Feature extraction by target propagation

de Souza Farias, T. and Maziero, J.

1

Abstract. *Opening the black box of neural networks is a problem that have being gaining more importance as neural neural networks develop and have more widespread application. Understanding how neural networks build up their understanding of a task is fundamental for the comprehension of why an algorithm makes some specific decisions. In this article, we introduce a novel technique for visualizing inputs that maximizes the neural activity of specific units of a trained network.*

1. Introduction

Learning algorithms are a paradigm shift from well-defined instructions for problem solving by computers. From well-defined datasets, neural networks produce accurate prediction models that are able to generalize for the tasks trained on. Despite neural networks outperforming humans on various tasks, they are described by a large number of parameters with no apparent meaning, leading to a black-box statistical tool. Even though they are very accurate models "The problem is that a single metric, such as classification accuracy, is an incomplete description of most real-world tasks." [Doshi-Velez and Kim 2017]. Besides, understanding why an algorithm makes a certain decision can be help for developing even better models.

In this sense, methods have emerged for the interpretability [Miller 2019] of neural networks. High interpretability of a machine learning model means to be easier for someone to comprehend why certain decisions or predictions have been made by the model [Molnar 2019]. Feature visualization is an attempt to bridge the parameters of a model to the ability of explanation for decision making. We define features as the elements of input by which the activity of a specific group of neurons is optimized. Understanding which features feed in the input layer entail internal neuron activity or the final output behavior, help us to comprehend how neural networks build up their understanding of a task, such as image classification.

Features visualization is an approach of making the learned features explicit. This is done by finding the input that maximizes the activation of a certain neuron [Erhan et al. 2009], which can reveal what the neuron "wants to see" or "what it is looking for". Activation maximization problem of a neuron is realized assuming that the weights of the neural network are fixed, which means that the network is trained. Approach through optimization we can use derivatives to iteratively tweak the input generating new images, starting from random noise [Olah et al. 2018]. Although feature visualization is a powerful tool, current techniques for visualizing output neurons and hidden neurons involve a number of complicated details.

This work focuses on the use of a simple technique for features extraction of a image classifier by using gradient target propagation learning rule

[de Souza Farias and Maziero 2018]. While backpropagation calculates the weight gradient of the neural network through error propagation to previous layers, target propagation does it by propagating targets layer by layer. The mathematical formulation of this technique naturally leads to targets for the input layer.

2. Feature extraction

After training a neural network, one must choose a layer and a neuron (or set of neurons) to extract its feature. In the forward operation, we set the input as a zero vector to activate all network until the chosen layer. Then a target vector is set (e.g. one-hot vector) and a loss function. In the backward operation, the targets are propagated to the earlier layers until the input, which corresponds to the feature. Algorithm 1 summarizes the process of feature extraction.

Algorithm 1: Feature extraction

Choose a trained neural network;
Initialize input as a zero vector;
Forward the network until the desired layer L to extract a feature ;
Set a target $\hat{a}_L$ for the layer and a loss function $\mathcal{L}_L$;
for *l=L to 0* **do**
  Employ the gradient target propagation algorithm until the input layer l=0
end
Manipulate the target $\hat{a}_0$ for visualization or other operation.

While there are many ways to interpret the explainability of features, here we chose feature visualization and measurement. The visual features aid human interpretation so that they correspond to knowledge about the training data. One-hot vectors are useful to maximize the activity of one single neuron, extracting specific characteristics that leads to high activation.

However, visualization of features does not guarantee a human comprehensible image. Feature measurement compares two distinct features with a distance function. Cosine similarity offers a direct interpretation for the relationship between features, which helps to understand how similar, or different, they are from each other. High positive correlation between features corresponds to a value of 1, meaning high visual similarity, while high negative correlation has a value of -1.

3. Results

All trained models are full feed-forward networks with variable number of hidden layers, hidden neurons and epochs. The parameters of the models were trained by the gradient target propagation with stochastic gradient descent optimizer, sigmoid function for activation of layers and batch size 100. The neural networks were trained on the MNIST image classification problem [Y. LeCun and Haffner 1998].

With the method described in Sec. 2, we extract features from neurons of the output layer. These features are directly related to the classes of the data since the output layer corresponds to the classification of the images, and thus we have a prior expectation of the features shape.

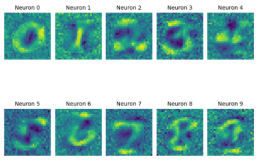
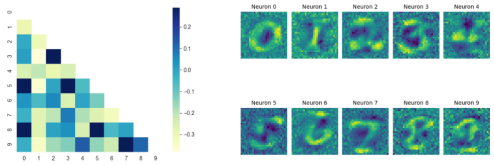


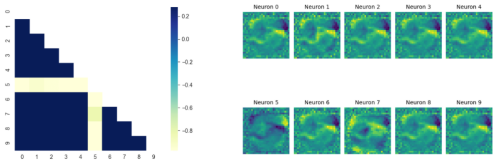
Figure 1. Visual features. Network trained with 1 epoch, 128 hidden neurons, 1 hidden layer, test accuracy: 85.88

Image 1 shows ten features extracted from ten orthogonal target one-hot vectors. There is a visual similarity between each feature and the corresponding class from the training data.

Image 2a shows the visual features and the cosine similarity between each one. As we increased performance, varying the learning rate, features were changed to a visually similar pattern each other but that differ from training data (figure 2b).



(a) 1 epoch, test accuracy: 85.88



(b) 10 epoch, test accuracy: 94.86

Figure 2. Transition by increase epoch. Left: cosine similarity between features. Right: visual features. Network trained with 128 hidden neurons, 1 hidden layer

As a counter-example to human interpretability, figure 2b has almost no distinguishable visual features. The results from the cosine similarity also confirm the absence of similarity of features with training data, with high correlation coefficients, despite the high accuracy.

3.1. Phase transition

All results show a consistent phenomena of training: the features go from random images to high explainable representations of the training data, and further to new images with almost no interpretability. These transitions can be measured with the accuracy on the test data: after about 90% performance, the neural network models change for high to low explainable features. The features behaviour are present in different phases, defined by their explainability.

Image 2 present a transition from increase performance, these phase transition suggests a change in the behaviour of neural networks in order to minimize the loss.

The space of one-hot vectors appears to be distinct from the data projection space, that can motivate this change of behavior. Image 3 outlines the relation between phases and accuracy.

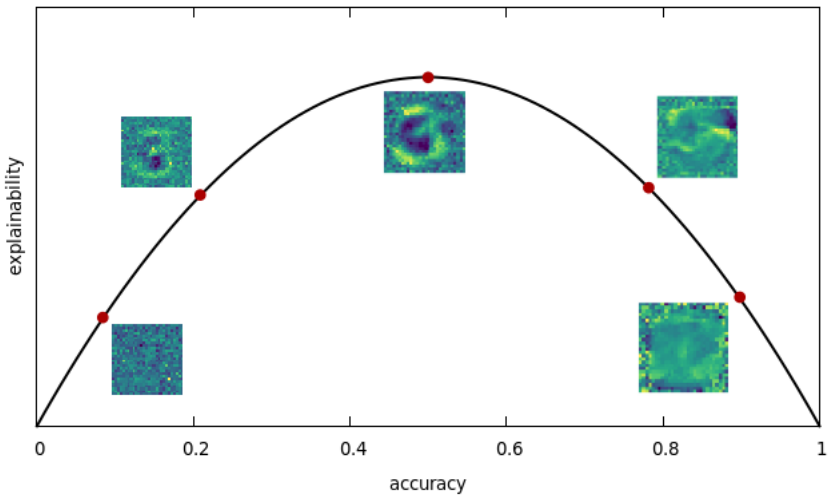


Figure 3. Features phase transitions across training neural network models. Higher accuracies leads to low feature explainability.

Additional results are present in appendix A. Images 5, 6 suggests that low neurons number tend to resist more to phase transition phenomena.

4. Conclusion

We presented an adaptation of the gradient target propagation training rule as a method to extract visual features from neural networks trained on a visual classification tasks. The visual features are useful for the explainability of neural networks, by showing image representations that lead to high activity of selected neurons. These features are related to the training phase, and highlight the importance of specific patterns of the data.

We observed phase transitions with the visual features. They go from random to high and then low interpretability with training time. We proposed that the distinctness between the one-hot and data projection spaces as a justification for this behaviour.

Our results may contribute to improvements on optimization algorithms, like modifications on the loss functions, that can enhance the neural interpretability and better understand the training of neural networks.

References

de Souza Farias, T. and Maziero, J. (2018). Gradient target propagation. *arXiv preprint arXiv:1810.09284*.

Doshi-Velez, F. and Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Erhan, D., Bengio, Y., Courville, A., and Vincent, P. (2009). Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1.

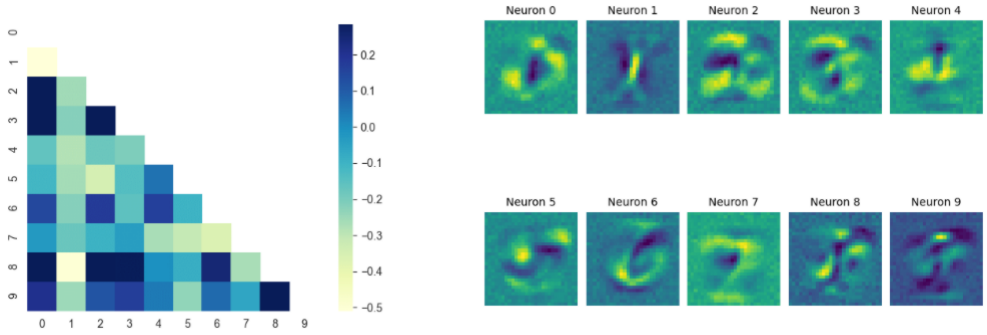
Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38.

Molnar, C. (2019). Interpretable machine learning. *Lulu. com*.

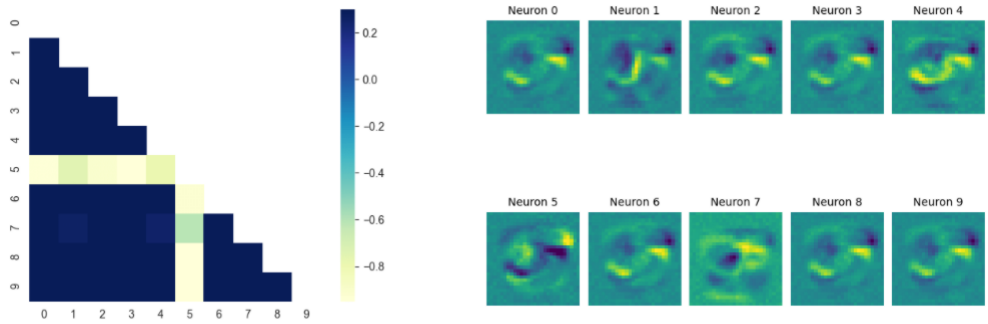
Olah, C., Mordvintsev, A., and Schubert, L. (2018). Feature visualization: How neural networks build up their understanding of images. distill.

Y. LeCun, L. Bottou, Y. B. and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278.

A. Additional results

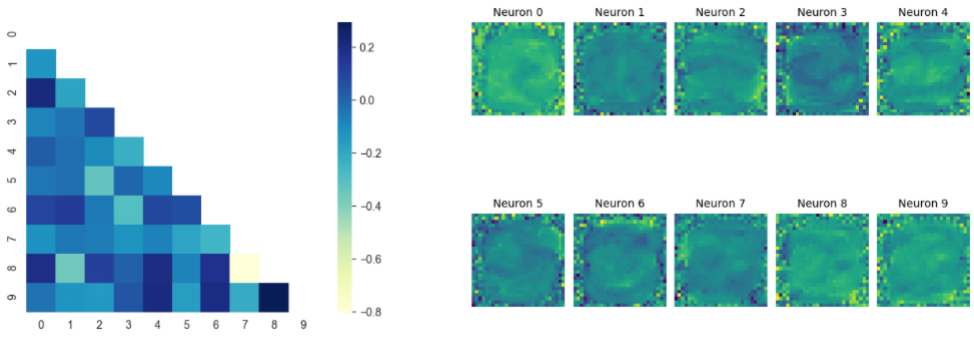


(a) 1 epoch, test accuracy: 89.04

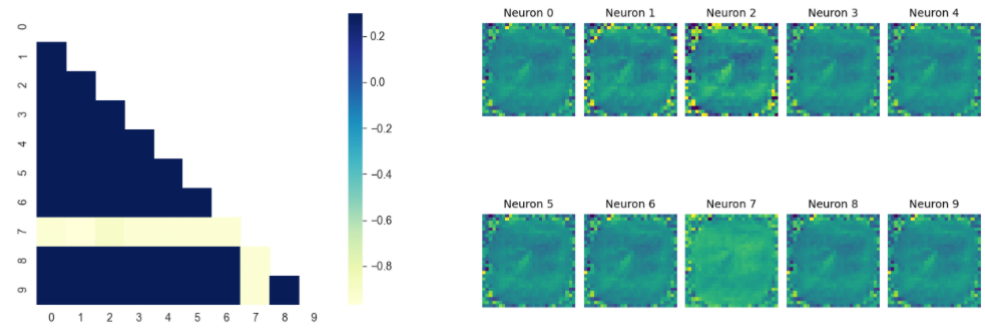


(b) 10 epoch, test accuracy: 92.83

Figure 4. Transition by increase epoch. Left: cosine similarity between features. Right: visual features. Network trained with 512 hidden neurons, 1 hidden layer

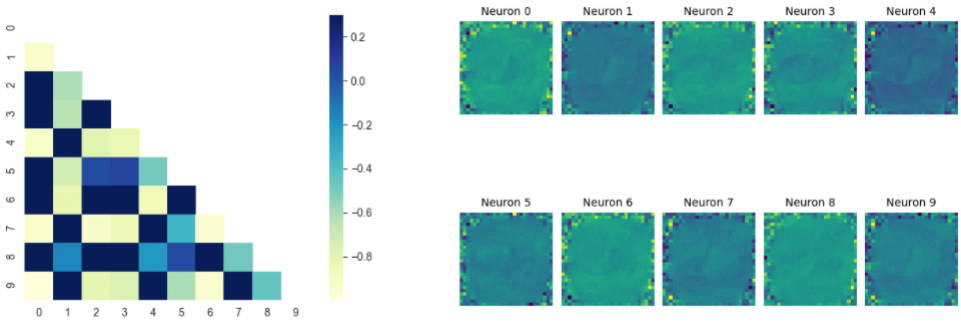


(a) 1 hidden layer, test accuracy: 93.08

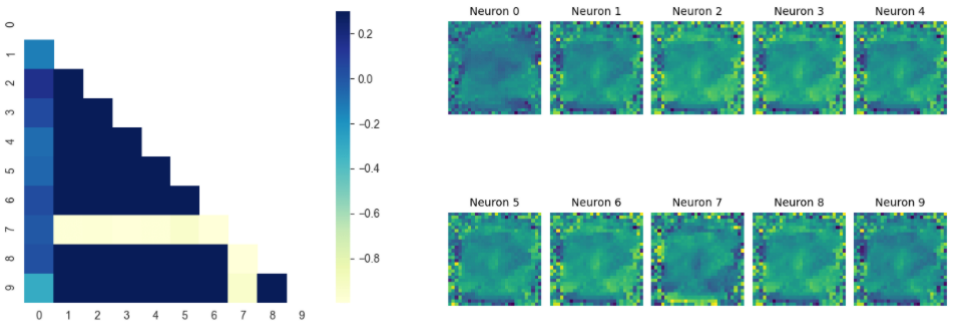


(b) 3 hidden layer, test accuracy: 93.83

Figure 5. Transition by increase hidden layer. Left: cosine similarity between features. Right: visual features. Network trained with 10 epoch, 16 hidden neurons



(a) 8 hidden neurons, test accuracy: 91.26



(b) 16 hidden neurons, test accuracy: 91.26

Figure 6. Transition by increase hidden neurons. Left: cosine similarity between features. Right: visual features. Network trained with 10 epoch, 3 hidden layer

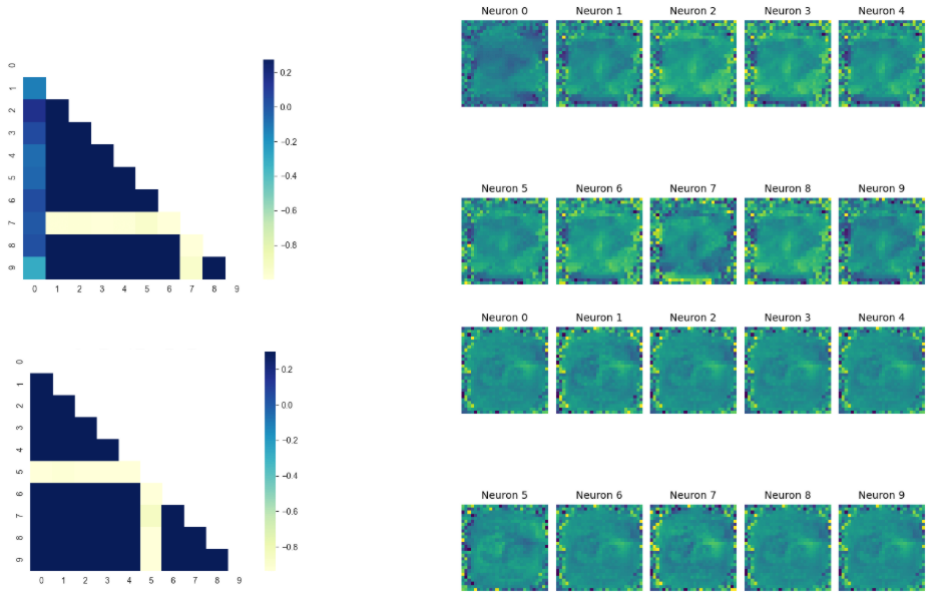


Figure 7. Left: cosine similarity between features. Right: visual features. Network trained with 10 epoch, 16 hidden neurons, 3 hidden layer, test accuracy: 91.266, 94.78

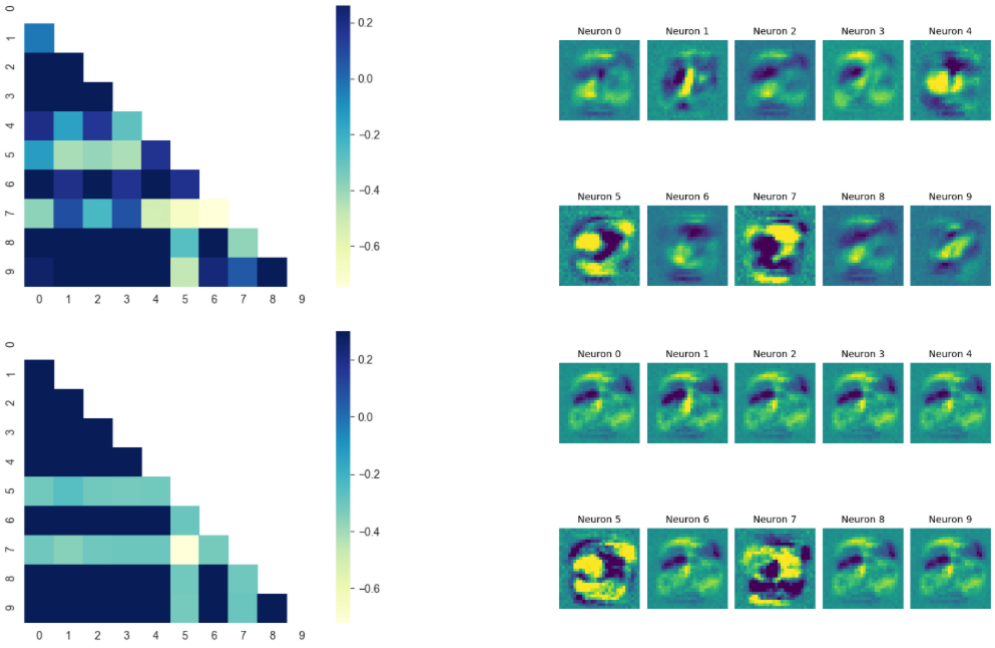


Figure 8. Left: cosine similarity between features. Right: visual features. Network trained with 10 epoch, 1024 hidden neurons, 3 hidden layer. Top test accuracy: 89.64. Bottom test accuracy: 95.17

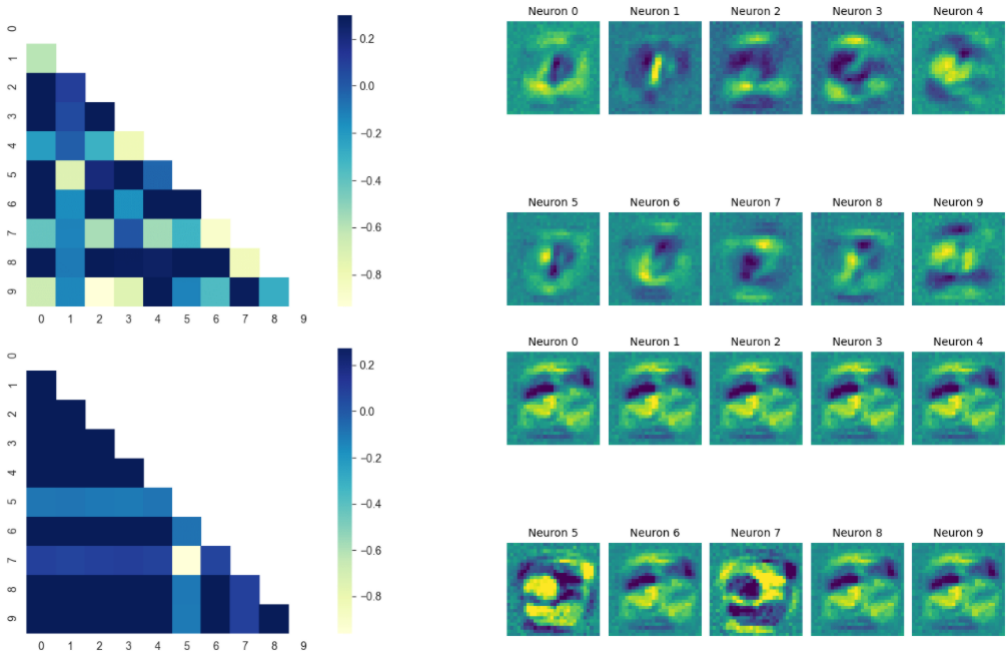


Figure 9. Left: cosine similarity between features. Right: visual features. Network trained with 10 epoch, 512 hidden neurons, 3 hidden layer. Top test accuracy: 84.07. Bottom test accuracy:' 94.03

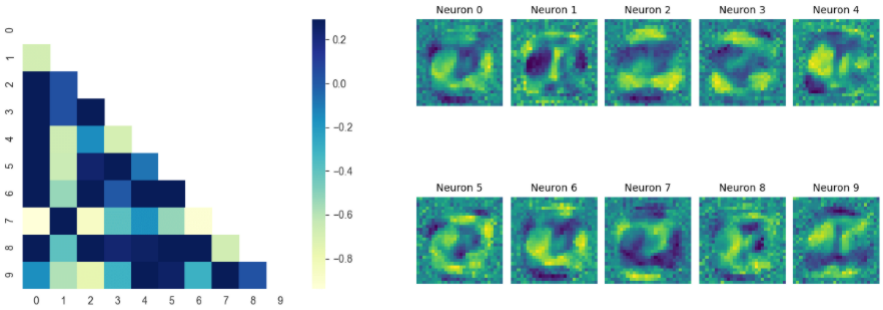


Figure 10. Left: cosine similarity between features. Right: visual features. Network trained with 1 epoch, 128 hidden neurons, 3 hidden layer. test accuracy: 83.05