**Convolution Neural Networks (CNN)**

**Instructions:**

Please share your answers filled in-line in the word document. Submit code separately wherever applicable.

Please ensure you update all the details:

**Name: DHEERAJ MISHRA Batch ID:**  DS\_01072021

**Topic: Convolutional Neural Networks**

**Grading Guidelines:**

**1. An assignment submission is considered complete only when correct and executable code(s) are submitted along with the documentation explaining the method and results. Failing to submit either of those will be considered an invalid submission and will not be considered for evaluation.**

**2. Assignments submitted after the deadline will affect your grades.**

**Grading:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ans** | **Date** |  |  | **Ans** | **Date** |
| Correct | On time | A | 100 |  |  |
| 80% & above | On time | B | 85 | Correct | Late |
| 50% & above | On time | C | 75 | 80% & above | Late |
| 50% & below | On time | D | 65 | 50% & above | Late |
|  |  | E | 55 | 50% & below |  |
| Copied/No Submission |  | F | 45 |  |  |

* **Grade A: (>= 90):** When all assignments are submitted on or before the given deadline.
* **Grade B: (>= 80 and < 90):** 
  + When assignments are submitted on time but less than 80% of problems are completed.

(OR)

* + All assignments are submitted after the deadline.
* **Grade C: (>= 70 and < 80):** 
  + When assignments are submitted on time but less than 50% of the problems are completed.

(OR)

* + Less than 80% of problems in the assignments are submitted after the deadline.
* **Grade D: (>= 60 and < 70):**
  + Assignments submitted after the deadline and with 50% or less problems.
* **Grade E: (>= 50 and < 60):** 
  + Less than 30% of problems in the assignments are submitted after the deadline.

(OR)

* + Less than 30% of problems in the assignments are submitted before the deadline.
* **Grade F: (< 50):** No submission (or) malpractice.

1. **Business Problem**
   1. **What is the business objective?**
   2. **Are there any constraints?**
2. **Work on each feature of the dataset to create a data dictionary as displayed in the below image:**



**2.1. Make a table as shown above and provide information about the features such as its data type and its relevance to the model building. And if not relevant, provide reasons and a description of the feature.**

1. **Data Pre-processing**

**3.1 Data Cleaning, Feature Engineering, etc.**

**3.2 Outlier treatment if applicable.**

1. **Model Building**
   1. **Build a convolution neural network model.**
   2. **Train and test the model.**
   3. **Briefly explain the model output in the documentation.**



1. **Write about the benefits/impact of the solution - in what way does the business (client) benefit from the solution provided?**
2. **Use Tensorflow for this assignment. Depending on your system configuration, either Tensorflow GPU or Tensorflow CPU versions.**

**Problem Statement: -**

1. Build a CNN model on CIFAR-10 dataset by applying few regularization techniques like drop out and data augmentation
2. BUSINESS OBJECTIVE:-

Maximize followup wrt time

1. DATA UNDERSTANDING:-

CIFAR-10 dataset of image from tensorflow

1. DATA CLEANSING :-
2. Taking input layers as input shape 32\*32\*3 activation function as relu
3. Applying five filters as activation function relu
4. Flatten function for dimension change
5. Splitting the data to train and test images
6. Scaling the images
7. MODEL BUILDING:-
8. Model builded
9. optimizer='rmsprop'
10. loss='categorical\_crossentropy'
11. metrics=['accuracy']
12. Test accuracy = 0.6933000087738037
13. Train accuracy = 0.7607399821281433

OUTPUT:-

It is right fit model

1. BENEFITS :-

From above information we can predict for images

1. Find out the differences between Convnet filter and the Maxpool layers

Solution:

Convent filter:

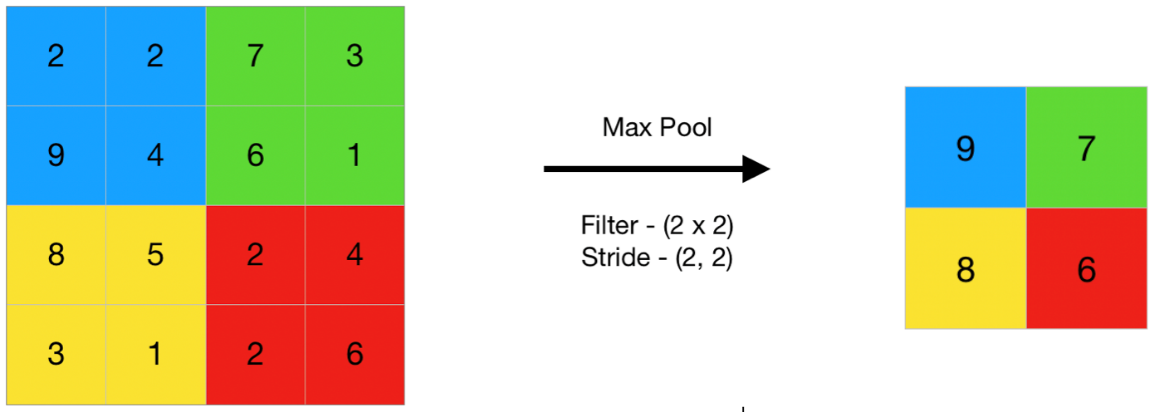
1. The convolutional layer serves to detect (multiple) patterns in multiple sub- regions in the input field using receptive fields.
2. In Convolutional neural network, the kernel is nothing but a filter that is used to extract the features from the images. The kernel is a matrix that moves over the input data, performs the dot product with the sub-region of input data, and gets the output as the matrix of dot product
3. A convolution is the simple application of a filter to an input that results in an activation. Repeated application of the same filter to an input results in a map of activations called a feature map, indicating the locations and strength of a detected feature in an input, such as an image.

**Pooling layer:**

1. The pooling layer serves to progressively reduce the spatial size of the representation, to reduce the number of parameters and amount of computation in the network, and hence to also control over fitting. There are no parameters learned in pooling layer.

b) The intuition is that the exact location of a feature is less important than its rough location relative to other features

**c)** Max pooling where max. value is considered.



1. Max pooling is done to in part to help over-fitting by providing an abstracted form of the representation. As well, it reduces the computational cost by reducing the number of parameters to learn and provides basic translation invariance to the internal representation.
2. If the input of an image is 64x64x3 which has been convolved by 10 5x5 filters with stride 1 and padding 2:
3. How many activation maps are obtained?

Solution:

Activation maps = no. of filters =10

1. What is the size of the activation maps?

Solution:

(input size-filter size + 2\*pad)/stride

Size of activation maps = [( 64-5+2\*2)/1 ] +1=64

ie. 64x64

1. How many parameters are calculated?

Solution:

[((f\*f\*d)+b)\*no. of filters],f =filter size ,b =bias=1

(5\*5\*3)+1=76 parameters for each filter

Total= 76\*10 =760 parameters

1. What are the different techniques that need to be applied to overcome the issue of overfitting? Provide brief explanations of how these techniques address the issue.
2. Regularization:

Regularization optimizes a model by penalizing complex models, therefore minimizing loss and complexity. Thus this forces our neural network to be simpler. L1 & L2 are the most common ones.

1. Weight initialization:

Weight initialization sets up the weights vector for all neurons for the first time before the training process begins. Choosing the correct weights is crucial because we want to get as close as possible to the global minimum of our cost function in an adequate amount of time

1. Dropout regularization:

Dropout regularization ignores a random subset of units in a layer while setting their weights to zero during that phase of training.

1. Weight constraints:

A weight constraint checks the size of the network weights and rescales them if the size exceeds a predefined limit. The weight constraint works as required.