lec11\_speech\_analysis.ipynb and notebook lec11\_speech\_recognition.ipynb and file preprocess.py. Please keep the original files until you finish your assignment. File preprocess.py is important. Keep it in the same directory as your Jupyter notebooks. File images.zip contains images from the first notebook. Unzip and keep directory images in the directory where you run your notebooks. Those images are illustrations in the above notebooks. Also, download Jupyter notebooks scipy\_fft.ipynb and ctc\_asr.ipynb. The former is a very brief tutorial on FFT. The later illustrates CTC, Connectionist Temporal Classification alogithm which is a more powerful technique used in Automated Speech Recognition.

**Problem 1.** Use scipy\_fft.ipynb or code in lec11\_speech\_analysis.ipynb to determine real, imaginary and the absolute value of the FFT of a Gaussian function with . What happens with FFT if you change to 2 and then to 4. Draw on one diagram the Gaussian functions with three values of and on the other diagram the absolute values of respective Fourier transforms. Explain the changes in the later functions.

**(20%)**

**Problem 2.** Download the dataset referred to in the lecture notes from <https://storage.cloud.google.com/download.tensorflow.org/data/speech_commands_v0.02.tar.gz>. Expand the archive into directory data\_35. Move the directory with noise to another directory for safe keeping. Create a small directory and call it data. Move subdirectories for words: dog, right, follow, learn, and three to directory data. For files in the subdirectories of the above directory data generate NumPy arrays with respective MFCC values. Use 40 MFCC vectors per one wav file and 12 elements per MFCC vector. Do not use Deltas and energies. Create corresponding NumPy archives \*.npy for all words in the directory data. Report on the volume of your npy files.

**(20%)**

**Problem 3.** Please train the sequential model as presented in the notebook speech\_recognition\_lec11.ipynb. When selecting train and validation set, please leave a small number of wav files from every class (word) for manual examination. You are working with words which have larger number of samples and you are using a larger number of MFFC vectors per wav file. Please report whether you achieved a higher accuracy than what is reported in the downloaded notebook lec11\_speech\_recognition\_cnn.ipynb. You should make this run in Google Colab, unless you have a GPU card of your own. Just report whatever result you get. Examine your trained model with several of your set-a-side test files. Report on results.

**(30%)**

**Problem 4.** Notebook lec11\_speech\_recognition\_cnn.ipynb has images with the validation and training accuracy. Curves are inverted. Usually, the training accuracy is higher. A possible reason for the inversion of the training accuracy and the validation accuracy is that we have too many samples in the validation set that are very similar to samples in the training set. If you examine your data directories you will see that some speakers contributed up to 5 samples of the same word. SciKit-learn’s train\_test\_split() function takes random files from each directory. As the result, we have files for the same speaker falling in both training and the validation set. As a consequence the validation set has sound files which are very similar to some files in the training set and the validation accuracy is reported as very high. You could easily make train and validation sets that do not share speakers. You could write a program or you could do it manually. Just take the first 80 (70)% of files and place them in the training set. Make sure the break is not splitting any one speaker into two groups. Run your training with new training and validation sets and report whether the curves flipped their positions.

**(30%)**

Please, describe every step of your work and present all intermediate and final results in your notebook(s). Please submit a working copy of the \*.ipynb file(s) and HTML images of the notebook(s).

Your notebook should include all results, images and comments.

Please place your name on any artifact you produce and submit.