Deep Learning with Keras, Tensorflow and Statistical Programming Language, R

Spring 2018 - Gui Larangeira, M.S. Computational Statistics. Mentor: Dr. Eric Suess

1. Case Study: The MNIST database comprises 60,000 training examples and 10,000 test examples of the handwritten digits 0–9, formatted as 28x28-pixel matrices, with each pixel carrying a grayscale value 0-255:

2. Building a handwritten digit classifier in R:
Open-source numerical libraries such as Keras and Tensorflow are now available in the R programming language environment. We show a well-known image recognition application for the MNIST database using the Keras library in R.

3. The Statistical Machine Learning workflow:
   a) Data representation and pre-processing: Preprocess data into chosen representation and divide data into training and test datasets
   b) Define a network architecture, number of layers and elements in each layer, using Keras/TF modules
   c) Adjust network configurations, such as functions and learning rates. Train the model on training set
   d) Evaluate model on the test set and iterate back to c until satisfied

4. The heart of the Network are the layers, comprised of linear transformation and non-linear activation function:

5. The advantage of Keras/Tensorflow libraries is how easily one can define the network architecture and how portable the code is between platform, from powerful GPUs to mobile phones. The model above is represented by:

   ```r
   Library(keras)
   network <- keras_model_sequential() %>%
     layer_dense(units = 512, activation = "relu",
                 input_shape = c(28 * 28)) %>%
     layer_dense(units = 10, activation = "softmax")
   ```

6. Explain what epochs, the training loop and learning rate (steps) are in the illustration on each side

7. Ultimately, our model is able to achieve an accuracy of ___% after ___seconds and of training