[*Mnih et al.*, 2015] report using the Deep Q-Network to let the computer to learn how to play the Atari 2600 games. This paper has the detailed information about the structure, parameter and the way they train the algorithm. It has two key ideas, (1) use the experience replay, and iterative update that adjusts the action-values towards target values. It uses the discounted award function, which discount the future actions for the rewards. The learning essentially is a regression problem to estimate the Q-value from the observed images. The code is written all in Lua, but there is a similar project using it to train the flappyBird, which you can find it here: <https://yanpanlau.github.io/2016/07/10/FlappyBird-Keras.html>.

[*Araya-Polo et al.*, 2018] talks about their way to do tomography velocity model using deep neural network. They are still only using the synthetic data by first randomly generating velocity model and then generate the seismic waveforms, and use these as the pair to train the network. After that, they will generate more velocity model and test the model against these newly generated models. Also note that, their output needs to be processed in order to have the same demission as the velocity models.

[*Shi and Dustdar*, 2016] discusses the edge computing. An edge device is any computing or networking resource residing between data sources and cloud-based datacenters. It has many advantages over the cloud computing, i.e. reduces the communication significantly, computing on the edge device will be much smaller than the cloud counterpart, improve the battery usages at the end devices, and solves partially the privacy issue, etc. It has a lot of different applications, and this paper shows two different applications: online shopping and finding missing children. It can also apply to the disaster response, which will be really useful.