906  0.0159
     45      0.5200        0.6914       0.5200        0.6905  0.0159
     46      0.5200        0.6910       0.5200        0.6905  0.0158

     47      0.5200        0.6926       0.5200        0.6904  0.0164
     48      0.5200        0.6959       0.5200        0.6904  0.0173
     49      0.5200        0.6901       0.5200        0.6903  0.0180
     50      0.5200        0.6927       0.5200        0.6902  0.0179
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.4800        0.6925       0.4800        0.6922  0.0183
      2      0.4800        0.6939       0.4800        0.6921  0.0171
      3      0.4800        0.6921       0.4800        0.6921  0.0181
      4      0.4800        0.6933       0.4800        0.6920  0.0198
      5      0.4800        0.6932       0.4800        0.6919  0.0186
      6      0.4800        0.6910       0.4800        0.6919  0.0260
      7      0.4800        0.6904       0.4800        0.6918  0.0179
      8      0.4800        0.6944       0.4800        0.6917  0.0214
      9      0.4800        0.6907       0.4800        0.6917  0.0207
```

```
     10      0.4800        0.6921       0.4800        0.6916  0.0188
     11      0.4867        0.6941       0.4867        0.6916  0.0187
     12      0.5000        0.6899       0.5000        0.6915  0.0178
     13      0.5067        0.6910       0.5067        0.6915  0.0186
     14      0.5400        0.6920       0.5400        0.6914  0.0186
     15      0.6133        0.6935       0.6133        0.6914  0.0219
     16      0.6867        0.6909       0.6867        0.6913  0.0208
     17      0.8067        0.6926       0.8067        0.6913  0.0215
     18      0.8867        0.6935       0.8867        0.6912  0.0204
     19      0.8933        0.6917       0.8933        0.6912  0.0187
     20      0.8867        0.6928       0.8867        0.6911  0.0215
     21      0.8333        0.6922       0.8333        0.6911  0.0202
     22      0.7867        0.6916       0.7867        0.6910  0.0197
     23      0.7467        0.6916       0.7467        0.6910  0.0174
     24      0.7067        0.6925       0.7067        0.6909  0.0195
     25      0.6533        0.6917       0.6533        0.6909  0.0172
     26      0.6133        0.6920       0.6133        0.6908  0.0202
     27      0.6067        0.6918       0.6067        0.6908  0.0207
     28      0.5667        0.6911       0.5667        0.6907  0.0207
     29      0.5400        0.6905       0.5400        0.6906  0.0199
     30      0.5400        0.6920       0.5400        0.6906  0.0198
     31      0.5400        0.6905       0.5400        0.6905  0.0206
     32      0.5400        0.6924       0.5400        0.6905  0.0204
     33      0.5400        0.6904       0.5400        0.6905  0.0221
     34      0.5400        0.6894       0.5400        0.6904  0.0196
     35      0.5267        0.6917       0.5267        0.6904  0.0182
     36      0.5200        0.6901       0.5200        0.6903  0.0165
     37      0.5200        0.6908       0.5200        0.6902  0.0162
     38      0.5200        0.6919       0.5200        0.6902  0.0228
     39      0.5200        0.6905       0.5200        0.6902  0.0169
     40      0.5200        0.6915       0.5200        0.6901  0.0181
     41      0.5200        0.6940       0.5200        0.6901  0.0170
     42      0.5200        0.6899       0.5200        0.6900  0.0189
     43      0.5200        0.6908       0.5200        0.6899  0.0170
     44      0.5200        0.6884       0.5200        0.6899  0.0170
     45      0.5200        0.6892       0.5200        0.6898  0.0167
     46      0.5200        0.6902       0.5200        0.6898  0.0164
     47      0.5200        0.6885       0.5200        0.6897  0.0167
     48      0.5200        0.6911       0.5200        0.6897  0.0154
     49      0.5200        0.6893       0.5200        0.6896  0.0147
     50      0.5200        0.6901       0.5200        0.6896  0.0167
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.7867        0.6810       0.7867        0.6409  0.0186
      2      0.8133        0.6108       0.8133        0.5007  0.0209
      3      0.8267        0.5062       0.8267        0.4784  0.0187
      4      0.8133        0.4745       0.8133        0.4818  0.0188
      5      0.7867        0.4911       0.7867        0.5154  0.0212
      6      0.8200        0.4851       0.8200        0.4842  0.0195
```

7	0.8200	0.4602	0.8200	0.4908	0.0196
8	0.8133	0.4618	0.8133	0.4937	0.0215
9	0.8200	0.4776	0.8200	0.4897	0.0192
10	0.8133	0.4510	0.8133	0.4913	0.0176
11	0.8133	0.4656	0.8133	0.4879	0.0169
12	0.8267	0.4572	0.8267	0.4851	0.0169
13	0.8267	0.4571	0.8267	0.4862	0.0185
14	0.8267	0.4651	0.8267	0.4846	0.0168
15	0.8200	0.4784	0.8200	0.4878	0.0186
16	0.7933	0.4854	0.7933	0.5094	0.0177
17	0.8133	0.4704	0.8133	0.4863	0.0182
18	0.8133	0.4695	0.8133	0.4915	0.0195
19	0.8133	0.4724	0.8133	0.4919	0.0197
20	0.8067	0.4648	0.8067	0.4884	0.0197
21	0.8200	0.4625	0.8200	0.4856	0.0193
22	0.8133	0.4666	0.8133	0.4859	0.0198
23	0.8067	0.4521	0.8067	0.4828	0.0203
24	0.8200	0.4516	0.8200	0.4881	0.0202
25	0.8133	0.4557	0.8133	0.4904	0.0195
26	0.8200	0.4589	0.8200	0.4853	0.0204
27	0.8267	0.4620	0.8267	0.4815	0.0200
28	0.8200	0.4526	0.8200	0.4789	0.0195
29	0.8333	0.4448	0.8333	0.4855	0.0213
30	0.8200	0.4497	0.8200	0.4895	0.0198
31	0.8267	0.4499	0.8267	0.4774	0.0216
32	0.8200	0.4613	0.8200	0.4810	0.0205
33	0.8267	0.4574	0.8267	0.4805	0.0195
34	0.8200	0.4493	0.8200	0.4855	0.0188
35	0.8333	0.4502	0.8333	0.4779	0.0192
36	0.8267	0.4522	0.8267	0.4831	0.0207
37	0.8133	0.4647	0.8133	0.4773	0.0232
38	0.8267	0.4654	0.8267	0.4808	0.0213
39	0.8200	0.4531	0.8200	0.4909	0.0241
40	0.8333	0.4531	0.8333	0.4711	0.0200
41	0.8333	0.4493	0.8333	0.4772	0.0199
42	0.8333	0.4600	0.8333	0.4787	0.0201
43	0.8333	0.4493	0.8333	0.4785	0.0202
44	0.8133	0.4546	0.8133	0.4953	0.0213
45	0.8333	0.4592	0.8333	0.4801	0.0190
46	0.8200	0.4401	0.8200	0.4841	0.0198
47	0.8267	0.4378	0.8267	0.4797	0.0181
48	0.8267	0.4637	0.8267	0.4817	0.0173
49	0.8267	0.4513	0.8267	0.4753	0.0189
50	0.8200	0.4380	0.8200	0.4896	0.0191
51	0.8333	0.4402	0.8333	0.4748	0.0172
52	0.8333	0.4378	0.8333	0.4822	0.0189
53	0.8267	0.4592	0.8267	0.4794	0.0174
54	0.8267	0.4641	0.8267	0.4749	0.0186
55	0.8333	0.4617	0.8333	0.4788	0.0184

```
     56      0.8200        0.4672       0.8200        0.4787  0.0176
     57      0.8333        0.4465       0.8333        0.4806  0.0185
     58      0.8267        0.4415       0.8267        0.4820  0.0180
     59      0.8400        0.4437       0.8400        0.4773  0.0176
     60      0.8200        0.4353       0.8200        0.4913  0.0178
     61      0.8200        0.4595       0.8200        0.4791  0.0195
     62      0.8267        0.4509       0.8267        0.4740  0.0185
     63      0.8400        0.4528       0.8400        0.4763  0.0168
     64      0.8267        0.4525       0.8267        0.4854  0.0181
     65      0.8267        0.4753       0.8267        0.4804  0.0170
     66      0.8267        0.4569       0.8267        0.4817  0.0176
     67      0.8200        0.4460       0.8200        0.4906  0.0168
     68      0.8333        0.4460       0.8333        0.4803  0.0167
     69      0.8067        0.4512       0.8067        0.4990  0.0167
     70      0.8200        0.4678       0.8200        0.4857  0.0171
     71      0.8267        0.4439       0.8267        0.4856  0.0191
     72      0.8267        0.4482       0.8267        0.4851  0.0172
     73      0.8333        0.4399       0.8333        0.4816  0.0175
     74      0.8333        0.4507       0.8333        0.4787  0.0176
     75      0.8200        0.4533       0.8200        0.4788  0.0181
     76      0.8267        0.4434       0.8267        0.4810  0.0169
     77      0.8333        0.4442       0.8333        0.4755  0.0167
     78      0.8267        0.4422       0.8267        0.4799  0.0166
     79      0.8333        0.4501       0.8333        0.4719  0.0175
Stopping since valid_loss has not improved in the last 40 epochs.
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.7933        0.6862       0.7933        0.6379  0.0165
      2      0.8933        0.6151       0.8933        0.4534  0.0164
      3      0.9067        0.5150       0.9067        0.4077  0.0161
      4      0.8800        0.4870       0.8800        0.4324  0.0173
      5      0.8733        0.4577       0.8733        0.4361  0.0174
      6      0.8867        0.4585       0.8867        0.4170  0.0165
      7      0.9067        0.4470       0.9067        0.4073  0.0169
      8      0.8933        0.4534       0.8933        0.4149  0.0174
      9      0.8800        0.4428       0.8800        0.4323  0.0163
     10      0.9067        0.4462       0.9067        0.4009  0.0171
     11      0.9067        0.4409       0.9067        0.3982  0.0155
     12      0.8333        0.4595       0.8333        0.4715  0.0152
     13      0.9067        0.4642       0.9067        0.4001  0.0163
     14      0.9133        0.4584       0.9133        0.3993  0.0158
     15      0.8867        0.4605       0.8867        0.4257  0.0143
     16      0.9067        0.4370       0.9067        0.4002  0.0161
     17      0.8933        0.4340       0.8933        0.4120  0.0173
     18      0.9133        0.4557       0.9133        0.4042  0.0157
     19      0.9067        0.4491       0.9067        0.3988  0.0175
     20      0.8867        0.4302       0.8867        0.4203  0.0168
     21      0.9067        0.4476       0.9067        0.3978  0.0184
     22      0.9067        0.4450       0.9067        0.4042  0.0205
```

```
     23      0.9133        0.4377       0.9133        0.3972  0.0169
     24      0.9000        0.4354       0.9000        0.4020  0.0168
     25      0.9133        0.4374       0.9133        0.3959  0.0167
     26      0.8933        0.4383       0.8933        0.4108  0.0169
     27      0.8933        0.4362       0.8933        0.4065  0.0167
     28      0.8867        0.4343       0.8867        0.4133  0.0167
     29      0.9000        0.4448       0.9000        0.4113  0.0170
     30      0.9067        0.4540       0.9067        0.4051  0.0171
     31      0.9000        0.4390       0.9000        0.4055  0.0187
     32      0.8933        0.4589       0.8933        0.4142  0.0188
     33      0.9000        0.4343       0.9000        0.4147  0.0198
     34      0.8867        0.4495       0.8867        0.4281  0.0197
     35      0.9000        0.4408       0.9000        0.4093  0.0200
     36      0.9200        0.4394       0.9200        0.3961  0.0200
     37      0.9000        0.4409       0.9000        0.4140  0.0192
     38      0.9000        0.4433       0.9000        0.4086  0.0181
     39      0.9000        0.4466       0.9000        0.4125  0.0190
     40      0.9000        0.4371       0.9000        0.4139  0.0169
     41      0.8867        0.4577       0.8867        0.4300  0.0187
     42      0.9133        0.4384       0.9133        0.3999  0.0179
     43      0.9000        0.4510       0.9000        0.4066  0.0178
     44      0.8933        0.4425       0.8933        0.4122  0.0214
     45      0.9133        0.4756       0.9133        0.4007  0.0194
     46      0.9067        0.4398       0.9067        0.4001  0.0193
     47      0.8933        0.4449       0.8933        0.4142  0.0203
     48      0.8933        0.4291       0.8933        0.4046  0.0186
     49      0.9133        0.4373       0.9133        0.4010  0.0168
     50      0.8933        0.4473       0.8933        0.4119  0.0177
     51      0.8933        0.4419       0.8933        0.4143  0.0186
     52      0.9067        0.4412       0.9067        0.4090  0.0172
     53      0.8867        0.4366       0.8867        0.4195  0.0168
     54      0.8867        0.4453       0.8867        0.4204  0.0169
     55      0.8867        0.4404       0.8867        0.4273  0.0175
     56      0.8933        0.4498       0.8933        0.4093  0.0178
     57      0.8933        0.4335       0.8933        0.4152  0.0174
     58      0.8933        0.4425       0.8933        0.4084  0.0182
     59      0.8867        0.4416       0.8867        0.4201  0.0171
     60      0.8933        0.4394       0.8933        0.4206  0.0176
     61      0.8933        0.4728       0.8933        0.4184  0.0166
     62      0.8867        0.4609       0.8867        0.4203  0.0177
     63      0.9133        0.4451       0.9133        0.4021  0.0172
     64      0.8867        0.4414       0.8867        0.4179  0.0167
Stopping since valid_loss has not improved in the last 40 epochs.
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
      1      0.5200        0.6926       0.5200        0.6925  0.0138
      2      0.5200        0.6930       0.5200        0.6925  0.0139
      3      0.5200        0.6912       0.5200        0.6925  0.0146
      4      0.5200        0.6938       0.5200        0.6924  0.0136
```

5	0.5200	0.6909	0.5200	0.6924	0.0137
6	0.5200	0.6929	0.5200	0.6924	0.0146
7	0.5200	0.6925	0.5200	0.6924	0.0141
8	0.5200	0.6921	0.5200	0.6923	0.0139
9	0.5200	0.6928	0.5200	0.6923	0.0138
10	0.5200	0.6942	0.5200	0.6923	0.0137
11	0.5200	0.6941	0.5200	0.6923	0.0138
12	0.5200	0.6930	0.5200	0.6923	0.0140
13	0.5200	0.6911	0.5200	0.6922	0.0136
14	0.5200	0.6925	0.5200	0.6922	0.0120
15	0.5200	0.6924	0.5200	0.6922	0.0120
16	0.5200	0.6939	0.5200	0.6922	0.0136
17	0.5200	0.6918	0.5200	0.6922	0.0121
18	0.5200	0.6961	0.5200	0.6921	0.0129
19	0.5200	0.6939	0.5200	0.6921	0.0123
20	0.5200	0.6938	0.5200	0.6921	0.0142
21	0.5200	0.6923	0.5200	0.6921	0.0119
22	0.5200	0.6914	0.5200	0.6920	0.0126
23	0.5200	0.6930	0.5200	0.6920	0.0137
24	0.5200	0.6913	0.5200	0.6920	0.0133
25	0.5200	0.6936	0.5200	0.6920	0.0129
26	0.5200	0.6926	0.5200	0.6920	0.0121
27	0.5200	0.6928	0.5200	0.6919	0.0129
28	0.5200	0.6906	0.5200	0.6919	0.0120
29	0.5200	0.6934	0.5200	0.6919	0.0130
30	0.5200	0.6925	0.5200	0.6919	0.0145
31	0.5200	0.6914	0.5200	0.6919	0.0136
32	0.5200	0.6916	0.5200	0.6918	0.0139
33	0.5200	0.6943	0.5200	0.6918	0.0140
34	0.5200	0.6933	0.5200	0.6918	0.0138
35	0.5200	0.6926	0.5200	0.6918	0.0146
36	0.5200	0.6928	0.5200	0.6917	0.0149
37	0.5200	0.6921	0.5200	0.6917	0.0143
38	0.5200	0.6913	0.5200	0.6917	0.0139
39	0.5200	0.6921	0.5200	0.6917	0.0139
40	0.5200	0.6898	0.5200	0.6917	0.0130
41	0.5200	0.6937	0.5200	0.6916	0.0142
42	0.5200	0.6914	0.5200	0.6916	0.0162
43	0.5200	0.6932	0.5200	0.6916	0.0139
44	0.5200	0.6931	0.5200	0.6916	0.0140
45	0.5200	0.6930	0.5200	0.6916	0.0166
46	0.5200	0.6913	0.5200	0.6916	0.0191
47	0.5200	0.6916	0.5200	0.6915	0.0149
48	0.5200	0.6919	0.5200	0.6915	0.0144
49	0.5200	0.6928	0.5200	0.6915	0.0150
50	0.5200	0.6918	0.5200	0.6915	0.0147
51	0.5200	0.6918	0.5200	0.6915	0.0147
52	0.5200	0.6927	0.5200	0.6915	0.0146
53	0.5200	0.6923	0.5200	0.6915	0.0145

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
54	0.5200	0.6921	0.5200	0.6914	0.0144
55	0.5200	0.6910	0.5200	0.6914	0.0160
56	0.5200	0.6907	0.5200	0.6914	0.0181
57	0.5200	0.6916	0.5200	0.6914	0.0170
58	0.5200	0.6929	0.5200	0.6913	0.0175
59	0.5200	0.6927	0.5200	0.6913	0.0169
60	0.5200	0.6934	0.5200	0.6913	0.0152
61	0.5200	0.6919	0.5200	0.6913	0.0169
62	0.5200	0.6923	0.5200	0.6913	0.0156
63	0.5200	0.6903	0.5200	0.6912	0.0162
64	0.5200	0.6925	0.5200	0.6912	0.0167
65	0.5200	0.6902	0.5200	0.6912	0.0146
66	0.5200	0.6920	0.5200	0.6912	0.0180
67	0.5200	0.6909	0.5200	0.6911	0.0163
68	0.5200	0.6900	0.5200	0.6911	0.0164
69	0.5200	0.6932	0.5200	0.6911	0.0159
70	0.5200	0.6917	0.5200	0.6911	0.0166
71	0.5200	0.6907	0.5200	0.6910	0.0168
72	0.5200	0.6906	0.5200	0.6910	0.0152
73	0.5200	0.6906	0.5200	0.6910	0.0155
74	0.5200	0.6900	0.5200	0.6910	0.0151
75	0.5200	0.6920	0.5200	0.6910	0.0153
76	0.5200	0.6899	0.5200	0.6910	0.0159
77	0.5200	0.6917	0.5200	0.6909	0.0145
78	0.5200	0.6922	0.5200	0.6909	0.0162
79	0.5200	0.6911	0.5200	0.6909	0.0154
80	0.5200	0.6916	0.5200	0.6908	0.0181
81	0.5200	0.6919	0.5200	0.6908	0.0170
82	0.5200	0.6914	0.5200	0.6908	0.0178
83	0.5200	0.6924	0.5200	0.6908	0.0165
84	0.5200	0.6911	0.5200	0.6907	0.0158
85	0.5200	0.6916	0.5200	0.6907	0.0160
86	0.5200	0.6914	0.5200	0.6907	0.0164
87	0.5200	0.6898	0.5200	0.6907	0.0168
88	0.5200	0.6915	0.5200	0.6906	0.0173
89	0.5200	0.6898	0.5200	0.6906	0.0152
90	0.5200	0.6927	0.5200	0.6906	0.0163
91	0.5200	0.6925	0.5200	0.6906	0.0157
92	0.5200	0.6923	0.5200	0.6905	0.0150
93	0.5200	0.6929	0.5200	0.6905	0.0161
94	0.5200	0.6911	0.5200	0.6905	0.0172
95	0.5200	0.6936	0.5200	0.6905	0.0161
96	0.5200	0.6940	0.5200	0.6905	0.0165
97	0.5200	0.6903	0.5200	0.6904	0.0157
98	0.5200	0.6898	0.5200	0.6904	0.0161
99	0.5200	0.6901	0.5200	0.6904	0.0161
100	0.5200	0.6919	0.5200	0.6903	0.0171

1	0.5200	0.6933	0.5200	0.6934	0.0168
2	0.5200	0.6944	0.5200	0.6934	0.0162
3	0.5200	0.6936	0.5200	0.6933	0.0153
4	0.5200	0.6921	0.5200	0.6933	0.0142
5	0.5200	0.6937	0.5200	0.6933	0.0137
6	0.5200	0.6940	0.5200	0.6932	0.0134
7	0.5200	0.6933	0.5200	0.6932	0.0144
8	0.5200	0.6935	0.5200	0.6932	0.0139
9	0.5200	0.6939	0.5200	0.6932	0.0137
10	0.5200	0.6928	0.5200	0.6931	0.0146
11	0.5200	0.6939	0.5200	0.6931	0.0138
12	0.5200	0.6917	0.5200	0.6931	0.0137
13	0.5200	0.6929	0.5200	0.6931	0.0133
14	0.5200	0.6921	0.5200	0.6930	0.0137
15	0.5200	0.6936	0.5200	0.6930	0.0136
16	0.5200	0.6922	0.5200	0.6930	0.0135
17	0.5200	0.6936	0.5200	0.6929	0.0147
18	0.5200	0.6923	0.5200	0.6929	0.0135
19	0.5200	0.6933	0.5200	0.6929	0.0144
20	0.5200	0.6930	0.5200	0.6929	0.0144
21	0.5200	0.6929	0.5200	0.6928	0.0141
22	0.5200	0.6937	0.5200	0.6928	0.0137
23	0.5200	0.6924	0.5200	0.6928	0.0142
24	0.5200	0.6923	0.5200	0.6927	0.0129
25	0.5200	0.6905	0.5200	0.6927	0.0130
26	0.5200	0.6918	0.5200	0.6927	0.0120
27	0.5200	0.6910	0.5200	0.6927	0.0141
28	0.5200	0.6914	0.5200	0.6926	0.0128
29	0.5200	0.6926	0.5200	0.6926	0.0136
30	0.5200	0.6939	0.5200	0.6926	0.0147
31	0.5200	0.6919	0.5200	0.6925	0.0136
32	0.5200	0.6939	0.5200	0.6925	0.0153
33	0.5200	0.6917	0.5200	0.6925	0.0136
34	0.5200	0.6946	0.5200	0.6924	0.0133
35	0.5200	0.6923	0.5200	0.6924	0.0134
36	0.5200	0.6897	0.5200	0.6924	0.0136
37	0.5200	0.6916	0.5200	0.6924	0.0141
38	0.5200	0.6896	0.5200	0.6923	0.0142
39	0.5200	0.6920	0.5200	0.6923	0.0143
40	0.5200	0.6902	0.5200	0.6923	0.0127
41	0.5200	0.6918	0.5200	0.6922	0.0146
42	0.5200	0.6925	0.5200	0.6922	0.0153
43	0.5200	0.6912	0.5200	0.6921	0.0134
44	0.5200	0.6925	0.5200	0.6921	0.0142
45	0.5200	0.6935	0.5200	0.6921	0.0143
46	0.5200	0.6910	0.5200	0.6921	0.0140
47	0.5200	0.6912	0.5200	0.6920	0.0141
48	0.5200	0.6912	0.5200	0.6920	0.0139
49	0.5200	0.6932	0.5200	0.6920	0.0125

50	0.5200	0.6918	0.5200	0.6919	0.0134
51	0.5200	0.6909	0.5200	0.6919	0.0134
52	0.5200	0.6906	0.5200	0.6918	0.0125
53	0.5200	0.6921	0.5200	0.6918	0.0148
54	0.5200	0.6918	0.5200	0.6918	0.0120
55	0.5200	0.6931	0.5200	0.6917	0.0159
56	0.5200	0.6920	0.5200	0.6917	0.0139
57	0.5200	0.6918	0.5200	0.6916	0.0152
58	0.5200	0.6919	0.5200	0.6916	0.0134
59	0.5200	0.6931	0.5200	0.6916	0.0139
60	0.5200	0.6918	0.5200	0.6915	0.0154
61	0.5200	0.6914	0.5200	0.6915	0.0136
62	0.5200	0.6923	0.5200	0.6915	0.0140
63	0.5200	0.6918	0.5200	0.6914	0.0146
64	0.5200	0.6901	0.5200	0.6914	0.0125
65	0.5200	0.6903	0.5200	0.6914	0.0129
66	0.5200	0.6923	0.5200	0.6913	0.0126
67	0.5200	0.6910	0.5200	0.6913	0.0143
68	0.5200	0.6907	0.5200	0.6912	0.0138
69	0.5200	0.6929	0.5200	0.6912	0.0129
70	0.5200	0.6919	0.5200	0.6912	0.0128
71	0.5200	0.6917	0.5200	0.6911	0.0120
72	0.5200	0.6930	0.5200	0.6911	0.0135
73	0.5200	0.6916	0.5200	0.6911	0.0125
74	0.5200	0.6890	0.5200	0.6910	0.0136
75	0.5200	0.6915	0.5200	0.6910	0.0123
76	0.5200	0.6916	0.5200	0.6910	0.0136
77	0.5200	0.6923	0.5200	0.6909	0.0125
78	0.5200	0.6921	0.5200	0.6909	0.1470
79	0.5200	0.6908	0.5200	0.6908	0.0582
80	0.5200	0.6907	0.5200	0.6908	0.0402
81	0.5200	0.6901	0.5200	0.6908	0.0271
82	0.5200	0.6907	0.5200	0.6907	0.0247
83	0.5200	0.6911	0.5200	0.6907	0.0259
84	0.5200	0.6896	0.5200	0.6906	0.0220
85	0.5200	0.6918	0.5200	0.6906	0.0213
86	0.5200	0.6911	0.5200	0.6905	0.0210
87	0.5200	0.6898	0.5200	0.6905	0.0209
88	0.5200	0.6906	0.5200	0.6905	0.0204
89	0.5200	0.6929	0.5200	0.6904	0.0177
90	0.5200	0.6902	0.5200	0.6904	0.0166
91	0.5200	0.6899	0.5200	0.6903	0.0168
92	0.5200	0.6906	0.5200	0.6903	0.0165
93	0.5200	0.6899	0.5200	0.6903	0.0166
94	0.5200	0.6918	0.5200	0.6902	0.0150
95	0.5200	0.6901	0.5200	0.6902	0.0139
96	0.5200	0.6890	0.5200	0.6901	0.0138
97	0.5200	0.6900	0.5200	0.6901	0.0140
98	0.5200	0.6915	0.5200	0.6900	0.0152

```
    99      0.5200        0.6884       0.5200        0.6900  0.0211
   100      0.5200        0.6906       0.5200        0.6899  0.0153
  epoch    accuracy    train_loss    valid_acc    valid_loss     dur
-------  ----------  ------------  -----------  ------------  ------
     1      0.8800        0.6175       0.8800        0.4382  0.0651
     2      0.8833        0.4808       0.8833        0.4287  0.0640
     3      0.8267        0.4770       0.8267        0.4705  0.0763
     4      0.8533        0.4831       0.8533        0.4605  0.2126
     5      0.8767        0.4702       0.8767        0.4376  0.1017
     6      0.8800        0.4627       0.8800        0.4284  0.1207
     7      0.8767        0.4527       0.8767        0.4309  0.2384
     8      0.8833        0.4705       0.8833        0.4273  0.1697
     9      0.8833        0.4626       0.8833        0.4289  0.1547
    10      0.8833        0.4688       0.8833        0.4293  0.0836
    11      0.8800        0.4557       0.8800        0.4301  0.0631
    12      0.8800        0.4503       0.8800        0.4294  0.0592
    13      0.8667        0.4597       0.8667        0.4389  0.0540
    14      0.8833        0.4561       0.8833        0.4326  0.0528
    15      0.8700        0.4646       0.8700        0.4376  0.0552
    16      0.8700        0.4486       0.8700        0.4367  0.0597
    17      0.8667        0.4346       0.8667        0.4345  0.0552
    18      0.8433        0.4386       0.8433        0.4678  0.0511

    19      0.8767        0.4461       0.8767        0.4296  0.0487
    20      0.8733        0.4501       0.8733        0.4345  0.0531
    21      0.8700        0.4424       0.8700        0.4384  0.0532
    22      0.8800        0.4616       0.8800        0.4324  0.0510
    23      0.8833        0.4430       0.8833        0.4282  0.0477
    24      0.8800        0.4712       0.8800        0.4323  0.0518
    25      0.8767        0.4486       0.8767        0.4335  0.0456
    26      0.8800        0.4580       0.8800        0.4317  0.0478
    27      0.8767        0.4683       0.8767        0.4312  0.0512
    28      0.8300        0.4672       0.8300        0.4798  0.0538
    29      0.8567        0.4840       0.8567        0.4526  0.0536
    30      0.8667        0.4549       0.8667        0.4405  0.0522
    31      0.8600        0.4526       0.8600        0.4453  0.0520
    32      0.8767        0.4546       0.8767        0.4359  0.0605
    33      0.8800        0.4467       0.8800        0.4311  0.0550
    34      0.8800        0.4537       0.8800        0.4313  0.0517
    35      0.8733        0.4450       0.8733        0.4339  0.0518
    36      0.8833        0.4424       0.8833        0.4306  0.0515
    37      0.8600        0.4525       0.8600        0.4485  0.0534
    38      0.8733        0.4518       0.8733        0.4360  0.0549
    39      0.8700        0.4457       0.8700        0.4341  0.0549
    40      0.8800        0.4435       0.8800        0.4299  0.0565
    41      0.8767        0.4410       0.8767        0.4347  0.0525
    42      0.8767        0.4502       0.8767        0.4386  0.0542
    43      0.8767        0.4441       0.8767        0.4355  0.0534
    44      0.8733        0.4413       0.8733        0.4349  0.0454
    45      0.8733        0.4310       0.8733        0.4331  0.0481
```

```
     46      0.8700        0.4465       0.8700        0.4315  0.0509
     47      0.8733        0.4394       0.8733        0.4359  0.0512
Stopping since valid_loss has not improved in the last 40 epochs.
Best parameters found for scaled data:  {'batch_size': 32, 'lr': 0.1,
'max_epochs': 50, 'optimizer': <class 'torch.optim.adam.Adam'>}
Time taken for grid search: 130.20 seconds

#The best paramaeters according to gridsearch learning rate 0.01, max
epochs: 100, adam optimizer and batch size = 64

# Building MLP model based on best parameters received after grid
search

# gridserach best parameters
epochs = 100
input_size = 12
hidden = 100
lr = 0.01
criterion = nn.CrossEntropyLoss
activation = F.relu
num_classes = 2
dropout = 0.3
batch_size = 64
weight_decay = 1e-4
optimizer=optim.Adam

class final_mlp_scaled(nn.Module):
    def __init__(self, input_size=input_size, hidden=hidden,
output_size=num_classes, dropout=dropout, activation=activation):
        super(final_mlp_scaled, self).__init__()
        self.dropout = nn.Dropout(dropout)
        self.activation = activation
        self.fc1 = nn.Linear(input_size, hidden)
        self.fc2 = nn.Linear(hidden, hidden)
        self.fc3 = nn.Linear(hidden, hidden)
        self.output = nn.Linear(hidden, output_size)

    def forward(self, x):
        x = self.activation(self.fc1(x))
        x = self.dropout(x)
        x = self.activation(self.fc2(x))
        x = self.dropout(x)
        x = self.activation(self.fc3(x))
        x = self.dropout(x)
        x = F.softmax(self.output(x), dim=-1)

        return x

# Reproducibility
random.seed(1)
```

```
torch.manual_seed(1)
np.random.seed(1)

# Starting timer
start_time = time.time()

# Final neural network classifier
net_final_train_scaled = NeuralNetClassifier(
    final_mlp_scaled,
    max_epochs=epochs,
    optimizer=optimizer,
    criterion=criterion(),
    optimizer__lr=lr,
    optimizer__weight_decay=weight_decay,
    callbacks=[
        EpochScoring(scoring='accuracy', lower_is_better=False),
        EarlyStopping(monitor='valid_loss', patience=5,
lower_is_better=True)
    ]
)

# Fitting the model
net_final_train_scaled.fit(X_train_scaled, y_train_scaled)

# Stop timer
end_time = time.time()

# Calculate the time taken for training
training_time = end_time - start_time
print(f"Training completed in {training_time:.2f} seconds.")
```

epoch	accuracy	train_loss	valid_acc	valid_loss	dur
1	0.8800	0.6023	0.8800	0.4343	0.0300
2	0.8567	0.4627	0.8567	0.4519	0.0287
3	0.8600	0.4804	0.8600	0.4444	0.0272
4	0.8833	0.4648	0.8833	0.4313	0.0258
5	0.8800	0.4523	0.8800	0.4339	0.0257
6	0.8800	0.4378	0.8800	0.4294	0.0276
7	0.8767	0.4404	0.8767	0.4301	0.0266
8	0.8700	0.4455	0.8700	0.4379	0.0269
9	0.8633	0.4366	0.8633	0.4348	0.0271
10	0.8700	0.4318	0.8700	0.4310	0.0257

```
Stopping since valid_loss has not improved in the last 5 epochs.
Training completed in 0.33 seconds.
```

```
# Saving the final model on pickle to load on testing notebook
with open('final_mlp_model_scaled.pkl', 'wb') as f:
    pickle.dump(net_final_train_scaled, f)
```

```
# Calculating accuracy on validation set
print("Accuracy on validation data")
accuracy_score(y_val_scaled,
net_final_train_scaled.predict(X_val_scaled)) * 100
```

```
Accuracy on validation data
```

```
89.4
```

```
# Confusion matrix for validation set
print("Confusion Matrix:")
print(confusion_matrix(y_val_scaled,
net_final_train_scaled.predict(X_val_scaled)))
```

```
Confusion Matrix:
[[206  34]
 [ 19 241]]
```

```
random.seed(1)
torch.manual_seed(1)
np.random.seed(1)
#Test set
# Use the final trained model to predict on the test data and obtain
accuracy score
print("Accuracy on test data")
accuracy_score(y_test_scaled,
net_final_train_scaled.predict(X_test_scaled)) * 100
```

```
Accuracy on test data
```

```
86.8
```

```
# Get the classification report
print(classification_report(y_test_scaled,net_final_train_scaled.predi
ct(X_test_scaled)))
```

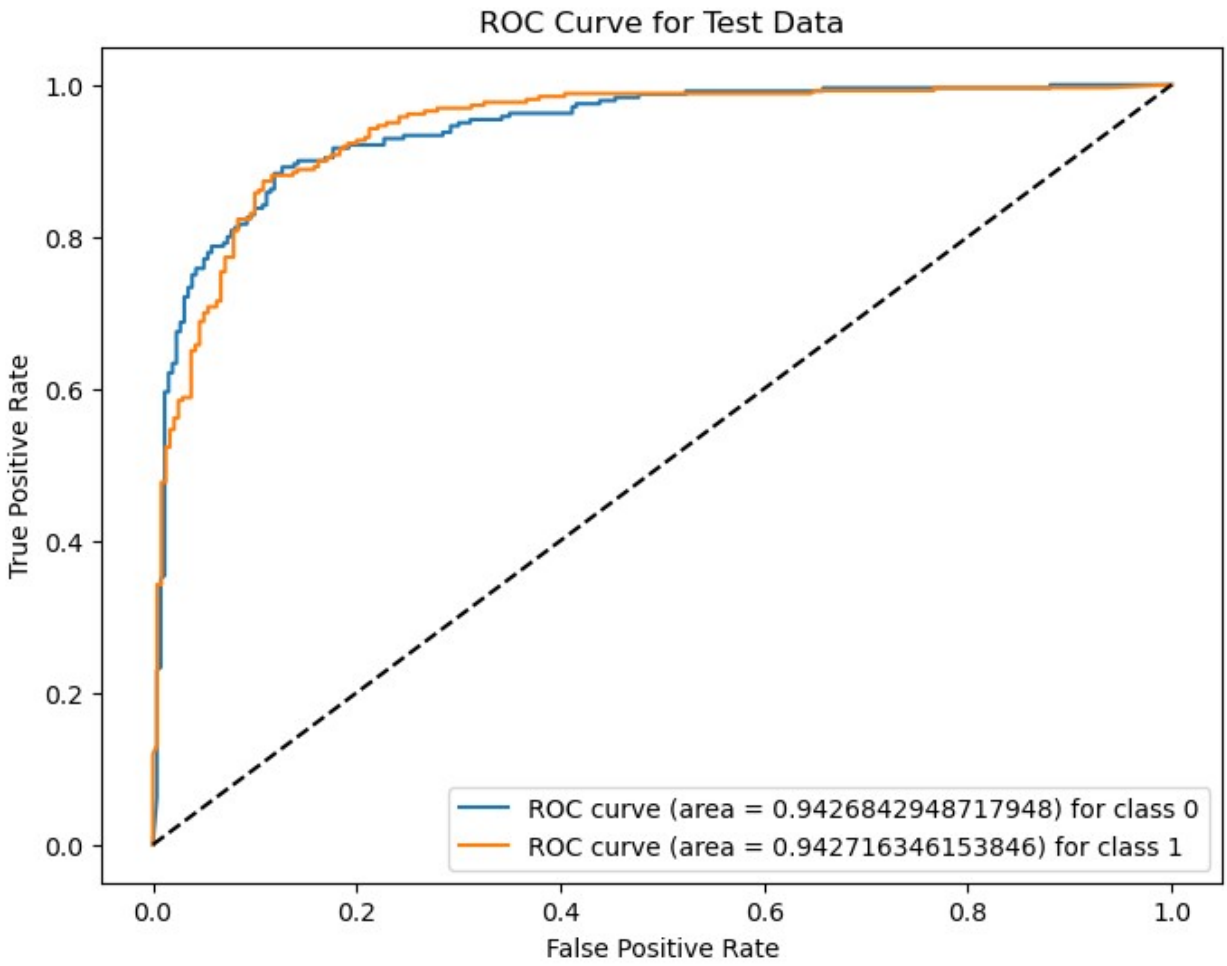
```
              precision    recall  f1-score   support

           0       0.90      0.82      0.86       240
           1       0.84      0.92      0.88       260

    accuracy                           0.87       500
   macro avg       0.87      0.87      0.87       500
weighted avg       0.87      0.87      0.87       500
```

```
#confusion matrix for test set
print("Confusion Matrix:")
```

```
print(confusion_matrix(y_test_scaled,
net_final_train_scaled.predict(X_test_scaled)))
```

```
Confusion Matrix:
[[196  44]
 [ 22 238]]
```

```
#Reference [12]: https://www.projectpro.io/recipes/plot-roc-curve-in-
python
#Plotting ROC curve

# Computing ROC curve and ROC area for each class
y_probs_test = net_final_train_scaled.predict_proba(X_test_scaled)

# Compute ROC curve and ROC area for class 0
false_positive_rate0, true_positive_rate0, _ =
roc_curve(y_test_scaled, y_probs_test[:, 0], pos_label=0)
roc_auc0 = auc(false_positive_rate0, true_positive_rate0)

# Compute ROC curve and ROC area for class 1
false_positive_rate1, true_positive_rate1, _ =
roc_curve(y_test_scaled, y_probs_test[:, 1], pos_label=1)
roc_auc1 = auc(false_positive_rate1, true_positive_rate1)

# Plotting ROC curve
plt.figure(figsize=(8, 6))
plt.plot(false_positive_rate0, true_positive_rate0, label=f'ROC curve
(area = {roc_auc0}) for class 0')
plt.plot(false_positive_rate1, true_positive_rate1, label=f'ROC curve
(area = {roc_auc1}) for class 1')

plt.plot([0, 1], [0, 1], 'k--')

plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve for Test Data')
plt.legend(loc="lower right")
plt.show()
```



Support Vector Machine (SVM)

```
random.seed(1)
torch.manual_seed(1)
np.random.seed(1)

#Reference [13]: https://www.geeksforgeeks.org/classifying-data-using-
support-vector-machinessvms-in-python/
#Base SVM model
#Linear kernel

#Probability= True for estimates for roc curve
classifier = SVC(kernel='linear', probability=True, random_state=10)

# fitting train data
classifier.fit(X_train_scaled, y_train_scaled)


SVC(kernel='linear', probability=True, random_state=10)
```

```
# Predict on the training data
print("Training Accuracy")
accuracy_score(y_train_scaled,classifier.predict(X_train_scaled) )*
100
```

```
Training Accuracy
```

```
87.53333333333333
```

```
# Predict on the validation data
print("Validation Accuracy")
accuracy_score(y_val_scaled,classifier.predict(X_val_scaled)) * 100
```

```
Validation Accuracy
```

```
89.4
```

```
#Reference [14]: https://www.geeksforgeeks.org/svm-hyperparameter-
tuning-using-gridsearchcv-ml/
#SVM gridsearch

# Start timer
start_time = time.time()

#Gridsearch criteria
svm_grid = {'C': [0.0001, 0.1,1.0, 10],
              'kernel': ['linear', 'rbf'],
              'gamma': [0.0001, 0.1],
              'degree': [2, 3, 4]}


#GridSearchCV, cross validate =2 , refitting on the training dataset
grid_search_svm = GridSearchCV(classifier, svm_grid, cv=2,refit=True,
scoring='accuracy', verbose=3)

#GridSearchCV is fit to the data
grid_search_svm.fit(X_train_scaled, y_train_scaled)

#display best parameters
print("Best parameters found: ", grid_search_svm.best_params_)

# Stop timer
end_time = time.time()

# Calculate and display the total time taken
total_time = end_time - start_time
print(f"Total time taken: {total_time} seconds")
```

```
Fitting 2 folds for each of 48 candidates, totalling 96 fits
[CV 1/2] END C=0.0001, degree=2, gamma=0.0001, kernel=linear;,
score=0.520 total time=   0.1s
[CV 2/2] END C=0.0001, degree=2, gamma=0.0001, kernel=linear;,
```

```
score=0.520 total time=   0.1s
[CV 1/2] END C=0.0001, degree=2, gamma=0.0001, kernel=rbf;,
score=0.520 total time=   0.1s
[CV 2/2] END C=0.0001, degree=2, gamma=0.0001, kernel=rbf;,
score=0.520 total time=   0.1s
[CV 1/2] END C=0.0001, degree=2, gamma=0.1, kernel=linear;,
score=0.520 total time=   0.1s
[CV 2/2] END C=0.0001, degree=2, gamma=0.1, kernel=linear;,
score=0.520 total time=   0.1s
[CV 1/2] END C=0.0001, degree=2, gamma=0.1, kernel=rbf;, score=0.520
total time=   0.1s
[CV 2/2] END C=0.0001, degree=2, gamma=0.1, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=0.0001, degree=3, gamma=0.0001, kernel=linear;,
score=0.520 total time=   0.1s
[CV 2/2] END C=0.0001, degree=3, gamma=0.0001, kernel=linear;,
score=0.520 total time=   0.1s
[CV 1/2] END C=0.0001, degree=3, gamma=0.0001, kernel=rbf;,
score=0.520 total time=   0.1s
[CV 2/2] END C=0.0001, degree=3, gamma=0.0001, kernel=rbf;,
score=0.520 total time=   0.1s
[CV 1/2] END C=0.0001, degree=3, gamma=0.1, kernel=linear;,
score=0.520 total time=   0.1s
[CV 2/2] END C=0.0001, degree=3, gamma=0.1, kernel=linear;,
score=0.520 total time=   0.1s
[CV 1/2] END C=0.0001, degree=3, gamma=0.1, kernel=rbf;, score=0.520
total time=   0.1s
[CV 2/2] END C=0.0001, degree=3, gamma=0.1, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=0.0001, degree=4, gamma=0.0001, kernel=linear;,
score=0.520 total time=   0.1s
[CV 2/2] END C=0.0001, degree=4, gamma=0.0001, kernel=linear;,
score=0.520 total time=   0.1s
[CV 1/2] END C=0.0001, degree=4, gamma=0.0001, kernel=rbf;,
score=0.520 total time=   0.1s
[CV 2/2] END C=0.0001, degree=4, gamma=0.0001, kernel=rbf;,
score=0.520 total time=   0.1s
[CV 1/2] END C=0.0001, degree=4, gamma=0.1, kernel=linear;,
score=0.520 total time=   0.1s
[CV 2/2] END C=0.0001, degree=4, gamma=0.1, kernel=linear;,
score=0.520 total time=   0.1s
[CV 1/2] END C=0.0001, degree=4, gamma=0.1, kernel=rbf;, score=0.520
total time=   0.1s
[CV 2/2] END C=0.0001, degree=4, gamma=0.1, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=0.1, degree=2, gamma=0.0001, kernel=linear;,
score=0.879 total time=   0.0s
[CV 2/2] END C=0.1, degree=2, gamma=0.0001, kernel=linear;,
score=0.853 total time=   0.0s
```

```
[CV 1/2] END C=0.1, degree=2, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 2/2] END C=0.1, degree=2, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=0.1, degree=2, gamma=0.1, kernel=linear;, score=0.879
total time=   0.0s
[CV 2/2] END C=0.1, degree=2, gamma=0.1, kernel=linear;, score=0.853
total time=   0.0s
[CV 1/2] END C=0.1, degree=2, gamma=0.1, kernel=rbf;, score=0.863
total time=   0.1s
[CV 2/2] END C=0.1, degree=2, gamma=0.1, kernel=rbf;, score=0.848
total time=   0.1s
[CV 1/2] END C=0.1, degree=3, gamma=0.0001, kernel=linear;,
score=0.879 total time=   0.0s
[CV 2/2] END C=0.1, degree=3, gamma=0.0001, kernel=linear;,
score=0.853 total time=   0.0s
[CV 1/2] END C=0.1, degree=3, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 2/2] END C=0.1, degree=3, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=0.1, degree=3, gamma=0.1, kernel=linear;, score=0.879
total time=   0.0s
[CV 2/2] END C=0.1, degree=3, gamma=0.1, kernel=linear;, score=0.853
total time=   0.0s
[CV 1/2] END C=0.1, degree=3, gamma=0.1, kernel=rbf;, score=0.863
total time=   0.1s
[CV 2/2] END C=0.1, degree=3, gamma=0.1, kernel=rbf;, score=0.848
total time=   0.1s
[CV 1/2] END C=0.1, degree=4, gamma=0.0001, kernel=linear;,
score=0.879 total time=   0.0s
[CV 2/2] END C=0.1, degree=4, gamma=0.0001, kernel=linear;,
score=0.853 total time=   0.0s
[CV 1/2] END C=0.1, degree=4, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 2/2] END C=0.1, degree=4, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=0.1, degree=4, gamma=0.1, kernel=linear;, score=0.879
total time=   0.0s
[CV 2/2] END C=0.1, degree=4, gamma=0.1, kernel=linear;, score=0.853
total time=   0.0s
[CV 1/2] END C=0.1, degree=4, gamma=0.1, kernel=rbf;, score=0.863
total time=   0.1s
[CV 2/2] END C=0.1, degree=4, gamma=0.1, kernel=rbf;, score=0.848
total time=   0.1s
[CV 1/2] END C=1.0, degree=2, gamma=0.0001, kernel=linear;,
score=0.883 total time=   0.0s
[CV 2/2] END C=1.0, degree=2, gamma=0.0001, kernel=linear;,
score=0.865 total time=   0.0s
[CV 1/2] END C=1.0, degree=2, gamma=0.0001, kernel=rbf;, score=0.520
```

```
total time=   0.1s
[CV 2/2] END C=1.0, degree=2, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=1.0, degree=2, gamma=0.1, kernel=linear;, score=0.883
total time=   0.0s
[CV 2/2] END C=1.0, degree=2, gamma=0.1, kernel=linear;, score=0.865
total time=   0.0s
[CV 1/2] END C=1.0, degree=2, gamma=0.1, kernel=rbf;, score=0.880
total time=   0.1s
[CV 2/2] END C=1.0, degree=2, gamma=0.1, kernel=rbf;, score=0.867
total time=   0.1s
[CV 1/2] END C=1.0, degree=3, gamma=0.0001, kernel=linear;,
score=0.883 total time=   0.0s
[CV 2/2] END C=1.0, degree=3, gamma=0.0001, kernel=linear;,
score=0.865 total time=   0.0s
[CV 1/2] END C=1.0, degree=3, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 2/2] END C=1.0, degree=3, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=1.0, degree=3, gamma=0.1, kernel=linear;, score=0.883
total time=   0.0s
[CV 2/2] END C=1.0, degree=3, gamma=0.1, kernel=linear;, score=0.865
total time=   0.0s
[CV 1/2] END C=1.0, degree=3, gamma=0.1, kernel=rbf;, score=0.880
total time=   0.1s
[CV 2/2] END C=1.0, degree=3, gamma=0.1, kernel=rbf;, score=0.867
total time=   0.1s
[CV 1/2] END C=1.0, degree=4, gamma=0.0001, kernel=linear;,
score=0.883 total time=   0.0s
[CV 2/2] END C=1.0, degree=4, gamma=0.0001, kernel=linear;,
score=0.865 total time=   0.0s
[CV 1/2] END C=1.0, degree=4, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 2/2] END C=1.0, degree=4, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=1.0, degree=4, gamma=0.1, kernel=linear;, score=0.883
total time=   0.0s
[CV 2/2] END C=1.0, degree=4, gamma=0.1, kernel=linear;, score=0.865
total time=   0.0s
[CV 1/2] END C=1.0, degree=4, gamma=0.1, kernel=rbf;, score=0.880
total time=   0.1s
[CV 2/2] END C=1.0, degree=4, gamma=0.1, kernel=rbf;, score=0.867
total time=   0.1s
[CV 1/2] END C=10, degree=2, gamma=0.0001, kernel=linear;, score=0.887
total time=   0.0s
[CV 2/2] END C=10, degree=2, gamma=0.0001, kernel=linear;, score=0.869
total time=   0.0s
[CV 1/2] END C=10, degree=2, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
```

```
[CV 2/2] END C=10, degree=2, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=10, degree=2, gamma=0.1, kernel=linear;, score=0.887
total time=   0.0s
[CV 2/2] END C=10, degree=2, gamma=0.1, kernel=linear;, score=0.869
total time=   0.0s
[CV 1/2] END C=10, degree=2, gamma=0.1, kernel=rbf;, score=0.887 total
time=   0.1s
[CV 2/2] END C=10, degree=2, gamma=0.1, kernel=rbf;, score=0.875 total
time=   0.0s
[CV 1/2] END C=10, degree=3, gamma=0.0001, kernel=linear;, score=0.887
total time=   0.0s
[CV 2/2] END C=10, degree=3, gamma=0.0001, kernel=linear;, score=0.869
total time=   0.0s
[CV 1/2] END C=10, degree=3, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 2/2] END C=10, degree=3, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=10, degree=3, gamma=0.1, kernel=linear;, score=0.887
total time=   0.0s
[CV 2/2] END C=10, degree=3, gamma=0.1, kernel=linear;, score=0.869
total time=   0.0s
[CV 1/2] END C=10, degree=3, gamma=0.1, kernel=rbf;, score=0.887 total
time=   0.1s
[CV 2/2] END C=10, degree=3, gamma=0.1, kernel=rbf;, score=0.875 total
time=   0.1s
[CV 1/2] END C=10, degree=4, gamma=0.0001, kernel=linear;, score=0.887
total time=   0.0s
[CV 2/2] END C=10, degree=4, gamma=0.0001, kernel=linear;, score=0.869
total time=   0.0s
[CV 1/2] END C=10, degree=4, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 2/2] END C=10, degree=4, gamma=0.0001, kernel=rbf;, score=0.520
total time=   0.1s
[CV 1/2] END C=10, degree=4, gamma=0.1, kernel=linear;, score=0.887
total time=   0.0s

[CV 2/2] END C=10, degree=4, gamma=0.1, kernel=linear;, score=0.869
total time=   0.0s
[CV 1/2] END C=10, degree=4, gamma=0.1, kernel=rbf;, score=0.887 total
time=   0.1s
[CV 2/2] END C=10, degree=4, gamma=0.1, kernel=rbf;, score=0.875 total
time=   0.1s
Best parameters found:  {'C': 10, 'degree': 2, 'gamma': 0.1, 'kernel':
'rbf'}
Total time taken: 7.746504068374634 seconds

random.seed(1)
torch.manual_seed(1)
np.random.seed(1)
```

```
#Creating the final SVM model with best hyper parameters
# new model with the best parameters
final_classifier = SVC(C=10, gamma=0.1, kernel='rbf')
start_time = time.time()
# Fit the model on the training data
final_classifier.fit(X_train_scaled, y_train_scaled)


end_time = time.time()

total_time = end_time - start_time
print(f"Total time taken: {total_time} seconds")
```

```
Total time taken: 0.02767801284790039 seconds
```

```
# Save the trained model to a file
with open('final_svm_model.pkl', 'wb') as file:
    pickle.dump(final_classifier, file)

# Predict on the training data
print("Training Accuracy")
accuracy_score(y_train_scaled,
final_classifier.predict(X_train_scaled)) * 100
```

```
Training Accuracy
```

```
88.8
```

```
# Predict on the validation data
print("Validation Accuracy")
accuracy_score(y_val_scaled,final_classifier.predict(X_val_scaled) )*
100
```

```
Validation Accuracy
```

```
89.60000000000001
```

```
# Predict on the test data
print("Test Accuracy")
accuracy_score(y_test_scaled,final_classifier.predict(X_test_scaled) )
* 100
```

```
Test Accuracy
```

```
87.2
```

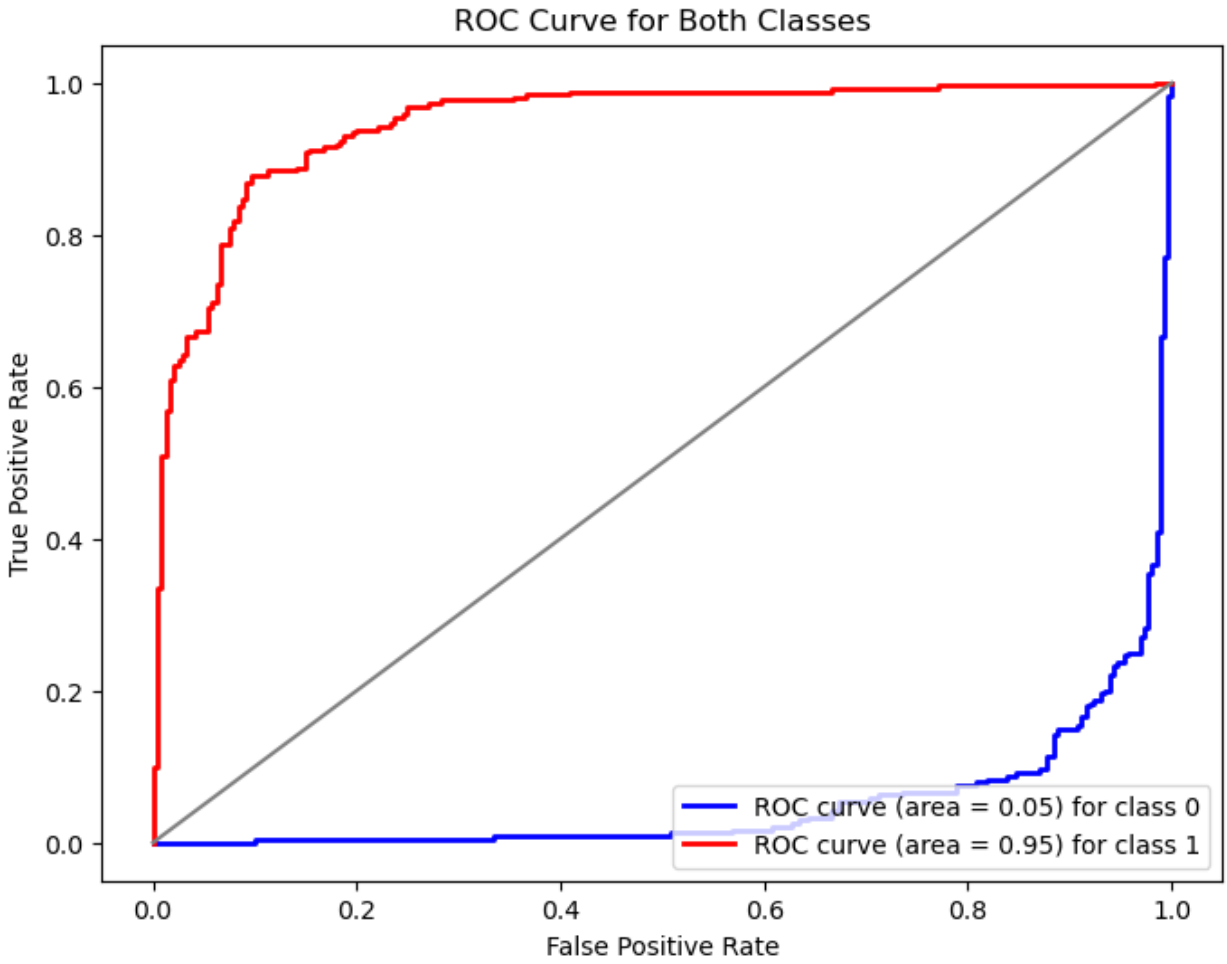
```
print("Classification Report for Test Data:")
print(classification_report(y_test_scaled,
final_classifier.predict(X_test_scaled)))
```

```
Classification Report for Test Data:
              precision    recall  f1-score   support

           0       0.90      0.82      0.86       240
           1       0.85      0.92      0.88       260

    accuracy                           0.87       500
   macro avg       0.88      0.87      0.87       500
weighted avg       0.87      0.87      0.87       500
```

```
# Compute the confusion matrix
confusion_matrix(y_test_scaled,final_classifier.predict(X_test_scaled)
)
```

```
array([[198,  42],
       [ 22, 238]])
```

```
#ROC CURVE for svm
# Calculate decision function for test data
y_scores = final_classifier.decision_function(X_test_scaled)

# Compute ROC curve and ROC area for each class
fpr0, tpr0, _ = roc_curve(y_test_scaled, y_scores, pos_label=0)
roc_auc0 = auc(fpr0, tpr0)

fpr1, tpr1, _ = roc_curve(y_test_scaled, y_scores, pos_label=1)
roc_auc1 = auc(fpr1, tpr1)

# Plotting ROC curve
plt.figure(figsize=(8, 6))
plt.plot(fpr0, tpr0, color='blue', lw=2, label=f'ROC curve (area =
{roc_auc0:.2f}) for class 0')
plt.plot(fpr1, tpr1, color='red', lw=2, label=f'ROC curve (area =
{roc_auc1:.2f}) for class 1')
plt.plot([0, 1], [0, 1], color='gray')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve for Both Classes')
plt.legend(loc="lower right")
plt.show()
```



References

[1] "Dataframe to CSV – How to Save Pandas Dataframes by Exporting," freeCodeCamp.org, Mar. 24, 2023. https://www.freecodecamp.org/news/dataframe-to-csv-how-to-save-pandas-dataframes-by-exporting/

[2] "Remap values in pandas column with a dict, None if KeyError," Stack Overflow. https://stackoverflow.com/questions/73371405/remap-values-in-pandas-column-with-a-dict-none-if-keyerror (accessed Apr. 19, 2024).

[3] "Neural-Computing-Project/project.ipynb at main · vighnesh32/Neural-Computing-Project," GitHub. https://github.com/vighnesh32/Neural-Computing-Project/blob/main/project.ipynb (accessed Apr. 19, 2024).

[4] "Chart Visualization — pandas 1.3.3 documentation," pandas.pydata.org. https://pandas.pydata.org/pandas-docs/stable/user_guide/visualization.html

[5] "Violinplot using Seaborn in Python," GeeksforGeeks, Jun. 24, 2020. https://www.geeksforgeeks.org/violinplot-using-seaborn-in-python/

[6] "Python | Pandas dataframe.skew()," GeeksforGeeks, Nov. 22, 2018. https://www.geeksforgeeks.org/python-pandas-dataframe-skew/

[7] Tutorial Lab (7)

[8] Tutorial lab (4)

[9] scikit-learn, “sklearn.preprocessing.MinMaxScaler — scikit-learn 0.22.1 documentation,” Scikit-learn.org, 2019. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html

[10] “How to Use Python Pickle [+Examples],” blog.hubspot.com, Oct. 07, 2023. https://blog.hubspot.com/website/python-pickle#:~:text=To%20use%20pickle%20in%20Python (accessed Apr. 19, 2024).

[11] P. Training, “GridSearchCV with Scikit-Learn and Python,” Pierian Training, Jun. 23, 2023. https://pieriantraining.com/gridsearchcv-with-scikit-learn-and-python/ (accessed Apr. 19, 2024).

[12] “How to plot a ROC Curve in Python? -,” ProjectPro. https://www.projectpro.io/recipes/plot-roc-curve-in-python

[13] “Classifying data using Support Vector Machines(SVMs) in Python - GeeksforGeeks,” GeeksforGeeks, Apr. 30, 2017. https://www.geeksforgeeks.org/classifying-data-using-support-vector-machinessvms-in-python/

[14] “SVM Hyperparameter Tuning using GridSearchCV | ML,” GeeksforGeeks, Jul. 05, 2019. https://www.geeksforgeeks.org/svm-hyperparameter-tuning-using-gridsearchcv-ml/

[15] Zach, “How to Create a Precision-Recall Curve in Python,” Statology, Sep. 09, 2021. https://www.statology.org/precision-recall-curve-python/