

## Questions and answers - Ben Nachman Lecture

The following questions were submitted through Google Form. Some may have been answered in the Q&A session already. Nevertheless, we request our lecturers to provide written answers here for the benefit of those who could not attend that session. Thank you!

Slide 66. Can you please explain a bit more on the Score-based generative models?

*I think these will be covered in more detail next week during the dedicated lectures on generative models, but briefly: instead of learning  $\log p(x)$  as in normalizing flows, score-based models learn  $\nabla \log p(x)$  (called the score function). The challenge with learning the density directly is that the function needs to be normalized to one, which imposes many constraints on the architecture. This restriction is absent for score-based models.*

Slide 60. About simulation purposes. How would you handle nonlinearities using a neural network to replace a slow, physics-based simulation?

Putting into context, and moving away from the calorimeter example you showed, what if I consider a light propagation simulation (in the context of a water-Cherenkov detector, for example), how can I handle the light scattering using a neural network?

*Neural networks are highly expressive and can readily model non-linearities. Water-Cherenkov should also work - the name of the game is starting with a representation of the data and then picking an architecture that can accommodate it.*

Slide 37. I understand that for unsupervised learning, we have a few algorithm which we use such as clustering algorithms or Dimensionality Reduction algorithms. My question is, what type of algorithm do we use for weakly-supervised or semi-supervised learning?

*Weak- and semi-supervised learning algorithms are usually trying to learn some kind of likelihood ratio, just like supervised learning, and can use basically all of the same architectures with suitably modified learning approaches. That said, sometimes there can also be a mix of unsupervised approaches and supervised approaches (e.g. combining autoencoders with supervised classifiers).*

