Jake Zhu

MATH 6373 Final Exam Part 1: Application of Convolution Neural Network on Fonts Dataset for Classification

1. Data Set Presentation

The set of data, Character Font Images Dataset, is obtained from University of California Irvine's machine learning repository and may be found in the following link:

https://archive.ics.uci.edu/ml/datasets/Character+Font+Images

We will select 8 fonts among the 50 fonts available, each font considered as a class for prediction.

The image height and width in this sample are always 20x20, 400 columns grayscale pixels representing images of characters and numbers.

Furthermore, we will combine the 8 fonts csv files together and create a "true class" column based on the given font for classification.

The input variables (based on computer vision):

- CL1 – Castellar: size(CL1) = 1056
- CL2 – Rage: size(CL2) = 976
- CL3 – French: size(CL3) = 984
- CL4 – English: size(CL4) = 968
- CL5 – Gloucester: size(CL5) = 956
- CL6 – Brush: size(CL6) = 956
- CL7 – Matura: size(CL7) = 956
- CL8 – Californian: size(CL8) = 1004

There is no categorical variable and the output variable in which we will attempt to classify would be the class of fonts.

Some potential use cases include:

- Optical character recognition: The dataset can be used to train machine learning models for recognizing and transcribing handwritten text into digital format.
- Handwriting analysis: The dataset can be used to develop algorithms that can identify individual handwriting styles and distinguish between different writers.
- Signature verification: The dataset can be used to train models that can identify and verify signatures on documents.
- Linguistics research: The dataset can be used to study the characteristics of different alphabets and scripts, and to investigate the similarities and differences between handwriting styles across different languages and cultures.
- Education: The dataset can be used to develop educational tools and resources for teaching handwriting and handwriting recognition to students.

Data Preparation:

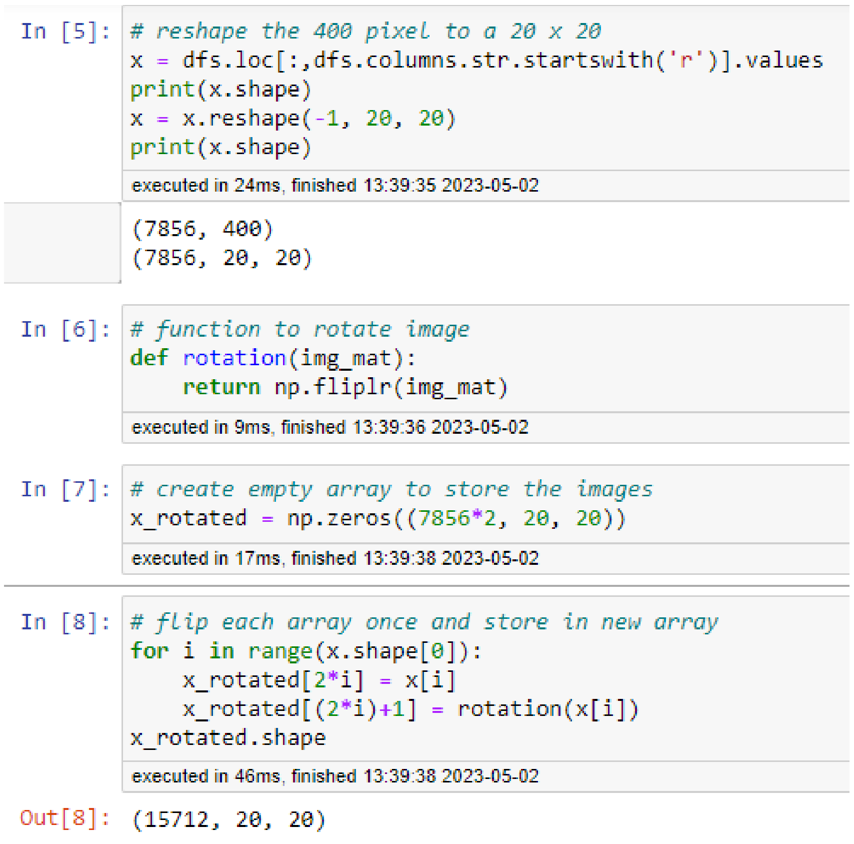
In total, there are about 7856 cases in this dataset. All classes of chosen datasets are fairly balanced as observed in our original proposal. But we want to increase the number of cases per class in order to have a more robust neural network. We will implement horizontal flipping, in which we will flip the images horizontally to increase the number of cases by a multiple of 2 to each class, while also maintaining the original pixelated data.

The new dataset with horizontally flipped images will contain 15712 total cases:

- CL1 – Castellar:
  size(CL1) = 2112
- CL2 – Rage:
  size(CL2) = 1952
- CL3 – French:
  size(CL3) = 1968
- CL4 – English:
  size(CL4) = 1936
- CL5 – Gloucester:
  size(CL5) = 1912
- CL6 – Brush:
  size(CL6) = 1912
- CL7 – Matura:
  size(CL7) = 1912
- CL8 – Californian:
  size(CL8) = 2008

Each font case from our dataset is represented by a row of 400 pixels, we then can reshape the 400 pixels from dimensions of (15712 x 400) into (15712 x 20 x 20), thereby representing a grid of 20 x 20 for imaging. A single case of the font is represented by (1 x 20 x 20), where we will reserve 90% of total cases for training and 10% for testing.

Below is the code that was used to reshape the 400 pixels into a 20 x 20 format, then flip horizontally.



```
In [5]: # reshape the 400 pixel to a 20 x 20
        x = dfs.loc[:,dfs.columns.str.startswith('r')].values
        print(x.shape)
        x = x.reshape(-1, 20, 20)
        print(x.shape)
        executed in 24ms, finished 13:39:35 2023-05-02

        (7856, 400)
        (7856, 20, 20)

In [6]: # function to rotate image
        def rotation(img_mat):
            return np.fliplr(img_mat)
        executed in 9ms, finished 13:39:36 2023-05-02

In [7]: # create empty array to store the images
        x_rotated = np.zeros((7856*2, 20, 20))
        executed in 17ms, finished 13:39:38 2023-05-02

In [8]: # flip each array once and store in new array
        for i in range(x.shape[0]):
            x_rotated[2*i] = x[i]
            x_rotated[(2*i)+1] = rotation(x[i])
        x_rotated.shape
        executed in 46ms, finished 13:39:38 2023-05-02

Out[8]: (15712, 20, 20)
```

Below is an example of the fonts and their respective horizontal flipping applied.

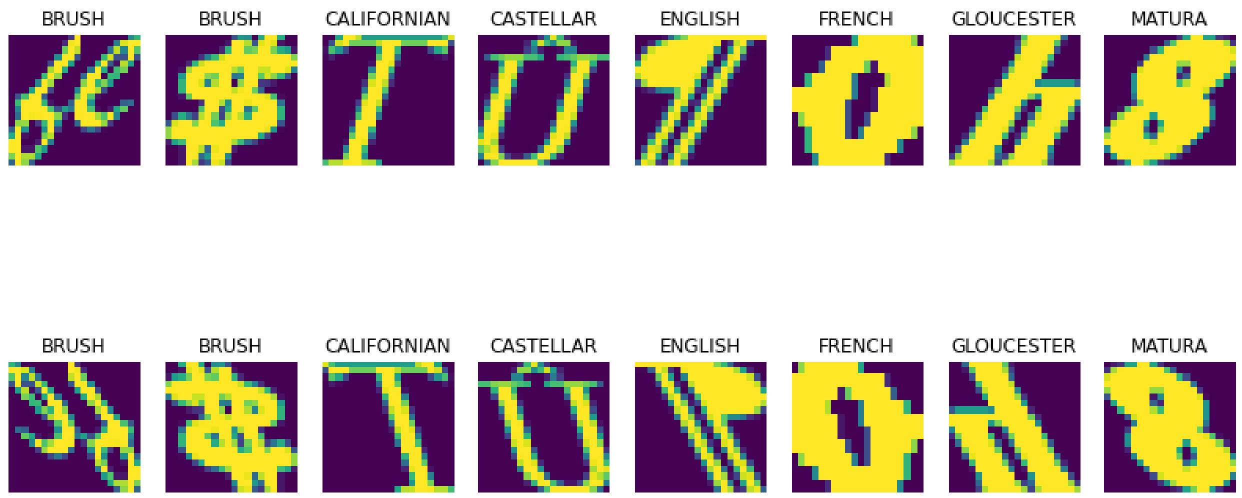


Figure 1: Fonts with their respective horizontal flipping applied below

2. CNN Architecture & First Automatic Training

Our CNN model will feature the following architecture:

Input → Convolution #1 → Maxpool #1 → Convolution #2 → Maxpool #2 → flattening layer (F) → hidden layer (H) → pre-output → softmax → probability output

We will attempt the following sensitivity changes on our parameters:

1. Number of channels in convolution layer 1, CL1
   a. Varying numbers: 8, 16, 24
2. Kernel size of convolution layer 1, KL1
   a. Varying window sizes: 3x3, 5x5
3. Number of channels in convolution layer 2, CL2
   a. Same number of channels as first convolutional layer
4. Kernel size of convolution layer 2, KL2
   a. Varying window sizes: 2x2, 3x3

We will try h = dim(F)/2 for our first run of automatic training, F = dimension of our flattened layer; note this can be improved later on after our first run-through.

All max pool layers will be kept at stride=2 and window 2x2 with no padding.

Our output dimensions for each layer can be computed with the following formulas:

Width_out = (Width_in – Kernel size + 2*padding size) / (stride size) + 1

Height_out = (Height_in – Kernel size + 2*padding size) / (stride size) + 1

Dimensions of our varying CNN models:

Case Model	CL1	KL1	Output Dim1	Maxpool1	Output Dim2	CL2	KL2	Output Dim3	Maxpool2	Output Dim4	Flattened	H = F/2	Output Layer
Model_8_3_16_2	8	3x3	18x18x8	2x2	9x9x8	16	2x2	8x8x16	2x2	4x4x16	256	128	8
Model_8_3_16_3	8	3x3	18x18x8	2x2	9x9x8	16	3x3	7x7x16	2x2	3x3x16	144	72	8
Model_8_3_8_3	8	3x3	18x18x8	2x2	9x9x8	8	3x3	7x7x8	2x2	3x3x8	72	36	8
Model_8_3_8_2	8	3x3	18x18x8	2x2	9x9x8	8	2x2	8x8x8	2x2	4x4x8	128	64	8
Model_8_5_24_2	8	5x5	16x16x8	2x2	8x8x8	24	2x2	7x7x24	2x2	3x3x24	216	108	8
Model_8_5_24_3	8	5x5	16x16x8	2x2	8x8x8	24	3x3	6x6x24	2x2	3x3x24	216	108	8
Model_16_5_32_2	16	5x5	16x16x16	2x2	8x8x16	32	2x2	7x7x32	2x2	3x3x32	288	144	8
Model_16_5_32_3	16	5x5	16x16x16	2x2	8x8x16	32	3x3	6x6x32	2x2	3x3x32	288	144	8
Model_24_5_48_2	24	5x5	16x16x24	2x2	8x8x24	48	2x2	7x7x58	2x2	3x3x48	432	216	8
Model_24_5_48_3	24	5x5	16x16x24	2x2	8x8x24	48	3x3	6x6x48	2x2	3x3x48	432	216	8
Model_16_3_16_2	16	3x3	18x18x16	2x2	9x9x16	16	2x2	8x8x16	2x2	4x4x16	256	128	8
Model_16_5_16_3	16	5x5	16x16x16	2x2	8x8x16	16	3x3	6x6x16	2x2	3x3x16	144	72	8

Example for the naming convention of our models: Model_8_3_16_2

- 8 channels in convolution layer 1
- 3x3 kernel size in convolution layer 1
- 16 channels in convolution layer 2
- 2x2 kernel size in convolution layer 2

From our table column naming convention:

- Case model: models with different cases
- CL1: channels of convolution layer 1
- KL1: kernel size of convolution layer 1
- Output Dim1: dimensions of convolution layer 1
- Maxpool1: max pool of stride 2 applied to convolution layer 1
- Output Dim2: dimensions of max pool 1
- CL2: channels of convolution layer 2
- KL2: kernel size of convolution layer 2
- Output Dim3: dimensions of convolution layer 2
- Maxpool2: max pool of stride 2 applied to convolution layer 2
- Output Dim4: dimensions of max pool 2
- Flattened: dimension of flattened max pool 2 output
- H = hidden layer dimensions after flattening
- Output layer: size of predicted class

<u>Selection of Best Epoch criterion:</u>

We can select the best epoch for individual models by observing the loss and accuracy during training. We should look for an epoch where the loss decreasing stabilizes, no longer decreasing for each epoch thereafter. The epoch should also be where accuracy of test set < accuracy of train set but high enough to be considered a good result. Each "best epoch" will be shown on the respective training curves.

<u>Learning Algorithm Parameters:</u>

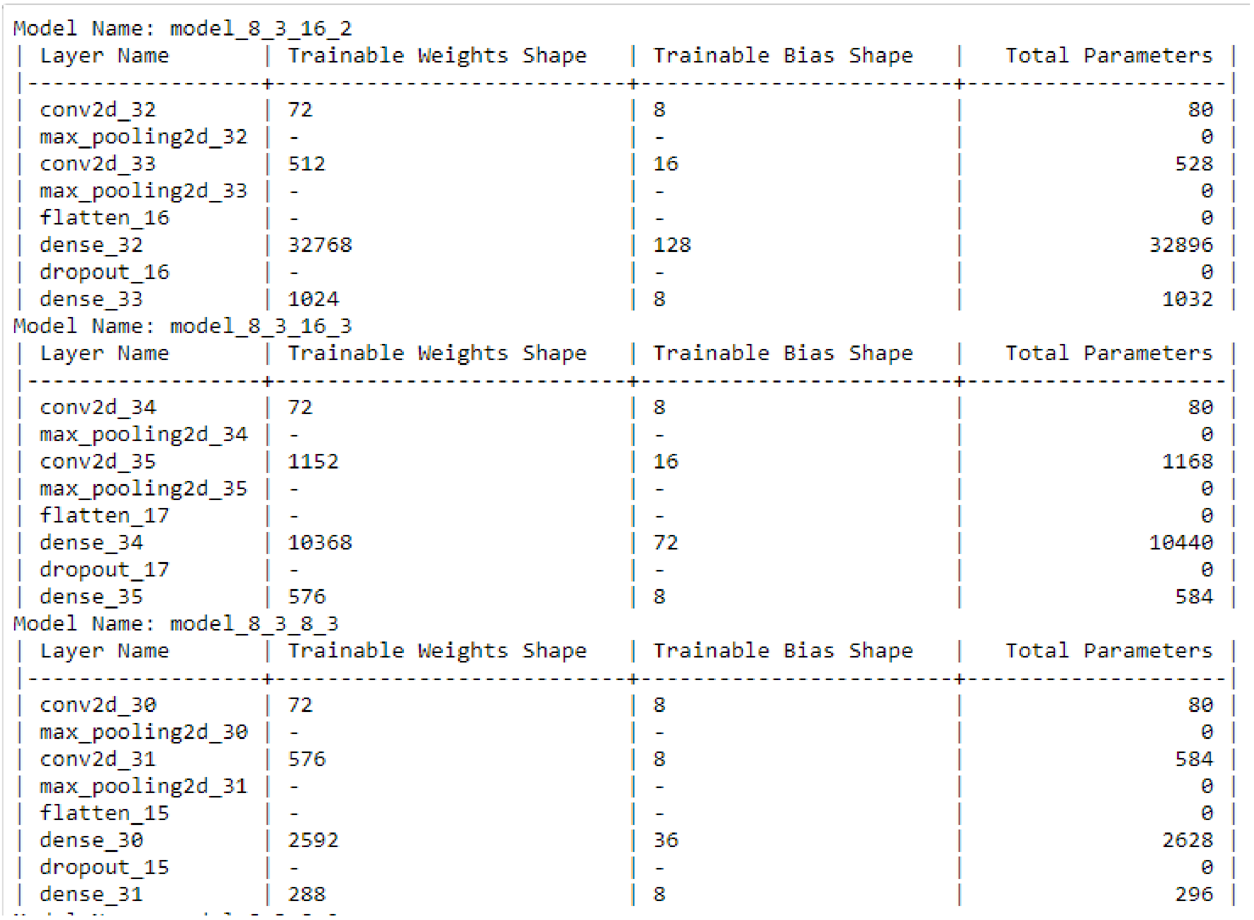
- Batch size: 100; recommended between [80, sqrt(trainset size)=118]
- Optimizer: categorical cross entropy for multi-class classification tasks
- Learning rate: 0.003, epoch: 150
- Dropout: dropout layer with value of 0.2 to increase robustness
- Activation function: RELU used in convolution and hidden layers
- Kernel Initializer: glorot uniform

Parsimony Ratio:

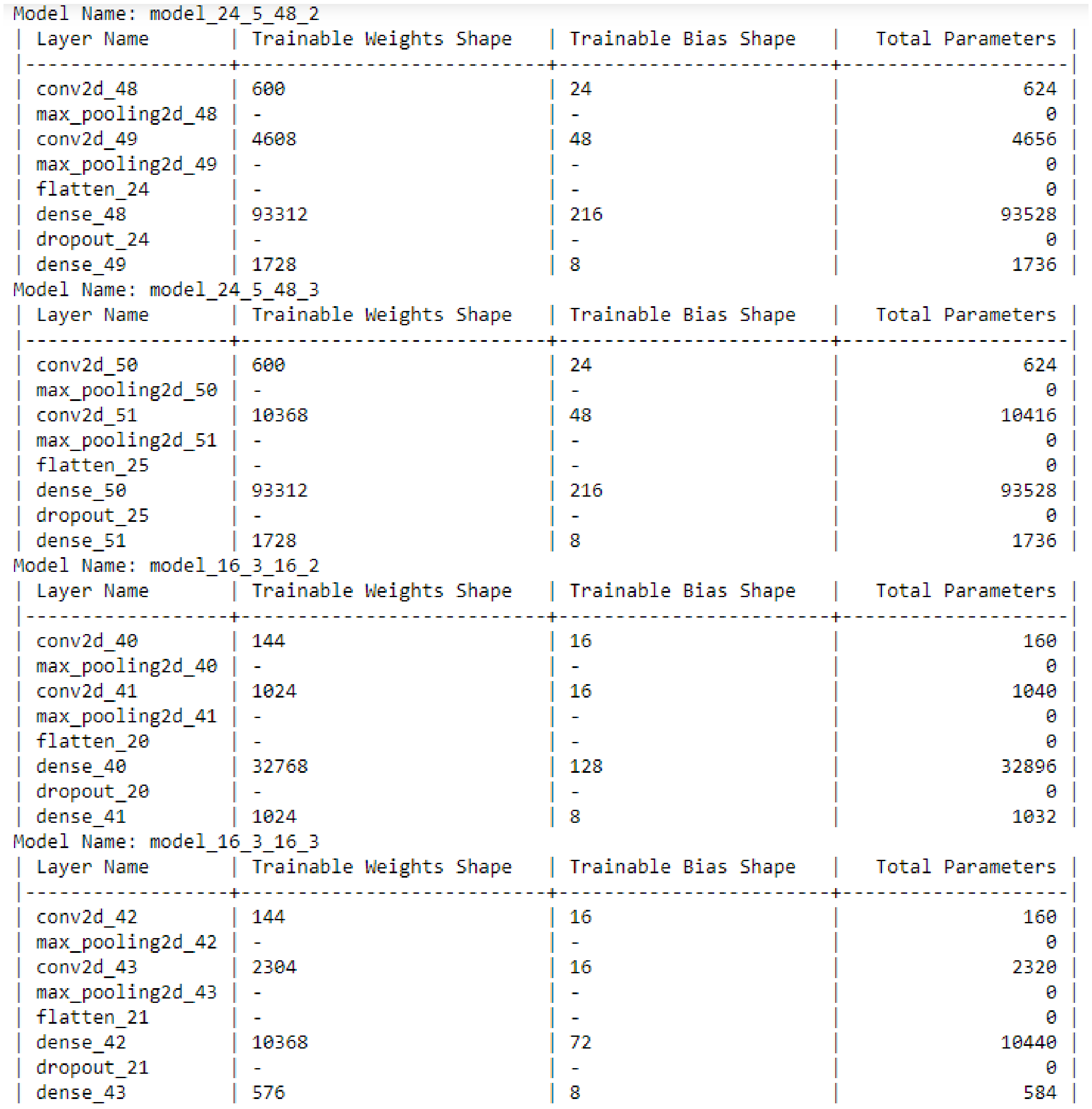
Number of information = number of classes to predict * training set cases = 8 * 14140 = 113120

Model	# of Parameters	# of Information	Parsimony Ratio
Model_8_3_16_2	34536	113120	3.28
Model_8_3_16_3	12272	113120	9.22
Model_8_3_8_3	3588	113120	31.53
Model_8_3_8_2	9120	113120	12.40
Model_8_5_24_2	25308	113120	4.47
Model_8_5_24_3	26268	113120	4.31
Model_16_5_32_2	45272	113120	2.50
Model_16_5_32_3	47832	113120	2.36
Model_24_5_48_2	100544	113120	1.13
Model_24_5_48_3	106304	113120	1.06
Model_16_3_16_2	35128	113120	3.22
Model_16_5_16_3	13504	113120	8.38

Weights and Thresholds of CNN:

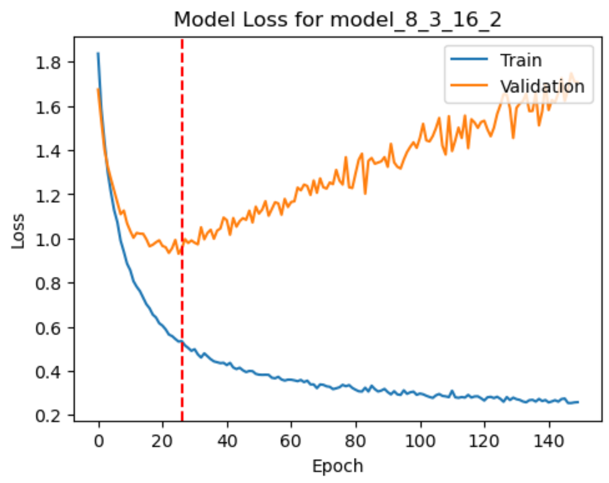


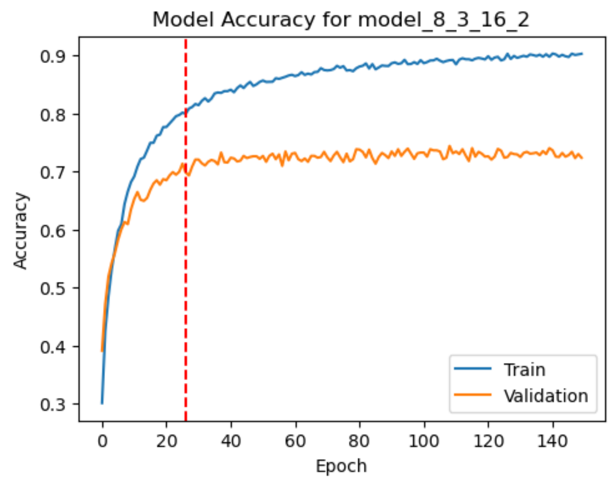
```
Model Name: model_8_3_16_2
 | Layer Name         | Trainable Weights Shape     | Trainable Bias Shape     |   Total Parameters |
 |--------------------+-----------------------------+--------------------------+--------------------|
 | conv2d_32          | 72                          | 8                        |                 80 |
 | max_pooling2d_32   | -                           | -                        |                  0 |
 | conv2d_33          | 512                         | 16                       |                528 |
 | max_pooling2d_33   | -                           | -                        |                  0 |
 | flatten_16         | -                           | -                        |                  0 |
 | dense_32           | 32768                       | 128                      |              32896 |
 | dropout_16         | -                           | -                        |                  0 |
 | dense_33           | 1024                        | 8                        |               1032 |
Model Name: model_8_3_16_3
 | Layer Name         | Trainable Weights Shape     | Trainable Bias Shape     |   Total Parameters |
 |--------------------+-----------------------------+--------------------------+--------------------|
 | conv2d_34          | 72                          | 8                        |                 80 |
 | max_pooling2d_34   | -                           | -                        |                  0 |
 | conv2d_35          | 1152                        | 16                       |               1168 |
 | max_pooling2d_35   | -                           | -                        |                  0 |
 | flatten_17         | -                           | -                        |                  0 |
 | dense_34           | 10368                       | 72                       |              10440 |
 | dropout_17         | -                           | -                        |                  0 |
 | dense_35           | 576                         | 8                        |                584 |
Model Name: model_8_3_8_3
 | Layer Name         | Trainable Weights Shape     | Trainable Bias Shape     |   Total Parameters |
 |--------------------+-----------------------------+--------------------------+--------------------|
 | conv2d_30          | 72                          | 8                        |                 80 |
 | max_pooling2d_30   | -                           | -                        |                  0 |
 | conv2d_31          | 576                         | 8                        |                584 |
 | max_pooling2d_31   | -                           | -                        |                  0 |
 | flatten_15         | -                           | -                        |                  0 |
 | dense_30           | 2592                        | 36                       |               2628 |
 | dropout_15         | -                           | -                        |                  0 |
 | dense_31           | 288                         | 8                        |                296 |
Model Name: model_8_3_8_2
| Layer Name         | Trainable Weights Shape    | Trainable Bias Shape     |     Total Parameters |
|--------------------+----------------------------+--------------------------+----------------------|
| conv2d_28          | 72                         | 8                        |                   80 |
| max_pooling2d_28   | -                          | -                        |                    0 |
| conv2d_29          | 256                        | 8                        |                  264 |
| max_pooling2d_29   | -                          | -                        |                    0 |
| flatten_14         | -                          | -                        |                    0 |
| dense_28           | 8192                       | 64                       |                 8256 |
| dropout_14         | -                          | -                        |                    0 |
| dense_29           | 512                        | 8                        |                  520 |
Model Name: model_8_5_24_2
| Layer Name         | Trainable Weights Shape    | Trainable Bias Shape     |     Total Parameters |
|--------------------+----------------------------+--------------------------+----------------------|
| conv2d_36          | 200                        | 8                        |                  208 |
| max_pooling2d_36   | -                          | -                        |                    0 |
| conv2d_37          | 768                        | 24                       |                  792 |
| max_pooling2d_37   | -                          | -                        |                    0 |
| flatten_18         | -                          | -                        |                    0 |
| dense_36           | 23328                      | 108                      |                23436 |
| dropout_18         | -                          | -                        |                    0 |
| dense_37           | 864                        | 8                        |                  872 |
Model Name: model_8_5_24_3
| Layer Name         | Trainable Weights Shape    | Trainable Bias Shape     |     Total Parameters |
|--------------------+----------------------------+--------------------------+----------------------|
| conv2d_38          | 200                        | 8                        |                  208 |
| max_pooling2d_38   | -                          | -                        |                    0 |
| conv2d_39          | 1728                       | 24                       |                 1752 |
| max_pooling2d_39   | -                          | -                        |                    0 |
| flatten_19         | -                          | -                        |                    0 |
| dense_38           | 23328                      | 108                      |                23436 |
| dropout_19         | -                          | -                        |                    0 |
| dense_39           | 864                        | 8                        |                  872 |
Model Name: model_16_5_32_2
| Layer Name         | Trainable Weights Shape    | Trainable Bias Shape     |     Total Parameters |
|--------------------+----------------------------+--------------------------+----------------------|
| conv2d_44          | 400                        | 16                       |                  416 |
| max_pooling2d_44   | -                          | -                        |                    0 |
| conv2d_45          | 2048                       | 32                       |                 2080 |
| max_pooling2d_45   | -                          | -                        |                    0 |
| flatten_22         | -                          | -                        |                    0 |
| dense_44           | 41472                      | 144                      |                41616 |
| dropout_22         | -                          | -                        |                    0 |
| dense_45           | 1152                       | 8                        |                 1160 |
Model Name: model_16_5_32_3
| Layer Name         | Trainable Weights Shape    | Trainable Bias Shape     |     Total Parameters |
|--------------------+----------------------------+--------------------------+----------------------|
| conv2d_46          | 400                        | 16                       |                  416 |
| max_pooling2d_46   | -                          | -                        |                    0 |
| conv2d_47          | 4608                       | 32                       |                 4640 |
| max_pooling2d_47   | -                          | -                        |                    0 |
| flatten_23         | -                          | -                        |                    0 |
| dense_46           | 41472                      | 144                      |                41616 |
| dropout_23         | -                          | -                        |                    0 |
| dense_47           | 1152                       | 8                        |                 1160 |
```

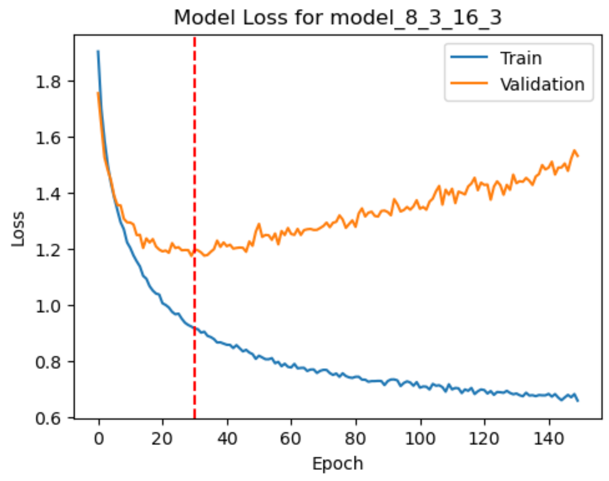


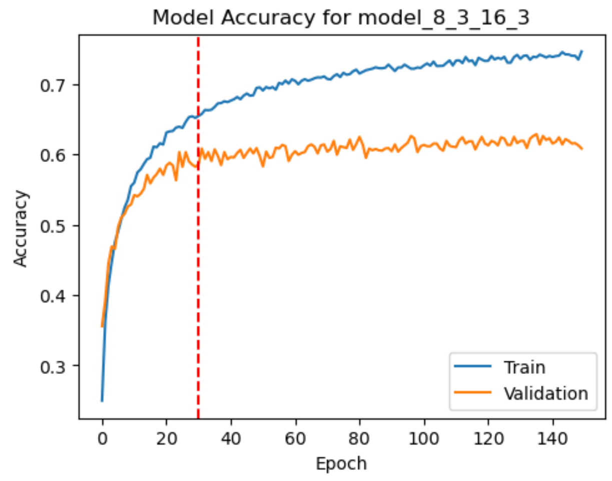
```
Model Name: model_24_5_48_2
 | Layer Name         | Trainable Weights Shape     | Trainable Bias Shape      |    Total Parameters |
 |--------------------+-----------------------------+---------------------------+---------------------|
 | conv2d_48          | 600                         | 24                        |                 624 |
 | max_pooling2d_48   | -                           | -                         |                   0 |
 | conv2d_49          | 4608                        | 48                        |                4656 |
 | max_pooling2d_49   | -                           | -                         |                   0 |
 | flatten_24         | -                           | -                         |                   0 |
 | dense_48           | 93312                       | 216                       |               93528 |
 | dropout_24         | -                           | -                         |                   0 |
 | dense_49           | 1728                        | 8                         |                1736 |
Model Name: model_24_5_48_3
 | Layer Name         | Trainable Weights Shape     | Trainable Bias Shape      |    Total Parameters |
 |--------------------+-----------------------------+---------------------------+---------------------|
 | conv2d_50          | 600                         | 24                        |                 624 |
 | max_pooling2d_50   | -                           | -                         |                   0 |
 | conv2d_51          | 10368                       | 48                        |               10416 |
 | max_pooling2d_51   | -                           | -                         |                   0 |
 | flatten_25         | -                           | -                         |                   0 |
 | dense_50           | 93312                       | 216                       |               93528 |
 | dropout_25         | -                           | -                         |                   0 |
 | dense_51           | 1728                        | 8                         |                1736 |
Model Name: model_16_3_16_2
 | Layer Name         | Trainable Weights Shape     | Trainable Bias Shape      |    Total Parameters |
 |--------------------+-----------------------------+---------------------------+---------------------|
 | conv2d_40          | 144                         | 16                        |                 160 |
 | max_pooling2d_40   | -                           | -                         |                   0 |
 | conv2d_41          | 1024                        | 16                        |                1040 |
 | max_pooling2d_41   | -                           | -                         |                   0 |
 | flatten_20         | -                           | -                         |                   0 |
 | dense_40           | 32768                       | 128                       |               32896 |
 | dropout_20         | -                           | -                         |                   0 |
 | dense_41           | 1024                        | 8                         |                1032 |
Model Name: model_16_3_16_3
 | Layer Name         | Trainable Weights Shape     | Trainable Bias Shape      |    Total Parameters |
 |--------------------+-----------------------------+---------------------------+---------------------|
 | conv2d_42          | 144                         | 16                        |                 160 |
 | max_pooling2d_42   | -                           | -                         |                   0 |
 | conv2d_43          | 2304                        | 16                        |                2320 |
 | max_pooling2d_43   | -                           | -                         |                   0 |
 | flatten_21         | -                           | -                         |                   0 |
 | dense_42           | 10368                       | 72                        |               10440 |
 | dropout_21         | -                           | -                         |                   0 |
 | dense_43           | 576                         | 8                         |                 584 |
```

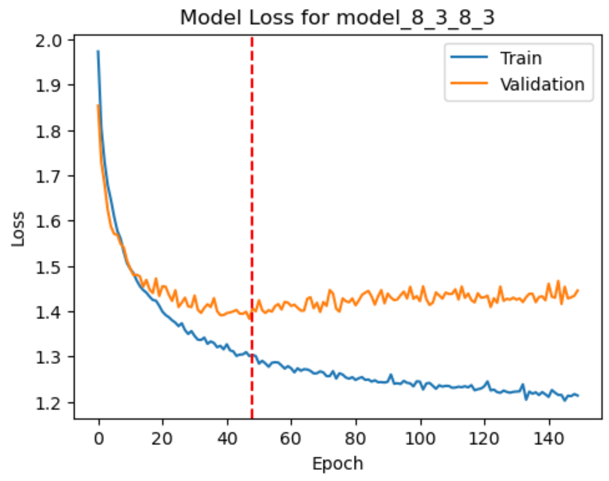
<u>Automatic Training Loss and Accuracy for First Run:</u>

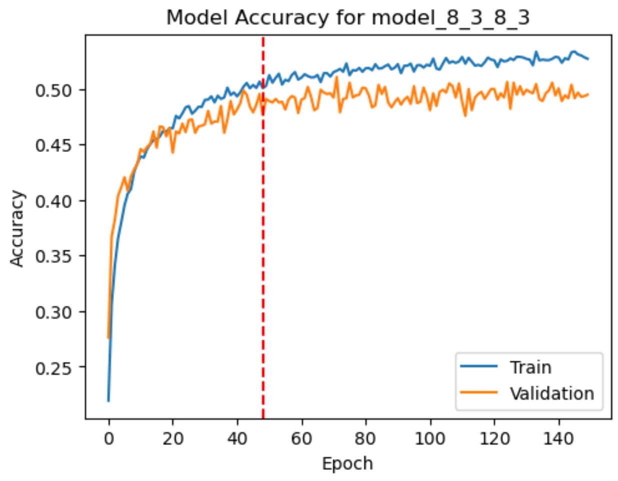


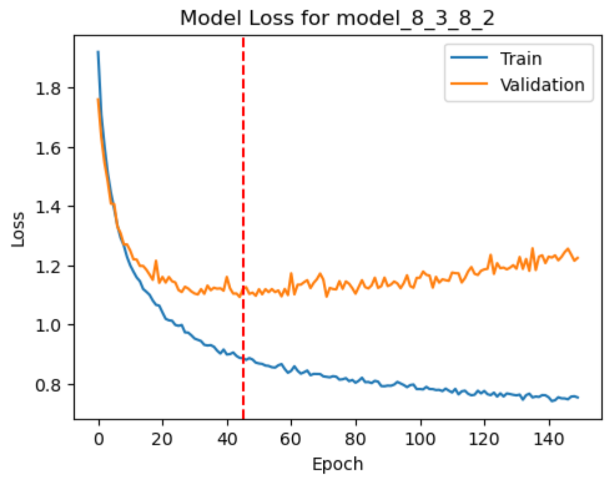


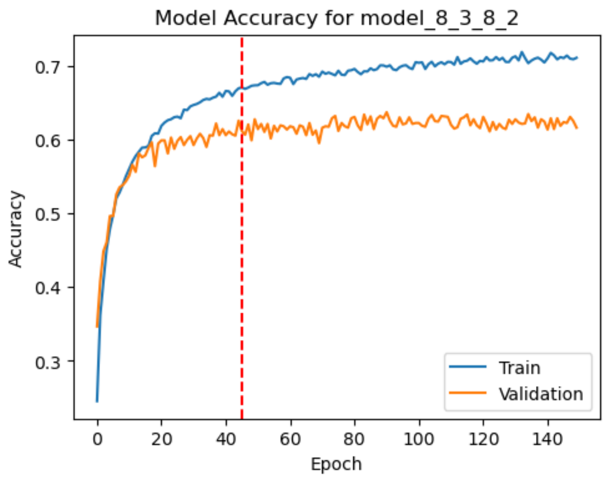


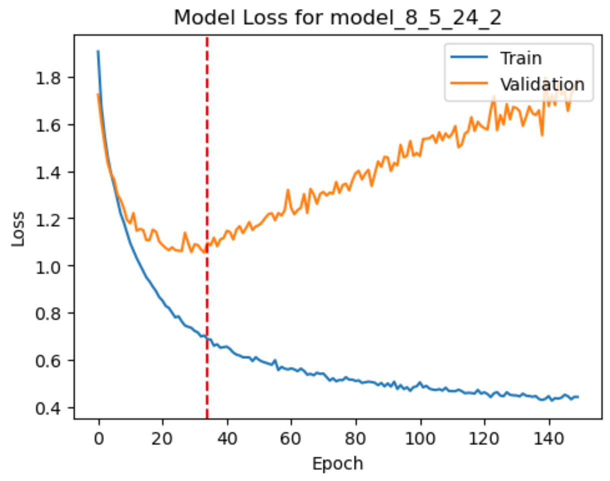


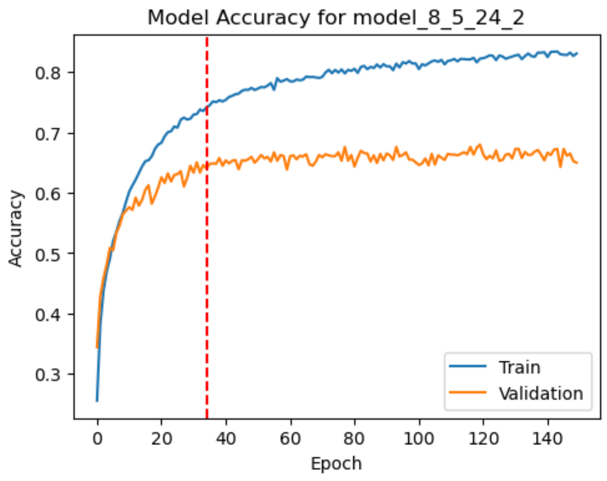


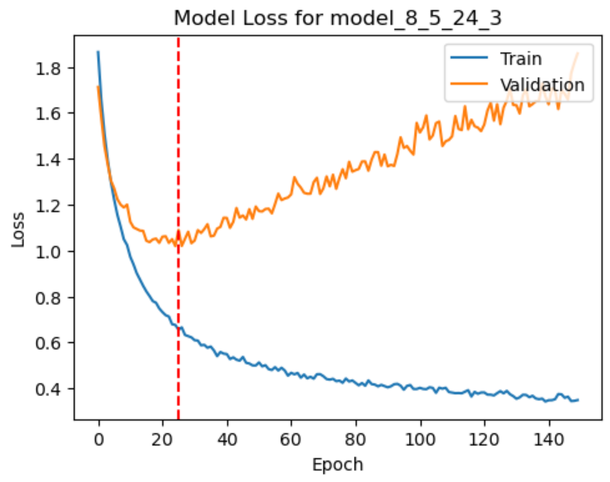


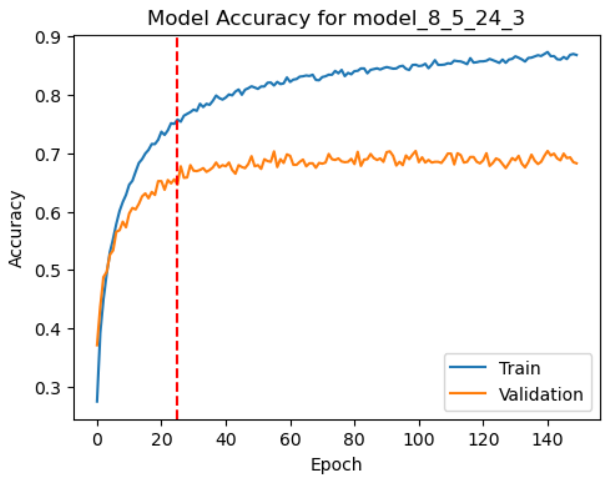

Model Loss for model_8_3_8_2
Loss
Epoch
Train
Validation
0.8
1.0
1.2
1.4
1.6
1.8
0
20
40
60
80
100
120
140

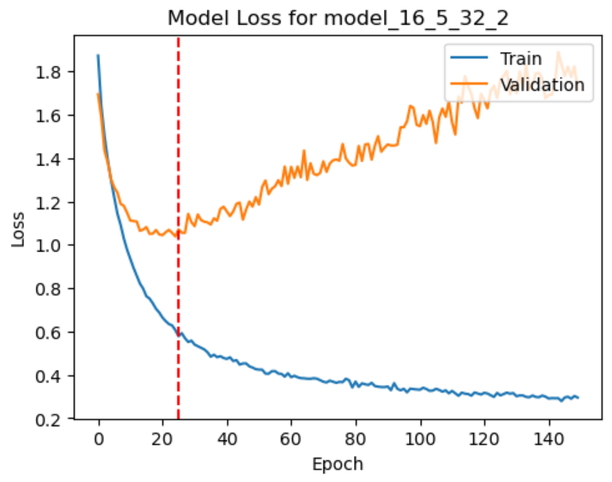

Model Accuracy for model_8_3_8_2
Accuracy
Epoch
Train
Validation
0.3
0.4
0.5
0.6
0.7
0
20
40
60
80
100
120
140

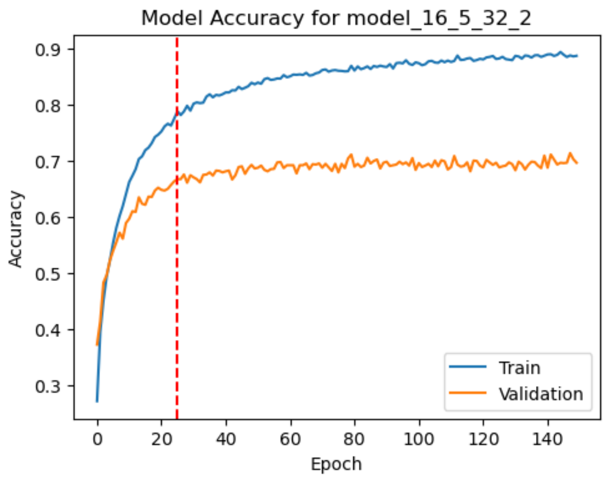

Model Loss for model_8_5_24_2
Loss
Epoch
Train
Validation
0.4
0.6
0.8
1.0
1.2
1.4
1.6
1.8
0
20
40
60
80
100
120
140

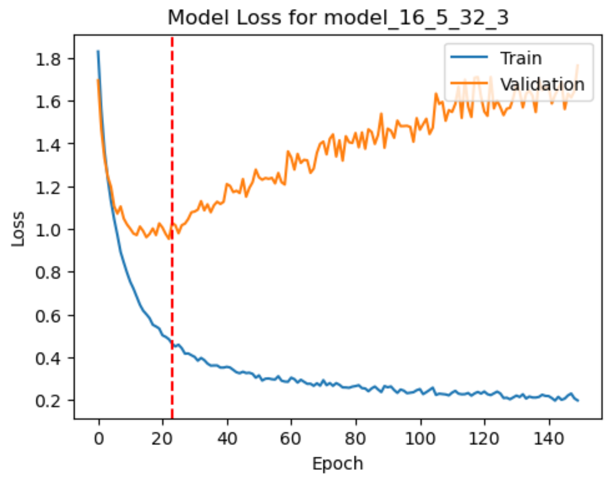

Model Accuracy for model_8_5_24_2
Accuracy
Epoch
Train
Validation
0.3
0.4
0.5
0.6
0.7
0.8
0
20
40
60
80
100
120
140

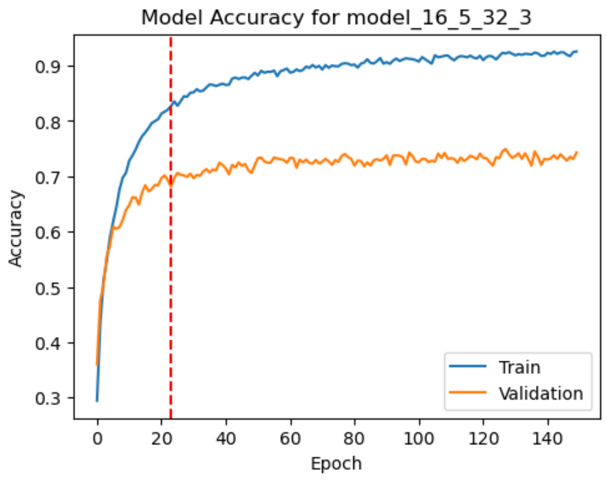

Model Loss for model_8_5_24_3
Loss
Epoch
Train
Validation
0.4
0.6
0.8
1.0
1.2
1.4
1.6
1.8
0
20
40
60
80
100
120
140

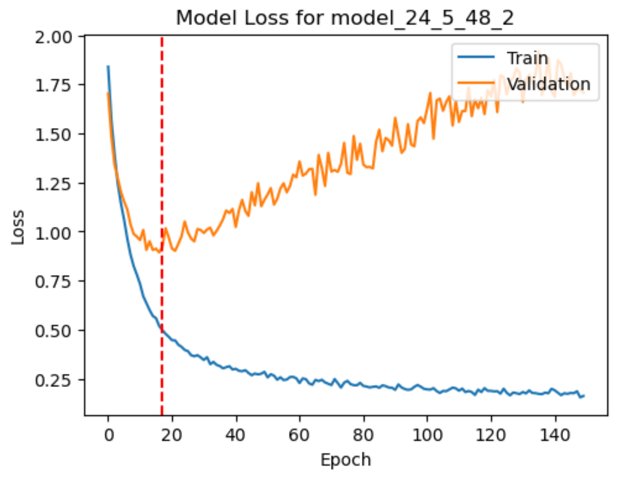

Model Accuracy for model_8_5_24_3
Accuracy
Epoch
Train
Validation
0.3
0.4
0.5
0.6
0.7
0.8
0.9
0
20
40
60
80
100
120
140

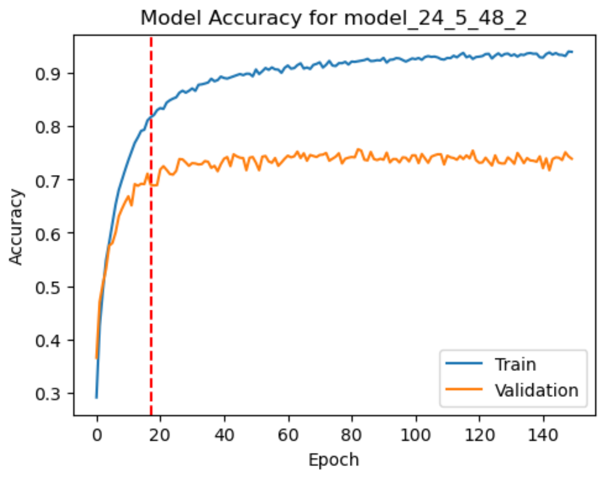

Model Loss for model_16_5_32_2
Train
Validation
Loss
Epoch

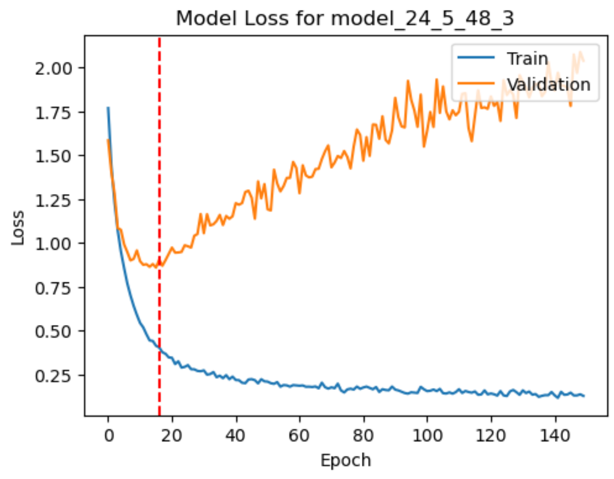

Model Accuracy for model_16_5_32_2
Train
Validation
Accuracy
Epoch

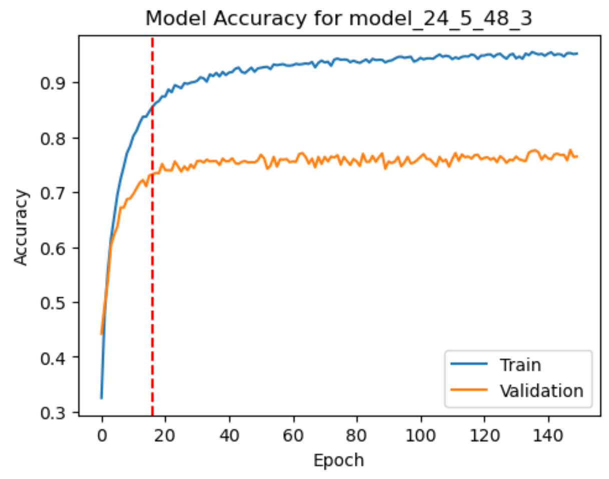

Model Loss for model_16_5_32_3
Train
Validation
Loss
Epoch

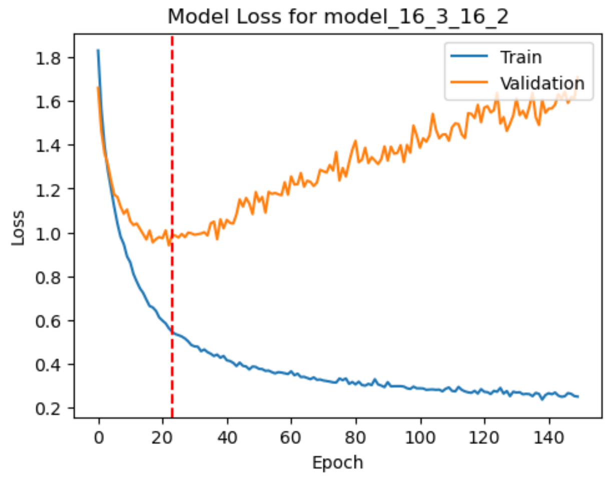

Model Accuracy for model_16_5_32_3
Train
Validation
Accuracy
Epoch

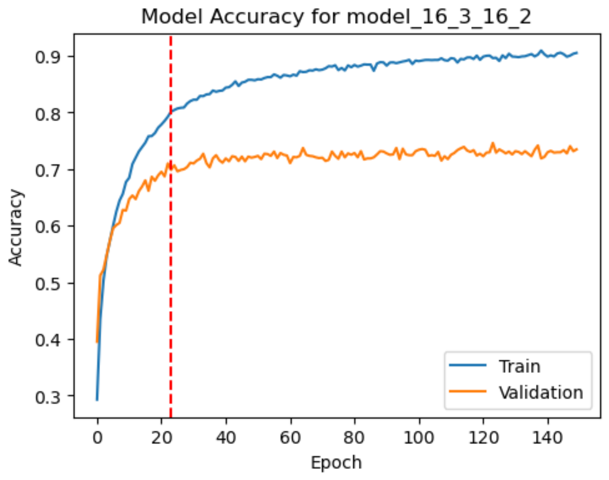

Model Loss for model_24_5_48_2
Train
Validation
Loss
Epoch

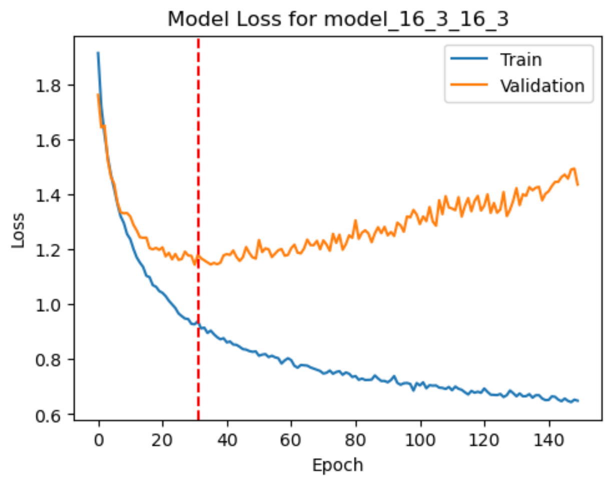

Model Accuracy for model_24_5_48_2
Train
Validation
Accuracy
Epoch

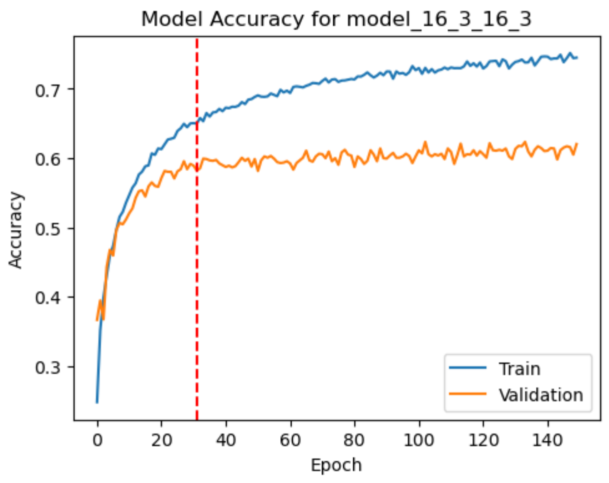

Model Loss for model_24_5_48_3
Train
Validation
Loss
Epoch
0
20
40
60
80
100
120
140
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00


Model Accuracy for model_24_5_48_3
Train
Validation
Accuracy
Epoch
0
20
40
60
80
100
120
140
0.3
0.4
0.5
0.6
0.7
0.8
0.9

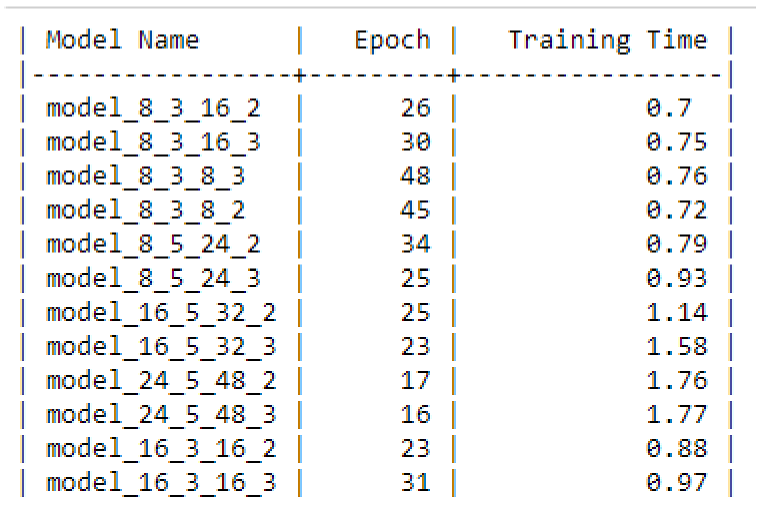

Model Loss for model_16_3_16_2
Train
Validation
Loss
Epoch
0
20
40
60
80
100
120
140
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
1.8


Model Accuracy for model_16_3_16_2
Train
Validation
Accuracy
Epoch
0
20
40
60
80
100
120
140
0.3
0.4
0.5
0.6
0.7
0.8
0.9


Model Loss for model_16_3_16_3
Train
Validation
Loss
Epoch
0
20
40
60
80
100
120
140
0.6
0.8
1.0
1.2
1.4
1.6
1.8


Model Accuracy for model_16_3_16_3
Train
Validation
Accuracy
Epoch
0
20
40
60
80
100
120
140
0.3
0.4
0.5
0.6
0.7

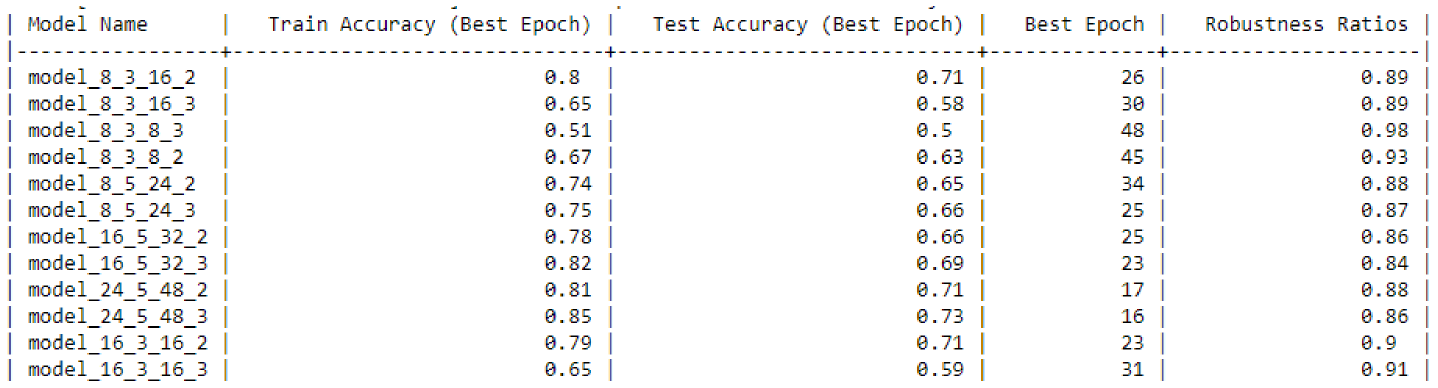
3. Epoch k* of Best Performance for Each CNN Model



```
| Model Name       |   Epoch |   Training Time |
|------------------+---------+-----------------|
| model_8_3_16_2   |      26 |            0.7  |
| model_8_3_16_3   |      30 |            0.75 |
| model_8_3_8_3    |      48 |            0.76 |
| model_8_3_8_2    |      45 |            0.72 |
| model_8_5_24_2   |      34 |            0.79 |
| model_8_5_24_3   |      25 |            0.93 |
| model_16_5_32_2  |      25 |            1.14 |
| model_16_5_32_3  |      23 |            1.58 |
| model_24_5_48_2  |      17 |            1.76 |
| model_24_5_48_3  |      16 |            1.77 |
| model_16_3_16_2  |      23 |            0.88 |
| model_16_3_16_3  |      31 |            0.97 |
```

Note: training time is in seconds (s)

4. Compare Best Performance Amongst Difference CNN Models



```
| Model Name      |   Train Accuracy (Best Epoch) |   Test Accuracy (Best Epoch) |   Best Epoch |   Robustness Ratios |
|-----------------+-------------------------------+------------------------------+--------------+---------------------|
| model_8_3_16_2  |                          0.8  |                         0.71 |           26 |                0.89 |
| model_8_3_16_3  |                          0.65 |                         0.58 |           30 |                0.89 |
| model_8_3_8_3   |                          0.51 |                         0.5  |           48 |                0.98 |
| model_8_3_8_2   |                          0.67 |                         0.63 |           45 |                0.93 |
| model_8_5_24_2  |                          0.74 |                         0.65 |           34 |                0.88 |
| model_8_5_24_3  |                          0.75 |                         0.66 |           25 |                0.87 |
| model_16_5_32_2 |                          0.78 |                         0.66 |           25 |                0.86 |
| model_16_5_32_3 |                          0.82 |                         0.69 |           23 |                0.84 |
| model_24_5_48_2 |                          0.81 |                         0.71 |           17 |                0.88 |
| model_24_5_48_3 |                          0.85 |                         0.73 |           16 |                0.86 |
| model_16_3_16_2 |                          0.79 |                         0.71 |           23 |                0.9  |
| model_16_3_16_3 |                          0.65 |                         0.59 |           31 |                0.91 |
```

For the performance of our models, the best was a test accuracy of 73% obtained by model_24_5_48_3, followed by model_8_3_16_2, model_24_5_48_2, and model_16_3_16_2. All four models have relatively similar robustness ratios which are in the range [0.86, 0.9]. We select model_24_5_48_3, for having the highest test accuracy, and for the other 3, we select the one with the highest robustness ratio, which would be model_16_3_16_2 with an accuracy of 71% and robustness ratio of 0.9. When comparing the two models, model_16_3_16_2's worst class of prediction was English with an accuracy of 61%, while it's best class of prediction was Castellar at 83%. Model_24_5_48_3's worst class of prediction was Rage at 64% while its best class was Castellar at 90%. For sake of accuracy, we will pick the best model as model_24_5_48_3.

Confusion Matrices of each CNN Model

Model_8_3_16_2	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	88%	0%	0%	4%	2%	2%	1%	3%
Californian	1%	66%	12%	9%	4%	4%	0%	1%
French	2%	7%	68%	6%	6%	8%	1%	2%
Rage	4%	2%	3%	74%	4%	7%	4%	3%
English	1%	5%	8%	4%	67%	5%	3%	6%
Brush	5%	3%	4%	10%	2%	66%	5%	6%
Gloucester	2%	2%	3%	11%	1%	6%	72%	2%
Matura	8%	2%	4%	5%	6%	3%	3%	71%

Model_8_3_16_3	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	73%	2%	2%	1%	6%	5%	2%	9%
Californian	3%	54%	17%	3%	11%	4%	3%	2%
French	4%	11%	61%	3%	9%	7%	2%	4%
Rage	8%	7%	2%	52%	8%	15%	5%	3%
English	2%	11%	7%	7%	53%	7%	7%	8%
Brush	5%	8%	8%	6%	5%	58%	5%	5%
Gloucester	9%	5%	2%	5%	9%	11%	55%	2%
Matura	13%	6%	4%	3%	8%	4%	5%	59%

Model_8_3_8_3	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	65%	4%	1%	3%	2%	3%	13%	9%
Californian	3%	42%	23%	7%	9%	4%	8%	3%
French	5%	10%	53%	3%	7%	7%	9%	5%
Rage	7%	8%	3%	42%	4%	16%	14%	5%
English	7%	10%	7%	6%	38%	5%	14%	13%
Brush	8%	6%	3%	12%	3%	48%	17%	4%
Gloucester	12%	7%	2%	6%	1%	6%	62%	4%
Matura	19%	5%	4%	4%	6%	3%	13%	49%

Model_8_3_8_2	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	73%	3%	2%	2%	1%	1%	4%	15%
Californian	6%	62%	13%	3%	6%	4%	3%	1%
French	3%	12%	62%	4%	6%	6%	4%	2%
Rage	4%	5%	1%	59%	7%	8%	11%	5%
English	4%	7%	8%	4%	57%	5%	7%	8%
Brush	7%	6%	3%	6%	8%	58%	3%	9%
Gloucester	7%	6%	2%	9%	4%	8%	61%	3%
Matura	9%	3%	4%	4%	7%	2%	6%	68%

Model_8_5_24_2	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	79%	4%	3%	3%	2%	0%	6%	4%
Californian	1%	70%	12%	4%	5%	3%	2%	1%
French	3%	12%	61%	4%	7%	4%	6%	4%
Rage	7%	5%	4%	60%	4%	11%	9%	0%
English	1%	12%	5%	7%	58%	7%	6%	4%
Brush	6%	5%	4%	10%	7%	56%	7%	5%
Gloucester	6%	8%	2%	4%	2%	5%	68%	5%
Matura	10%	3%	3%	5%	5%	3%	6%	67%

Model_8_5_24_3	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	82%	3%	1%	4%	1%	1%	2%	7%
Californian	2%	67%	9%	7%	2%	6%	3%	2%
French	2%	7%	65%	4%	9%	6%	3%	4%
Rage	7%	4%	3%	56%	4%	11%	13%	2%
English	4%	8%	7%	5%	60%	4%	6%	5%
Brush	5%	3%	3%	9%	7%	60%	8%	5%
Gloucester	3%	4%	0%	9%	3%	5%	70%	7%
Matura	9%	4%	4%	6%	4%	4%	4%	66%

Model_16_5_32_2	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	84%	0%	2%	1%	4%	1%	3%	5%
Californian	1%	58%	12%	3%	12%	6%	5%	1%
French	4%	3%	66%	3%	9%	7%	4%	4%
Rage	5%	2%	3%	52%	9%	13%	15%	1%
English	4%	8%	6%	5%	58%	8%	4%	7%
Brush	6%	1%	2%	5%	5%	71%	5%	5%
Gloucester	3%	1%	1%	2%	9%	6%	74%	4%
Matura	6%	3%	3%	2%	8%	5%	4%	70%

Model_16_5_32_3	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	80%	2%	1%	8%	2%	0%	3%	5%
Californian	1%	75%	7%	4%	5%	4%	1%	1%
French	2%	8%	69%	3%	9%	6%	1%	2%
Rage	5%	5%	5%	69%	2%	7%	3%	4%
English	5%	11%	8%	4%	60%	3%	5%	4%
Brush	4%	6%	3%	10%	5%	65%	5%	2%
Gloucester	6%	4%	1%	8%	3%	6%	69%	2%
Matura	6%	7%	5%	3%	5%	2%	5%	69%

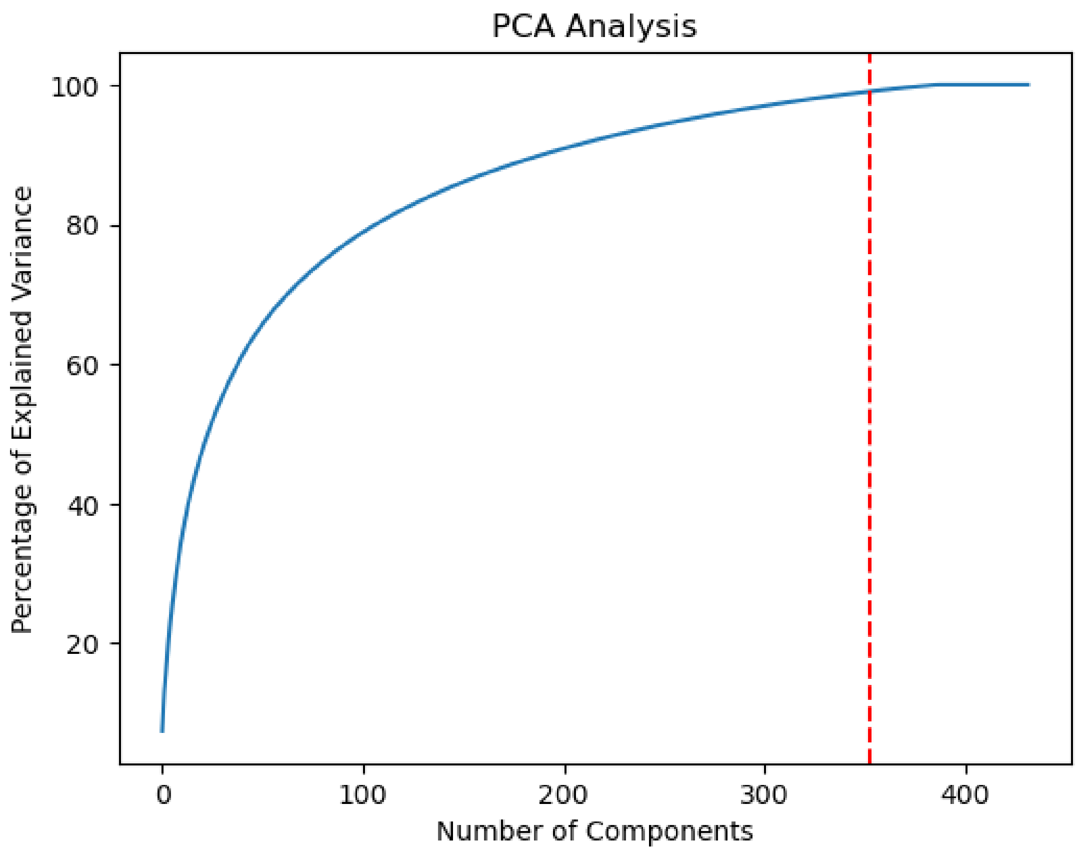
Model_24_5_48_2	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	83%	0%	3%	2%	1%	6%	3%	3%
Californian	1%	76%	6%	5%	2%	7%	2%	0%
French	4%	5%	74%	3%	3%	5%	3%	3%
Rage	6%	4%	3%	65%	2%	14%	4%	1%
English	3%	9%	7%	9%	60%	5%	3%	4%
Brush	5%	5%	4%	8%	3%	66%	6%	4%
Gloucester	2%	3%	4%	9%	0%	6%	76%	1%
Matura	9%	3%	4%	2%	3%	4%	5%	71%

Model_24_5_48_3	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	90%	0%	1%	0%	2%	6%	1%	2%
Californian	1%	75%	8%	1%	5%	8%	1%	0%
French	2%	3%	75%	4%	8%	5%	2%	1%
Rage	5%	5%	4%	64%	4%	13%	4%	2%
English	1%	8%	9%	4%	67%	5%	2%	4%
Brush	4%	4%	6%	5%	8%	68%	3%	2%
Gloucester	4%	2%	1%	4%	2%	8%	75%	3%
Matura	6%	6%	4%	2%	6%	4%	2%	72%

Model_16_3_16_2	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	83%	0%	2%	1%	2%	3%	5%	4%
Californian	2%	72%	9%	2%	3%	2%	5%	2%
French	3%	5%	66%	3%	9%	6%	5%	3%
Rage	5%	3%	1%	66%	3%	9%	10%	3%
English	3%	4%	10%	5%	61%	3%	7%	7%
Brush	5%	3%	3%	5%	6%	68%	9%	2%
Gloucester	4%	2%	1%	4%	3%	5%	79%	4%
Matura	5%	2%	1%	3%	6%	4%	5%	76%

Model_16_3_16_3	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	73%	7%	0%	2%	2%	2%	6%	9%
Californian	2%	61%	11%	5%	8%	5%	4%	2%
French	1%	11%	57%	5%	12%	6%	2%	4%
Rage	5%	7%	2%	45%	7%	21%	11%	3%
English	4%	12%	8%	3%	50%	5%	8%	12%
Brush	8%	7%	2%	4%	4%	61%	7%	6%
Gloucester	8%	12%	1%	6%	3%	11%	56%	5%
Matura	9%	6%	4%	3%	3%	3%	5%	69%

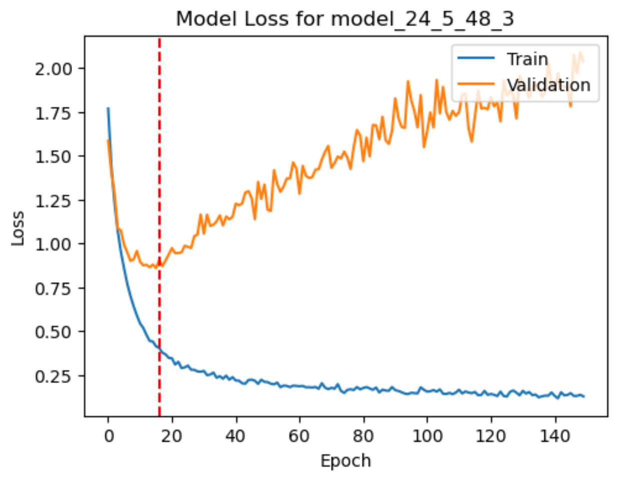
PCA of Model (24-5-48-3)

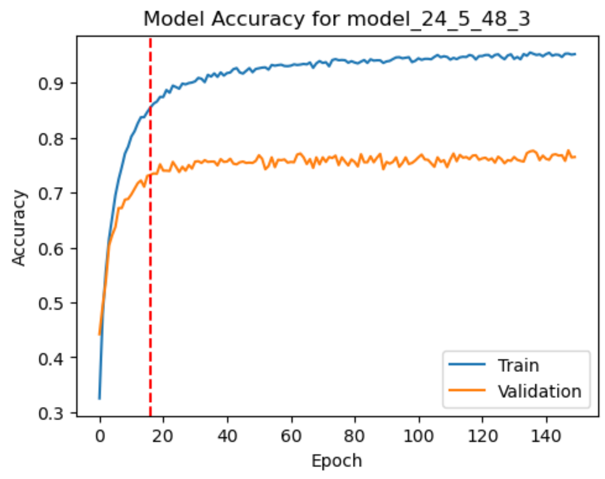


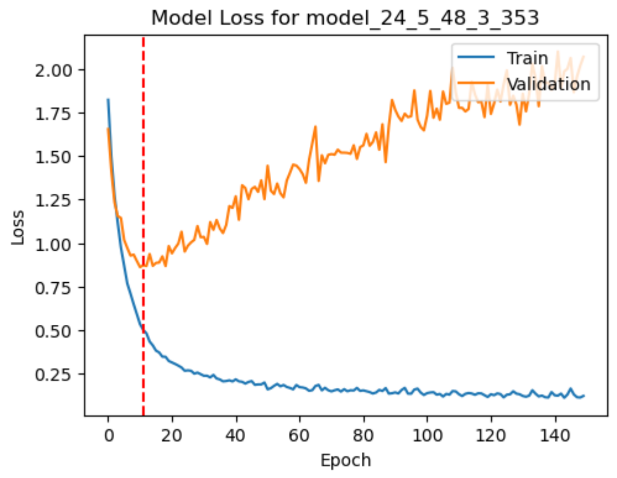
After running a PCA analysis on this model, we find that the number of components that explain 99% of variance is 353 components. So, our R = 353 for our further analysis.

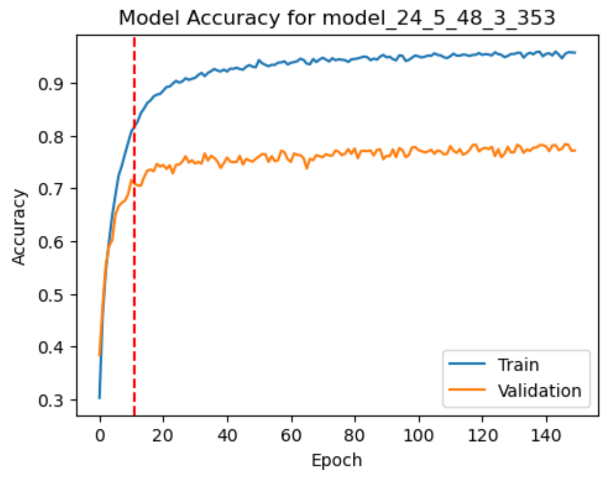
New Analysis with h=R and h=2R for Best Model (24-5-48-3)

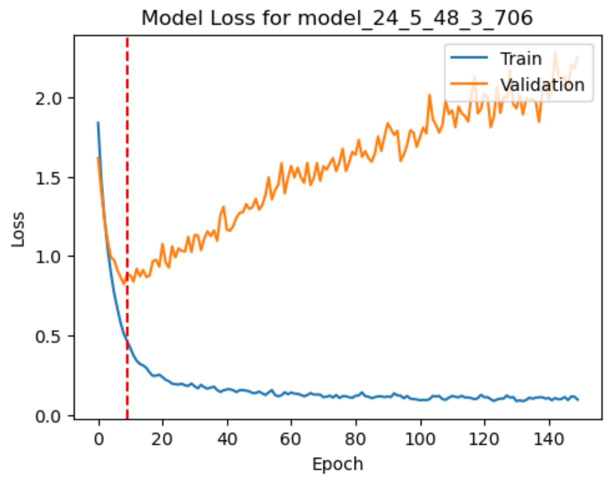
Our original hidden layer size was 216, we will now change this to 353 and 706 to compare performance amongst the 3 models (1 original, 2 new models containing new hidden layer sizes).

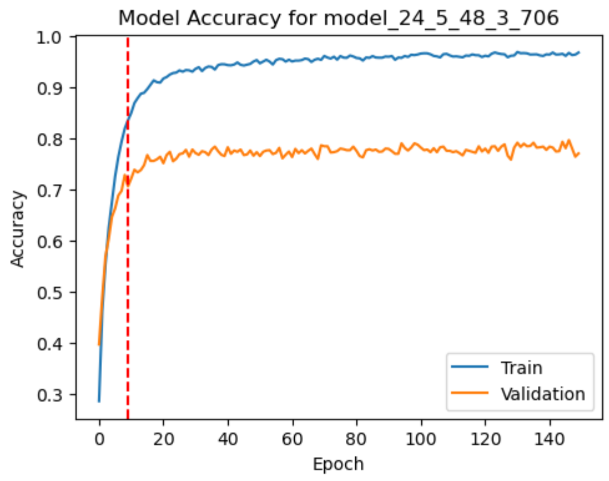


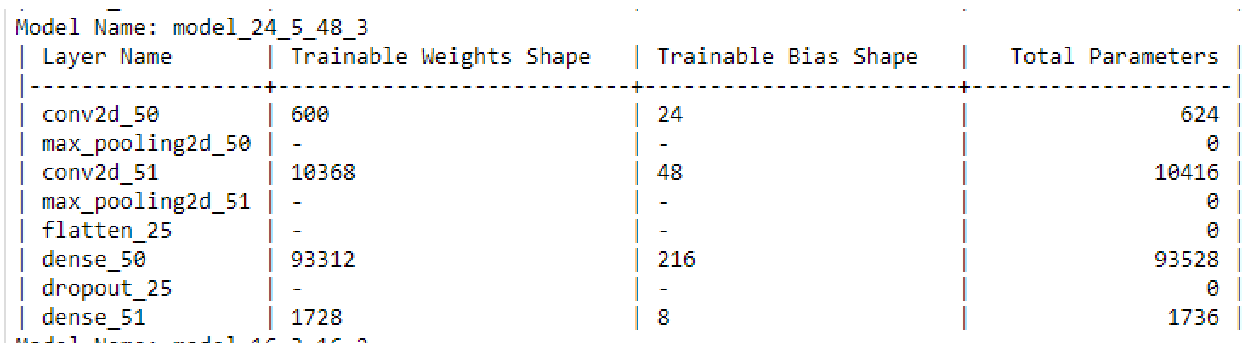




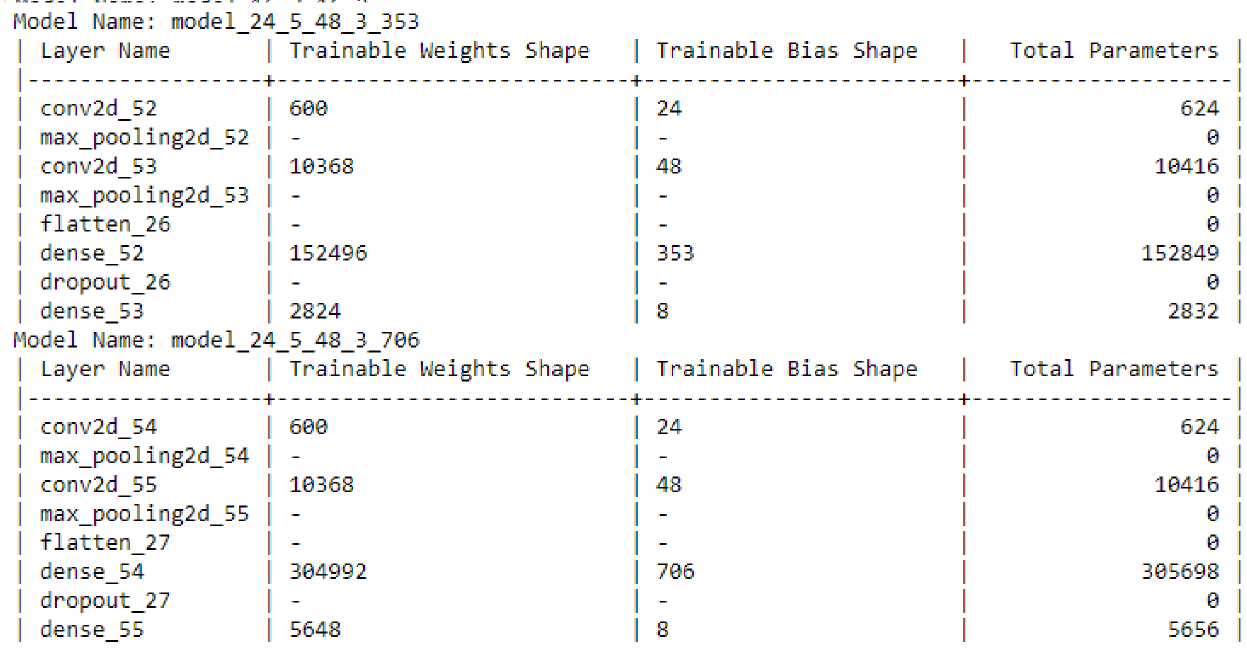








```
Model Name: model_24_5_48_3
 | Layer Name        | Trainable Weights Shape    | Trainable Bias Shape    |    Total Parameters |
 |-------------------+----------------------------+-------------------------+---------------------|
 | conv2d_50         | 600                        | 24                      |                 624 |
 | max_pooling2d_50  | -                          | -                       |                   0 |
 | conv2d_51         | 10368                      | 48                      |               10416 |
 | max_pooling2d_51  | -                          | -                       |                   0 |
 | flatten_25        | -                          | -                       |                   0 |
 | dense_50          | 93312                      | 216                     |               93528 |
 | dropout_25        | -                          | -                       |                   0 |
 | dense_51          | 1728                       | 8                       |                1736 |
```



```
Model Name: model_24_5_48_3_353
 | Layer Name        | Trainable Weights Shape    | Trainable Bias Shape    |    Total Parameters |
 |-------------------+----------------------------+-------------------------+---------------------|
 | conv2d_52         | 600                        | 24                      |                 624 |
 | max_pooling2d_52  | -                          | -                       |                   0 |
 | conv2d_53         | 10368                      | 48                      |               10416 |
 | max_pooling2d_53  | -                          | -                       |                   0 |
 | flatten_26        | -                          | -                       |                   0 |
 | dense_52          | 152496                     | 353                     |              152849 |
 | dropout_26        | -                          | -                       |                   0 |
 | dense_53          | 2824                       | 8                       |                2832 |
Model Name: model_24_5_48_3_706
 | Layer Name        | Trainable Weights Shape    | Trainable Bias Shape    |    Total Parameters |
 |-------------------+----------------------------+-------------------------+---------------------|
 | conv2d_54         | 600                        | 24                      |                 624 |
 | max_pooling2d_54  | -                          | -                       |                   0 |
 | conv2d_55         | 10368                      | 48                      |               10416 |
 | max_pooling2d_55  | -                          | -                       |                   0 |
 | flatten_27        | -                          | -                       |                   0 |
 | dense_54          | 304992                     | 706                     |              305698 |
 | dropout_27        | -                          | -                       |                   0 |
 | dense_55          | 5648                       | 8                       |                5656 |
```

Model_24_5_48_3	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	90%	0%	1%	0%	2%	6%	1%	2%
Californian	1%	75%	8%	1%	5%	8%	1%	0%
French	2%	3%	75%	4%	8%	5%	2%	1%
Rage	5%	5%	4%	64%	4%	13%	4%	2%
English	1%	8%	9%	4%	67%	5%	2%	4%
Brush	4%	4%	6%	5%	8%	68%	3%	2%
Gloucester	4%	2%	1%	4%	2%	8%	75%	3%
Matura	6%	6%	4%	2%	6%	4%	2%	72%

Model_24_5_48_3_353	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	81%	2%	1%	1%	6%	3%	3%	4%
Californian	1%	79%	5%	4%	5%	3%	1%	1%
French	1%	9%	67%	4%	7%	6%	4%	3%
Rage	4%	5%	2%	70%	6%	7%	5%	1%
English	2%	5%	10%	4%	67%	1%	4%	7%
Brush	5%	13%	6%	7%	5%	57%	6%	1%
Gloucester	3%	4%	1%	4%	4%	4%	79%	2%
Matura	3%	5%	4%	5%	6%	2%	2%	75%

Model_24_5_48_3_706	Castellar	Californian	French	Rage	English	Brush	Gloucester	Matura
Castellar	79%	2%	2%	3%	3%	2%	2%	7%
Californian	0%	74%	8%	4%	7%	2%	2%	2%
French	2%	9%	69%	6%	8%	3%	2%	2%
Rage	3%	4%	2%	74%	2%	8%	6%	2%
English	2%	7%	7%	5%	68%	2%	4%	5%
Brush	5%	5%	4%	8%	4%	65%	6%	3%
Gloucester	1%	1%	1%	5%	6%	2%	81%	3%
Matura	6%	2%	4%	4%	5%	2%	3%	76%

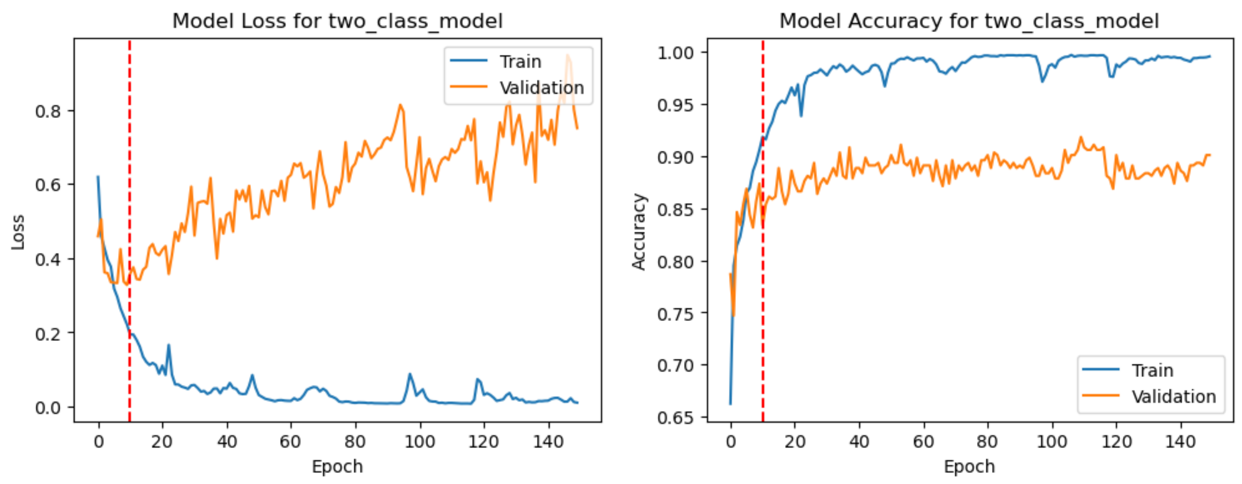
Model	# of Parameters	# of Information	Parsimony Ratio	Best Epoch	Training Time (s)	Robustness Ratio	Train Accuracy (Best Epoch)	Test Accuracy (Best Epoch)
Model_24_5_48_3	106304	113120	1.06	16	1.77	0.86	0.85	0.73
Model_24_5_48_3_353	166722	113120	0.68	11	1.62	0.89	0.81	0.72
Model_24_5_48_3_706	322394	113120	0.35	9	2.1	0.89	0.82	0.73

Our changing of the hidden layer size shows no improvement in test accuracy but improve in robustness ratio. Both increased hidden layer models have 0.89 robustness ratio which is slightly better than our original model. When comparing the confusion matrix of the new models, we can see the h=353 has a maximum of 81% for Castellar and minimum of 57% for Brush; and for h=706, maximum of 81% for Gloucester and minimum of 65% for Brush. Our original model has a max of 90% for Castellar and a min of 64% for Rage. Thus, it is marginally close for our original model and the model with h=706, and based on the parsimony ratio, we should choose h=706 as our best model.

Two Classes Analysis

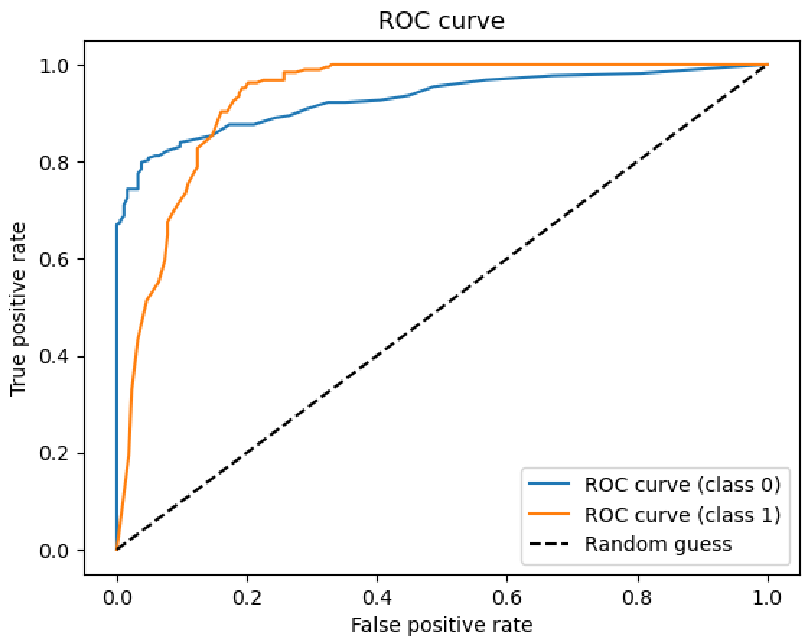
The two highest classes of our classification is Castellar at 79% and Gloucester at 81%.

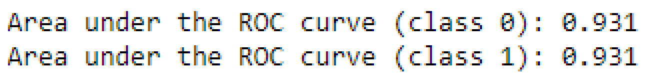
We proceeded with the best windows+channels and h=706 configurations to implement a new CNN classifier with only these two classes.



Two Class Analysis	Castellar	Gloucester
Castellar	81%	19%
Gloucester	5%	95%

The best epoch for our two-class model is epoch 10. For the confusion matrix, we only slightly increased the accuracy of Castellar compared to our multiclass classifier, from 79% to 81%; while the classification of Gloucester increased drastically from 81% to 95%.





```
Area under the ROC curve (class 0): 0.931
Area under the ROC curve (class 1): 0.931
```

Based on our AUC curve, we obtain a good classifier as its value is close to 1.