Radio signal modulation classification

Updates: 21 Jul 2020

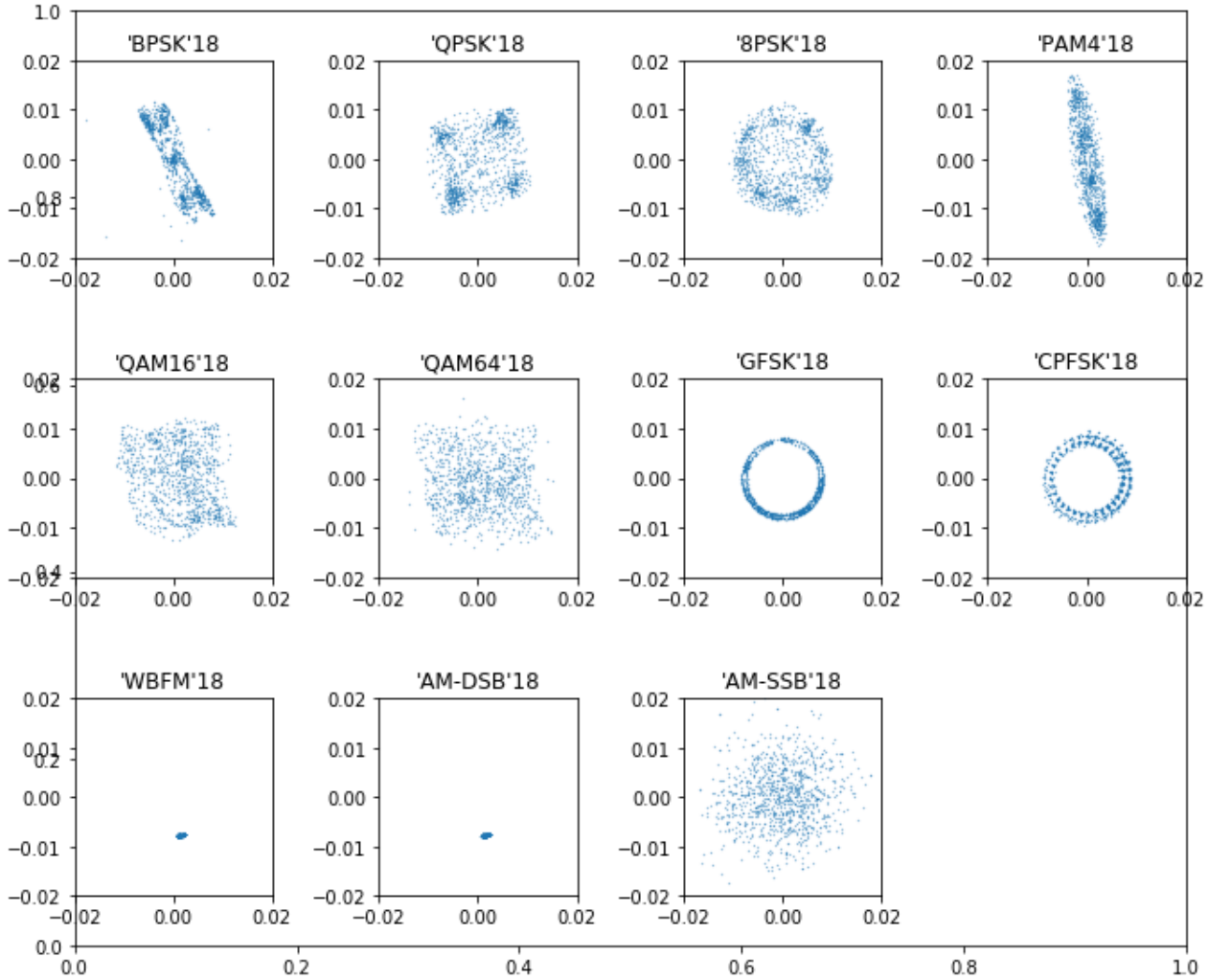
Summary

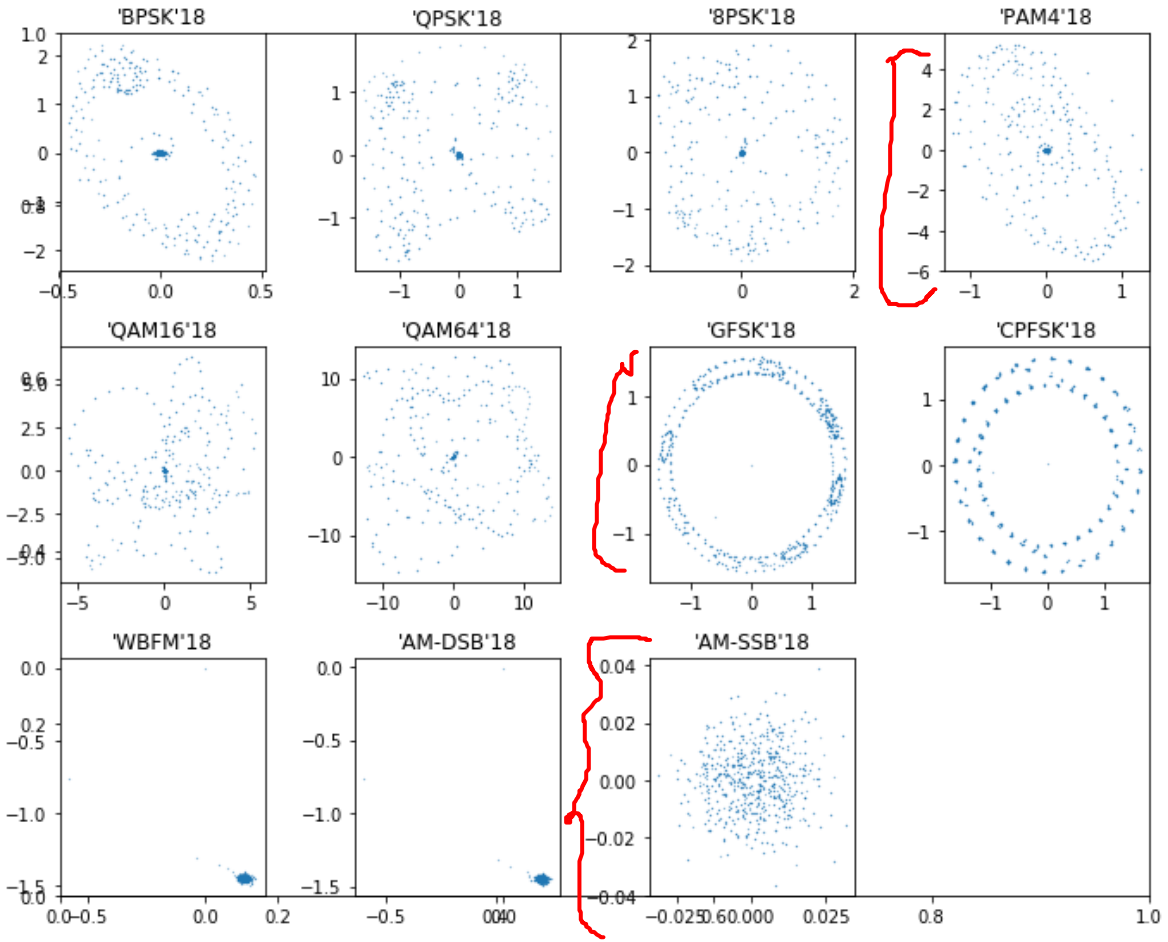
1. Data Exploration
2. Features tried for training
3. Next steps

1. Data Exploration - source

- Two radioML datasets, **RADIOML 2016.04C** and **RADIOML 2016.10A**, (available from https://www.deepsig.ai/datasets)
- 11 modulation classes
  - (8 digital + 3 analog)
  - Variable SNR
  - Channel effects simulated
- Samples are time-series in I/Q form of array shape (2, 128) Difference: **2016.10A** is normalized and cleaner

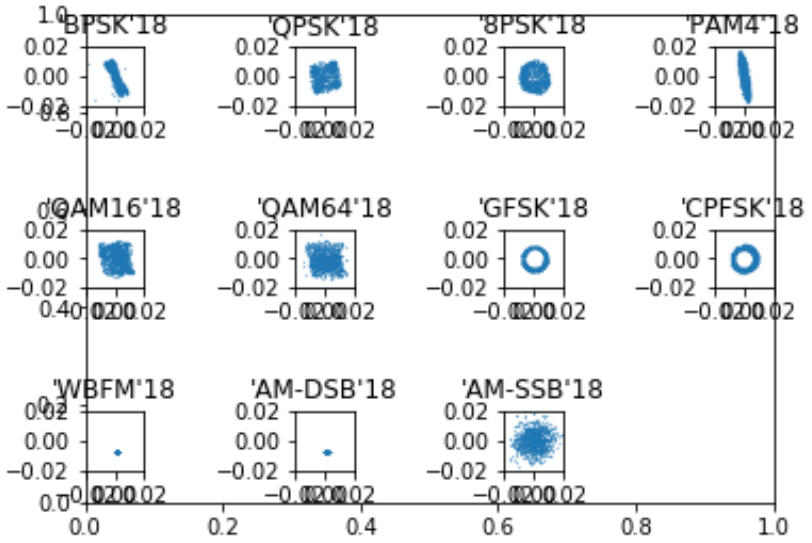
1. Data: **2016.10A** vs **2016.04C** (Constellation)



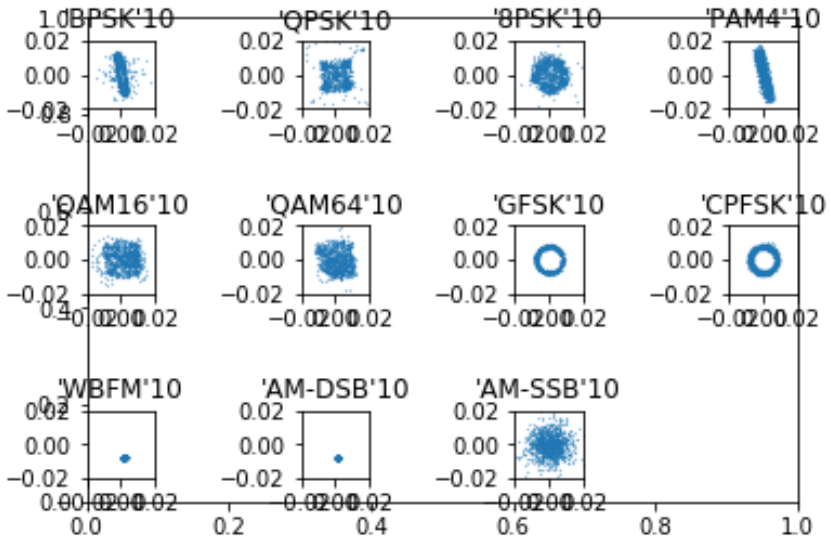


1. Data: Clarity at different SNRs

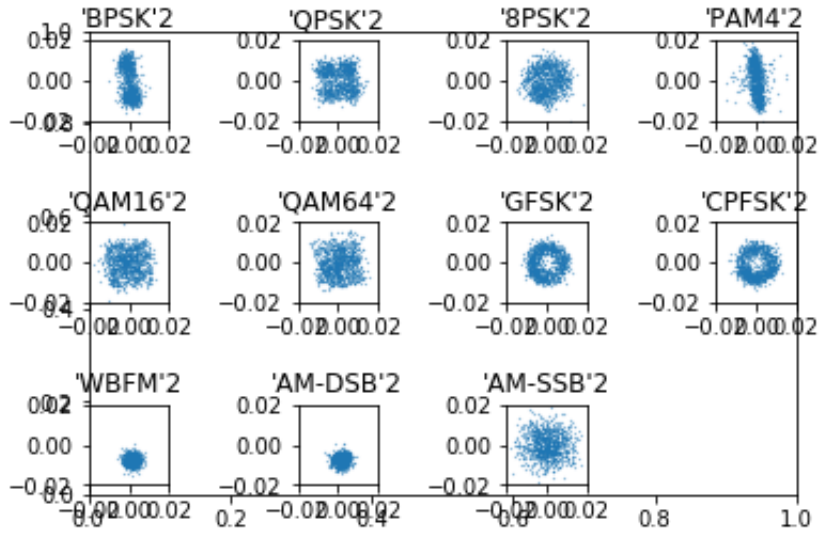
18



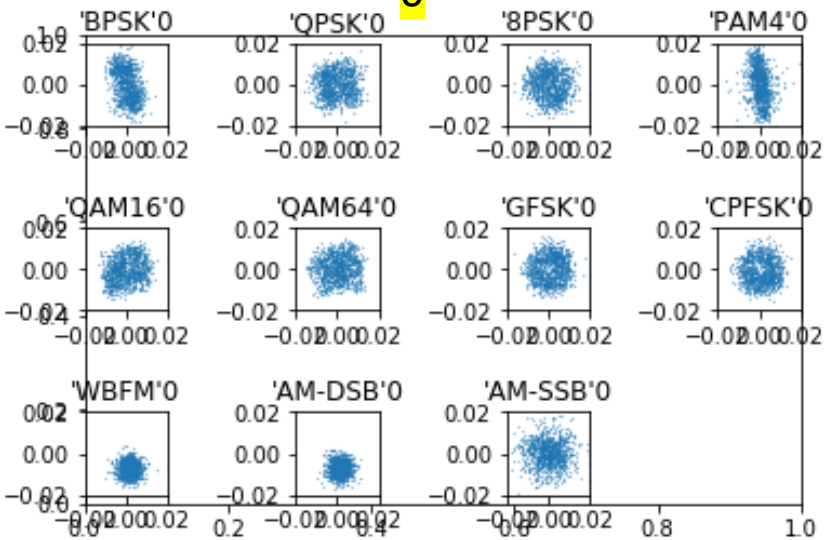
10



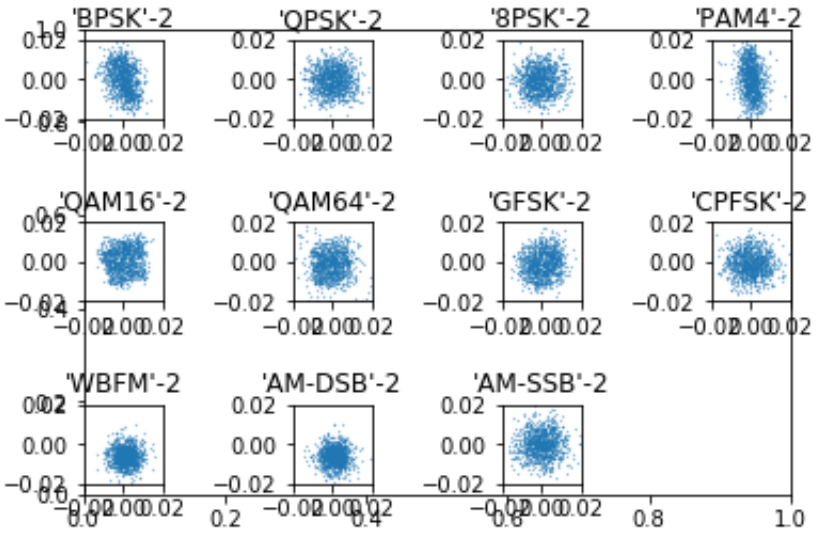
2



0



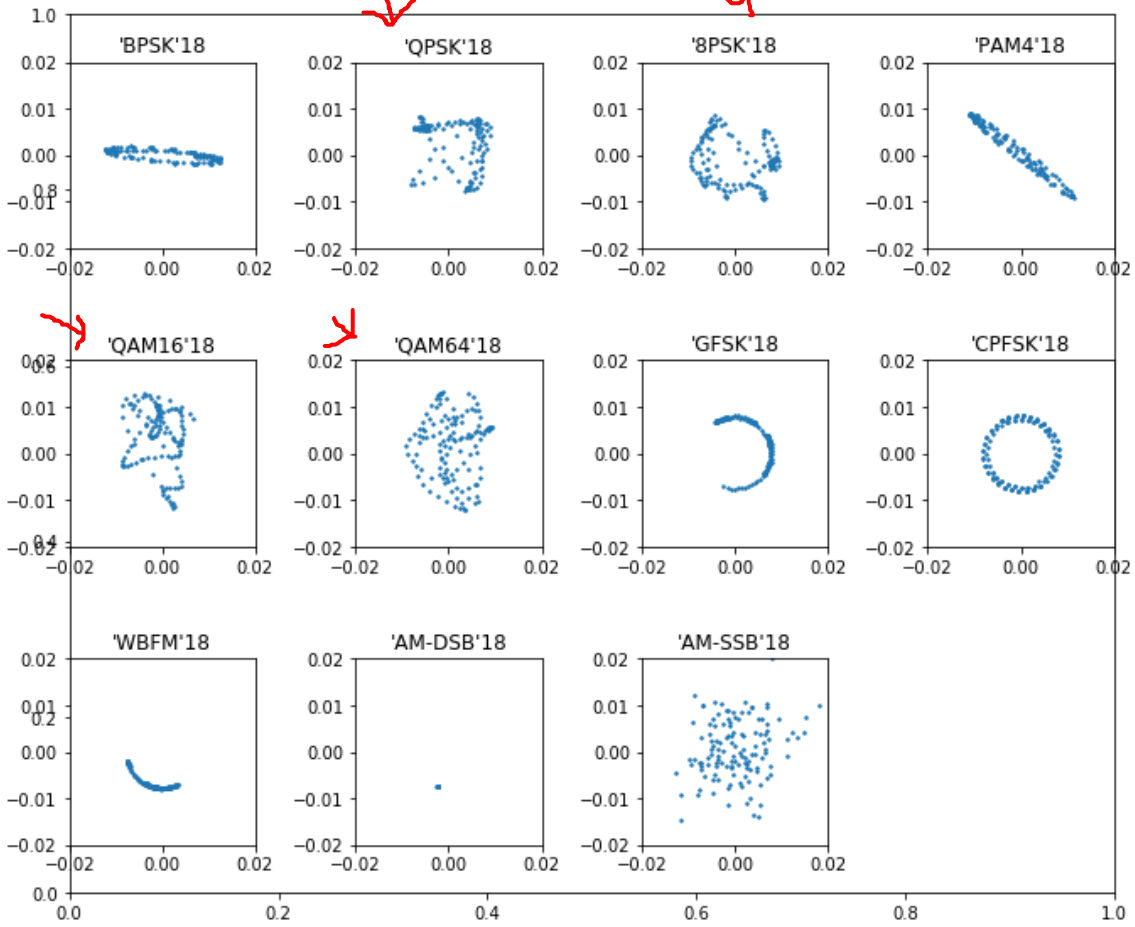
-2



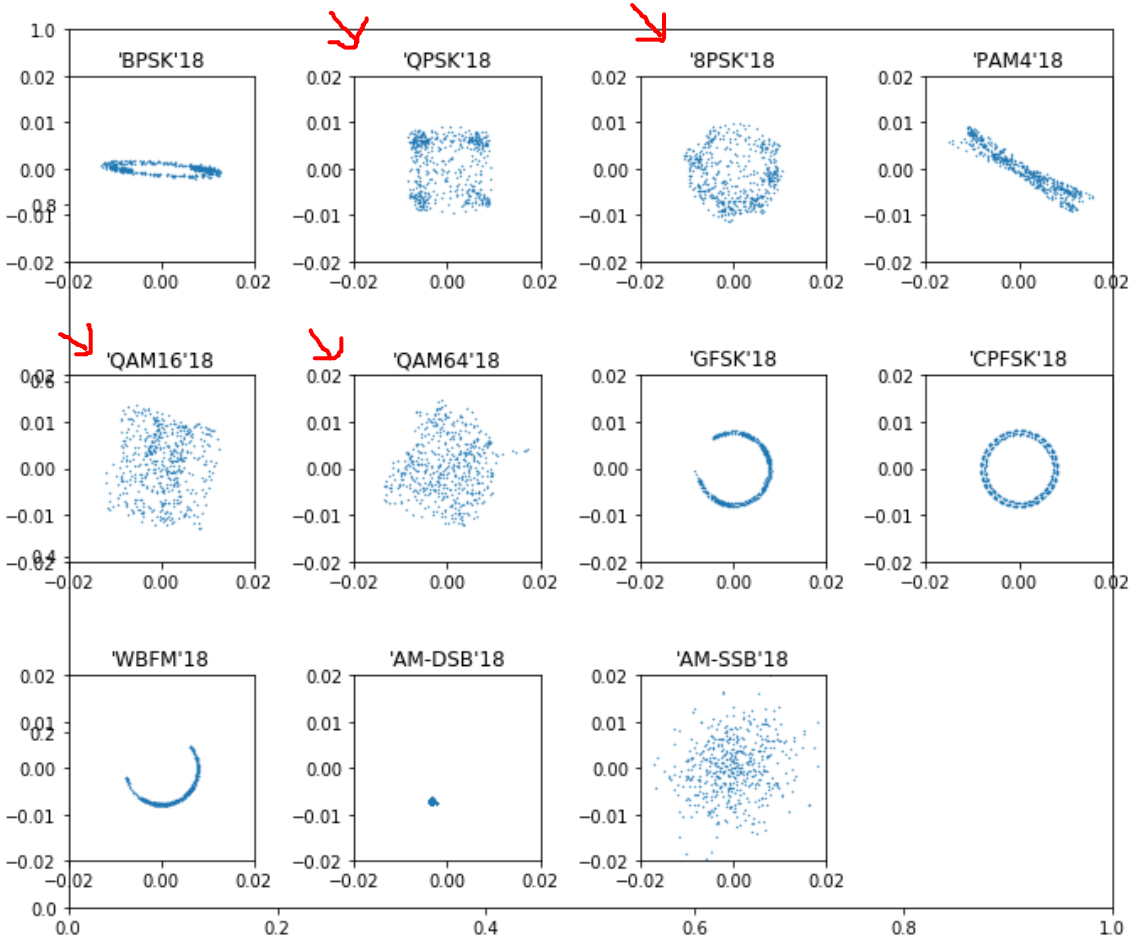
From roughly 0dB and below, the constellation plots become indistinguishable by naked eye.

1. Data: Clarity with different number of points

128



1024 (4 sets dumpted tgt)



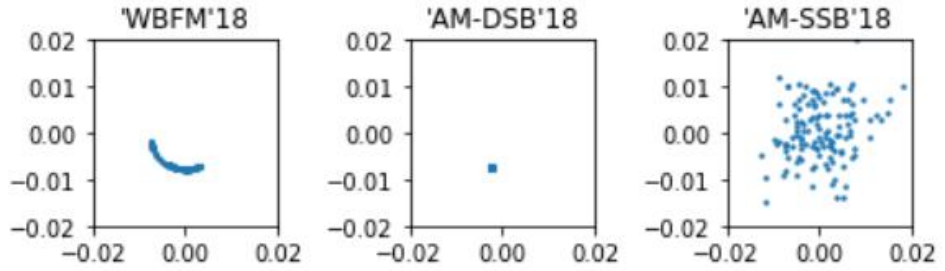
1. Data – problems with analogue mod

- Analogue dataset comes from audio stream with pauses in the middle
- Impossible to differentiate WBFM and AM-DSB during pauses

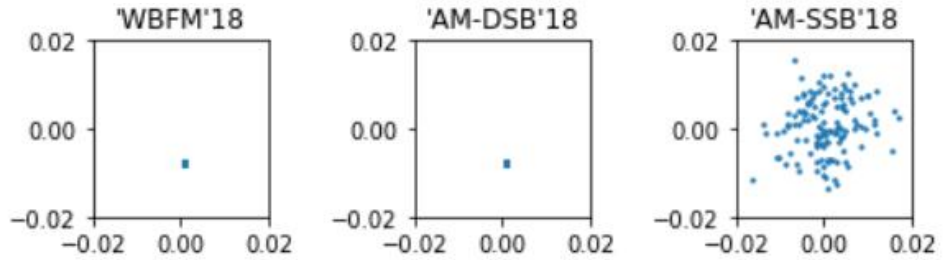
Understanding issues…

- AM-DSB amplitude doesn't vary…

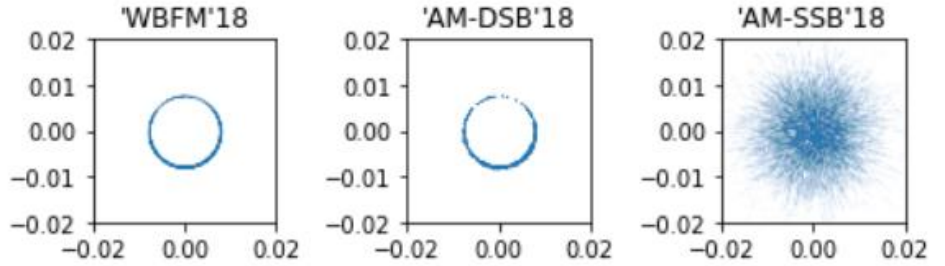
Signal present



No signal

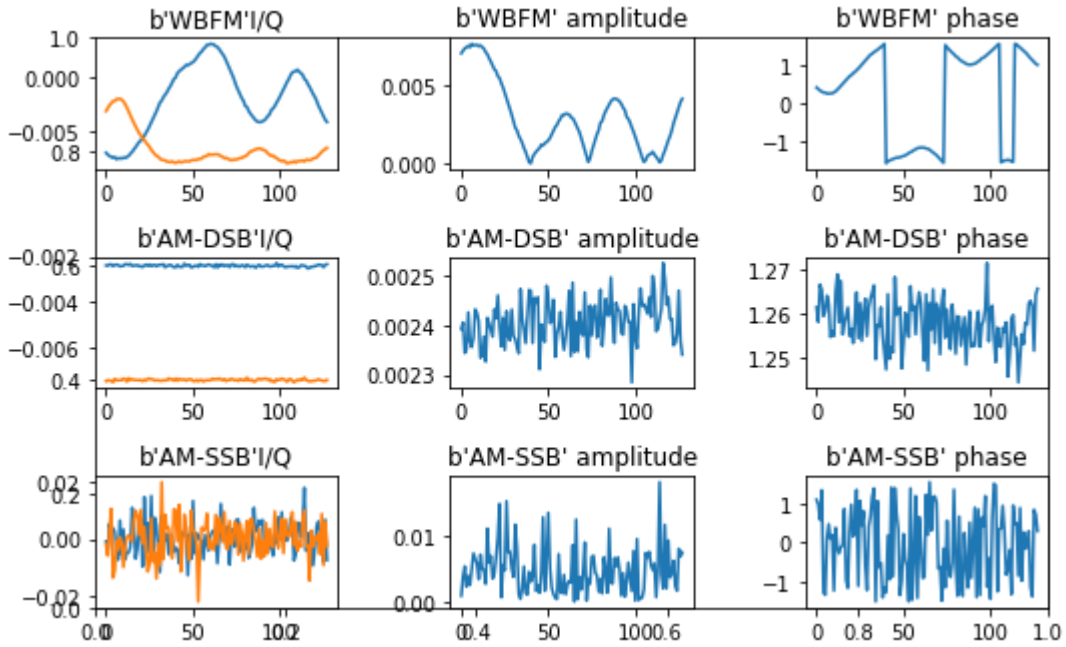


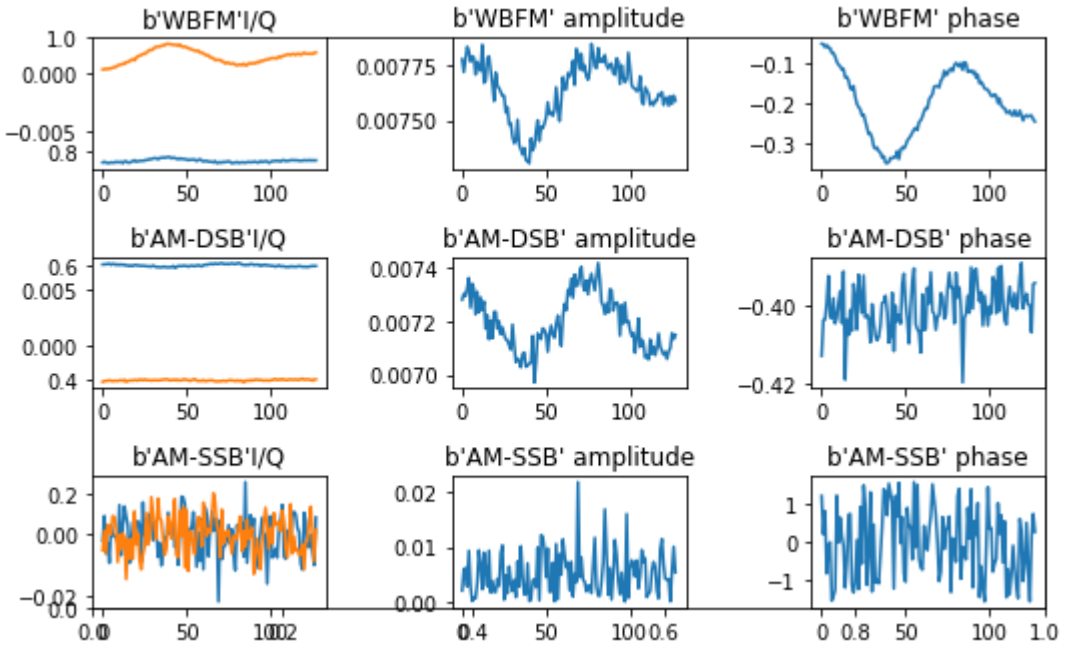
all samples dumped together - 128k points



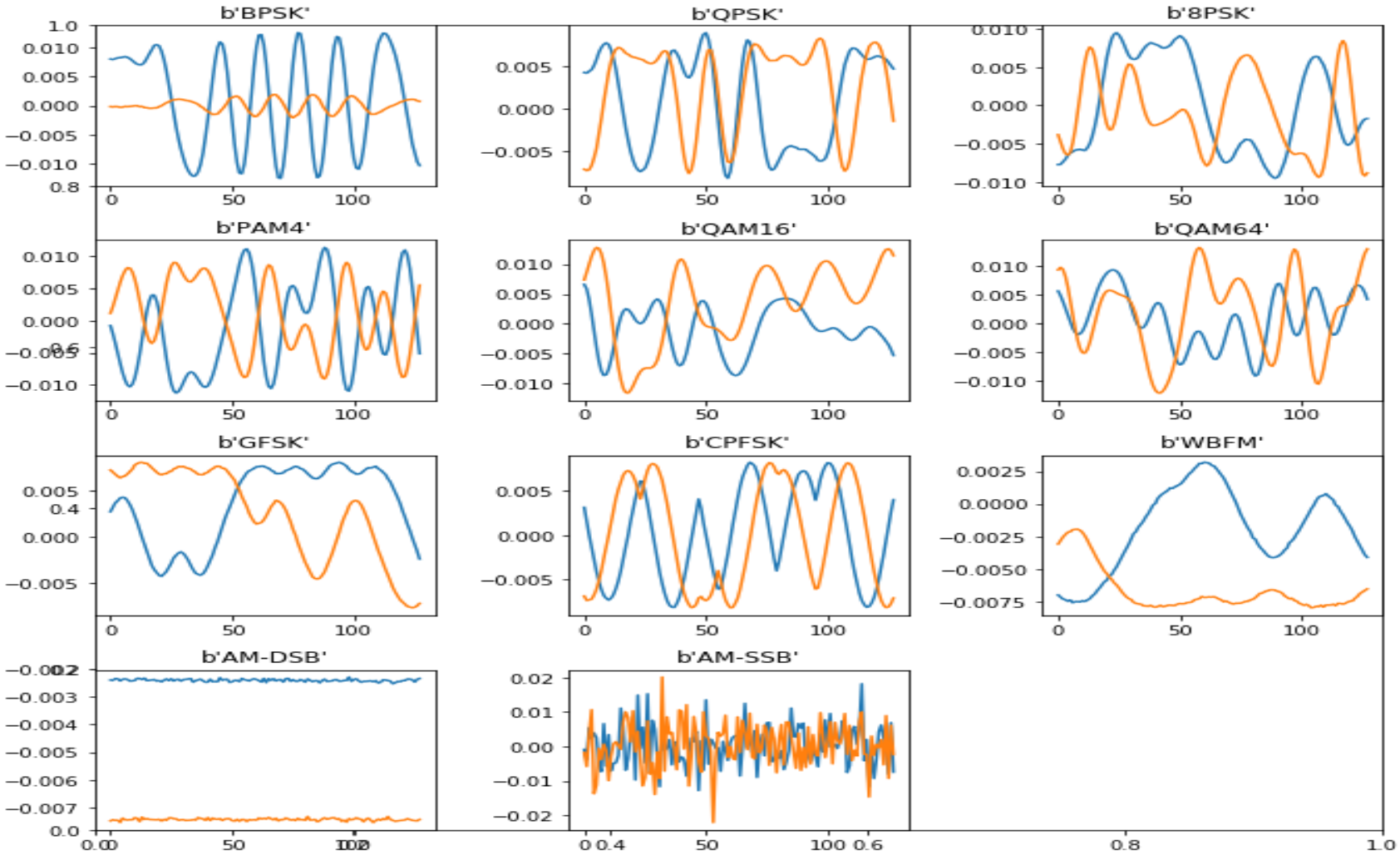
1. Data – problems with analogue mod

- I/Q, amplitude, and phase plots
- AM-DSB still weird…

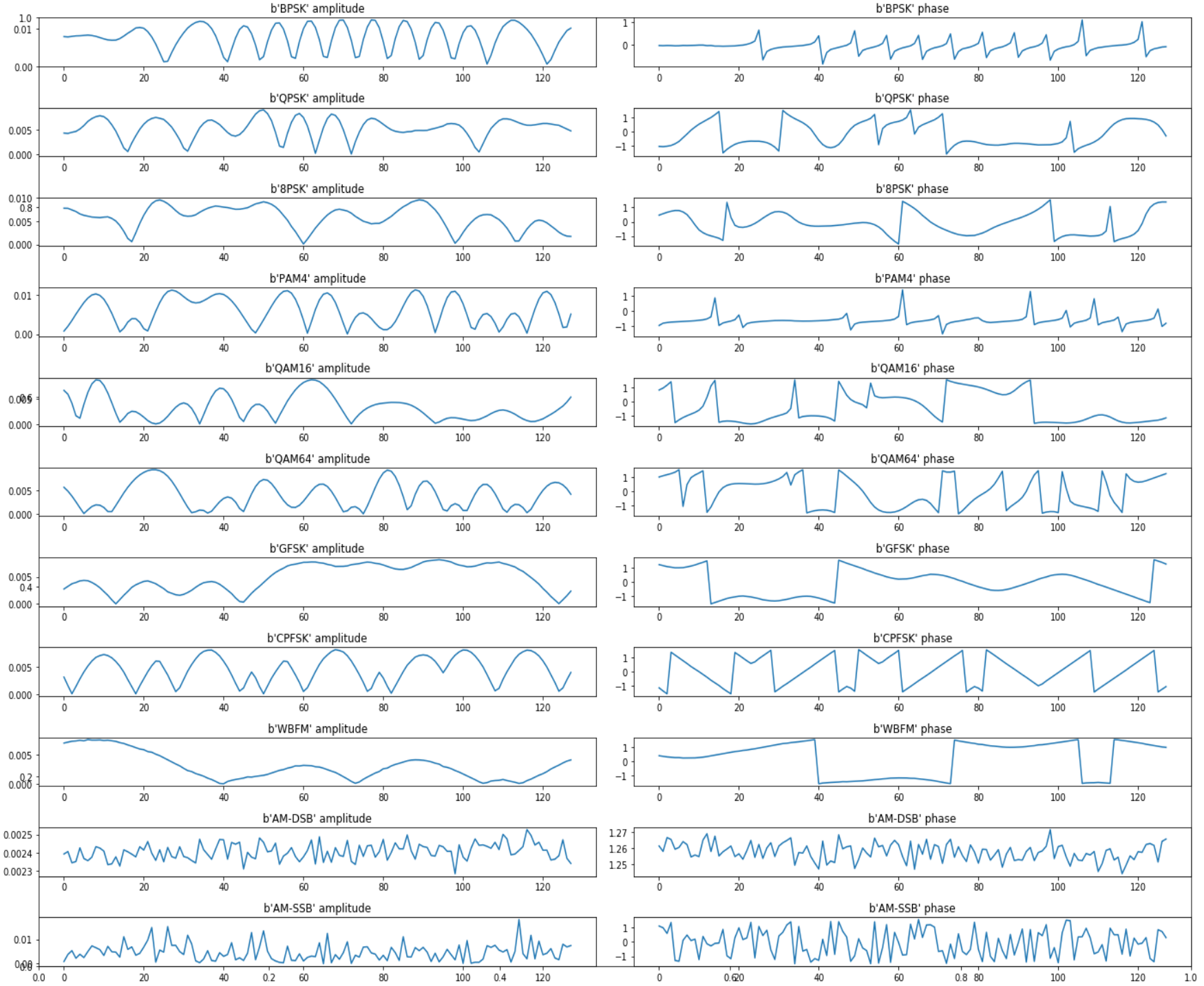




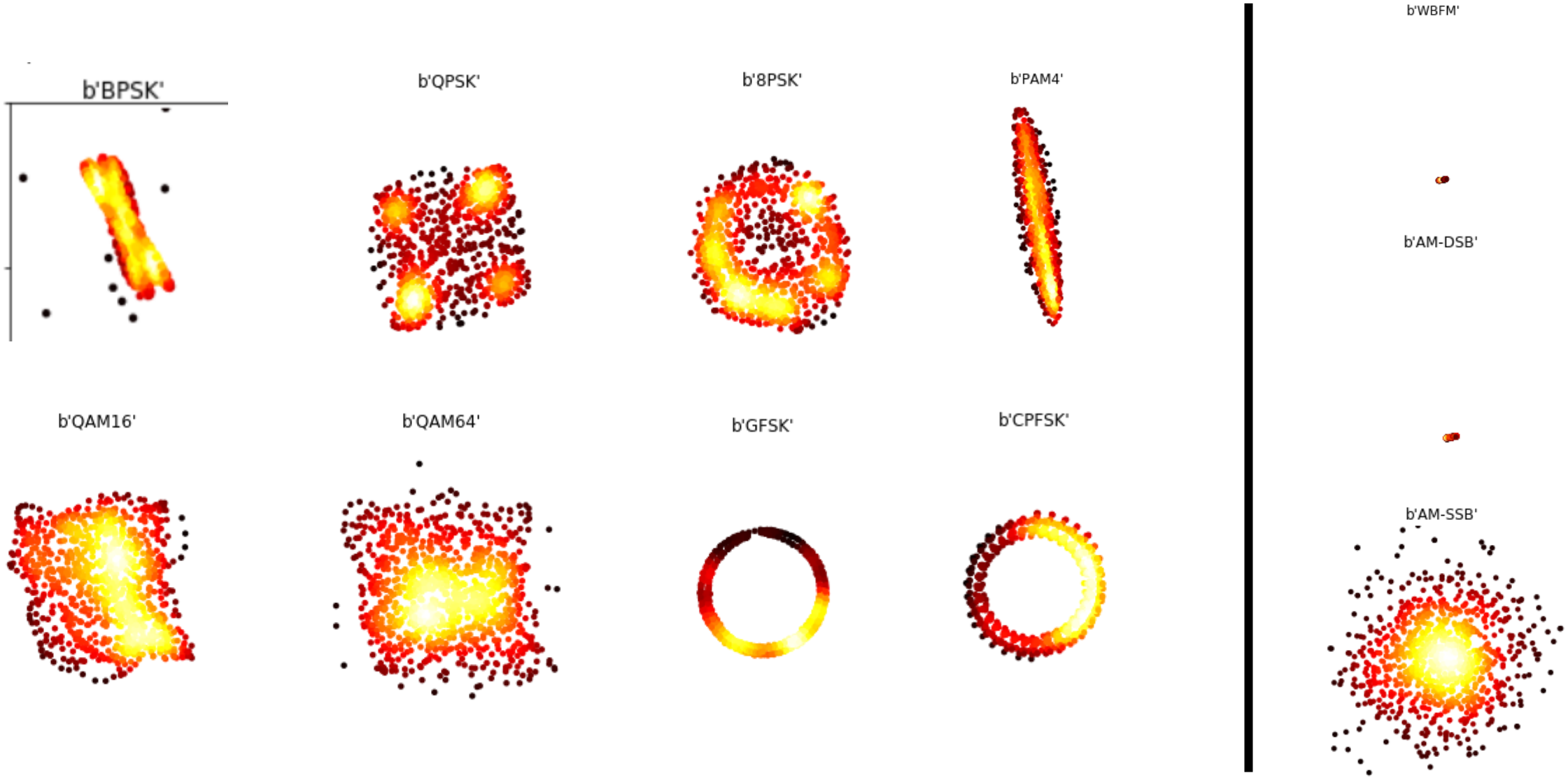
1. Data: I/Q time domain



1. Data: Amplitude-phase against time



1. Data: coloured constellation density plot



2. Training with different features with CNN

1. 2x128 time-series data (1000 samples per (mod, snr) class)
   - I/Q raw data 2x128
   - Amplitude phase data 2x128
2. Constellation image 128x128x3
   - With 128 points, single colour
   - With 1024 points, ‘hot’ heatmap

2.1 Training for 2x128 data: CNN models used

- CNN architectures from DeepSig papers (1)Convolutional Radio Modulation Recognition Networks and (2) Over-the-Air Deep Learning Based Radio Signal Classification.

- End-to-end Learning from Spectrum Data: A Deep Learning approach for Wireless Signal Identification in Spectrum Monitoring applications found that training with amplitude phase data improved accuracy by 2% compared to just I/Q data

2.1 Training: CNN Architectures

(1) CNN with 2 Conv + 2 Dense

TABLE 1. CNN structure.

Layer type	Input size	Parameters	Activation function
Convolutional layer	2×128	256 filters, filter size 1×3, dropout=0.6	ReLU
Convolutional layer	256×2×128	80 filters, filter size 2×3, dropout=0.6	ReLU
Fully connected layer	10240×1	256 neurons, dropout=0.6	ReLU
Fully connected layer	256×1	11 neurons or 15 neurons	Softmax

~50+ to 60+ epochs

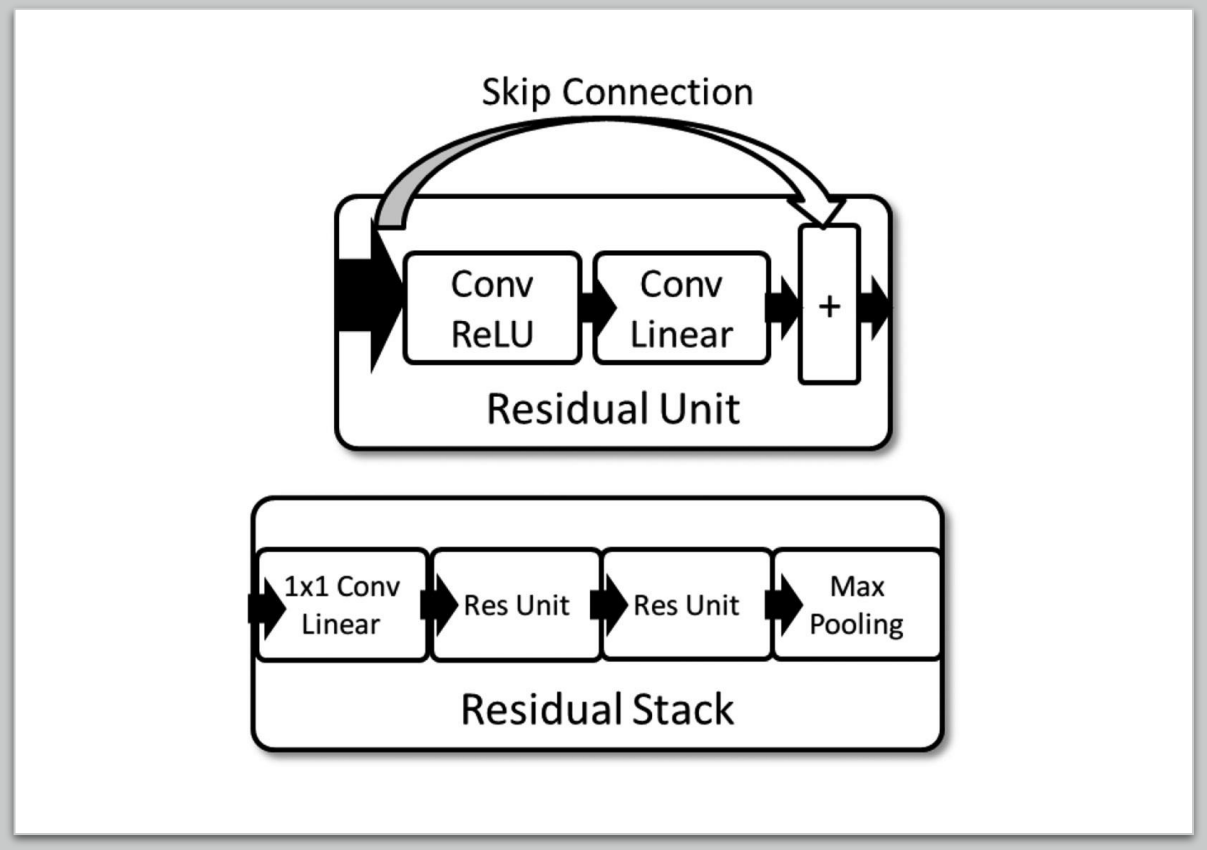
(2) ResNet → BETTER

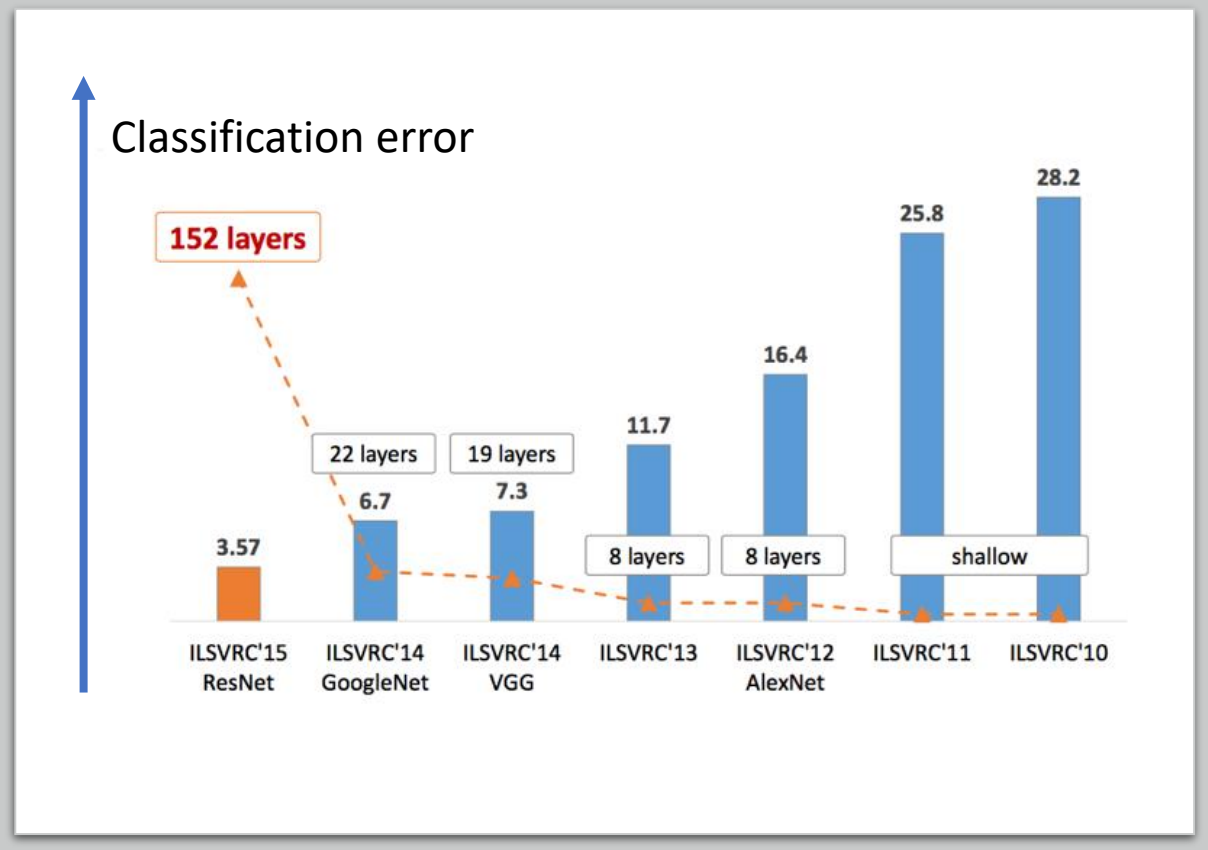
TABLE IV
RESNET NETWORK LAYOUT

Layer	Output dimensions
Input	2 × 128
Residual Stack	32 × 512
Residual Stack	32 × 256
Residual Stack	32 × 128
Residual Stack	32 × 64
Residual Stack	32 × 32
Residual Stack	32 × 16
FC/SeLU	128
FC/SeLU	128
FC/Softmax	24

~20+ epochs

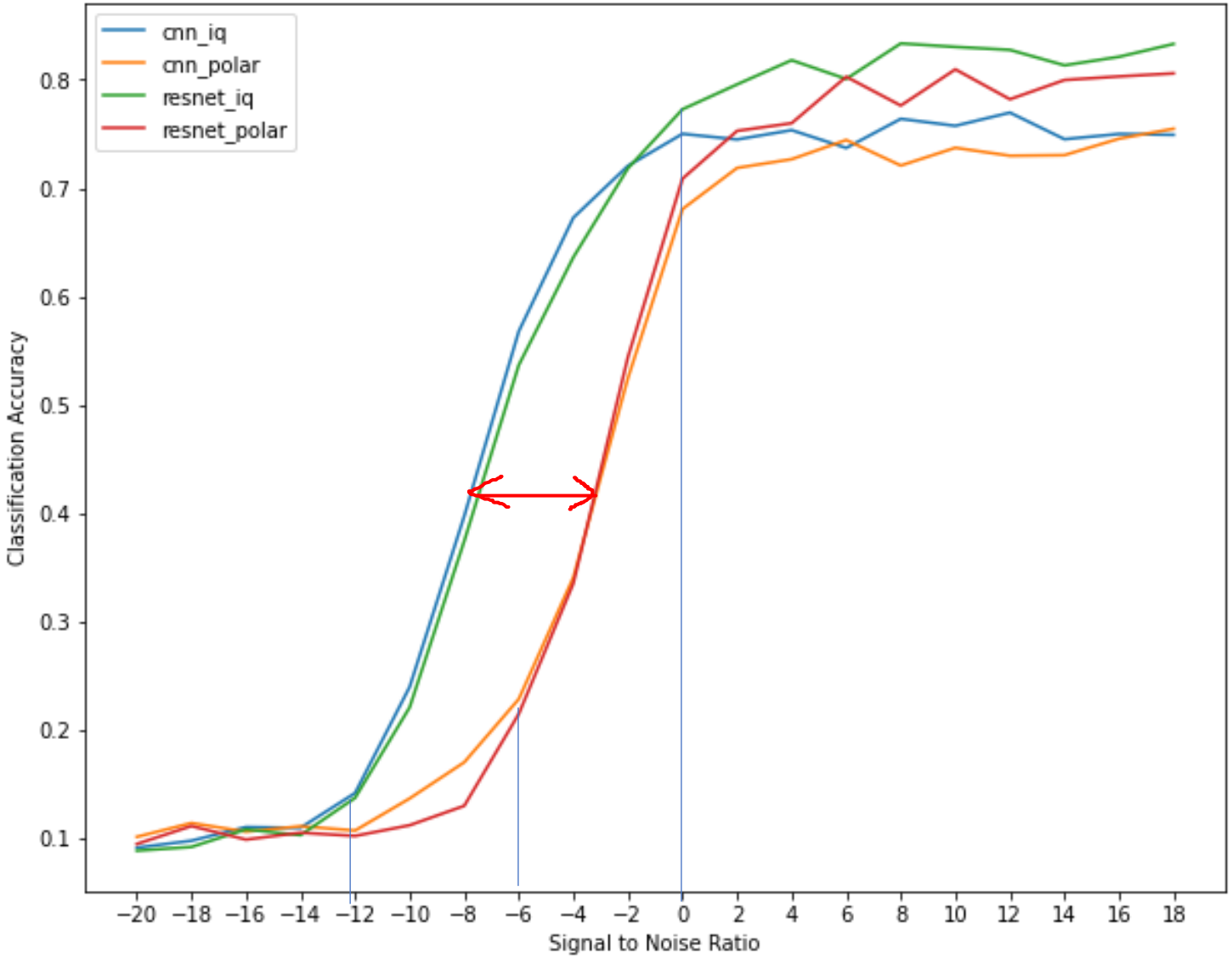
2.1 Training: ResNet Skip Connection



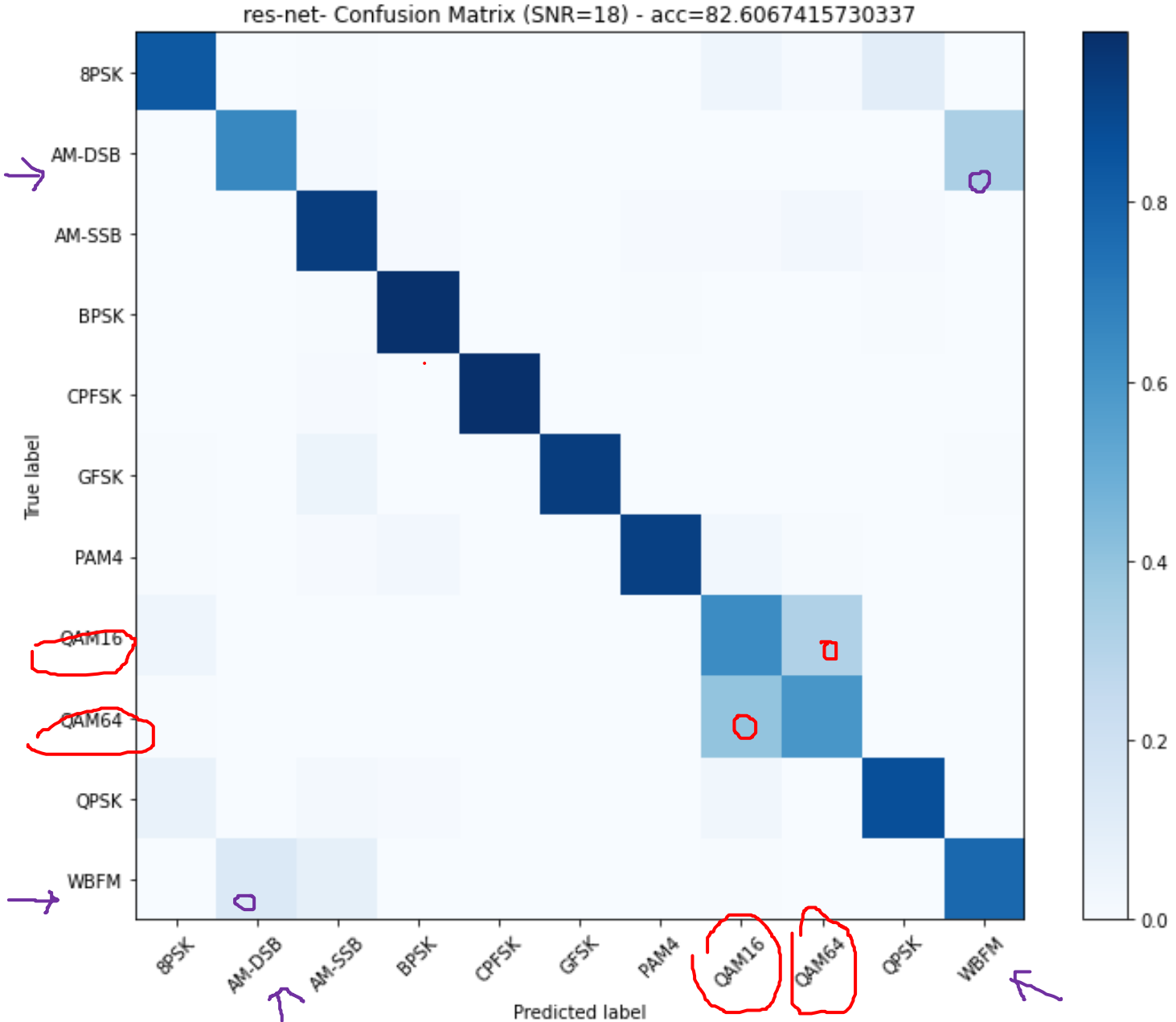


2.1 Training results – ResNet with I/Q best

- ResNet better → deeper, Multi-scale
- At high SNRs, ResNet ~82% accuracy, cnn ~72%
- Training with amp/phase worse, less resistant to low SNR → doesn't correspond to paper
- Why CNN even work with time-series data?



2.1 Training results – Confusion Matrix

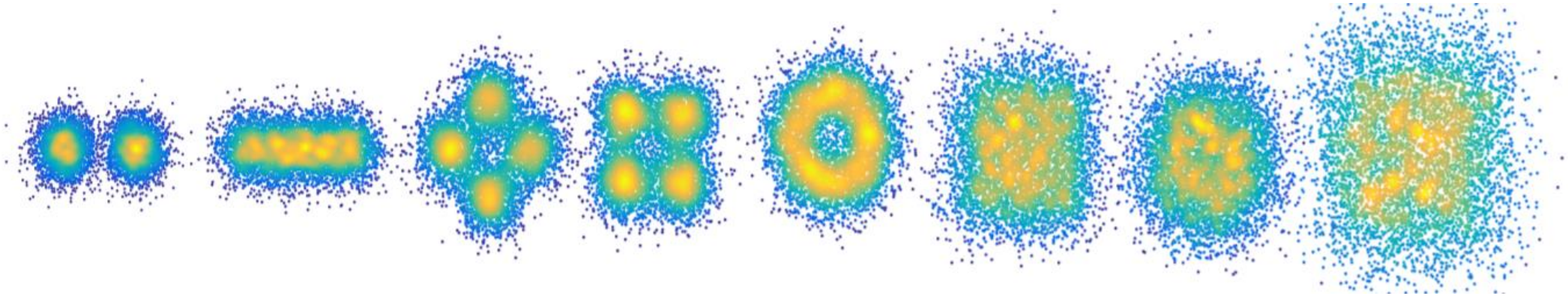


QAM16 ⇔ QAM64

AM-DSB ⇔ WBFM

2.2 Train with constellation image

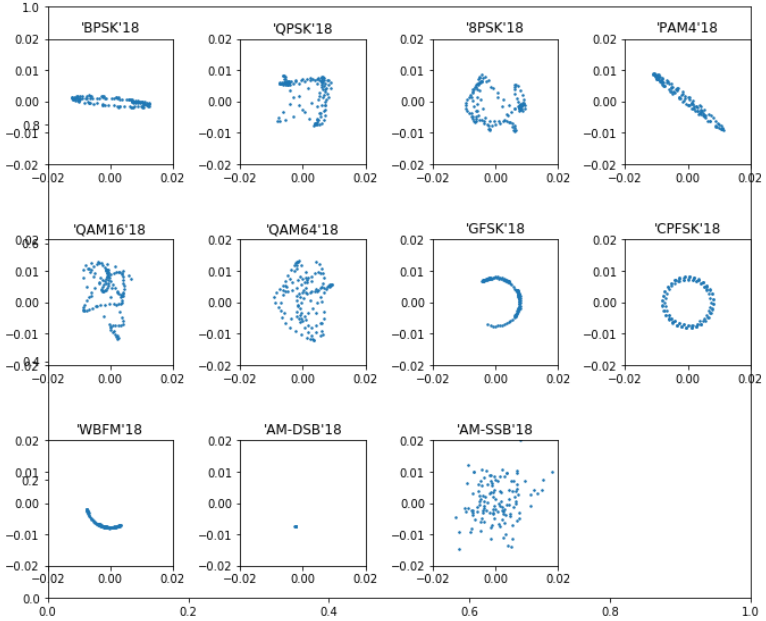
- Works better than time series with CNN because local neighbourhood got more defining features
- Papers (1) Data-Driven Deep Learning for Automatic Modulation Recognition in Cognitive Radios (2) Digital Signal Modulation Classification With Data Augmentation Using Generative Adversarial Nets in Cognitive Radio Networks achieved ~100% accuracy with coloured constellations → can distinguish different QAM



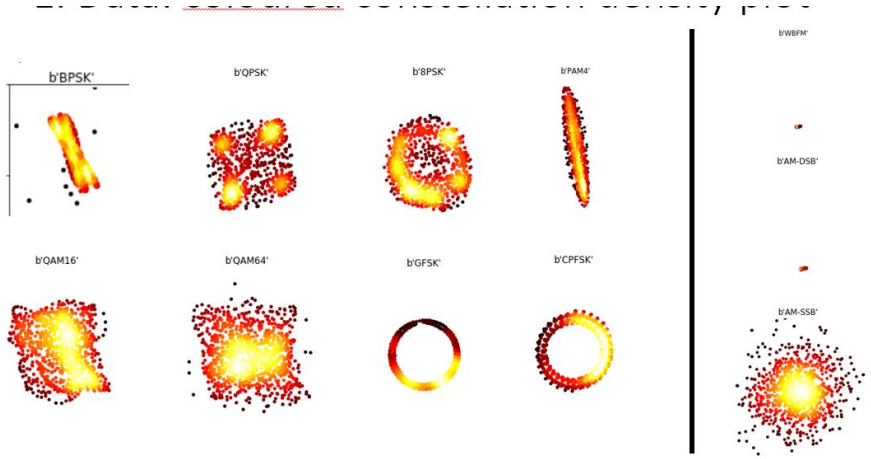
Paper's constellation diagrams at 4dB (dunno how many points used)

2.2 Constellation images used for training

1. 128 points, single colour
   - 200 images per class



1. 1024 points, colour
   - 125 images per class



*Haven't had time for 1024 single colour

2.2 Training: CNN architecture

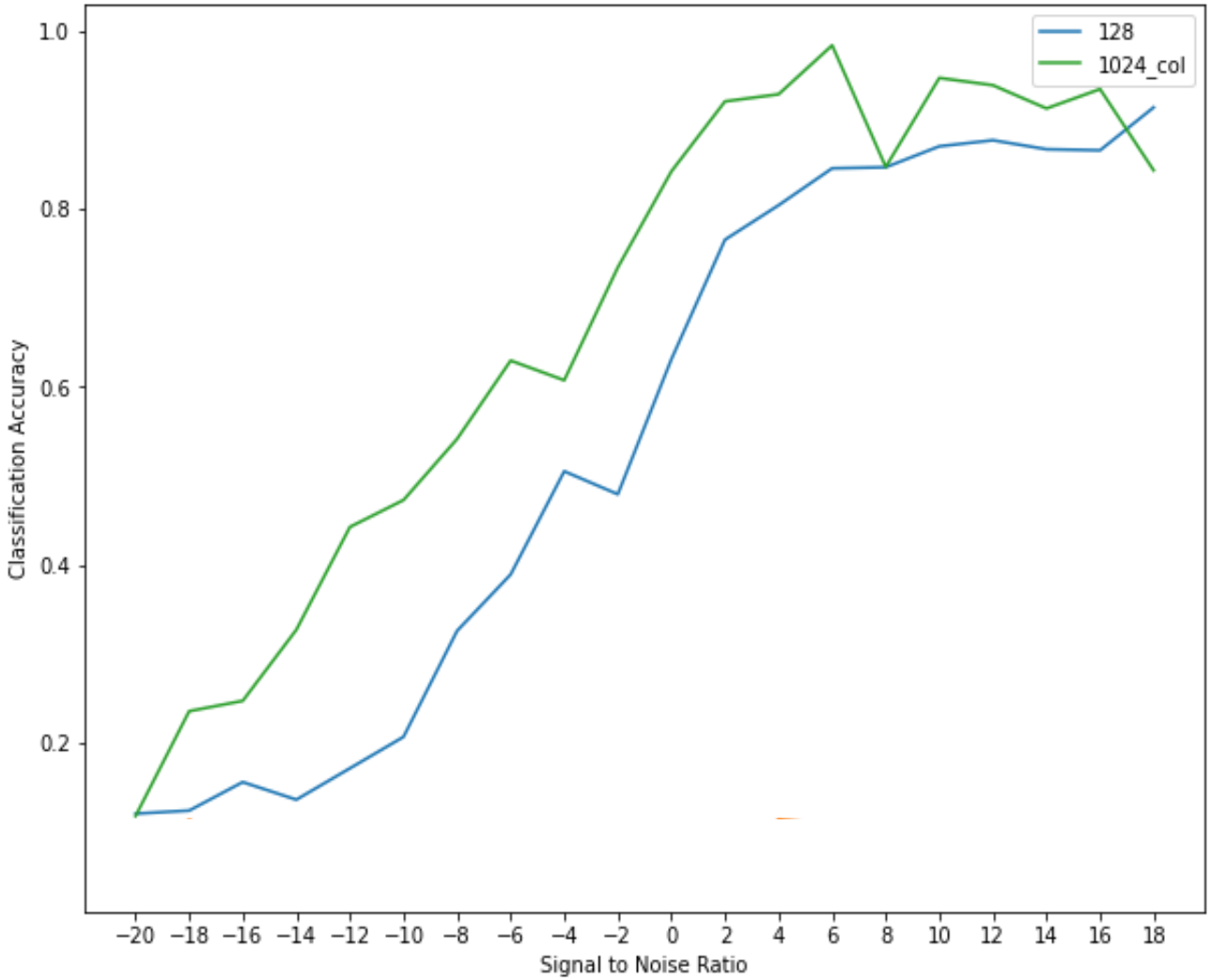
- From Data-Driven Deep Learning for Automatic Modulation Recognition in Cognitive Radios
- I used ReLU instead of PReLU for activation
- Tried making a ResNet but not much better than this
- Train ~10+ epochs

Layer	Output dimensions
Input	$128 \times 128 \times 3$
Conv2D(filters 128, size 5×5) + PReLU	$124 \times 124 \times 128$
Avepooling (size = 2, stride = 2)	$62 \times 62 \times 128$
Conv2D(filters 64, size 3×3) + PReLU	$62 \times 62 \times 64$
Conv2D(filters 64, size 3×3) + PReLU	$62 \times 62 \times 64$
Avepooling (size = 2, stride = 2)	$30 \times 30 \times 64$
Conv2D(filters 32, size 3×3) + PReLU	$30 \times 30 \times 32$
Conv2D(filters 32, size 3×3) + PReLU	$30 \times 30 \times 32$
Avepooling (size = 2, stride = 2)	$15 \times 15 \times 32$
Flatten	7200
Dense + PReLU	1024
Dropout (0.5)	/
Dense + PReLU	512
Dropout (0.5)	/
Dense + Softmax	modulation modes

2.2 Training with constellation results

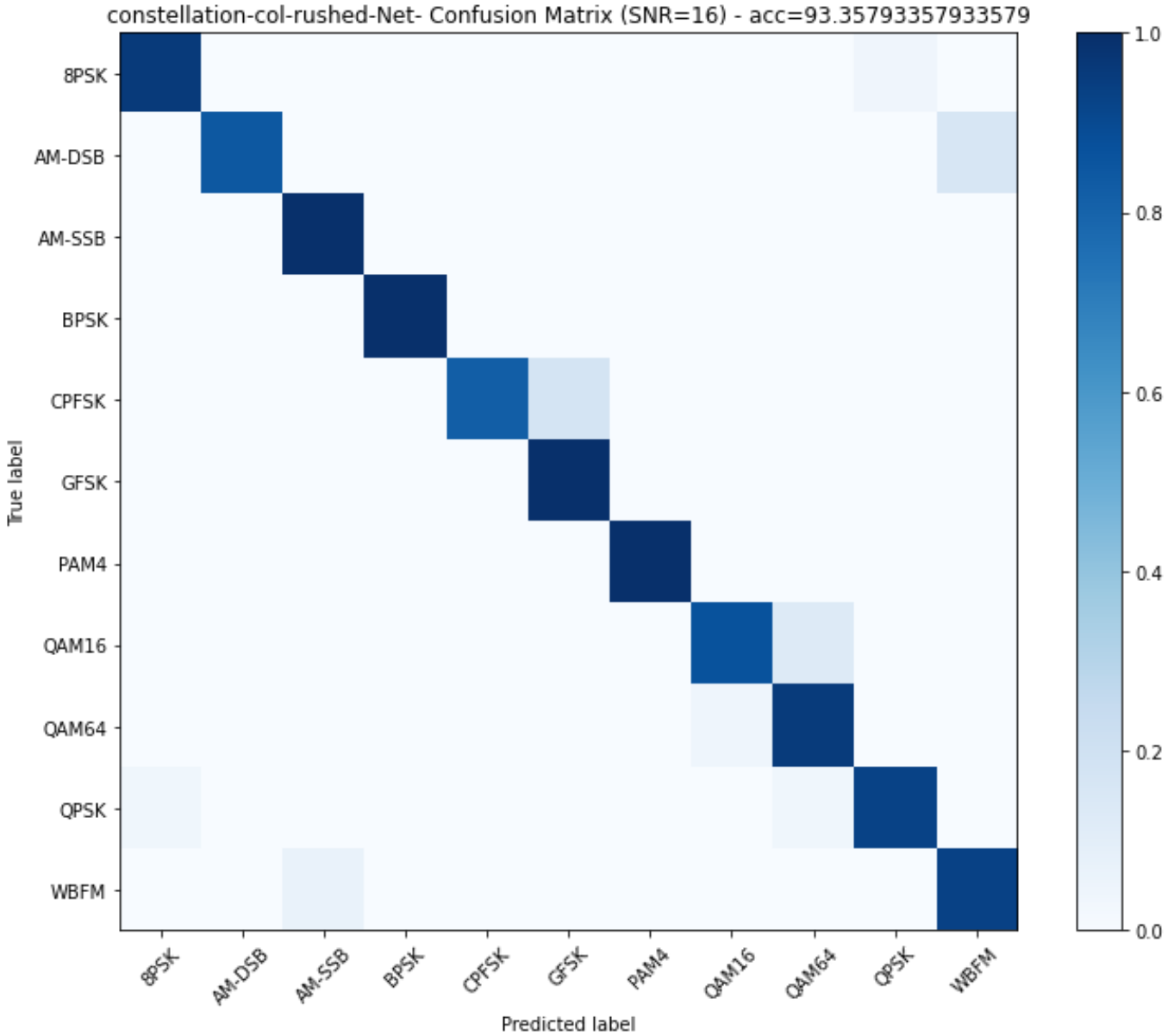
- For high SNRS, 1024_col ~90% accuracy, 128 ~82%
- 1024_col more resistant to noise

- Clearly, image better than time series
- Likely more points better
- Likely coloured better



2.2 Training results: Confusion Matrix

- QAM confusion less
- CPFSK, GFSK confuse more



3. Next Steps

1. Finish up current work, neaten up documentation, push to github
   - Analysing training results: visualize features extracted by layers, LIME?, why kinks in constellation graphs (try different test set)
   - Examine why my reproductions of papers (Amp-phase, I/Q), (coloured constellations) not as good
     - Hyperparameters (Talos)? Dataset differences? Train-test splits? Epochs?
   - Can further work on constellation classification with CNN
     - Fine tune / new architectures, different colour maps, different number of points
2. More work on I/Q modulation theory, better understanding of how dataset is generated
   - Explore 2018.01A dataset with 24 modulation schemes
   - Try running data generation code
   - Try getting different dataset
3. Try to exploit temporal feature? RNN?