Addressing the Need for Raw-Valued Dataset Exploration in Neural Network Visualization

Filip Dabek[*]
filip.j.dabek.ctr@mail.mil

Peter Hoover[*]
peter.j.hoover2.ctr@mail.mil

Jesus J. Caban[*]
jesus.j.caban.civ@mail.mil

Abstract

Over the last decade, through the increased computational power and availability of large scale datasets, deep neural networks have become an instrumental tool for a wide range of machine learning tasks. While these networks continue to provide breakthroughs across many domains and a large amount of research has focused on visualizing and understanding these networks, we recognize that much of the focus has been concentrated on networks that handle images or text and the need to explore raw-valued datasets is increasingly important. For example, in the medical domain interpretability is of high importance and through visualization it would be possible to not only embed clinical knowledge to speed up/skip parts of the training process, but clinicians could also use their expertise to fine tune a trained network. Therefore, in this paper we identify this need, show some early exploration, and state the future work that should be focused on in this area of raw-valued datasets.

1 Introduction

Over the last decade, through the increased computational power and availability of large scale datasets, deep neural networks have become an instrumental tool for a wide range of machine learning tasks. While these networks continue to provide breakthroughs across many domains, they continue to be considered black boxes due to their complexity and their near impossible interpretability. As a first step in unraveling the effectiveness of these black boxes, research in recent years has identified the layer-wise structure of convolutional neural networks (CNNs) [11] as well as the images that the network is learning behind the scenes [10, 15]. Even though this work has enhanced our knowledge into these networks, much of the focus has been concentrated on networks that handle images or text. While these are two important mediums to evaluate, we believe that it is paramount for the community to also place our focus on raw valued datasets that cannot be easily visualized.

Therefore, in this paper we discuss some of the previous work in visualizing neural networks and state our position on the need to explore raw-valued datasets for this task. To present this position, we will show the early explorations that we have made in this realm in order to foster discussion and to show some initial work in this challenging task.

2 Background

In the past couple of years, a vast amount of literature has focused on uncovering the hidden mysteries embedded within deep neural networks [20, 3, 19, 14]. With the breakthrough in convolutional

[*]National Intrepid Center of Excellence, Walter Reed National Military Medical Center, Bethesda, MD

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

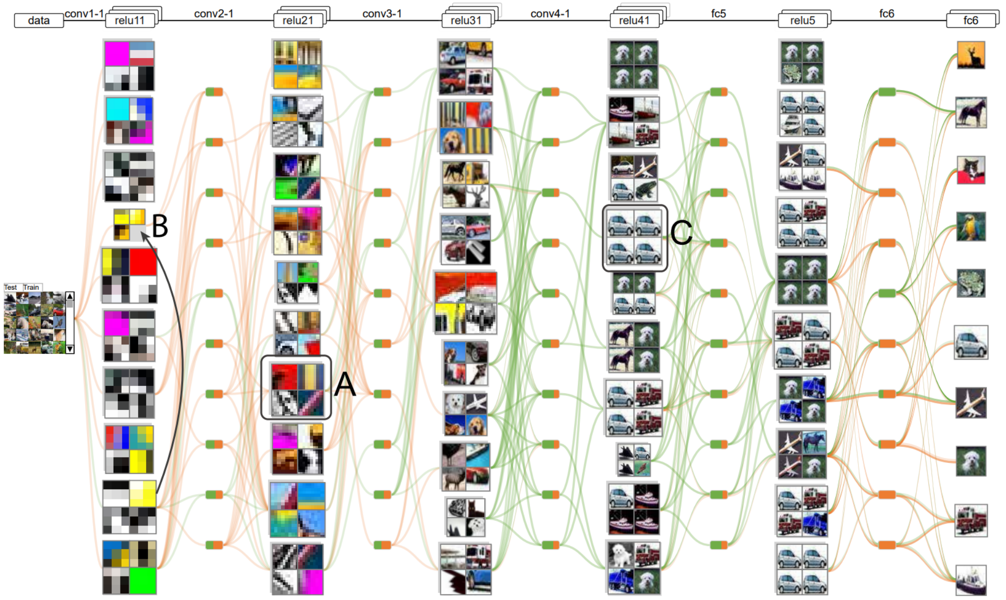


Figure 1: Previous work, titled CNNVis, in which the learned features and concepts of each layer and its respective neurons is displayed in a visualization.

neural networks (CNNs), the activation maps of large networks such as AlexNet were visualized to understand which nodes had been activated for images [10]. Beyond that, Google has attempted to reverse deep CNNs in an attempt to visualize the learned weights by the networks in what has been referred to as “DeepDream” [15].

Most recently at the last two IEEE Visualization Conferences (2016 & 2017) several papers have attempted to provide tools to inspect deep neural networks: ActiVis, developed by Facebook, supports instance level and node level inspection in which diagnosing where a network made an incorrect decision can be identified [8]; Alsallakh et al. visualized CNNs and identified performance improvements based on the class hierarchy [1]; and CNNVis built on the rich existing literature on understanding CNNs and built a representation shown in Figure 1 [11].

3 Position

While the existing research has been able to provide explanations in CNNs, there exists a need to explore raw-valued datasets that are difficult to visualize. This is especially true for the medical domain that greatly relies on being able to interpret and explain how a decision is made. Much of the current clinical informatics research focuses on utilizing decision trees due to the high interpretability associated with the method [7, 9, 17]. However, decision trees are severely limited in their performance on large datasets and complex tasks, but compared to neural networks, support vector machines, random forests, and many other methods; they are very easy to understand for a domain that heavily relies on explanations.

However, while decision trees have been utilized greatly in the medical literature, there has been research utilizing deep neural networks that has shown breakthrough performance on clinical tasks using raw-valued datasets, such as the work in Deep Patient [13]. Understanding the network that was trained by this approach and understanding how clinical decisions are arrived at would reveal a lot. Ultimately, we envision an approach that is able to build a visualization similar to that of CNNVis (Figure 1) where the progressive learning of concepts can be easily seen and culminated into a decision made by the network. Furthermore, by visualizing these networks it would be possible to: (i) embed clinical knowledge in order to speed up/skip parts of the training process and (ii) utilize clinicians’ expertise to fine tune a trained network to remove incorrect decisions that are made.

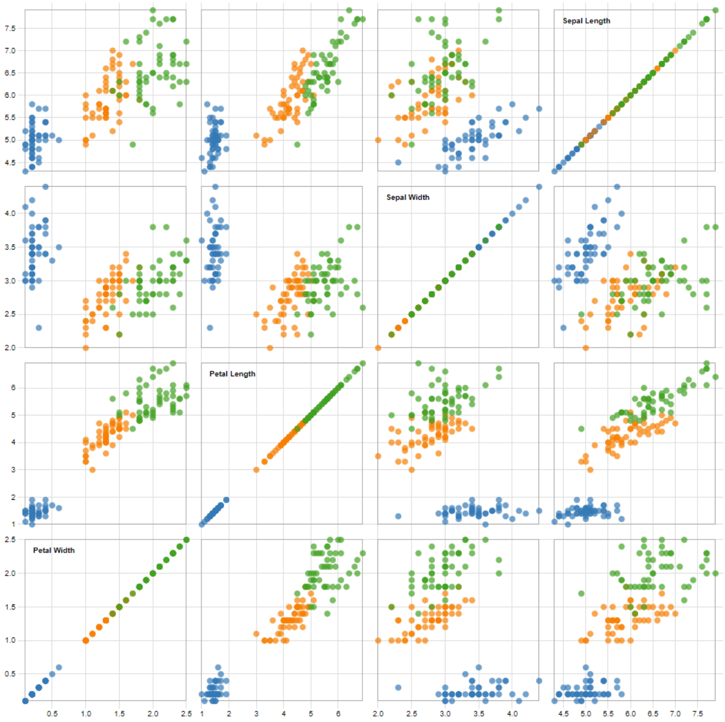


Figure 2: A scatterplot matrix of the iris dataset where each color is a different class of flower. This matrix shows the different properties of the classes and how they are related.

While a lot of work exists for visualizing and understanding deep neural networks with raw-valued datasets, we will present some of the initial explorations that we have taken with a basic dataset. With this, we hope to foster discussion and simultaneously move to visualizing some of the large neural networks that we have trained on raw-valued medical data.

4 Visualizing Iris

4.1 Dataset

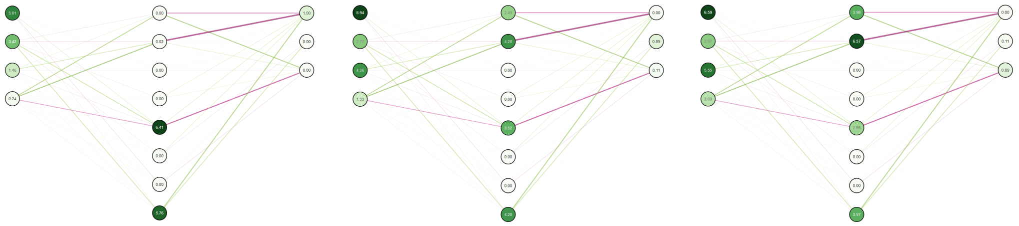
To provide an easy to understand example of visualizing and understanding a neural network, we will present our initial explorations into a network trained on the iris dataset [6, 2]. The iris dataset is commonly used as an introduction to machine learning due to its simpleness and relatively easy separability. A visualization, in the form of a scatterplot matrix [2], is shown in Figure 2 where each class is mapped to a specific color. Inspecting this matrix, it is possible to see that the blue class is vastly different from the other two classes; while the orange and green classes are very similar and even can be difficult to differentiate between at certain points.

4.2 Network Design

Using the neural network library, Keras [4], we constructed a network that contained a single hidden layer of 8 nodes and an output layer of 3 nodes, where each output node corresponded to a class within the dataset. For activation functions, the hidden layer utilized ReLU [16, 18] and the output layer utilized softmax [12]. These functions were chosen due to their simplicity and relatively easy interpretability as ReLU is a simple linear function.

Next, we trained the network with the categorical cross entropy loss function for 100 epochs on the entire dataset. Running the trained network against the same dataset, it achieved a 97% accuracy in predicting the correct class.

[2] `https://bl.ocks.org/mbostock/4063663`



(a) Class 1

(b) Class 2

(c) Class 3

Figure 3: The average node activations across the network for each class within the network.

4.3 Per Class Networks

Using the trained network, we both saved the architecture and its weights as well as ran each instance through the network to log the activation at each node. These logs of the network at each instance would allow us to generate visualizations of the network and understand how the data flows through the network.

The first analysis step that we took was to aggregate the activations of the instances for each class and generate a visualization based on the average values for each node. The results of this can be seen in Figure 3. Analyzing the hidden layer of each class, an interesting insight into the data can quickly be obtained: class 1 primarily activates two nodes while classes 2 and 3 activate a series of different nodes. This instantly reveals to us a similar insight that we gained from the scatterplot matrix in Figure 2 that is widely known for this dataset: one of the classes is vastly different than the other two.

4.4 Overall Network

To gain a better overview of the entire network, we attempted to create a visualization for the network that would encompass all three classes. We accomplished this by utilizing the average activations from the per class networks building a pie chart at each node location indicating the relative activation of each class for a particular node. This can be seen in Figure 4 where each class is mapped to a color: class 1 (blue), class 2 (orange), class 3 (green). Additionally, the weights between nodes are drawn with both color and size mappings: red to white to green where a dark red indicates a negative weight, white indicates a zero weight, and a dark green indicates a positive weight. For the size mapping, the absolute value of the weight is utilized to determine the width of the line: $width = |weight| * 5$, causing for weights close to zero to be near non-existent.

The first thing that is possible to notice with this network are the activations at the output layer: class 1 is always predicted blue while classes 2 and 3 have some minimal alternate predictions mixed in. Additionally, it is easy to notice that a network of 8 nodes was too large for this task as there exists 4 nodes within the network that have an activation of 0 and hardly any weight influence to future layers of the network.

Moving past these simple observations, looking at the hidden layer we see similar trends to those that were evident within the per class networks: blue is only activated for nodes 5 and 8 within the hidden layer and not for any others. Furthermore, the weights between the nodes reveal why these nodes are limited to only certain classes. Nodes 1 and 2, because they do not activate at all for the blue class, they have a very negative weight to the blue output node. This can be explained by the fact that the blue output node can determine that the instance is not blue if these two nodes are activated.

Moving on to the two blue activated nodes in the hidden layer, node 5 has a high negative weight to the green class while node 8 has a relatively high positive weight to the blue output node. With this, we can see how these nodes influence the output.

The weights between the hidden and output layer explain what each node learned for the output, the weights between the input and hidden layer reveal what these nodes are learning. Looking at the 4th input, petal width, we can see that it is heavily positively weighted towards nodes 1 and 2, while it is very negatively weighted towards node 5. This indicates that the blue class cannot be explained by the petal width and that both the green and orange classes can.

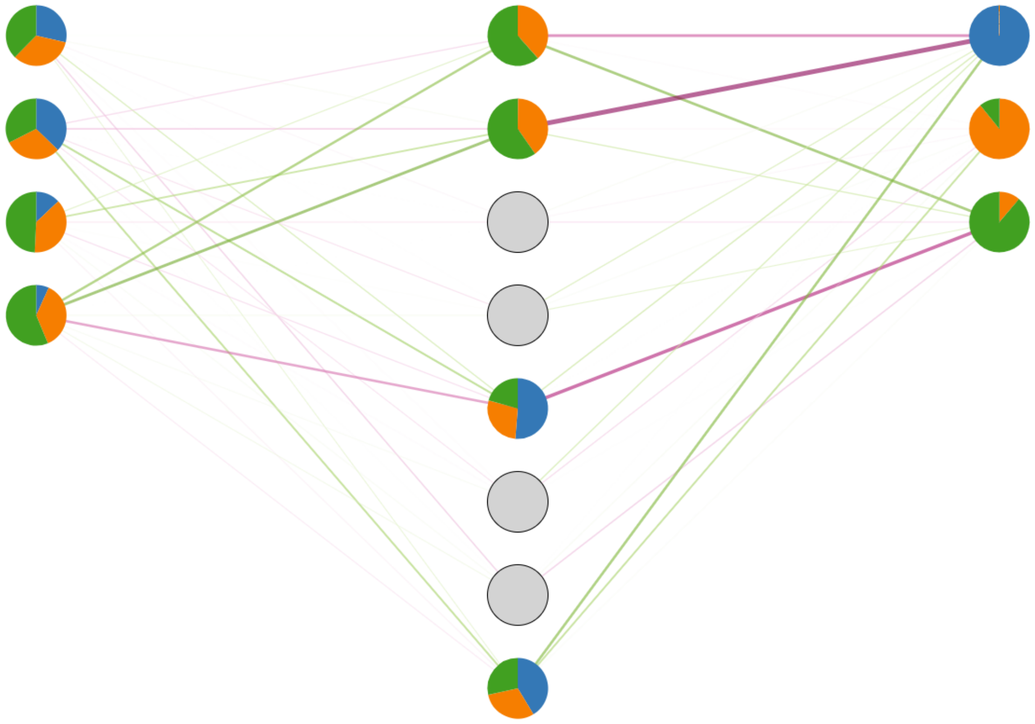


Figure 4: A visualization of the entire iris neural network in which each node contains a plot showing which of the three classes (blue, orange, and green) activated the particular node.

4.5 Feature Influence Network

While the overall network in Figure 4 explains the connection of the input layer on the hidden layer, understanding the exact influence of each flower feature is another important task. Figure 5 displays a visualization of the result of varying each input and the effect/change that it has on each node's activation. The features of the iris dataset displayed in this visualizations consists of: sepal length (blue), sepal width (orange), petal length (green), petal width (red). To construct this visualization, the minimum and maximum were taken for each feature and then a pass through was made for the network for each value in between the two extremes, while the other features were remained constant at 0.

With this influence graph we can see that the nodes that had been considered "useless" and a zero node, actually does receive influence from several features. Looking back at the nodes described in the overall network, it is possible to see that nodes 1 and 2 mainly are updated with the green and red features, while nodes 5 and 8 are mainly influenced by the blue and orange features.

In the output layer, it is possible to see that the blue and orange features heavily influence the first class (that was the blue class in previous visualizations). Beyond the blue and orange features, the green and red features are curved in each of the output nodes and it can be seen that there exists a range for each of these features where the node's activation is maximized. Using this information, we can infer the range of the values for both of these features where the two classes can be separated.

4.6 Clustering Network Nodes

While these visualizations have provided insights into the Iris dataset, we seek to explore datasets that are much more complex and networks that are much larger. To be able to accomplish this, visualizing the hundreds, thousands, or millions of nodes within a deep neural network would simply be infeasible and impossible. Therefore, we explore a potential solution to simplifying the Iris network in Figure 6.

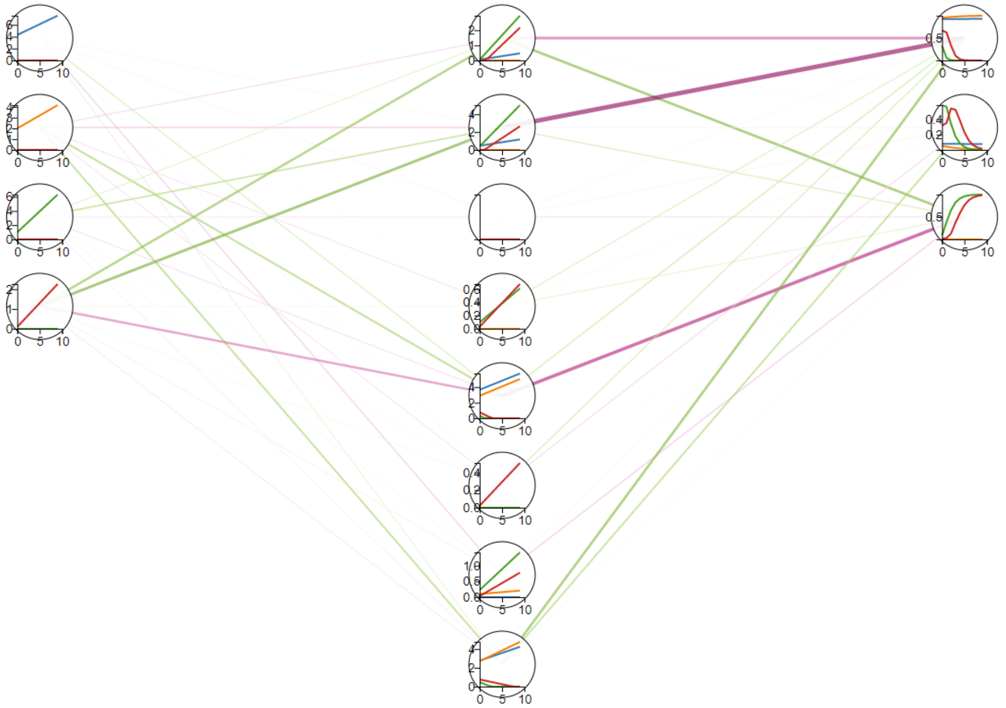


Figure 5: The influence of the change in each input feature on the activation of each node in the neural network.

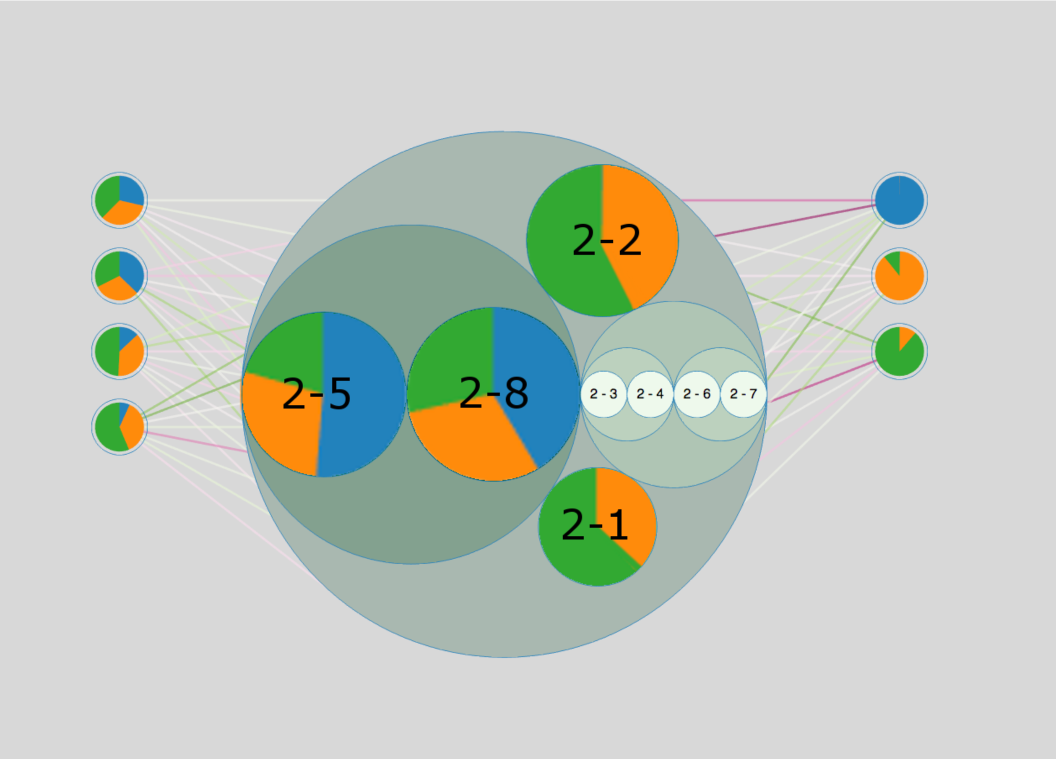


Figure 6: An example of clustering nodes in order to simplify a hidden layer, which would be useful for future work in which layers can contain large amounts of nodes.

In this network, the hidden layer was clustered using the average activations for each class and utilizing hierarchical clustering [5] to achieve a hierarchical representation of the layer. For this figure, the nodes in the hidden layer are labeled with numbers indicating the node's position in the original overall network from Figure 4. We can see how the nodes with zero activations are clustered together while the remaining nodes are clustered according to the patterns that we had identified earlier: 5 and 8 are the only nodes with the blue class. While, this is just an example method, we seek to explore these types of methods further to be able to achieve an explanation of a large hidden layer.

5 Discussion

Through the position stated in this paper, the need for understanding deep neural networks for raw valued datasets has been described. Through the example visualizations and exploration that we have performed, it is clear that analyzing the nodes within the network is important for understanding how decisions are made and could be applied to real-world examples, such as medical diagnoses.

For the future, we encourage and also seek to build on this exploration by aiming towards building a network visualization, similar to that of CNNVis in Figure 1, where a large network can be summarized into the concepts learned by each set of nodes and layer. Furthermore, we also look forward to utilizing this network visualization to be able to evaluate the training process of the network and being able to embed clinical knowledge into the training process to: (1) speed up the learning and (2) optimize parts of the network that make incorrect decisions.

References

[1] B. Alsallakh, A. Jourabloo, M. Ye, X. Liu, and L. Ren. Do convolutional neural networks learn class hierarchy? *IEEE Transactions on Visualization and Computer Graphics*, 2017.

[2] E. Anderson. The species problem in iris. *Annals of the Missouri Botanical Garden*, 23(3):457–509, 1936.

[3] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

[4] F. Chollet et al. Keras. `https://github.com/fchollet/keras`, 2015.

[5] F. Corpet. Multiple sequence alignment with hierarchical clustering. *Nucleic acids research*, 16(22):10881–10890, 1988.

[6] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of human genetics*, 7(2):179–188, 1936.

[7] J. M. Jerez-Aragonés, J. A. Gómez-Ruiz, G. Ramos-Jiménez, J. Muñoz-Pérez, and E. Alba-Conejo. A combined neural network and decision trees model for prognosis of breast cancer relapse. *Artificial intelligence in medicine*, 27(1):45–63, 2003.

[8] M. Kahng, P. Andrews, A. Kalro, and D. H. Chau. Activis: Visual exploration of industry-scale deep neural network models. *arXiv preprint arXiv:1704.01942*, 2017.

[9] M. A. Karaolis, J. A. Moutiris, D. Hadjipanayi, and C. S. Pattichis. Assessment of the risk factors of coronary heart events based on data mining with decision trees. *IEEE Transactions on information technology in biomedicine*, 14(3):559–566, 2010.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[11] M. Liu, J. Shi, Z. Li, C. Li, J. Zhu, and S. Liu. Towards better analysis of deep convolutional neural networks. *IEEE transactions on visualization and computer graphics*, 23(1):91–100, 2017.

[12] P. McCullagh. Generalized linear models. *European Journal of Operational Research*, 16(3):285–292, 1984.

[13] R. Miotto, L. Li, B. A. Kidd, and J. T. Dudley. Deep patient: An unsupervised representation to predict the future of patients from the electronic health records. *Scientific reports*, 6:26094, 2016.

[14] G. Montavon, W. Samek, and K.-R. Müller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 2017.

[15] A. Mordvintsev, C. Olah, and M. Tyka. Inceptionism: Going deeper into neural networks. *Google Research Blog. Retrieved June*, 20:14, 2015.

[16] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[17] S. Rizoli, A. Petersen, E. Bulger, R. Coimbra, J. D. Kerby, J. Minei, L. Morrison, A. Nathens, M. Schreiber, and A. L. de Oliveira Manoel. Early prediction of outcome after severe traumatic brain injury: a simple and practical model. *BMC emergency medicine*, 16(1):32, 2016.

[18] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2892–2900, 2015.

[19] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.

[20] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.