6.7720/18.619/15.070 Lecture 5
Shannon's Noisy Coding Theorem

Kuikui Liu

February 18, 2025

Acknowledgements & Disclaimers *In the process of writing these notes, we consulted materials by Venkatesan Guruswami and Michel X. Goemans. Please be advised that these notes have not been subjected to the usual scrutiny reserved for formal publications. If you do spot an error, please contact the instructor.*

1 Error-Correction and Shannon's Theorem

In this lecture, we use the probabilistic method combined with the Weak Law of Large Numbers to prove a remarkable characterization of the fundamental limits of error-correcting mechanisms for communication in the presence of noise. We will work with the following bare-bones definition of an error-correcting code.

Definition 1 (Error-Correcting Code)**.** *For positive integers $k, n \in \mathbb{N}$ satisfying $n \geq k$, an* (error-correcting) $[n, k]$-code *is a pair of functions* $\mathsf{Enc} : \{0,1\}^k \to \{0,1\}^n$ *and* $\mathsf{Dec} : \{0,1\}^n \to \{0,1\}^k$*. We will often refer to the former as the* encoder*, and the latter as the* decoder*. The* rate *of the code is the ratio $r = \frac{k}{n}$. We often refer to the elements of $\{0,1\}^k$ as* messages*, and the elements of $\{0,1\}^n$ as* codewords*.*

One should imagine that a sender would like to transmit a message $\boldsymbol{m} \in \{0,1\}^k$ to a receiver, but only has access to a noisy communication channel. Hence, the sender will first encode the message, and then send $\mathsf{Enc}(\boldsymbol{m})$ through the channel. The hope is that even though the codeword $\mathsf{Enc}(\boldsymbol{m})$ can suffer errors during transmission, the receiver will be able to recover $\boldsymbol{m}$ by applying the decoder Dec to the (possibly corrupted) codeword they received. In practice, we typically want our error-correcting codes to have additional properties, e.g. *linearity*, *locality*, *efficient computability* of $\mathsf{Enc}, \mathsf{Dec}$, etc. We will not be concerned with these in this lecture.

To make the problem well-defined, we also need to fix a model for how errors will be introduced. In this lecture, we will only consider a very special type of error, namely every bit of the codeword can be flipped independently with some probability $0 \leq p \leq 1/2$. This *channel* should be familiar; we saw it in the context of the broadcast process on trees.

Definition 2 (Binary Symmetric Channel)**.** *For $0 \leq p \leq 1/2$, the* binary symmetric channel *with parameter p, denoted BSC_p, is described by the following randomized operation: For any bit-string $\boldsymbol{x} \in \{0,1\}^*$ of any length n, $\boldsymbol{y} = \mathsf{BSC}_p(\boldsymbol{x})$ is a random length-n bit-string such that independently for each $i = 1, \ldots, n$, $\boldsymbol{y}_i = \boldsymbol{x}_i$ with probability $1 - p$ and $\boldsymbol{y}_i \neq \boldsymbol{x}_i$ with probability p.*

One can, of course, consider more exotic error models (e.g. random erasures, random swaps, correlated bit flips, etc.), as well as *worst-case* error models.

For an error-correcting code $(\mathsf{Enc}, \mathsf{Dec})$, we will refer to

$$\Pr\left[\mathsf{Dec}\left(\mathsf{BSC}_p\left(\mathsf{Enc}(\boldsymbol{m})\right)\right) \neq \boldsymbol{m}\right] \qquad \text{and} \qquad \Pr\left[\mathsf{Dec}\left(\mathsf{BSC}_p\left(\mathsf{Enc}(\boldsymbol{m})\right)\right) = \boldsymbol{m}\right]$$

as the *failure/error* and *success* probabilities of the message $\boldsymbol{m}$, respectively. The former (resp. latter) is the probability that if we send the encoded message $\mathsf{Enc}(\boldsymbol{m})$ through the noisy BSC_p channel, then the receiver will be unable (resp. able) to recover $\boldsymbol{m}$ using the decoding function $\mathsf{Dec}(\cdot)$.

Naturally, there is a tradeoff between the rate of an error-correcting code, which quantifies its efficiency, and the (worst-case) failure probability of any given message. For instance, if you allow

arbitrarily small rates (i.e. arbitrarily inefficient codes), then you can drive down the failure probabilities uniformly by using something like the *repetition code*, which simply copies each message bit some large number of times.

Remarkably, there turns out to be a specific threshold for the rate, above which every error-correcting code must fail for some message with probability close to 1, and below which there exists an error-correcting code with failure probabilities uniformly close to 0. This phase transition phenomenon is the content of Shannon's seminal Noisy Coding Theorem. To state it, define the *binary entropy function* by

$$H(p) \stackrel{\mathsf{def}}{=} -p \log_2 p - (1-p)\log_2(1-p), \qquad \forall p \in [0,1].$$

Theorem 1.1 (Shannon's Noisy Coding Theorem; **[???]**)**.** *Fix any constant $0 \le p < 1/2$.*

- *Suppose $r > 1 - H(p)$. Then for every $n \in \mathbb{N}$ and every $[n,k]$-code $(\mathsf{Enc}, \mathsf{Dec})$ with $k = rn$,*

$$\max_{\boldsymbol{m} \in \{0,1\}^k} \Pr\left[\mathsf{Dec}\left(\mathsf{BSC}_p\left(\mathsf{Enc}(\boldsymbol{m})\right)\right) \neq \boldsymbol{m}\right] \ge 1 - o(1).$$

- *Suppose $r < 1 - H(p)$. Then for every $n \in \mathbb{N}$, there exists an $[n,k]$-code $(\mathsf{Enc}, \mathsf{Dec})$ with $k = rn$ such that*

$$\max_{\boldsymbol{m} \in \{0,1\}^k} \Pr\left[\mathsf{Dec}\left(\mathsf{BSC}_p\left(\mathsf{Enc}(\boldsymbol{m})\right)\right) \neq \boldsymbol{m}\right] \le o(1).$$

Remark 1. We note that Shannon's Noisy Coding Theorem actually holds for much more general channels beyond the BSC_p, although one must replace $1 - H(p)$ by a more general information-theoretic quantity known as the *channel capacity*.

To prove Theorem 1.1, we will need an extremely weak concentration bound for sums of independent random variables, which can be obtained directly from the second moment. By leveraging more powerful concentration estimates like *Chernoff bounds* (which we introduce in the next lecture), one can obtain strong quantitative control on the $o(1)$ terms in Theorem 1.1.

Theorem 1.2 (Weak Law of Large Numbers)**.** *Let $\{X_i\}_{i=1}^{\infty}$ be a sequence of independent and identically distributed (i.i.d.) random variables with mean μ and variance $\sigma^2 < \infty$. Then*

$$\lim_{n \to \infty} \Pr\left[\left|\frac{1}{n}\sum_{i=1}^{n} X_i - \mu\right| < \epsilon\right] = 1, \qquad \forall \epsilon > 0.$$

In other words, we have convergence in probability $\frac{1}{n}\sum_{i=1}^{n} X_i \xrightarrow{\mathsf{P}} \mu$ as $n \to \infty$.

Remark 2. As the proof shows, one can significantly relax the joint independence assumption to merely *pairwise independence*, i.e. for all $i \neq j$ and all x_i, x_j, $\Pr[X_i = x_i, X_j = x_j] = \Pr[X_i = x_i] \cdot \Pr[X_j = x_j]$. A direct calculation reveals this latter condition is sufficient to guarantee that the variance of the sum is equal to the sum of the variances.

Proof. By independence, we have $\mathrm{Var}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) = \frac{\sigma^2}{n}$. Hence, by Chebyshev's Inequality,

$$\Pr\left[\left|\frac{1}{n}\sum_{i=1}^{n} X_i - \mu\right| \ge \epsilon\right] \le \frac{\mathrm{Var}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right)}{\epsilon^2} = \frac{\sigma^2}{\epsilon^2} \cdot \frac{1}{n}.$$

□

Finally, to quantify the amount of errors introduced to a codeword, we define a metric on bit-strings known as *Hamming distance*:

$$d_H(\boldsymbol{x}, \boldsymbol{y}) \stackrel{\mathsf{def}}{=} \|\boldsymbol{x} - \boldsymbol{y}\|_1 = \#\{i \in [n] : \boldsymbol{x}_i \neq \boldsymbol{y}_i\}, \qquad \forall \boldsymbol{x}, \boldsymbol{y} \in \{0,1\}^n.$$

We have the following bound on the volume of Hamming balls, similar to what we used when we looked at the connectivity threshold of Erdös–Rényi random graphs.

Lemma 1.3 (Volume of a Hamming Ball)**.** *For* $0 \le p \le 1/2$*,*

$$\#\{\boldsymbol{x} \in \{0,1\}^n : d_H(\mathbf{0}, \boldsymbol{x}) \le pn\} = \sum_{k=0}^{pn} \binom{n}{k} \le 2^{H(p)\cdot n}.$$

Proof. The first identity is immediate. For the second identity, we have

$$1 = (p + (1-p))^n \ge \sum_{k=0}^{pn} \binom{n}{k} p^k (1-p)^{n-k} \ge p^{pn}(1-p)^{(1-p)n} \sum_{k=0}^{pn} \binom{n}{k} = 2^{-H(p)\cdot n} \sum_{k=0}^{pn} \binom{n}{k},$$

where in the second inequality, we used $0 \le p \le 1/2$ and the fact that the function $k \mapsto p^k(1-p)^{n-k}$ is monotone decreasing for $0 \le k \le pn$. Rearranging then yields the upper bound. □

2 Proof of Theorem 1.1: The Case $r > 1 - H(p)$

We will go via the contrapositive, and show that if there exists an error-correcting code for which the success probability is uniformly lower bounded by some positive constant $\epsilon > 0$ for all possible messages $\boldsymbol{m} \in \{0,1\}^k$, then it must be that $r \le 1 - H(p) + o(1)$. To achieve this, our goal will be to argue that any message $\boldsymbol{m}$ with constant success probability must have at least $2^{(H(p)-o(1))\cdot n}$ codewords $\boldsymbol{x} \in \{0,1\}^n$ which decode to $\boldsymbol{m}$, i.e. $\mathsf{Dec}(\boldsymbol{x}) = \boldsymbol{m}$. Since the preimages of the decoder Dec partition $\{0,1\}^n$, this will imply a bound on the number of possible such messages $\boldsymbol{m}$. This is a classic example of a *volume argument*.

Let us now formalize our strategy.

Proposition 2.1. *For any constant* $\epsilon > 0$ *independent of* n*, if* $\boldsymbol{m} \in \{0,1\}^k$ *is a message with success probability at least* ϵ*, then*

$$\#\{\boldsymbol{x} \in \{0,1\}^n : \mathsf{Dec}(\boldsymbol{x}) = \boldsymbol{m}\} \ge 2^{(H(p)-o(1))\cdot n}.$$

Let us first quickly see how this can be used to show $r \le 1 - H(p) + o(1)$ and complete the proof of the first case of Theorem 1.1. For $r \in [0,1]$, suppose there exists $\epsilon > 0$ such that for infinitely many $n \in \mathbb{N}$, there exist $\mathsf{Enc} : \{0,1\}^k \to \{0,1\}^n$ and $\mathsf{Dec} : \{0,1\}^n \to \{0,1\}^k$ with $k = rn$ such that

$$\max_{\boldsymbol{m} \in \{0,1\}^k} \Pr\left[\mathsf{Dec}\left(\mathsf{BSC}_p\left(\mathsf{Enc}(\boldsymbol{m})\right)\right) \ne \boldsymbol{m}\right] \le 1 - \epsilon$$

Then the success probability of every $\boldsymbol{m} \in \{0,1\}^k$ is uniformly lower bounded by ϵ. Hence,

$$\begin{aligned} 2^n &\ge \sum_{\boldsymbol{m} \in \{0,1\}^k} \#\{\boldsymbol{x} \in \{0,1\}^n : \mathsf{Dec}(\boldsymbol{x}) = \boldsymbol{m}\} && \text{(Preimages of Dec partition } \{0,1\}^n) \\ &\ge 2^k \cdot 2^{(H(p)-o(1))\cdot n}. && \text{(Invoking Proposition 2.1)} \end{aligned}$$

Rearranging then yields the inequality $r \le 1 - H(p) + o(1)$. Since this holds for infinitely many n, we must have $r \le 1 - H(p)$ as desired.

Proof of Proposition 2.1. Our assumption says that applying BSC_p to $\mathsf{Enc}(\boldsymbol{m})$ yields a random bit-string which happens to fall in the preimage $\{\boldsymbol{x} \in \{0,1\}^n : \mathsf{Dec}(\boldsymbol{x}) = \boldsymbol{m}\}$ with constant probability:

$$\Pr\left[\mathsf{BSC}_p\left(\mathsf{Enc}(\boldsymbol{m})\right) \in \mathsf{Dec}^{-1}(\boldsymbol{m})\right] \ge \epsilon.$$

Our goal is to use this lower bound to deduce a lower bound on the cardinality of the preimage $\mathsf{Dec}^{-1}(\boldsymbol{m})$. Observe that if $A \subseteq \{0,1\}^n$ is any subset of codewords, then

$$\left|\mathsf{Dec}^{-1}(\boldsymbol{m})\right| \ge \left|\mathsf{Dec}^{-1}(\boldsymbol{m}) \cap A\right| = \Pr_{\boldsymbol{x} \sim \mathsf{Unif}(A)}\left[\boldsymbol{x} \in \mathsf{Dec}^{-1}(\boldsymbol{m})\right] \cdot |A|.$$

We will employ an inequality of this flavor. Somewhat inconveniently, the probability in our assumption is with respect to a nonuniform distribution over $\{0,1\}^n$. However, we can get around this by using concentration. For convenience, let $\boldsymbol{y} = \mathsf{Enc}(\boldsymbol{m})$. For $\delta > 0$, let A_δ denote the *annulus* around $\boldsymbol{y}$

$$A_\delta \stackrel{\mathsf{def}}{=} \{\boldsymbol{x} \in \{0,1\}^n : d_H(\boldsymbol{x}, \boldsymbol{y}) \in [p - \delta, p + \delta]\}.$$

The idea is to choose $\delta(n) \to 0$ so that we get some "approximate uniformity"; in particular, all codewords in $\mathsf{Dec}^{-1}(\boldsymbol{m}) \cap A_{\delta(n)}$ will have approximately the same probability mass under $\mathsf{BSC}_p(\boldsymbol{y})$. By Theorem 1.2 (or its proof), we can make the decay $\delta(n) \to 0$ sufficiently slow while maintaining

$$\Pr\left[\mathsf{BSC}_p(\boldsymbol{y}) \in A_{\delta(n)}\right] \geq 1 - \frac{\epsilon}{2}.$$

This will ensure that we do not incur too much loss when we pass from $\mathsf{BSC}_p(\boldsymbol{y})$ to the conditional distribution $\mathsf{BSC}_p(\boldsymbol{y}) \mid A_{\delta(n)}$. Let us now make this formal.

Applying the Law of Total Probability, we have

$$\begin{aligned}
\epsilon &\leq \Pr\left[\mathsf{BSC}_p(\boldsymbol{y}) \in \mathsf{Dec}^{-1}(\boldsymbol{m})\right] \\
&= \Pr\left[\mathsf{BSC}_p(\boldsymbol{y}) \in \mathsf{Dec}^{-1}(\boldsymbol{m}) \cap A_{\delta(n)}\right] \\
&\qquad + \Pr\left[\mathsf{BSC}_p(\boldsymbol{y}) \in \mathsf{Dec}^{-1}(\boldsymbol{m}) \mid \mathsf{BSC}_p(\boldsymbol{y}) \notin A_{\delta(n)}\right] \cdot \Pr\left[\mathsf{BSC}_p(\boldsymbol{y}) \notin A_{\delta(n)}\right] \\
&\leq \Pr\left[\mathsf{BSC}_p(\boldsymbol{y}) \in \mathsf{Dec}^{-1}(\boldsymbol{m}) \mid \mathsf{BSC}_p(\boldsymbol{y}) \in A\right] + \frac{\epsilon}{2}. \qquad \text{(Using Theorem 1.2)}
\end{aligned}$$

Hence, we obtain on the one hand

$$\Pr\left[\mathsf{BSC}_p(\boldsymbol{y}) \in \mathsf{Dec}^{-1}(\boldsymbol{m}) \cap A_{\delta(n)}\right] \geq \frac{\epsilon}{2} = \Omega(1).$$

On the other hand,

$$\begin{aligned}
\Pr\left[\mathsf{BSC}_p(\boldsymbol{y}) \in \mathsf{Dec}^{-1}(\boldsymbol{m}) \cap A_{\delta(n)}\right] &= \sum_{\boldsymbol{x} \in \mathsf{Dec}^{-1}(\boldsymbol{m}) \cap A_{\delta(n)}} \Pr\left[\mathsf{BSC}_p(\boldsymbol{y}) = \boldsymbol{x}\right] \\
&\leq p^{(p-\delta(n))n}(1-p)^{(1-p+\delta(n))n} \cdot \left|\mathsf{Dec}^{-1}(\boldsymbol{m}) \cap A_{\delta(n)}\right| \\
&= 2^{-(H(p)-o(1))\cdot n} \cdot \left|\mathsf{Dec}^{-1}(\boldsymbol{m}) \cap A_{\delta(n)}\right|,
\end{aligned}$$

where the $o(1)$ is due to the fact that $\delta(n) \to 0$ as $n \to \infty$. Rearranging then yields

$$\left|\mathsf{Dec}^{-1}(\boldsymbol{m})\right| \geq \left|\mathsf{Dec}^{-1}(\boldsymbol{m}) \cap A_{\delta(n)}\right| \geq \frac{\epsilon}{2} \cdot 2^{(H(p)-o(1))\cdot n} \geq 2^{(H(p)-o(1))\cdot n},$$

where we may absorb $\frac{\epsilon}{2}$ into $2^{o(n)}$. □

3 Proof of Theorem 1.1: The Case $r < 1 - H(p)$

Our task is to design an error-correcting code. This isn't easy to do by hand, so let's use the probabilistic method. We will let $\mathsf{Enc} : \{0,1\}^k \to \{0,1\}^n$ be a randomly chosen function, and let $\mathsf{Dec} : \{0,1\}^n \to \{0,1\}^k$ be the *maximum likelihood decoder*:

$$\mathsf{Dec}(\boldsymbol{x}) \stackrel{\mathsf{def}}{=} \operatorname*{arg\,min}_{\boldsymbol{m} \in \{0,1\}^k} d_H(\boldsymbol{x}, \mathsf{Enc}(\boldsymbol{m})).$$

In the case where multiple messages $\boldsymbol{m}$ attain the minimum, we select one arbitrarily. This decoder is by no means efficient, but it is a natural one to use and can be computed by brute force. Note that since $\mathsf{Enc} : \{0,1\}^k \to \{0,1\}^n$ is a uniformly random function, meaning $\mathsf{Enc}(\boldsymbol{m}) \sim \mathsf{Unif}\{0,1\}^n$ independently over all $\boldsymbol{m} \in \{0,1\}^k$, there is a (small) collision probability we'll have to address in the overall error probability of decoding.

Ultimately, our goal is to guarantee that the success probability is at least $1 - o(1)$ uniformly for all $\boldsymbol{m} \in \{0,1\}^k$. As a stepping stone, we begin by establishing an average-case guarantee.

Proposition 3.1. *Suppose $r < 1 - H(p)$ and $k = rn$. Then there exists $\eta(n)$ decaying to 0 as $n \to \infty$ such that the following holds: For a uniformly random function $\mathsf{Enc} : \{0,1\}^k \to \{0,1\}^n$ paired with the maximum likelihood decoder $\mathsf{Dec} : \{0,1\}^n \to \{0,1\}^k$,*

$$\mathbb{E}\left[\Pr\left[\mathsf{Dec}\left(\mathsf{BSC}_p\left(\mathsf{Enc}(\boldsymbol{m})\right)\right) \neq \boldsymbol{m}\right]\right] \leq \eta(n), \qquad \forall \boldsymbol{m} \in \{0,1\}^k,$$

where the inner probability is with respect to the binary symmetric channel, and the outer expectation is with respect to the random choice of encoder.

Proof. Because we're using maximum likelihood decoding and because our encoder is chosen uniformly at random, the hope is that $\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m}))$ will be close to $\mathsf{Enc}(\boldsymbol{m})$ and far from $\mathsf{Enc}(\boldsymbol{m}')$ for $\boldsymbol{m}' \neq \boldsymbol{m}$. In particular, for a parameter $t > 0$ to be determined later, observe that $\mathsf{Dec}(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m}))) = \boldsymbol{m}$ holds if simultaneously $d_H(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m})), \mathsf{Enc}(\boldsymbol{m})) \leq t$, while $d_H(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m})), \mathsf{Enc}(\boldsymbol{m}')) > t$ for all $\boldsymbol{m}' \neq \boldsymbol{m}$. Taking the contrapositive and applying the Union Bound, we have that

$$\begin{aligned}&\Pr[\mathsf{Dec}(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m}))) \neq \boldsymbol{m}] \\ &\leq \Pr[d_H(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m})), \mathsf{Enc}(\boldsymbol{m})) > t] + \sum_{\boldsymbol{m}' \neq \boldsymbol{m}} \Pr[d_H(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m})), \mathsf{Enc}(\boldsymbol{m}')) \leq t].\end{aligned}$$

Now choose $\delta(n), \epsilon(n) \to 0$ and set $t = (p + \delta(n)) \cdot n$ so that

$$\Pr[d_H(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m})), \mathsf{Enc}(\boldsymbol{m})) > t] \leq \epsilon(n),$$

which is possible by Theorem 1.2 (or its proof). For the remaining terms, fix $\boldsymbol{m}' \neq \boldsymbol{m}$. By writing the probability as an expectation of an indicator random variable, and then exchanging the order of the expectations, we have

$$\mathbb{E}_{\mathsf{Enc}}\left[\Pr_{\mathsf{BSC}_p}[d_H(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m})), \mathsf{Enc}(\boldsymbol{m}')) \leq t]\right] = \mathbb{E}_{\mathsf{BSC}_p}\left[\Pr_{\mathsf{Enc}}[d_H(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m})), \mathsf{Enc}(\boldsymbol{m}')) \leq t]\right].$$

The inner probability in the right-hand side is much easier to estimate, since $\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m}))$ can be treated as some arbitrarily fixed codeword $\boldsymbol{y} \in \{0,1\}^n$, and the randomness is taken over a uniformly random $\boldsymbol{x} = \mathsf{Enc}(\boldsymbol{m}')$. More precisely, it is equal to the probability that a uniformly random bit-string $\mathsf{Enc}(\boldsymbol{m}') \sim \mathsf{Unif}\{0,1\}^n$ lands in the radius-t Hamming ball around the point $\boldsymbol{y} = \mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m}))$. This probability is the same regardless of what $\boldsymbol{y}$ is, and is upper bounded by

$$\frac{1}{2^n}\sum_{k=0}^{(p+\delta(n))\cdot n}\binom{n}{k} \leq 2^{(H(p)-1+\delta(n))\cdot n}. \qquad \text{(Using Lemma 1.3)}$$

Combining these inequalities, we obtain

$$\mathbb{E}[\Pr[\mathsf{Dec}(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m}))) \neq \boldsymbol{m}]] \leq \epsilon(n) + 2^{(H(p)-1+\delta(n))\cdot n} \stackrel{\mathsf{def}}{=} \eta(n) \leq o(1),$$

using that $p < 1/2$ so that $H(p) < 1$ is constant independent of n. □

To complete the proof of Theorem 1.1 in the case $r < 1 - H(p)$, observe that by averaging Proposition 3.1 over $\boldsymbol{m} \in \{0,1\}^k$, we have

$$\mathbb{E}_{\mathsf{Enc}}\left[\mathbb{E}_{\boldsymbol{m}\sim\mathsf{Unif}\{0,1\}^k}[\Pr[\mathsf{Dec}(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m}))) \neq \boldsymbol{m}]]\right] \leq \eta(n).$$

Hence, there exists an encoder $\mathsf{Enc} : \{0,1\}^k \to \{0,1\}^n$ possessing the average-case guarantee

$$\mathbb{E}_{\boldsymbol{m}\sim\mathsf{Unif}\{0,1\}^k}[\Pr[\mathsf{Dec}(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m}))) \neq \boldsymbol{m}]] \leq \eta(n).$$

Recall that ultimately, we want this bound for hold for all $\boldsymbol{m} \in \{0,1\}^k$, not in expectation.

Here's the final idea. Observe that the above bound on the expectation implies that there exists a subset of "good messages" $\mathcal{G} \subseteq \{0,1\}^k$ with $|\mathcal{G}| = 2^{k-1}$ such that for all $\boldsymbol{m} \in \mathcal{G}$, we have

$$\Pr[\mathsf{Dec}(\mathsf{BSC}_p(\mathsf{Enc}(\boldsymbol{m}))) \neq \boldsymbol{m}] \leq 2 \cdot \eta(n).$$

Indeed, if more than half of the elements of $\{0,1\}^k$ have failure probability exceeding $2 \cdot \eta(n)$, then we would obtain a contradiction with the aforementioned bound on the expectation. Hence, we have found an encoder with the desired properties for a set of messages $\mathcal{G} \subseteq \{0,1\}^k$ of size 2^{k-1}. By mapping $\mathcal{G}$ bijectively to $\{0,1\}^{k-1}$, it follows we have constructed a $[n, k-1]$-code with worst-case failure probability $2 \cdot \eta(n) \leq o(1)$. Replacing k with $k-1$, which incurs $o(1)$ in the rate r, then completes the proof.