EECE 571 Course Project: Modulation Classification Using Neural Networks

Akshay Viswakumar (Student# 32971665),
Department of Electrical & Computer Engineering,
The University of Birtish Columbia

I. Introduction

THE goal of Automatic Modulation Classification (AMC) is to be able to infer the type of modulation technique by observing samples of received signals. This is a pattern recognition task and a good classifier would need to base its decision solely on key features extracted from the received signal and no other prior knowledge. A good classifier must also be robust and capable of making inferences from real world signals which are subject to degradation due to effects of channel fading and noise.

An intuitive choice of a solution to the problem of AMC would be to design a classifier that is trained using supervised learning techniques. This notion is intuitive because: (1) Supervised learning techniques continue to demonstrate time and again, their superiority over conventional algorithms and techniques for popular classification problems, given enough labeled training data. (2) At present, there is no dearth of labeled training data for this problem with plenty of datasets which may be sampled from actual over-the-air recordings or synthetically generated using tools like GNU Radio. The dataset that has been provided for this project, "RADIOML 2016.10A" is a synthetic dataset (with simulated effects of channel conditions) that was created and made available to the community by DEEPSIG [1].

The goal of this project was to design an Artificial Neural Network (ANN) solution and a Convolutional Neural Network (CNN) solution to the AMC problem. This report documents the work carried out in pursuit of the aforementioned goal. The remainder of this report is organised in the following manner. Section 2 is an analysis of the available dataset and a discussion of choices (common to both solutions) made to pre-process and segment the data. Section 3 and Section 4 deal with the ANN and CNN solutions respectively. Each of these sections start with a small background sub-section, a discussion of related work followed by design choices, observations and results. Finally, both solutions are compared in Section 5. The choice to include related work and background within the sections they are discussed in may be a little unconventional, but I believe this will greatly improve the way this report reads.

II. RADIOML Data

A. Description of Data

The dataset consists of 220,000 labelled signals. Signal samples are split and their real and imaginary components are stored separately. In this way, each signal is a 2x128 array representing 128 μ s of a received waveform sampled at 10^6 samples/second. Each signal is labelled based on both the modulation technique as well as the signal-to-noise ratio (SNR) value. There are 11 different classes based on modulation techniques (8PSK, AM-DSB, AM-SSB, BPSK, CPFSK, GFSK, PAM4, QAM16, QAM64, QPSK, WBFM). There are 20 different classes based on the SNR (-20, -18, -16, -14, -12, -10, -8, -6, -4, -2, 0, 2, 4, 6, 8, 10, 12, 14, 16, 18).

B. Pre-Processing

The dataset is available as a pickle file that stores a python data structure in a serialized manner. Data from the pickle file were loaded into Numpy Arrays. The loaded label data consisted of binary strings which aren't otherwise meaningful to a neural network. The modulation labels were converted into on-hot encoded vectors where each vector was 1x11 and each bit represented one of the 11 modulation techniques.

Normalization is carried out on feature vectors before they are fed into the respective neural networks. This will be discussed in subsequent sections since the kind of feature vectors are different between the ANN and CNN networks.

C. Partitioning via Stratified Sampling

The available dataset was split in half to form training and testing datasets consisting of 110,000 samples each. The training set was then partitioned once again where 90% of the samples went on to form the actual Training set and 10% of the samples used to form a Validation set.

Each partition of the data set was formed in a way that there was adequate representation from all modulation and SNR classes. This was to make sure that the neural network could observe nearly the same number of samples from all possible classes and that there wouldn't be any over-representation or bias for one class or the other. Refer to Figure 1 and Figure 2 which display the histograms of the partitioned dataset based on SNR and Modulation Techniques. This sort of Stratified Sampling [2] was possible because the original dataset has been designed in such a way that there is equal representation from all classes.

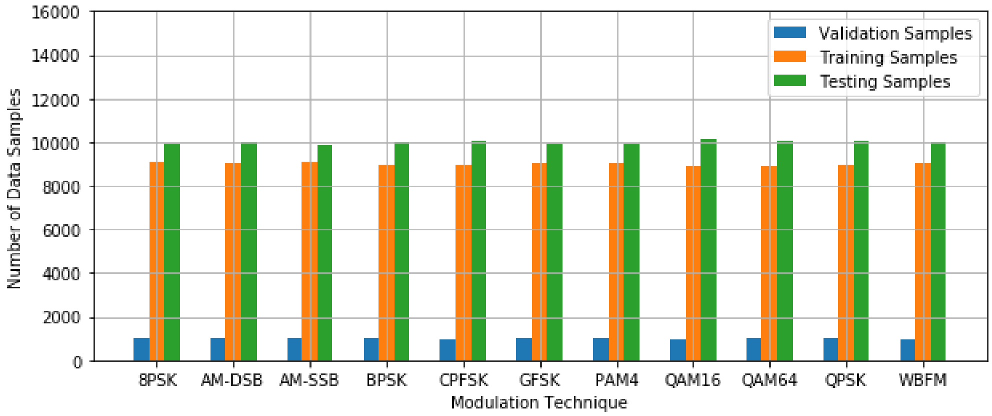

Fig. 1: Histogram of Partitioned Data, grouped by Modulation Technique

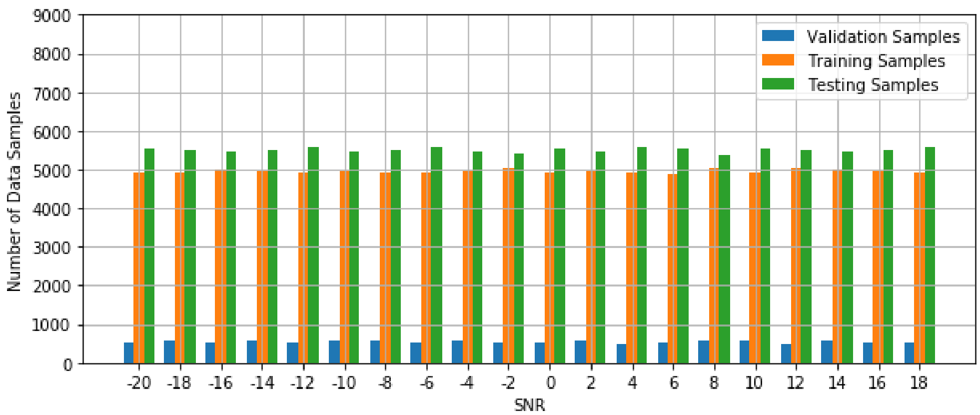

Fig. 2: Histogram of Partitioned Data, grouped by SNR

The Keras framework is able to reserve a portion of the training set for validation purposes at the time of training. However, the source of entropy for this is not within the user's control. Data was manually partitioned (using seeded pseudo-random number generators) in an effort to ensure that datasets stayed the same across multiple runs.

III. Artificial Neural Network (ANN) Solution

A. Background

B. Related Work

C. Network Design

D. Observations

E. Results

IV. Convolutional Neural Network (CNN) Solution

A. Background

B. Related Work

C. Network Design

D. Observations

E. Results

V. Conclusion

References

[1] T. O'Shea and N. West, "Radio machine learning dataset generation with gnu radio," *Proceedings of the GNU Radio Conference*, vol. 1, no. 1, 2016. [Online]. Available: https://pubs.gnuradio.org/index.php/grcon/article/view/11

[2] "Encyclopedia of survey research methods au - lavrakas, paul," Thousand Oaks, California, 2008. [Online]. Available: https://methods.sagepub.com/reference/encyclopedia-of-survey-research-methods