Simulation Setup

Let $H^{(k)} = (h_1^{(k)}, h_2^{(k)}, \cdots, h_{n_k}^{(k)})^T$ be the column vector of node functions in the k^{th} layer, where n_k is the number of nodes in the k^{th} layer, $k = 1, 2, \cdots, L$, and let $h_j^{(1)} = \sigma(w_j^{(0)}x + b_j^{(0)})$, $h_j^{(2)} = \sigma(w_j^{(1)}H^{(1)} + b_j^{(1)})$, $\cdots$, $h_j^{(k)} = \sigma(w_j^{(k-1)}H^{(k-1)} + b_j^{(k-1)})$ be the node functions, where $j = 1, \cdots, n_k$, $b_j^{(0)}, b_j^{(1)}, \cdots b_j^{(k-1)} \in \mathbb{R}$, and $w_j^{(0)}, w_j^{(1)}, \cdots, w_j^{(k-1)} \in \mathbb{R}^p$ are row vectors with $w_j^{(0)} = (w_{j,1}^{(0)}, \cdots, w_{j,p}^{(0)})$ as the weights from the input layer to the j^{th} node in the first layer, where $w_{j,i}^{(0)} = \frac{a_i c}{\sum_{k=1}^{p} |a_k|}$ is the weight of the i^{th} node in the input layer to the j^{th} node in the first layer, for $c > 0$ and $i = 1, \cdots, p$, where p is the input dimension. Suppose $b_j^{(0)} \sim N(0, 1)$ and denote $a_1, \cdots, a_p$ to be the p elements in $w_j^{(0)}$ that is generated from a distribution of interest (simulations can be tried with Normal distribution, t distribution, Cauchy distribution, etc.). Generate $w_j^{(1)}, b_j^{(1)}, w_j^{(2)}, b_j^{(2)}, \cdots, w_j^{(k-1)}, b_j^{(k-1)}$ similarly to obtain the teacher network. Compute the sparsity index $TSI_j^{(k-1)} = \frac{|w_j^{(k-1)}|_1}{|w_j^{(k-1)}|_q}$ for each node j, and the average sparsity index of each layer $TSI^{(k-1)} = \frac{1}{n_{k-1}} \sum_{j=1}^{n_{k-1}} TSI_j^{(k-1)}$. The function for the j^{th} node in the k^{th} layer can be represented as $g_j^{(k)} = w_j^{(k-1)}H^{(k-1)} + b_j^{(k-1)}$, i.e. $h_j^{(k)} = \sigma(g_j^{(k)})$. The output function is $y = w^{(L)}H^{(L)} + b^{(L)}$, and its sparsity index $TSI^{(L)} = \frac{|w^{(L)}|_1}{|w^{(L)}|_q}$. The average sparsity index across all layers is $\overline{TSI} = \frac{1}{L} \sum_{k=1}^{L} TSI^{(k)}$, where $k = 1, 2, \cdots, L$.

For the training data, $x^{(i)}$, where $i = 1, 2, \cdots, n$ (n is the sample size), compute $H_i^{(1)}, H_i^{(2)}, \cdots, H_i^{(L)}$ and y_i.

After approximation, denote $N_y^{(L)}$ as the set of nodes in layer L that are selected to connect with output y, with cardinality $K_L := |N_y^{(L)}|$, and $\forall j \in N_y^{(L)}$, denote $N_j^{(L-1)}$ as the set of nodes in layer $L-1$ that are selected to connect with the j nodes in layer L, with cardinality $n_j^{(L-1)} := |N_j^{(L-1)}|$, and denote $K_{L-1} := |N^{(L-1)}|$, where $N^{(L-1)} = \bigcup_{j \in N_y^{(L)}} N_j^{(L-1)}$. Suppose $N^{(k)}$ is the set of nodes selected in layer k. $\forall j \in N^{(k)}$, denote $N_j^{(k-1)}$ as the set of nodes in layer $k-1$ that are selected to connect with the j nodes in layer k, with cardinality $n_j^{(k-1)} := |N_j^{(k-1)}|$, and denote $K_{k-1} := |N^{(k-1)}|$, where $N^{(k-1)} = \bigcup_{j \in N^{(k)}} N_j^{(k-1)}$, $k = L, L-1, \cdots, 2, 1$ (suppose the input layer is the 0^{th} layer).

Below is an outline for the backward approximation learning algorithm.

1. Compute $Loss^{(L)} = \sum_{i=1}^{n} (w^{(L)}H_i^{(L)} + b^{(L)} - y_i)^2 + \lambda_1^{(L)}|w^{(L)}|_1 + \lambda_2^{(L)}|b^{(L)}|_1$, where $H_i^{(L)}$ is

calculated based on $x^{(i)}$.

2. Select K_L nodes in the L^{th} layer and their corresponding $\widetilde{w}^{(L)}$, $\widetilde{b}^{(L)}$
3. Denote the set of the nodes selected as $N_y^{(L)}$, and let $k = L$, $N^{(k)} = N_y^{(L)}$
4. Set $N^{(k-1)} = \emptyset$
5. For $j \in N^{(k)}$, let $y^{(k)} = g_j^k$
6. $Loss_j^{(k-1)} = \sum_{i=1}^n (w_j^{(k-1)} H_i^{(k-1)} + b_j^{(k-1)} - y_i^{(k)})^2 + \lambda_1^{(k-1)} |w_j^{(k-1)}|_1 + \lambda_2^{(k-1)} |b^{(k-1)}|_1$
7. Select $n_j^{(k-1)}$ nodes in the $(k-1)^{th}$ layer and their corresponding $\widetilde{w}_j^{(k-1)}$, $\widetilde{b}_j^{(k-1)}$
8. Compute $N^{(k-1)} = N^{(k-1)} \bigcup N_j^{(k-1)}$; end for
9. Compute $K_{k-1} := |N^{(k-1)}|$
10. Let $k = k-1$, if $k \geq 1$, continue Step 4 to Step 9 to obtain the student network $\widetilde{y}$
11. Calculate $\widetilde{y}_i$ with $x^{(i)}$, where $i = 1, \cdots, n$
12. Compute total number of nodes, $K = \sum_{j=1}^L K_j$
13. Compute total number of edges, $K_E = K_L + \sum_{j=1}^{K_L} n_j^{(L-1)} + \sum_{j=1}^{K_{L-1}} n_j^{(L-2)} + \cdots \sum_{j=1}^{K_1} n_j^{(0)}$
14. MSE $= \frac{1}{n} \sum_{i=1}^n (\widetilde{y}_i - y_i)^2$
15. Return $\widetilde{y}$, K, K_E, MSE

To Explore

For the teacher model, vary the following to explore different cases:

1. Sample size: $n = 2000, 5000, 10000$ in training data $x^{(i)} \sim N(0,1)$, $i = 1, \cdots, n$ with input dimension $p = 20$
2. Constant c in computing $w_{j,i}^{(0)}$: $c = 0.5, 1, 5, 10$
3. Distribution: Normal, student t, Cauchy (can also vary the parameters of each distribution)

4. Average sparsity index across all layers for the teacher model (by changing parameters in the distributions): $\overline{TSI} \approx 0.01, 0.1, 0.5, 0.9$ might not be able to achieve 0.9)

5. $q = 0.1, 0.3, 0.5$ for calculating TSI

6. Total number of layers: $L = 5, 10$ (same as the number of layers in the student network)

7. Number of nodes in each layer (consistent across all layers): $n_k = 100, 200, 500, 1000$

To Record

1. Information for the different cases from above for the teacher notwork

2. Output from Step 15 for the student model for each case explored from the above

3. Total number of nodes ($\sum_{j=1}^{L} n_j$) and edges ($pn_1 + n_1n_2 + \cdots + n_{L-1}n_L + n_L$) for the teacher network

References

Fuchang Gao, Ching-Kang Ing, and Yuhong Yang. Metric entropy and sparse linear approximation of q-hulls for 0¡ q 1. *Journal of Approximation Theory*, 166:42–55, 2013.