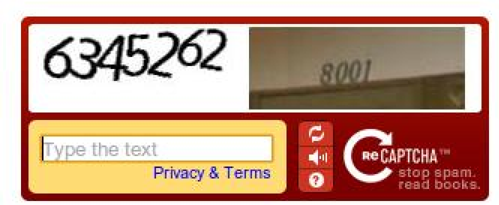
Deep Learning Capstone Project

Background

Neural networks are effective function approximators, but it turns out that the deeper the neural net is, the more complicatedare the tasks it can perform.

Among these complicated tasks is digit recognition.



Problem Statement

This project seeks to identify and output numbers which are contained in images.

Dataset and Inputs

The MNIST dataset will be used to develop the neural network model.

Once a neural net model has been determined, the SVHN dataset will be used to train the model.

Once this is done, the model will be fed images from the wild to see how it performs.

Solution Statement

This will be accomplished through the creation and training of a deep neural net to recognize numeric content within an image.

Python 2.7 and publicly-available libraries will be used to accomplish this task.
These are expected to include `numpy`, `jupyter`, `TensorFlow`, and `opencv`.

Benchmark Model

Goodfellow et al. achieved 91% whole-sequence recognition accuracy on the SVHN dataset. This project attempts to approximate, but not achieve, that performance.

Evaluation Metrics

Performance will be evaluated on a whole-sequence recognition basis, with a target accuracy of 80% or better.

Project Design

The workflow for this project will closely approximate the steps set forth in the Deep Learning Capstone Project description.

More specifically, the project design will be structured as follows:

1. **Design and test** a model architecture that can identify sequences of digits in an image.

   i. This will largely follow the work of Goodfellow et al., as they have already developed an effective an efficient model for this task.
   ii. This project will use a deep neural network as implemented by the TensorFlow library. "Deep" here refers to the fact that there are several hidden layers in the neural network.
   iii. Model development will largely focus on the MNIST dataset, as it contains simplified depictions of the digits the neural net will eventually be expected to recognize.

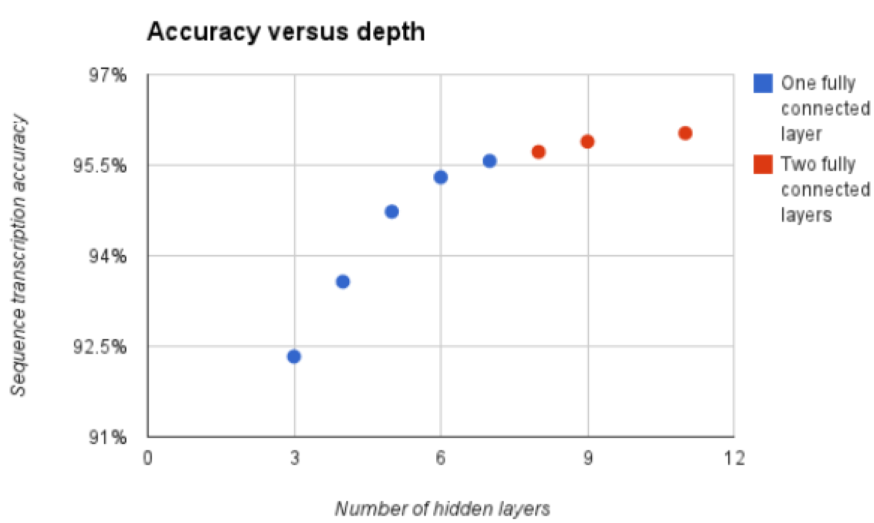
      It is expected that performance will degrade when moving to the SVHN dataset, so performance on the MNISTdataset will need to exceed 80%.

      It is expected that the neural network will employ **softmax regressions** in order to choose between competing interpretations of a given digit image.

      For training on the MNIST dataset, the already-provided training and test data split will be used.

2. **Train** a model on realistic data.

   i. This phase will focus on the SVHN dataset, and will attempt to replicate the performance achieved on the MNISTdataset, while recognizing that the digits in SVHN are more difficult to recognize.
   ii. As suggested by Goodfellow et al. (see their Figure 4 below), it is expected that additional model features, such as **convolutional layers** may be necessary in order to detect digits within the SVHN dataset, which were not necessary for success on the MNIST dataset.



3. Feed the model new number-containing **images from the wild**.

   This phase will involve one or both of the following:

   i. hand-photographing digits available locally, or
   ii. Creating (e.g. drawing) digits, either on-screen or on paper.

   After obtaining images from the wild, these images will be processed so that they are in a form which the neural net expects, and they will be input to the neural net to examine its digit-recognition performance.

4. **Localization** will be employed to display a box around detected sequences of digits.

   This will be made possible by meta-data within the SVHN dataset, and as Goodfellow et al. suggest, will likely require additional hidden layers to perform the localization task.