Assignment 1

Classifying CIFAR-10 with MLP classifier

Jai Soni

NUID : 001822913

1. Base Model

Paramaters\Model	model
Batch Size	200
Epochs	30
Number of Neurons per layer	512
Number of layers	2
Learning Rate	0.01
Activation Functions	Relu
Dropout Rates	0.2

Results:

Train Loss	0.0713
Train Accuracy	0.4318
Validation Loss	0.0701
Validation Accuracy	0.4458

Let's consider the results of this configuration of MLP model as Benchmark/Base. We will compare all other configurations against this model and try to come up with a more optimized model which provides better accuracy and less loss.

- Test Accuracy : 0.4458
- Test Loss: 0.07010

Number of Epochs

Lets Change the Number of Epochs and see the difference in model's accuracy and loss

	Epoch
Paramaters\Model	model2
Batch Size	200
Epochs	70
Number of Neurons per layer	512
Number of layers	2

Learning Rate	0.01
Activation Functions	Relu
Dropout Rates	0.2

Results:

Train Loss	0.0652
Train Accuracy	0.4922
Validation Loss	0.0654
Validation Accuracy	0.4865

Observation on Changing Epochs:

Increasing the Epochs to 70 results in increasing the accuracy on both test and validation data.

- Test Loss: 0.06539
- Test Accuracy: 0.4865

Batch Size

Lets Change the Batch Size, We will test same model with batch sizes of 100,200 and 300. We already have batch size of 200 in out benchmark model.

	Batch Size	
Paramaters\Model	model3_1	model3_2
Batch Size	100	300
Epochs	30	30
Number of Neurons per layer	512	512

Number of layers	2	2
Learning Rate	0.01	0.01
Activation Functions	Relu	Relu
Dropout Rates	0.2	0.2

Results:

Train Loss	0.0607	0.0737
Train Accuracy	0.4735	0.4064
Validation Loss	0.0665	0.0722
Validation Accuracy	0.4722	0.4238

Model3_1


Model Accuracy- Training Vs Validation
Train
Validtaion
Accuracy
Epoch
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0
5
10
15
20
25
30


Model Loss- Training Vs Validation
Train
Validtaion
Loss
Epoch
0.0875
0.0850
0.0825
0.0800
0.0775
0.0750
0.0725
0.0700
0
5
10
15
20
25
30

Model3_2

Observations on changing batch size

Batch size -100

- Test Loss: 0.0665
- Test Accuracy: 0.4772

Batch size -300

- Test Loss: 0.0721
- Test Accuracy: 0.4238

Accuracy increases when batch size is less and decreases when batch size is more, similarly Loss is decreased on decreasing batch size and increased on increasing batch size.

Network Configuration - A- Number of Neurons in a layer

Lets change the number of neuron's in a layer in model, We will increase the number of neurons to 1.5 times and 3 times that of benchmark model

	Number of Neurons in a Layer	
Paramaters\Model	model4_1	model4_2
Batch Size	200	200
Epochs	30	30
Number of Neurons per layer	768	1536
Number of layers	2	2
Learning Rate	0.01	0.01
Activation Functions	Relu	Relu
Dropout Rates	0.2	0.2

Results:

Train Loss	0.0715	0.0693
Train Accuracy	0.4296	0.4516
Validation Loss	0.0703	0.0687
Validation Accuracy	0.4409	0.4581

Model4_1


Model Loss- Training Vs Validation
Train
Validtaion
0.0875
0.0850
0.0825
0.0800
0.0775
0.0750
0.0725
0.0700
Loss
0
5
10
15
20
25
30
Epoch

Model4_2


Model Accuracy- Training Vs Validation
Train
Validtaion
0.45
0.40
0.35
0.30
0.25
0.20
Accuracy
0
5
10
15
20
25
30
Epoch

Observations on changing Number of Neurons per layer

Number of Neurons is 1.5 times, i.e 512*1.5 =768

- Test Loss: 0.0695
- Test Accuracy: 0.4491

Number of Neurons is 3 times, i.e 512*3 =1536

- Test Loss: 0.0687
- Test Accuracy: 0.4581

Accuracy increases marginally on increasing the Number of neurons per layer, and similarly there is marginal decrease in loss.

Network Configuration -B- Number of Layers

Let's increase the layers by 2 and 4 from thebenchmark model.

	Number of Layers	
Paramaters\Model	model5_1	model5_2
Batch Size	200	200
Epochs	30	30
Number of Neurons per layer	512	512
Number of layers	4	6
Learning Rate	0.01	0.01
Activation Functions	Relu	Relu
Dropout Rates	0.2	0.2

Result:

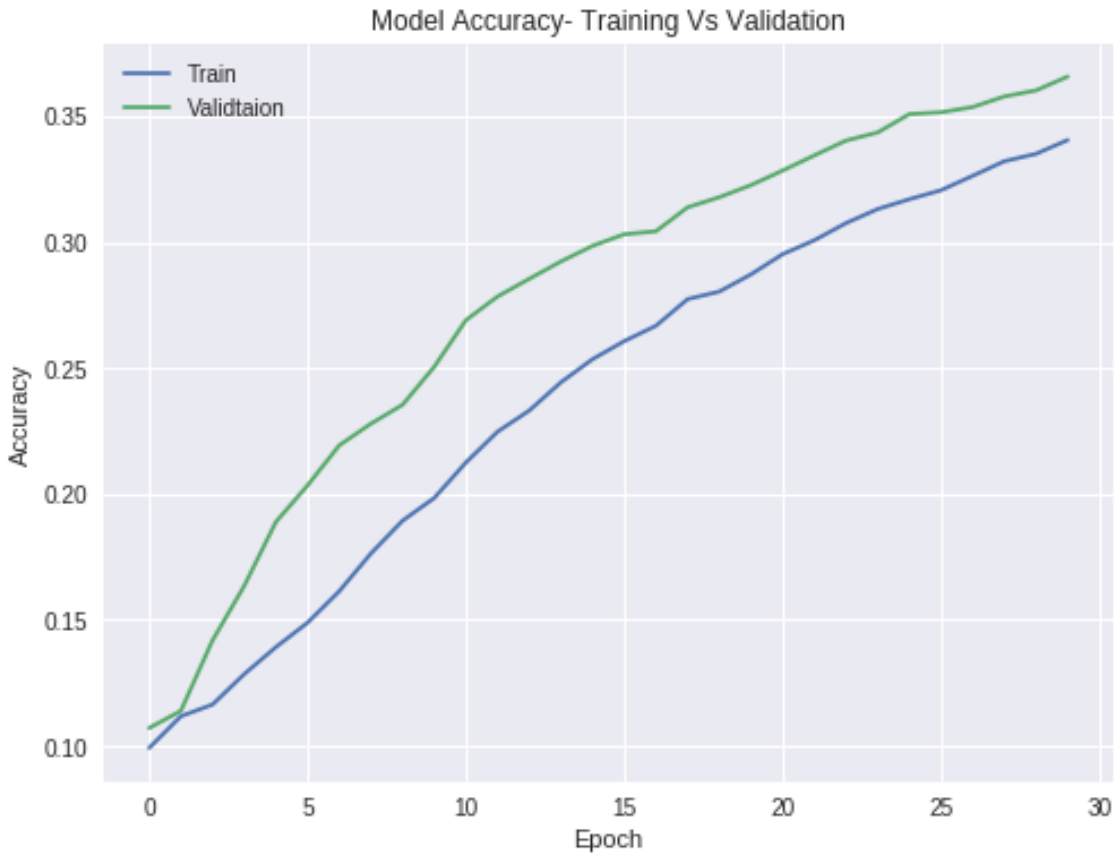
Train Loss	0.0739	0.0783
Train Accuracy	0.396	0.3405
Validation Loss	0.0715	0.0758
Validation Accuracy	0.4226	0.3657

Model5_1


Model Loss- Training Vs Validation
Train
Validtaion
0.0900
0.0875
0.0850
0.0825
0.0800
0.0775
0.0750
0.0725
Loss
0
5
10
15
20
25
30
Epoch


Model Accuracy- Training Vs Validation
Train
Validtaion
0.40
0.35
0.30
0.25
0.20
0.15
Accuracy
0
5
10
15
20
25
30
Epoch

Model5_2


Model Accuracy- Training Vs Validation
Train
Validtaion
Accuracy
0.35
0.30
0.25
0.20
0.15
0.10
0
5
10
15
20
25
30
Epoch


Model Loss- Training Vs Validation
Train
Validtaion
Loss
0.090
0.088
0.086
0.084
0.082
0.080
0.078
0.076
0
5
10
15
20
25
30
Epoch

Observations on changing Number layers

Total number of layers is 4

- Test Loss: 0.0714
- Test Accuracy: 0.4226

Total number of layers is 6

- Test Loss: 0.0758
- Test Accuracy: 0.3657

There is a consistent decrease in Accuracy and consistent increase in loss , on increasing the number of layers

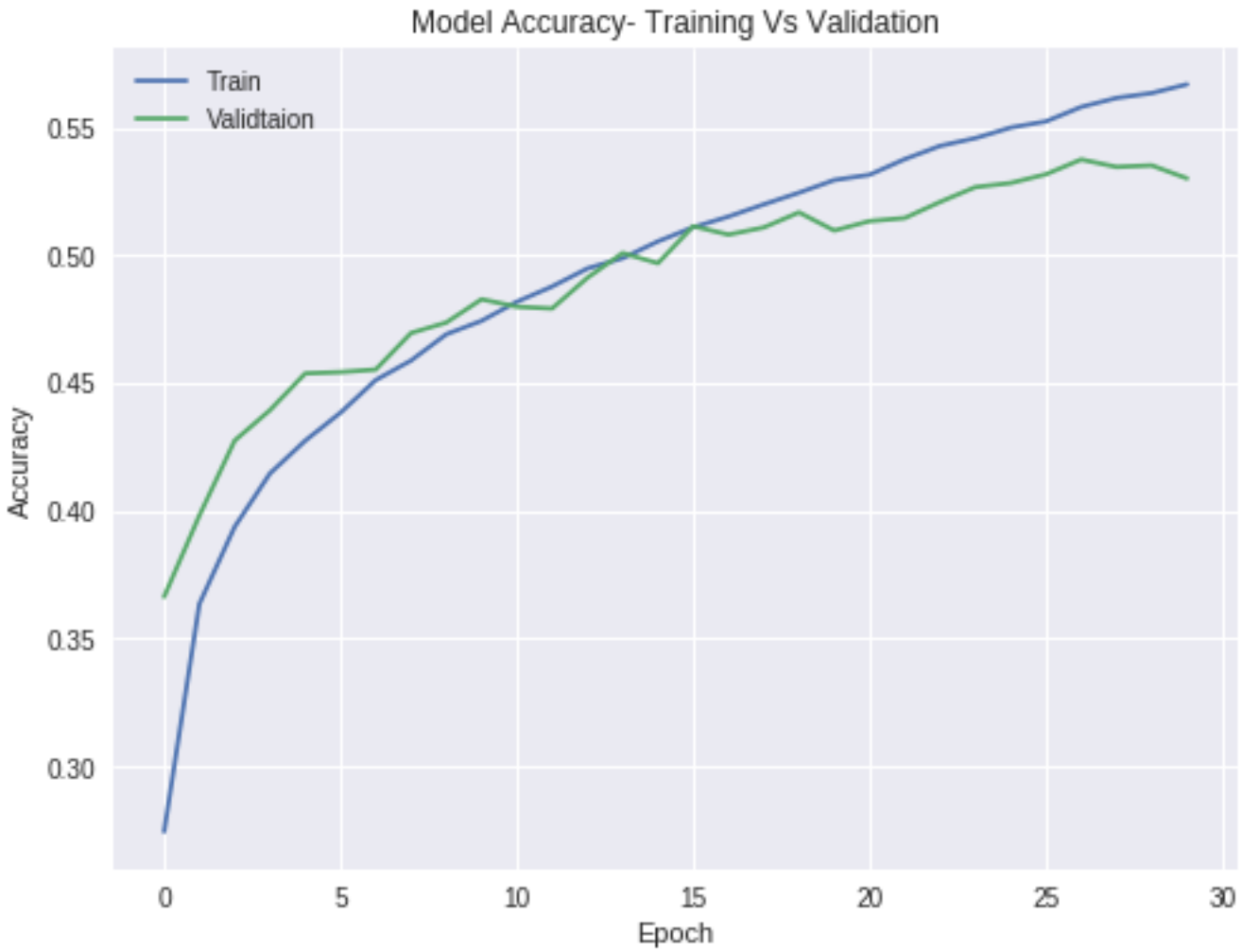
Learning Rate

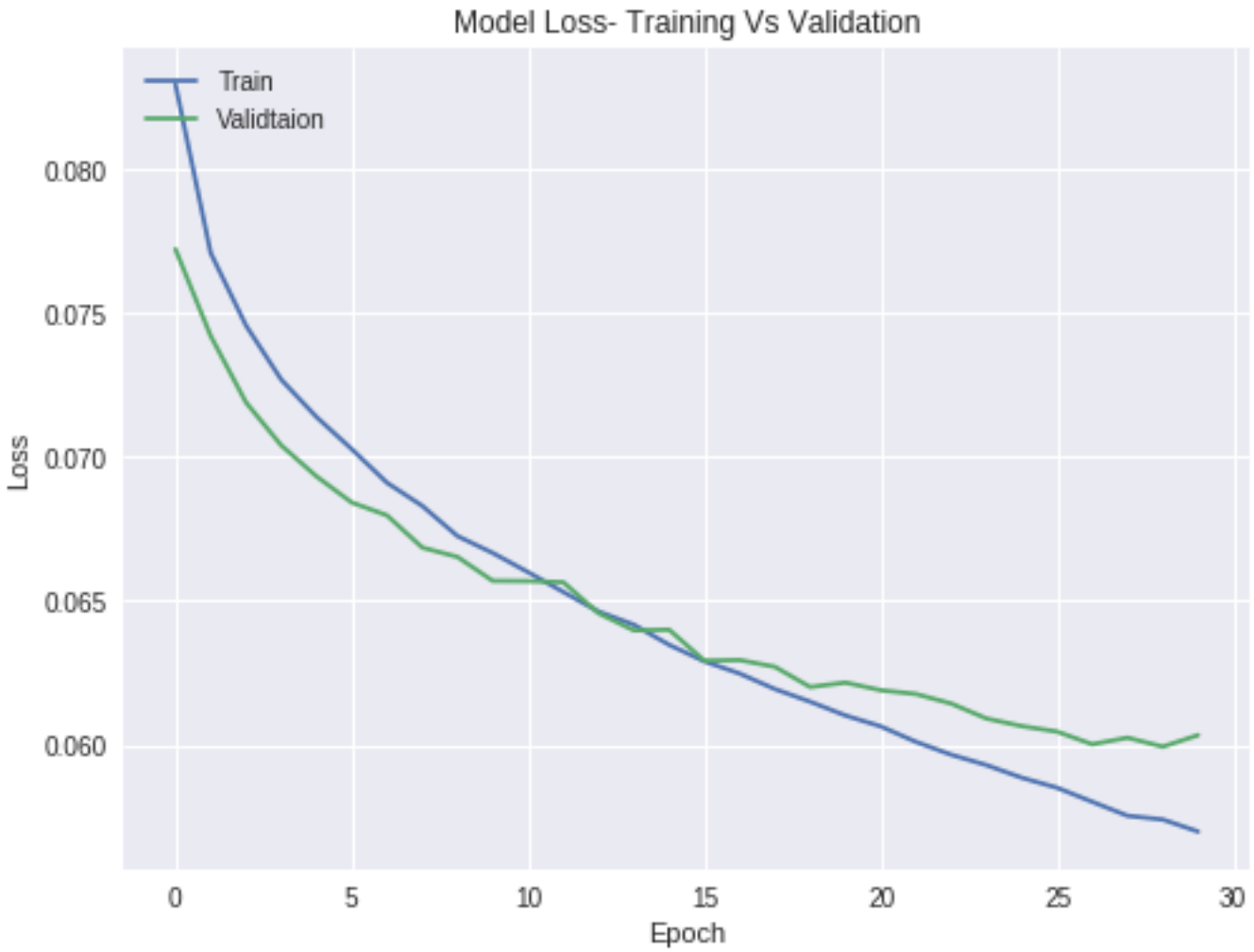
	Learning Rate	
Paramaters\Model	model6_1	model6_2
Batch Size	200	200
Epochs	30	30
Number of Neurons per layer	512	512
Number of layers	2	2
Learning Rate	0.1	0.005
Activation Functions	Relu	Relu
Dropout Rates	0.2	0.2

Result:

Train Loss	0.057	0.0756
Train Accuracy	0.5668	0.3814
Validation Loss	0.0603	0.074
Validation Accuracy	0.53	0.4043

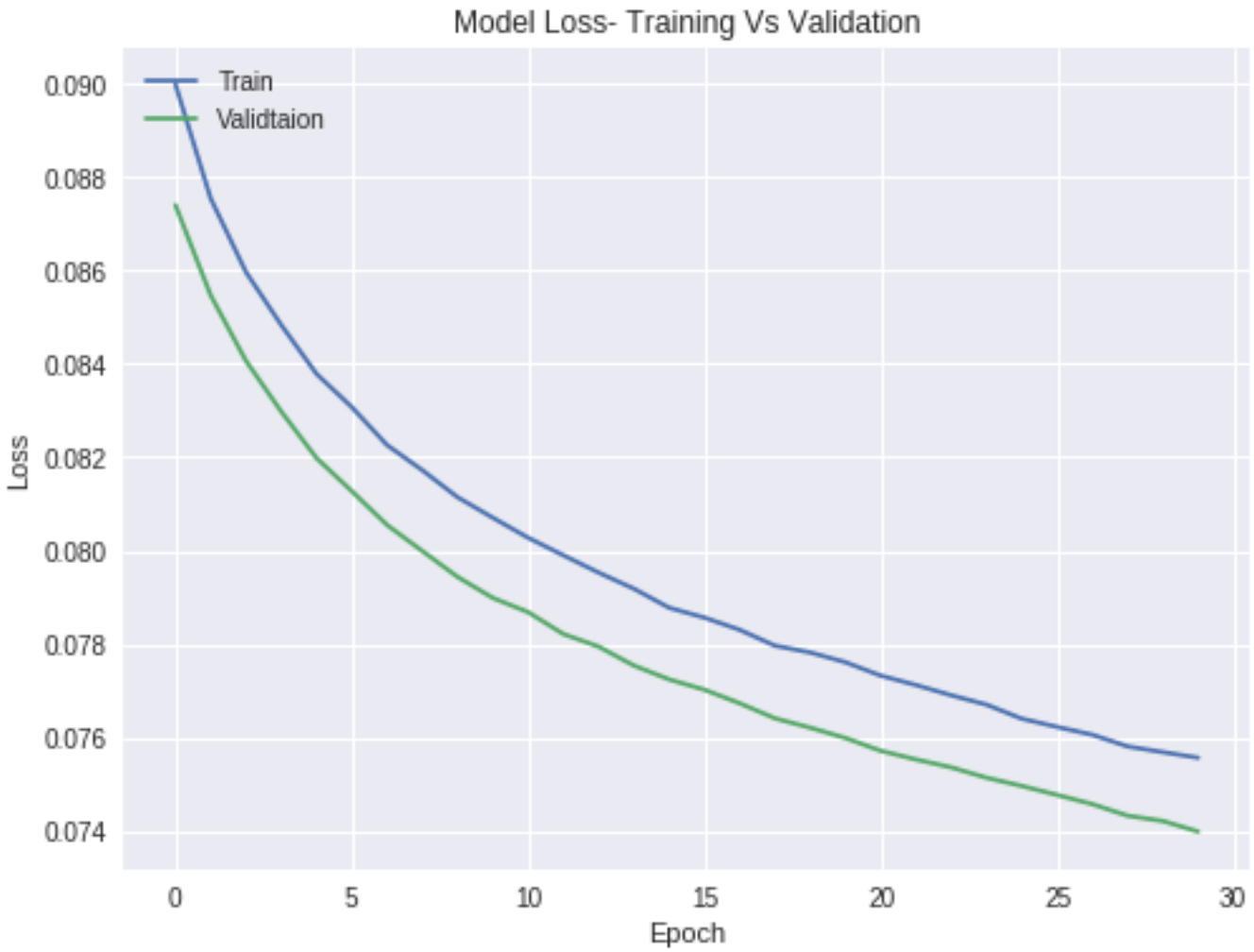
Model6_1


Model Accuracy- Training Vs Validation
Train
Validtaion
Accuracy
0.55
0.50
0.45
0.40
0.35
0.30
0
5
10
15
20
25
30
Epoch


Model Loss- Training Vs Validation
Train
Validtaion
0.080
0.075
0.070
0.065
0.060
Loss
0
5
10
15
20
25
30
Epoch

Model6_2


Model Accuracy- Training Vs Validation
Train
Validtaion
Accuracy
0.40
0.35
0.30
0.25
0.20
0.15
0
5
10
15
20
25
30
Epoch



Observations on changing Learning rate

Increasing learning rate from 0.01 to 0.1

- Test Loss: 0.0603
- Test Accuracy: 0.53

Decreasing learning rate from 0.01 to 0.005

- Test Loss: 0.0739
- Test Accuracy: 0.4043

Accuracy incerases on increasing the learning rate and decreases on decreasing the learning rate. Similarly, learning rate is inversly propotional to Loss

Activation Function

We will compare our benchmark model against tanh, sigmoid, elu and Leaky-Relu activation functions

	Activation Function			
Paramaters\Model	model7_1	model7_2	model7_3	model7_4
Batch Size	200	200	200	200
Epochs	30	30	30	30
Number of Neurons per layer	512	512	512	512
Number of layers	2	2	2	2
Learning Rate	0.01	0.01	0.01	0.01
Activation Functions	tanh	sigmoid	elu	Leaky_Relu
Dropout Rates	0.2	0.2	0.2	0.2

Results:

Train Loss	0.0727	0.0883	0.0722	0.0713
Train Accuracy	0.4171	0.1756	0.4257	0.4308
Validation Loss	0.0724	0.0872	0.0716	0.0702
Validation Accuracy	0.4183	0.2413	0.4271	0.4398

Observation on changing Activation Function

Tanh

- Test Loss: 0.0724
- Test Accuracy: 0.4183

Sigmoid

- Test Loss: 0.0871
- Test Accuracy: 0.2413

Elu

- Test Loss: 0.0716
- Test Accuracy: 0.4271

Leaky Relu

- Test Loss: 0.0702
- Test Accuracy: 0.4398

Accuracy is maximum in the the benchmark model with Relu activation function.

Dropout Rates

	Droupout Rate	
Paramaters\Model	model8_1	model8_2
Batch Size	200	200
Epochs	30	30
Number of Neurons per layer	512	512
Number of layers	2	2
Learning Rate	0.01	0.01
Activation Functions	Relu	Relu
Dropout Rates	0.1	0.5

Result:

Train Loss	0.0698	0.0765
Train Accuracy	0.4474	0.3713
Validation Loss	0.0695	0.0734
Validation Accuracy	0.4498	0.4133

Observation on changing Dropout Factor

Dropout - 0.1

Test Loss: 0.0694
Test Accuracy: 0.4498

Dropout - 0.005

Test Loss: 0.0733
Test Accuracy: 0.4133

Accuracy increases on in increasing the dropout factor, and decreases on decreasing the dropout.

You are expected to provide a recommendation for the best model you would recommend for classification. Which model (with parameter values) would you choose and why?

I would recommend the model with the following configuration:

batch_size = 200
num_classes = 10
epochs = 30
Number of neurons per layer = 512
Number of hidden layers = 2
Learning Rate = 0.1
Activation Function = Relu
Dropout Factor = 0.2

I have chosen this model as it provides best accuracy with less overfitting and underfitting and thus the training and validation accuracy and loss are close to each other.

Resulting Metrics:

- Train Accuracy: 0.5668
- Validation Accuracy: 0.5300
- Train Loss: 0.0570
- Validation Loss: 0.0603

Comment on how good your model is ? Does it overfit/underfit data ? What could you do to improve the model?

As per various models tested, I have found the recommended model is better than others as it has one of the least overfitting on the validation data. We can try to increasing the dropout factor to reduce its overfitting.