LARGER LANGUAGE MODELS PROVABLY GENERALIZE BETTER

Anonymous authors

Paper under double-blind review

ABSTRACT

Why do larger language models generalize better? To explore this question, we develop generalization bounds on the pretraining objective of large language models (LLMs) in the compute-optimal regime, as described by the Chinchilla scaling laws. We introduce a novel, fully empirical Freedman-type martingale concentration inequality that tightens existing bounds by accounting for the variance of the loss function. The generalization bound can be broken into three contributions: the number of parameters per token, the loss variance, and the quantization error at a fixed bitrate. As language models are scaled up, the number of parameters per data point stays constant; however, both the loss variance and the quantization error decrease, implying that larger models should have *smaller* generalization gaps. We examine why larger models tend to be more quantizable from an information theoretic perspective, showing that the rate at which they can integrate new information grows slower than their capacity on the compute optimal frontier. From these findings we produce a scaling law for the generalization gap, showing that our bounds decrease in a predictable way.

1 INTRODUCTION

Large language models (LLMs) have demonstrated a remarkable general purpose problem solving capacity across a wide range of complex tasks which humans are able to perform, from classical NLU (Brown, 2020), forecasting (Gruver et al., 2023), mathematics (Trinh et al., 2024), spatial reasoning (Patel & Pavlick, 2022), and many other areas. For a large majority of individual tasks, model capabilities increase monotonically as the next token prediction loss from the pretraining objective decreases.

A conceptually useful story about the learning process involves the model accommodating predictive subprograms of progressively larger computational depth and complexity. During pretraining, shallow details are absorbed first: the log likelihood loss is most easily decreased by learning word frequencies, syntax, and grammar. As these details are absorbed, slightly higher level structures such as facts, relations, and idioms then become the next batch of the lowest hanging fruit, giving way to yet higher level structures and pattern matching.

For reasons that are not yet well understood, this process is reflected in the pretraining objective as a power law for LLMs and other generative models on natural data. The frontier of the best achievable performance given a fixed computational budget C obeys a fixed and predictable power law relationship $L(C) \propto C^{-\alpha}$ over many orders of magnitude (Kaplan et al., 2020). This relationship appears to vary considerably with the kind of data (Henighan et al., 2020) (e.g. natural text vs images vs math) and only weakly on the details of the model architecture and training method (Bahri et al., 2021).

Effort in quantifying *what* this relationship is in a given domain and *how* it varies as model size and dataset size are traded off has been extremely valuable in guiding where resources are spent in constructing more capable AI models (Brown, 2020; Besiroglu et al., 2024; OpenAI, 2023; Dubey et al., 2024) and charting a path for the future. In this work, we target the *why* of scaling laws. While mathematically simple toy models or toy data are valuable, we aim to study the *why* of scaling laws on real models and real data by focusing on just one contribution to the scaling law curve: the token-wise generalization gap. Constructing a generalization bound sensitive enough to capture the

small differences between architectures and yet simple enough to write down in a short formula is likely impossible; however, even the broad strokes of behavior such as how generalization scales with compute have not been addressed. Thus, here we focus high level understanding rather than algorithmic intervention.

In order to construct the relevant generalization bounds, we introduce a novel empirical Freedman concentration inequality (Freedman, 1975). Our generalization bound highlights three critical components—the ratio of parameters per token in compute-optimal scaling (which is roughly constant), the token-wise loss variance (which decreases with model size), and the performance gap between quantized and unquantized models (which also decreases with model size). As an alternative to quantization, we bound the information transfer between dataset and the model, showing that the information content in the model grows sublinearly with model size, and thus the complexity decreases with model size. These components collectively contribute to a predictable reduction in the generalization gap as models grow larger.

2 Background

2.1 Generalization Bounds

At a high level, we are interested in the expected test error (population risk) $\mathbb{E}_{X' \sim p_{\mathcal{D}}}[R_{h(X)}(X')]$ for a given model (hypothesis) h depending on the training set X but evaluated on a test set X' sampled from the data distribution $p_{\mathcal{D}}$. One conceptually convenient way of breaking down this quantity is into the irreducible error, approximation gap, and generalization gap:[1]

$$\mathbb{E}_{X' \sim p_{\mathcal{D}}}[R_{h(X)}(X')] = \underbrace{R_*(X)}_{\text{Irreducible Error } E} + \underbrace{R_{h(X)}(X) - R_*(X)}_{\text{Approximation Gap } A} + \underbrace{\mathbb{E}_{X' \sim p_{\mathcal{D}}}[R_{h(X)}(X')] - R_{h(X)}(X)}_{\text{Generalization Gap } G}.$$

The first term describes the entropy of natural text, e.g. the amount of truly random information content in the data, which cannot be further explained even when knowing the true data generating process. The second term describes the approximation gap, capturing the extent to which the trained model is able to fit the training data. This term combines both model capacity, e.g. as described by universal approximation theorems (Cybenko, 1989), as well as optimization via how well the training algorithm is able to find the given solution. Finally, we have the generalization gap, capturing the extent to which training and testing performance diverge on account of overfitting to the statistically irrelevant regularities in X. Though generalization bounds focus on the last term, all three quantities are of interest for understanding LLM behavior. Empirically, it has been observed that the generalization gap for LLMs (at least in the low epoch regime) tends to be extremely small or even negligible compared to the other two terms.

Among the simplest generalization bounds is the finite hypothesis with prior generalization bound applied to IID data (Shalev-Shwartz & Ben-David, 2014). With probability at least $1 - \delta$,

$$\mathbb{E}_{X' \sim p_{\mathcal{D}}}[R_{h(X)}(X')] - R_{h(X)}(X) \leq \Delta \sqrt{\frac{\log 1/P(h) + \log 1/\delta}{2m}}$$

where m is the number of IID data points, Δ is an upper bound on the range of values the risk can take, and $P(h)$ is a prior distribution over hypotheses in a discrete hypothesis class $\mathcal{H}$. With a judicious choice of prior, $\log 1/P(h)$ can be related to the compressed size of the model measured in nats (Lotfi et al., 2022).

During text pretraining, the individual tokens are not sampled IID. Thus, a generalization bound requires treating entire documents (often thousands of tokens) as the elements the empirical risk is computed over. Note that modern language models have hundreds of times more parameters than documents they were trained on. With the help of very extreme compression methods and using smoothing to bound Δ, it is possible to construct nonvacuous bounds (Lotfi et al., 2024a). However, the required compression (greater than 100 times) is so severe that it cripples model performance.

[1] We note this differs from the commonly referred to estimation-approximation error breakdown (Bottou & Bousquet, 2007) or the bias-variance decomposition (Brown & Ali, 2024); however, the train error-generalization gap is more useful for our purposes.

In a recent work, Lotfi et al. (2024c) explore breaking down generalization into tokenwise generalization, e.g. how the loss varies with each individual predicted token being resampled under the distribution but keeping the context the same. Splitting up the training dataset X into the sequence of tokens $[X_k]_{k=1}^D$, the authors bound

$$T = \frac{1}{D}\sum_{k=1}^{D} \mathbb{E}[R_h(X_k \mid X_{<k}) \mid X_{<k}] - R_h(X),$$

where $R_h(X_k \mid X_{<k})$ is the negative log likelihood for token k given the context $X_{<k}$, and the expectation is taken with respect to $p(X_k|X_{<k})$ from the data distribution. The authors bound T using Azuma's inequality to arrive at a bound scaling as $\Delta\sqrt{\frac{\log 1/P(h)}{2D}}$. We improve upon this bound, reducing to a leading term of $\Delta\frac{\log 1/P(h)}{D}$.

2.2 Chinchilla Scaling Laws

A key insight from the current machine learning paradigm is that the dataset should not be considered a fixed quantity. Rather than optimizing to find the best model for a given dataset, one should instead try to find the best performing model and dataset for a given computational budget. Muennighoff et al. (2024) describes the optimal allocation of resources for increasing the size of the model and increasing the size of the dataset under the assumption that data is plentiful relative to the computational budget.

Let N be the number of parameters and D be the number of training tokens. In the one epoch regime of LLM pretraining, the negative log likelihood loss is well-approximated by the power law

$$R(N, D) = E + \frac{A}{N^\alpha} + \frac{B}{D^\beta},$$

where A, B are empirically estimated constants, exponents α, β have similar values, and E is the irreducible error. Optimizing $N(C)$ and $D(C)$ under the constraint of a fixed compute budget $C \approx 6ND$ (Kaplan et al., 2020), one arrives at

$$N^*(C) = G(C/6)^a \quad D^*(C) = G^{-1}(C/6)^b$$

for constants G, a, b that depend on A, B, α, β (Hoffmann et al., 2022).

Within the margin of statistical error, we have $a = b = 0.5$ in the optimal allocation of compute (Besiroglu et al., 2024). Therefore, the ratio of parameters per token, $N^*(C)/D^*(C) = G^2$, is a fixed constant. Evaluating the constants from Muennighoff et al. (2024), we have $G^2 \approx 1/20$. Note that many open source models optimize performance amortized over both training time and inference time compute, which leads to smaller than Chinchilla optimal models, e.g. models with a ratio $N/D < G^2$. In the context of this paper, we will assume the Chinchilla optimal scaling $N/D = G^2$ with the understanding that any generalization bounds we construct would only be tighter if the ratio N/D is smaller.

To test our theory, we use the open source Pythia model family (Biderman et al., 2023) ranging from 70 million to 12 billion parameters. Unlike other open source LLMs, we have full access to both the Pythia model checkpoints from training and the Pile dataset they were trained on (Gao et al., 2020). From these intermediate checkpoints, we choose the set of models along the compute optimal frontier to match $N/D = 1/20$, reflecting the choice for number of training steps and model size that one would have made optimizing only for performance at the given computational budget. The chosen checkpoints are plotted in the training frontier of these models in Figure 1.

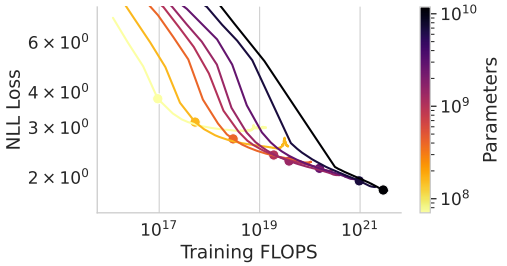


Figure 1: Pythia models and checkpoints chosen along the compute optimal frontier (checkpoints given by the marked values).

3 GENERALIZATION BOUND

In this section, we will build up the components for our generalization bounds. To capture the relevant behavior, we derive a new concentration inequality for martingales. We apply a prior weighted union bound to this concentration inequality so that we can apply it to models in a large hypothesis class, taking advantage of the low complexity inherent in compressible models. Bounding the worst case loss behavior using prediction smoothing, we apply this bound to LLMs.

3.1 AN EMPIRICAL FREEDMAN'S CONCENTRATION INEQUALITY

Theorem 3.1. *Let $X_1, \ldots, X_n$ be a sequence of $\mathcal{F}_k$-measurable random variables. Let $Y_1, \ldots Y_n$ be any other sequence of $\mathcal{F}_{k-1}$-measurable random variables such that the difference is bounded below: $A_k = (Y_k - X_k)/\Delta > -1$ for some $\Delta \geq 0$. Let K be a finite subset of $(0, 1)$. Then, with probability at least $1 - \delta$,*

$$\frac{1}{n}\sum_{k=1}^{n}\left(\mathbb{E}[X_k \mid \mathcal{F}_{k-1}] - X_k\right) \leq \Delta C + \Sigma\sqrt{C}, \tag{1}$$

where $C := \frac{1}{n}\log\frac{|K|}{\delta}$, and

$$\Sigma = \Sigma(C, \Delta, \{X_k - Y_k\}_{k=1}^n, K) := \min_{s \in K} \Delta\sqrt{C}(1-s)/s + \frac{\Delta}{\sqrt{C}}\frac{1}{n}\sum_{k=1}^{n} v\big(sA_k\big)/s$$

and $v(x) = x - \log(1 + x)$.

Proof: See Section A.1. This concentration inequality states that the sequence of random variables concentrates on their conditional means with a term Σ depending on the empirical variation of the loss value. We note that Σ can be viewed as a variance term. As we show in Appendix A, using a small K, the variance proxy can be upper bounded: $\Sigma \leq 2\sqrt{\frac{1}{n}\sum_{k=1}^{n}(X_k - Y_k)^2}$. This is explicitly related to the empirical variance but with the mean replaced by Y_k. Although the minimization form above is unwieldy, it produces significantly tighter estimates of Σ (a factor of ~ 5x smaller). When the loss variation Σ is small, concentration happens at a rate linear in the complexity C rather the slower $\sqrt{C}$ rate.

This concentration inequality provides the core result for our generalization bounds, and to the best of our knowledge it is the first martingale concentration inequality to incorporate a variance term which can be evaluated on the original data. We can view this bound as aiming to achieve the benefits that Freedman's inequality has over Azuma's inequality while being fully empirical, replacing the population variance with a fully empirical proxy. Our approach is analogous to the fully empirical Bernstein bound derived in Maurer & Pontil (2009), but in the martingale rather than IID setting. Unfortunately, the proof technique of Maurer & Pontil (2009) does not carry over to the martingale case and instead we take quite a different approach. We derive our concentration inequality in Theorem A.5 making use of a proxy Y_k that is $\mathcal{F}_{k-1}$-measurable but which can take the place of $\mathbb{E}[X_k \mid \mathcal{F}_{k-1}]$ in the variance. In practice, we choose this quantity to be the mean of the model NLL under resampling of the given token according to the *model* distribution in place of the data distribution.

3.2 EXTENDING TO A DISCRETE HYPOTHESIS CLASS

From the concentration inequality in Equation 1, we derive the following discrete hypothesis class generalization bound.

Theorem 3.2. *Let $X_1, \ldots, X_n$ be a sequence of (possibly dependent) random variables. Let $R_h(X_k \mid X_{<k})$ denote the risk for element X_k given the previous elements of the sequence for hypothesis h in a countable hypothesis class $\mathcal{H}$. Let $p_h(X_k \mid X_{<k})$ be any (hypothesis dependent) distribution over X_k conditioned on $X_{<k}$. Consider a prefix free coding of each $h \in \mathcal{H}$ and let $L(h)$ be the length of that code measured in nats. Let K be a finite subset of $\mathbb{R}^+$. Assuming $R_h(X_k \mid X_{<k}) - \mathbb{E}_{Y_k \sim p_h}[R_h(Y_k \mid X_{<k})] \leq \Delta$ for some $\Delta > 0$, we have that simultaneously for all $h \in \mathcal{H}$, with probability at least*

$1-\delta$,

$$\frac{1}{n}\sum_k \mathbb{E}[R_h(X_k \mid X_{<k}) \mid X_{<k}] \leq \frac{1}{n}\sum_k R_h(X_k \mid X_{<k})) + \Delta\mathcal{C} + \Sigma\sqrt{\mathcal{C}}, \tag{2}$$

where the complexity $\mathcal{C}$ is given by

$$\mathcal{C} := \frac{L(h) + \log|K|/\delta}{n}$$

and $\Sigma = \Sigma(\mathcal{C}, \Delta, \{A_k\}_{k=1}^n, K)$ from Theorem 3.1 for $A_k = R_h(X_k \mid X_{<k}) - \mathbb{E}_{Y_k \sim p_h}[R_h(Y_k \mid X_{<k})]$.

Proof: See Section A.2.

3.3 Worst Case Behavior and Smoothing

The last component of our bounds is the smoothing to bound the worst case behavior of the model, which in general for the negative log likelihood can be arbitrarily large. We employ the prediction smoothing idea from Lotfi et al. (2024a), where the model is mixed with a uniform distribution over the tokens with a given mixing parameter. Unlike application in previous work, we optimize over this parameter analytically so that we can remove it from the bounds and evaluation entirely, instead merely as a tool for constructing bounds while all evaluations are with the unsmoothed model.

Theorem 3.3. *For the categorical negative log likelihood objective $\hat{R}_h = -\frac{1}{n}\sum_{k=1}^n \log p_h(X_k \mid X_{<k})$ on V classes and $C \in \mathbb{R}^+$, there exists a prediction smoothed model $p_s(\cdot) = (1-\alpha)p_h(\cdot) + \alpha/V$ which has a worst case loss $\Delta_s = \sup_{X_k, X_{<k}} -\log p_s(X_k \mid X_{<k}) \leq \log(V/\alpha)$, and the risk satisfies*

$$\hat{R}_s + C\Delta_s \leq \hat{R}_h + C\log V + \sqrt{2C}, \tag{3}$$

for some value $\alpha \in (0,1)$ (approximately $C/(1+C)$).

The proof is provided in Section A.3.

3.4 Generalization Bound for Chinchilla Language Models

Finally, we assemble these three components into a generalization bound that we can empirically evaluate for language models. Combining the prediction smoothing bound with Theorem 3.2 applied to the smoothed quantized model produces the result of Theorem 3.4. Note that each term in the expression has an interpretable meaning.

Theorem 3.4. *Let $X_1, \ldots, X_D$ be the sequence of D (possibly dependent) tokens formed from concatenating each sequence in the dataset together into a single stream of tokens. Let $R_h(X_k \mid X_{<k}) = -\log p_h(X_k \mid X_{<k})$ denote the NLL for element X_k given the previous elements for a given model h and with vocabulary size V and N parameters. Let $\hat{R}_h = \frac{1}{D}\sum_{k=1}^D R_h(X_k \mid X_{<k})$ be the empirical risk and $R_h = \frac{1}{D}\sum_{k=1}^D \mathbb{E}[R_h(X_k \mid X_{<k}) \mid X_{<k}]$ be the tokenwise expected risk for that model. Let K be a finite subset of $(0,1)$. For any given quantization q of h using b bits per parameter with expected risk R_q, there exists a label smoothed model sq with $R_{sq}(X_k \mid X_{<k}) = (1-\alpha)R_q(X_k \mid X_{<k}) + \alpha/V$ for fixed $\alpha \in (0,1)$ which achieves a tokenwise population risk with probability $1-\delta$*

$$R_{sq} \leq \hat{R}_h + \underbrace{\mathcal{C}\log V}_{\text{Random Guess NLL}} + \underbrace{\Sigma\sqrt{\mathcal{C}}}_{\text{Loss Variation}} + \underbrace{\sqrt{2\mathcal{C}}}_{\text{Smoothing Cost}} + \underbrace{(\hat{R}_q - \hat{R}_h)}_{\text{Quantization Gap}}, \tag{4}$$

where the complexity $\mathcal{C}$ is given by

$$\mathcal{C} = \left(\tfrac{N}{D}\right)b\log 2 + \tfrac{1}{D}\log\tfrac{|K|}{\delta},$$

and $\Sigma = \Sigma(\mathcal{C}, \Delta, \{A_k\}_{k=1}^n, K)$ (defined in Theorem 3.2) which can be upper bounded in terms of the empirical loss variance

$$\Sigma \leq 2\sqrt{\tfrac{1}{D}\sum_{k=1}^D \left(R_q(X_k \mid X_{<k}) - \mathbb{E}_{Y_k \sim p_h}[R_q(Y_k \mid X_{<k})]\right)^2}.$$

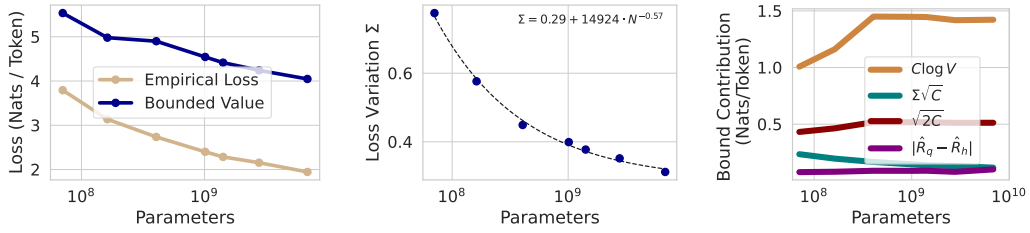


Figure 2: **Left:** A direct comparison of our evaluated generalization bound, and the empirical loss as a function of model scale. As the model is scaled up, our bound improves just like the empirical loss. **Center:** Loss variation Σ entering into the generalization bound. As the loss deviation decreases, so does the largest term in our bound. **Right:** Comparison of the relative scale of the contributions to Theorem 3.4.

To make sense of the bound, let's consider the various terms. The bounded quantity on the left hand side, R_{sq}, is the expected tokenwise risk of the smoothed and quantized version of hypothesis h. The bound is written in terms of the empirical risk of the original model h, with $\hat{R}_q - \hat{R}_{sq}$ controlled by the smoothing cost and quantization gap. Typically, the largest contribution to the bound is $\mathcal{C} \log V$, e.g. the complexity times the negative log likelihood of random guessing. The loss variation relates to how spread the empirical loss is and can be seen as a model realizability term. If there existed a 0 loss model in the model class, then this term could be brought to 0. Given the nonzero entropy of natural text, this will not be the case; however, as models improve and approach the irreducible error, so too will the empirical loss variation.

In this setup, the complexity $\mathcal{C}$ is just the ratio of parameters to data points, $\frac{N}{D} = G^2$, times the number of bits per parameter used in the quantization b, plus a negligible additional term. The decreased complexity of using fewer bits for b trades off with the quantization gap, and in principle this parameter should be optimized to achieve the best bound. As all terms in the expression can be evaluated empirically, we can determine how much of the empirical observation it can explain and how much remains to be understood.

To get a sense for the scale of the different terms, consider the following typical scenario. For simplicity, suppose $V = 50000$, $\log V \approx 11$, $b = 3$, $G^2 = 1/20$, which yields $\mathcal{C} \approx 1/9$. Σ varies with model scale but is of scale $1/10$, and the quantization gap is around $1/10$. Evaluating these terms, $R_{sq} - \hat{R}_h \leq 11/9 + 1/30 + 1.4/3 + 1/10 \approx 1.8$ nats per token, and we see that the random guess and smoothing terms contribute most to the size of the bound. To put this value into perspective, the empirical risk $\hat{R}_h$ itself is around 2 nats per token and the boundary between vacuous and nonvacuous bounds is at $\log V \approx 11$ nats per token.

4 Empirical Evaluation

As Theorem 3.4 is fully empirical, we simply need to empirically evaluate the loss variation term Σ along with the quantization gap and we can evaluate the generalization bound. We compute these quantities on the given Pythia checkpoints on the Pile dataset on which they were trained and quantized using GPTQ (Frantar et al., 2023) to $b = 4$ bits per parameter, and we evaluate the bounds with failure probability $\delta = 0.01$. The results are shown in Figure 2 for the Chinchilla compute optimal checkpoints within the Pythia model family. Additional evaluation details are provided below.

We compute Σ with K given by 1000 equally spaced points between $[0, 1]$ excluding the endpoints. We estimate the risk and loss variation on an IID subsample from the collection of token-context pairs in the training dataset of size 10^4 and bound the difference with standard techniques. We note that largest 12B parameter model failed to quantize properly (possibly due to the learning rate drop as it was the only checkpoint taken at the end of training) and so we removed it from the evaluation. As we must pick out the compute optimal checkpoints from training runs that were not designed for this purpose, for the two smallest models (70m and 160m parameters), which the relevant region of

the training curve is more sparsely sampled with checkpoints, the closest models consistently have too small a value of N/D, biasing these two initial data points towards lower values and we urge the reader to keep this in mind while reading the plots.

In Figure 2, we can observe several points. In Figure 2 center, we see that the loss variation decreases with model size in a predictable way, also obeying a power law relationship with a constant offset. As additional compute is spent, the models explain a larger fraction of the variance in the data and the loss variation decreases, but like with the irreducible error there is a minimum value that it is converging towards. Beyond language models, we would expect to see this behavior also, but the predictable improvements with compute give it the simple relationship observed here. In Figure 2 right, we break down the individual contributions to the generalization bound, with the quantization error and loss variance being small and decreasing whereas the $\mathcal{C} \log V$ term and smoothing terms make up the majority and are not decreasing (if the complexity $\mathcal{C}$ is constant). Figure 2 left shows the comparison with the bound value R_{sq} and the empirical risk $\hat{R}_h$. The fact that the quantization gap at a fixed number of bits is decreasing model size has been observed to an even larger extent in other work (Chee et al., 2024; Tseng et al., 2024) with more advanced quantization methods. This property suggests that if b were able to be freely optimized in the bound, the complexity $\mathcal{C}$ would actually decrease with model size, and we explore evidence for and consequences of this idea in the following section.

5 Compressibility and the Sublinear Information Growth in LLMs

While not obvious from efficient quantization algorithms like GPTQ (Frantar et al., 2023), there are good reasons to believe that the model complexity term $L(h)/D$ decreases with model scale on the compute optimal frontier.

5.1 Quantizability from the Hessian Spectrum and QuIP

So far we have split the compressed size of the model $L(h)$ featured in the complexity term into the number of parameters N times the number of bits per parameter used in the quantization b: $L(h) \leq bN \log 2$. In this splitting increased compressibility of a model shows up in terms of requiring a smaller number of bits b to achieve a given quantization gap $\hat{R}_q - \hat{R}_h$. In Appendix B, we provide a theoretical argument using the QuIP quantization framework (Chee et al., 2024) for why we should expect that larger models can be more easily quantized. If the hessian around the solution weights is PSD and the spectrum decays sufficiently rapidly, then we should expect the quantization error to decrease with model size. In Section B.1, we investigate the Hessian spectrum empirically finding that it indeed decays sufficiently quickly (though not always PSD). Unfortunately, the version of QuIP needed to construct this argument cannot be run in practice due to the impractically large computational constraints. Empirically it has been observed by some that practical quantization algorithms also reveal that larger models are more quantizable (Tseng et al., 2024), though the effect is not very pronounced with the GPTQ algorithm we use here.

Alternatively, we present a more abstract information theoretic argument to provide evidence for the fact that $L(h)/D$ decreases with model scale even if we do not have an explicit compression scheme that makes it so.

5.2 Information Accumulation in LLMs

Despite the many parameters, at initialization the information content in a neural network (and thus the size a compression scheme can achieve) is extremely small. It suffices to specify the model architecture and a random seed used to initialize the weights, both of which can be specified ahead of time without seeing the data. As the loss decreases, information is transferred incrementally from the dataset to the weights of the model with each additional data point. This information can be quantified abstractly using algorithmic information theory and prequential coding (Rissanen, 1984; Dawid, 1984). Let X be the training dataset with D tokens and h the LLM with N parameters. Let $K(X)$ be the (prefix) Kolmogorov complexity of X, that is the length of a shortest self delimiting program that produces X when run, and we will consider the description of the LLM architecture as well as code for performing arithmetic encoding and decoding as part of the language in question. From

the symmetry of information property (the analog of Bayes rule), $K(X, h) = K(h) + K(X|h) + c$, where c is a small constant and $K(X|h)$ is the length of a shortest (self delimiting) program which takes as input h and produces X. As described in Blier & Ollivier (2018); Voita & Titov (2020) and more specifically in Zhang et al. (2020) for measuring the information transfer, rearranging $K(h) = K(X, h) - K(X|h) - c$ one can estimate an upper bound on the information stored in a given model using prequential coding.

In a prequential code (Rissanen, 1984; Dawid, 1984), we consider the hypothesis h_k obtained from training after seeing only $k-1$ of the data points in X. Starting with the initialization at h_0, one can use h_{k-1} to encode X_k using arithmetic coding (or any entropy stream code) with $-\log_2 p_{h_{k-1}}(X_k|X_{<k})$ bits and transmit this data. After transmitting X_k, the model is trained on X_k and the next data point is transmitted and so forth until the entire dataset has been encoded and transmitted. On the other end, this data allows reconstructing both the entire dataset and the model if the details of the training algorithm are known. One simply uses h_k to decode the given data point, train one step, and repeat so that the exact sequence of models $[h_0, h_1, h_2, \dots, h_D]$ is repeated. The length of the code for X, h is then only $-\sum_{k=1}^{D} \log_2 p_{h_{k-1}}(X_k|X_{<k})$, the area under the loss curve.

Though used for classification problems in Zhang et al. (2020), we can readily repurpose the approach for the autoregressive unsupervised learning task. For the coding of $X|h$, one can estimate this using the code length of the final model $-\sum_{k=1}^{D} \log_2 p_{h_D}(X_k|X_{<k})$. Assembling these two components together, one can estimate an upper bound on $K(h)$. Notably, one can convert a regular code of ℓ bits into a prefix free code of $L = \ell + 2\log_2 \ell + 1$ bits. Written in terms of the empirical risk and up to additive logarithmic factors $\tilde{O}(1)$,

$$K(h)\log(2) \leq \sum_{k=1}^{D} \left[R_{h_{k-1}}(X_k|X_{<k}) - R_h(X_k|X_{<k})\right] + \tilde{O}(1). \tag{5}$$

If we plug in the Chinchilla scaling law $R(N, D) = E + AN^{-\alpha} + BD^{-\beta}$ as an estimate for the risk along the training trajectory, (and noting that $\frac{1}{D}\sum_{k=1}^{D} f(k) \to \frac{1}{D}\int_1^D f(x)dx$ as $D \to \infty$) we would have

$$K(h)\log(2) \leq \left(\sum_{k=1}^{D} R(N, k)\right) - DR(N, D) + \tilde{O}(1) = \tilde{O}(D^{1-\beta}). \tag{6}$$

Looking at the right hand side $K(h) = \tilde{O}(D^{1-\beta})$ we have an insightful result for LLMs: the information content in the model grows sublinearly in the size of the training dataset, with a coefficient depending on the scaling law.

As shown in Figure 3 left, using the actual loss curves to evaluate $\sum_{k=1}^{D}\left[R_{h_{k-1}}(X_k|X_{<k}) - R_h(X_k|X_{<k})\right]$ and fitting the results to a power law we get a very good agreement with the predicted $\tilde{O}(D^{1-\beta})$. Notably the empirical fit yields $2 \times 10^4 \cdot N^{0.64} \propto D^{0.64}$, and from the constants estimated in Besiroglu et al. (2024), $\beta = 0.37$, which would yield $D^{1-\alpha} = D^{0.63}$. While the empirical values for the upper bound lie above the straightforward value one gets from quantization and parameter counting bN over the range of Pythia models, the curves predict a crossover point at $\approx$ 30B parameter models. This would mean that despite the large number of parameters, the information stored in each one decreases with scale.

5.3 Implications for Generalization Bounds

If we apply this observation to upper bound the complexity featured in our generalization bounds (Theorem 3.4) $\mathcal{C} = \frac{L(h)+\log|K|/\delta}{D} = K(h)\log(2)/D + \frac{\log|K|/\delta}{D} = \tilde{O}(D^{-\beta})$, we see that the complexity will actually decrease with the size of the dataset even as the ratio with parameters is held constant. With this scaling we can derive a version of the generalization bound Theorem 3.4 without needing to consider quantization or considering the number of explicit parameters in the model, provided that the Chinchilla scaling law holds.

We evaluate the non asymptotic generalization bound of Theorem 3.4 but using the complexity derived from empirical prequential coding bound in Figure 3 (right) and break it down into the scaling

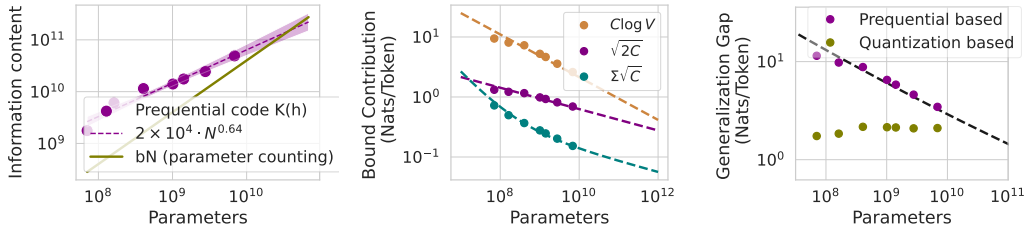


Figure 3: **Left:** Information content contained in the model as upper bounded by $K(h)$ from the information transfer prequential coding approach vs parameter counting and quantization. Fitting a power law to the prequential $K(h)$ yields $2 \times 10^4 \cdot N^{0.64}$. While parameter counting gives a better upper bound over the range of Pythia models, the sublinear scaling of the prequential bound means that it overtakes it eventually, somewhere around 30B sized models. **Center:** The contributions of the various terms to our generalization bounds when using prequential coding complexity, along with their power law fits. **Right:** Comparison of generalization bounds produced by the prequential vs quantization based approaches. While the prequential bounds are worse, they follow a power law and improve substantially with scale.

of the individual terms (center), with the Σ term scaling law extrapolated from the fit in Figure 2. Like before, the $\mathcal{C} \log V$ term dominates; however, the $\sqrt{2\mathcal{C}}$ smoothing term threatens to overtake it with very large model sizes. We can see that the bounds based on the prequential coding are worse than their quantization counterparts, however the bounds improve with scale and can be extrapolated with scaling laws.

At a high level just considering the asymptotics, the generalization gap of Theorem 3.4 will be dominated by the scaling of the smoothing term $\sqrt{2\mathcal{C}}$: $R_s - \hat{R}_h = \tilde{O}(D^{-\beta/2})$. To speculate, it seems likely that with a more sophisticated approach for dealing with the unbounded loss, the $\sqrt{C} = \tilde{O}(D^{-\beta/2})$ could be removed, letting the $\tilde{O}(D^{-\beta})$ shine through. If that were the case, then the tokenwise generalization gap could be hidden within the $D^{-\beta}$ of the original scaling law, and we leave such investigations to future work.

6 Additional Related Work

Generalization Bounds. Historically, generalization bounds for neural networks have been limited due to their large parameter count; however, significant progress has been made in explaining the generalization behavior of moderately large machine learning models over the years (Dziugaite & Roy, 2017; Zhou et al., 2018; Arora et al., 2018; Lotfi et al., 2022). PAC-Bayes has proven to be a convenient unifying framework for accommodating many of these techniques (Catoni, 2007). Lotfi et al. (2024a) demonstrate how to construct the first non-vacuous generalization bounds for LLMs by handling the unbounded objective with prediction smoothing and employing extreme compression with subspace LoRA (Hu et al., 2021) training.

While Lotfi et al. (2024a) focus on constructing generalization bounds at the level of documents, Lotfi et al. (2024b) take a different approach by applying Azuma's inequality to derive a martingale-based bound at the token level, leveraging the much larger number of data points. We adopt this approach here, and improve on the $O(\sqrt{\mathcal{C}})$ complexity on the dominating term from Azuma's ineqaulity to the $O(\mathcal{C})$ of the dominating term of our inequality through the use of the loss variation. Generalization bounds have also been used to constrain the context learning performance of LLMs (Li et al., 2023), and generalization over the hypothesis space of prompts in vision-language models has been explored (Akinwande et al., 2023).

Chugg et al. (2023) develop a broad family of generalization bounds applicable to both IID and martingale settings, generalizing many previous theoretical results. The martingale bounds presented in their work are more similar to ours than most other approaches; however, they do not provide fully empirical bounds suitable for our purposes. In terms of concentration inequalities, the closest work is

Maurer & Pontil (2009), where the authors derive a fully empirical Bernstein inequality. However, their result and proof technique do not extend to the non-IID martingale setting.

Post-Training Quantization. For hardware efficiency, there has been significant research into how low the precision of a model can be reduced post-training without substantially degrading its performance (Hassibi et al., 1993; Hubara et al., 2021; Yao et al., 2022; Dettmers et al., 2022). Empirically, the community has found that 3 or 4 bits provide a reasonable tradeoff between model compression and performance. Further research has also focused on applying quantization at the scale of LLMs, achieving a small number of bits per parameter, with Ma et al. (2024) pushing the limits to 1.58 bits per parameter. With some fine-tuning, even binary networks have shown promise (Liu et al., 2024).

We specifically highlight a few particularly relevant post-training quantization (PTQ) methods. Frantar et al. (2023) introduce GPTQ (formerly known as OPTQ), which demonstrates *extreme quantization* that scales to billions of parameters without significant degradation while being computationally efficient. This performance is achieved through iterative rounding of the weight columns. In this work, we use GPTQ with 4 bits to post-quantize the models. Chee et al. (2024) propose QuIP, which relies on the insight of *incoherence* in approximate Hessian estimation, ensuring that weights are not too large along a single direction, thereby suppressing outliers, computed successively at each layer. QuIP# further improves upon this incoherence processing (Tseng et al., 2024). In this work, we adopt the GPTQ quantizer, as it is easier to work with and to adapt to our setting.

7 Discussion

Here we have provided generalization theory to better explain why large language models trained in the compute-optimal regime generalize, with particular attention on how generalization changes with scale. For the term that contributes the most to the generalization bound, we are able to improve the scaling over alternate methods from $\sqrt{C}$ to C while still being fully empirical. We explore two approaches for constraining model complexity in the generalization bounds, directly via quantization and parameter counting, and indirectly, via information transfer as quantified by prequential coding. While the quantization approach yields more practical bounds, the information transfer approach yields insights into how generalization scales with model size without requiring an explicit quantization strategy that behaves that way.

While we believe that these insights help advance understanding, there are a number of limitations of our approach and many questions that remain unaddressed. As previously mentioned, the $\sqrt{2C}$ smoothing term seems pessimistic and could likely be improved with a different approach. Additionally, while the information transfer argument provides evidence *that* the complexity of a model is low based on the training curve, it falls short of explaining *why* the complexity is low. In principle the training curve could look different, leading to a different information transfer rate. Similarly with the Hessian based argument for why larger models are more quantizable as it depends on the empirical spectrum of the Hessian which we have not explained. Furthermore, why does the loss variation term Σ scale in the way that it does?

Even more broadly, our generalization bounds constrain only the token-wise generalization gap. While it is intuitive that generalizing well on next token prediction over the training contexts should imply generalization on the full sequences, we are not aware of work that does so and that gap remains to be understood. Similarly, constraining generalization on the NLL objective over data drawn from the natural distribution may be less pertinent. Instead, it may be more relevant to constrain the gap between the quality metrics of model generations and natural data. Farther removed, there is the question of why the training loss scales in the way that it does, and how does that relate to approximation theory and the architecture of the model? Though many questions remain, we hope that the techniques here can yield generalizable insights for tackling this broader set of problems.

References

Victor Akinwande, Yiding Jiang, Dylan Sam, and J Zico Kolter. Understanding prompt engineering may not require rethinking generalization. *arXiv preprint arXiv:2310.03957*, 2023.

Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. Stronger generalization bounds for deep nets via a compression approach. In *International Conference on Machine Learning*, pp. 254–263. PMLR, 2018.

Yasaman Bahri, Ethan Dyer, Jared Kaplan, Jaehoon Lee, and Utkarsh Sharma. Explaining neural scaling laws. *Proceedings of the National Academy of Sciences of the United States of America*, 121, 2021.

Tamay Besiroglu, Ege Erdil, Matthew Barnett, and Josh You. Chinchilla scaling: A replication attempt. *arXiv preprint arXiv:2404.10102*, 2024.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pp. 2397–2430. PMLR, 2023.

Léonard Blier and Yann Ollivier. The description length of deep learning models. *Advances in Neural Information Processing Systems*, 31, 2018.

Léon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. *Advances in neural information processing systems*, 20, 2007.

Gavin Brown and Riccardo Ali. Bias/variance is not the same as approximation/estimation. *Transactions on Machine Learning Research*, 2024.

Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

Joël Bun, Jean-Philippe Bouchaud, and Marc Potters. Cleaning large correlation matrices: tools from random matrix theory. *Physics Reports*, 666:1–109, 2017.

Olivier Catoni. Pac-bayesian supervised classification: the thermodynamics of statistical learning. *arXiv preprint arXiv:0712.0248*, 2007.

Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher M De Sa. Quip: 2-bit quantization of large language models with guarantees. *Advances in Neural Information Processing Systems*, 36, 2024.

Ben Chugg, Hongjian Wang, and Aaditya Ramdas. A unified recipe for deriving (time-uniform) pac-bayes bounds. *Journal of Machine Learning Research*, 24(372):1–61, 2023.

G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, 1989. doi: 10.1007/BF02551274. URL https://doi.org/10.1007/BF02551274.

A Philip Dawid. Present position and potential developments: Some personal views statistical theory the prequential approach. *Journal of the Royal Statistical Society: Series A (General)*, 147(2): 278–290, 1984.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Llm.int8(): 8-bit matrix multiplication for transformers at scale. *ArXiv*, abs/2208.07339, 2022.

Kun Dong, David Eriksson, Hannes Nickisch, David Bindel, and Andrew G Wilson. Scalable log determinants for gaussian process kernel learning. *Advances in Neural Information Processing Systems*, 30, 2017.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Cantón Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab A. AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip

Radenovic, Frank Zhang, Gabriele Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guanglong Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Laurens Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Ju-Qing Jia, Kalyan Vasuden Alwala, K. Upasani, Kate Plawiak, Keqian Li, Ken-591 neth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline C. Muzzi, Mahesh Babu Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melissa Hall Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri S. Chatterji, Olivier Duchenne, Onur cCelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasić, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Chandra Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yiqian Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zhengxu Yan, Zhengxing Chen, Zoe Papakipos, Aaditya K. Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adi Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Ben Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Shang-Wen Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzm'an, Frank J. Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory G. Sizov, Guangyi Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Han Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kaixing(Kai) Wu, U KamHou, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, A Lavender, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari,

Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollár, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sung-Bae Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Andrei Poenaru, Vlad T. Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xia Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. The llama 3 herd of models. *ArXiv*, abs/2407.21783, 2024.

Gintare Karolina Dziugaite and Daniel M Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *arXiv preprint arXiv:1703.11008*, 2017.

Jack Fitzsimons, Diego Granziol, Kurt Cutajar, Michael Osborne, Maurizio Filippone, and Stephen Roberts. Entropic trace estimates for log determinants. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18–22, 2017, Proceedings, Part I 10*, pp. 323–338. Springer, 2017.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers, 2023.

David A. Freedman. On tail probabilities for martingales. 1975.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Behrooz Ghorbani, Shankar Krishnan, and Ying Xiao. An investigation into neural net optimization via hessian eigenvalue density. In *International Conference on Machine Learning*, pp. 2232–2241. PMLR, 2019.

Diego Granziol, Xingchen Wan, Timur Garipov, Dmitry Vetrov, and Stephen Roberts. Mlrg deep curvature: An open-source package to analyse and visualise neural network curvature and loss surface. *stat*, 2018.

Diego Granziol, Timur Garipov, Stefan Zohren, Dmitry Vetrov, Stephen Roberts, and Andrew Gordon Wilson. The deep learning limit: are negative neural network eigenvalues just noise? In *ICML 2019 workshop on theoretical physics for deep learning*, 2019.

Diego Granziol, Stefan Zohren, and Stephen Roberts. Learning rates as a function of batch size: A random matrix theory approach to neural network training. *Journal of Machine Learning Research*, 23(173):1–65, 2022.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. Large language models are zero-shot time series forecasters. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 19622–19635. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/3eb7ca52e8207697361b2c0fb3926511-Paper-Conference.pdf.

Babak Hassibi, David G. Stork, and Gregory J. Wolff. Optimal brain surgeon and general network pruning. *IEEE International Conference on Neural Networks*, pp. 293–299 vol.1, 1993.

Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B Brown, Prafulla Dhariwal, Scott Gray, et al. Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701*, 2020.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and L. Sifre. Training compute-optimal large language models. *ArXiv*, abs/2203.15556, 2022.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Itay Hubara, Yury Nahshan, Yair Hanani, Ron Banner, and Daniel Soudry. Accurate post training quantization with small calibration sets. In *International Conference on Machine Learning*, 2021.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeff Wu, and Dario Amodei. Scaling laws for neural language models. *ArXiv*, abs/2001.08361, 2020.

Yingcong Li, Muhammed Emrullah Ildiz, Dimitris Papailiopoulos, and Samet Oymak. Transformers as algorithms: Generalization and stability in in-context learning. In *International Conference on Machine Learning*, pp. 19565–19594. PMLR, 2023.

James Liu, Guangxuan Xiao, Kai Li, Jason D. Lee, Song Han, Tri Dao, and Tianle Cai. Bitdelta: Your fine-tune may only be worth one bit, 2024.

Sanae Lotfi, Marc Finzi, Sanyam Kapoor, Andres Potapczynski, Micah Goldblum, and Andrew G Wilson. Pac-bayes compression bounds so tight that they can explain generalization. *Advances in Neural Information Processing Systems*, 35:31459–31473, 2022.

Sanae Lotfi, Marc Finzi, Yilun Kuang, Tim G. J. Rudner, Micah Goldblum, and Andrew Gordon Wilson. Non-vacuous generalization bounds for large language models, 2024a.

Sanae Lotfi, Yilun Kuang, Brandon Amos, Micah Goldblum, Marc Finzi, and Andrew Gordon Wilson. Unlocking tokens as data points for generalization bounds on larger language models. *ArXiv*, abs/2407.18158, 2024b.

Sanae Lotfi, Yilun Kuang, Brandon Amos, Micah Goldblum, Marc Finzi, and Andrew Gordon Wilson. Unlocking tokens as data points for generalization bounds on larger language models. *arXiv preprint arXiv:2407.18158*, 2024c.

Shuming Ma, Hongyu Wang, Lingxiao Ma, Lei Wang, Wenhui Wang, Shaohan Huang, Li Dong, Ruiping Wang, Jilong Xue, and Furu Wei. The era of 1-bit llms: All large language models are in 1.58 bits, 2024.

Andreas Maurer and Massimiliano Pontil. Empirical bernstein bounds and sample variance penalization. *arXiv preprint arXiv:0907.3740*, 2009.

Gérard Meurant and Zdeněk Strakoš. The lanczos and conjugate gradient algorithms in finite precision arithmetic. *Acta Numerica*, 15:471–542, 2006.

Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A Raffel. Scaling data-constrained language models. *Advances in Neural Information Processing Systems*, 36, 2024.

Markus Nagel, Rana Ali Amjad, Mart Van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? adaptive rounding for post-training quantization. In *International Conference on Machine Learning*, pp. 7197–7206. PMLR, 2020.

OpenAI. Gpt-4 technical report. 2023.

Vardan Papyan. Measurements of three-level hierarchical structure in the outliers in the spectrum of deepnet hessians. *arXiv preprint arXiv:1901.08244*, 2019.

Roma Patel and Ellie Pavlick. Mapping language models to grounded conceptual spaces. In *International Conference on Learning Representations*, 2022.

Barak A Pearlmutter. Fast exact multiplication by the hessian. *Neural computation*, 6(1):147–160, 1994.

Andres Potapczynski, Marc Finzi, Geoff Pleiss, and Andrew Gordon Wilson. CoLA: Exploiting Compositional Structure for Automatic and Efficient Numerical Linear Algebra. *arXiv preprint arXiv:2309.03060*, 2023.

Jorma Rissanen. Universal coding, information, prediction, and estimation. *IEEE Transactions on Information theory*, 30(4):629–636, 1984.

Farbod Roosta-Khorasani and Uri Ascher. Improved bounds on sample size for implicit matrix trace estimators. *Foundations of Computational Mathematics*, 15(5):1187–1212, 2015.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Trieu H. Trinh, Yuhuai Wu, Quoc V. Le, He He, and Thang Luong. Solving olympiad geometry without human demonstrations. *Nature*, 625:476 – 482, 2024.

Albert Tseng, Jerry Chee, Qingyao Sun, Volodymyr Kuleshov, and Christopher De Sa. Quip: Even better llm quantization with hadamard incoherence and lattice codebooks, 2024.

Shashanka Ubaru, Jie Chen, and Yousef Saad. Fast estimation of tr(f(a)) via stochastic lanczos quadrature. *SIAM Journal on Matrix Analysis and Applications*, 38(4):1075–1099, 2017.

Jean Ville. *Étude critique de la notion de collectif*. 1939. URL http://eudml.org/doc/192893.

Elena Voita and Ivan Titov. Information-theoretic probing with minimum description length. *arXiv preprint arXiv:2003.12298*, 2020.

Zhewei Yao, Reza Yazdani Aminabadi, Minjia Zhang, Xiaoxia Wu, Conglong Li, and Yuxiong He. Zeroquant: Efficient and affordable post-training quantization for large-scale transformers. *ArXiv*, abs/2206.01861, 2022.

Xiao Zhang, Xingjian Li, Dejing Dou, and Ji Wu. Measuring information transfer in neural networks. *arXiv preprint arXiv:2009.07624*, 2020.

Wenda Zhou, Victor Veitch, Morgane Austern, Ryan P Adams, and Peter Orbanz. Non-vacuous generalization bounds at the imagenet scale: a pac-bayesian compression approach. *arXiv preprint arXiv:1804.05862*, 2018.

A PROOFS

A.1 A FULLY EMPIRICAL MARTINGALE FREEDMAN CONCENTRATION INEQUALITY.

In this section, unless otherwise specified, $\log$ is used to denote the natural logarithm. We start with three technical lemmas.

Lemma A.1. *Consider the function $v(a) = a - \log(1+a)$. Let $\mu \in \mathbb{R}$. For any random variable Z with $\mathbb{E}[Z] = 0$ that satisfies $Z - \mu > -1$, we have*

$$\mathbb{E}[\exp(Z - v(Z-\mu))] = (1-\mu)e^{\mu} \leq 1.$$

Proof. We see that

$$\mathbb{E}[\exp(Z - v(Z-\mu))] = e^{\mu}\mathbb{E}[\exp(Z - \mu - v(Z-\mu))] = e^{\mu}\mathbb{E}[1 + (Z-\mu)].$$

As $\mathbb{E}[Z] = 0$, we know $\mathbb{E}[\exp(Z - v(Z-\mu))] = (1-\mu)e^{\mu}$. Additionally as $1 - \mu \leq e^{-\mu}$, we have $\mathbb{E}[\exp(Z - v(Z-\mu))] \leq 1$, as desired. □

Lemma A.2. *Consider the function $v(a) = a - \log(1+a)$. For all $a \in (-1, \infty)$, we have*

$$v(a) \leq a^2/(1+a)$$

Proof. Let $f(a) = a^2/(1+a) - v(a)$. By direct calculation, we see that $f'(a) = a/(1+a^2)$, which is a strictly negative function passing through 0 at $a = 0$. As $f(0) = 0$, we have $f(a) \geq 0$ for all $a \geq 0$. Note that $\lim_{a \to -1^-} f(a) = +\infty$, so $f(a)$ must be positive on $(-1, 0)$. The claim follows. □

For the following lemma, consider on the filtered probability space $(\Omega, \mathcal{F}, \{\mathcal{F}_k\}_{k\in\mathbb{N}}, \mathbb{P})$, and we consider expectations with respect to $\mathbb{P}$.

Theorem A.3. *Let $X_1, \ldots, X_n$ be a sequence of $\mathcal{F}_k$-measurable random variables. Let $Y_1, \ldots Y_n$ be any other sequence of $\mathcal{F}_{k-1}$-measurable random variables such that the difference is bounded above: $A_k = Y_k - X_k > -\Delta$ for some $\Delta \geq 0$. Define $C = \frac{1}{n}\log\frac{1}{\epsilon}$,and $B = \frac{1}{n}\sum_k \big(\mathbb{E}[X_k \mid \mathcal{F}_{k-1}] - X_k\big)$. For any $0 < t < 1/\Delta$ and simultaneously for all n, we have*

$$\mathbb{P}\big[tB \leq C + \frac{1}{n}\sum_{k=1}^{n} v(tA_k)\big] \geq 1 - \epsilon. \tag{7}$$

Proof. Let $v(a) = a - \log(1+a)$. Define the random variable

$$M_k = \exp\left(t(\mathbb{E}[X_k \mid \mathcal{F}_{k-1}] - X_k) - v(t(Y_k - X_k))\right).$$

Consider $Z = t(\mathbb{E}[X_k \mid \mathcal{F}_{k-1}] - X_k)$ and $\mu = t(\mathbb{E}[X_k \mid \mathcal{F}_{k-1}] - Y_k)$. By construction, we have that $\mathbb{E}[Z \mid \mathcal{F}_{k-1}] = 0$ and $Z - \mu = t(Y_k - X_k) > -t\Delta > -1$. Thus, applying Lemma A.1, we have

$$\mathbb{E}[M_k \mid \mathcal{F}_{k-1}] \leq 1.$$

Therefore $U_n = \prod_{k=1}^{n} M_k$ is a supermartingale. By Ville's inequality (Ville, 1939), we have

$$\sup_n U_n \leq \frac{\mathbb{E}[U_0]}{\epsilon} \leq 1/\epsilon$$

with probability at least $1 - \epsilon$. Using the definition of U and taking the log of both sides, we have for all n,

$$t\sum_{k=1}^{n}\big(\mathbb{E}[X_k \mid \mathcal{F}_{k-1}] - X_k\big) - \sum_{k=1}^{n} v(t(Y_k - X_k)) \leq \log\frac{1}{\epsilon}. \tag{8}$$

Defining $B = \frac{1}{n}\sum_k \big(\mathbb{E}[X_k \mid \mathcal{F}_{k-1}] - X_k\big)$, $C = \frac{1}{n}\log\frac{1}{\epsilon}$, $A_k = Y_k - X_k$, rearrange Equation 8 to obtain

$$tB \leq C + \frac{1}{n}\sum_{k=1}^{n} v(tA_k). \tag{9}$$

□

Corollary A.4. *Let $X_1, \dots, X_n$ be a sequence of $\mathcal{F}_k$-measurable random variables. Let $Y_1, \dots Y_n$ be any other sequence of $\mathcal{F}_{k-1}$-measurable random variables such that the difference is bounded below: $A_k = (Y_k - X_k)/\Delta > -1$ for some $\Delta \geq 0$. Let K be a finite subset of $(0, 1)$. Then, with probability at least $1 - \delta$,*

$$\frac{1}{n}\sum_{k=1}^{n}\big(\mathbb{E}[X_k \mid \mathcal{F}_{k-1}] - X_k\big) \leq \Delta C + \Sigma\sqrt{C}, \tag{10}$$

where $C := \frac{1}{n}\log|K|/\delta$, and

$$\Sigma(C, \Delta, \{X_k - Y_k\}_{k=1}^n, K) := \min_{s\in K} \Delta\sqrt{C}(1-s)/s + \frac{\Delta}{\sqrt{C}}\frac{1}{n}\sum_{k=1}^{n} v\big(sA_k\big)/s$$

Proof. Let $s = t\Delta$. Apply a union bound to Theorem A.3 over the different values of s in K, taking the one that minimizes the bound. Rearrange and isolate terms. □

Theorem A.5. *Let $X_1, \dots, X_n$ be a sequence of $\mathcal{F}_{i-1}$-measurable random variables. Let $Y_0, \dots Y_{n-1}$ be any other sequence of $\mathcal{F}_{i-1}$ measurable sequence of random variables such that the difference is bounded above: $X_k - Y_k \leq \Delta$ for some $\Delta \geq 0$. Define $V = \frac{1}{n}\sum_k (X_k - Y_k)^2$ and let $\delta \in (0, 1)$. Then, with probability at least $1 - \delta$, we have*

$$\frac{1}{n}\sum_{k}\big(\mathbb{E}[X_k \mid \mathcal{F}_{k-1}] - X_k\big) \leq \Delta\mathcal{C} + 2\sqrt{V\mathcal{C}}, \tag{11}$$

where $\mathcal{C} \leq n^{-1}\big(\log 1/\delta + 4\log\log n/\delta + 6\big)$.

Proof. Starting from Theorem A.3, we apply Theorem A.2 of $v(a) \leq a^2(1+a)$. For our convenience, here we will define $A_k = Y_k - X_k$.

$$\begin{aligned} tB &\leq C + \frac{1}{n}\sum_{k}\frac{t^2A_k^2}{1 + tA_k} \\ &\leq C + \frac{1}{n}\sum_{k}\frac{t^2A_k^2}{1 - t\Delta}, \end{aligned} \tag{12}$$

where the second inequality follows from the assumption that $A_k \geq -\Delta$.

Finally, by defining a variance term $V = \frac{1}{n}\sum_k A_k^2 = \frac{1}{n}\sum_k (X_k - Y_k)^2$ and rearranging Equation 12, we see

$$0 \leq t^2(V + B\Delta) - (B + \Delta C)t + C, \tag{13}$$

which we recall holds with probability at least $1 - \epsilon$.

Inequality Sketch: This inequality is very close to what we need. The approach would be to optimize over t and then read off the constraint on B. The minimizer of the quadratic is at $t^* = \frac{B+\Delta C}{2(V+\Delta B)}$. Plugging in this value one would arrive at,

$$(1/4)\frac{(B+\Delta C)^2}{(V+\Delta B)} - (1/2)\frac{(B+\Delta C)^2}{(V+\Delta B)} + C \geq 0$$

Rearranging,

$$\frac{1}{4}(B + \Delta C)^2 - B\Delta C \leq VC$$

$$\frac{1}{4}(B - \Delta C)^2 \leq VC$$

and finally,

$$B \leq \Delta C + 2\sqrt{VC}.$$

At a high level, this determines the overall form of Theorem A.5, however some technical complications arise from the fact that t must be deterministic and chosen ahead of time (it must not depend on the random variables B and C). Instead we will consider optimizing t over a discrete set of

possibilities (not depending on B or C), and consider a union bound over the different possibilities.
Full derivation: Consider the quadratic, Equation 13. Its minimizer is given by

$$t^* = \frac{B + \Delta C}{2(V + \Delta B)}.$$

Consider two cases: $t^* \geq \frac{1}{\Delta}$ and $t^* < \frac{1}{\Delta}$. If $t^* \geq \frac{1}{\Delta}$, then rearranging and solving for B, we see

$$B \leq \Delta C - 2V/\Delta,$$

which is strictly less than the value $\Delta\mathcal{C} + 2\sqrt{V\mathcal{C}}$, and we are done.

Therefore it suffices to consider the case $t^* < 1/\Delta$, where we can apply Lemma A.3. Note that this result applies for a single t, so it cannot be directly applied to t^*. Instead, we will turn to quantization and apply a union bound. Note that if $B > 0$, using that $V \leq \Delta^2$, we have $t^* \geq \frac{\Delta C}{2\Delta^2} = \frac{C}{2\Delta}$. Therefore we only need to consider the range: $t^* \in (\frac{C}{2\Delta}, \frac{1}{\Delta}) =: T$.

Drawing inspiration from floating point numbers, consider a discrete set Q defined as

$$Q = \left\{ \tfrac{1}{\Delta} 2^{-b} \left(1 + \frac{k}{K}\right) \;\middle|\; k = 0, 1, ..., K-1, b \in \mathbb{N}^+ \right\}$$

for some $K \in \mathbb{N}$. Let

$$q(a) = \arg\min_{q \in Q} |q - a|.$$

From this we can determine that the quantization error is bounded by

$$\sup_{a \in T} \frac{|q(a) - a|}{a} \leq \frac{1}{K}.$$

Define a prior over the values of Q:

$$P(q_{k,b}) = P(k)P(b) = \frac{1}{K} \frac{1}{Z(b+2)(\log_2(b+2))^2}.$$

By direct calculation, we see that $1 = \sum_{k,b} P(k)P(b) = \left(\sum_{b=0}^{\infty} \frac{1}{(b+2)(\log_2(b+2))^2}\right)/Z \leq 1/Z$, therefore $Z \leq 1$.

Now we apply a union bound for Lemma A.3 over values of $t \in Q$. For each $t \in Q$, we set $\epsilon(t) = \delta P(t)$. We have

$$\begin{aligned}
&\mathbb{P}\left[\forall t \in Q : t^2(V + B\Delta) - (B + \Delta C_{\epsilon(t)})t + C_{\epsilon(t)} < 0\right] \\
&\quad \leq \sum_{t \in Q} \mathbb{P}\left[t^2(V + B\Delta) - (B + \Delta C_{\epsilon(t)})t + C_{\epsilon(t)} < 0\right] \\
&\quad \leq \sum_{t \in Q} \epsilon(t) = \delta \sum_{t \in Q} P(t) = \delta.
\end{aligned}$$

Therefore, uniformly for all $t \in Q$,

$$t^2(V + B\Delta) - (B + \Delta C_{\epsilon(t)})t + C_{\epsilon(t)} \geq 0 \tag{14}$$

with probability at least $1 - \delta$.

Now we plug in $t = q(t^*)$. Note that $0 \leq b \leq \log_2 \frac{2}{C}$, so we have

$$\begin{aligned}
\log \frac{1}{\epsilon(t)} = \log \frac{1}{\delta P(t)} &\leq \log \frac{K}{\delta} + \log(3 + \log_2 1/C) + 2 \log\log_2(3 + \log_2 1/C) \\
&\leq \log \frac{K}{\delta} + 2 + 2 \log\log 1/C \leq \log \frac{K}{\delta} + 2 + 2 \log\log n
\end{aligned} \tag{15}$$

Plugging in this quantized value of t to Equation 14 and using the quantized error bound, Equation 15, we have

$$\begin{aligned}
\frac{(B + \Delta C)^2}{4(V + \Delta B)} &\leq C + \frac{3}{K}\left(1 + \frac{1}{K}\right)\frac{1}{4}\frac{(B + \Delta C)^2}{(V + \Delta B)} \\
&\leq C(1 + 4/K),
\end{aligned} \tag{16}$$

where in the second line we chose a $K \geq 6$ so that we have $\left(1 - \frac{3}{K}\left(1 + \frac{1}{K}\right)\right)^{-1} \leq 1 + \frac{4}{K}$. Solving Equation 16 for B, we have the inequality

$$\begin{aligned} B &\leq \Delta C(1 + 8/K) + \sqrt{\Delta^2 C^2 (8/K)^2 + 4CV(1 + 4/K)} \\ &\leq \Delta C(1 + 16/K) + 2\sqrt{VC(1 + 16/K)}, \end{aligned} \tag{17}$$

where the second line follows from the fact that $\sqrt{x + y} \leq \sqrt{x} + \sqrt{y}$. Define $\mathcal{C} = C(1 + 16/K)$. Choosing $K = \lceil 16 \log 1/\delta \rceil$ (which is > 6), we have

$$n\mathcal{C} \leq \log 1/\delta + 1 + \left[\log(\lceil 16 \log 1/\delta \rceil) + 2 + 2 \log\log n\right]\left(1 + 1/\log(1/\delta)\right) \tag{18}$$

Applying some simplifications to Equation 17 and Equation 18, we obtain

$$B \leq \Delta\mathcal{C} + 2\sqrt{V\mathcal{C}}.$$

with

$$\mathcal{C} \leq \frac{1}{n}\left(\log 1/\delta + 4 \log\log n/\delta + 6\right).$$

□

A.2 Generalization Bound

Converting to Prefix Free Codes Let $\ell(h)$ be the length of a prefix free code and $L(h)$ be the length of a non prefix free code. We wish to make a normalized prior over the hypotheses. We would like to use approximately $2^{-L(h)}$ where $L(h)$ is the codelength of hypothesis h, but this sum would diverge. Instead consider $\ell(L) = L + 2\log_2(L) + 1$. Computing the sum, $\sum_h 2^{-\ell(h)} = \sum_{L=1}^{\infty} 2^{L - \ell(L)} = \sum_{L=1}^{\infty} \frac{1}{2L^2} = \pi^2/12 < 1$. Thus with the prior $P(h) = \frac{12}{\pi^2} 2^{-\ell(h)}$ can be used for any countable hypothesis class, placing higher mass on elements with shorter descriptions, which is closely related to Kolmogorov complexity as explored in Lotfi et al. (2022).

If we know the length of the object ahead of time, then we are free to use a regular code in place of a prefix free code. For a fixed number of parameters N and bits per parameter b, we know the length of the code, and thus we can use $\ell(h) \leq bN$.

Applying Theorem 3.1 to the sequence $R_h(X_k \mid X_{<k})$ with $\delta(h) = \epsilon P(h)$ for each hypothesis individually, the probability that the bound is violated for an arbitrary hypothesis constrained with a union bound $\sum_h \epsilon P(h) = \epsilon$, and therefore Theorem 3.2 holds with probability at least $1 - \epsilon$ (replacing δ with ϵ in its expression). The $\log 1/\delta$ in Theorem A.5 becomes $\log 1/\delta + \log 1/P(h) \leq \log 1/\delta + \ell(h) \log 2$.

A.3 Prediction Smoothing

Theorem A.6. *For the categorical negative log likelihood objective $\hat{R}_h = -\frac{1}{n}\sum_{k=1}^{n} \log p_h(X_k \mid X_{<k})$ on V classes and $C \in \mathbb{R}^+$, there exists a prediction smoothed model $p_s(\cdot) = (1 - \alpha)p_h(\cdot) + \alpha/V$ with worst case loss $\sup_{X_k, X_{<k}} -\log p_s(X_k \mid X_{<k}) \leq \Delta_s$ that satisfies*

$$\hat{R}_s + C\Delta_s \leq \hat{R}_h + C \log V + \sqrt{2C}, \tag{19}$$

for some value $\alpha(C, V) \in (0, 1)$ (approximately $C/(1 + C)$).

Proof. We have

$$\begin{aligned} -\log p_s &= -\log\left((1 - \alpha)p_h + \alpha/V\right) \\ &\leq -\log p_h - \log\left(1 - \alpha + \alpha/V\right). \end{aligned}$$

Noting that, $-\log p_s(X_k | X_{<k}) \leq \Delta_s = \log(V/\alpha)$, so adding $C\Delta$ to both sides yields

$$\hat{R}_s + C\Delta_s \leq \hat{R}_h - \log(1 - \alpha + \alpha/V) + C \log(V/\alpha), \tag{20}$$

where the right-hand side is minimized at

$$\alpha = \frac{VC}{(V - 1)(1 + C)}.$$

Note that α is a *deterministic quantity* that we can compute ahead of time based on the model we are bounding. Therefore we need not pay additional bits for a union bound over values of α. Substituting α into Equation 20, we have

$$\begin{aligned}\hat{R}_s - \hat{R}_h + C\Delta_s &\leq \log(1+C) + C\log\frac{(V-1)(1+C)}{C} \\ &\leq (1+C)\log(1+C) + C\log(V/C) \\ &\leq C\log V + \sqrt{2C},\end{aligned}$$

where the last line follows from $(1+x)\log(1+x) - x\log x \leq \sqrt{2x}$ for $x > 0$. The claim follows. □

B Quantizability from the Hessian

We have shown that the quantization gap tends to be quite small in practice, but why is this the case? A more complete explanation of why LLMs generalize would need to explain why they are readily quantizable, not just why they should achieve a small generalization gap if they are quantizable. In this section we attempt to shed light on this question on the existence of a quantized solution for a model which achieves low quantization error at a small bitrate, regardless of how computationally efficient it is to actually produce said quantized model .

As a starting point in the analysis of many quantization schemes (Nagel et al., 2020), consider the Lagrange remainder form of the quadratic Taylor expansion of the loss around a given solution of the weights θ, with $\hat{\theta}$ being our desired quantization,

$$L(\hat{\theta}) = L(\theta) + g^\top(\hat{\theta} - \theta) + (\hat{\theta} - \theta)^\top H(\hat{\theta} - \theta)$$

holding with equality for g evaluated at θ and the Hessian H evaluated at an unknown but fixed point ξ on the linear path between θ and $\hat{\theta}$. If we use a stochastic rounding algorithm that is unbiased, then the first order term can be neglected as

$$\mathbb{E}[g^\top(\hat{\theta} - \theta)] = 0,$$

and a high dimensional vector θ ensures the sum will concentrate around the expectation. This leaves the quadratic form with the Hessian. .

A key property for low precision quantization of the weights (while minimizing the quadratic quantization error) is that the scale of the components of the eigenvectors of H do not differ by a large extent. If they do, then the quantization range must simultaneously provide coverage over a large range of values. This criterion is formalized through the notion of incoherence, introduced in Chee et al. (2024), which we briefly present below with a simplification of their analysis.

B.0.1 Incoherence

A Hessian is μ-incoherent if the eigenvectors in the decomposition $Q\Lambda Q^\top = H \in \mathbb{R}^{N\times N}$ satisfy

$$\forall i, j : |Q_{ij}| \leq \mu/\sqrt{N},$$

and a parameter vector is μ-incoherent if it satisfies $\forall j : |\theta_j| \leq \mu\|\theta\|/\sqrt{N}$. Following QuIP (Chee et al., 2024), rather than quantizing the weights θ directly we will instead quantize the weights after applying a random orthogonal transformation matrix $P \in \mathbb{R}^{N\times N}$. Let $w = P^\top\theta$ and likewise $\theta = Pw$. Applying this rotation, the Hessian is transformed: $H_w = P^\top H_\theta P$ and likewise the eigenvectors Q from $H_\theta = Q\Lambda Q^\top$ are also multiplied $Q^\top \mapsto Q^\top P$.

If we choose P as a random Gaussian matrix: $\mathcal{N}(0, 1/N)^{N\times N}$, applying a rotation by $Q^\top$ preserves the spherically symmetric distribution. Therefore the eigenvectors $Q^\top P$ of H_w are $\mathcal{N}(0, 1/N)^{N\times N}$ distributed. Applying a union bound over the Gaussian tail probability of the N^2 elements, the maximum absolute value entry of Q is at most $\sqrt{\frac{2\log(2N^2/\delta)}{N}}$ with probability $1-\delta$ and therefore incoherent with $\mu = \sqrt{2\log(2N^2/\delta)}$.

B.0.2 SCALAR LDLQ

QuIP introduces the LDLQ quantization algorithm which quantizes weights sequentially and autoregressively taking into account how previous quantized values impact the quadratic Taylor expansion of the loss. Applying LDLQ to the entire vector of weights w rather than block by block, one has the following relation on the quantized weights $\hat{w}$. Let $L^\top D L = H_w$ be the LDL decomposition of $H_w = P^\top H_\theta P$, then we can express the quantization of the weights as

$$\hat{w} = \mathcal{Q}(w + (L - I)(w - \hat{w}))$$

where $\mathcal{Q}$ quantizes the weights elementwise with nearest or unbiased stochastic rounding. As $L - I$ is a lower triangular matrix, the full $\hat{w}$ can be quantized sequentially in an autoregressive manner. With this quantization scheme, Tseng et al. (2024) prove that the error of the quadratic in the Taylor expansion satisfies

$$(\hat{w} - w)^\top H (w - \hat{w}) \leq \frac{\mu^2 \sigma^2}{n} \mathrm{Tr}(H^{1/2})^2, \tag{21}$$

where the pointwise quantization error of the scalar quantizer is assumed to be $\mathbb{E}[\big(\mathcal{Q}(x) - x\big)^2] \leq \sigma^2$ (see Theorem 4.1 applied to block size 1 and a single $N \times 1$ weight matrix), where σ^2 is a function of the bitrate. For example $x \in [0, 1]$, then a uniform grid would achieve $\sigma^2 = 2^{-2b-2}$ for b bits per parameter. From incoherence processing μ grows only logarithmically with the dimension N.

The achievable bits per parameter for a fixed quantization error depends crucially on the spectrum of the Hessian H, and how it scales with the dimension . We now pursue Krylov subspace routines with Hessian vector products to evaluate these quantities numerically for LLMs.

B.1 ESTIMATING $\mathrm{Tr}(H^{1/2})$

To estimate the trace of the square root of the Hessian matrix, $\mathrm{Tr}(H^{1/2})$, we begin by assuming that the Hessian is positive semi-definite (i.e., it contains no negative eigenvalues). The square root of the Hessian, denoted as S, can be expressed as:

$$S = \sum_{i=1}^{P} \sqrt{\lambda_i} \phi_i \phi_i^T,$$

where λ_i and ϕ_i represent the eigenvalues and corresponding eigenvectors of the Hessian, respectively. Consequently, the trace of the square root of the Hessian is:

$$\mathrm{Tr}(H^{1/2}) = \sum_{i=1}^{P} \sqrt{\lambda_i} = n \int_0^\infty p(\lambda) \sqrt{\lambda}\, d\lambda,$$

where $p(\lambda)$ is the spectral density function associated with the Hessian's eigenvalues.

A direct computation of the full eigendecomposition to obtain $\mathrm{Tr}(H^{1/2})$ has a computational complexity of $\mathcal{O}(n^3)$, which is infeasible for large models. Instead, we employ stochastic spectral density estimation techniques (Granziol et al., 2018; Papyan, 2019; Ghorbani et al., 2019), which scale linearly with the number of parameters. The key idea involves using the Pearlmutter trick (Pearlmutter, 1994) to efficiently compute Hessian-vector products:

$$\nabla(\nabla L^T v) = Hv,$$

where v is a random vector. This allows us to approximate the trace by leveraging the identity:

$$\mathrm{Tr}(H) = \mathbb{E}[\mathrm{Tr}(vv^T H)] = \mathbb{E}[v^T H v],$$

assuming v has zero mean and unit variance. These stochastic methods are well-established in machine learning (Fitzsimons et al., 2017; Dong et al., 2017).

Building upon the work of Ubaru et al. (2017), we can derive an explicit bound on the estimation of $\mathrm{Tr}(H^{1/2})$ using stochastic Lanczos quadrature (SLQ).

Theorem B.1. *Let $\boldsymbol{H} \in \mathbb{R}^{n\times n}$ be a symmetric positive definite matrix with eigenvalues ordered as $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ and condition number $\kappa = \frac{\lambda_1}{\lambda_n}$. For any $\epsilon, \eta \in (0,1)$, if the SLQ parameters satisfy*

$$m \geq \frac{\sqrt{\kappa}}{4} \log \frac{K}{\epsilon} \quad \textit{(Lanczos steps)},$$

$$n_v \geq \frac{24}{\epsilon^2} \log \frac{2}{\eta} \quad \textit{(Rademacher vectors)},$$

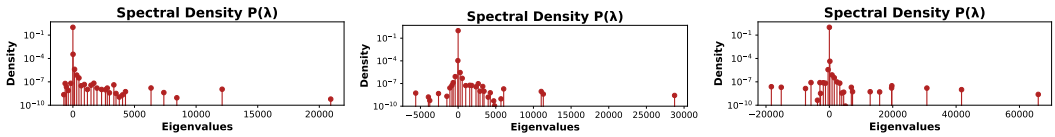
where $K = (\lambda_{\max} - \lambda_{\min})(\sqrt{\kappa} - 1)^2$, then the output Γ of stochastic Lanczos quadrature satisfies:

$$\Pr\left[\left|\frac{\operatorname{Tr}(\sqrt{\boldsymbol{H}}) - \Gamma}{\operatorname{Tr}(\sqrt{\boldsymbol{H}})}\right| \leq \epsilon\right] \geq 1 - \eta.$$

The proof of this theorem is provided in Appendix D. However, we observe that the bound on the trace provided here is overly conservative for practical purposes. Therefore, we also establish a result demonstrating self-averaging, which shows that the estimator converges to the true value based on a single random vector.

Theorem B.2. *For a single random vector v, the signal-to-noise ratio of the trace estimator for a matrix $\boldsymbol{H} \in \mathbb{R}^{n\times n}$, where the spectral moments of $\boldsymbol{H}$ do not depend on the matrix dimension, scales as:*

$$\frac{\sqrt{\operatorname{Var}(v^T \boldsymbol{H} v)}}{\mathbb{E}(v^T \boldsymbol{H} v)} = \mathcal{O}(n^{-\frac{1}{2}}).$$



(a) Dataset Fraction 0.01 (b) Dataset Fraction 0.001 (c) Dataset Fraction 0.0001

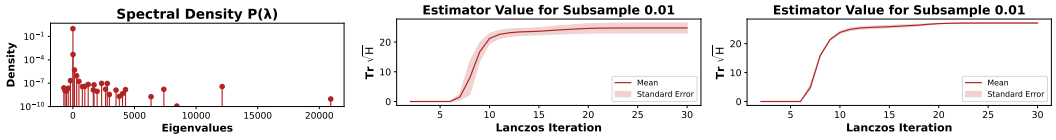
Figure 4: Spectral density plots of the $70M$ parameter Pythia model trained on varying fractions of the Pile dataset using the same data and random vector seed.

We utilize the CoLA (Potapczynski et al., 2023) library to compute the spectral approximation of the Hessian. This involves leveraging the relationship between the Lanczos $\boldsymbol{T}$ matrix and Gaussian quadrature (Meurant & Strakoš, 2006; Granziol et al., 2019). However, these concepts are highly specialized and may not be familiar to all readers. Therefore, we provide a high-level overview without delving into the intricate mathematical details.

Figure 4 illustrates the spectral density of a $70M$ parameter Pythia model trained on different subsets of the Pile dataset (Gao et al., 2020). Specifically, as we decrease the number of training samples—from 1% (Figure 4a) to 0.1% (Figure 4b) and further to 0.01% (Figure 4c)—we observe an increase in the largest eigenvalue and an increase in the mass of negative spectral density. These phenomena are consistent with previous studies on ResNets and VGGs, where spiked Wigner random matrix theory models have been employed to understand such behaviors (Granziol et al., 2022).

Future work aimed at establishing a tighter empirical bound could explore advanced random matrix theory techniques (Bun et al., 2017), potentially utilizing the variance of the Hessian (Granziol et al., 2022). In this study, we adopt a simpler approach by shifting the Hessian spectrum by the magnitude of the largest negative eigenvalue, thereby ensuring a positive semi-definite Hessian and providing a trivial upper bound.

From Figure 4b, we observe that the variance of each estimator remains low and that convergence is achieved with relatively few Lanczos iterations. Additionally, Figure 5 demonstrates that varying the random vector introduces minimal variance, while different data subsets do exhibit some variance, as indicated by the error bars computed over three different seeds (see Figure 5b). For clarity, Figure 5a provides another example of the spectrum with a different seed vector on the same subsampled dataset, showing negligible differences compared to Figure 4a.



(a) with Different Vector (b) Convergence with Iterations (c) with Fixed Data Subset

Figure 5: Comparison of spectral density and $\mathrm{Tr}(\sqrt{\boldsymbol{H}})$ estimations for different subsample sizes and configurations.

With confidence in the accuracy of our estimations for $\mathrm{Tr}(H^{1/2})$ and the Hessian spectrum, we can interpret the implications for model quantization. Despite the high dimensionality and the presence of many distinct eigenvalues, the Hessian spectrum decays rapidly in density. This indicates that $\mathrm{Tr}(H^{1/2})$ grows sublinearly with the model dimension, rather than exhibiting the worst-case linear scaling. Consequently, as model size increases, the ratio $L(h)/D$ is expected to decrease, allowing for a more favorable tradeoff between the bitrate and the quantization gap. This supports the hypothesis that larger models on the compute-optimal frontier are more easily quantizable, thereby contributing to their improved generalization performance.

C Stochastic Trace Estimation Improvement with Model Size

Here, we provide the proof that for a spectrum independent of model dimension, the stochastic trace estimator has a bigger signal to noise ratio as a function of dimension.

Lemma C.1. *Let $\boldsymbol{u} \in \mathbb{R}^{P\times 1}$ random vector, where $\boldsymbol{u}_i$ is zero mean and unit variance and finite 4th moment $\mathbb{E}[\boldsymbol{u}_i^4] = m_4$. Then for $\boldsymbol{H} \in \mathbb{R}^{P\times P}$, then*

(i) $\mathbb{E}[\boldsymbol{u}^T\boldsymbol{H}\boldsymbol{u}] = \mathrm{Tr}\,\boldsymbol{H}$,

(ii) $\mathrm{Var}[\boldsymbol{u}^T\boldsymbol{H}\boldsymbol{u}] \leq (2+m_4)\,\mathrm{Tr}(\boldsymbol{H}^T\boldsymbol{H})$.

Proof. For the expectation, we see

$$\mathbb{E}[\boldsymbol{u}^T\boldsymbol{H}\boldsymbol{u}] = \sum_{i,j=1}^{P} \boldsymbol{H}_{i,j}\mathbb{E}[\boldsymbol{u}_i\boldsymbol{v}_j] = \sum_{i=1}^{P} \boldsymbol{H}_{i,i} = \mathrm{Tr}\,\boldsymbol{H}.$$

For the variance, we have

$$\begin{aligned}\mathbb{E}[||\boldsymbol{u}^T\boldsymbol{H}\boldsymbol{u}||^2] &= \sum_{i,j}\sum_{k,l} \boldsymbol{H}_{i,j}\boldsymbol{H}^T_{k,l}\mathbb{E}[\boldsymbol{u}_i\boldsymbol{u}_j^T\boldsymbol{u}_k\boldsymbol{u}_l^T] \\ &= \sum_{i,j}\sum_{k,l} \boldsymbol{H}_{i,j}\boldsymbol{H}^T_{k,l}[\delta_{i,j}\delta_{k,l} + \delta_{i,l}\delta_{j,k} + \delta_{i,k}\delta_{j,l} + m_4\delta_{i,j,k,l}] \\ &= (\mathrm{Tr}\,\boldsymbol{H})^2 + (2+m_4)\,\mathrm{Tr}(\boldsymbol{H}^2),\end{aligned}$$

whence (ii) follows.

Let us consider the signal to noise ratio for some positive definite $\boldsymbol{H} \succ c\boldsymbol{I}$

$$\frac{\sqrt{\mathrm{Var}[\boldsymbol{u}^T\boldsymbol{H}\boldsymbol{u}]}}{\mathbb{E}[\boldsymbol{u}^T\boldsymbol{H}\boldsymbol{u}]} = \sqrt{2+m_4}\sqrt{\frac{\mathrm{Tr}\,\boldsymbol{H}^2}{\mathrm{Tr}^2\,\boldsymbol{H}}} = \sqrt{\frac{2+m_4}{P}}\sqrt{\frac{\langle\lambda^2\rangle}{\langle\lambda\rangle}} \tag{22}$$

where we denote the mean eigenvalue $\langle\lambda\rangle$ and the mean square eigenvalue similarly. □

Remark C.2. Note that m_4 is 3 for the Gaussian case and 1 for the Hutchinson trace estimator where the entries are ± 1 with probability half, which justifies its use.

C.1 Deriving the impact of Low precision Lanczos

Consider a number taken from our Hessian matrix $a_{i,j}$, which can be represented as $(-1)^s 2^e s$. As the exponent for FP16 has 5 bits, it has a range of $2^5 - 1$. Since the exponent is always integer, there is no loss of information in the range. This means the error is in the significand, which has 6 bits after the 1. Thus, we have $\epsilon = 10^{-7}$.

Then, we see that

$$\tilde{\boldsymbol{H}} = \begin{bmatrix} a_{1,1}(1+\mathcal{N}(0,\epsilon)) & a_{1,2}(1+\mathcal{N}(0,\epsilon)) & \cdots & a_{1,n}(1+\mathcal{N}(0,\epsilon)) \\ a_{2,1}(1+\mathcal{N}(0,\epsilon)) & a_{2,2}(1+\mathcal{N}(0,\epsilon)) & \cdots & a_{2,n}(1+\mathcal{N}(0,\epsilon)) \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1}(1+\mathcal{N}(0,\epsilon)) & a_{m,2}(1+\mathcal{N}(0,\epsilon)) & \cdots & a_{m,n}(1+\mathcal{N}(0,\epsilon)) \end{bmatrix}$$

$$= \boldsymbol{H} + \begin{bmatrix} a_{1,1}\mathcal{N}(0,\epsilon) & a_{1,2}\mathcal{N}(0,\epsilon) & \cdots & a_{1,n}\mathcal{N}(0,\epsilon) \\ a_{2,1}\mathcal{N}(0,\epsilon) & a_{2,2}\mathcal{N}(0,\epsilon) & \cdots & a_{2,n}\mathcal{N}(0,\epsilon) \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1}\mathcal{N}(0,\epsilon) & a_{m,2}\mathcal{N}(0,\epsilon) & \cdots & a_{m,n}\mathcal{N}(0,\epsilon) \end{bmatrix}.$$

Now under certain assumptions on the elements of the perturbation matrix (essentially the $a_{i,j}$ does not vary too wildly or have wild dependencies) this becomes a Gaussian orthogonal ensemble (GOE) again. Then using the Frobenius Norm, we see that the spectral width will be of order $\epsilon\sqrt{\langle\lambda^2\rangle}$, which depends on the square root of the average eigenvalue squared of $\boldsymbol{H}$. Anything within this will be noise. This is because $\sum_{i,j} a_{i,j}^2 = P\langle\lambda^2\rangle$. An obvious upper bound of this would be $\epsilon\lambda_1$ but this will likely be super loose. Note that the vast majority of the already broadened spectrum is already very close to zero, so we would expect this to be even more extreme for the unbroadened version. A better strategy might be to sample the noisy version of $a_{i,j}^2$ perhaps using the diagonal approximation, and note that in expectations we expect the square to be $(1+\epsilon^2)$ the size of its non noisy counter part, which gives an an estimation equation

$$\sqrt{\frac{P\epsilon^2 \sum_k^N a_{i,j}^2}{N(1+\epsilon^2)}}.$$

D Stochastic Lanczos Quadrature Proof

Theorem D.1. *Consider a symmetric positive definite matrix $\boldsymbol{A} \in \mathbf{R}^{n\times n}$ with eigenvalues enumerated in reverse order of size $\lambda_1 \geq \lambda_2 \cdots \geq \lambda_n$ and condition number $\kappa = \frac{\lambda_1}{\lambda_n}$. For $\epsilon, \eta \in (0,1)$ and SLQ parameters satisfying*

(i) $m \geq \frac{\log \frac{K}{\epsilon}}{2\log\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}} >= \frac{\sqrt{\kappa}}{4}\log\frac{K}{\epsilon}$ *Lanczos steps*

(ii) $n_v \geq \frac{24}{\epsilon^2}\log\frac{2}{\eta}$ *Rademacher vectors,*

where $K = (\lambda_{max} - \lambda_{min})(\sqrt{\kappa}-1)^2$. The output Γ of stochastic lanczos quadrature is such that

$$\Pr\left[\left|\frac{\mathrm{Tr}(\sqrt{\boldsymbol{A}}) - \Gamma}{\mathrm{Tr}(\sqrt{\boldsymbol{A}})}\right| \leq \epsilon\right] \geq 1-\eta \tag{23}$$

Proof. The proof follows trivially from Ubaru et al. (2017), where we simply take the more general proof and instead of the general function $f(\boldsymbol{A})$, we take $f(x) = \sqrt{x}$. The second inequality for m is directly from the paper, but the tighter bound is also available just buried. □

The proof sketch goes as follows. We bound the error from the Gauss quadrature rule. We start with a function analytic in the interval $[-1,1]$. Knowing that the Gauss quadrature rule is exact for any

polynomial up to degree $2m+1$, we bound the sum from $2m+1$ to infinity using Cauchy-Schwarz. We use results from Chebyshev coefficients, symmetry and the interval boundaries to get

$$|I - I_m| \leq \frac{4\sqrt{\lambda_1}}{(\rho^2 - 1)\rho^{2m}},$$

where ρ is the sum of the major and minor axis of the Bernstein elipse. We shift the spectrum so that it is in the interval $[-1, 1]$, e.g this implies the factor of $\frac{\lambda_1 - \lambda_n}{2}$. The shifted function is not analytic for $\alpha = -\frac{-\kappa+1}{\kappa-1}$, so this will serve as our major axis. Now as $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$ and the focus is $1 = \sqrt{a^2 - b^2}$, where we take our major axis a in this case to be α. We then have our rate of convergence $\rho = a + b$ through some algebra to be $\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}$. This gives us the value of K. This is combined with the error of the trace estimator from Roosta-Khorasani & Ascher (2015) and Cauchy-Schwartz to obtain the final result.