THE PREIMAGE OF RECTIFIER NETWORK ACTIVITIES

Stefan Carlsson, Hossein Azizpour, Ali Razavian, Josephine Sullivan and Kevin Smith
School of Computer Science and Communication
KTH
Stockholm, Sweden
email stefanc@kth.se

ABSTRACT

We give a procedure for explicitly computing the complete preimage of activities of a layer in a rectifier network with fully connected layers, from knowledge of the weights in the network. The most general characterisation of preimages is as piecewise linear manifolds in the input space with possibly multiple branches. This work therefore complements previous demonstrations of preimages obtained by heuristic optimisation and regularization algorithms Mahendran & Vedaldi (2015; 2016) We are presently empirically evaluating the procedure and it's ability to extract complete preimages as well as the general structure of preimage manifolds.

1 PREIMAGES OF FULLY CONNECTED RECTIFIER NETWORKS

We will investigate preimages for fully connected multi layer networks where the mapping at layer (l) is described by the matrix W and bias vector b. This is followed by a rectifier linear unit (ReLU) that maps all negative components of the output vector to 0. We can then write for the mapping between successive layers:

$$x^{(l+1)} = [Wx^{(l)} + b\,]_+$$

where $[x]_+$ denotes the ReLU function.

For each element $x^{(l+1)}$ the preimage set of this mapping will be the set:

$$P(x^{(l+1)}) \;=\; \{x : x^{l+1} = \;[Wx + b\,]_+\}$$

which can be specified in more detail as:

$$P(x^{(l+1)}) \;=\; \{x : w_i^T x + b_i = x_i^{l+1} \;\; \forall x_i^{l+1} > 0, \;\; w_i^T x + b_i \leq 0 \;\; \forall x_i^{l+1} = 0\}$$

Let $i_1, i_2, \ldots i_p$ be the indices of the components of x^{l+1} that are $= 0$ and $j_1, j_2, \ldots j_q$ those that are > 0. If x is in n dimensional space we have $p + q = n$ and:

$$w_{i_1}^T x^{(l)} + b_{i_1} \leq 0, \quad w_{i_2}^T x^{(l)} + b_{i_2} \leq 0 \quad \ldots \quad w_{i_p}^T x^{(l)} + b_{i_p} \leq 0 \tag{1}$$

$$w_{j_1}^T x^{(l)} + b_{j_1} > 0, \quad w_{j_2}^T x^{(l)} + b_{j_2} > 0 \quad \ldots \quad w_{j_q}^T x^{(l)} + b_{j_q} > 0$$

For the case $p = 0$ we have a trivial linear mapping from the previous layer to only positive values of the output. This means that the preimage is just the point $x^{(l)}$. In the general case where $p > 0$ the preimage will contain elements x such that $w_i^T x + b_i < 0$ for $i_1, i_2, \ldots i_p$. In order to identify these we will define the null spaces of the linear mappings w_i:

$$\Pi_i \;=\; \{x : w_i^T x + b_i = 0 \quad i = 1 \ldots n\}$$

These null spaces are hyperplanes in space of activities at layer (l). Obviously, any input element x that is mapped to the negative side of the hyperplane generated by the mapping w^i will get mapped to this hyperplane by the ReLU function. In order to identify this mapping we will define a set of basis vectors for elements of the input space from the one dimensional linear subspaces generated by the intersections:

$$\pi_i \;=\; \Pi_1 \cap \Pi_2 \cap \ldots \cap \Pi_{i-1} \cap \Pi_{i+1} \cap \ldots \cap \Pi_n$$

Each one dimensional subspace π_i is generated by intersecting the hyperplanes associated with the nullspaces of the remaining linear mapping kernels. The fact that these intersections generate one dimensional subspaces can be seen most easily by just noting that each intersection of a succession of n-dimensional hyperplanes gives rise to a linear manifold with dimension one lower at each intersection. For each subspace π_i we can now define a basis unit vector e_i such that each element of π_i can be expressed as $x = \alpha_i e_i$. We can also define the direction and length of e_i by requiring that $w_i^T e_i = 1$. The assumed full rank of the mapping W guarantees that the system $e_1, e_2 \dots e_n$ is complete in the input space. We can therefore express any vector as:

$$x \ = \ \sum_1^n \alpha_i e_i$$

Since e_i is in the nullspace of every remaining kernel except i we have: $w_j^T e_i = 0 \quad i \neq j$ This means that:

$$w_j^T x \ = \ \sum_1^n \alpha_i w_j^T e_i = \alpha_j$$

The subspace coordinates α_i are therefore a convenient tool for identifying the preimage of the mapping between the successive layers in a rectifier network. Since for $j = i_1, i_2, \dots i_p$ we will have $\alpha_j > 0$ and for $j = j_1, j_2, \dots j_q$ we will have $\alpha_j \leq 0$. By definition, the actual computation of the bases e_i is done by finding the nullspace of the matrix W where the i:th row is deleted. We also have that the matrix $(e_1, e_2, \dots, e_n)$ is the inverse of W.

We can therefore finally formulate the procedure for identifying the preimage of a mapping between successive layers in a rectifying network as:

> Given the mapping where the activity of the j:th node is computed as:
>
> $$x_j^{(l+1)} \ = [w_j^T x^{(l)} + b_j]_+ \tag{2}$$
>
> we identify indices $j = i_1, i_2, \dots i_p$ where $w_j^T x^{(l)} + b_j > 0$ and $j = j_1, j_2, \dots j_q$ where $w_j^T x^{(l)} + b_j \ \leq \ 0$ Using kernels $w_1 \dots w_n$ to define their corresponding null-space hyperplanes $\Pi_1 \dots \Pi_n$ we generate one dimensional subspaces π_i by intersecting the complementary set of null-space hyperplanes:
>
> $$\pi_i \ = \ \Pi_1 \cap \Pi_2 \cap \dots \cap \Pi_{i-1} \cap \Pi_{i+1} \cap \dots \cap \Pi_n$$
>
> and define basis vectors for these as e_i. Any element in the input space can now be expressed as a linear combination:
>
> $$x \ = \ \alpha_{i_1} e_{i_1} + \alpha_{i_2} e_{i_2} + \dots \alpha_{i_p} e_{i_p} - \ \alpha_{j_1} e_{j_1} - \alpha_{j_2} e_{j_2} - \dots \alpha_{j_q} e_{j_q}$$
>
> where all $\alpha_i \geq 0$. The preimage set is then generated by assigning arbitrary values > 0 to the coefficients $\alpha_{j_1}, \alpha_{j_2}, \dots \alpha_{j_q}$

Figure 1 illustrates the associated hyperplanes Π_1, Π_2, Π_3 in the case of three nodes and the respective unit vectors e_1, e_2, e_3 with positive directions indicated by arrows. For the all positive octant, i.e. all $w_i^T x > 0$ the linear mapping is just full rank and the preimage is just the associated input (x_1, x_2, x_3). For three other octants the preimages for three selected points are illustrated:

1. For $w_1^T x + b_1 > 0, w_2^T x + b_2 > 0, w_3^T x + b_3 < 0$, the preimage of a point on the plane Π_3 consist of all points on the indicated arrow.
2. For $w_1^T x + b_1 > 0, w_2^T x + b_2 < 0, w_3^T x + b_3 > 0$, the preimage of a point on the plane Π_2 consist of all points on the indicated arrow.
3. For $w_1^T x + b_1 > 0, w_2^T x + b_2 < 0, w_3^T x + b_3 < 0$, the preimage of a point on the intersection of planes Π_2 and Π_3 consist of all points on the indicated grey shaded area

In general, for points that are not in the all positive $w_i^T x > 0 \forall i$ region, they will be located on a linear submanifold spanned by the unit vectors $e_{i_1}, e_{i_2}, \dots, e_{i_p}$

$$x \ = \ \alpha_{i_1} e_{i_1} + \alpha_{i_2} e_{i_2} + \dots \alpha_{i_p} e_{i_p}$$

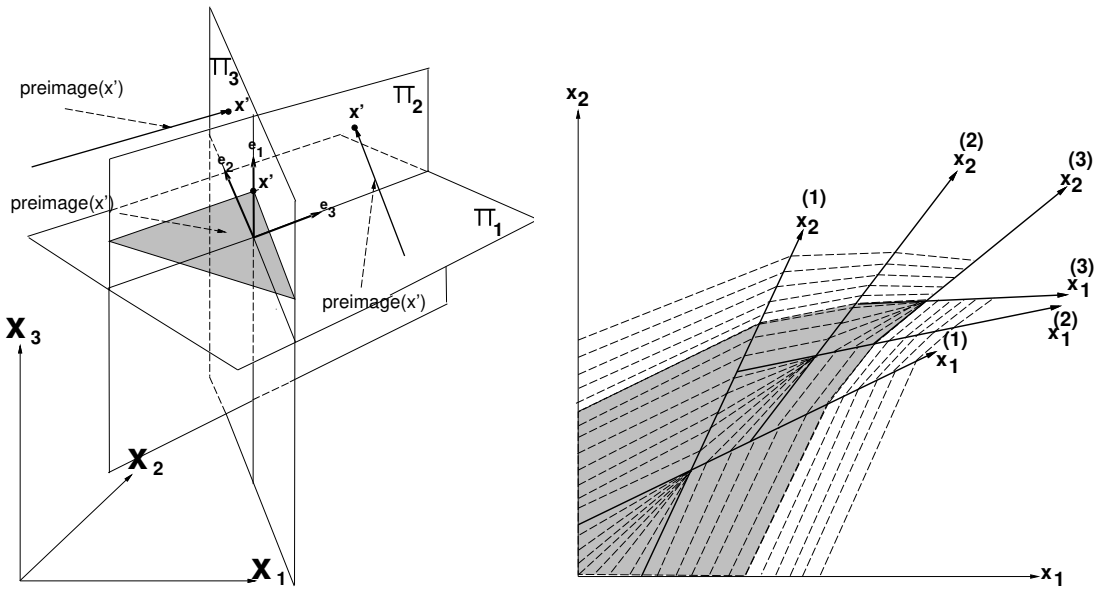

Figure 1:

Left: *Hyperplanes* $\Pi_1, \Pi_2\Pi_3$ *of nullspaces for transformation kernels and the associated unit vectors* e_1, e_2, e_3 *from pairwise intersections* $(\Pi_2\Pi_3)$ (Π_1, Π_3) *and* (Π_1, Π_2) *respectively. The preimages of various points in the output are indicated as arrows or the shaded area*

Right: *Preimages at various levels of a rectifier network with input* (x_1, x_2) *and output activity* $(x_1^{(3)}, x_2^{(3)})$ *All elements in the grey shaded area eventually get mapped to output activity* $(0, 0)$ *and are irreversibly mixed.*

The preimage then consists of all points on the linear manifold:

$$x \ - \ \alpha_{j_1} e_{j_1} - \alpha_{j_2} e_{j_2} - \ldots \alpha_{j_q} e_{j_q}$$

where all $\alpha_i \geq 0$.

For a multi level network , preimages for elements that are mappings between successive levels will therefore consist of pieces of linear manifolds in the input space at that level of dimensions determined by the number of nodes with positive output for that element. By mapping back to the original input space, preimages for specific elements at a certain level will be piecewise linear manifolds, the elements of which all map to that specific element. This is exactly what is illustrated in figure 1 for the case of 2-dimensional inputs and a network with three levels of two nodes at each level.These piecewise linear manifolds can therefore be considered as fundamental building blocks for mapping input distributions to node outputs at any level of the network.

REFERENCES

Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Aravindh Mahendran and Andrea Vedaldi. Visualizing deep convolutional neural networks using natural pre-images. *International Journal of Computer Vision (IJCV)*, 2016.