Algorithms for the Communication of Samples

Lucas Theis [1] **Noureldin Yosri** [2]

Abstract

The efficient communication of noisy data has applications in several areas of machine learning, such as neural compression or differential privacy, and is also known as reverse channel coding or the channel simulation problem. Here we propose two new coding schemes with practical advantages over existing approaches. First, we introduce *ordered random coding* (ORC) which uses a simple trick to reduce the coding cost of previous approaches. This scheme further illuminates a connection between schemes based on importance sampling and the so-called *Poisson functional representation*. Second, we describe a hybrid coding scheme which uses dithered quantization to more efficiently communicate samples from distributions with bounded support.

1. Introduction

Consider a problem where a sender has information $\mathbf{x}$ and wants to communicate a noisy version of it over a digital channel,

$$\mathbf{Z} \sim \mathbf{x} + \mathbf{U}. \tag{1}$$

The sender does not care which value the noise $\mathbf{U}$ takes as long as it follows a given distribution. For example, it may be desired that the noise is a fair sample from a Gaussian distribution. Can we exploit the sender's indifference to the exact value of the noise to save bits in the communication? More generally, we may want to send a sample from a given distribution,

$$\mathbf{Z} \sim q_{\mathbf{x}}. \tag{2}$$

How can we communicate such a sample most efficiently? This is the problem of *reverse channel coding*. While channel coding tries to communicate digital information over a noisy channel with as few errors as possible, reverse channel coding attempts to do the opposite, namely to simulate a noisy channel over a digital channel. This problem has therefore also been referred to as "channel simulation" (e.g., Cuff, 2008) and is closely related to "relative entropy coding" (Flamich et al., 2020).

The reverse channel coding problem occurs in many applications. In neural compression, differentiable channels enable gradient-based methods to optimize encoders but these channels are necessarily noisy if we want to limit their capacity. For example, it is common to approximate quantization with uniform noise when training neural networks for lossy compression (Ballé et al., 2017). Reverse channel coding allows us to implement such noisy channels at test time (Agustsson & Theis, 2020) and to use arbitrary distributions in place of uniform noise (Havasi et al., 2019).

In differential privacy, most mechanisms seek to limit the amount of sensitive information revealed to another party by adding noise to the data (Dwork et al., 2006). Efficiently communicating such private information is an active area of research (e.g., Chen et al., 2020; Shah et al., 2022) and the goal of reverse channel coding.

Quantum teleportation can be viewed as another instance of reverse channel coding where classical bits are used to communicate stochastic information in the form of a qubit. Some of the earliest results on reverse channel coding were obtained in quantum mechanics (Bennett & Shor, 2002).

The naive approach to our problem would be to let the sender generate a sample and then to encode this noisy data. If the data or the noise stems from a continuous distribution, lossless coding is impossible as it would require an infinite number of bits. On the other hand, lossy coding (by first quantizing) leads to further corruption of the data and still wastes bits on encoding the remaining noise. In contrast, efficient reverse channel coding techniques are able to communicate such information with a coding cost which is close to the mutual information between the data $\mathbf{X}$ and the sample $\mathbf{Z}$ (e.g., Li & El Gamal, 2018),

$$I[\mathbf{X}, \mathbf{Z}] = h[\mathbf{Z}] - h[\mathbf{Z} \mid \mathbf{X}] = \mathbb{E}[D_{\mathrm{KL}}[q_{\mathbf{X}} \parallel p]], \tag{3}$$

where h is the differential entropy and p is the marginal distribution of $\mathbf{Z}$. Unlike the naive approach, the coding cost actually decreases as we introduce more noise, that is,

[1] Google, London, UK [2] Google, Dublin, Ireland. Correspondence to: Lucas Theis <theis@google.com>.

Proceedings of the 39^{th} *International Conference on Machine Learning*, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

when the (differential) entropy of $q_{\mathbf{X}}$ increases.

More formally, a *reverse channel code* consists of an encoder f and a decoder g,

$$f : \mathcal{X} \times [0, 1) \to \mathbb{N}_0, \quad g : \mathbb{N}_0 \times [0, 1) \to \mathcal{Z}, \tag{4}$$

where $\mathcal{X}$ and $\mathcal{Z}$ are arbitrary spaces. The encoder takes $\mathbf{x} \in \mathcal{X}$ together with a possibly infinite number of bits represented by a real number $u \in [0, 1)$ and outputs a discrete representation $k \in \mathbb{N}_0$. The decoder accepts k and u and outputs $\mathbf{z} \in \mathcal{Z}$.

Assume $\mathbf{X}$ and $\mathbf{Z}$ are random variables jointly distributed over $\mathcal{X} \times \mathcal{Z}$ such that $I[\mathbf{X}, \mathbf{Z}] < \infty$. Let $K = f(\mathbf{X}, U)$ where U is a random variable uniformly distributed over $[0, 1)$ and independent of $\mathbf{X}$. For simplicity, we will assume that for all $\mathbf{x} \in \mathcal{X}$ the conditional distribution of $\mathbf{Z}$ given $\mathbf{X} = \mathbf{x}$ has a density $q_{\mathbf{x}}$ with respect to a Borel or counting measure on $\mathcal{Z}$. A *reverse channel coding problem* is any problem which tries to find a code such that $H[K \mid U]$ is small and the distribution of $g(K, U)$ is simultaneously close to $q_{\mathbf{x}}$ in some well-defined sense. In this paper, we will measure closeness in terms of the total variation divergence (Eq. 15).

Here we assume that the encoder and decoder have access to a *shared source of randomness* U which we may therefore also be used to encode K at a coding cost close to $H[K \mid U]$. Other variants of reverse channel coding limit the amount of shared randomness which can be used but are not considered here.

In the following, we will first provide an overview of a few useful algorithms implementing reverse channel codes. For instance, we will describe a practical algorithm based on the so-called Poisson functional representation (Li & El Gamal, 2018). We will then introduce new algorithms with practical advantages over existing approaches along with theoretical results on their properties, which represents our main contribution. As a further contribution, we will present a unifying view of some of the algorithms which helps to clarify the relationship between them and sheds light on their empirical behavior. Finally, we will provide the first direct empirical comparison between different reverse channel coding algorithms. All proofs and additional empirical results can be found in the appendix.

2. Related work

The *reverse Shannon theorem* of Bennett & Shor (2002) shows that a sender who has access to $\mathbf{X}$ can communicate an instance of $\mathbf{Z}$ at a cost which is close to the two random variables' mutual information. Many papers have considered problems related to reverse channel coding and derived bounds on the coding cost of communicating a sample (e.g., Cover & Permuter, 2007; Harsha et al., 2007; Braverman & Garg, 2014). To our knowledge, the sharpest known upper bound was provided by Li & Anantharam (2021) who showed that an optimal code using shared randomness does not require more than

$$I[\mathbf{X}, \mathbf{Z}] + \log(I[\mathbf{X}, \mathbf{Z}] + 1) + 4.732 \tag{5}$$

bits on average to communicate an exact sample. On the other hand, Li & El Gamal (2018) showed that distributions exist for which the coding cost is at least

$$I[\mathbf{X}, \mathbf{Z}] + \log(I[\mathbf{X}, \mathbf{Z}] + 1) - 1. \tag{6}$$

That is, the bound in Eq. 5 cannot be improved significantly without making additional assumptions about the distributions involved (see also Braverman & Garg, 2014). Note that the communication overhead (the second and third term) becomes relatively less important as the transmitted amount of information increases.

Most general reverse channel coding algorithms operate on the same basic principle. First, a potentially large number of candidates is generated from a fixed distribution which is known to the sender and receiver,

$$\mathbf{Z}_n \sim p, \tag{7}$$

where $n \in \mathbb{N}$ or $n \in \{1, \ldots, N\}$. Both the sender and receiver are able to generate these candidates without communication by using a shared source of randomness. In practice, this will typically be a pseudorandom number generator with a common seed that has been established in advance. The sender selects an index N^* according to some distribution such that, at least approximately,

$$\mathbf{Z}_{N^*} \sim q_{\mathbf{x}}. \tag{8}$$

Note that only N^* needs to be communicated and this can be done efficiently if $H[N^*]$ is small. The main difference between algorithms is in how N^* is decided.

Li & El Gamal (2017) described an algorithm for communicating samples from distributions with log-concave PDFs without common randomness. Without a shared source of randomness, the number of bits required is at least *Wyner's common information* (Wyner, 1975; Cuff, 2008), which can be significantly larger than the mutual information (Xu et al., 2011). In the following, we therefore focus on algorithms with access to common randomness.

Agustsson & Theis (2020) showed that there is no general algorithm whose computational complexity is polynomial in the communication cost. That is, as the amount of information transmitted increases, general purpose algorithms become prohibitively expensive. One solution to this problem is to split information into chunks and to encode these chunks separately (Havasi et al., 2019; Flamich et al., 2020).

However, this reduces statistical efficiency as each chunk will contribute its own overhead to the overall coding cost. We therefore typically find tension between the computational efficiency and the coding efficiency of a scheme.

A more well-known idea in machine learning is *bits-back coding* (Wallace, 1990; Hinton & Van Camp, 1993) which at first glance may appear closely related to reverse channel coding. Here, the goal is to losslessly compress a source $\mathbf{X}$ using a model of its joint distribution with a set of latent variables $\mathbf{Z}$. Encoding an instance $\mathbf{x}$ involves sampling $\mathbf{Z} \sim q_\mathbf{x}$ while using previously encoded bits as a source of randomness. The data and latent variables are subsequently encoded using the model's joint distribution (Townsend et al., 2019). Unlike reverse channel coding, however, bits-back coding necessarily transmits a perfect copy of the data, that is, it is an implementation of lossless source coding. On the other hand, reverse channel coding can be viewed as a generalization of source coding. Lossless source coding is recovered as a special case when choosing $q_\mathbf{x}(\mathbf{z}) = \delta(\mathbf{z}-\mathbf{x})$.

3. Algorithms

We will first continue the discussion of related work by introducing existing algorithms for the simulation of noisy channels. New methods and results are presented in Sections 3.4, 3.5, and 3.7.

3.1. Rejection sampling

Rejection sampling (RS) is a method for generating a sample from one distribution given samples from another distribution. As an introductory example, we show how RS can be turned into a reverse channel coding scheme.

Let $\mathbf{Z}_n$ be candidates drawn independently from a proposal distribution p. Further, let $U_n \sim \text{Uniform}([0,1))$. RS selects the first index N^*_RS such that

$$U_{N^*_\text{RS}} \leq w_\text{min} \frac{q_\mathbf{x}(\mathbf{Z}_{N^*_\text{RS}})}{p(\mathbf{Z}_{N^*_\text{RS}})} \tag{9}$$

where w_min is any number such that

$$w_\text{min} \leq \inf_\mathbf{z} \frac{p(\mathbf{z})}{q_\mathbf{x}(\mathbf{z})}, \tag{10}$$

ensuring that the right-hand side in Eq. 9 is smaller than 1. If $w_\text{min} > 0$, then N^* will be finite and, crucially,

$$\mathbf{Z}_{N^*_\text{RS}} \sim q_\mathbf{x}. \tag{11}$$

A sender could thus communicate a sample from $q_\mathbf{x}$ by sending N^*_RS, assuming the receiver already has access to the candidates $\mathbf{Z}_n$. Note that this works even when the distribution is continuous since N^*_RS will still be discrete. While w_min can be chosen to depend on $\mathbf{x}$, in the following analysis we assume for simplicity that the same value is chosen for all target distributions $q_\mathbf{x}$.

Let us consider the coding cost of encoding N^*_RS. The average probability of accepting a candidate is

$$\int p(\mathbf{z}) w_\text{min} \frac{q_\mathbf{x}(\mathbf{z})}{p(\mathbf{z})} \, d\mathbf{z} = w_\text{min}. \tag{12}$$

The marginal distribution of N^*_RS is therefore a geometric distribution whose entropy can be bounded by

$$H[N^*_\text{RS}] \leq -\log w_\text{min} + 1/\ln 2. \tag{13}$$

Rejection sampling is efficient if $H[N^*_\text{RS}]$ is not much more than the information contained in $\mathbf{Z}$. While it is easy to construct examples where RS is efficient, it is also easy to construct examples where $-\log w_\text{min}$ is significantly larger than $I[\mathbf{X}, \mathbf{Z}]$. For instance, the density ratio in Eq. 10 may be unbounded. However, if we are willing to accept an approximate sample, then there are ways to limit the coding cost even then. For example, we may unconditionally accept the Nth candidate if the first $N-1$ candidates are rejected. In this case, the distribution of $\mathbf{Z}_{N^*}$ will be a mixture distribution with density

$$\beta p(\mathbf{z}) + (1-\beta) q_\mathbf{x}(\mathbf{z}), \tag{14}$$

where $\beta = (1 - w_\text{min})^{N-1}$ is the probability of rejecting all $N-1$ candidates. The quality of a sample is often measured in terms of the *total variation distance* (TVD),

$$D_\text{TV}[p, q] = \frac{1}{2} \int |p(\mathbf{z}) - q(\mathbf{z})| \, d\mathbf{z}. \tag{15}$$

When measuring the distance of the mixture distribution from the target distribution $q_\mathbf{x}$, we obtain

$$D_\text{TV}[\beta p + (1-\beta) q_\mathbf{x}, q_\mathbf{x}] = \beta D_\text{TV}[p, q_\mathbf{x}]. \tag{16}$$

That is, the divergence decays exponentially with N.

An alternative approach to limiting the coding cost is to choose an invalid but larger w_min. Harsha et al. (2007) described a related approach which effectively uses $w_\text{min} = 1$ but is nevertheless able to produce an exact sample by adjusting the target distribution after each candidate rejection (Appendix A). However, their approach is computationally expensive and often infeasible for continuous distributions.

3.2. Minimal random coding

An approach closely related to *importance sampling* was first considered by Cuff (2008) and later dubbed *likelihood encoder* (Song et al., 2016). The approach was independently rediscovered in machine learning by Havasi et al. (2019) who referred to it as *minimal random coding* (MRC) and used it for model compression. It has since also been used for lossy image compression (Flamich et al., 2020). Unlike Havasi et al. (2019), Cuff (2008) only considered discrete distributions and assumed that p is the true marginal distribution of the data. But Cuff (2008) also described a more general approach where the amount of shared randomness between the sender and receiver is limited.

In MRC, the sender picks one of N candidates by sampling an index N^*_{MRC} from the distribution

$$\pi_{\mathbf{x}}(n) \propto q_{\mathbf{x}}(\mathbf{Z}_n)/p(\mathbf{Z}_n). \tag{17}$$

Unlike RS, the distribution of $\mathbf{Z}_{N^*_{\text{MRC}}}$ (call it $\tilde{q}_{\mathbf{x}}$) will in general only approximate $q_{\mathbf{x}}$. On the other hand, the coding cost of MRC can be significantly smaller. Havasi et al. (2019) showed that under reasonable assumptions, samples from $\tilde{q}_{\mathbf{x}}$ will be similar to samples from $q_{\mathbf{x}}$ if the number of candidates is

$$N = 2^{D_{\text{KL}}[q_{\mathbf{x}} \parallel p]+t} \tag{18}$$

for some $t > 0$. That is, the number of candidates required to guarantee a sample of high quality grows exponentially with the amount of information gained by the receiver.

Since acceptable candidates may appear anywhere in the sequence of candidates, each index is a priori equally likely to be picked. That is, the marginal distribution of N^*_{MRC} is uniform and its entropy is

$$H[N^*_{\text{MRC}}] = \log N. \tag{19}$$

3.3. Poisson functional representation

Li & El Gamal (2018) studied the following *Poisson functional representation* (PFR) of a random variable. Let T_n be the arrival times of a homogeneous Poisson process on the non-negative real line such that $T_n \geq 0$ and $T_n \leq T_{n+1}$ for all $n \in \mathbb{N}$. Let

$$\mathbf{Z}_n \sim p, \qquad N^*_{\text{PFR}} = \operatorname*{argmin}_{n \in \mathbb{N}} T_n \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)}. \tag{20}$$

for all $n \in \mathbb{N}$. Then $\mathbf{Z}_{N^*_{\text{PFR}}}$ has the distribution $q_{\mathbf{x}}$. The same construction was already considered by Maddison (2016) for the purpose of Monte Carlo integration but not for channel simulation.

As in RS, the index N^*_{PFR} picks one of infinitely many candidates and we obtain an exact sample from the target distribution. However, Li & El Gamal (2018) provided much stronger guarantees for the coding cost of the N^*_{PFR}. In particular,

$$H[N^*_{\text{PFR}}] \leq I[\mathbf{X}, \mathbf{Z}] + \log(I[\mathbf{X}, \mathbf{Z}] + 1) + 4. \tag{21}$$

The distribution of N^*_{PFR} takes a more complicated form than in RS or MRC (Li & Anantharam, 2021, Eq. 29). Nevertheless, a coding cost corresponding to the bound above can be achieved by entropy encoding N^*_{PFR} with a simple Zipf distribution $p_\lambda(n) \propto n^{-\lambda}$ where (Li & El Gamal, 2018)

$$\lambda = 1 + 1/(I[\mathbf{X}, \mathbf{Z}] + e^{-1} \log e + 1). \tag{22}$$

Algorithm 1 PFR

```
Require: $\mathtt{p}, \mathtt{q}, w_{\text{min}}$
$t, n, s^* \leftarrow 0, 1, \infty$

repeat
    $z \leftarrow \mathtt{simulate}(n, \mathtt{p})$        ▷ Candidate generation
    $t \leftarrow t + \mathtt{expon}(n, 1)$              ▷ Poisson process
    $s \leftarrow t \cdot \mathtt{p}(z)/\mathtt{q}(z)$    ▷ Candidate's score

    if $s < s^*$ then                                     ▷ Accept/reject candidate
        $s^*, n^* \leftarrow s, n$
    end if

    $n \leftarrow n + 1$
until $s^* \leq t \cdot w_{\text{min}}$

return $n^*$
```

A downside of the PFR is that it depends on an infinite number of candidates. Unlike rejection sampling, we cannot consider the candidates in sequence but generally have to consider the scores of all candidates (Eq. 20). However, if we can bound the density ratio as in rejection sampling (Eq. 10), then we can terminate our search for the best candidate after considering a finite number of them. Let

$$S^*_n = \min_{i \leq n} T_i \frac{p(\mathbf{Z}_i)}{q_{\mathbf{x}}(\mathbf{Z}_i)} \tag{23}$$

be the smallest score observed after taking into account n candidates. Since $T_m \geq T_n$ for all $m > n$, all further scores will be at least $T_n w_{\text{min}}$. Hence, if $S^*_n \leq T_n w_{\text{min}}$, we can terminate the search. Algorithm 1 summarizes this idea. Here, $\mathtt{simulate}(n,\ \mathtt{p})$ is a function which simulates a distribution $\mathtt{p}$ by returning the nth pseudorandom sample derived from n and an implicit random seed. Similarly, $\mathtt{expon}(n, 1)$ simulates an exponential distribution with rate 1. The computational complexity of this algorithm is the same as RS, with an expected number of iterations of $1/w_{\text{min}}$ (Maddison, 2016).

3.4. Ordered random coding

In the following we show that, interestingly, a slight modification of MRC is able to reduce the entropy of the selected index while maintaining the exact distribution of the communicated sample.

To generate a sample from $\pi_{\mathbf{x}}$ (Eq. 17), we can write

$$N^* = \operatorname*{argmax}_{n \leq N} \log q_{\mathbf{x}}(\mathbf{Z}_n) - \log p(\mathbf{Z}_n) + G_n, \tag{24}$$

where the G_n are Gumbel distributed random variables (Gumbel, 1954) with scale parameter 1. This is the well-known Gumbel-max trick for sampling from a categorical distribution (Maddison et al., 2014). We can show that this trick still works if we arbitrarily permute the Gumbel random variables, as long as the permutation does not depend on the values of the candidates $\mathbf{Z}_n$. In particular, we have the following result.

Theorem 3.1. *Let $\tilde{G}_n$ be the result of sorting the Gumbel random variables in decreasing order such that $\tilde{G}_1 \geq \cdots \geq \tilde{G}_N$ and define*

$$\tilde{N}^* = \operatorname*{argmax}_{n \leq N} \log q_{\mathbf{x}}(\mathbf{Z}_n) - \log p(\mathbf{Z}_n) + \tilde{G}_n. \tag{25}$$

Then $\mathbf{Z}_{N^*} \sim \mathbf{Z}_{\tilde{N}^*}$.

While the distribution of $\mathbf{Z}_{\tilde{N}^*}$ remains the same, the distribution of $\tilde{N}^*$ is no longer uniform but biased towards smaller indices. Since N^* is uniform, we must have $H[\tilde{N}^*] \leq H[N^*]$. In the next section, we show that $H[\tilde{N}^*]$ is in fact on a par with $H[N^*_{\text{PFR}}]$. We dub the approach *ordered random coding* (ORC) and pseudocode for ORC is provided in Appendix B.

As for the sample quality, it improves quickly once the coding cost exceeds the information contained in a sample. In particular, we have the following corollary to the results of Chatterjee & Diaconis (2018) and Havasi et al. (2019).

Corollary 3.2. *Let $\tilde{q}_{\mathbf{x}}$ be the distribution of $\mathbf{Z}_{\tilde{N}^*}$ where $\tilde{N}^*$ is defined as in Theorem 3.1. If the number of candidates is $N = 2^{D_{\text{KL}}[q_{\mathbf{x}} \,\|\, p]+t}$ for some $t \geq 2e^{-1}\log e$ and $p(\mathbf{z})/q_{\mathbf{x}}(\mathbf{z}) \geq w_{\min} > 0$ for all $\mathbf{z}$, then*

$$D_{\text{TV}}[\tilde{q}_{\mathbf{x}}, q_{\mathbf{x}}] \leq 4\varepsilon \tag{26}$$

where with $B = -\log w_{\min}$ we have

$$\epsilon \leq 2^{-t/8} + \sqrt{2}\exp\left(-\frac{1}{4B^2}(t/2 - e^{-1}\log e)^2\right). \tag{27}$$

3.5. A unifying view

In this section we show a close connection between methods based on importance sampling and the PFR. First, we can rewrite Eq. 24 as follows,

$$N^*_{\text{MRC}} = \operatorname*{argmin}_{n \leq N} S_n \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)} \tag{28}$$

where S_n are exponentially distributed with rate 1. This is true because $-\log S_n$ is Gumbel distributed. Note the similarity to the PFR,

$$N^*_{\text{PFR}} = \operatorname*{argmin}_{n \in \mathbb{N}} T_n \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)}, \tag{29}$$

where $T_n \sim \sum_{m=1}^n S_m$. ORC, on the other hand, first sorts the exponential random variables, $\tilde{S}_1 \leq \cdots \leq \tilde{S}_N$, before choosing

$$N^*_{\text{ORC}} = \operatorname*{argmin}_{n \leq N} \tilde{S}_n \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)}. \tag{30}$$

It turns out that (Rényi, 1953)

$$\tilde{S}_n \sim \sum_{m=1}^{n} \frac{S_m}{N - m + 1}. \tag{31}$$

This allows us to generate the sorted exponential variables in $O(N)$ instead of $O(N \log N)$ time with sorting. More interestingly, Eq. 31 reveals a close connection to the PFR. Where the PFR uses cumulative sums of exponential random variables, ORC uses weighted sums. This representation allows us to arrive at the following result.

Theorem 3.3. *Let S_n be exponentially distributed RVs and $\mathbf{Z}_n \sim p$ for all $n \in \mathbb{N}$ (i.i.d.), and let*

$$T_n = \sum_{m=1}^{n} S_m, \quad \tilde{T}_{N,n} = \sum_{m=1}^{n} \frac{N}{N - m + 1} S_m \tag{32}$$

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

$$N^*_{\text{PFR}} = \operatorname*{argmin}_{n \in \mathbb{N}} T_n \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)}, \tag{33}$$

$$N^*_{\text{ORC}} = \operatorname*{argmin}_{n \leq N} \tilde{T}_{N,n} \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)}. \tag{34}$$

Then $N^*_{\text{ORC}} \leq N^*_{\text{PFR}}$. Further, if N^*_{PFR} is finite then there exists an $M \in \mathbb{N}$ such that for all $N \geq M$ we have $N^*_{\text{ORC}} = N^*_{\text{PFR}}$.

Note that the additional factor N in the definition of $\tilde{T}_{N,n}$ compared to $\tilde{S}_n$ does not change the output of argmin. Using Theorem 3.3, it is not difficult to see that the bound on the coding cost of the PFR also applies to ORC, or the following result.

Corollary 3.4. *Let $C = \mathbb{E}_{\mathbf{X}}[D_{\text{KL}}[q_{\mathbf{X}} \,\|\, p]]$ and let N^*_{ORC} be defined as in Theorem 3.3. Then*

$$H[N^*_{\text{ORC}}] < C + \log(C + 1) + 4. \tag{35}$$

To achieve this bound, a Zipf distribution $p_\lambda(n) \propto n^{-\lambda}$ with $\lambda = 1 + 1/(C + e^{-1} \log e + 1)$ can be used to entropy encode the index, analogous to the PFR (Eq. 22).

The significance of these results is as follows. Li & El Gamal (2018) showed that the PFR is near-optimal in the sense that the entropy of N^*_{PFR} is close to the worst-case coding cost needed for communicating a perfect sample. However, the construction of the PFR relies on an infinite number of candidates and there are no theoretical guarantees of the sample quality if we naively limit the number of candidates to N. In particular, we do not know how quickly the quality of a communicated sample deteriorates as we decrease N, or how large N should be. On the other hand, we do have some idea of the quality of a sample obtained via importance sampling from a finite number of candidates (e.g., Corollary 3.2 or the results of Cuff, 2008; Chatterjee & Diaconis, 2018; Havasi et al., 2019). But the coding cost of MRC is relatively large and continues to grow unbounded as N increases. ORC combines the best of both by inheriting the bounds on the coding cost of the PFR and the sample quality of MRC.

Unlike the PFR, the guarantees of ORC hold for a finite number of samples. Unlike MRC, we can make N arbitrarily large without having to worry about an exploding coding cost, which makes it easier to tune this parameter.

3.6. Dithered quantization

Dithered quantization, also known as *universal quantization* (Ziv, 1985), refers to quantization with a randomly shifted lattice. Consider a scalar $y \in \mathbb{R}$ and a random variable U uniform over any interval of length one, such as $[0, 1)$. Then (e.g., Schuchman, 1964)

$$\lfloor y - U \rceil + U \sim y + U_0, \tag{36}$$

where U_0 is uniform over $[-0.5, 0.5)$. More generally, let Q be a quantizer which maps inputs to the nearest point on a lattice and let $\mathbf{V}$ be a random variable which is uniformly distributed over an arbitrarily placed Voronoi cell of the lattice. Further, let $\mathbf{V}_0$ be uniform over the Voronoi cell which contains the lattice point at zero. Then (Zamir, 2014)

$$Q(\mathbf{y} - \mathbf{V}) + \mathbf{V} \sim \mathbf{y} + \mathbf{V}_0. \tag{37}$$

Dithered quantization has been mainly used as a tool for studying quantization from a theoretical perspective. However, already Roberts (1962) considered some of its practical advantages over uniform quantization for compressing grayscale images, especially in terms of perceptual quality. Theis & Agustsson (2021) showed that universal quantization can outperform vector quantization where a realism constraint is considered. Universal quantization also recently started being used in neural compression (Choi et al., 2019), in particular to realize differentiable training losses at inference time (Agustsson & Theis, 2020).

Algorithm 2 Hybrid coding

```
Require: $c, \mathrm{q}, w_{\min}, N$
$t, n, s^* \leftarrow 0, 1, \infty$

repeat
    $u \leftarrow$ uniform($n, 0, 1$)          ▷ Candidate generation
    $k \leftarrow$ round($c - u$)
    $z \leftarrow k + u$

    $v \leftarrow N/(N - n + 1)$
    $t \leftarrow t + v \cdot$ expon($n, 1$)
    $s \leftarrow t/\mathrm{q}(z)$                ▷ Candidate's score

    if $s < s^*$ then                  ▷ Accept/reject candidate
        $s^*, n^*, k^* \leftarrow s, n, k$
    end if

    $n \leftarrow n + 1$
until $s^* \leq t \cdot w_{\min}$ or $n > N$

return $n^*, k^*$
```

To communicate a sample of a uniform distribution centered around y, the sender encodes $K = \lfloor y - U \rceil$. The receiver decodes K and computes $Z = K + U$, which is distributed as $y + U_0$. Like the reverse channel coding schemes discussed so far, this requires a shared source of randomness in the form of U. Zamir & Feder (1992) showed that dithered quantization is statistically efficient in the sense that

$$H[K \mid U] = I[Y, Z]. \tag{38}$$

That is, the cost of encoding K is as close as can be to the amount of information contained in Z. Note that we can condition on U when encoding K since U is known to both sender and receiver. Another advantage of dithered quantization is that it is computationally highly efficient, at least for the simple case of the integer lattice.

3.7. Hybrid coding

While dithered quantization is computationally much more efficient than general reverse channel coding schemes, it is also much more limited in terms of the distributions it can simulate. Here we propose a hybrid coding scheme for continuous distributions which retains most of the flexibility of general purpose schemes but is computationally more efficient when the support of the target distribution is small.

The general idea is as follows. Instead of drawing candidates from a fixed distribution p, candidates $\mathbf{Z}_n$ are drawn from a distribution $r_\mathbf{x}$ which acts as a bridge and more closely

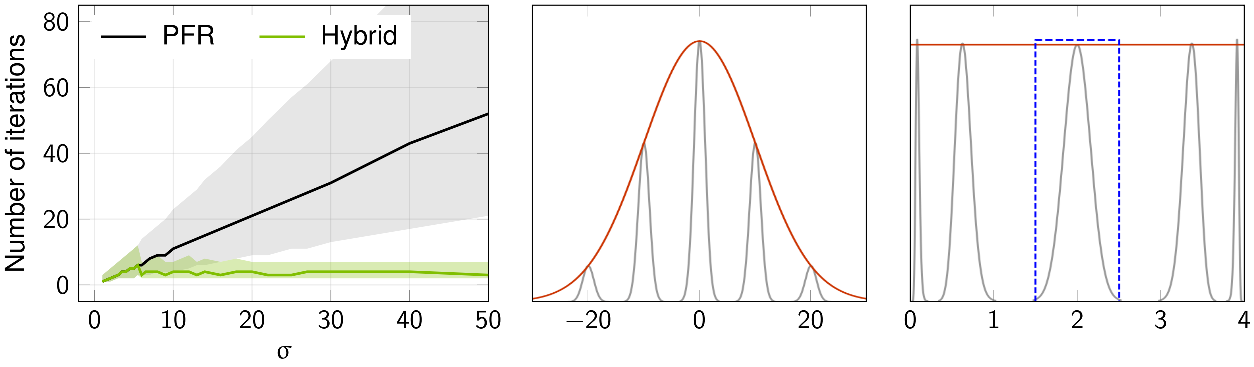

Figure 1. Left: Computational cost of communicating a sample from a Gaussian whose mean varies with standard deviation σ. Solid lines indicate the median number of candidates considered by an algorithm while shaded regions indicate the 25th and 75th percentile. The initial rise and sharp drop of the green curve is due to the discontinuity of M as a function of σ. *Middle:* Illustration of example target distributions $\tilde{q}_\mathbf{x}$ (gray; scaled for visualization purposes) and the marginal distribution $\mathbb{E}[\tilde{q}_\mathbf{x}]$ (red). *Right:* The same distributions after transformation with the marginal's CDF, Φ_{σ^2+1}, and scaling by $M = 4$. The dashed blue line indicates $r_\mathbf{x}$.

resembles the target distribution $q_\mathbf{x}$. Since $r_\mathbf{x}$ is closer to $q_\mathbf{x}$, we will require fewer candidates to find one that is suitable. Let

$$N^* = \operatorname*{argmin}_{n \leq N} \tilde{T}_{N,n} \frac{r_\mathbf{x}(\mathbf{Z}_n)}{q_\mathbf{x}(\mathbf{Z}_n)}, \tag{39}$$

be the index of the selected candidate. Unlike before, only the sender has access to the candidates and so knowing N^* alone will not allow us to reconstruct $\mathbf{Z}_{N^*}$. Our hybrid coding scheme relies on two insights. First, the receiver does not require access to all candidates but only to the selected candidate. Second, the missing information can be encoded easily and efficiently if $r_\mathbf{x}$ can be simulated with dithered quantization.

We assume for now that there exist vectors $\mathbf{c}_\mathbf{x}$ such that the support of $q_\mathbf{x}$ is contained in the support of

$$r_\mathbf{x}(\mathbf{z}) = \begin{cases} 1 & \text{if } \mathbf{z} \in \mathbf{c}_\mathbf{x} + [-0.5, 0.5)^D, \\ 0 & \text{else} \end{cases} \tag{40}$$

which can be simulated via dithered quantization,

$$\mathbf{K}_n = \lfloor \mathbf{c}_\mathbf{x} - \mathbf{U}_n \rceil, \qquad \mathbf{Z}_n = \mathbf{K}_n + \mathbf{U}_n, \tag{41}$$

where $\mathbf{U}_n \sim \text{Uniform}([0,1)^D)$. Define $\mathbf{K}^* = \mathbf{K}_{N^*}$. Hybrid coding transmits the pair $(N^*, \mathbf{K}^*)$, which the receiver uses to reconstruct the selected candidate via

$$\mathbf{Z}_{N^*} = \mathbf{K}^* + \mathbf{U}_{N^*}. \tag{42}$$

Theorem 3.5. *Let N^* and $\mathbf{K}^*$ be defined as in Eq. 39 and below Eq. 41 and let p be the uniform distribution over $[0, M_1) \times \cdots \times [0, M_D)$ for some $M_i \in \mathbb{N}$. Then*

$$H[N^*, \mathbf{K}^*] < C + \log(C - \textstyle\sum_i \log M_i + 1) + 4,$$

where $C = \mathbb{E}_\mathbf{X}[D_\text{KL}[q_\mathbf{X} \,\|\, p]]$.

Theorem 3.5 shows that $(N^*, \mathbf{K}^*)$ is an efficient representation if the marginal distribution of $\mathbf{Z}$ is uniform over some box whose sides have lengths M_i, since then $C = I[\mathbf{X}, \mathbf{Z}]$. However, for continuous random variables this can always be achieved through a transformation Ψ. If $\tilde{q}_\mathbf{x}$ is the desired target distribution before the transformation, then

$$q_\mathbf{x}(\mathbf{z}) = \tilde{q}_\mathbf{x}(\Psi(\mathbf{z}))|D\Psi(\mathbf{z})| \tag{43}$$

is the target distribution in transformed space. Note that after the transformation, the support of $q_\mathbf{x}$ is always bounded. Moreover, for small enough M_i the support will be contained in the support of $r_\mathbf{x}$, satisfying our earlier assumption. This is illustrated for Gaussian distributions in Fig 1.

To achieve the bound suggested by Theorem 3.5, the sender can first encode N^* (e.g., using arithmetic coding; Rissanen & Langdon, 1979) while assuming a Zipf distribution $p_\lambda(n) \propto n^{-\lambda}$ with

$$\lambda = 1 + 1/(C - \textstyle\sum_i \log M_i + e^{-1} \log e + 1). \tag{44}$$

The K_i^* are subsequently added to the bit stream using a fixed rate of $\log M_i$ bits.

Algorithm 2 describes how N^* and K^* are found during the encoding process. Here, $\mathtt{q}$ is the transformed density and $\mathtt{p}$ is not needed as it is assumed to be uniform. Note that hybrid coding effectively reduces to ORC when the support is unconstrained ($M_i = 1$). For larger M_i, the bound on the coding cost improves only slightly but the computational cost reduces significantly. Since the number of candidates required for a sample of high quality grows exponentially in the KLD (Eq. 18) and

$$D_\text{KL}[q_\mathbf{x} \,\|\, r_\mathbf{x}] = D_\text{KL}[q_\mathbf{x} \,\|\, p] - \log M_i, \tag{45}$$

we can expect a speedup on the order of $\prod_i M_i$. We thus want to maximize M_i while making sure that the support of $q_\mathbf{x}$ is still contained within that of $r_\mathbf{x}$.

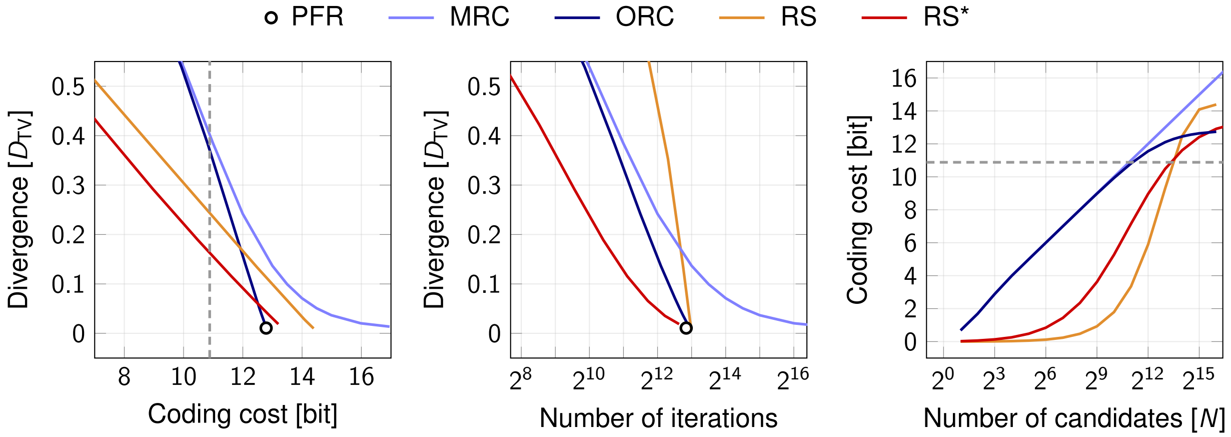

Figure 2. A comparison of various reverse channel coding algorithms for 2^{16}-dimensional categorical distributions which are themselves randomly distributed according to a Dirichlet distribution. Dashed lines indicate the average KLD between the target distribution and the uniform candidate generating distribution. *Left:* The sample quality as a function of the coding cost. *Middle:* The sample quality as a function of the computational cost (for which the number of iterations is a proxy, except for RS*). *Right:* The coding cost as a function of the maximum number of candidates considered.

4. Experiments

We run two sets of empirical experiments to compare the reverse channel coding schemes discussed above. We first investigate the effect of hybrid coding on the computational cost of communicating a (truncated) Gaussian sample. We then compare the performance of a wider set of algorithms for the task of approximately simulating a categorical distribution.

4.1. Gaussian distribution

Consider the task of communicating a sample from a D-dimensional Gaussian with a random mean,

$$\mathbf{Z} \sim \mathcal{N}(\mathbf{X}, \mathbf{I}), \qquad \mathbf{X} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}), \tag{46}$$

where $\mathbf{I}$ is the identity matrix and the mean $\mathbf{X}$ itself is Gaussian distributed with covariance $\sigma^2\mathbf{I}$. The marginal distribution of $\mathbf{Z}$ is Gaussian with mean zero and covariance $\sigma^2\mathbf{I} + \mathbf{I}$ and so we use this distribution as our candidate generating distribution p. The average information gained by obtaining a sample is

$$I[\mathbf{X}, \mathbf{Z}] = -\tfrac{D}{2} \log \left(1 - \tfrac{\sigma^2}{\sigma^2 + 1}\right). \tag{47}$$

To be able to apply the hybrid coding scheme, we slightly truncate the target distribution by assigning zero density to a small fraction θ of values with the lowest density. The TVD between the truncated Gaussian and the Gaussian distribution is θ. A classifier observing $\mathbf{Z}$ would be able to distinguish between these two distributions with an accuracy of at most $1/2 + \theta/2$. In our experiments, we fix $\theta = 10^{-4}$ so that this accuracy is close to chance.

We compare hybrid coding (Algorithm 2) to ORC with $N = \infty$, which reduces to the PFR (Algorithm 1). Using an unlimited number of candidates allows us to avoid any further approximations and to focus on the computational cost. Appropriate values for $w_{\min}$ and M are provided in Appendix H.

Figure 1 shows the average number of iterations an algorithm runs before identifying a suitable candidate of a 1-dimensional (truncated) Gaussian. The computational cost of the PFR grows exponentially with the amount of information transmitted, which is approximately $\log \sigma$. On the other hand, the computational cost of the hybrid coding scheme quickly saturates and remains low throughout, allowing for much quicker communication of the Gaussian sample. Results for higher-dimensional Gaussians are provided in Appendix I.

4.2. Categorical distribution

As another example we consider D-dimensional categorical distributions distributed according to a Dirichlet distribution with concentration parameter $\boldsymbol{\alpha}$. We chose $D = 2^{16}$ and $\alpha_i = 3 \cdot 10^{-4}$ for all i, leading to sparse target distributions and a uniform marginal distribution. We include rejection sampling (RS) with an optimal choice for $w_{\min}$ (a different value for each distribution) as well as the greedy rejection sampler (RS*) of Harsha et al. (2007) in the comparison. For each method and target distribution, we simulate 10^5 samples and measure the TVD between the resulting histogram and the target distribution. As a measure of the coding cost, we estimate the entropy of the index distribution obtained by averaging index histograms of 20 different target distributions.

We explore the effects of limiting the number of candidates available to an algorithm. We find that the sample quality of all methods deteriorates quickly as the coding cost drops below the information contained in exact samples (Fig. 2, left). RS performed well in the bit-rate constrained regime but not as well when constraining computational cost (Fig. 2, middle). RS* performed very well in the low bit-rate regime but we point out that its computational complexity is larger by a factor D per iteration compared to the other methods. PFR and ORC performed best for samples of high quality.

Although MRC is the method which is currently the most widely used in machine learning (e.g., Havasi et al., 2019; Flamich et al., 2020; Shah et al., 2022), we find that here it performs worse than the other methods, mostly due to its coding and computational cost growing unboundedly with the number of candidates (Fig. 2, right). ORC addresses this issue such that its coding cost converges to that of the PFR.

5. Discussion

We demonstrated a close connection between minimal random coding (MRC; Havasi et al., 2019), or likelihood encoding (Cuff, 2008; Song et al., 2016), and the Poisson functional representation (PFR; Li & El Gamal, 2018). We introduced ordered random coding (ORC), which occupies a space between the two and benefits from the theoretical guarantees of both. In practice, we found that ORC can significantly outperform MRC, especially as the desired sample quality increases (achieving a 20% coding cost reduction at a TVD of 0.02).

Our second coding scheme enables much more efficient communication of samples from distributions with concentrated support. When the target distributions' support is unbounded, hybrid coding may still be applied after truncation, as in the Gaussian example. While the hybrid scheme is much more efficient than other approaches for the Gaussian example, its cost still grows exponentially with dimensionality. A potential solution may be to generate candidates using more sophisticated lattices than the integer lattice considered here (e.g., Leech, 1967; Zamir, 2014). The Gaussian channel is ubiquitous in machine learning and plays an important role in, for instance, variational autoencoders (Kingma & Welling, 2014) and differential privacy (Dwork et al., 2006). An important task for future research is to characterize other distributions which can be simulated efficiently and which are therefore of great relevance for practical applications.

Acknowledgements

We would like to thank Abhin Shah, Johannes Ballé, Eirikur Agustsson, and Aaron B. Wagner for helpful discussions of the ideas presented in this manuscript.

References

Agustsson, E. and Theis, L. Universally Quantized Neural Compression. In *Advances in Neural Information Processing Systems 33*, 2020.

Ballé, J., Laparra, V., and Simoncelli, E. P. End-to-end Optimized Image Compression. In *International Conference on Learning Representations*, 2017.

Bennett, C. H. and Shor, P. W. Entanglement-Assisted Capacity of a Quantum Channel and the Reverse Shannon Theorem. *IEEE Trans. Info. Theory*, 48(10), 2002.

Braverman, M. and Garg, A. Public vs private coin in bounded-round information. In Esparza, J., Fraigniaud, P., Husfeldt, T., and Koutsoupias, E. (eds.), *Automata, Languages, and Programming*, pp. 502–513. Springer, 2014.

Chatterjee, S. and Diaconis, P. The sample size required in importance sampling. *The Annals of Applied Probability*, 28(2):1099–1135, 2018.

Chen, W.-N., Kairouz, P., and Ozgur, A. Breaking the communication-privacy-accuracy trilemma. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 3312–3324. Curran Associates, Inc., 2020.

Choi, Y., El-Khamy, M., and Lee, J. Variable rate deep image compression with a conditional autoencoder. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

Cover, T. M. and Permuter, H. H. Capacity of coordinated actions. In *2007 IEEE International Symposium on Information Theory*, pp. 2701–2705, 2007. doi: 10.1109/ISIT.2007.4557184.

Cuff, P. Communication requirements for generating correlated random variables. In *2008 IEEE International Symposium on Information Theory*, pp. 1393–1397, 2008.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. Calibrating noise to sensitivity in private data analysis. In Halevi, S. and Rabin, T. (eds.), *Theory of Cryptography*, pp. 265–284. Springer Berlin Heidelberg, 2006.

Flamich, G., Havasi, M., and Hernández-Lobato, J. M. Compressing Images by Encoding Their Latent Representations with Relative Entropy Coding, 2020. Advances in Neural Information Processing Systems 34.

Gumbel, E. J. Statistical Theory of Extreme Values and Some Practical Applications. *U.S. Department of Commerce, National Bureau of Standards*, 33, 1954.

Harsha, P., Jain, R., McAllester, D., and Radhakrishnan, J. The Communication Complexity of Correlation. In *Twenty-Second Annual IEEE Conference on Computational Complexity*, pp. 10–23, 2007.

Havasi, M., Peharz, R., and Hernández-Lobato, J. M. Minimal Random Code Learning: Getting Bits Back from Compressed Model Parameters. In *International Conference on Learning Representations*, 2019.

Hinton, G. E. and Van Camp, D. Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the sixth annual conference on Computational learning theory*, pp. 5–13, 1993.

Kingma, D. and Welling, M. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

Leech, J. Notes on sphere packings. *Canadian Journal of Mathematics*, 19:251–267, 1967.

Li, C. T. and Anantharam, V. A unified framework for one-shot achievability via the poisson matching lemma. *IEEE Transactions on Information Theory*, 67(5):2624–2651, 2021. doi: 10.1109/TIT.2021.3058842.

Li, C. T. and El Gamal, A. Distributed simulation of continuous random variables. *IEEE Transactions on Information Theory*, 63(10):6329–6343, 2017. doi: 10.1109/TIT.2017.2735438.

Li, C. T. and El Gamal, A. Strong Functional Representation Lemma and Applications to Coding Theorems. *IEEE Transactions on Information Theory*, 64(11):6967–6978, 2018. doi: 10.1109/TIT.2018.2865570.

Maddison, C. J. A Poisson process model for Monte Carlo. In Hazan, T., Papandreou, G., and Tarlow, D. (eds.), *Perturbation, Optimization, and Statistics*. MIT Press, 2016.

Maddison, C. J., Tarlow, D., and Minka, T. $A*$ Sampling. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

Rissanen, J. and Langdon, G. G. Arithmetic coding. *IBM Journal of research and development*, 23(2):149–162, 1979.

Roberts, L. G. Picture Coding Using Pseudo-Random Noise. *IRE Transactions on Information Theory*, 1962.

Rényi, A. On the theory of order statistics. *Acta Mathematica Hungarica*, 4:191—231, 1953.

Schuchman, L. Dither signals and their effect on quantization noise. *IEEE Transactions on Communication Technology*, 12(4):162–165, 1964.

Shah, A., Chen, W.-N., Balle, J., Kairouz, P., and Theis, L. Optimal compression of locally differentially private mechanisms. In *Artificial Intelligence and Statistics*, 2022. URL `https://arxiv.org/abs/2111.00092`.

Song, E. C., Cuff, P., and Poor, H. V. The likelihood encoder for lossy compression. *IEEE Transactions on Information Theory*, 62(4):1836–1849, 2016. doi: 10.1109/TIT.2016.2529657.

Sriperumbudur, B. K., Fukumizu, K., Gretton, A., Schölkopf, B., and Lanckriet, G. R. G. On integral probability metrics, ϕ-divergences and binary classification, 2009.

Theis, L. and Agustsson, E. On the advantages of stochastic encoders. In *Neural Compression Workshop at ICLR 2021*, 2021.

Townsend, J., Bird, T., and Barber, D. Practical lossless compression with latent variables using bits back coding. *arXiv preprint arXiv:1901.04866*, 2019.

Wallace, C. S. Classification by minimum-message-length inference. In *International Conference on Computing and Information*, pp. 72–81. Springer, 1990.

Wyner, A. The common information of two dependent random variables. *IEEE Transactions on Information Theory*, 21(2):163–179, 1975. doi: 10.1109/TIT.1975.1055346.

Xu, G., Liu, W., and Chen, B. Wyners common information for continuous random variables - a lossy source coding interpretation. In *45th Annual Conference on Information Sciences and Systems*, pp. 1–6, 2011. doi: 10.1109/CISS.2011.5766249.

Zamir, R. *Lattice Coding for Signals and Networks*. Cambridge University Press, 2014.

Zamir, R. and Feder, M. On universal quantization by randomized uniform/lattice quantizers. *IEEE Transactions on Information Theory*, 38(2):428–436, 1992.

Ziv, J. On universal quantization. *IEEE Transactions on Information Theory*, 31(3):344–347, 1985.

A. Pseudocode for greedy rejection sampling

Algorithm 3 Greedy rejection sampling (RS*; Harsha et al., 2007)

```
Require: $\mathtt{p}, \mathtt{q}$
$n \leftarrow 0$
$\mathtt{q}' \leftarrow 0$                                        ▷ Portion of distribution simulated thus far

repeat
    $n \leftarrow n + 1$
    $z \leftarrow \mathtt{simulate}(n, \mathtt{p})$                ▷ Generate candidate
    $\mathtt{p}' \leftarrow (1 - \mathtt{sum}(\mathtt{q}')) \cdot \mathtt{p}$          ▷ Scaled proposal distribution
    $\mathtt{a} \leftarrow \mathtt{min}(1, (\mathtt{q} - \mathtt{q}')/\mathtt{p}')$        ▷ Acceptance probability
    $\mathtt{q}' \leftarrow \mathtt{q}' + \mathtt{a} \cdot \mathtt{p}'$
until $\mathtt{uniform}(n) < \mathtt{a}(z)$                       ▷ Accept/reject candidate

return $n$
```

Algorithm 3 (RS*) contains pseudocode for the approach taken by Harsha et al. (2007) to prove a one-shot achievability result for the coding cost of reverse channel coding, the "one-shot reverse Shannon theorem". Our algorithm computes slightly different intermediate results and also differs in notation from the proofs of Harsha et al. (2007). For pseudocode closer in style to the paper, see Appendix A of Havasi et al. (2019).

The algorithm effectively divides a distribution into slices and each iteration attempts to sample from one of these slices. The success probability in each iteration is proportional to the probability mass contained in a slice.

For an intuitive understanding of the algorithm, assume that $\mathtt{p}$ and $\mathtt{q}$ are vectors representing categorical proposal and target distributions, respectively. At the beginning of an iteration, $\mathtt{q}'$ represents the portion of the target distribution which we have already attempted to sample from (a sum of previously considered slices). We always have $\mathtt{q}'(z) \leq \mathtt{q}(z)$ and $\mathtt{sum}(\mathtt{q}')$ is the probability of having accepted a candidate by the current iteration. $\mathtt{p}'(z)$ is thus the probability of reaching the nth iteration *and* producing the candidate z. The vector $\mathtt{a}$ provides an acceptance probability for each value of the candidate so that $\mathtt{a} \cdot \mathtt{p}'$ gives the probability of reaching the nth iteration, sampling a candidate value, and accepting it.

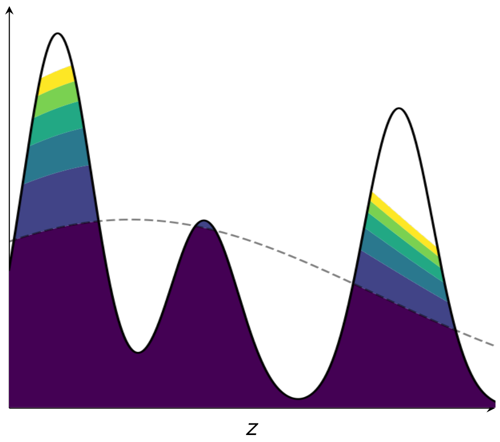

Figure 3. Visualization of 6 iterations of RS*. The solid line corresponds to the target distribution $\mathtt{q}$ while the dashed line indicates the proposal distribution $\mathtt{p}$. Shaded regions correspond to $\mathtt{a} \cdot \mathtt{p}'$.

The main difference between the algorithm described above and rejection sampling is in the calculation of the acceptance probability. If we used the acceptance probability

$$\mathtt{a} \leftarrow \mathtt{min}(1, w_{\text{min}} \cdot (\mathtt{q} - \mathtt{q}')/\mathtt{p}') \tag{48}$$

instead, the algorithm would already reduce to rejection sampling. Note that in this case $\mathtt{min}$ would not be needed since w_{min} already makes sure that the acceptance probabilities do not exceed 1. Not using w_{min} allows RS* to accept candidates with higher probability. The algorithm corrects for this "greedy" approach by targeting $\mathtt{q} - \mathtt{q}'$ instead of $\mathtt{q}$ in each iteration. The algorithm is visualized in Figure 3.

Note that RS* requires integration and pointwise multiplication of vectors or functions where other algorithms only require evaluation of densities at a single point. In practice, this means RS* is computationally more demanding and more difficult to implement, especially for continuous distributions.

B. Pseudocode for ordered random coding

Algorithm 4 Ordered random coding (ORC)

```
Require: $\mathtt{p}, \mathtt{q}, w_{\text{min}}, N$
 1: $t, n, s^* \leftarrow 0, 1, \infty$

 2: repeat
 3:     $z \leftarrow \mathtt{simulate}(n, \mathtt{p})$                ▷ Candidate generation

 4:     $v \leftarrow N/(N - n + 1)$
 5:     $t \leftarrow t + v \cdot \mathtt{expon}(n, 1)$                ▷ Candidate scoring
 6:     $s \leftarrow t \cdot \mathtt{p}(z)/\mathtt{q}(z)$

 7:     if $s < s^*$ then                                 ▷ Accept/reject candidate
 8:         $s^* \leftarrow s$
 9:         $n^* \leftarrow n$
10:     end if

11:     $n \leftarrow n + 1$
12: until $s^* \leq t \cdot w_{\text{min}}$ or $n > N$

13: return $n^*$
```

Algorithm 4 contains pseudocode for ordered random coding. In contrast to the Poisson functional representation, the algorithm considers only a finite number of candidates N, which means that it can still be used in practice when $w_{\text{min}} = 0$ (either because the density ratio is unbounded or a bound is unknown). Another difference is that the exponential variables are weighted.

We note that for better numerical accuracy, log-density ratios and log-sum-exp operations should be used in practice where the pseudocode uses density ratios and sums (lines 5 and 6).

C. Proof of Theorem 3.1

Theorem 3.1. *Let $\mathbf{Z}_n \sim p$ and $G_n \sim \text{Gumbel}(0,1)$ for $n \in \{1, \dots, N\}$ (i.i.d.). Let $\tilde{G}_n$ be the result of sorting the random variables in decreasing order such that $\tilde{G}_1 \geq \dots \geq \tilde{G}_N$. We further define*

$$N^* = \operatorname*{argmax}_{n \leq N} \log \frac{q_{\mathbf{x}}(\mathbf{Z}_n)}{p(\mathbf{Z}_n)} + G_n, \qquad \tilde{N}^* = \operatorname*{argmax}_{n \leq N} \log \frac{q_{\mathbf{x}}(\mathbf{Z}_n)}{p(\mathbf{Z}_n)} + \tilde{G}_n. \tag{49}$$

Then $\mathbf{Z}_{N^*} \sim \mathbf{Z}_{\tilde{N}^*}$.

Proof. First, let us define the functions

$$n^*(\mathbf{z}_1, \dots, \mathbf{z}_N, g_1, \dots, g_N) = \text{argmax}_n \log \frac{q_{\mathbf{x}}(\mathbf{z}_n)}{p(\mathbf{z}_n)} + g_n \tag{50}$$

and

$$z(\mathbf{z}_1, \dots, \mathbf{z}_N, g_1, \dots, g_N) = \mathbf{z}_{n^*(\mathbf{z}_1, \dots, \mathbf{z}_N, g_1, \dots, g_N)}. \tag{51}$$

It is not difficult to see that

$$z(\mathbf{z}_{\sigma(1)}, \dots, \mathbf{z}_{\sigma(N)}, g_{\sigma(1)}, \dots, g_{\sigma(N)}) = z(\mathbf{z}_1, \dots, \mathbf{z}_N, g_1, \dots, g_N) \tag{52}$$

for any permutation σ of the indices $1, \dots, N$. Also note that since the candidates are i.i.d. and therefore exchangeable, permuting the candidates does not change their distribution,

$$\mathbf{Z}_1, \dots, \mathbf{Z}_N \sim \mathbf{Z}_{\sigma(1)}, \dots, \mathbf{Z}_{\sigma(N)} \sim \prod_n p, \tag{53}$$

and that this is true for any permutation that is independent of $\mathbf{Z}_1, \dots, \mathbf{Z}_n$ even if the permutation depends on the G_n. We therefore have

$$z(\mathbf{Z}_{\sigma(1)}, \dots, \mathbf{Z}_{\sigma(N)}, G_1, \dots, G_N) \sim z(\mathbf{Z}_1, \dots, \mathbf{Z}_N, G_1, \dots, G_N). \tag{54}$$

Choose $\tilde{\sigma}$ such that $\tilde{G}_n = G_{\tilde{\sigma}(n)}$. Then

$$\begin{aligned}
\mathbf{Z}_{\tilde{N}^*} &= z(\mathbf{Z}_1, \dots, \mathbf{Z}_N, G_{\tilde{\sigma}(1)}, \dots, G_{\tilde{\sigma}(N)}) && (55)\\
&= z(\mathbf{Z}_{\tilde{\sigma}^{-1}(1)}, \dots, \mathbf{Z}_{\tilde{\sigma}^{-1}(N)}, G_1, \dots, G_N) && (56)\\
&\sim z(\mathbf{Z}_1, \dots, \mathbf{Z}_N, G_1, \dots, G_N) && (57)\\
&= \mathbf{Z}_{N^*}. && (58)
\end{aligned}$$

□

D. Proof of Corollary 3.2

We first prove the following result which follows from the results of Havasi et al. (2019) and Chatterjee & Diaconis (2018) and does not assume bounded density ratios.

Lemma D.1. *Let $\tilde{q}_{\mathbf{x}}$ be the distribution of $\mathbf{Z}_{\tilde{N}^*}$ where $\tilde{N}^*$ is defined as in Theorem 3.1. If the number of candidates is $N = 2^{D_{\text{KL}}[q_{\mathbf{x}} \,\|\, p]+t}$ for some $t \geq 0$ and $\mathbf{Z} \sim q_{\mathbf{x}}$, then*

$$D_{\text{TV}}[\tilde{q}, q] \leq 4\epsilon \tag{59}$$

where

$$\epsilon = \left(2^{-t/4} + 2\sqrt{\mathbb{P}\left(\log \frac{q_{\mathbf{x}}(\mathbf{Z})}{p(\mathbf{Z})} > D_{\text{KL}}[q_{\mathbf{x}} \,\|\, p] + t/2\right)}\right)^{\frac{1}{2}}. \tag{60}$$

Proof. If $\varepsilon \geq 1/4$ then Eq. 59 is automatically true and there is nothing left to show. Assume therefore that q, p, and t are such that $\varepsilon < 1/4$.

Since $\mathbf{Z}_{\tilde{N}^*} \sim \mathbf{Z}_{N^*}$ by Theorem 3.1, $\tilde{q}_{\mathbf{x}}$ is also the distribution of $\mathbf{Z}_{N^*}$. Let $\Omega = \{\mathbf{Z}_1, \dots, \mathbf{Z}_N\}$ be the set of candidates and let $\tilde{q}_{\mathbf{x},\Omega}$ be the distribution of $\mathbf{Z}_{N^*}$ for a fixed set of candidates. That is,

$$\tilde{q}_{\mathbf{x},\Omega}(\mathbf{z}) = \sum_{n=1}^{N} \pi(n)\delta(\mathbf{z} - \mathbf{Z}_n) \tag{61}$$

where $\pi(n) \propto q_{\mathbf{x}}(\mathbf{Z}_n)/p(\mathbf{Z}_n)$. Theorem 3.2 of Havasi et al. (2019) tells us that

$$\mathbb{P}\left(\left|\mathbb{E}_{q_{\mathbf{x},\Omega}}[f(\tilde{\mathbf{Z}})] - \mathbb{E}_{q_{\mathbf{x}}}[f(\mathbf{Z})]\right| \geq 2\|f\|_q \frac{\epsilon}{1-\epsilon}\right) < 2\epsilon, \tag{62}$$

for any measurable function f, where $\|f\|_q = \sqrt{\mathbb{E}_{q_{\mathbf{x}}}[f(\mathbf{Z})^2]}$ and the probability arises due to the randomness in the set of candidates Ω. We choose

$$f(\mathbf{z}) = \begin{cases} 1 & \text{if } \tilde{q}_{\mathbf{x}}(\mathbf{z}) > q_{\mathbf{x}}(\mathbf{z}), \\ -1 & \text{else.} \end{cases} \tag{63}$$

For notational convenience, we further define the event

$$A = \left[\!\left[\left|\mathbb{E}_{q_{\mathbf{x},\Omega}}[f(\tilde{\mathbf{Z}})] - \mathbb{E}_{q_{\mathbf{x}}}[f(\mathbf{Z})]\right| \geq \|f\|_q \frac{2\epsilon}{1-\epsilon}\right]\!\right], \tag{64}$$

where $[\![\cdot]\!]$ is 1 if its argument is true and 0 otherwise. We have

$$\begin{aligned}
2D_{\mathrm{TV}}[\tilde{q}, q] &= |\mathbb{E}_{\tilde{q}}[f(\tilde{\mathbf{Z}})] - \mathbb{E}_{q_{\mathbf{x}}}[f(\mathbf{Z})]| && (65)\\
&= |\mathbb{E}_{\Omega}[\mathbb{E}_{\tilde{q}_{\mathbf{x},\Omega}}[f(\tilde{\mathbf{Z}})]] - \mathbb{E}_{q_{\mathbf{x}}}[f(\mathbf{Z})]| && (66)\\
&\leq \mathbb{E}_{\Omega}[|\mathbb{E}_{\tilde{q}_{\mathbf{x},\Omega}}[f(\tilde{\mathbf{Z}})] - \mathbb{E}_{q_{\mathbf{x}}}[f(\mathbf{Z})]|] && (67)\\
&= P(A=1)\mathbb{E}_{\Omega}[|\mathbb{E}_{\tilde{q}_{\mathbf{x},\Omega}}[f(\tilde{\mathbf{Z}})] - \mathbb{E}_{q_{\mathbf{x}}}[f(\mathbf{Z})]| \mid A=1] + P(A=0)\mathbb{E}_{\Omega}[|\mathbb{E}_{\tilde{q}_{\mathbf{x},\Omega}}[f(\tilde{\mathbf{Z}})] - \mathbb{E}_{q_{\mathbf{x}}}[f(\mathbf{Z})]| \mid A=0] && (68)\\
&\leq 2P(A=1) + (1 - P(A=1))\tfrac{2\epsilon}{1-\epsilon}\|f\|_q && (69)\\
&= 2P(A=1)\left(1 - \tfrac{\epsilon}{1-\epsilon}\right) + \tfrac{2\epsilon}{1-\epsilon} && (70)\\
&\leq 4\epsilon\left(1 - \tfrac{\epsilon}{1-\epsilon}\right) + \tfrac{2\epsilon}{1-\epsilon} && (71)\\
&\leq 4\epsilon + 4\epsilon && (72)\\
&= 8\epsilon, && (73)
\end{aligned}$$

where Eq. 65 is a known identity (Sriperumbudur et al., 2009), Eq. 67 follows from Jensen's inequality, Eq. 69 follows from the definitions of f and A, Eq. 70 follows from $\|f(\mathbf{z})\|_q = 1$, and Eq. 71 follows from Eq. 62. □

Corollary 3.2. *Let $\tilde{q}_{\mathbf{x}}$ be the distribution of $\mathbf{Z}_{\tilde{N}^*}$ where $\tilde{N}^*$ is defined as in Theorem 3.1. If the number of candidates is $N = 2^{D_{\mathrm{KL}}[q_{\mathbf{x}} \,\|\, p]+t}$ for some $t \geq 2e^{-1}\log e$ and $p(\mathbf{z})/q_{\mathbf{x}}(\mathbf{z}) \geq w_{\min} > 0$ for all $\mathbf{z}$, then*

$$D_{\mathrm{TV}}[\tilde{q}_{\mathbf{x}}, q_{\mathbf{x}}] \leq 4\varepsilon \tag{74}$$

where with $B = -\log w_{\min}$ we have

$$\epsilon \leq 2^{-t/8} + \sqrt{2}\exp\left(-\frac{1}{4B^2}(t/2 - e^{-1}\log e)^2\right). \tag{75}$$

Proof. By Lemma D.1, we have

$$D_{\mathrm{TV}}[\tilde{q}_{\mathbf{x}}, q_{\mathbf{x}}] \leq 4\epsilon \tag{76}$$

where

$$\epsilon = \left(2^{-t/4} + 2\sqrt{\mathbb{P}\left(\log \frac{q_{\mathbf{x}}(\mathbf{Z})}{p(\mathbf{Z})} > D_{\mathrm{KL}}[q_{\mathbf{x}} \parallel p] + t/2\right)}\right)^{\frac{1}{2}}. \tag{77}$$

To prove our claim we need to bound ϵ. Define

$$l(\mathbf{z}) = \max(0, \log q_{\mathbf{x}}(\mathbf{z}) - \log p(\mathbf{z})). \tag{78}$$

By Claim A.2 of Harsha et al. (2007), we have

$$\mathbb{E}_{q_{\mathbf{x}}}[l(\mathbf{Z})] = \mathbb{E}_{q_{\mathbf{x}}}\left[\max\left(0, \log \frac{q_{\mathbf{x}}(\mathbf{Z})}{p(\mathbf{Z})}\right)\right] = \mathbb{E}_{q_{\mathbf{x}}}\left[\log \frac{q_{\mathbf{x}}(\mathbf{Z})}{p(\mathbf{Z})} - \min\left(0, \log \frac{q_{\mathbf{x}}(\mathbf{Z})}{p(\mathbf{Z})}\right)\right] \leq D_{\mathrm{KL}}[q_{\mathbf{x}} \parallel p] + e^{-1}\log e. \tag{79}$$

Let $B = -\log w_{\text{min}}$ so that $l(\mathbf{z}) \leq B$ for all $\mathbf{z}$. We have

$$\begin{aligned}
\mathbb{P}\left(\log \frac{q_{\mathbf{x}}(\mathbf{Z})}{p(\mathbf{Z})} > D_{\mathrm{KL}}[q_{\mathbf{x}} \parallel p] + t/2\right) &= \mathbb{P}\left(l(\mathbf{Z}) > D_{\mathrm{KL}}[q_{\mathbf{x}} \parallel p] + t/2\right) &(80)\\
&= \mathbb{P}\left(l(\mathbf{Z}) - \mathbb{E}_{q_{\mathbf{x}}}[l(\mathbf{Z})] > D_{\mathrm{KL}}[q_{\mathbf{x}} \parallel p] - \mathbb{E}_{q_{\mathbf{x}}}[l(\mathbf{Z})] + t/2\right) &(81)\\
&\leq \exp\left(-(D_{\mathrm{KL}}[q_{\mathbf{x}} \parallel p] - \mathbb{E}_{q_{\mathbf{x}}}[l(\mathbf{Z})] + t/2)^2/B^2\right) &(82)\\
&\leq \exp\left(-(t/2 - e^{-1}\log e)^2/B^2\right) &(83)
\end{aligned}$$

where the first equality follows from the non-negativity of the KL divergence, Eq. 82 follows from Hoeffding's inequality, and the last inequality follows from Eq. 79 and $t/2 \geq e^{-1}\log e$. Thus, we have

$$\begin{aligned}
\epsilon &\leq \left(2^{-t/4} + 2\exp\left(-\frac{1}{2B^2}(t/2 - e^{-1}\log e)^2\right)\right)^{\frac{1}{2}} &(84)\\
&\leq 2^{-t/8} + \sqrt{2}\exp\left(-\frac{1}{4B^2}(t/2 - e^{-1}\log e)^2\right) &(85)
\end{aligned}$$

where the second inequality follows from $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$. □

E. Proof of Theorem 3.3

Theorem 3.3. *Let S_n be exponentially distributed RVs and $\mathbf{Z}_n \sim p$ for all $n \in \mathbb{N}$ (i.i.d.), and let*

$$T_n = \sum_{m=1}^{n} S_m, \qquad \tilde{T}_{N,n} = \sum_{m=1}^{n} \frac{N}{N-m+1} S_m \tag{86}$$

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

$$N^*_{\text{PFR}} = \operatorname*{argmin}_{n \in \mathbb{N}} T_n \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)}, \tag{87}$$

$$N^*_{\text{ORC}} = \operatorname*{argmin}_{n \leq N} \tilde{T}_{N,n} \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)}. \tag{88}$$

Then $N^*_{\text{ORC}} \leq N^*_{\text{PFR}}$. Further, if N^*_{PFR} is finite then there exists an $M \in \mathbb{N}$ such that for all $N \geq M$ we have $N^*_{\text{ORC}} = N^*_{\text{PFR}}$.

Proof. We first show that $N^*_{\text{ORC}} \leq N^*_{\text{PFR}}$. For any $\Delta \geq 0$ and n with $n + \Delta \leq N$ we have

$$\begin{aligned}
\frac{\tilde{T}_{N,n+\Delta}}{\tilde{T}_{N,n}} &= \frac{\sum_{m=1}^{n+\Delta} \frac{N}{N-m+1} S_m}{\sum_{m=1}^{n} \frac{N}{N-m+1} S_m} && (89)\\
&= \frac{\sum_{m=1}^{n+\Delta} \frac{N-n+1}{N-m+1} S_m}{\sum_{m=1}^{n} \frac{N-n+1}{N-m+1} S_m} && (90)\\
&= \frac{\sum_{m=1}^{n} \frac{N-n+1}{N-m+1} S_m + \sum_{m=n+1}^{n+\Delta} \frac{N-n+1}{N-m+1} S_m}{\sum_{m=1}^{n} \frac{N-n+1}{N-m+1} S_m} && (91)\\
&\geq \frac{\sum_{m=1}^{n} \frac{N-n+1}{N-m+1} S_m + \sum_{m=n+1}^{n+\Delta} S_m}{\sum_{m=1}^{n} \frac{N-n+1}{N-m+1} S_m} && (92)\\
&\geq \frac{\sum_{m=1}^{n} S_m + \sum_{m=n+1}^{n+\Delta} S_m}{\sum_{m=1}^{n} S_m} && (93)\\
&= \frac{T_{n+\Delta}}{T_n} && (94)
\end{aligned}$$

That is, $\tilde{T}_{N,n}$ increases at a faster rate than T_n. Eq. 92 is true because $S_m \geq 0$ and $(N - n + 1)/(N - m + 1) > 1$ for $m \geq n + 1$. Eq. 93 is true because $(N - n + 1)/(N - m + 1) \leq 1$ for $m \leq n$ and

$$f : [0, \infty) \to [0, \infty), \quad t \mapsto \frac{t + a}{t} \tag{95}$$

is a monotonically decreasing function for any $a \geq 0$.

Now assume that N^*_{PFR} takes on some value $n^* \in \mathbb{N}$. If $n^* \geq N$ then $N^*_{\text{ORC}} \leq N^*_{\text{PFR}}$ since $N^*_{\text{ORC}} \leq N$. If $n^* < N$, to show that N^*_{ORC} does not exceed N^*_{PFR} it is enough to show that[1]

$$\tilde{T}_{N,n^*} \frac{p(\mathbf{Z}_{n^*})}{q_{\mathbf{x}}(\mathbf{Z}_{n^*})} \leq \tilde{T}_{N,n^*+\Delta} \frac{p(\mathbf{Z}_{n^*+\Delta})}{q_{\mathbf{x}}(\mathbf{Z}_{n^*+\Delta})} \tag{96}$$

for any $\Delta > 0$ with $n^* + \Delta \leq N$, since N^*_{ORC} then must be either n^* or take on a smaller value. By definition of N^*_{PFR}, we have

$$T_{n^*} \frac{p(\mathbf{Z}_{n^*})}{q_{\mathbf{x}}(\mathbf{Z}_{n^*})} \leq T_{n^*+\Delta} \frac{p(\mathbf{Z}_{n^*+\Delta})}{q_{\mathbf{x}}(\mathbf{Z}_{n^*+\Delta})} \tag{97}$$

for any $\Delta > 0$. From this and Eqs. 89 to 94 it follows that

$$\begin{aligned}
\frac{p(\mathbf{Z}_{n^*})}{q_{\mathbf{x}}(\mathbf{Z}_{n^*})} &\leq \frac{T_{n^*+\Delta}}{T_{n^*}} \frac{p(\mathbf{Z}_{n^*+\Delta})}{q_{\mathbf{x}}(\mathbf{Z}_{n^*+\Delta})}, && (98)\\
\frac{p(\mathbf{Z}_{n^*})}{q_{\mathbf{x}}(\mathbf{Z}_{n^*})} &\leq \frac{\tilde{T}_{N,n^*+\Delta}}{\tilde{T}_{N,n^*}} \frac{p(\mathbf{Z}_{n^*+\Delta})}{q_{\mathbf{x}}(\mathbf{Z}_{n^*+\Delta})}, && (99)\\
\tilde{T}_{N,n} \frac{p(\mathbf{Z}_{n^*})}{q_{\mathbf{x}}(\mathbf{Z}_{n^*})} &\leq \tilde{T}_{N,n^*+\Delta} \frac{p(\mathbf{Z}_{n^*+\Delta})}{q_{\mathbf{x}}(\mathbf{Z}_{n^*+\Delta})}, && (100)\\
& && (101)
\end{aligned}$$

which concludes the proof of $N^*_{\text{ORC}} \leq N^*_{\text{PFR}}$.

Next, we show that $N^*_{\text{ORC}} = N^*_{\text{PFR}}$ for large enough N. First note that we can equivalently define N^*_{ORC} as follows,

$$N' = \min\{N^*_{\text{PFR}}, N\}, \qquad N^*_{\text{ORC}} = \operatorname*{argmin}_{n \leq N'} \tilde{T}_{N,n} \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)}. \tag{102}$$

[1] We assume that in case of a tie, argmin returns the smaller index.

We have

$$\lim_{N\to\infty} \tilde{T}_{N,n} = \sum_{m=1}^{n} \left(\lim_{N\to\infty} \frac{N}{N-m+1} \right) S_m = T_n \tag{103}$$

for all $n \leq N^*_{\text{PFR}}$ and therefore

$$\begin{aligned}
\lim_{N\to\infty} N^*_{\text{ORC}} &= \lim_{N\to\infty} \operatorname*{argmin}_{n\leq N'} \tilde{T}_{N,n} \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)} && (104)\\
&= \operatorname*{argmin}_{n\leq N'} \lim_{N\to\infty} \tilde{T}_{N,n} \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)} && (105)\\
&= \operatorname*{argmin}_{n\leq N^*_{\text{PFR}}} \lim_{N\to\infty} \tilde{T}_{N,n} \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)} && (106)\\
&= \operatorname*{argmin}_{n\leq N^*_{\text{PFR}}} T_n \frac{p(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)} && (107)\\
&= N^*_{\text{PFR}}. && (108)
\end{aligned}$$

Hence, since N^*_{PFR} is assumed to be finite, there is an $M \in \mathbb{N}$ such that $N^*_{\text{ORC}} = N^*_{\text{PFR}}$ for $N \geq M$. □

F. Proof of Corollary 3.4

Corollary 3.4. *Let $C = \mathbb{E}_{\mathbf{X}}[D_{\text{KL}}[q_{\mathbf{X}} \parallel p]]$ and let N^*_{ORC} be defined as in Theorem 3.3. Then*

$$H[N^*_{\text{ORC}}] < C + \log(C+1) + 4. \tag{109}$$

Proof. Let N^*_{PFR} be defined as in Theorem 3.3. Li & El Gamal (2018, Appendix A) showed that

$$\mathbb{E}[\log N^*_{\text{PFR}} \mid \mathbf{X} = \mathbf{x}] \leq D_{\text{KL}}[q_{\mathbf{x}} \parallel p] + e^{-1}\log e + 1. \tag{110}$$

While Li & El Gamal (2018) were only considering the case where $p(\mathbf{z}) = \mathbb{E}_{\mathbf{X}}[q_{\mathbf{X}}(\mathbf{z})]$, their proof of the above statement does not make use of this assumption. Since $N^*_{\text{ORC}} \leq N^*_{\text{PFR}}$, we also have

$$\mathbb{E}[\log N^*_{\text{ORC}}] \leq \mathbb{E}[\log N^*_{\text{PFR}}] \leq \mathbb{E}_{\mathbf{X}}[D_{\text{KL}}[q_{\mathbf{X}} \parallel p]] + e^{-1}\log e + 1. \tag{111}$$

Li & El Gamal (2018, Appendix B) further showed that for any random variable N^* with values in $\mathbb{N}$, we have

$$\mathbb{E}[-\log p_\lambda(N^*)] \leq \mathbb{E}[\log N^*] + \log(\mathbb{E}[\log N^*] + 1) + 1, \tag{112}$$

where $p_\lambda(n) \propto n^{-\lambda}$ is a Zipf distribution with

$$\lambda = 1 + 1/\mathbb{E}[\log N^*]. \tag{113}$$

Applied to N^*_{ORC} and using Eq. 111, we get

$$\begin{aligned}
H[N^*_{\text{ORC}}] &\leq \mathbb{E}[-\log p_\lambda(N^*_{\text{ORC}})] && (114)\\
&\leq \mathbb{E}[\log N^*_{\text{ORC}}] + \log(\mathbb{E}[\log N^*_{\text{ORC}}] + 1) + 1 && (115)\\
&\leq C + \log(C + e^{-1}\log e + 2) + e^{-1}\log e + 2 && (116)\\
&< C + \log(C+1) + 4. && (117)
\end{aligned}$$

□

G. Proof of Theorem 3.5

Let us first briefly repeat the relevant definitions from the main text. We have

$$r_{\mathbf{x}}(\mathbf{z}) = \begin{cases} 1 & \text{if } \mathbf{z} \in \mathbf{c}_{\mathbf{x}} + [-0.5, 0.5)^D, \\ 0 & \text{else,} \end{cases} \tag{118}$$

where $\mathbf{c}_{\mathbf{x}}$ is chosen such that the support of $r_{\mathbf{x}}$ is contained within $[0, M_1) \times \cdots [0, M_D)$. Candidates are generated via dithered quantization

$$\mathbf{U}_n \sim \text{Uniform}([0,1)^D), \qquad \mathbf{K}_n = \lfloor \mathbf{c}_{\mathbf{x}} - \mathbf{U}_n \rceil, \qquad \mathbf{Z}_n = \mathbf{K}_n + \mathbf{U}_n, \tag{119}$$

so that $\mathbf{Z}_n \sim r_{\mathbf{x}}$. One of the candidates is then selected according to

$$\tilde{T}_{N,n} = \sum_{m=1}^{n} \frac{N}{N - m + 1} S_m, \qquad N^* = \operatorname*{argmin}_{n \leq N} \tilde{T}_{N,n} \frac{r_{\mathbf{x}}(\mathbf{Z}_n)}{q_{\mathbf{x}}(\mathbf{Z}_n)}, \tag{120}$$

where the support of $q_{\mathbf{x}}$ is assumed to be contained in the support of $r_{\mathbf{x}}$. For notational convenience, further define

$$\mathbf{K}^* = \mathbf{K}_{N^*}, \qquad \mathbf{U}^* = \mathbf{U}_{N^*}, \qquad \mathbf{Z}^* = \mathbf{Z}_{N^*}. \tag{121}$$

Note that $\mathbf{Z}^* = \mathbf{K}^* + \mathbf{U}^*$. The following theorem bounds the coding cost of optimally encoding N^* and $\mathbf{K}^*$.

Theorem 3.5. *Let N^* and $\mathbf{K}^*$ be defined as in Eqs. 120 and 121 and let p be the uniform distribution over $[0, M_1) \times \cdots \times [0, M_D)$ for some $M_i \in \mathbb{N}$. Then*

$$H[N^*, \mathbf{K}^*] < C + \log(C - \textstyle\sum_i \log M_i + 1) + 4,$$

where $C = \mathbb{E}_{\mathbf{X}}[D_{\text{KL}}[q_{\mathbf{X}} \parallel p]]$.

Proof. By Corollary 3.4, we have

$$H[N^*] < C' + \log(C' + 1) + 4 \tag{122}$$

where $C' = \mathbb{E}_{\mathbf{X}}[D_{\text{KL}}[q_{\mathbf{X}} \parallel r_{\mathbf{X}}]] = C - \sum_i \log M_i$, assuming the support of $q_{\mathbf{x}}$ is contained in the support of $r_{\mathbf{x}}$.

Next consider the coding cost of $\mathbf{K}^*$. Note that for each entry K_i^* in $\mathbf{K}^*$ we must have $0 \leq K_i^* < M_i$ since otherwise $K_i^* + U_i^* < 0$ or $K_i^* + U_i^* \geq M_i$, that is, $\mathbf{Z}^* = \mathbf{K}^* + \mathbf{U}^*$ would be outside the support of $r_{\mathbf{X}}$. Hence,

$$\begin{aligned} H[\mathbf{K}^*] &\leq \sum_i \log M_i && (123) \\ &= \log \frac{1}{p(\mathbf{z})} && (124) \\ &= \mathbb{E}_{\mathbf{X}}\left[\mathbb{E}_{\mathbf{Z} \sim q_{\mathbf{X}}}\left[\log \frac{r_{\mathbf{X}}(\mathbf{Z})}{p(\mathbf{Z})}\right]\right] && (125) \\ &= \mathbb{E}_{\mathbf{X}}\left[\mathbb{E}_{\mathbf{Z} \sim q_{\mathbf{X}}}\left[\log \frac{q_{\mathbf{X}}(\mathbf{Z}) r_{\mathbf{X}}(\mathbf{Z})}{p(\mathbf{Z}) q_{\mathbf{X}}(\mathbf{Z})}\right]\right] && (126) \\ &= C - C'. && (127) \end{aligned}$$

Taken together, we have

$$H[N^*, \mathbf{K}^*] \leq H[N^*] + H[\mathbf{K}^*] < C' + \log(C' + 1) + 4 + C - C' = C + \log(C - \sum_i \log M_i + 1) + 4, \tag{128}$$

proving the claim. □

H. Hybrid coding for Gaussian distributions

Let $\tilde{q}_{\mathbf{x}}$ be a truncated Gaussian with mean $\mathbf{x}$ and covariance $\mathbf{I}$ and let θ be the fraction of mass which has been truncated. We assume that the mean is itself Gaussian distributed with covariance $\sigma^2\mathbf{I}$ so that a Gaussian with covariance $(\sigma^2+1)\mathbf{I}$ is a suitable candidate generating distribution $\tilde{p}$.

To make the marginal distribution uniform, we first transform each coordinate with the CDF of a univariate Gaussian with variance σ^2+1, Φ_{σ^2+1}. After this transformation, different target distributions have supports of varying widths. The distribution with the widest support is centered at zero. The support of the truncated Gaussian is limited to the left and right by

$$a = \Phi^{-1}(\theta'/2), \qquad b = \Phi^{-1}(1-\theta'/2), \tag{129}$$

along each coordinate, where $\theta' = 1-(1-\theta)^{1/D}$ and Φ is the CDF of a standard normal. After the transformation, the limits of the support become $\Phi_{\sigma^2+1}(a)$ and $\Phi_{\sigma^2+1}(b)$, respectively, so that we can scale the distributions by

$$M = \left\lfloor \frac{1}{\Phi_{\sigma^2+1}(b) - \Phi_{\sigma^2+1}(a)} \right\rfloor \tag{130}$$

along the ith coordinate while still ensuring that the distributions fit into a unit interval. For $D=1$, the target distribution becomes

$$q_x(z) = \frac{\tilde{q}_x(\tilde{z})}{M\Phi'_{\sigma^2+1}(\tilde{z})} = \frac{1}{M}\frac{\tilde{q}_x(\tilde{z})}{\mathcal{N}(\tilde{z};0,\sigma^2+1)} \tag{131}$$

where $\tilde{z} = \Phi^{-1}_{\sigma^2+1}(z/M)$ and $z \in [0, M)$. This is illustrated in Figure 1. For $D > 1$, we have

$$q_{\mathbf{x}}(\mathbf{z}) = \prod_i q_{x_i}(z_i). \tag{132}$$

For w_{min}, we choose

$$\inf_{\mathbf{z}} \frac{p(\mathbf{z})}{q_{\mathbf{x}}(\mathbf{z})} = \inf_{\tilde{\mathbf{z}}} \frac{\tilde{p}(\tilde{\mathbf{z}})}{\tilde{q}_{\mathbf{x}}(\tilde{\mathbf{z}})} \geq \inf_{\tilde{\mathbf{z}}} \frac{(1-\theta)\mathcal{N}(\tilde{\mathbf{z}};0,(\sigma^2+1)\mathbf{I})}{\mathcal{N}(\tilde{\mathbf{z}};\mathbf{x},\mathbf{I})} = (1-\theta)\frac{\mathcal{N}(\tilde{\mathbf{z}}_{\text{min}};0,(\sigma^2+1)\mathbf{I})}{\mathcal{N}(\tilde{\mathbf{z}}_{\text{min}};\mathbf{x},\mathbf{I})} = w_{\text{min}} \tag{133}$$

where

$$\tilde{\mathbf{z}}_{\text{min}} = \frac{\sigma^2+1}{\sigma^2}\mathbf{x}. \tag{134}$$

is the minimizer of the infimum on the right-hand side. Since the density ratios are invariant under transformation, we can use the same w_{min} for ORC/PFR and the hybrid coding scheme.

I. Additional results

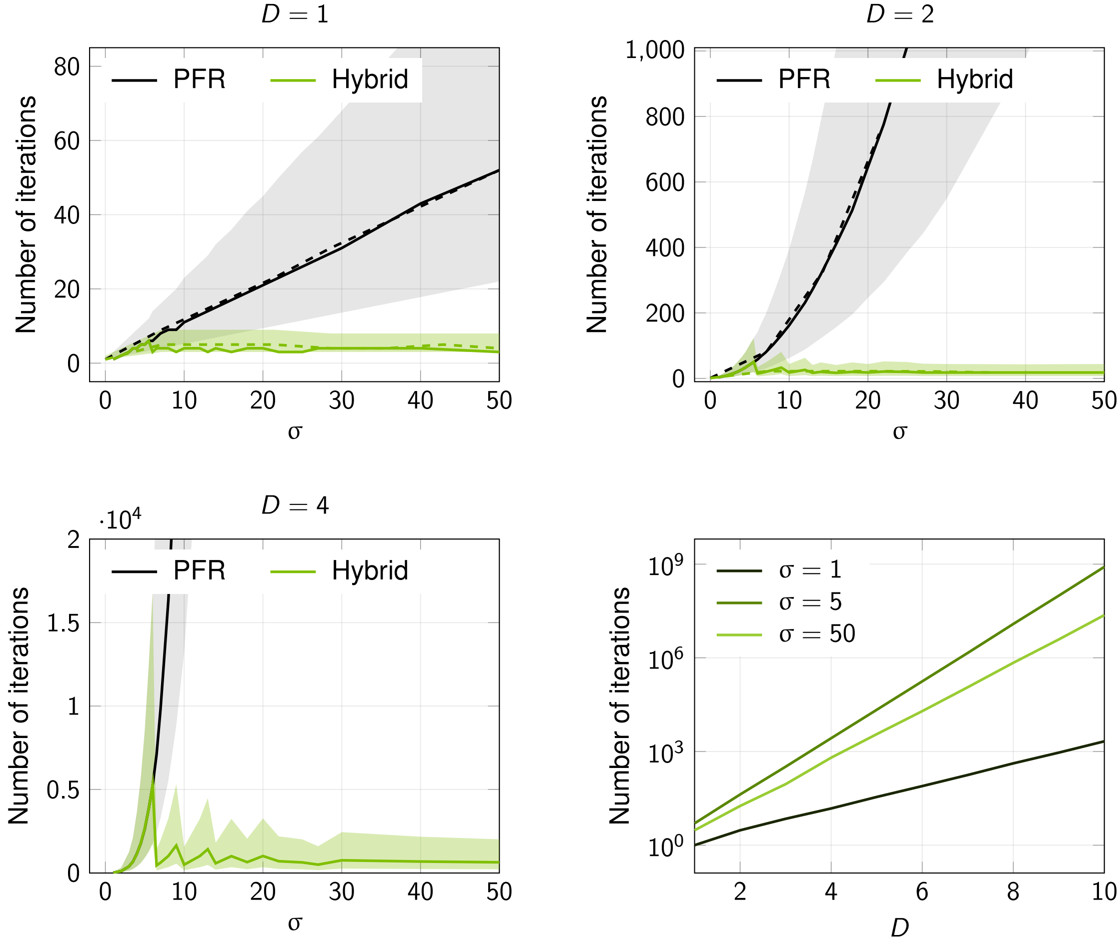

Figure 4. Additional results comparing the computational cost of the hybrid coding scheme with PFR/ORC on multivariate Gaussian distributions (25th, median, and 75th percentile). Dashed lines correspond to empirical performance measured by running the full algorithm while solid lines were obtained through simulations by sampling the number of iterations from their known distributions.

Figure 4 illustrates the computational cost of encoding Gaussian samples with either the Poisson functional representation (PFR) or the hybrid coding scheme ($N = \infty$). For a fixed target distribution and corresponding w_{min}, the number of iterations of the PFR is geometrically distributed with parameter w_{min} (Maddison, 2016). For the hybrid algorithm, the number of iterations is also geometrically distributed but with parameter $w_{\text{min}} \prod_i M_i$. Solid lines in Figure 4 were estimated by repeatedly sampling a target distribution $q_{\mathbf{x}}$, then computing w_{min} and finally sampling from the corresponding geometric distribution. Dashed lines correspond to empirical results by running the full algorithm. The empirical results match our simulated results.

For small σ the performance of the hybrid coding scheme matches that of the PFR. Once σ is large enough that $M_i > 1$, the hybrid coding scheme substantially outperforms the PFR. The wiggliness in the green curves is explained by the fact that the M_i are discontinuous functions of σ.

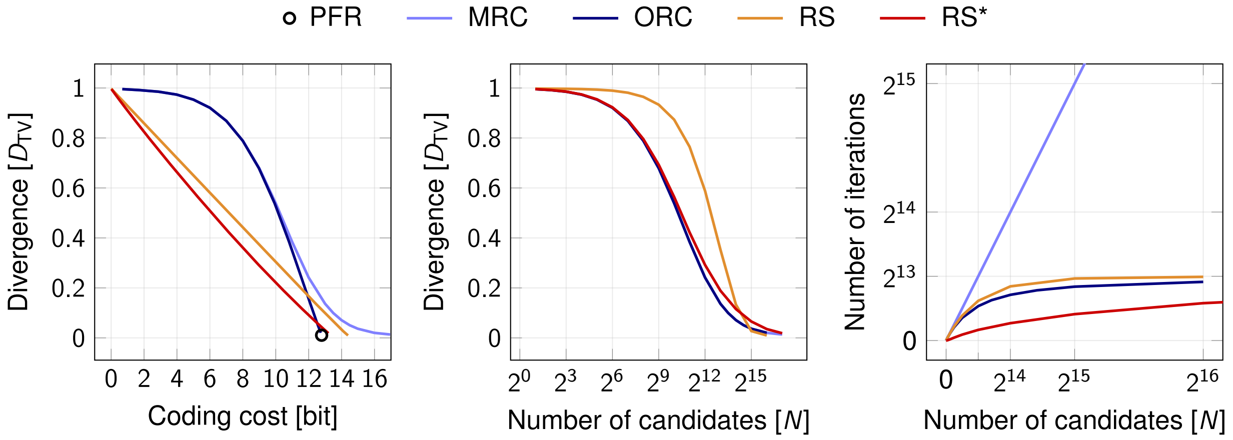

Figure 5. Additional figures for the example of communicating categorical samples. *Left:* The sample quality as a function of the coding cost, as in the main text but for a wider range of values. Note that samples of low quality (high D_{TV}) are rarely of interest. *Middle:* The sample quality as a function of the maximum number of candidates available to an algorithm. *Right:* The average number of candidates considered (that is, the number of iterations before termination) as a function of the maximum number of candidates.