cs512 Assignment 3: Review Questions

Amit Nikam
Department of Computer Science
Illinois Institute of Technology

November 9,2020.

1 Neural Networks:

1.a In template matching interpretation of a linear classifier we consider rows of Θ^T as templates. With k rows of Θ^T we have k templates(one template per class). $\Theta^T X$ measures how well X matches with the template. High similarity indicates high membership to the particular class.
The decision boundary of a linear classifier is interpreted as a hyper-plane that splits the data into classes.

1.b To convert similarity score to probability, we use sigmoid for the last layer of the network for 2 class classification. Sigmoid gives a certain value which is useful as a output. For a K class classification we use we use softmax which also takens into account the output of the other classes.

1.c L1 loss: $Li(\Theta) = \Sigma_{\text{for each class}}$ |predicted value – known value|

L2 loss: $Li(\Theta) = \Sigma_{\text{for each class}}$ (predicted value – known value)^2

Huber loss: $Li(\Theta) = \Sigma_{\text{for each class}}\, \rho\sigma$ (predicted value – known value), where $\rho\sigma$ is quadratic for small d and linear for larger d.

Cross-entropy loss: $Li(\Theta) = - \Sigma_{\text{for each image}} \Sigma_{\text{for each class}}$ (known value) x log(predicted value)

1.d Since complex models can be overfitted(not desirable), we regularize the loss function. This helps us generalize better and the solution is stable with smaller coefficient values.

1.e We take steps in the opposite direction of the gradient in gradient descent algorithm to find a local minimum of the function.

1.f Gradient descent algorithm is implemented after processing all the examples to minimize the loss, while stochastic gradient descent is implemented after each exmaple. SGD converges faster but could be bad or wrong which is not disered, while gradient descent gives much better results. We never perform true SGD in real world, but a hybrid type can be used.

1.g Learning rate can be computed dynamically or can be manually selected by the user during implementation as a hyper-parameter. This learning rate need not be fixed and should become smaller as iterations progress to avoid overshooting and infinite learning time.

1.h While using gradient descent, we might get a local minimum and get stuck. This does not mean that the solution is optimal, there exists another minimum which can give us better results and so to get out of this local minimum, momentum is used.

1.i In back propagation algorithm, we push the input through the network to compute all intermediate node values. In backward pass, starting with end nodes we push the gradients towards the beginning. The back propagated gradient from back is multiplied by current gradients and propagated further backwards.

1.j In a fully connected layer each input unit is connected to each of the nodes in the FC layer. Each of the nodes in the FC layer generate one output each. Other the other hand, a convolution layer is one where the input unit is only processed with corresponding convolution unit in convolution layer / frame.

1.k A dropout is a regularization method used in neural network to prevent overfitting. In dropout at each training stage of FC some of the units are dropped out with a probablity of (1-P). Removed nodes are reinstated with original weights in the subsequent stages. This helps reduce the co-dependency between the nodes.

2 Convolutional Neural Networks:

2.a The convolution of this image will be $\begin{bmatrix}45,45\\54,54\end{bmatrix}$

2.b The convolution of this image will be $\begin{bmatrix}18,27,27,18\\30,45,45,30\\36,54,54,36\\28,39,39,28\end{bmatrix}$

2.c The convolution of this image will be $\begin{bmatrix}24,24\\20,20\end{bmatrix}$

2.d In convolution the filter which entend the depth of the image and match the template within the filter to the input. This only gives high values for the units that show high similarity. This is how template matching interpretation of convolution works.

2.e By using multiple convolution layers of fixed receptive fields and proper stride we can perform multiple scale analysis on the input image.

2.f To compensate for spatial resolution decrease resulting in reduced coefficients, we increase the depth such that we can keep the same number of coefficients. This helps us retain capture details as we convolve.

2.g The size of the resulting tensor will be 126x126x16.

2.h The size of the resulting tensor will be 63x63x16.

2.i In a 1x1 convoltion the filter is of dimension 1x1xdepth$_{image}$, but the output tensor is (H$_{image}$) x (W$_{image}$)x (# of filters). Thus by specifying the total number of filters we can reduce the number of channels.

2.j The early convolution layers interpret the low-level features from the images. These features are learnt as they train. While the deeper convolution layers interpret the mid level features and even slight color patterns. Deeper layers can also correlate between early layers.

2.k Pooling is done to downsample the spatial dimensions of the tensor. Various strategies like Max pooling or Average pooling can be used.

2.l The result obtained is, $R=\begin{bmatrix}1,1\\1,1\end{bmatrix}$, $G=\begin{bmatrix}2,2\\2,2\end{bmatrix}$, $B=\begin{bmatrix}2,2\\4,4\end{bmatrix}$