Information-Theoretic Understanding of Population Risk Improvement with Model Compression

Yuheng Bu,[1*] **Weihao Gao,**[1†] **Shaofeng Zou,**[2] **Venugopal V. Veeravalli**[1]
[1]University of Illinois at Urbana-Champaign, Urbana, IL, USA
[2]University at Buffalo, The State University of New York, Buffalo, NY, USA
buyuheng@mit.edu, weihao.gao@bytedance.com, szou3@buffalo.edu, vvv@illinois.edu

Abstract

We show that model compression can improve the population risk of a pre-trained model, by studying the tradeoff between the decrease in the generalization error and the increase in the empirical risk with model compression. We first prove that model compression reduces an information-theoretic bound on the generalization error; this allows for an interpretation of model compression as a regularization technique to avoid overfitting. We then characterize the increase in empirical risk with model compression using rate distortion theory. These results imply that the population risk could be improved by model compression if the decrease in generalization error exceeds the increase in empirical risk. We show through a linear regression example that such a decrease in population risk due to model compression is indeed possible. Our theoretical results further suggest that the Hessian-weighted K-means clustering compression approach can be improved by regularizing the distance between the clustering centers. We provide experiments with neural networks to support our theoretical assertions.

1 Introduction

The recent success of deep neural networks has dramatically boosted the applications of machine learning (Krizhevsky, Sutskever, and Hinton 2012; Silver et al. 2017; Goodfellow et al. 2016). However, implementing a deep neural network model on resource-limited devices becomes increasingly difficult, as deep neural networks usually have a large number of parameters. For example, for the problem of image classification, it takes over 200MB to save the parameters of AlexNet (Krizhevsky, Sutskever, and Hinton 2012), and more than 500MB for VGG-16 net (Simonyan and Zisserman 2014). It is difficult to port such large models to mobile devices and embedded systems, due to their limited storage, bandwidth, energy and computational resources.

For this reason there has been a flurry of recent work on compressing the parameters of deep neural networks (see (Cheng et al. 2017; Krishnamoorthi 2018; Guo 2018) for recent surveys). Existing studies mainly focus on designing compression algorithms to reduce the memory and computational cost, while keeping the same population risk. However, in some recent works (Choi, El-Khamy, and Lee 2016; Zhu et al. 2016; Lin et al. 2017), it has been observed empirically that the population risk of the compressed model can sometimes be *better* than that of the original model. This phenomenon is counterintuitive at a first glance, since compression generally leads to information loss.

Indeed, as neural networks are usually trained by minimizing the empirical risk, a compressed model has a larger empirical risk than the original one. Despite this fact, model compression could possibly improve 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 characterizing the tradeoff between generalization error and 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 tightens the information-theoretic generalization error bound in (Raginsky et al. 2016), and it can therefore be interpreted as a regularization method to reduce overfitting. Furthermore, we define the distortion as the difference in the empirical risk between the original and compressed models, and use rate distortion theory to characterize the distortion 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 and population risk improvement are achieved simultaneously (details are given in Section 7). To better demonstrate our theoretical results, we investigate an example of linear regression comprehensively, where we develop explicit bounds on the generalization error and the distortion.

Our generalization error bound also suggests that the

[*] Currently with Massachusetts Institute of Technology, Cambridge, MA, USA

[†] Currently with Bytedance Inc., Bellevue, WA, USA

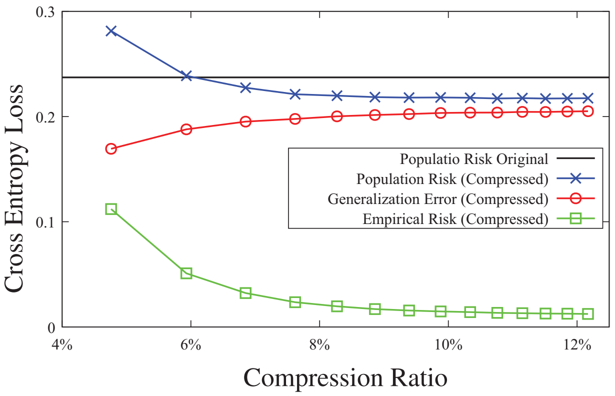

Figure 1: Population risk of the compressed model $\hat{W}$ and the original model W vs. compression ratio (ratio of the number of bits used for compressed model to the number of bits used for original model). The generalization error of $\hat{W}$ decreases and the empirical risk of $\hat{W}$ increases with more compression, i.e., smaller compression ratio. The population risk of $\hat{W}$ is less than that of W for compression ratio larger than 6%.

Hessian-weighted K-means clustering compression approach (Choi, El-Khamy, and Lee 2016) can be improved by further regularizing the distance between the clustering centers. Our numerical 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 happen by varying the training process, e.g., network structure optimization (Howard et al. 2017), low precision neural networks (Gupta et al. 2015) and neural networks with binary weights (Courbariaux, Bengio, and David 2015; Rastegari et al. 2016). 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 (Mozer and Smolensky 1989; LeCun, Denker, and Solla 1990; Hassibi and Stork 1993). Recently, an iterative pruning and retraining algorithm to further reduce the size of deep models is proposed in (Han et al. 2015). In addition, the method of network quantization or weight sharing is investigated in (Gong et al. 2014; Han, Mao, and Dally 2015; Choi, El-Khamy, and Lee 2016; Ullrich, Meeds, and Welling 2017; Louizos, Ullrich, and Welling 2017), where a clustering algorithm is employed to group the weights in a neural network. Matrix factorization, i.e., low-rank approximation of the weights in neural networks has also been widely studied in (Denton et al. 2014; Tai et al. 2015; Novikov et al. 2015).

All of the aforementioned works demonstrate the effectiveness of their methods via comprehensive numerical experiments. Little research has been done to develop a theoretical understanding of how model compression affects performance. In recent work (Gao, Wang, and Oh 2018), an information-theoretic view of model compression via rate-distortion theory is provided, with the focus on purely minimizing the empirical risk of the compressed model. In (Zhou et al. 2018), a non-vacuous generalization error bound based on the small complexity of the compressed model using a PAC-Bayesian framework is discussed.

In contrast to these works, we study the problem from the perspective of the population risk of the compressed model. We develop an understanding as to why model compression can improve population risk based on an analysis of both the empirical risk and generalization error. More importantly, our theoretical studies offer insights on designing practical model compression algorithms, i.e., the increase in empirical risk and the decrease in generalization error should be considered jointly, so that the population risk can be improved.

Notation: 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 natural ones.

2 Preliminaries

2.1 Generalization Error

Consider an instance space $\mathcal{Z}$, a hypothesis space $\mathcal{W}$, and a nonnegative 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:

$$L_\mu(w) \triangleq \mathbb{E}_{Z\sim\mu}[\ell(w, Z)]. \tag{1}$$

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

$$L_S(w) \triangleq \frac{1}{n}\sum_{i=1}^{n} \ell(w, Z_i). \tag{2}$$

A learning algorithm can be characterized by a randomized mapping from the training data set S to a hypothesis W according to a conditional distribution $P_{W|S}$. The 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:

$$\text{gen}(\mu, P_{W|S}) \triangleq \mathbb{E}_{W,S}[L_\mu(W) - L_S(W)], \tag{3}$$

where the expectation is taken over the joint distribution $P_{S,W} = P_S \otimes P_{W|S}$.

2.2 Review of Rate Distortion Theory

Rate distortion theory, firstly introduced by (Shannon 1959), is a major branch of information theory which 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 a given 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

$$d(w^m, \hat{w}^m) \triangleq \frac{1}{m} \sum_{i=1}^{m} d(w_i, \hat{w}_i). \tag{4}$$

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

Definition 1. *An (m, M, D)-pair 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 $\mathbb{E}[d(W^m; g_m(f_m(W^m)))] \leq D$.*

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

$$R(D) \triangleq \lim_{m\to\infty} \frac{1}{m} \log_2 M^*(m, D), \tag{5}$$

$$D(R) \triangleq \lim_{m\to\infty} D^*(m, R), \tag{6}$$

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

The main theorem of rate distortion theory is as follows.

Lemma 1. *(Cover and Thomas 2012) For an i.i.d. source W with distribution P_W and distortion function $d(w, \hat{w})$, it follows that*

$$R(D) = \min_{P_{\hat{W}|W} : \mathbb{E}[d(W, \hat{W})] \leq D} I(W; \hat{W}), \tag{7}$$

$$D(R) = \min_{P_{\hat{W}|W} : I(W; \hat{W}) \leq R} \mathbb{E}[d(W, \hat{W})]. \tag{8}$$

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.

3 Compression Improves Generalization

In this section, we prove that a lossy compression algorithm can be used to improve the generalization error of a supervised learning algorithm via an information-theoretic generalization error bound. We start from the following lemma which provides an upper bound on the generalization error using the mutual information $I(S; W)$ between training data set S and the output of the learning algorithm W.

Lemma 2. *(Xu and Raginsky 2017) Suppose $\ell(w, Z)$ is σ-sub-Gaussian[1] under $Z \sim \mu$ for all $w \in \mathcal{W}$, then*

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

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*

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

Note that the generalization error bound in Theorem 1 for the compressed model is tighter than the one in Lemma 2. Thus, a compression algorithm can be interpreted as a regularization technique to reduce the generalization error.

4 Tradeoff between Generalization Error and Distortion

In this section, we define the distortion metric in model compression, and connect the distortion with the generalization error bound using rate-distortion theory. We show that the population risk can possibly be improved by trading-off between the generalization error and the distortion.

4.1 Distortion Metric in Model Compression

Consider the expected population risk of the compressed model $\hat{W}$,

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

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 distortion 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:

$$d_S(w, \hat{w}) \triangleq L_S(\hat{w}) - L_S(w), \tag{11}$$

[1] A random variable X is σ-sub-Gaussian if $\Lambda_X(\lambda) \leq \frac{\sigma^2 \lambda^2}{2}$, $\forall \lambda \in \mathbb{R}$.

which is the difference in the empirical risk between the compressed model $\hat{W}$ and the original model W. By Theorem 1, it follows that

$$\begin{aligned}&\mathbb{E}_{S,W,\hat{W}}[L_\mu(\hat{W}) - L_S(W)]\\ &\leq \sqrt{\frac{2\sigma^2}{n}I(W;\hat{W})} + \mathbb{E}_{S,W,\hat{W}}[d_S(\hat{W},W)]\\ &\triangleq \mathcal{L}_{S,W}(P_{\hat{W}|W}),\end{aligned} \tag{12}$$

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.

4.2 Population Risk Improvement

By Lemma 1, the tightest bound in (12) that can be achieved at rate R is given in the following theorem.

Theorem 2. *Suppose the assumptions in Theorem 1 hold, and $I(W;\hat{W}) = R$, then*

$$\begin{aligned}\min_{P_{\hat{W}|W}:I(W;\hat{W})=R} &\mathbb{E}_{S,W,\hat{W}}[L_\mu(\hat{W}) - L_S(W)]\\ &\leq \sqrt{\frac{2\sigma^2}{n}R} + D(R).\end{aligned} \tag{13}$$

From the properties of the distortion-rate function (Cover and Thomas 2012), we know that $D(R)$ is a decreasing function of R. Thus, to minimize the population risk of the compressed model $\hat{W}$, there is a tradeoff between the rate R, which upper bounds the generalization error, and the distortion $D(R)$ on the empirical risk. Such a tradeoff is similar to the relationship between the complexity of the hypothesis space, e.g., VC dimension, and the empirical risk, where a simple and small model could have a small generalization error, but may underfit the training data. As will be shown Section 7, such a tradeoff can be observed in practice, and it is possible to improve the population risk of $\hat{W}$ with a properly chosen compression algorithm and compression ratio.

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 bound for generalization error and distortion-rate function $D(R)$. All the proofs are provided in (Bu et al. 2019).

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 $w^* = (w^{*(1)}, \cdots, w^{*(d)}) \in \mathbb{R}^d$,

$$Y_i = X_i^\top w^* + \varepsilon_i,\ i = 1, \cdots, n, \tag{14}$$

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 empirical risk on S is

$$L_S(w) = \frac{1}{n}\sum_{i=1}^n (Y_i - X_i^\top w)^2 = \frac{1}{n}\|Y - X^\top w\|_2^2, \tag{15}$$

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

$$W = (XX^\top)^{-1}XY, \tag{16}$$

which is deterministic given S. Its generalization error can be computed exactly as in the following lemma.

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

$$\text{gen}(\mu, P_{W|S}) = \frac{\sigma'^2 d}{n}(2 + \frac{d+1}{n-d-1}). \tag{17}$$

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 use Theorem 1 to upper bound the generalization error by $I(\hat{W};W)$.

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

$$C(w^*) \triangleq \sup_{\hat{w}\in\hat{\mathcal{W}}} \|\hat{w} - w^*\|_2^2, \tag{18}$$

which measures the largest distance between the reconstruction $\hat{w}$ and the optimal weights w^*. The following theorem provides an upper bound on the generalization error of the compressed model $\hat{W}$.

Theorem 3. *Consider the ERM solution $W = (XX^\top)^{-1}XY$, and suppose $\hat{\mathcal{W}}$ is compact, then*

$$\text{gen}(\mu, P_{\hat{W}|S}) \leq 2\sigma_\ell^{*2}\sqrt{\frac{I(W;\hat{W})}{n}}, \tag{19}$$

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

5.2 Distortion-Rate Function for Linear Model

In this subsection, we provide an upper bound on the distortion-rate function $D(R)$ for the linear regression model, and further demonstrate the tradeoff between generalization error and distortion.

Note that $\nabla L_S(W) = 0$, since W minimizes the empirical risk. The Hessian matrix of the loss function is

$$H_S(W) = \frac{1}{n}XX^\top, \tag{20}$$

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

$$\begin{aligned}&\mathbb{E}_{S,W,\hat{W}}[d_S(\hat{W},W)]\\ &= \mathbb{E}_{S,W,\hat{W}}[L_S(\hat{W}) - L_S(W)]\\ &= \mathbb{E}_{S,W,\hat{W}}[(\hat{W} - W)^\top \frac{1}{n}XX^\top(\hat{W} - W)].\end{aligned} \tag{21}$$

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

Theorem 4. *For the ERM solution $W = (XX^\top)^{-1}XY$, we have*

$$R(D) \le \frac{d}{2}\Big(\ln \frac{d\sigma'^2}{(n-d-1)D}\Big)^+, \quad D \ge 0, \tag{22}$$

$$D(R) \le \frac{d\sigma'^2}{n-d-1}e^{-\frac{2R}{d}}, \quad R \ge 0, \tag{23}$$

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

Remark 1. *As shown in (Vershynin 2010), if* $n = O(d/\epsilon^2)$, $\|\frac{1}{n}XX^\top - \Sigma_X\| \le \epsilon$ *holds with high probability. Then, the following lower bound on* $R(D)$ *holds if we can approximate* $\frac{1}{n}XX^\top$ *in* (21) *using* Σ_X,

$$R(D) \gtrsim \frac{d}{2}\Big(\ln \frac{d\sigma'^2}{(n-d-1)D}\Big)^+ - D(P_W \| P_{W_G}), \tag{24}$$

where W_G *denotes a Gaussian random vector with the same mean and variance as* W.

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,

$$P_{\hat{W}|W} = \mathcal{N}\big((1-\alpha)W + \alpha w^*, (1-\alpha)\frac{D}{d}\Sigma_X^{-1}\big), \tag{25}$$

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

Combing Theorems 3 and 4, we have the following result.

Corollary 1. *Under the same assumptions as in Theorem 3, we have*

$$\min_{P_{\hat{W}|W}: I(W;\hat{W}) = R} \mathbb{E}_{S,W,\hat{W}}[L_\mu(\hat{W}) - L_S(W)] \le 2\sigma_\ell^{*2}\sqrt{\frac{R}{n}} + \frac{d\sigma'^2}{n-d-1}e^{-\frac{2R}{d}}, \quad R \ge 0. \tag{26}$$

It is clear that in (26) 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 data set S using the linear model in (14) 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 (25), 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 10000 Monte-Carlo trials in the simulation. We note that $I(W;\hat{W})$ equals 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.

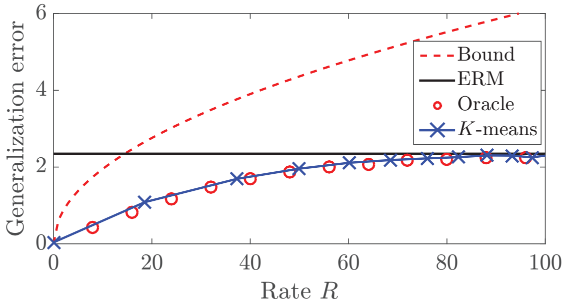

Figure 2: Comparison between the generalization error bound and generalization errors of different algorithms for linear regression.

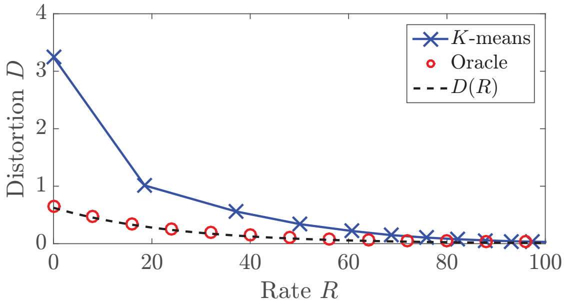

Figure 3: Distortions achieved by different algorithms for linear regression.

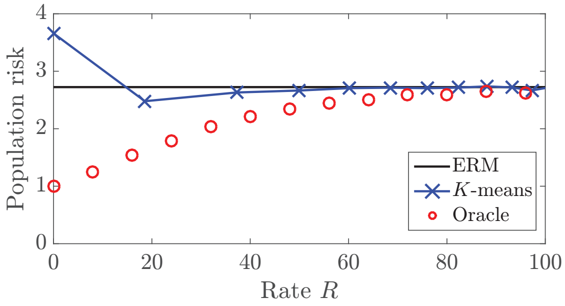

Figure 4: Comparison of the population risks achieved by different algorithms for linear regression.

In Figure 2, we plot the generalization error bound in Theorem 3 as a function of the rate R, and compare the generalization errors of the Oracle and K-means algorithms. It can be seen that Theorem 3 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 3 plots the upper bound on the distortion-rate function in Theorem 4 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 (Linder, Lugosi, and Zeger 1994).

We plot the population risks of the ERM solution W, the Oracle and K-means algorithms in Figure 4. 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.

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

In this section, we propose an improvement of the Hessian-weighted (HW) K-means clustering algorithm (Choi, El-Khamy, and Lee 2016) 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.

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:

$$\begin{aligned} &d_S(\hat{W}, W)\\ &\approx (\hat{W} - W)^T \nabla L_S(W) + \frac{1}{2}(\hat{W} - W)^T H_S(W)(\hat{W} - W), \end{aligned}$$

where $H_S(W)$ is the Hessian matrix. Assuming that W is a local minimum of $L_S(W)$ 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 (Choi, El-Khamy, and Lee 2016) partitions them into K disjoint clusters, using a set of cluster centers $c = \{c^{(1)}, \cdots, c^{(K)}\}$, and a cluster assignment $C = \big\{C^{(1)}, \cdots, C^{(K)}\big\}$, while solving the following optimization problem:

$$\min \sum_{k=1}^{K} \sum_{w^{(j)} \in C^{(k)}} h^{(j)} |w^{(j)} - c^{(k)}|^2. \tag{27}$$

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

$$\mathcal{L}_{S,W}(P_{\hat{W}|W}) = \sqrt{\frac{2\sigma^2}{n} I(W; \hat{W})} + \mathbb{E}[d_S(\hat{W}, W)]. \tag{28}$$

Here, we fix the number of clusters K 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, 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 Theorem 3, we have

$$\mathrm{gen}(\mu, P_{\hat{W}|S}) \leq 2\big(C(w^*)\|\Sigma_X\| + \sigma'^2\big) \sqrt{\frac{I(W; \hat{W})}{n}}, \tag{29}$$

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 (29) by

$$\beta \max_{k_1, k_2} |c^{(k_1)} - c^{(k_2)}|^2, \beta \geq 0, \tag{30}$$

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:

$$\min \sum_{k=1}^{K} \sum_{w^{(j)} \in C^{(k)}} h^{(j)} |w^{(j)} - c^{(k)}|^2 + \beta \max_{k_1, k_2} |c^{(k_1)} - c^{(k_2)}|^2. \tag{31}$$

An iterative algorithm to solve this optimization problem is provided in (Bu et al. 2019).

7 Experiments

In this section, we provide some real-world experiments to validate our theoretical assertions and the DRHW K-means algorithm.[2] Our experiments include compression of: (i) a three-layer fully connected network on MNIST; and (ii) a convolutional neural network with five conv layers and three linear layers on CIFAR10.[3]

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), 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 Figures 5 and 6, we compare DRHW K-means with HW K-means for different compression ratios on the MNIST

[2] All the codes of our experiments are available at the following link https://github.com/wgao9/weight-quant.

[3] We downloaded the pre-trained model in PyTorch from https://github.com/aaron-xichen/pytorch-playground.

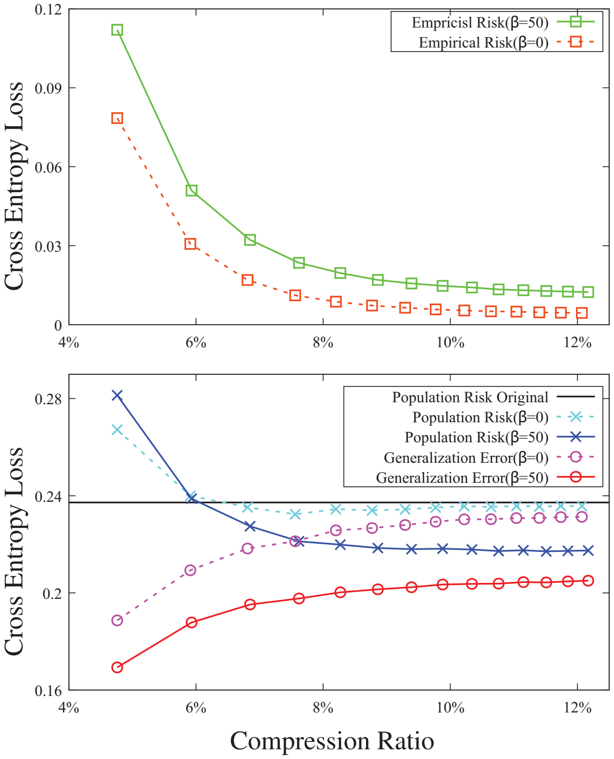

Figure 5: Comparison between the diameter regularized Hessian weighted K-means algorithm ($\beta = 50$) and the original one ($\beta = 0$) on MNIST. Top: empirical risks. Bottom: population risks and generalization errors.

and CIFAR10 datasets. Both figures demonstrate that the proposed quantization algorithm increases the empirical risk, but decreases the generalization error, and the net effect is that the proposed algorithm has a smaller population risk than the original model. More importantly, DRHW K-means algorithm has a better population risk than the HW K-means algorithm.

In Figure 7, we study how β affects the performance of our diameter-regularized Hessian-weighted K-means algorithm. It can be seen that as β increases, the generalization error decreases and 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 can be chosen via cross-validation in practice.

8 Conclusion

In this paper, we have provided an information-theoretical understanding of how model compression affects the population risk of a compressed model. We have shown that compression controls the tradeoff between generalization error and empirical risk. Our theoretical studies convey an important message for designing practical model compression algorithms, which is that we should consider the increase in empirical risk and the decrease in generalization error jointly, so as to achieve a smaller population risk.

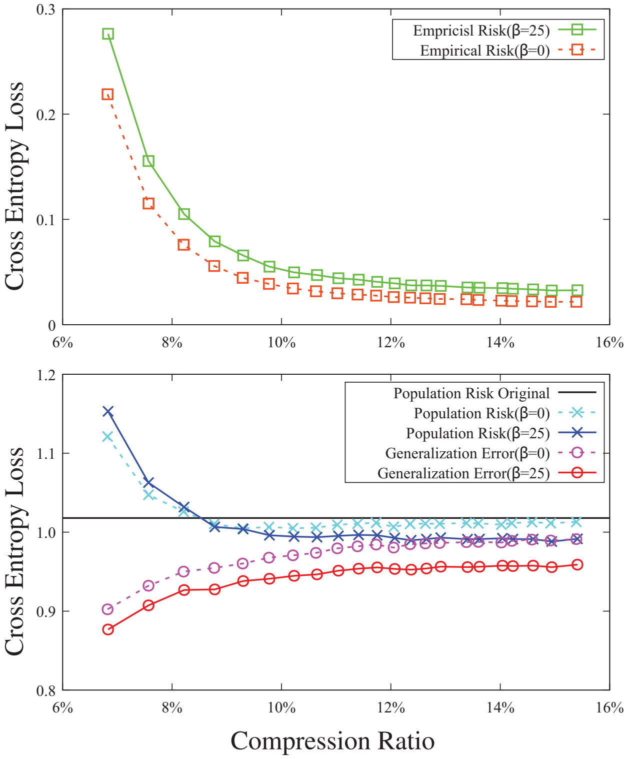

Figure 6: Comparison between the diameter regularized Hessian weighted K-means algorithm ($\beta = 25$) and the original one ($\beta = 0$) on CIFAR10. Top: empirical risks. Bottom: population risks and generalization errors.

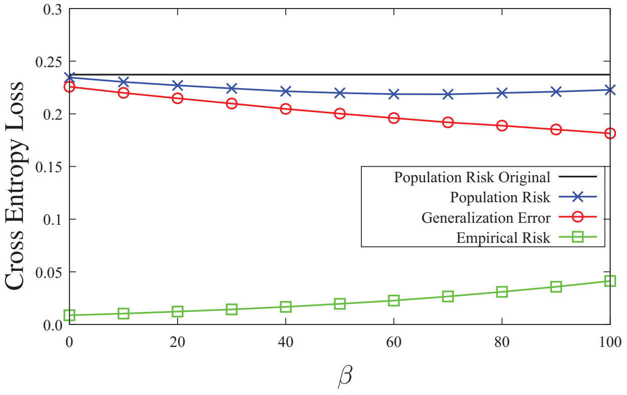

Figure 7: Diameter-regularized Hessian-weighted K-means with different β on the MNIST dataset with $K = 7$.

9 Acknowledgments

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

References

Bu, Y.; Gao, W.; Zou, S.; and Veeravalli, V. V. 2019. Information-theoretic understanding of population risk improvement with model compression. *arXiv preprint arXiv:1901.09421*.

Cheng, Y.; Wang, D.; Zhou, P.; and Zhang, T. 2017. A survey of model compression and acceleration for deep neural networks. *arXiv preprint arXiv:1710.09282*.

Choi, Y.; El-Khamy, M.; and Lee, J. 2016. Towards the limit of network quantization. *arXiv preprint arXiv:1612.01543*.

Courbariaux, M.; Bengio, Y.; and David, J.-P. 2015. Binaryconnect: Training deep neural networks with binary weights during propagations. In *Advances in neural information processing systems*, 3123–3131.

Cover, T. M., and Thomas, J. A. 2012. *Elements of information theory*. John Wiley & Sons.

Denton, E. L.; Zaremba, W.; Bruna, J.; LeCun, Y.; and Fergus, R. 2014. Exploiting linear structure within convolutional networks for efficient evaluation. In *Advances in neural information processing systems*, 1269–1277.

Gao, W.; Wang, C.; and Oh, S. 2018. Rate distortion for model compression: From theory to practice. *arXiv preprint arXiv:1810.06401*.

Gong, Y.; Liu, L.; Yang, M.; and Bourdev, L. 2014. Compressing deep convolutional networks using vector quantization. *arXiv preprint arXiv:1412.6115*.

Goodfellow, I.; Bengio, Y.; Courville, A.; and Bengio, Y. 2016. *Deep learning*, volume 1. MIT press Cambridge.

Guo, Y. 2018. A survey on methods and theories of quantized neural networks. *arXiv preprint arXiv:1808.04752*.

Gupta, S.; Agrawal, A.; Gopalakrishnan, K.; and Narayanan, P. 2015. Deep learning with limited numerical precision. In *International Conference on Machine Learning*, 1737–1746.

Han, S.; Pool, J.; Tran, J.; and Dally, W. 2015. Learning both weights and connections for efficient neural network. In *Advances in neural information processing systems*, 1135–1143.

Han, S.; Mao, H.; and Dally, W. J. 2015. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*.

Hassibi, B., and Stork, D. G. 1993. Second order derivatives for network pruning: Optimal brain surgeon. In *Advances in neural information processing systems*, 164–171.

Howard, A. G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; and Adam, H. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Krishnamoorthi, R. 2018. Quantizing deep convolutional networks for efficient inference: A whitepaper. *arXiv preprint arXiv:1806.08342*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.

LeCun, Y.; Denker, J. S.; and Solla, S. A. 1990. Optimal brain damage. In *Advances in neural information processing systems*, 598–605.

Lin, Y.; Han, S.; Mao, H.; Wang, Y.; and Dally, W. J. 2017. Deep gradient compression: Reducing the communication bandwidth for distributed training. *arXiv preprint arXiv:1712.01887*.

Linder, T.; Lugosi, G.; and Zeger, K. 1994. Rates of convergence in the source coding theorem, in empirical quantizer design, and in universal lossy source coding. *IEEE Transactions on Information Theory* 40(6):1728–1740.

Louizos, C.; Ullrich, K.; and Welling, M. 2017. Bayesian compression for deep learning. In *Advances in Neural Information Processing Systems*, 3288–3298.

Mozer, M. C., and Smolensky, P. 1989. Skeletonization: A technique for trimming the fat from a network via relevance assessment. In *Advances in neural information processing systems*, 107–115.

Novikov, A.; Podoprikhin, D.; Osokin, A.; and Vetrov, D. P. 2015. Tensorizing neural networks. In *Advances in Neural Information Processing Systems*, 442–450.

Raginsky, M.; Rakhlin, A.; Tsao, M.; Wu, Y.; and Xu, A. 2016. Information-theoretic analysis of stability and bias of learning algorithms. In *Proc. Information Theory Workshop (ITW)*, 26–30.

Rastegari, M.; Ordonez, V.; Redmon, J.; and Farhadi, A. 2016. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European Conference on Computer Vision*, 525–542. Springer.

Shannon, C. E. 1959. Coding theorems for a discrete source with a fidelity criterion. *IRE Nat. Conv. Rec* 4(142-163):1.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. 2017. Mastering the game of go without human knowledge. *Nature* 550(7676):354.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Tai, C.; Xiao, T.; Zhang, Y.; Wang, X.; et al. 2015. Convolutional neural networks with low-rank regularization. *arXiv preprint arXiv:1511.06067*.

Ullrich, K.; Meeds, E.; and Welling, M. 2017. Soft weight-sharing for neural network compression. *arXiv preprint arXiv:1702.04008*.

Vershynin, R. 2010. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*.

Xu, A., and Raginsky, M. 2017. Information-theoretic analysis of generalization capability of learning algorithms. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, 2524–2533.

Zhou, W.; Veitch, V.; Austern, M.; Adams, R. P.; and Orbanz, P. 2018. Non-vacuous generalization bounds at the imagenet scale: a pac-bayesian compression approach. *arXiv preprint arXiv:1804.05862*.

Zhu, C.; Han, S.; Mao, H.; and Dally, W. J. 2016. Trained ternary quantization. *arXiv preprint arXiv:1612.01064*.