STA 221: Homework 3

- Homework due in Canvas: 06/03/2020 by 11:59PM. Please follow the instructions provided in Canvas about homeworks, carefully.

Q1: Image Classification Using Deep Features (15 Points)

In the housing price prediction problem in a previous assignment , we tried some manual feature engineering and higher-order features constructed manually. In this problem, we will play with the most powerful *automatic* feature engineering tool in the machine learning field – deep neural networks. As we saw in the class, deep networks learn *layer-wise representation of the input* and is particularly useful for computer vision problems such as image classification. For example, the following deep neural network model can achieve greater than 95% accuracy on MNIST classification.

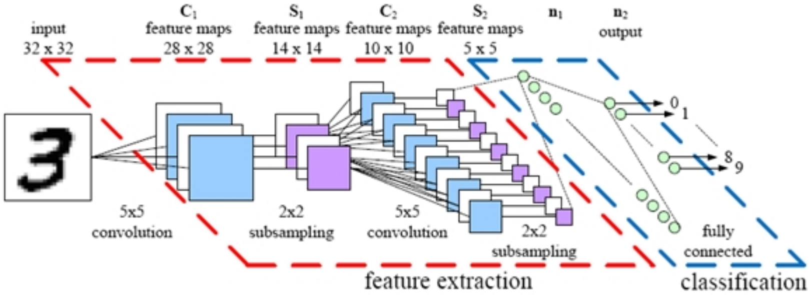

Figure 1: Deep neural network is mostly about feature engineering.

Deep neural nets are useful but painfully slow to train. We have trained a deep neural network (CNN; similar to the above architecture) on a large set of 32 × 32 RGB images for you. With a trained neural network, we can extract features from any new image of the same dimensions. From a different set of 32 × 32 RGB images, we have extracted deep features using the trained deep neural network and stored them in `DeepFeature.RData`. Note that this data is in RData format and hence use `pyreadr` to import this dataset in python. Load this RData file and you will see two matrices `deep.features`, `image.array` and a vector `label`. Every row of the matrices represent an image sample. The vector `label` has only two categories "cat" or "dog" with the i-th component representing the true label of the i-th image.

Load and split the data randomly into 70% training data and 30% test data. Use the `image.array` and `deep.features` as inputs respectively to classify the images into "cat" and "dog" using the following different approaches in PyTorch.

1. Use logistic regression to classify the raw data (`image.array`) and CNN representation (`deep.features`) and report the accuracy on test set.

2. Use SVM (with rbf kernel) to classify the raw data (`image.array`) and CNN representation (`deep.features`) and report the accuracy on test set.

3. Use a 2-layer neural network (with ReLU activation function) to classify the raw data (`image.array`) and CNN representation (`deep.features`) and report the accuracy on test set.

Q2: Summarizing ADAM (7.5 Points)

In this question, you will be required to read-up and summarize certain sections from the following paper: `https://arxiv.org/pdf/1412.6980.pdf`. ADAM is a variant of vanilla SGD that works particularly well for training deep neural networks.

- Summarize Algorithm 1 (see also Section 2) from the above paper and highlight what is the main difference from vanilla SGD.
- Summarize the main idea described in Section 3 and 7.1.

Q3: Summarizing Batch Normalization (7.5 Points)

In this question, you will be required to read-up and summarize certain sections from the following paper: `https://arxiv.org/pdf/1502.03167.pdf`. Batch Normalization is yet another trick used for efficiently training deep neural networks.

- Explain what is Internal Covariate Shift and how it is related to training of Deep Neural Networks in your own words.
- Explain what is Batch-Normalization and how it helps overcome the internal covariate shift problem.