

# Deep Learning

---

## PROGRAMMING ASSIGNMENT 2

Submitted By:

**Gaurav Arora (CS18M021)**

**Sinzith Tatikonda (CS18M064)**

### Tasks

#### Checklist :

- ☒ We have read all the instruction carefully and followed them to our best ability.
- ☒ We have written the name, roll no in report.
- ☒ Run sanity\_check.sh.
- ☒ We will be submitting only single submission on behalf of our team.
- ☒ We have not included unnecessary text, pages, logos in the assignment.
- ☒ We have not used any high level APIs(Keras, Estimators for e.g.).
- ☒ We have not copied anything for this assignment.

---

## Contents

<b>1</b>	<b>Best Model</b>	<b>2</b>
1.1	Configurations . . . . .	2
1.2	Training Details . . . . .	2
<b>2</b>	<b>Performance on Test Data of model that performs best on Validation data.</b>	<b>2</b>
<b>3</b>	<b>Learning Curve</b>	<b>3</b>
<b>4</b>	<b>Parameter Setting</b>	<b>3</b>
<b>5</b>	<b>Layer-1 Filters</b>	<b>4</b>
<b>6</b>	<b>Batch Normalization</b>	<b>4</b>
<b>7</b>	<b>Dimensions, Parameters and Neurons</b>	<b>5</b>
<b>8</b>	<b>Extra</b>	<b>5</b>
8.1	Data Augmentation . . . . .	5
<b>9</b>	<b>Kaggle Accuracy</b>	<b>5</b>

---

# 1 Best Model

## 1.1 Configurations

We have used 6 convolution layers, 3 pooling layers and one fully connected layer.

*Conv1(Conv  $\Rightarrow$  BatchNorm  $\Rightarrow$  Relu)  $\Rightarrow$  Conv2(Conv  $\Rightarrow$  BatchNorm  $\Rightarrow$  Relu)  $\Rightarrow$  Maxpool1  $\Rightarrow$  Conv3(Conv  $\Rightarrow$  BatchNorm  $\Rightarrow$  Relu)  $\Rightarrow$  Conv4(Conv  $\Rightarrow$  BatchNorm  $\Rightarrow$  Relu)  $\Rightarrow$  Maxpool2  $\Rightarrow$  Conv5(Conv  $\Rightarrow$  BatchNorm  $\Rightarrow$  Relu)  $\Rightarrow$  Conv6(Conv  $\Rightarrow$  BatchNorm  $\Rightarrow$  Relu)  $\Rightarrow$  Maxpool3  $\Rightarrow$  FC1  $\Rightarrow$  Relu  $\Rightarrow$  Dropout  $\Rightarrow$  FC2  $\Rightarrow$  BatchNorm  $\Rightarrow$  Softmax*

## 1.2 Training Details

Hyper Parameters

- learning rate= 0.001
- batch size= 300
- initialization =Xavier
- data augmentation =True(Augmented 7 times)
- batch normalization=True
- dropout=True(0.5)

## 2 Performance on Test Data of model that performs best on Validation data.

The performance of the model on the test and validation is relatively same. With increase in training data it is able to perform well.

Test	Validation
52.5	49.5
47.5	48

---

### 3 Learning Curve

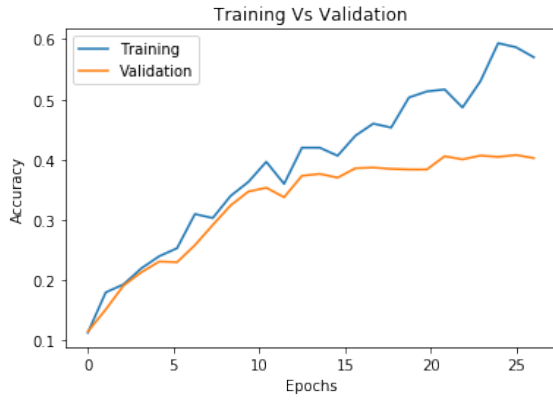


Figure 1: Accuracy

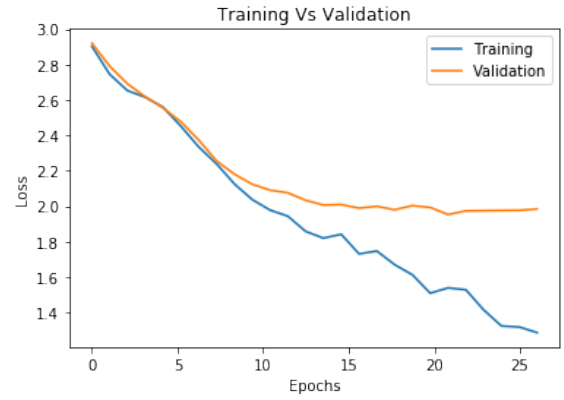


Figure 2: Loss

### 4 Parameter Setting

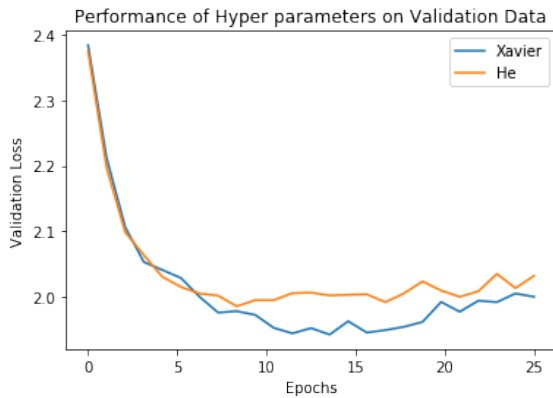


Figure 3: lr=0.001, batch-size=300

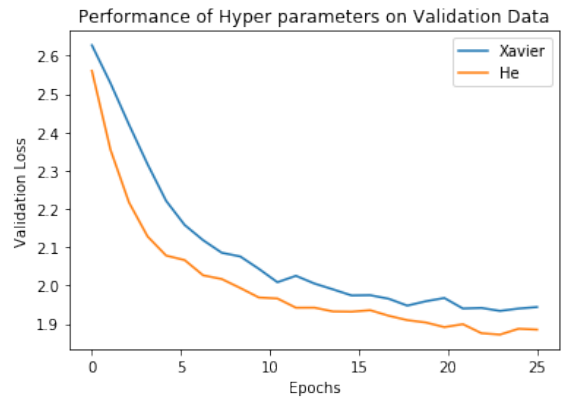


Figure 4: lr=0.01, batch-size=200

---

## 5 Layer-1 Filters

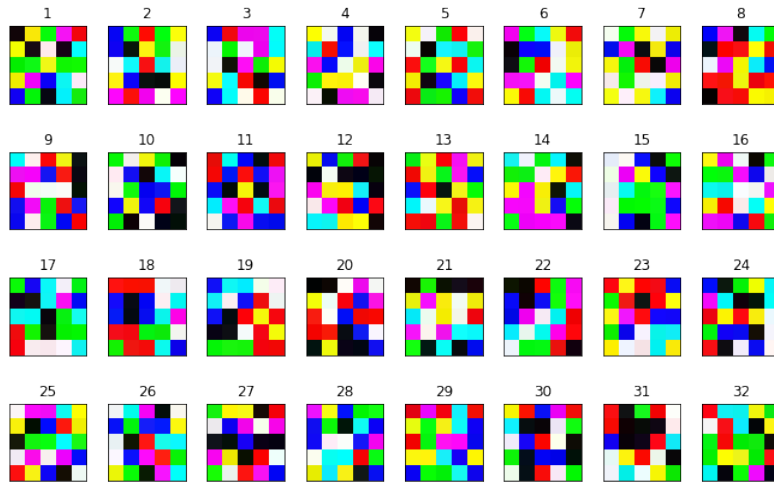


Figure 5: 5x5 filters

## 6 Batch Normalization

With batch normalization, we are feeding the network with data points that belong to the same distribution so that the training will be faster. If we feed the network with points of different distribution, then it will be slow as it will learn and unlearn features.

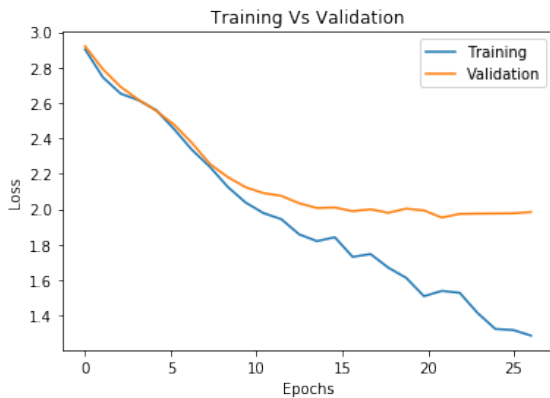


Figure 6: Loss With BN

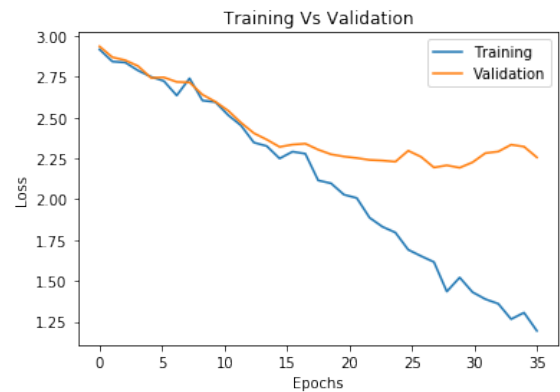


Figure 7: Loss Without BN

---

## 7 Dimensions, Parameters and Neurons

Layer	Input	Output	Parameters
conv1	64x64x3	64x64x32	5x5x3x32=2400
conv2	64x64x32	64x64x32	5x5x32x32=25600
pool-1	64x64x32	32x32x32	0
conv3	32x32x32	32x32x64	3x3x32x64=18432
conv4	32x32x64	32x32x64	3x3x64x64=36864
pool-2	32x32x64	16x16x64	0
conv5	16x16x64	16x16x64	3x3x64x64=36864
conv6	16x16x64	14x14x128	3x3x64x128=73728
pool-3	14x14x128	7x7x128	0
FC1	6272	256	1,605,632
OUT	256	20	5120

Layer	Parameters	Neurons
convolution	194,272	490,112
Fully Connected	1,611,028	276

## 8 Extra

### 8.1 Data Augmentation

Techniques used

- salt and pepper
- flipping (left-right, up-down, transpose)
- Rotation(90,180,270)

## 9 Kaggle Accuracy

Kaggle accuracy is **52.577**