Layerwise Derived Valid Inequalities for the Binarized Neural Network Verification Problem

Woojin Kim [1], Jim Luedtke [2]

[1]Department of Mathematics, University of Wisconsin-Madison, wkim73@wisc.edu [2]Department of Industrial & Systems Engineering, University of Wisconsin-Madison, jim.luedtke@wisc.edu

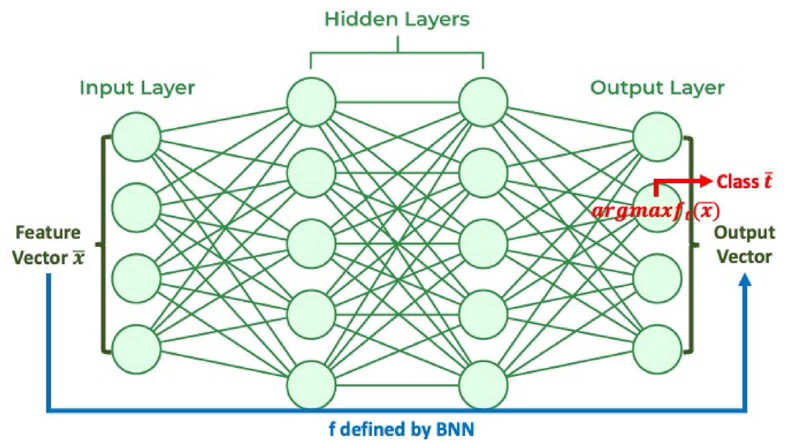
Binarized Neural Networks (BNNs)

Definition

Feedforward neural networks with binary weights and activation functions (Hubara et al. [2016])

Strengths

- Reduce memory size and improve power-efficiency (Hubara et al. [2016])
- Applied in small embedded devices (McDanel et al. [2017])
- Achieve comparable results as deep neural networks in image classification (Hubara et al. [2016]) and image super resolution (Ma et al. [2019])



BNN Verification Problem

Notation

- L: number of hidden layers
- n^ℓ: number of neurons in the ℓ^{th} layer ($\ell \in \{0, \cdots, L+1\}$)
- N^ℓ: set of neurons in the ℓ^{th} layer ($\ell \in \{0, \cdots, L+1\}$)
- $q \in \mathbb{N}$: coordinates of feature vectors are quantized as multiples of $\frac{1}{q}$

Problem

Is there a perturbed feature vector $\mathbf{x}^0$ close to $\bar{\mathbf{x}}$ that a given BNN classifies as a class $t \neq \bar{t}$?

$$z^*_\epsilon(\bar{\mathbf{x}}) := \max_{\substack{\mathbf{x}^0 \in \frac{1}{q}\mathbb{Z}_+^{n^0} \cap [0,1]^{n^0} \\ \|\mathbf{x}^0 - \bar{\mathbf{x}}\|_1 \le \epsilon}} \{f_t(\mathbf{x}^0) - f_{\bar{t}}(\mathbf{x}^0) : t \in N^{L+1} \setminus \{\bar{t}\}\} > 0?$$

Application

- Measure the robustness of BNNs by solving the BNN verification problem for many feature vectors

Previous Work

- Narodytska et al. [2018] investigated the BNN verification problem as Boolean satisfiability problems
- Fischetti and Jo [2018] proposed a MILP formulation for the deep neural network verification problem and applied the idea of fixing variables to solve the obtained MILP problem

MIP Formulation

Decision Variables

- $\mathbf{x}^0$: decision variables for the perturbed feature vector
- $\mathbf{x}^\ell$: binary decision variables for the output vector of the ℓ^{th} hidden layer

Notation

- $X^0 := \{\mathbf{x}^0 \in \frac{1}{q}\mathbb{Z}_+^{n^0} \cap [0,1]^{n^0} : \left\|\mathbf{x}^0 - \bar{\mathbf{x}}\right\|_1 \le \epsilon\}$
- $\mathbf{W}^\ell$: weight matrix between the $(\ell-1)^{\text{th}}$ layer and the ℓ^{th} layer
- $\mathbf{b}^\ell$: bias vector between the $(\ell-1)^{\text{th}}$ layer and the ℓ^{th} layer
- $a^\ell(\mathbf{x}^{\ell-1}) := \mathbf{W}^\ell(2\mathbf{x}^{\ell-1} - \mathbf{1}) + \mathbf{b}^\ell$
- $g^\ell(\mathbf{x}^{\ell-1}) := \mathbb{1}_{\mathbb{R}_+}(a^\ell(\mathbf{x}^{\ell-1}))$

Formulation

$$\begin{aligned}
\max_{\mathbf{x}^0, \cdots, \mathbf{x}^L} \quad & \max\{a_t^{L+1}(\mathbf{x}^L) - a_{\bar{t}}^{L+1}(\mathbf{x}^L) : t \in N^{L+1} \setminus \{\bar{t}\}\} \\
\text{s.t.} \quad & \mathbf{x}^\ell = g^\ell(\mathbf{x}^{\ell-1}),\ \forall \ell \in \{1, \cdots, L\}, \\
& \mathbf{x}^0 \in X^0, \\
& \mathbf{x}^\ell \in \{0,1\}^{n^\ell},\ \forall \ell \in \{1, \cdots, L\}
\end{aligned}$$

Objective Function Linearization

Two Ways to Linearize the Objective Function

Consider each alternative class $t \in N^{L+1} \setminus \{\bar{t}\}$ individually

- Used in previous work on MIP methods to solve the BNN verification problem
- Solve the obtained MIP problem to obtain the maximum $z^*_\epsilon(\bar{\mathbf{x}}, t)$ for each $t \in N^{L+1} \setminus \{\bar{t}\}$
- Find $z^*_\epsilon(\bar{\mathbf{x}}) = \max\{z^*_\epsilon(\bar{\mathbf{x}}, t) : t \in N^{L+1} \setminus \{\bar{t}\}\}$

Solve a single MIP problem incorporating all decisions on $t \in N^{L+1} \setminus \{\bar{t}\}$

- Developed for better MIP methods to solve the BNN verification problem in our work
- Add a binary decision variable z_t indicating whether t is selected as an alternative class ($t \in N^{L+1} \setminus \{\bar{t}\}$)
- Add a binary decision variable v_{ti} for $z_t x_i^L$ ($t \in N^{L+1} \setminus \{\bar{t}\}, i \in N^L$)

Layerwise Derived Valid Inequalities

Notation

- $X^\ell := g^\ell(X^{\ell-1})$ ($\ell \in \{1, \cdots, L\}$)
- $X^0_{\text{out}} := X^0$
- $X^\ell_{\text{out}} \subset \{0,1\}^{n^\ell}$: set containing X^ℓ ($\ell \in \{1, \cdots, L\}$)

Observation

- With access to a description for X^L, $z^*_\epsilon(\bar{\mathbf{x}})$ can be obtained by solving a MIP problem on X^L

Goal

- Find valid inequalities for X^ℓ using an outer approximation $X^{\ell-1}_{\text{out}}$ for $X^{\ell-1}$ by each layer
- Add obtained valid inequalities to the MIP formulation to solve the MIP problem for the BNN verification problem more efficiently

Valid Inequalities: Variable Fixing

Question

Is $c_i x_i^\ell \le \frac{c_i - 1}{2}$ valid for X^ℓ? ($i \in N^\ell$, $c_i \in \{-1, 1\}$)

- Motivated by Fischetti and Jo [2018]'s idea to fix variables in the deep neural network verification problem

Answer (Case with $c_i = 1$)

Yes if

$$\begin{aligned}
&\max\{x_i^\ell : x_i^\ell = g_i^\ell(\mathbf{x}^{\ell-1}), \mathbf{x}^{\ell-1} \in X^{\ell-1}_{\text{out}}\} \le 0 \\
\Leftrightarrow\ &\max\{a_i^\ell(\mathbf{x}^{\ell-1}) : \mathbf{x}^{\ell-1} \in X^{\ell-1}_{\text{out}}\} < 0
\end{aligned}$$

Valid Inequalities: Two-variable Inequalities

Question

Is $c_i x_i^\ell + c_k x_k^\ell \le \frac{c_i + c_k}{2}$ valid for X^ℓ? ($i, k \in N^\ell$ satisfying $i > k$, $c_i, c_k \in \{-1, 1\}$)

Answer (Case with $c_i = 1$ and $c_k = 1$)

Yes if

$$\begin{aligned}
&\max\{x_i^\ell + x_k^\ell : x_i^\ell = g_i^\ell(\mathbf{x}^{\ell-1}), x_k^\ell = g_k^\ell(\mathbf{x}^{\ell-1}), \mathbf{x}^{\ell-1} \in X^{\ell-1}_{\text{out}}\} \le 1 \\
\Leftrightarrow\ &\max\{a_i^\ell(\mathbf{x}^{\ell-1}) : \mathbf{x}^{\ell-1} \in X^{\ell-1}_{\text{out}}, a_k^\ell(\mathbf{x}^{\ell-1}) \ge 0\} < 0
\end{aligned}$$

Algorithm with Layerwise Derived Valid Inequalities

Main Ideas

- For each candidate, check whether a layerwise derived valid inequality is valid by solving the MIP subproblem
- Add obtained valid inequalities to the MIP formulation for the BNN verification problem

Bottleneck: Solving MIP Subproblems

- Create an inner approximation X^ℓ_{in} for X^ℓ
- Rule out valid inequalities violated by a vector in X^ℓ_{in} to avoid solving MIP subproblems

Finding Two-variable Inequalities is Harder than Finding Variable Fixings

- Find only variable fixings first
- Solve the MIP problem by exploring the root node
- Find two-variable inequalities only if it fails to solve to the optimality

Computational Results

- Basic(Indiv): solve the MIP problem obtained by considering each alternative class individually for each alternative class $t \in N^{L+1} \setminus \{\bar{t}\}$
- Basic(Incorp): solve the single MIP problem obtained by incorporating decisions on alternative classes
- Fix: solve the single MIP problem with added variable fixing
- Fix+TwoVar: solve the single MIP problem with added variable fixing and (if needed) two-variable inequalities

Method	Mean Relative LP Gap	Mean Verification Time (sec.)
Basic(Indiv)	271.6%	1279.1
Basic(Incorp)	271.6%	568.0
Fix	80.6%	355.0
Fix+TwoVar	39.8%	112.7

Table 1. Results for Instances with Non-ϵ-perturbed $\bar{\mathbf{x}}$

Method	Mean Relative LP Gap	Time Limit (1 hour) Fraction
Basic(Indiv)	247.7%	98.5%
Basic(Incorp)	252.2%	100.0%
Fix	269.9%	100.0%
Fix+TwoVar	243.5%	100.0%

Table 2. Results for Instances with ϵ-perturbed $\bar{\mathbf{x}}$

Conclusion

- The MIP method employing layerwise derived valid inequalities outperforms the other MIP methods as a method to solve the BNN verification problem
- The objective function linearization incorporating decisions on alternative classes results in a more efficient method to solve the BNN verification problem than the other linearization considering each alternative class individually