Topology as an explanatory tool for deep neural networks

Clara I. López González
Universidad Complutense de Madrid

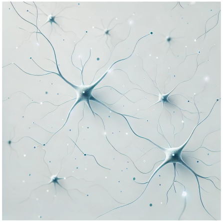
Seminario de TDA
UAM-CUNEF

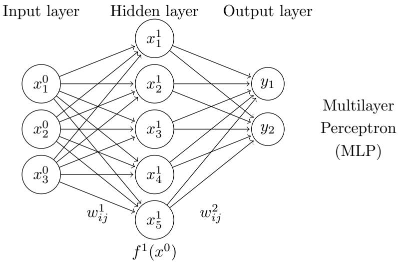
Noviembre 2024

UNIVERSIDAD COMPLUTENSE MADRID

1. Deep Learning
2. TDA
3. Analyzing MLPs
4. Analyzing CNNs
5. Explainable Artificial Intelligence

FEEDFORWARD NEURAL NETWORKS



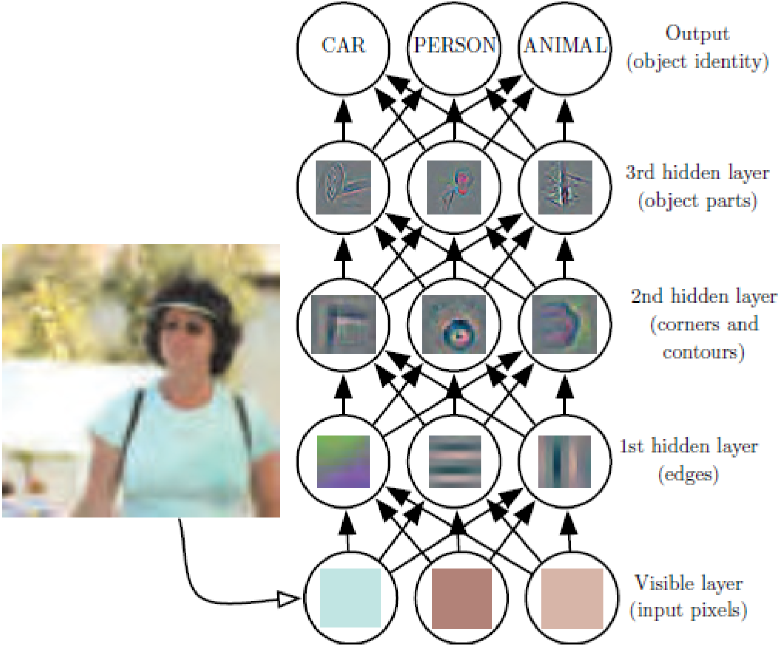


$$y : \mathbb{R}^{K_0} \xrightarrow{f^1} \mathbb{R}^{K_1} \xrightarrow{f^2} \cdots \xrightarrow{f^{N-1}} \mathbb{R}^{K_{N-1}} \xrightarrow{f^N} \mathbb{R}^{K_N}$$

$$f^n = h^n \circ g^n \text{ with } \begin{cases} g^n : \mathbb{R}^{K_{n-1}} \to \mathbb{R}^{K_n}, & g^n(x) = W^n x + b^n, \\ h^n : \mathbb{R}^{K_n} \to \mathbb{R}^{K_n}, & h^n(x) = (h^n(x_1), \dots, h^n(x_{K_n}))^\top, \end{cases}$$

where:

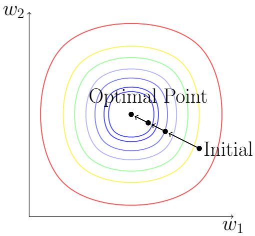
- N is the number of layers and K_n is the number of neurons.
- $W^n = (w_{ij}^n) \in \mathbb{R}^{K_n \times K_{n-1}}$ and b^n are the weights and biases.
- h^n is a non-linear activation function: ReLU, sigmoid, softmax, etc.


CAR
PERSON
ANIMAL
Output
(object identity)
3rd hidden layer
(object parts)
2nd hidden layer
(corners and
contours)
1st hidden layer
(edges)
Visible layer
(input pixels)

1. The loss function $\mathcal{L}$ quantifies the goodness of the model to perform a task.
   - Regression: $\mathcal{L}(y, \mathcal{D}) = \frac{1}{2} \sum_{m=1}^{M} \|y_m - t_m\|^2$.
   - Classification: $\mathcal{L}(y, \mathcal{D}) = -\sum_{m=1}^{M} \sum_{n=1}^{K_N} t_{ml} \log(y_{ml})$.

1. The loss function $\mathcal{L}$ quantifies the goodness of the model to perform a task.
2. The dataset $\mathcal{D}$ is divided into training $\mathcal{D}_0$ and test $\mathcal{D}_1$ sets. Using gradient descent, the weights are updated to minimize the loss:

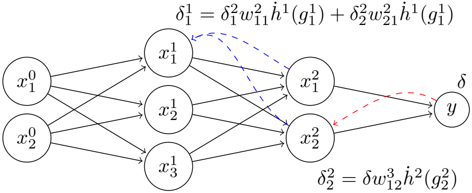
$$W^n - \eta \nabla \mathcal{L}(W^n) \to W^n, \quad b^n - \eta \nabla \mathcal{L}(b^n) \to b^n,$$



1. The loss function $\mathcal{L}$ quantifies the goodness of the model to perform a task.
2. The dataset $\mathcal{D}$ is divided into training $\mathcal{D}_0$ and test $\mathcal{D}_1$ sets. Using gradient descent, the weights are updated to minimize the loss:

$$W^n - \eta \nabla \mathcal{L}(W^n) \to W^n, \quad b^n - \eta \nabla \mathcal{L}(b^n) \to b^n,$$

3. The gradient is computed with the backpropagation algorithm.

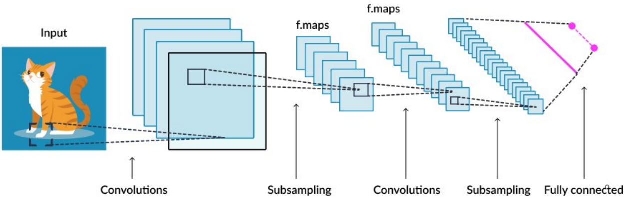


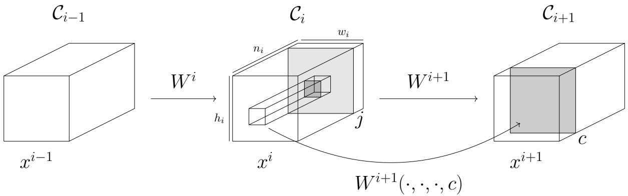
1. The loss function $\mathcal{L}$ quantifies the goodness of the model to perform a task.
2. The dataset $\mathcal{D}$ is divided into training $\mathcal{D}_0$ and test $\mathcal{D}_1$ sets. Using gradient descent, the weights are updated to minimize the loss:

$$W^n - \eta \nabla \mathcal{L}(W^n) \to W^n, \quad b^n - \eta \nabla \mathcal{L}(b^n) \to b^n,$$

3. The gradient is computed with the backpropagation algorithm.
4. The performance of the model is evaluated with $\mathcal{D}_1$:
   - Accuracy, Intersection over Union (IoU), etc.

CONVOLUTIONAL NEURAL NEWORKS (CNNS)

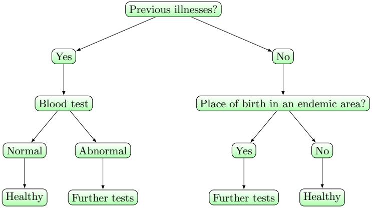


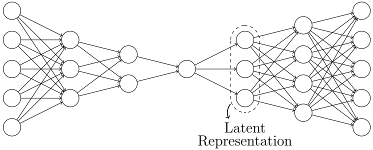


$$x^{i+1}(s,t,c) = \sum_{m,l,r} x^i(s+m,t+l,r) \times W^{i+1}(m,l,r,c) + b^{i+1}(c).$$

DEEP LEARNING

BLACK BOX





In neural networks...

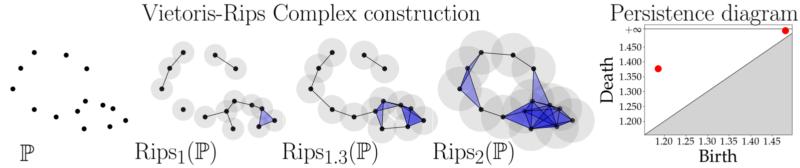
The output of each layer is an abstraction of the input, called **latent representation**, and constitutes a reduced and meaningful representation of the data.

Idea

What if we analyze the topology of the latent representations?

1. Deep Learning
2. TDA
3. Analyzing MLPs
4. Analyzing CNNs
5. Explainable Artificial Intelligence

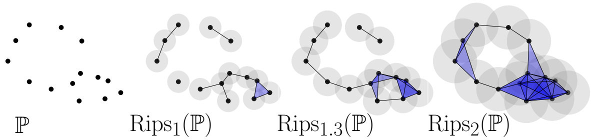
TOPOLOGICAL DATA ANALYSIS (TDA)



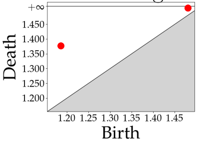
$$
\begin{tikzcd}
\mathrm{Rips}_{\alpha_0}(\mathbb{P}) \arrow[r] \arrow[d, "H_1"] & \mathrm{Rips}_{\alpha_1}(\mathbb{P}) \arrow[r] \arrow[d, "H_1"] & \cdots \arrow[r] & \mathrm{Rips}_{\alpha_n}(\mathbb{P}) \arrow[d, "H_1"] \\
H_1(\mathrm{Rips}_{\alpha_0}(\mathbb{P})) \arrow[r] & H_1(\mathrm{Rips}_{\alpha_1}(\mathbb{P})) \arrow[r] & \cdots \arrow[r] & H_1(\mathrm{Rips}_{\alpha_n}(\mathbb{P}))
\end{tikzcd}
\qquad \alpha_0 < \cdots < \alpha_n
$$

TOPOLOGICAL DATA ANALYSIS (TDA)

Vietoris-Rips Complex construction

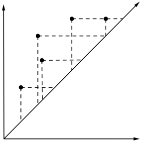


Persistence diagram

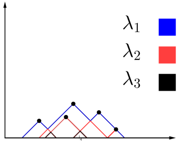


$$
\begin{tikzcd}
\mathrm{Rips}_{\alpha_0}(\mathbb{P}) \arrow[r] \arrow[d, "H_1"] & \mathrm{Rips}_{\alpha_1}(\mathbb{P}) \arrow[r] \arrow[d, "H_1"] & \cdots \arrow[r] & \mathrm{Rips}_{\alpha_n}(\mathbb{P}) \arrow[d, "H_1"] \\
H_1(\mathrm{Rips}_{\alpha_0}(\mathbb{P})) \arrow[r] & H_1(\mathrm{Rips}_{\alpha_1}(\mathbb{P})) \arrow[r] & \cdots \arrow[r] & H_1(\mathrm{Rips}_{\alpha_n}(\mathbb{P}))
\end{tikzcd}
\qquad \alpha_0 < \cdots < \alpha_n
$$

Persistence diagram



Persistence landscape



$\{\lambda_k\}_{k\in\mathbb{N}}$,

$\lambda_k(t) = \mathrm{kmax}_{p\in PD}\Lambda_p(t)$,

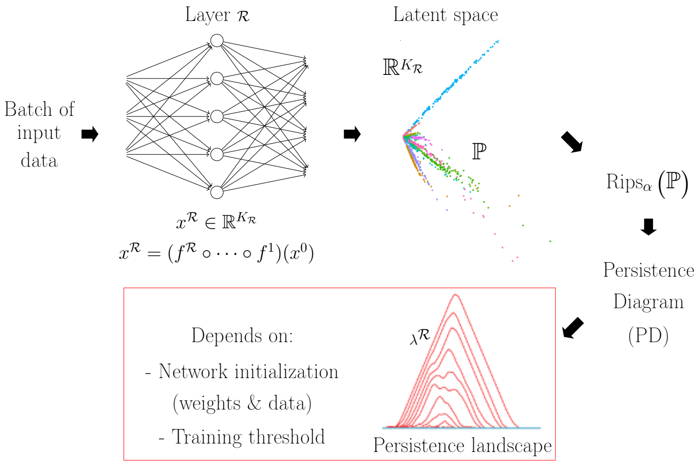
$\Lambda_p(t) = \max\{0, \min\{t-b, d-t\}\}$,

with $t \in \mathbb{R}$, $p = (b, d) \in PD$.

1. Deep Learning
2. TDA
3. Analyzing MLPs
4. Analyzing CNNs
5. Explainable Artificial Intelligence

ACTIVATION LANDSCAPES $\lambda^{\mathcal{R}}$

MLPS

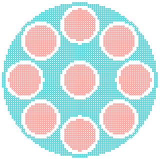


Matthew Wheeler et al. "Activation landscapes as a topological summary of neural network performance", IEEE Int. Conf. on Big Data, 2021.

EXPERIMENTS

SYNTHETIC DATA

- Two categoris: C_1 with 9 disks and C_2 as the complement.



- 100 MLPs trained with different initializations.
- Compute[1] the average activation landscape for class C_2, each layer and selection of training threshold $s \in S$,

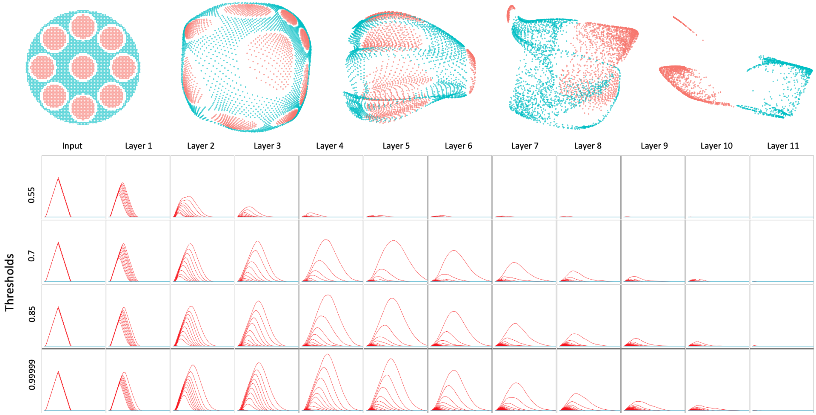
$$\{\lambda^0[s], \cdots, \lambda^N[s]\}_{s \in S}, \quad \lambda^{\mathcal{R}}[s] = \frac{1}{100} \sum_{j=1}^{100} \lambda^{\mathcal{R}}[s,j] \quad \mathcal{R} \in \{0, \dots, N\},$$

where $\lambda^{\mathcal{R}}[s,j]$ corresponds to the j-th MLP and threshold s.

[1] `https://github.com/jjbouza/nn-activation-landscapes`

EXPERIMENTS

SYNTHETIC DATA

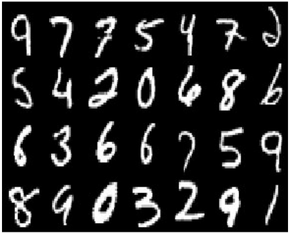


- The first layer detects 9 holes.
- The activation landscapes of a fully trained network accentuate the most significant topological features of the activations.

EXPERIMENTS

REAL DATA

- MNIST dataset.

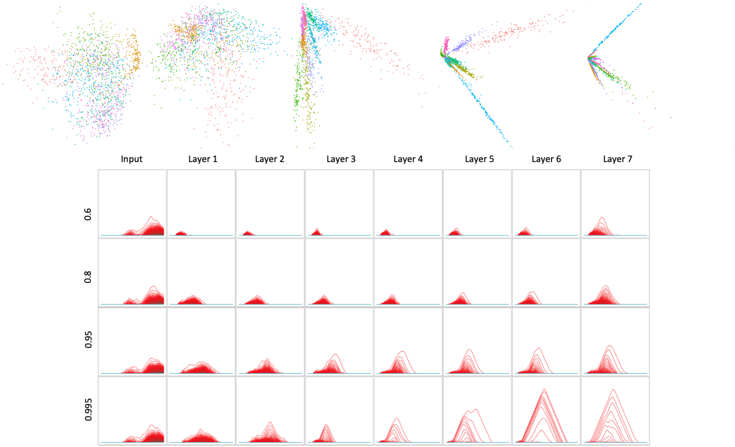


- 10 MLPs trained with different initializations.
- Compute the average activation landscape for each layer and selection of training threshold over choices of trained networks and batch of input data.

Y. LeCun et al. "Gradient-based learning applied to document recognition." Proceedings of the IEEE, 1998.

EXPERIMENTS

REAL DATA



- The activation landscape of the last layer detects the clustering by classes in 1D subspaces.

EXPERIMENTS

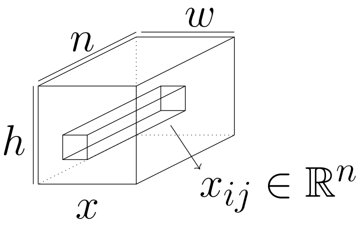
Activation landscapes...

- Provide a complete summary of the persistent homology of the activations in each layer.
- Illuminate aspects of training ddynamics.
- Show that topological complexity increases with training...
- but does not decrease monotonically with each layer, contradicting previous observations.[3]

[3] Gregoty Naitzat et al. "Topology of deep neural networks". J. Mach. Learn. Res., 2020.

1. Deep Learning
2. TDA
3. Analyzing MLPs
4. Analyzing CNNs
5. Explainable Artificial Intelligence

CNNS



$x = x' * W, \quad W \in \mathbb{R}^{k \times k \times n' \times n}$

$x \in \mathbb{R}^{h \times w \times n}$ is a latent representation of the input image which codes contextual information:

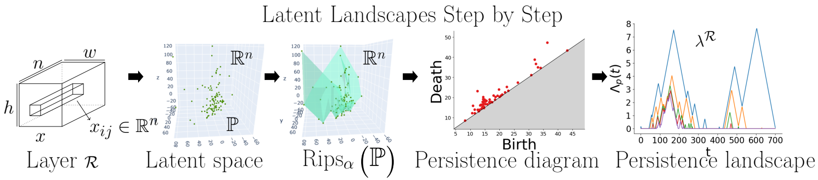
- Each filter $W(\cdot, \cdot, \cdot, c) \in \mathbb{R}^{k \times k \times n'}$ detects the presence of a feature by regions (a pixel and its neighbors).
- Each unit $x(i, j, c)$ encodes the value of that feature in a certain area.

By considering all filters, channel vectors $x_{ij} := x(i, j, \cdot)$ are latent representations of a region that encodes contextual information.

Clara I. López-González et al. "Analyzing and interpreting CNNs using latent space topology", Neurocomputing, 2024.

LATENT LANDSCAPES

CNNS



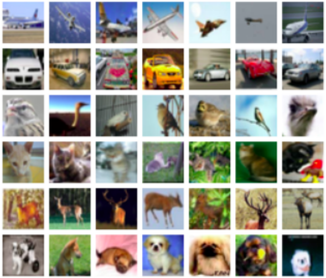
- Close x_{ij} = areas with similar contextual information.
- Connections between x_{ij} in the Rips $\Rightarrow$ regions with similarities = same feature's category.
- Holes created/destroyed $\Rightarrow$ categories distinguished $\Rightarrow$ richer encoded information.
- Trained NN $\Rightarrow$ code varied features = interesting non trivial topology.
- Homogeneous activations $\Rightarrow$ trivial topology $\Rightarrow$ poorer performance.

---

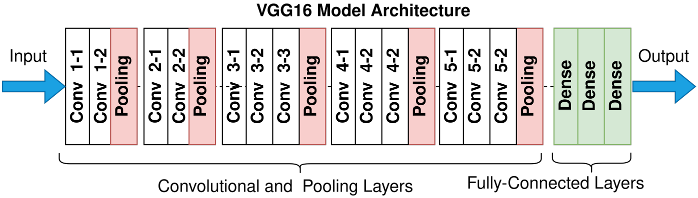
https://github.com/claraisl/cnn-latent-landscapes

EXPERIMENTS

EVOLUTION OF INFORMATION



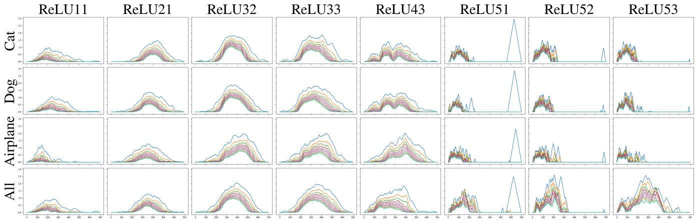
Classification of CIFAR-10 with VGG-16.

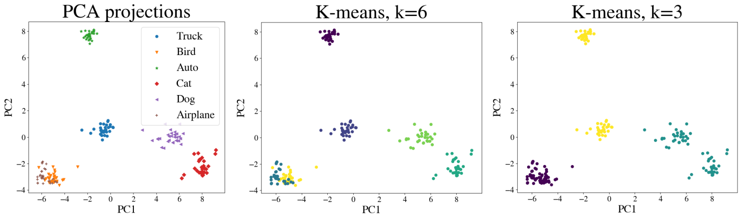


Alex Krizhevsky, "Learning multiple layers of features from tiny images, 2009.

EXPERIMENTS

EVOLUTION OF INFORMATION

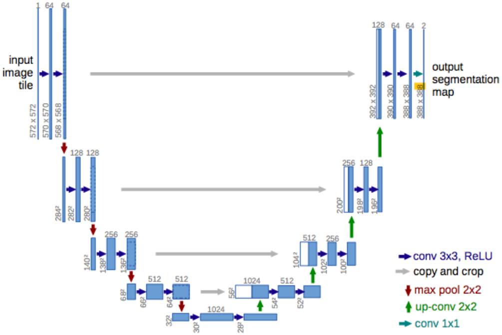




- Different classes could be distinguished by the last layer latent landscape, even detecting similarities between them.

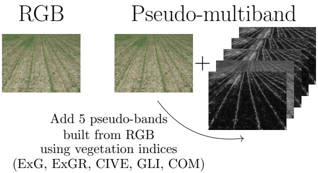
EXPERIMENTS

EXPLAINING PERFORMANCE



Semantic segmentation with U-Net on Crop Row Benchmark Dataset:

- RGB images.
- Pseudo-multiband images (M*x* models).

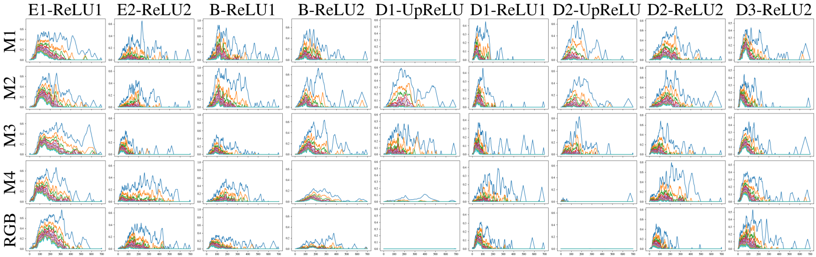


Ivan Vidovic et al. "Crop row detection by global energy minimization", Ptrn. Recognit., 2016.

Olaf Ronneberger et al. "U-net: Convolutional networks for biomedical image segmentation." Int. Conf. MICCAI, 2015.

EXPERIMENTS

EXPLAINING PERFORMANCE



- The UpReLU layers before the skip connections do not store relevant information.
- RGB is neither capable of capturing complex and diverse features, nor of benefiting from the skip connections.
- The difference within M*x* is explained by comparing how they codify the information provided.

1. Deep Learning
2. TDA
3. Analyzing MLPs
4. Analyzing CNNs
5. Explainable Artificial Intelligence

... EXPLAINABLE ARTIFICIAL INTELLIGENCE

This is known as explainability and falls within eXplainable Artificial Intelligence (XAI).

> XAI is a collection of methods that allow us to explain, interpret and understand the decision and predictions made by an AI model.

XAI is a recent and relevant field, given the use of black box algorithms in areas like medicine:

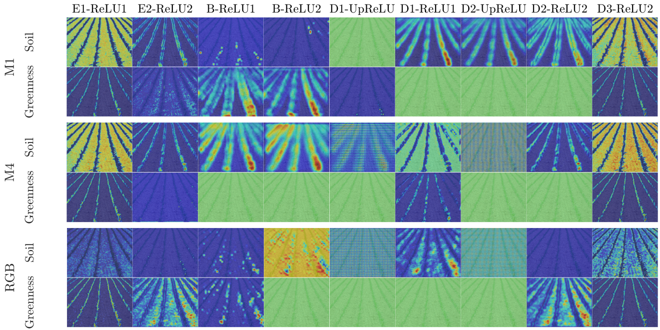
- It is important to ensure that the right decisions are being made correctly

OTHER METHODS

GRAD-CAM

Let I be the input image and $S^\beta(I)$ the output of the network for class β. Grad-CAM is computed as:

$$\mathrm{ReLU}\left(\sum_c \alpha_c^\beta x(\cdot,\cdot,c)\right), \quad \alpha_c^\beta = \frac{1}{hw}\sum_{i,j}\frac{\partial S^\beta(I)}{\partial x(i,j,c)}.$$

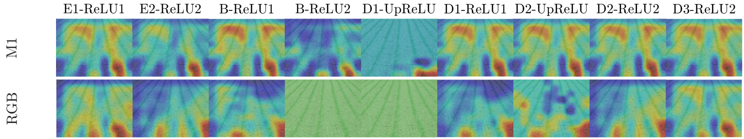


Ramprasaath R. Selvaraju et al. "Grad-CAM: Visual explanations from deep networks via gradient-based localization", ICCV, 2017.

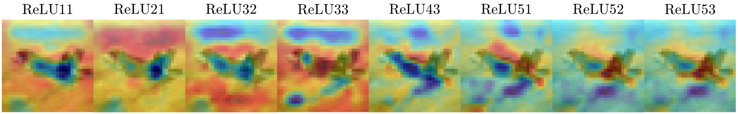
OTHER METHODS

OCCLUSION SENSITIVITY

Consists in occluding regions of the input image and measuring the change in the output.



Crop Row Dataset.



CIFAR-10, airplane.

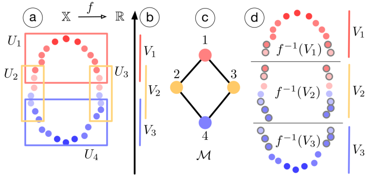
Matthew D. Zeiler et al. "Visualizing and understanding convolutional networks", ECCV, 2014.

OTHER METHODS

TOPOACT

Visualization that show the shape of the activation space and the relationships within a layer:

- Point cloud of channel vectors obtained by randomly sampling a single spatial activation from each input.
- Mapper construction built from these point cloud to summarize clusters and cluster relations behind neuron activations.



- Feature visualization applied to channel vectors and averaged channel vectors per cluster.

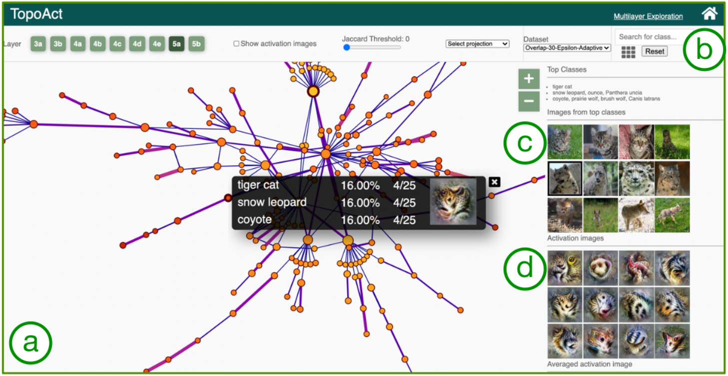
Archit Rathore et al. "TopoAct: Visually exploring the shape of activations in deep learning", Computer Graphics, 2021.

OTHER METHODS

TOPOACT

Captures topological structures, such as branches (separations among classes) or loops (different aspects of the same object), in the space of activations:

`https://tdavislab.github.io/TopoAct/`

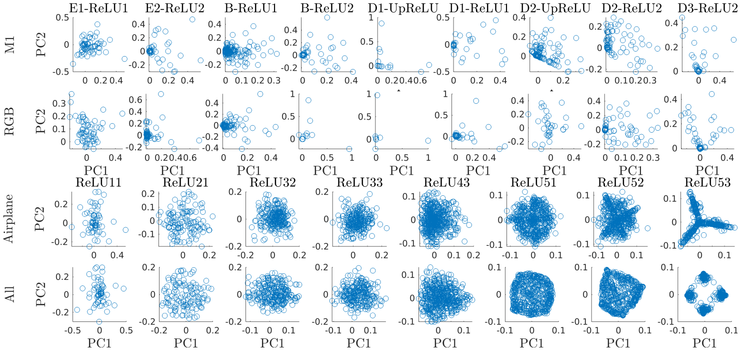


OTHER METHODS

PCA

Instead of computing latent landscapes, what if we just perform PCA on the latent space?

- Knowledge about how the coded features are distributed and how varied they are is lost.



2D PCA projections of the channel vectors.

Thank you
for your attention!

Feel free to get in touch: `claraisl@ucm.es`