CSCI 145 Problem 13: A Residual Network Is an Ensemble of 2^L Paths

July 20, 2026

Submission Instructions

Please upload your work to Gradescope by **11:59pm on DATE**.

- Turn in a single PDF of *handwritten* solutions. Typed (including LaTeX) solutions will not be accepted.
- Problems are grouped into three-week **cycles**. Once per cycle, you will meet with the grader and present the solution to *one* problem from that cycle, selected at random when you arrive — so bring your work and be ready to explain any of them.

Problem 13: A Residual Network Is an Ensemble of 2^L Paths

A residual network 50 blocks deep is usually described as one very deep network, but it is better described as a collection of roughly 10^{15} shallower ones, of which only a vanishing fraction matter. Lecture kept one identity path through each block; here we count *all* of the paths, and the counting explains why residual networks are so forgiving to train.

Throughout, the stack has L blocks of one width d, so every matrix is $d \times d$: $\mathbf{I}$ is the identity and $\mathbf{J}_F^{(l)}$ is the Jacobian of block l's learned part $F^{(l)}$, as in lecture. By the chain rule, the stack's end-to-end Jacobian is $\prod_{l=1}^{L} \left(\mathbf{I} + \mathbf{J}_F^{(l)}\right)$.

Part A: Two Blocks, by Hand

Take $L = 2$ and expand $\left(\mathbf{I} + \mathbf{J}_F^{(1)}\right)\left(\mathbf{I} + \mathbf{J}_F^{(2)}\right)$ by hand into four terms. Identify each term with a "path" through the two blocks: which term skips both blocks entirely, which takes both blocks' learned transformations, and which two take one block's shortcut and the other's learned path?

Part B: L Blocks, 2^L Paths

Generalize Part A. Show by induction on L (Part A is the case $L = 2$) that

$$\prod_{l=1}^{L} \left(\mathbf{I} + \mathbf{J}_F^{(l)}\right) = \sum_{T \subseteq \{1,\ldots,L\}} \prod_{l \in T} \mathbf{J}_F^{(l)},$$

where each inner product is taken in increasing order of l, and the empty product is $\mathbf{I}$. Each of the 2^L subsets T names one path: through the learned part of every block in T, and along the identity shortcut through every block outside it. Which subset is the all-shortcuts path, the one term that survives no matter what the blocks learn?

Part C: The Gradient Concentrates on Short Paths

Suppose every block's learned Jacobian is small, $\|\mathbf{J}_F^{(l)}\|_2 \le \epsilon$ for some $\epsilon < 1$, a fair description early in training. (Recall that $\|\cdot\|_2$ is the largest singular value, the SVD quantity this course has tracked since the power method.) Using the triangle inequality and submultiplicativity of the spectral norm ($\|\mathbf{AB}\|_2 \le \|\mathbf{A}\|_2 \|\mathbf{B}\|_2$), show that the paths through exactly k blocks contribute at most

$$\left\| \sum_{|T|=k} \prod_{l \in T} \mathbf{J}_F^{(l)} \right\|_2 \le \binom{L}{k} \epsilon^k.$$

Summing over k and applying the binomial theorem recovers lecture's bound $(1+\epsilon)^L$ for the whole stack, so no mass has gone missing. Now fix $L = 50$ and $\epsilon = 0.1$, where that total is $1.1^{50} \approx 117$, and evaluate $\binom{L}{k}\epsilon^k$ at $k = 1$, $k = 5$, and $k = 25$. Conclude in one sentence where essentially all of the end-to-end Jacobian's mass lives: on paths through many of the 50 blocks, or on paths through a handful of them?

Where this goes next. A sum over exponentially many paths with the mass on the short ones is why a residual network of depth L trains like a network of much smaller *effective* depth, and why it behaves like an ensemble of shallower networks, one per short path, all sharing the same weights. Every transformer in the Architectures unit is a residual stack, which is why those models can be tens or even hundreds of blocks deep.