Full report from the project

The application starts at the main method inside the Application.java file. In the Application class there's the NeuralNetwork instance that is initialised with a *new Data instance*, the *number* of *hidden layers* and the *hidden layer size*. Going through in this method we see that it trains the NeuralNetwork for MAX_EPOCH times. At the end, the application will read the actual values from the test file.
The actual NeuralNetwork is defined in the NeuralNetwork class. In the constructor it needs to initialise input layers and hidden layers as well. When it's initialising input layer it's just adding to n_layers a new Node instance. When it's initialising the hidden layer it needs to iterate over all hidden layer configuring it as a new H_Layers and adding H_Units on each one.
In order to train the neural network it need to use the train method. It goes through every Y inside the data pack and calculates the foreword neurone with the purpose of getting it calculated. After that if goes in back propagation calculating the error, the output weights, the propagate hidden layer weights and performs improvements. To perform the entire back propagation it needs to loop through every hidden layer calling the perform method.
After the back propagation it calculates the global error increasing it by the o_Units value then it clears inputs to avoid inaccuracy