Multi-Layer Perceptron

COMP4901K and MATH 4824B

Fall 2018

Prerequisites

- Sign up a Intel®AI DevCloud account (done in Lab5 at `https://plan.seek.intel.com//ww_en_software_registration-form-intelaidevcloudsignup_html?registration_source=salesforce&activityID=OPTY-0015197`).
- You need to install keras and tensorflow packages:

```
pip3 install --upgrade keras tensorflow
```

- Some nltk files that you should have downloaded in Lab4:

```
nltk.download('punkt')
nltk.download('stopwords')
```

1 Random Projection

We've used sparse matrix to speed up computations. However, as the documents vary in length, it's not easy to feed the sparse vector to a perceptron. In this lab, we'll use Gaussian Random Projection to reduce the dimensionality of document vectors. sklearn provides a handful tool to perform this, see the example given by the offical document:

```
import numpy as np
from sklearn import random_projection
X = np.random.rand(100, 10000)
transformer = random_projection.GaussianRandomProjection()
X_new = transformer.fit_transform(X)
X_new.shape
```

2 Keras

Keras makes it easier to develop a neural network model.

Some basic keras modules:

- keras.models.Sequential: is a linear stack of neural network layers. Create one by using

```
model = Sequential()
```

- keras.layers: defines different types of neural network layers, e.g.
  - keras.layers.Dense: for an input vector x a dense layer would produce an output

$$y = g(Wx + b)$$

where W is a matrix and b is the bias term, and $g(x)$ is the activation function. W and b is automatically created by Keras based on input(x) and output(y) dimensions.

3 Assignment: Build a Multi-Layer Perceptron

Build a Keras Sequential model, with two Dense hidden layers, each hidden layer contains 100 hidden units, using ReLu as activation function. Use SGD optimizer with learning rate as 0.1, and train 10 epochs.

4 Intel®AI DevCloud

After registration and wait for few days, you should have received an email containing your user name, node name, and an instructions address. Read "Learn", "Connect", and "Compute" section carefully to access and make use of Intel®AI DevCloud .

Warning Do not give your password, private key, or the invitation link to anyone else.

As keras may not be installed on Intel®AI DevCloud , you can install the package locally.

```
pip3 install --user --upgrade keras nltk
```

Some useful shell commands:

- connect remote server: ssh
- file transfer: sftp, scp, rsync, or a graphical user interface(GUI) tool FileZilla.
- `https://www-xray.ast.cam.ac.uk/~jss/lecture/computing/notes/out/commands_basic/`

(Optional)After finish this lab, you can try to upload the program together with data files to Intel®AI DevCloud , and submit a job using the given script launch.sh.

```
rsync -r lab6 colfax
ssh colfax
cd lab6
qsub launch.sh
```

Appendices

A Documentations

- Keras: `https://keras.io/`.
- sklearn random projection: `http://scikit-learn.org/stable/modules/random_projection.html`
- Shell commands: `https://www.learnshell.org/` or `https://www-xray.ast.cam.ac.uk/~jss/lecture/computing/notes/out/commands_basic/`
- (Advanced) SFTP command: `https://www.computerhope.com/unix/sftp.htm`