Teorema da Codificação da Fonte

O problema das 12 bolas

- Você recebe 12 bolas idênticas, exceto por uma delas que pode ser ligeiramente mais pesada ou mais leve que as demais. Você também recebe uma balança com dois pratos para usar. Em cada uso da balança, você pode colocar qualquer quantidade de bolas no prato da esquerda e o mesmo número de bolas no prato da direita e apertar um botão para que seja feita a medida. Há três resultados possíveis: os pesos são iguais; o prato da direita é mais pesado; o prato da esquerda é mais pesado.

    - Sua tarefa é bolar uma estratégia para determinar qual é a bola diferente E se ela é mais pesada ou mais leve que as demais utilizando a balança pelo menor número possível de vezes.

O problema das 12 bolas

- Dicas:
  - Como otimizar o processo de medidas?
    - O que, de fato, estamos tentando maximizar no processo para reduzir o número de usos da balança?
  - É possível **medir** informação?
    - **Quanto** de informação você ganha ao descobrir a bola diferente?
    - Qual é o máximo de informação que você pode obter em 1 uso da balança?
  - Usando a balança 3 vezes temos $3^3 = 27$ resultados possíveis. Há 24 estados possíveis espaço amostral (12 bolas x 2 possibilidades para cada).

Jogos de adivinhação

- 63:
  - Qual o menor número de perguntas com respostas binárias (sim ou não) para se determinar um número inteiro entre 0 e 63?

Jogos de adivinhação

- Submarino:
  - Batalha naval com um único submarino.
    - Quanto de informação você obteve ao final do jogo?
    - Seu resultado depende de quando você chegou à resposta?

Jogos de adivinhação

- Uma lingua inventada (Wenglish):
  - Dicionário composto por $2^{15} = 32.768 << 26^5 \sim 10^7$ palavras de 5 caracteres formadas aleatoriamente de acordo com as frequências de ocorrência das letras na língua Inglesa
  - Para um texto formado por palavras escolhidas aleatoriamente no dicionário, h(x)=log(32768)=15 bits por palavra, o que dá uma média de 3 bits por caracter
  - Analisando caracter por caracter, como a probabilidade da letra a é ~0,0625 e a da letra z é ~0,001, ao verificar que a primeira letra de uma palavra é a, ganhamos $\log(1/0{,}0625) \approx 4$ bits. Se a primeira letra for z, ganhamos $\log(1/0{,}001) \approx 10$ bits. O total, no entanto, continua sendo 15 bits

1	aaail
2	aaaiu
3	aaald
⋮	⋮
129	abati
⋮	⋮
2047	azpan
2048	aztdn
⋮	⋮
⋮	⋮
16 384	odrcr
⋮	⋮
⋮	⋮
32 737	zatnt
⋮	⋮
32 768	zxast

Figure 4.4. The Wenglish dictionary.

Compressão de Dados

- Quantos bits são necessários para codificar uma mensagem?
  - Texto em ASCII - 127 símbolos codificados em 1 byte = 8 bits
    - Podemos comprimir por um fator de 7/8 - já conhecemos 1 bit
    - Funciona quando há redundância na codificação
    - Maiores compressões podem ser obtidas considerando que:
      - Letras aparecem com frequências diferentes
      - Alguns pares de letras são mais prováveis que outros
      - Palavras podem ser preditas de acordo com o contexto

Compressão de Dados

- Conteúdo bruto de X:

$$H_0(X) = \log(|A_X|)$$

- Limite inferior para o número de questões binárias para determinar uma saída do ensemble X
- $H_0(X, Y) = H_0(X) + H_0(Y)$
- Não há compressão nesse processo. Apenas mapeamos as saídas em strings binários de comprimento fixo.
- Pode haver um compressor que mapeia cada saída x em um código binário, $c(x)$, e um tradutor que mapeia c de volta em x, tal que toda saída possível é comprimida em um código binário de comprimento menor que $H_0(X)$ bits?

Compressão de Dados

- Compressão com perdas (Lossy compression):
  - Mapeia algumas saídas no mesmo código
  - Probabilidade δ de errar
  - Se δ for pequeno o suficiente, a compressão com perdas pode ser útil
- Compressão sem perdas (Lossless compression):
  - Se for capaz de comprimir uma entrada, necessariamente tornará outras maiores
  - Desenvolvido de forma a tornar a probabilidade de aumentar o tamanho ser muito pequena

Quantidade de informação em termos da compressão com perdas

- Assumimos que pode haver erros com probabilidade δ
- Exemplo simples: Simplesmente ignoramos símbolos cuja probabilidade de ocorrência é muito pequena
    - Exemplo: Excluir os caracteres !, #, %, ^, ~, *, <, >, _, {, }, [,], |, /, \ do código ASCII
- Formalizando: Temos que escolher um conjunto S_δ tal que $P(x \notin S_\delta) \leq \delta$
- O menor subconjunto δ-suficiente, S_δ, é o menor subconjunto no alfabeto $\mathcal{A}_X$ que satisfaz

$$P(x \in S_\delta) \geq 1 - \delta$$

Quantidade de informação em termos da compressão com perdas

- Conteúdo essencial de bits:

$$H_\delta(X) = \log(|S_\delta|)$$

- Podemos fazer melhor se considerarmos blocos de símbolos ao invés dos símbolos individualmente?

- Saída: $\mathbf{x} = (x_1, x_2, \cdots, x_N)$ - conjunto de N variáveis aleatórias independentes e identicamente distribuídas

- Ensemble: X^N. Como a entropia é aditiva para variáveis independentes: $H(X^N) = NH(X)$

Quantidade de informação em termos da compressão com perdas

- Moeda viciada jogada N vezes $p_0 = 0.9, p_1 = 0.1$. Seja $r(\mathbf{x})$ o número de 1's.

$$P(\mathbf{x}) = p_0^{N-r(\mathbf{x})} p_1^{r(\mathbf{x})}$$

- Encontramos o menor subconjunto suficiente ordenando as saídas possíveis em ordem crescente até um dado r_{max}.

Quantidade de informação em termos da compressão com perdas

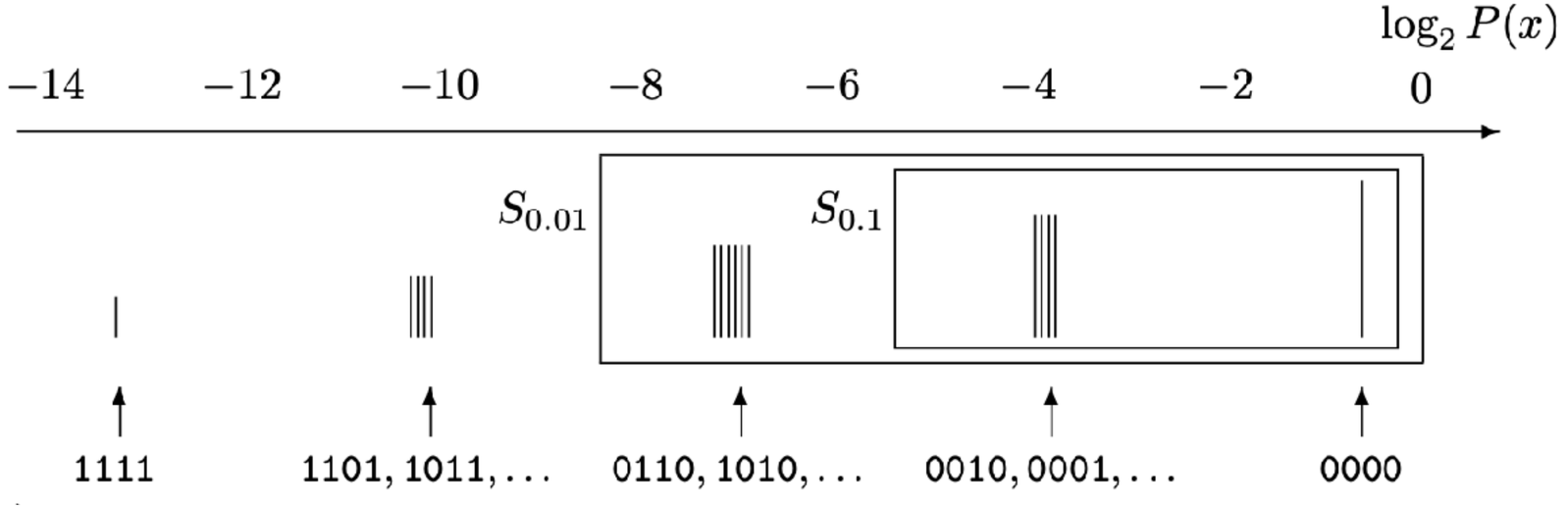
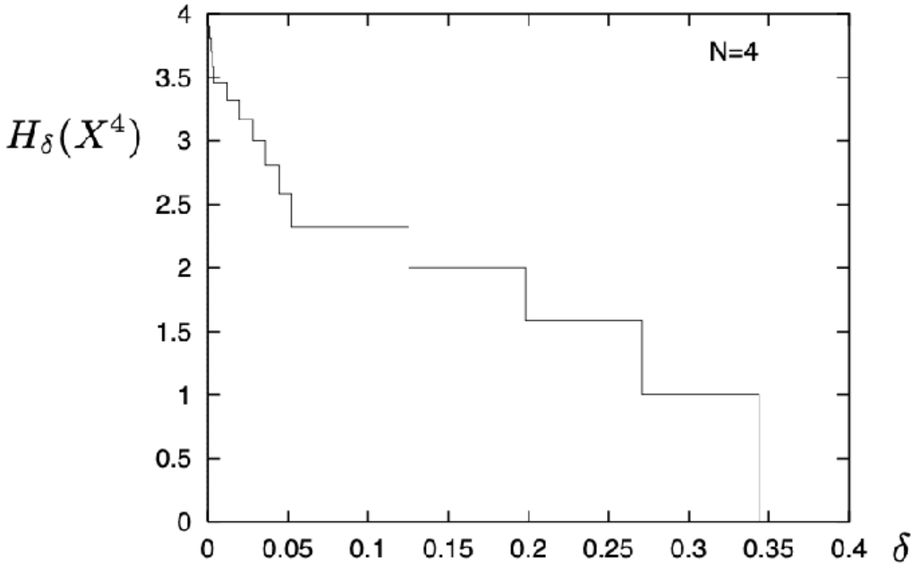


Figure 4.7. (a) The sixteen outcomes of the ensemble X^4 with $p_1 = 0.1$, ranked by probability. (b) The essential bit content $H_\delta(X^4)$. The upper schematic diagram indicates the strings' probabilities by the vertical lines' lengths (not to scale).



Quantidade de informação em termos da compressão com perdas

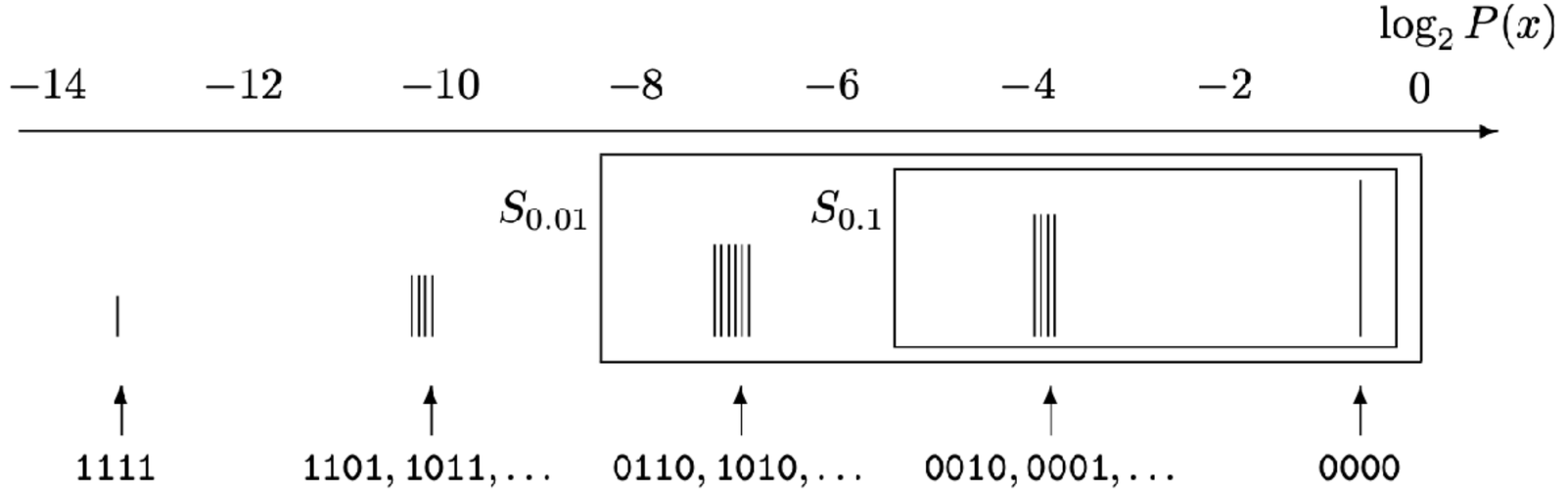
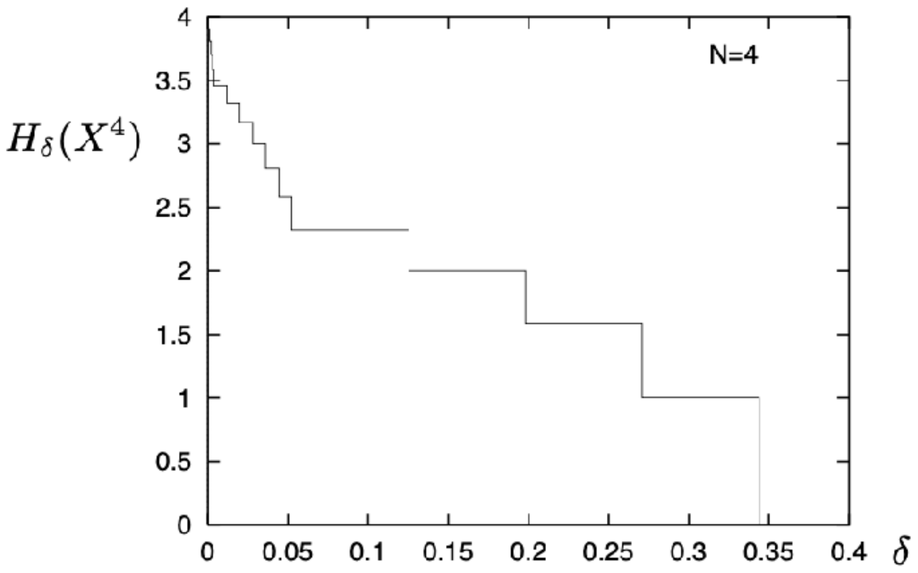
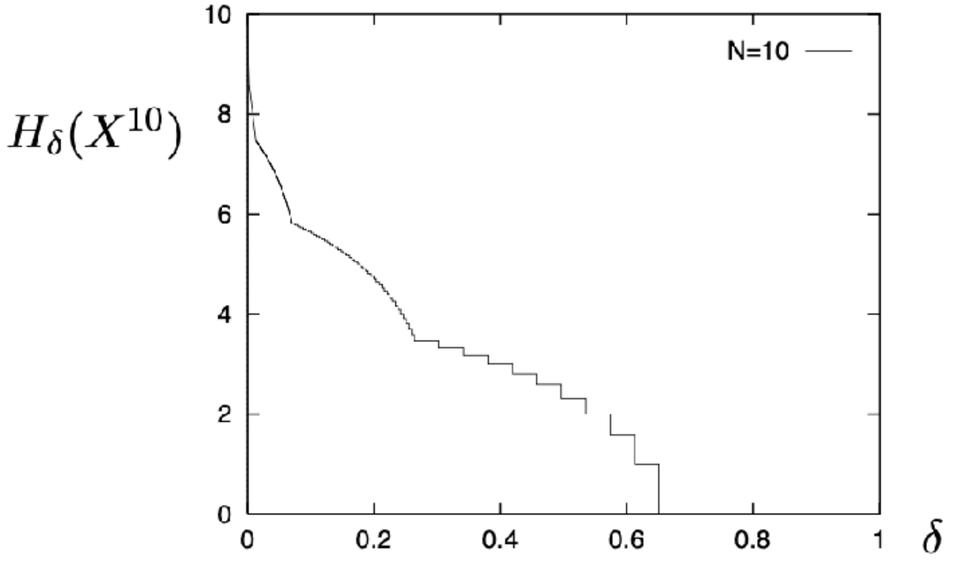


Figure 4.7. (a) The sixteen outcomes of the ensemble X^4 with $p_1 = 0.1$, ranked by probability. (b) The essential bit content $H_\delta(X^4)$. The upper schematic diagram indicates the strings' probabilities by the vertical lines' lengths (not to scale).





Quantidade de informação em termos da compressão com perdas

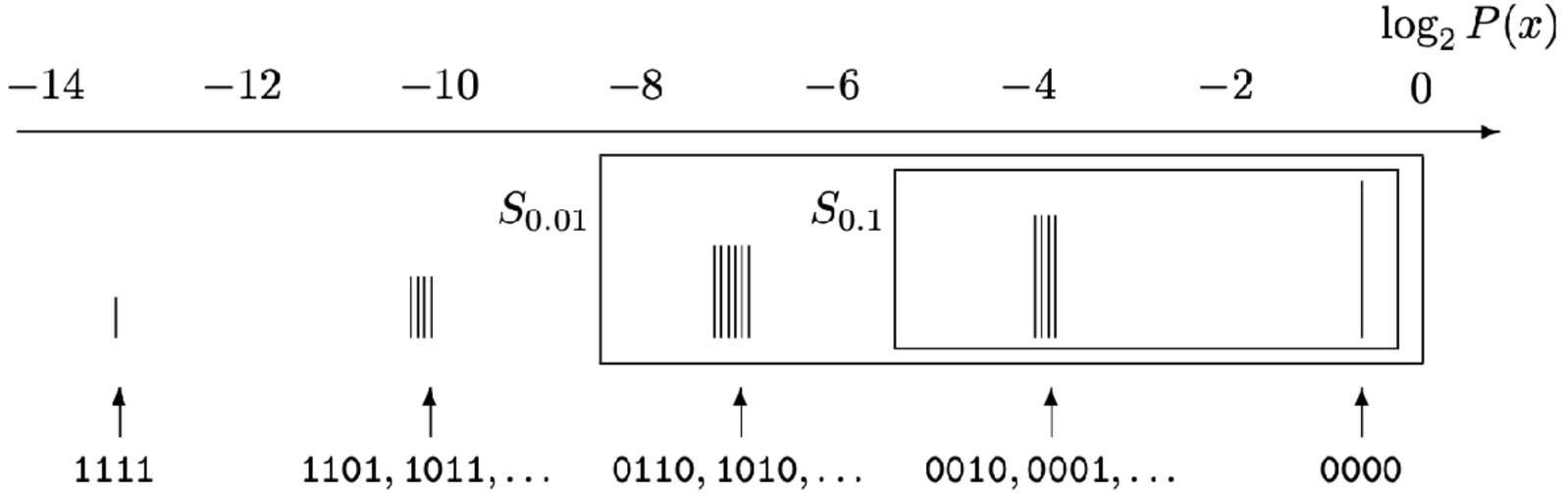
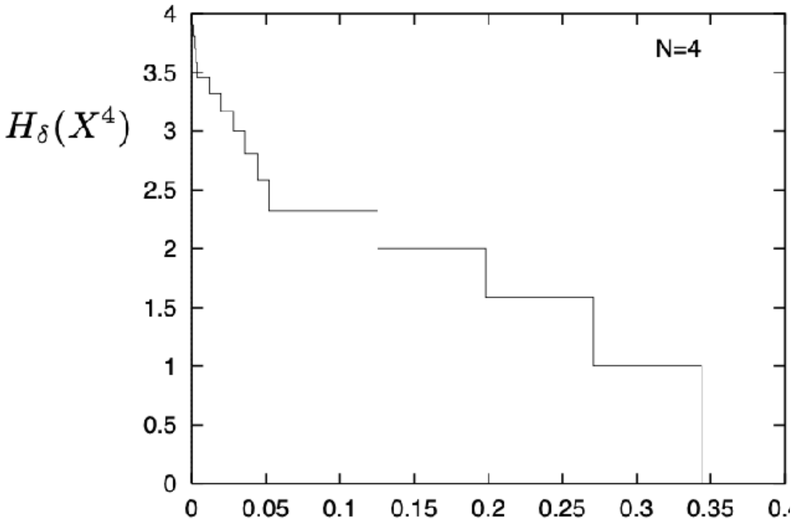
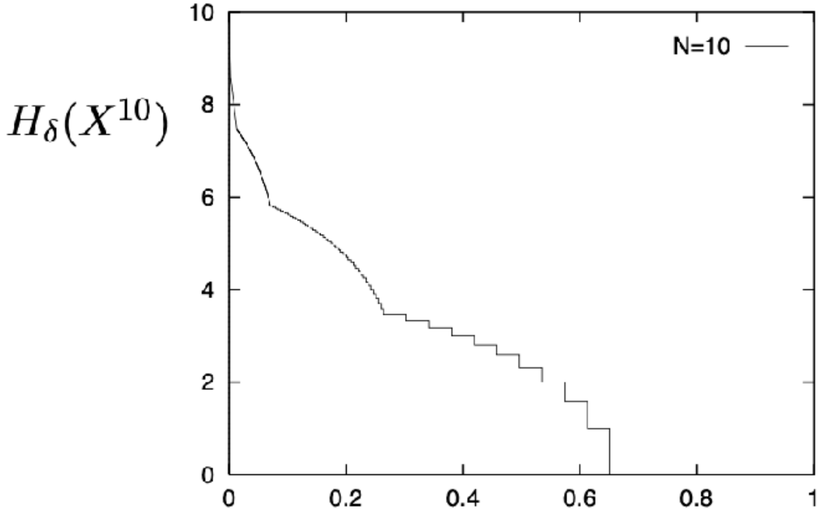
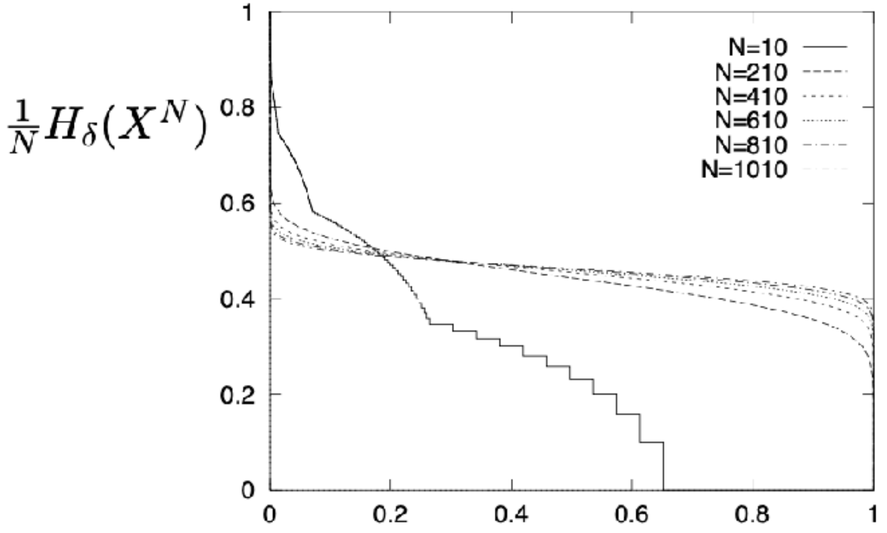


Figure 4.7. (a) The sixteen outcomes of the ensemble X^4 with $p_1 = 0.1$, ranked by probability. (b) The essential bit content $H_\delta(X^4)$. The upper schematic diagram indicates the strings' probabilities by the vertical lines' lengths (not to scale).







Teorema da Codificação da Fonte

- Seja X um ensemble com $H(X) = H$ bits. Dado $\epsilon > 0$ e $0 < \delta < 1$, existe um inteiro positivo N_0 tal que para $N > N_0$,

$$\left| \frac{1}{N} H_\delta(X^N) - H \right| < \epsilon$$

Teorema da Codificação da Fonte

- N variáveis aleatórias independentes e identicamente distribuídas cada uma com entropia $H(X)$ podem ser compactadas em mais de $NH(X)$ bits com risco insignificante de perda de informação quando $N \to \infty$. Inversamente, se forem compactadas em menos que $NH(X)$ bits é praticamente certo que haverá perda de informação.