COLD RAO-BLACKWELLIZED STRAIGHT-THROUGH GUMBEL-SOFTMAX GRADIENT ESTIMATOR

Anonymous authors
Paper under double-blind review

ABSTRACT

The problem of estimating the gradient of an expectation in discrete random variables arises in many applications: learning with discrete latent representations, training neural networks with quantized weights, activations, conditional blocks, etc. This work contributes to the development of the popular Gumbel-Softmax family of estimator, which is based on approximating argmax with a temperature-parametrized softmax. The state-of-the art in this family, the Gumbel-Rao estimator uses internal MC samples to reduce the variance. We show that in the limit of zero temperature the internal integration has a closed form solution. The limit estimator, called ZGR, has a favorable bias and variance, is simple to implement and computationally inexpensive and is obviously free of the temperature hyper-parameter. Furthermore, ZGR is unbiased for the class of quadratic functions of categorical variables and can be decomposed into a sum of two simple but not very well performing on their own estimators: the straight through estimator and the DARN estimator. Experiments thoroughly validate the method.

1 INTRODUCTION

Discrete variable or discrete structures are important in machine learning. For example, the semantic hashing idea (Salakhutdinov & Hinton, 2009) is to learn a binary vector representation such that semantically similar instances (*e.g.* images or text documents) would have similar bit representations. This allows for a quick retrieval via a lookup table or via nearest neighbor search with respect to the Hamming distance of binary encodings. Recent works employ variational autoencoders with binary latent states (Shen et al., 2018; Dadaneh et al., 2020; Ñanculef et al., 2020).

Another example is neural networks with discrete (binary or quantized) weights and activations. They allow for low-latency and energy-efficient test-time inference. Recent results indicate that quantization can achieve similar accuracy with a better efficiency in various applications (Nie et al., 2022). The training methods used are however often very empirical in nature. A sound training approach is to consider a stochastic relaxation, replacing all discrete weights and activations by discrete random variables, leading in the binary case to stochastic binary networks (Peters & Welling, 2018; Roth et al., 2019; Shekhovtsov & Yanush, 2021). A parameter-efficient stochastic relaxation for quantization was proposed by Louizos et al. (2019).

Other potential applications include conditional computation (Bengio et al., 2013; Yang et al., 2019; Bulat et al., 2021) reinforcement learning (Yin et al., 2019), learning task-specific tree structures for agglomerative neural networks (Choi et al., 2018), neural architecture search (Chang et al., 2019).

The learning problem in the presence of stochastic variables is usually formulated as minimization of the expected loss (or involves such expectation). The gradient-based optimization requires a gradient of the expectation in the probabilities of random variables (or parameters of networks inferring those probabilities). Unbiased gradient estimators have been developed (Williams, 1992; Grathwohl et al., 2018; Tucker et al., 2017; Gu et al., 2016; Yin et al., 2019), however they often require multiple forward passes of the model per one estimate and their variance can be still prohibitive.

Single-sample and low variance estimators are of high practical interest in applications. Gumbel Softmax (Jang et al., 2017) dubbed as Concrete relaxation (Maddison et al., 2017)[1] enable differ-

[1]These two works have contemporaneously introduced the same relaxation, but the name Gumbel Softmax (GS) became more popular in the literature.

entiability through discrete variables by relaxing them to real-valued variables with a distribution approximating the original discrete distribution. These estimators are biased, meaning that they are systematically off. The tightness of the relaxation is controlled by the temperature parameter $\tau > 0$ and the bias can be reduced by choosing a smaller temperature. However, the variance of the estimator grows as $O(1/\tau)$ (Shekhovtsov, 2021). The Gumbel-Rao (GR) estimator (Paulus et al., 2021) is a recent improvement of the popular Gumbel-Softmax Straight-Through (GS-ST) variant (Jang et al., 2017), which uses discrete samples on the forward pass but relaxed ones on the backward. GR can substantially reduce the variance of GS-ST. however it involves an intractable local integration in multiple variables, which is approximated by sampling. The experiments (Paulus et al., 2021) suggest that this estimator performs better at lower temperatures.

Inspired by the performance of GR at low temperatures, we analyze its behavior for temperatures close to and in the limit of the absolute zero. In the case of binary variables we give an asymptotic series expansion around $\tau = 0$ and show that the zero temperature limit of GR matches DARN, an already known estimator by Gregor et al. (2014). In the general categorical case, the limit of the multivariate integral is substantially more difficult, but is still analytically tractable. This zero temperature limit of GR leads to a new estimator called ZGR with a simple formula $\frac{1}{2}$(ST+DARN), where ST is the basic straight-through estimator. We show that ZGR is unbiased for all quadratic functions of categorical variables and experimentally show that it achieves a useful bias-variance tradeoff. We also contribute a refined experimental comparison of GS family with the SOTA unbiased ARSM estimator (Yin et al., 2019).

2 Background

Let x be a categorical random variable taking values in $\mathcal{K} = \{0, \dots K-1\}$ with probabilities $p(x;\eta)$ parametrized by η. Let $\phi(x) \in \mathbb{R}^d$ be an *embedding* of the discrete state x in the vector space. Categorical variables are usually represented using 1-hot embedding, in this case $d = K$. The value of x itself will still be used as an index. For binary variables and in quantization we will adopt the embedding $\phi(x) = x$, in this case $d = 1$.

Let $\mathcal{L}\colon \mathbb{R}^d \to \mathbb{R}$ be a differentiable loss function. We will denote the Jacobian of $\mathcal{L}$ with respect to an input ϕ as $J_\phi = \frac{\mathrm{d}\mathcal{L}}{\mathrm{d}\phi}$. It is the transposed gradient of $\mathcal{L}$ with respect to ϕ, nevertheless, we will informally refer to such Jacobians as gradients. For brevity, let us also use a shorthand $\mathcal{L}(x) = \mathcal{L}(\phi(x))$. The goal is to estimate the gradient of the expected loss

$$J_\eta = \tfrac{\mathrm{d}}{\mathrm{d}\eta}\mathbb{E}[\mathcal{L}(x)] = \tfrac{\mathrm{d}}{\mathrm{d}\eta}\textstyle\sum_x \mathcal{L}(x)p(x;\eta). \tag{1}$$

It has proven efficient in practice to consider *single-sample* estimators that take a single realization $x \sim p(x;\eta)$ and make an estimate of J_η based on the value $\mathcal{L}(x)$ and or the gradient $J_\phi = \frac{\mathrm{d}\mathcal{L}(\phi(x))}{\mathrm{d}\phi}$ at the sample x. Such estimators can be easily extended to losses in multiple discrete variables, *e.g.* defined by a stochastic computation graph, by simply applying the elementary estimator whenever the respective Jacobian is needed in backpropagation. In this case J_ϕ is the gradient in a specific variable at a joint sample.

Let us review the basic estimators with which we compare theoretically and experimentally and the GS family from which we derive.

Likelihood Ratio LR or REINFORCE (Williams, 1992) estimator is as follows. We rewrite

$$\tfrac{\mathrm{d}}{\mathrm{d}\eta}\textstyle\sum_x \mathcal{L}(x)p(x;\eta) = \sum_x \mathcal{L}(x)\frac{\mathrm{d}p(x;\eta)}{\mathrm{d}\eta} = \sum_x \mathcal{L}(x)p(x;\eta)\frac{\mathrm{d}\log p(x;\eta)}{\mathrm{d}\eta} = \mathbb{E}\big[\mathcal{L}(x)\frac{\mathrm{d}\log p(x;\eta)}{\mathrm{d}\eta}\big] \tag{2}$$

and define the estimate to be $J_\eta^{\mathrm{LR}} = \mathcal{L}(x)\frac{\mathrm{d}\log p(x;\eta)}{\mathrm{d}\eta}$, where $x \sim p(x;\eta)$. This estimator is clearly unbiased as $\mathbb{E}[J_\eta^{\mathrm{LR}}] = J_\eta$, but may have a high variance.

ST Let $\bar\phi(\eta) = \mathbb{E}[\phi(x)] = \sum_x \phi(x)p(x;\eta)$ – the *mean embedding* in $\mathbb{R}^d$ under the current distribution of x. The Straight-Through estimator is

$$J_\eta^{\mathrm{ST}} = J_\phi \frac{\mathrm{d}\bar\phi(\eta)}{\mathrm{d}\eta}. \tag{3}$$

Note, ST estimators of different empirical forms exist. The present definition for categorical variables is the same as, *e.g.*, by Gu et al. (2016) and is consistent with Hinton (2012) and Shekhovtsov & Yanush (2021) in the binary case.

DARN Gregor et al. (2014) use LR with a baseline $b(x) = \mathcal{L}(x) + J_\phi(\bar\phi - \phi(x))$, which is a first order Taylor approximation of $\mathcal{L}$ about $\phi(x)$ evaluated at some point $\bar\phi$. This results in the estimator

$$J_\eta^{\text{DARN}} = J_\phi(\phi(x) - \bar\phi)\tfrac{\mathrm{d}\log p(x;\eta)}{\mathrm{d}\eta}. \tag{4}$$

When x is binary, then the choice $\bar\phi = \frac{1}{2}\sum_x \phi(x)$ – the mean embedding under the uniform distribution, ensures that the estimator is unbiased for any quadratic function. However, for categorical variables no $\bar\phi$ with such property exist. Gu et al. (2016) have experimentally tested several choices for $\bar\phi$, including $\bar\phi = \bar\phi(\eta)$, and found that none performed well in the categorical case.

GS The Gumbel-Softmax (GS) estimator (Jang et al., 2017) is a relaxation of the Gumbel-argmax sampling scheme. Let $\theta_k = \log p(x{=}k;\eta)$. Let $G_k \sim \text{Gumbel}(0,1),\ k \in \mathcal{K}$, where $\text{Gumbel}(0,1)$ is Gumbel distribution with cdf $F(u) = e^{-e^u}$. Then

$$x = \arg\max_k(\theta_k + G_k) \tag{5}$$

is a sample from $p(x;\eta)$. The relaxation is obtained by using a $\mathrm{softmax}$ instead of $\arg\max$. This construction assumes one-hot embedding ϕ and creates relaxed (continuous) samples in the simplex Δ^K. Formally, introducing temperature hyperparameter τ, it can be written as

$$\tilde\phi = \mathrm{softmax}((\theta + G)/\tau) =: \mathrm{softmax}_\tau(\theta + G); \tag{6a}$$

$$J_\theta^{\text{ST-GS}} = \tfrac{\mathrm{d}\mathcal{L}(\tilde\phi)}{\mathrm{d}\theta} = \tfrac{\mathrm{d}\mathcal{L}(\tilde\phi)}{\mathrm{d}\tilde\phi}\tfrac{\mathrm{d}\tilde\phi}{\mathrm{d}\eta}. \tag{6b}$$

There are two practical concerns. First, the loss function is evaluated at a relaxed sample, which in a large computation graph can offset estimates of all other gradients, even not related to the discrete variable we relax. This effect can be mitigated by using a smaller temperature, causing $\tilde\phi$ to concentrate in a corner of the simplex. However, and this is the second concern, the variance of the estimator grows as $O(\frac{1}{\tau})$ if τ is decreased towards zero (Shekhovtsov, 2021).

GS-ST The Straight-Through Gumbel-Softmax (ST-GS) estimator (Jang et al., 2017) is an empirical modification of GS, addressing the first concern above. It uses discrete samples in the forward pass but swaps in the Jacobian of the continuous relaxation in the backward pass. The complete receipt is as follows

$$G_k \sim \text{Gumbel}(0,1),\ k \in \mathcal{K}; \tag{7a}$$

$$x = \arg\max_k(\theta_k + G_k); \tag{7b}$$

$$\tilde\phi = \mathrm{softmax}_\tau(\theta + G); \tag{7c}$$

$$J_\theta^{\text{ST-GS}} = \tfrac{\mathrm{d}\mathcal{L}(\phi(x))}{\mathrm{d}\phi}\tfrac{\mathrm{d}\tilde\phi}{\mathrm{d}\theta}. \tag{7d}$$

Notice that the hard sample $\phi(x)$ and the relaxed sample $\tilde\phi$ are entangled through G. Although, x has the law of $p(x;\eta)$ as desired, not biasing other variables, there is still bias in estimating the gradient in θ, which is typically larger than that of GS (see Fig. 1). To make the bias smaller the temperature τ should be decreased, however, the variance still grows as $O(1/\tau)$ (Shekhovtsov, 2021). Values of τ between 0.1 and 1 are used in practice (Jang et al., 2017). Notice also that the value of $\tilde\phi$ and of the Gumbel sample need not be stored in the forward pass, it can be re-sampled from the distribution of $G|x$, which is tractable (Maddison et al., 2017).

GR Finally, Paulus et al. (2021) proposed the Gumbel-Rao estimator, which decreases the variance of ST-GS estimator by computing via stochastic approximation the conditional expectation over G given x:

$$J_\theta^{\text{GR}} = \mathbb{E}_{G|x}\left[J_\theta^{\text{ST-GS}}(G)\right] = \tfrac{\mathrm{d}\mathcal{L}(\phi(x))}{\mathrm{d}\phi}\mathbb{E}_{G|x}\left[\tfrac{\mathrm{d}\tilde\phi}{\mathrm{d}\theta}\right]. \tag{8}$$

Because the value of the loss $\mathcal{L}(x)$ and its gradient do not depend on the specific realization of G, enabling the equality above, the expectation is localized and can be computed in the backward pass. However, this expectation is in multiple variables and is not analytically tractable. Paulus et al. (2021) use Monte Carlo integration with M samples from $G|x$. In their experiments they report improvement of the mean squared error of the estimator with τ decreasing from 1 down to 0.1, especially with $M = 1000$. The trend suggests that it would improve even further below $\tau = 0.1$ provided that the conditional expectation is approximated accurately enough. Unfortunately, the analytic form of this conditional expectation is not known even for Bernoulli variables and an accurate approximation requires many samples for low temperatures (Paulus et al., 2021).

3 METHOD

Given the experimental evidence about the GR estimator, we took the challenge to study its cold asymptotic behavior, *i.e.* for $\tau \to 0$. We first analyze the binary case, where derivations are substantially simpler. Proofs of all formal claims can be found in Appendix A.

3.1 BINARY CASE

In the case with two categories we can simplify the initial ST-GS estimator as follows. We assume $x \in \{0, 1\}$ and $\phi(x) = x$. The argmax trick can be expressed as $x = [\![\theta_1 + G_1 \geq \theta_0 + G_0]\!]$, where $[\![\cdot]\!]$ is the Iverson bracket. It is convenient to assume that the distribution of x is parametrized so that $p(x{=}1; \eta) = \sigma(\eta)$, the logistic sigmoid function. This is without loss of generality, because any other parametrization will result in just an extra deterministic Jacobian. Recalling that $\theta_k = \log p(x{=}\,k; \eta)$, we have $\theta_1 - \theta_0 = \eta$. Next, denoting $G_1 - G_0 = Z$, we can write the argmax trick compactly as $x = [\![\eta + Z \geq 0]\!]$. The difference of two Gumbel variables $Z = G_1 - G_0$ follows logistic distribution with cdf $\sigma(z)$. The GR estimator of gradient in η simplifies as

$$Z \sim \text{Logistic}; \quad x = [\![\eta + Z \geq 0]\!]; \quad \tilde{x} = \sigma_\tau(\eta + Z); \quad J_\eta^{\text{GR}} = \tfrac{\mathrm{d}\mathcal{L}(x)}{\mathrm{d}x} \mathbb{E}_{Z|x}\left[\tfrac{\mathrm{d}\tilde{x}}{\mathrm{d}\eta}\right], \tag{9}$$

where $\sigma_\tau(u) = \sigma(u/\tau)$ is the temperatured logistic sigmoid function. Although there is no closed form expression, we can compute the series expansion around $\tau = 0$.

Proposition 1.

$$J_\eta^{\text{GR}} = \tfrac{\mathcal{L}'(x)}{p(x)} p_Z(\eta)\Big(\tfrac{1}{2} + (2x-1)c_1 \log(2)\tau\Big) + O(\tau^2), \tag{10}$$

where p_Z is the logistic density and $c_1 = 2p - 1$.

Corollary 1. In the limit $\tau \to 0$ the GR estimator becomes the DARN estimator.

This establishes an interesting theoretical link, but discovers no new method for practical applications, as the DARN estimator is already known. Using the same expansion, we can study the asymptotic bias and variance of GR around $\tau = 0$.

Corollary 2. The mean and variance of the GR estimator (9) in the asymptote $\tau \to 0$ are:

$$\mathbb{E}[J_\eta^{\text{GR}}] = p(1-p)\Big(\tfrac{1}{2}(\mathcal{L}'(1) + \mathcal{L}'(0)) + (\mathcal{L}'(1) - \mathcal{L}'(0))\tilde{c}_1\tau\Big) + O(\tau^2), \tag{11a}$$

$$\mathbb{V}[J_\eta^{\text{GR}}] = (p(1-p))^3\Big(\tfrac{1}{4}(a-b)^2 + \tfrac{1}{2}(a^2 - b^2)\tilde{c}_1\tau\Big) + O(\tau^2), \tag{11b}$$

where $p = \sigma(\eta)$, $a = \frac{\mathcal{L}'(1)}{p}$, $b = \frac{\mathcal{L}'(0)}{1-p}$ and $\tilde{c}_1 = (2p-1)\log(2)$.

This allows to make some predictions, in particular in the case of a linear objective $\mathcal{L}$. In this case the bias is $O(\tau^2)$ and the squared bias is $O(\tau^4)$. Therefore the MSE is determined by the variance alone up to $O(\tau^4)$. The dependence of variance on τ for a linear objective is negative in the first order term. Therefore the temperature corresponding to the minimum MSE will be non-zero.

3.2 GENERAL CATEGORICAL CASE

In the general categorical case, the analysis is more complicated but gives a novel result:

Theorem 1 (ZGR)**.** *The Gumbel-Rao estimator for one-hot embedding ϕ in the limit of zero temperature is given by*

$$J^{ZGR}_{\theta_i} = \begin{cases} \frac{1}{2}(J_{\phi_i} - J_{\phi_x})p(x{=}i;\eta) & \text{if } i \neq x; \\ -\frac{1}{2}\sum_{j\neq x}(J_{\phi_j} - J_{\phi_x})p(x{=}j;\eta) & \text{if } i = x. \end{cases} \tag{12}$$

Proposition 2. ZGR estimator decomposes as

$$\boxed{J^{\text{ZGR}} = \tfrac{1}{2}(J^{\text{ST}} + J^{\text{DARN}}),} \tag{13}$$

with the choice $\bar{\phi} = \bar{\phi}(\eta)$ in DARN.

As we have given ST and DARN estimators for a general embedding, we claim that the form (2) is the correct expression of ZGR for any embedding (since a change of the embedding is just a linear transform). This form shows a surprising connection. Neither ST nor DARN estimators perform particularly well in categorical VAEs on their own (Gu et al., 2016; Paulus et al., 2021). However ZGR, being effectively their sum, appears superior. It has the following property, up to date unique amongst single-sample categorical estimators:

Theorem 2. *ZGR is unbiased for quadratic loss functions $\mathcal{L}$.*

By quadratic functions we understand quadratic functions $\mathbb{R}^d \to \mathbb{R}$ with arbitrary coefficients, *i.e.*, including all linear functions. Because the bias of ST-GS, GR and GR-MC is the same and we have shown that ZGR is the limit of GR at $\tau \to 0$ there following is straightforward.

Corollary 3. ST-GS and GR are asymptotically unbiased for quadratic functions with $\tau \to 0$.

These results extend to multiple independent discrete variables as follows.

Corollary 4. Let $x_1, \ldots, x_n$ be independent categorical variables and $\mathcal{L}(x_1, \ldots, x_n)$ be such that for all i and all configurations x the restriction $\mathcal{L}(x_i)$ is a quadratic function. Then ZGR is unbiased.

The unbiased property for quadratic function gives us some intuition about applicability limits of ZGR. Namely, if the loss function is reasonably smooth, such that it can be approximated well by a quadratic function, we expect gradient estimates to be accurate. Compared to ST, which is unbiased for multilinear functions only, we hypothesize that ZGR can capture interactions more accurately.

4 Experiments

We will compare ZGR with Gumbel-Softmax (**GS**), Straight-through Gumbel-Softmax (**GS-ST**) (Jang et al., 2017)[2], Gumbel-Rao with MC samples (**GR**) (Paulus et al., 2021)[3], the state-of-the art unbiased estimator **ARSM** (Yin et al., 2019)[4] and the basic (and very popular) **ST** estimator (3).

4.1 Variational Autoencoder

A common benchmark used in the literature for evaluating the quality of discrete latent variable models has been the stochastically binarized MNIST data (Salakhutdinov & Murray, 2008). Unfortunately this exact dataset is no longer available. In or experiments, following Yin et al. (2019), we binarize MNIST with a deterministic threshold of 0.5.

4.1.1 Zero Temperature Limit and Estimation Accuracy in VAE

First, we measure the gradient estimation accuracy at a particular point of VAE training, comparing GS family of estimators at different temperatures as in (Paulus et al., 2021, Fig 2b.) and verify whether ZGR is indeed the zero temperature limit of GR estimator. The referenced plot of Paulus et al. shows a steady decrease of MSE with the decrease of temperature down to 0.1 and we expect ZGR to achieve the lowest MSE.

[2]Pytorch function `torch.functional.gumbel_softmax`

[3]Public reimplementation https://github.com/nshepperd/gumbel-rao-pytorch

[4]Our reimplementation cross-checked with https://github.com/ARM-gradient/ARSM

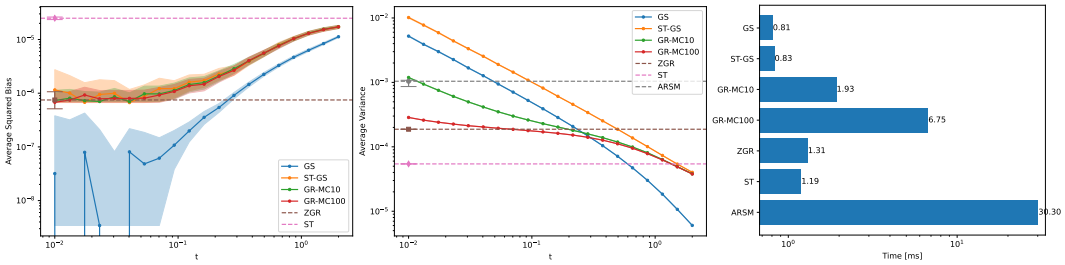

Figure 1: Gradient estimation accuracy and computation cost in VAE on MNIST. *Left* and *middle*: average (per parameter) squared bias and variance of gradient estimators versus temperature. ST-GS and all GR estimators share the same bias according to the theory but differ in variance. ZGR achieves the lowest bias of this family and has variance lower than GR and GS estimators for any temperature < 0.1 and substantially lower than that of unbiased ARSM. GS estimator is asymptotically unbiased but its variance grows asymptotically as $O(1/\tau)$. *Right:* time of a forward-backward pass on GPU. GS and GS-ST estimators are built-in in pytorch while other estimators are implemented by us in pytorch. ZGR is significantly faster than GR-MC variants and ARSM. Note that the network has only a single layer with 16 8-way categorical variables.

As an evaluation point we take a VAE model after 100 epochs of training with ARSM. We then estimate bias and variance of all gradient estimators as follows. The loss function is the average ELBO of a randomly fixed mini-batch of size 200. Let X be the reference unbiased estimator (ARSM) and Y be the tested estimator. We want to measure the average over parameters squared bias, which can be written as $b^2 = \frac{1}{d}\|\mathbb{E}[X] - \mathbb{E}[Y]\|^2$. We obtain $n_1 = 10000$ independent samples $X_i \in \mathbb{R}^d$ from ARSM and $n_2 = 10000$ independent samples $Y_i \in \mathbb{R}^d$ from the tested estimator and measure an unbiased estimate of b^2:

$$\widehat{b^2} = \frac{1}{d}\|\hat{\mu}_1 - \hat{\mu}_2\|^2 - \frac{V_1}{n_1} - \frac{V_2}{n_2}, \tag{14}$$

where $\hat{\mu}_1$ is the sample mean of X and V_1 is the average (over parameters) sample variance of X and $\hat{\mu}_2$ and V_2 are likewise for Y. The variance of the evaluated estimator is just V_2.

The results are shown in Fig. 1. The confidence intervals are 95% empirical intervals of 100 bootstrap samples of the estimator $\widehat{b^2}$. Since the estimate is unbiased, the result is sometimes negative and is cropped in the figure.

We observe that the variances are by several orders larger (at best) than the squared bias. Therefore the mean squared error, which is the sum of the variance and the squared bias, is dominated by the variance alone for all estimators. This is in high contradiction with (Paulus et al., 2021, Fig 2b.), in particular meaning that the optimal operation point of GR estimators in terms of MSE is going to be at high temperatures, however many MC samples we use. This also means that ZGR is not improving the MSE as we expected. However, it still fulfills our expectations of the zero limit estimator: it has the limiting bias, which is the lowest, and the limiting variance which is moderate. The timing plot in Fig. 1 verifies that ZGR is computationally inexpensive.

Note that the effect of the bias-variance trade-off in learning is not straightforward: common optimization methods use momentum as an effective way of variance reduction while the bias can be correlated across iterations and may potentially accumulate.

4.1.2 Training Performance in VAE

Yin et al. (2019) have shown that their unbiased ARSM substantially outperforms other unbiased estimators as well as GS and GS-ST on MNIST VAE. At the same time, Paulus et al. (2021) have shown that their GR-MC also substantially outperforms GS and GS-ST on the same task. We will now compare all estimators with ZGR under a constant setup detailed in Appendix B. The results are presented in Fig. 2. The variant ZGR $s = 2$ uses two forward passes as a variance reduction.

In this experiment, ARSM was found to performs not better than the GS family, in contradiction with Yin et al. (2019). We believe the reason is that they use temperature $\tau = 1$ and attempt to learn it as parameter instead of treating it as a hyper-parameter. We imply therefore that the variance of ARSM is still too high for practical applications even in cases where it is computationally feasible.

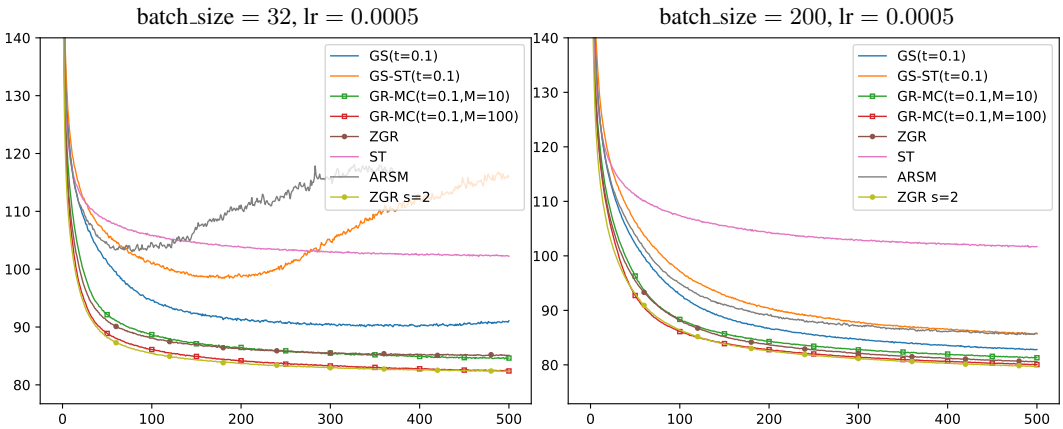

Figure 2: MNIST VAE. Training loss versus epochs with batch size 32 (*left*) and 200 (*right*), the rest of the setup held fixed. ZGR method performs not worse than the more computationally demanding Gumbel-Rao with 10 or 100 MC samples. For small batch size, the performance of ARSM, GS and GS-ST is clearly affected by their higher variance. The effect is consistent with the comparison of variance in Fig. 1 at t=0.1. The performance of ST, which has the least variance, is affected by its bias for any batch size.

GR methods perform better than GS and GS-ST in the case of a small batch size and a relatively large learning rate. This agrees with the experimental findings of Paulus et al. (2021).

Finally, we observe that the proposed ZGR performs not worse than GR-MC with $K = 10$ or $K = 100$ MC samples, which are more computationally costly. Its performance appears to be still bounded by the variance, as using the two-sample estimate (ZRG $s = 2$) leads to an improvement.

4.2 Training Quantized Neural Networks

We consider the stochastic relaxation for quantization based on the relaxed quantization model of Louizos et al. (2019). In this model the distribution of a quantized weight or activation x is defined by a single real-valued input η via: $x = \lfloor \eta + z \rceil_K$, where $\lfloor \cdot \rceil$ rounds to the nearest integer in the range $0 \dots K-1$ and z is an injected noise, such as logistic noise.

4.2.1 Single Unit Quantization

We first evaluate bias and variance of all estimators on a simple function of a single quantized variable. Let η be a real-valued parameter. Let $p(x;\eta)$ be given by the stochastic quantization model with $K = 4$ states and a particular noise type. Given a test function $\mathcal{L}(x)$ we can compute the true gradient of $\mathbb{E}[\mathcal{L}(x)]$. For each estimator we draw 10^4 samples to compute its mean and standard deviation for each value of η. The results are presented in Fig. 3. In this plot we show several combinations of loss functions and noises. The test functions are: *linear* $\mathcal{L}(x) = x$; *quadratic* $\mathcal{L}(x) = \frac{1}{2}(x-c)^2$ and *sigmoid* $\mathcal{L}(x) = \sigma(2(x-c))$, where $c = (K-1)/2$ is chosen for centering. Note, this significantly extends the toy experiment (Yin et al., 2019, Fig.1), where only a linear objective was tested. The noises shown refer to the *logistic* noise with std = 1/3 as used at initialization by Louizos et al. 2019 and the *triangular* noise with the density $p(z) = \max(0, 1-|z|)$.

The bias of the GS family quickly decreases with the temperature. ZRG estimator achieves the same expected value as GS-ST in the limit of small temperature illustrated by GS-ST(t=0.1) and the variance comparable to that of GR(t=0.1, M=100). We also verify that ZGR has zero bias for quadratic objectives as we have shown theoretically.

4.2.2 Deep Quantized Networks

In this experiment we train a convolutional network, closely following Louizos et al. (2019), for MNIST. We do not quantize the input (it has 8 bit resolution in the dataset), the first and last weight matrices are quantized to 4 bits. All inner layer weights and activations are quantized to

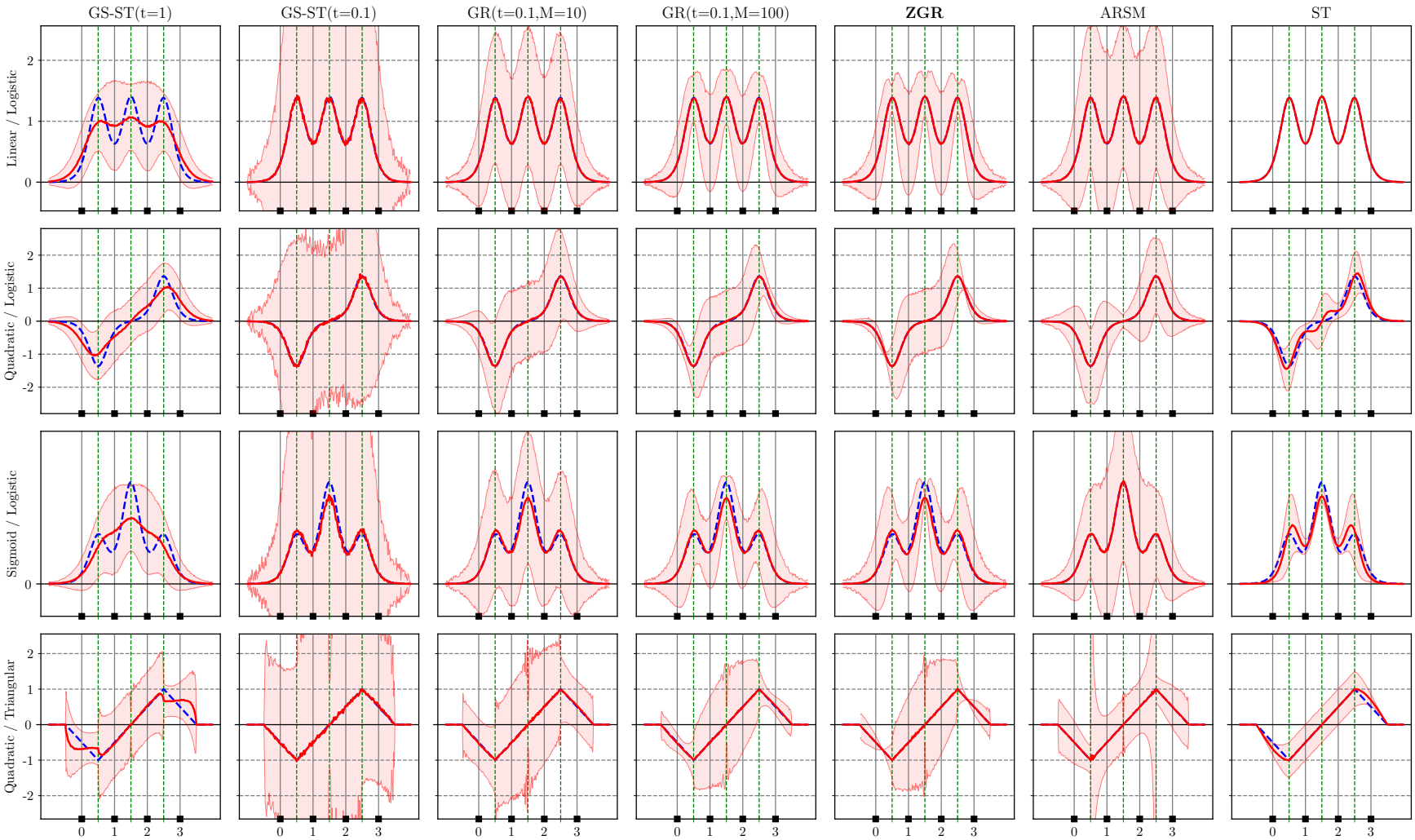

Figure 3: Performance of estimators in stochastic quantization with selected combinations of test functions and injected noise for varied input η of the stochastic quantizer. The dashed blue line is the exact gradient. Red line is the mean of the estimator and the red shaded area shows ± 1 std. GR with 100 is able to reduce the variance of GS-ST substantially. ZGR reduces the variance by an edge further while keeping the bias equal to the theoretical bias of GR with zero temperature. The variance of ARSM is substantially higher. Plain ST estimator has a yet smaller variance but a larger bias, which may accumulate during taring.

Table 1: MNIST test error$[\%]$ in deterministic mode (no injected noises at test time) for different bit-width per weigh and activation (T denotes ternary). Hyperparameters are selected on the validation set. Reference test errors: ReLU 0.69% , Clamp 0.64%.

Logistic Noise Relaxation

	Weights [bits] / Activations [bits]			
method	2/2	T/T	T/1	1/1
GS(t=2)	0.80	0.78	0.87	1.11
GS(t=1)	0.86	0.72	0.98	0.86
GS(t=0.5)	0.94	0.82	0.99	1.18
GS-ST(t=2)	**0.63**	0.75	0.70	**0.53**
GS-ST(t=1)	0.71	0.76	0.82	0.60
GS-ST(t=0.5)	0.82	0.82	0.88	1.16
GR(t=0.5,K=10)	0.70	0.79	0.85	0.99
GR(t=0.1,K=10)	0.80	1.02	0.86	1.04
ZGR	0.69	0.89	0.97	0.99
ST	0.74	**0.62**	**0.68**	0.78

Triangular Noise Relaxation

	Weights [bits] / Activations [bits]			
method	2/2	T/T	T/1	1/1
GR(t=0.5,K=10)	0.57	0.55	**0.52**	0.69
GR(t=0.1,K=10)	0.65	0.58	0.61	0.96
ZGR	**0.55**	0.66	0.59	0.91
ST	0.62	**0.54**	0.65	**0.66**

2 bits or below: T - ternary, 1 - binary. Each activation quantization is preceded by batch normalization (Ioffe & Szegedy, 2015). We form two real-valued baselines applying ReLU or Clamp ($x \mapsto \min(\max(x, 0), K - 1)$) as activations instead of quantization. We expect that with enough bits, quantized training should achieve performance of at least the Clamp variant. ARSM is not included in this experiment because its complexity grows quadratically with both: the number of categories and the number of discrete stochastic layers. Further detail are specified in Appendix B.

The results are presented in Table 1. First, we note that our results with 2/2 bits are very close to (Louizos et al., 2019, Table 1), we however achieve good results even for lower bit-widths. We see that the best results for logistic noise relaxation, as in Louizos et al. (2019), are obtained with estimators having the least variance, prominently ST and GS-ST(t=2). It appears that the bias is not important or not harmful in this application. With each step decreasing the bias but increasing the variance, *i.e.*, when decreasing the temperature with any method of the GS family, the results tend to worsen.

We additionally tested the triangular noise and observed an improvement in results for all low bias methods. The gap between ST and ZGR, GR-MC closes, especially so in cases with more quantization levels (2/2, T/T, T/1). The variance at low quantization levels (esp. 1/1) is still detrimental. In both relaxations, our limit ZGR estimator is comparable but more efficient than GR-MC with $t = 0.1$, or, based on the trends in Fig. 1, we could say it is better than GR with $t \leq 0.1$.

5 Conclusion

We obtained a new single-sample gradient estimator ZGR, which is extremely simple, more computationally efficient than GR-MC, has no hyperparameter, and experimentally achieves the lowest bias possible with GS-ST or GR-MC family of estimators. The variance is moderate, experimentally lower than that of the best unbiased ARSM estimator (which is also significantly more costly). In VAE, we found that ZGR is a viable replacement of GR-MC, achieving the best bias-variance tradeoff. By cooling down the GR-MC estimator, ZGR helps to keep the GPUs cool :) In training quantized networks we have found that the bias of all estimators, including ST and GS at high temperatures $\tau = 1, 2$, is of no importance and the estimators with the least variance perform the best. The picture may reverse though if a different learning formulation is considered, *e.g.* Bayesian learning of weights or learning with the multi-sample objective such as in Raiko et al. (2015). We hope that our theoretical and experimental findings will be insightful and inspiring for the future research.

Reproducibility Statement

Proofs of all formal claims are presented in Appendix A. Details of the experiments are described in Appendix B. The source code of our implementation will be made publicly available upon pub-

lication. During the review period, we will be happy to answer questions and share the code with reviewers confidentially through the OpenReview platform.

REFERENCES

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

Adrian Bulat, Brais Martinez, and Georgios Tzimiropoulos. High-capacity expert binary networks. In *ICLR*, 2021.

Jianlong Chang, xinbang zhang, Yiwen Guo, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. DATA: Differentiable architecture approximation. In *NeurIPS*, volume 32, 2019.

Jihun Choi, Kang Min Yoo, and Sang-goo Lee. Learning to compose task-specific tree structures. In *AAAI*, 2018. ISBN 978-1-57735-800-8.

Siamak Zamani Dadaneh, Shahin Boluki, Mingzhang Yin, Mingyuan Zhou, and Xiaoning Qian. Pairwise supervised hashing with Bernoulli variational auto-encoder and self-control gradient estimator. *ArXiv*, abs/2005.10477, 2020.

Will Grathwohl, Dami Choi, Yuhuai Wu, Geoff Roeder, and David Duvenaud. Backpropagation through the void: Optimizing control variates for black-box gradient estimation. In *ICLR*, 2018.

Karol Gregor, Ivo Danihelka, Andriy Mnih, Charles Blundell, and Daan Wierstra. Deep autoregressive networks. In *ICML*, 2014.

Shixiang Gu, Sergey Levine, Ilya Sutskever, and Andriy Mnih. Muprop: Unbiased backpropagation for stochastic neural networks. In *ICLR*, May 2016.

Geoffrey Hinton. Lecture 15d - Semantic hashing : 3:05 - 3:35, 2012.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, volume 37, pp. 448–456, 2015.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with Gumbel-softmax. In *ICLR*, 2017.

Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. *CoRR*, abs/1312.6114, 2013.

Christos Louizos, Matthias Reisser, Tijmen Blankevoort, Efstratios Gavves, and Max Welling. Relaxed quantization for discretized neural networks. In *ICLR*, 2019.

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *ICLR*, 2017.

Henrick J. Malik and Bovas Abraham. Multivariate logistic distributions. *The Annals of Statistics*, 1(3):588–590, 1973.

Ricardo Ñanculef, Francisco Alejandro Mena, Antonio Macaluso, Stefano Lodi, and Claudio Sartori. Self-supervised bernoulli autoencoders for semi-supervised hashing. *CoRR*, abs/2007.08799, 2020.

Guhong Nie, Lirui Xiao, Menglong Zhu, Dongliang Chu, Yue Shen, Peng Li, Kang Yang, Li Du, and Bo Chen. Binary neural networks as a general-propose compute paradigm for on-device computer vision, 2022. URL https://arxiv.org/abs/2202.03716.

Max B Paulus, Chris J. Maddison, and Andreas Krause. Rao-Blackwellizing the straight-through Gumbel-softmax gradient estimator. In *ICLR*, 2021.

Jorn WT Peters and Max Welling. Probabilistic binary neural networks. *arXiv preprint arXiv:1809.03368*, 2018.

Tapani Raiko, Mathias Berglund, Guillaume Alain, and Laurent Dinh. Techniques for learning binary stochastic feedforward neural networks. In *ICLR*, 2015.

Wolfgang Roth, Günther Schindler, Holger Fröning, and Franz Pernkopf. Training discrete-valued neural networks with sign activations using weight distributions. 2019.

Ruslan Salakhutdinov and Geoffrey Hinton. Semantic hashing. *Int. J. Approx. Reasoning*, 50(7):969–978, 2009.

Ruslan Salakhutdinov and Iain Murray. On the quantitative analysis of deep belief networks. In *ICML*, pp. 872–879, 2008.

Alexander Shekhovtsov. Bias-variance tradeoffs in single-sample binary gradient estimators. In *GCPR*, 2021.

Alexander Shekhovtsov and Viktor Yanush. Reintroducing straight-through estimators as principled methods for stochastic binary networks. In *GCPR*, 2021.

Dinghan Shen, Qinliang Su, Paidamoyo Chapfuwa, Wenlin Wang, Guoyin Wang, Ricardo Henao, and Lawrence Carin. NASH: Toward end-to-end neural architecture for generative semantic hashing. In *Annual Meeting of the Association for Computational Linguistics*, 2018.

George Tucker, Andriy Mnih, Chris J Maddison, John Lawson, and Jascha Sohl-Dickstein. REBAR: Low-variance, unbiased gradient estimates for discrete latent variable models. In *NeurIPS*, 2017.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3):229–256, May 1992.

Inc. Wolfram Research. Mathematica, Version 13.1, 2021. URL https://www.wolfram.com/mathematica. Champaign, IL, 2022.

Brandon Yang, Gabriel Bender, Quoc V Le, and Jiquan Ngiam. CondConv: Conditionally parameterized convolutions for efficient inference. In *NeurIPS*, volume 32, 2019.

Mingzhang Yin, Yuguang Yue, and Mingyuan Zhou. ARSM: augment-REINFORCE-swap-merge estimator for gradient backpropagation through categorical variables. In *ICML*, volume 97, pp. 7095–7104, 2019.

A PROOFS

A.1 BINARY CASE

Proposition 1.

$$J_\eta^{\mathrm{GR}} = \tfrac{\mathcal{L}'(x)}{p(x)} p_Z(\eta)\Big(\tfrac{1}{2} + (2x-1)c_1 \log(2)\tau\Big) + O(\tau^2), \tag{10}$$

where p_Z is the logistic density and $c_1 = 2p - 1$.

Proof. The conditional density $p(z|x)$ is

$$p(z|x) = \begin{cases} p_Z(z)[\![\eta + z \geq 0]\!]/p(x{=}1), & \text{if } x = 1; \\ p_Z(z)[\![\eta + z < 0]\!]/p(x{=}0), & \text{if } x = 0. \end{cases} \tag{15}$$

Let us denote $p(x{=}1)$ as just p. The GR estimator expands as

$$J_\eta^{\mathrm{GR}} = \begin{cases} \frac{\mathcal{L}'(1)}{p} \int_{-\infty}^{\eta} \sigma'_\tau(\eta + z) p(z) \mathrm{d}z, & \text{if } x = 1; \\ \frac{\mathcal{L}'(0)}{1-p} \int_{\eta}^{\infty} \sigma'_\tau(\eta + z) p(z) \mathrm{d}z, & \text{if } x = 0. \end{cases} \tag{16}$$

Using the change of variables $v = \sigma_\tau(\eta + z)$, with the inverse $z = \tau \mathrm{logit}(v) - \eta$, we have

$$\mathrm{d}v = \sigma'_\tau(\eta + z)\mathrm{d}z \tag{17}$$

and can write the estimator as

$$J_\eta^{\mathrm{GR}} = \tfrac{\mathcal{L}'(x)}{p(x)} \Big(x \int_{\frac{1}{2}}^{1} p_Z(\eta - \tau \mathrm{logit} v)\mathrm{d}v + (1-x) \int_0^{\frac{1}{2}} p_Z(\eta - \tau \mathrm{logit} v)\mathrm{d}v \Big). \tag{18}$$

Note that $p_Z(\eta - \tau \mathrm{logit} v)$ is bounded above by a constant $\sup_z p_Z(z) = \frac{1}{4}$. A constant is integrable on $[0, 1]$. By dominated convergence theorem we can take the limits $\tau \to 0$ under the integral. In particular we can use

$$\lim_{\tau \to 0} p_Z(\eta - \tau \mathrm{logit}(v)) = \tfrac{1}{2} p_Z(\eta) \tag{19}$$

under the integral. In order to get a more detailed view, we make the Taylor series expansion of $p_Z(\eta - \tau \mathrm{logit}(v))$ and substitute it under the integral. With the help of Mathematica (Wolfram Research, 2021) we obtain:

$$J_\eta^{\mathrm{GR}} = \tfrac{\mathcal{L}'(x)}{p(x)} p_Z(\eta)\Big(\tfrac{1}{2} + (2x-1)c_1 \log(2)\tau + c_2 \tfrac{\pi^2}{6}\tau^2\Big) + O(\tau^3), \tag{20}$$

where $c_1 = \tanh(\eta/2) = 2p - 1$ and $c_2 = \frac{1}{2}(1 - 3/(\cosh(\eta) + 1))$. □

Corollary 1. In the limit $\tau \to 0$ the GR estimator becomes the DARN estimator.

Proof. From the series expansion, the limit $\tau- \to 0$ is

$$J_\eta^{\mathrm{GR}} = \tfrac{1}{2} \tfrac{\mathcal{L}'(x)}{p(x)} p_Z(\eta) = \tfrac{1}{2} \tfrac{\mathcal{L}'(x)}{p(x)} p(1-p), \tag{21}$$

where $p = \sigma(\eta)$. It remains to show that it matches DARN as defined in (4). Note that $\frac{\mathrm{d} \log p(x{=}1;\eta)}{\mathrm{d}\eta} = \sigma(\eta)(1 - \sigma(\eta)) = p(1-p)$ and $\frac{\mathrm{d} \log p(x{=}0;\eta)}{\mathrm{d}\eta} = -p(1-p)$. By expanding the cases for $x = 1$ and $x = 0$ we verify that

$$(x - \bar{x}) \tfrac{\mathrm{d} \log p(x{=}1;\eta)}{\mathrm{d}\eta} = \tfrac{1}{2} p(1-p), \tag{22}$$

where $\bar{x} = \frac{1}{2}$. □

Corollary 2. The mean and variance of the GR estimator (9) in the asymptote $\tau \to 0$ are:

$$\mathbb{E}[J_\eta^{\mathrm{GR}}] = p(1-p)\Big(\tfrac{1}{2}(\mathcal{L}'(1) + \mathcal{L}'(0)) + (\mathcal{L}'(1) - \mathcal{L}'(0))\tilde{c}_1 \tau\Big) + O(\tau^2), \tag{11a}$$

$$\mathbb{V}[J_\eta^{\mathrm{GR}}] = (p(1-p))^3 \Big(\tfrac{1}{4}(a-b)^2 + \tfrac{1}{2}(a^2 - b^2)\tilde{c}_1 \tau\Big) + O(\tau^2), \tag{11b}$$

where $p = \sigma(\eta)$, $a = \frac{\mathcal{L}'(1)}{p}$, $b = \frac{\mathcal{L}'(0)}{1-p}$ and $\tilde{c}_1 = (2p - 1)\log(2)$.

Proof. The mean of the estimator is computed from the series expansion up to the first order as

$$\begin{aligned}
&p\tfrac{\mathcal{L}'(1)}{p}p_Z(\eta)\Big(\tfrac{1}{2}+\tilde{c}_1\tau\Big)+O(\tau^2)\\
&+(1-p)\tfrac{\mathcal{L}'(0)}{1-p}p_Z(\eta)\Big(\tfrac{1}{2}+-1\tilde{c}_1\tau\Big)+O(\tau^2)\\
&=p_Z(\eta)\Big(\tfrac{1}{2}(\mathcal{L}'(1)+\mathcal{L}'(1))+\tilde{c}_1(\mathcal{L}'(1)-\mathcal{L}'(0))\tau\Big)+O(\tau^2). \qquad (23)
\end{aligned}$$

Since the GR estimator $J_\eta^{\mathrm{GR}}(x)$ is a Bernoulli variable with values $J_\eta^{\mathrm{GR}}(0)$ and $J_\eta^{\mathrm{GR}}(1)$ with probabilities p and $1-p$, respectively, we can compute is variance simply as

$$\big(J_\eta^{\mathrm{GR}}(1)-J_\eta^{\mathrm{GR}}(0)\big)^2 p(1-p). \qquad (24)$$

Using that $p_Z(\eta)=p(1-p)$, the asymptotic expansion of variance up to first order in τ is

$$(p(1-p))^3\Big(\big(a(\tfrac{1}{2}+\tilde{c}_1\tau)-b(\tfrac{1}{2}-\tilde{c}_1\tau)\big)\Big)^2+O(\tau^2) \qquad (25a)$$

$$=(p(1-p))^3\Big(\tfrac{1}{2}(a-b)+(a+b)\tilde{c}_1\tau\Big)^2+O(\tau^2), \qquad (25b)$$

where $a=\frac{f'(1)}{p}$, $b=\frac{f'(0)}{1-p}$, $\tilde{c}_1=\log(2)c_1$. The first order term is

$$\tfrac{1}{2}(p(1-p))^3(a^2-b^2)\tilde{c}_1\tau. \qquad (26)$$

It could be positive or negative depending on the values of the derivatives and of p. Let us expand a,b and $c_1=\tanh(\eta/2)=2p-1$. We obtain, up to positive constants,

$$p(1-p)(f'(1)^2(1-p)^2-f'(0)^2p^2)(2p-1)\tau. \qquad (27)$$

We see that for the corner points, where p approaches either 0 or 1, this linear term is negative. In particular for a linear objective we have $f'(1)=f'(0)$ and the linear term becomes

$$-p(1-p)f'(1)^2(2p-1)^2\tau, \qquad (28)$$

which is non-positive for any p and is zero for $p=\frac{1}{2}$. □

A.2 General Categorical Case

This case is significantly more difficult, as we are dealing with multivariate integration in K Gumbel variables. We will make use of the following statistical relationship.

Lemma A.1. *Let $G_1,\dots G_K$ be independent standard Gumbel random variables. Then Z with components $Z_i=G_i-G_K$ for $i=1\dots K-1$ has the multivariate logistic distribution (Malik & Abraham, 1973) with cdf*

$$F_Z(z)=\tfrac{1}{1+\sum_{i=1}^{K-1}e^{-z_i}}. \qquad (29)$$

Proof. The cdf and density of Gumbel$(0,1)$ distribution are given respectively by

$$F_G(x)=e^{-e^{-x}};\quad p_G(x)=e^{-(x+e^{-x})}. \qquad (30)$$

The conditional distribution of Z_i given G_K is $F_{Z_i|G_K}(z_i|y)=e^{-e^{-(z_i+y)}}$. The conditional joint distribution of Z given G_K is respectively

$$F_{Z|G_K}(z|y)=\prod_{i=1}^{K-1}e^{-e^{-(z_i+y)}}=\exp(\sum_{i=1}^{K-1}-e^{-(z_i+y)})=\exp(-e^{-y}\sum_{i=1}^{K-1}e^{-z_i})=e^{-e^{-y}S}, \qquad (31)$$

where $S=\sum_{i=1}^{K-1}e^{-z_i}$. The cdf of Z is obtained by computing the expectation of $F_{Z|G_K}$ in G_K:

$$\begin{aligned}
F_Z(z)&=\int_{-\infty}^{\infty}e^{-e^{-y}S}e^{-(y+e^{-y})}\mathrm{d}y=\int_{-\infty}^{\infty}e^{-y-e^{-y}(1+S)}\mathrm{d}y\\
&=\tfrac{1}{1+S}\int_{-\infty}^{\infty}(1+S)e^{-y-(1+S)e^{-y}}\mathrm{d}y=\tfrac{1}{1+S}\int_{-\infty}^{\infty}e^{-v-e^{-v}}\mathrm{d}v=\tfrac{1}{1+S}, \qquad (32)
\end{aligned}$$

where $v=y-\log(S+1)$ and the last equality is by recognizing the Gumbel density under the integral. □

Theorem 1 (ZGR)**.** *The Gumbel-Rao estimator for one-hot embedding ϕ in the limit of zero temperature is given by*

$$J^{ZGR}_{\theta_i} = \begin{cases} \frac{1}{2}(J_{\phi_i} - J_{\phi_x})p(x{=}i;\eta) & \text{if } i \neq x; \\ -\frac{1}{2}\sum_{j\neq x}(J_{\phi_j} - J_{\phi_x})p(x{=}j;\eta) & \text{if } i = x. \end{cases} \tag{12}$$

Proof. We can take $J_\phi = \frac{\mathrm{d}\mathcal{L}(\phi(x))}{\mathrm{d}\phi}$ out of the conditional expectation since it does not depend on G for fixed x. Furthermore, X is distributed as $p(x)$ by the sampling procedure and therefore $\mathbb{P}_G(X{=}x) = p(x)$. We can thus rewrite the conditional expectation (8) as

$$J_\phi \int [\![X(u) = x]\!] \frac{\mathrm{d}\tilde{\phi}(u)}{\mathrm{d}\theta} \mathrm{d}F_G(u)/\mathbb{P}_G(X{=}x) = \frac{1}{p(x)} J_\phi \int\limits_{\arg\max(\theta+u)=x} \frac{\mathrm{d}\tilde{\phi}(u)}{\mathrm{d}\theta} \mathrm{d}F_G(u), \tag{33}$$

where F_G is the joint cdf of G. The condition $\arg\max_k(\theta_k + G_k) = x$ can be expressed as

$$\theta_j + G_j - (\theta_x + G_x) \leq 0, \quad \forall j \neq x. \tag{34}$$

Let us define $\beta_j = \theta_j - \theta_x$ and $Z_j = G_j - G_x$ for $j = 1 \dots K$. Note that $\beta_x = Z_x = 0$ by this definition. Then the constraint can be written as

$$Z \leq -\beta. \tag{35}$$

The integrand $\tilde{\phi}$ expresses in variables β, Z as

$$\tilde{\phi} = \mathrm{softmax}_\tau(\theta + G) = \mathrm{softmax}_\tau(\beta + Z). \tag{36}$$

Let us denote $Z_{\neg x} = (Z_j | j \neq x)$. The joint distribution of $Z_{\neg x}$ is the $(K-1)$-variate multivariate logistic distribution (Malik & Abraham, 1973), as detailed in Lemma A.1, with cdf:

$$F_{Z_{\neg x}}(z_{\neg x}) = \frac{1}{1+\sum_{i\neq x} e^{-z_i}}. \tag{37}$$

To simplify notation, we let Z_x have the discrete law with mass 1 at a single point $z_x = 0 = -\beta_x$ and extend $F_{Z_{\neg x}}$ to the full joint F_Z accordingly. We then can rewrite the integral as

$$\Big(\int\limits_{z\leq -\beta} \frac{\partial}{\partial\beta} \mathrm{softmax}_\tau(\beta + z) \mathrm{d}F_Z(z) \Big) \frac{\partial\beta}{\partial\theta}. \tag{38}$$

The Jacobian $\frac{\partial}{\partial\beta}\mathrm{softmax}_\tau(\beta + z)$ is a $K \times K$ matrix with indices (k, j) where the column $j = x$ is zero by definition. Let us consider one component of the above integral for $j \neq x$:

$$I_{k,j} = \int\limits_{z\leq -\beta} \frac{\partial}{\partial\beta_j} \mathrm{softmax}_\tau(\beta + z)_k \mathrm{d}F_Z(z). \tag{39}$$

We want to evaluating its limit for $\tau \to 0$. We cannot push the limit under the integral in this form, we need to transform it first. To shorten the notation, let us denote $a_i = e^{(z_i+\beta_i)/\tau}$. We change the variable z_j by the mapping

$$T\colon z_j \mapsto v_j = (2 + S_j)\frac{a_j}{1+a_j+S_j}. \tag{40}$$

where $S_j = \sum_{i\neq x,j} a_j$. This mapping is monotone increasing and one-to one from $(-\infty, -\beta)$ to $(0, 1)$, therefore the constraint $z_j \leq -\beta_j$ will trivialize.

Let $A_k = \mathrm{softmax}_\tau(\beta + z)_k$. We can rewrite the integrand $\frac{\partial}{\partial\beta_j} A_k \mathrm{d}z_j$ as follows:

$$\begin{aligned} \frac{\mathrm{d}A_k}{\mathrm{d}\beta_j}\mathrm{d}z_j &= \frac{\mathrm{d}A_k}{\mathrm{d}z_j}\mathrm{d}z_j && (A_k \text{ depends on } \beta_j \text{ in the same way as on } z_j) \\ &= \frac{\mathrm{d}A_k}{\mathrm{d}z_j}\frac{\mathrm{d}z_j}{\mathrm{d}v}\mathrm{d}v = \frac{\mathrm{d}A_k}{\mathrm{d}v_j}\mathrm{d}v. \end{aligned} \tag{41}$$

We will thus need to evaluate $C_j := \frac{\mathrm{d}A_k}{\mathrm{d}v_j} = \frac{\mathrm{d}A_k}{\mathrm{d}a_j}\frac{\mathrm{d}a_j}{\mathrm{d}v_j} = \frac{\mathrm{d}A_k}{\mathrm{d}a_j}(\frac{\mathrm{d}v_j}{\mathrm{d}a_j})^{-1}$. For $j = k$ we simply have

$$A_k = \frac{a_j}{1+a_j+S_j} = \frac{1}{2+S_j}v_j; \;\; C_j = \frac{1}{2+S_j}. \tag{42}$$

For $j \neq k$ we have

$$\frac{\mathrm{d}A_k}{da_j} = \frac{\mathrm{d}}{da_j}\frac{a_k}{1+a_j+S_j} = \frac{-a_k}{(1+a_j+S_j)^2} \tag{43a}$$

$$\frac{\mathrm{d}v_j}{da_j} = (2+S_j)\Big(\frac{1}{1+a_j+S_j} - \frac{a_j}{(1+a_j+S_j)^2}\Big) = \frac{(2+S_j)(1+S_j)}{(1+a_j+S_j)^2} \tag{43b}$$

$$C_j = \frac{-a_k}{(2+S_j)(1+S_j)}. \tag{43c}$$

The integral $I_{k,j}$ with the change of variable $z_j \mapsto v_j$ expresses as

$$\int\limits_{z_i \leq -\beta_i \quad \forall i \neq j} \int_0^1 C_j f_\tau(v_j|z_{\neg j})\mathrm{d}v_j \mathrm{d}F_{Z_{\neg j}}(z_{\neg j}), \tag{44}$$

where $z_{\neg j} = (z_i | i \neq j)$ and

$$f_\tau(v_j|z_{\neg j}) = p_{Z_j|Z_{\neg j}}(T^{-1}(v_j)|z_{\neg j}). \tag{45}$$

The dependance of f_τ on τ is through T, while C_j depends on τ and $z_{\neg j}$. Note that f is a squashed density and is itself not a density.

Next we show dominated convergence of $h_\tau(v_j, z_{\neg j}) = C_j f_\tau(v_j|z_{\neg j})$ in $\tau \to 0$. If $h_\tau(v_j, z_{\neg j})$ converges point-wise and bounded above by an integrable function, then the limit $\tau \to 0$ can be taken under the integral.

We show a constant bound on $h_\tau(v_j, z_{\neg j})$ as follows. Note that $|C_j| \leq 1$. We then have

$$|h_\tau(v_j, z_{\neg j})| \leq \sup_{v \in (0,1), z_{\neg j}} f(v_j|z_{\neg j}) = \sup_z p_{Z_j|Z_{\neg j}}(z_j|z_{\neg j}), \tag{46}$$

which is the supremum of the conditional density of the standard multivariate distribution and is equal to some constant c independent of τ. The integral of a constant function c over $(0,1) \times \mathbb{R}^{K-1}$ with respect to the measure $\mathrm{d}v_j \mathrm{d}F_{Z_{\neg j}}(z_{\neg j})$ is c.

The point-wise limit is as follows. For z satisfying the constraints $z_i + \beta_i \leq 0$ strictly for $i \neq j, x$, we have

$$\lim_{\tau \to 0} a_i = \lim_{\tau \to 0} e^{(\beta_i + z_i)/\tau} = 0 \text{ and } \lim_{\tau \to 0} S_j = 0. \tag{47}$$

Therefore we have

$$\lim_{\tau \to 0} C_j = \begin{cases} \frac{1}{2} & \text{if } j = k; \\ 0 & \text{if } j \neq k \wedge k \neq x; \\ -\frac{1}{2} & \text{if } j \neq k \wedge k = x. \end{cases} \tag{48}$$

The inverse of mapping T is given by the relations

$$a_j = \frac{v_j(1+S_j)}{2+S_j-v_j}; \quad z_j = -\beta_j + \tau \log(a_j). \tag{49}$$

It is seen that the limit of $\log(a_j)$ is finite and therefore

$$\lim_{\tau \to 0} T^{-1}(v_j) = -\beta_j. \tag{50}$$

and

$$\begin{aligned} \lim_{\tau \to 0} f_\tau(v_j|z_{\neg j}) = \lim_{\tau \to 0} p_{Z_j|Z_{\neg j}}(T^{-1}(v_j)|z_{\neg j}) &= p_{Z_j|Z_{\neg j}}(\lim_{\tau \to 0} T^{-1}(v_j)|z_{\neg j}) \\ &= p_{Z_j|Z_{\neg j}}(-\beta_j|z_{\neg j}). \end{aligned} \tag{51}$$

By dominated convergence theorem, we can now claim

$$\lim_{\tau \to 0} I_{k,j} = 0 \text{ if } j \neq k \text{ and } k \neq x. \tag{52}$$

And elsewise, if $j = k$ or $k = x$,

$$\lim_{\tau \to 0} I_{k,j} = \int\limits_{z_i \leq -\beta_i \quad \forall i \neq j} \pm\tfrac{1}{2} p_{Z_j|Z_{\neg j}}(-\beta_j|z_{\neg j})\mathrm{d}F_{Z_{\neg j}}(z_{\neg j}) = \pm\tfrac{1}{2}\frac{\partial}{\partial z_j}F_Z(z)\Big|_{z=-\beta} \tag{53}$$

$$= \mp\tfrac{1}{2}\frac{\partial}{\partial \beta_j}\frac{1}{1+\sum_{i \neq x} e^{\beta_i}} = \mp\tfrac{1}{2}\frac{\partial}{\partial \beta_j}\frac{e^{\beta_x}}{\sum_i e^{\beta_i}} = \mp\tfrac{1}{2}\frac{\partial}{\partial \beta_j}\operatorname{softmax}(\beta)_x \tag{54}$$

$$= \pm\tfrac{1}{2}p(x)p(j), \tag{55}$$

where the upper sign corresponds to the case $j = k$ and the lower to $k = x$.

Let us denote $\hat{I} = \lim_{\tau\to 0} I$. Multiplying it with the incoming derivative J_ϕ on the left, we obtain:

$$(J_\phi \hat{I})_j = \tfrac{1}{2}(J_{\phi_j} - J_{\phi_x})p(x)p(j). \tag{56}$$

And finally, multiplying (56) with the Jacobian $\frac{\partial\beta}{\partial\theta}$ on the right per (38) and with the factor $\frac{1}{p(x)}$ per (33), we obtain

$$J^{\text{ZGR}}_{\theta_i} = \begin{cases} \frac{1}{2}(J_{\phi_i} - J_{\phi_x})p(i) & \text{if } i \neq x; \\ -\frac{1}{2}\sum_{j\neq x}(J_{\phi_j} - J_{\phi_x})p(j) & \text{if } i = x. \end{cases} \tag{57}$$

□

Proposition 2. ZGR estimator decomposes as

$$\boxed{J^{\text{ZGR}} = \tfrac{1}{2}(J^{\text{ST}} + J^{\text{DARN}}),} \tag{13}$$

with the choice $\bar{\phi} = \bar{\phi}(\eta)$ in DARN.

Proof. Let p denote the vector of probabilities $(p(x{=}k;\eta)|k = 0,\dots,K-1)$. Recall that we have derived ZGR under the assumption of on-hot embedding ϕ, inherited from GS. In this case $J_\phi\phi(i) = J_{\phi_i}$ and $\bar{\phi}_k = \sum_i \phi(i)_k p_i = p_k$.

Note that ZGR (57) defines the gradient in the parametrization θ used in Gumbel Rao and initially in Gumbel-Softmax, while ST and DARN estimators are given by us with respect to η. We need to bring these two to a common basis. We chose to reconstruct J^{ZGR}_p because both J^{ST}_p and J^{DARN}_p are particularly simple:

$$J^{\text{ST}}_{p_i} = J_\phi\phi(i) = J_{\phi_i}, \tag{58}$$

$$J^{\text{DARN}}_{p_i} = J_\phi(\phi_i - \bar{\phi})[\![x{=}i]\!]/p(x) = (J_{\phi_i} - J_\phi p)[\![x{=}i]\!]/p(x). \tag{59}$$

Note, because p lies in the simplex, gradients in p are defined up to an additive constant to all coordinates. In other words any such additive constant is irrelevant and will not affect the gradient in η.

In order to reconstruct J^{ZGR}_p we represent $J^{\text{ZGR}}_\theta = J^{\text{ZGR}}_p P$, where P is the Jacobian of softmax, given by

$$P = \operatorname{diag}(p) - pp^\mathsf{T} = \operatorname{diag}(p)(I - \mathbf{1}p^\mathsf{T}). \tag{60}$$

We first note that J^{ZGR}_θ satisfies $\sum_i J^{\text{ZGR}}_{\theta_i} = 0$ (as any gradient should, but not necessarily a stochastic estimator) and therefore

$$J^{\text{ZGR}}_\theta = J^{\text{ZGR}}_\theta(I - \mathbf{1}p^\mathsf{T}) = J^{\text{ZGR}}_\theta \operatorname{diag}(p)^{-1}P. \tag{61}$$

We obtained:

$$J^{\text{ZGR}}_{p_i} = \begin{cases} \frac{1}{2}(J_{\phi_i} - J_{\phi_x}) & \text{if } i \neq x; \\ -\frac{1}{2}\sum_{j\neq x}(J_{\phi_j} - J_{\phi_x})p(j)/p(x) & \text{if } i = x, \end{cases} \tag{62}$$

up to a constant, *i.e.* adding the same number c to all components. We further add the constant $\frac{1}{2}J_{\phi_x}$ and obtain

$$J^{\text{ZGR}}_{p_i} = \begin{cases} \frac{1}{2}J_{\phi_i} & \text{if } i \neq x; \\ \frac{1}{2}J_{\phi_x} - \frac{1}{2}\sum_{j\neq x}(J_{\phi_j} - J_{\phi_x})p(j)/p(x) & \text{if } i = x, \end{cases} \tag{63}$$

Subtracting $\frac{1}{2}J^{\text{ST}}_p$, the reminder is $\frac{1}{2}J^{\text{RE}}_p$ with

$$J^{\text{RE}}_{p_i} = [\![i{=}x]\!]\tfrac{1}{p(x)}\textstyle\sum_{j\neq x}(J_{\phi_x} - J_{\phi_j})p(j). \tag{64}$$

Simplifying

$$\textstyle\sum_{j\neq x}(J_{\phi_x} - J_{\phi_j})p(j) = \sum_j (J_{\phi_x} - J_{\phi_j})p(j) = J_{\phi_x} - \sum_j J_{\phi_j}p(j) \tag{65}$$

we obtain

$$J^{\text{RE}}_{p_i} = [\![i{=}x]\!]\tfrac{1}{p(x)}\Big(J_{\phi_x} - \textstyle\sum_j J_{\phi_j}p(j)\Big). \tag{66}$$

and we see that $J^{\text{RE}}_p = J^{\text{DARN}}_p$ with $\bar{\phi} = p = \bar{\phi}(\eta)$. □

Theorem 2. *ZGR is unbiased for quadratic loss functions $\mathcal{L}$.*

Proof. Since ZGR estimator is linear in $\mathcal{L}$ (estimate for a linear combination of two loss functions is the linear combination of estimates), it is sufficient to prove the claim for one-hot embedding ϕ and some elementary functions forming a basis for all quadratic functions. With one-hot embedding we have $\bar{\phi}(\eta)_i = p(x{=}i;\eta) = p_i$.

Let us start with a linear monomial $\mathcal{L}(x) = \phi(x)_i$. The expected loss is $\mathbb{E}[\mathcal{L}(x)] = p(x{=}i;\eta)$. The true gradient is

$$J_\eta = \tfrac{\mathrm{d}}{\mathrm{d}\eta} p(x{=}i;\eta). \tag{67}$$

Substituting $J_{\phi_k} = [\![k{=}i]\!]$ in ST we have

$$J_\eta^{\mathrm{ST}} = \tfrac{\mathrm{d}\mathcal{L}(\phi(x))}{\mathrm{d}\phi} \tfrac{\mathrm{d}\bar{\phi}(\eta)}{\mathrm{d}\eta} = \tfrac{\mathrm{d}\bar{\phi}(\eta)_i}{\mathrm{d}\eta} = J_\eta. \tag{68}$$

This may come as a surprise for someone, but ST for a single categorical variable is exact (zero bias and zero variance). The expectation of J^{DARN} simplifies as follows for any $\bar{\phi}$ and a linear loss function, ensuring that J_ϕ is constant in x:

$$\begin{aligned}\mathbb{E}[J_\eta^{\mathrm{DARN}}] &= \textstyle\sum_x p(x) J_\phi(\phi(x) - \bar{\phi}) \frac{1}{p(x)} \frac{\mathrm{d}p(x;\eta)}{\mathrm{d}\eta} = J_\phi \textstyle\sum_x (\phi(x) - \bar{\phi}) \frac{\mathrm{d}p(x;\eta)}{\mathrm{d}\eta} \\ &= J_\phi \textstyle\sum_x \phi(x) \frac{\mathrm{d}p(x;\eta)}{\mathrm{d}\eta} - J_\phi \bar{\phi} \frac{\mathrm{d}}{\mathrm{d}\eta} \textstyle\sum_x p(x;\eta) = J_\phi \frac{\mathrm{d}\bar{\phi}(\eta)}{\mathrm{d}\eta}.\end{aligned} \tag{69}$$

Substituting $J_{\phi_k} = [\![k{=}i]\!]$ and $\bar{\phi}(\eta) = p$ we obtain

$$\mathbb{E}[J_\eta^{\mathrm{DARN}}] = \tfrac{\mathrm{d}p(x{=}i;\eta)}{\mathrm{d}\eta} = J_\eta, \tag{70}$$

reconfirming that DARN is unbiased for linear function of categorical variables as expected. It follows that $\frac{1}{2}(J_\eta^{\mathrm{ST}} + J_\eta^{\mathrm{DARN}})$ is also unbiased.

Let us now consider the elementary quadratic function $\mathcal{L}(\phi(x)) = \phi(x)_i^2 - \phi(x)_i$. For all discrete assignments it is zero, therefore the true gradient of its expected value is zero. We have

$$J_{\phi_k}(x) = \begin{cases} 2\phi(x)_i - 1 & k = i \\ 0 & k \neq i. \end{cases} \tag{71}$$

Therefore $J_{p_k}^{\mathrm{ST}} = 0$ for $k \neq i$ and

$$\mathbb{E}[J_{p_i}^{\mathrm{ST}}] = \mathbb{E}[2\phi(x)_i - 1] = 2p_i - 1. \tag{72}$$

For $J_{p_i}^{\mathrm{DARN}}$ we have

$$J_{p_i}^{\mathrm{DARN}} = (J_{\phi_i}(x) - J_\phi(x)p) \tfrac{1}{p(x)} [\![x{=}i]\!] \tag{73}$$

$$= (1 - p_i)(2\phi(x)_i - 1) \tfrac{1}{p(x)} [\![x{=}i]\!]. \tag{74}$$

Its expectation is

$$(1 - p_i)(2\phi_i(i) - 1) = 1 - p_i. \tag{75}$$

For $J_{p_k}^{\mathrm{DARN}} = 0$ for $k \neq i$ we have

$$J_{p_k}^{\mathrm{DARN}} = (J_{\phi_k}(x) - J_\phi(x)p) \tfrac{1}{p(x)} [\![x{=}k]\!] \tag{76a}$$

$$= -J_\phi(x) p \tfrac{1}{p(x)} [\![x{=}k]\!] \tag{76b}$$

$$= -p_i(2\phi(x)_i - 1) \tfrac{1}{p(x)} [\![x{=}k]\!]. \tag{76c}$$

Its expectation is

$$-p_i(2\phi_i(k) - 1) = p_i. \tag{77}$$

By subtracting p_i from all ordinates of J_p^{DARN}, we obtain an equivalent (having identical derivative in η) form where $\mathbb{E}[J_{p_k}^{\mathrm{DARN}}] = 0$ for all $k \neq i$ and $\mathbb{E}[J_{p_i}^{\mathrm{DARN}}] = 1 - 2p_i$, which cancels with $\mathbb{E}[J_{p_i}^{\mathrm{ST}}]$.

Next we consider a bilinear monomial in ϕ: $\mathcal{L}(\phi(x)) = \phi(x)_1\phi(x)_2$, where we have taken indices 1 and 2, without loss of generality. Its is zero for all discrete assignments and therefore the gradient of its expectation is zero. We have

$$J_{phi_1} = \phi_2(x) = [\![x{=}2]\!] \tag{78}$$
$$J_{phi_2} = \phi_1(x) = [\![x{=}1]\!]. \tag{79}$$

For ST we have $J_p^{\mathrm{ST}} = J_\phi$ and

$$\mathbb{E}[J_{p_1}^{\mathrm{ST}}] = p_2, \tag{80}$$
$$\mathbb{E}[J_{p_2}^{\mathrm{ST}}] = p_1, \tag{81}$$
$$\mathbb{E}[J_{p_k}^{\mathrm{ST}}] = 0, \quad k \neq 1, 2. \tag{82}$$

For DARN part we have:

$$J_{p_1}^{\mathrm{DARN}} = [\![x{=}1]\!]\tfrac{1}{p_1}(J_{\phi_1} - J_{\phi_1}p_1 - J_{\phi_2}p_2), \tag{83}$$
$$J_{p_2}^{\mathrm{DARN}} = [\![x{=}2]\!]\tfrac{1}{p_2}(J_{\phi_2} - J_{\phi_1}p_1 - J_{\phi_2}p_2), \tag{84}$$
$$J_{p_k}^{\mathrm{DARN}} = [\![x{=}k]\!]\tfrac{1}{p_k}(-J_{\phi_1}p_1 - J_{\phi_2}p_2), \quad k \neq 1, 2. \tag{85}$$

In the expectation, substituting J_ϕ:

$$\mathbb{E}[J_{p_1}^{\mathrm{DARN}}] = [\![x{=}1]\!]\tfrac{1}{p_1}(\phi_2(1) - \phi_2(1)p_1 - \phi_1(1)p_2) = -p_2, \tag{86}$$
$$\mathbb{E}[J_{p_2}^{\mathrm{DARN}}] = [\![x{=}2]\!]\tfrac{1}{p_2}(\phi_1(2) - \phi_2(2)p_1 - \phi_1(2)p_2) = -p_1, \tag{87}$$
$$\mathbb{E}[J_{p_k}^{\mathrm{DARN}}] = [\![x{=}k]\!]\tfrac{1}{p_k}(-\phi_2(k)p_1 - \phi_1(k)p_2) = 0, \quad k \neq 1, 2. \tag{88}$$

This exactly cancels with ST.

The elementary functions we have considered form a basis in the space of all quadratic functions. By linearity argument, $J^{ZGR} = \frac{1}{2}(J^{ST} + J^{DARN})$ is unbiased for all quadratic functions. □

B Details of Experiments

Here we give detailed specifications of our experiments. The implementation of all experiments will be made publicly available upon publication. During the review period, we will be happy to answer questions and share the code with reviewers confidentially through the OpenReview platform.

Dataset We use MNIST data set[5]. It contains 60000 training and 10000 test images of handwritten digits. We used 54000 images for trainig and 6000 for validation.

B.1 VAE

Preprocessing The grayscale image intensities in $[0, 1]$ were binarized with the threshold 0.5. This is in order to replicate similar benchmarking experiments in the literature for VAE with Bernoulli observation model.

Model Each categorical variable is encoded as a vector of ± 1, corresponding to the bit representation of x. The network architecture is adopted from Yin et al. (2019):

Linear(784,512) → LReLU → Linear(512,256) → LReLU → Linear(256, D*K),

Where in the last layer we have D K-way categorical units. The results in Fig. 2 use $D = 16$ and $K = 2^4$. The output of the encoder defines logits of the encoder Bernoulli model $q(z_i{=}1|x)$, where x is the input binary image and z is the latent discrete state.

The decoder has exactly the reverse Linear-LReLU architecture and outputs logits of conditionally independent Bernoulli generative model $p(x_i{=}1|z)$. We optimize the standard evidence lower bound Kingma & Welling (2013) with prior distribution $p(z)$ uniform and not learned.

[5] http://yann.lecun.com/exdb/mnist/

Optimization We compute the KL term in ELBO analytically for a mini-batch and use its exact gradient. The gradient of the expectation of the data term estimated.

In this setup we measure the gradient accuracy and the training performance and do not make use of validation or test sets.

B.2 Quantized Neural Networks

Experimental Setup In this experiment we train a convolutional network 32C5-MP2-64C5-MP2-512FC-10FC, closely replicating the model evaluated by Louizos et al. (2019) for MNIST.

All gradient estimators are working with the same network, parametrization and initialization. In the case of logistic noise the noise standard deviation is learnable and is initialized to $1/3$. All methods are applied with Adam optimizer for 200 epochs. For every method we select the bast validation performance with the grid search for the learning rate from $\{10^{-3}, 3.3 * 10^{-4}, 10^{-4}\}$ and weight decay in $\{0, 10^{-4}\}$. We used the step-wise learning rate schedule decreasing the learning rate 10 times at epochs 100 and 150.

For validation and testing, we evaluate the network in the 'deterministic' mode, turning off all injected noises. This corresponds to a simple deterministic quantized model to be deployed.