CH5650 Assignment

Ayesha Ulde
MM19B021

February 25, 2022

1 Introduction

The objective of this assignment is to predict the radius of gyration of a polymer which is a sequence of styrene and stearate monomers. The data set provided has 3030 sequences, in which the two monomers are represented by the numbers '1' and '2', and the computed radius of gyration for each sequence is given at the end of their respective file. The model can be found in the Jupyter notebook *model.ipynb*.
The following sections describe my approach to fit a neural network to the given data set for predicting the radius of gyration and the results obtained.

2 Model Building Approach

2.1 Data Pre-Processing

2.1.1 Data Augmentation

The radius of gyration of a polymer is independent of the beginning and ending of the polymer sequence. To introduce this invariance, 500 sequences were randomly chosen from the data set and reversed while keeping their respective radius of gyration constant. These reversed sequences were added to the data set and the model was trained on it. For the same model architecture, it was observed that the addition of the reversed sequences gave a higher R^2 score and lower Mean Absolute Error (MAE) and Mean Squared Error (MSE).

2.1.2 Feature Scaling

The input sequences were scaled using the $StandardScaler$ class in the $scikit-learn$ package. Model trained on scaled data outperformed the model trained on unscaled data. Note that $StandardScaler$ was chosen over $MinMaxScaler$ because it gave a higher R^2 score and lower MAE.

2.2 Model Architecture

The machine learning model is a feedforward neural network. It has five layers: an input layer which has 256 nodes, a dropout layer with 0.1 dropout rate, a hidden layer having 500 nodes, another dropout layer with 0.1 dropout rate and an output layer which has one node. The activation function used throughout the model is ReLu.

2.3 Model Hyperparameterization

It was observed that using Keras Tuner, the hyperparameter tuner in Keras, did not give significant improvements in errors and R^2 score. Therefore, it was not used in this model.

2.4 Model training

The model was compiled using MAE as the loss function and Adam optimizer. The metrics used were MAE and MSE. Using MAE as the loss function gave a higher R^2 score as compared to the models that used MSE as the loss function.
The neural network was trained for 500 epochs with a batch size of 32. The train:test split is 0.67:0.33. One-fifth of the training set was used to validate the model.

3 Results

The model gives the following predictions on test and train data sets. In the plots below, Y train and Y test are the radii of gyration in the train and test data sets respectively, and Y pred is the radius of gyration predicted by the model.

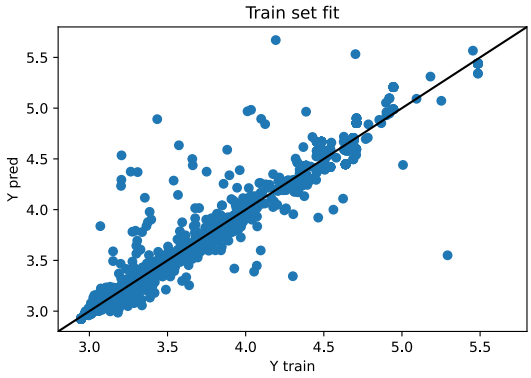

Figure 1: Model predictions on the train data set

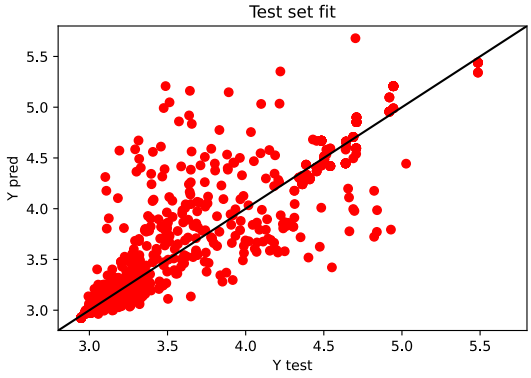

Figure 2: Model predictions on the test data set

The loss and MSE plots are given below. It can be observed that even though the MSE converges after 250 epochs, the MAE starts converging only after 450 epochs.

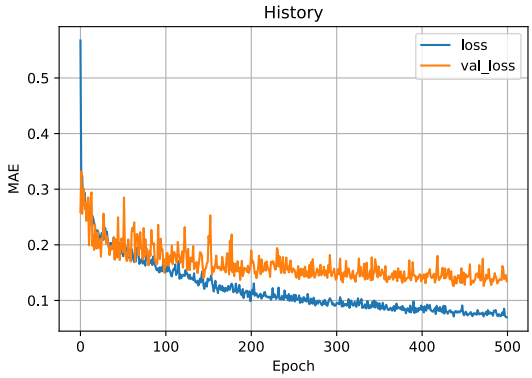

Figure 3: Plot of loss function (MAE)

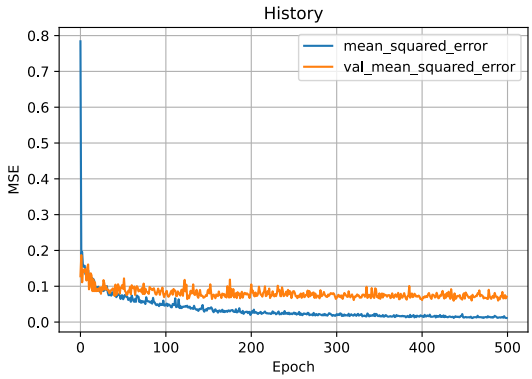

Figure 4: Plot of the MSE

The R^2 scores obtained for train and test data sets are **0.953** and **0.820** respectively. The errors for train and test data sets are tabulated below.

Error	Train set	Test set
MSE	0.018	0.072
RMSE	0.134	0.269
MAE	0.062	0.141