MSc in Photonics

PHOTONICS**BCN**

Universitat Politècnica de Catalunya (UPC)
Universitat Autònoma de Barcelona (UAB)
Universitat de Barcelona (UB)
Institut de Ciències Fotòniques (ICFO)

http://www.photonicsbcn.eu

Master in Photonics

MASTER THESIS WORK

APPLICATIONS OF MACHINE LEARNING TO STUDIES OF QUANTUM PHASE TRANSITIONS

Laura Malo Roset

Supervised by Prof. M. Lewenstein and Dr. A. Dauphin, (ICFO)

Presented on date 6th September 2019

Registered at

Escola Tècnica Superior d'Enginyeria de Telecomunicació de Barcelona

Applications of Machine Learning to Studies of Quantum Phase Transitions

Laura Malo Roset

ICFO-Institut de Ciencies Fotoniques, The Barcelona Institute of Science and Technology, 088060 Castelldefels (Barcelona), Spain

E-mail: laumaro95@gmail.com

Abstract. In the past years Machine Learning has shown to be a useful tool in quantum many-body physics to detect phase transitions. Being able to identify phases via machine learning introduces the question of how did the algorithm learn to classify them, and thus how to interpret the model's prediction. In this thesis we present a study of the transition from a normal insulator to a topological insulator. We study this quantum phase transition in the framework of the Su-Schrieffer-Heeger model. In the area of Deep Learning, we introduce two models, a normal convolutional neural network and a model based on deep residual learning. In particular, we focus on the interpretability of the model and its prediction by generating class activation maps (CAM) using a global average pooling (GAP) layer. We show the application of this technique by applying it on the model without disorder and with disorder. Here we give further analysis of the detection of states using transfer learning from no disordered to disordered systems. We conclude that the neural network is able to detect edge states when there is no disorder but unable to distinguish between edge states and Anderson localized states when disorder is introduced.

Keywords: Phase classification, Topological Insulator, Anderson localization, Machine learning for physicists, Interpretability of Machine Learning in physics.

1. Introduction

Phase transition is an important area of research. A successful theory to characterize phase transition is the Ginzburg-Landau theory (1). There, the phases appear through a spontaneous symmetry-breaking and are characterized by a local order parameter. There are however phases that escape this classification. A famous example are the topological insulators. These exotic phases are characterized by a global order parameter, which makes them robust against local perturbations. The task of classifying phases can be seen as putting labels to the different states. This leads to a growing interest in introducing Machine Learning algorithms (2) to perform this task. With that, being able to explore the link between the two fields. This new approach to phase transitions can be done by using unsupervised learning, when the labels are unknown, or with supervised learning, when they are known in advance. We here focus on the supervised learning algorithms. Several works have been able to prove the good performance of machine learning when it comes to identify the order parameter in a phase transition (3; 4). But for physicists, the interpretation (5) on how the decision has been made is key in order to understand if any physical meaning has been taken into account during the prediction. In this work we combine the performance of machine learning algorithms on phase transitions ruled by

a topological order parameter (6) with the idea of making machine learning models and their decisions interpretable.

2. Physical background

We consider a one-dimensional topological insulator (TI), the Su-Schrieffer-Hegger (SSH) model (7; 8). This model describes a one-dimensional tight-binding Hamiltonian of spinless fermions with a staggered hopping, described by

$$\hat{H} = \sum_{n=1}^{N} t_n \hat{a}_n^\dagger \hat{a}_{n+1} + h.c., \tag{1}$$

where t_n is the position-dependent hopping amplitude, $\hat{a}^\dagger$ and $\hat{a}$ are the creation and anihilation operators and N the total number of states. For a staggered hopping $t_1 - t_2 - t_1 - t_2 - ...$, one can introduce a two-atom unit cell [see Fig. 1(a)] and the Hamiltonian reads,

$$\hat{H} = \sum_{m=1}^{M} t_1 \hat{a}_{2m-1,A}^\dagger \hat{a}_{2m,B} + t_2 \hat{a}_{2m,B}^\dagger \hat{a}_{2m+1,A} + h.c. \tag{2}$$

where $\hat{a}_{m,\alpha}/\hat{a}_{m,\alpha}^\dagger$ anihilates/creates a particle in the unit cell m and in the sublattice α where $\alpha \in \{A, B\}$.

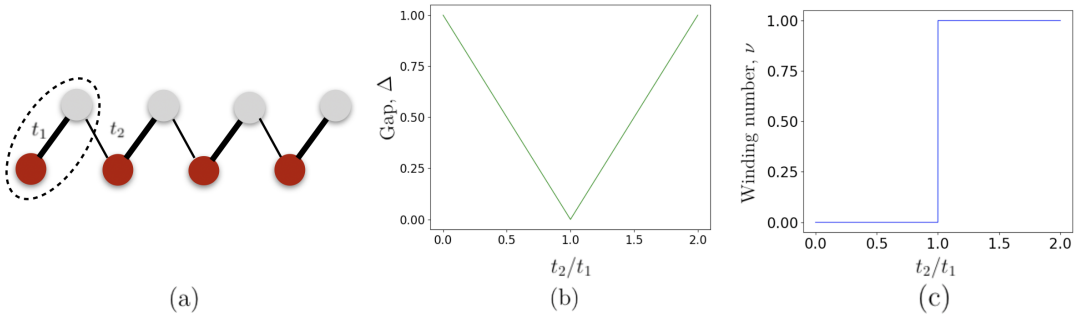

Figure 1. **(a)** Scheme of a SSH chain. The hopping amplitudes are denoted by t_1, the intracell hopping amplitude (thick line), and t_2, the intercell hopping amplitude (thin line). Two sublattices can be observed, sublattice A (red dots) and sublattice B (grey dots). In a doted line is marked the unit cell $m = 1$, where m is the index for the different unit cells and the total number of unit cells, M (in our case M=4). Each unit cell is formed by two sites. In case of m=1, $n = 1$ and $n = 2$ where n is the index of the different states and the total number of states is $N = 2M$ which in our case reads $N = 8$. **(b)** Energy gap as a function of the ratio t_2/t_1. If the hopping amplitudes are staggered, $t_1 \neq t_2$ the model describes an insulator, for $t_1 = t_2$ describes a conductor. **(c)** Winding number as a function of the ratio t_2/t_1. Two different phases can be clearly identified, the phase transition occurs at $t_2/t_1 = 1$.

If we consider a two-atom unit cell, one can apply the Bloch's theorem and write the Hamiltonian in momentum space. The latter is described by a 2x2 matrix,

$$H(k) = d(k)\hat{\sigma}, \tag{3}$$

where $\hat{\sigma} = (\hat{\sigma}_x, \hat{\sigma_y}, \hat{\sigma}_z)$ is the Pauli vector and the components of the k-dependent 3-dimensional vector $d(k)$ read, $d(k) = (t_1 + t_2 \cos k, t_1 \sin k, 0)$.

The SSH model for staggered hopping amplitudes ($t_1 \neq t_2$) describes an insulator. The gap, Δ, behaves as described in Fig. 1(b), and two different types of insulators can be observed. For $t_1 > t_2$ it describes a normal insulator whereas for $t_1 < t_2$ a topological insulator. These two kinds of insulators described by the SSH model can be characterized by a topological invariant that is an integer number characterizing an insulating Hamiltonian. In this case, this topological invariant is the winding number, ν. The winding number predicts the number of protected edge states appearing at each edge of a finite size chain with open boundary conditions. Analytically can be computed as,

$$\nu = \frac{1}{2\pi i}\int_{-\pi}^{\pi} dk \frac{d}{dk}\log h(k) \tag{4}$$

where the parameter h is $h(k) = d_x(k) - id_y(k) = t_1 + t_2\cos k - it_2 \sin k$. Therefore, the final expression to be computed for the winding number reads,

$$\nu = \frac{-t_2}{2\pi i}\int_{-\pi}^{\pi} dk \frac{(\sin k + i\cos k)}{t_1 + t_2\cos k - it_2\sin k} \tag{5}$$

In the SSH model, the winding number can take two possible values depending on the dominant hopping amplitude, thus two different cases can be considered, $t_1 \gg t_2 \Rightarrow \nu = 0$ and $t_1 \ll t_2 \Rightarrow \nu = 1$.

The mentioned edge states correspond to a pair of zero energy eigenstates, whose wavefunctions are localized at each end of the chain, and it only appear when the winding number is nonzero. This allows us to easily differentiate between the topological phase [Fig. 2(b)], where edge states appear, from the trivial phase [Fig. 2(a)], where they do not.

So far we have considered a model without disorder, where the two phases are well characterized by the presence of edge states in the topological phase. It has been a matter of interest the combination of topology and disorder to prove the robustness of the topological insulator. It is indeed the case as the TI is characterized by a global topological invariant, therefore not sensitive to local perturbations such as disorder. In 1D for any amount of disorder, the states are Anderson localized. This Anderson localization can be diagnosed by eigenstate characterization in where eigenfunctions are localized around some localization center and are decaying exponentially away from it with a certain localization length.

It is interesting to prove the interplay between Anderson localization (9; 10) and topology. We introduce the disorder in the Hamiltonian given in Eq.(2) via creating precisely defined disorder in the off-diagonal hopping amplitude terms,

$$\begin{aligned} t_{1,n} &= t_1(1 + W_1\omega_n) \\ t_{2,n'} &= t_2(1 + W_2\omega_{n'}) \end{aligned} \tag{6}$$

where ω_n and $\omega_{n'}$ are independent randomly generated numbers drawn from the uniform distribution [-0.5,0.5] and W_1 and W_2 are the tunneling disorder strengths that we consider $W = W_2 = 2W_1$ as is done in Refs. (11; 12), where W is the disorder strength.

We take into consideration the phase diagram for the disorder-driven transition from topological to trivial wires obtained in Ref. (12) [See Fig. 2(e)] and we use as weak disorder the value ($W = 0.5$) and strong disorder, ($W = 5$). We can observe [see Fig. 2 (d)] how topological features can eventually disappear when the disorder strength becomes too large. It has also been proved that the point in which the phase transition happens when disorder is added is no longer at $t_2/t_1 = 1$ when the disorder strength is increased. We consider a disorder strength of $W \in \{0,0.75\}$ where the transition point is considered close to $t_2/t_1 = 1$.

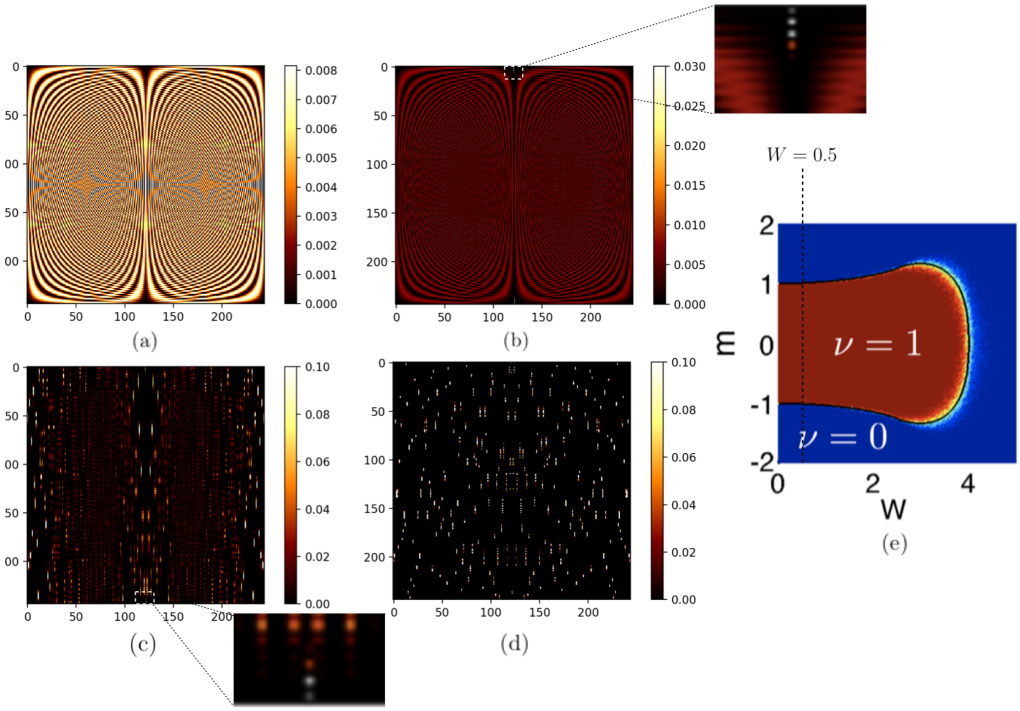

Figure 2. Density profile for the SSH model with different hopping amplitudes and disorder strength. **(a)** Hopping amplitudes set to $t_1 = 2$ and $t_2 = 1$ and no disorder, $W = 0$. Corresponds to a trivial phase. **(b)** Hopping amplitudes set to $t_1 = 1$ and $t_2 = 2$ and no disorder, $W = 0$. Corresponds to a topological phase, edge states appear at the top and bottom of the center of the density profile and are indicated by a blue dashed line. T **(c)** Hopping amplitudes set to $t_1 = 1$ and $t_2 = 2$ with weak disorder, W=0.5. Corresponds to a topological phase as the edge states can still be identified. **(d)** Hopping amplitudes set to $t_1 = 1$ and $t_2 = 2$ with strong disorder (W=5). The topological features start to disappear for this disorder strength. **(e)** Phase diagram for the disorder-driven transition from topological to trivial wires. Figure adapted from Ref. (12). Red corresponds to the topological phase and blue to the trivial phase. The ratio m is described as $m = t_2/t_1$ and W, the disorder strength. A black dashed line indicates the area where we work. Images (b-d) have been saturated for a better visualization.

3. Machine Learning

3.1. Introduction

Machine Learning (ML) has opened a whole new area of research in science where these algorithms can be applied to different fields such as biology, meteorology or quantum physics (13) among others. In this work, we focus on Deep Learning (DL), a subfield of machine learning in which learning algorithms inspired by the brains neurons, called Artificial Neural Networks (ANNs), are used for different tasks such as classification. Different structures of ANNs can be considered depending on the type of layers that formed their structure. Fully connected layers correspond to layers in which all the neurons of each layer are connected with a specific weights to all the neurons of the next layer [see Fig. 3 (a)]. NN formed exclusively by this type of layers have proved to have a good performance in datasets such as the handwritten digits dataset MNIST (14) but, when considering more dificult datasets, for instance CIFAR (15), the number of weights to be trained quickly grows, the neural network shows poor performances and other architectures show better results.

In particular, convolutional neural networks (CNNs), a discrete convolution where the

weights of the convolution are learned during the training, have shown in the recent years outstanding results when it comes to classification tasks. CNNs are neural networks with a high performance in identifying particular patterns in images having as its main feature the fact that the connections between the different layers of neurons are implemented by filters [see Fig. 3(b)], reducing in that way the number of weights in the training.

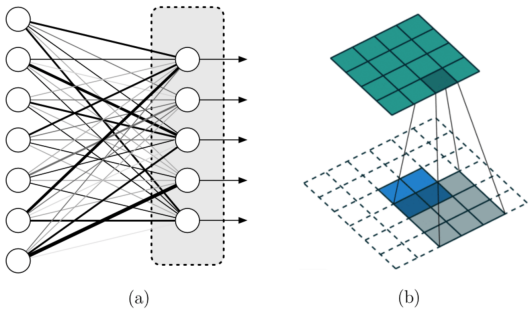

Figure 3. (a) Representation of a FCNN, all the neurons of the first layer are connected to all the filters of the second layer. **(b)** Representation of the application of a filter in a CNN. The filter (green) of size 4x4 is applied to the input layer. Figures are adapted from Ref. (16).

Different types of learning processes can be considered during the training, we focus on supervised learning in which the input of the neural network is an image with its corresponding label indicating to which class it belongs. The output of the neural network here predicts the label.

The training consists in the minimization of a loss function, in particular we consider the *Binary Cross Entropy Loss* that measures the performance of a classification model whose output is a probability value between 0 and 1. Cross entropy loss increases as the predicted probability diverges from the actual label.

$$L = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)), \tag{7}$$

where y is the label (1 for one class and 0 for the other) and $p(y)$ is the predicted probability of the point being class 1 for all N points.

During the training the weights have to be updated following an optimization process, a common method is to apply *Gradient Descent (GD)* that is an iterative method which is used to find the values that minimizes the loss function,

$$\omega^{t+1} = \omega^t - \alpha \frac{\partial L}{\partial \omega} \tag{8}$$

where ω are the weights that are being update and L is the loss function that we want to minimize. We will add L_2 regularization to prevent overfitting[1]. When adding L_2 regularization the computed loss reads,

$$L_{L_2} = L + \frac{1}{2}\lambda \sum_j |\omega_j|^2, \tag{9}$$

[1] Overfitting is one of the most common problems machine learning can face during the training and it is a consequence of the model being very good on the training set but unable to classify well data that is not part of this set. Leading to a model very well trained but unable to generalize.

which adds an extra term in the loss function to penalize high weights. Therefore, the weight's update reads,

$$\omega_j^{t+1} = (1-\lambda)\omega_j^t - \alpha\frac{\partial L}{\partial \omega_j} \tag{10}$$

where λ and α are hyperparameters.

In particular, we use in the training *Stochastic Gradient Descent (SGD)*, where a few samples are selected randomly instead of the whole data set for each iteration in the optimization process. For instance, if the data set is (X,Y) where X are the images and Y their corresponding labels being (x,y) a sample (where $x \in X$ and $y \in Y$), we take a random subset of (X,Y). While in GD all the samples of the training set have to go through the neural network to do a single update for the weights, in SGD only one random sample is needed at each update. It has been proved that SGD often converges much faster compared to GD and reduces the training time for larger training sets. More precisely, we use Stochastic Gradient Descent with momentum. The addition of a momentum makes SGD accelerate gradient vectors in the right direction, thus leading to faster converging.

3.2. Model architecture

The first architecture we study is a CNN formed by five convolutional layers stacked as a sequential model where after the last convolutional layer and before the linear layer, a global average pooling layer is implemented [see Fig. 4(a)]. A global average pooling layer is a pooling layer [2] that takes each of the channels of the previous layer and returns their spatial average.

After some of the plain blocks, a max pooling layer is introduced. This type of pooling layer pass over the different matrices of pixels of the image and pool them into the highest value, saving only the important features with its location.

First it seemed the deeper the CNN, the better. However, when the CNN becomes too deep, the problem of the vanishing gradient appears: as the gradient comes from the chain rule (backpropagation) and when you backpropagate the error it becomes smaller and smaller and thus the vanishing gradient. In Ref. (18) the degradation problem is address by introducing a deep residual learning framework. By using this residual CCN, you avoid the vanishing gradient problem.

Introducing residual connections means connecting the output of previous layers to the output of new layers (see Fig.4(d)). In a residual setup you wouldn't only pass the output of layer 1 to layer 2 and on, but you would also add up the outputs of layer 1 to the output of layer 2 in aim to avoid learning unreferenced functions. Network architectures based on residual blocks are the so called ResNets.

Many ResNets have been implemented and tested on different datasets (19) showing optimal results in the training error and accuracy of the models. This introduces our second model, the 18 layers deep ResNet18 (see Fig. 4 (b)).

3.3. Interpretability

Machine Learning algorithms give information about the label predicted but, knowing the reason why this label has been picked over the others can also provide useful information. The interpretability techniques become essential for humans to validate the decisions of the machine (5).

[2] Pooling layers are one of the types of CNNs layers (convolutional layers, pooling layers, normalization layers and fully connected layers) that extract the features that better characterize the data and get rid of the rest. This type of layers reduce the dimension of the image by dividing the image in a serie of blocks of pixels and do some operation on these blocks (for instance, taking the maximum or taking the average). Max pooling is taking the maximum on a typical block of 2x2 pixels and the Global Average Pooling is taking the average over the whole image as unique block.

Reference (20) proposed an interpretatbility for CNNs with a GAP layer, called Class Activation Map (CAM). Class activation mapping is described as a technique to get the discriminative regions used by a CNN to identify a specific class in an image. This technique is based in projecting back the weights of the output layer on the convolutional feature maps.

The CAM technique creates a heatmap in which are shown the parts of the image the CNN is focusing on. To construct this heatmap we take the activated features of the last convolutional layer, the weights of the fully connected layer (on the side of the average pooling) and the class index we want to investigate. We index into the fully-connected layer to get the weights for that class and calculate the dot product with the features from the image.

For being able to apply this technique to a model, the network architecture must fulfill some requirements, after the last convolutional layer and before a unique linear (dense) layer that will produce the output, a global average pooling layer must be placed. With that, it's able to give enough information in order to localize the discriminative regions.

Therefore, the overall result we obtain by applying this technique is to be able to identify which regions of the image are being used for discrimination as can be seen in Fig. 5.

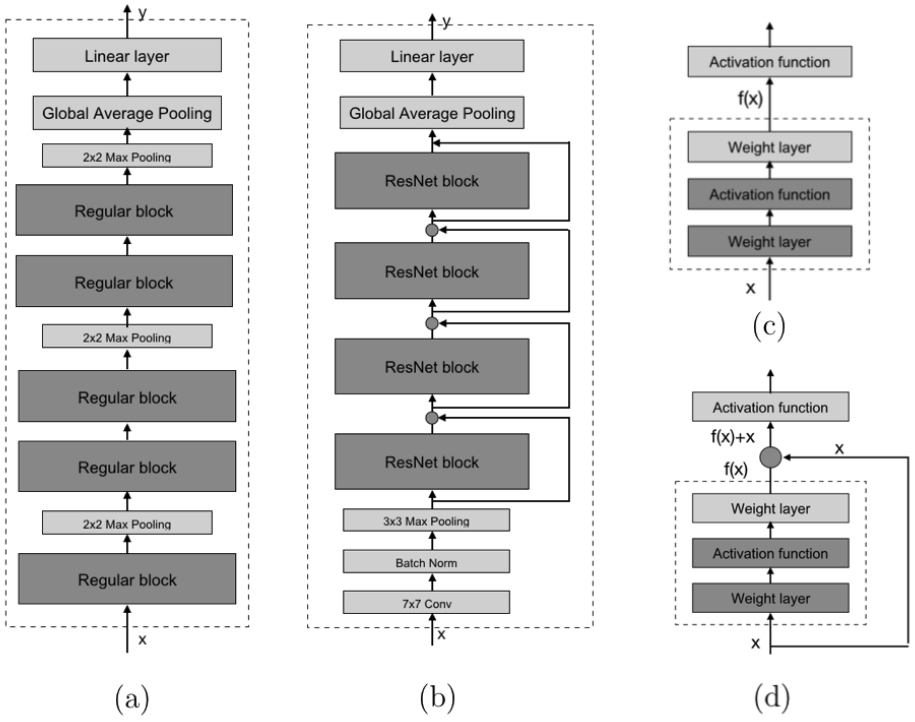

Figure 4. **(a)**Architecture of our CNN. **(b)**Architecture of ResNet18. **(c)**Regular block. **(d)**Residual block.

3.4. Application to the SSH model

We train a CNN on the SSH model. We produce the data via exact diagonalization of the Hamiltonian of the SSH model described in Eq. (2). The input data for the CNN are the density profiles of the states for different ratios t_2/t_1 where we set $t_2 = 1$ and t_1 goes from 0 to 2. The matrices obtained have a size 2Mx2M being M the cell number, for the ResNet18 M=122 and for our CNN model M=30 since the input sizes are 244x244 and 60x60 respectively. For each dataset we produce a training set, a validation set and a test set formed by 10000, 4000 and 6000 images respectively. We produce different datasets considering no disorder, W=0, and disorder with values compressed in W$\in$ {0.25,0.75}.

The training for the two models is done with the same set of hyperparameters, trained for 100 epochs using a Batch size of 100 images. In the optimization process the learning rate is set to 0.001 and the momentum to 0.9. L_2 regularization is introduced [Eq. (10)] with a weight decay value of 0.01.

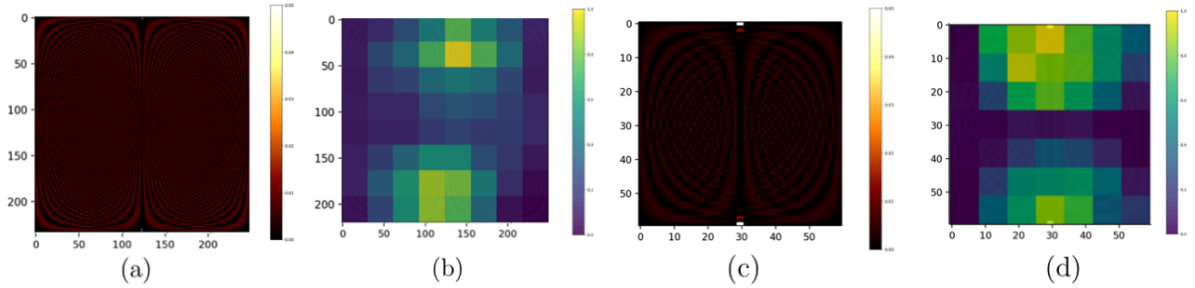

Figure 5. **(a)** Image of size 244x244 of a topological phase. **(b)** Class Activation map of the image in (a) using the ResNet18 architecture. **(c)** Image of size 60x60 of a topological phase. **(d)** Class Activation map of the image in (c) using our CNN architecture.

In Fig.5 a comparison between the heatmaps obtained by the ResNet18 and the CNN can be observed. Even if both methods are able to detect the edges as the discriminative regions, the former exhibits a better performance where a better precision is achieved.

4. Transfer learning

We now benchmark transfer learning between the SSH model without disorder and with disorder and use the CAM to interpret the results. Transfer learning is a method in which a model trained for one task is reused as a starting point for a model on a second related task.

We train our neural network with a training set and validation set without disorder and then, via transfer learning, test the model on a test set where disorder is included. We also do the same with datasets with disorder but with different disorder strengths.

4.1. Transfer learning to non-disorder to disorder

The model is trained using images without disorder and tested via transfer learning with a new dataset where disorder (W=0.5) has been introduced. The test accuracy for images without disorder is 99,7%. However, when testing it on images with disorder, we observe a drop in the accuracy and find an accuracy of 53,73%. Our aim is to figure out via interpretability why this latter classification fails and understand if there is a physical explanation behind it.

By analysing the Class Activation Maps (see Fig. 6), we show that the discriminative regions are Anderson localized states. This result can be contrasted in the Topological case where for no disorder edge states where the discriminative regions. By computing other magnitudes (TP,FP,TN,FN)[3] we conclude that almost all the images where classified as Topological even if they where labeled as Trivial. We propose as an explanation for that that the NN learns to identify if there are localized states or not, and when in the presence of a localize state classifies the image as Topological without being able to distinguish between edge states and Anderson localized states.

[3] True positive (**TP**): when a data point is classified as class 1 and it actually belongs to class 1. False positive (**FP**): when a data point is classified as class 1 but it actually belongs to class 0. True negative (**TN**): when a data point is classified as class 0 and it actually belongs to class 0. False negative (**FN**): when a data point is classified as class 0 but it actually belongs to class 1.

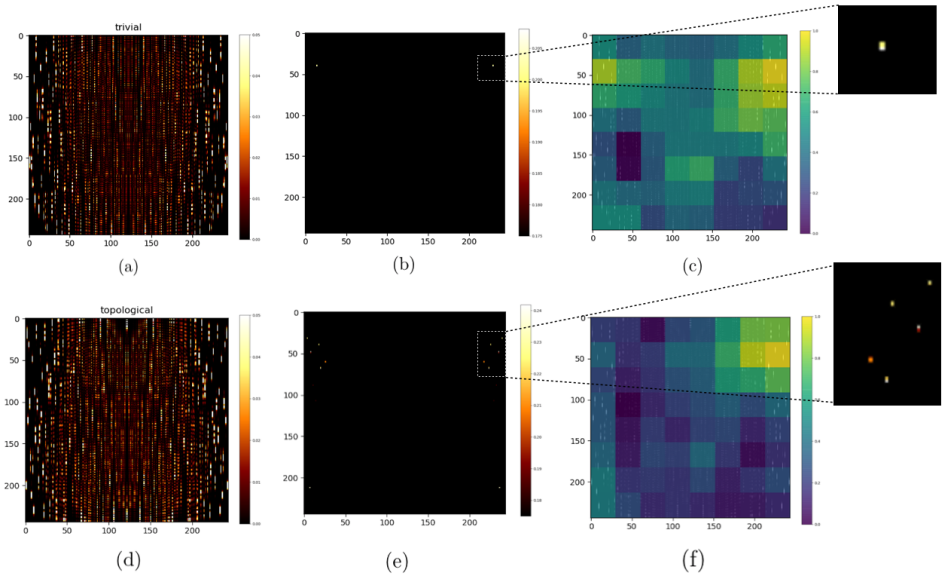

Figure 6. **(a)** Density profile corresponding to a trivial phase. **(b)**Maximums of the density profile in (a). **(c)**Class Activation Map of image (a). **(d)** Density profile corresponding to a topological phase. (e) Maximums of the density profile in (d). **(f)** Class Activation Map of image (d).

4.2. Transfer learning for different disorder strengths

We address the problem of knowing if transfer learning still works when disorder is increased or decreased by training our model with small (W=0.25) and bigger (W=0.75) disorder and testing it with different disorders compressed in this range. We obtain that the training is local and a slight modification of the disorder makes the accuracy rapidly drop (see Fig. 7). Our results also show that the performance is better when going from big disorder to smaller disorder that in the other direction.

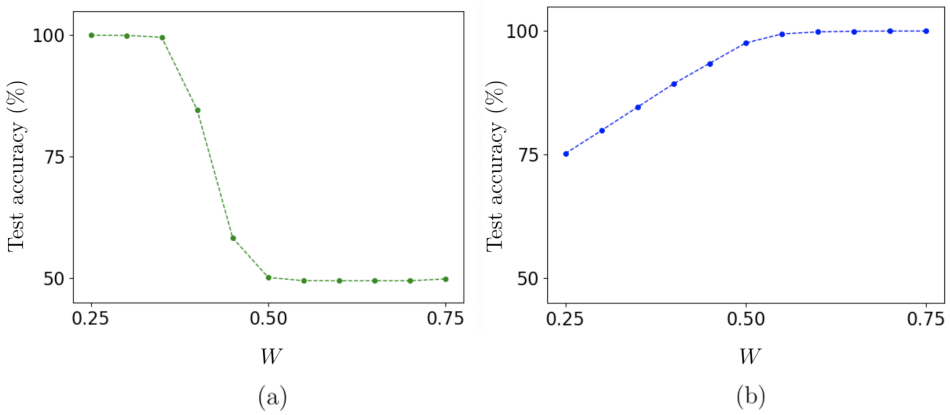

Figure 7. **(a)** Transfer learning from small disorder to bigger disorder. Test accuracy obtained for different disorder strength, W, using a model trained with W=0.25. **(b)** Transfer learning from big disorder to smaller disorder. Test accuracy obtained for different disorder strength, W, using a model trained with W=0.75.

5. Conclusions

To summarize our work, we have studied the topological insulator by characterizing the two phases with the winding number. We have trained two neural networks, a CNN architecture and a ResNet18, for this classification task. We have obtained a good performance when considering the non-disordered case but, it failed when introducing disorder. We have been able through interpretability to find a physical explanation of that, concluding that Anderson localized states due to disorder make the algorithm unable to distinguish them from edge states.

We have also showed that transfer learning for disorder systems only works locally. A further step could be adding more disorder realizations to test if its performance would increase or use data augmentation.

We suggest that other neural network, such as domain adversarial neural networks, can be used to address this problem aiming for a different interpretability of the predictions.

Acknowledgements

I want to acknowledge Prof. M. Lewenstein and Quantum Optics Theory group for the opportunity to do this reseach with them. Also, A. Dauphin and P. Huembeli for supervising me and for useful discussions during all the thesis.

References

[1] Hohenberg, P. and Krekhov, A. An introduction to the Ginzburg-Landau theory of phase transitions and nonequilibrium patterns. *Phys. Rep.* **572**, 142 (2015). *arXiv:1410.7285*

[2] Giuseppe Carleo, Ignacio Cirac, Kyle Cranmer, Laurent Daudet, Maria Schuld, Naftali Tishby, Leslie Vogt-Maranto, Lenka Zdeborov. Machine learning and the physical sciences. *Phys. Rev. Lett.* **113**, 046802 (2014)

[3] Juan Carrasquilla and Roger G. Melko, Machine learning phases of matter. *Nature Physics* **13**, 431434

[4] Frank Schindler, Nicolas Regnault, and Titus Neupert, Probing many-body localization with neural networks. *Physical Review* B **95**, 245134 (2017). *arXiv:1704.01578*

[5] Doshi-Velez, Finale, and Been Kim. Towards a rigorous science of interpretable machine learning, *no. ML* 113 (2017) *arxiv:1702.08608*

[6] Deng, D.-L., Li, X. and Sarma, S. D. Machine learning topological states. *Phys. Rev.* B **96**, 195145 (2017). *arXiv:1609.09060*

[7] A. J. Heeger, S. Kivelson, J. R. Schrieffer, and W. P. Su. Solitons in conducting polymers. *Rev. Mod. Phys.* **60**, 781.

[8] Janos K. Asboth, Laszlo Oroszlany, Andras Palyi. A Short Course on Topological Insulators: Band-structure topology and edge states in one and two dimensions. *Lecture Notes in Physics*, **919** (2016). *arXiv:1509.02295*

[9] Serge Aubry and Gilles Andr. Analyticity breaking and Anderson localization in incommensurate lattices. *Ann. Israel Phys.* Soc **3**, 18 (1980).

[10] Jian Li, Rui-Lin Chu, J. K. Jain, and Shun-Qing Shen. Topological Anderson insulator. *Physical Review Letters* **102**, 136806 (2009). *arXiv:0811.3045*

[11] Eric J. Meier, Fangzhao Alex An, Alexandre Dauphin, Maria Maffei, Pietro Massignan, Taylor L. Hughes, Bryce Gadway. Observation of the topological Anderson insulator in disordered atomic wires. *Science* **362**, 929 (2018).

[12] Ian Mondragon-Shem, Juntao Song, Taylor L. Hughes, Emil Prodan. Topological Criticality in the Chiral-Symmetric AIII Class at Strong Disorder. *PRL* **113** , 046802 (2014).

[13] Lenka Zdeborova. New tool in the box. *Nature Physics* **13**, 420421 (2017).

[14] MNIST Database: http://yann.lecun.com/exdb/mnist/

[15] CIFAR Database: https://www.cs.toronto.edu/ kriz/cifar.html

[16] https://github.com/vdumoulin/conv_arithmetic

[17] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. textitIEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2009).

[18] He, K., Zhang, X., Ren, S., Sun, J. Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778, (2016). *arXiv:1512.03385*

[19] Khan, R.U.; Zhang, X.; Kumar, R.; Aboagye, E.O. Evaluating the Performance of ResNet Model Based on Image Recognition. *In Proceedings of the 2018 International Conference on Computing and Articial Intelligence (ICCAI 2018)*, 8690 (2018).

[20] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A. Torralba, A. Learning deep features for discriminative localization. *In Proc. CVPR* 29212929 (2016). *arXiv:1512.04150* (2016).