**MLiM 25440 Homework 2 : Programming Assignment**

**Classifying Normal Samples and Tumors Samples from RNAseq profiles**
**and**
**Classifying Types of Tumors from RNAseq profiles**
**and**
**Autoencoding LINCs1000 genes**

**Due Thursday Oct 17th at Midnight.**

For each of these problems, use 5-fold cross validation and display the confusion matrix, f1 scores and other evaluation metrics in addition to the accuracy achieved.

There are two input files for each problem, one containing all the genes (60,483 features) and one containing just the protein coding genes (19,560 features). You will likely want to use some kind of normalization for these features.

The input files are **nt.coding.csv**, **nt.all.csv** and **type.coding.csv**, **type.all.csv** and **combined_rnaseq_data_lincs1000_combat.tsv**

They can be found in the shared **MLiM-Datasets/HW2** directory.

There are other files in that directory which map Ensembl IDs to Gene Symbol and Gene Types. You might find those useful when developing gene signatures. There are 18 types of cancer represented, a subset of the GDC types. Those are just labelled 1-18.

For each part please turn in your code (a python notebook is a reasonable way, but a python script is also fine), turn in output from the program (text of graphics, graphics preferred). And a short (1 paragraph write up for each part and section, explaining what you did and your critique of the results, comments on problems or difficulties and possible future approaches that might do better).

## Part 1. Normal and Tumor Match Pair Analysis (30 points)

a. Using SciKit Learn build a machine learning classifier that takes RNAseq profiles from matched normal tumor pairs and classifies the sample as Normal or Tumor. Compare the **nt.coding.csv** vs the **nt.all.csv**.

b. Using model selection methods of your choice determine which classical ML method performs best on the NT classification problem.

c. Using feature selection methods of your choice determine a < 100 gene signature that can be used to classify Normal vs Tumor.

d. Using **Keras**, build a deep learning classifier that performs the same classification task, and determine the learning curve (relationship of number of training samples to prediction accuracy) for your network, recommend using at least 10 training set sizes to estimate the learning curve.

e. **Extra Credit**: Use the TPOT autoML system to search for a better solution to part a. (**10 points**)

Part 2. **Cancer Type Classifier for 18 Common Tumor Types (40 points)**

a. Using SciKit Learn build a machine learning classifier that classifies Cancer Type from the **type.coding.csv** and **type.all.csv** files. Compare the coding vs all genes cases.

b. Using model selection methods of your choice, determine which classical ML method performs best.

c. Using feature selection methods of your choice, determine a < 100 gene signature that can be used to classify tumor type.

d. Using **Keras**, build a deep learning classifier that performs the same classification task, and determine the learning curve (relationship of number of training samples to prediction accuracy) for your network.

e. **Extra Credit 1**: Use the **AutoKeras** system to search for a better solution to part d. (**10 points)**

f. **Extra Credit 2**: Use the **Modal** system to explore active learning on this problem. (**10 points)**

Part 3. **Cancer Gene Expression Autoencoder (40 points)**

a. Using **Keras**, build an autoencoder that takes the gene expression values as input, encodes to 50 dimensions and then decodes back to the original width of the input.
b. Experiment with changing the width of the bottleneck layer (10, 20, 30, 40, 60, 80) explain what is happening to the loss value when you change the width.
c. Experiment with changing the number of layers in the encoder and decoder. What happens if the depth is too small (i.e. 1 ?) what happens when the depth is too large (~20)?
d. What are a two uses of an autoencoder for gene expression data?