1)Please explain the pros and cons of Instance-Based Learning and Model-Based Learning respectively. (4points)

Instance-Based Learning =>

Advantages:

1. Instance-Based Learning method is especially appropriate because not all examples are available from the outset, but are gathered online. The new example observed in this case needs only an update to the database. That unlike global approximators, is worth noting.

2.Instance-Based learning does not suffer from interference with knowledge (Atkeson, Moore, and Schaal, 1997). That is, it does not degrade modeling efficiency for others by obtaining examples of an operating regime.

Disadvantages:

1. Definition of the distance metric to determine the validity of the examples adjacent to the question point, selection of the local approximator structure e.g. selection of polynomials of varying degrees), selection of the number of examples to be used for each local model identification and their relative weight, and, finally, selection of the characteristics to be considered.

2.A large amount of memory to store the data, possibly, and with the fact that each request for information requires starting from scratch to define a local model.

Model Based Learning =>

Pros:

1.Model needs to be trained only once.

2. After model is trained, data need not be stored.

Cons:

1.Outliers affect the model more significantly.

2.Model cannot adapt to new data.

2)Please draw the diagram of Convolutional Neural Networks (CNN). Then explain the functionality of each layer of CNN. Name several latest algorithms of CNN (e.g., AlexNet etc.). (10points)

The Degree of Convolution - The first layer that removes the features from the input image is Convolution. By using small input data squares for learning image attributes, Convolution maintains the relationship between pixels. This is a two-input mathematical operation, such as an image matrix and a filter or a kernel.

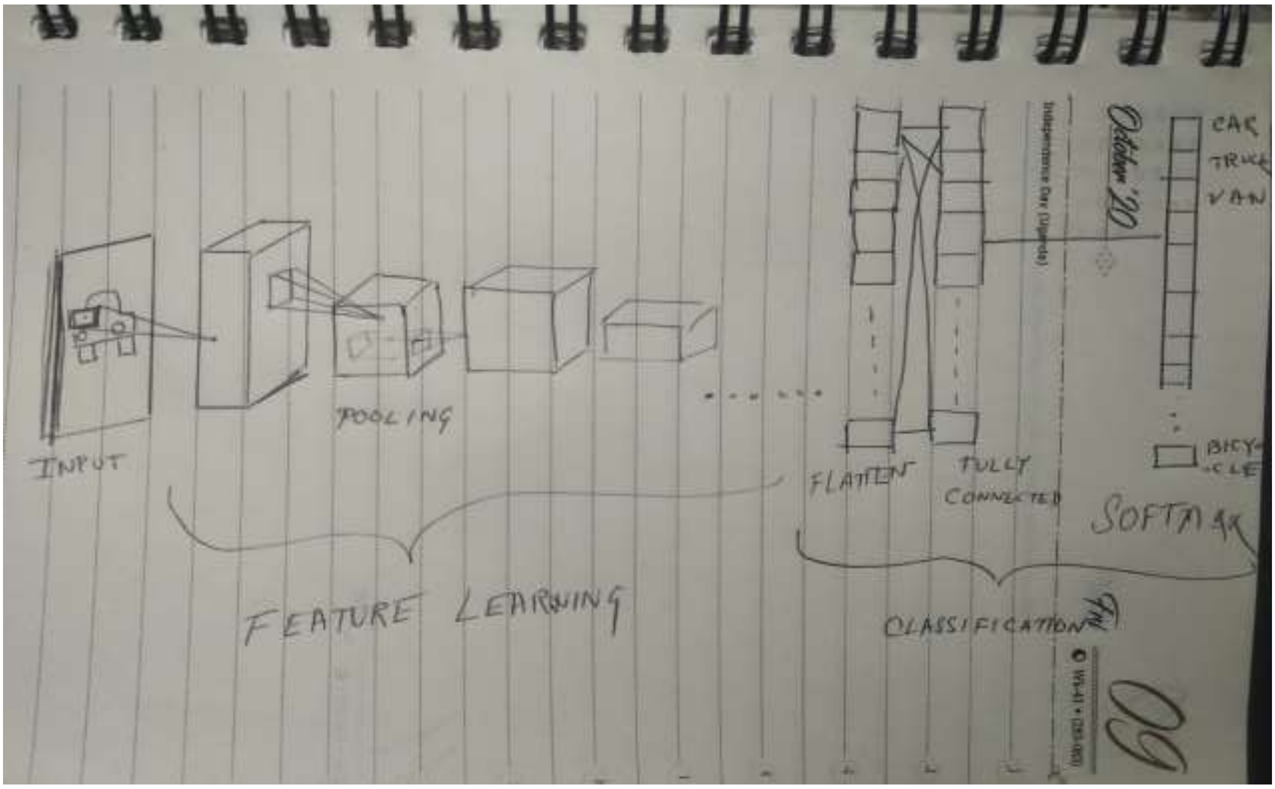
Stride- Stride is the number of pixels in the input matrix that pass through. We turn the filters to 1 pixel at a time if the phase is 1. We move the filters to 2 pixels at a time if the step is 2, and so on. The figure below shows that with a step of 2, the convolution would succeed.

Padding- In case the filter does not match the input picture exactly, we can do one of the following:

1. Pad the image with zeros (zero padding) such that it matches.
2. Drop the part of the photo where the filter did not match thereby preserving only a valid part of the picture.

Layer of pooling - The segment of the pooling layer will minimize the number of parameters when the images are too large. Spatial pooling is also referred to as subsampling or downsampling, which reduces each map's size but retains critical detail. There can be different forms of spatial pooling: Max pooling takes the largest element out of the modified feature diagram. The average pooling could also involve taking the biggest element. Definition of all the features in the Map Call as a pool of numbers.

CNN Algorithms => LeNet-5, AlexNet, VGGNet, ResNet



3)When training deep networks using Backpropagation, one difficulty is so-called “diffusion of gradient”, i.e., the error will attenuate as it propagates to early layers. Please explain how to address this problem. (6points)

When more layers utilizing such activation functions are applied to the neural networks, the loss function gradients reach zero, making the network difficult to train.

Certain triggering functions squirt a large input space into a small input space between 0 and 1, such as the sigmoid function. As a result, a major change in the sigmoid function's input can cause a small change in the output.

By the chain rule, to evaluate the derivatives of the initial layers, the derivatives of each layer get multiplied down the network (from the final layer to the initial layer). N small derivatives are multiplied together, however when n hidden layers are allowed as a sigmoid function. Thus the gradient declines exponentially when we propagate to the original layers.

A limited gradient means that each training session cannot update the weights and biases of the initial layers effectively. Since these initial layers are always critical to the identification of the core elements of input data, they may contribute to the general inaccuracy of the entire network.

Solutions for this problem:

1. The easiest approach is to use other activation functions, such as ReLU, which does not cause a slight change derivative.
2. Residual networks are another solution, since they have residual links directly to earlier layers.
3. Finally, the problem can also be overcome by batch normalization layers.