CSc 84020 Neural Networks and Deep Learning
Homework 2: Classification and Prediction with DL Multi-Layer Perceptron
Andrea Ceres
Shao Liu

Dataset

Google *Quick, Draw!*
Source: https://github.com/googlecreativelab/quickdraw-dataset


Fig.1

The *Quick, Draw!* dataset is a collection of contributed drawings to the *Quick, Draw!* website. Players on the site are given a short period of time, such as 20 seconds, to draw an assigned category, such as a pond. Refer to *Fig.1* for 52 examples of such drawings.

In its totality, the dataset offers 345 classes and 50 million examples. The image files used for this assignment have been preprocessed into 28x28 grayscale Numpy bitmaps. From the full dataset, we selected 6 classes that fit a theme—bugs and critters: [*ant*, *bee*, *butterfly*, *mosquito*, *scorpion*, *spider*]. For each class, we sampled 25,000 examples, for a total of 150,000 images.

DL Multi-Layer Perceptron topology

Each pixel has values in the range [0, 255]. These features were normalized to the range [0,1] by dividing all feature values by 255.
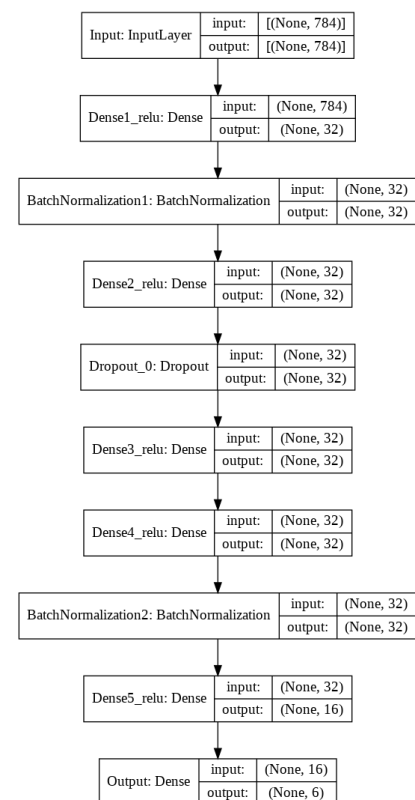
Fig.2

As shown in *Fig.2*, our base model includes five total hidden layers: four layers with 32 neurons each and one layer with 16 neurons. For each layer, the activation function used is *ReLU*, and the weight initialization used is *Glorot Normal*. Batch normalization is added after the first and fourth dense layers. Dropout is added after the second dense layer, and is set to 0.0 for the base model having no dropout. As this is a multiclass classification, the last layer's activation function is *softmax*, and the loss function we seek to minimize is *categorical crossentropy*. Given the format of the dataset and that each class is mutually exclusive, we opt to use an equivalent loss function, *sparse categorical crossentropy*. Batch size is set to 128, and 30 epochs are run for each experiment.