Assignment 4

1. Introduction

To develop a model that takes training, test dataset of Street View House Numbers and classifies the house numbers. Here the program process the data, build and train the network, and evaluate its accuracy on the test dataset.

2. Network Architecture

We have implemented Convolution Neural Network (CNN) architecture to build the model. The architecture consists of *input layer, convolution layers, pooling, flattening layer and output layer*.

The model consists of 6 convolution layers .The convolution layer consists of convolution, activation and batch normalization. Then pooling is applied and then dropout is performed in order to reduce the over fitting .Then the flattening is applied which results into the output layer that has outputs.

Here the *activation function used is ELU*: ELU has improved learning characteristics compared to the other activation functions. In contrast to RELU, ELUs have negative values which allow them to push mean unit activations closer to zero.

Regularization involves dropout and batch normalization techniques to reduce overfitting. Dropout randomly drops units from neural network during training. Batch normalization normalizes the activation of the previous layer at each batch, i.e. applies a transformation that maintains the mean activation close to 0 and the activation standard deviation close to 1.

3. Justification for selecting this Architecture:

We have implemented CNN as it is able to successfully capture the Spatial and Temporal dependencies in an image through the application of relevant filters. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights.

Pooling can be applied using CNN. Here we have used maximum pooling that returns the maximum value from the portion of the image covered by the Kernel.

4. Training the Network:

The *loss function implemented here is categorical_crossentropy: It* is the default loss function used for multi-class classification problems. Cross-entropy will calculate a score that summarizes the average difference between the actual and predicted probability distributions for all classes in the problem.

The *optimizer used here is rmsprop:* It is the average of the squared gradients for each weight. And then we divide the gradient by square root the mean square.

The metrics used here is accuracy to judge the performance of the model.

The number of epochs used is 125 . The evaluation is run after each epoch.

5. Data pre-processing steps

Here data preprocessing is achieved by using the function. The image data is generated by transforming the actual training images by rotation, crop, shifts, shear, zoom, flip, reflection, normalization etc.

6. Github Link:

https://github.com/yaminikosaraju/MLAssignment-4