entropy

MDPI

Article

Population Risk Improvement with Model Compression: An Information-Theoretic Approach †

Yuheng Bu [1,‡], Weihao Gao [2], Shaofeng Zou [3] and Venugopal V. Veeravalli [1,*]

[1] Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61820, USA; buyuheng@mit.edu
[2] Bytedance Inc., Bellevue, WA 98004, USA; weihao.gao@bytedance.com
[3] Department of Electrical Engineering, University at Buffalo, The State University of New York, Buffalo, NY 14221, USA; szou3@buffalo.edu
* Correspondence: vvv@illinois.edu
† This paper is an extended version of our paper published in AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020.
‡ Current address: Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02142, USA.

Abstract: It has been reported in many recent works on deep model compression that the population risk of a compressed model can be even better than that of the original model. In this paper, an information-theoretic explanation for this population risk improvement phenomenon is provided by jointly studying the decrease in the generalization error and the increase in the empirical risk that results from model compression. It is first shown that model compression reduces an information-theoretic bound on the generalization error, which suggests that model compression can be interpreted as a regularization technique to avoid overfitting. The increase in empirical risk caused by model compression is then characterized using rate distortion theory. These results imply that the overall population risk could be improved by model compression if the decrease in generalization error exceeds the increase in empirical risk. A linear regression example is presented to demonstrate that such a decrease in population risk due to model compression is indeed possible. Our theoretical results further suggest a way to improve a widely used model compression algorithm, i.e., Hessian-weighted *K*-means clustering, by regularizing the distance between the clustering centers. Experiments with neural networks are provided to validate our theoretical assertions.

Keywords: empirical risk; generalization error; K-means clustering; model compression; population risk; rate distortion theory; vector quantization

Citation: Bu, Y.; Gao, W.; Zou, S.; Veeravalli, V.V. Population Risk Improvement with Model Compression: An Information-Theoretic Approach. *Entropy* **2021**, *23*, 1255. https://doi.org/10.3390/e23101255

Academic Editors: Lizhong Zheng and Chao Tian

Received: 13 August 2021
Accepted: 23 September 2021
Published: 27 September 2021

1. Introduction

Although deep neural networks have achieved remarkable success in various domains [1], e.g., computer vision [2], playing games like Go [3], and autonomous driving [4], the improvement of the performance of deep models often comes with deeper layers and more complex network structures, which usually have a large number of parameters. For example, in the application of image classification, it takes over 200 MB to save the parameters of AlexNet [2] and more than 500 MB for VGG-16 net [5]. Hence, it is difficult to port such large models to resource-limited devices such as mobile devices and embedded systems, due to their limited storage, bandwidth, energy, and computational resources.

Due to this reason there has been a flurry of work on compressing deep neural networks (see [6–8] for recent surveys). Existing studies mainly focus on designing compression algorithms to reduce the memory and computational cost, while keeping the same level of population risk. In some recent papers [9–12], aggressive model compression algorithms have been proposed, which require 10% or fewer bits to store the compressed model compared to the storage required by the original model. Surprisingly, it has been

observed empirically in these works that the population risk of the compressed model can often be even *better* than that of the original model. This phenomenon is counter-intuitive at first glance, since more compression generally leads to more information loss.

Indeed, a compressed model would usually have a larger empirical risk than the original one, since machine learning methods are usually trained by minimizing the empirical risk. On the other hand, model compression could possibly decrease the generalization error, since it can be interpreted as a regularization technique to avoid overfitting. As the population risk is the sum of the empirical risk and the generalization error, it is possible for the population risk to be reduced by model compression.

1.1. Contributions

In this paper, we provide an information-theoretic explanation for the population risk improvement with model compression by jointly characterizing the decrease in generalization error and the increase in empirical risk. Specifically, we focus on the case where the model is compressed based on a pre-trained model.

We first prove that model compression leads to a tightening of the information-theoretic generalization error bound in [13], and it can therefore be interpreted as a regularization method to reduce overfitting. Furthermore, by defining a distortion metric based on the difference in the empirical risk between the original model obtained by empirical risk minimization (ERM) and compressed models, we use rate distortion theory to characterize the increase in empirical risk as a function of the number of bits R used to describe the model. If the decrease in generalization error exceeds the increase in empirical risk, the population risk can be improved. An empirical illustration of this result for the MNIST dataset is provided in Figure 1, where model compression can lead to population risk improvement (details are given in Section 7). To better demonstrate our theoretical results, we investigate the example of linear regression comprehensively, where we develop explicit bounds on the generalization error and the increase in empirical risk.

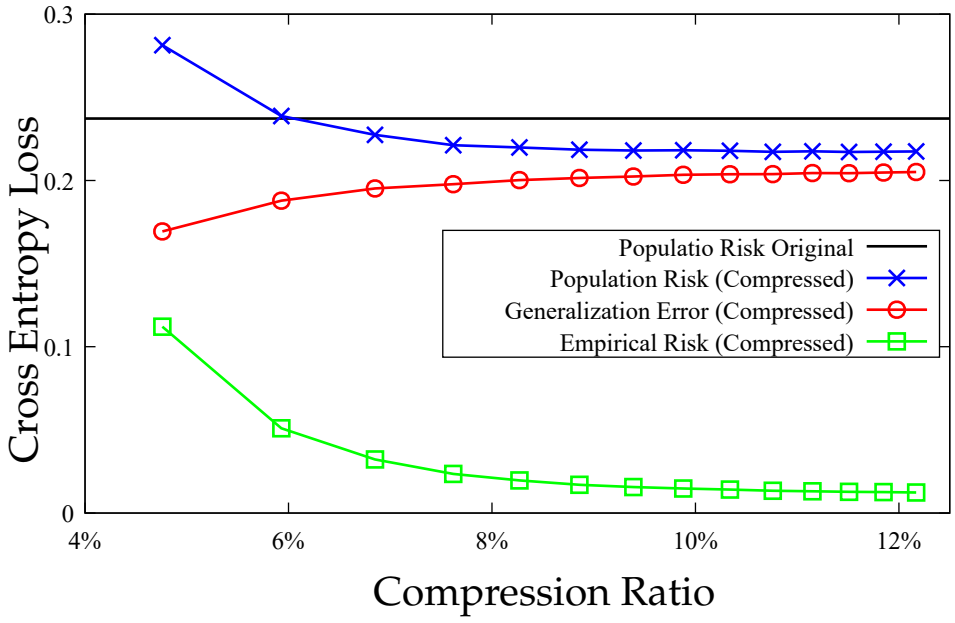

Figure 1. Population risk of the compressed model $\hat{W}$ and the original model W vs. compression ratio (ratio of the number of bits used for compressed model to the number of bits used for original model). The generalization error of $\hat{W}$ decreases and the empirical risk of $\hat{W}$ increases with more compression (smaller compression ratio). The population risk of $\hat{W}$ is less than that of W for compression ratios larger than 6% in this figure. As the compression ratio goes to 100% (no compression), the population risk of $\hat{W}$ will converge to that of the original model W.

Our results also suggest a way to improve a method for compression based on Hessian-weighted K-means clustering [11] in both scalar and vector case, by regularizing the distance between the clustering centers. Our experiments with neural networks validate our theoretical assertions and demonstrate the effectiveness of the proposed regularizer.

1.2. Related Works

There have been many studies on model compression for deep neural networks. The compression could be achieved by varying the training process, e.g., network structure optimization [14], low precision neural networks [15], and neural networks with binary weights [16,17]. Here we mainly discuss compression approaches that are applied on a pre-trained model.

Pruning, quantization, and matrix factorization are the most popular approaches to compressing pre-trained deep neural networks. The study of pruning algorithms for model compression which remove redundant parameters from neural networks dates back to the 1980s and 1990s [18–20]. More recently, an iterative pruning and retraining algorithm to further reduce the size of deep models was proposed in [9,21]. The method of network quantization or weight sharing, i.e., employing a clustering algorithm to group the weights in a neural network, and its variants, including vector quantization [22], soft quantization [23,24], fixed point quantization [25], transform quantization [26], and Hessian weighted quantization [11], have been extensively investigated. Matrix factorization, where low-rank approximation of the weights in neural networks is used instead of the original weight matrix, has also been widely studied in [27–29].

All of the aforementioned works demonstrate the effectiveness of their compression methods via comprehensive numerical experiments. Little research has been done to develop a theoretical understanding of how model compression affects performance. In work [30], an information-theoretic view of model compression via rate-distortion theory is provided, with the focus on characterizing the tradeoff between model compression and only the *empirical risk* of the compressed model. In [31–33], using a PAC-Bayesian framework, a non-vacuous generalization error bound for compressed model is derived based on its smaller model complexity.

In contrast to these works, instead of focusing on minimizing only the empirical risk as in [30], or minimizing only the generalization error as in [33], we use the mutual information based generalization error bound developed in [13,34] jointly with rate distortion theory to connect analyses of generalization error and empirical risk. This way, we are able to characterize the tradeoff between decrease in generalization error and the increase in empirical risk that results from model compression, and thus provide an understanding as to why model compression can improve the population risk. More importantly, our theoretical studies offer insights on designing practical model compression algorithms.

The rest of the paper is organized as follows. In Section 2, we provide relevant definitions and review relevant results from rate distortion theory. In Section 3, we prove that model compression results in the tightening of an information-theoretic generalization error upper bound. In Section 4, we use rate distortion theory to characterize the tradeoff between the increase in empirical risk and the decrease in generalization error that results from model compression. In Section 5, we quantify this tradeoff for a linear regression model. In Section 6, we discuss how the Hessian-weighted K-means clustering compression approach can be improved by using a regularizer motivated by our theoretical results. In Section 7, we provide some experiments with neural network models to validate our theoretical results and demonstrate the effectiveness of the proposed regularizer.

Notation 1. *For a random variable X generated from a distribution μ, we use $\mathbb{E}_{X\sim\mu}$ to denote the expectation taken over X with distribution μ. We use I_d to denote the d-dimensional identity matrix and $\|A\|$ to denote the spectral norm of a matrix A. The cumulant generating function (CGF) of a random variable X is defined as $\Lambda_X(\lambda) \triangleq \ln \mathbb{E}[e^{\lambda(X-\mathbb{E}X)}]$. All logarithms are the natural ones.*

2. Preliminaries

2.1. Review of Rate Distortion Theory

Rate distortion theory, introduced by Shannon [35], is a major branch of information theory that studies the fundamental limits of lossy data compression. It addresses the minimal number of bits per symbol, as measured by the rate R, to transmit a random variable W such that the receiver can reconstruct W without exceeding distortion D.

Specifically, let $W^m = \{W_1, W_2, \cdots, W_m\}$ denote a sequence of m i.i.d. random variables $W_i \in \mathcal{W}$ generated from a source distribution P_W. An encoder $f_m : \mathcal{W}^m \to \{1, 2, \cdots, M\}$ maps the message W^m into a codeword, and a decoder $g_m : \{1, 2, \cdots, M\} \to \hat{\mathcal{W}}^m$ reconstructs the message by an estimate $\hat{W}^m$ from the codeword, where $\hat{\mathcal{W}} \subseteq \mathcal{W}$ denotes the range of $\hat{W}$. A distortion metric $d : \mathcal{W} \times \mathcal{W} \to \mathbb{R}^+$ quantifies the difference between the original and reconstructed messages. The distortion between sequences w^m and $\hat{w}^m$ is defined to be

$$d(w^m, \hat{w}^m) \triangleq \frac{1}{m} \sum_{i=1}^{m} d(w_i, \hat{w}_i). \tag{1}$$

A commonly used distortion metric is the square distortion: $d(w, \hat{w}) = (w - \hat{w})^2$.

Definition 1. *An (m, M, D)-triple is achievable, if there exists a (probabilistic) encoder-decoder pair (f_m, g_m) such that the alphabet of codeword has size M and the expected distortion $\mathbb{E}[d(W^m; g_m(f_m(W^m)))] \leq D$.*

Now we define the following rate-distortion and distortion-rate function for lossy data compression.

Definition 2. *The rate-distortion function and the distortion-rate function are defined as*

$$R(D) \triangleq \lim_{m \to \infty} \frac{1}{m} \log_2 M^*(m, D), \tag{2}$$

$$D(R) \triangleq \lim_{m \to \infty} D^*(m, R), \tag{3}$$

where $M^*(m, D) \triangleq \min\{M : (m, M, D)$ is achievable$\}$ and $D^*(m, R) \triangleq \min\{D : (m, 2^{mR}, D)$ is achievable$\}$.

The main theorem of rate distortion theory is as follows.

Lemma 1 ([36])**.** *For an i.i.d. source W with distribution P_W and distortion function $d(w, \hat{w})$:*

$$R(D) = \min_{P_{\hat{W}|W} : \mathbb{E}[d(W, \hat{W})] \leq D} I(W; \hat{W}), \tag{4}$$

$$D(R) = \min_{P_{\hat{W}|W} : I(W; \hat{W}) \leq R} \mathbb{E}[d(W, \hat{W})], \tag{5}$$

where $I(W; \hat{W}) \triangleq \mathbb{E}_{W, \hat{W}}[\ln \frac{P_{W, \hat{W}}}{P_W P_{\hat{W}}}]$ denotes the mutual information between W and $\hat{W}$.

The rate-distortion function quantifies the smallest number of bits required to compress the data given the distortion, and the distortion-rate function quantifies the minimal distortion that can be achieved under the rate constraint.

2.2. Generalization Error

Consider an instance space $\mathcal{Z}$, a hypothesis space $\mathcal{W}$, and a non-negative loss function $\ell : \mathcal{W} \times \mathcal{Z} \to \mathbb{R}^+$. A training dataset $S = \{Z_1, \cdots, Z_n\}$ consists of n i.i.d samples $Z_i \in \mathcal{Z}$

drawn from an unknown distribution μ. The goal of a supervised learning algorithm is to find an output hypothesis $w \in \mathcal{W}$ that minimizes the population risk:

$$L_\mu(w) \triangleq \mathbb{E}_{Z\sim\mu}[\ell(w, Z)]. \tag{6}$$

In practice, μ is unknown, and therefore $L_\mu(w)$ cannot be computed directly. Instead, the empirical risk of w on the training dataset S is studied, which is defined as

$$L_S(w) \triangleq \frac{1}{n}\sum_{i=1}^{n} \ell(w, Z_i). \tag{7}$$

A learning algorithm can be characterized by a randomized mapping from the training dataset S to a hypothesis W according to a conditional distribution $P_{W|S}$. The (expected) generalization error of a supervised learning algorithm is the expected difference between the population risk of the output hypothesis and its empirical risk on the training dataset:

$$\text{gen}(\mu, P_{W|S}) \triangleq \mathbb{E}_{W,S}[L_\mu(W) - L_S(W)], \tag{8}$$

where the expectation is taken over the joint distribution $P_{S,W} = P_S \otimes P_{W|S}$. The generalization error is used to measure the extent to which the learning algorithm overfits the training data.

3. Compression Can Improve Generalization

In this section, we show that lossy compression can lead to a tighter mutual information based generalization error upper bound, which potentially reduces the generalization error of a supervised learning algorithm.

We start from the following lemma which provides an upper bound on the generalization error using the mutual information $I(S;W)$ between training dataset S and the output of the learning algorithm W.

Lemma 2 ([13]). *Suppose $\ell(w, Z)$ is σ-sub-Gaussian (A random variable X is σ-sub-Gaussian if $\Lambda_X(\lambda) \leq \frac{\sigma^2\lambda^2}{2}, \forall\lambda \in \mathbb{R}$.) under $Z \sim \mu$ for all $w \in \mathcal{W}$, then*

$$|\text{gen}(\mu, P_{W|S})| \leq \sqrt{\frac{2\sigma^2}{n} I(S;W)}. \tag{9}$$

Compression can be viewed as a post-processing of the output of a learning algorithm. The output model W generated by a learning algorithm can be quantized, pruned, factorized, or even perturbed by noise, which results in a compressed model $\hat{W}$. Assume that the compression algorithm is only based on W and can be described by a conditional distribution $P_{\hat{W}|W}$. Then the following Markov chain holds: $S \to W \to \hat{W}$. By the data processing inequality,

$$I(S;\hat{W}) \leq \min\{I(W;\hat{W}), I(S,W)\}.$$

Thus, we have the following theorem characterizing the generalization error of the compressed model.

Theorem 1. *Consider a learning algorithm $P_{W|S}$, a compression algorithm $P_{\hat{W}|W}$, and suppose $\ell(\hat{w}, Z)$ is σ-sub-Gaussian under $Z \sim \mu$ for all $\hat{w} \in \hat{\mathcal{W}}$. Then*

$$|\text{gen}(\mu, P_{\hat{W}|S})| \leq \sqrt{\frac{2\sigma^2}{n} \min\{I(W;\hat{W}), I(S,W)\}}. \tag{10}$$

Note that the generalization error upper bound in Theorem 1 for the compressed model is always no greater than the one in Lemma 2. This allows for the interpretation of compression as a regularization technique to reduce the generalization error.

4. Generalization Error and Model Distortion

In this section, we define a distortion metric in model compression that allows us to relate the distortion (the increase in empirical risk) due to compression with the reduction in the generalization error bound discussed in Section 3.

4.1. Distortion Metric in Model Compression

The expected population risk of a model W can be written as

$$\mathbb{E}_W[L_\mu(W)] = \mathbb{E}[L_S(W)] + \text{gen}(\mu, P_{W|S}), \tag{11}$$

where the first term, which is the expected empirical risk, reflects how well the model W fits the training data, while the second term demonstrates how well the model generalizes. In the empirical risk minimization framework, we control both terms by (1) minimizing the empirical risk of W directly or using other stochastic optimization algorithms, and (2) using regularization methods to control the generalization error, e.g., early stopping and dropout [1].

Now, consider the expected population risk of the compressed model $\hat{W}$:

$$\begin{aligned}\mathbb{E}_{\hat{W}}[L_\mu(\hat{W})] &= \mathbb{E}[L_\mu(\hat{W}) - L_S(\hat{W}) + L_S(\hat{W}) - L_S(W) + L_S(W)] \\ &= \mathbb{E}[L_S(W)] + \text{gen}(\mu, P_{\hat{W}|S}) + \mathbb{E}[L_S(\hat{W}) - L_S(W)].\end{aligned} \tag{12}$$

Compared with (11), we note that the first empirical risk term is independent of the compression algorithm, the second generalization error term can be upper bounded by Theorem 1, and the third term $\mathbb{E}[L_S(\hat{W}) - L_S(W)]$ quantifies the increase in the empirical risk if we use the compressed model $\hat{W}$ instead of the original model W. We then define the following distortion metric for model compression:

$$d_S(w, \hat{w}) \triangleq L_S(\hat{w}) - L_S(w), \tag{13}$$

which is the difference in the empirical risk between the compressed model $\hat{W}$ and the original model W. In general, function $d_S(w, \hat{w})$ is not always non-negative. However, for ERM solution W, which is obtained by minimizing the empirical risk $L_S(W)$, $d_S(w, \hat{w}) \geq 0$, which ensures that $d_S(w, \hat{w})$ is a valid distortion metric. By Theorem 1, it follows that

$$\mathbb{E}_{S,W,\hat{W}}[L_\mu(\hat{W}) - L_S(W)] \leq \sqrt{\frac{2\sigma^2}{n} I(W;\hat{W})} + \mathbb{E}_{S,W,\hat{W}}[d_S(\hat{W}, W)] \triangleq \mathcal{L}_{S,W}(P_{\hat{W}|W}), \tag{14}$$

where $\mathcal{L}_{S,W}(P_{\hat{W}|W})$ is an upper bound on the expected difference between the population risk of $\hat{W}$ and the empirical risk of the original model W on training dataset S. Note that $L_S(W)$ is independent of the compression algorithm. Therefore, the bound in (14) can be viewed as an upper bound of the population risk of the compressed model $\hat{W}$.

4.2. Population Risk Improvement

By Lemma 1, the smallest distortion that can be achieved at rate R is $D(R) = \min_{I(W;\hat{W}) \leq R} \mathbb{E}_{S,W,\hat{W}}[d_S(\hat{W}, W)]$. Thus, the tightest bound in (14) that can be achieved at rate R is given in the following theorem.

Theorem 2. *Suppose the assumptions in Theorem 1 hold, $P_{W|S}$ minimizes the empirical risk $L_S(W)$, and $I(W;\hat{W}) = R$, then*

$$\min_{P_{\hat{W}|W}: I(W;\hat{W})=R} \mathbb{E}_{S,W,\hat{W}}[L_\mu(\hat{W}) - L_S(W)] \leq \sqrt{\frac{2\sigma^2}{n} R} + D(R). \tag{15}$$

From the properties of the distortion-rate function [36], we know that $D(R)$ is a decreasing function of R. Thus, we see that as R decreases the first term in (15), which corresponds to the generalization error, decreases, while the second term, which corresponds to the empirical risk, increases. Due to this tradeoff, it may be possible for the bound in (15) to be smaller due to compression, i.e., using a smaller rate R. This indicates that the population risk could improve with compression algorithm, which minimizes the upper bound $\mathcal{L}_{S,W}(P_{\hat{W}|W})$.

Remark 1. *In order to conclude definitively that the population risk can be improved with compression, we need to find a lower bound (as a function of R) to match (at least in the order sense) the upper bound in Theorem 2. This appears to be difficult to construct in general. One approach might be to use the same decomposition as in* (12) *and develop lower bounds for* $\min_{I(W;\hat{W})=R} \mathrm{gen}(\mu, P_{\hat{W}|S})$ *and* $\min_{I(W;\hat{W})=R} \mathbb{E}_{S,W,\hat{W}}[d_S(\hat{W}, W)]$ *independently. However, such an approach runs into the following issues: (1) such a lower bound would be loose since the compression algorithm* $P_{\hat{W}|W}$ *that minimizes generalization error, the one that minimizes the distortion, and the one that minimizes the sum of the two can be quite different; and (2) a lower bound for generalization error needs to be developed, which appears to be difficult, with existing literature mainly focusing on lower bounding the excess risk, e.g., [37].*

As will be shown in Section 7, we can actually improve the population risk with a well designed compression algorithm in practical applications.

5. Example: Linear Regression

In this section, we comprehensively explore the example of linear regression to get a better understanding of the results in Section 4. To this end, we develop explicit upper bounds for generalization error and distortion-rate function $D(R)$. All the proofs of the lemmas and theorems are provided in the Appendixes A–D.

Suppose that the dataset $S = \{Z_1, \cdots, Z_n\} = \{(X_1, Y_1), \cdots, (X_n, Y_n)\}$ is generated from the following linear model with weight vector $w^* = (w^{*(1)}, \cdots, w^{*(d)}) \in \mathbb{R}^d$,

$$Y_i = X_i^\top w^* + \varepsilon_i, \quad i = 1, \cdots, n, \tag{16}$$

where X_i's are i.i.d. d-dimensional random vectors with distribution $\mathcal{N}(0, \Sigma_X)$, and $\varepsilon_i \sim \mathcal{N}(0, \sigma'^2)$ denotes i.i.d. Gaussian noise. We adopt the mean squared error as the loss function, and the corresponding empirical risk on S is

$$L_S(w) = \frac{1}{n}\sum_{i=1}^{n}(Y_i - X_i^\top w)^2 = \frac{1}{n}\|Y - X^\top w\|_2^2, \tag{17}$$

for $w \in \mathcal{W} = \mathbb{R}^d$, where $X \in \mathbb{R}^{d\times n}$ denotes all the input samples, and $Y \in \mathbb{R}^n$ denotes the responses. If $n > d$, the ERM solution is

$$W = (XX^\top)^{-1}XY, \tag{18}$$

which is deterministic given S. Its generalization error can be computed exactly as in the following lemma (see Appendix A for detailed proof).

Lemma 3. *If* $n > d + 1$, *then*

$$\mathrm{gen}(\mu, P_{W|S}) = \frac{\sigma'^2 d}{n}\left(2 + \frac{d+1}{n-d-1}\right). \tag{19}$$

5.1. Information-Theoretic Generalization Bounds for Compressed Linear Model

We note that the mutual information based bound in Lemma 2 is not applicable for this linear regression model, since W is a deterministic function of S, and $I(S; W) = \infty$. However,

this issue can be resolved if we post-process the ERM solution W by a compression algorithm and upper bound the generalization error by $I(\hat{W};W)$ as shown in Theorem 1.

Consider a compression algorithm, which maps the original weights $W \in \mathbb{R}^d$ to the compressed model $\hat{W} \in \hat{\mathcal{W}} \subseteq \mathbb{R}^d$. For a fixed and compact $\hat{\mathcal{W}}$, we define

$$C(w^*) \triangleq \sup_{\hat{w}\in\hat{\mathcal{W}}} \|\hat{w} - w^*\|_2^2, \tag{20}$$

which measures the largest distance between the reconstruction $\hat{w}$ and the optimal weights w^*. The following proposition provides an upper bound on the generalization error of the compressed model $\hat{W}$, and the detailed proof is provided in Appendix B.

Proposition 1. *Consider the ERM solution $W = (XX^\top)^{-1}XY$, and suppose $\hat{\mathcal{W}}$ is compact, then*

$$\mathrm{gen}(\mu, P_{\hat{W}|S}) \leq 2\sigma_\ell^{*2}\sqrt{\frac{I(W;\hat{W})}{n}}, \tag{21}$$

where $\sigma_\ell^{*2} \triangleq C(w^*)\|\Sigma_X\| + \sigma'^2$.

5.2. Distortion-Rate Function for Linear Model

We now provide an upper bound on the distortion-rate function $D(R)$ for the linear regression model. Note that $\nabla L_S(W) = 0$, since W minimizes the empirical risk. The Hessian matrix of the loss function is

$$H_S(W) = \frac{1}{n}XX^\top, \tag{22}$$

which is not a function of W. Then, the distortion function can be written as:

$$\begin{aligned}\mathbb{E}_{S,W,\hat{W}}[d_S(\hat{W},W)] &= \mathbb{E}_{S,W,\hat{W}}[L_S(\hat{W}) - L_S(W)] \\ &= \mathbb{E}_{S,W,\hat{W}}[(\hat{W}-W)^\top \frac{1}{n}XX^\top(\hat{W}-W)].\end{aligned} \tag{23}$$

The following theorem characterizes upper bounds for $R(D)$ and $D(R)$ for linear regression.

Proposition 2. *For the ERM solution $W = (XX^\top)^{-1}XY$, we have*

$$R(D) \leq \frac{d}{2}\Big(\ln\frac{d\sigma'^2}{(n-d-1)D}\Big)^+, \quad D \geq 0, \tag{24}$$

$$D(R) \leq \frac{d\sigma'^2}{n-d-1}e^{-\frac{2R}{d}}, \quad R \geq 0, \tag{25}$$

where $(x)^+ = \max\{0,x\}$.

Proof sketch. The proof of the upper bound for $R(D)$ is based on considering a Gaussian random vector which has the same mean and covariance matrix as W. In addition, the upper bound is achieved when $W - \hat{W}$ is independent of the dataset S with the following conditional distribution,

$$P_{\hat{W}|W} = \mathcal{N}\big((1-\alpha)W + \alpha w^*, (1-\alpha)\frac{D}{d}\Sigma_X^{-1}\big), \tag{26}$$

where $\alpha \triangleq \frac{nD}{d\sigma'^2} \leq 1$. Note that this "compression algorithm" requires the knowledge of optimal weights w^*, which is unknown in practice.

The details can be found in Appendix C. □

Remark 2. *As shown in [38], if* $n > d/\epsilon^2$, $\|\frac{1}{n}XX^\top - \Sigma_X\| \leq \epsilon$ *holds with high probability. Then, the following lower bound on* $R(D)$ *holds if we can approximate* $\frac{1}{n}XX^\top$ *in (23) using* Σ_X,

$$R(D) \gtrsim \frac{d}{2}\Big(\ln \frac{d\sigma'^2}{(n-d-1)D}\Big)^+ - D(P_W \| P_{W_G}), \tag{27}$$

where W_G *denotes a Gaussian random vector with the same mean and variance as* W. *The details can be found in Appendix D.*

Combing Propositions 1 and 2, we have the following result.

Corollary 1. *Under the same assumptions as in Propositions 1, we have*

$$\min_{P_{\hat{W}|W}: I(W;\hat{W})=R} \mathbb{E}_{S,W,\hat{W}}[L_\mu(\hat{W}) - L_S(W)] \leq 2\sigma_\ell^{*2}\sqrt{\frac{R}{n}} + \frac{d\sigma'^2}{n-d-1}e^{-\frac{2R}{d}}, \quad R \geq 0. \tag{28}$$

In (28) the first term corresponds to the generalization error, which decreases with compression, and the second term corresponds to the empirical risk, which increases with compression.

5.3. Evaluation and Visualization

In the following plots, we generate the training dataset S using the linear model in (16) by letting $d = 50$, $n = 80$, $\Sigma_X = I_d$ and $\sigma'^2 = 1$. We consider the following two compression algorithms. The first one is the conditional distribution $P_{\hat{W}|W}$ in the proof of achievability (26), which requires the knowledge of w^* and is denoted as "Oracle". The second one is the well-known K-means clustering algorithm, where the weights in W are grouped into K clusters and represented by the cluster centers in the reconstruction $\hat{W}$. By changing the number of clusters K, we can control the rate R, i.e., $I(W;\hat{W})$. We average the performance and estimate $I(W;\hat{W})$ of these algorithms with 10,000 Monte-Carlo trials in the simulation.

We note that $I(W;\hat{W})$ is equal to the number of bits used in compression only in the asymptotic regime of large number of samples. In practice, we may have only one sample of the weights W, and therefore $I(W;\hat{W})$ simply measures the extent to which compression is performed by the compression algorithm.

In Figure 2a, we plot the generalization error bound in Proposition 1 as a function of the rate R and compare the generalization errors of the Oracle and K-means algorithms. It can be seen that Proposition 1 provides a valid upper bound for the generalization error, but this bound is tight only when R is small. Moreover, both compression algorithms can achieve smaller generalization errors compared to that of the ERM solution W, which validates the result in Theorem 1.

Figure 2b plots the upper bound on the distortion-rate function in Theorem 2 and the distortions achieved by the Oracle and K-means algorithms. The distortion of the Oracle decreases as we increase the rate R and matches the $D(R)$ function well. However, there is a large gap between the distortion achieved by K-means algorithms and $D(R)$. One possible explanation is that since w^* is unknown, it is impossible for the K-means algorithm to learn the optimal cluster center with only one sample of W. Even if we view $W^{(j)}, j = 1, \cdots, d$ as i.i.d. samples from the same distribution, there is still a gap between the distortion achieved by the K-means algorithm and the optimal quantization as studied in [39].

We plot the population risks of the ERM solution W, the Oracle, and K-means algorithms in Figure 2c. It is not surprising that the Oracle algorithm achieves a small population risk, since $\hat{W}$ is a function of w^* and $\hat{W} = w^*$ when $R = 0$. However, it can be seen that the K-means algorithm achieves a smaller population risk than the original model W, since the decrease in generalization error exceeds the increase in empirical risk,

when we use fewer clusters in the K-means algorithm, i.e., a smaller rate R. We note that the minimal population risk is achieved when $K = 2$, since we initialize w^* so that $w^{*(i)}$, $1 \leq i \leq d$, can be well approximated by two cluster centers.

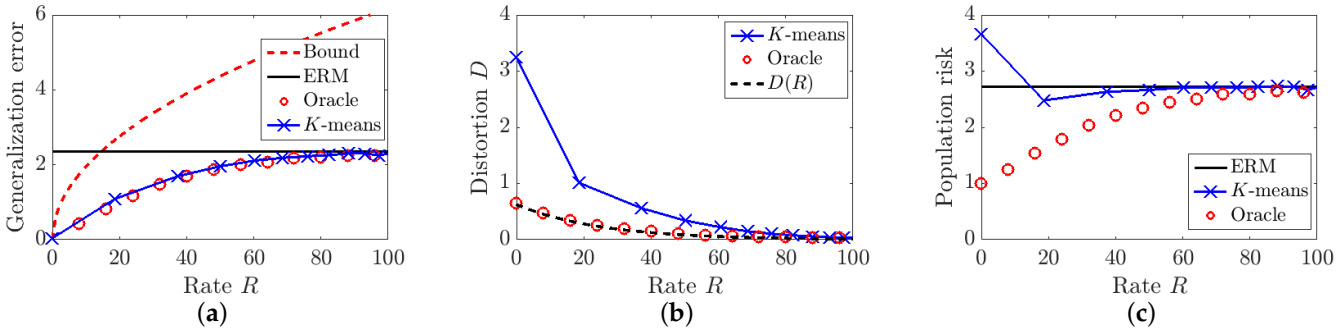

Figure 2. Comparison of three different quantities for linear regression as a function of rate R in bits. (**a**) Generalization error. (**b**) Distortion. (**c**) Population risk.

6. Clustering Algorithm Minimizing $\mathcal{L}_{S,W}$

In this section, we propose an improvement of the Hessian-weighted (HW) K-means clustering algorithm [11] for model compression by regularizing the distance between the cluster centers, which minimizes the upper bound $\mathcal{L}_{S,W}(P_{\hat{W}|W})$, as suggested by our theoretical results in Section 4.

6.1. Hessian-Weighted K-Means Clustering

The goal of HW K-means is to minimize the distortion on the empirical risk $d_S(\hat{W}, W)$, which has the following Taylor series approximation:

$$d_S(\hat{W}, W) \approx (\hat{W} - W)^T \nabla L_S(W) + \frac{1}{2}(\hat{W} - W)^T H_S(W)(\hat{W} - W), \tag{29}$$

where $H_S(W)$ is the Hessian matrix. Assuming that W is a local minimum of $L_S(W)$ (ERM solution) and $\nabla L_S(W) \approx 0$, the first term can be ignored. Furthermore, the Hessian matrix $H_S(W)$ can be approximated by a diagonal matrix, which further simplifies the objective to $d_S(\hat{W}, W) \approx \sum_{j=1}^{d} h^{(j)}(W^{(j)} - \hat{W}^{(j)})^2$, where $h^{(j)}$ is the j-th diagonal element of the Hessian matrix.

Given network parameters $w = \{w^{(1)}, \cdots, w^{(d)}\}$, the HW K-means clustering algorithm [11] partitions them into K disjoint clusters, using a set of cluster centers $c = \{c^{(1)}, \cdots, c^{(K)}\}$, and a cluster assignment $C = \left\{C^{(1)}, \cdots, C^{(K)}\right\}$, while solving the following optimization problem:

$$\min \sum_{k=1}^{K} \sum_{w^{(j)} \in C^{(k)}} h^{(j)} |w^{(j)} - c^{(k)}|^2. \tag{30}$$

6.2. Diameter Regularization

In contrast to HW K-means which only cares about empirical risk, our goal is to obtain as small a population risk as possible by minimizing the upper bound

$$\mathcal{L}_{S,W}(P_{\hat{W}|W}) = \sqrt{\frac{2\sigma^2}{n} I(W; \hat{W})} + \mathbb{E}[d_S(\hat{W}, W)]. \tag{31}$$

Here, we let the number of clusters K to be an input argument of the algorithm, so that $I(W; \hat{W}) \leq \log_2 K$, and we want to minimize $\mathcal{L}_{S,W}(P_{\hat{W}|W})$ by carefully designing the reconstructed weights given K, i.e., by choosing cluster centers $\{c^{(1)}, \cdots, c^{(K)}\}$. Then,

minimizing the sub-Gaussian parameter σ is one way to control the generalization error of the compression algorithm. Recall that in Proposition 1, we have

$$\mathrm{gen}(\mu, P_{\hat{W}|S}) \leq 2\big(C(w^*)\|\Sigma_X\| + \sigma'^2\big)\sqrt{\frac{I(W;\hat{W})}{n}}, \tag{32}$$

where the sub-Gaussian parameter is related to $C(w^*) = \sup_{\hat{w}\in\hat{\mathcal{W}}} \|\hat{w} - w^*\|_2^2$ in linear regression. Note that this quantity can be interpreted as the diameter of the set $\mathcal{W}$. Since the ground truth w^* is unknown in practice, we then propose the following diameter regularization by approximating $C(w^*)$ in (32) by

$$\beta \max_{k_1,k_2} |c^{(k_1)} - c^{(k_2)}|^2, \quad \beta \geq 0, \tag{33}$$

where β is a parameter controls the penalty term and can be selected by cross validation in practice. Our diameter-regularized Hessian-weighted (DRHW) K-means algorithm solves the following optimization problem:

$$\min \sum_{k=1}^{K} \sum_{w^{(j)}\in C^{(k)}} h^{(j)}|w^{(j)} - c^{(k)}|^2 + \beta \max_{k_1,k_2} |c^{(k_1)} - c^{(k_2)}|^2. \tag{34}$$

Such an optimization problem can be easily extended to the vector case which leads to a vector quantization algorithm. Suppose that we group the d-dimensional weights $w = \{w^{(1)}, \cdots, w^{(d)}\}$ into $d' = d/m$ vectors with length m, i.e., $\{\mathbf{w}^{(1)}, \cdots, \mathbf{w}^{(d')}\}$, $\mathbf{w}^{(j)} \in \mathbb{R}^m$, then our goal is to find cluster centers $\mathbf{c}^k \in \mathbb{R}^m$ and assignments minimizing the following cost function:

$$\min \sum_{k=1}^{K} \sum_{\mathbf{w}^{(j)}\in C^{(k)}} (\mathbf{w}^{(j)} - \mathbf{c}^{(k)})^\top H^{(j)} (\mathbf{w}^{(j)} - \mathbf{c}^{(k)}) + \beta \max_{k_1,k_2} \|\mathbf{c}^{(k_1)} - \mathbf{c}^{(k_2)}\|_2^2, \tag{35}$$

where $H^{(j)}$ is the diagonal Hessian matrix corresponding to the vector $\mathbf{w}^{(j)}$. An iterative algorithm to solve the above optimization problem for vector quantization is provided in Algorithm 1.

The algorithm alternates between minimizing the objective function over the cluster centers and the assignments. In the Assignment step, we first fix centers and assign each $\mathbf{w}^{(j)}$ to its nearest neighbor. We then fix assignments and update the centers by the weighted mean of each cluster in the Update step. For the farthest pair of centers, the diameter regularizer pushes them toward each other, so that the output centers have potentially smaller diameters than those of regular K-means. We note that the time complexity of the proposed diameter-regularized Hessian weighted K-means algorithm is the same as that of the original K-means algorithm.

Algorithm 1 Diameter-regularized Hessian weighted K-means in vector case

Input: Weights vector $\{\mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(d')}\}$, Hessian matrices $\{H^{(1)}, \ldots, H^{(d')}\}$, diameter regularizer $\beta > 0$, number of clusters K, iterations T

Initialize the K cluster centers $\{\mathbf{c}_0^{(1)}, \ldots, \mathbf{c}_0^{(K)}\}$ randomly

for $t = 1$ to T **do**

Assignment step:

Initialize $C_t^{(k)} = \emptyset$ for all $k \in [K]$.

for $j = 1$ to d' **do**

Assign $\mathbf{w}^{(j)}$ to the nearest cluster center, i.e., find $k_t^{(j)} = \arg\min_{k\in[K]} \|\mathbf{w}^{(j)} - \mathbf{c}_{t-1}^{(k)}\|_2^2$ and let

$$C_t^{(k_t^{(j)})} \leftarrow C_t^{(k_t^{(j)})} \cup \{\mathbf{w}^{(j)}\} \tag{36}$$

end for
Update step:
Find current farthest pair of centers $(k_1, k_2) = \arg\max_{k_1,k_2} \|\mathbf{c}_{t-1}^{(k_1)} - \mathbf{c}_{t-1}^{(k_2)}\|_2^2$.
Update $\mathbf{c}_t^{(k_1)}$ and $\mathbf{c}_t^{(k_2)}$ by

$$\mathbf{c}_t^{(k_1)} = \Big(\sum_{\mathbf{w}^{(j)} \in C_t^{(k_1)}} H^{(j)} + \beta I_m \Big)^{-1} \Big(\sum_{\mathbf{w}^{(j)} \in C_t^{(k_1)}} H^{(j)} \mathbf{w}^{(j)} + \beta \mathbf{c}_t^{(k_2)} \Big)$$
$$\mathbf{c}_t^{(k_2)} = \Big(\sum_{\mathbf{w}^{(j)} \in C_t^{(k_2)}} H^{(j)} + \beta I_m \Big)^{-1} \Big(\sum_{\mathbf{w}^{(j)} \in C_t^{(k_2)}} H^{(j)} \mathbf{w}^{(j)} + \beta \mathbf{c}_t^{(k_1)} \Big) \tag{37}$$

for $k = 1$ to K, $k \notin \{k_1, k_2\}$ **do**
Update the cluster centers by

$$\mathbf{c}_t^{(k)} = \Big(\sum_{\mathbf{w}^{(j)} \in C_t^{(k)}} H^{(j)} \Big)^{-1} \Big(\sum_{\mathbf{w}^{(j)} \in C_t^{(k)}} H^{(j)} \mathbf{w}^{(j)} \Big) \tag{38}$$

end for
end for
Output: centers $\{\mathbf{c}_T^{(1)}, \ldots, \mathbf{c}_T^{(K)}\}$ and assignments $\{C_T^{(1)}, \ldots, C_T^{(K)}\}$.

7. Experiments

In this section, we provide some real-world experiments to validate our theoretical assertions and the DRHW K-means algorithm. (The code for our experiments is available at the following link https://github.com/wgao9/weight-quant (accessed on 13 August 2021)) Our experiments include compression of: (i) a three-layer fully connected network on the MNIST dataset [40]; and (ii) a convolutional neural network with five convolutional layers and three linear layers on the CIFAR10 dataset [41] (We downloaded the pre-trained model in PyTorch from https://github.com/aaron-xichen/pytorch-playground (accessed on 13 August 2021)).

In Theorem 1, an upper bound on the *expected* generalization error is provided, and therefore we independently train 50 different models (with the same structure but different parameter initializations) using different subset of training samples, and average the results. We use 10% of the training data to train the model for MNIST and use 20% of the training data to train the model for CIFAR10. For each experiment, we use the same number of clusters for each convolutional layer and fully connected layer.

In the following experiments, we plot the cross entropy loss as a function of compression ratio. Note that compression ratio can be controlled by changing the number of clusters K in the quantization algorithm. To see this, suppose that the neural networks have total of d parameters that need to be compressed, and each parameter is of b bits. Let $C^{(k)}$ be the set of weights in cluster k and let b_k be the number of bits of the codeword assigned to the network parameters in cluster k for $1 \leq k \leq K$. For a lookup table to decode quantized values, we need Kb bits to store all the reconstructed weights, i.e., cluster centers $c = \{c^{(1)}, \cdots, c^{(K)}\}$. Then, the compression ratio is given by

$$\text{Compression Ratio} = \frac{\sum_{k=1}^{K} |C^{(k)}| b_k + Kb}{db}, \tag{39}$$

where $|\cdot|$ denotes the number of elements in the set. In our experiments, we use a variable-length code such as the Huffman code to compute the compression ratio under different numbers of clusters K.

In Figures 3 and 4, we compare the scalar DRHW K-means algorithm with the scalar HW K-means algorithm for different compression ratios on the MNIST and CIFAR10

datasets. Both figures demonstrate that the compression algorithm increases the empirical risk but decreases the generalization error, and the net effect is that the both compressed models have smaller population risks than those of the original models. More importantly, the DRHW K-means algorithm produces a compressed model that has a better population risk than that of the HW K-means algorithm.

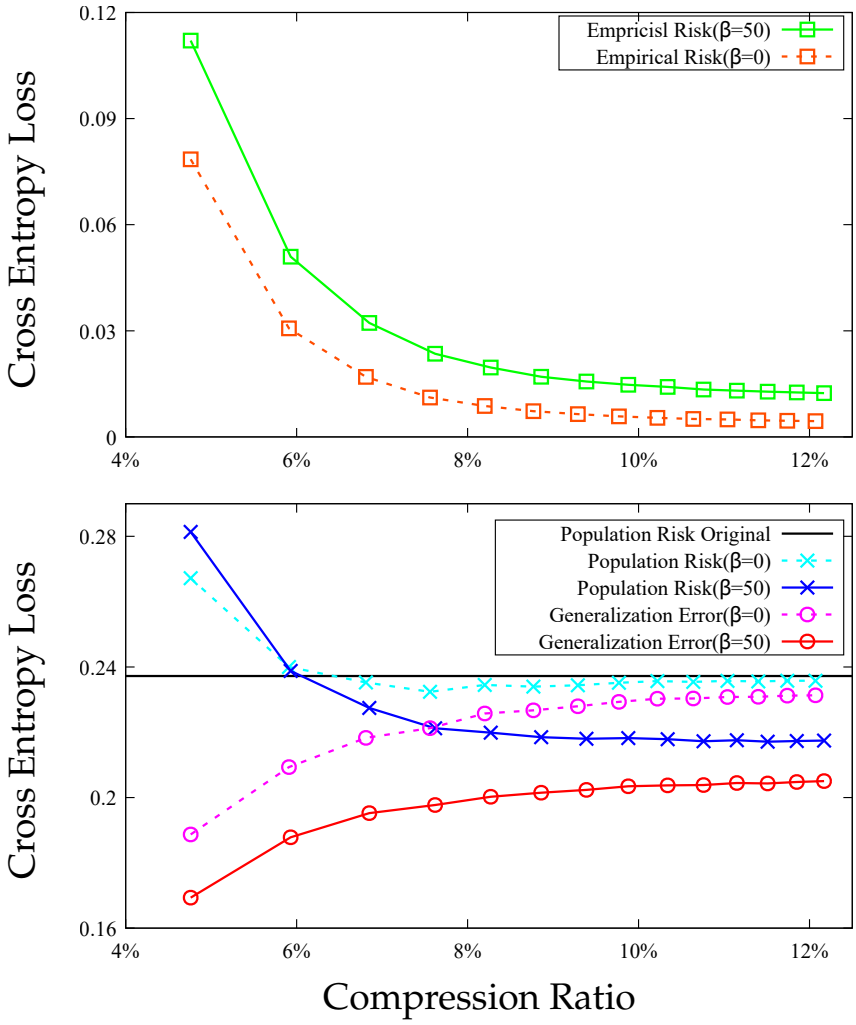

Figure 3. Comparison between DRHW K-means ($\beta = 50$) and HW K-means ($\beta = 0$) on MNIST. **Top**: empirical risks. **Bottom**: population risks and generalization errors.

In Figure 5, we compare the population risk of scalar DRHW K-means algorithm and that of the vector DRHW K-means algorithm with block length $m = 2$ for different compression ratios on the MNIST dataset. It can be seen from the figure that the improvement by using vector quantization ($m = 2$) is quite modest, which implies that the dependence between the weights $W^{(j)}$ is weak. However, we can still observe the improvement of adding the diameter regularizer in vector DRHW K-means algorithm by comparing the curves with $\beta = 50$ and $\beta = 0$.

In Figure 6, we demonstrate how β affects the performance of our diameter-regularized Hessian-weighted K-means algorithm in scalar case. It can be seen that as β increases, the generalization error decreases and the distortion in empirical risk increases, which validates the idea that this proposed diameter regularizer can be used to reduce the generalization error. The value of β that results in the best population risk therefore can be chosen via cross-validation in practice.

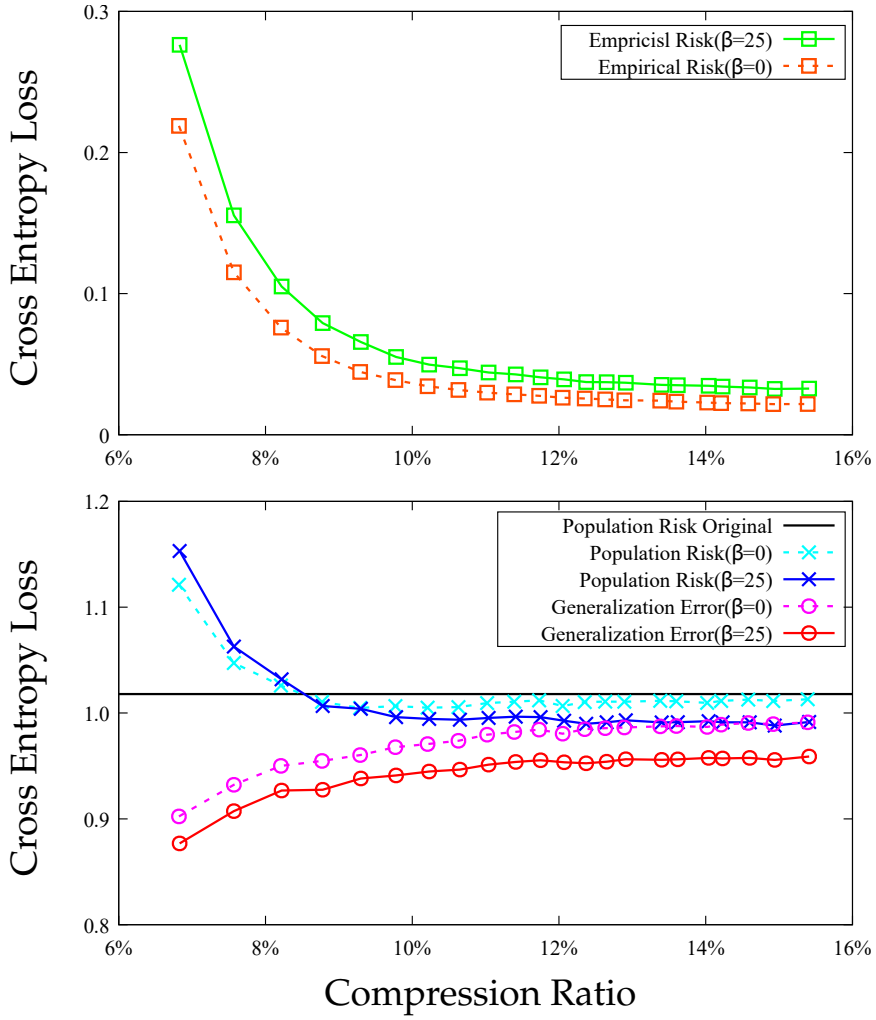

Figure 4. Comparison between DRHW *K*-means ($\beta = 25$) and HW *K*-means ($\beta = 0$) on CIFAR10. **Top**: empirical risks. **Bottom**: population risks and generalization errors.

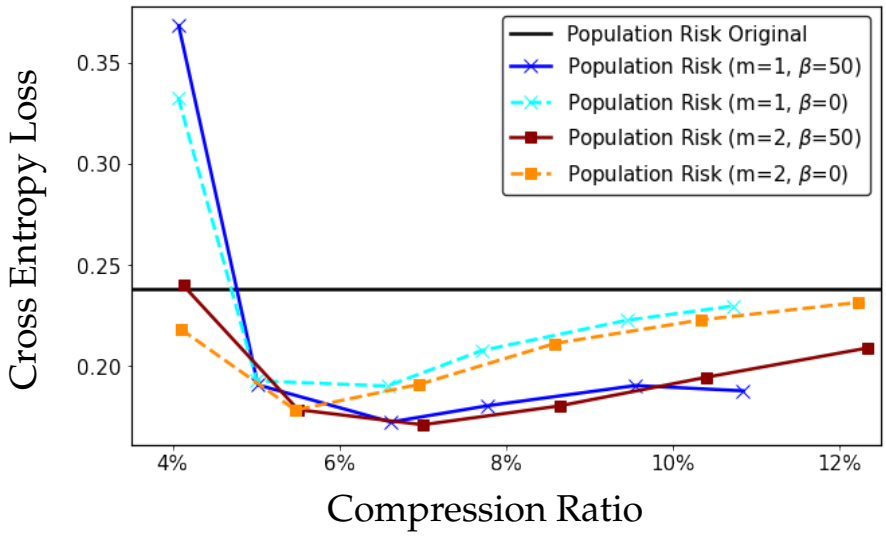

Figure 5. Comparison between scalar DRHW *K*-means ($m = 1$) and vector DRHW *K*-means ($m = 2$) on the MNIST dataset.

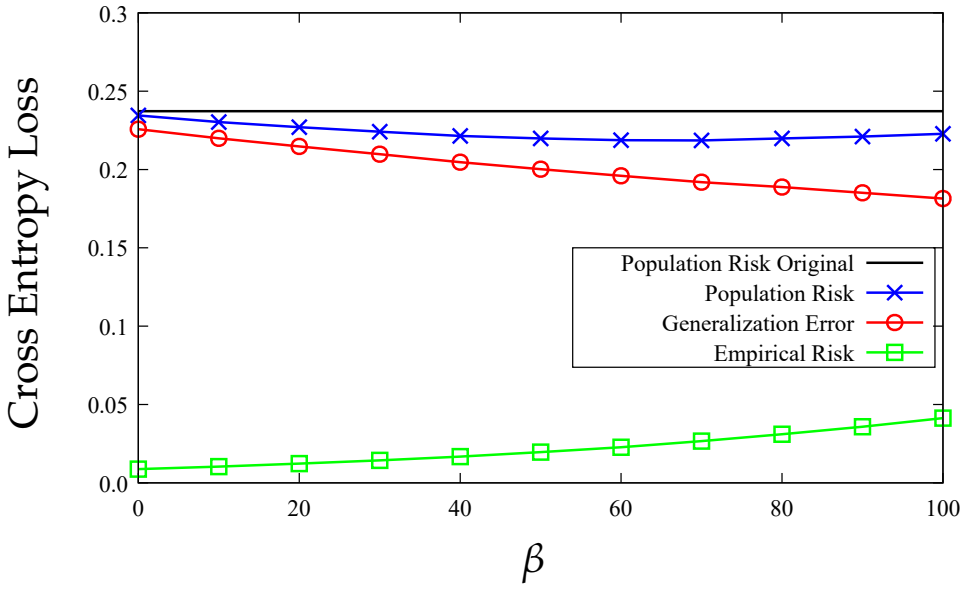

Figure 6. DRHW K-means with different β on the MNIST dataset with $K = 7$.

8. Conclusions

In this paper, we have provided an information-theoretical understanding of how model compression affects the population risk of a compressed model. In particular, our results indicate that model compression may increase the empirical risk but decrease the generalization error. Therefore, it might be possible to achieve a smaller population risk via model compression. Our experiments validate these theoretical findings. Furthermore, we showed how our information-theoretic bound on the population risk can be used to optimize practical compression algorithms.

We note that our results could be applied to improve other compression algorithms, such as pruning and matrix factorization. Moreover, we believe that the information-theoretic analysis adopted here could be generalized to characterize a similar tradeoff between the generalization error and empirical risk in other applications beyond compressing pre-trained models, e.g., distributed optimization [42] and low precision training [15].

Author Contributions: Y.B.: theoretical analysis, methodology, conceptualization, writing—original draft. W.G.: software, methodology and visualization. S.Z.: writing—review and editing. V.V.V.: supervision, funding acquisition, and writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Army Research Laboratory under Cooperative Agreement W911NF-17-2-0196, through the University of Illinois at Urbana-Champaign.

Data Availability Statement: Data and code can be found in Section 7.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Proof of Lemma 3

Let $\widetilde{Z} = (\widetilde{X}, \widetilde{Y})$, $\widetilde{X} \in \mathbb{R}^d$ and $\widetilde{Y} \in \mathbb{R}$ denote an independent copy of the training sample Z_i. Then, it can be shown that

$$\begin{aligned}
\mathrm{gen}(\mu, P_{W|S}) &= \mathbb{E}_{W,S}[L_\mu(W) - L_S(W)] \\
&= \mathbb{E}_{W,S}\left[\mathbb{E}_{\widetilde{Z}}[(\widetilde{Y} - \widetilde{X}^\top W)^2] - \frac{1}{n}\|Y - X^\top W\|_2^2\right] \\
&= \mathbb{E}_S\left[\mathbb{E}_{\widetilde{Z}}[(\widetilde{Y} - \widetilde{X}^\top (XX^\top)^{-1}XY)^2] - \frac{1}{n}\|Y - X^\top (XX^\top)^{-1}XY\|_2^2\right], \qquad \text{(A1)}
\end{aligned}$$

where $\widetilde{Y} = \widetilde{X}^\top w^* + \widetilde{\varepsilon}$ and $Y = X^\top w^* + \varepsilon$. Then, we have

$$\begin{aligned}\mathrm{gen}(\mu, P_{W|S}) &= \mathbb{E}_{\varepsilon,\widetilde{\varepsilon},X,\widetilde{X}}\big[(\widetilde{\varepsilon} - \widetilde{X}^\top (XX^\top)^{-1} X\varepsilon)^2\big] - \frac{1}{n}\mathbb{E}_{\varepsilon,X}\big[\|\varepsilon - X^\top (XX^\top)^{-1} X\varepsilon\|_2^2\big] \\ &= \mathbb{E}_{\varepsilon,X,\widetilde{X}}\big[\varepsilon^\top X^\top (XX^\top)^{-1} \widetilde{X}\widetilde{X}^\top (XX^\top)^{-1} X\varepsilon + \frac{1}{n}\varepsilon^\top X^\top (XX^\top)^{-1} X\varepsilon\big] \\ &= \mathbb{E}_{\varepsilon,X}\big[\mathrm{Tr}(X^\top (XX^\top)^{-1}\Sigma_X (XX^\top)^{-1} X\varepsilon\varepsilon^\top)\big] + \frac{\sigma'^2 d}{n} \\ &= \sigma'^2 \mathbb{E}_X\big[\mathrm{Tr}((XX^\top)^{-1}\Sigma_X)\big] + \frac{\sigma'^2 d}{n}. \end{aligned} \tag{A2}$$

Note that X_i's are i.i.d. samples from $\mathcal{N}(0, \Sigma_X)$, then we have $(XX^\top)^{-1}$ distributed according to Wishart$^{-1}(\Sigma_X^{-1}, n)$, where Wishart^{-1} denotes the inverse Wishart distribution with n degrees of freedom, and $\mathbb{E}[(XX^\top)^{-1}] = \frac{\Sigma_X^{-1}}{n-d-1}$. It then follows that

$$\mathrm{gen}(\mu, P_{W|S}) = \frac{\sigma'^2}{n-d-1}\big[\mathrm{Tr}(\Sigma_X^{-1}\Sigma_X)\big] + \frac{\sigma'^2 d}{n} = \frac{\sigma'^2 d}{n}(2 + \frac{d+1}{n-d-1}). \tag{A3}$$

Appendix B. Proof of Proposition 1

For all $\hat{w} \in \hat{\mathcal{W}}$, it can be shown that

$$\ell(\hat{w}, \widetilde{Z}) = (\widetilde{Y} - \widetilde{X}^\top \hat{w})^2 = (\widetilde{X}^\top (w^* - \hat{w}) + \widetilde{\varepsilon})^2. \tag{A4}$$

Since $\widetilde{X} \sim \mathcal{N}(0, \Sigma_X)$ and $\widetilde{\varepsilon} \sim \mathcal{N}(0, \sigma'^2)$, then $\ell(\hat{w}, \widetilde{Z}) \sim \sigma_\ell^2 \chi_1^2$, where

$$\sigma_\ell^2 \triangleq (\hat{w} - w^*)^\top \Sigma_X (\hat{w} - w^*) + \sigma'^2,$$

and χ_1^2 denotes the chi-squared distribution with one degree of freedom. Then, the CGF of $\ell(\hat{w}, \widetilde{Z})$ is

$$\Lambda_{\ell(\hat{w},\widetilde{Z})}(\lambda) = -\sigma_\ell^2 \lambda - \frac{1}{2}\ln(1 - 2\sigma_\ell^2 \lambda),\ \lambda \in (-\infty, \frac{1}{2\sigma_\ell^2}). \tag{A5}$$

Thus, $\ell(\hat{w}, \widetilde{Z})$ is not sub-Gaussian for all $\lambda \in \mathbb{R}$. However, it can be shown that

$$\Lambda_{\ell(\hat{w},\widetilde{Z})}(\lambda) \le \sigma_\ell^4 \lambda^2, \quad \lambda < 0. \tag{A6}$$

We need the following lemma from the Theorem 1 of [43] to proceed our analysis.

Lemma A1 ([43]). *Assume that for all $\hat{w} \in \hat{\mathcal{W}}$, $\Lambda_{\ell(\hat{w},\widetilde{Z})}(\lambda) \le \frac{\sigma^2\lambda^2}{2}$ for $\lambda \le 0$. Then,*

$$\mathrm{gen}(\mu, P_{\hat{W}|S}) \le \sqrt{\frac{2\sigma^2}{n} I(\hat{W}; S)}. \tag{A7}$$

Recall that $C(w^*) = \sup_{\hat{w}\in\hat{\mathcal{W}}} \|\hat{w} - w^*\|_2^2$. We then have the following bound on the CGF of $\ell(\hat{w}, \widetilde{Z})$,

$$\Lambda_{\ell(\hat{w},\widetilde{Z})}(\lambda) \le \lambda^2 \max_{\hat{w}\in\hat{\mathcal{W}}} \sigma_\ell^4 \le \lambda^2 \big(C(w^*)\|\Sigma_X\| + \sigma'^2\big)^2, \quad \lambda < 0. \tag{A8}$$

Applying Lemma A1 and data processing inequality, we have

$$\mathrm{gen}(\mu, P_{\hat{W}|S}) \le 2\big(C(w^*)\|\Sigma_X\| + \sigma'^2\big)\sqrt{\frac{I(\hat{W}; W)}{n}}. \tag{A9}$$

Appendix C. Proof of Proposition 2

The constraint on the distortion function can be written as follows:

$$D \geq \mathbb{E}_{S,W,\hat{W}}[d_S(\hat{W},W)] = \frac{1}{n}\mathbb{E}_{S,W,\hat{W}}[(\hat{W}-W)^\top XX^\top(\hat{W}-W)]. \tag{A10}$$

It follows from Lemma 1 that

$$R(D) = \min_{P_{\hat{W}|W}} I(\hat{W};W), \quad \text{s.t.} \quad \mathbb{E}_{S,W,\hat{W}}[(\hat{W}-W)^\top \frac{1}{n}XX^\top(\hat{W}-W)] \leq D. \tag{A11}$$

Note that $\mathbb{E}[W] = w^*$ and $\mathrm{Cov}[W] = \frac{\sigma'^2}{n-d-1}\Sigma_X^{-1}$ since W is the ERM solution. In the following proof, we consider a Gaussian random vector with the same mean and covariance matrix $W_G \sim \mathcal{N}(w^*, \frac{\sigma'^2}{n-d-1}\Sigma_X^{-1})$ as W.

For the upper bound of $R(D)$, consider the channel $P^*_{\hat{W}|W} = \mathcal{N}\big((1-\alpha)W + \alpha w^*, (1-\alpha)\frac{D}{d}\Sigma_X^{-1}\big)$, where $\alpha = \frac{nD}{d\sigma'^2} \leq 1$. It can be verified that this channel satisfies the constraint on the distortion:

$$\begin{aligned}
&\mathbb{E}_{S,W,\hat{W}}[d_S(\hat{W},W)] \\
&= \alpha^2\mathbb{E}[(W-w^*)^\top \frac{1}{n}XX^\top(W-w^*)] + (1-\alpha)\frac{D}{d}\mathrm{Tr}\Big(\mathbb{E}[\frac{1}{n}XX^\top]\Sigma_X^{-1}\Big) \\
&= \alpha^2\mathbb{E}[\big((XX^\top)^{-1}X\varepsilon\big)^\top \frac{1}{n}XX^\top\big((XX^\top)^{-1}X\varepsilon\big)] + (1-\alpha)D \\
&= \alpha^2\frac{1}{n}\mathbb{E}[\varepsilon^\top X^\top(XX^\top)^{-1}X\varepsilon] + (1-\alpha)D \\
&= D.
\end{aligned} \tag{A12}$$

If we let $\xi \sim \mathcal{N}(0, (1-\alpha)\frac{D}{d}\Sigma_X^{-1})$, it follows that

$$\begin{aligned}
R(D) &\leq I(W; (1-\alpha)W + \alpha w^* + \xi) \\
&\overset{(a)}{\leq} I(W_G; (1-\alpha)W_G + \xi) \\
&= \frac{d}{2}\ln\Big(\frac{d\sigma'^2}{(n-d-1)D} - \frac{n}{n-d-1} + 1\Big) \\
&\leq \frac{d}{2}\Big(\ln\frac{d\sigma'^2}{(n-d-1)D}\Big)^+,
\end{aligned} \tag{A13}$$

where (a) is due to the fact that Gaussian distribution maximizes the mutual information in an additive white Gaussian noise channels.

The upper bound on $D(R)$ follows immediately from the upper bound on $R(D)$.

Appendix D. Discussion of Remark 2

Suppose that $\frac{1}{n}XX^\top$ can be approximated by Σ_X for large n in (A10). It then follows that

$$R(D) = \min_{P_{\hat{W}|W}} I(\hat{W};W), \quad \text{s.t.} \quad \mathbb{E}_{S,W,\hat{W}}[(\hat{W}-W)^\top\Sigma_X(\hat{W}-W)] \leq D. \tag{A14}$$

It can be easily verified that the channel $P^*_{W|\hat{W}} = \mathcal{N}(\hat{W}, \frac{D}{d}\Sigma_X^{-1})$ satisfies the distortion constraint. For any $P_{W|\hat{W}}$ such that $\mathbb{E}_{S,W,\hat{W}}[d_S(\hat{W},W)] \leq D$, it follows that

$$
\begin{aligned}
I(W;\hat{W}) &= \mathbb{E}_{W,\hat{W}}\Big[\ln\frac{P_{W|\hat{W}}}{P_W}\Big] \\
&= \mathbb{E}_{W,\hat{W}}\Big[\ln\frac{P_{W|\hat{W}}}{P^*_{W|\hat{W}}}\Big] + \mathbb{E}_{W,\hat{W}}\Big[\ln\frac{P^*_{W|\hat{W}}}{P_{W_G}}\Big] - \mathrm{KL}(P_W\|P_{W_G}) \\
&\geq \mathbb{E}_{W,\hat{W}}\Big[\ln\frac{P^*_{W|\hat{W}}}{P_{W_G}}\Big] - \mathrm{KL}(P_W\|P_{W_G}),
\end{aligned} \tag{A15}
$$

where $\mathrm{KL}(P_W\|P_{W_G})$ is the Kullback–Leibler divergence between the two distributions, and the last step follows from the fact that $\mathrm{KL}(P_{W,\hat{W}}\|P^*_{W,\hat{W}}) \geq 0$. Note that

$$
\begin{aligned}
&\mathbb{E}_{W,\hat{W}}\Big[\ln\frac{P^*_{W|\hat{W}}}{P_{W_G}}\Big] \\
&= \mathbb{E}_{W,\hat{W}}\Big[\frac{(n-d-1)(W-w^*)^\top\Sigma_X(W-w^*)}{2\sigma'^2} - \frac{d(\hat{W}-W)^\top\Sigma_X(\hat{W}-W)}{2D}\Big] \\
&\quad + \frac{d}{2}\ln\frac{d\sigma'^2}{(n-d-1)D} \\
&\overset{(a)}{=} \frac{d}{2}\ln\frac{d\sigma'^2}{(n-d-1)D} + \mathbb{E}_{W,\hat{W}}\Big[\frac{d}{2} - \frac{d(\hat{W}-W)^\top\Sigma_X(\hat{W}-W)}{2D}\Big] \\
&\overset{(b)}{\geq} \frac{d}{2}\ln\frac{d\sigma'^2}{(n-d-1)D},
\end{aligned} \tag{A16}
$$

where (a) follows from the fact that $\mathbb{E}[W] = w^*$ and $\mathrm{Cov}[W] = \frac{\sigma'^2}{n-d-1}\Sigma_X^{-1}$, and (b) is due to the fact that $P_{\hat{W}|W}$ satisfies the distortion constraint. Thus,

$$
R(D) \geq \frac{d}{2}\ln\frac{d\sigma'^2}{(n-d-1)D} - \mathrm{KL}(P_W\|P_{W_G}). \tag{A17}
$$

References

1. Goodfellow, I.; Bengio, Y.; Courville, A.; Bengio, Y. *Deep Learning*; MIT Press: Cambridge, UK, 2016; Volume 1.
2. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
3. Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. Mastering the game of Go without human knowledge. *Nature* **2017**, *550*, 354. [CrossRef]
4. Huval, B.; Wang, T.; Tandon, S.; Kiske, J.; Song, W.; Pazhayampallil, J.; Andriluka, M.; Rajpurkar, P.; Migimatsu, T.; Cheng-Yue, R. An empirical evaluation of deep learning on highway driving. *arXiv* **2015**, arXiv:1504.01716.
5. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
6. Cheng, Y.; Wang, D.; Zhou, P.; Zhang, T. A survey of model compression and acceleration for deep neural networks. *arXiv* **2017**, arXiv:1710.09282.
7. Krishnamoorthi, R. Quantizing deep convolutional networks for efficient inference: A whitepaper. *arXiv* **2018**, arXiv:1806.08342.
8. Guo, Y. A survey on methods and theories of quantized neural networks. *arXiv* **2018**, arXiv:1808.04752.
9. Han, S.; Mao, H.; Dally, W.J. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv* **2015**, arXiv:1510.00149.
10. Zhu, C.; Han, S.; Mao, H.; Dally, W.J. Trained ternary quantization. *arXiv* **2016**, arXiv:1612.01064.
11. Choi, Y.; El-Khamy, M.; Lee, J. Towards the limit of network quantization. *arXiv* **2016**, arXiv:1612.01543.
12. Lin, Y.; Han, S.; Mao, H.; Wang, Y.; Dally, W.J. Deep gradient compression: Reducing the communication bandwidth for distributed training. *arXiv* **2017**, arXiv:1712.01887.
13. Xu, A.; Raginsky, M. Information-theoretic analysis of generalization capability of learning algorithms. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 2524–2533.

14. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
15. Gupta, S.; Agrawal, A.; Gopalakrishnan, K.; Narayanan, P. Deep learning with limited numerical precision. In Proceedings of the International Conference on Machine Learning (ICML), Lille, France, 6–11 July 2015; pp. 1737–1746.
16. Courbariaux, M.; Bengio, Y.; David, J.P. Binaryconnect: Training deep neural networks with binary weights during propagations. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–12 December 2015; pp. 3123–3131.
17. Rastegari, M.; Ordonez, V.; Redmon, J.; Farhadi, A. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 525–542.
18. Mozer, M.C.; Smolensky, P. Skeletonization: A technique for trimming the fat from a network via relevance assessment. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Denver, CO, USA, 27–30 November 1989; pp. 107–115.
19. LeCun, Y.; Denker, J.S.; Solla, S.A. Optimal brain damage. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Denver, CO, USA, 26–29 November 1990; pp. 598–605.
20. Hassibi, B.; Stork, D.G. Second order derivatives for network pruning: Optimal brain surgeon. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), San Francisco, CA, USA, 30 November–3 December 1992; pp. 164–171.
21. Han, S.; Pool, J.; Tran, J.; Dally, W. Learning both weights and connections for efficient neural network. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–12 December 2015; pp. 1135–1143.
22. Gong, Y.; Liu, L.; Yang, M.; Bourdev, L. Compressing deep convolutional networks using vector quantization. *arXiv* **2014**, arXiv:1412.6115.
23. Ullrich, K.; Meeds, E.; Welling, M. Soft weight-sharing for neural network compression. *arXiv* **2017**, arXiv:1702.04008.
24. Louizos, C.; Ullrich, K.; Welling, M. Bayesian compression for deep learning. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 3288–3298.
25. Lin, D.; Talathi, S.; Annapureddy, S. Fixed point quantization of deep convolutional networks. In Proceedings of the International Conference on Machine Learning, PMLR, New York, NY, USA, 19–24 June 2016; pp. 2849–2858.
26. Young, S.; Wang, Z.; Taubman, D.; Girod, B. Transform Quantization for CNN Compression. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, 1. [CrossRef] [PubMed]
27. Denton, E.L.; Zaremba, W.; Bruna, J.; LeCun, Y.; Fergus, R. Exploiting linear structure within convolutional networks for efficient evaluation. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 8–13 December 2014; pp. 1269–1277.
28. Tai, C.; Xiao, T.; Zhang, Y.; Wang, X. Convolutional neural networks with low-rank regularization. *arXiv* **2017**, arXiv:1511.06067.
29. Novikov, A.; Podoprikhin, D.; Osokin, A.; Vetrov, D.P. Tensorizing neural networks. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–12 December 2015; pp. 442–450.
30. Gao, W.; Liu, Y.H.; Wang, C.; Oh, S. Rate distortion for model compression: From theory to practice. In Proceedings of the International Conference on Machine Learning (ICML), PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 2102–2111.
31. Dziugaite, G.K.; Roy, D.M. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *arXiv* **2017**, arXiv:1703.11008.
32. Neyshabur, B.; Bhojanapalli, S.; McAllester, D.; Srebro, N. Exploring generalization in deep learning. *arXiv* **2017**, arXiv:1706.08947.
33. Zhou, W.; Veitch, V.; Austern, M.; Adams, R.P.; Orbanz, P. Non-vacuous generalization bounds at the imagenet scale: A PAC-bayesian compression approach. *arXiv* **2018**, arXiv:1804.05862.
34. Russo, D.; Zou, J. Controlling bias in adaptive data analysis using information theory. In Proceedings of the International Conference on Artifical Intelligence and Statistics (AISTATS), Cadiz, Spain, 9–11 May 2016; pp. 1232–1240.
35. Shannon, C.E. Coding theorems for a discrete source with a fidelity criterion. *IRE Nat. Conv. Rec* **1959**, *4*, 142–163.
36. Cover, T.M.; Thomas, J.A. *Elements of Information Theory*; John Wiley & Sons: Hoboken, NJ, USA, 2012.
37. Grønlund, A.; Kamma, L.; Larsen, K.G.; Mathiasen, A.; Nelson, J. Margin-Based Generalization Lower Bounds for Boosted Classifiers. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Vancouver, BC, Canada, 8–14 December 2019; pp. 11940–11949.
38. Vershynin, R. Introduction to the non-asymptotic analysis of random matrices. *arXiv* **2010**, arXiv:1011.3027.
39. Linder, T.; Lugosi, G.; Zeger, K. Rates of convergence in the source coding theorem, in empirical quantizer design, and in universal lossy source coding. *IEEE Trans. Inf. Theory* **1994**, *40*, 1728–1740. [CrossRef]
40. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
41. Krizhevsky, A.; Hinton, G. *Learning Multiple Layers of Features from Tiny Images*; Technical Report; University of Toronto: Toronto, ON, Canada, 2009.

42. Basu, D.; Data, D.; Karakus, C.; Diggavi, S. Qsparse-local-SGD: Distributed SGD with Quantization, Sparsification and Local Computations. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Vancouver, BC, Canada, 8–14 December 2019; pp. 14668–14679.
43. Bu, Y.; Zou, S.; Veeravalli, V.V. Tightening Mutual Information-Based Bounds on Generalization Error. *IEEE J. Sel. Areas Inf. Theory* **2020**, *1*, 121–130. [CrossRef]