**LAB-07**

**NAME-MAHIKA JADHAV REG NO-21BDA53**

**1.Have you come across Grid Search Cross Validation? Fit any two models covered in previous classes and optimize them using Grid search CV.**

**Ans:**

**No ,I haven’t come across Grid search CV.**

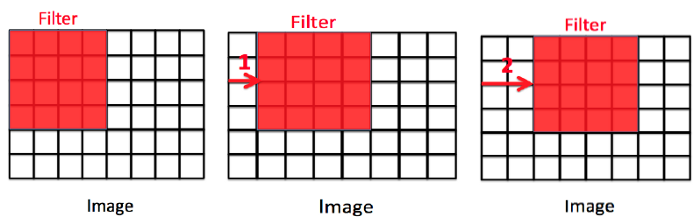
**This is the first time that I’ve tried.**

**The accuracy is reduced in both random forest and decision tree after applying grid search cv.**

**2.What is Stride, Padding & Pooling? Explain with an example.**

**Strides**

When the array is created, the pixels are shifted over to the input matrix. The number of pixels turning to the input matrix is known as the strides. When the number of strides is 1, we move the filters to 1 pixel at a time. Similarly, when the number of strides is 2, we carry the filters to 2 pixels, and so on. They are essential because they control the convolution of the filter against the input, i.e., Strides are responsible for regulating the features that could be missed while flattening the image. They denote the number of steps we are moving in each convolution. The following figure shows how the convolution would work.

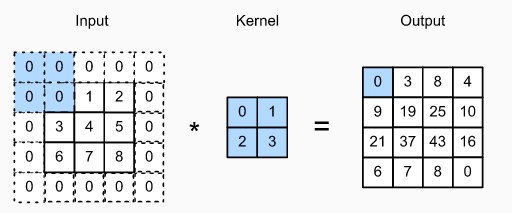
****

## Padding

**The padding plays a vital role in creating CNN. After the convolution operation, the original size of the image is shrunk. Also, in the image classification task, there are multiple convolution layers after which our original image is shrunk after every step, which we don’t want.**

**Secondly, when the kernel moves over the original image, it passes through the middle layer more times than the edge layers, due to which there occurs an overlap.**

**To overcome this problem, a new concept was introduced named padding. It is an additional layer that can add to the borders of an image while preserving the size of the original picture. For example:**

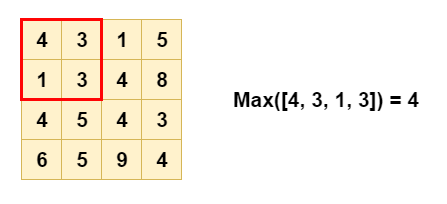


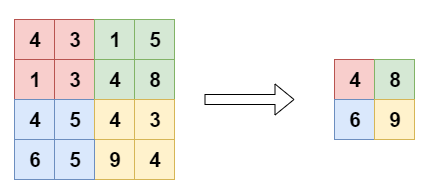
## Pooling

The pooling layer is another building block of a CNN and plays a vital role in pre-processing an image. In the pre-process, the image size shrinks by reducing the number of parameters if the image is too large. When the picture is shrunk, the pixel density is also reduced, the downscaled image is obtained from the previous layers. Basically, its function is to progressively reduce the spatial size of the image to reduce the network complexity and computational cost. Spatial pooling is also known as downsampling or subsampling that reduces the dimensionality of each map but retains the essential features. A rectified linear activation function, or ReLU, is applied to each value in the feature map. Relu is a simple and effective nonlinearity that does not change the values in the feature map but is present because later subsequent pooling layers are added. Pooling is added after the nonlinearity is applied to the feature maps. There are three types of spatial pooling:

### 1. Max Pooling

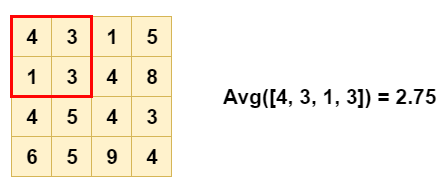
Max pooling is a rule to take the maximum of a region and help to proceed with the most crucial features from the image. It is a sample-based process that transfers continuous functions into discrete counterparts. Its primary objective is to downscale an input by reducing its dimensionality and making assumptions about features contained in the sub-region that were rejected.



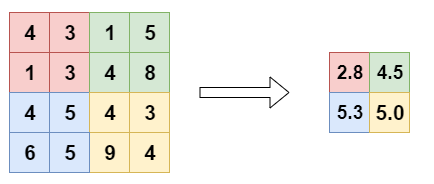


### 2. Average Pooling

It is different from Max Pooling; it retains information about the lesser essential features. It simply downscales by dividing the input matrix into rectangular regions and calculating the average values of each area.



Source: [Link](https://miro.medium.com/max/448/1*vKn1L0Im3B9yFWCzhNOQSw.png)



Source: [Link](https://miro.medium.com/max/434/1*SUuwziBInxQ_Z1Vl9y0_CQ.png)

### 3. Sum Pooling

It is similar to Max pooling, but instead of calculating the maximum value, we calculate the mean of each sub-region.

4.What is overfitting? How to overcome overfitting in an ML model?

Ans: It is a common pitfall in [deep learning](https://www.v7labs.com/blog/deep-learning-guide) algorithms in which a model tries to fit the[training data](https://www.v7labs.com/blog/quality-training-data-for-machine-learning-guide)entirely and ends up memorizing the data patterns and the noise and random fluctuations.

These models fail to generalize and perform well in the case of unseen data scenarios, defeating the model's purpose.

**How to Avoid Overfitting :**

1. Collect/Use more data. This makes it possible for algorithms to properly detect the signal to eliminate mistakes. ...
2. Data augmentation. ...
3. Simplify the data/Remove features. ...
4. Ensemble Learning. ...
5. Cross-validation. ...
6. Early stopping. ...
7. Regularization.