ML in Fundamental Physics

Ludwig-Maximilians-Universität München – Dr. Sven Krippendorf

Exercise Sheet 8

Submission by Wednesday 8th 17:00 via Moodle.
Corrections due by Saturday 11th 16:00 via Moodle.
Format for all questions: individual Jupyter notebooks .ipynb.

1 Variational Autoencoder (3+2 points)

The aim of this exercise is to implement the variational autoencoder and to test it on the polynomial dataset from the previous exercise sheet.

- (1 point) Implement the loss function of the variational autoencoder in your previous architecture.
- (2 point) Compare the performance and the latent space representations of the traditional autoencoder and your variational autoencoder.
- * (2 points) There is a quicker way of implementing the variational autoencoder using the tensorflow probability package. Using this package implement your variational autoencoder. As indicated, this exercise is optional.

2 GAN optimisation (2 points)

For the example $V(x, y) = xy$ perform alternating gradient descent updates with respect to x and y as we would perform when training a GAN, i.e. one player is performing updates to optimise V and one player to optimise $-V$. Identify the trajectory of this update. Which 'late-time' behaviour do you find?

3 GAN (3 + 2 points)

The aim of this exercise is to implement a GAN architecture and to test it on the polynomial dataset from the previous exercise sheets. You can adapt the architecture from the MNIST example in the lectures.

$\star$ Update your loss function to implement a Wasserstein GAN.

Note that in case you do not have a local environment where you can calculate on a GPU, you can calculate online on `collab` on a GPU.