On the Sparsity-Promoting Effect of Weight Decay in Deep Learning

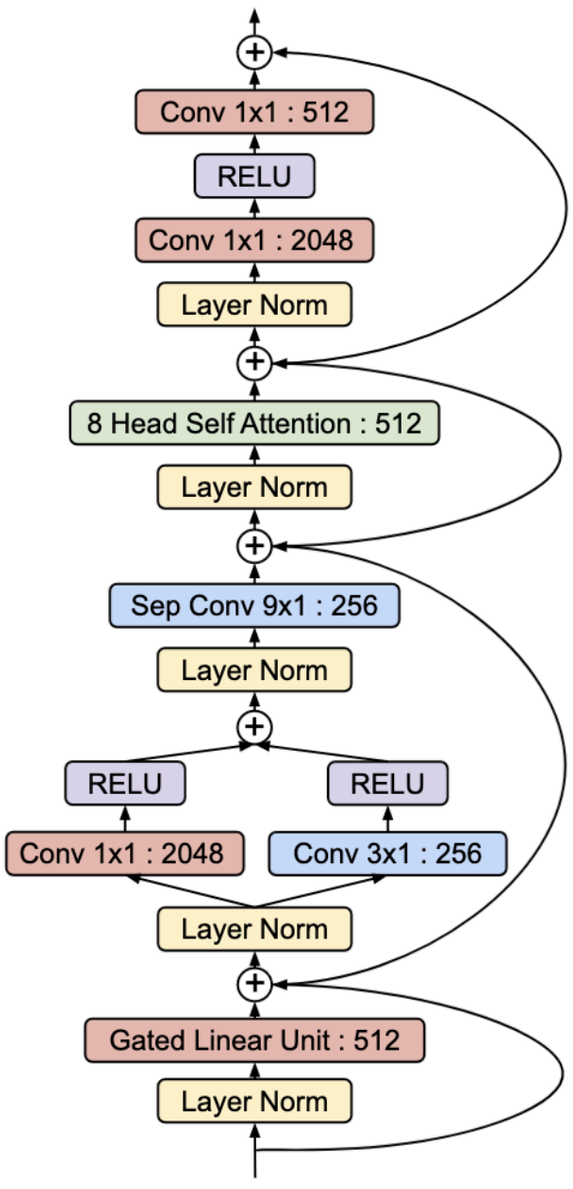
Rahul Parhi
Biomedical Imaging Group
École polytechnique fédérale de Lausanne

Conference on Parsimony and Learning
4 January 2024

Deep Neural Network Architectures

The Evolved Transformer

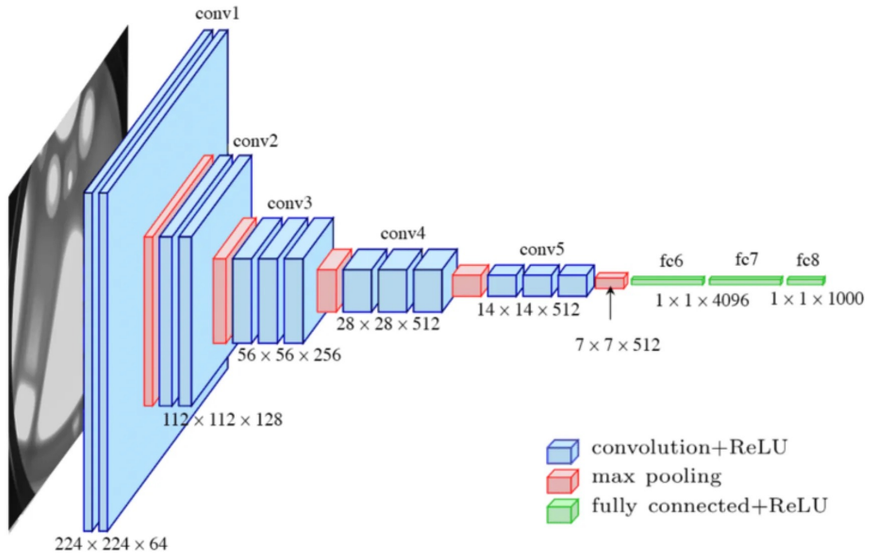
David So, Quoc Le, Chen Liang *Proceedings of the 36th International Conference on Machine Learning*, PMLR 97:5877-5886, 2019.



Google DeepMind 2023-10-26

ConvNets Match Vision Transformers at Scale

Samuel L Smith[1], **Andrew Brock**[1], **Leonard Berrada**[1] **and Soham De**[1]
[1]Google DeepMind

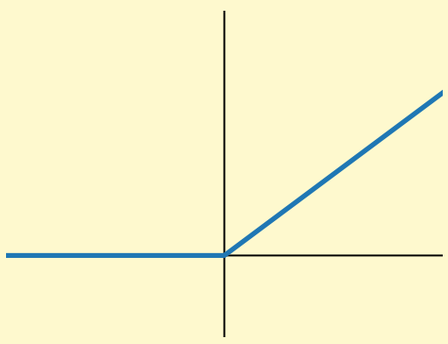


Very deep convolutional networks for large-scale image recognition

K Simonyan, A Zisserman

In this work we investigate the effect of the convolutional network depth on its accuracy in the large-scale image recognition setting. Our main contribution is a thorough evaluation of …

☆ Save 99 Cite Cited by 112161 Related articles »

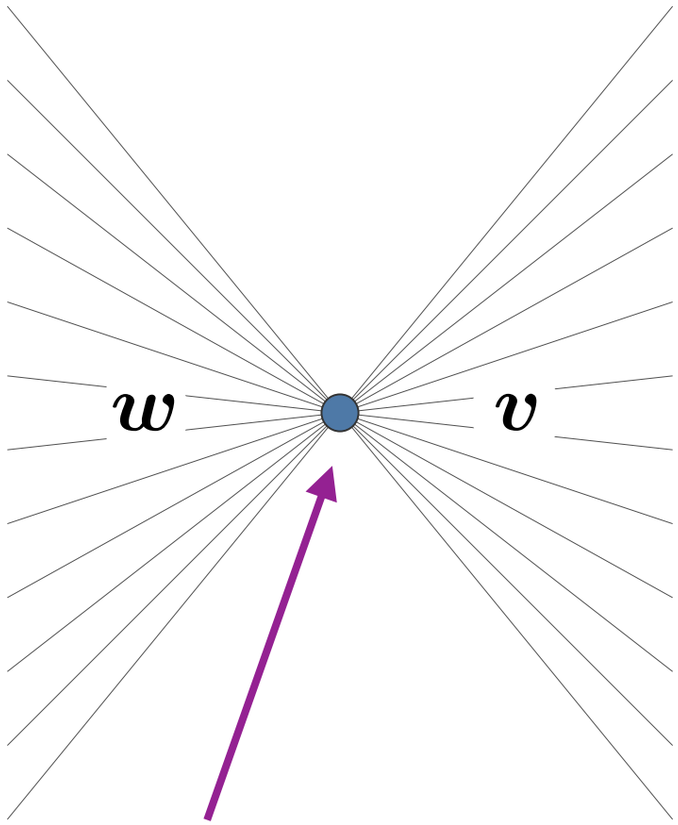


Rectified Linear Unit (ReLU)

$$\mathrm{ReLU}(t) = \max\{0, t\} = t_+$$

$+$ weight decay in training

Neural Balance in Deep Neural Networks



mathematical expression for a single ReLU neuron

$$\boldsymbol{v}(\boldsymbol{w}^\mathsf{T}\boldsymbol{z})_+ \in \mathbb{R}^D$$

weight decay in training is equivalent to adding $\|\boldsymbol{w}\|_2^2 + \|\boldsymbol{v}\|_2^2$ to the training objective

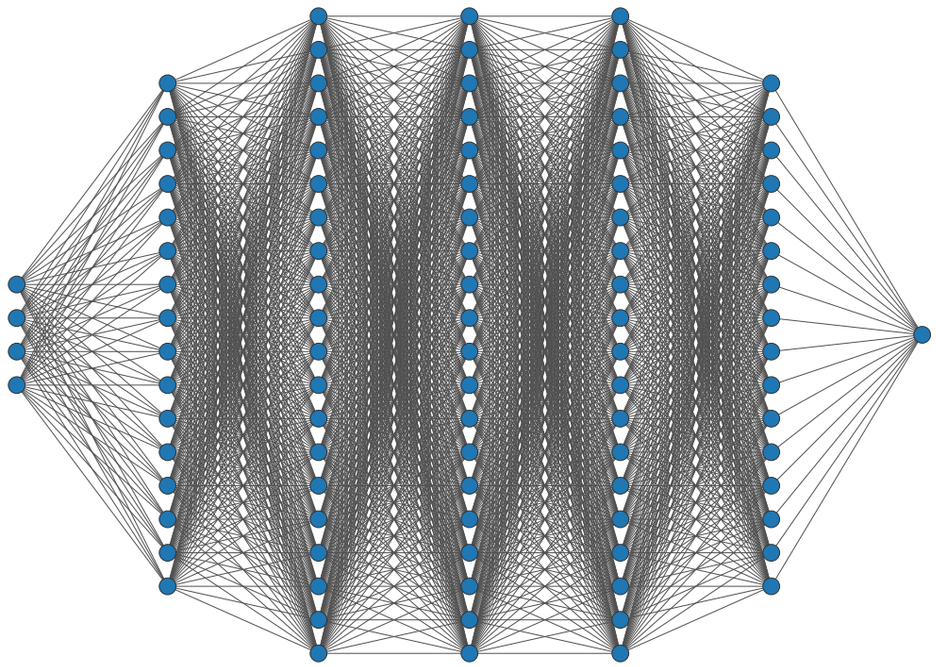
Neural Balance Theorem

If a DNN is trained with weight decay, then the 2-norms of the input and output weights to each ReLU neuron must be **balanced**.

$$\|\boldsymbol{w}\|_2 = \|\boldsymbol{v}\|_2$$

P. and Nowak (2023, IEEE Signal Processing Magazine)

Neural Network Training

$\boldsymbol{x}$  $f_{\boldsymbol{\theta}}(\boldsymbol{x})$

parameterized by a vector $\boldsymbol{\theta} \in \mathbb{R}^P$ of neural network **weights**

Neural network training problem for the data $\{(\boldsymbol{x}_n, y_n)\}_{n=1}^N$.

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^P} \underbrace{\sum_{n=1}^{N} \mathcal{L}(y_n, f_{\boldsymbol{\theta}}(\boldsymbol{x}_n))}_{\text{data fidelity}} + \underbrace{\frac{\lambda}{2} \|\boldsymbol{\theta}\|_2^2}_{\text{regularization}}$$

Neural Balance

The ReLU activation is **homogeneous**

$$\boldsymbol{v}(\boldsymbol{w}^{\mathsf{T}}\boldsymbol{z})_+ = \gamma^{-1}\boldsymbol{v}(\gamma\boldsymbol{w}^{\mathsf{T}}\boldsymbol{z})_+, \quad \text{for any } \gamma > 0.$$

At a global minimizer of the weight decay objective, $\|\boldsymbol{v}\|_2 = \|\boldsymbol{w}\|_2$.

Proof. The solution to

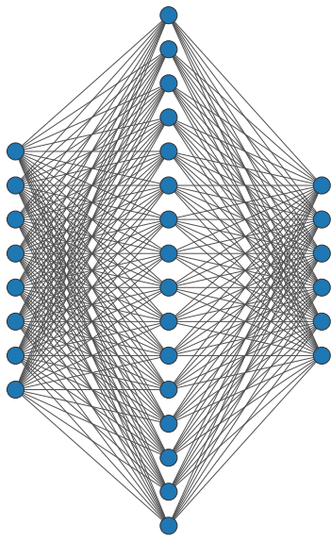
$$\min_{\gamma>0} \|\gamma^{-1}\boldsymbol{v}\|_2 + \|\gamma\boldsymbol{w}\|_2$$

is $\gamma = \sqrt{\|\boldsymbol{v}\|_2/\|\boldsymbol{w}\|_2}$. □

At a global minimizer, $\dfrac{\|\boldsymbol{v}\|_2^2 + \|\boldsymbol{w}\|_2^2}{2} = \|\boldsymbol{v}\|_2\|\boldsymbol{w}\|_2$.

Grandvalet (1998, ICANN)
Neyshabur et al. (2015, ICLR Workshop)

Secret Sparsity/Implicit Parsimony of Weight Decay



$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sum_{k=1}^{K} \boldsymbol{v}_k (\boldsymbol{w}_k^\mathsf{T} \boldsymbol{x})_+$$

$$\boldsymbol{\theta} = \{(\boldsymbol{w}_k, \boldsymbol{v}_k)\}_{k=1}^K$$

weight decay

$$\min_{\boldsymbol{\theta}=\{(\boldsymbol{w}_k,\boldsymbol{v}_k)\}_{k=1}^K} \sum_{n=1}^{N} \mathcal{L}(\boldsymbol{y}_n, f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)) + \frac{\lambda}{2} \sum_{k=1}^{K} \|\boldsymbol{v}_k\|_2^2 + \|\boldsymbol{w}_k\|_2^2$$

path-norm

$$\min_{\boldsymbol{\theta}=\{(\boldsymbol{w}_k,\boldsymbol{v}_k)\}_{k=1}^K} \sum_{n=1}^{N} \mathcal{L}(\boldsymbol{y}_n, f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)) + \lambda \sum_{k=1}^{K} \|\boldsymbol{v}_k\|_2 \|\boldsymbol{w}_k\|_2$$

multitask lasso

$$\min_{\substack{\boldsymbol{\theta}=\{(\boldsymbol{w}_k,\boldsymbol{v}_k)\}_{k=1}^K \\ \|\boldsymbol{w}_k\|_2=1}} \sum_{n=1}^{N} \mathcal{L}(\boldsymbol{y}_n, f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)) + \lambda \sum_{k=1}^{K} \|\boldsymbol{v}_k\|_2$$

Rebalancing

Secret Sparsity/Implicit Parsimony of Weight Decay

$$\text{weight decay} \iff \min_{\substack{\boldsymbol{\theta}=\{(\boldsymbol{w}_k,\boldsymbol{v}_k)\}_{k=1}^K \\ \|\boldsymbol{w}_k\|_2=1}} \sum_{n=1}^N \mathcal{L}(\boldsymbol{y}_n, f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)) + \lambda \sum_{k=1}^K \|\boldsymbol{v}_k\|_2$$

- Weight decay is equivalent to a **non-convex** multitask lasso.

What Kinds of Functions Do Neural Networks Learn?

Why Do Neural Networks Work Well in High-Dimensional Problems?

Principled Compression Algorithms for Pre-Trained DNNs.

A Banach Space Representer Theorem

Neural Network Representer Theorem (**P.** and Nowak 2021)

For any data set $\{(\boldsymbol{x}_n, y_n)\}_{n=1}^N$ and lower semicontinuous $\mathcal{L}(\cdot,\cdot)$, there exists a solution to

$$\min_{f \in \mathscr{R}\,\mathrm{BV}^2} \sum_{n=1}^{N} \mathcal{L}(y_n, f(\boldsymbol{x}_n)) + \lambda \|f\|_{\mathscr{R}\,\mathrm{BV}^2}, \qquad \lambda > 0,$$

that admits a representation of the form

$$f_{\mathrm{ReLU}}(\boldsymbol{x}) = \sum_{k=1}^{K} v_k \underbrace{(\boldsymbol{w}_k^\mathsf{T}\boldsymbol{x} - b_k)_+}_{\text{ReLU neurons}} + \underbrace{\boldsymbol{w}_0^\mathsf{T}\boldsymbol{x} + b_0}_{\text{skip connection}}, \qquad \underbrace{K < N.}_{\text{sparse solution}}$$

Training a **sufficiently parameterized** neural network ($K \geq N$) with weight decay (to a global minimizer) is a solution to the Banach space problem.

Neural networks learn $\mathscr{R}\,\mathrm{BV}^2$-functions.

P. and Nowak (2021, Journal of Machine Learning Research)

Minimax Optimality of Neural Networks

Suppose that $\{\boldsymbol{x}_n\}_{n=1}^N$ are i.i.d. uniform on a bounded domain $\Omega \subset \mathbb{R}^d$. If $y_n = f^\star(\boldsymbol{x}_n) + \varepsilon_n$ with $\|f^\star\|_{\mathscr{R}\mathrm{BV}^2} < \infty$, then any solution to

$$f_{\mathrm{ReLU}} \in \arg\min_{\boldsymbol{\theta}} \sum_{n=1}^N \mathcal{L}(y_n, f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)) + \frac{\lambda}{2} \sum_{k=1}^K |v_k|^2 + \|\boldsymbol{w}_k\|_2^2$$

weight decay objective

satisfies

$$\mathbf{E}\|f^\star - f_{\mathrm{ReLU}}\|_{L^2(\Omega)}^2 = \widetilde{O}(N^{-\frac{d+3}{2d+3}}) = \widetilde{O}(N^{-\frac{1}{2}}).$$

no curse

minimax rate

Linear methods (thin-plate splines, kernel methods, neural tangent kernels, etc.) **necessarily** suffer the curse of dimensionality.

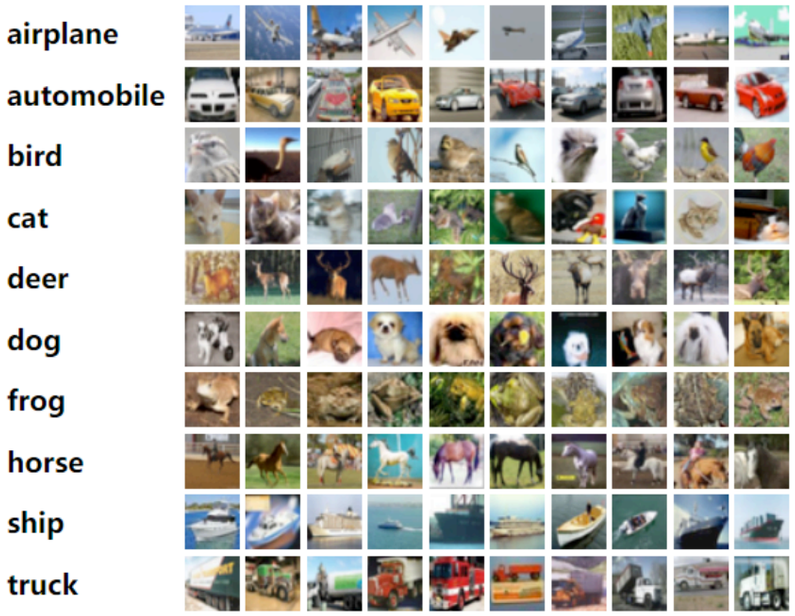
Linear minimax lower bound: $N^{-\frac{3}{d+3}}$

the curse

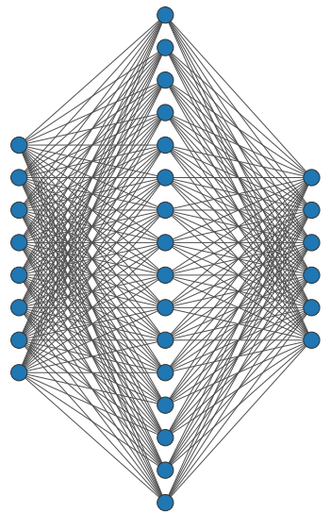
P. and Nowak (2023, IEEE Transactions on Information Theory)

Principled DNN Compression

VGG-19 trained with weight decay on CIFAR-10.



final ReLU layer
$K = 512$ neurons



output dimension
$D = 10$

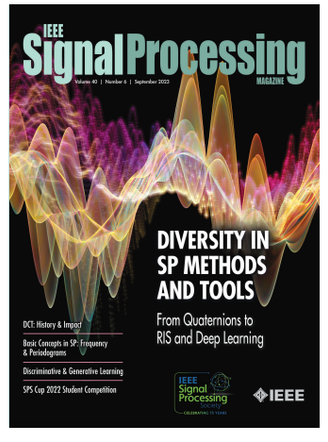
Theory: There exists a representation with ≤ 100 neurons that can be found by solving a **convex** multitask lasso problem.

	original network	compressed network
active neurons	512	47
test accuracy	93.92%	93.88%
train loss	0.0104	0.0112

$10\times$ compression!
no change in performance!

Shenouda, **P.**, Lee, and Nowak (2023+)

Conclusion



- General audience overview article:
  - *Deep Learning Meets Sparse Regularization: A signal processing perspective*
    Rahul Parhi and Robert D. Nowak
    IEEE Signal Processing Magazine, vol. 40, no. 6, pp. 63–74, Sept. 2023.
- Technical articles:
  - *Banach Space Representer Theorems for Neural Networks and Ridge Splines*
    Rahul Parhi and Robert D. Nowak
    Journal of Machine Learning Research, vol. 22, no. 43, pp. 1–40, 2021.
  - *What Kinds of Functions Do Deep Neural Networks Learn? Insights from Variational Spline Theory*
    Rahul Parhi and Robert D. Nowak
    SIAM Journal on Mathematics of Data Science, vol. 4, no. 2, pp. 464–489, 2022.
  - *Near-Minimax Optimal Estimation With Shallow ReLU Neural Networks*
    Rahul Parhi and Robert D. Nowak
    IEEE Transactions on Information Theory, vol. 69, no. 2, pp. 1125-1140, Feb. 2023.
  - *Vector-Valued Variation Spaces and Width Bounds for DNNs: Insights on Weight Decay Regularization*
    Joseph Shenouda, **Rahul Parhi**, Kangwook Lee, Robert D. Nowak
    arXiv preprint arXiv:2305.16534, 2023+

Questions?

The Structured Sparsity of Weight Decay

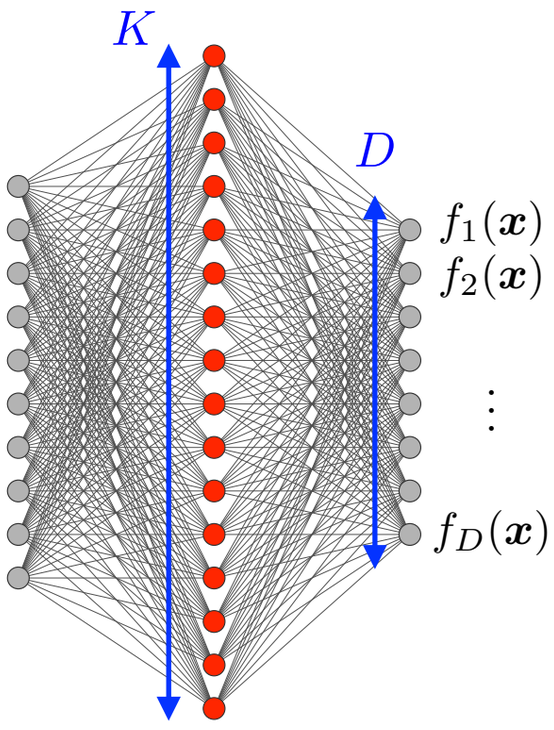
$$\min_{\substack{\boldsymbol{\theta}=\{(\boldsymbol{w}_k,\boldsymbol{v}_k)\}_{k=1}^K \\ \|\boldsymbol{w}_k\|_2=1}} \sum_{n=1}^{N} \mathcal{L}(\boldsymbol{y}_n, f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)) + \lambda \sum_{k=1}^{K} \|\boldsymbol{v}_k\|_2$$

weight decay

$\Longleftrightarrow$

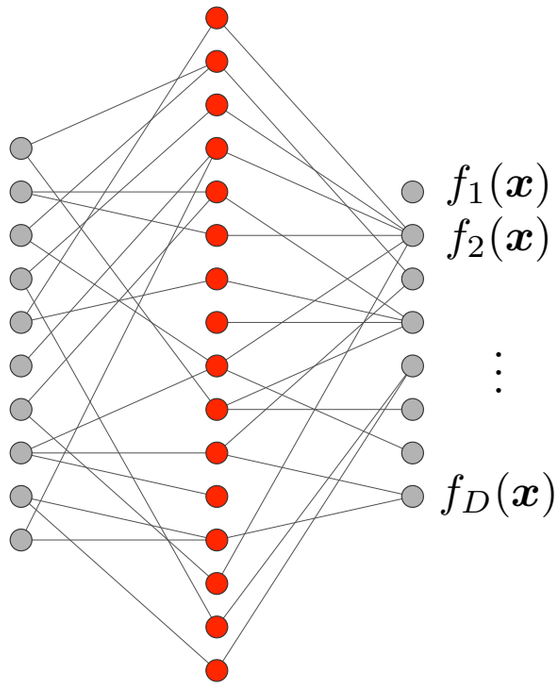
non-convex multitask lasso

dense weights



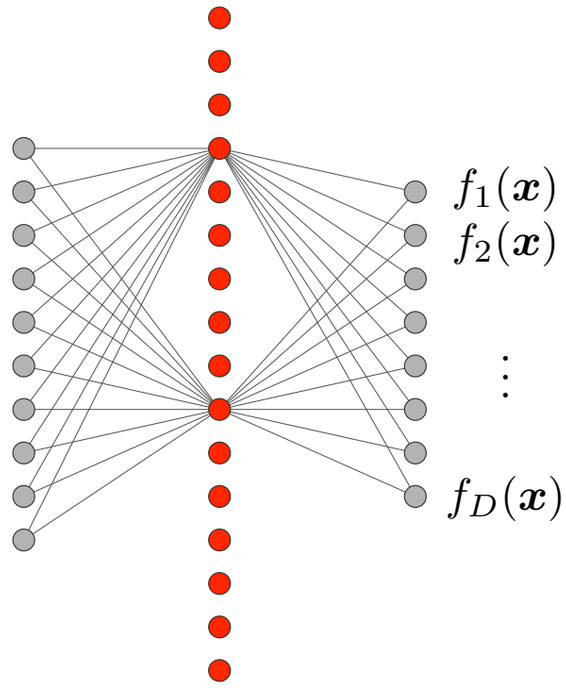
$\|f\|_{\mathcal{V}} = O(K\sqrt{D})$

sparse weights



$\|f\|_{\mathcal{V}} = O(D)$

sparse neurons

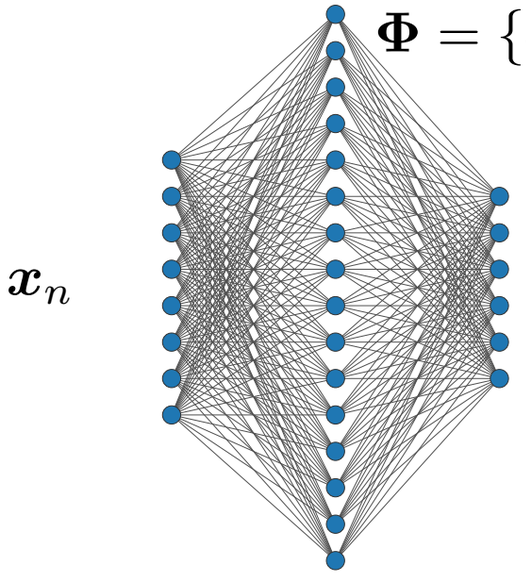


$\|f\|_{\mathcal{V}} = O(\sqrt{D})$

Weight decay favors variation in only a few directions (sparse weights)

Weight decay favors outputs that "share" neurons (sparse neurons)

Tight Bounds on Widths



$$\min_{\{\boldsymbol{v}_k\}_{k=1}^K} \sum_{k=1}^K \|\boldsymbol{v}_k\|_2 \quad \text{s.t.} \quad \boldsymbol{\Psi} = \mathbf{V}\boldsymbol{\Phi}.$$

Low-rank data embeddings have been observed empirically by Huh et al. (2022).

Layer Width Theorem (Shenouda, **P.**, Lee and Nowak 2023+)

Let $\boldsymbol{\Phi}$ denote the post-activation features and $\boldsymbol{\Psi}$ denote the neuron outputs of any ReLU layer in a **trained** DNN (minimizes the weight decay objective). Then, there exists a representation with

$$K \leq \operatorname{rank}(\boldsymbol{\Phi}) \operatorname{rank}(\boldsymbol{\Psi}) \leq N^2$$

Bound of Jacot (2023): $N(N+1)$.

neurons. The representation can be found by solving a **convex multitask lasso** problem.

Shenouda, **P.**, Lee, and Nowak (2023+)