Chem361: Machine Learning in Chemistry
Spring 2025, UW-Madison
Homework 4

Due date: 5pm, Apr 15, 2025

Instructions: This homework covers "Chapter 7: Generating Chemical Data - AI Generative Models" and "Chapter 8: Transforming Chemistry with Large Language Models – From Chemical to Protein Language Models". Use this PDF file as a template to complete your homework. Submit your work on time as a ZIP file to **Canvas**, including:

(1) A single PDF or Word document containing your answers (handwritten or typed).

(2) Your code in a Jupyter Notebook.

You may refer to the provided reference notebooks in chapter 7 and 8, or online resources to help you solve the questions. If needed, you can use Google Colab (`https://colab.research.google.com/`) instead of a local Python environment. For Python package functionality, consult the official documentation for detailed explanations and examples.

There are 4 homework assignments; the 3 highest scores will each contribute 20% to your final grade. ***Late homework submissions will not be accepted.***

1 Generating SMILES using Variational AutoEncoder (VAE) (50 points)

Fetch the selected QM9 dataset (`https://raw.githubusercontent.com/aspuru-guzik-group/selfies/master/examples/vae_example/datasets/0SelectedSMILES_QM9.txt`) to complete the tasks in this question. In this notebook, each compound is stored as a SMILES string in the column "**SMILES**".

In class, we went through the reference notebook `Reference_Ch7_VAE_colab.ipynb`, which demonstrated how to train a VAE using RNN as encoder and decoder for SMILES generation. In this question, we will train the model on compounds containing exactly 8 heavy atoms.

1.1 Please use RDKit to calculate the number of heavy atoms in each compound from the QM9 dataset and report how many compounds contain exactly 8 heavy atoms. **(10 points)**

1.2 Based on Q1.1, randomly split the dataset into training and validation sets using a 90:10 ratio. Identify a set of hyperparameters (including embedding dimensions, number of layers, learning rate, training epochs, etc.) that reduce the **cross-entropy (CE)** loss of the validation set to below 0.9. Plot the CE loss and KL divergence (KLD) loss against epochs for both the training and validation processes. (Hint: You can reduce the model size to save training time, as the size of the training dataset is reduced). **(20 points)**

1.3 Continuing from Q1.2, randomly select one compound from the validation set, pass it through the trained model, and report its SMILES and mean vector in the latent space. **(10 points)**

1.4 Continuing from Q1.2, compute the mean vector of the Gaussian distribution in the VAE latent space for each compound in the validation set. Report the average of these mean vectors across all compounds.**(10 points)**

2 Predicting peptide activities using Protein Language Models (50 points)

Fetch the training dataset (https://raw.githubusercontent.com/xuhuihuang/uwmadisonchem361/main/CNN_training_data.csv), which has been preprocessed based on data from **Machine learning designs new GCGR/GLP-1R dual agonists with enhances biological potency** (https://www.nature.com/articles/s41557-024-01532-x). Please refer to the reference notebook `Reference_Ch8_plm_colab.ipynb`.

2.1 Please tokenize one peptide sequence from the dataset using the tokenizer of **esm2_t6_8M_UR50D**. Explain the difference between the length of tokens and the length of token ids. **(10 points)**

2.2 Please pass the tokenized sequence from Q2.1 to the **esm2_t6_8M_UR50D** model. Report the dimensions of model output and share your understanding about the output. (Hint: The output is a dictionary containing two keys: "last_hidden_state" and "pooler_output". Please explain separately). **(10 points)**

2.3 Please build your own model to apply **esm2_t6_8M_UR50D** for a downstream task of predicting the log EC50 of peptide agonist binding to GCGR. Report the number of parameters contained in your model. **(10 points)**

2.4 Please report the number of trainable parameters contained in your model. **(5 points)**

2.5 Continuing from Q2.3, randomly split the dataset into training and validation sets using a 90:10 ratio. Train the modified CNN model using the AdamW optimizer with a learning rate of 0.001, a batch size of 10, and 100 epochs. Please report the MSE and R^2 values of the predicted log EC50 of GCGR on the validation set. **(15 points)**