Neural Network Visualization

Charlie Vu, Kenneth Worden, Rachel Kitchen, Paul Noble, Chris Headley

Final Presentation December 7, 2017

1 Overview

Machine learning has seen a growth in popularity after the appearance of Convolutional Neural Networks (CNNs).

CNNs are useful for time-series data such as signals, but the inner workings of a signal-classifying network are still relatively unknown.

2 Goal

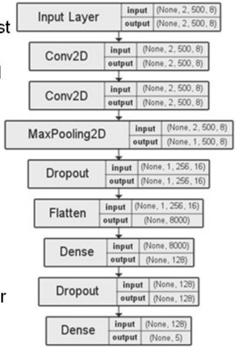
To visualize the inner processes and decisions made by a signal-classification neural network.

We hope to find a meaningful way to visualize the data to determine the network's decisions.

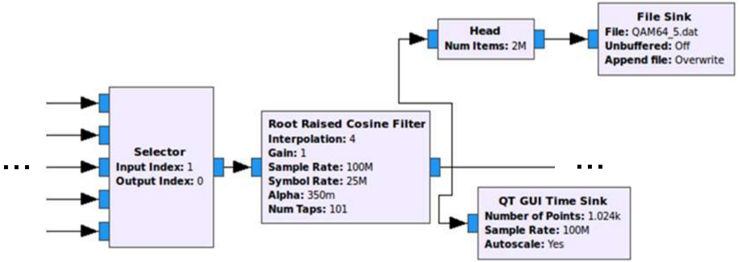
3 Network Architecture

Convolutional neural networks typically consist of two stages- a convolutional stage, and a classifier stage. Our network has convolutional layers for feature detection and a max pooling layer to reduce computed parameters.

The classifier stage consists of a flatten layer which "flattens" the 2d samples, then multiple dense layers which map the parameters to actual activations.

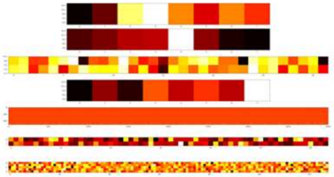


4 Data Generation



5 Visualization

Visualizing the weights from layers as a heat map:

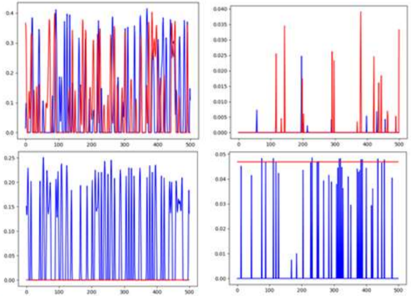


Network trained on 4 signals

There is a positive correlation between diversity of the dataset and the range of weights in the final dropout layer (6th layer) of the network model.

dropout layer from 4 signals (top) vs. 1 signal (bottom)

When the same network is trained on shuffled data, the weights are completely different.



Convolutional filters identifying QAM16 (top) and BPSK (bottom)

network activation for real data

network activation for imaginary data

Network will sometimes apply an arbitrary bias to the imaginary values in BPSK (non-existent)

6 Takeaways

Networks relate unknown signals to signals it has seen before.
The more diverse the training data is, the more diverse the model weights are.
The differences in the dropout layers show the complexity of the final selection process.
A network can learn to classify the same signals using completely different sets of weights.
The bias applied to imaginary values helps the network identify signals with imaginary components.

We have a better understanding of what the network is doing, but unfortunately we haven't found a meaningful way to apply this knowledge.

VIRGINIA TECH

Ted and Karyn Hume Center for National Security and Technology