## Advanced ML&NNs

## Coding Assignment

### 

**Problem 1.** Consider Jupyter notebook Classifying-movie-reviews.ipynb we discussed in class. Run provided notebook, if anything needs fixing, please fix. Use RMSPROP optimizer.

**(10%)**

**ANS.**

**CODE for QUESTION 1, 2, 3**

**Done in the HTML FILE and ipynb file attached in the folder.**

**Problem 2.** Above model is overfitting early. Please add L2 and L1 regularizations to the first and the second dense hidden layer. Run the model with following pairs of L1, L2 parameters: (0.01,0.01), (0.02,0.02), (0.01,0.02) and (0.02,0.01). Compare accuracy and loss function plots with those you generated in Problem 1. Report whether applied regularizations affected the overfitting and if yes in what way.

**(20%)**

**ANS.**

**There is no clear evidence of overfitting being mitigated by regularization in this case as the model without regularization is also not overfitting.**

**Problem 3.** Let us do some more tests with Classifying-movie-reviews.ipynb notebook. This time, in order to fight overfitting instead of regularization add a Dropout layer after the first and the second hidden Dense layer. Run tests with Dropout layers which pass 25% of signal from the preceding Dense layers and 10%. Compare accuracy and loss function plots with those you generated in Problem 1 and 2. Report whether applied Dropout layers affected the overfitting and if yes in what way.

**(20%)**

**ANS.**

**25% Dropout Rate:**

**Training Loss and Accuracy: The training loss continuously decreases, and the accuracy improves, indicating the model is learning the training data well.**

**Validation Loss and Accuracy: The validation loss seems to start increasing after around Epoch 6, indicating the model is starting to overfit on the training data around this point.**

**However, the overfitting is not as severe as it would be without Dropout, and the model maintains relatively high validation accuracy.**

**10% Dropout Rate:**

**Training Loss and Accuracy: The training loss continuously decreases, and the accuracy improves, indicating the model is learning the training data well.**

**Validation Loss and Accuracy: The validation loss starts increasing after around Epoch 5, similar to the model with a 25% Dropout rate. However, the validation accuracy is relatively stable, indicating the model is generalizing well to unseen data.**

**Problem 4.** Consider Jupyter notebook Predicting-house-prices.ipynb we discussed in class. Run the notebook with K-fold validation with K=3. We are reducing K to speedup your training times. Capture results and compare with results presented in the lecture notes.

**(10%)**

**Problem 5.** To the model in Problem 4, please add Dropout layers with probability 0.5 after the first and the second Dense hidden layer. Compare your training curves with those in Problem 4. Report on effects of those dropout layers, if any.

**(20%)**

**Problem 6.** Attached notebook Classifying-newswires.ipynb demonstrates multi-class classification. Run the code as provided. Then, experiment with reducing the number of hidden layers to 1 and increasing the number of hidden layers to 3. You are free to choose the number of neurons any your model. Compare training results with those of the original model and report.

**(20%)**

**Please, place your full name in the form “Last name, first name” on all submitted document.**

Your main submission should be made of one or several Jupyter notebook(s) and (its) HTML image(s). Your notebook(s) should include all results, images and comments. In this problem set, it might be convenient to submit 3 notebooks.

Please include full statement of every problem in a markdown field before you start implementing that problem.

**It is not acceptable that you describe your solution of any of these problems on Piazza.**