Quirks of Training Deep Learning Systems for Future Wireless Networks

Sarunas Kalade, University of Strathclyde

linkedin.com/skalade

orcid.org/0000-0001-5512-7402

github.com/skalade

16 Sept, 2021 | UK Spectrum Policy Forum Event

Overview

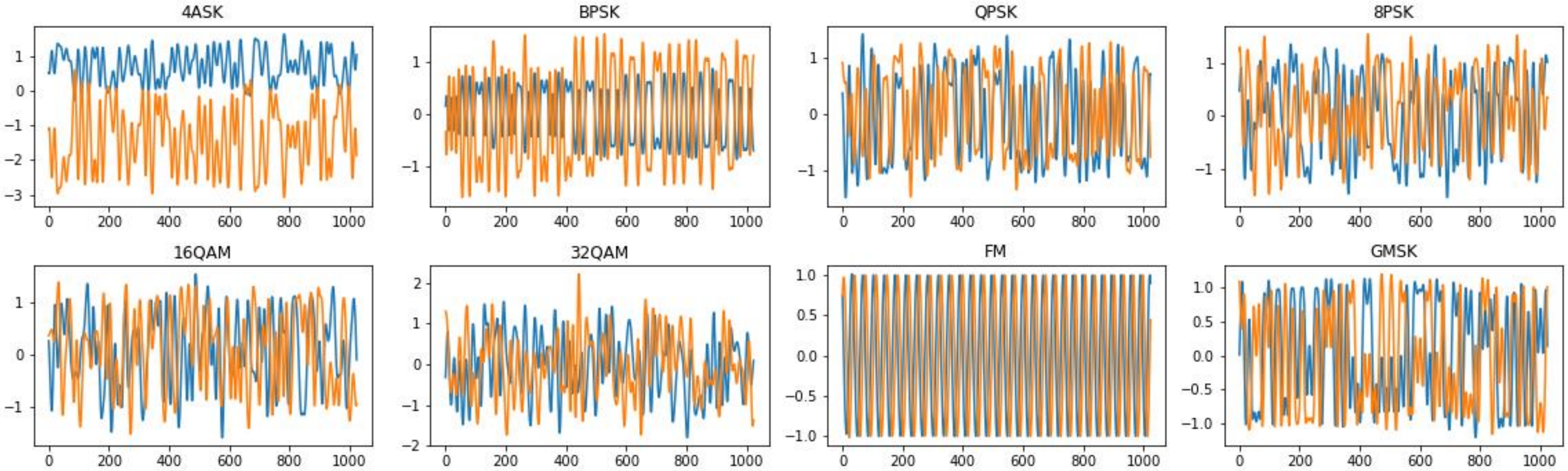
- Deep Learning For Wireless Comms Applications
  - Unique Challenges
  - Techniques
  - Physical layer mainly
- Automatic Modulation Classification
- Physical Layer Frame Sync

Deep Learning on Comms Data

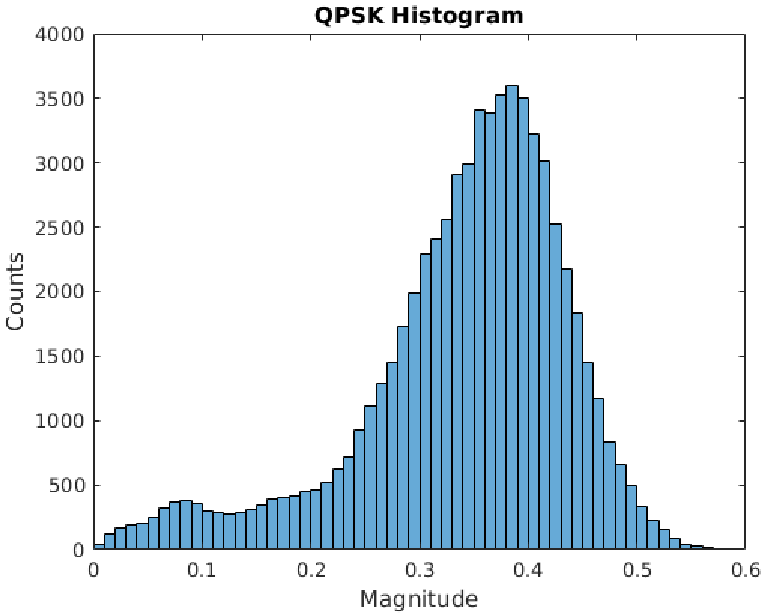
- What's unique?
  - SNR
  - Channel effects
  - Synchronization
  - Our networks have to be tiny (no dedicated cloud servers / data centers)
- What we got going for us?
  - We can generate infinite data
    - Python, MATLAB toolboxes, GNU Radio
  - SDRs enable vast amounts of data collection and in DL data is king
  - Have you seen how much Hz of instantaneous bandwidth the RFSoC boards can grab??

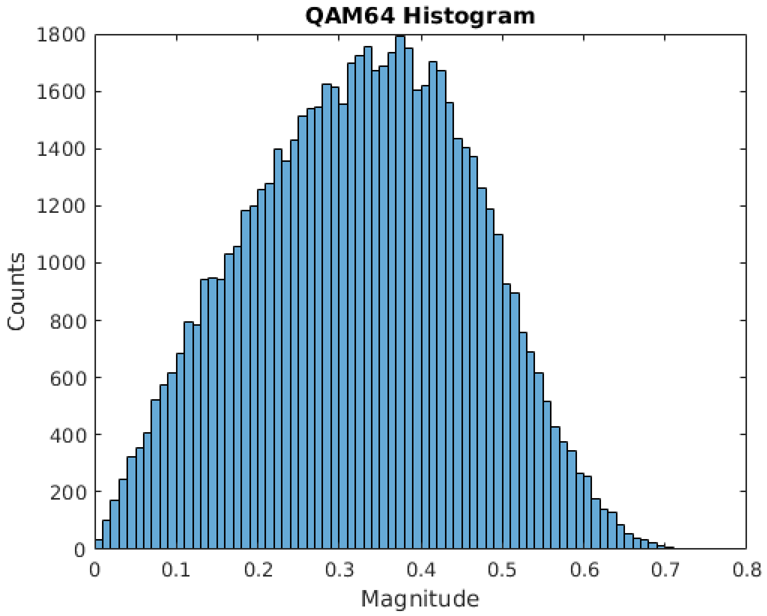
Automatic Modulation Classification

- Signals all around us, and with 6G around the corner there's only going to be more
- Next generation networks will employ some form of cognitive radio for better spectrum utilization
- We can quite successfully apply machine learning to achieve this task



Automatic Modulation Classification

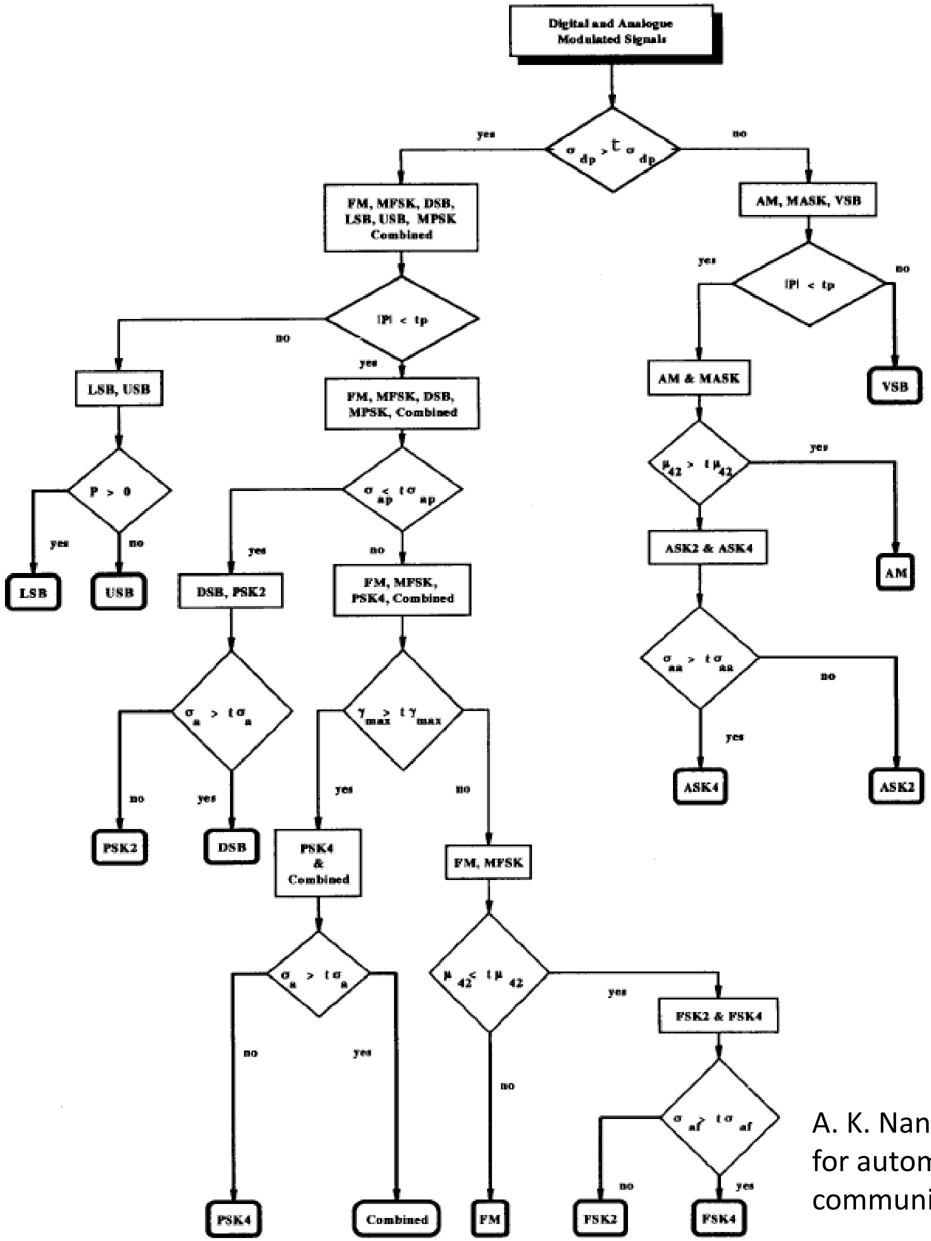




- If we are an expert we can deduce the modulation schemes using statistics
- Kurtosis of QPSK > 64-QAM
- If we have enough insights we can build a tree

Automatic Modulation Classification

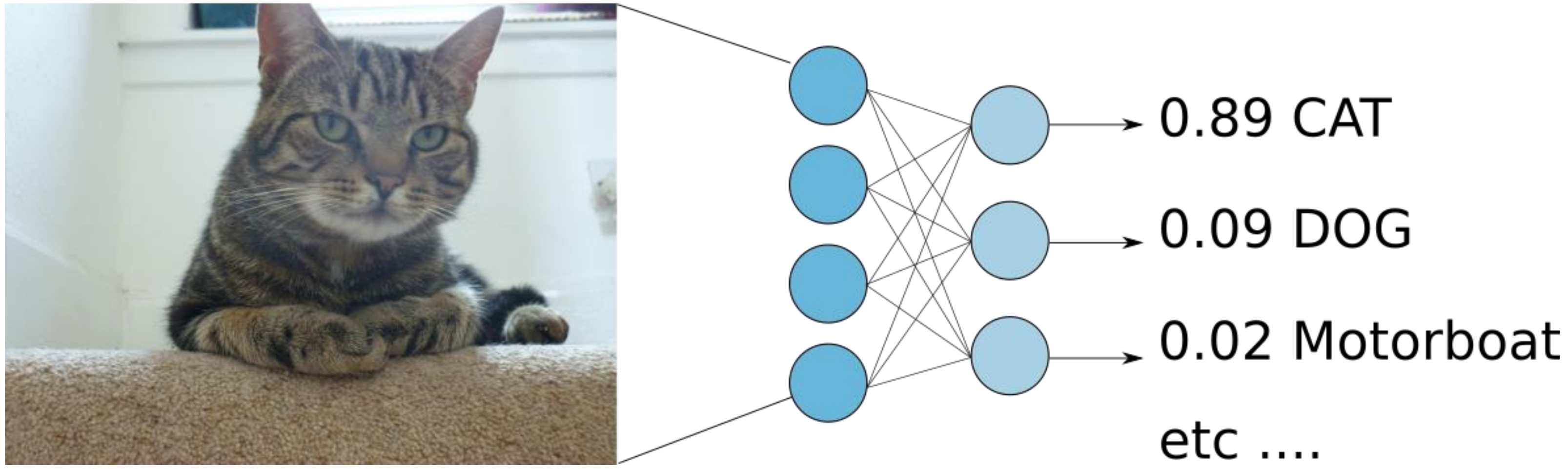
- The decision tree approach
- Manually crafted features (based on instantaneous phase, amplitude, frequency, spectral, etc.)
- Requires expert knowledge and engineering effort (costs)
- ML helps decide how to better use these features
- What about DL?



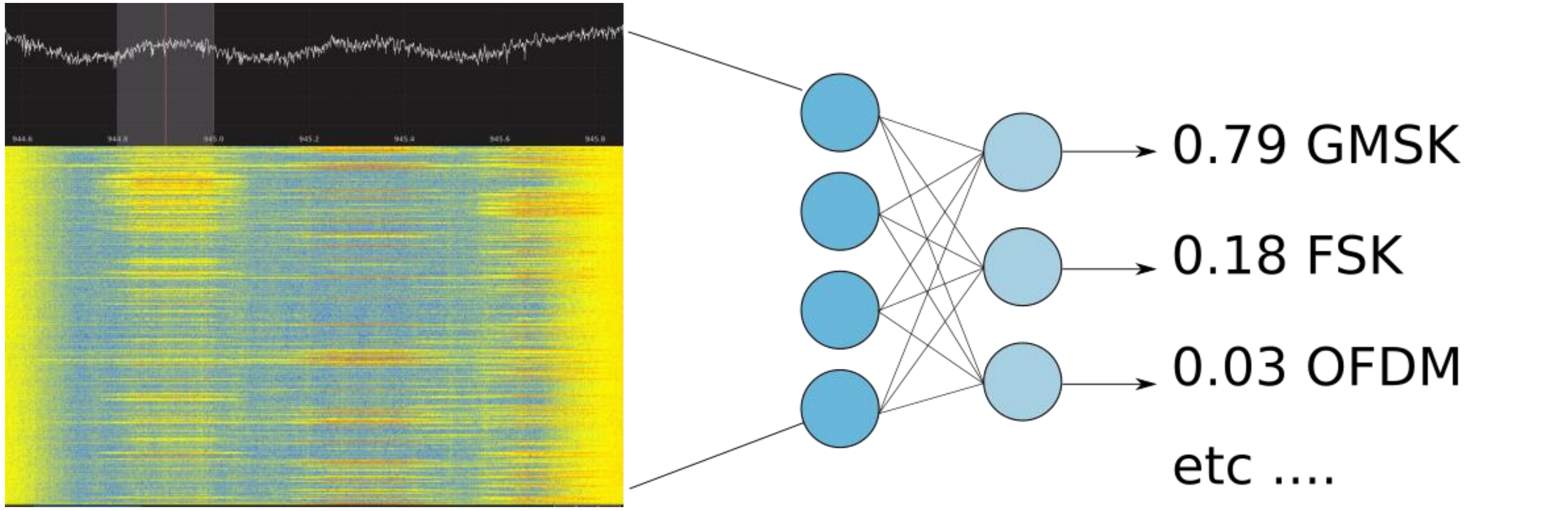
A. K. Nandi and E. E. Azzouz, "Algorithms for automatic modulation recognition of communication signals," (1998)

Automatic Modulation Classification

- Neural networks have dominated cat picture classification since 2012 (Alexnet)

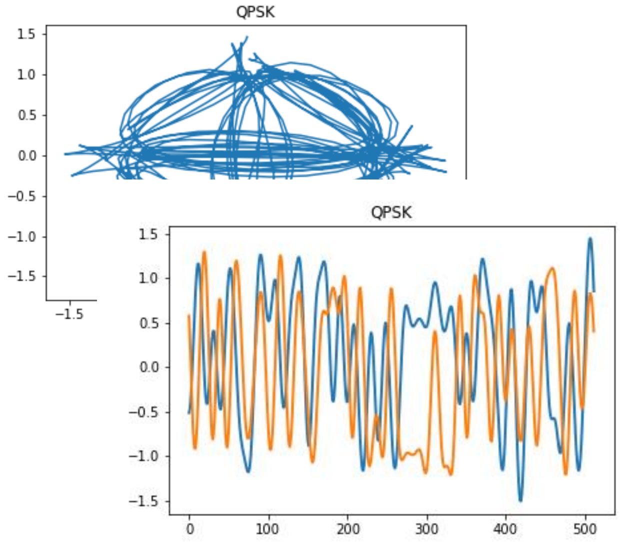


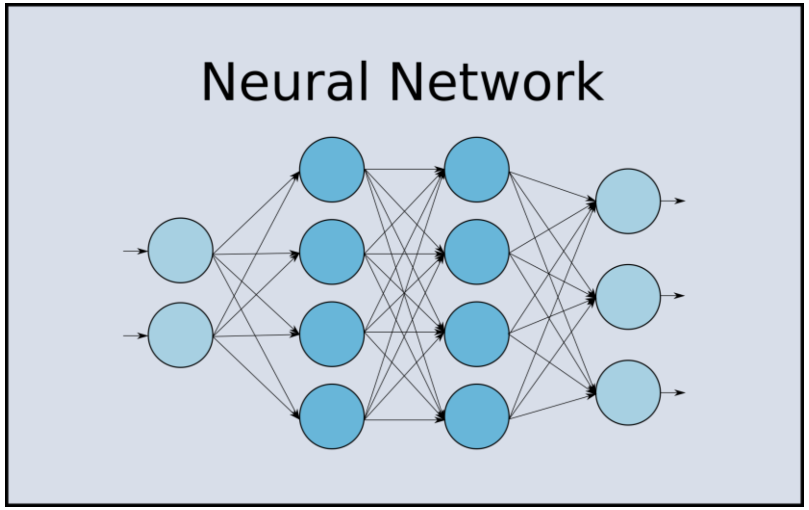
- Why not do this with signals?
- We can!



Automatic Modulation Classification

- We generate examples our network can learn from by applying different SNR levels and channel effects
  - 1 example is a combination of a waveform (I/Q samples) and a label (class name e.g. "BPSK")
- Iterate over the same examples (dataset) until the network accuracy is good enough
- Another strategy is to train multiple models for low/high SNR



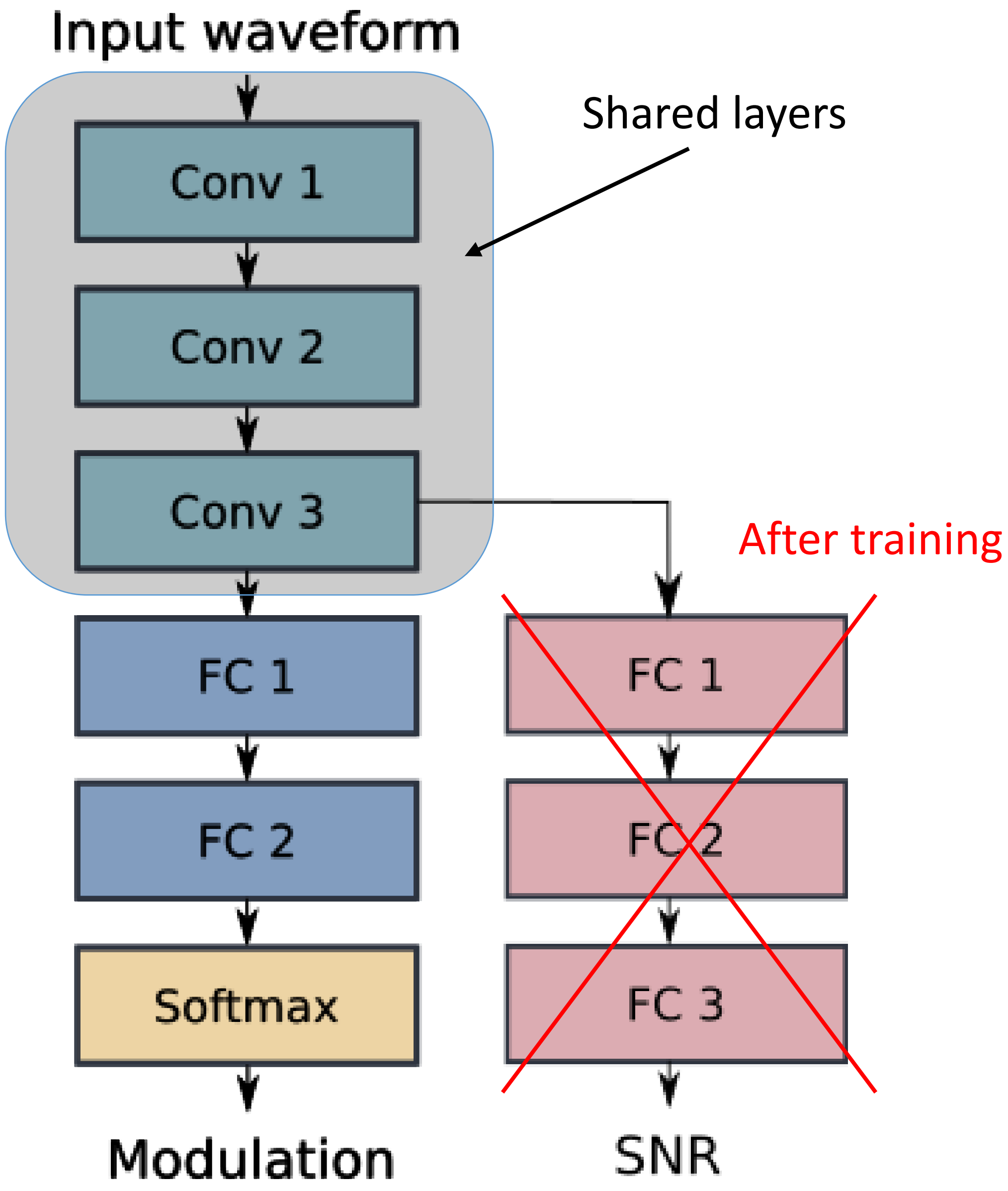


BPSK?
QPSK!!!
OFDM?
FM?

AMC: Multitask Learning

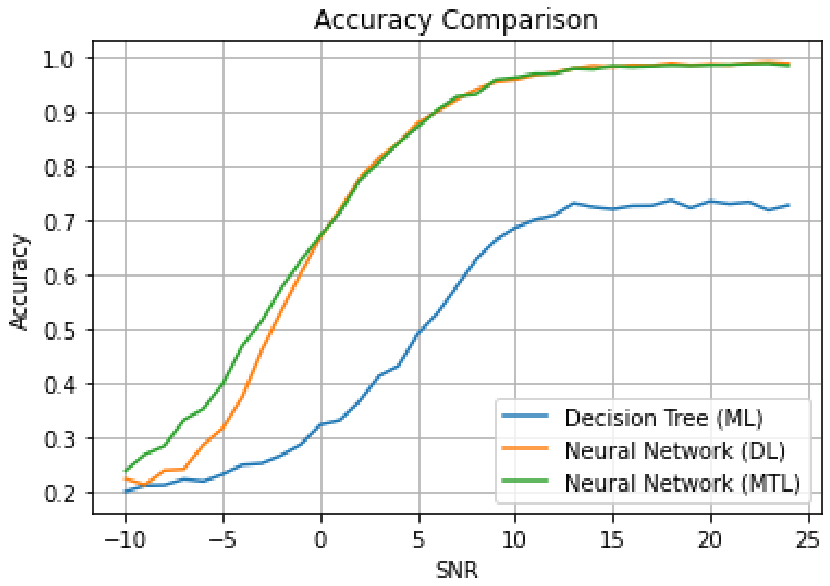
StrathSDR

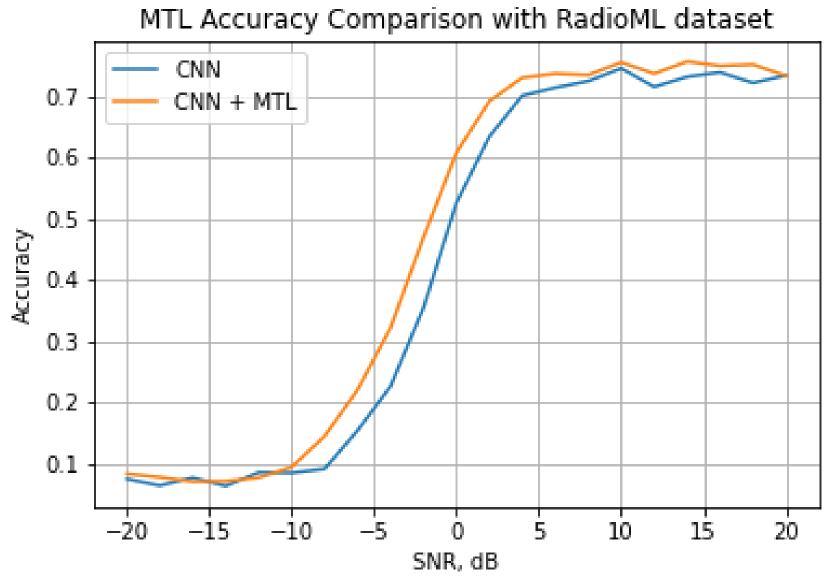
- Typical classification setup
  - Convolutions for feature extraction
  - Fully Connected (FC) for predictive power
  - Softmax for final classification (output probabilities)
- Normally we generate waveforms at different SNR values to train this network
- To apply multitask learning we do not have to change our main network
- Attach additional SNR estimation head
- Now we can use data we would normally throw away...



AMC: Multitask Learning

- Tested on synthetic data with 5 classes and harder 14 classes from RadioML 2018 dataset
- We gain a roughly ~2% average accuracy increase on a real dataset





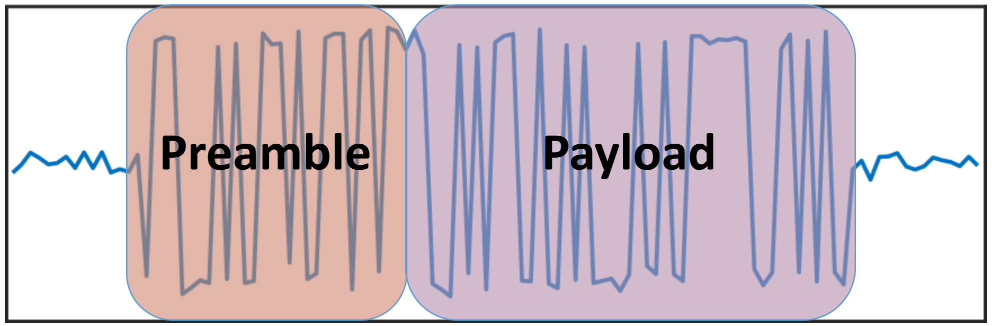
- **Key takeaway**: we improve performance by utilizing data we would normally throw away, with no additional costs in hardware (for inference)

Physical Layer Frame Synchronization

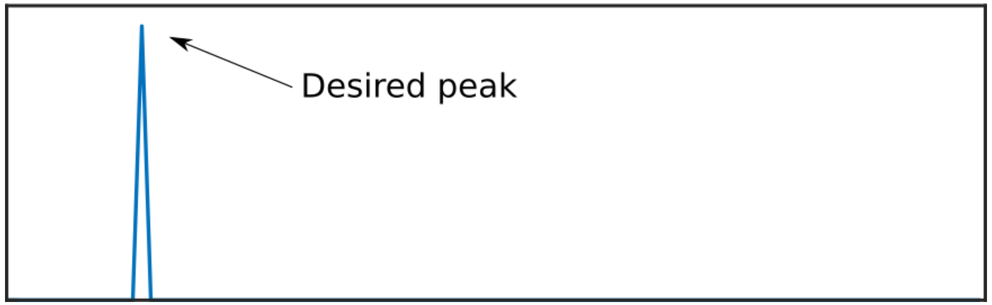
- Frame sync is an essential step many comms systems
- Can implement with simple correlation with known sequence to detect when packets start

- **Why use DL for this??**
- For small sensor transmitters saving as much as a byte can have big energy implications
- Reduction in overheads will increase spectrum efficiency and comm link throughput

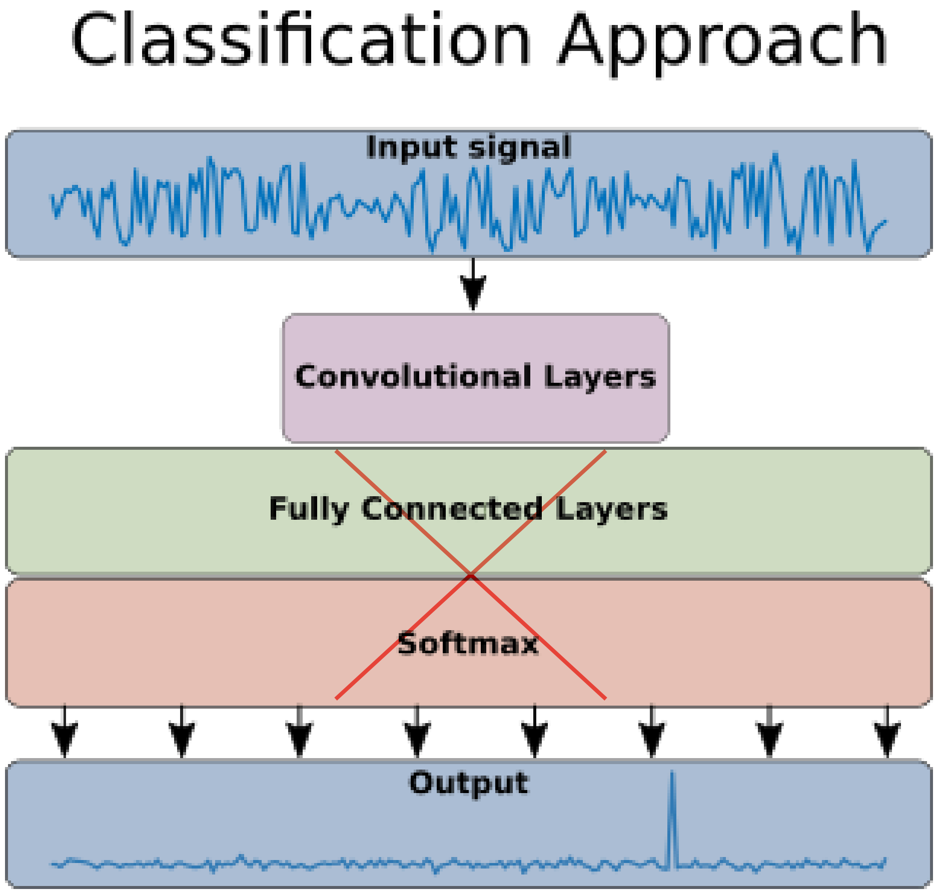
Training waveform



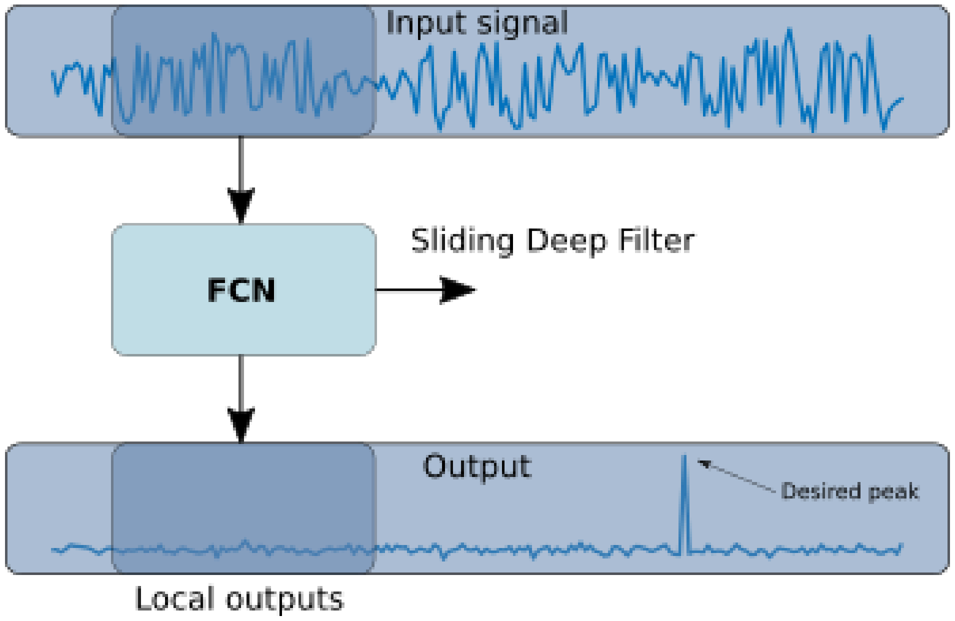
Training label



Physical Layer Frame Synchronization

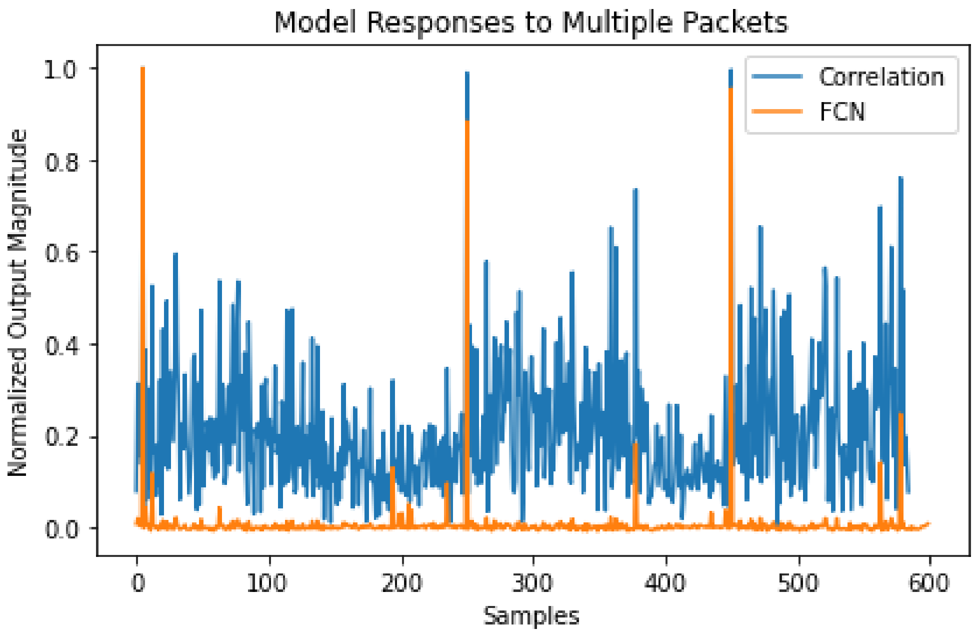


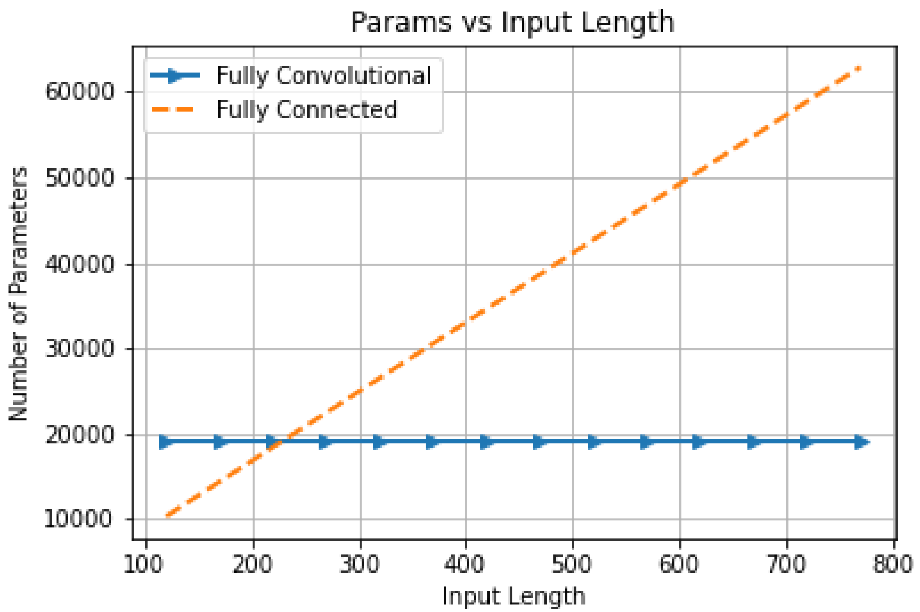
- Gut instinct is to use classification
- What we care about is putting things into hardware/FPGAs
- Quantization can only get us so far



Physical Layer Frame Synchronization

- Pleasant side-effect of using FCNs is the ability to deploy them on input of any size
- Network never saw more than 2 spikes in the training data
- Can save a lot of compute by training on small windows and deploying anywhere





- Memory footprint for on-chip weights storage is greatly reduced as we scale input size
- We are essentially dealing with a filter, rather than a network full of scary matrix multiplies

Conclusions

StrathSDR

Conclusions

- We saw 2 DL examples for typical PHY layer wireless comms
  - Automatic Modulation Classification
  - Frame Synchronization
- If can be helped, try to use all of your data
- Try multitask learning at home – it's free**

Thanks for listening!

Engage with us:

https://sdr.eee.strath.ac.uk

@strathSDR

github.com/strath-sdr