SPLEX TME 13

Data Integration with Deep Learning

The goal of the TME is to learn some techniques of data integration.

Data

- Molecular classification of leukemia data set of *Golub et al. 1999* contains gene expressions of 72 patients and 3562 genes.
- Breast cancer data set
- Micr-Obes data (provided during the TME)

Libraries
You will need to load at least the following packages:

```
import pandas as pd
import numpy as np
import keras
from keras.models import Sequential
from sklearn.neural_network import MLPClassifier
```

Analysis
We will use `sklearn` and `keras` which is already installed on the machines.

Repeat the same analyses for the data sets and make conclusions.

1. Using `sklearn`
   - Use a multi-layer perceptron `http://scikit-learn.org/stable/modules/neural_networks_supervised.html` to define a model (this is an example, you can choose different parameters)

     ```
     model = MLPClassifier(solver='lbfgs', alpha=1e-7,hidden_layer_sizes=(10, 5))
     ```

     to learn

     ```
     model.fit(XTrain, yTrain)
     ```

     to predict

     ```
     yTest_predicted =model.predict(XTest)
     ```

     and find its accuracy.

2. Using `keras`
   Here is a tutorial on the `keras`

   `https://keras.io/getting-started/sequential-model-guide/`

   - Look in particular how to learn MLP for binary classification
   - Below it is an example! You can freely use a different number of layers and another optimizer!

To define a model

```
model = Sequential()
model.add(Dense(32, activation='relu', input_dim=29))
model.add(Dense(2, activation='softmax'))
model.compile(optimizer='rmsprop',
              loss='categorical_crossentropy',
              metrics=['accuracy'])
```

to learn

```
history = model.fit(XTrain, yTrain,
                    batch_size=batch_size,
                    epochs=epochs,
                    verbose=1,
                    validation_split=0.1)
   score = model.evaluate(XTest, yTest,
                    batch_size=batch_size, verbose=1)
```

to predict

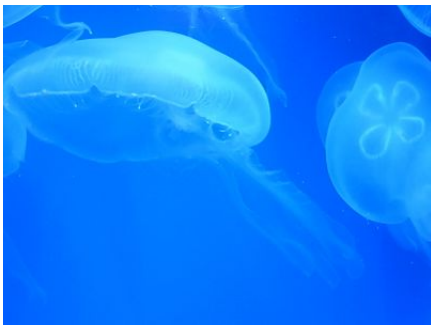
```
score = model.evaluate(XTest, yTest,
                       batch_size=batch_size, verbose=1)
print('Test score:', score[0])
print('Test accuracy:', score[1])
```

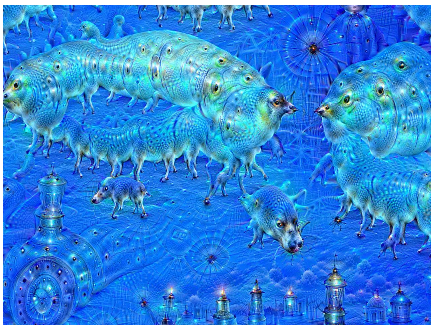
3. Perform 10 fold-cross validation to compare the MLP classifiers of `keras` and `sklearn` on the Breast Cancer and Leukemia data sets.

4. In the Micr-Obes cohort, we aim to predict from three heterogeneous data sets (environment, host, and gut microbiota) whether a patient is a High Gene count person (i.e. healthy) or a Low Gene count individual (i.e. less healthy). Do not consider patients whose class in not known.

   - Run prediction on each separate data source (environment, host, and gut microbiota) and save accuracies
   - Do prediction from all provided data (concatenate environment, host, and gut microbiota) to predict HGC/LGC.
   - What model is optimal in terms of accuracy?

P.S. Be careful when you use deep learning methods. While trying to learn the image which is on the left, you can get the image that is on the right.





https://en.wikipedia.org/wiki/DeepDream