Solution 7

Robustness of Neural Networks

Program Analysis for System Security and Reliability 2018
ETH Zurich

August 5, 2018

In this exercise, we will consider the verification of *robustness* property for neural networks. We first define robustness below:

Robustness A robustness property for a neural network is a pair (X, C) consisting of a robustness region X and robustness condition C. A network is robust if all samples from X satisfy the condition C.

Consider the neural network in Figure 1 which takes input x at neuron v_{11} and produces output y at neuron v_{31} and has one hidden layer with two neurons v_{21} and v_{22}. The labels on the edges show the weights of the connections and the biases are assumed to be zero.

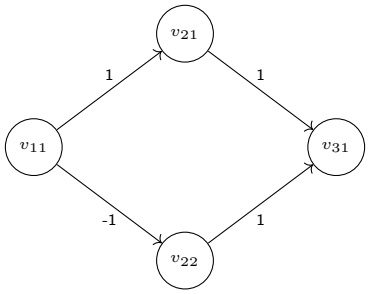

Figure 1: Original neural network.

For convenience , we split both neurons in the hidden layer into two to represent affine transformation and RELU separately. The modified network is shown in Figure 2.

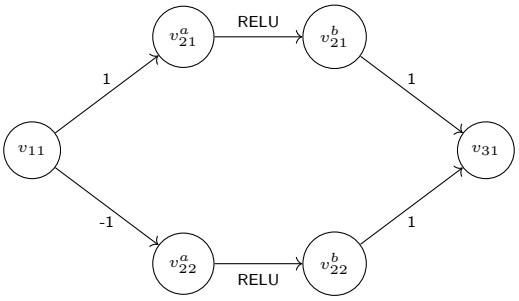

Figure 2: Neural network after splitting the neurons in the hidden layer.

Consider,

$$X = \{x \in \mathbb{R} \mid x \in [0.5, 1]\} \text{ and,}$$
$$C = \{y \in \mathbb{R} \mid y \in (0.4, 1]\}.$$

Show that the given neural network is robust for (X, C) using Reluplex.

Solution: We need to prove that the output cannot take values outside the range $(0.4, 1]$ when the input is in the range $[0.5, 1]$. We show that the Reluplex procedure cannot find a satisfying tableau configuration when the output v_{31} is in the range $[0, 0.4]$ for the input $v_{11} \in [0.5, 1]$. The cases for the remaining values of v_{31} can be solved similarly.

The Reluplex procedure maintains lower bound L_x, upper bound U_x and current satisfying assignment α_x for each variable x. Further, the procedure introduces new basic variables a_1, a_2, a_3 for encoding the network with the equations:

$$\begin{aligned} a_1 &= v_{11} - v_{21}^a \\ a_2 &= v_{11} + v_{22}^a \\ a_3 &= v_{31} - v_{21}^b - v_{22}^b \end{aligned} \tag{1}$$

These equations form the initial tableau T_0, and the initial set of basic variables is $\mathcal{B} = \{a_1, a_2, a_3\}$. The set of ReLU constraints is $\mathcal{R} = \{< v_{21}^a, v_{21}^b >, < v_{22}^a, v_{22}^b >\}$. The initial assignment of all variables is set to 0. The lower and upper bounds of the basic variables are set to 0, in order to enforce the equalities that they represent. The bounds for the input and output variables are set to $[0.5, 1]$ and $[0, 0.4]$ respectively. The hidden variables of the neural network are unbounded, except that forward-facing variables v_{21}^b, v_{22}^b are, by definition, non-negative.

In the initial configuration, we notice that the assignment to v_{11} violates its bounds and therefore we fix it by updating its assignment to 0.5. As a result, a_1 and a_2, which depend on v_{11} are also set to 0.5. Since a_1 and a_2 are basic and out of bounds, so we

	v_{11}	v_{21}^a	v_{21}^b	v_{22}^a	v_{22}^b	v_{31}	a_1	a_2	a_3
L	0.5	$-\infty$	0	$-\infty$	0	0	0	0	0
α	0	0	0	0	0	0	0	0	0
U	1	∞	∞	∞	∞	0.4	0	0	0

pivot them with v_{21}^a and v_{22}^a and then update a_1 and a_2 back to 0. The tableau now consists of the equations:

$$\begin{aligned} v_{21}^a &= v_{11} - a_1 \\ v_{22}^a &= a_2 - v_{11} \\ a_3 &= v_{31} - v_{21}^b - v_{22}^b \end{aligned} \tag{2}$$

The assignments for v_{21}^a and v_{22}^a are updated to 0.5 and -0.5 respectively. The updated tableau has the following configuration:

	v_{11}	v_{21}^a	v_{21}^b	v_{22}^a	v_{22}^b	v_{31}	a_1	a_2	a_3
L	0.5	$-\infty$	0	$-\infty$	0	0	0	0	0
α	0.5	0.5	0	-0.5	0	0	0	0	0
U	1	∞	∞	∞	∞	0.4	0	0	0

Now the ReLU constraint between v_{21}^a and v_{21}^b is violated as the assigment to $v_{21}^a = 0.5$ whereas v_{21}^b is assigned to 0. We therefore, update the assignmnent to v_{21}^b. As a result of this assignment, a_3 which depends on v_{21}^b is set to 0.5. Since a_3 is basic and out of bounds, so we pivot it with v_{31} and then update a_3 to 0. The tableau now consists of the equations:

$$\begin{aligned} v_{21}^a &= v_{11} - a_1 \\ v_{22}^a &= a_2 - v_{11} \\ v_{31} &= a_3 + v_{21}^b + v_{22}^b \end{aligned} \tag{3}$$

The assignment to v_{31} is updated to 0.5. The updated tableau has the following configuration:

	v_{11}	v_{21}^a	v_{21}^b	v_{22}^a	v_{22}^b	v_{31}	a_1	a_2	a_3
L	0.5	$-\infty$	0	$-\infty$	0	0	0	0	0
α	0.5	0.5	0.5	-0.5	0	0.5	0	0	0
U	1	∞	∞	∞	∞	0.4	0	0	0

The bounds for v_{31} are now violated. We set it to 0 and pivot with v_{21}^b which now becomes the basic variable. Its assignment is changed to 0. The tableau now consists of the equations:

$$
\begin{aligned}
v_{21}^a &= v_{11} - a_1 \\
v_{22}^a &= a_2 - v_{11} \\
v_{21}^b &= v_{31} - a_3 - v_{22}^b
\end{aligned} \tag{4}
$$

The updated tableau has the following configuration:

	v_{11}	v_{21}^a	v_{21}^b	v_{22}^a	v_{22}^b	v_{31}	a_1	a_2	a_3
L	0.5	$-\infty$	0	$-\infty$	0	0	0	0	0
α	0.5	0.5	0	-0.5	0	0	0	0	0
U	1	∞	∞	∞	∞	0.4	0	0	0

Now, the ReLU constraint between v_{21}^a and v_{21}^b is not satisfied. For satisfying the constraint we set v_{21}^a to 0. The tableau now has the following configuration:

	v_{11}	v_{21}^a	v_{21}^b	v_{22}^a	v_{22}^b	v_{31}	a_1	a_2	a_3
L	0.5	$-\infty$	0	$-\infty$	0	0	0	0	0
α	0.5	0	0	-0.5	0	0	0	0	0
U	1	∞	∞	∞	∞	0.4	0	0	0

Since v_{21}^a is basic, its assignment to 0 makes the equality involving it invalid. We notice that the assignment to none of the non-basic variables a_1 and v_{11} in the equation can be changed without violating their bounds for making the equality valid. The Reluplex procedure determines that the configuration is unsatisfiable and finishes with `UNSAT` answer proving that the given robustness property holds for the neural network.