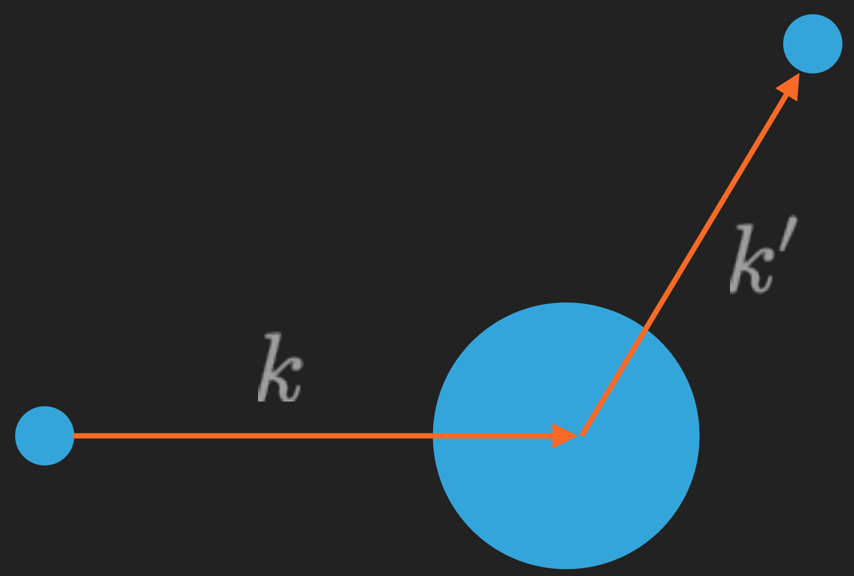
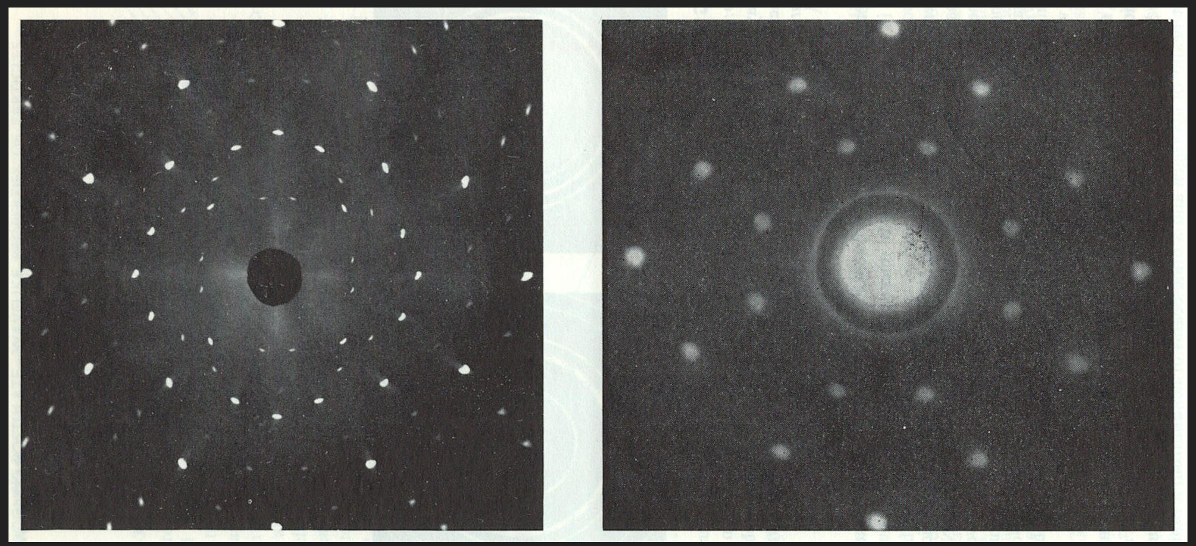
KEITH T. BUTLER

ANALYSING AND UNDERSTANDING INELASTIC NEUTRON SCATTERING WITH DEEP LEARNING

INELASTIC NEUTRON SCATTERING

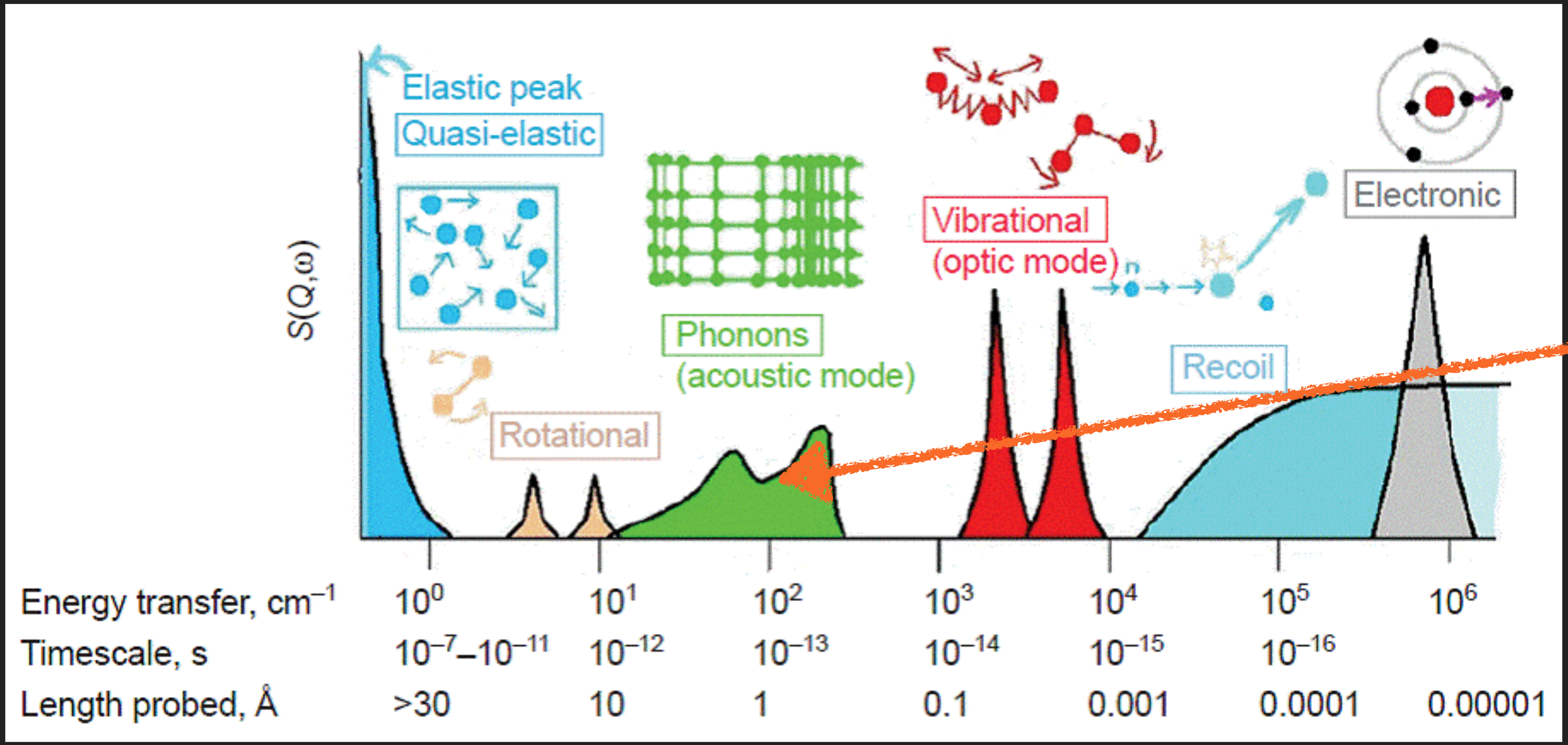


- Elastic scattering - no energy is transferred to the sample - diffraction pattern
- Inelastic scattering some energy is transferred to the sample



INELASTIC NEUTRON SCATTERING

- Inelastic energy transfer can occur due to many processes
- Inelastic events give spectra

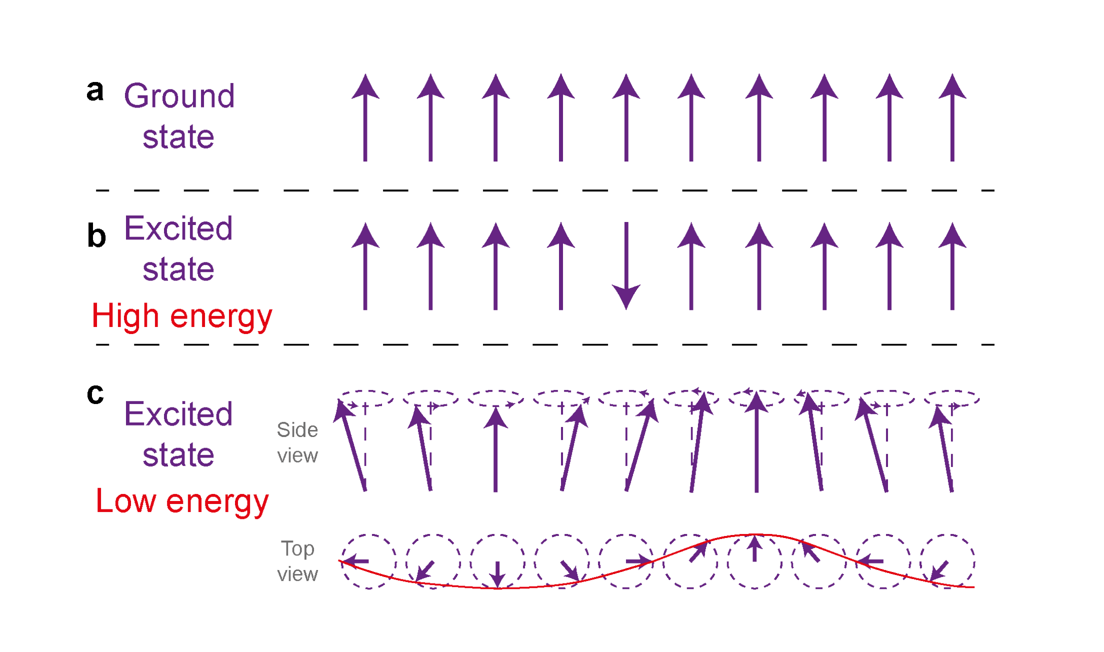


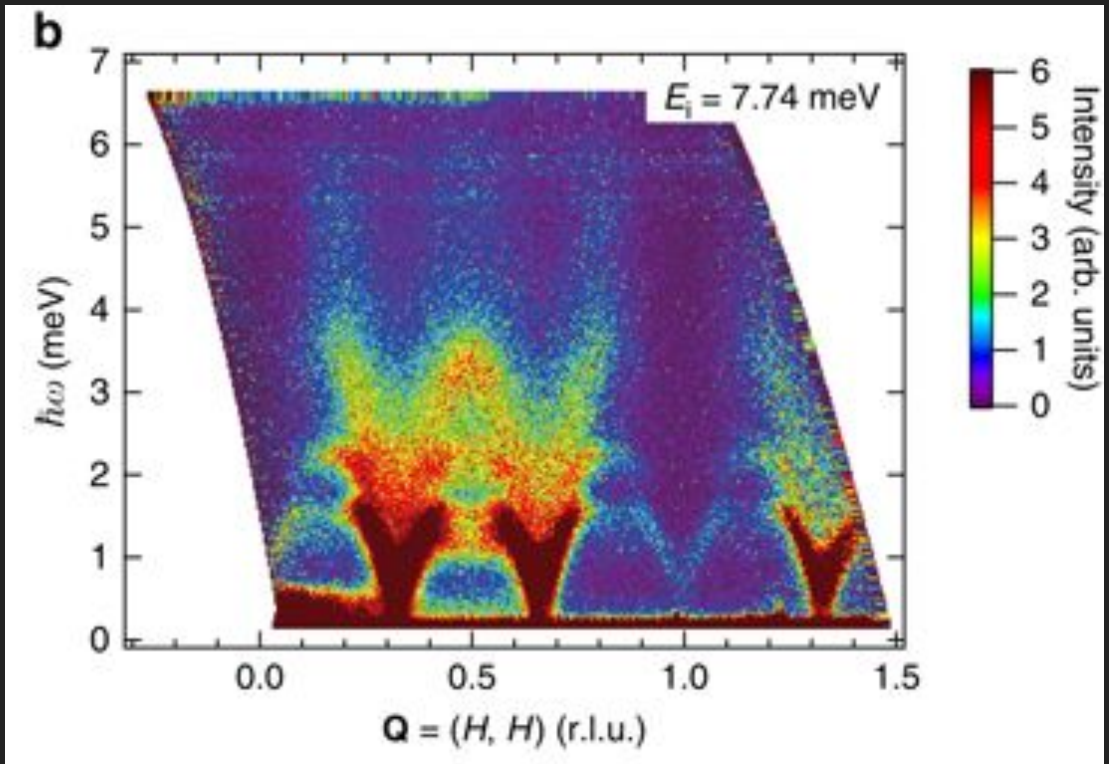
And magnons

Science and Technology Facilities Council

MAGNONS

- Magnons are low energy excited states of electrons
- Spin on one electron is perturbed and propagates through the lattice resulting in a wave of reorganisation
- Dependent on the magnetic structure of a material



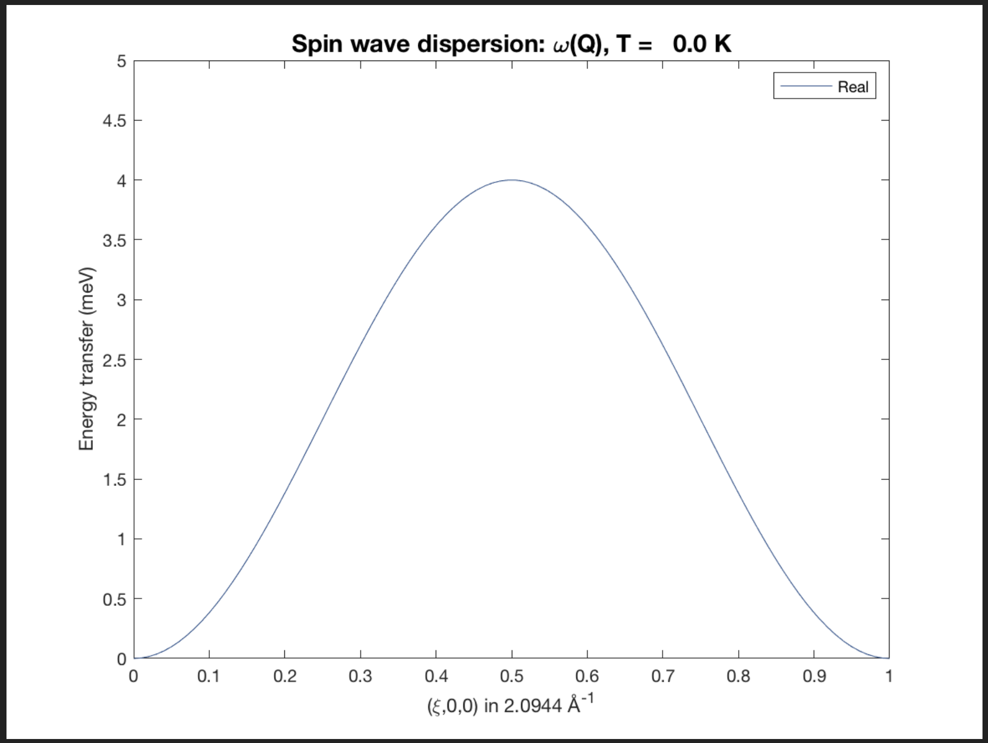


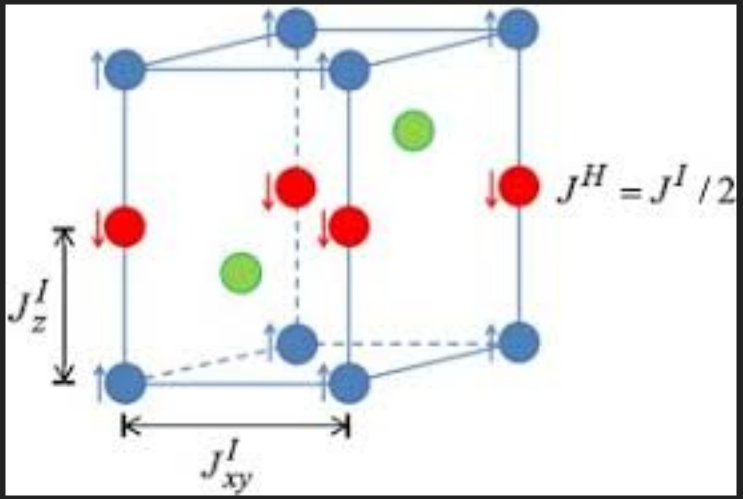
Nature Communications **volume 8**, Article number: 235 (2017)

LINEAR SPIN-WAVE THEORY

$$H = \sum_{mi,nj} J_{mi,nj} S_i S_j$$

$$H = \sum_{i,j} J_{i,j}(q)[\frac{\sqrt{S_i S_j}}{2}(b_{i,q} b^{\dagger}_{j,q} + b^{\dagger}_{i,q} b_{j,q}) - S_i b_{i,q} b^{\dagger}_{i,q} - S_j b^{\dagger}_{j,q} b_{j,q}]$$





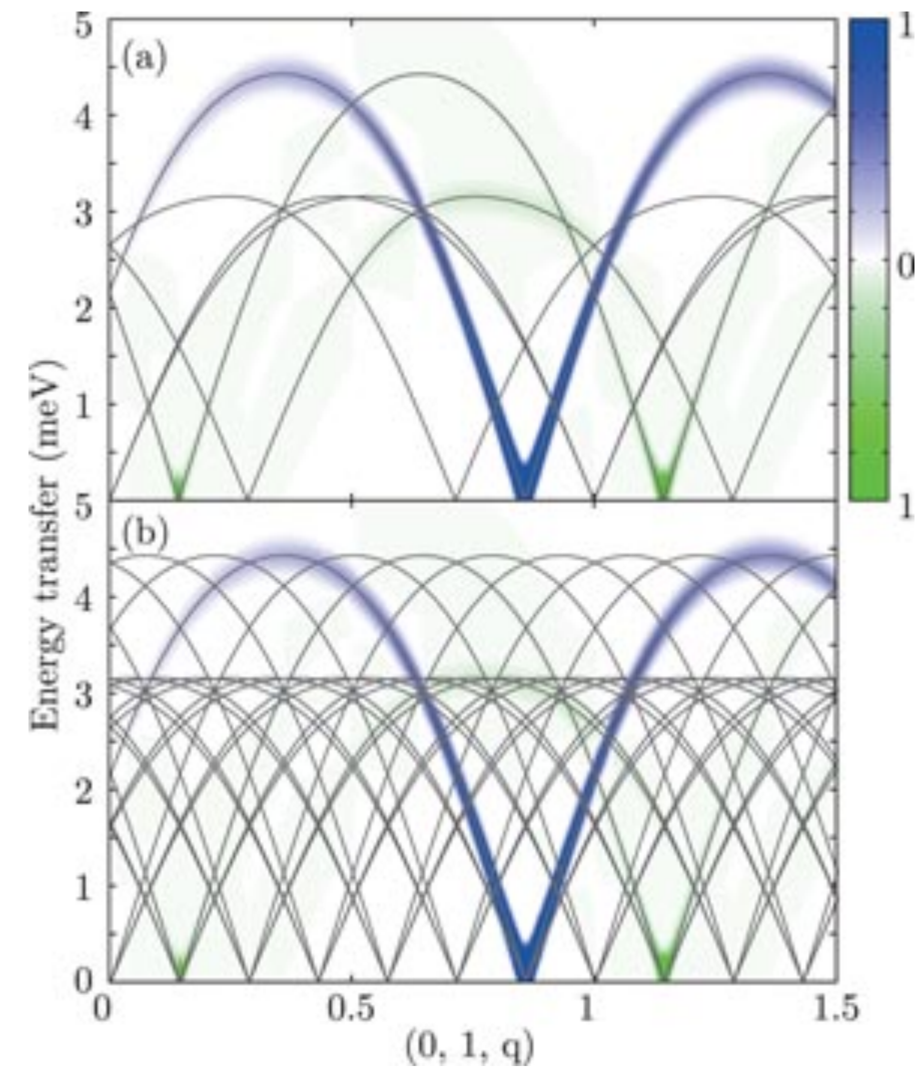
Linearise

Fourier transform

Provides a 'dispersion relation' energy vs q

SOLVING LINEAR SPIN WAVE THEORY: SPIN-W

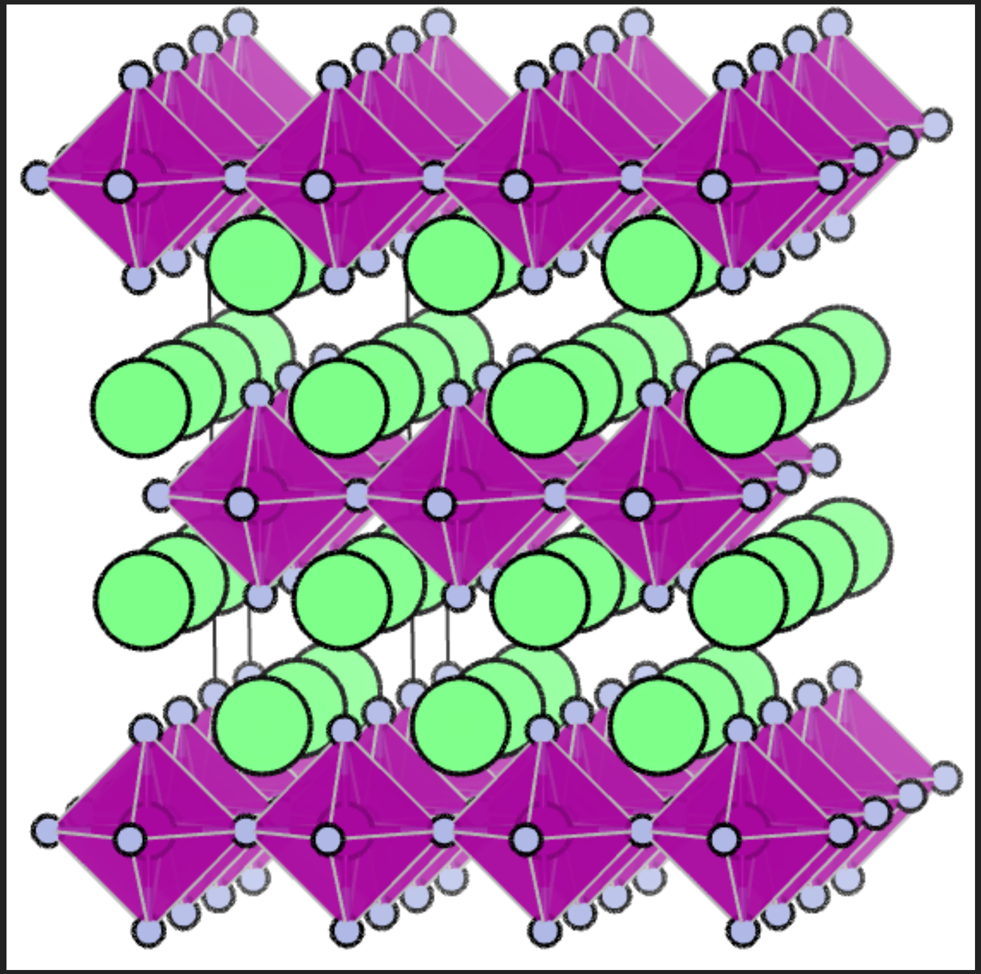
- Numerical solver for the linear spin wave Hamiltonian
- **Input** : Magnetic moments, lattice, model of interactions
- **Output**: Simulated spectrum - can numerically fit to experiment



Journal of Physics: Condensed Matter, Volume 27, Number 16

RB2MNF4

- 2D Antiferromagnet
- Interactions in planes of MnF
- Mostly described by linear spin wave theory

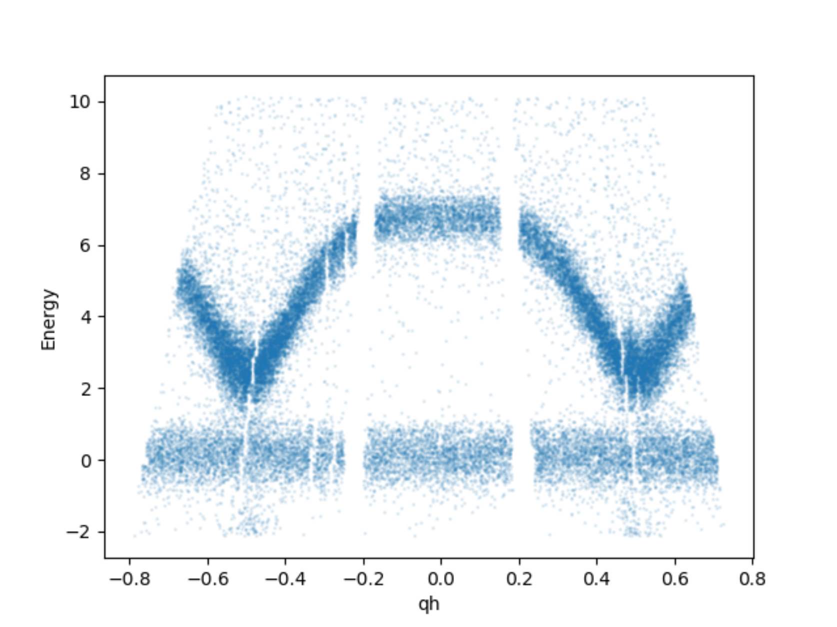


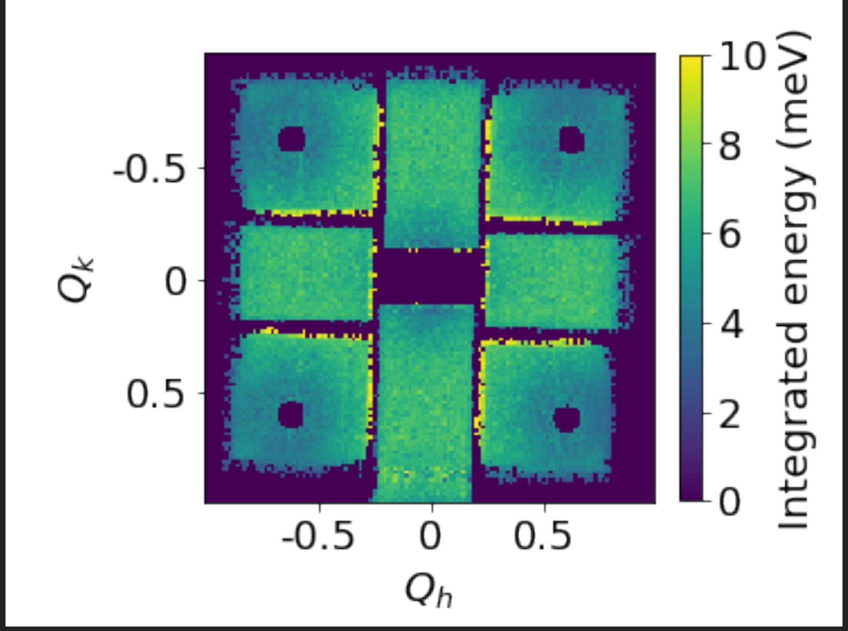
Two-magnon excitations observed by neutron scattering in the two-dimensional spin-$\frac{5}{2}$ Heisenberg antiferromagnet $\mathrm{Rb_2\,Mn\,F_4}$

T. Huberman, R. Coldea, R. A. Cowley, D. A. Tennant, R. L. Leheny, R. J. Christianson, and C. D. Frost
Phys. Rev. B **72**, 014413 – Published 6 July 2005

RB2MNF4 THE DATA

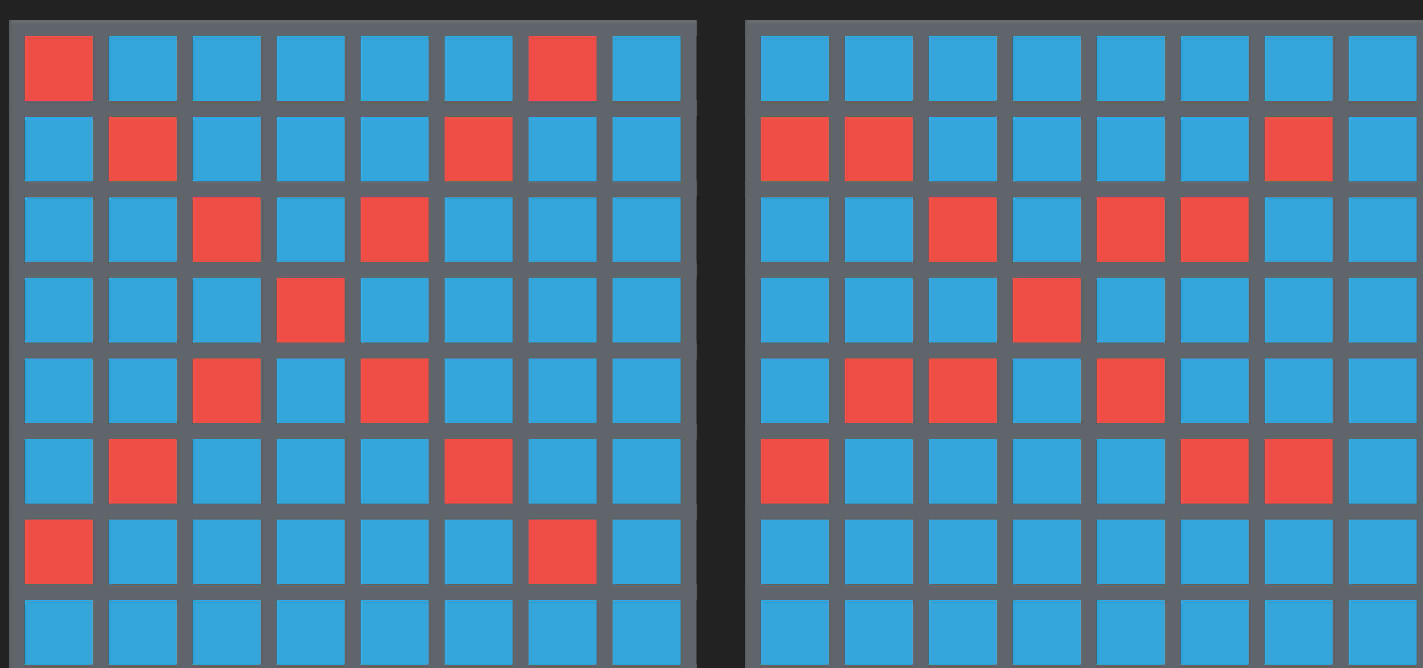
- Clean data-set
- Single magnon dispersion band
- Remove Bragg peaks and integrate the signal intensity across the energy range
- 2D map in Qh/Qk
- Can we train a model to estimate the exchange constants?

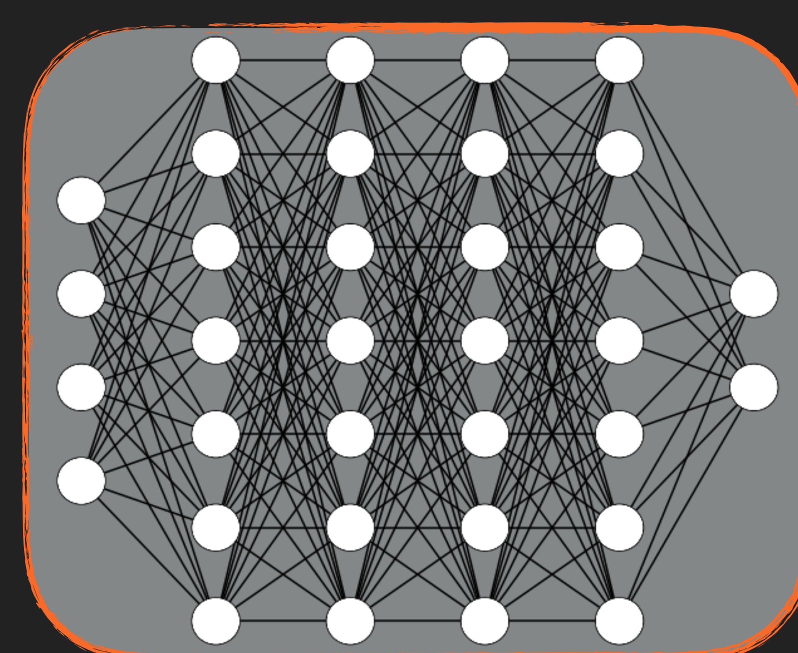




MLPS STRUGGLE WITH IMAGE RECOGNITION

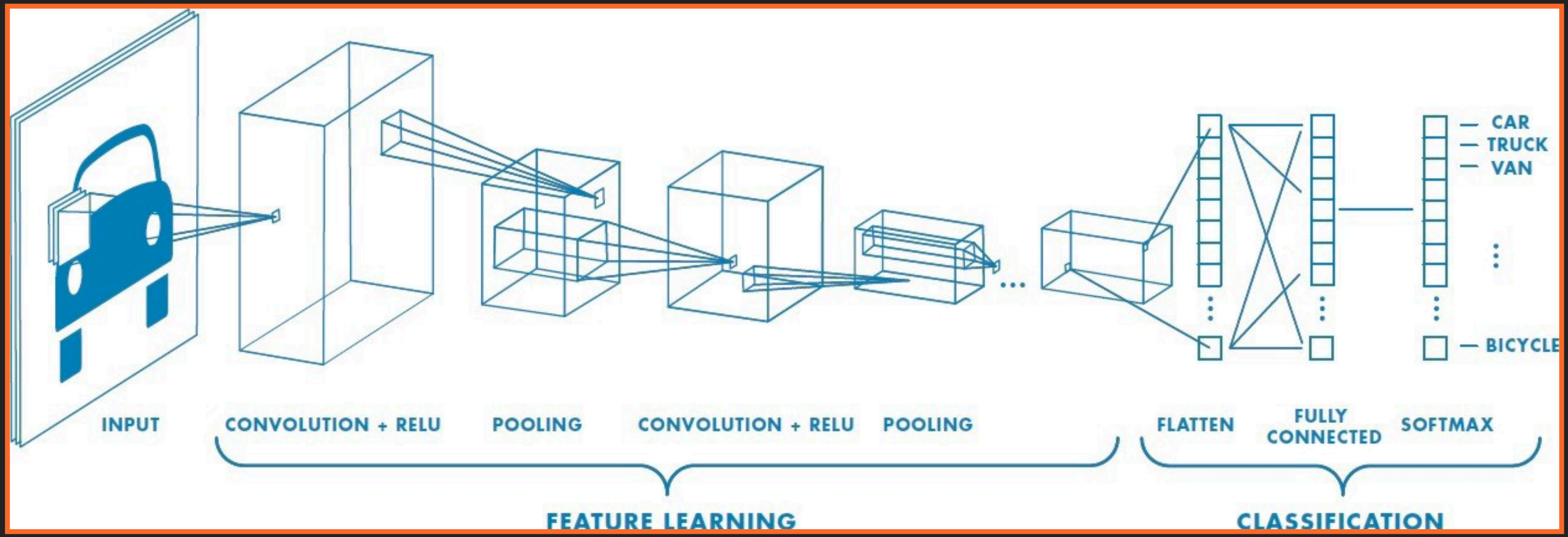
- For a computer, literally these do not match
- The MLP has no real concept of the spatial relations
- Also, dense connections lead to parametric explosions for many pixel images

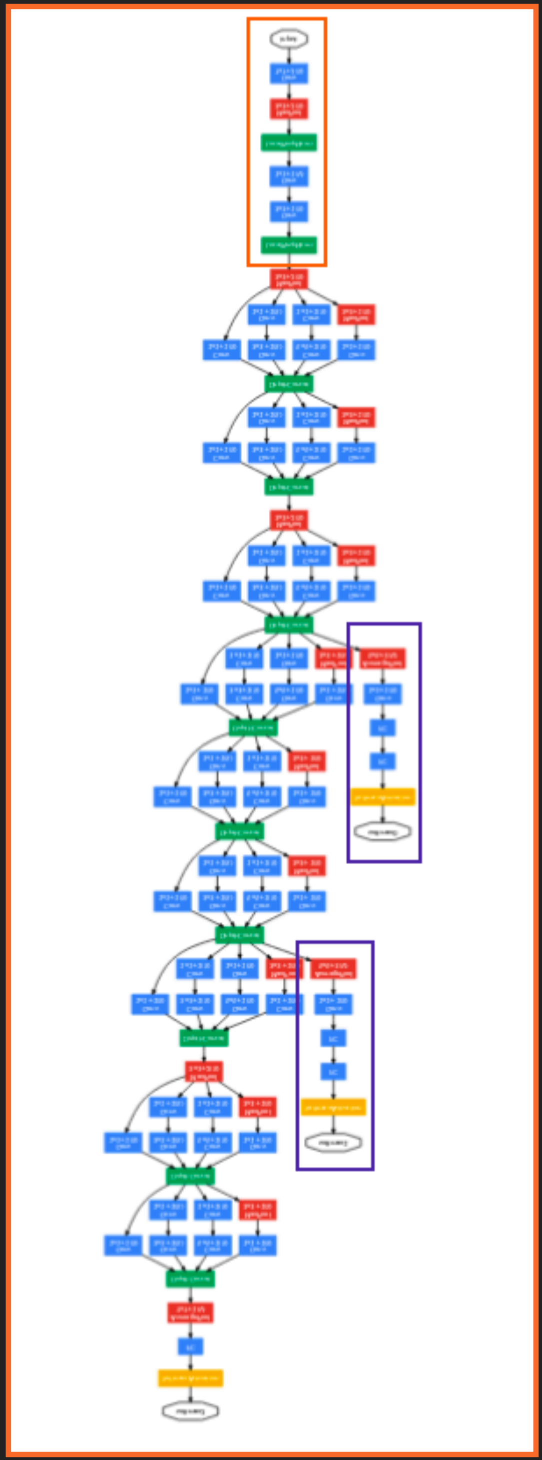




CONVOLUTIONAL NEURAL NETS (CNNS)

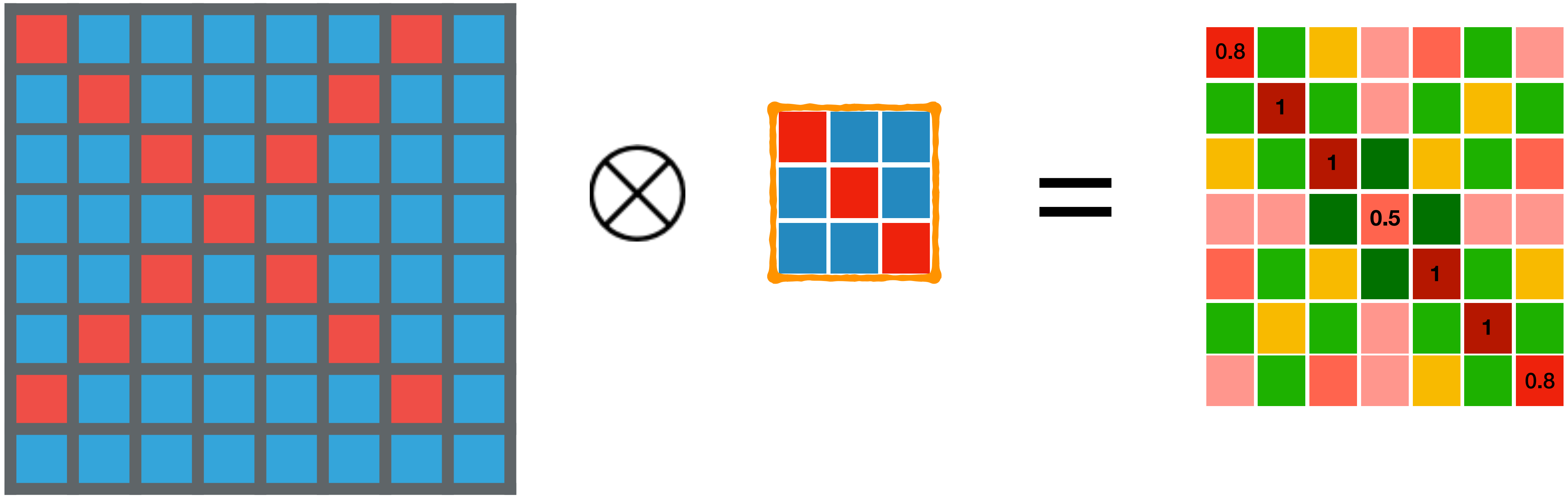
- Uses filters to pick out important features
- Compresses image information
- Is finally connected to a typical NN layer
- Successful CNNs are often very deep





HOW CNNS WORK

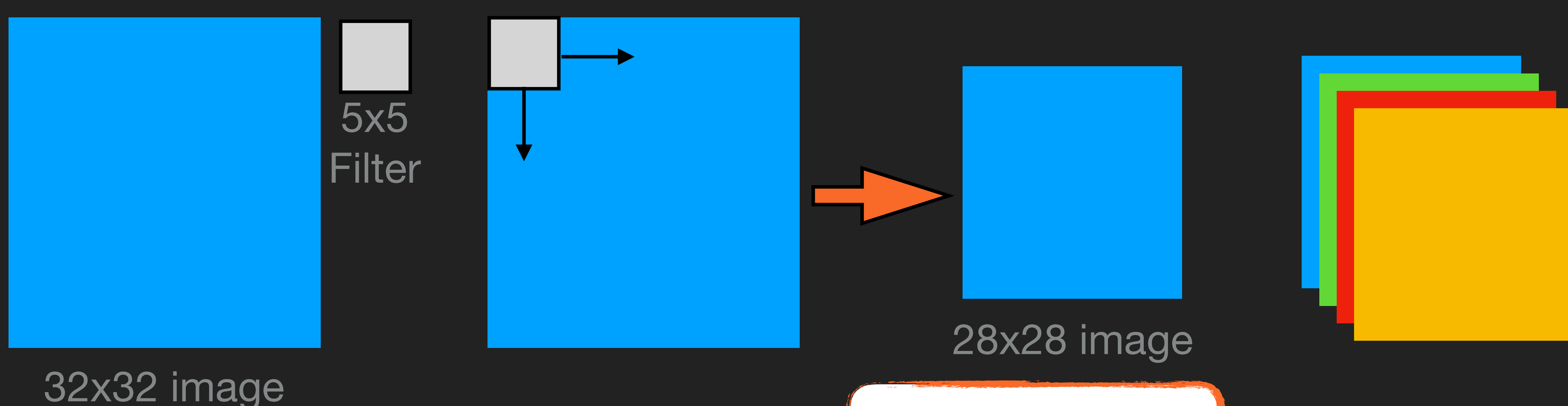
- Filters work to pick out features in the image



HOW CNNS WORK THE FILTERS

$$\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{15} \\ x_{21} & x_{22} & \cdots & x_{25} \\ \cdots & \cdots & \cdots & \cdots \\ x_{51} & x_{52} & \cdots & x_{55} \end{bmatrix}$$

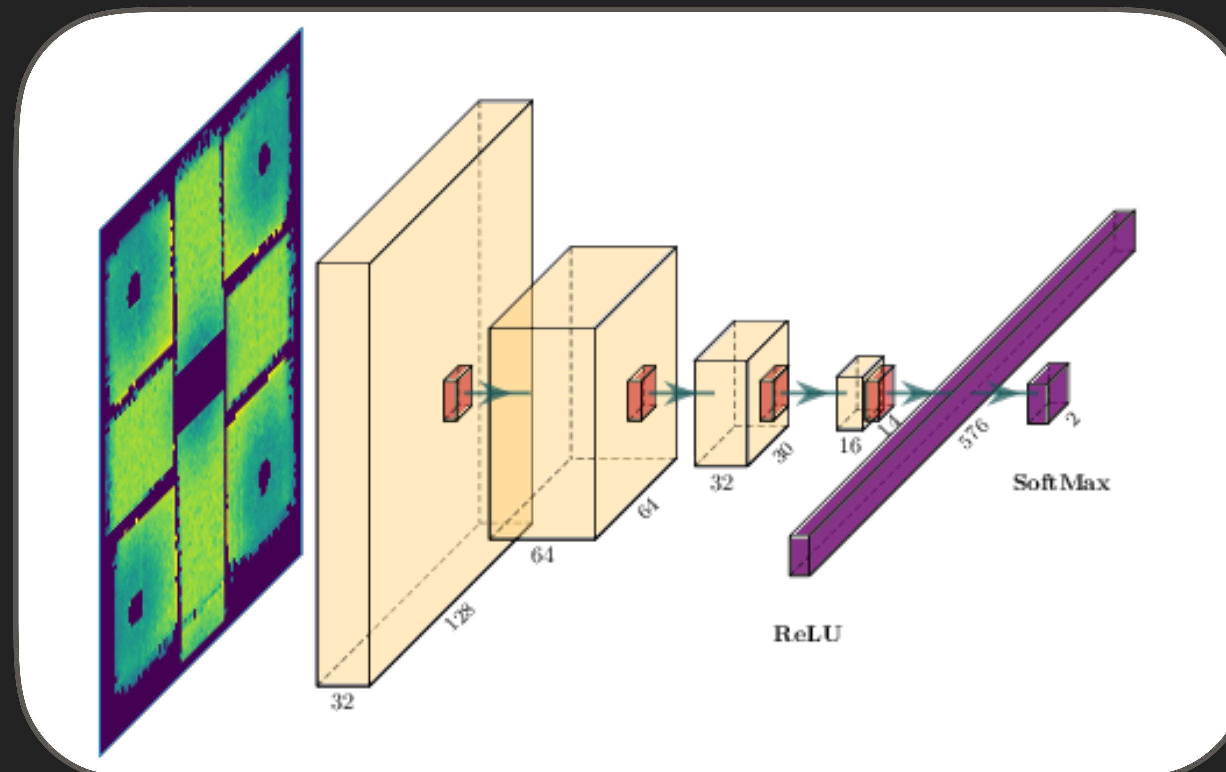
- Filter is a matrix
- Filter dot product with image to produce scalar
- A number of filters are added at each layer



$$I \cdot F = s$$

RB2MNF4 RESULTS

- Simple neural network with 4 convolutional layers - extract features
- Function approximation from two layer MLP



Literature values:

- $J_1 = 0.657 \pm 0.002$
- $J_2 = 0.006 \pm 0.003$

or:

- $J_1 = 0.673 \pm 0.028$
- $J_2 = 0.012 \pm 0.002$

$$J_1 = 0.676$$
$$J_2 = 0.014$$

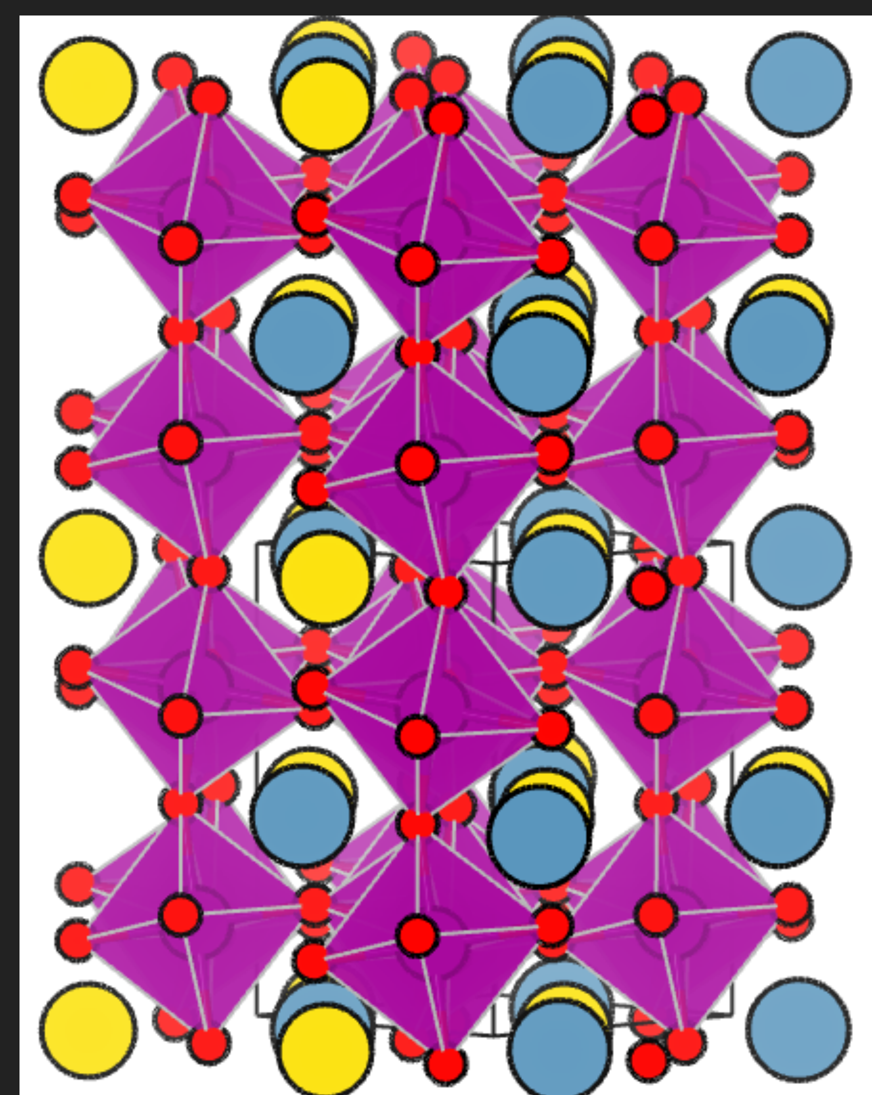
PCSMO THE SYSTEM

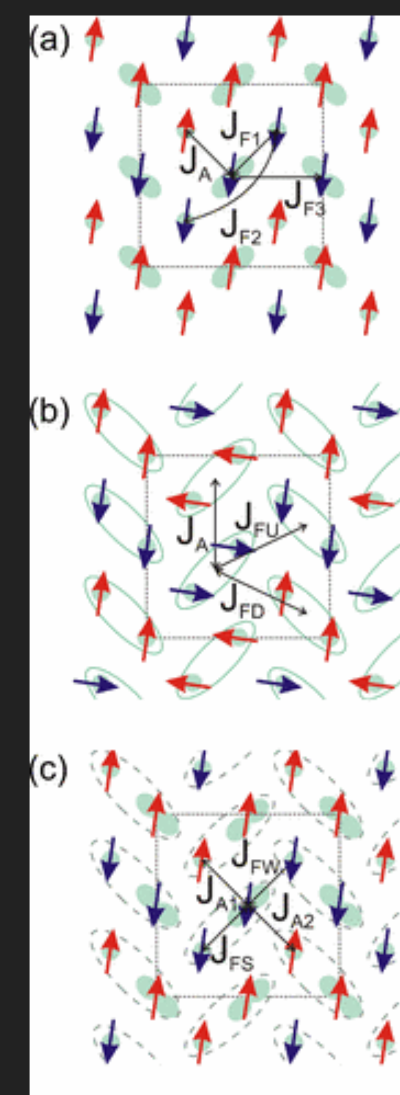
- Double perovskite
- Mixed A-site
- Several possible models for the magnetism
- Goodenough model
- Zener polaron
- Dimer model

Ground State in a Half-Doped Manganite Distinguished by Neutron Spectroscopy

G. E. Johnstone, T. G. Perring, O. Sikora, D. Prabhakaran, and A. T. Boothroyd

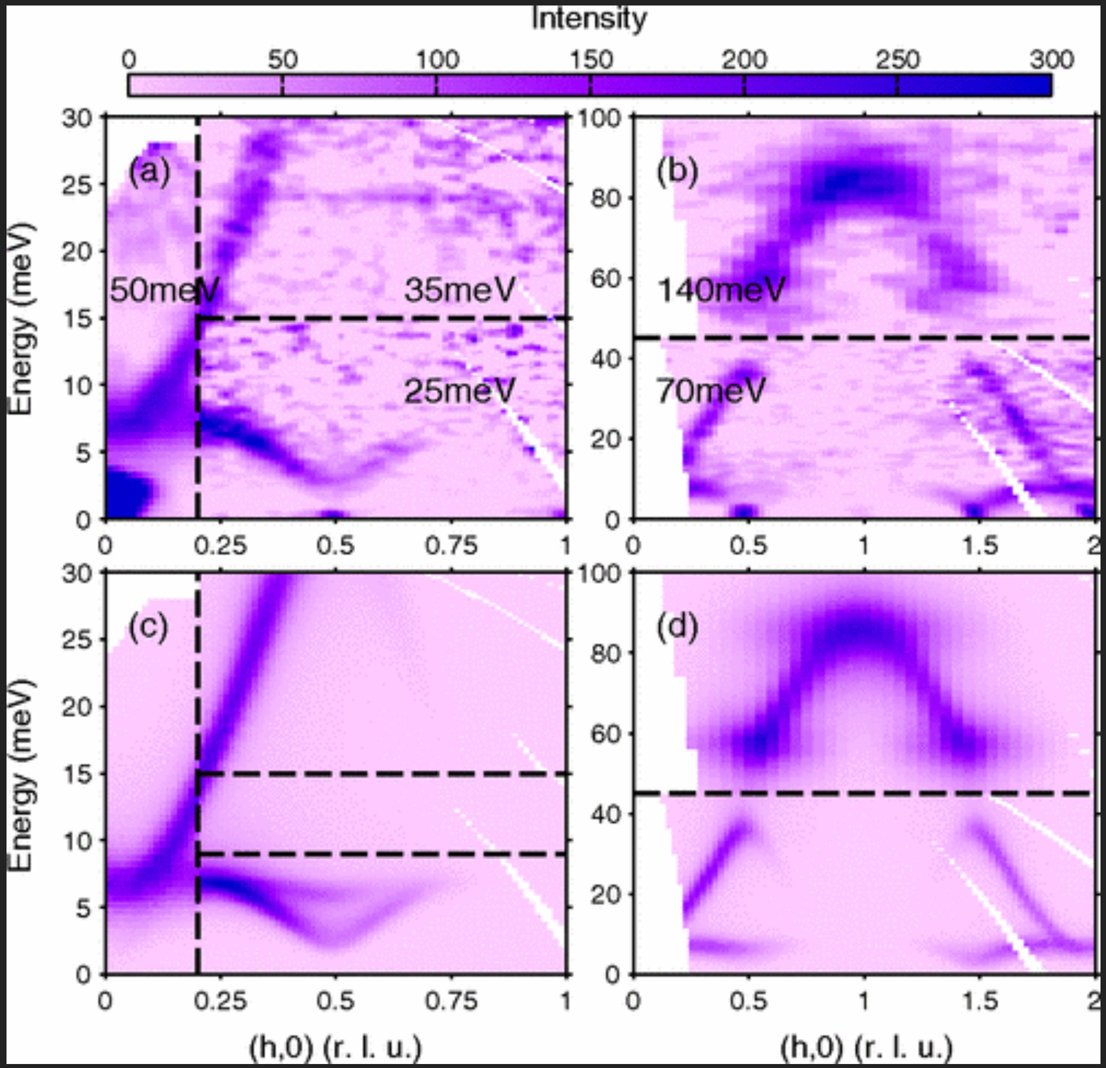
Phys. Rev. Lett. **109**, 237202 – Published 3 December 2012





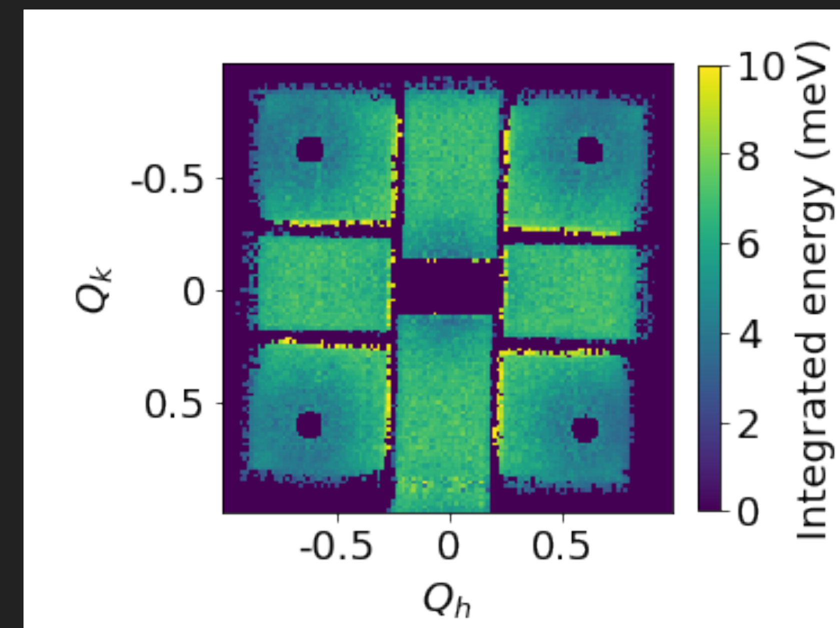
PCSMO THE DATA

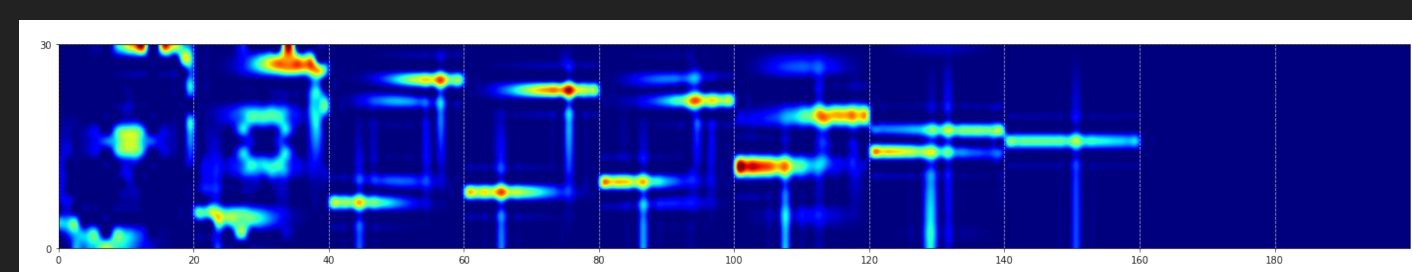
- Significantly messier dataset
- Noisy experimental data
- Multiple bands
- Presence of phonons



PCSMO THE DATA PART II: MULTI-BANDS AND HOW TO DEAL WITH THEM

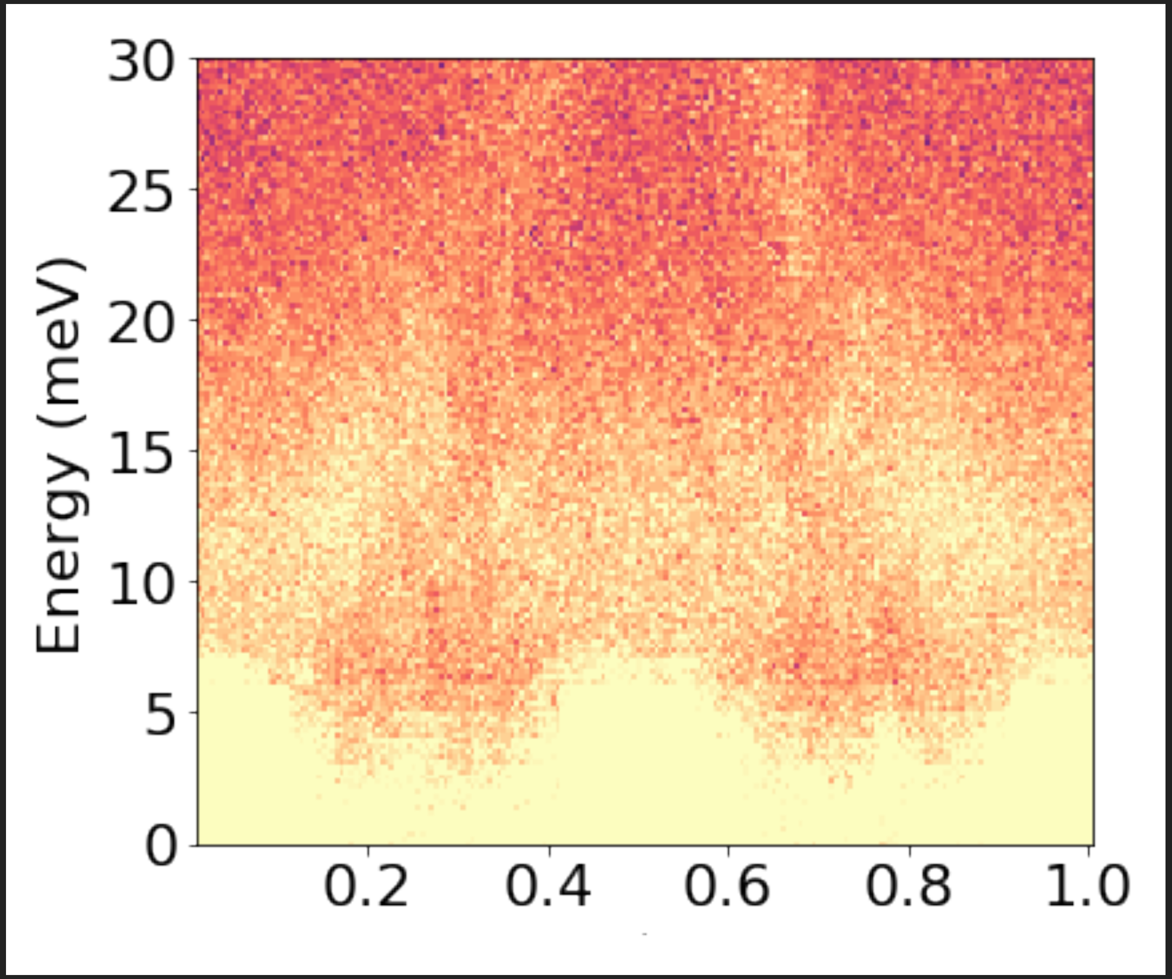
- In Rb2MnF4 we could integrate across the energy spectrum
- In PCSMO this would lead to loss of information
- Develop an image with interactions across energy slices



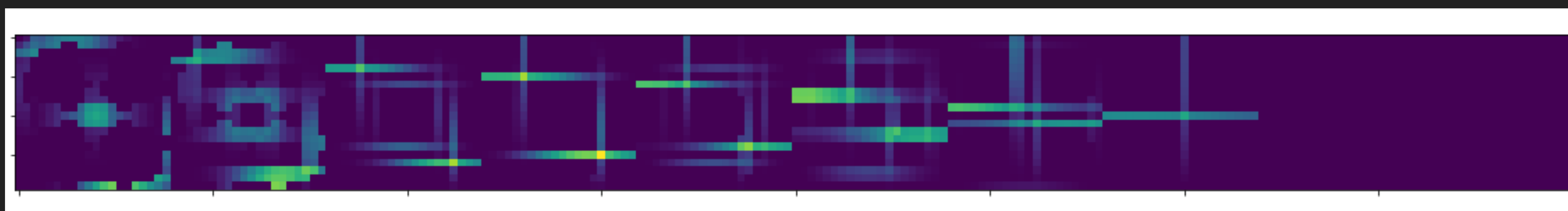


PCSMO THE DATA PART III : “NOISE”

- There is a large contribution from the phonon spectrum
- This can obfuscate the magnon spectrum
- Would like to remove this if possible



PCSMO RESULTS: PHASE DISCRIMINATION (SIMULATED DATA)

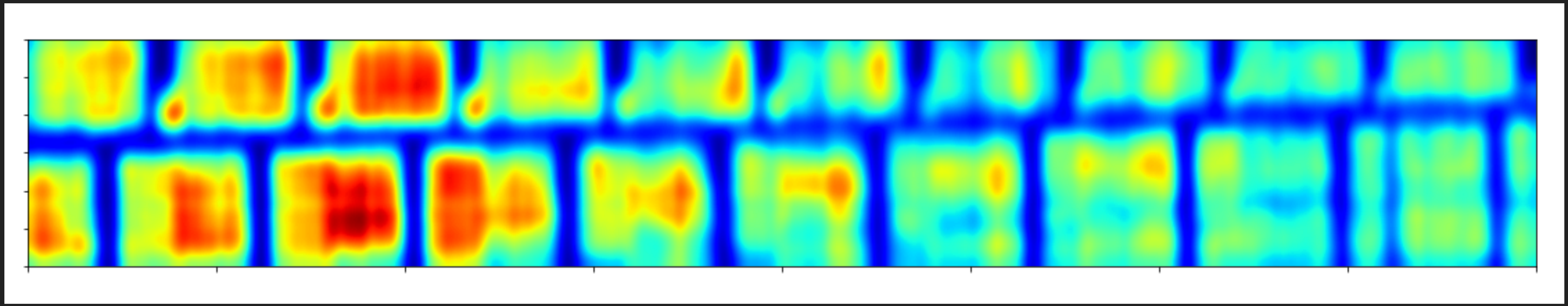


```
[conv_outputs, predictions] = get_output([test_i
conv_outputs = conv_outputs[0, :, :, :]
maxval = np.argmax(np.array(predictions))
```

```
Prediction Goodenough
```

PCMSO RESULTS II: EXPERIMENTAL DATA

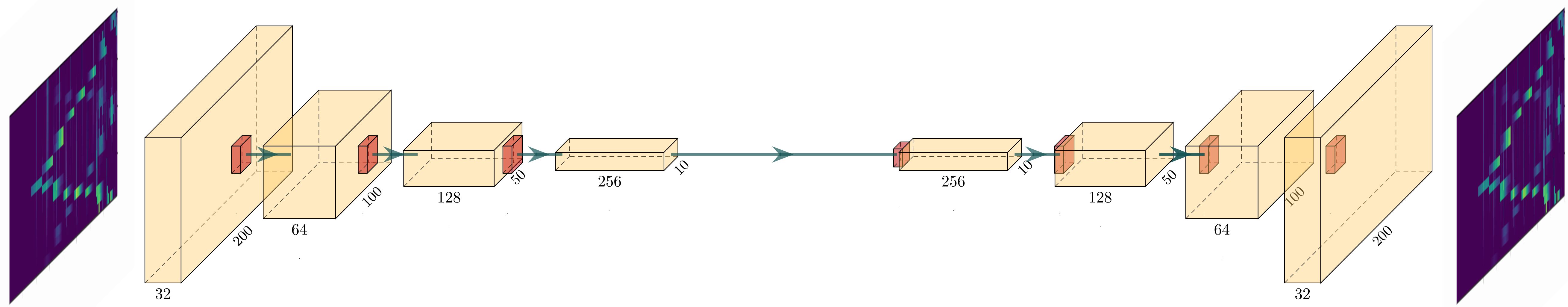
- Failure - noise :(



```
ylist = np.linspace(0, dim[0]*2, c
X, Y = np.meshgrid(xlist, ylist)
## Add Gaussian smoothening to con
sigma = 0.2 # this depends on how
camg = gaussian_filter(cam, sigma)
```

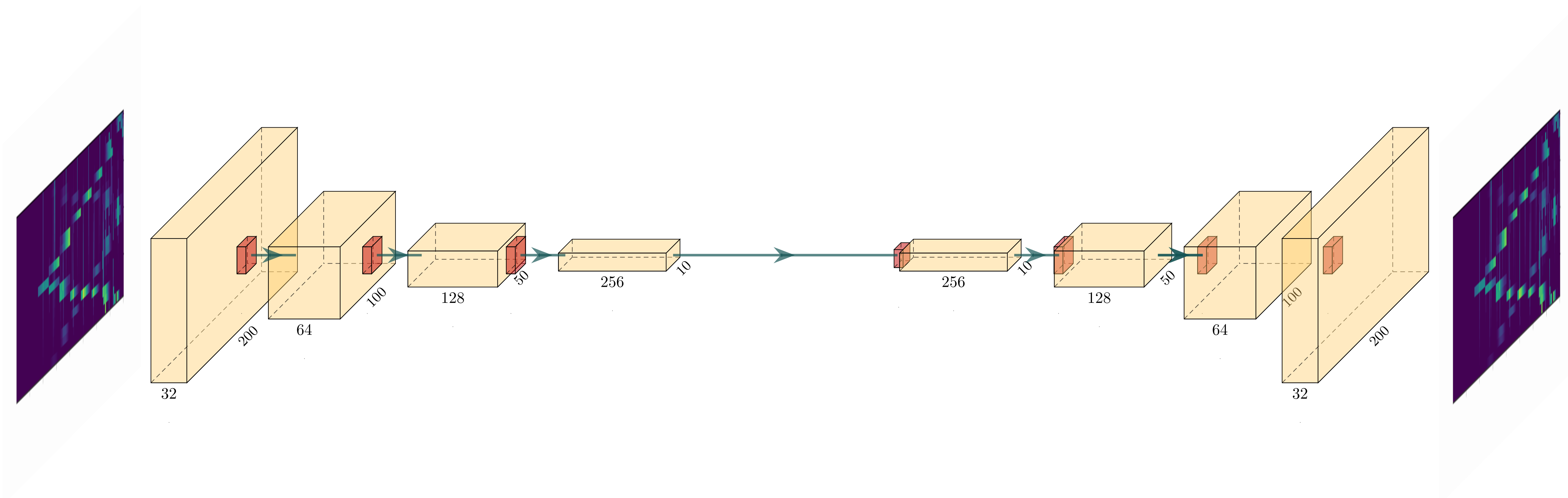
```
Prediction Dimer
[[8.2898813e-01 4.3244651e-08]]
```

PCSMO: REMOVING THE NOISE (AUTOENCODERS)



- Can we remove the experimental 'noise'
- Noise = instrument noise + other signals
- We can try to use a denoising auto encoder

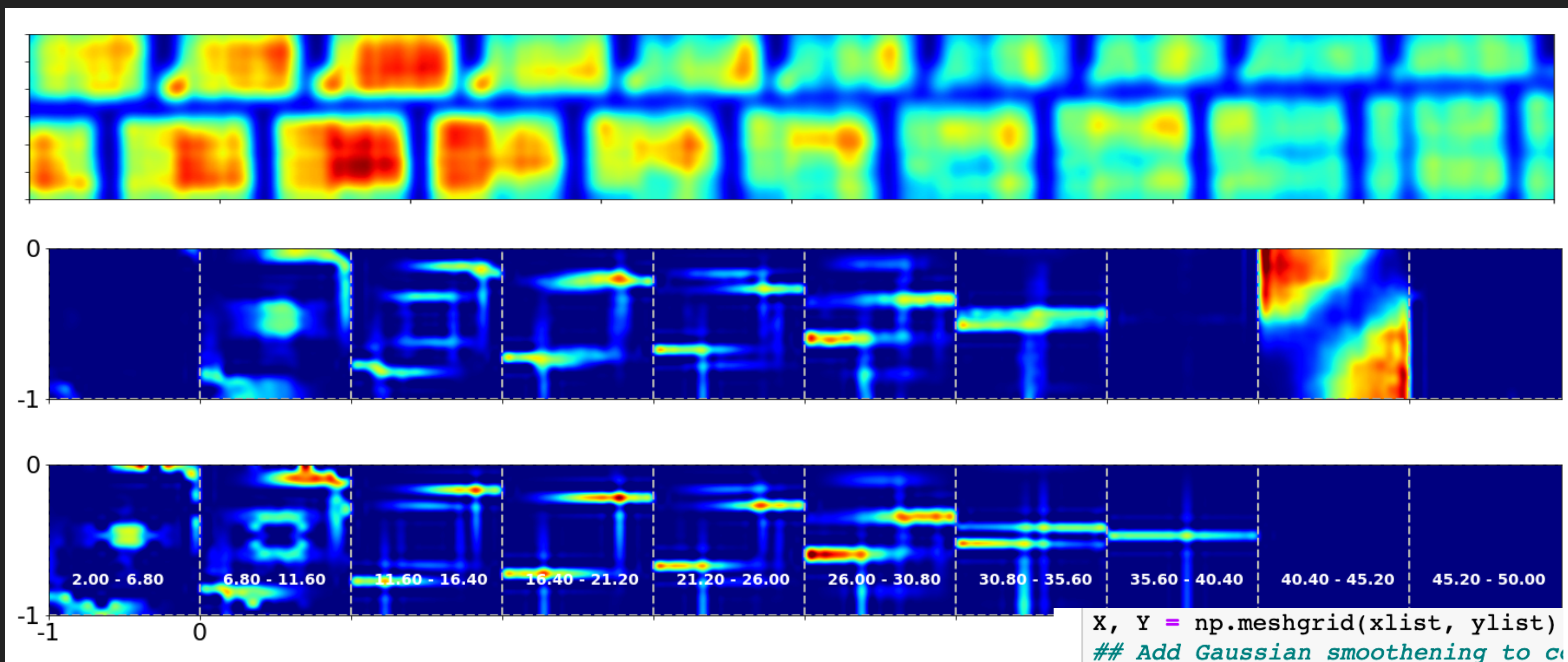
PCSMO: REMOVING THE NOISE (AUTOENCODERS)



- An unsupervised machine learning approach
- Learn compressed representations
- Can be similar to classical methods like PCA

PCSMO RESULTS III: AUTOENCODER + DISCRIMINATION

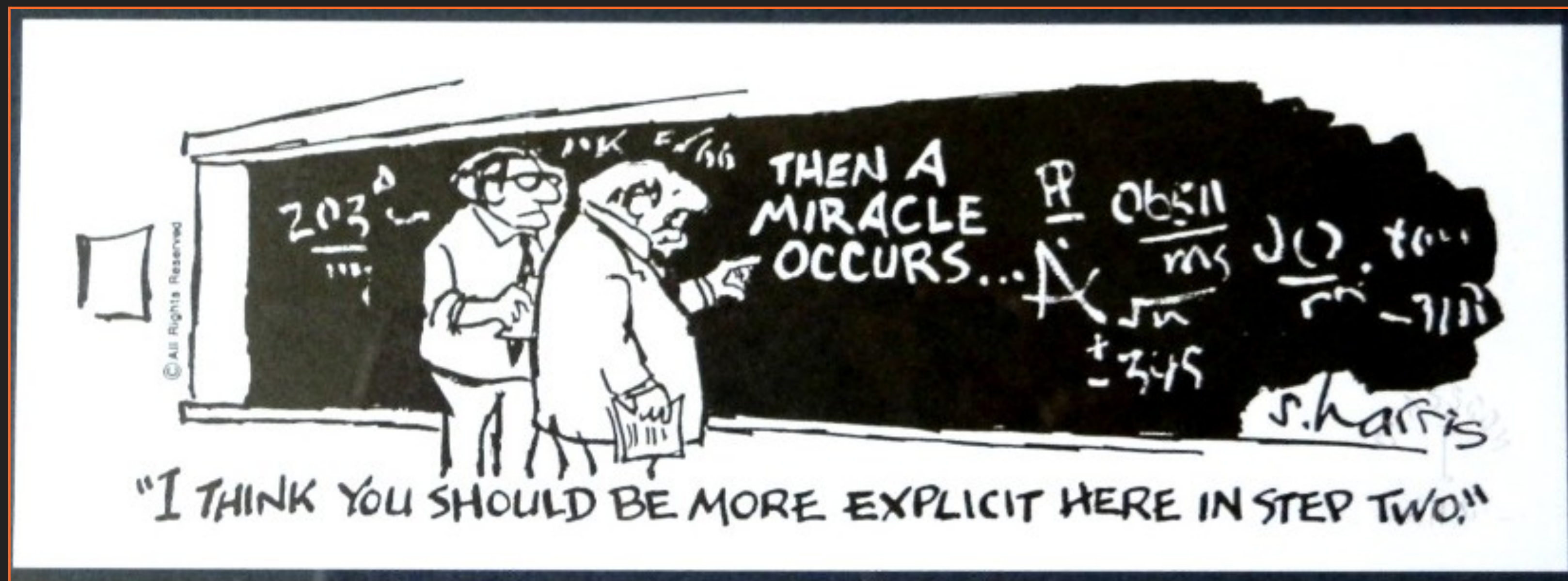
- (Qualified) Success :)




```
X, Y = np.meshgrid(xlist, ylist)
## Add Gaussian smoothening to co
sigma = 0.2 # this depends on how
camg = gaussian_filter(cam, sigma
```

```
Prediction Goodenough
[[0.00203549 0.90476966]]
```

MAKING MODELS INTERPRETABLE



"I THINK YOU SHOULD BE MORE EXPLICIT HERE IN STEP TWO."

- Classical models are often easy to interpret
- Deep models, learned representations can be more opaque

MAKING MODELS INTERPRETABLE

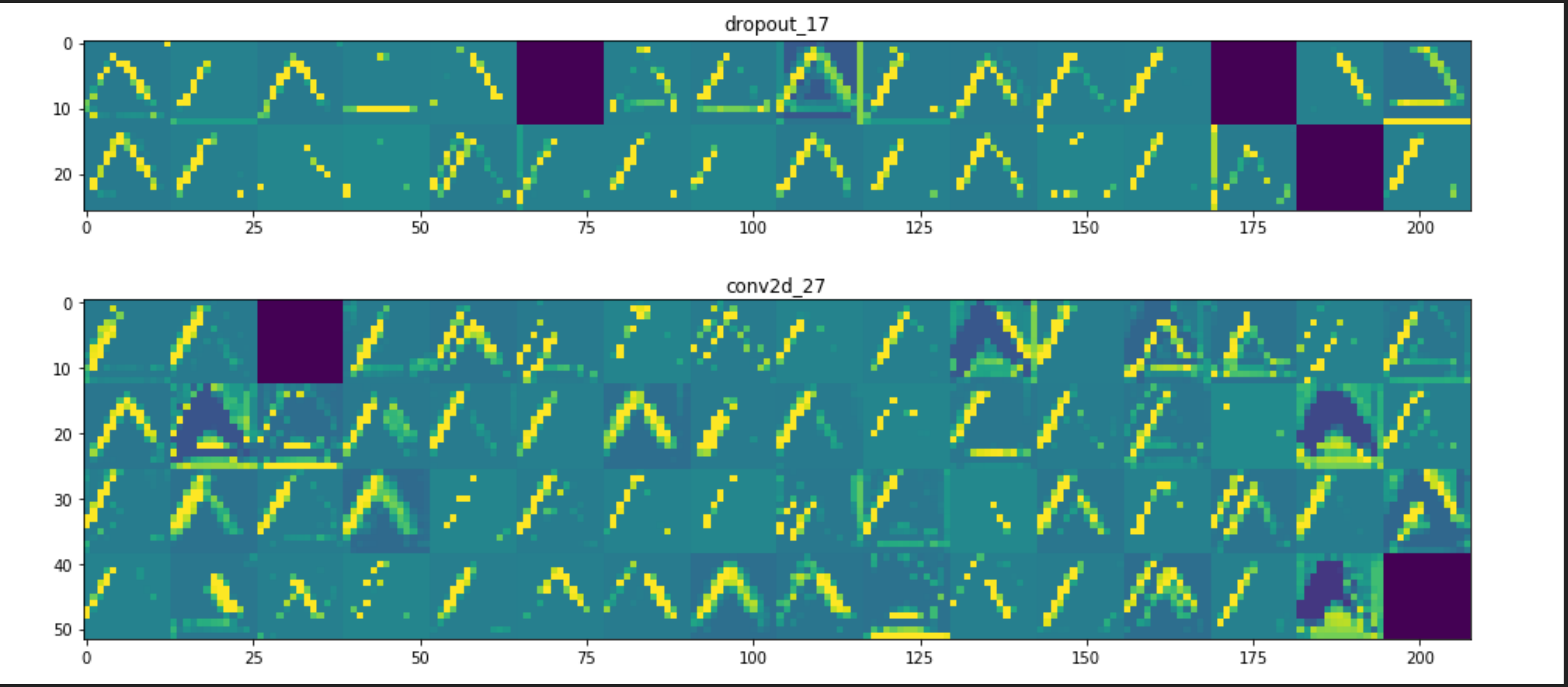
Model performance	Interpretability use
Sub-human	Debug and improve
Human	Increase confidence
Super-human	Learn from successs

INTERPRETING CNNS

- Visualising the activations
- Visualise layer weights
- Retrieving images that maximally activate a neuron
- Filter maximisation
- Embedding the codes with t-SNE
- Class activation maps

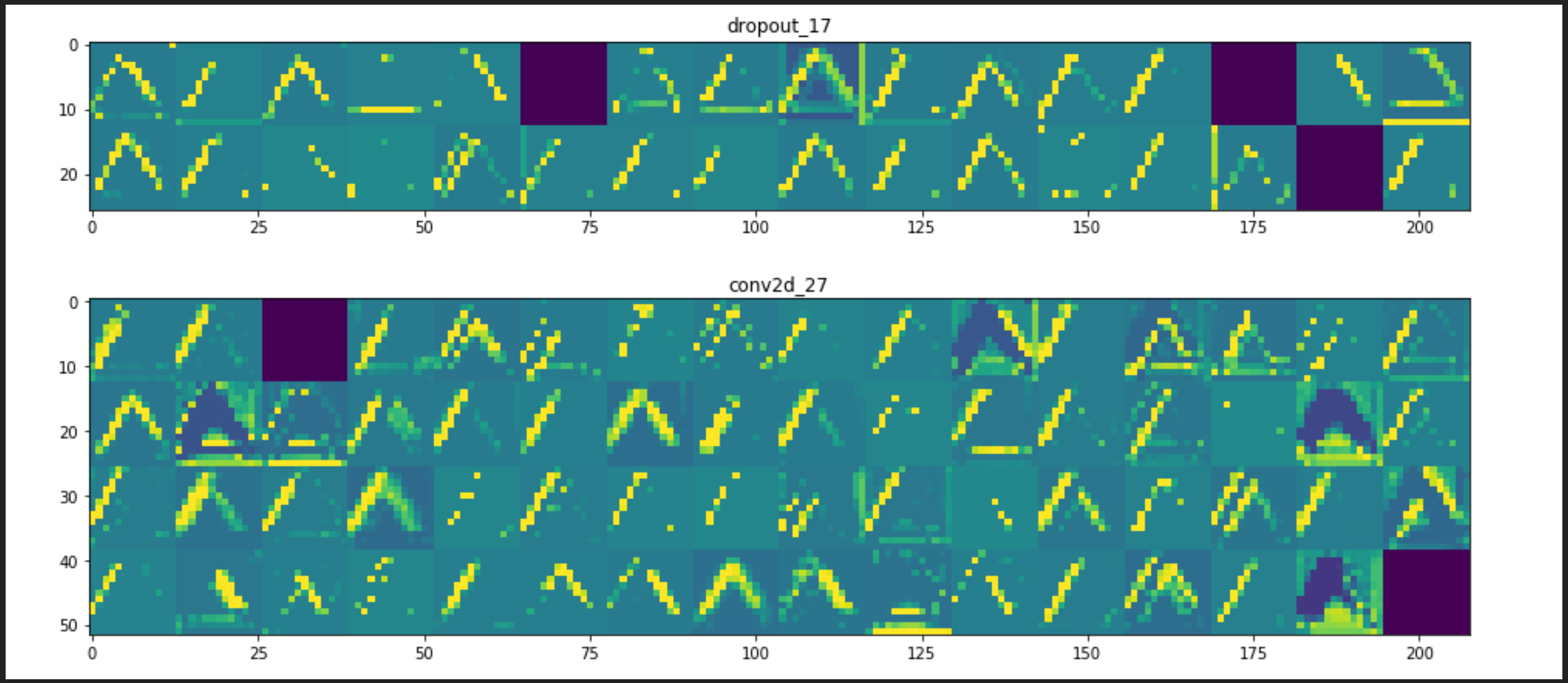
VISUALISE ACTIVATIONS

```
activations = activation_model.predict(img_tensor)
first_layer_activation = activations[0]
plt.matshow(first_layer_activation[0, :, :, 4], cmap='viridis')
```



https://towardsdatascience.com/visualizing-intermediate-activation-in-convolutional-neural-networks-with-keras-260b36d60d0

VISUALISE ACTIVATIONS



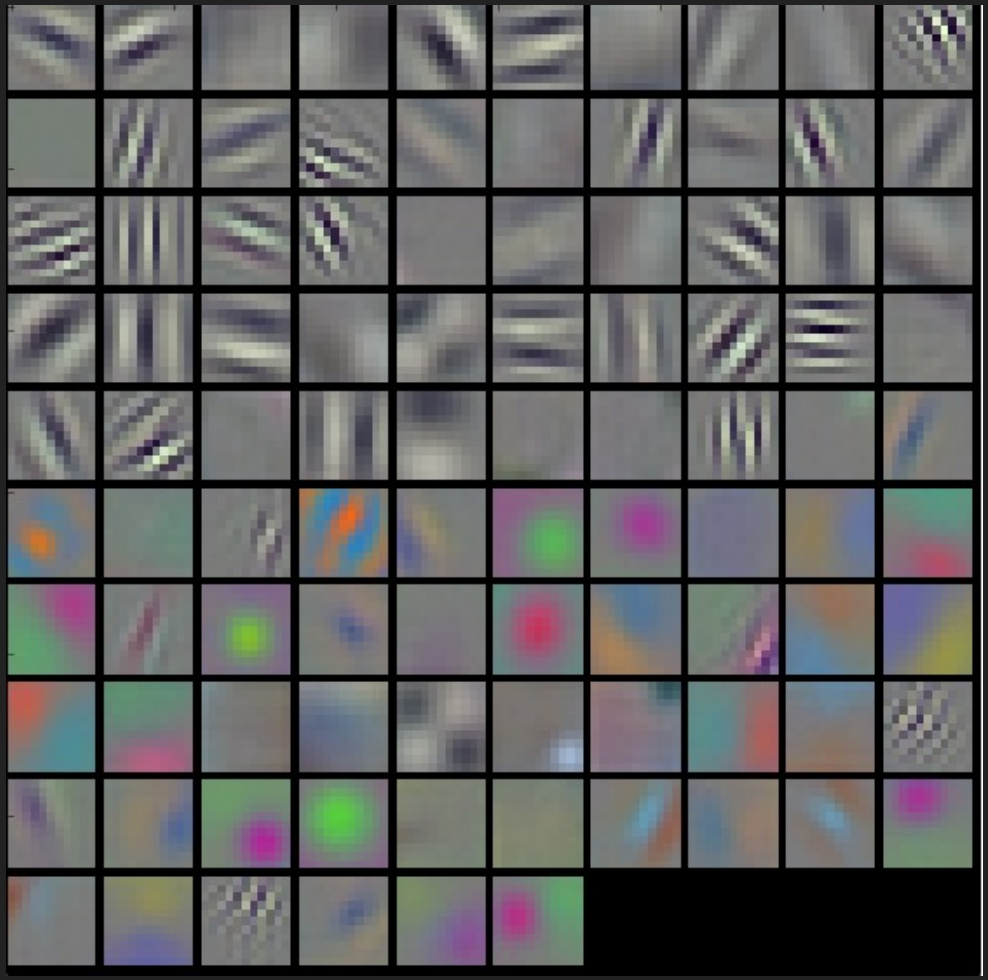
Filters start out dense and blobby - sharpen up during training

Very useful for identifying dead filters

https://towardsdatascience.com/visualizing-intermediate-activation-in-convolutional-neural-networks-with-keras-260b36d60d0

VISUALISE WEIGHTS

```
for i in range(nlayers):
    filters, biases = model.layers[n].get_weights()
```

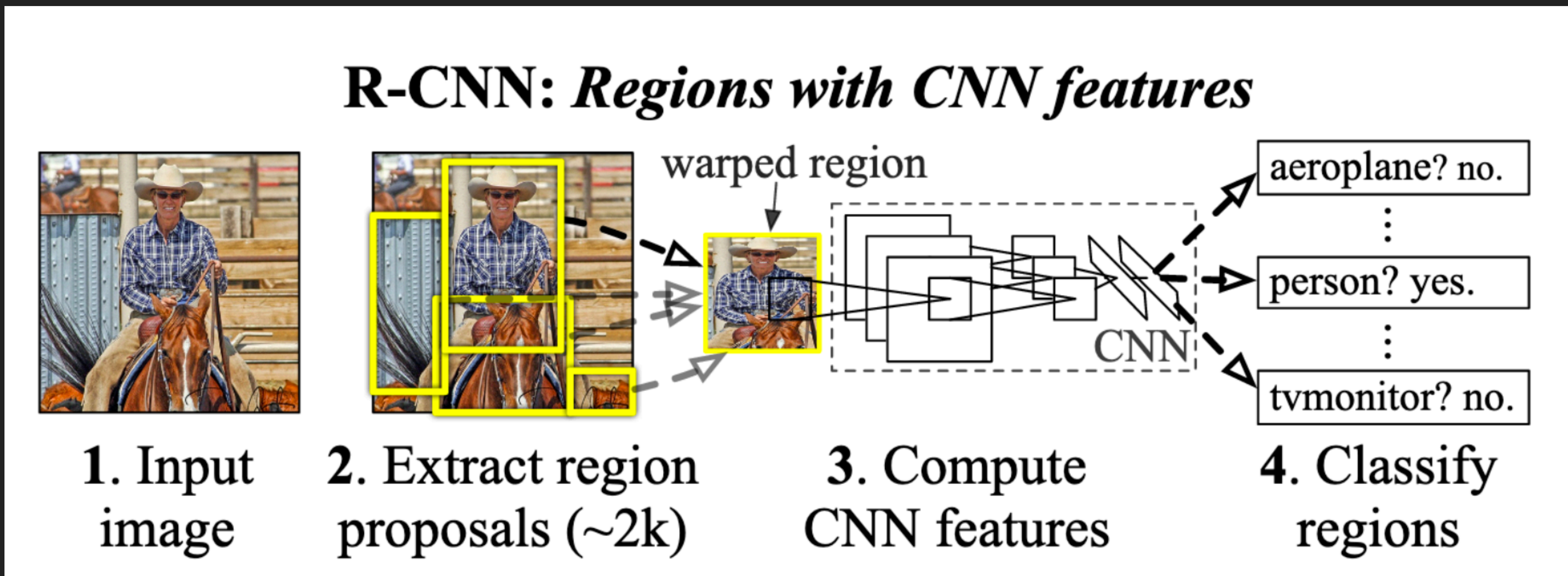


Most useful for early layers

Working network - smooth weights features

Noisy filters - network not trained long enough or overfit

MAXIMAL ACTIVATION



- Can look at class activations or filter activations
- Neurons do not have semantic meaning by themselves

arXiv:1311.2524

FILTER MAXIMISATION

- Develop a loss function find the image that maximises a given filter response

```
# nth filter of the layer considered
layer_output = layer_dict[layer_name].output
loss = K.mean(layer_output[:, :, :, filter_index])

# compute the gradient of the input picture wrt this loss
grads = K.gradients(loss, input_img)[0]

# normalization trick: we normalize the gradient
grads /= (K.sqrt(K.mean(K.square(grads))) + 1e-5)

# this function returns the loss and grads given the input picture
iterate = K.function([input_img], [loss, grads])
```

https://blog.keras.io/how-convolutional-neural-networks-see-the-world.html

FILTER MAXIMISATION

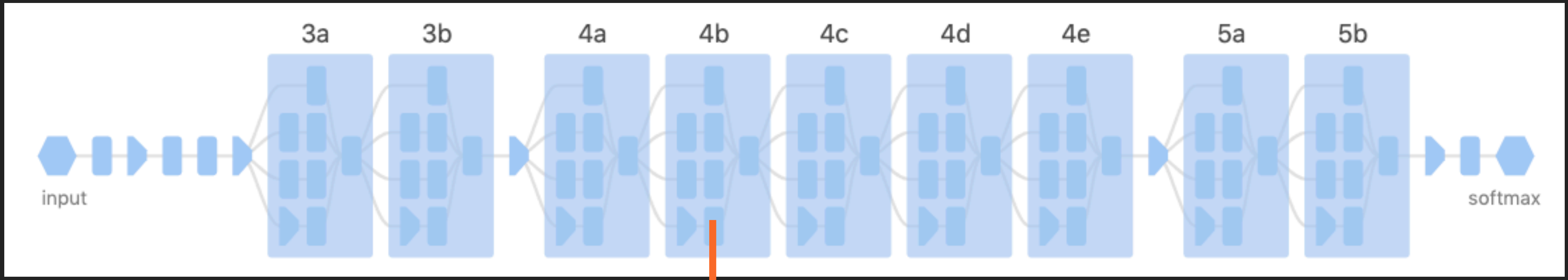
- Convert random noise to a maximising image

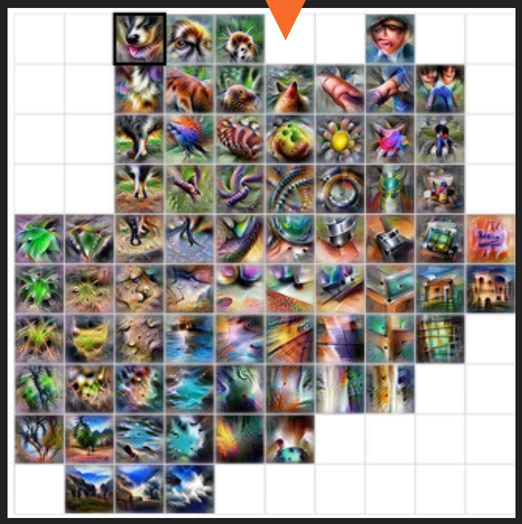
```
# we start from a gray image with some noise
input_img_data = np.random.random((1, 3, img_width, img_height)) * 20
+ 128.
# run gradient ascent for 20 steps
for i in range(20):
    loss_value, grads_value = iterate([input_img_data])
    input_img_data += grads_value * step
```

SHOW NOTEBOOK EXAMPLE

ACTIVATION ATLAS

- Take an image - pass through a network
- Map the vectors of different layers using dimensionality reduction of the vectors
- Distance on the map grid corresponds to the the similarity of the vectors





https://distill.pub/2019/activation-atlas/

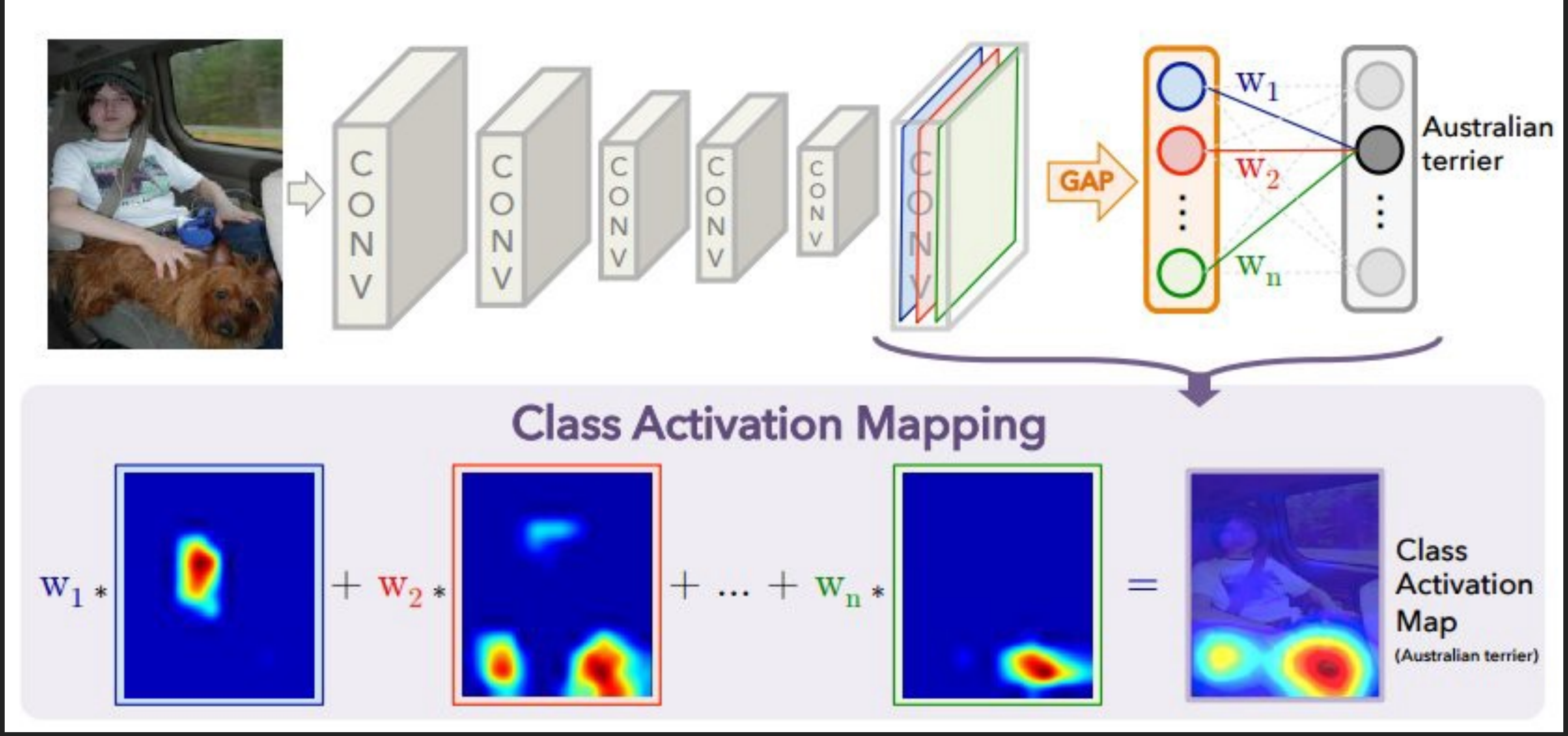
Science and Technology Facilities Council

CLASS ACTIVATION MAPS

- Show which regions of an input are responsible for classification

arXiv:1512.04150v1

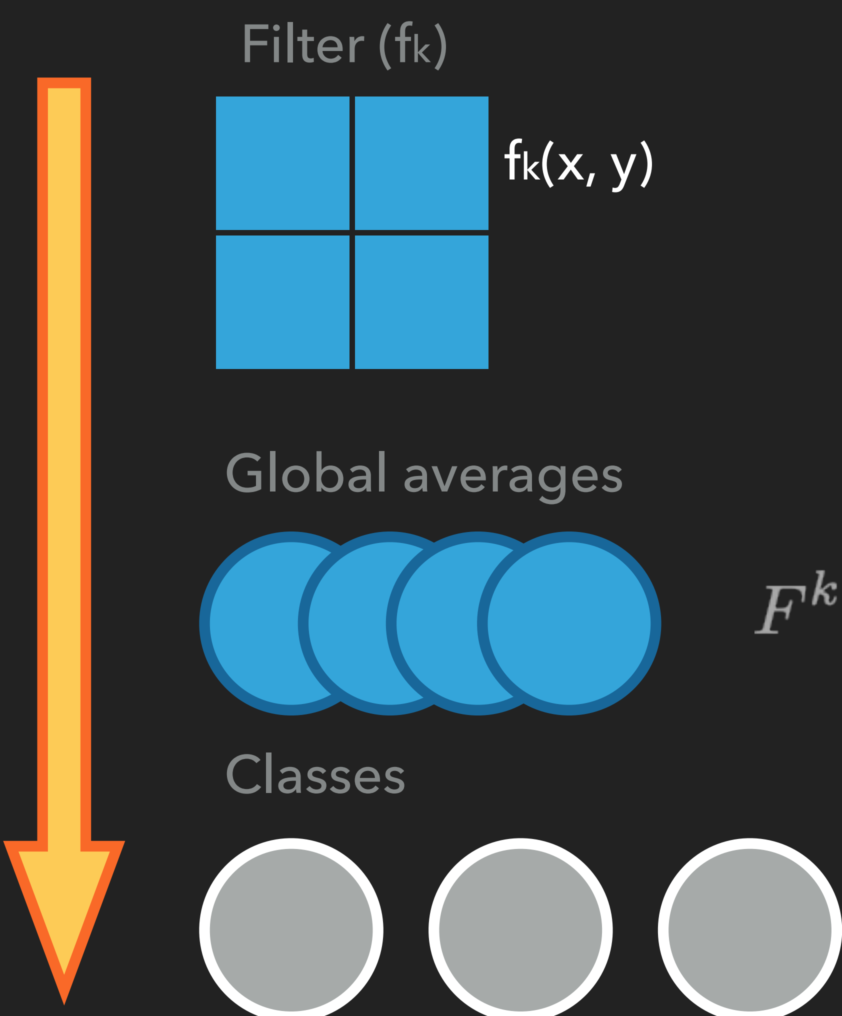
CLASS ACTIVATION MAPS NETWORK ARCHITECTURE



- Global average pooling
- Apply to final convolutional layer

arXiv:1512.04150v1

CAM HOW IT WORKS



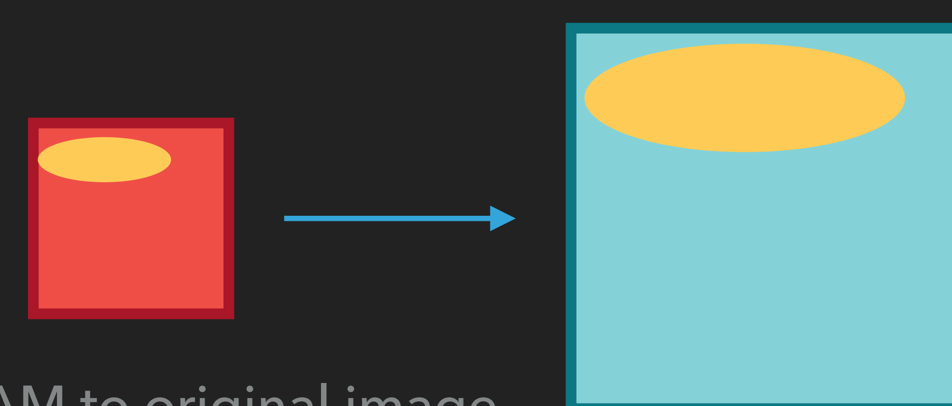
$$F^k = \sum_{x,y} f_k(x, y)$$

$$S_c = \sum_k w_k^c F^k$$

$$S_c = \sum_k \sum_{x,y} w_k^c f_k(x, y)$$

$$M_c(x, y) = \sum_k w_k^c f_k(x, y)$$

Directly indicates the importance of a location (x, y) to the class activation S_c.



Upsample CAM to original image size

GRAD-CAM GENERALISING CAM

$$\alpha_k^c = \frac{1}{Z} \sum_{x,y} \frac{\partial S_c}{\partial f_k(x,y)}$$

The importance of feature map k for classification c

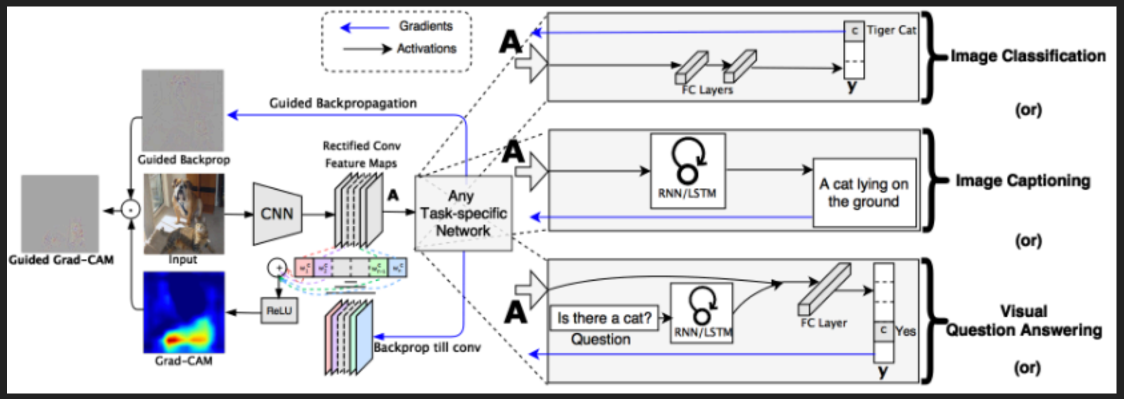
$$M_c(x,y) = ReLU\left(\sum_k \alpha_k^c f_k(x,y)\right)$$

Summation of feature map time importance

ReLU ensures that only positive contributions are measured

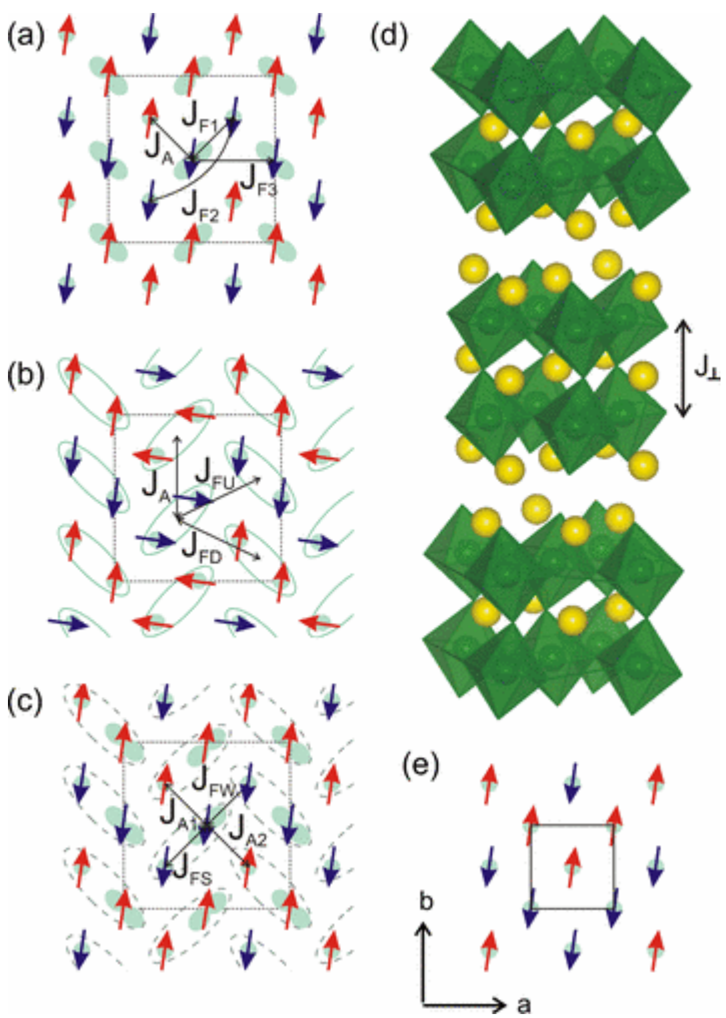
Grad-CAM does not require network reconfiguration

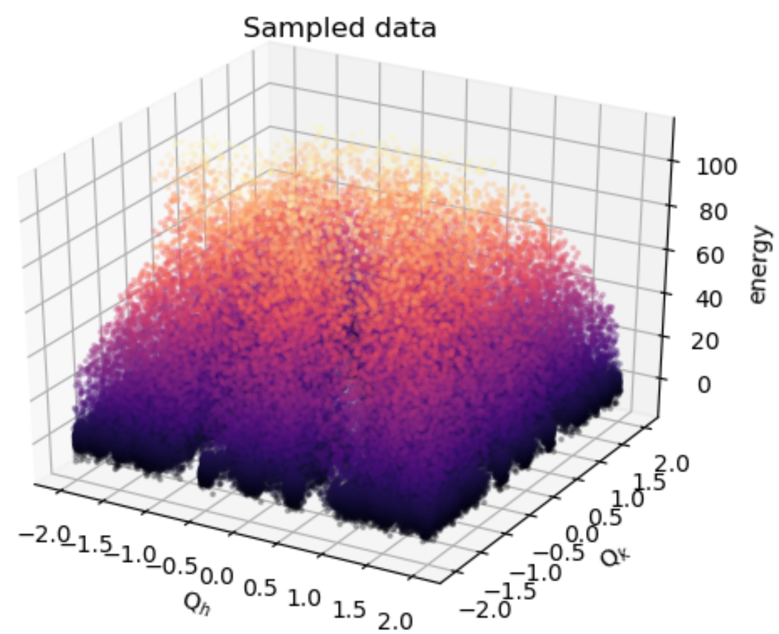
Grad-CAM can work with any network type, not just CNNs



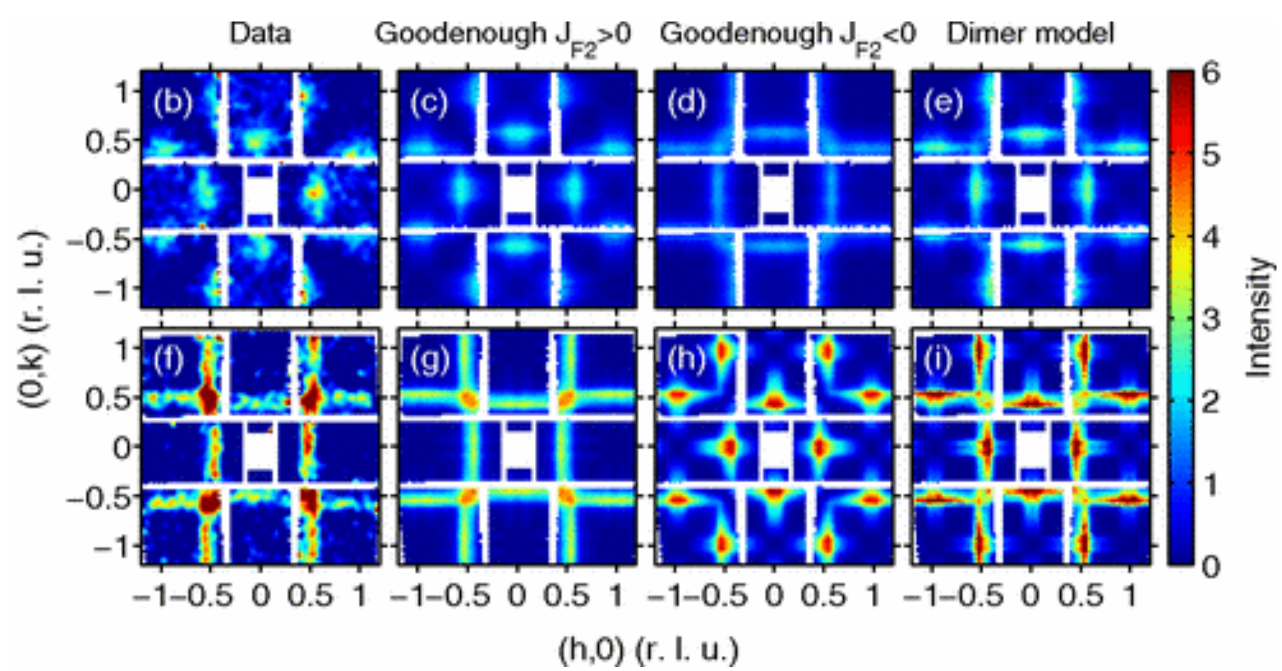
arXiv:1610.02391v3

INTERPRETABLE MODELS FOR NEUTRON SCATTERING

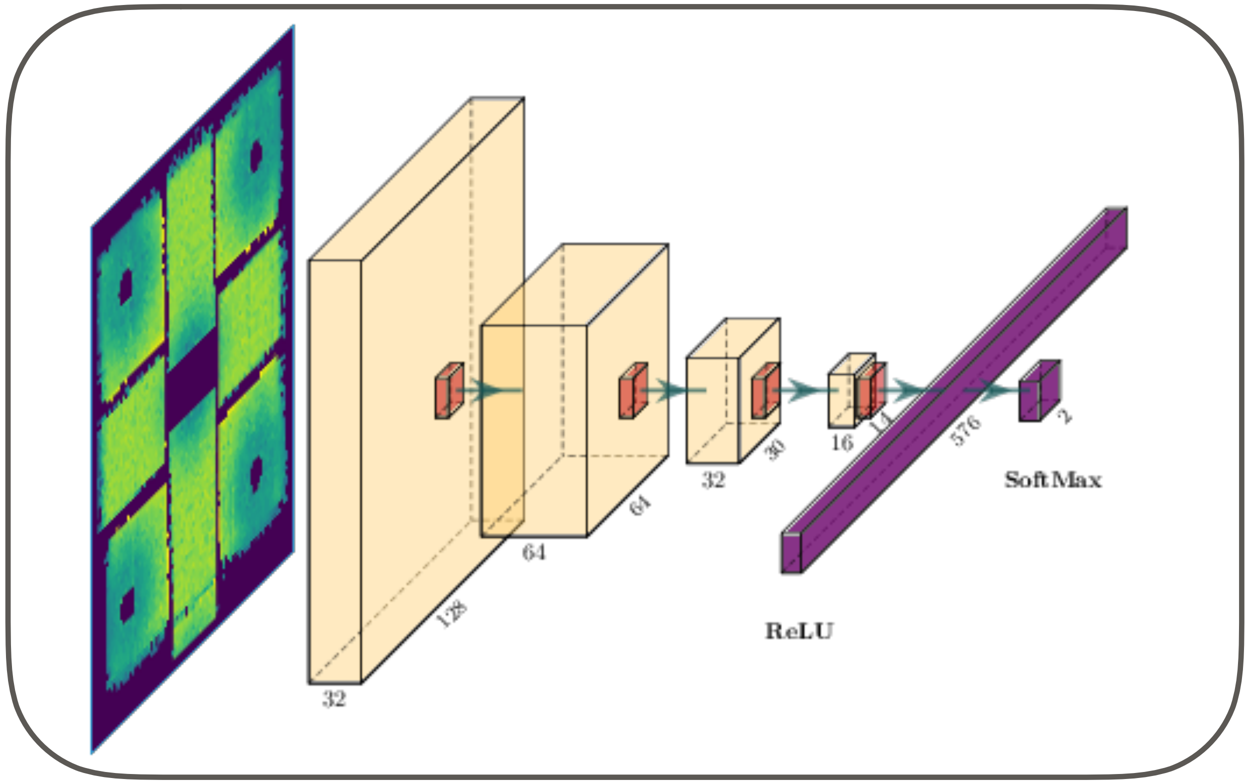




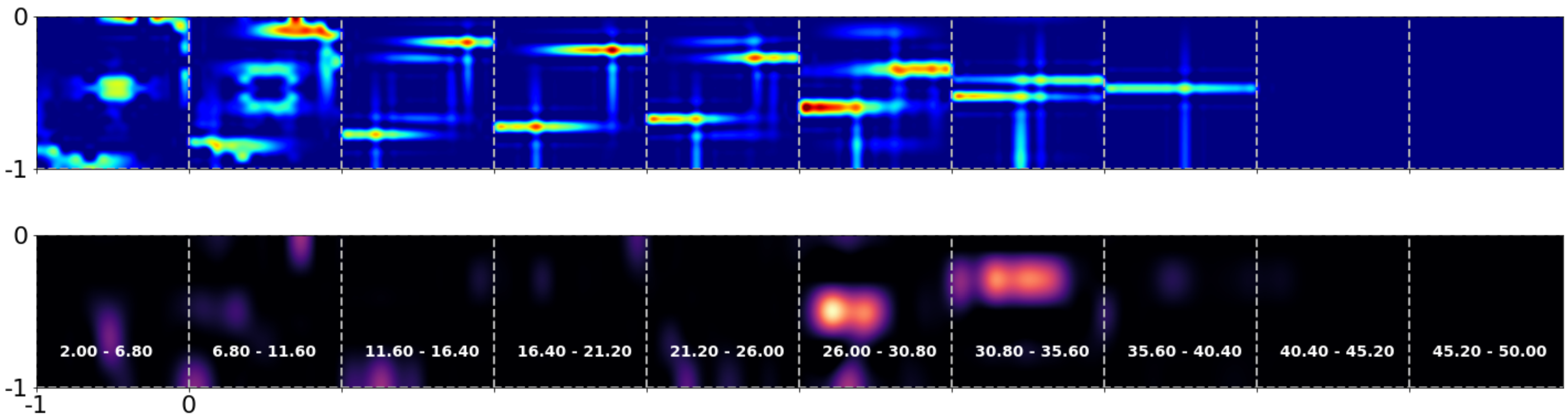
Finding the right signal can be a needle in a haystack



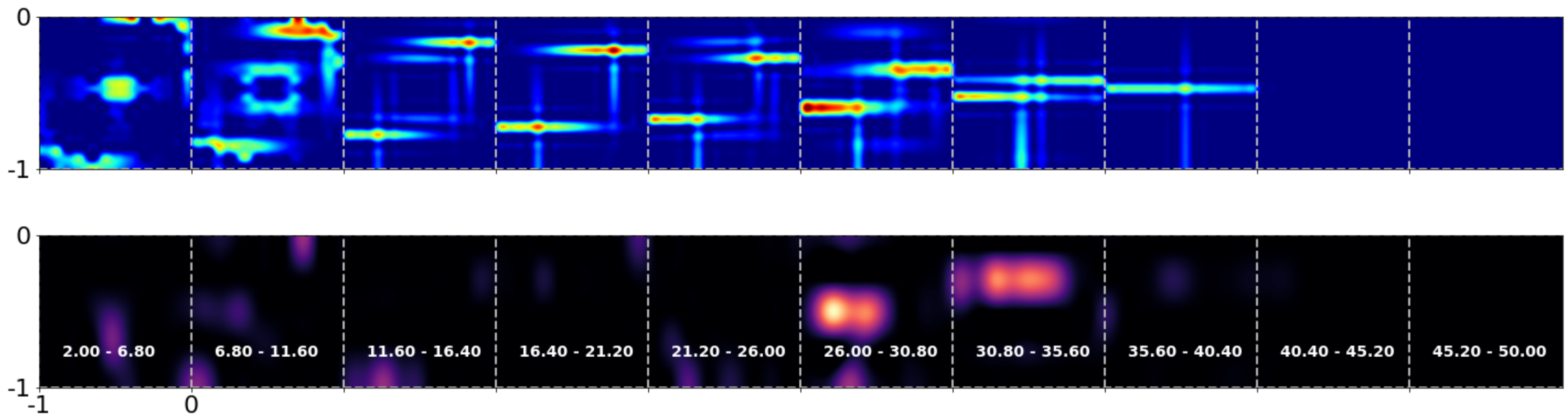
INTERPRETABLE MODELS FOR NEUTRON SCATTERING

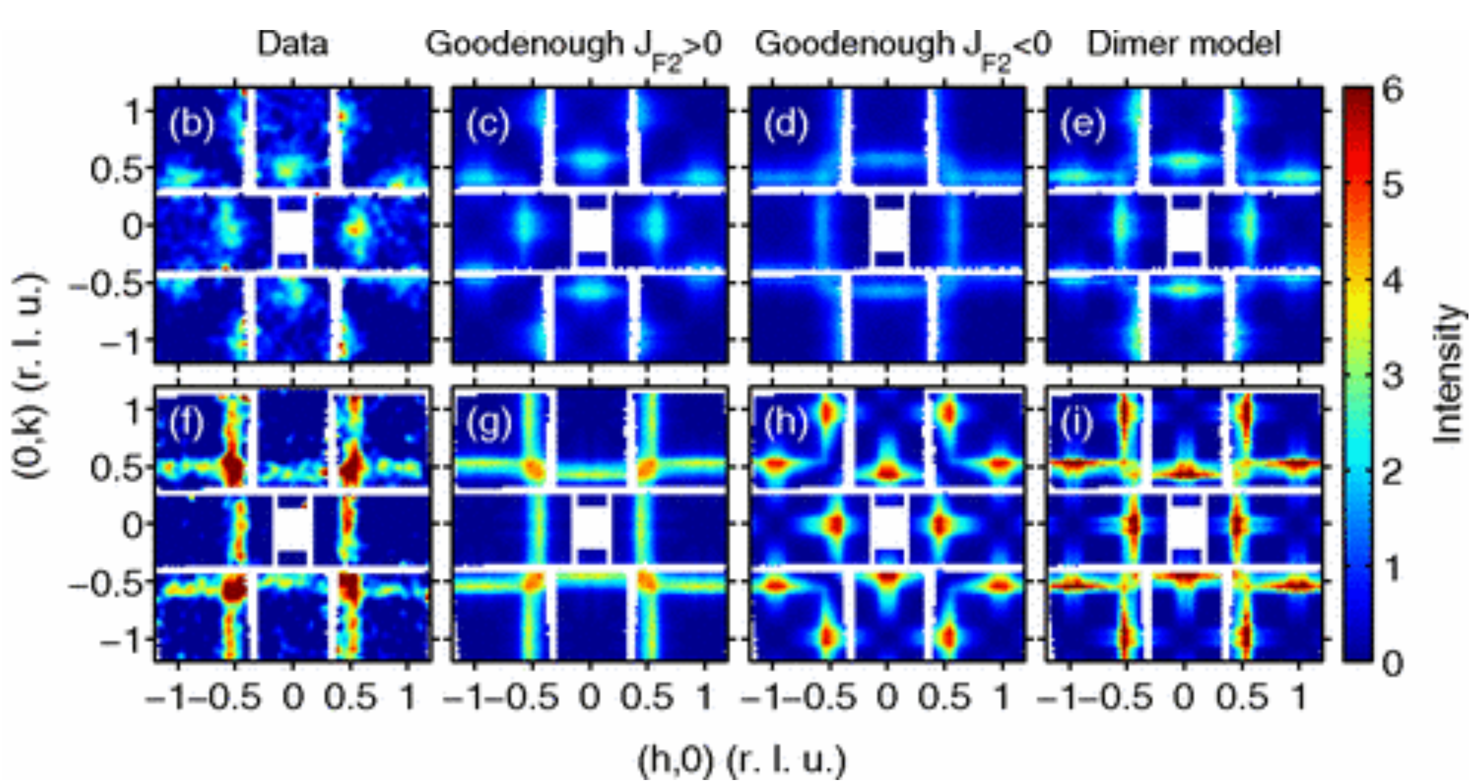


Build a model discrimination network - ask it WHY it makes the choice.



INTERPRETABLE MODELS FOR NEUTRON SCATTERING





The network identifies the same regions of E/Q space as a trained physicist.

Could, in future, guide experiments of the same type.

SUMMARY

- Inelastic neutron scattering requires complex data analysis to extract useful information
- Combining physics simulations with deep neural networks can help in interpreting experimental spectra
- Understanding how neural networks arrive at answers is generally a good idea!
- Understanding network results can provide guidance on how to sample experimental space

ACKNOWLEDGMENTS

- SciML
  - Rebecca, Tony, Jeyan, Sam, Patrick
- ISIS
  - Duc, Toby

@keeeto2000
@ml_sci

keeeto.github.io

www.scd.stfc.ac.uk/Pages/Scientific-Machine-Learning.aspx