Renormalization is a tool commonly employed in the field of physics. However, recent papers have attempted to explore the connection between renormalization and deep learning. In this assignment, I will walk through the paper of Mehta and Schwab (2014), which claims that we could achieve an one-on-one mapping between variational renormalization group and deep learning. The goal is to compare my own implementation of the techniques mentioned in the paper (i.e., stacked restricted Boltzmann machines) with an existing implementation that follows the authors' instructions more closely (i.e., hand-coded RBM function) and compare the results. The process would shed light on the various conditions for the RBM to work as intended, and therefore lead to discussions on the strengths and limitations of the paper.

The write-up contains a summary of the paper and arguments against its generalizability. The quantitative component consists of a replication of the deep learning algorithm in the paper.

Paper summary:

The paper by Mehta and Schwab shows an one-on-one mapping between renormalization group and deep learning algorithm (Restricted Boltzmann Machine in this case), thus proving that deep learning inherently follows a process of renormalization, therefore attaining its impressive learning power frequently touted among the academia and public alike. Since deep learning algorithms have a tendency to discover hidden but useful patterns and strategies normally overlooked by humans, the claim made by authors is extrapolated by Wolchover (2014) to imply a "door to something exciting". In other words, renormalization might be the hidden strategy we humans could learn to apply to a wider range of problems, not only pertaining to physics, to come closer to hidden rules of nature.

Quantitative:

In this section, I replicate the stacked RBM model described in Mehta-Schwab, with identical parameter values where applicable. My own implementation is then compared with an existing implementation that follows exactly the authors' specifications, including modifications to the RBM. The result shows that an off the shelf usage of sklearn's RBM does not work in learning to renormalize the Ising model. Instead, the model failed to demonstrate the block renormalization behavior (it still reconstructs the original data pretty well in a rather renormalized manner, nevertheless -- I only treat the model as failed since it does not display block renormalization which is the center of the Mehta-Schwab paper). I found out that the author has made changes to the update function to use L1 regularization instead of weight decay (see Mehta & Schwab Appendix A). This substitution's impact is two-folded. One, it prevents overfitting. Two, and more importantly, as explained by the authors, it prevents all-to-all couplings between layers of the DNN. This is suspected to be the main component that leads to the block coupling behavior seen in the Mehta-Schwab paper. Indeed, results in image processing, such as Ho (2013), found that L1 regularization leads to localized learning, while weight decay leads to random weight patterns.

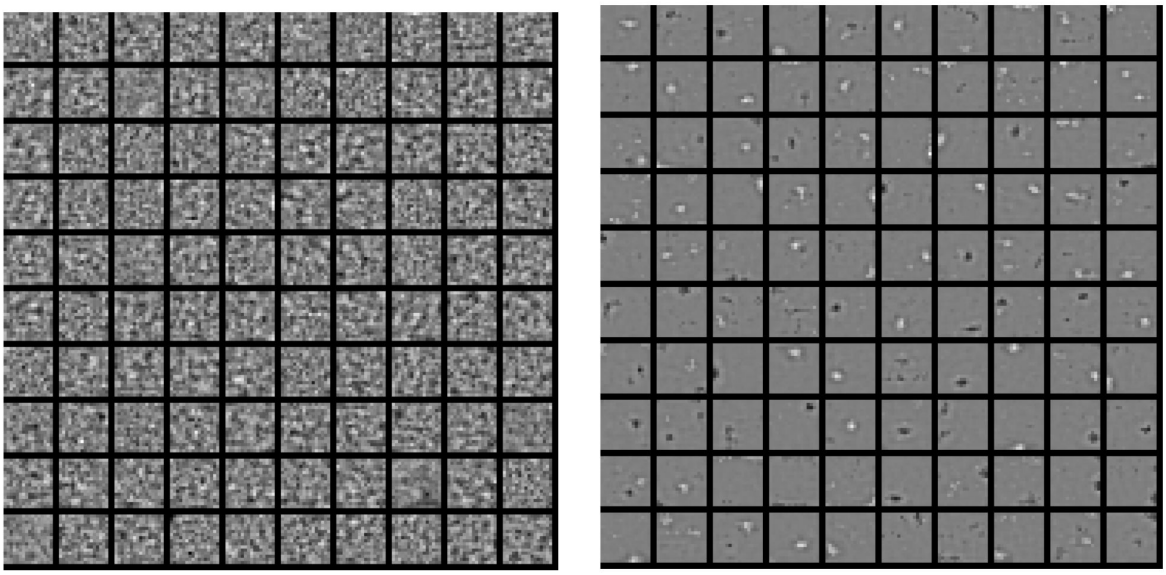

Fig 1: From Ho (2013). Left side is the weight image of 100 hidden nodes without L1 regularization. Right is 100 hidden nodes with L1 regularization. Notice that while the images are not Ising models, the behavior of the right plot is similar to that described in Mehta-Schwab.

The quantitative part (code and commentaries) could be accessed here.

Discussion:

In the section above, we dissect the model (i.e., section 2 of the paper). In this discussion, we look at the one-on-one mapping of the paper.

The one-on-one mapping is proved mathematically by Mehta and Schwab by showing the Hamiltonian between the variational renormalization group process and the RBM are the same.

$$H(\boldsymbol{v}) = H_{\lambda}^{RBM}(\boldsymbol{v})$$

However, this is achieved by an arbitrary selection of the operator T of the variational renormalization group. In particular,

$$\mathbf{T}(\{v_i\},\{h_j\}) = -\mathbf{E}(\{v_i\},\{h_j\}) + \mathbf{H}[\{v_i\}], \qquad (18)$$

Which allows the mapping of the Hamiltonians to be possible. However, substituting eq.18 back to eq.6, which is:

$$e^{-\mathbf{H}_{\lambda}^{RG}[\{h_j\}]} \equiv \mathrm{Tr}_{v_i} e^{\mathbf{T}_{\lambda}(\{v_i\},\{h_j\})-\mathbf{H}(\{v_i\})}. \qquad (6)$$

We have an equation that depends entirely on the energy function $E(\{v_i\},\ \{h_j\})$ of the RBM. That is, we have transformed the operator T to $T' = -E(\{v_i\},\ \{h_j\})$ and therefore implies that the parameters λ for the operator T now is the parameters λ', which is the weights in the RBM model. However, this does not entirely characterize the renormalization group. Instead, it only shows the equivalence between the distribution of the spin configuration between RG and RBM, which is rather obvious since the RBM is based on a Boltzmann joint distribution between the visible and hidden states. The same equivalence could be drawn between RBM and any other technique with a similar distribution function. What truly characterizes the RG, hence required for the one-on-one mapping, is that some form of logical renormalization, which is essentially what the operator T does, is shown in both models. As shown in the quantitative report, and as supported by Koch-Janusz and Ringel (2016), while the RBM did show some form of renormalization, the weights do not make physical sense. At this stage, it seems fair to conclude that what the RBM does is just the act of discarding noises in the initial Ising model data due to the state space reduction from one layer to another (i.e., limited learning capacity). The RBM does not demonstrate localized weight distribution unless specified to do so using L1 regularization. Therefore, I do not believe the proof is entirely rigorous, and that RBM (let alone deep learning in general) does not employ renormalization (in its physical sense) naturally during learning.

Connection to LOs:

#renomalizingData: Our application of renormalizingData is supported mainly by the quantitative component of the project. Given a set of Ising model data, the task at hand is to coarse-grain them via a deep belief network. While the model could not successfully coarse-grain the data, I explored the reasons for failure to explain why the existing models that accomplish the job are not generalizable. In particular, a set of detailed instructions either on the cost function or the update process is needed to ensure localized couplings between the original configuration and the coarse-grained configuration. Therefore, deep learning does not inherently conduct renormalization on its own, which invalidates the premise of Mehta-Schwab.

#renormalizingModels: This process is, again, demonstrated in the quantitative component of the project. In particular, we compare the models after iterations of coarse-graining (fit-transforming) of the data, in absolute terms by comparing the weight plots, and in relative terms by visualizing the effective receptive fields. The comparison aforementioned shows the changes in parameters (weights of hidden nodes in this case) in learning coarse-grained

versions of the given data, as well as how weights in one layer of the stacked RBM model influence the weights in another layer. In our case (that the model successfully coarse-grains the data, albeit not with block renormalization inherently), this process demonstrates the fact that the deep neural network model (characterized by the cost function) remains unchanged throughout iterations of coarse-graining (only the parameter values - characterized by matrices in a linear system - change), which means the effective theory has not been altered.

#renormalizationFlow: In contrast to #renormalizingModels, which inspects the relation of the sets of weights produced by each layer of the model, #renormalizationFlow inspects the flow of weights (or parameters in the general sense) of each layer of the model. In particular, the weights are continually varied after each epoch to optimize a pseudo-likelihood metric associated with RBM. This process is run in a long enough time (50 epochs, as specified by Mehta and Schwab) to attain a fixed state. That is, the pseudo-likelihood stops improving, thus weights stop being changed. In the context of machine learning, however, this fixed point should not be interpreted in isolation without considering overfitting. Optimization until metrics stop improving makes the model prone to overfitting, which is demonstrated when the data are reconstructed from hidden nodes in the RBM stack. Due to time constraint, this aspect of fixed point is not thoroughly explored in this project. However, Koch-Janusz and Ringel (2016) has applied a similar model on the dimer configuration to obtain renormalization that does not make physical sense. This shows that the model works well for Ising model but could not be generalized to more complicated ones, thus overfitted in a sense.

Reference

Ho, N. 2013. RBM, L1 vs L2 weight decay penalty for images. Retrieved from http://nghiaho.com/?p=1796

Koch-Janusz, M., Ringel, Z. 2016. Mutual information, neural networks, and the renormalization group. Retrieved from https://arxiv.org/pdf/1704.06279.pdf.

Mehta, P., Schwab, D. J. 2014. An exact mapping between variational renormalization group and deep learning. Retrieved from https://arxiv.org/pdf/1410.3831.pdf.

Wolchover, N. 2016. A common logic to seeing cats and cosmos. Retrieved from https://www.quantamagazine.org/deep-learning-relies-on-renormalization-physicists-find-20141204/.