Tissue classification from Gene Expression

Take-home Coding Exam

This exam aims to evaluate your practical knowledge in data manipulation, deep learning, [shallow] machine learning, data analysis, and presentation. You do not need to train state of the art models. You may opt to do **any or all** of Task 2.

Note: We will check your code. You may use “vanilla” pytorch (preferred) or tensorflow for deep learning. Do not use Pytorch lightning or Keras.

Data
Data can be downloaded from https://depmap.org/portal/download/
Features: Gene expression (CCLE_expression.csv)
Labels: tissue type (`sample_collection_site` column of sample_info.csv)

Main goal: Using the gene expression of a sample, classify its tissue type

Task 1: Data Manipulation
Download the data and preprocess *as you see fit*. Split the data into training, validation, and test sets. Process the labels by assigning integers or one-hot vectors to each tissue type. No need to do N-fold CV splits. Make sure that there is no data leakage during preprocessing.

Hint: check out drug response prediction papers (*e.g.*, Hostallero et al.’s BiG-DRP, Huang et al.’s TG-LASSO)

Task 2a: Unsupervised Deep Learning
Train an autoencoder that encodes the gene expression (feature vector) into a vector of size 512. You may choose the hyperparameters and architecture according to the validation set. You may also choose to use other types of autoencoder (sparse, denoising, variational, etc.)

Log your results (i.e. save your training/validation error curves).
After training, encode all the samples (train, val, test) and save the encodings.

Task 2b: Shallow Supervised Learning
If you finished task 2a, use the encodings as the features in this task. Otherwise, you may use the gene expressions or process it further to reduce dimensions.

Train a shallow classifier (non deep learning, *e.g.*, SVM, RF) to classify the tissue type of the sample.

Task 2c: Supervised Deep Learning
Train an MLP that takes as an input the gene expression, and classifies the tissue type.
Log your results (i.e. save your training/validation error curves).

Task 3: Analysis (open-ended)
Test your model(s) on the test set and analyze the results. You may need to train more models if comparison is needed. You may also analyze the data itself, instead of the prediction results.

Task 4: Present your model/results (schedule TBA)
Make appropriate visualizations.
Key points: design choices (e.g., model choice, preprocessing, hyperparameters, architecture, training, loss function, metrics), results, analysis.

You must convince us that you know what you are doing. We will ask questions.