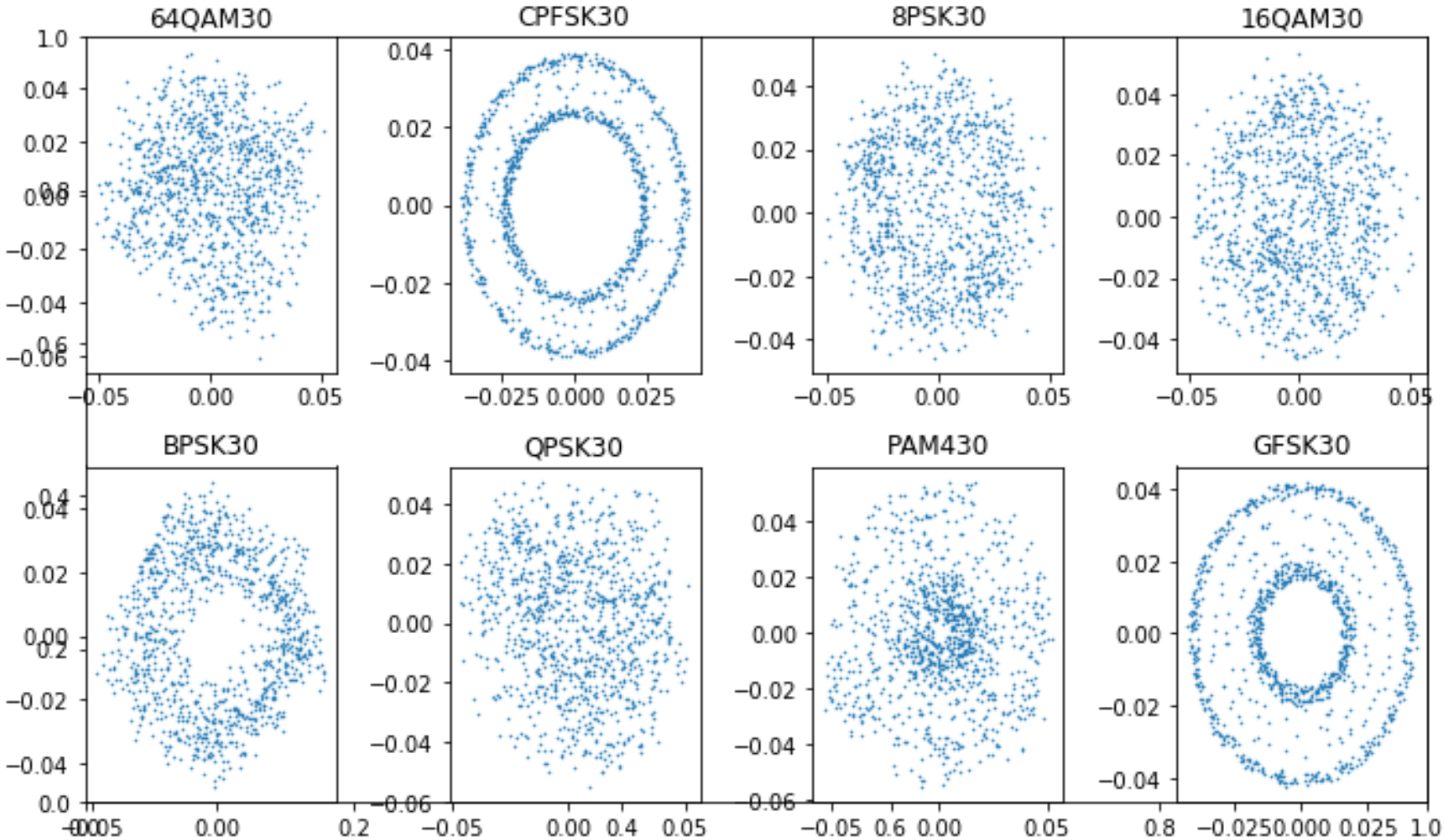
Updates 11 Aug

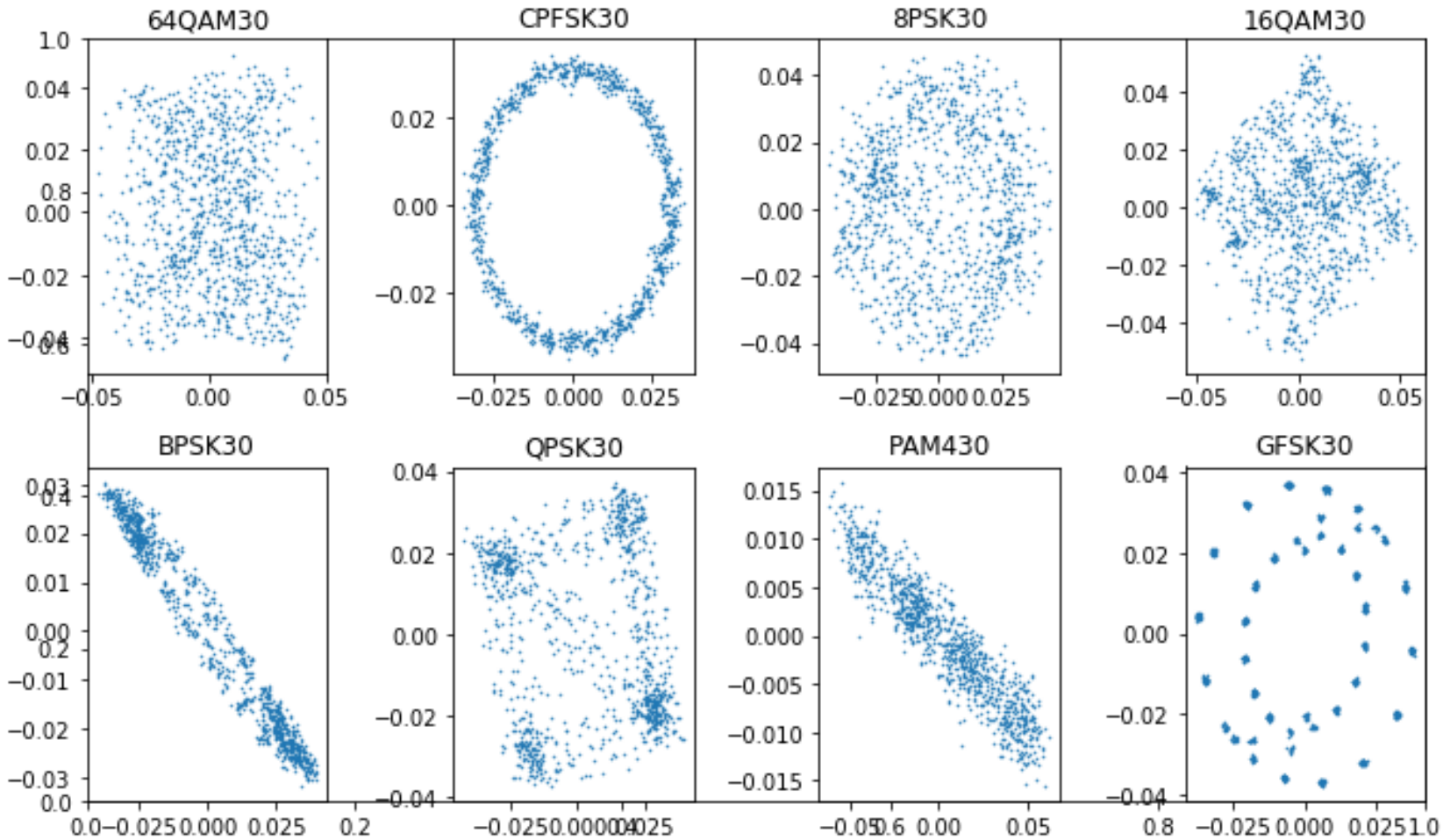
Recap from last week

- Supposed to try to simulate different kinds of real-life conditions (didn’t do) and assess effects on classification accuracy
- On Matlab data with Rician fading: try how models perform, and how well models trained on easier dataset performs on hard dataset
  - Especially whether image method still works given how undiscernible the hard Matlab data was
- Try combining both time and image into one feature

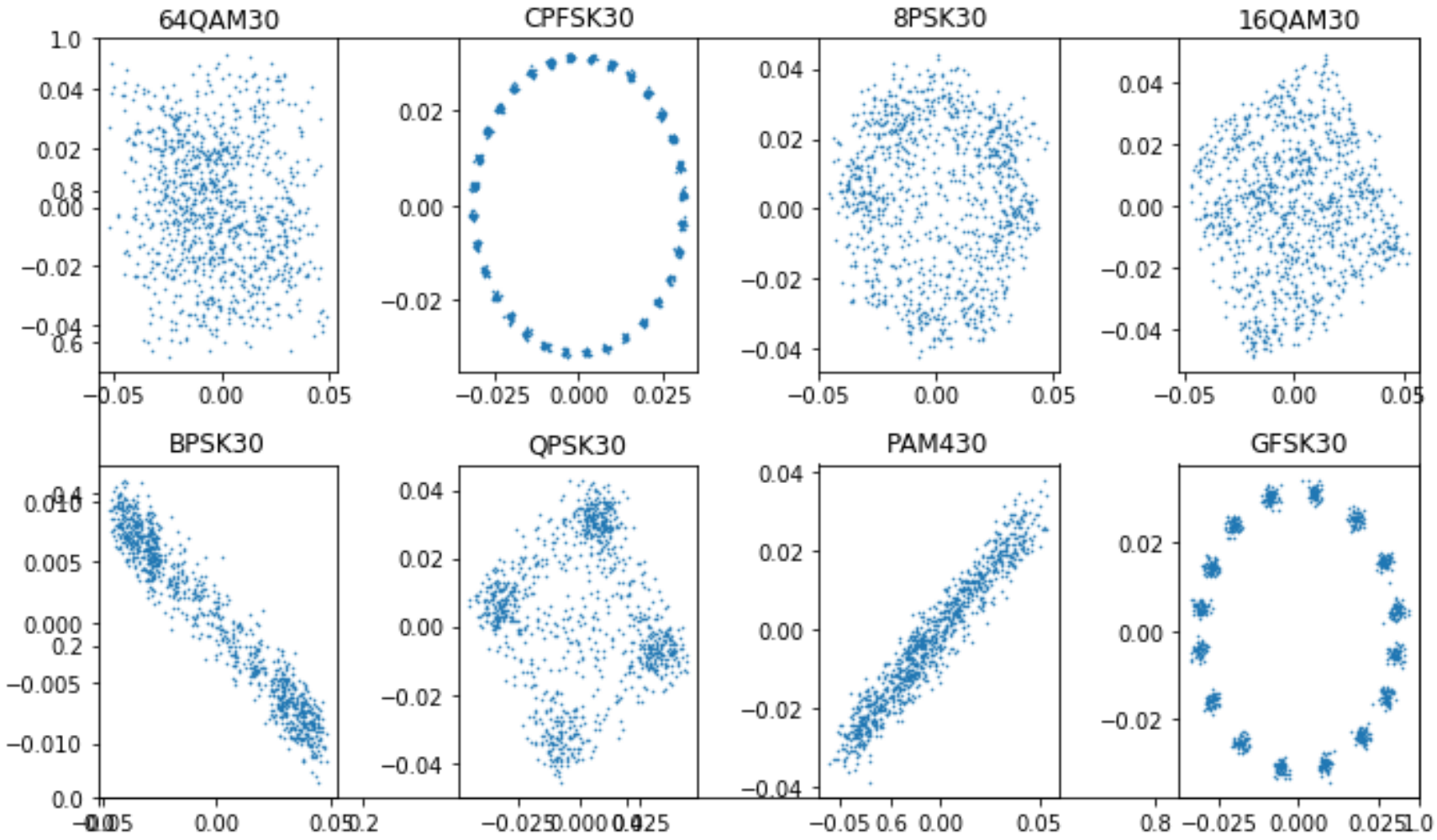
Hard Dataset – AWGN, fading, heavy clock shift, Doppler



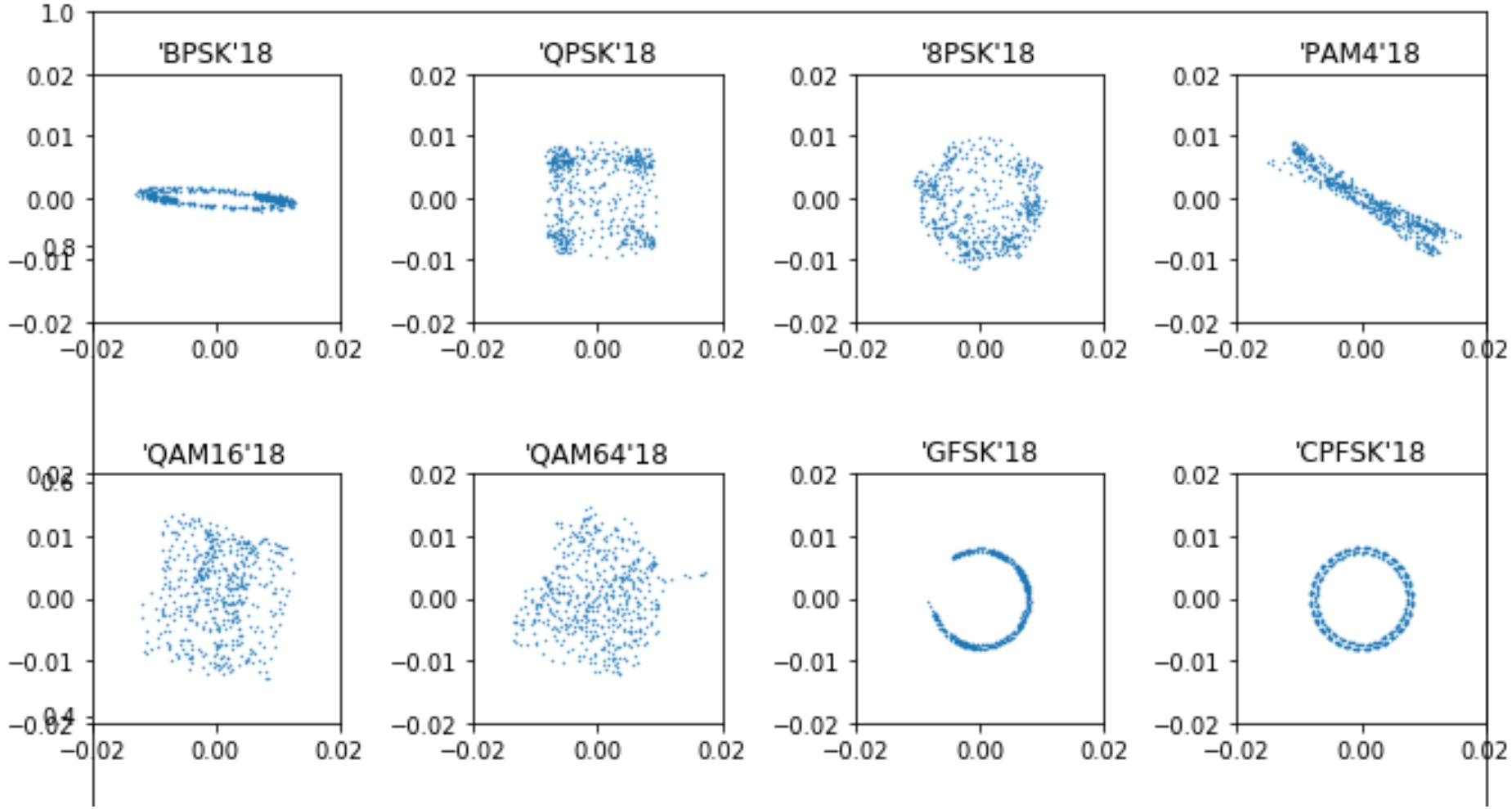
Medium Dataset – AWGN, fading, light clock shift



Easy Dataset – only AWGN



RadioML


'BPSK'18
'QPSK'18
'8PSK'18
'PAM4'18
'QAM16'18
'QAM64'18
'GFSK'18
'CPFSK'18

Summary

1. Model performance on Matlab data
    1. Resnet-iq model
    2. Constellation model
2. Some ideas on combining time and image feature

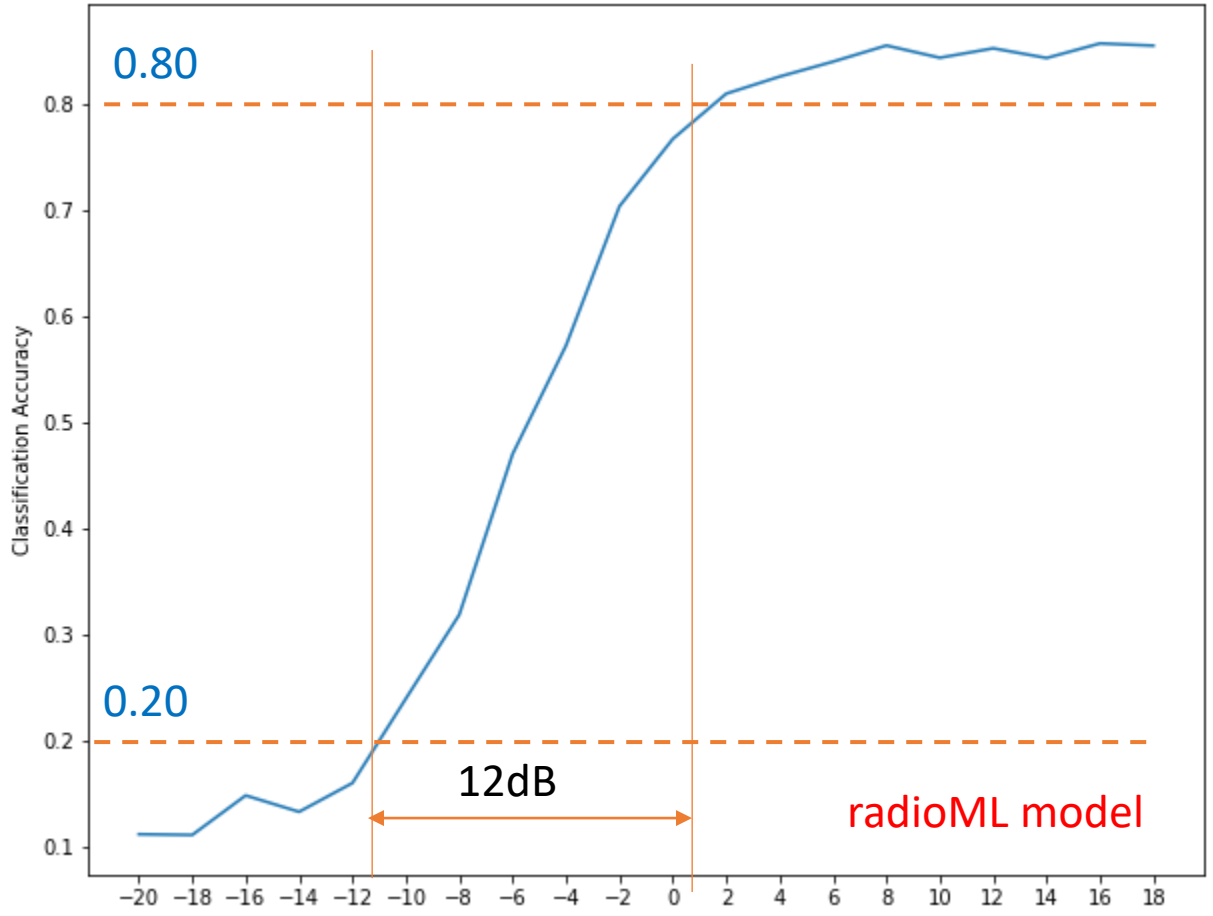
1. 1 Performance of resnet-iq model on data

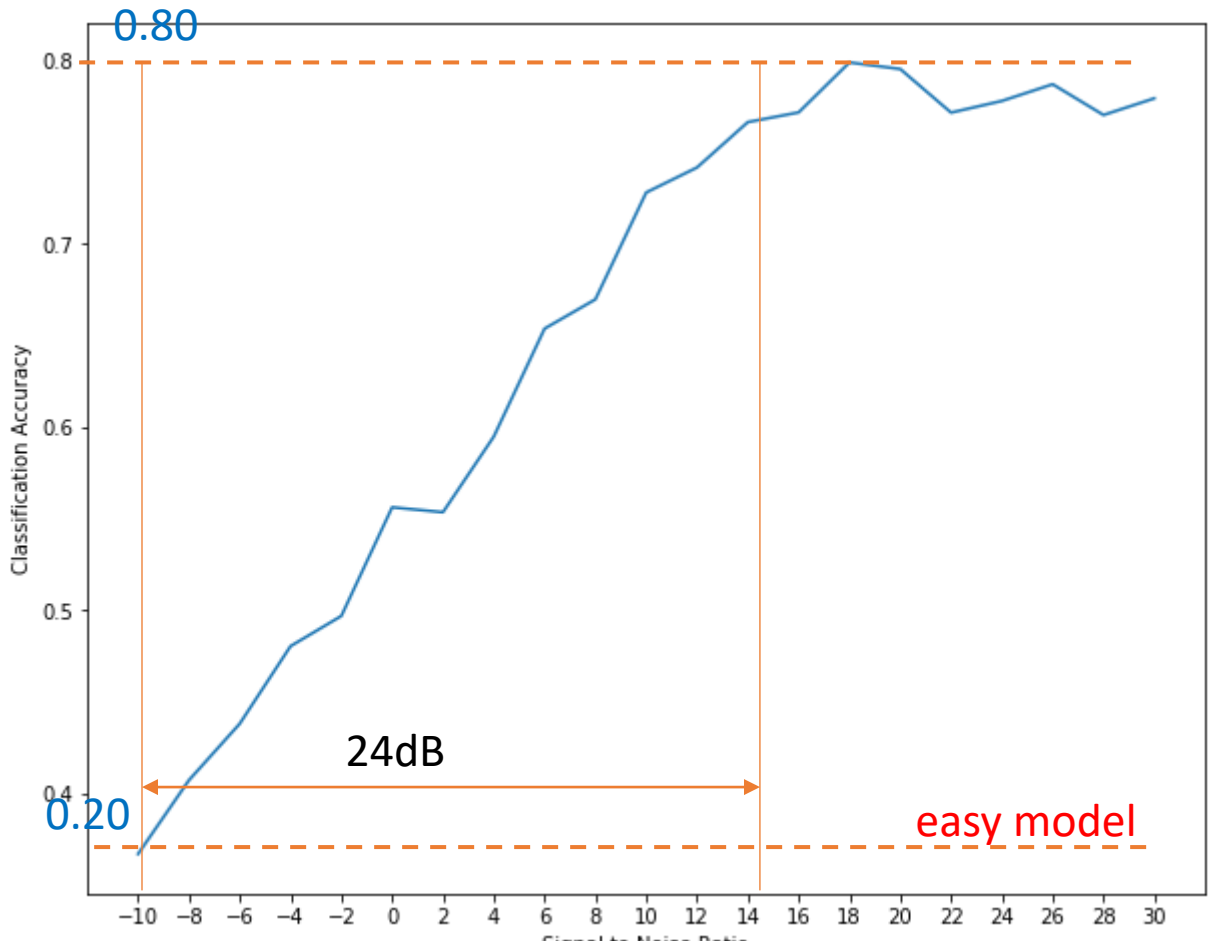
Procedure:

- Trained resnet model on iq data (only the 8 digital modulation classes) from easy, medium, hard and radioML datasets → so 4 models
- Tested how models perform on other datasets to assess adaptability

1. 1 Resnet-iq: Easy vs radioML

- Quite hard to compare because SNR in Matlab and radioML doesn't seem to be defined the same way
- But can roughly see that Resnet-iq model performs similarly for both easy and radioML datasets over the SNR range

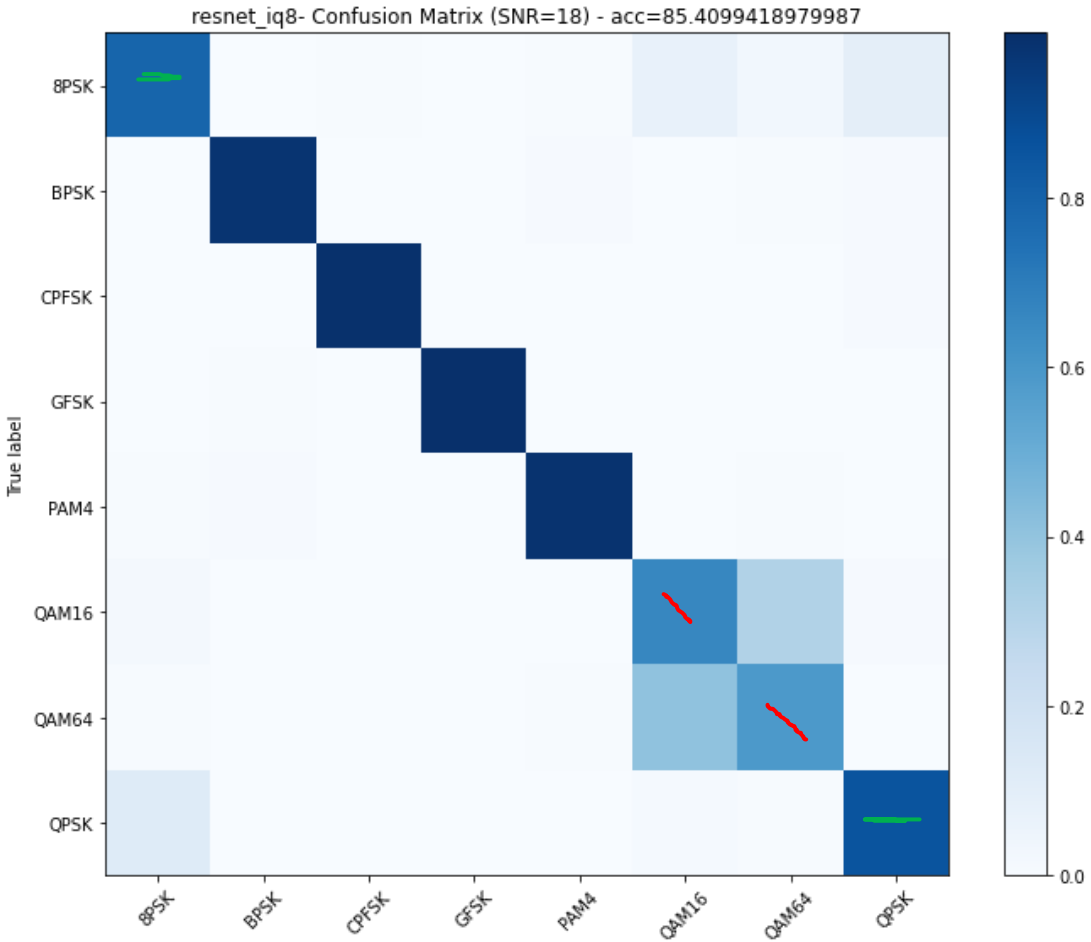




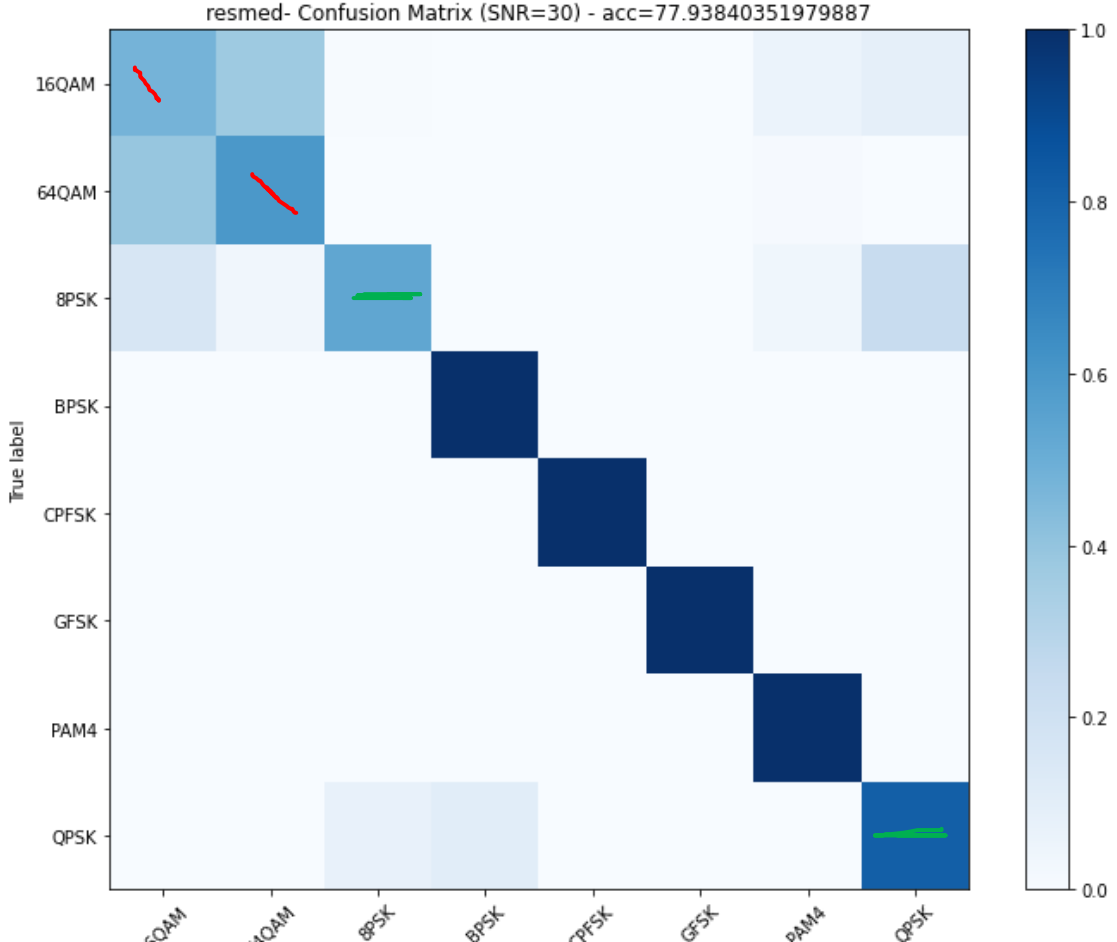
1. 1 Resnet-iq: Easy vs radioML

- Areas of confusion (QAMs, high order PSKs) at highest SNR also same
- From now on, use easy model as reference for comparing Matlab models

radioML model

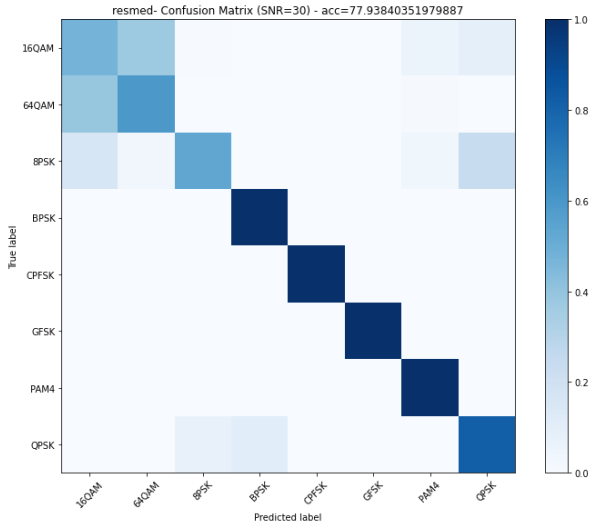


easy model

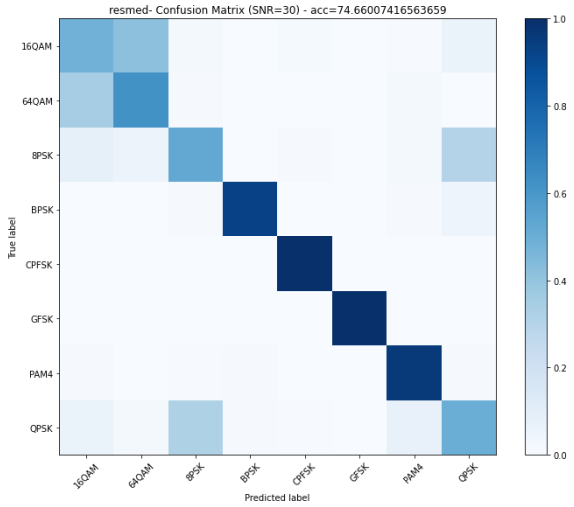


1.1 Resnet-iq

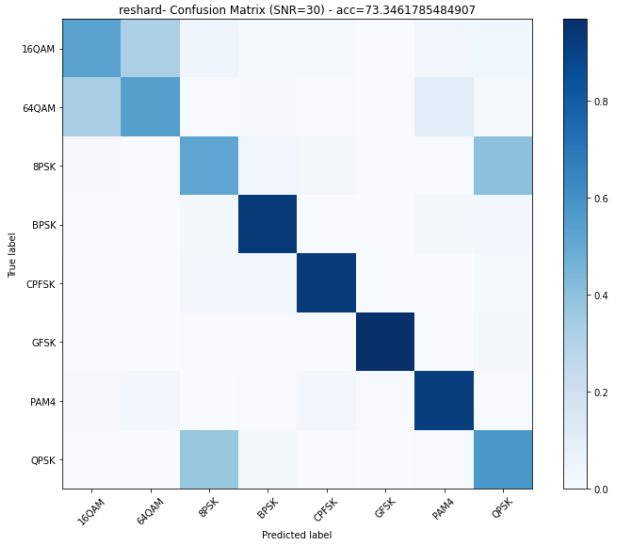
Easy – 77.9%



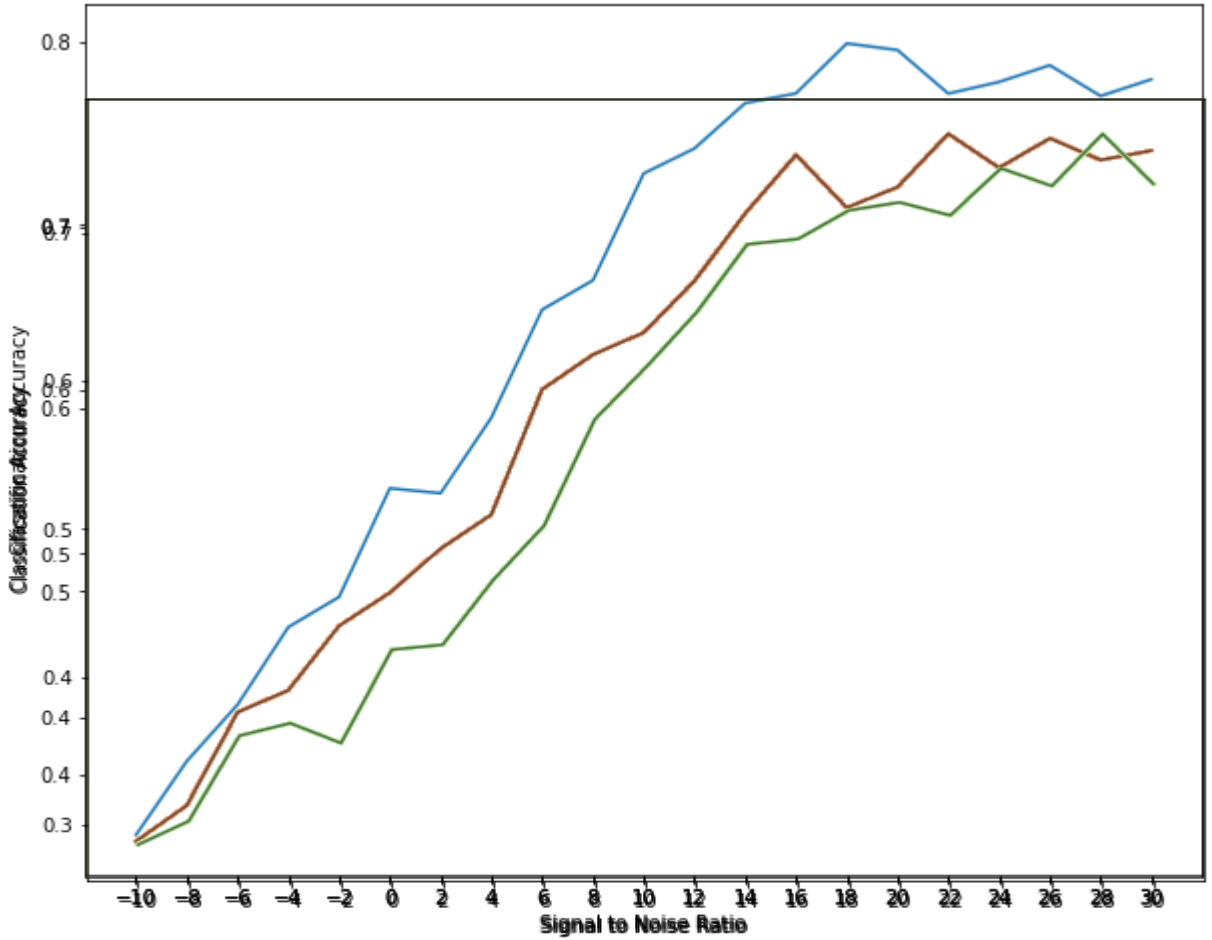
Medium – 74.7%



Hard – 73.3%



As expected, accuracy decreases with increasing difficulty of dataset.

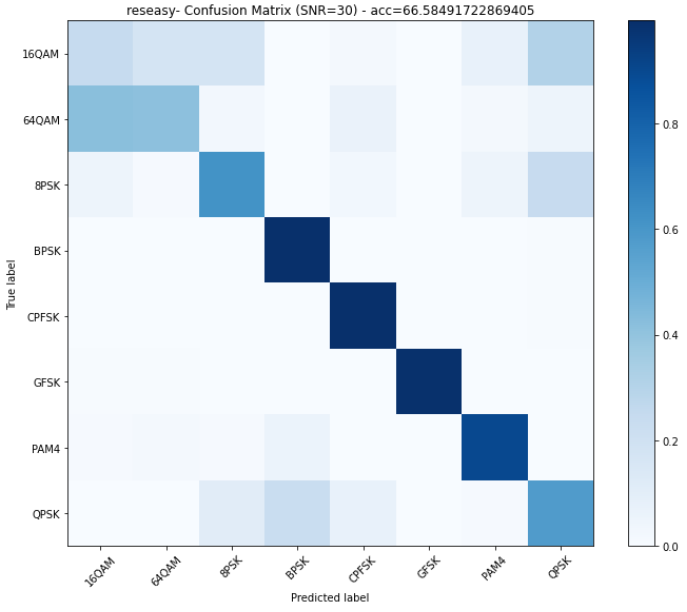


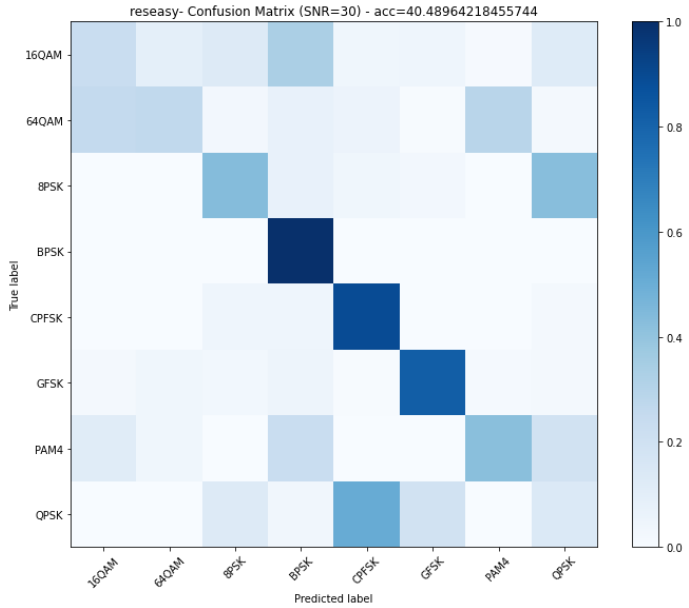
Disclaimer – not to scale

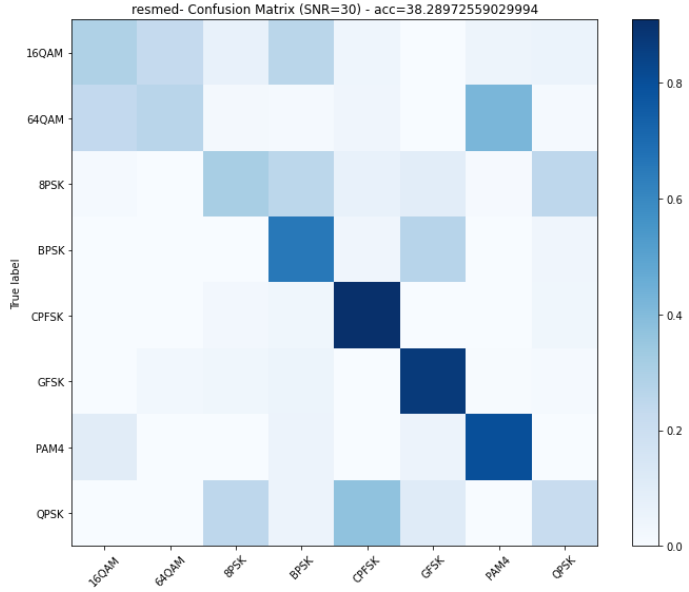
1.1 Resnet-iq cross-evaluated on other datasets

Classification accuracy at 30dB/ %		Tested on:		
		Easy	**Medium**	**Hard**
Trained on:	**Easy**	78	66	40
	Medium	-	75	38
	Hard	-	65	73

- Cross-evaluation on datasets harder than was used to train the model works better than random (12.5%)
- Quite good with BPSK, CPFSK, GFSK, PAM4, and very bad with QAMS and high order PSKs.

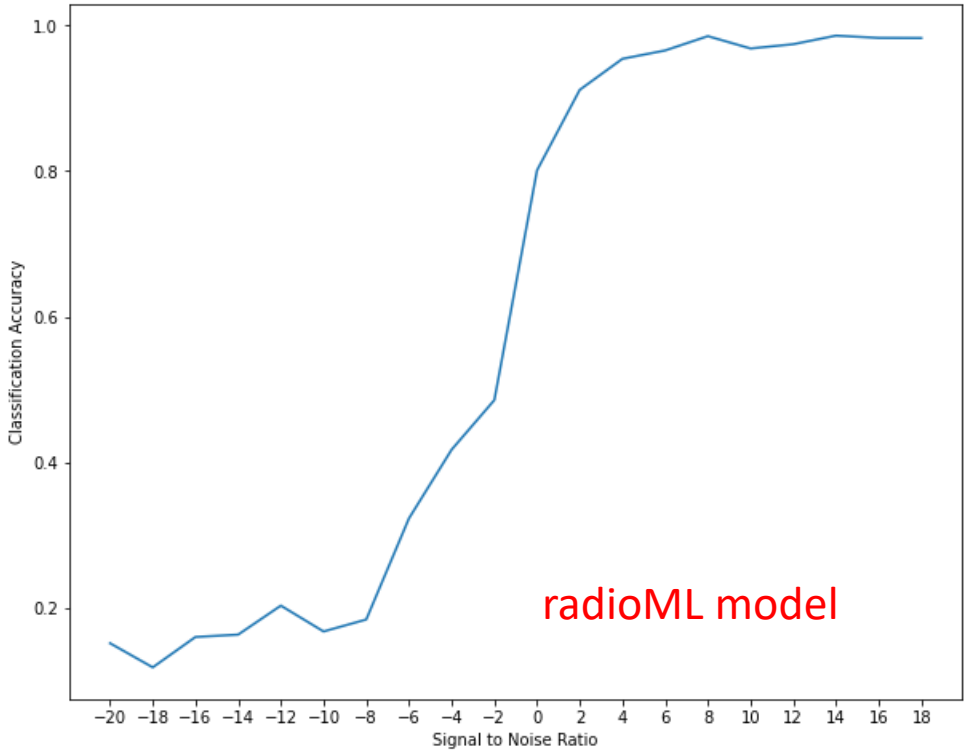


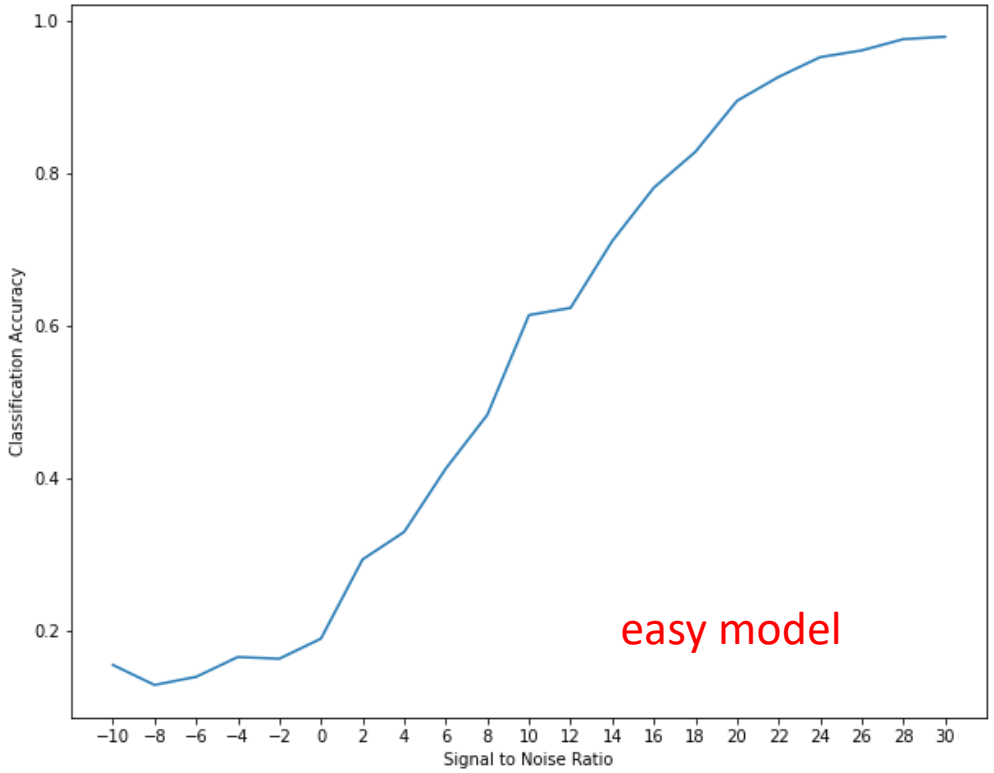




1.2 Constellation: easy vs radioML

- At high SNR, both constellation models trained on easy and radioML dataset achieve nearly 100% accuracy

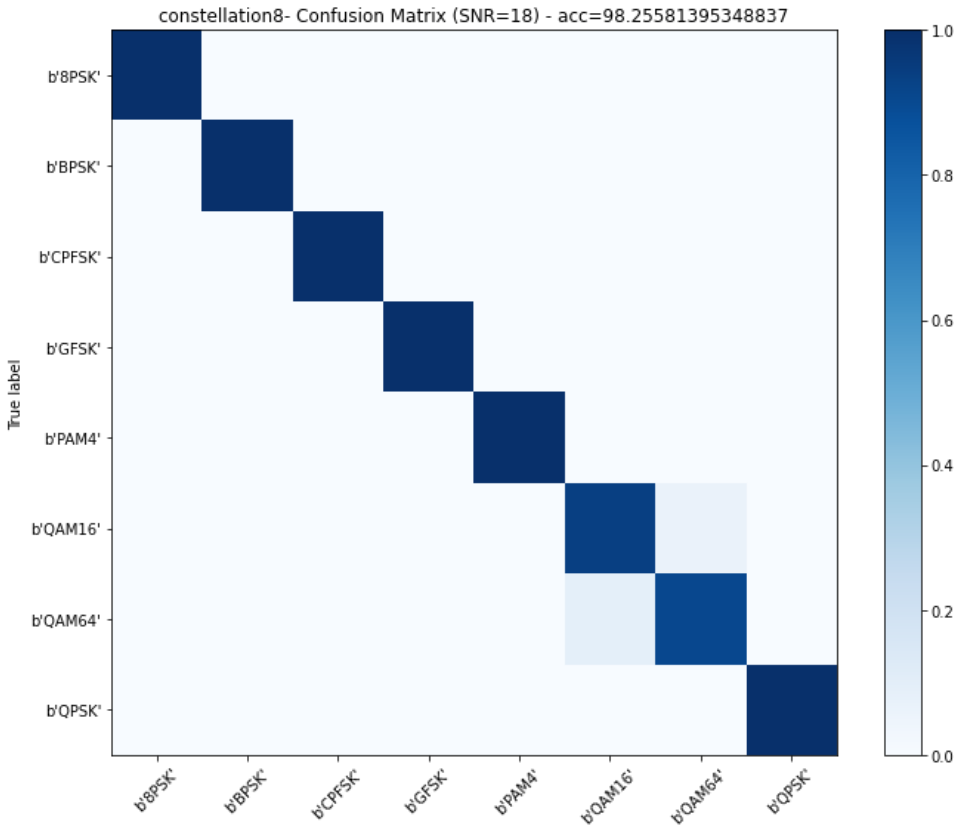




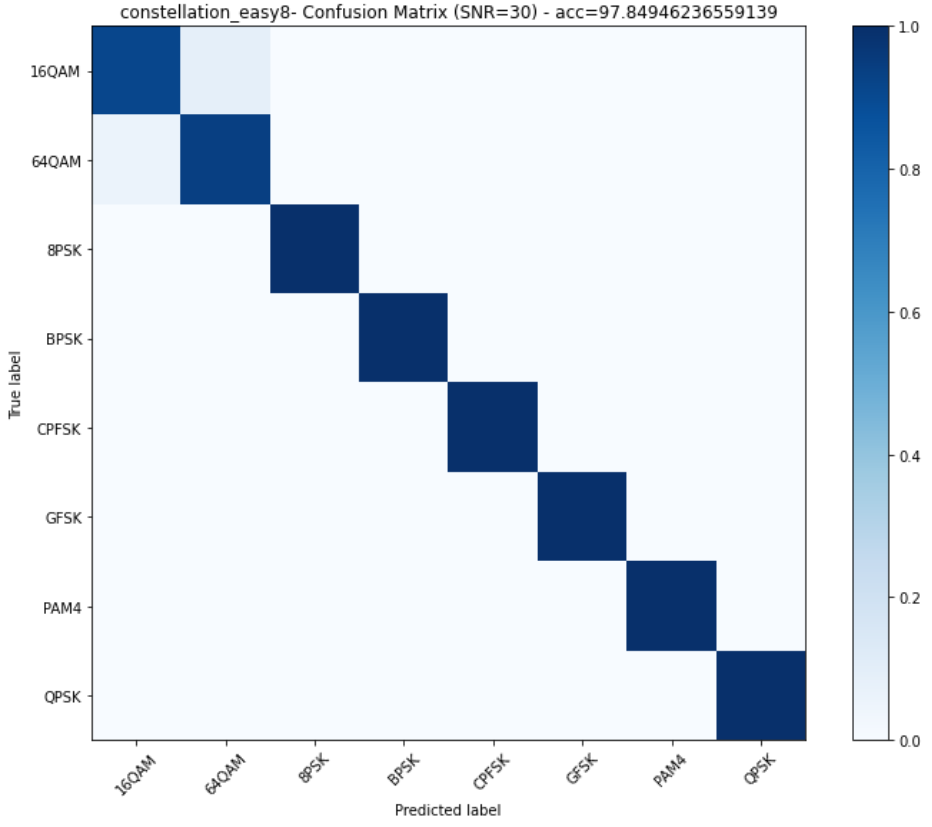
1.2 Constellation: easy vs radioML

- Confusion matrices at high SNR look almost identical, with slight confusion between QAMs
- Kind of expected, but what we were interested in is how constellations perform for hard dataset

radioML model



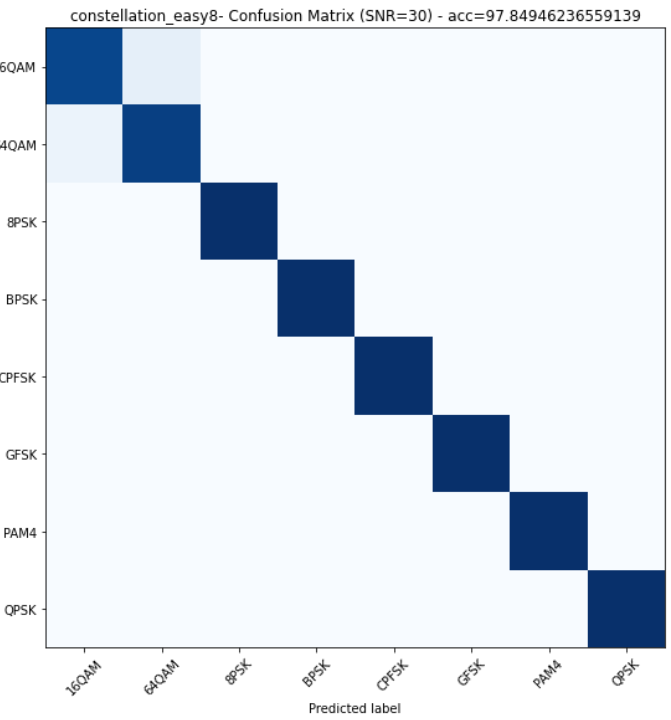
easy model



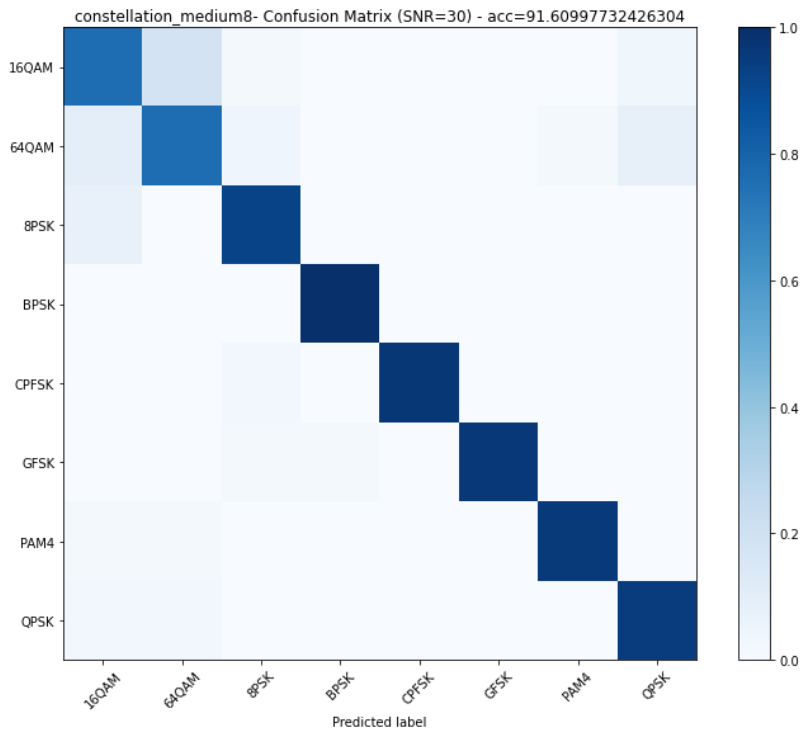
1.2 Constellation – accuracy at 30dB

- Constellations don't work well for hard dataset
- Significant confusion between CPFSK, GFSK (which resnet-iq model wasn't confused about) → maybe combine both models for classification?

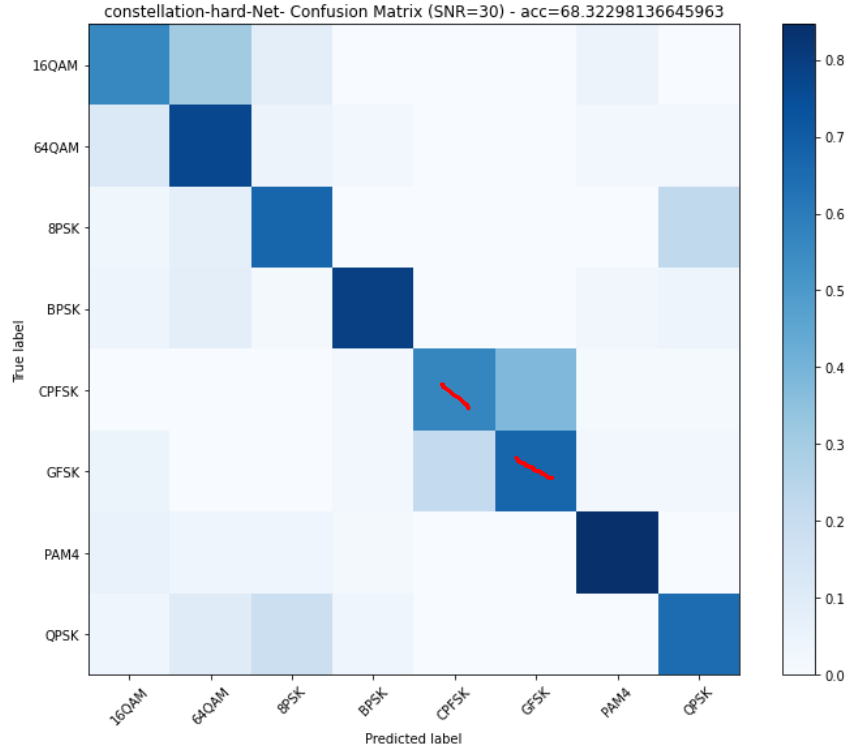
Easy – 97.8%



Medium – 91.6%



Hard – 68.3%

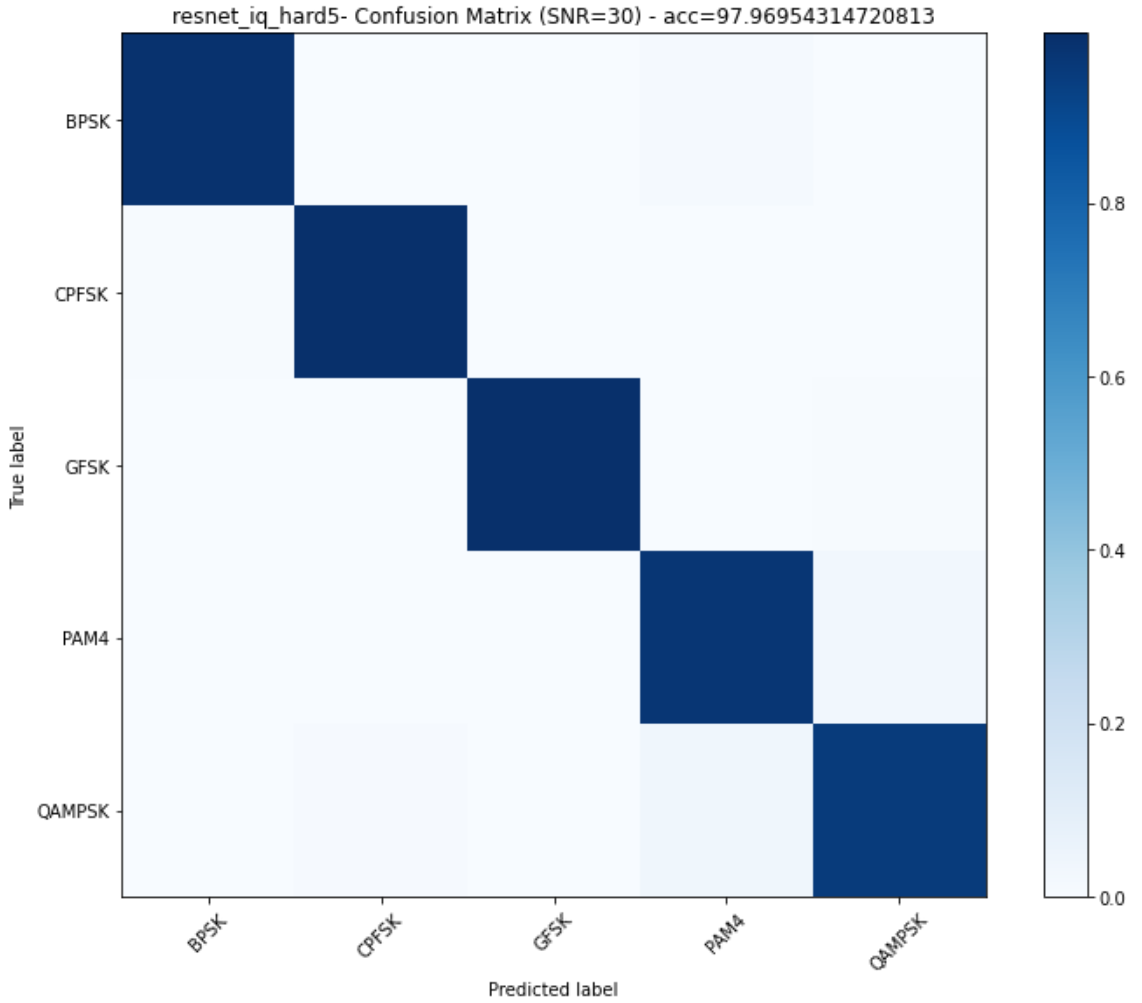


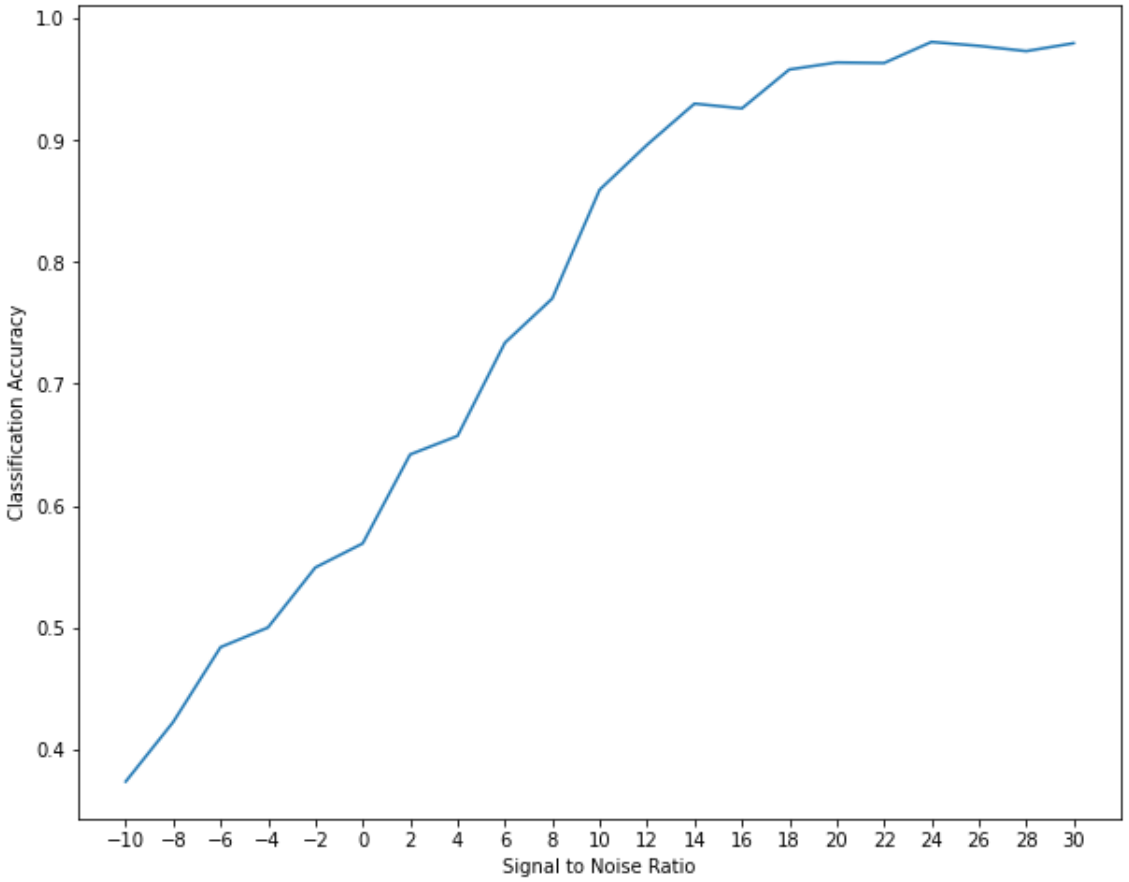
2. Ideas about combining time and image features

1. Use resnet-iq to differentiate 5 classes (QAMPSK, BPSK, PAM4, GFSK, CPFSK), then use constellation to differentiate 4 classes (16QAM, 64QAM, QPSK, 8PSK)
2. Combine time and image into video – video classification?
3. Graph classification? Turn each I/Q data point into nodes of a graph, with edges corresponding to time relation and Euclidean distance, then apply graph convolutional networks?

2.1 Resnet-iq model on 5 classes (hard dataset)

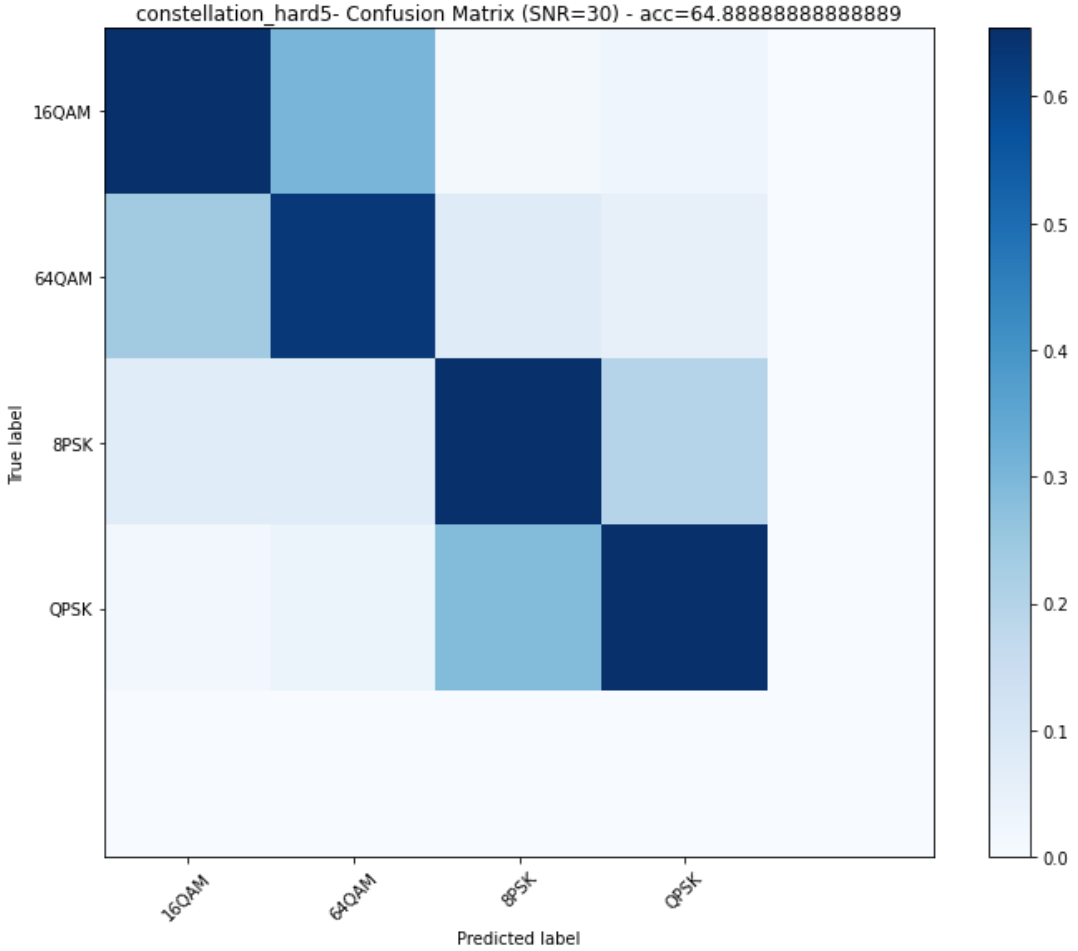
- Almost perfect at high SNRs
- Suggests that resnet-iq is a good first stage to differentiate high order QAMs and PSKs from the rest

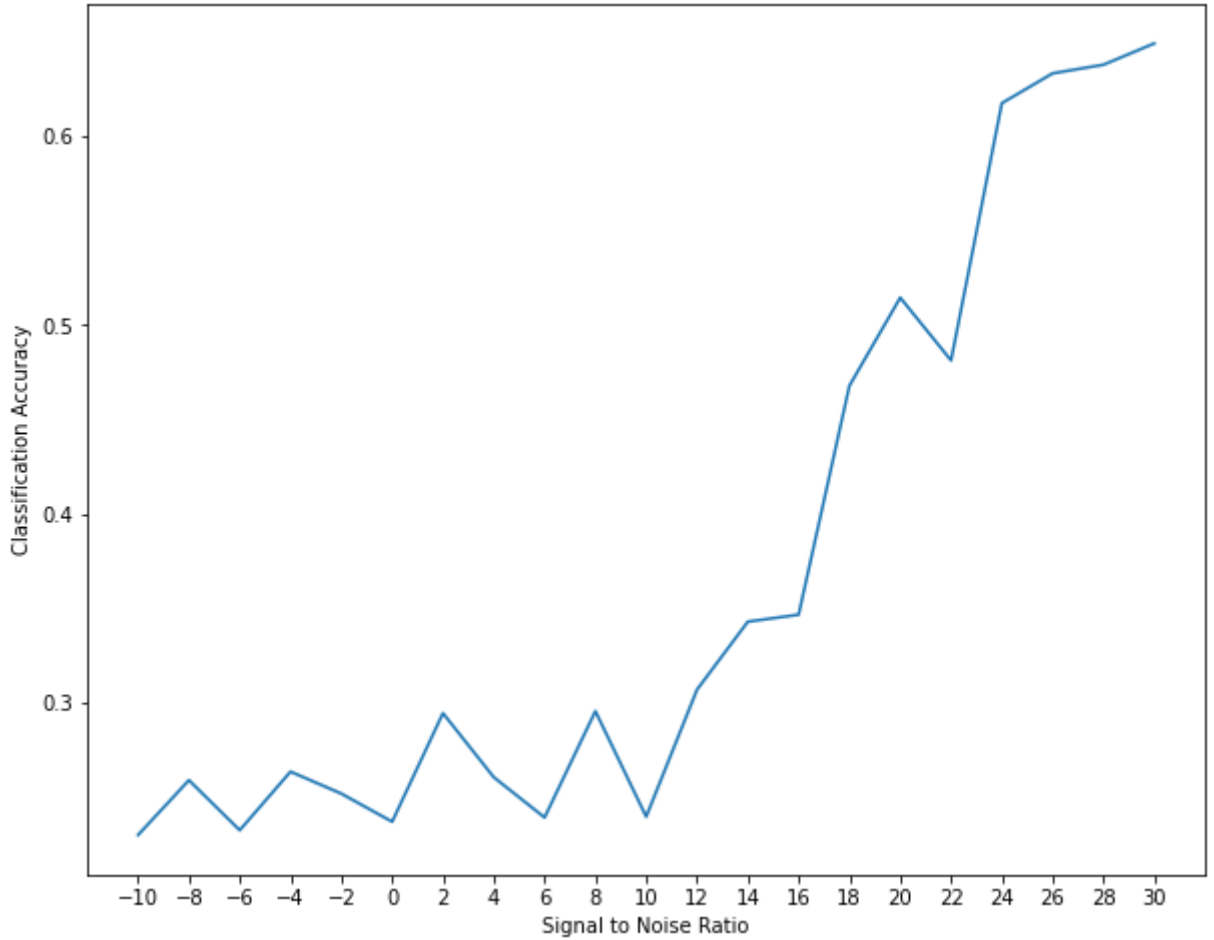




2.1 Constellation model on 4 classes (hard dataset)

- Constellation model still cannot differentiate QAMs and PSKs in hard dataset properly





Other ideas to combine time and image into one feature

1. Use resnet-iq to differentiate 5 classes (QAMPSK, BPSK, PAM4, GFSK, CPFSK), then use constellation to differentiate 4 classes (16QAM, 64QAM, QPSK, 8PSK)
2. Combine time and image into video – video classification?
3. Graph classification? Turn each I/Q data point into nodes of a graph, with edges corresponding to time relation and Euclidean distance, then apply graph convolutional networks?

Next steps

- Stop dataset generation
- Last thing to try: combine time and image feature
  - Weighted average of resnet-iq and constellation?
  - Video classification?
  - Graph classification?
- Updates on fri – stop new stuff on thur, start preparing updates and documenting