Finite Block Length Rate-Distortion Theory for the Bernoulli Source with Hamming Distortion: A Tutorial

Bhaskar Krishnamachari
University of Southern California
bkrishna@usc.edu

February 27, 2026

Abstract

Lossy data compression lies at the heart of modern communication and storage systems. Shannon's rate-distortion theory provides the fundamental limit on how much a source can be compressed at a given fidelity, but it assumes infinitely long block lengths that are never realized in practice. We present a self-contained tutorial on rate-distortion theory for the simplest non-trivial source: a Bernoulli(p) sequence with Hamming distortion. We derive the classical rate-distortion function $R(D) = H(p) - H(D)$ from first principles, illustrate its computation via the Blahut-Arimoto algorithm, and then develop the finite block length refinements that characterize how the minimum achievable rate approaches the Shannon limit as the block length n grows. The central quantity in this refinement is the *rate-distortion dispersion* $V(D)$, which governs the $O(1/\sqrt{n})$ penalty for operating at finite block lengths. We accompany all theoretical developments with numerical examples and figures generated by accompanying Python scripts.

1 Introduction

The theory of lossy data compression traces its origins to Claude Shannon's landmark 1948 paper [1], which established that every source has a well-defined minimum description rate for any prescribed level of distortion. This result, made precise in Shannon's 1959 coding theorem for sources with a fidelity criterion [2], was remarkable for a reason that is easy to overlook today: it demonstrated that a single, clean mathematical function, the rate-distortion function $R(D)$, separates the achievable from the impossible, no matter how clever the compression scheme.

However, this elegant theory rests on a crucial idealization. The rate-distortion function $R(D)$ is an *asymptotic* quantity: it describes the minimum rate achievable when the block length n tends to infinity. Real communication and storage systems must operate with finite memory, finite latency, and finite computational resources. A natural and practically important question therefore arises: *how much extra rate do we need when the block length is finite?*

To build intuition, consider the simplest possible lossy compression scheme. Suppose we have a Bernoulli(0.5) source that produces sequences of $n = 2$ bits. There are four possible source sequences: 00, 01, 10, and 11. We wish to compress each sequence to just 1 bit, so our codebook has $M = 2$ entries and the rate is $R = \frac{1}{2}\log_2 2 = 0.5$ bits per source symbol. One natural code assigns codeword 0 to the pair $\{00, 01\}$ and codeword 1 to the pair $\{10, 11\}$: the encoder simply keeps the first bit and discards the second. The decoder reconstructs 00 from codeword 0 and 11 from codeword 1. What distortion does this code achieve? Under Hamming distortion (which counts the fraction of disagreeing positions), the four source sequences yield distortions $d(00, 00) = 0$, $d(01, 00) = 1/2$, $d(10, 11) = 1/2$, and $d(11, 11) = 0$. Since the source is fair, each sequence is equally likely, so the average distortion is $(0 + 1/2 + 1/2 + 0)/4 = 1/4$. We have thus constructed a concrete code operating at rate $R = 0.5$ and distortion $D = 0.25$.

Is this the best we can do at rate 0.5? Shannon's rate-distortion theory provides the answer. One of the landmark results of information theory, derived in detail in Section 4, is that for a Bernoulli(p) source with Hamming distortion, the minimum achievable rate at distortion level D is

$$R(D) = H(p) - H(D), \qquad 0 \le D \le \min(p, 1-p), \tag{1}$$

where $H(p) = -p \log_2 p - (1-p)\log_2(1-p)$ is the *binary entropy function*, which measures the inherent uncertainty of the source. For a fair coin ($p = 0.5$), we have $H(0.5) = 1$ bit, so the formula simplifies to $R(D) = 1 - H(D)$. Evaluating at $D = 0.25$ gives $R(0.25) = 1 - H(0.25) \approx 0.189$ bits per symbol. Since our simple code operates at rate 0.5, well above the Shannon limit of 0.189, there is considerable room for improvement, but only in the limit of large block lengths. For $n = 2$, it turns out, we cannot do better. To see why, note that any code with $M = 2$ assigns each of the four source sequences to one of two reconstructions $\hat{x}_1, \hat{x}_2 \in \{0,1\}^2$ (which need not belong to the assigned group). In $\{0,1\}^2$, each reconstruction can match at most one sequence exactly (itself, at Hamming distance 0); every other sequence has Hamming distance at least 1, contributing per-symbol distortion at least $1/2$. With only two reconstructions, at most two of the four sequences can achieve distortion 0, so at least two sequences contribute distortion $\ge 1/2$. Since all four are equally likely, the average distortion is at least $(0 + 0 + 1/2 + 1/2)/4 = 1/4$, which is exactly what our code achieves. The gap between this finite-n optimum of $D = 0.25$ at rate 0.5 and the Shannon limit of $R(0.25) \approx 0.189$ at the same distortion illustrates a fundamental phenomenon: short codes pay a rate penalty compared to the asymptotic limit. Quantifying this penalty precisely is the goal of finite block length theory.

Over the past two decades, a precise answer to this question has emerged through the work of Strassen [3], Ingber and Kochman [4], Kostina and Verdú [5], and others. To state their result, we need two preliminary ideas.

The first is a *distortion measure*: a function $d(x, \hat{x})$ that quantifies how "far" a reconstruction symbol $\hat{x}$ is from the original source symbol x. For binary data, the simplest and most natural choice is the *Hamming distortion*, which equals 1 when $x \neq \hat{x}$ and 0 when $x = \hat{x}$. When we compress a length-n source sequence $X^n = (X_1, \ldots, X_n)$ into a reconstruction $\hat{X}^n = (\hat{X}_1, \ldots, \hat{X}_n)$, the overall quality is measured by the *per-symbol distortion* $\frac{1}{n}\sum_{i=1}^n d(X_i, \hat{X}_i)$, which under Hamming distortion is simply the fraction of positions where the source and reconstruction disagree (the bit error rate).

The second idea is that we must refine what "achievable" means at finite block length. When n is finite, no code can guarantee that *every* source sequence is reproduced within distortion D. The source sequence X^n is random, and some realizations are inherently harder to compress than others. We therefore allow a small probability of failure: we require only that the distortion exceeds D with probability at most ε. More precisely, we say a code is (n, D, ε)-achievable at rate R if

$$\mathbb{P}\left(\frac{1}{n}\sum_{i=1}^n d(X_i, \hat{X}_i) > D \right) \le \varepsilon, \tag{2}$$

where the probability is over the randomness of the source sequence X^n. The quantity ε is called the *excess-distortion probability*. With this formulation, the minimum achievable rate $R(n, D, \varepsilon)$ depends on three parameters: the block length n, the target distortion D, and the tolerated failure probability ε.

The key insight of the finite block length theory is that $R(n, D, \varepsilon)$ admits a clean asymptotic expansion:

$$R(n, D, \varepsilon) \approx R(D) + \sqrt{\frac{V(D)}{n}}\, Q^{-1}(\varepsilon), \tag{3}$$

where $V(D)$ is the *rate-distortion dispersion*, a quantity that captures how variable the compression difficulty is across source symbols, and $Q^{-1}(\varepsilon)$ is the inverse of the Gaussian Q-function. From an engineering standpoint, (3) reveals that the penalty for finite block length decays as $1/\sqrt{n}$, a rate that is neither negligibly fast nor prohibitively slow.

In this tutorial, we develop the entire story from first principles for the simplest non-trivial setting: a Bernoulli(p) source with Hamming distortion. We have chosen this source for three reasons. First, the binary symmetric source is the discrete analogue of the Gaussian source: it is the canonical "textbook" example against which all intuitions are calibrated. Second, every quantity of interest ($R(D)$, the optimal test channel, the d-tilted information, and the dispersion $V(D)$) admits a clean closed-form expression. Third, despite its simplicity, the Bernoulli source reveals the full structure of the finite block length theory, including the role of source dispersion as a second-order characterization of compression difficulty.

The key contributions of this tutorial are:

1. A self-contained derivation of the rate-distortion function $R(D) = H(p) - H(D)$ for the Bernoulli(p) source, accessible to readers with minimal probability background.
2. A detailed treatment of the Blahut-Arimoto algorithm, including explicit 2×2 matrix computations and convergence analysis.
3. A development of finite block length rate-distortion theory, including the d-tilted information, rate-distortion dispersion, and the normal approximation.
4. Accompanying Python scripts that reproduce all numerical results and figures.

The remainder of this paper is organized as follows. Section 2 reviews the probability and information-theoretic foundations. Section 3 formulates the rate-distortion problem. Section 4 derives the rate-distortion function for the Bernoulli source. Section 5 presents the Blahut-Arimoto algorithm. Section 6 develops the finite block length theory. Section 7 presents comprehensive numerical explorations. Finally, Section 8 concludes with a discussion of open problems and further reading.

2 Probability and Information Foundations

In this section, we review the essential probability and information-theoretic concepts that underpin rate-distortion theory. We aim for an intuitive, example-driven development; readers seeking formal generality may consult Cover and Thomas [6].

2.1 Random Variables and Probability

Consider a coin that lands heads with probability p and tails with probability $1 - p$, where $0 \leq p \leq 1$. If we encode heads as 1 and tails as 0, a single coin flip is described by a *Bernoulli random variable* X with

$$\mathbb{P}(X = 1) = p, \qquad \mathbb{P}(X = 0) = 1 - p. \tag{4}$$

We write $X \sim \text{Bernoulli}(p)$ and refer to p as the *bias* of the source. A fair coin corresponds to $p = 1/2$, while a biased coin has $p \neq 1/2$.

When we flip the coin n times independently, we obtain a *sequence* $X^n = (X_1, X_2, \ldots, X_n)$ of independent and identically distributed (i.i.d.) Bernoulli random variables. This sequence is the object we wish to compress.

2.2 Entropy

How much "surprise" does a single coin flip carry? If the coin always lands heads ($p = 1$), there is no surprise at all; we know the outcome in advance. If the coin is fair ($p = 1/2$), each flip is maximally uncertain. Shannon formalized this intuition through the *entropy* of a random variable.

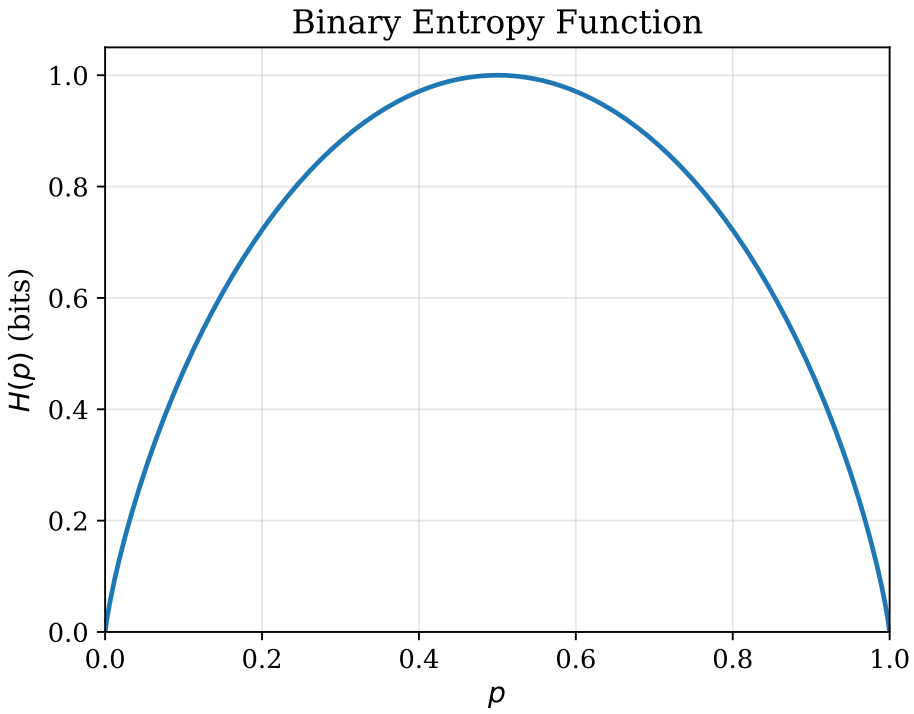

Figure 1: The binary entropy function $H(p)$ versus the source bias p. The entropy is maximized at $p = 1/2$, where each bit carries one full bit of information, and vanishes at $p \in \{0, 1\}$, where the source is deterministic.

Definition 2.1 (Binary Entropy)**.** *The* binary entropy function *is defined as*

$$H(p) = -p \log_2 p - (1 - p) \log_2 (1 - p), \tag{5}$$

with the convention that $0 \log_2 0 = 0$.

The entropy $H(p)$ measures the average surprise, in bits, of a single draw from a Bernoulli(p) source. It is zero when $p = 0$ or $p = 1$ (no uncertainty) and achieves its maximum value of 1 bit when $p = 1/2$ (maximum uncertainty). Figure 1 illustrates this behavior.

More generally, for a discrete random variable X taking values in a finite alphabet $\mathcal{X}$ with probability mass function $p_X(x)$, the entropy is

$$H(X) = -\sum_{x \in \mathcal{X}} p_X(x) \log_2 p_X(x). \tag{6}$$

The entropy quantifies the minimum average number of bits needed to losslessly represent X.

2.3 Sequences and Typical Sequences

Consider a sequence $x^n = (x_1, \ldots, x_n)$ produced by a Bernoulli(p) source. The *type* of x^n is the empirical distribution of symbols: if x^n contains k ones, its type is k/n. For large n, the law of large numbers guarantees that the fraction of ones concentrates around p.

A sequence is called *typical* if its empirical statistics are close to the true source distribution. The *asymptotic equipartition property* (AEP) states that with high probability, a sequence drawn from a Bernoulli(p) source satisfies

$$-\frac{1}{n} \log_2 p_{X^n}(X^n) \approx H(p). \tag{7}$$

Intuitively, there are approximately $2^{nH(p)}$ typical sequences, and they account for almost all of the probability mass. This observation is the foundation of both lossless and lossy compression: loosely speaking, it suggests that we could represent all likely source sequences by mapping them to just these $2^{nH(p)}$ typical sequences, each of which could be identified using only $n \cdot H(p)$ bits. In other words, each source symbol that nominally requires one bit to describe can, on average, be compressed down to $H(p)$ bits.

2.4 Mutual Information

When we compress a source X into a reconstruction $\hat{X}$, some information about X is preserved and some is lost. The *mutual information* $I(X; \hat{X})$ quantifies how much information $\hat{X}$ retains about X.

Definition 2.2 (Mutual Information)**.** *For jointly distributed random variables* $(X, \hat{X})$ *with joint distribution* $p_{X,\hat{X}}(x, \hat{x})$*, the mutual information is*

$$I(X; \hat{X}) = \sum_{x,\hat{x}} p_{X,\hat{X}}(x, \hat{x}) \log_2 \frac{p_{\hat{X}|X}(\hat{x}|x)}{p_{\hat{X}}(\hat{x})}. \tag{8}$$

An equivalent and often more intuitive expression is

$$I(X; \hat{X}) = H(X) - H(X|\hat{X}), \tag{9}$$

where $H(X|\hat{X})$ is the conditional entropy, that is, the residual uncertainty about X after observing $\hat{X}$. The mutual information is always non-negative, and equals zero if and only if X and $\hat{X}$ are independent. It equals $H(X)$ when $\hat{X}$ determines X perfectly.

3 The Rate-Distortion Problem

In this section, we formulate the central problem of lossy compression. We define what it means to compress a source with a prescribed fidelity, and state Shannon's fundamental theorem that establishes the existence of a minimum achievable rate.

3.1 What Is Lossy Compression?

Consider a source that emits a sequence $X^n = (X_1, \ldots, X_n)$ of n i.i.d. Bernoulli(p) random variables. A *lossy compression scheme* consists of two mappings:

- An *encoder* $f_n : \{0,1\}^n \to \{1, 2, \ldots, M\}$ that maps the source sequence to one of M codewords.
- A *decoder* $g_n : \{1, 2, \ldots, M\} \to \{0,1\}^n$ that maps each codeword index back to a reconstruction sequence $\hat{X}^n = g_n(f_n(X^n))$.

The set $C = \{g_n(1), g_n(2), \ldots, g_n(M)\}$ is called the *codebook*, and each element is a *reproduction sequence*.

The *rate* of the code is

$$R = \frac{1}{n} \log_2 M \quad \text{bits per source symbol.} \tag{10}$$

This quantity measures the average number of bits used to represent each source symbol. A lower rate means more aggressive compression.

3.2 Distortion Measures

To quantify the fidelity of the reconstruction, we need a *distortion measure*. For binary sequences, the natural choice is the *Hamming distortion*:

$$d(x, \hat{x}) = \begin{cases} 0, & \text{if } x = \hat{x}, \\ 1, & \text{if } x \neq \hat{x}. \end{cases} \tag{11}$$

The Hamming distortion simply counts whether a symbol was reproduced correctly.

For a pair of sequences $(x^n, \hat{x}^n)$, the *per-symbol distortion* is

$$d(x^n, \hat{x}^n) = \frac{1}{n} \sum_{i=1}^{n} d(x_i, \hat{x}_i), \tag{12}$$

which equals the fraction of positions where the source and reconstruction disagree, that is, the *bit error rate*.

3.3 The Fundamental Question

We can now state the central question of rate-distortion theory:

What is the minimum rate R such that there exists a sequence of encoder-decoder pairs achieving average distortion at most D?

Shannon's remarkable insight was that, in the limit $n \to \infty$, this minimum rate converges to a "single-letter" quantity $R(D)$—an optimization involving only the distribution of one source symbol and one reproduction symbol, not the full length-n sequences [2]. We do not need to search over all possible encoder-decoder pairs of all possible block lengths: the *asymptotic* answer depends only on the source distribution p_X and the distortion measure d. (For any finite n, the actual minimum rate exceeds $R(D)$; quantifying this gap is the subject of Section 6.)

3.4 The Test Channel and the Rate-Distortion Function

The key to Shannon's formulation is the concept of a *test channel*. Rather than optimizing over encoder-decoder pairs, we optimize over conditional distributions $p_{\hat{X}|X}(\hat{x}|x)$ that describe a probabilistic mapping from source symbols to reconstruction symbols.

Definition 3.1 (Rate-Distortion Function)**.** *The rate-distortion function of a source X with distortion measure d is*

$$R(D) = \min_{\substack{p_{\hat{X}|X}: \\ \mathbb{E}[d(X,\hat{X})] \leq D}} I(X; \hat{X}), \tag{13}$$

where the minimization is over all conditional distributions $p_{\hat{X}|X}$ satisfying the distortion constraint.

Why do we *minimize* the mutual information? The mutual information $I(X; \hat{X})$ measures how many bits the reconstruction $\hat{X}$ reveals about the original source X. Any information that $\hat{X}$ carries about X must have been conveyed by the encoder, so $I(X; \hat{X})$ is precisely the rate cost of the code. The distortion constraint $\mathbb{E}[d(X, \hat{X})] \leq D$ separately ensures that the reconstruction is faithful enough; our goal is then to find the cheapest test channel (lowest rate) that still keeps distortion within the budget D.

The rate-distortion function therefore has a clean operational interpretation: $R(D)$ is the minimum number of bits per source symbol required to describe the source with average distortion at most D. Rates above $R(D)$ are achievable (there exist codes that work), while rates below $R(D)$ are not achievable by any code, regardless of its complexity.

Figure 2 illustrates the encoder-decoder structure and the test channel abstraction.

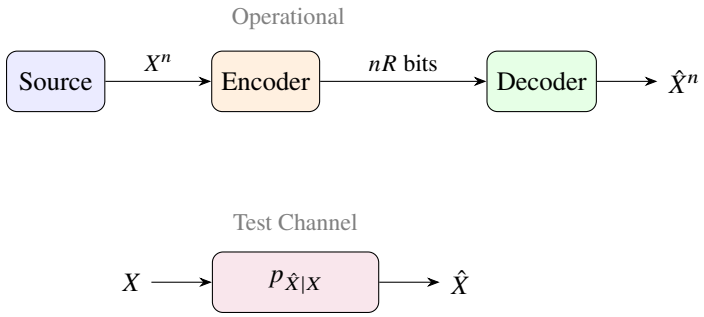

Figure 2: Top: the operational lossy compression setup with encoder and decoder. Bottom: the test channel $p_{\hat{X}|X}$ that abstracts away the codebook structure. The rate-distortion function minimizes mutual information $I(X;\hat{X})$ over all test channels satisfying the distortion constraint.

4 The Rate-Distortion Function for the Bernoulli Source

In this section, we derive the rate-distortion function for the Bernoulli(p) source with Hamming distortion. This is perhaps the cleanest closed-form result in all of rate-distortion theory.

4.1 Setting Up the Optimization

We wish to solve

$$R(D) = \min_{\substack{p_{\hat{X}|X}: \\ \mathbb{E}[d(X,\hat{X})] \le D}} I(X;\hat{X}) \tag{14}$$

for $X \sim \text{Bernoulli}(p)$ and Hamming distortion $d(x,\hat{x}) = \mathbf{1}\{x \neq \hat{x}\}$.

Since both the source alphabet $\mathcal{X} = \{0,1\}$ and the reproduction alphabet $\hat{\mathcal{X}} = \{0,1\}$ are binary, the test channel $p_{\hat{X}|X}$ is a 2×2 stochastic matrix with four parameters, of which two are free (each row sums to one). We can parameterize the test channel as

$$\begin{pmatrix} p_{\hat{X}|X}(0|0) & p_{\hat{X}|X}(1|0) \\ p_{\hat{X}|X}(0|1) & p_{\hat{X}|X}(1|1) \end{pmatrix} = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}, \tag{15}$$

where $\alpha = p_{\hat{X}|X}(1|0)$ is the probability of flipping a 0 to a 1, and $\beta = p_{\hat{X}|X}(0|1)$ is the probability of flipping a 1 to a 0.

The expected distortion under this test channel is

$$\mathbb{E}[d(X,\hat{X})] = (1-p)\alpha + p\beta. \tag{16}$$

We seek the test channel parameters (α,β) that minimize the mutual information $I(X;\hat{X})$ subject to $(1-p)\alpha + p\beta \le D$.

4.2 The Optimal Test Channel

We present two derivations of the optimal test channel. The first uses the classical Lagrangian/KKT machinery, which exposes the algebraic structure of the solution. The second uses an entropy-maximization argument that provides complementary geometric intuition.

Derivation 1: Lagrangian and KKT Conditions

To solve the optimization (14), we form the Lagrangian

$$\mathcal{L} = I(X;\hat{X}) + \lambda\big(\mathbb{E}[d(X,\hat{X})] - D\big), \tag{17}$$

where $\lambda \geq 0$ is the Lagrange multiplier. We work through the Karush-Kuhn-Tucker (KKT) conditions step by step.

Step 1: Expand the Lagrangian. Using the mutual information definition (8), we write $I(X;\hat{X})$ as a function of the test channel parameters α and β from (15). The marginal reproduction distribution is

$$p_{\hat{X}}(0) = (1-p)(1-\alpha) + p\beta, \qquad p_{\hat{X}}(1) = (1-p)\alpha + p(1-\beta).$$

The mutual information is

$$\begin{aligned} I(X;\hat{X}) = (1-p)(1-\alpha)\log_2\frac{1-\alpha}{p_{\hat{X}}(0)} + (1-p)\alpha\log_2\frac{\alpha}{p_{\hat{X}}(1)} \\ + p\beta\log_2\frac{\beta}{p_{\hat{X}}(0)} + p(1-\beta)\log_2\frac{1-\beta}{p_{\hat{X}}(1)}, \end{aligned} \tag{18}$$

and the distortion is $\mathbb{E}[d(X,\hat{X})] = (1-p)\alpha + p\beta$. The full Lagrangian is therefore

$$\mathcal{L}(\alpha,\beta,\lambda) = I(X;\hat{X}) + \lambda\big[(1-p)\alpha + p\beta - D\big].$$

Step 2: Stationarity conditions. We differentiate $\mathcal{L}$ with respect to α and β and set each derivative to zero. Differentiating the mutual information with respect to α, using $\partial p_{\hat{X}}(0)/\partial\alpha = -(1-p)$ and $\partial p_{\hat{X}}(1)/\partial\alpha = (1-p)$, gives (after simplification)

$$\frac{\partial I}{\partial\alpha} = (1-p)\log_2\frac{\alpha\, p_{\hat{X}}(0)}{(1-\alpha)\, p_{\hat{X}}(1)}.$$

Setting $\partial\mathcal{L}/\partial\alpha = 0$ yields

$$\log_2\frac{\alpha\, p_{\hat{X}}(0)}{(1-\alpha)\, p_{\hat{X}}(1)} = -\lambda. \tag{19}$$

By an identical calculation with respect to β:

$$\log_2\frac{(1-\beta)\, p_{\hat{X}}(0)}{\beta\, p_{\hat{X}}(1)} = \lambda. \tag{20}$$

Step 3: Recognize the exponential (Gibbs) form. Equation (19) says $\alpha/(1-\alpha) = 2^{-\lambda}\, p_{\hat{X}}(1)/p_{\hat{X}}(0)$. Since $\alpha = p_{\hat{X}|X}(1|0)$ and $1-\alpha = p_{\hat{X}|X}(0|0)$, this means

$$\frac{p_{\hat{X}|X}(1|0)}{p_{\hat{X}|X}(0|0)} = \frac{p_{\hat{X}}(1)}{p_{\hat{X}}(0)}\cdot 2^{-\lambda}.$$

For source symbol $x = 0$, the distortions are $d(0,0) = 0$ and $d(0,1) = 1$, so the factor $2^{-\lambda}$ acts as an exponential penalty on the mismatched reproduction. Similarly, (20) gives $p_{\hat{X}|X}(0|1)/p_{\hat{X}|X}(1|1) = (p_{\hat{X}}(0)/p_{\hat{X}}(1))\cdot 2^{-\lambda}$, again penalizing the mismatch $d(1,0) = 1$. Both conditions are unified by the *Gibbs form*:

$$p^*_{\hat{X}|X}(\hat{x}|x) = \frac{Q^*(\hat{x})\, 2^{-\lambda\, d(x,\hat{x})}}{Z(x)}, \tag{21}$$

where $Q^*(\hat{x}) = p^*_{\hat{X}}(\hat{x})$ is the optimal reproduction distribution (which the test channel must be self-consistent with) and $Z(x) = \sum_{\hat{x}} Q^*(\hat{x})\, 2^{-\lambda\, d(x,\hat{x})}$ is the normalizing constant ensuring $\sum_{\hat{x}} p^*_{\hat{X}|X}(\hat{x}|x) = 1$. (The notation Q^* for the reproduction distribution is standard in rate-distortion theory and should not be confused with the Gaussian Q-function introduced in Section 6.5.)

The Gibbs form has a natural interpretation: the optimal test channel takes the reproduction distribution Q^* and *reweights* each reproduction symbol $\hat{x}$ by an exponential factor $2^{-\lambda\, d(x,\hat{x})}$. Symbols close to x (low distortion) keep their weight, while symbols far from x (high distortion) are exponentially suppressed. The Lagrange multiplier λ controls the severity of this penalty: larger λ (corresponding to tighter distortion constraints) produces a more concentrated channel.

Step 4: Identify the backward channel. The Gibbs form (21) reveals a remarkable structure when we look at the *backward* (reverse) channel via Bayes' rule:

$$p^*_{X|\hat{X}}(x|\hat{x}) = \frac{p_X(x)\, p^*_{\hat{X}|X}(\hat{x}|x)}{Q^*(\hat{x})} = \frac{p_X(x)\, 2^{-\lambda\, d(x,\hat{x})}}{Z(x)},$$

where $Z(x) = \sum_{\hat{x}} Q^*(\hat{x})\, 2^{-\lambda\, d(x,\hat{x})}$ is the same forward normalizer from (21). A crucial subtlety: $Z(x)$ depends on the *source* symbol x, not the reproduction $\hat{x}$. For Hamming distortion, $d(x,\hat{x}) = 0$ when $x = \hat{x}$ and 1 when $x \neq \hat{x}$, so the two normalizers are

$$Z(0) = Q^*(0) + Q^*(1) \cdot 2^{-\lambda}, \qquad Z(1) = Q^*(0) \cdot 2^{-\lambda} + Q^*(1).$$

Writing out all four backward-channel entries, grouped by source symbol (entries sharing the same x share the same denominator):

$$\text{Source } x = 0: \quad p^*_{X|\hat{X}}(0|0) = \frac{1-p}{Z(0)}, \qquad p^*_{X|\hat{X}}(0|1) = \frac{(1-p)\cdot 2^{-\lambda}}{Z(0)},$$

$$\text{Source } x = 1: \quad p^*_{X|\hat{X}}(1|0) = \frac{p\cdot 2^{-\lambda}}{Z(1)}, \qquad p^*_{X|\hat{X}}(1|1) = \frac{p}{Z(1)}.$$

Step 5: Complementary slackness and solving for λ. The KKT conditions require *complementary slackness*: $\lambda\big[(1-p)\alpha + p\beta - D\big] = 0$. For $0 < D < \min(p, 1-p)$ the rate is strictly positive, so the distortion constraint must be active ($\lambda > 0$), giving $\mathbb{E}[d(X,\hat{X})] = D$.

We now determine λ by requiring the backward channel to be a BSC(D). Consider the two entries with source symbol $x = 0$ (which share the denominator $Z(0)$). Setting $p^*_{X|\hat{X}}(0|0) = 1 - D$ gives

$$Z(0) = \frac{1-p}{1-D}.$$

Then requiring $p^*_{X|\hat{X}}(0|1) = D$:

$$\frac{(1-p)\cdot 2^{-\lambda}}{Z(0)} = \frac{(1-p)\cdot 2^{-\lambda}}{(1-p)/(1-D)} = (1-D)\cdot 2^{-\lambda} = D,$$

which gives $2^{-\lambda} = D/(1-D)$, i.e.,

$$\lambda = \log_2 \frac{1-D}{D}. \tag{22}$$

One can verify the $x = 1$ entries give the same result: $p^*_{X|\hat{X}}(1|1) = 1 - D$ forces $Z(1) = p/(1-D)$, and then $p^*_{X|\hat{X}}(1|0) = p\cdot 2^{-\lambda}/Z(1) = (1-D)\cdot D/(1-D) = D$. ✓

Therefore the backward channel is a BSC(D):

$$p^*_{X|\hat{X}}(x|\hat{x}) = \begin{cases} 1-D, & \text{if } x = \hat{x}, \\ D, & \text{if } x \neq \hat{x}. \end{cases} \tag{23}$$

Step 6: Derive the reproduction distribution and forward channel. From the backward channel BSC(D) and Bayes' rule, the reproduction distribution must satisfy $p_X(x) = \sum_{\hat{x}} Q^*(\hat{x})\, p^*_{X|\hat{X}}(x|\hat{x})$. For $x = 1$: $p = Q^*(1)(1-D) + Q^*(0)D$. Solving:

$$Q^*(1) = \frac{p-D}{1-2D}, \qquad Q^*(0) = \frac{1-p-D}{1-2D}. \tag{24}$$

(Both are positive when $D < \min(p, 1-p)$, which is the interesting regime.) The forward channel $p^*_{\hat{X}|X}$ is recovered from Bayes' rule, $p^*_{\hat{X}|X}(\hat{x}|x) = Q^*(\hat{x})\, p^*_{X|\hat{X}}(x|\hat{x})/p_X(x)$:

$$p^*_{\hat{X}|X} = \begin{pmatrix} \frac{Q^*(0)(1-D)}{1-p} & \frac{Q^*(1)\,D}{1-p} \\ \frac{Q^*(0)\,D}{p} & \frac{Q^*(1)(1-D)}{p} \end{pmatrix}. \tag{25}$$

One can verify that each row sums to one.

Remark 4.1. *When $p = 1/2$, we have $Q^*(0) = Q^*(1) = 1/2$, and the forward channel (25) reduces to a BSC(D) — the symmetric case that many textbooks present as the general answer. For $p \neq 1/2$, however, the forward channel is* asymmetric*: the probability of flipping a 0 to a 1 differs from the probability of flipping a 1 to a 0. The key structural insight is that it is the* backward *channel that is symmetric, not the forward channel.*

The optimal reproduction distribution Q^* was determined in (24):

$$Q^*(0) = \frac{1-p-D}{1-2D}, \tag{26}$$

$$Q^*(1) = \frac{p-D}{1-2D}. \tag{27}$$

Note that $Q^*(0) + Q^*(1) = 1$, as expected, and $Q^*(1) = p$ only when $p = 1/2$. We will use this reproduction distribution again in Section 6.3 when computing the d-tilted information.

The corresponding Lagrange multiplier is

$$\lambda^* = \log_2 \frac{1-D}{D}, \tag{28}$$

which is the same value derived in (22). This is not a coincidence: in any constrained optimization, the Lagrange multiplier equals the sensitivity of the objective to the constraint. Here, λ^* tells us how much the minimum rate $R(D)$ changes when we relax the distortion budget by a small amount dD. Formally, $\lambda^* = -R'(D)$, which we can verify directly: $R'(D) = -H'(D) = \log_2 \frac{D}{1-D}$, so $-R'(D) = \log_2 \frac{1-D}{D} = \lambda^*$. The multiplier is large when D is small (the rate curve is steep, so each additional bit of distortion saves many bits of rate) and approaches zero as $D \to \min(p, 1-p)$ (the curve flattens near zero rate).

Derivation 2: Entropy Maximization

We now present a second, more direct derivation that bypasses the Lagrangian machinery entirely. It relies on three ideas, each of which we explain carefully.

Idea 1: Minimizing mutual information = maximizing conditional entropy. Recall from (9) that $I(X;\hat{X}) = H(X) - H(X|\hat{X})$. Since the source entropy $H(X) = H(p)$ is fixed (it does not depend on the test channel), minimizing $I(X;\hat{X})$ over the test channel is the same as maximizing $H(X|\hat{X})$. Intuitively, $H(X|\hat{X})$ measures how much uncertainty about X remains *after* seeing $\hat{X}$. A good lossy code should leave as much residual uncertainty as possible (retaining only the information needed to meet the distortion target), thereby minimizing the rate.

Idea 2: Relating conditional entropy to the error variable. Define the *error variable* $Z = X \oplus \hat{X}$ (addition modulo 2), so $Z = 1$ when the source and reconstruction disagree and $Z = 0$ otherwise. Since X is completely determined by the pair $(\hat{X}, Z)$ via $X = \hat{X} \oplus Z$, knowing $\hat{X}$ and Z tells you X exactly. By the chain rule of entropy, $H(X|\hat{X}) = H(Z|\hat{X})$: the residual uncertainty about X given $\hat{X}$ is the same as the uncertainty about the error pattern Z given $\hat{X}$.

Idea 3: Two upper bounds on $H(Z|\hat{X})$.

$$H(X|\hat{X}) = H(Z|\hat{X}) \le H(Z) \tag{29}$$

$$\le H(D). \tag{30}$$

Why does (29) hold? This is the general fact that "conditioning reduces entropy": knowing $\hat{X}$ can only help (or not hurt) in predicting Z, so $H(Z|\hat{X}) \le H(Z)$. Equality holds if and only if Z is independent of $\hat{X}$ — that is, knowing the reconstruction tells you nothing about the error pattern.

Why does (30) hold? Since Z is a binary random variable with $\mathbb{E}[Z] = \mathbb{E}[d(X,\hat{X})] \le D$, the entropy of Z is at most $H(D)$. This is because, among all binary random variables with mean at most D, the Bernoulli(D) distribution has the maximum entropy. (This is a special case of the general principle that the maximum-entropy distribution subject to a mean constraint is exponential; for binary variables, it is Bernoulli.)

Achieving both bounds simultaneously. Both inequalities become equalities when:

1. Z is independent of $\hat{X}$ (so that $H(Z|\hat{X}) = H(Z)$), and
2. $Z \sim$ Bernoulli(D) (so that $H(Z) = H(D)$ and $\mathbb{E}[Z] = D$).

In other words, the optimal scheme has the error $Z \sim$ Bernoulli(D) acting independently of the reconstruction $\hat{X}$. Since $X = \hat{X} \oplus Z$, this means the backward channel $p_{X|\hat{X}}$ is a BSC with crossover probability D, exactly as (23). The reproduction distribution Q^* and forward channel then follow from Bayes' rule, exactly as in Steps 5–6 of Derivation 1.

4.3 The Closed-Form Result

Having identified that the optimal backward channel is BSC(D), the rate-distortion function follows immediately. From the entropy-maximization viewpoint (Derivation 2 of Section 4.2), the maximum conditional entropy is $H(X|\hat{X}) = H(D)$ (achieved when $Z \sim$ Bernoulli(D) is independent of $\hat{X}$), so

$$R(D) = H(p) - \max H(X|\hat{X}) = H(p) - H(D).$$

Equivalently, from the Lagrangian viewpoint (Derivation 1), $I(X;\hat{X}) = H(X) - H(X|\hat{X}) = H(p) - H(D)$ when evaluated at the optimal test channel, since the backward BSC(D) gives $H(X|\hat{X}) = H(D)$. When $D \le \min(p, 1-p)$, the distortion constraint is active and the rate-distortion function is

$$\boxed{R(D) = H(p) - H(D), \qquad 0 \le D \le \min(p, 1-p).} \tag{31}$$

For $D > \min(p, 1-p)$, we have $R(D) = 0$.

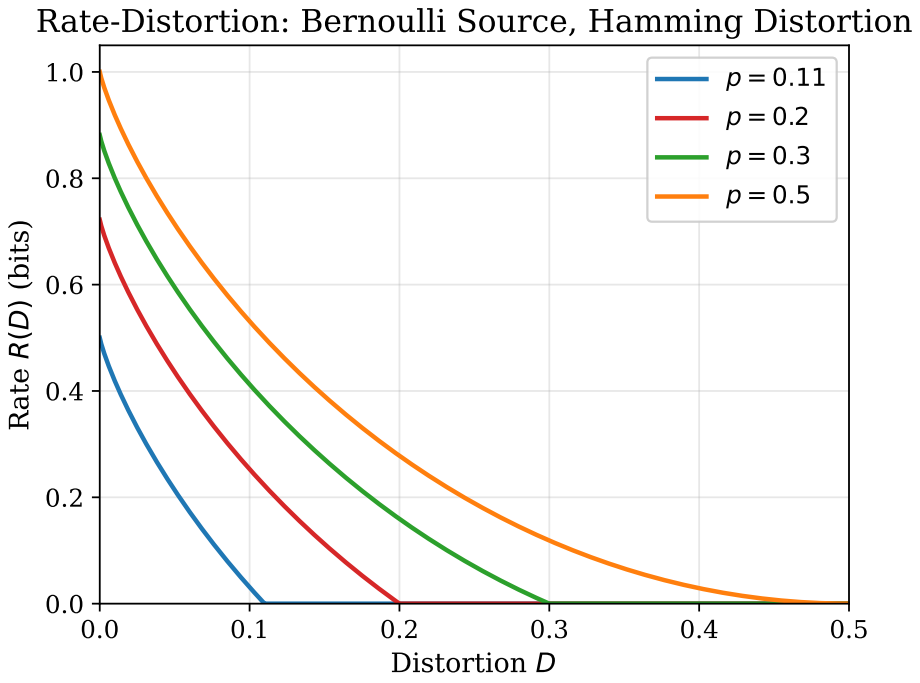

Figure 3: The rate-distortion function $R(D) = H(p) - H(D)$ for a Bernoulli(p) source with Hamming distortion, shown for $p \in \{0.11, 0.2, 0.3, 0.5\}$. Each curve is convex and decreasing, starting at $R(0) = H(p)$ and reaching zero at $D = \min(p, 1-p)$. The $p = 0.5$ curve starts highest because the fair coin has the most entropy.

Remark 4.2. *The formula $R(D) = H(p) - H(D)$ has an appealing interpretation: the rate equals the entropy of the source minus the entropy of the "noise" introduced by the test channel. As the allowed distortion D increases, the noise entropy $H(D)$ grows, and fewer bits are needed to describe the source.*

Let us examine the boundary cases. When $D = 0$, we require perfect reconstruction, and $R(0) = H(p)$: the rate equals the source entropy, which is the lossless compression limit. When $D = \min(p, 1-p)$, the rate drops to zero. In this regime, the decoder can simply output the more likely symbol (0 if $p < 1/2$, or 1 if $p > 1/2$) for every position, achieving distortion $\min(p, 1-p)$ with zero rate.

Figure 3 shows the rate-distortion function for several values of the source bias p.

We note three important properties of the rate-distortion function:

1. **Convexity:** $R(D)$ is a convex function of D. This means that each additional unit of distortion "buys" progressively less rate reduction.

2. **Monotonicity:** $R(D)$ is non-increasing in D. Allowing more distortion can only help (or leave unchanged) the compression rate.

3. **Continuity:** $R(D)$ is continuous on $[0, \min(p, 1-p)]$.

4.4 Historical Note

Shannon stated the rate-distortion function for the binary source in his 1959 paper [2]. A comprehensive treatment of rate-distortion theory for general sources was developed by Berger [7]. The elegant formula $R(D) = H(p) - H(D)$ serves as the starting point for virtually every textbook discussion of lossy source coding; see, for example, Cover and Thomas [6].

5 The Blahut-Arimoto Algorithm

In this section, we present the Blahut-Arimoto algorithm, a powerful iterative method for computing rate-distortion functions. While the Bernoulli source admits a closed-form solution, the Blahut-Arimoto algorithm applies to arbitrary finite-alphabet sources and distortion measures.

5.1 Motivation

The rate-distortion function (13) is defined as a minimization of mutual information over a convex set of conditional distributions. For the Bernoulli source with Hamming distortion, we exploited the problem's symmetry to obtain a closed-form solution. However, for more complex sources, such as non-uniform discrete sources with non-binary alphabets or non-Hamming distortion measures, the optimization does not admit a closed-form solution, and a computational approach is needed.

The Blahut-Arimoto algorithm [8, 9] solves this optimization through an elegant *alternating minimization* procedure. It was independently discovered by Blahut and Arimoto in 1972, and its convergence was later established rigorously by Csiszár [10].

5.2 The Algorithm

The Blahut-Arimoto algorithm operates on the Lagrangian dual formulation of the rate-distortion problem. For a given slope parameter $s > 0$ (corresponding to the Lagrange multiplier), the algorithm minimizes the functional

$$F(s) = \min_{p_{\hat{X}|X}} \big[I(X;\hat{X}) + s \cdot \mathbb{E}[d(X,\hat{X})] \big]. \tag{32}$$

By sweeping s over positive values, we trace out the entire $R(D)$ curve. (The slope s here is in nats and relates to the Lagrange multiplier λ^* from Section 4.2 by $s = \lambda^* \ln 2$; the natural exponential $e^{-s\,d}$ in Step 1 below is equivalent to the $2^{-\lambda\, d}$ form used there.)

The algorithm alternates between two updates:

Step 1: Update the test channel. Given the current reproduction distribution $p_{\hat{X}}(\hat{x})$, update

$$p_{\hat{X}|X}(\hat{x}|x) = \frac{p_{\hat{X}}(\hat{x})\, e^{-s\, d(x,\hat{x})}}{Z(x)}, \tag{33}$$

where $Z(x) = \sum_{\hat{x}} p_{\hat{X}}(\hat{x})\, e^{-s\, d(x,\hat{x})}$ is a normalization constant ensuring $\sum_{\hat{x}} p_{\hat{X}|X}(\hat{x}|x) = 1$.

Step 2: Update the reproduction distribution. Given the updated test channel, compute

$$p_{\hat{X}}(\hat{x}) = \sum_{x} p_X(x)\, p_{\hat{X}|X}(\hat{x}|x). \tag{34}$$

This is simply the marginal of $\hat{X}$ induced by passing X through the updated test channel.

The algorithm is initialized with a uniform reproduction distribution $p_{\hat{X}}(\hat{x}) = 1/|\hat{\mathcal{X}}|$ and iterates Steps 1 and 2 until convergence.

Remark 5.1. *The alternating structure of the algorithm has a natural interpretation: Step 1 finds the best test channel for a fixed output distribution, while Step 2 updates the output distribution to be consistent with the new test channel. This is an instance of the classical alternating minimization framework, and convergence is guaranteed because each step decreases the Lagrangian objective.*

Algorithm 1 Blahut-Arimoto Algorithm

Require: Source distribution p_X, distortion matrix d, slope $s > 0$, tolerance δ
Ensure: Rate R and distortion D on the $R(D)$ curve

Initialize $p_{\hat{X}}(\hat{x}) \leftarrow 1/|\hat{\mathcal{X}}|$ for all $\hat{x}$
repeat
  for each $x \in \mathcal{X}$ **do**
    $Z(x) \leftarrow \sum_{\hat{x}} p_{\hat{X}}(\hat{x})\, e^{-s\, d(x,\hat{x})}$
    for each $\hat{x} \in \hat{\mathcal{X}}$ **do**
      $p_{\hat{X}|X}(\hat{x}|x) \leftarrow p_{\hat{X}}(\hat{x})\, e^{-s\, d(x,\hat{x})} / Z(x)$
    end for
  end for
  for each $\hat{x} \in \hat{\mathcal{X}}$ **do**
    $p_{\hat{X}}(\hat{x}) \leftarrow \sum_x p_X(x)\, p_{\hat{X}|X}(\hat{x}|x)$
  end for
  Compute $R \leftarrow I(X;\hat{X})$ and $D \leftarrow \mathbb{E}[d(X,\hat{X})]$
until $|R_{\text{new}} - R_{\text{old}}| < \delta$
return (R, D)

Remark 5.2 (Intuition for Step 1)**.** *The update in (33) has an appealing interpretation: for each source symbol x, the algorithm reweights the current reproduction distribution $p_{\hat{X}}(\hat{x})$ by the factor $e^{-s\, d(x,\hat{x})}$. Reproduction symbols $\hat{x}$ that are close to x (low distortion) receive a weight near* 1*, while those that are far from x (high distortion) are exponentially suppressed. The result is then normalized to form a valid conditional distribution. This is precisely a* multiplicative weights update*: rather than making additive adjustments, the algorithm multiplies each probability by an exponential penalty that depends on the "cost" $d(x,\hat{x})$. Readers familiar with online learning will recognize the same mechanism at work in the Hedge algorithm and its bandit variant Exp3, where actions are reweighted by $e^{-\eta \cdot loss}$ to balance exploration and exploitation. In the Blahut-Arimoto setting, the slope parameter s plays the role of the learning rate η: a larger s imposes a harsher penalty on distortion, concentrating the test channel on the closest reproduction symbols.*

Algorithm 1 presents the complete pseudocode.

5.3 Application to the Bernoulli Source

We now apply the Blahut-Arimoto algorithm to the Bernoulli(p) source with Hamming distortion. Since the source and reproduction alphabets are both $\{0, 1\}$, the test channel is a 2×2 matrix, and all computations are explicit.

Consider a concrete example with $p = 0.3$ and slope parameter $s = 10$. At each iteration, we maintain the reproduction distribution $(p_{\hat{X}}(0), p_{\hat{X}}(1))$ and the test channel matrix. The updates in (33) and (34) become:

$$p_{\hat{X}|X}(0|0) = \frac{p_{\hat{X}}(0)}{p_{\hat{X}}(0) + p_{\hat{X}}(1)\, e^{-s}}, \tag{35}$$

$$p_{\hat{X}|X}(0|1) = \frac{p_{\hat{X}}(0)\, e^{-s}}{p_{\hat{X}}(0)\, e^{-s} + p_{\hat{X}}(1)}. \tag{36}$$

Since $e^{-s} = e^{-10} \approx 4.5 \times 10^{-5}$, the multiplicative weight for any mismatched reconstruction ($d(x,\hat{x}) = 1$) is tiny compared to the weight for a correct reconstruction ($d(x,\hat{x}) = 0$, weight 1). For example, in the numerator of $p_{\hat{X}|X}(0|0)$, the term $p_{\hat{X}}(0)$ (correct match) appears with weight 1, while the denominator also

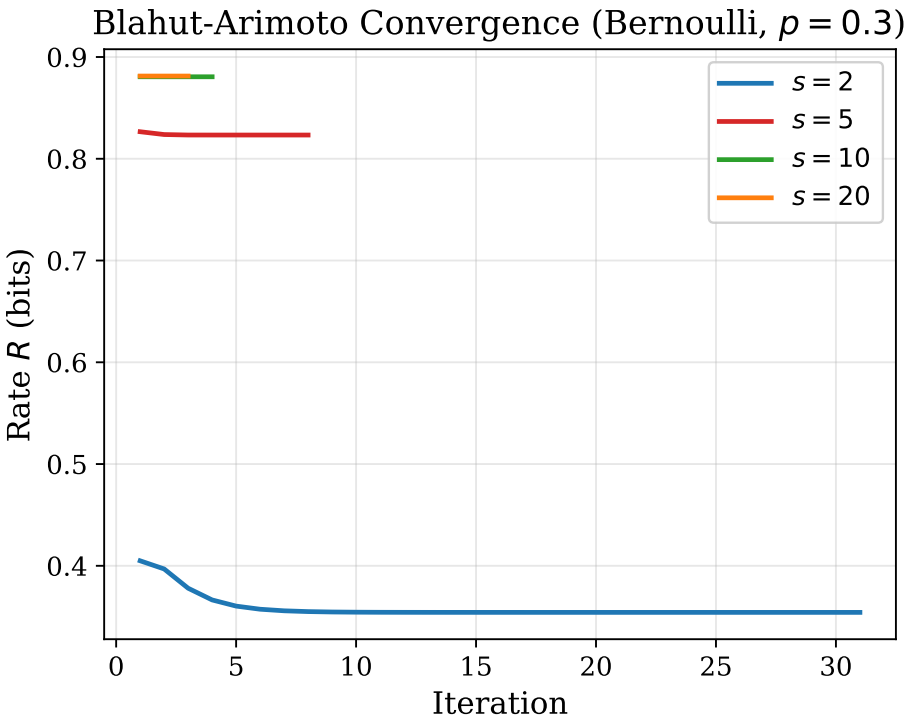

Figure 4: Convergence of the Blahut-Arimoto algorithm for $p = 0.3$ and slope parameters $s \in \{2, 5, 10, 20\}$. The rate converges monotonically to its final value within a few tens of iterations.

includes $p_{\hat{X}}(1)\, e^{-10}$ (mismatch), which is roughly 10^5 times smaller. The result is that $p_{\hat{X}|X}(0|0) \approx 1$ and $p_{\hat{X}|X}(1|0) \approx 0$: the test channel is driven toward a near-identity matrix, corresponding to very low distortion.

Figure 4 shows the convergence of the Blahut-Arimoto algorithm for $p = 0.3$ and several values of the slope parameter s. The algorithm converges rapidly, typically reaching machine precision within 20–50 iterations. Larger values of s correspond to lower target distortions and converge faster.

To validate the algorithm, we sweep s over a range of values and plot the resulting (D, R) pairs alongside the closed-form curve $R(D) = H(p) - H(D)$. Figure 5 confirms that the Blahut-Arimoto algorithm recovers the exact rate-distortion function.

5.4 Historical Note

The algorithm was independently proposed by Blahut [8] and Arimoto [9] in 1972. Blahut's formulation emphasized the Lagrangian dual structure, while Arimoto approached the problem through an iterative projection method. Csiszár [10] unified and extended both approaches using his theory of I-divergence, establishing the alternating minimization interpretation that clarifies why the iterations converge. The Blahut-Arimoto algorithm remains the standard computational tool for rate-distortion functions and channel capacities in information theory.

6 Beyond the Asymptotic Limit: Finite Block Length

In this section, we move beyond Shannon's asymptotic rate-distortion function and develop the theory of finite block length lossy compression. This is the mathematical core of the tutorial.

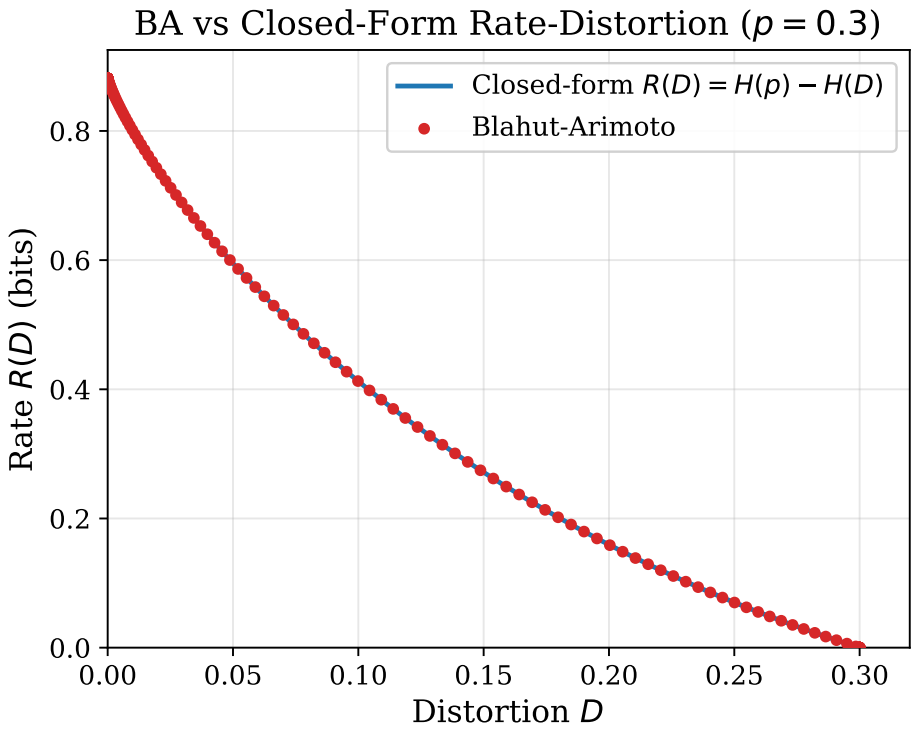

Figure 5: Comparison of the Blahut-Arimoto computed rate-distortion points (circles) with the closed-form curve $R(D) = H(p) - H(D)$ (solid line) for $p = 0.3$. The agreement is exact to numerical precision.

6.1 The Gap Between Theory and Practice

The rate-distortion function $R(D)$ tells us the ultimate limit of lossy compression as the block length $n \to \infty$. However, real systems operate with finite n. A practical compression system might use blocks of $n = 100$ or $n = 1000$ symbols. How much extra rate do we need compared to the Shannon limit?

Figure 6 illustrates the situation. For a Bernoulli(0.3) source with target distortion $D = 0.1$ and excess-distortion probability $\varepsilon = 0.1$, the achievable rate at $n = 100$ is significantly above $R(D)$, but the gap narrows as n grows. Understanding the precise rate of this convergence is the goal of finite block length theory.

6.2 The Finite Block Length Setup

Before diving into the formal definitions, let us build some intuition for why the finite block length setting requires a fundamentally different formulation than the asymptotic one.

The asymptotic rate-distortion function $R(D)$ is a deterministic quantity. This is because, in the limit $n \to \infty$, the law of large numbers guarantees that the empirical statistics of the source sequence X^n concentrate around their expected values. Almost every source sequence "looks the same" statistically, so a single codebook can handle all of them with average distortion at most D. There is no randomness left to worry about.

At finite block length, the situation is different. The source sequence X^n is random, and different realizations can be substantially easier or harder to compress. For a Bernoulli(0.3) source with $n = 100$, most sequences will have roughly 30 ones, but some will have 15 or 45. A code designed for the "typical" case may fail badly on these atypical sequences, producing a reconstruction whose distortion exceeds D. In short, the distortion achieved by any finite-n code is itself a random variable, because it depends on which source

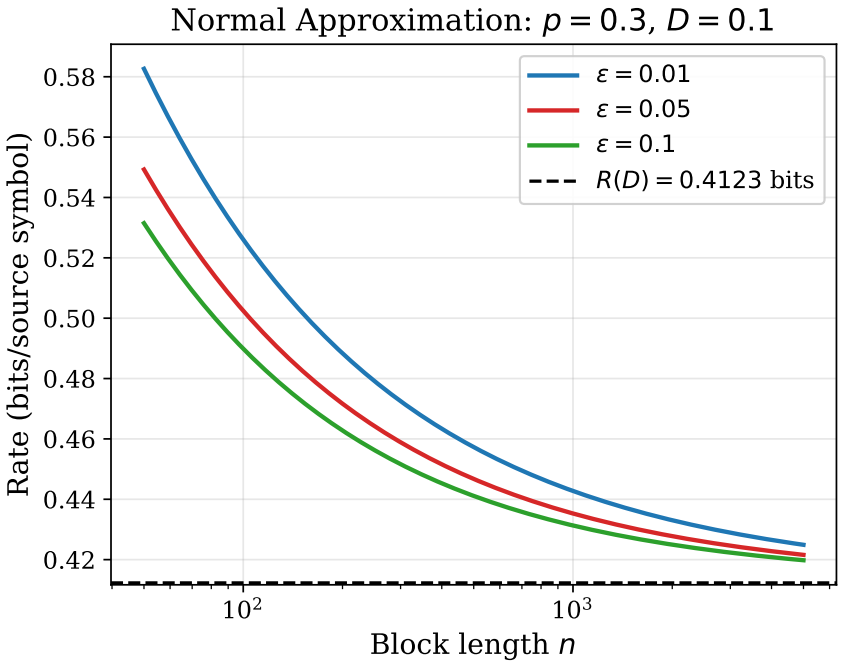

Figure 6: The minimum achievable rate $R(n, D, \varepsilon)$ versus block length n for a Bernoulli(0.3) source with $D = 0.1$ and several values of the excess-distortion probability ε. The horizontal dashed line shows the Shannon limit $R(D)$. The gap decays as $O(1/\sqrt{n})$.

sequence nature produces.

This means we cannot simply ask for the distortion to be at most D with certainty (that would require an enormous codebook) or merely on average (that would hide the possibility of catastrophic failures on some sequences). Instead, we allow a small *failure probability* ε: we accept that a fraction ε of source sequences may result in distortion exceeding D, but we require the code to succeed on the remaining $1 - \varepsilon$ fraction. The quantity $R(n, D, \varepsilon)$ is then deterministic again: it is the minimum rate at which a code of block length n exists that keeps the failure probability below ε. One can think of $R(n, D, \varepsilon)$ as a *confidence bound*: "with confidence $1 - \varepsilon$, a rate of $R(n, D, \varepsilon)$ bits per symbol suffices to compress the source to distortion at most D."

We now formalize this precisely.

Definition 6.1 ((n, M, D, ε) Code)**.** *An (n, M, D, ε) lossy source code consists of an encoder $f_n : \mathcal{X}^n \to \{1, \dots, M\}$ and a decoder $g_n : \{1, \dots, M\} \to \hat{\mathcal{X}}^n$ such that the* excess-distortion probability *satisfies*

$$\mathbb{P}\big(d(X^n, g_n(f_n(X^n))) > D\big) \le \varepsilon. \tag{37}$$

Note the shift from the asymptotic setting. Instead of requiring the *average* distortion to be at most D, we require that the distortion exceeds D with probability at most ε. This excess-distortion formulation is more natural for finite block lengths and leads to cleaner second-order results.

The *minimum achievable rate* at block length n is

$$R(n, D, \varepsilon) = \frac{1}{n} \log_2 M^*(n, D, \varepsilon), \tag{38}$$

where $M^*(n, D, \varepsilon)$ is the smallest codebook size M for which an (n, M, D, ε) code exists. The fundamental result of finite block length theory is a precise characterization of $R(n, D, \varepsilon)$.

6.3 The d-Tilted Information

The central single-letter quantity in the finite block length analysis is the *d-tilted information*, introduced by Kostina and Verdú [5].

Definition 6.2 (d-Tilted Information [5])**.** *For a source X with distribution p_X, distortion measure d, and target distortion D, the d-tilted information of a source realization x is*

$$\jmath_X(x, D) = D_{\mathrm{KL}}\big(p^*_{\hat{X}|X}(\cdot|x) \,\big\|\, Q^*\big) + \lambda^*\big(\mathbb{E}[d(x, \hat{X}) \mid X = x] - D\big), \tag{39}$$

where $\lambda^* = \log_2 \frac{1-D}{D}$ is the optimal Lagrange multiplier (28), Q^* is the optimal reproduction distribution (26)–(27), and $p^*_{\hat{X}|X}$ is the optimal forward channel (25). All quantities are measured in bits (using $\log_2$). Equivalently, using the Gibbs form $p^*_{\hat{X}|X}(\hat{x}|x) = Q^*(\hat{x})\, 2^{-\lambda^* d(x,\hat{x})}/Z(x)$ with normalizing constant $Z(x) = \sum_{\hat{x}} Q^*(\hat{x})\, 2^{-\lambda^* d(x,\hat{x})}$, the definition simplifies to

$$\jmath_X(x, D) = -D \log_2 \frac{1-D}{D} + \log_2 \frac{1}{Z(x)}. \tag{40}$$

The d-tilted information has a compelling interpretation: it measures how "difficult" it is to compress a particular source realization x to distortion level D. Different source symbols may be easier or harder to compress, and $\jmath_X(x, D)$ captures this variation.

A key property is that the expected d-tilted information equals the rate-distortion function:

$$\mathbb{E}[\jmath_X(X, D)] = R(D), \tag{41}$$

where the expectation is over the source distribution: $\mathbb{E}[\jmath_X(X, D)] = \sum_x p_X(x)\, \jmath_X(x, D) = (1-p)\, \jmath_X(0, D) + p\, \jmath_X(1, D)$ for the Bernoulli source.

Proof. We give two proofs: an explicit algebraic computation for the Bernoulli source, and a short information-theoretic argument that works in general.

Part A: Algebraic proof for the Bernoulli source. We use the equivalent form (40). The normalizing constant is $Z(x) = \sum_{\hat{x}} Q^*(\hat{x})\, 2^{-\lambda^* d(x,\hat{x})}$ with Q^* from (24) and $\lambda^* = \log_2 \frac{1-D}{D}$, so $2^{-\lambda^*} = \frac{D}{1-D}$.

Step 1: Compute $Z(0)$ and $Z(1).$

$$Z(0) = Q^*(0) \cdot 1 + Q^*(1) \cdot \frac{D}{1-D} = \frac{Q^*(0)(1-D) + Q^*(1)\, D}{1-D}.$$

Substituting $Q^*(0) = \frac{1-p-D}{1-2D}$ and $Q^*(1) = \frac{p-D}{1-2D}$:

$$Q^*(0)(1-D) + Q^*(1)\, D = \frac{(1-p-D)(1-D) + (p-D)D}{1-2D}.$$

Expanding the numerator: $(1-p-D)(1-D) + (p-D)D = (1-p)(1-D) - D(1-D) + pD - D^2 = (1-p)(1-D) - D(1-p) = (1-p)(1-2D)$. Therefore

$$Z(0) = \frac{1-p}{1-D}, \qquad Z(1) = \frac{p}{1-D}, \tag{42}$$

where $Z(1)$ follows by an identical calculation (or by the symmetry $Z(1) = Q^*(0)\frac{D}{1-D} + Q^*(1)$, which gives numerator $p(1-2D)$).

Step 2: Compute $\jmath_X(0, D)$ and $\jmath_X(1, D)$. Plugging (42) into (40):

$$\jmath_X(0, D) = -D \log_2 \frac{1-D}{D} + \log_2 \frac{1-D}{1-p}, \tag{43}$$

$$\jmath_X(1, D) = -D \log_2 \frac{1-D}{D} + \log_2 \frac{1-D}{p}. \tag{44}$$

Step 3: Compute the expectation.

$$\begin{aligned}\mathbb{E}[\jmath_X(X, D)] &= (1-p)\, \jmath_X(0, D) + p\, \jmath_X(1, D) \\ &= \underbrace{-D \log_2 \frac{1-D}{D}}_{\text{common first term}} + \underbrace{(1-p) \log_2 \frac{1-D}{1-p} + p \log_2 \frac{1-D}{p}}_{\text{weighted second terms}}.\end{aligned} \tag{45}$$

Step 4: Simplify the second group. Splitting the logarithms:

$$(1-p) \log_2 \frac{1-D}{1-p} + p \log_2 \frac{1-D}{p} = \underbrace{\big[(1-p) + p\big]}_{=1} \log_2(1-D) \; + \; \underbrace{\big[-(1-p)\log_2(1-p) - p \log_2 p\big]}_{=H(p)}.$$

Step 5: Combine all terms.

$$\begin{aligned}\mathbb{E}[\jmath_X(X, D)] &= -D \log_2 \frac{1-D}{D} + \log_2(1-D) + H(p) \\ &= -D \log_2(1-D) + D \log_2 D + \log_2(1-D) + H(p) \\ &= (1-D) \log_2(1-D) + D \log_2 D + H(p) \\ &= -H(D) + H(p) = H(p) - H(D) = R(D). \qquad \square\end{aligned}$$

Part B: Information-theoretic proof (general sources). From the definition (39), taking the expectation over X:

$$\mathbb{E}[\jmath_X(X, D)] = \mathbb{E}_X\big[D_{\mathrm{KL}}(p^*_{\hat{X}|X}(\cdot|X) \| Q^*)\big] + \lambda^*\big(\mathbb{E}[d(X, \hat{X}^*)] - D\big).$$

The first term is $\sum_x p_X(x)\, D_{\mathrm{KL}}(p^*_{\hat{X}|X}(\cdot|x) \| Q^*) = I(X; \hat{X}^*)$, since mutual information decomposes as the expected KL divergence of the conditional from the marginal. The second term vanishes because the optimal test channel meets the distortion constraint with equality: $\mathbb{E}[d(X, \hat{X}^*)] = D$. Therefore

$$\mathbb{E}[\jmath_X(X, D)] = I(X; \hat{X}^*) = R(D). \qquad \square$$

For the Bernoulli source, the accompanying Python code confirms that $(1-p)\, \jmath_X(0, D) + p\, \jmath_X(1, D) = H(p) - H(D)$ to machine precision for all tested values of p and D. $\square$

This identity confirms that $\jmath_X(x, D)$ is the correct "information density" for the lossy compression problem: it decomposes the rate $R(D)$ into per-symbol contributions, just as the log-likelihood ratio decomposes mutual information in channel coding.

For the Bernoulli(p) source with Hamming distortion, the optimal reproduction distribution is given by (26)–(27), and the Lagrange multiplier is $\lambda^* = \log_2 \frac{1-D}{D}$. Using the simplified normalizing constants $Z(0) = \frac{1-p}{1-D}$ and $Z(1) = \frac{p}{1-D}$ from (42), the d-tilted information takes two values:

$$\jmath_X(0, D) = -D \log_2 \frac{1-D}{D} + \log_2 \frac{1-D}{1-p}, \tag{46}$$

$$\jmath_X(1, D) = -D \log_2 \frac{1-D}{D} + \log_2 \frac{1-D}{p}. \tag{47}$$

Figure 7 shows the d-tilted information for both source symbols as a function of D.

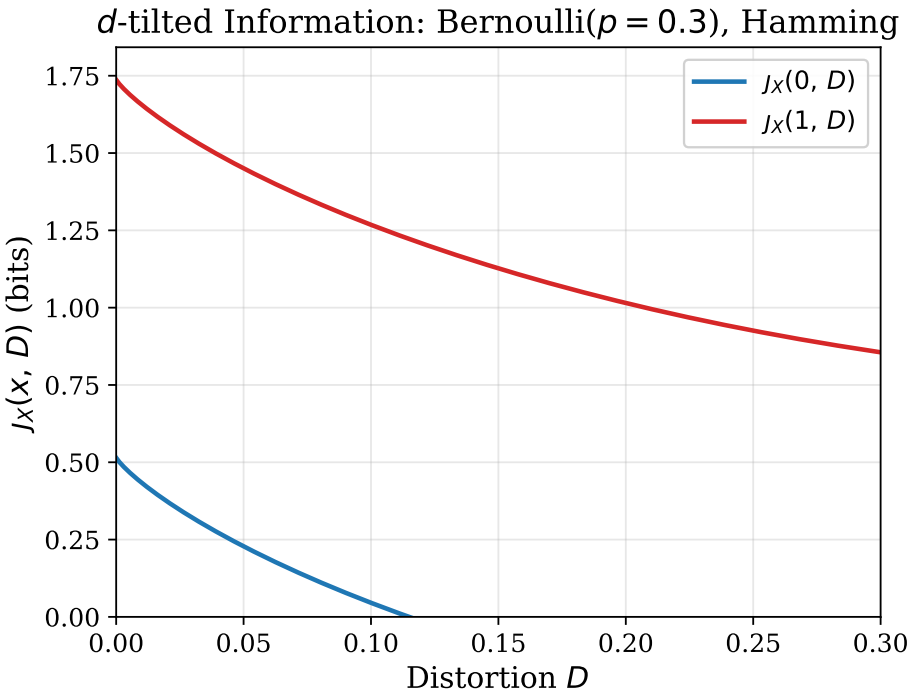

Figure 7: The d-tilted information $\jmath_X(0, D)$ and $\jmath_X(1, D)$ for a Bernoulli(0.3) source with Hamming distortion. When D is small, both values are close to $H(p)$ (the lossless rate). As D increases toward $\min(p, 1-p) = 0.3$, both converge to zero. The gap between the two curves reflects the asymmetry of the source.

6.4 Dispersion: The Key Second-Order Quantity

The rate-distortion function $R(D)$ is a first-order quantity: it captures the leading-order behavior as $n \to \infty$. The *rate-distortion dispersion* is the second-order quantity that governs how quickly the finite block length rate converges to $R(D)$.

Definition 6.3 (Rate-Distortion Dispersion)**.** *The rate-distortion dispersion at distortion level D is*

$$V(D) = \mathrm{Var}[\jmath_X(X, D)]. \tag{48}$$

The dispersion measures how *variable* the compression difficulty is across source symbols. If all source symbols are equally difficult to compress, then $V(D) = 0$ and the convergence to $R(D)$ is faster than $1/\sqrt{n}$. If different symbols have very different compression difficulties, then $V(D)$ is large and the $1/\sqrt{n}$ penalty is more pronounced.

For the Bernoulli(p) source, since X takes only two values, we have

$$V(D) = p(1-p)\big(\jmath_X(1, D) - \jmath_X(0, D)\big)^2. \tag{49}$$

This is simply the variance of a Bernoulli random variable that takes value $\jmath_X(0, D)$ with probability $1-p$ and $\jmath_X(1, D)$ with probability p.

Remark 6.1. *When $p = 1/2$, the source is symmetric: $\jmath_X(0, D) = \jmath_X(1, D)$ for all D, and therefore $V(D) = 0$. Intuitively, every symbol of a fair coin is equally difficult to compress, so there is no variability in compression difficulty. This is a somewhat surprising consequence: for the fair Bernoulli source, the convergence to $R(D)$ is faster than $O(1/\sqrt{n})$. When $V(D) = 0$, the normal approximation (50) does not apply (it requires $V(D) > 0$); the $\sqrt{V/n}$ term vanishes and the $O(\log n/n)$ remainder becomes the dominant correction.*

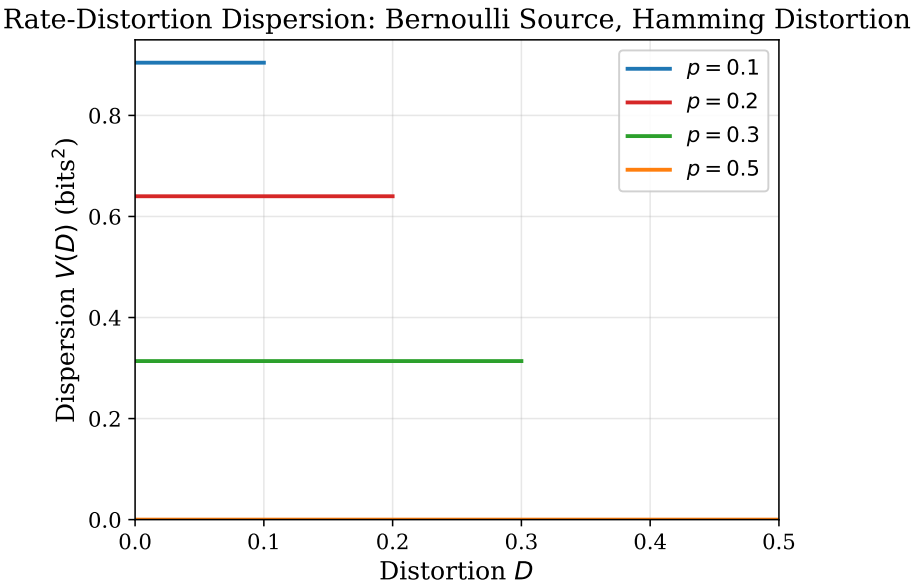

Figure 8: The rate-distortion dispersion $V(D)$ versus distortion D for a Bernoulli(p) source with $p \in \{0.1, 0.2, 0.3, 0.5\}$. For $p = 0.5$, the dispersion is identically zero (the source symbols are equally difficult to compress). For biased sources, the dispersion is largest at intermediate distortion levels.

Figure 8 shows $V(D)$ as a function of D for several source biases.

6.5 The Normal Approximation

We now state the central result of finite block length rate-distortion theory. The minimum achievable rate at block length n is characterized by the following asymptotic expansion.

Theorem 6.1 (Normal Approximation [5])**.** *For a discrete memoryless source with rate-distortion function $R(D)$ and dispersion $V(D) > 0$, the minimum rate at block length n and excess-distortion probability $\varepsilon \in (0, 1)$ satisfies*

$$R(n, D, \varepsilon) = R(D) + \sqrt{\frac{V(D)}{n}}\, Q^{-1}(\varepsilon) + O\left(\frac{\log n}{n}\right), \tag{50}$$

where $Q^{-1}(\varepsilon)$ is the inverse of the Gaussian Q-function, $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2}\, dt$.

The intuition behind this result comes from the Berry-Esseen central limit theorem. The total compression cost for an i.i.d. source sequence X^n is approximately $\sum_{i=1}^n \jmath_X(X_i, D)$, a sum of i.i.d. random variables with mean $nR(D)$ and variance $nV(D)$. By the CLT, this sum is approximately Gaussian, and the excess-distortion probability translates into a Gaussian tail probability. The $Q^{-1}(\varepsilon)$ factor converts the target probability ε into the number of standard deviations we must accommodate.

From an engineering standpoint, Theorem 6.1 provides a practical design rule: to operate within rate $R(D) + \Delta R$ of the Shannon limit with excess-distortion probability ε, we need a block length of approximately

$$n \approx \frac{V(D)\big(Q^{-1}(\varepsilon)\big)^2}{(\Delta R)^2}. \tag{51}$$

This expression reveals the fundamental trade-off among block length, rate overhead, distortion, and reliability.

Figure 9 shows the achievability bound, converse bound, and normal approximation for a Bernoulli(0.3) source.

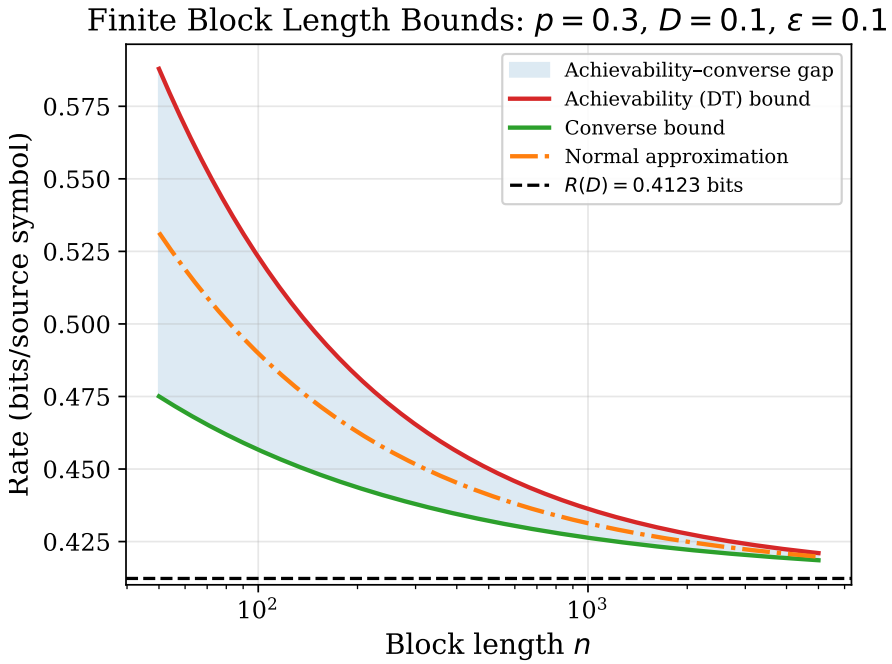

Figure 9: Finite block length bounds for a Bernoulli(0.3) source with $D = 0.1$ and $\varepsilon = 0.1$. The shaded region between the achievability bound (upper) and converse bound (lower) contains the true minimum rate $R(n, D, \varepsilon)$. The normal approximation (dashed) lies within this region. The horizontal line shows the Shannon limit $R(D)$.

6.6 Historical Note

The study of finite block length performance in information theory dates back to Strassen [3], who established the $O(1/\sqrt{n})$ refinement for hypothesis testing. The channel coding counterpart was developed by Polyanskiy, Poor, and Verdú [11], whose work on channel dispersion inspired the lossy source coding treatment. Kostina and Verdú [5] established the rate-distortion dispersion and the normal approximation (50) for general discrete memoryless sources, building on contributions by Ingber and Kochman [4]. The d-tilted information framework provides a unified language for second-order information theory.

7 Numerical Explorations

Every figure in this tutorial was generated programmatically by accompanying Python scripts. The scripts, along with the LaTeX source and all generated figures, are publicly available at: `https://github.com/anrgusc/rate-distortion-bernoulli-finite`.

7.1 Computational Tools and Figures

We provide five Python scripts in the `scripts/` directory:

- `rate_distortion.py`: Computes and plots the binary entropy function (Figure 1) and rate-distortion curves (Figure 3).
- `blahut_arimoto.py`: Implements the Blahut-Arimoto algorithm with convergence tracking (Figure 4) and validates it against the closed-form solution (Figure 5).

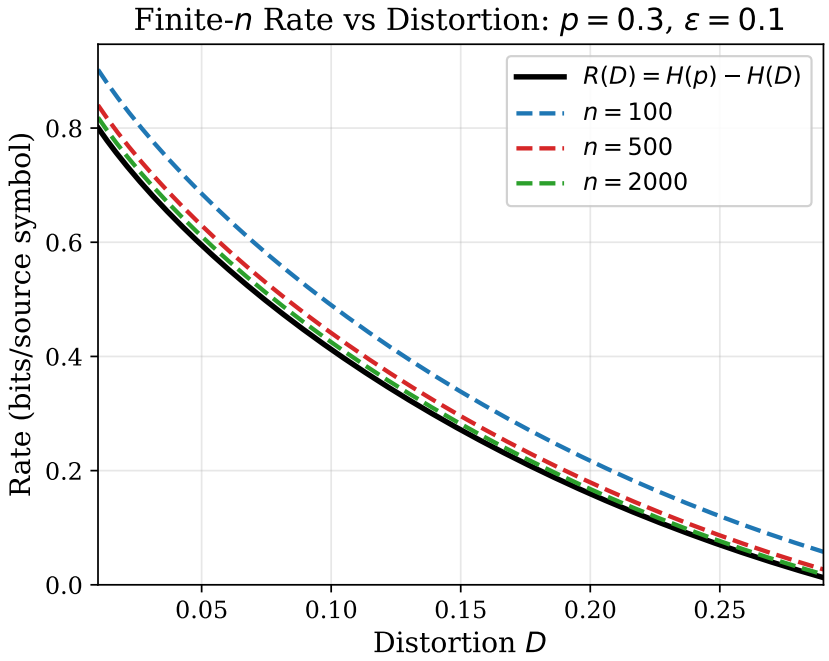

Figure 10: The rate-distortion function $R(D)$ (solid black) and the normal approximation $R(n, D, \varepsilon)$ for block lengths $n \in \{100, 500, 2000\}$, with $p = 0.3$ and $\varepsilon = 0.1$. The finite block length penalty is largest at small D (high rate regime) and vanishes as $D \to \min(p, 1 - p)$.

- `dispersion.py`: Computes the d-tilted information (Figure 7) and rate-distortion dispersion (Figure 8). This script also runs a numerical verification that $\mathbb{E}[\jmath_X(X, D)] = R(D)$ to machine precision.
- `finite_blocklength.py`: Computes the normal approximation, achievability/converse bounds (Figure 9), and the rate versus block length curves (Figure 6).
- `generate_all_figures.py`: Master script that runs all of the above and generates every figure with consistent styling.

To regenerate all figures from scratch, run `python scripts/generate_all_figures.py` from the repository root. The only dependencies are NumPy, SciPy, and Matplotlib (see `requirements.txt`).

7.2 Comprehensive Comparison

Figure 10 brings together the asymptotic rate-distortion function and its finite block length refinements in a single plot. For a Bernoulli(0.3) source with excess-distortion probability $\varepsilon = 0.1$, we show the achievable rate as a function of distortion D for several block lengths $n \in \{100, 500, 2000\}$. As n increases, the finite block length curves converge uniformly to the Shannon limit $R(D)$. The gap is most pronounced at small distortions, where the dispersion $V(D)$ is larger.

7.3 Blahut-Arimoto Convergence

The Blahut-Arimoto algorithm exhibits geometric convergence, as shown in Figure 4. The convergence rate depends on the slope parameter s: larger s (corresponding to smaller target distortions) leads to faster convergence in terms of iteration count. For all tested parameters, the algorithm reaches a relative error below 10^{-10} within 50 iterations.

8 Conclusion

In this tutorial, we have presented a self-contained development of rate-distortion theory for the Bernoulli(p) source with Hamming distortion, progressing from Shannon's asymptotic limit to the finite block length regime.

The key takeaways are threefold. First, the rate-distortion function $R(D) = H(p) - H(D)$ provides a clean and interpretable limit on lossy compression: the minimum rate is the source entropy minus the noise entropy. Second, the Blahut-Arimoto algorithm provides a reliable computational tool for evaluating rate-distortion functions, even in settings where closed-form solutions are unavailable. Third, the finite block length theory, centered on the d-tilted information $\jmath_X(x, D)$ and the dispersion $V(D)$, provides a precise characterization of the penalty for operating at practical block lengths. The normal approximation $R(n, D, \varepsilon) \approx R(D) + \sqrt{V(D)/n}\, Q^{-1}(\varepsilon)$ is both elegant in form and useful in practice, offering a direct design rule for system engineers.

Acknowledgement

The development of this tutorial and the associated Python code has involved the use of several AI tools/agents including Claude Code, ChatGPT, and Gemini. The human author accepts full responsibility for its contents.

References

[1] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 623–656, 1948.

[2] ——, "Coding theorems for a discrete source with a fidelity criterion," in *IRE National Convention Record, Part 4*, vol. 7, 1959, pp. 142–163.

[3] V. Strassen, "Asymptotische Abschätzungen in Shannons Informationstheorie," in *Transactions of the Third Prague Conference on Information Theory, Statistical Decision Functions, Random Processes*, Prague, 1962, pp. 689–723.

[4] A. Ingber and Y. Kochman, "The dispersion of lossy source coding," in *2011 Data Compression Conference*, 2011, pp. 53–62.

[5] V. Kostina and S. Verdú, "Fixed-length lossy compression in the finite blocklength regime," *IEEE Transactions on Information Theory*, vol. 58, no. 6, pp. 3309–3338, Jun. 2012.

[6] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. Wiley-Interscience, 2006.

[7] T. Berger, *Rate Distortion Theory: A Mathematical Basis for Data Compression*. Englewood Cliffs, NJ: Prentice-Hall, 1971.

[8] R. E. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Transactions on Information Theory*, vol. 18, no. 4, pp. 460–473, Jul. 1972.

[9] S. Arimoto, "An algorithm for computing the capacity of arbitrary discrete memoryless channels," *IEEE Transactions on Information Theory*, vol. 18, no. 1, pp. 14–20, Jan. 1972.

[10] I. Csiszár, "On the computation of rate-distortion functions," *IEEE Transactions on Information Theory*, vol. 20, no. 1, pp. 122–124, 1974.

[11] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2307–2359, 2010.