MAX-TOTAL-FLOW GENERALIZATION BOUNDS FOR DAG-STRUCTURED RELU DEEP NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Understanding generalization in over-parameterized neural networks remains a fundamental challenge. While recent path-norm frameworks address general directed acyclic graph (DAG) networks with skip connections, branching, and merging, they require computing astronomically large path-norms that render generalization bounds vacuous in practice. In this study, a new perspective is proposed by interpreting Karush–Kuhn–Tucker (KKT) optimality conditions as a flow conservation law over the DAG. At a KKT point, the squared Frobenius norms of weight matrices define non-negative flows on edges that satisfy classical conservation equations at every interior node. Using flow decomposition, we prove that at every KKT point the total parameter norm is bounded in terms of the longest source-to-sink path length T_G of the DAG G and the regularization parameter λ. Combining this norm control with a held-out validation set and a change-of-measure argument under a Gaussian perturbation of the weights with noise level ϵ, we show that, with probability at least $1-\delta$, the validation loss of the trained network exceeds its expected training loss by at most $O\big(\sqrt{\ln(1/\delta)/r} + \sqrt{T_G/(\epsilon^2\lambda n)}\big)$, for n training and r validation samples. The results cover binary and multi-class classification and regression, and extend to Leaky ReLU activations.

1 INTRODUCTION

Understanding why over-parameterized neural networks generalize is a central problem in learning theory. Classical analyses based on the VC-dimension tie sample complexity to the number of parameters (1), and therefore become vacuous when parameters greatly outnumber training examples, as in modern networks. This motivated a shift toward *norm-based* capacity control, where complexity is measured by the magnitude of the weights rather than their count. Representative results bound the Rademacher complexity or PAC-Bayes risk by products of layer-wise spectral or Frobenius norms (2; 3; 4), or by the *path norm* (5), which aggregates the product of weights along every input-to-output path.

However, most of these bounds are derived for chain-structured, fully connected networks, whereas modern architectures contain skip connections, branching, and merging, i.e., general directed acyclic graphs (DAGs). The recent work of Gonon et al. (6) closes this gap by developing a path-norm toolkit for general DAG ReLU networks. Building on the path-lifting representation (7; 8; 9), they define path-activations for arbitrary DAGs, show that path-norms can be computed in a single forward pass, and prove a generalization bound with only logarithmic depth dependence. Yet, as the authors acknowledge, the bound is vacuous in practice: for a dense ResNet-18 trained on ImageNet, the ℓ_1 path-norm is on the order of 10^{30}.

The vacuity of norm-based bounds is not specific to path norms. Bounds scaling with products of Frobenius norms (5; 4) are numerically very large in practice (10), and for deep networks such as ResNet-110 and DenseNet-100 the product of spectral norms appearing in margin-based bounds overflows numerically unless spectral normalization is imposed (11). Nagarajan and Kolter (10) identified a further, more troubling symptom: for SGD-trained networks, the weight norms entering these bounds grow polynomially with the number of training samples n, so the bounds can *increase* with n (for small batch sizes) even though the test error decreases. They also construct overparameterized settings in which even the tightest algorithm-dependent uniform convergence bound is vacuous. Their analysis, however, concerns training without explicit regularization, and they note that it does not

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

preclude guarantees when explicit regularization such as weight decay is used. This is the regime we study.

Worst-case bounds sit at one extreme. Rademacher-complexity and norm-based bounds hold uniformly over every hypothesis in a class, and PAC-Bayesian bounds hold simultaneously for every posterior, so their guarantees are valid for every solution and every learning algorithm. This generality is their strength, but it comes at a cost: they do not use which optimization problem is solved, and must therefore account for the worst hypothesis in the class rather than the solution that training returns. This matters in practice: Nagarajan and Kolter (10) show that, in the unregularized settings they study, even restricting uniform convergence to the hypotheses actually output by the algorithm can leave the bound vacuous.

At the other extreme, algorithm-specific analyses study the training procedure itself: the implicit bias of gradient descent (12), the stability of SGD (13), or the geometry and noise of training trajectories (14; 15). These works give precise insight into particular optimizers, but their guarantees typically hinge on specific architectures, initializations, or restrictive regimes, such as carefully chosen learning rates and batch sizes (16; 17), which makes it hard to obtain broad, a priori guarantees that are both quantitative and predictive.

Our approach is complementary to both. Rather than tracking the dynamics of an optimizer, our guarantees concern the weights returned at a Karush–Kuhn–Tucker (KKT) point of the regularized empirical risk. The learning algorithm enters only through the requirement that training reaches such a point. Since the KKT conditions are a property of the optimization problem rather than of the trajectory, they then constrain the learned weights whatever the optimizer, initialization, learning rate, or batch size. We show that they confine the learned weights to an ℓ_2 ball whose radius depends on the architecture only through the length of its longest path. A Gaussian perturbation of the weights turns this norm control into a generalization guarantee: it makes the divergence between the distribution of the training set given the learned weights and its unconditional distribution finite and controlled by the norm of the weights. A change-of-measure argument converts this divergence into a bound on the training gap, and a held-out validation set controls the remaining term. Combining the two, we obtain generalization guarantees for ReLU networks with arbitrary DAG architectures, including skip connections, branching, and merging.

Explicit regularization is essential here: the weight-decay parameter $\lambda > 0$ enters the KKT conditions directly, and it is through λ that they bound the squared parameter norm $\|U(S)\|^2$ by $(T_G + 2)/\lambda$, independently of the sample size n. Our analysis therefore lies in the regularized regime, which is not covered by the negative results of (10): those concern training without explicit regularization, where the relevant weight norms grow with n. The price is scope: the guarantee applies to KKT points of the regularized risk rather than to arbitrary hypotheses, and it requires setting aside a validation set.

For clarity, we take the underlying DAG G to have a single source node and a single sink node, with $L - 1$ multi-dimensional neurons, and we consider networks without bias terms. The source receives the input $x \in \mathbb{R}^d$ and the sink outputs a vector in $\mathbb{R}^{m_{L-1}}$, so d and m_{L-1} are the input and output dimensions. Every edge (p, q) carries a trainable weight block included in the ℓ_2 penalty; the matrix $U^{(q)}$ of node q stacks the blocks carried by its incoming edges. We treat training as a deterministic map $S \mapsto \mathcal{U}(S)$ (with initialization and data order fixed) that returns a KKT point of the empirical risk regularized with weight decay $\lambda > 0$, where the weights of $\mathcal{U}(S)$ are the matrices $U^{(l)}$, one for each node l of G, together with the output vector. The data are split into n training samples and r validation samples. Our bounds concern the trained weights $\mathcal{U}(S)$ themselves; the Gaussian perturbation $W = \mathcal{U}(S) + \epsilon\kappa$, $\kappa \sim \mathcal{N}(0, I)$, is only an analysis device that controls how much the trained weights reveal about the training set.

In order to bound the total parameter norm, the key idea is to read the KKT conditions as a flow-conservation principle. At a KKT point, the regularized squared norms of the weights carried by the edges of G behave as flows: they are nonnegative, they satisfy a conservation law at every interior node, and the flow entering the sink is bounded by the margin conditions. The conservation law is a node-level form of the balancedness of homogeneous networks (19), which holds exactly at stationary points of weight-decayed objectives (20). What is new here is treating these balanced norms as a flow on G. Bounding the total parameter norm thus becomes a maximum-total-flow problem on a DAG. By standard flow decomposition, the total flow is a sum of contributions along source-to-sink paths, each using at most T_G edges, where $T_G \leq L - 1$ is the length of the longest such path. This gives

Table 1: Summary of generalization bounds for neural networks under consistent notation, now considering DAG architectures instead of fully connected layers. T_G is the network depth (longest path in the DAG), n the number of training samples, $U^{(l)}$ the weight matrix for multi-dimensional neuron l, λ a regularization parameter, and $\mathbb{E}[.]$ the expectation over perturbations. Moreover, $\mathcal{L}$ and $\hat{\mathcal{L}}$ denote expected and empirical errors and x_i denotes the i-th training sample, m_l the width of node l, and y the vector of labels. Also, $\Phi(U)$ is defined as the sum over all input-to-output paths, where each contribution is given by the product of the weights along the corresponding path. Results are reported for binary classification (B-C), multi-class classification (M-C), and regression (Reg).

Paper	**Activation Function**	**Architecture**	**Bound Form (Simplified), O-notation**	**Problem**	**Data-dependent**	$W = \mathcal{U}(S)$
(18), Theorem 28	Sigmoid	LFCN	$\frac{\left(\max_{l=1}^{L} \|U^{(l)}\|_\infty\right)^{\frac{L(L+1)}{2}}}{\sqrt{n}}$	**B-C** ✓ **M-C** × **Reg** ×	No	No
(5), Corollary 7	ReLU, Ramp, Sigmoid	DAG	$\frac{(\max_i \|x_i\|_{p^*}) 2^{L-1} \Phi(U)}{\sqrt{n}}$	**B-C** ✓ **M-C** × **Reg** ×	No	No
(3), Theorem 1	ReLU	LFCN	$\frac{\max_i \|x_i\|_2 L^{3/2} \prod_{l=1}^{L} \|U^{(l)}\|_2}{\sqrt{n}}$	**B-C** ✓ **M-C** ✓ **Reg** ×	No	No
(2), Theorem 1.1	Lipschitz activations, $\sigma(0) = 0$	LFCN	$\frac{\|X\|_F L^{3/2} \prod_{l=1}^{L} \|U^{(l)}\|_2}{\sqrt{n}}$	**B-C** ✓ **M-C** ✓ **Reg** ×	No	No
(4), Theorem 3.2	Lipschitz activations, $\sigma(0) = 0$	LFCN	$\frac{\|X\|_2 L^{1/2} \prod_{l=1}^{L} \|U^{(l)}\|_F}{\sqrt{n}}$	**B-C** ✓ **M-C** ✓ **Reg** ×	No	No
(6), Theorem 3.1	ReLU, identity, k-max-pooling.	DAG	$\frac{(\max_i \|x_i\|_\infty) \Phi(U)}{\sqrt{n}}$	**B-C** ✓ **M-C** ✓ **Reg** ×	No	No
Our KKT-based bound, Theorems 6, 7 and 8	ReLU, Leaky ReLU, ReLU6	DAG	**Classification:** $\sqrt{\frac{\ln(1/\delta)}{2r}} + \sqrt{\frac{5\,(T_G+2)}{4\,\epsilon^2 \lambda\, n}}$ **Regression:** $\sqrt{\frac{\ln(1/\delta)}{2r}} + \sqrt{\frac{5\,(T_G+2)\,B^2}{16\,\epsilon^2 \lambda\, n}}$	**B-C** ✓ **M-C** ✓ **Reg** ✓	Yes	Yes

$\sum_{l=1}^{L-1} \|U^{(l)}\|_F^2 \leq \frac{T_G+1}{\lambda}$, a bound that depends on the architecture only through T_G, on the training procedure only through λ, and, unlike the norms observed in unregularized training (10), not on the sample size n.

For binary classification, Theorem 6 shows that, with probability at least $1 - \delta$ over the validation set, the validation loss of the trained weights exceeds their expected training loss by at most $\sqrt{\ln(1/\delta)/(2r)} + \sqrt{5(T_G+2)/(4\epsilon^2 \lambda n)}$, with extensions to multi-class classification, regression, and Leaky ReLU activations (Theorems 7 and 8). The parameter ϵ controls how much conditioning on the trained weights reveals about the training set. For large ϵ, the training term is small, but the expected training loss averages over many training sets and moves towards the population risk; for small ϵ, it approaches the observed training loss, but the bound grows as $1/\epsilon$. Unlike path-norm bounds, the analysis requires neither computing path norms nor any other aggregation over paths: the weight control comes from the graph structure and λ alone.

Table 1 reports influential generalization bounds for neural networks under a common notation (logarithmic factors omitted). These results are derived in different settings: they differ in the

activation functions, in the architecture (chain-shaped networks versus general DAGs), in the learning problem (binary classification, multi-class classification, regression), and in the object the guarantee applies to. The bounds of prior work hold uniformly over a hypothesis class, whereas ours concerns the trained weights at a KKT point of the regularized empirical risk, with high probability over a held-out validation set. The results are therefore not directly comparable, and the table is meant to place our bound in context rather than to rank the bounds. For context on the quantities involved, our bound depends on the architecture only through the longest path T_G, on the regularization parameter λ, and on the noise level ϵ, while the bounds of (5; 4; 6) depend on norm-based quantities such as $\prod_{l=1}^{L} \|U^{(l)}\|_F$ and $\Phi(U)$. Table 3 of (6) reports values of $\Phi(U)$ of order 10^{30} or higher for pretrained ResNets, and Table 2 (Appendix A) reports the values of $\Phi(U)$ for pretrained TorchVision models.

2 PRELIMINARIES

In this section, we introduce the notation and basic concepts used throughout the paper. Lowercase letters (e.g., u, v) denote vectors, while uppercase letters (e.g., A, B) denote matrices. For a matrix A, we denote by a_i its i-th column and by a_{ij} its (i, j)-th entry. Given a matrix A, we define $B = \max(0, A)$ entry-wise, i.e., $b_{ij} = \max(0, a_{ij})$. The inner product of two vectors u and v is defined as $\langle u, v\rangle = v^\top u$.

Let x be a vector, the *infinity norm* of x is defined by $\|x\|_\infty = \max_i |x_i|$. The *Euclidean norm* of x is defined by $\|x\|_2 = \left(\sum_i x_i^2\right)^{1/2}$. Generalizing the above, let $p \geq 1$ and let $p^* \geq 1$ denote the conjugate exponent, i.e., $\frac{1}{p} + \frac{1}{p^*} = 1$. The *$p$-norm* of a vector x is defined as $\sum_i |x_i|$ for $p = 1$, $\left(\sum_i |x_i|^p\right)^{1/p}$ for $1 < p < \infty$, and $\max_i |x_i|$ for $p = \infty$.

Let $A = [a_{ij}]$ be a matrix. The infinity norm of A is defined by $\|A\|_\infty = \max_i \sum_j |a_{ij}|$. The spectral norm of A is defined by $\|A\|_2 = \max_{\|x\|_2=1} \|Ax\|_2$, which is equal to the square root of the largest eigenvalue of $A^\top A$. The Frobenius norm of A is defined by $\|A\|_F = \left(\sum_{ij} a_{ij}^2\right)^{1/2}$. For $p, q \in [1, \infty)$, the matrix (p, q)-norm induced by the vector p- and q-norms is defined by $\|A\|_{p,q} = \max_{x \neq 0} \frac{\|Ax\|_q}{\|x\|_p} = \max_{\|x\|_p=1} \|Ax\|_q$. The Kullback–Leibler divergence between two distributions P and Q is denoted as $D_{\mathrm{KL}}(P\|Q)$.

Our generalization analysis combines two standard tools. The first is a held-out validation set, on which the loss of a hypothesis chosen independently of it concentrates around the population risk (21). The second is a change of measure. For a fixed hypothesis w, the distribution $\mathcal{P}_{S|W=w}$ of the training set, given that the algorithm produced w, generally differs from the data distribution $\mathcal{P}_S$. The Donsker–Varadhan variational formula (22) bounds expectations under $\mathcal{P}_{S|W=w}$ by expectations under $\mathcal{P}_S$, at the cost of the divergence $\mathrm{KL}(\mathcal{P}_{S|W=w} \,\|\, \mathcal{P}_S)$, which measures how much observing $W = w$ reveals about the training set. The Gaussian perturbation of the learned weights makes this divergence finite and explicit. Together, these tools give a high-probability statement for a specific hypothesis.

Let $\mathcal{Z} = (Z_1, \ldots, Z_{n+r}) \sim \mathcal{P}^{n+r}$ be an i.i.d. sample, split into a fixed training subset $S \subset \mathcal{Z}$ with $|S| = n$ and the validation subset $V = \mathcal{Z} \setminus S$ with $|V| = r$. A deterministic learning algorithm maps the training subset to $U(S) \in \mathbb{R}^d$ with $\|U(S)\| \leq c$ for every training subset S, and $W \mid S \sim \mathcal{N}\big(U(S), \epsilon^2 I_d\big)$ denotes its Gaussian perturbation. For $w \in \mathbb{R}^d$ and a loss $\ell(w, z)$, define the population risk $\mathcal{L}(w, \mathcal{P}) := \mathbb{E}_{Z\sim\mathcal{P}}[\ell(w, Z)]$ and the empirical risks on the two subsets,

$$\hat{\mathcal{L}}(w, S) := \frac{1}{n} \sum_{Z_i \in S} \ell(w, Z_i), \qquad \hat{\mathcal{L}}(w, V) := \frac{1}{r} \sum_{Z_i \in V} \ell(w, Z_i).$$

Theorem 1. *Assume $\ell(w, z) \in [0, 1]$ for all w and z. Then for every fixed $w \in \mathbb{R}^d$ and every $\delta \in (0, 1)$, with probability at least $1 - \delta$ over the draw of the validation subset V,*

$$\hat{\mathcal{L}}(w, V) - \mathbb{E}_{S|W=w}\big[\hat{\mathcal{L}}(w, S)\big] \;\leq\; \underbrace{\sqrt{\frac{\ln(1/\delta)}{2r}}}_{\textit{validation concentration}} \;+\; \underbrace{\sqrt{\frac{4c\|w\| + c^2}{4n\epsilon^2}}}_{\textit{training dependence}}. \tag{1}$$

The first term is a standard concentration term for the validation set, and it applies because w is chosen independently of V. The second term bounds the expected training gap given that the algorithm produced w. It is obtained by a change of measure from $\mathcal{P}_S$ to $\mathcal{P}_{S|W=w}$, which costs the divergence $\mathrm{KL}(\mathcal{P}_{S|W=w}\|\mathcal{P}_S)$, and the Gaussian form of $W \mid S$ turns this divergence into the explicit quantity above. The only property of the learning algorithm that the theorem uses is the norm bound $\|U(S)\| \le c$ for every training set. In Section 3 we show that at a KKT point this bound holds with $c = \sqrt{(T_G+2)/\lambda}$, which depends on the architecture only through its longest path. The proof of Theorem 1 is given in Appendix B.

3 Main Results

Consider a directed acyclic graph (DAG) $G = (\mathcal{V}, E)$, where $\mathcal{V} = \{1, \dots, L-1\}$ is the set of vertices and E is the set of directed edges. Each vertex represents a multi-dimensional neuron. For a neuron $q \in \mathcal{V}$, define the sets of antecedents and successors as $\pi_q := \{\, p \in \mathcal{V} \mid (p,q) \in E \,\}$, $\phi_q := \{\, p \in \mathcal{V} \mid (q,p) \in E \,\}$. Assume that neuron 1 is the source node, so $\pi_1 = \varnothing$, and neuron $L-1$ is the sink node, so $\phi_{L-1} = \varnothing$.

The ReLU function is defined as $\sigma(x) = \max\{0, x\}$ for $x \in \mathbb{R}$. Let $M_l = \sum_{p \in \pi_l} m_p$ and $U^{(l)} \in \mathbb{R}^{M_l \times m_l}$ be the weight matrix corresponding to the l-th multi-dimensional neuron, $v \in \mathbb{R}^{m_{L-1}}$ be the output layer weight vector, and $y_i \in \{-1, +1\}$. Here $m_0 = d$ denotes the input dimension. We study the following problem under the hinge loss:

$$\underset{U^{(l)}, v, X^{(l)}}{\text{minimize}} \quad \frac{1}{n}\sum_{i=1}^{n} \max(0, 1 - y_i \langle x_i^{(L)}, v\rangle) + \frac{\lambda}{2}\left(\sum_{l=1}^{L-1} \left\|U^{(l)}\right\|_F^2 + \|v\|^2\right) \tag{2}$$

$$\text{subject to} \quad x_{ji}^{(l)} = \sigma(\left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle), \quad \forall i = 1, \dots, n, \forall j = 1, \dots, m_{l-1}, \forall l = 2, \dots, L \tag{3}$$

(The KKT conditions for this problem are provided in Appendix C)

Rather than computing each neuron's Frobenius norm independently, we interpret the KKT conditions as defining a flow network over the DAG. Let define,

$$F^{(p\to q)} = \sum_{k=1}^{m_p}\sum_{i=1}^{n}\sum_{j=1}^{m_q} \left(\eta_{ji}^{(q+1)} x_{ji}^{(q+1)} - \theta_{ji}^{(q+1)}\right) u_{kj}^{(q)} x_{ki}^{(p+1)}. \tag{4}$$

The quantity $F^{(p\to q)}$ in equation equation 4 is associated with the edge $(p,q) \in E$ and has a simple meaning. Partition $U^{(q)}$ into row blocks $U_{[p]}^{(q)} \in \mathbb{R}^{m_p \times m_q}$, $p \in \pi_q$, where $U_{[p]}^{(q)}$ multiplies the output of parent p; these are exactly the weights carried by edge (p,q). At a KKT point, $F^{(p\to q)} = \lambda \|U_{[p]}^{(q)}\|_F^2$ (equation 49, Appendix D). Thus, the flow on an edge is the regularized squared Frobenius norm of the weights it carries. Moreover, $\lambda\|U^{(q)}\|_F^2 = \sum_{p\in\pi_q} F^{(p\to q)}$, so $\lambda\|\mathcal{U}^{(q)}\|_F^2$ is precisely the total flow entering node q.

Theorem 2 shows that for every interior node q, the regularized norm $\lambda\|U^{(q)}\|_F^2$ equals the total flow leaving q along its outgoing edges, while for the sink node this quantity is bounded. Theorem 3 then establishes flow conservation at every interior node—the total flow entering a node equals the total flow leaving it—which naturally leads to a Max-Total-Flow problem (see eqs. equation 10–equation 14).

Theorem 2. *Let $U^{(q)}$ denote the weight matrix at multi-dimensional neuron q. Under the problem constraints, the following holds:*

- ***Sink node.*** *For $q = L-1$ where $\phi_q = \varnothing$, we have*

$$\lambda\|U^{(q)}\|_F^2 \le 1. \tag{5}$$

- ***Interior node.*** *If $\phi_q \neq \varnothing$, then*

$$\lambda\|U^{(q)}\|_F^2 = \sum_{c\in\phi_q} F^{(q\to c)}. \tag{6}$$

- ***Non-negativity.*** *If $\pi_q \neq \varnothing$, then for every parent $p \in \pi_q$,*

$$F^{(p\to q)} \geq 0. \tag{7}$$

(The proof can be found in Appendix D)

Theorem 3. *For any node q in the DAG, the following holds:*

- *If $\phi_q \neq \varnothing$, then the flow entering q equals the flow leaving q:*

$$\sum_{p\in\pi_q} F^{(p\to q)} = \sum_{c\in\phi_q} F^{(q\to c)}. \tag{8}$$

- *If $\phi_q = \varnothing$ (i.e., q is the sink node), then the total inflow into q is bounded by one:*

$$\sum_{p\in\pi_q} F^{(p\to q)} \leq 1. \tag{9}$$

(The proof can be found in Appendix E)

By equation 49, Theorems 2 and 3 state that, at every interior node, the squared norm of the incoming weights equals that of the outgoing weights. This is a node-level manifestation of the balancedness property of positively homogeneous networks. Along gradient flow without regularization, the differences between incoming and outgoing squared norms are conserved (19; 9) for a general characterization of such conservation laws). Under ℓ_2 regularization, the rescaling symmetry of positively homogeneous networks implies that this balance holds exactly at stationary points (20). Theorems 2 and 3 recover this balance within our KKT framework and, in addition, identify each edge's regularized norm as a flow on G.

In the following, we first show that the sum over all weight matrices is bounded by the total flow on all edges, which establishes a bridge to derive a generalization bound using graph-theoretic properties.

Theorem 4. *Let $G = (\mathcal{V}, E)$ be a DAG with a single source and a single sink, and let the multi-dimensional neurons be indexed so that $\mathcal{V} = \{1, 2, \ldots, L-1\}$ correspond to the hidden layers. Then*

$$\lambda \sum_{q=1}^{L-1} \|U^{(q)}\|_F^2 \;\leq\; 1 + \sum_{q=1}^{L-2} \sum_{c\in\phi_q} F^{(q\to c)} = 1 + \sum_{(q,c)\in E} F^{(q\to c)}.$$

(The proof can be found in Appendix F)

By combining equation 5-equation 9 with Theorem 4, any solution to the following Max-Total-Flow problem provides a valid upper bound for $\lambda \sum_{q=1}^{L-1} \left\|U^{(q)}\right\|_F^2$. Therefore, instead of explicitly computing the sum of the Frobenius norms of the weight matrices at each neuron, it suffices to solve the following optimization problem. We assume that in the DAG G, neuron 1 is the source node, so $\pi_1 = \varnothing$, and neuron $L-1$ is the sink node, so $\phi_{L-1} = \varnothing$.

$$\underset{F^{(q\to c)}}{\text{maximize}} \quad \sum_{(q,c)\in E} F^{(q\to c)} \tag{10}$$

$$\text{subject to} \quad \sum_{p\in\pi_q} F^{(p\to q)} = \sum_{c\in\phi_q} F^{(q\to c)}, \quad \forall q = 2, \ldots, L-2, \tag{11}$$

$$F^{(p\to q)} \geq 0, \quad \forall (p,q) \in E, \tag{12}$$

$$\sum_{p\in\pi_{L-1}} F^{(p\to L-1)} \leq 1, \tag{13}$$

$$\sum_{c\in\phi_q} F^{(q\to c)} \geq 0, \quad \forall q = 1, \ldots, L-2. \tag{14}$$

Flow decomposition into path and cycle components is a classical result in network flow theory (23; 24; 25). It states that any feasible flow can be expressed as a non-negative combination of

source-to-sink paths (and possibly cycles). In a DAG with a single source and a single sink, cycles cannot appear, so the flow reduces to a sum of path flows. This representation is particularly useful for bounding the total amount of flow circulating in the network: each unit of flow on a path contributes to every edge along that path, so the sum over all edges counts each path as many times as its length. If the total flow value entering the sink is at most one, then the total edge flow is at most the length of the longest source-to-sink path. The following theorem formalizes this observation, which is central to our analysis of the Max-Total-Flow problem.

Theorem 5. *Let $G = (\mathcal{V}, E)$ be a DAG with a single source s and a single sink t, and let $\mathcal{P}$ be the set of all directed paths from s to t in G. Let T_G denote the length of the longest directed path from s to t. Suppose a feasible flow $F^{(p\to q)} : E \to \mathbb{R}_{\geq 0}$ satisfies flow conservation at every intermediate node and the total flow into the sink, as specified in equation 11-equation 14. Then $\sum_{(p,q)\in E} F^{(p\to q)} \leq T_G$.*

(The proof can be found in Appendix G)

In the next lemma, we establish an upper bound for the norm of v. This result will then allow us to derive a bound on the generalization error for binary classification. Let $S = (X, y)$ denote a dataset of n i.i.d. samples, where $X = (x_1, \ldots, x_n) \in \mathbb{R}^{d\times n}$ and $y = (y_1, \ldots, y_n)^T \in \{-1, +1\}^n$.

Lemma 1. *In the setting above, let $v(S)$ denote the weight vector of the output layer. Then we have $\|v(S)\|^2 \leq \frac{1}{\lambda}$.*

(The proof can be found in Appendix H)

As a consequence of Theorem 5, and by relating the total flow to the sum of squared Frobenius norms of the weight matrices, we obtain, $\lambda \sum_{l=1}^{L-1} \|U^{(l)}\|_F^2 \leq T_G + 1$. Let $\mathcal{U}(S) = [\mathrm{vec}(U^{(1)}(S)), \ldots, \mathrm{vec}(U^{(L-1)}(S)), v(S)]$ denote the concatenated parameter vector. Using Lemma 1 and Theorems 2-5, the total squared norm satisfies

$$\|\mathcal{U}(S)\|^2 = \sum_{l=1}^{L-1} \left\|U^{(l)}\right\|_F^2 + \|v(S)\|^2 \leq \frac{1}{\lambda}(1 + \sum_{(q,c)\in E} F^{(q\to c)} + 1) \leq \frac{T_G + 2}{\lambda}. \tag{15}$$

With this norm control, we can now state the main generalization guarantee.

Theorem 6. *Consider the setting of Theorem 1, with a training subset S of size n and a validation subset V of size r. Let $\mathcal{U}(S)$ be a KKT point of the regularized problem 2-3 trained on S, and let $W \mid S \sim \mathcal{N}\big(\mathcal{U}(S), \epsilon^2 I\big)$. For a data point $Z_i = (x_i, y_i)$, let $\ell(w, Z_i) = \mathbf{1}\big[y_i \langle x_i^{(L)}, v\rangle \leq 0\big]$ be the 0-1 loss, where $x_i^{(L)}$ denotes the activation at the final hidden node of the DAG G and v the output weight vector of w. Set $w := \mathcal{U}(S)$. Then for every $\delta \in (0, 1)$, with probability at least $1 - \delta$ over the draw of V,*

$$\hat{\mathcal{L}}(w, V) - \mathbb{E}_{S|W=w}\big[\hat{\mathcal{L}}(w, S)\big] \;\leq\; \sqrt{\frac{\ln(1/\delta)}{2r}} \;+\; \sqrt{\frac{5\,(T_G + 2)}{4\,\epsilon^2 \lambda\, n}}. \tag{16}$$

Proof. Based on equation 15, we have $\|\mathcal{U}(S)\|^2 \leq \frac{T_G+2}{\lambda}$. Hence the assumption $\|U(S)\| \leq c$ of Theorem 1 holds with $c = \sqrt{(T_G + 2)/\lambda}$, and the 0-1 loss takes values in $[0, 1]$. The vector $w = U(S)$ depends on the data only through S, so it is independent of V, and Theorem 1 applies to it as a fixed vector. Since $\|w\| \leq c$,

$$4c\|w\| + c^2 \;\leq\; 5c^2 \;=\; \frac{5\,(T_G + 2)}{\lambda},$$

and substituting this into equation 1 gives equation 16. □

Beyond ReLU: Leaky ReLU and ReLU6. The same analysis applies to Leaky ReLU networks, whose activation can also be written through equivalent constraints (Appendix I). The flow must be redefined to account for the negative slope, but the flow properties of Theorems 2 and 3, and hence Theorems 5 and 6, carry over unchanged (Lemma 5).

3.1 MULTI-CLASS CLASSIFICATION

For labels $y_i \in \{1, \dots, \mathcal{K}\}$, we use a separate output vector v_κ for each class κ and the multi-class hinge loss; the resulting problem and its KKT conditions are given in Appendix J.

Lemma 2. *At a KKT point of the multi-class problem, $\sum_{\kappa=1}^{\mathcal{K}} \|v_\kappa\|^2 \le 1/\lambda$. (The proof can be found in Appendix K.)*

Theorem 7. *In the setting of Theorem 6, let $y_i \in \{1, \dots, \mathcal{K}\}$, let $\mathcal{U}(S) = \big[\mathrm{vec}(U^{(1)}(S)), \dots, \mathrm{vec}(U^{(L-1)}(S)), v_1(S), \dots, v_{\mathcal{K}}(S)\big]$ be a KKT point of the multi-class problem, and let $\ell(w, Z_i) = \mathbf{1}\big[y_i \neq \arg\max_\kappa \langle x_i^{(L)}, v_\kappa\rangle\big]$ be the multi-class 0-1 loss. Set $w := \mathcal{U}(S)$. Then for every $\delta \in (0,1)$, with probability at least $1-\delta$ over the draw of V,*

$$\hat{\mathcal{L}}(w, V) - \mathbb{E}_{S|W=w}\big[\hat{\mathcal{L}}(w, S)\big] \;\le\; \sqrt{\frac{\ln(1/\delta)}{2r}} \;+\; \sqrt{\frac{5\,(T_G+2)}{4\,\epsilon^2 \lambda\, n}}. \tag{17}$$

Proof. Lemma 2 together with Theorems 2, 3 and 5 gives $\|\mathcal{U}(S)\|^2 \le (T_G+2)/\lambda$ for every training set. The rest is identical to the proof of Theorem 6, since the multi-class 0-1 loss also takes values in $[0,1]$. □

3.2 REGRESSION EXTENSION

We extend our analysis to regression settings in which the response variable satisfies $y_i \in \mathbb{R}$. The KKT-based framework previously developed for classification carries over naturally to this context. While the objective function is modified from hinge loss to squared loss, the structural constraints induced by the ReLU activations remain unchanged, allowing analogous properties to be established. For the purpose of generalization analysis, we adopt a truncated squared loss, bounded by 1, since Theorem 1 requires a loss with values in $[0,1]$. Concretely, a deep ReLU network is trained to minimize the mean squared error between the predictions $\left\langle x_i^{(L)}, v \right\rangle$ and the targets y_i, subject to the same piecewise-linear constraints. The resulting KKT conditions give rise to comparable relationships among the weight norms, which in turn yield generalization bounds for the truncated loss. This demonstrates that the proposed approach extends consistently across both classification and regression settings. Here we analyze the following problem:

$$\begin{aligned} \underset{U^{(l)}, v, X^{(l)}}{\text{minimize}} \quad & \frac{1}{2n}\sum_{i=1}^{n}\left(y_i - \left\langle x_i^{(L)}, v\right\rangle\right)^2 + \frac{\lambda}{2}\left(\sum_{l=1}^{L-1}\left\|U^{(l)}\right\|_F^2 + \|v\|^2\right) \\ \text{subject to} \quad & x_{ji}^{(l)} = \sigma(\left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)}\right\rangle), \forall i = 1, \dots, n, \forall j = 1, \dots, m_{l-1} \forall l = 2, \dots, L, \end{aligned}$$

also, $x_i^{(1)} = x_i$ and $m_0 = d$. For brevity, we defer the full constrained formulation and the detailed optimality conditions to Appendix L

The following lemma shows that data-independent upper bound on the norm of v in terms of the regularization parameter λ, the sample size n, and the norm of the response vector y.

Lemma 3. *Let v denote the weight vector of the output layer, in the neural network regression problem and using the notation $v(S)$ to denote the dependence on S, dataset of n i.i.d. samples. Assuming the stated constraints, we obtain, $\|v(S)\| \le \frac{1}{2\sqrt{n\lambda}}\|y\|$.*

(The proof can be found in Appendix M)

As in the classification setting, the KKT conditions yield a fundamental norm equality between the last hidden layer and the output layer:

Lemma 4. *Let $U^{(L-1)}$ denote the weight matrices of the $L-1$-th neuron of assumed DAG and v denotes the weight vector of the output layer. Under the specified model constraints, we have $\left\|U^{(L-1)}\right\|_F^2 = \|v\|^2$.*

(The proof can be found in Appendix N)

The flow definition in equation 4 and the properties established in Theorems 2 and 3 for the classification setting extend directly to the regression case. To avoid redundancy, we omit their proofs in

this context. The only modification arises in the sink node property, specifically in equation 5, as detailed in Lemma 4. Accordingly, by replacing equation 13 with the updated sink node condition in the total flow formulation, $\sum_{p\in\pi_{L-1}} F^{(p\to q)} \le \left\|U^{(L-1)}\right\|_F^2$, the optimal solution derived in Theorem 5 for classification takes the following form in the regression setting,$\sum_{(p,q)\in E} F^{(p\to q)} \le T_G\lambda \left\|U^{(L-1)}\right\|_F^2$. This follows from the identity $\sum_{P\in\mathcal{P}} \delta_P = \left\|U^{(L-1)}\right\|_F^2$ in that theorem. The following theorem provides a generalization bound for regression with a capped squared loss, which takes values in $[0, 1]$.

Theorem 8. *Consider the setting of Theorem 1, with a training subset S of size n and a validation subset V of size r, where each $Z_i = (x_i, y_i)$ consists of an input $x_i \in \mathbb{R}^d$ and a label $y_i \in \mathbb{R}$ with $|y_i| \le B$. Let $\mathcal{U}(S)$ be a KKT point of the regularized regression problem trained on S (the weight matrices of all nodes together with the output weight vector v), and let $W \mid S \sim \mathcal{N}\big(\mathcal{U}(S), \epsilon^2 I\big)$. For a data point $Z_i = (x_i, y_i)$, let $\ell(w, Z_i) = \min\big((y_i - \langle x_i^{(L)}, v\rangle)^2,\ 1\big)$ be the capped squared loss, where $x_i^{(L)}$ is the output of the last node and v the output weight vector of w. Set $w := \mathcal{U}(S)$. Then for every $\delta \in (0, 1)$, with probability at least $1 - \delta$ over the draw of V,*

$$\hat{\mathcal{L}}(w, V) - \mathbb{E}_{S|W=w}\big[\hat{\mathcal{L}}(w, S)\big] \ \le\ \sqrt{\frac{\ln(1/\delta)}{2r}} \ +\ \sqrt{\frac{5\,(T_G + 2)\,B^2}{16\,\epsilon^2\lambda\,n}}. \tag{18}$$

(The proof can be found in Appendix O)

In the binary classification setting of Theorem 6, each label satisfies $y_i \in \{-1, +1\}$, so the label bound of Theorem 8 holds with $B = 1$.

4 Concluding Remarks

We introduced a KKT-based flow interpretation for DAG-structured networks, showing that weight norms propagate like flow and that the total parameter norm is bounded in terms of the longest path length T_G. Combined with a held-out validation set and a change-of-measure argument, this yields high-probability bounds whose training term is of order $O\big(\sqrt{T_G/(\epsilon^2\lambda n)}\big)$ for classification and regression, avoiding intractable path-norm computations. Our bounds are conditional on training reaching a KKT point of the regularized objective. Extending them to approximate stationary points, such as those reached by stochastic training with a finite step size, is an important direction for future work. Further directions include tightening the divergence estimate, extending the analysis to other loss functions, and investigating whether T_G correlates with generalization in modern architectures.

AI Use Statement

In this work, we used generative AI tools solely to edit the manuscript to improve readability and language. We did not use generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, assist with proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, clean or reformat datasets, perform qualitative or thematic data analysis, or interpret results. We reviewed all AI-assisted text and take responsibility for the final content of this work.

References

[1] Martin Anthony and Peter L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 2009.

[2] Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30, 2017.

[3] Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nathan Srebro. A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. In *Proceedings of the 35th International Conference on Machine Learning*, pages 4080–4089. PMLR, 2017.

[4] Noah Golowich, Alexander Rakhlin, and Ohad Shamir. Size-independent sample complexity of neural networks. In *Conference On Learning Theory*, pages 297–299. PMLR, 2018.

[5] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. Norm-based capacity control in neural networks. In *Proceedings of The 28th Conference on Learning Theory*, pages 1376–1401. PMLR, 2015.

[6] Antoine Gonon, Nicolas Brisebarre, Elisa Riccietti, and Rémi Gribonval. A path-norm toolkit for modern networks: Consequences, promises and challenges. In *International Conference on Learning Representations (ICLR)*, 2024.

[7] Joachim Bona-Pellissier, François Malgouyres, and François Bachoc. Local identifiability of deep relu neural networks: the theory. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[8] Pierre Stock and Rémi Gribonval. An embedding of relu networks and an analysis of their identifiability. *Constructive Approximation*, 57(2):853–899, 2023.

[9] Sibylle Marcotte, Rémi Gribonval, and Gabriel Peyré. Abide by the law and follow the flow: Conservation laws for gradient flows. *arXiv preprint arXiv:2307.00144*, 2023.

[10] Vaishnavh Nagarajan and J Zico Kolter. Uniform convergence may be unable to explain generalization in deep learning. *Advances in neural information processing systems*, 32, 2019.

[11] Amartya Sanyal, Philip HS Torr, and Puneet K Dokania. Stable rank normalization for improved generalization in neural networks and gans. *arXiv preprint arXiv:1906.04659*, 2019.

[12] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19(70):1–57, 2018.

[13] Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1225–1234, 2016.

[14] Stanisław Jastrzębski, Zachary Kenton, Devansh Arpit, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos Storkey. Three factors influencing minima in sgd. *arXiv preprint arXiv:1711.04623*, 2017.

[15] Umut Simsekli, Levent Sagun, and Mert Gürbüzbalaban. Fractional underdamped langevin dynamics: Retargeting sgd with momentum under heavy-tailed gradient noise. *arXiv preprint arXiv:1902.02440*, 2019.

[16] Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[17] Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. Learning and generalization in overparameterized neural networks, going beyond two layers. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[18] Peter L Bartlett. The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. *IEEE Transactions on Information Theory*, 44(2):525–536, 1998.

[19] Simon S Du, Wei Hu, and Jason D Lee. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced. *Advances in neural information processing systems*, 31, 2018.

[20] Daniel Kunin, Javier Sagastuy-Brena, Surya Ganguli, Daniel LK Yamins, and Hidenori Tanaka. Neural mechanics: Symmetry and broken conservation laws in deep learning dynamics. In *ICLR*, 2021.

[21] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[22] Monroe D. Donsker and S. R. Srinivasa Varadhan. Asymptotic evaluation of certain Markov process expectations for large time, I. *Communications on Pure and Applied Mathematics*, 28(1):1–47, 1975.

[23] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1993.

[24] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press, 3rd edition, 2009.

[25] Christos H. Papadimitriou and Kenneth Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Prentice Hall, 1982.

[26] Miguel Á. Carreira-Perpiñán and Weiran Wang. Distributed optimization of deeply nested systems. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 33 of *Proceedings of Machine Learning Research*, pages 10–19, 2014.

[27] Armin Askari, Geoffrey Negiar, Rajiv Sambharya, and Laurent El Ghaoui. Lifted neural networks. *arXiv preprint arXiv:1805.01532*, 2018.

[28] Fangda Gu, Armin Askari, and Laurent El Ghaoui. Fenchel lifted networks: A Lagrange relaxation of neural network training. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 108 of *Proceedings of Machine Learning Research*, pages 3362–3371, 2020.

[29] Dominic Yang, Prasanna Balaprakash, and Sven Leyffer. Modeling design and control problems involving neural network surrogates. *arXiv preprint arXiv:2111.10489*, 2021.

Supplementary Material: Max-Total-Flow Generalization Bounds for DAG-Structured ReLU Deep Neural Networks

A Additional Evidence

Table 2 reports $\Phi(U)$ as the sum over all input-to-output paths, where each term is given by the product of weights along the path (found in the analysis of (5; 6)) for 75 pre-trained TorchVision models. These astronomically large values—often exceeding 10^{100}—demonstrate that generalization bounds depending on path-norm become vacuous in practice, even for moderately deep networks. Most of these models contain parameter-free identity shortcuts, normalization layers and biases, and were trained with cross-entropy, so they lie outside the setting of our theorems; Table 2 only illustrates the size of path-norm quantities in trained networks.

Table 2: For pre-trained TorchVision models, we compute $\Phi(U)$ as the sum over all input-to-output paths, where each term is given by the product of weights along the path. This quantity dominates the generalization bounds of (5; 6) and related path-norm-based approaches. The extremely large values suggest that such bounds can become vacuous for practical deep networks.

Deep Network	T_G	$\Phi(U)$	**Deep Network**	T_G	$\Phi(U)$
alexnet	7	7.5×10^{11}	densenet169	337	8.0×10^{158}
convnext_large	148	2.0×10^{229}	efficientnet_b0	130	3.7×10^{163}
densenet201	401	6.5×10^{179}	efficientnet_b1	184	5.5×10^{255}
efficientnet_b7	436	∞	efficientnet_b2	184	6.0×10^{224}
efficientnet_v2_l	556	∞	efficientnet_b3	208	5.1×10^{274}
googlenet	42	1.6×10^{50}	efficientnet_b4	256	∞
inception_v3	94	9.5×10^{98}	efficientnet_b5	310	∞
mnasnet1_3	104	5.7×10^{138}	efficientnet_b6	358	∞
mobilenet_v2	104	5.1×10^{131}	efficientnet_v2_m	402	∞
mobilenet_v3_large	109	6.8×10^{135}	efficientnet_v2_s	280	∞
regnet_x_32gf	140	4.3×10^{203}	mnasnet0_5	104	3.4×10^{197}
regnet_y_32gf	162	1.5×10^{218}	mnasnet0_75	104	2.9×10^{133}
resnet152	302	∞	mnasnet1_0	104	9.0×10^{115}
resnext101_64x4d	200	1.7×10^{272}	mobilenet_v3_small	87	1.1×10^{98}
squeezenet1_1	17	2.0×10^{21}	regnet_x_16gf	134	2.7×10^{203}
swin_b	105	4.9×10^{155}	regnet_x_1_6gf	120	1.3×10^{171}
swin_v2_b	105	1.2×10^{154}	regnet_x_3_2gf	124	1.8×10^{199}
vgg19	18	1.2×10^{25}	regnet_x_400mf	104	6.2×10^{141}
vit_l_32	98	2.2×10^{135}	regnet_x_800mf	112	1.3×10^{156}
wide_resnet101_2	200	2.0×10^{268}	regnet_x_8gf	132	2.2×10^{201}
convnext_base	148	1.8×10^{209}	regnet_y_128gf	168	∞
convnext_small	148	8.1×10^{191}	regnet_y_16gf	146	3.9×10^{223}
convnext_tiny	76	1.1×10^{100}	regnet_y_1_6gf	122	2.8×10^{199}
densenet121	241	3.1×10^{130}	regnet_y_3_2gf	126	5.4×10^{210}
densenet161	321	1.1×10^{160}	regnet_y_400mf	104	7.2×10^{143}
regnet_y_800mf	112	1.9×10^{160}	ssd300_vgg16	35	2.2×10^{35}
regnet_y_8gf	134	3.5×10^{205}	swin_s	105	2.1×10^{153}
resnet101	200	1.5×10^{275}	swin_t	105	8.4×10^{88}
resnet18	62	2.5×10^{46}	swin_v2_s	105	3.9×10^{150}
resnet34	110	3.2×10^{82}	swin_v2_t	105	7.1×10^{87}
resnet50	122	6.8×10^{136}	vgg11	12	3.8×10^{13}
resnext101_32x8d	200	1.4×10^{276}	vgg11_bn	12	2.6×10^{23}
resnext50_32x4d	122	2.9×10^{133}	vgg13	14	7.9×10^{15}
shufflenet_v2_x0_5	50	2.1×10^{77}	vgg13_bn	14	3.5×10^{27}
shufflenet_v2_x1_0	52	3.5×10^{101}	vgg16	16	2.2×10^{20}
shufflenet_v2_x1_5	54	1.1×10^{111}	vgg16_bn	16	9.8×10^{35}
shufflenet_v2_x2_0	56	3.2×10^{119}	vgg19_bn	18	1.7×10^{43}
squeezenet1_0	17	9.5×10^{20}			

B Proof of Theorem 1

Proof. Fix w. The proof splits the left-hand side of equation 1 into a validation part and a training part, bounds the first by concentration and the second by a change-of-measure argument, and finally makes the resulting divergence explicit using the Gaussian form of $W \mid S$.

Since $\mathcal{L}(w, \mathcal{P})$ does not depend on the data, $\mathbb{E}_{S|W=w}[\mathcal{L}(w, \mathcal{P})] = \mathcal{L}(w, \mathcal{P})$, and therefore

$$\hat{\mathcal{L}}(w, V) - \mathbb{E}_{S|W=w}\big[\hat{\mathcal{L}}(w, S)\big] = \underbrace{\hat{\mathcal{L}}(w, V) - \mathcal{L}(w, \mathcal{P})}_{\text{(A)}} + \underbrace{\mathbb{E}_{S|W=w}\big[\mathcal{L}(w, \mathcal{P}) - \hat{\mathcal{L}}(w, S)\big]}_{\text{(B)}}. \quad (19)$$

Term (A) compares the validation loss with the population risk; term (B) is the expected training gap, given that the algorithm produced w.

Since w is fixed and the samples in V are i.i.d., the losses $\ell(w, Z_i)$, $Z_i \in V$, are i.i.d., take values in $[0, 1]$, and have mean $\mathcal{L}(w, \mathcal{P})$. Hoeffding's inequality (21) gives $\mathbb{P}\big(\hat{\mathcal{L}}(w, V) - \mathcal{L}(w, \mathcal{P}) > \gamma\big) \leq \exp(-2r\gamma^2)$ for every $\gamma > 0$. Setting the right-hand side equal to δ shows that, with probability at least $1 - \delta$ over V,

$$\text{(A)} \leq \sqrt{\frac{\ln(1/\delta)}{2r}}. \quad (20)$$

Term (B) is an expectation under $\mathcal{P}_{S|W=w}$, the distribution of the training subset given that the algorithm produced w, which differs from the data distribution $\mathcal{P}_S$. We change measure from $\mathcal{P}_S$ to $\mathcal{P}_{S|W=w}$. Let

$$g(S) := \mathcal{L}(w, \mathcal{P}) - \hat{\mathcal{L}}(w, S) = \frac{1}{n} \sum_{Z_i \in S} \big(\mathcal{L}(w, \mathcal{P}) - \ell(w, Z_i)\big).$$

Under $\mathcal{P}_S$, the n summands are independent, have mean zero, and take values in an interval of length 1. By Hoeffding's lemma, each has log-moment-generating function at most $s^2/8$; applying this with $s = t/n$ and summing over the n independent terms gives

$$\ln \mathbb{E}_{\mathcal{P}_S}\big[e^{tg(S)}\big] \leq \frac{t^2}{8n}, \qquad t \in \mathbb{R}. \quad (21)$$

The Donsker–Varadhan inequality states that $\mathbb{E}_Q[f] \leq \mathrm{KL}(Q \,\|\, P) + \ln \mathbb{E}_P[e^f]$ for any measurable f and any $Q \ll P$. Apply it with $Q = \mathcal{P}_{S|W=w}$, $P = \mathcal{P}_S$ (here $Q \ll P$, since the Gaussian density of $W \mid S$ is positive) and $f = tg$ with $t > 0$. Using equation 21 and dividing by t,

$$\text{(B)} = \mathbb{E}_{S|W=w}[g(S)] \leq \frac{\mathrm{KL}\big(\mathcal{P}_{S|W=w} \,\|\, \mathcal{P}_S\big)}{t} + \frac{t}{8n}.$$

Choosing $t = \sqrt{8n\,\mathrm{KL}(\mathcal{P}_{S|W=w} \| \mathcal{P}_S)}$, which minimizes the right-hand side, gives

$$\text{(B)} \leq \sqrt{\frac{\mathrm{KL}\big(\mathcal{P}_{S|W=w} \,\|\, \mathcal{P}_S\big)}{2n}}. \quad (22)$$

It remains to bound $\mathrm{KL}(\mathcal{P}_{S|W=w} \| \mathcal{P}_S)$ using the Gaussian form of $W \mid S$. Write $p_{W|S}$ for the conditional density of W and $p_W(w) = \mathbb{E}_S[p_{W|S}(w \mid S)]$ for its marginal. By Bayes' rule, and inserting the reference density q of $\mathcal{N}(0, \epsilon^2 I_d)$,

$$\mathrm{KL}\big(\mathcal{P}_{S|W=w} \,\|\, \mathcal{P}_S\big) = \mathbb{E}_{S|W=w}\left[\ln \frac{p_{W|S}(w \mid S)}{p_W(w)}\right] = \underbrace{\mathbb{E}_{S|W=w}\left[\ln \frac{p_{W|S}(w \mid S)}{q(w)}\right]}_{\text{(C)}} + \underbrace{\ln \frac{q(w)}{p_W(w)}}_{\text{(D)}}.$$

Since both $p_{W|S}(\cdot \mid S)$ and q are Gaussian with covariance $\epsilon^2 I_d$, their normalizing constants cancel, and

$$\ln \frac{p_{W|S}(w \mid S)}{q(w)} = \frac{\|w\|^2 - \|w - U(S)\|^2}{2\epsilon^2} = \frac{2\langle w, U(S)\rangle - \|U(S)\|^2}{2\epsilon^2} \leq \frac{c\|w\|}{\epsilon^2},$$

by Cauchy–Schwarz, $\|U(S)\| \le c$ and $-\|U(S)\|^2 \le 0$. Hence $(\mathrm{C}) \le c\|w\|/\epsilon^2$. For (D), the triangle inequality gives $\|w - U(S)\| \le \|w\| + c$ for every S, so $p_W(w) \ge (2\pi\epsilon^2)^{-d/2} \exp\big(-(\|w\| + c)^2/(2\epsilon^2)\big)$ and

$$(\mathrm{D}) \le \frac{(\|w\| + c)^2 - \|w\|^2}{2\epsilon^2} = \frac{2c\|w\| + c^2}{2\epsilon^2}.$$

Adding the two estimates,

$$\mathrm{KL}\big(\mathcal{P}_{S|W=w} \,\|\, \mathcal{P}_S\big) \le \frac{2c\|w\| + \frac{1}{2}c^2}{\epsilon^2}. \tag{23}$$

Substituting equation 23 into equation 22 gives $(\mathrm{B}) \le \sqrt{(4c\|w\| + c^2)/(4n\epsilon^2)}$. Combining this with equation 20 in the decomposition equation 19 proves equation 1 for every fixed w. □

C KKT CONDITIONS OF THE RELU NEURAL NETWORK CLASSIFICATION

The ReLU activation is replaced by the following three equivalent constraints in order to ensure that derivatives are well-defined everywhere:

$$x_{ji}^{(l)} \ge 0, \tag{24}$$

$$x_{ji}^{(l)} \ge \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle, \tag{25}$$

$$x_{ji}^{(l)} \left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle \right) = 0. \tag{26}$$

The problem 2-3, can be written as:

$$\underset{\xi_i, U^{(l)}, v, X^{(l)}}{\text{minimize}} \quad \frac{1}{n}\sum_{i=1}^{n} \xi_i + \frac{\lambda}{2}\left(\sum_{l=1}^{L-1} \left\| U^{(l)} \right\|_F^2 + \|v\|^2 \right) \tag{27}$$

$$\text{subject to} \quad \xi_i \ge 0, \tag{28}$$

$$y_i \langle x_i^{(L)}, v \rangle \ge 1 - \xi_i, \tag{29}$$

$$x_{ji}^{(l)} \ge 0, \tag{30}$$

$$x_{ji}^{(l)} \ge \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle, \tag{31}$$

$$x_{ji}^{(l)} \left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle \right) = 0. \tag{32}$$

Replacing activation functions by equivalent constraints on auxiliary activation variables is a standard device, both in the study of neural-network training (26; 27; 28) and in optimization over trained networks (29). Since constraints equation 24–equation 26 are equivalent to the ReLU, the activations are uniquely determined by the weights, so the lifted problem 27-32 and problem 2-3have the same global and local minimizers.

The Lagrangian function of the problem as follows:

$$\begin{aligned}
\mathscr{L}(\xi_i, U^{(l)}, v, X^{(l)}) = {} & \frac{1}{n}\sum_{i=1}^{n} \xi_i + \frac{\lambda}{2}\sum_{l=1}^{L-1} \left\| U^{(l)} \right\|_F^2 + \frac{\lambda}{2}\|v\|^2 - \sum_{i=1}^{n} \alpha_i \xi_i \\
& - \sum_{i=1}^{n} \beta_i \left(y_i \langle x_i^{(L)}, v \rangle - 1 + \xi_i \right) - \sum_{l=2}^{L}\sum_{i=1}^{n}\sum_{j=1}^{m_{l-1}} \gamma_{ji}^{(l)} x_{ji}^{(l)} \\
& - \sum_{l=2}^{L}\sum_{i=1}^{n}\sum_{j=1}^{m_{l-1}} \theta_{ji}^{(l)} \left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle \right) \\
& + \sum_{l=2}^{L}\sum_{i=1}^{n}\sum_{j=1}^{m_{l-1}} \eta_{ji}^{(l)} x_{ji}^{(l)} \left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle \right),
\end{aligned}$$

and the stationary conditions of KKT are:

$$\frac{\partial \mathscr{L}}{\partial v} = \lambda v - \sum_{i=1}^{n} \beta_i y_i x_i^{(L)} = 0, \tag{33}$$

$$\frac{\partial \mathscr{L}}{\partial u_{tk}^{(l)}} = \lambda u_{tk}^{(l)} + \sum_{i=1}^{n} \theta_{ki}^{(l+1)} x_{ti}^{\pi_l} - \sum_{i=1}^{n} \eta_{ki}^{(l+1)} x_{ki}^{(l+1)} x_{ti}^{\pi_l} = 0, \forall t = 1, \ldots, M_l, \forall k = 1, \ldots, m_l \tag{34}$$

$$\frac{\partial \mathscr{L}}{\partial \xi_i} = \frac{1}{n} - \alpha_i - \beta_i = 0, \forall i = 1, \ldots, n \tag{35}$$

$$\begin{aligned}\frac{\partial \mathscr{L}}{\partial x_{tk}^{(l)}} &= -\gamma_{tk}^{(l)} - \theta_{tk}^{(l)} + \eta_{tk}^{(l)} \left(2x_{tk}^{(l)} - \left\langle x_k^{\pi_{l-1}}, u_t^{(l-1)} \right\rangle\right) \\ &\quad + \sum_{q \in \phi_{l-1}} \sum_{j=1}^{m_q} \left(\theta_{jk}^{(q+1)} - \eta_{jk}^{(q+1)} x_{jk}^{(q+1)}\right) u_{tj}^{(q)} = 0, \forall t = 1, \ldots, m_l, \forall k = 1, \ldots, n\end{aligned} \tag{36}$$

$$\frac{\partial \mathscr{L}}{\partial x_{tk}^{(L)}} = -\beta_k y_k v_t - \gamma_{tk}^{(L)} - \theta_{tk}^{(L)} + \eta_{tk}^{(L)} \left(2x_{tk}^{(L)} - \left\langle x_k^{\pi_{L-1}}, u_t^{(L-1)} \right\rangle\right) = 0. \tag{37}$$

In addition, the complementary slackness conditions of KKT for $i = 1, \ldots, n$, and $l = 2, \ldots, L$ are:

$$\alpha_i \xi_i = 0, \tag{38}$$

$$\beta_i \left(1 - \xi_i - y_i \left\langle x_i^{(L)}, v \right\rangle\right) = 0, \tag{39}$$

$$\gamma_{ji}^{(l)} \, x_{ji}^{(l)} = 0, \tag{40}$$

$$\theta_{ji}^{(l)} \left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle\right) = 0, \tag{41}$$

Moreover, the dual feasibility conditions of KKT for $i = 1, \ldots, n$ and $j = 1, \ldots, m_{l-1}$ where $l = 2, \ldots, L$ are:

$$\begin{aligned}&\alpha_i, \beta_i, \gamma_{ji}^{(l)}, \theta_{ji}^{(l)} \geq 0, \\ &\eta_{ji}^{(l)} \in \mathbb{R},\end{aligned} \tag{42}$$

also, eqs.equation 24-equation 26 are primal conditions.

D Proof of Theorem 2

As we assumed that the graph has one source and one sink node, the only node q, such that $\phi_q = \varnothing$, is $q = L - 1$. By multiplying eq.equation 34 by $u_{tk}^{(L-1)}$, summing over t, and using eq. equation 41, we obtain:

$$\begin{aligned}\lambda \|u_k^{(L-1)}\|^2 &= \sum_{t=1}^{M_{L-1}} \sum_{i=1}^{n} \left[\eta_{ki}^{(L)} x_{ki}^{(L)} x_{ti}^{\pi_{L-1}} u_{tk}^{(L-1)} - \theta_{ki}^{(L)} x_{ti}^{\pi_{L-1}} u_{tk}^{(L-1)}\right] \\ &= \sum_{i=1}^{n} \left[\eta_{ki}^{(L)} x_{ki}^{(L)} \left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)} \right\rangle - \theta_{ki}^{(L)} \left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)} \right\rangle\right] \\ &= \sum_{i=1}^{n} \left[\eta_{ki}^{(L)} \left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)} \right\rangle - \theta_{ki}^{(L)}\right] x_{ki}^{(L)}.\end{aligned} \tag{43}$$

Summing over k, we have

$$\lambda \left\| U^{(L-1)} \right\|_F^2 = \sum_{i=1}^{n} \sum_{k=1}^{m_{L-1}} \left[\eta_{ki}^{(L)} \left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)} \right\rangle - \theta_{ki}^{(L)}\right] x_{ki}^{(L)}.$$

Considering the primal feasibility constrains equation 24-equation 26, we obtain

$$\lambda \left\| U^{(L-1)} \right\|_F^2 = \sum_{(k,i)\in T_+} \left[\eta_{ki}^{(L)} \left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)} \right\rangle - \theta_{ki}^{(L)} \right] x_{ki}^{(L)} + \underbrace{\sum_{(k,i)\in T_-} \left[\eta_{ki}^{(L)} \left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)} \right\rangle - \theta_{ki}^{(L)} \right] x_{ki}^{(L)}}_{0},$$

where

$$T_+ = \{(k,i) | x_{ki}^{(L)} = \left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)} \right\rangle > 0\}$$

and

$$T_- = \{(k,i) |\, x_{ki}^{(L)} = 0, \,\&\, \left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)} \right\rangle \leq 0 \,\}$$

Then by applying the stationarity conditions equation 35 and equation 37 along with the complementary slackness conditions equation 38 and equation 39, we obtain:

$$\begin{aligned}
\lambda \left\| U^{(L-1)} \right\|_F^2 &= \sum_{(k,i)\in T_+} \left[\eta_{ki}^{(L)} \left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)} \right\rangle - \theta_{ki}^{(L)} \right] x_{ki}^{(L)} \\
&= \sum_{(k,i)\in T_+} \left[\gamma_{ki}^{(L)} + \beta_i y_i v_k \right] x_{ki}^{(L)} = \sum_{(k,i)\in T_+} \beta_i y_i v_k x_{ki}^{(L)} \\
&= \sum_{i=1}^{n} \sum_{k=1}^{m_{L-1}} \beta_i y_i v_k x_{ki}^{(L)} = \sum_{i=1}^{n} \beta_i y_i \left\langle x_i^{(L)}, v \right\rangle \\
&= \sum_{i=1}^{n} \beta_i (1-\xi_i) = \sum_{i=1}^{n} (\frac{1}{n} - \alpha_i)(1-\xi_i) \\
&= \sum_{i=1}^{n} (\frac{1}{n} - \frac{1}{n}\xi_i - \alpha_i + \alpha_i \xi_i) \leq \sum_{i=1}^{n} \frac{1}{n} = 1,
\end{aligned} \tag{44}$$

which is equivalent to eq. equation 5.

In addition, if $\phi_q \neq \varnothing$ multiplying eq. equation 34 by $u_{tk}^{(q)}$ and summing over t, we have:

$$\begin{aligned}
\lambda \| u_k^{(q)} \|^2 &= \sum_{t=1}^{M_q} \sum_{i=1}^{n} \left[\eta_{ki}^{(q+1)} x_{ki}^{(q+1)} x_{ti}^{\pi_q} u_{tk}^{(q)} - \theta_{ki}^{(q+1)} x_{ti}^{\pi_q} u_{tk}^{(q)} \right] \\
&= \sum_{i=1}^{n} \left[\eta_{ki}^{(q+1)} x_{ki}^{(q+1)} \left\langle x_i^{\pi_q}, u_k^{(q)} \right\rangle - \theta_{ki}^{(q+1)} \left\langle x_i^{\pi_q}, u_k^{(q)} \right\rangle \right] \\
&= \sum_{i=1}^{n} \left[\eta_{ki}^{(q+1)} \left\langle x_i^{\pi_q}, u_k^{(q)} \right\rangle - \theta_{ki}^{(q+1)} \right] x_{ki}^{(q+1)}.
\end{aligned}$$

Therefore,

$$\lambda \| U^{(q)} \|_F^2 = \sum_{k=1}^{m_q} \sum_{i=1}^{n} \left[\eta_{ki}^{(q+1)} \left\langle x_i^{\pi_q}, u_k^{(q)} \right\rangle - \theta_{ki}^{(q+1)} \right] x_{ki}^{(q+1)}. \tag{45}$$

Applying the primal feasibility condition equation 26, equation equation 45 can be separated into two parts:

$$\lambda \left\| U^{(q)} \right\|_F^2 = \sum_{(k,i)\in T_+} \left[\eta_{ki}^{(q+1)} \left\langle x_i^{\pi_q}, u_k^{(q)} \right\rangle - \theta_{ki}^{(q+1)} \right] x_{ki}^{(q+1)} + \underbrace{\sum_{(k,i)\in T_-} \left[\eta_{ki}^{(q+1)} \left\langle x_i^{\pi_q}, u_k^{(q)} \right\rangle - \theta_{ki}^{(q+1)} \right] x_{ki}^{(q+1)}}_{0}, \tag{46}$$

where $T_+ = \{(k,i)|x_{ki}^{(q+1)} = \left\langle x_i^{(q)}, u_k^{(q)} \right\rangle > 0\}$ and $T_- = \{(k,i)|x_{ki}^{(q)} = 0, \ \& \ \left\langle x_i^{(q)}, u_k^{(q)} \right\rangle \leq 0\}$. Therefore, using eqs. equation 36 and equation 40, we obtain:

$$\begin{aligned}
\lambda\|U^{(q)}\|_F^2 &= \sum_{(k,i)\in T_+} \left[\eta_{ki}^{(q+1)} \left\langle x_i^{\pi_q}, u_k^{(q)} \right\rangle - \theta_{ki}^{(q+1)}\right] x_{ki}^{(q+1)} \\
&= \sum_{k=1}^{m_q}\sum_{i=1}^{n} \left[\eta_{ki}^{(q+1)} \left\langle x_i^{\pi_q}, u_k^{(q)} \right\rangle - \theta_{ki}^{(q+1)}\right] x_{ki}^{(q+1)} \\
&= \sum_{k=1}^{m_q}\sum_{i=1}^{n} \left[\gamma_{ki}^{(q+1)} + \sum_{c\in\phi_q}\sum_{j=1}^{m_c} \left(\eta_{ji}^{(c+1)} x_{ji}^{(c+1)} - \theta_{ji}^{(c+1)}\right) u_{kj}^{(c)}\right] x_{ki}^{(q+1)} \\
&= \sum_{c\in\phi_q}\sum_{k=1}^{m_q}\sum_{i=1}^{n} \left[\sum_{j=1}^{m_c} \left(\eta_{ji}^{(c+1)} x_{ji}^{(c+1)} - \theta_{ji}^{(c+1)}\right) u_{kj}^{(c)}\right] x_{ki}^{(q+1)} \\
&= \sum_{c\in\phi_q} F^{(q\to c)}
\end{aligned} \tag{47}$$

Let $p \in \pi_q$. Equation equation 7 can be derived by multiplying equation equation 34 by $u_{kj}^{(q)}$ and summing over k such that $x_{ki}^{\pi_q}$ corresponds to the k-th component of $x_i^{(p+1)}$, i.e., $x_{ki}^{\pi_q} = x_{ki}^{(p+1)}$. Since $x_i^{(p+1)}$ has dimension m_p, the summation over k is effectively from 1 to m_p, and $x_{ki}^{\pi_q}$ can be replaced by $x_i^{(p+1)}$. Then we have

$$\begin{aligned}
\lambda\sum_{k=1}^{m_p}(u_{kj}^{(q)})^2 &= \sum_{k=1}^{m_p}\sum_{i=1}^{n} \left(\eta_{ji}^{(q+1)} x_{ji}^{(q+1)} - \theta_{ji}^{(q+1)}\right) x_{ki}^{\Pi_q} u_{kj}^{(q)} \\
&= \sum_{k=1}^{m_p}\sum_{i=1}^{n} \left(\eta_{ji}^{(q+1)} x_{ji}^{(q+1)} - \theta_{ji}^{(q+1)}\right) x_{ki}^{(p+1)} u_{kj}^{(q)},
\end{aligned} \tag{48}$$

therefore,

$$\begin{aligned}
\lambda\sum_{k=1}^{m_p}\sum_{j=1}^{m_q}(u_{kj}^{(q)})^2 &= \sum_{k=1}^{m_p}\sum_{j=1}^{m_q}\sum_{i=1}^{n} \left(\eta_{ji}^{(q+1)} x_{ji}^{(q+1)} - \theta_{ji}^{(q+1)}\right) x_{ki}^{(p+1)} u_{kj}^{(q)} \\
&= \sum_{k=1}^{m_p}\sum_{i=1}^{n}\sum_{j=1}^{m_q} \left(\eta_{ki}^{(q+1)} x_{ki}^{(q+1)} - \theta_{ki}^{(q+1)}\right) x_{ki}^{(p+1)} u_{tk}^{(q)} = F^{(p\to q)} \geq 0.
\end{aligned} \tag{49}$$

E PROOF OF THEOREM3

Lets $\phi_q \neq \varnothing$ and $p \in \pi_q$, we have:

$$\begin{aligned}
\sum_{p\in\pi_q} F^{(p\to q)} &= \sum_{p\in\pi_q}\sum_{k=1}^{m_p}\sum_{i=1}^{n} \left[\sum_{j=1}^{m_q} \left(\eta_{ji}^{(q+1)} x_{ji}^{(q+1)} - \theta_{ji}^{(q+1)}\right) u_{kj}^{(q)}\right] x_{ki}^{(p+1)} \\
&= \sum_{k=1}^{m_p}\sum_{i=1}^{n}\sum_{j=1}^{m_q} \left(\sum_{p\in\pi_q} \left(\eta_{ji}^{(q+1)} x_{ji}^{(q+1)} - \theta_{ji}^{(q+1)}\right) u_{kj}^{(q)} x_{ki}^{(p+1)}\right) \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m_q} \left(\eta_{ji}^{(q+1)} x_{ji}^{(q+1)} - \theta_{ji}^{(q+1)}\right) \left\langle x_i^{\pi_q}, u_j^{(q)} \right\rangle \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m_q} \left(\eta_{ji}^{(q+1)} \left\langle x_i^{\pi_q}, u_j^{(q)} \right\rangle - \theta_{ji}^{(q+1)}\right) x_{ji}^{(q+1)}
\end{aligned}$$

If we apply similar technique which is used in eqs. equation 46-equation 47, we have:

$$\begin{aligned}\sum_{p\in\pi_q} F^{(p\to q)} &= \sum_{i=1}^{n}\sum_{j=1}^{m_q}\left(\gamma_{ji}^{(q+1)} + \sum_{c\in\phi_q}\sum_{z=1}^{m_c}\left(\eta_{zi}^{(c+1)}x_{zi}^{(c+1)} - \theta_{zi}^{(c+1)}\right)u_{jz}^{(c)}\right)x_{ji}^{(q+1)} \\ &= \sum_{c\in\phi_q}\sum_{i=1}^{n}\sum_{j=1}^{m_q}\sum_{z=1}^{m_c}\left(\eta_{zi}^{(c+1)}x_{zi}^{(c+1)} - \theta_{zi}^{(c+1)}\right)u_{jz}^{(c)}x_{ji}^{(q+1)} \\ &= \sum_{c\in\phi_q} F^{(q\to c)}\end{aligned}$$

For the proof of eq. equation 9, we have:

$$\begin{aligned}\sum_{p\in\pi_q} F^{(p\to q)} &= \sum_{p\in\pi_q}\sum_{k=1}^{m_p}\left(\sum_{i=1}^{n}\sum_{j=1}^{m_q}\eta_{ji}^{(q+1)}x_{ji}^{(q+1)} - \theta_{ji}^{(q+1)}\right)u_{kj}^{(q)}x_{ki}^{(p+1)} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m_q}\left(\eta_{ji}^{(q+1)}x_{ji}^{(q+1)} - \theta_{ji}^{(q+1)}\right)\left\langle x_i^{\pi_q}, u_j^{(q)}\right\rangle \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m_q}\left(\eta_{ji}^{(q+1)}\left\langle x_i^{\pi_q}, u_j^{(q)}\right\rangle - \theta_{ji}^{(q+1)}\right)x_{ji}^{(q+1)}.\end{aligned}$$

As the only node that has no children in our assumed graph is the sink node, we can set $q = L-1$. Then by using eq. equation 37, we have:

$$\begin{aligned}\sum_{p\in\pi_q} F^{(p\to q)} &= \sum_{i=1}^{n}\sum_{j=1}^{m_{L-1}}\left(\eta_{ji}^{(L)}\left\langle x_i^{\pi_{L-1}}, u_j^{(L-1)}\right\rangle - \theta_{ji}^{(L)}\right)x_{ji}^{(L)} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m_{L-1}}\left(\beta_i y_i v_j + \gamma_{ji}^{(L)}\right)x_{ji}^{(L)} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m_{L-1}}\left(\beta_i y_i v_j\right)x_{ji}^{(L)} \\ &= \sum_{i=1}^{n}\beta_i y_i\left\langle x_i^{(L)}, v\right\rangle = \sum_{i=1}^{n}\beta_i(1-\xi_i) \le 1.\end{aligned}$$

F Proof of Theorem 4

Decompose the sum into the sink node $q = L-1$ and the remaining interior nodes. Applying Theorem 2:

$$\lambda\sum_{q=1}^{L-1}\|U^{(q)}\|_F^2 = \underbrace{\lambda\|U^{(L-1)}\|_F^2}_{\le 1} + \lambda\sum_{q=1}^{L-2}\|U^{(q)}\|_F^2 = 1 + \sum_{q=1}^{L-2}\sum_{c\in\phi_q}F^{(q\to c)},$$

where the equality for the interior nodes follows from the identity $\lambda\|U^{(q)}\|_F^2 = \sum_{c\in\phi_q}F^{(q\to c)}$ when $\phi_q \neq \varnothing$ (Theorem 2). The last expression is exactly $1 + \sum_{(q,c)\in E}F^{(q\to c)}$ because each edge (q,c) in the DAG corresponds to a flow term $F^{(q\to c)}$.

G Proof of Theorem 5

Any feasible flow in an acyclic network can be decomposed into a finite collection of source-to-sink paths with associated non-negative flows (23; 24; 25). The decomposition is obtained by the following iterative algorithm:

- While there exists an edge with positive flow:
  1. Find a directed path P from s to t using only edges with strictly positive flow.
  2. Let $\delta_P = \min_{(p,q)\in P} F^{(p\to q)}$.
  3. Subtract δ_P from the flow on every edge of P.
  4. Record a path component with flow δ_P.

The procedure terminates because each iteration reduces the total number of edges with positive flow or the total flow amount. The result is a representation

$$\sum_{P\in\mathcal{P}} \delta_P \, \mathbf{1}_P,$$

where $\delta_P \geq 0$, and $\mathbf{1}_P$ is the indicator vector of edges on P. From this decomposition, the total flow into the sink equals the sum of all path flows, based on eq. equation 9, we have:

$$\sum_{(u,t)\in E} F^{(u\to t)} = \sum_{P\in\mathcal{P}} \delta_P \leq 1.$$

The sum of flows over all edges can be expressed as

$$\sum_{(p,q)\in E} F^{(p\to q)} = \sum_{P\in\mathcal{P}} \sum_{(p,q)\in P} \delta_P \mathbf{1}_P((p,q)) = \sum_{P\in\mathcal{P}} \delta_P \, \ell(P),$$

where $\ell(P)$ denotes the length of path P. Since $\ell(P) \leq T_G$ for all $P \in \mathcal{P}$, we obtain

$$\sum_{(p,q)\in E} F^{(p\to q)} \leq \sum_{P\in\mathcal{P}} \delta_P \cdot T_G = T_G \sum_{P\in\mathcal{P}} \delta_P \leq T_G.$$

H Proof of Lemma 1

As $\alpha_i, \beta_i \geq 0$, from eq. equation 35 we have $\alpha_i \leq \frac{1}{n}$ and $\beta_i \leq \frac{1}{n}$. From eq. equation 33, we have:

$$v = \frac{1}{\lambda} \sum_{i=1}^{n} \beta_i y_i x_i^{(L)}.$$

Using the dual feasibility condition equation 39, we have:

$$\lambda \left\langle \frac{1}{\lambda} \sum_{i=1}^{n} \beta_i y_i x_i^{(L)}, v \right\rangle = \sum_{i=1}^{n} \beta_i - \sum_{i=1}^{n} \beta_i \xi_i \leq \sum_{i=1}^{n} \beta_i \leq \sum_{i=1}^{n} \frac{1}{n} = 1,$$

which is equivalent to

$$\|v\|^2 \leq \frac{1}{\lambda}.$$

I Generalization to Other Activations

The generalization bounds derived in Theorem 6 extend naturally to a broader class of piecewise-linear activation functions, including Leaky ReLU and ReLU6, with only minor adjustments to the associated constants. Our analysis is based on representing the ReLU activation through a set of equivalent constraints (eqs. equation 24–equation 26), which can be systematically adapted to other piecewise-linear activations by introducing the corresponding constraint formulations (see equation 52–equation 56 for Leaky ReLU and equation 76–equation 79 for ReLU6). The associated Lagrangian and KKT conditions are modified accordingly, yielding analogous structural properties. While the definition of the induced flow must be adjusted to reflect the specific activation function, for Leaky ReLU we obtain

$$F_{LR}^{(p\to q)} = \sum_{k=1}^{m_p} \sum_{i=1}^{n} \sum_{j=1}^{m_q} \left[\eta_{jk}^{(q+1)} \left((1+\alpha) x_{jk}^{(q+1)} - 2\alpha \left\langle x_k^{\pi_q}, u_j^{(q)} \right\rangle\right) - \gamma_{jk}^{(q+1)} - \alpha \theta_{jk}^{(q+1)}\right] u_{kj}^{(q)} x_{ki}^{(p+1)}, \tag{50}$$

and for ReLU6 we have

$$F_{R6}^{(p\to q)} = \sum_{k=1}^{m_p}\sum_{i=1}^{n}\sum_{j=1}^{m_q}\left(\eta_{jk}^{(q+1)}x_{jk}^{(q+1)} - \delta_{jk}^{(q+1)}(x_{jk}^{(q+1)} - 6)\right)u_{kj}^{(q)}x_{ki}^{(p+1)}. \tag{51}$$

The core arguments and resulting generalization bounds remain unchanged. Despite these changes, the core arguments remain intact, and the resulting generalization bounds follow through with only minor modifications.

$$\begin{aligned}\underset{U^{(1)},\ldots,U^{(L-1)},v,X^{(2)},\ldots,X^{(L)}}{\text{minimize}} \quad & \frac{1}{n}\sum_{i=1}^{n}\max\left(0, 1 - y_i\left\langle x_i^{(L)}, v\right\rangle\right) + \frac{\lambda}{2}\left(\sum_{l=1}^{L-1}\left\|U^{(1)}\right\|_F^2 + \|v\|^2\right)\\ \text{subject to} \quad & x_{ji}^{(l)} = \begin{cases}\left\langle x_i^{\pi_{l-1}}, u_j^{(l)}\right\rangle & \text{if } \left\langle x_i^{\pi_{l-1}}, u_j^{(l)}\right\rangle > 0,\\ \alpha\left\langle x_i^{\pi_{l-1}}, u_j^{(l)}\right\rangle & \text{if } \left\langle x_i^{\pi_{l-1}}, u_j^{(l)}\right\rangle \le 0,\end{cases}\end{aligned}$$

for $i = 1,\ldots,n$ and $j = 1,\ldots,m_{l-1}$ where $l = 2,\ldots,L$. The above can be written as:

$$\begin{aligned}\underset{U^{(1)},\ldots,U^{(L-1)},v,X^{(2)},\ldots,X^{(L)}}{\text{minimize}} \quad & \frac{1}{n}\sum_{i=1}^{n}\xi_i + \frac{\lambda}{2}\left(\sum_{l=1}^{L-1}\left\|U^{(l)}\right\|_F^2 + \|v\|^2\right)\\ \text{subject to} \quad & \xi_i \ge 0, && (52)\\ & y_i\left\langle x_i^{(L)}, v\right\rangle \ge 1 - \xi_i, && (53)\\ & x_{ji}^{(l)} \ge \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)}\right\rangle, && (54)\\ & x_{ji}^{(l)} \ge \alpha\left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)}\right\rangle, && (55)\\ & \left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)}\right\rangle\right)\left(x_{ji}^{(l)} - \alpha\left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)}\right\rangle\right) = 0. && (56)\end{aligned}$$

The Lagrangian is:

$$\begin{aligned}\mathscr{L}(\xi_i, U^{(l)}, v, X^{(l)}) = {} & \frac{1}{n}\sum_{i=1}^{n}\xi_i + \frac{\lambda}{2}\left(\sum_{l=1}^{L-1}\left\|U^{(l)}\right\|_F^2 + \|v\|^2\right) - \sum_{i=1}^{n}\beta_i\left(y_i\langle x_i^{(L)}, v\rangle - 1 + \xi_i\right)\\ & - \sum_{i=1}^{n}\alpha_i\xi_i - \sum_{l=2}^{L}\sum_{i=1}^{n}\sum_{j=1}^{m_{l-1}}\gamma_{ij}^{(l)}\left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)}\right\rangle\right)\\ & - \sum_{l=2}^{L}\sum_{i=1}^{n}\sum_{j=1}^{m_{l-1}}\theta_{ji}^{(l)}\left(x_{ji}^{(l)} - \alpha\left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)}\right\rangle\right)\\ & + \sum_{l=2}^{L}\sum_{i=1}^{n}\sum_{j=1}^{m_{l-1}}\eta_{ji}^{(l)}\left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)}\right\rangle\right)\left(x_{ji}^{(l)} - \alpha\left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)}\right\rangle\right),\end{aligned}$$

and the stationary conditions of KKT are:

$$\frac{\partial \mathscr{L}}{\partial v} = \lambda v - \sum_{i=1}^{n} \beta_i y_i x_i^{(L)} = 0, \tag{57}$$

$$\frac{\partial \mathscr{L}}{\partial u_{tk}^{(l)}} = \lambda u_{tk}^{(l)} + \sum_{i=1}^{n} \left(\gamma_{ki}^{(l+1)} + \alpha\theta_{ki}^{(l+1)}\right) x_{ti}^{\pi_l} - \sum_{i=1}^{n} \eta_{ki}^{(l+1)} x_{ti}^{\pi_l} \left((1+\alpha)x_{ki}^{(l+1)} - 2\alpha \left\langle x_i^{\pi_l}, u_t^{(l)} \right\rangle\right) = 0, \tag{58}$$

$$\frac{\partial \mathscr{L}}{\partial \xi_i} = \frac{1}{n} - \alpha_i - \beta_i = 0, \tag{59}$$

$$\frac{\partial \mathscr{L}}{\partial x_{tk}^{(l)}} = -\gamma_{tk}^{(l)} - \theta_{tk}^{(l)} + \eta_{tk}^{(l)} \left(2x_{tk}^{(l)} - (1+\alpha)\left\langle x_k^{\pi_{l-1}}, u_t^{(l-1)} \right\rangle\right) + \sum_{q\in\phi_{l-1}} \sum_{j=1}^{m_q} \left(\gamma_{jk}^{(q+1)} + \alpha\theta_{jk}^{(q+1)} - \eta_{jk}^{(q+1)} \left[(1+\alpha)x_{jk}^{(q+1)} - 2\alpha \left\langle x_k^{\pi_q}, u_j^{(q)} \right\rangle\right]\right) u_{tj}^{(q)} = 0, \tag{60}$$

$$\frac{\partial \mathscr{L}}{\partial x_{tk}^{(L)}} = -\beta_k y_k v_t - \gamma_{tk}^{(L)} - \theta_{tk}^{(L)} + \eta_{tk}^{(L)} \left(2x_{tk}^{(L)} - (1+\alpha)\left\langle x_k^{\pi_{L-1}}, u_t^{(L-1)} \right\rangle\right) = 0. \tag{61}$$

In addition, the complementary slackness conditions of KKT for $i = 1, \ldots, n$, and $l = 2, \ldots, L$ are:

$$\alpha_i \xi_i = 0, \tag{62}$$

$$\beta_i \left(1 - \xi_i - y_i \left\langle x_i^{(L)}, v \right\rangle\right) = 0, \tag{63}$$

$$\gamma_{ji}^{(l)} \left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle)\right) = 0, \tag{64}$$

$$\theta_{ji}^{(l)} \left(x_{ji}^{(l)} - \alpha \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle\right) = 0, \tag{65}$$

Moreover, the dual feasibility conditions of KKT for $i = 1, \ldots, n$ and $j = 1, \ldots, m_{l-1}$ where $l = 2, \ldots, L$ are:

$$\alpha_i, \beta_i, \gamma_{ji}^{(l)}, \theta_{ji}^{(l)} \geq 0, \tag{66}$$
$$\eta_{ij}^{(l)} \in \mathbb{R},$$

also, eqs.equation 52-equation 56 are primal conditions.

It should be noted that, for different activation functions, the definition of the flow must be updated, as mentioned in eqs. equation 50 and equation 51.

Lemma 5. *Let $U^{(q)}$ denote the weight matrix at node q. Under the model constraints equation 52-equation 66, the following hold:*

- ***Sink node.*** *If $\phi_q = \varnothing$, then*

$$\lambda \|U^{(q)}\|_F^2 \leq 1. \tag{67}$$

- ***Interior node.*** *If $\phi_q \neq \varnothing$, then*

$$\lambda \|U^{(q)}\|_F^2 = \sum_{c\in\phi_q} F_{LR}^{(q\to c)}. \tag{68}$$

- ***Non-negativity.*** *If $\pi_q \neq \varnothing$, then for every parent $p \in \pi_q$,*

$$F_{LR}^{(p\to q)} \geq 0. \tag{69}$$

- ***Flow conservation laws.*** *If $\phi_q \neq \varnothing$ (i.e., q is not the sink), then the flow entering q equals the flow leaving q:*

$$\sum_{p\in\pi_q} F_{LR}^{(p\to q)} = \sum_{c\in\phi_q} F_{LR}^{(q\to c)}. \tag{70}$$

Proof. As we assumed that the graph has one source and one sink node, the only node q, such that $\phi_q = \varnothing$, is $q = L-1$. By multiplying eq.equation 58 by $u_{tk}^{(L-1)}$ and summing over t, we obtain:

$$\begin{aligned}\lambda\|u_k^{(L-1)}\|^2 = &\sum_{t=1}^{m_{L-1}}\sum_{i=1}^{n}\big[\eta_{ki}^{(L)}\left((1+\alpha)x_{ki}^{(L)} - 2\alpha\left\langle x_i^{\pi_{L-1}}, u_t^{(L-1)}\right\rangle\right)\\ &-\gamma_{ki}^{(l+1)} - \alpha\theta_{ki}^{(l+1)}\big]x_{ti}^{\pi_{L-1}}u_{tk}^{(L-1)}\\ =&\sum_{i=1}^{n}\left[\eta_{ki}^{(L)}\left((1+\alpha)x_{ki}^{(L)} - 2\alpha\left\langle x_i^{\pi_{L-1}}, u_t^{(L-1)}\right\rangle\right) - \gamma_{ki}^{(L)} - \alpha\theta_{ki}^{(L)}\right]\left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)}\right\rangle\end{aligned}$$

Summing over k and applying complementary slackness condition equation 64 and equation 65, we have

$$\begin{aligned}\lambda\left\|U^{(L-1)}\right\|_F^2 = &\sum_{i=1}^{n}\sum_{k=1}^{m_L}\Big[\eta_{ki}^{(L)}\Big((1+\alpha)x_{ki}^{(L)} - 2\alpha\left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)}\right\rangle\Big)\left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)}\right\rangle\\ &-\gamma_{ki}^{(L)}x_{ki}^{(L)} - \theta_{ki}^{(L)}x_{ki}^{(L)}\Big]\end{aligned}$$

Considering the primal feasibility constrains equation 52-equation 56, we obtain

$$\begin{aligned}\lambda\left\|U^{(L-1)}\right\|_F^2 = &\sum_{(k,i)\in T_+}\big[\eta_{ki}^{(L)}\left((1+\alpha)x_{ki}^{(L)} - 2\alpha\left\langle x_i^{\pi_{L-1}}, u_t^{(L-1)}\right\rangle\right)\left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)}\right\rangle\\ &-\gamma_{ki}^{(L)}x_{ki}^{(L)} - \theta_{ki}^{(L)}x_{ki}^{(L)}\big]\\ &+\sum_{(k,i)\in T_-}\big[\eta_{ki}^{(L)}\left((1+\alpha)x_{ki}^{(L)} - 2\alpha\left\langle x_i^{\pi_{L-1}}, u_t^{(L-1)}\right\rangle\right)\left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)}\right\rangle\\ &-\gamma_{ki}^{(L)}x_{ki}^{(L)} - \theta_{ki}^{(L)}x_{ki}^{(L)}\big]\\ =&\sum_{(k,i)\in T_+}\left[(1-\alpha)\eta_{ki}^{(L)}x_{ki}^{(L)} - \gamma_{ki}^{(L)} - \theta_{ki}^{(L)}\right]x_{ki}^{(L)}\\ &+\sum_{(k,i)\in T_-}\left[\eta_{ki}^{(L)}(\alpha-1)\left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)}\right\rangle - \gamma_{ki}^{(L)} - \theta_{ki}^{(L)}\right]x_{ki}^{(L)},\end{aligned}$$

where

$$T_+ = \{(k,i)|x_{ki}^{(L)} = \left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)}\right\rangle > 0\}, \tag{71}$$

$$T_- = \{(k,i)|x_{ki}^{(L)} = \alpha\left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)}\right\rangle \le 0\}. \tag{72}$$

Then by applying the stationary condition equation 61 and complementary slackness condition equation 63, we obtain:

$$\begin{aligned}\lambda\left\|U^{(L-1)}\right\|_F^2 = &\sum_{(k,i)\in T_+}\left[(1-\alpha)\eta_{ki}^{(L)}x_{ki}^{(L)} - \gamma_{ki}^{(L)} - \theta_{ki}^{(L)}\right]x_{ki}^{(L)}\\ &+\sum_{(k,i)\in T_-}\left[(\alpha-1)\eta_{ki}^{(L)}\left\langle x_i^{\pi_{L-1}}, u_k^{(L-1)}\right\rangle - \gamma_{ki}^{(L)} - \theta_{ki}^{(L)}\right]x_{ki}^{(L)}\\ =&\sum_{(k,i)\in T_+}\left[\beta_i y_i v_k + \gamma_{ki}^{(L)} + \theta_{ki}^{(L)} - \gamma_{ki}^{(L)} - \theta_{ki}^{(L)}\right]x_{ki}^{(L)}\\ &+\sum_{(k,i)\in T_-}\left[\beta_i y_i v_k + \gamma_{ki}^{(L)} + \theta_{ki}^{(L)} - \gamma_{ki}^{(L)} - \theta_{ki}^{(L)}\right]x_{ki}^{(L)}\\ =&\sum_{k=1}^{m_L}\sum_{i=1}^{n}\beta_i y_i v_k x_{ki}^{(L)} = \sum_{i=1}^{n}\beta_i y_i\left\langle x_i^{(L)}, v\right\rangle = \sum_{i=1}^{n}\beta_i(1-\xi_i)\\ =&\sum_{i=1}^{n}(\frac{1}{n}-\alpha_i)(1-\xi_i) = \sum_{i=1}^{n}(\frac{1}{n} - \frac{1}{n}\xi_i - \alpha_i + \alpha_i\xi_i) \le \sum_{i=1}^{n}\frac{1}{n} = 1.\end{aligned}$$

In addition, if $\phi_q \neq \varnothing$ multiplying eq. equation 58 by $u_{tk}^{(q)}$ and summing over t, we have:

$$\begin{aligned}
\lambda\|u_k^{(q)}\|^2 &= \sum_{t=1}^{M_q}\sum_{i=1}^{n}\eta_{ki}^{(q+1)}\left((1+\alpha)x_{ki}^{(q+1)} - 2\alpha\left\langle x_i^{\pi_q}, u_t^{(q)}\right\rangle\right)x_{ti}^{\pi_q}u_{tk}^{(q)} \\
&\quad - \sum_{t=1}^{M_{q-1}}\sum_{i=1}^{n}\left(\gamma_{ki}^{(q+1)} + \alpha\theta_{ki}^{(q+1)}\right)x_{ti}^{\pi_q}u_{tk}^{(q)} \\
&= \sum_{i=1}^{n}\eta_{ki}^{(q+1)}\left((1+\alpha)x_{ki}^{(q+1)} - 2\alpha\left\langle x_i^{\pi_q}, u_t^{(q)}\right\rangle\right)\left\langle x_i^{\pi_q}, u_t^{(q)}\right\rangle \\
&\quad - \sum_{i=1}^{n}\left(\gamma_{ki}^{(q+1)} + \alpha\theta_{ki}^{(q+1)}\right)\left\langle x_i^{\pi_q}, u_t^{(q)}\right\rangle.
\end{aligned}$$

Therefore,

$$\begin{aligned}
\lambda\|U^{(q)}\|_F^2 = \sum_{k=1}^{m_q}\sum_{i=1}^{n}\big[&\eta_{ki}^{(q+1)}\left((1+\alpha)x_{ki}^{(q+1)} - 2\alpha\left\langle x_i^{\pi_q}, u_k^{(q)}\right\rangle\right) \\
&- \gamma_{ki}^{(q+1)} - \alpha\theta_{ki}^{(q+1)}\big]\left\langle x_i^{\pi_q}, u_k^{(q)}\right\rangle.
\end{aligned} \tag{73}$$

Equation equation 73 can be separated into two parts:

$$\lambda\left\|U^{(q)}\right\|_F^2 = \sum_{(k,i)\in T_+}\left[(1-\alpha)\eta_{ki}^{(q+1)}x_{ki}^{(q+1)} - \gamma_{ki}^{(q+1)} - \theta_{ki}^{(q+1)}\right]x_{ki}^{(q+1)} \tag{74}$$

$$+ \sum_{(k,i)\in T_-}\left[(\alpha-1)\eta_{ki}^{(q+1)}\left\langle x_i^{\pi_q}, u_t^{(q)}\right\rangle - \gamma_{ki}^{(q+1)} - \theta_{ki}^{(q+1)}\right]x_{ki}^{(q+1)}, \tag{75}$$

where T_+ and T_- are defined in eq. equation 71. Using eqs. equation 60, we obtain

$$\begin{aligned}
\lambda\|U^{(q)}\|_F^2 &= \sum_{(k,i)\in T_+}\sum_{c\in\phi_q}\sum_{j=1}^{m_c}\big[\eta_{ji}^{(c+1)}\left((1+\alpha)x_{ji}^{(c+1)} - 2\alpha\left\langle x_i^{\pi_c}, u_j^{(c)}\right\rangle\right) \\
&\qquad - \gamma_{ji}^{(c+1)} - \alpha\theta_{ji}^{(c+1)}\big]u_{kj}^{(c)}x_{ki}^{(q+1)} \\
&\quad + \sum_{(k,i)\in T_-}\sum_{c\in\phi_q}\sum_{j=1}^{m_c}\big[\eta_{ji}^{(c+1)}\left((1+\alpha)x_{ji}^{(c+1)} - 2\alpha\left\langle x_i^{\pi_c}, u_j^{(c)}\right\rangle\right) \\
&\qquad - \gamma_{ji}^{(c+1)} - \alpha\theta_{ji}^{(c+1)}\big]u_{kj}^{(c)}x_{ki}^{(q+1)} \\
&= \sum_{c\in\phi_q}\sum_{k=1}^{m_q}\sum_{i=1}^{n}\sum_{j=1}^{m_c}\big[\eta_{ji}^{(c+1)}\left((1+\alpha)x_{ji}^{(c+1)} - 2\alpha\left\langle x_i^{\pi_c}, u_j^{(c)}\right\rangle\right) \\
&\qquad - \gamma_{ji}^{(c+1)} - \alpha\theta_{ji}^{(c+1)}\big]u_{kj}^{(c)}x_{ki}^{(q+1)} \\
&= \sum_{c\in\phi_q}F_{LR}^{(q+1\to c)}.
\end{aligned}$$

Moreover, eq. equation 69 can be proved by a similar argument to that used in eqs. equation 48–equation 49.

Lets $\phi_q \neq \varnothing$ and $p \in \pi_q$, based on eq. equation 50, we have:

$$
\begin{aligned}
\sum_{p\in\pi_q} F_{LR}^{(p\to q)} &= \sum_{p\in\pi_q}\sum_{k=1}^{m_p}\sum_{i=1}^{n}\sum_{j=1}^{m_q} \big[\eta_{jk}^{(q+1)}\left((1+\alpha)x_{jk}^{(q+1)} - 2\alpha\left\langle x_k^{\pi_q}, u_j^{(q)}\right\rangle\right) \\
&\qquad - \gamma_{jk}^{(q+1)} - \alpha\theta_{jk}^{(q+1)}\big] u_{kj}^{(q)} x_{ki}^{(p+1)} \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m_q}\sum_{k=1}^{m_p}\Big\{\sum_{p\in\pi_q} \big[\eta_{jk}^{(q+1)}\left((1+\alpha)x_{jk}^{(q+1)} - 2\alpha\left\langle x_k^{\pi_q}, u_j^{(q)}\right\rangle\right) \\
&\qquad - \gamma_{jk}^{(q+1)} - \alpha\theta_{jk}^{(q+1)}\big]\Big\} u_{kj}^{(q)} x_{ki}^{(p+1)} \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m_q} \big[\eta_{jk}^{(q+1)}\left((1+\alpha)x_{jk}^{(q+1)} - 2\alpha\left\langle x_k^{\pi_q}, u_j^{(q)}\right\rangle\right) \\
&\qquad - \gamma_{jk}^{(q+1)} - \alpha\theta_{jk}^{(q+1)}\big]\left\langle x_i^{\pi_q}, u_j^{(q)}\right\rangle \\
&= \lambda\left\|U^{(q)}\right\|_F^2.
\end{aligned}
$$

Using the same technique as in the proof of equation 6, we obtain

$$\lambda\|U^{(q)}\|_F^2 = \sum_{c\in\phi_q} F_{LR}^{(q\to c)}.$$

Therefore, we have

$$\sum_{p\in\pi_q} F_{LR}^{(p\to q)} = \sum_{c\in\phi_q} F_{LR}^{(q\to c)}.$$

□

Consequently, the results established in Theorems 3, 5, the Max-Total-Flow problem, and Theorem 6 extend to the setting where Leaky ReLU is applied.

We extend the analysis to the ReLU6 activation by expressing it through an equivalent set of constraints that capture its bounded and piecewise-linear structure. This leads to a corresponding Lagrangian formulation and KKT conditions. Due to the saturation effect, the analysis requires a three-way partition of indices, and the definition of flow is adjusted accordingly. Nevertheless, the main arguments and resulting generalization bounds remain unchanged.

The ReLU6 activation, $\sigma(z) = \min(\max(0, z), 6)$, is equivalent to the following constraints

$$x_{ji}^{(l+1)} \geq 0, \tag{76}$$

$$6 - x_{ji}^{(l+1)} \geq 0, \tag{77}$$

$$x_{ji}^{(l+1)}\left(x_{ji}^{(l+1)} - \left\langle x_i^{\pi_l}, u_j^{(l)}\right\rangle\right) \leq 0, \tag{78}$$

$$\left(x_{ji}^{(l+1)} - 6\right)\left(x_{ji}^{(l+1)} - \left\langle x_i^{\pi_l}, u_j^{(l)}\right\rangle\right) \leq 0. \tag{79}$$

The corresponding Lagrangian function for the ReLU6 classification problem can be written using the primal conditions equation 76–equation 79.

Our analysis requires a three-way partition of indices to capture all three operational regions of the activation function:

$$T_0 = \{(k,i)|x_{ki}^{(l)} = 0,\ \&\left\langle x_i^{\pi_{l-1}}, u_k^{(l-1)}\right\rangle \leq 0\ \},$$

$$T_+ = \{(k,i)|\ x_{ki}^{(l)} = \left\langle x_i^{\pi_{l-1}}, u_k^{(l-1)}\right\rangle,\quad 0 < x_{ki}^{(l)} \leq 6\},$$

and

$$T_- = \{(k,i)|x_{ki}^{(l)} = 6,\ \left\langle x_i^{\pi_{l-1}}, u_k^{(l-1)}\right\rangle > 6\}.$$

This tripartite decomposition parallels the two-set approach used for Leaky ReLU but accounts for the additional saturation constraint. The complementary slackness conditions induced by the ReLU6 activation function ensure that the KKT-based norm analysis remains tractable, ultimately leading to the same generalization bound.

J KKT CONDITIONS OF THE MULTI-CLASS RELU NEURAL NETWORK CLASSIFICATION

The multi-class classification problem can be equivalently formulated as follows:

$$\begin{aligned}
\underset{\xi_i, U^{(l)}, v_\kappa, X^{(l)}}{\text{minimize}} \quad & \frac{1}{n}\sum_{i=1}^{n}\xi_i + \frac{\lambda}{2}\left(\sum_{l=1}^{L-1}\left\|U^{(l)}\right\|_F^2 + \sum_{\kappa=1}^{\mathcal{K}}\|v_\kappa\|^2\right) \\
& \text{subject to} \quad \xi_i \geq 0, && (80)\\
& \left\langle x_i^{(L)}, v_{y_i}\right\rangle \geq 1 + \left\langle x_i^{(L)}, v_\kappa\right\rangle - \xi_i, \quad \forall \kappa \neq y_i, && (81)\\
& x_{ji}^{(l)} \geq 0, && (82)\\
& x_{ji}^{(l)} \geq \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)}\right\rangle, && (83)\\
& x_{ji}^{(l)}\left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)}\right\rangle\right) = 0, && (84)\\
& \forall i = 1,\ldots,n, \forall j = 1,\ldots,m_{l-1}, \forall l = 2,\ldots,L.
\end{aligned}$$

Since most of the KKT conditions for this problem are similar to those of the binary classification problem, eqs. equation 2-equation 42, we only highlight the parts that need to be modified; the remaining equations remain unchanged:

$$\frac{\partial \mathscr{L}}{\partial v_m} = \lambda v_m - \sum_{i:y_i=m}\sum_{\substack{\kappa=1\\ \kappa\neq y_i}}^{\mathcal{K}}\beta_{i\kappa}x_i^{(L)} + \sum_{i:y_i\neq m}\beta_{im}x_i^{(L)} = 0, \tag{85}$$

$$\frac{\partial \mathscr{L}}{\partial \xi_i} = \frac{1}{n} - \alpha_i - \sum_{\substack{\kappa=1\\ y_\kappa\neq y_i}}^{\mathcal{K}}\beta_{i\kappa} = 0, \tag{86}$$

$$\frac{\partial \mathscr{L}}{\partial x_{tk}^{(L)}} = -\sum_{\substack{y_\kappa=1\\ \kappa\neq y_i}}^{\mathcal{K}}\beta_{k\kappa}(v_{ty_i} - v_{t\kappa}) - \gamma_{tk}^{(L)} - \theta_{tk}^{(L)} + \eta_{tk}^{(L)}\left(2x_{tk}^{(L)} - \left\langle x_k^{(L-1)}, u_t^{(L-1)}\right\rangle\right) = 0. \tag{87}$$

Complementary slackness condition equation 39, will change to:

$$\beta_{i\kappa}\left(\left\langle x_i^{(L)}, v_{y_i}\right\rangle - \left\langle x_i^{(L)}, v_\kappa\right\rangle - 1 + \xi_i\right) = 0. \tag{88}$$

K PROOF OF LEMMA 2

From equation 86 and dual feasibility, we immediately obtain $\alpha_i \leq \frac{1}{n}$ and $\sum_{\kappa\neq y_i}\beta_{i\kappa} \leq \frac{1}{n}$. Summing equation 88 yields:

$$\sum_{i=1}^{n}\sum_{\substack{\kappa=1\\ \kappa\neq y_i}}^{\mathcal{K}}\beta_{i\kappa}\left(\left\langle x_i^{(L)}, v_{y_i}\right\rangle - \left\langle x_i^{(L)}, v_\kappa\right\rangle\right) = \sum_{i=1}^{n}\sum_{\substack{\kappa=1\\ \kappa\neq y_i}}^{\mathcal{K}}\beta_{i\kappa}\left(1-\xi_i\right). \tag{89}$$

Multiplying eq. equation 85 by v_κ, summing over $\kappa = 1,\ldots,\mathcal{K}$, and using eq. equation 89, we obtain

$$\begin{aligned}
\lambda\sum_{\kappa=1}^{\mathcal{K}}\|v_\kappa\|^2 &= \sum_{i=1}^{n}\sum_{\substack{\kappa=1\\ \kappa\neq y_i}}^{\mathcal{K}}\beta_{i\kappa}\Big(\langle x_i^{(L)}, v_{y_i}\rangle - \langle x_i^{(L)}, v_\kappa\rangle\Big)\\
&= \sum_{i=1}^{n}\sum_{\substack{\kappa=1\\ \kappa\neq y_i}}^{\mathcal{K}}\beta_{i\kappa}\left(1-\xi_i\right) \leq \sum_{i=1}^{n}\sum_{\substack{\kappa=1\\ \kappa\neq y_i}}^{\mathcal{K}}\beta_{i\kappa}\\
&\leq \sum_{i=1}^{n}\frac{1}{n} = 1,
\end{aligned}$$

therefore, $\sum_{\kappa=1}^{\mathcal{K}}\|v_\kappa\|^2 \leq \frac{1}{\lambda}$.

L KKT CONDITIONS OF THE RELU REGRESSION SETTING

The problem can be written as ($\forall i = 1, \ldots, n$, $\forall j = 1, \ldots, m_{l-1}$ and $\forall l = 2, \ldots, L$):

$$\begin{aligned}
\underset{U^{(l)}, v, X^{(l)}}{\text{minimize}} \quad & \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \left\langle x_i^{(L)}, v \right\rangle \right)^2 + \frac{\lambda}{2} \left(\sum_{l=1}^{L-1} \left\| U^{(l)} \right\|_F^2 + \|v\|^2 \right) \\
\text{subject to} \quad & x_{ji}^{(l)} \geq 0, && (90) \\
& x_{ji}^{(l)} \geq \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle, && (91) \\
& x_{ji}^{(l)} \left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle \right) = 0. && (92)
\end{aligned}$$

Following the same methodology as in the classification case, we construct the Lagrangian and derive the KKT conditions. The Lagrangian of regression problem is as follows:

$$\begin{aligned}
\mathscr{L}(U^{(l)}, v, X^{(L)}) = & \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \left\langle x_i^{(L)}, v \right\rangle \right)^2 + \frac{\lambda}{2} \sum_{l=1}^{L-1} \left\| U^{(l)} \right\|_F^2 + \frac{\lambda}{2} \|v\|^2 \\
& - \sum_{l=2}^{L} \sum_{i=1}^{n} \sum_{j=1}^{m_{l-1}} \alpha_{ji}^{(l)} x_{ji}^{(l)} - \sum_{l=2}^{L} \sum_{i=1}^{n} \sum_{j=1}^{m_{l-1}} \beta_{ji}^{(l)} \left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle \right) \\
& + \sum_{l=2}^{L} \sum_{i=1}^{n} \sum_{j=1}^{m_{l-1}} \gamma_{ji}^{(l)} x_{ji}^{(l)} \left(x_{ji}^{(l)} - \left\langle x_i^{\pi_{l-1}}, u_j^{(l-1)} \right\rangle \right),
\end{aligned}$$

and the stationary conditions of KKT are:

$$\begin{aligned}
& \frac{\partial \mathscr{L}}{\partial v} = \lambda v - \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \left\langle x_i^{(L)}, v \right\rangle \right) x_i^{(L)} = 0, && (93) \\
& \frac{\partial \mathscr{L}}{\partial u_{tk}^{(l)}} = \lambda u_{tk}^{(l)} + \sum_{i=1}^{n} \left[\beta_{ki}^{(l+1)} - \gamma_{ki}^{(l+1)} x_{ki}^{(l+1)} \right] x_{ti}^{\pi_l} = 0, && (94) \\
& \frac{\partial \mathscr{L}}{\partial x_{tk}^{(l)}} = -\alpha_{tk}^{(l)} - \beta_{tk}^{(l)} + \gamma_{tk}^{(l)} \left(2x_{tk}^{(l)} - \left\langle x_k^{(l-1)}, u_t^{(l-1)} \right\rangle \right) \\
& \qquad + \sum_{q \in \phi_l} \sum_{j=1}^{m_q} \left(\beta_{jk}^{(q+1)} - \gamma_{jk}^{(q+1)} x_{jk}^{(q+1)} \right) u_{tj}^{(q)} = 0, && (95) \\
& \frac{\partial \mathscr{L}}{\partial x_{tk}^{(L)}} = -\frac{1}{n} (y_k - \langle x_k, v \rangle) v_t - \alpha_{tk}^{(L)} - \beta_{tk}^{(L)} + \gamma_{tk}^{(L)} \left(2x_{tk}^{(L)} - \left\langle x_k^{\pi_{L-1}}, u_t^{(L-1)} \right\rangle \right) = 0. && (96)
\end{aligned}$$

Additionally, the complementary slackness conditions of KKT for $i = 1, \ldots, n$ and $j = 1, \ldots, m_{l-1}$ where $l = 2, \ldots, L$ defines:

$$\begin{aligned}
& \alpha_{ji} x_{ji}^{(l)} = 0, && (97) \\
& \beta_{ji}^{(l)} \left(x_{ji}^{(l)} - \left\langle x_i^{(l-1)}, u_j^{(l-1)} \right\rangle \right) = 0. && (98)
\end{aligned}$$

Moreover, the dual feasibility conditions of KKT for $i = 1, \ldots, n$ and $j = 1, \ldots, m_l$ where $l = 2, \ldots, L$ are:

$$\begin{aligned}
& \alpha_{ji}^{(l)}, \beta_{ji}^{(l)} \geq 0, && (99) \\
& \gamma_{ij}^{(l)} \in \mathbb{R},
\end{aligned}$$

also, equation 90-equation 92 are primal conditions.

M Proof of Lemma 3

Applying the stationary condition equation 93, we have:

$$\lambda v + \frac{1}{n}\sum_{i=1}^{n}\left\langle x_i^{(L)}, v\right\rangle x_i^{(L)} = \frac{1}{n}\sum_{i=1}^{n} y_i x_i^{(L)},$$

In matrix form, with $X^{(L)} \in \mathbb{R}^{m_{L-1}\times n}$ (each column is a data point), this becomes:

$$\left(\lambda I + \frac{1}{n}X^{(L)}(X^{(L)})^T\right) v = \frac{1}{n}X^{(L)}y.$$

where I is the $m_{L-1} \times m_{L-1}$ identity matrix. Therefore,

$$v = \frac{1}{n}\left(\lambda I + \frac{1}{n}X^{(L)}(X^{(L)})^T\right)^{-1}\left(X^{(L)}y\right).$$

Using the singular value decomposition $X^{(L)} = UDV^T$, (with $U \in \mathbb{R}^{m_{L-1}\times m_{L-1}}$ orthogonal, $D \in \mathbb{R}^{m_{L-1}\times n}$ and $V \in \mathbb{R}^{n\times n}$ orthogonal), we have:

$$X^{(L)}(X^{(L)})^T = UDD^TU^T.$$

Thus,

$$v = \frac{1}{n}U\left(\lambda I + \frac{1}{n}DD^T\right)^{-1} DV^Ty.$$

The spectral norm of the matrix $\left(\lambda I + \frac{1}{n}DD^\top\right)^{-1} D$ is bounded by the maximum of the function $f(z) = \frac{z}{\lambda + z^2/n}$ for $z \geq 0$, which achieves its maximum value of $\frac{1}{2}\sqrt{n/\lambda}$ at $z = \sqrt{\lambda n}$. Consequently,

$$\left\|\left(\lambda I + \frac{1}{n}DD^\top\right)^{-1} D\right\|_2 \leq \frac{1}{2}\sqrt{\frac{n}{\lambda}}.$$

Using the orthogonality of U and V, we have

$$\|v\| \leq \frac{1}{n}\cdot\frac{1}{2}\sqrt{\frac{n}{\lambda}}\cdot\|y\| = \frac{1}{2\sqrt{n\lambda}}\|y\|.$$

N Proof of Lemma 4

Multiplying equation equation 94 by $u_{tk}^{(l)}$ and applying the same technique used in eqs. equation 43-equation 44, we obtain

$$\lambda\|U^{(L-1)}\|_F^2 = \sum_{i=1}^{n}\sum_{j=1}^{m_{L-1}}\left(\gamma_{ji}^{(L)}\left\langle x_i^{(L-1)}, u_j^{(L-1)}\right\rangle - \beta_{ji}^{(L)}\right)x_{ji}^{(L)}. \tag{100}$$

Combining the complementary slackness conditions equation 97-equation 98 with the stationary condition equation 96 implies that:

$$\begin{aligned}\lambda\|U^{(L-1)}\|_F^2 &= \frac{1}{n}\sum_{i=1}^{n}(y_i - \left\langle x_i^{(L)}, v\right\rangle)\left\langle x_i^{(L)}, v\right\rangle = \left\langle \frac{1}{n}\sum_{i=1}^{n}(y_i - \left\langle x_i^{(L)}, v\right\rangle)x_i^{(L)}, v\right\rangle \\ &= \langle \lambda v, v\rangle = \lambda\|v\|^2.\end{aligned}$$

O Proof of Theorem 8

By Lemma 3 and Lemma 4, the squared Frobenius norms of the weight matrices and the squared norm of the output weight vector satisfy

$$\sum_{l=1}^{L-1}\|U^{(l)}\|_F^2 \leq \frac{T_G+1}{4n\lambda}\|y\|^2, \qquad \|v\|^2 \leq \frac{1}{4n\lambda}\|y\|^2.$$

Hence, for the concatenated parameter vector $\mathcal{U}(S) = [\mathrm{vec}(U^{(1)}(S)), \ldots, \mathrm{vec}(U^{(L-1)}(S)), v(S)]$,

$$\|\mathcal{U}(S)\|^2 \leq \frac{T_G + 2}{4n\lambda}\|y\|^2 \leq \frac{(T_G + 2)\, B^2}{4\lambda},$$

where the last inequality uses $\|y\|^2 = \sum_{i=1}^{n} y_i^2 \leq nB^2$. This holds for every training set S, so the assumption of Theorem 1 holds with $c = \sqrt{(T_G + 2)B^2/(4\lambda)}$. The capped squared loss takes values in $[0, 1]$. The vector $w = \mathcal{U}(S)$ depends on the data only through S, so it is independent of V, and Theorem 1 applies to it as a fixed vector. Since $\|w\| \leq c$,

$$4c\|w\| + c^2 \leq 5c^2 = \frac{5\,(T_G + 2)\, B^2}{4\lambda},$$

and substituting this into equation 1 gives

$$\sqrt{\frac{5c^2}{4n\epsilon^2}} = \sqrt{\frac{5\,(T_G + 2)\, B^2}{16\,\epsilon^2 \lambda\, n}},$$

which is equation 18.