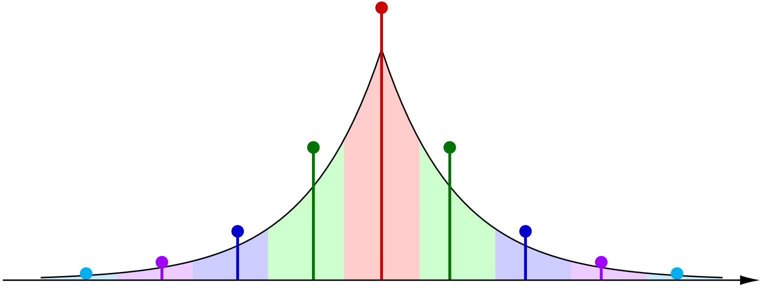
Scalar Quantization



Last Lectures: Lossless Coding

Variable-length Coding

- Scalar codes, conditional codes, block codes, V2V codes (using codeword tables)
- For given pmf: **Huffman algorithm** yields optimal codeword table
- Problem: Codeword tables become too large for practical application of block codes

Arithmetic Coding

- No codeword table: On-the-fly encoding and decoding
- Sub-optimal block code for arbitrarily large block sizes N (very close to optimum for $N \gg 1$)
- Straightforward combination with **conditional and adaptive probability models**

Reduction of Inter-Symbols Dependencies before Entropy Coding

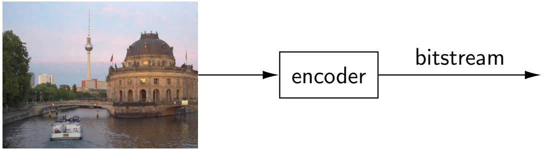
- **Affine and linear prediction**: Suitable for reducing dependencies in audio, image, video data
- **Lempel-Ziv coding** or **block sorting**: Suitable for text, source code, general files

→ **Lossless coding in practice**:
- Prediction followed by entropy coding of prediction errors
- Lempel-Ziv coding or block sorting followed by entropy coding

Lossy Coding

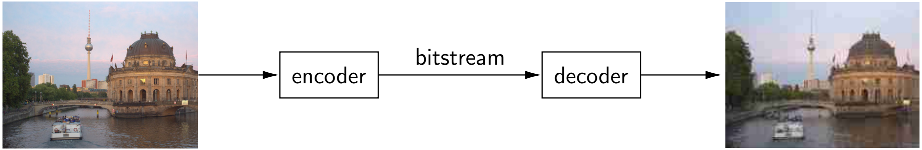
original $\boldsymbol{s}$

Lossy Coding



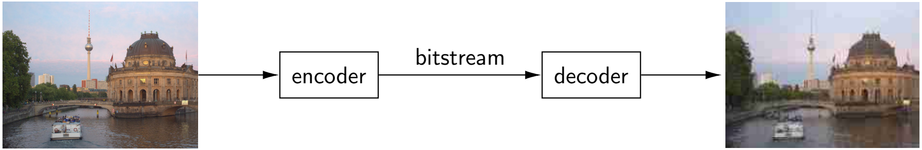
original $\boldsymbol{s}$

Lossy Coding



original $\boldsymbol{s}$ reconstructed $\boldsymbol{s}' \neq \boldsymbol{s}$

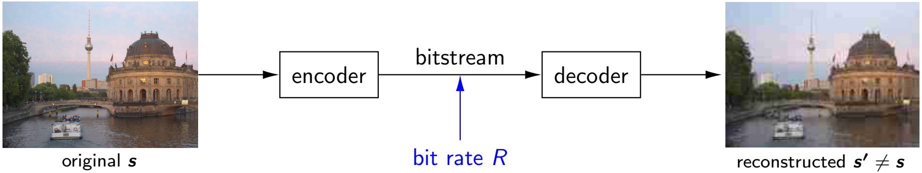
Lossy Coding



original s reconstructed $s' \neq s$

Lossy coding is characterized by two aspects:

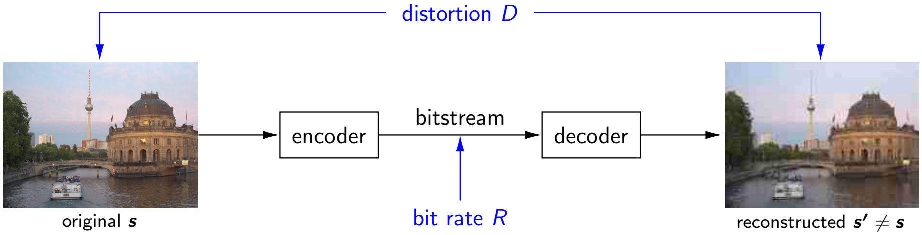
Lossy Coding



Lossy coding is characterized by two aspects:

- Bit rate R: Average number of bits per sample (or per time unit)

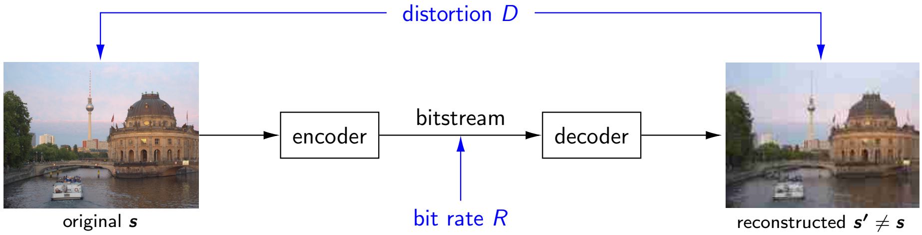
Lossy Coding



Lossy coding is characterized by two aspects:

- Bit rate R: Average number of bits per sample (or per time unit)
- Distortion D: Measure for deviation between original signal $\boldsymbol{s}$ and reconstructed signal $\boldsymbol{s}'$

Lossy Coding

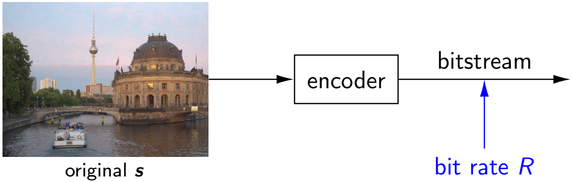


Lossy coding is characterized by two aspects:

- Bit rate R: Average number of bits per sample (or per time unit)
- Distortion D: Measure for deviation between original signal $\boldsymbol{s}$ and reconstructed signal $\boldsymbol{s}'$

Design Goal: Smallest possible bit rate for given maximum distortion, or
Smallest possible distortion for given maximum bit rate

Lossy Coding: Bit Rate



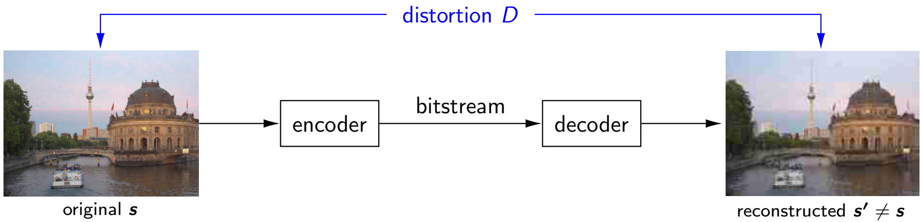
Bit Rate R:

- Images: Average number of bits per sample
- Audio or video: Average number of bits per time units

Often used Approximation:

- Assume that we have a close to optimal entropy coding (e.g., arithmetic coding)
- Bit rate = Entropy of symbols that are actually transmitted

Lossy Coding: Distortion



Distortion Measures used in Practice

- General p-norm distortion:

$$D_p = \frac{1}{N} \sum_{k=1}^{N} \left| s_k - s'_k \right|^p \qquad \text{or} \qquad D_p = \mathrm{E}\left\{ \left| S - S' \right|^p \right\}$$

- Most often: Mean squared error (MSE)

$$D_2 = \frac{1}{N} \sum_{k=1}^{N} \left(s_k - s'_k \right)^2 \qquad \text{or} \qquad D_2 = \mathrm{E}\left\{ \left(S - S' \right)^2 \right\}$$

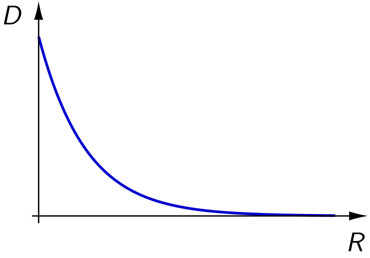
MSE Distortion as Signal-To-Noise Ratio (SNR)

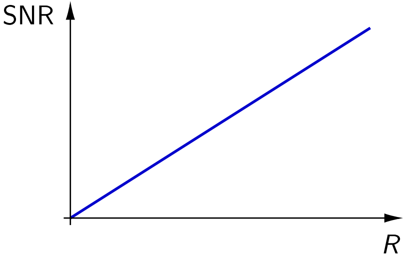
Signal-to-Noise Ratio (SNR)

- Logarithmic ratio of variance and MSE distortion

$$\mathrm{SNR} = 10 \cdot \log_{10}\left(\frac{\sigma^2}{D_2}\right)$$

- Measured in decibel (dB)





MSE Distortion as Signal-To-Noise Ratio (SNR)

Signal-to-Noise Ratio (SNR)

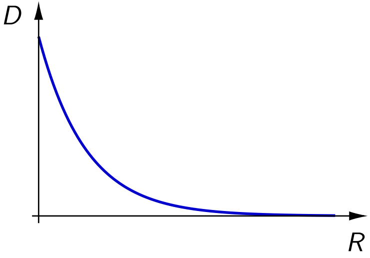
- Logarithmic ratio of variance and MSE distortion

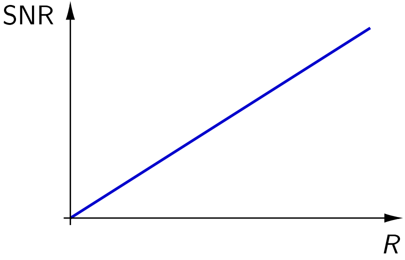
$$\text{SNR} = 10 \cdot \log_{10}\left(\frac{\sigma^2}{D_2}\right)$$

- Measured in decibel (dB)

Advantages of using SNR

- Independent of signal variance





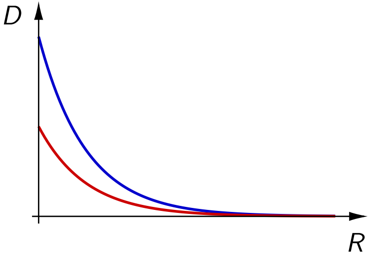
MSE Distortion as Signal-To-Noise Ratio (SNR)

Signal-to-Noise Ratio (SNR)

- Logarithmic ratio of variance and MSE distortion

$$\text{SNR} = 10 \cdot \log_{10}\left(\frac{\sigma^2}{D_2}\right)$$

- Measured in decibel (dB)



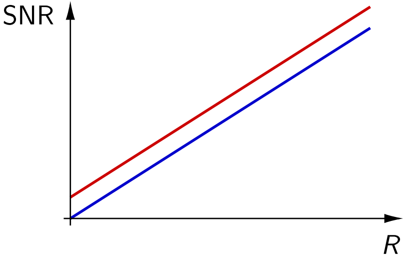
Advantages of using SNR

- Independent of signal variance
- Easy interpretation of differences

$$\Delta\text{SNR} = \text{SNR}_a - \text{SNR}_b = -10 \cdot \log_{10} \frac{D_a}{D_b}$$

- Examples:

$$\begin{aligned}
D_a &= D_b & &\rightarrow & \Delta\text{SNR} &\approx 0.0\,\text{dB} \\
D_a &= D_b/\sqrt{2} & &\rightarrow & \Delta\text{SNR} &\approx 1.5\,\text{dB} \\
D_a &= D_b/2 & &\rightarrow & \Delta\text{SNR} &\approx 3.0\,\text{dB} \\
D_a &= D_b/4 & &\rightarrow & \Delta\text{SNR} &\approx 6.0\,\text{dB} \\
D_a &= D_b/8 & &\rightarrow & \Delta\text{SNR} &\approx 9.0\,\text{dB}
\end{aligned}$$



Probabilistic Modeling of Sources

Source Coding in Practice

- Encoder and decoder are computer programs

➞ Actual input signals are **discrete-time** and **discrete-amplitude** signals

Real-world signals

- In most cases: Continuous-time and continuous-amplitude signals
- Discrete signals are obtained by sampling and quantization
- Typical scenarios: Initial quantization has negligible effect on source coding

Probabilistic Modeling of Sources

Source Coding in Practice

- Encoder and decoder are computer programs

➡ Actual input signals are **discrete-time** and **discrete-amplitude** signals

Real-world signals

- In most cases: Continuous-time and continuous-amplitude signals
- Discrete signals are obtained by sampling and quantization
- Typical scenarios: Initial quantization has negligible effect on source coding

Theoretical Analysis of Lossy Source Coding

- Will mostly use models for **discrete-time** and **continuous-amplitude** signals
- Main reason: Mathematical tractability

➡ Interpretation: Consider signal before initial quantization

Review: Random Variables and Cumulative Distribution Function (CDF)

Random Variable

- Function $X(\zeta)$ of the sample space $\mathcal{O}$ that assigns a real value $x = X(\zeta)$ to each possible outcome $\zeta \in \mathcal{O}$ of a random experiment

Cumulative Distribution Function (cdf)

- Cumulative distribution function $F_X(x)$ of a random variable X

$$F_X(s) = \mathrm{P}(X \leq x) = \mathrm{P}(\,\{\zeta : X(\zeta) \leq x\}\,)$$

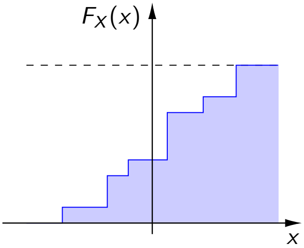
- Joint cdf of two random variables X and Y

$$F_{XY}(x, y) = \mathrm{P}(X \leq x, Y \leq y)$$

- Conditional cdf of a random variable X given another random variable Y

$$F_{X|Y}(x \,|\, y) = \mathrm{P}(X \leq x \,|\, Y \leq y) = \frac{\mathrm{P}(X \leq x, Y \leq y)}{\mathrm{P}(Y \leq y)} = \frac{F_{XY}(x, y)}{F_Y(y)}$$

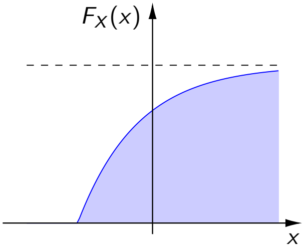
Review: Examples of Cumulative Distribution Functions



Staircase function

- Random variable X can only take a countable number of values

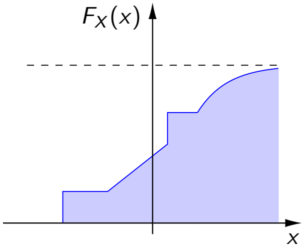
➜ **Discrete random variable**



Continuous function

- Random variable X can take all values inside one or more non-zero intervals

➜ **Continuous random variable**



Mixed type

- Random variable X can take all values inside one or more non-zero intervals and a countable number of additional values

Continuous Random Variables and Probability Density Function (PDF)

Continuous Random Variables

- A random variable X is called a **continuous random variable** if and only if its cdf $F_X(x)$ is a continuous function

Probability Density Function

- Probability density function (pdf) of a continuous random variable S

$$f_X(x) = \frac{\partial}{\partial x} F_X(x) \quad \iff \quad F_X(x) = \int_{-\infty}^{x} f_X(t)\,\mathrm{d}t$$

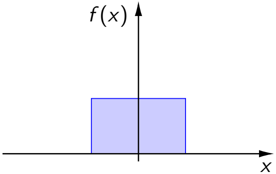
➡ Properties:

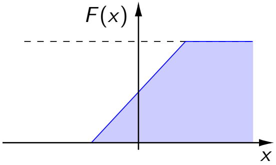
- $f_X(x) \geq 0, \ \forall x$
- $\int_{-\infty}^{\infty} f_X(t)\,\mathrm{d}t = 1$
- $\mathrm{P}(a < X \leq b) = \int_a^b f_X(t)\,\mathrm{d}t$

Examples for Continuous Distributions (Zero Mean)

Uniform

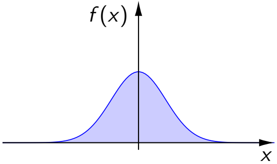
$$f(x) = \begin{cases} \frac{1}{2a} & : \ |x| \leq a \\ 0 & : \ \text{otherwise} \end{cases}$$

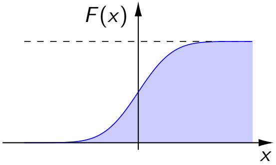




Gaussian

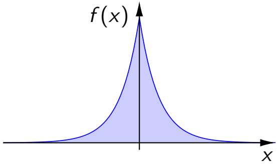
$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \, e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

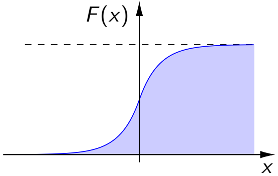




Laplacian

$$f(x) = \frac{1}{\sqrt{2\sigma^2}} \, e^{-\sqrt{\frac{2}{\sigma^2}}\,|x-\mu|}$$





Generalized Gaussian Distribution

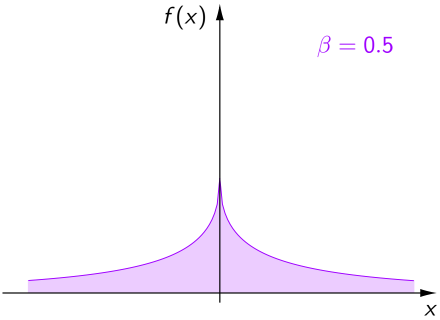
Shape parameter $\beta \in (0, \infty)$:

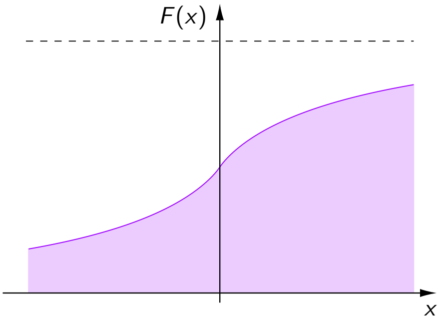
$$f(x) = \frac{\beta}{2\,\alpha\,\Gamma(1/\beta)}\, e^{-\left(\frac{|x-\mu|}{\alpha}\right)^{\beta}} \qquad \text{with} \qquad \Gamma(x) = \int_0^\infty e^{-t}\, t^{x-1}\,\mathrm{d}t$$

Generalized Gaussian Distribution

Shape parameter $\beta \in (0, \infty)$:

$$f(x) = \frac{\beta}{2\,\alpha\,\Gamma(1/\beta)}\, e^{-\left(\frac{|x-\mu|}{\alpha}\right)^{\beta}} \qquad \text{with} \qquad \Gamma(x) = \int_0^{\infty} e^{-t}\, t^{x-1}\, \mathrm{d}t$$

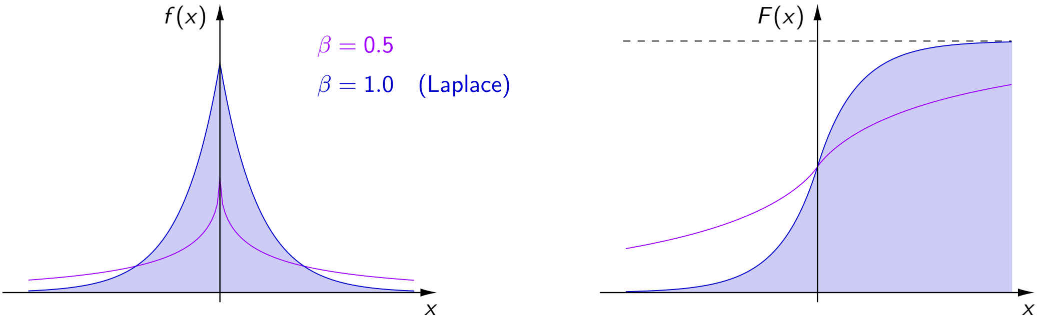




Generalized Gaussian Distribution

Shape parameter $\beta \in (0, \infty)$:

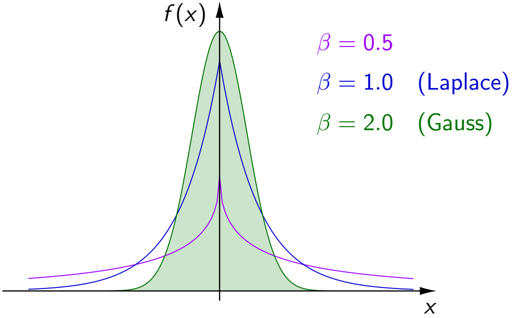
$$f(x) = \frac{\beta}{2\,\alpha\,\Gamma(1/\beta)}\, e^{-\left(\frac{|x-\mu|}{\alpha}\right)^{\beta}} \qquad \text{with} \qquad \Gamma(x) = \int_0^\infty e^{-t}\, t^{x-1}\, \mathrm{d}t$$

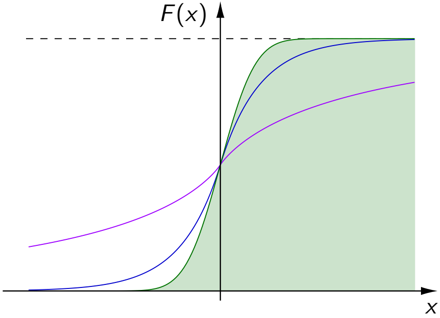


Generalized Gaussian Distribution

Shape parameter $\beta \in (0, \infty)$:

$$f(x) = \frac{\beta}{2\,\alpha\,\Gamma(1/\beta)}\, e^{-\left(\frac{|x-\mu|}{\alpha}\right)^{\beta}} \qquad \text{with} \qquad \Gamma(x) = \int_0^{\infty} e^{-t}\, t^{x-1}\, \mathrm{d}t$$

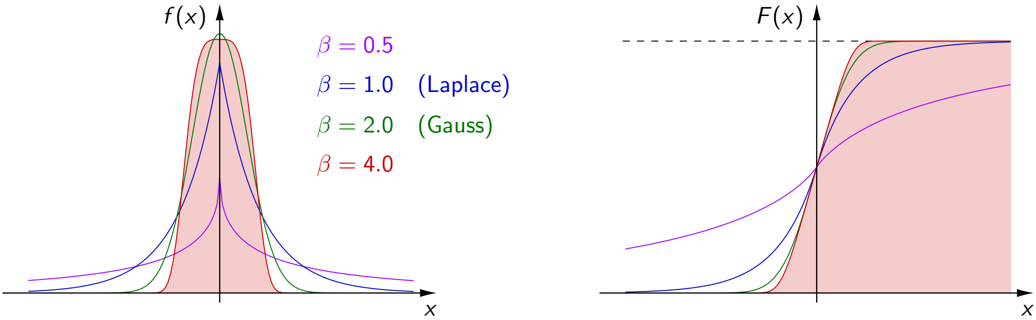




Generalized Gaussian Distribution

Shape parameter $\beta \in (0, \infty)$:

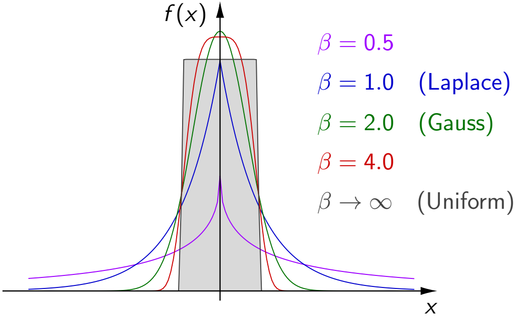
$$f(x) = \frac{\beta}{2\,\alpha\,\Gamma(1/\beta)}\, e^{-\left(\frac{|x-\mu|}{\alpha}\right)^{\beta}} \qquad \text{with} \qquad \Gamma(x) = \int_0^{\infty} e^{-t}\, t^{x-1}\, \mathrm{d}t$$

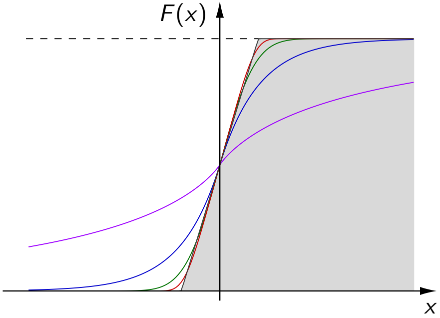


Generalized Gaussian Distribution

Shape parameter $\beta \in (0, \infty)$:

$$f(x) = \frac{\beta}{2\,\alpha\,\Gamma(1/\beta)}\, e^{-\left(\frac{|x-\mu|}{\alpha}\right)^{\beta}} \qquad \text{with} \qquad \Gamma(x) = \int_0^\infty e^{-t}\, t^{x-1}\, \mathrm{d}t$$

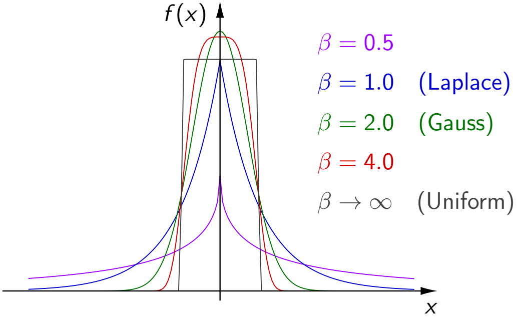


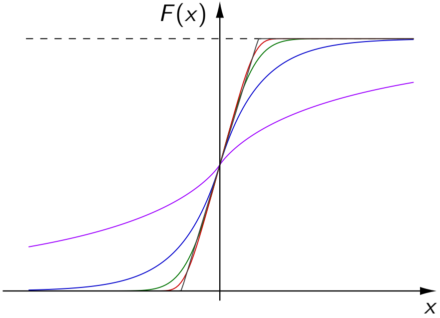


Generalized Gaussian Distribution

Shape parameter $\beta \in (0, \infty)$:

$$f(x) = \frac{\beta}{2\,\alpha\,\Gamma(1/\beta)}\, e^{-\left(\frac{|x-\mu|}{\alpha}\right)^{\beta}} \qquad \text{with} \qquad \Gamma(x) = \int_0^{\infty} e^{-t}\, t^{x-1}\,\mathrm{d}t$$





➡ **Suitable approximation for many distributions**

Joint and Conditional Probability Density Function

Joint Probability Density Function

- Joint pdf of two random variables X and Y

$$f_{XY}(x, y) = \frac{\partial^2}{\partial x \, \partial y} F_{XY}(x, y)$$

Joint and Conditional Probability Density Function

Joint Probability Density Function

- Joint pdf of two random variables X and Y

$$f_{XY}(x,y) = \frac{\partial^2}{\partial x \, \partial y} F_{XY}(x,y)$$

Conditional Probability Density Function

- Conditional pdf of a random variable X given another random variable Y

$$f_{X|Y}(x\,|\,y) = \frac{\partial}{\partial x} F_{X|Y}(x\,|\,y) = \frac{\partial}{\partial x} \, \frac{F_{XY}(x,y)}{F_Y(y)} = \frac{\frac{\partial^2}{\partial x \, \partial y} F_{XY}(x,y)}{\frac{\partial}{\partial y} F_Y(y)} = \frac{f_{XY}(x,y)}{f_Y(y)}$$

Expected Values for Continuous Random Variables

Expected Values

- Expected value of a function $g(X)$ of a continuous random variable X

$$\mathrm{E}\{\, g(X) \,\} = \mathrm{E}_X\{\, g(X) \,\} = \int_{-\infty}^{\infty} g(x)\, f_X(x)\, \mathrm{d}x$$

- Expected value of function $g(X, Y)$ of two continuous random variables X and Y

$$\mathrm{E}\{\, g(X,Y) \,\} = \mathrm{E}_{XY}\{\, g(X,Y) \,\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y)\, f_{XY}(x,y)\, \mathrm{d}x\, \mathrm{d}y$$

Conditional Expected Values

- Expected value of function $g(X)$ of a random variable X given another random variable Y

$$\mathrm{E}\{\, g(X) \,|\, Y \,\} = \int_{-\infty}^{\infty} g(x)\, f_{X|Y}(x \,|\, Y)\, \mathrm{d}x \qquad \text{(is another random variable)}$$

Properties of Expected Values

same properties as in discrete case

Important Properties

- Linearity of expected values

$$\mathrm{E}\{\, aX + bY \,\} = a \cdot \mathrm{E}\{\, X \,\} + b \cdot \mathrm{E}\{\, Y \,\}$$

- For independent random variables X and Y

$$\mathrm{E}\{\, XY \,\} = \mathrm{E}\{\, X \,\}\, \mathrm{E}\{\, Y \,\}$$

- Iterative expectation rule

$$\mathrm{E}\{\, \mathrm{E}\{\, g(X) \,|\, Y \,\} \,\} = \mathrm{E}\{\, g(X) \,\}$$

Important Expected Values

- **Mean** μ_X of a random variable X

$$\mu_X = \mathrm{E}\{\, X \,\} = \int_{-\infty}^{\infty} x \cdot f_X(x)\, \mathrm{d}x$$

- **Variance** σ_X^2 of a random variable X

$$\sigma_X^2 = \mathrm{E}\Big\{\, (X - \mathrm{E}\{\, X \,\})^2 \,\Big\} = \int_{-\infty}^{\infty} (x - \mu_X)^2 \cdot f_X(x)\, \mathrm{d}x$$

- **Covariance** σ_{XY}^2 of two random variables X and Y, and **correlation coefficient** ϕ_{XY}

$$\sigma_{XY}^2 = \mathrm{E}\Big\{\, (X - \mathrm{E}\{\, X \,\})\,(Y - \mathrm{E}\{\, Y \,\}) \,\Big\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (x - \mu_x)(y - \mu_y) \cdot f_{XY}(x, y)\, \mathrm{d}x\, \mathrm{d}y$$

$$\phi_{XY} = \frac{\sigma_{XY}^2}{\sqrt{\sigma_X^2 \cdot \sigma_Y^2}}$$

Continuous Random Processes

Discrete-Time Random Process

- Series of random experiments at time instants t_n, with $n = 0, 1, 2, \cdots$
- For each experiment: Random variable $X_n = X(t_n)$
- **Random process**: Series of random variables

$$\boldsymbol{X} = \{X_0, X_1, X_2, \cdots\} = \{X_n\}$$

Discrete-Time Continuous-Amplitude Random Process

- Random variables X_n are continuous random variables

➡ Type of random processes we consider for analyzing lossy coding

Statistical Properties of Continuous Random Processes

Characterization of Statistical Properties

- Consider N-dimensional random vector

$$\boldsymbol{X}_k^{(N)} = \{X_k, X_{k+1}, \cdots, X_{k+N-1}\}$$

- N-th order joint cdf

$$F_k^{(N)}(\boldsymbol{x}) = \mathrm{P}\left(\boldsymbol{X}_k^{(N)} \le \boldsymbol{x}\right) = \mathrm{P}(X_k \le x_0, X_{k+1} \le x_1, \cdots, X_{k+N-1} \le x_{N-1})$$

- N-th order joint pdf

$$f_k^{(N)}(\boldsymbol{x}) = \frac{\partial^N}{\partial x_0 \cdots \partial x_{N-1}} F_k^{(N)}(\boldsymbol{x})$$

- Also: Conditional cdfs and conditional pdfs

Models for Random Processes

Stationary Random Processes

- Statistical properties are invariant to a shift in time
- In this course: Typically restrict our considerations to stationary processes

Memoryless Random Processes

- All random variables X_n are independent of each other

Independent and Identically Distributed (IID) Random Processes

- Random processes that are **stationary** and **memoryless**

Markov Processes

- Markov property: Future outcomes do only depend on present outcome, but not on past outcomes

$$F(x_n \,|\, x_{n-1}, x_{n-2}, x_{n-3}, \cdots) = F(x_n \,|\, x_{n-1})$$
$$f(x_n \,|\, x_{n-1}, x_{n-2}, x_{n-3}, \cdots) = f(x_n \,|\, x_{n-1})$$

- Simple model for random processes with memory

Autoregressive (AR) Processes

General AR(p) Model

- Autoregressive model of order p for random variables X_n with mean μ

$$X_n = Z_n + \mu + \sum_{k=1}^{p} \varrho_k \cdot (X_{n-k} - \mu)$$

where $\boldsymbol{Z} = \{Z_n\}$ is a zero-mean iid process (innovation process)
and $\varrho_1, \cdots, \varrho_p$ are the model parameters

Autoregressive (AR) Processes

General AR(p) Model

- Autoregressive model of order p for random variables X_n with mean μ

$$X_n = Z_n + \mu + \sum_{k=1}^{p} \varrho_k \cdot (X_{n-k} - \mu)$$

where $\boldsymbol{Z} = \{Z_n\}$ is a zero-mean iid process (innovation process) and $\varrho_1, \cdots, \varrho_p$ are the model parameters

Special case: AR(1) model

- Autoregressive model of order $p = 1$

$$X_n = Z_n + \mu + \varrho \cdot (X_{n-1} - \mu)$$

Autoregressive (AR) Processes

General AR(p) Model

- Autoregressive model of order p for random variables X_n with mean μ

$$X_n = Z_n + \mu + \sum_{k=1}^{p} \varrho_k \cdot (X_{n-k} - \mu)$$

where $\boldsymbol{Z} = \{Z_n\}$ is a zero-mean iid process (innovation process)
and $\varrho_1, \cdots, \varrho_p$ are the model parameters

Special case: AR(1) model

- Autoregressive model of order $p = 1$

$$X_n = Z_n + \mu + \varrho \cdot (X_{n-1} - \mu)$$

➡ Completely specified by mean μ, correlation coefficient ϱ, and pdf $f_Z(z)$ of iid process $\{Z_n\}$

Autoregressive (AR) Processes

General AR(p) Model

- Autoregressive model of order p for random variables X_n with mean μ

$$X_n = Z_n + \mu + \sum_{k=1}^{p} \varrho_k \cdot (X_{n-k} - \mu)$$

where $\boldsymbol{Z} = \{Z_n\}$ is a zero-mean iid process (innovation process)
and $\varrho_1, \cdots, \varrho_p$ are the model parameters

Special case: AR(1) model

- Autoregressive model of order $p = 1$

$$X_n = Z_n + \mu + \varrho \cdot (X_{n-1} - \mu)$$

➡ Completely specified by mean μ, correlation coefficient ϱ, and pdf $f_Z(z)$ of iid process $\{Z_n\}$

➡ Important type of stationary Markov process for continuous random processes

Gaussian Processes

Gaussian Random Process

- All finite collections of random variables X_n are Gaussian random vectors
- N-th order pdf is given by N-th order auto-covariance matrix $\boldsymbol{C}_N$ and mean μ

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \frac{1}{\sqrt{(2\pi)^N |\boldsymbol{C}_N|}} \, e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{C}_N^{-1} (\boldsymbol{x}-\boldsymbol{\mu})} \qquad \text{with} \qquad \boldsymbol{\mu} = \begin{pmatrix} \mu \\ \vdots \\ \mu \end{pmatrix}$$

Gaussian Processes

Gaussian Random Process

- All finite collections of random variables X_n are Gaussian random vectors
- N-th order pdf is given by N-th order auto-covariance matrix $\boldsymbol{C}_N$ and mean μ

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \frac{1}{\sqrt{(2\pi)^N |\boldsymbol{C}_N|}} \, e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{C}_N^{-1} (\boldsymbol{x}-\boldsymbol{\mu})} \qquad \text{with} \qquad \boldsymbol{\mu} = \begin{pmatrix} \mu \\ \vdots \\ \mu \end{pmatrix}$$

Stationary Gauss-Markov Process

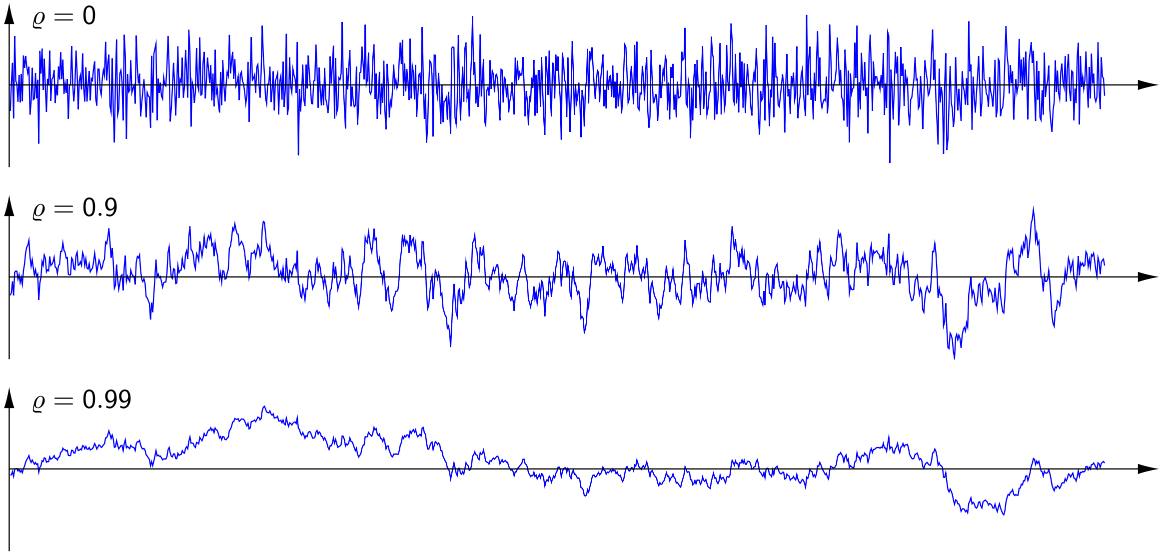
- Stationary Markov process that is also a Gaussian random process
- Can be constructed with Gaussian iid process $\boldsymbol{Z} = \{Z_n\}$ according to

$$X_n = \mu + \varrho\,(X_{n-1} - \mu) + Z_n$$

- Statistical properties are completely specified by **mean** μ, **variance** σ^2, **correlation coefficient** ϱ

➞ Will use it as very simple model for sources with memory

Examples of Gauss-Markov Processes (1000 Samples)



Summary of Mathematical Basics (for Continuous Case)

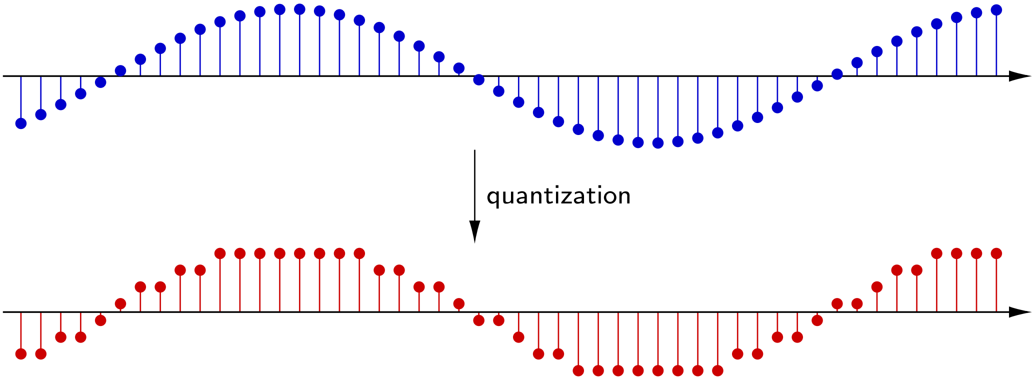
Continuous Random Variables

- Can take all values inside one or more non-zero intervals
- Cumulative distribution function (cdf): Continuous function
- Probability density function (pdf)
- Expected values: Mean, variance, covariance

Discrete-Time Continuous-Amplitude Random Processes

- Sequence of continuous random variables: Model for lossy source coding
- Types of random processes: Stationary, memoryless, iid, Markov
- Suitable model for real signals: Autoregressive processes
- Special importance for lossy source coding: Gaussian processes
- Simple model for sources with memory: Gauss-Markov process

Quantization



- "Lossy part" of source coding
- Non-reversible mapping from input to output samples
- Determines trade-off between signal fidelity and bit rate

Scalar Quantization: Functional Mapping

- Scalar Quantization: Functional mapping of an input sample to an output sample

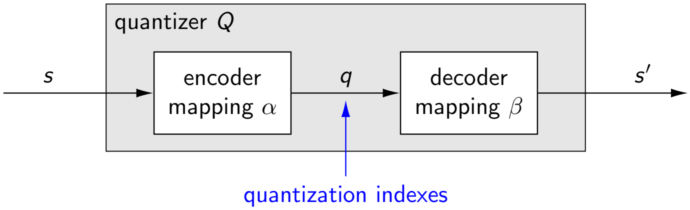
$$s' = Q(\,s\,)$$

- Input: • Discrete or continuous

- Output: • Set of obtainable output points is countable
  • Less obtainable output points than input points

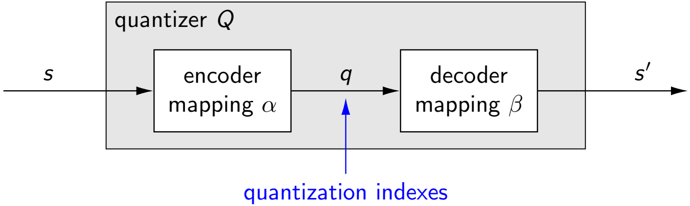
→ **Non-reversible loss in signal fidelity**

Structure of Scalar Quantizers: Encoder and Decoder Mapping



- Split quantizer Q into **encoder mapping** α and **decoder mapping** β

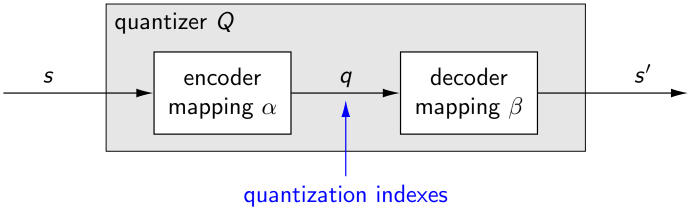
Structure of Scalar Quantizers: Encoder and Decoder Mapping



- Split quantizer Q into **encoder mapping** α and **decoder mapping** β
- Encoder mapping α: Maps input sample s to a quantizer index q (integer)

$$q = \alpha(\, s\,)$$

Structure of Scalar Quantizers: Encoder and Decoder Mapping



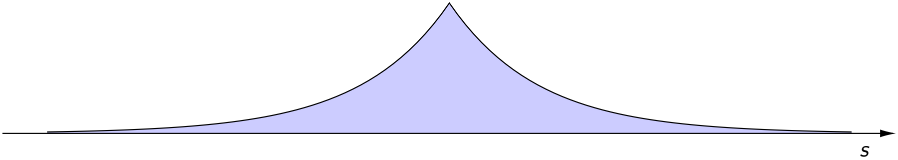
- Split quantizer Q into **encoder mapping** α and **decoder mapping** β
- Encoder mapping α: Maps input sample s to a quantizer index q (integer)

$$q = \alpha(s)$$

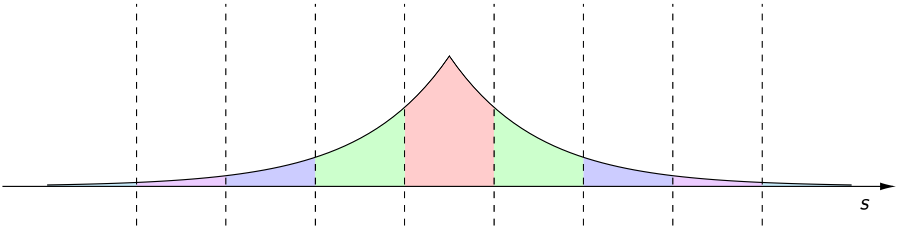
- Decoder mapping β: Maps quantizer index q to reconstructed samples s'

$$s' = \beta(q) = \beta(\alpha(s)) = Q(s)$$

Principle of Scalar Quantization

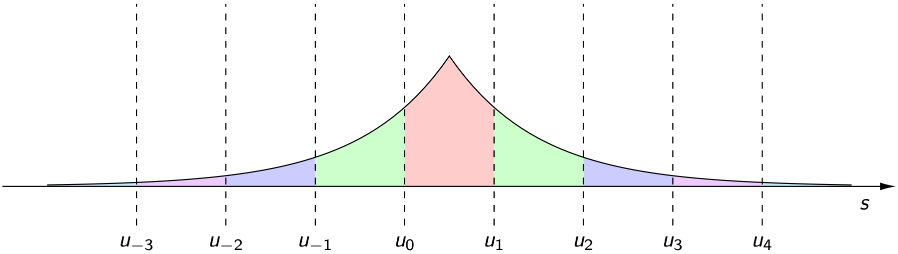


Principle of Scalar Quantization



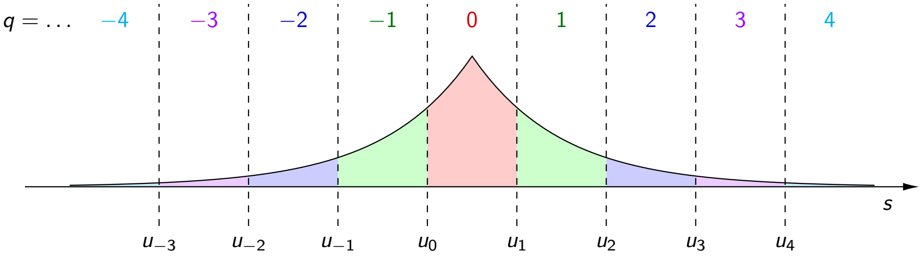
- Partition real line into a countable (typically finite) number of quantization intervals $\mathcal{I}_k$

Principle of Scalar Quantization



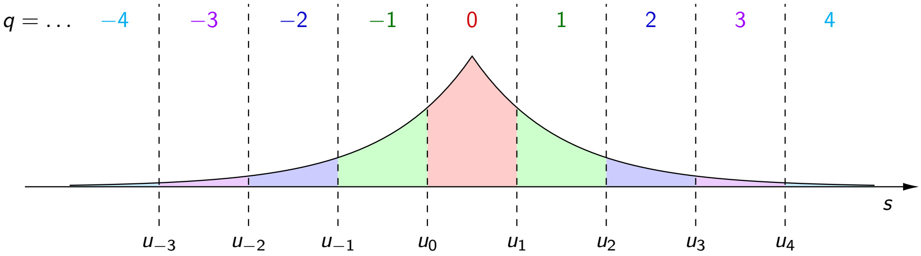
- Partition real line into a countable (typically finite) number of quantization intervals $\mathcal{I}_k$
  - Partitioning is given by decision thresholds $\{u_k\}$

Principle of Scalar Quantization



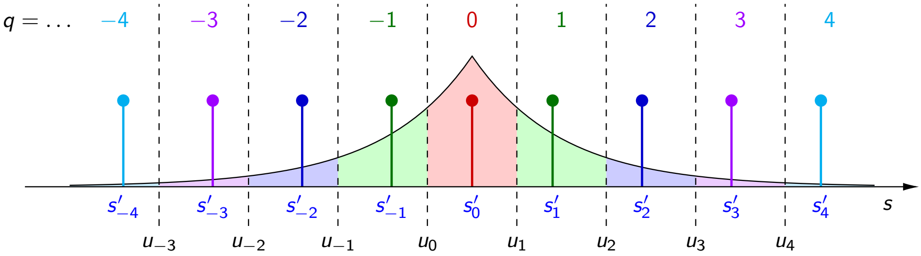
- Partition real line into a countable (typically finite) number of quantization intervals $\mathcal{I}_k$
  - Partitioning is given by decision thresholds $\{u_k\}$
  - Quantization intervals are labeled by quantization index q

Principle of Scalar Quantization



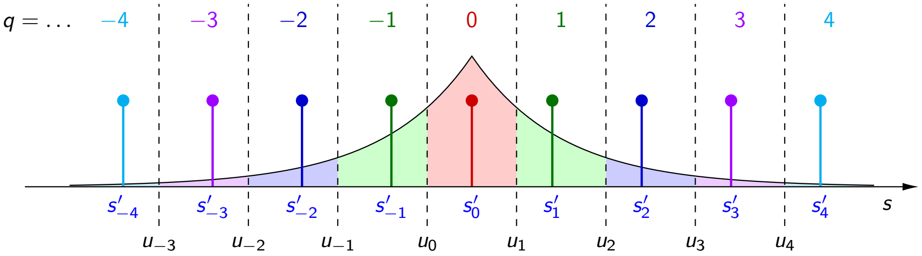
- Partition real line into a countable (typically finite) number of quantization intervals $\mathcal{I}_k$
  - Partitioning is given by decision thresholds $\{u_k\}$
  - Quantization intervals are labeled by quantization index q
  - A quantization interval is the given by $\mathcal{I}_k = [u_k, u_{k+1})$

Principle of Scalar Quantization



- Partition real line into a countable (typically finite) number of quantization intervals $\mathcal{I}_k$
  - Partitioning is given by decision thresholds $\{u_k\}$
  - Quantization intervals are labeled by quantization index q
  - A quantization interval is the given by $\mathcal{I}_k = [u_k, u_{k+1})$
- Each quantization interval $\mathcal{I}_k$ is associated with a reconstruction level $s'_k \in \mathcal{I}_k$

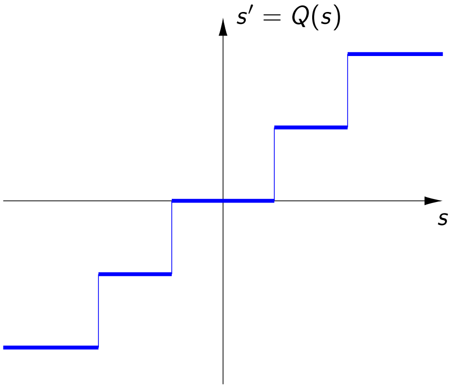
Principle of Scalar Quantization



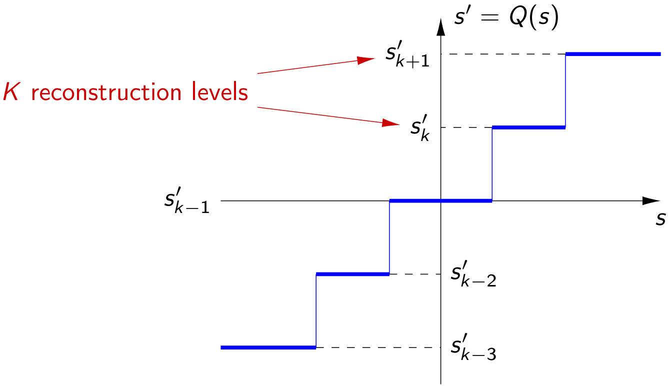
- Partition real line into a countable (typically finite) number of quantization intervals $\mathcal{I}_k$
  - Partitioning is given by decision thresholds $\{u_k\}$
  - Quantization intervals are labeled by quantization index q
  - A quantization interval is the given by $\mathcal{I}_k = [u_k, u_{k+1})$
- Each quantization interval $\mathcal{I}_k$ is associated with a reconstruction level $s'_k \in \mathcal{I}_k$

➡ **Scalar quantization**: Replace input value s that falls inside $\mathcal{I}_k$ with reconstruction value s'_k

Scalar Quantization: Input-Output Function

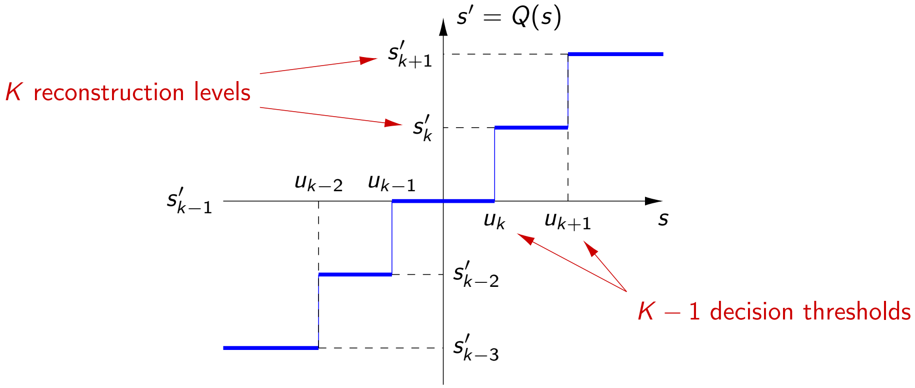


Scalar Quantization: Input-Output Function



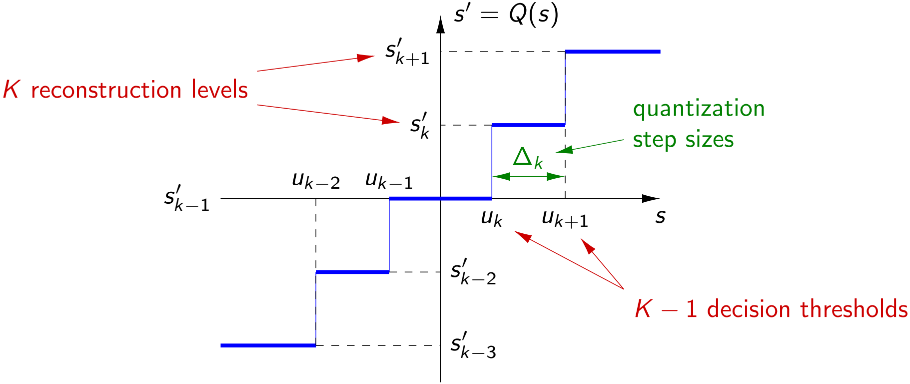
- Scalar quantizer mapping: $Q: \ \mathbb{R} \mapsto \{\cdots, s'_{k-1}, s'_k, s'_{k+1}, \cdots\}$

Scalar Quantization: Input-Output Function



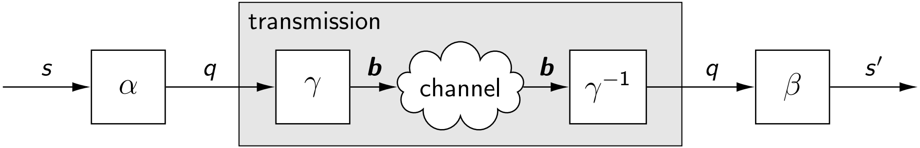
- Scalar quantizer mapping: $Q: \ \mathbb{R} \mapsto \{ \cdots, s'_{k-1}, s'_k, s'_{k+1}, \cdots \}$
- Quantization intervals: $\mathcal{I}_k = [u_k, u_{k+1})$

Scalar Quantization: Input-Output Function



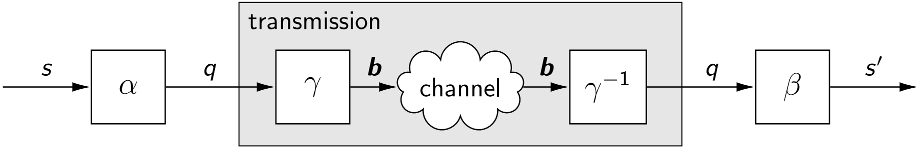
- Scalar quantizer mapping: $Q: \ \mathbb{R} \mapsto \{\cdots, s'_{k-1}, s'_k, s'_{k+1}, \cdots\}$
- Quantization intervals: $\mathcal{I}_k = [u_k, u_{k+1})$
- Quantization step sizes: $\Delta_k = u_{k+1} - u_k$

Scalar Quantization and Entropy Coding



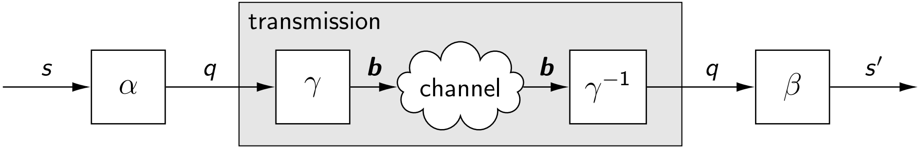
- Add **lossless coding** γ of quantization indexes (e.g., Huffman or arithmetic coding)

Scalar Quantization and Entropy Coding



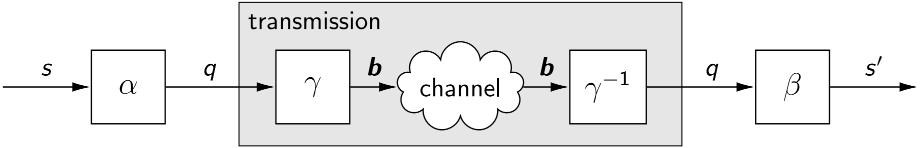
- Add **lossless coding** γ of quantization indexes (e.g., Huffman or arithmetic coding)
- Encoding/decoding process:

Scalar Quantization and Entropy Coding



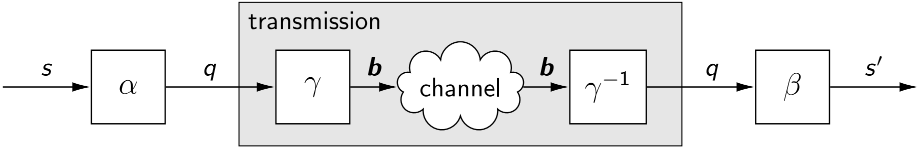
- Add **lossless coding** γ of quantization indexes (e.g., Huffman or arithmetic coding)
- Encoding/decoding process:
  1. Encoder mapping α: Input samples $s \mapsto$ quantization indexes q

Scalar Quantization and Entropy Coding



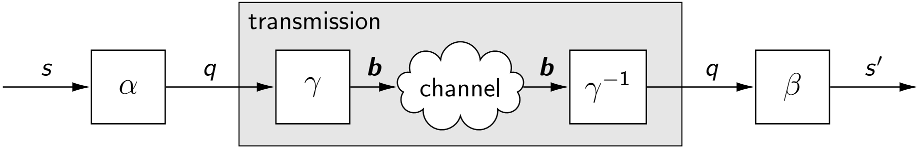
- Add **lossless coding** γ of quantization indexes (e.g., Huffman or arithmetic coding)
- Encoding/decoding process:
  1. Encoder mapping α: Input samples $s \mapsto$ quantization indexes q
  2. Lossless mapping γ: Quantization indexes $q \mapsto$ bitstream $\boldsymbol{b}$

Scalar Quantization and Entropy Coding



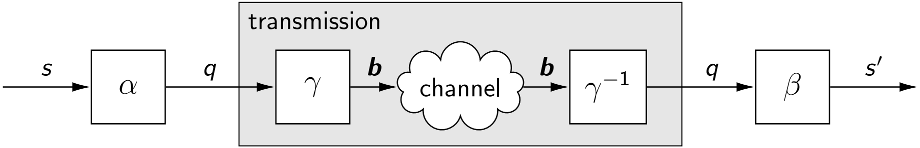
- Add **lossless coding** γ of quantization indexes (e.g., Huffman or arithmetic coding)
- Encoding/decoding process:
  1. Encoder mapping α: Input samples $s \mapsto$ quantization indexes q
  2. Lossless mapping γ: Quantization indexes $q \mapsto$ bitstream $\boldsymbol{b}$
  3. Transmission channel: Transmission of bitstream (assume: error-free)

Scalar Quantization and Entropy Coding



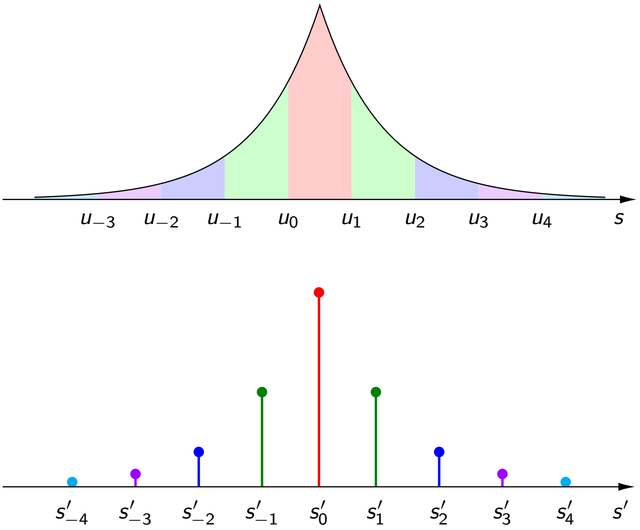
- Add **lossless coding** γ of quantization indexes (e.g., Huffman or arithmetic coding)
- Encoding/decoding process:
  1. Encoder mapping α: Input samples $s \mapsto$ quantization indexes q
  2. Lossless mapping γ: Quantization indexes $q \mapsto$ bitstream $\boldsymbol{b}$
  3. Transmission channel: Transmission of bitstream (assume: error-free)
  4. Lossless mapping γ^{-1}: Bitstream $\boldsymbol{b} \mapsto$ quantization indexes q

Scalar Quantization and Entropy Coding



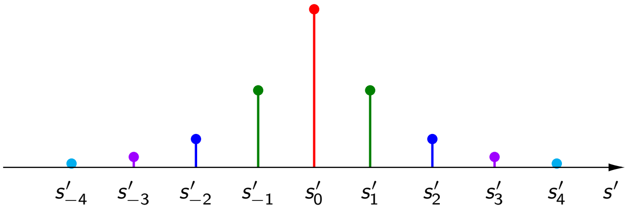
- Add **lossless coding** γ of quantization indexes (e.g., Huffman or arithmetic coding)
- Encoding/decoding process:
  1. Encoder mapping α: Input samples $s \mapsto$ quantization indexes q
  2. Lossless mapping γ: Quantization indexes $q \mapsto$ bitstream $\boldsymbol{b}$
  3. Transmission channel: Transmission of bitstream (assume: error-free)
  4. Lossless mapping γ^{-1}: Bitstream $\boldsymbol{b} \mapsto$ quantization indexes q
  5. Decoder mapping β: Quantization indexes $q \mapsto$ reconstructed samples s'

Scalar Quantization: Discretization of Pdf

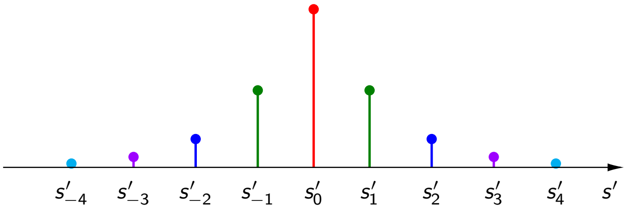


$$p_k = \mathrm{P}(S' = s'_k) = \int_{u_k}^{u_{k+1}} f(s)\,\mathrm{d}s$$

Performance of Scalar Quantizers: Bit Rate



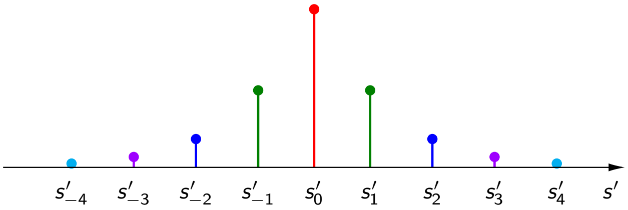
Performance of Scalar Quantizers: Bit Rate



- Average bit rate R (ℓ_k = codeword length for quantization index k)

$$R = \mathrm{E}\{\, \ell(\, S'\,)\,\} = \mathrm{E}\{\, \ell(\, \alpha(S)\,)\,\}$$

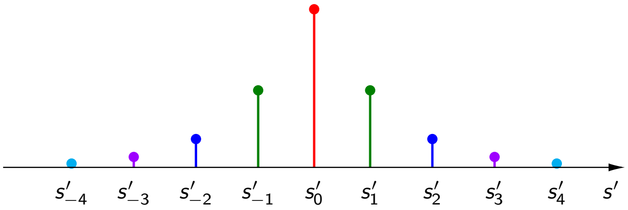
Performance of Scalar Quantizers: Bit Rate



- Average bit rate R (ℓ_k = codeword length for quantization index k)

$$R = \mathrm{E}\{\,\ell(\,S'\,)\,\} = \mathrm{E}\{\,\ell(\,\alpha(S)\,)\,\} = \sum_k p_k\,\ell_k$$

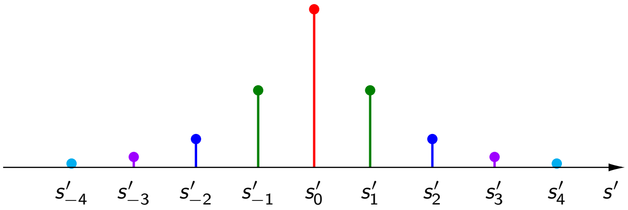
Performance of Scalar Quantizers: Bit Rate



- Average bit rate R (ℓ_k = codeword length for quantization index k)

$$R = \mathrm{E}\{\,\ell(\,S'\,)\,\} = \mathrm{E}\{\,\ell(\,\alpha(S)\,)\,\} = \sum_k p_k\,\ell_k \qquad \text{with} \qquad p_k = \int_{u_k}^{u_{k+1}} f(s)\,\mathrm{d}s$$

Performance of Scalar Quantizers: Bit Rate

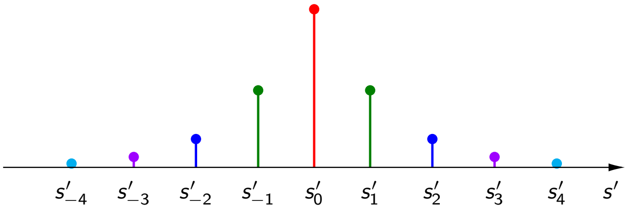


- Average bit rate R (ℓ_k = codeword length for quantization index k)

$$R = \mathrm{E}\{\,\ell(\,S'\,)\,\} = \mathrm{E}\{\,\ell(\,\alpha(S)\,)\,\} = \sum_k p_k\,\ell_k \qquad \text{with} \qquad p_k = \int_{u_k}^{u_{k+1}} f(s)\,\mathrm{d}s$$

- Approximations (without knowledge of actual entropy coding)
  - ➡ fixed-length coding: $R = \lceil\, \log_2 K \,\rceil$ (K: number of quantization intervals)

Performance of Scalar Quantizers: Bit Rate

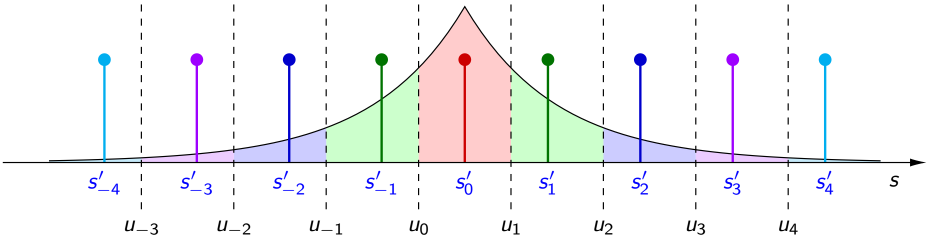


- Average bit rate R (ℓ_k = codeword length for quantization index k)

$$R = \mathrm{E}\{\,\ell(\,S'\,)\,\} = \mathrm{E}\{\,\ell(\,\alpha(S)\,)\,\} = \sum_k p_k\,\ell_k \qquad \text{with} \qquad p_k = \int_{u_k}^{u_{k+1}} f(s)\,\mathrm{d}s$$

- Approximations (without knowledge of actual entropy coding)
  - ➡ fixed-length coding: $R = \lceil\, \log_2 K \,\rceil$ (K: number of quantization intervals)
  - ➡ optimal entropy coding: $R = H(S') = H(\,\alpha(S)\,) = -\sum_k p_k \,\log_2 p_k$

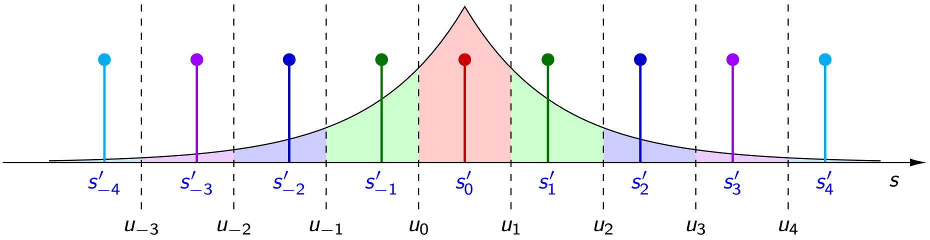
Performance of Scalar Quantizers: MSE Distortion



- Average MSE distortion D is given by

$$D = \mathrm{E}\Big\{ \big(S - Q(S) \big)^2 \Big\}$$

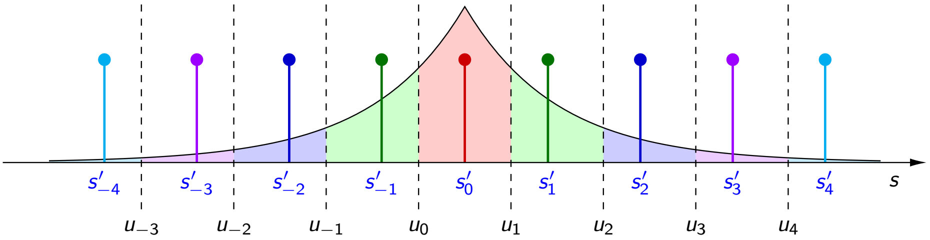
Performance of Scalar Quantizers: MSE Distortion



- Average MSE distortion D is given by

$$D = \mathrm{E}\left\{ \left(S - Q(S) \right)^2 \right\} = \int_{-\infty}^{\infty} (s - Q(s))^2 \, f(s) \, \mathrm{d}s$$

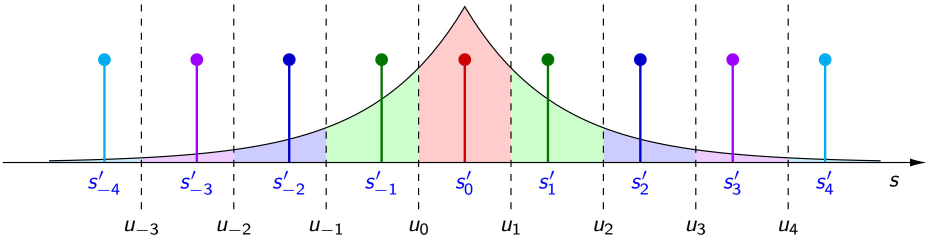
Performance of Scalar Quantizers: MSE Distortion



- Average MSE distortion D is given by

$$D = \mathrm{E}\Big\{ \big(S - Q(S) \big)^2 \Big\} = \int_{-\infty}^{\infty} (s - Q(s))^2 \, f(s) \, \mathrm{d}s = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s$$

Performance of Scalar Quantizers: MSE Distortion



- Average MSE distortion D is given by

$$D = \mathrm{E}\left\{ \left(S - Q(S) \right)^2 \right\} = \int_{-\infty}^{\infty} (s - Q(s))^2 \, f(s) \, \mathrm{d}s = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s$$

➡ Similar for other additive distortion measures (e.g., all p-norm distortion measures)

Optimal Scalar Quantizer for Fixed-Length Coding

Goal: Minimize MSE Distortion for Quantizer with K Quantization Intervals

- Neglect impact of entropy coding ➡ Consider fixed-length coding

➡ Rate R and MSE distortion D are given by

$$R = \lceil \log_2 K \rceil \qquad (\text{typically } K = 2^B, \text{ with } B \text{ being the bits per codeword})$$

$$D = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s$$

Optimal Scalar Quantizer for Fixed-Length Coding

Goal: Minimize MSE Distortion for Quantizer with K Quantization Intervals

- Neglect impact of entropy coding ➡ Consider fixed-length coding

➡ Rate R and MSE distortion D are given by

$$R = \lceil \log_2 K \rceil \qquad \text{(typically } K = 2^B\text{, with } B \text{ being the bits per codeword)}$$

$$D = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s$$

Optimize Quantizer of size K

- Bit rate R is independent on decision thresholds and reconstruction levels (R is given by K)

Optimal Scalar Quantizer for Fixed-Length Coding

Goal: Minimize MSE Distortion for Quantizer with K Quantization Intervals

- Neglect impact of entropy coding ➡ Consider fixed-length coding

➡ Rate R and MSE distortion D are given by

$$R = \lceil \log_2 K \rceil \qquad \text{(typically } K = 2^B\text{, with } B \text{ being the bits per codeword)}$$

$$D = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s$$

Optimize Quantizer of size K

- Bit rate R is independent on decision thresholds and reconstruction levels (R is given by K)
- Distortion (MSE) depends on
  - ➡ K reconstruction levels s'_k
  - ➡ $K - 1$ decision thresholds u_k

Centroid Condition

$$D = \sum_{\forall i} \int_{u_i}^{u_{i+1}} (s - s_i')^2 \, f(s) \, \mathrm{d}s$$

- Optimize reconstruction levels s_k' for given decision thresholds u_k

Centroid Condition

$$D = \sum_{\forall i} \int_{u_i}^{u_{i+1}} (s - s_i')^2 \, f(s) \, \mathrm{d}s$$

- Optimize reconstruction levels s_k' for given decision thresholds u_k

$$\frac{\partial}{\partial s_k'} D = \int_{u_k}^{u_{k+1}} 2 \cdot (s - s_k') \cdot (-1) \cdot f(s) \, \mathrm{d}s = 0$$

Centroid Condition

$$D = \sum_{\forall i} \int_{u_i}^{u_{i+1}} (s - s'_i)^2 \, f(s) \, \mathrm{d}s$$

- Optimize reconstruction levels s'_k for given decision thresholds u_k

$$\frac{\partial}{\partial s'_k} D = \int_{u_k}^{u_{k+1}} 2 \cdot (s - s'_k) \cdot (-1) \cdot f(s) \, \mathrm{d}s = 0$$

$$\int_{u_k}^{u_{k+1}} s \, f(s) \, \mathrm{d}s = s'_k \cdot \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

Centroid Condition

$$D = \sum_{\forall i} \int_{u_i}^{u_{i+1}} (s - s'_i)^2 \, f(s) \, \mathrm{d}s$$

- Optimize reconstruction levels s'_k for given decision thresholds u_k

$$\frac{\partial}{\partial s'_k} D = \int_{u_k}^{u_{k+1}} 2 \cdot (s - s'_k) \cdot (-1) \cdot f(s) \, \mathrm{d}s = 0$$

$$\int_{u_k}^{u_{k+1}} s \, f(s) \, \mathrm{d}s = s'_k \cdot \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

➡ **Centroid Condition for MSE Distortion**

$$\boxed{ s'_k = \mathrm{E}\{\, S \mid S \in \mathcal{I}_k \,\} = \frac{1}{p_k} \int_{u_k}^{u_{k+1}} s \, f(s) \, \mathrm{d}s = \frac{\int_{u_k}^{u_{k+1}} s \, f(s) \, \mathrm{d}s}{\int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s} }$$

➡ Optimal reconstruction level s'_k is given by conditional mean

Nearest Neighbour Condition

$$D = \sum_{\forall i} \int_{u_i}^{u_{i+1}} (s - s'_i)^2 \, f(s) \, \mathrm{d}s$$

- Optimize decision thresholds u_k for given reconstruction levels s'_k

Nearest Neighbour Condition

$$D = \sum_{\forall i} \int_{u_i}^{u_{i+1}} (s - s'_i)^2 \, f(s) \, \mathrm{d}s$$

- Optimize decision thresholds u_k for given reconstruction levels s'_k
  - Threshold u_k lies somewhere between neighboring reconstruction levels: $s'_{k-1} < u_k < s'_k$

Nearest Neighbour Condition

$$D = \sum_{\forall i} \int_{u_i}^{u_{i+1}} (s - s_i')^2 \, f(s) \, \mathrm{d}s$$

- Optimize decision thresholds u_k for given reconstruction levels s_k'
  - Threshold u_k lies somewhere between neighboring reconstruction levels: $s_{k-1}' < u_k < s_k'$
  - At the threshold u_k, we have the same distortion for both neighbouring intervals

$$(u_k - s_{k-1}')^2 = (u_k - s_k')^2$$

Nearest Neighbour Condition

$$D = \sum_{\forall i} \int_{u_i}^{u_{i+1}} (s - s'_i)^2 \, f(s) \, \mathrm{d}s$$

- Optimize decision thresholds u_k for given reconstruction levels s'_k
  - Threshold u_k lies somewhere between neighboring reconstruction levels: $s'_{k-1} < u_k < s'_k$
  - At the threshold u_k, we have the same distortion for both neighbouring intervals

$$(u_k - s'_{k-1})^2 = (u_k - s'_k)^2$$

$$u_k - s'_{k-1} = s'_k - u_k$$

Nearest Neighbour Condition

$$D = \sum_{\forall i} \int_{u_i}^{u_{i+1}} (s - s'_i)^2 \, f(s) \, \mathrm{d}s$$

- Optimize decision thresholds u_k for given reconstruction levels s'_k
  - Threshold u_k lies somewhere between neighboring reconstruction levels: $s'_{k-1} < u_k < s'_k$
  - At the threshold u_k, we have the same distortion for both neighbouring intervals

$$\begin{aligned} (u_k - s'_{k-1})^2 &= (u_k - s'_k)^2 \\ u_k - s'_{k-1} &= s'_k - u_k \\ 2\,u_k &= s'_{k-1} + s'_k \end{aligned}$$

Nearest Neighbour Condition

$$D = \sum_{\forall i} \int_{u_i}^{u_{i+1}} (s - s'_i)^2 \, f(s) \, \mathrm{d}s$$

- Optimize decision thresholds u_k for given reconstruction levels s'_k
  - Threshold u_k lies somewhere between neighboring reconstruction levels: $s'_{k-1} < u_k < s'_k$
  - At the threshold u_k, we have the same distortion for both neighbouring intervals

$$\begin{aligned}
(u_k - s'_{k-1})^2 &= (u_k - s'_k)^2 \\
u_k - s'_{k-1} &= s'_k - u_k \\
2\,u_k &= s'_{k-1} + s'_k
\end{aligned}$$

➡ **Nearest Neighbour Condition for MSE Distortion**

$$\boxed{u_k = \frac{1}{2}\left(s'_{k-1} + s'_k\right)}$$

➡ Optimal decision threshold u_k lies in the middle between the neighboring reconstruction levels

Lloyd Quantizer: Minimization of Distortion

Necessary Conditions for Minimizing MSE Distortion

1. Centroid condition

$$s'_k = \frac{\int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s}{\int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s}$$

Lloyd Quantizer: Minimization of Distortion

Necessary Conditions for Minimizing MSE Distortion

1. Centroid condition

$$s'_k = \frac{\int_{u_k}^{u_{k+1}} s\, f(s)\,\mathrm{d}s}{\int_{u_k}^{u_{k+1}} f(s)\,\mathrm{d}s}$$

2. Nearest neighbour condition

$$u_k = \frac{1}{2}\left(s'_k + s'_{k-1}\right)$$

Lloyd Quantizer: Minimization of Distortion

Necessary Conditions for Minimizing MSE Distortion

1. Centroid condition

$$s'_k = \frac{\int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s}{\int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s}$$

2. Nearest neighbour condition

$$u_k = \frac{1}{2}\left(s'_k + s'_{k-1}\right)$$

Design of Lloyd quantizers

- In general: Cannot be derived in closed form

Lloyd Quantizer: Minimization of Distortion

Necessary Conditions for Minimizing MSE Distortion

1. Centroid condition

$$s'_k = \frac{\int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s}{\int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s}$$

2. Nearest neighbour condition

$$u_k = \frac{1}{2}\left(s'_k + s'_{k-1}\right)$$

Design of Lloyd quantizers

- In general: Cannot be derived in closed form

→ Iterative algorithm consisting of

- Optimize decision thresholds u_k given reconstruction levels s'_k
- Optimize reconstruction levels s'_k given decision thresholds u_k

Lloyd Algorithm for Given Pdf (MSE Distortion)

Given is:

- the size K of the quantizer (i.e., the number of quantization intervals)
- the marginal probability density function $f(s)$ of the source

Lloyd Algorithm for Given Pdf (MSE Distortion)

Given is:
- the size K of the quantizer (i.e., the number of quantization intervals)
- the marginal probability density function $f(s)$ of the source

Iterative quantizer design

1. Choose an initial set of K reconstruction levels $\{s'_k\}$

Lloyd Algorithm for Given Pdf (MSE Distortion)

Given is:

- the size K of the quantizer (i.e., the number of quantization intervals)
- the marginal probability density function $f(s)$ of the source

Iterative quantizer design

1. Choose an initial set of K reconstruction levels $\{s'_k\}$
2. Update the $K-1$ decision thresholds $\{u_k\}$ according to

$$u_k = \frac{s'_k + s'_{k-1}}{2} \qquad \text{(nearest neighbor condition)}$$

Lloyd Algorithm for Given Pdf (MSE Distortion)

Given is:

- the size K of the quantizer (i.e., the number of quantization intervals)
- the marginal probability density function $f(s)$ of the source

Iterative quantizer design

1. Choose an initial set of K reconstruction levels $\{s'_k\}$
2. Update the $K-1$ decision thresholds $\{u_k\}$ according to

$$u_k = \frac{s'_k + s'_{k-1}}{2} \qquad \text{(nearest neighbor condition)}$$

3. Update the K reconstruction levels $\{s'_k\}$ according to

$$s'_k = \frac{\int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s}{\int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s} \qquad \text{(centroid condition)}$$

Lloyd Algorithm for Given Pdf (MSE Distortion)

Given is:
- the size K of the quantizer (i.e., the number of quantization intervals)
- the marginal probability density function $f(s)$ of the source

Iterative quantizer design

1. Choose an initial set of K reconstruction levels $\{s'_k\}$
2. Update the $K-1$ decision thresholds $\{u_k\}$ according to

$$u_k = \frac{s'_k + s'_{k-1}}{2} \qquad \text{(nearest neighbor condition)}$$

3. Update the K reconstruction levels $\{s'_k\}$ according to

$$s'_k = \frac{\int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s}{\int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s} \qquad \text{(centroid condition)}$$

4. Repeat the previous two steps until convergence

Lloyd Algorithm for a Training Set (MSE Distortion)

Given is:

- the size K of the quantizer (i.e., the number of quantization intervals)
- a sufficiently large realization $\{s_n\}$ of considered source

Lloyd Algorithm for a Training Set (MSE Distortion)

Given is:
- the size K of the quantizer (i.e., the number of quantization intervals)
- a sufficiently large realization $\{s_n\}$ of considered source

Iterative quantizer design

1. Choose an initial set of K reconstruction levels $\{s'_k\}$

Lloyd Algorithm for a Training Set (MSE Distortion)

Given is:

- the size K of the quantizer (i.e., the number of quantization intervals)
- a sufficiently large realization $\{s_n\}$ of considered source

Iterative quantizer design

1. Choose an initial set of K reconstruction levels $\{s'_k\}$
2. Associate all samples of the training set $\{s_n\}$ with one of the quantization intervals $\mathcal{I}_k$

$$q(s_n) = \arg\min_{\forall k} (s_n - s'_k)^2 \qquad \text{(nearest neighbor condition)}$$

Lloyd Algorithm for a Training Set (MSE Distortion)

Given is:

- the size K of the quantizer (i.e., the number of quantization intervals)
- a sufficiently large realization $\{s_n\}$ of considered source

Iterative quantizer design

1. Choose an initial set of K reconstruction levels $\{s'_k\}$
2. Associate all samples of the training set $\{s_n\}$ with one of the quantization intervals $\mathcal{I}_k$

$$q(s_n) = \arg\min_{\forall k} \, (s_n - s'_k)^2 \qquad \text{(nearest neighbor condition)}$$

3. Update the reconstruction levels $\{s'_k\}$ according to

$$s'_k = \frac{1}{N_k} \sum_{n:\, q(s_n)=k} s_n \qquad \text{(centroid condition)}$$

where N_k is the number of samples associated with $\mathcal{I}_k$

Lloyd Algorithm for a Training Set (MSE Distortion)

Given is:

- the size K of the quantizer (i.e., the number of quantization intervals)
- a sufficiently large realization $\{s_n\}$ of considered source

Iterative quantizer design

1. Choose an initial set of K reconstruction levels $\{s'_k\}$
2. Associate all samples of the training set $\{s_n\}$ with one of the quantization intervals $\mathcal{I}_k$

$$q(s_n) = \arg\min_{\forall k} \, (s_n - s'_k)^2 \qquad \text{(nearest neighbor condition)}$$

3. Update the reconstruction levels $\{s'_k\}$ according to

$$s'_k = \frac{1}{N_k} \sum_{n:\, q(s_n)=k} s_n \qquad \text{(centroid condition)}$$

   where N_k is the number of samples associated with $\mathcal{I}_k$

4. Repeat the previous two steps until convergence

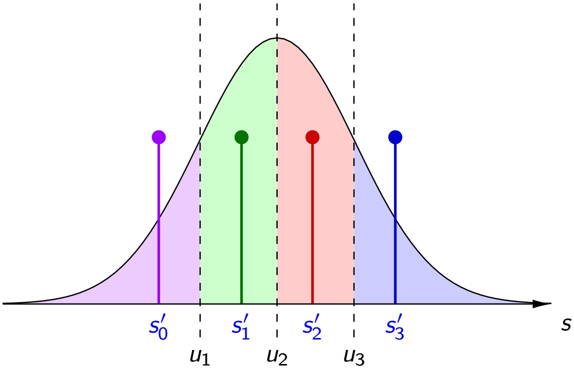
Example: Lloyd Algorithm for Gaussian Source

Gaussian Source

- Zero mean $\mu = 0$
- Unit variance $\sigma^2 = 1$

Lloyd Quantizer of size $K = 4$

- Decision thresholds: $u_1 = -0.982$, $u_2 = 0.000$, $u_3 = 0.982$
- Reconstruction levels: $s'_0 = -1.510$, $s'_1 = -0.453$, $s'_2 = 0.453$, $s'_3 = 1.510$

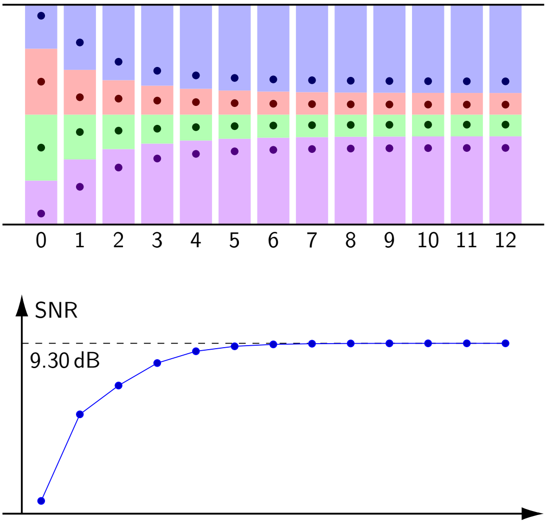


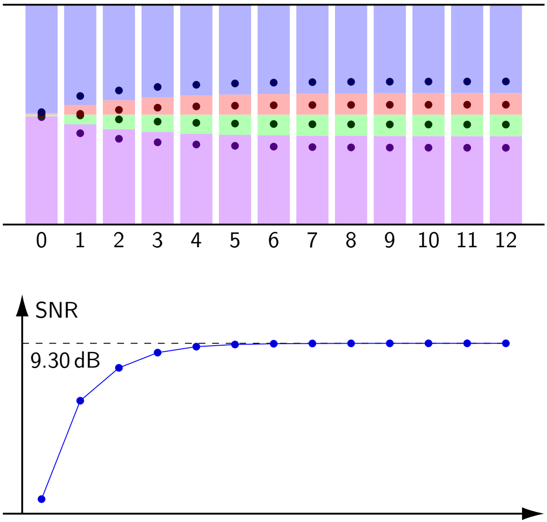
$R = 2.0$ (fixed-length coding)

$D = 0.117$

$\text{SNR} = 9.30$ dB

Example: Convergence of Lloyd Algorithm for Gaussian Source

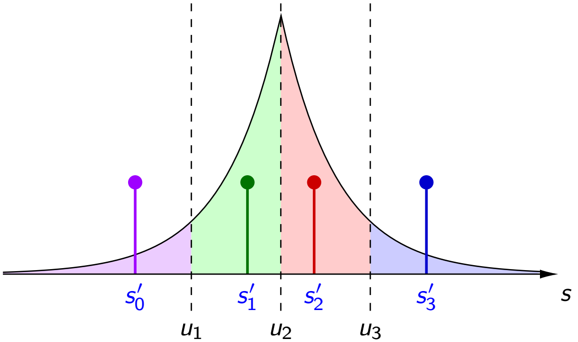




Example: Lloyd Algorithm for Laplacian Source

Laplacian Source

- Zero mean $\mu = 0$
- Unit variance $\sigma^2 = 1$

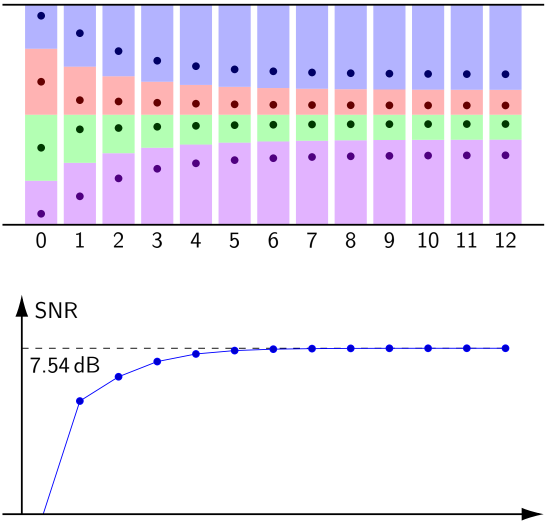


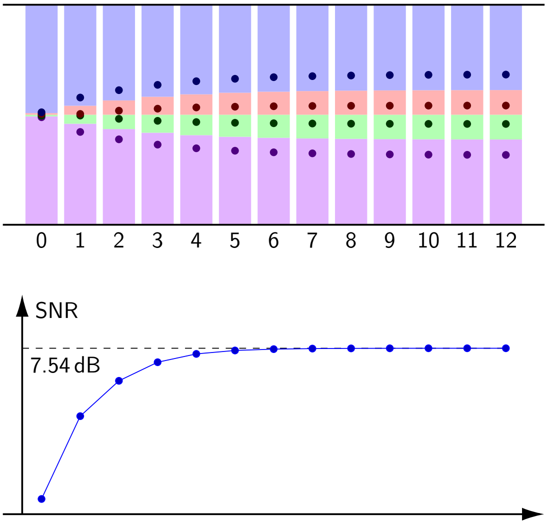
Lloyd Quantizer of size $K = 4$

- Decision thresholds: $$u_1 = -1.127 \quad u_2 = 0.000 \quad u_3 = 1.127$$
- Reconstruction levels: $$s'_0 = -1.834 \quad s'_1 = -0.420 \quad s'_2 = 0.420 \quad s'_3 = 1.834$$

$$R = 2.0 \quad \text{(fixed-length coding)}$$
$$D = 0.176$$
$$\text{SNR} = 7.54 \text{ dB}$$

Example: Convergence of Lloyd Algorithm for Laplacian Source





Centroid Quantizer at High Rates

High-Rate Approximation

- High rates: Pdf $f(s)$ is nearly constant inside each quantization interval

$$f(s) \approx \frac{p_k}{\Delta_k} = \frac{p_k}{u_{k+1} - u_k}$$

➡ Direct consequence: Reconstruction value s'_k lies in center of quantization interval $\mathcal{I}_k$

$$s'_k = \frac{1}{p_k} \int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s = \frac{1}{p_k} \cdot \frac{p_k}{u_{k+1} - u_k} \int_{u_k}^{u_{k+1}} s\, \mathrm{d}s$$

$$= \frac{1}{2} \cdot \frac{1}{u_{k+1} - u_k} \cdot \left(u_{k+1}^2 - u_k^2 \right) = \frac{1}{2} \cdot \frac{(u_{k+1} + u_k) \cdot (u_{k+1} - u_k)}{u_{k+1} - u_k}$$

$$= \frac{1}{2} \left(u_k + u_{k+1} \right) = u_k + \frac{\Delta_k}{2}$$

High-Rate Approximation of MSE Distortion for Centroid Quantizers

$$D = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \cdot f(s) \, \mathrm{d}s$$

High-Rate Approximation of MSE Distortion for Centroid Quantizers

$$D = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \cdot f(s)\,\mathrm{d}s$$

$$= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2\,\mathrm{d}s$$

High-Rate Approximation of MSE Distortion for Centroid Quantizers

$$
\begin{aligned}
D &= \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \cdot f(s)\, \mathrm{d}s \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2\, \mathrm{d}s \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{u_k - s'_k}^{u_{k+1} - s'_k} t^2\, \mathrm{d}t
\end{aligned}
$$

High-Rate Approximation of MSE Distortion for Centroid Quantizers

$$
\begin{aligned}
D &= \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \cdot f(s)\,\mathrm{d}s \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2\,\mathrm{d}s \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{u_k - s'_k}^{u_{k+1} - s'_k} t^2\,\mathrm{d}t \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{-\Delta_k/2}^{\Delta_k/2} t^2\,\mathrm{d}t
\end{aligned}
$$

High-Rate Approximation of MSE Distortion for Centroid Quantizers

$$
\begin{aligned}
D &= \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \cdot f(s)\,\mathrm{d}s \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2\,\mathrm{d}s \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{u_k - s'_k}^{u_{k+1} - s'_k} t^2\,\mathrm{d}t \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{-\Delta_k/2}^{\Delta_k/2} t^2\,\mathrm{d}t \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \cdot \frac{1}{3} \cdot \left(\frac{\Delta_k^3}{8} + \frac{\Delta_k^3}{8} \right)
\end{aligned}
$$

High-Rate Approximation of MSE Distortion for Centroid Quantizers

$$
\begin{aligned}
D &= \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \cdot f(s)\,\mathrm{d}s \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2\,\mathrm{d}s \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{u_k - s'_k}^{u_{k+1} - s'_k} t^2\,\mathrm{d}t \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \int_{-\Delta_k/2}^{\Delta_k/2} t^2\,\mathrm{d}t \\
&= \sum_{\forall k} \frac{p_k}{\Delta_k} \cdot \frac{1}{3} \cdot \left(\frac{\Delta_k^3}{8} + \frac{\Delta_k^3}{8} \right)
\end{aligned}
$$

$$
\boxed{D = \frac{1}{12} \sum_{\forall k} p_k\, \Delta_k^2}
$$

High-Rate Approximation: MSE Distortion for Lloyd Quantizer

- Will use: **Hölders inequality** in the following form (with $x_k \geq 0$ and $y_k \geq 0$)

$$\alpha + \beta = 1 \qquad \Longrightarrow \qquad \Big(\sum_k x_k\Big)^{\alpha} \cdot \Big(\sum_k y_k\Big)^{\beta} \geq \sum_k x_k^{\alpha}\, y_k^{\beta}$$

with equality iff y_k is proportional to x_k, i.e., $y_k = \text{const} \cdot x_k$

High-Rate Approximation: MSE Distortion for Lloyd Quantizer

- Will use: **Hölders inequality** in the following form (with $x_k \geq 0$ and $y_k \geq 0$)

$$\alpha + \beta = 1 \qquad \Longrightarrow \qquad \left(\sum\nolimits_k x_k\right)^{\alpha} \cdot \left(\sum\nolimits_k y_k\right)^{\beta} \geq \sum\nolimits_k x_k^{\alpha}\, y_k^{\beta}$$

with equality iff y_k is proportional to x_k, i.e., $y_k = \text{const} \cdot x_k$

Average MSE distortion of Lloyd quantizer of size K (at high rates)

- Approximation for centroid quantizers

$$D = \frac{1}{12} \sum_{i=0}^{K-1} p_i\, \Delta_i^2 = \frac{1}{12} \sum_{i=0}^{K-1} f(s_i')\, \Delta_i^3$$

High-Rate Approximation: MSE Distortion for Lloyd Quantizer

- Will use: **Hölders inequality** in the following form (with $x_k \geq 0$ and $y_k \geq 0$)

$$\alpha + \beta = 1 \qquad \Longrightarrow \qquad \left(\sum_k x_k \right)^{\alpha} \cdot \left(\sum_k y_k \right)^{\beta} \geq \sum_k x_k^{\alpha} \, y_k^{\beta}$$

with equality iff y_k is proportional to x_k, i.e., $y_k = \text{const} \cdot x_k$

Average MSE distortion of Lloyd quantizer of size K (at high rates)

- Approximation for centroid quantizers

$$D = \frac{1}{12} \sum_{i=0}^{K-1} p_i \, \Delta_i^2 = \frac{1}{12} \sum_{i=0}^{K-1} f(s_i') \, \Delta_i^3$$

- Rewrite expression using $\sum_{i=0}^{K-1} (1/K) = K \cdot (1/K) = 1$

$$D = \frac{1}{12} \left(\left(\sum_{i=0}^{K-1} f(s_i') \, \Delta_i^3 \right)^{\frac{1}{3}} \cdot \left(\sum_{i=0}^{K-1} \frac{1}{K} \right)^{\frac{2}{3}} \right)^3$$

High-Rate Approximation: MSE Distortion for Lloyd Quantizer

Average MSE distortion of Lloyd quantizer of size K (at high rates)

- Apply Hölders inequality

$$D \geq \frac{1}{12} \left(\sum_{i=0}^{K-1} \left(f(s_i') \, \Delta_i^3 \right)^{\frac{1}{3}} \left(\frac{1}{K} \right)^{\frac{2}{3}} \right)^3 = \frac{1}{12\,K^2} \left(\sum_{i=0}^{K-1} \sqrt[3]{f(s_i')} \, \Delta_i \right)^3$$

with equality iff $\Delta_i \sqrt[3]{f(s_i')} = \text{const}$

High-Rate Approximation: MSE Distortion for Lloyd Quantizer

Average MSE distortion of Lloyd quantizer of size K (at high rates)

- Apply Hölders inequality

$$D \geq \frac{1}{12} \left(\sum_{i=0}^{K-1} \left(f(s_i') \, \Delta_i^3 \right)^{\frac{1}{3}} \left(\frac{1}{K} \right)^{\frac{2}{3}} \right)^3 = \frac{1}{12\,K^2} \left(\sum_{i=0}^{K-1} \sqrt[3]{f(s_i')} \, \Delta_i \right)^3$$

  with equality iff $\Delta_i \sqrt[3]{f(s_i')} = \text{const}$

- Remember: Lloyd quantizer minimizes distortion for given size K

$$D = \frac{1}{12\,K^2} \left(\sum_{i=0}^{K-1} \sqrt[3]{f(s_i')} \, \Delta_i \right)^3$$

High-Rate Approximation: MSE Distortion for Lloyd Quantizer

Average MSE distortion of Lloyd quantizer of size K (at high rates)

- Apply Hölders inequality

$$D \geq \frac{1}{12} \left(\sum_{i=0}^{K-1} \left(f(s_i')\,\Delta_i^3 \right)^{\frac{1}{3}} \left(\frac{1}{K} \right)^{\frac{2}{3}} \right)^3 = \frac{1}{12\,K^2} \left(\sum_{i=0}^{K-1} \sqrt[3]{f(s_i')}\,\Delta_i \right)^3$$

with equality iff $\Delta_i\,\sqrt[3]{f(s_i')} = \text{const}$

- Remember: Lloyd quantizer minimizes distortion for given size K

$$D = \frac{1}{12\,K^2} \left(\sum_{i=0}^{K-1} \sqrt[3]{f(s_i')}\,\Delta_i \right)^3$$

- Asymptotic limit for large K ($\Delta_k \to 0$)

$$D = \frac{1}{12\,K^2} \left(\int_{-\infty}^{\infty} \sqrt[3]{f(s)}\,\mathrm{d}s \right)^3$$

High-Rate Approximation: Lloyd Quantizer with Fixed-Length Coding

MSE Distortion for Lloyd Quantizer at High Rates and Rate for Fixed-Length Coding

$$D = \frac{1}{12\,K^2} \left(\int_{-\infty}^{\infty} \sqrt[3]{f(s)}\,\mathrm{d}s \right)^3 \qquad \text{and} \qquad R = \log_2 K \quad \Longrightarrow \quad \frac{1}{K^2} = 2^{-2R}$$

High-Rate Approximation: Lloyd Quantizer with Fixed-Length Coding

MSE Distortion for Lloyd Quantizer at High Rates and Rate for Fixed-Length Coding

$$D = \frac{1}{12\,K^2} \left(\int_{-\infty}^{\infty} \sqrt[3]{f(s)}\,\mathrm{d}s \right)^3 \qquad \text{and} \qquad R = \log_2 K \quad \Longrightarrow \quad \frac{1}{K^2} = 2^{-2R}$$

Lloyd Quantizer with Fixed-Length Coding at High Rates

- Panter and Dite approximation for operational distortion-rate function

$$D_F(R) \;=\; \frac{1}{12} \left(\int_{-\infty}^{\infty} \sqrt[3]{f(s)}\,\mathrm{d}s \right)^3 \cdot 2^{-2R}$$

$$\boxed{D_F(R) \;=\; \varepsilon_F^2 \cdot \sigma^2 \cdot 2^{-2R}} \qquad \text{with} \qquad \boxed{\varepsilon_F^2 = \frac{1}{12\,\sigma^2} \left(\int_{-\infty}^{\infty} \sqrt[3]{f(s)}\,\mathrm{d}s \right)^3}$$

Lloyd Quantizer with Fixed-Length Coding vs Panter-Dite Approximation

Uniform pdf

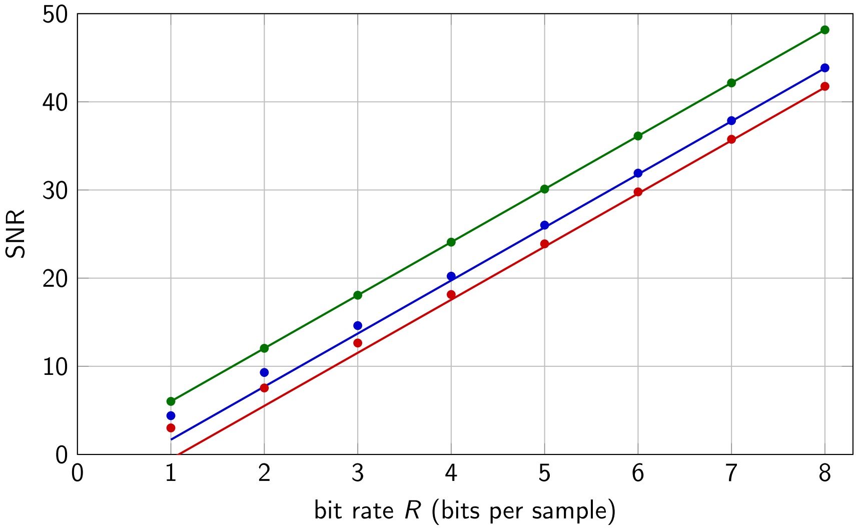
$$\varepsilon_F^2 = 1$$

Gaussian pdf

$$\varepsilon_F^2 = \frac{\sqrt{3}\,\pi}{2}$$

Laplacian pdf

$$\varepsilon_F^2 = 4.5$$



Summary of Lecture

Scalar Quantization

- Input-output function $s' = Q(s)$ is a staircase function
- Quantizer is characterized by K reconstruction levels s'_k and $K-1$ decision thresholds u_k

Lloyd Quantizer

- Minimizes distortion D for given number K of quantization intervals
- Two optimization criterions
  - Centroid condition (MSE): $s'_k = \mathrm{E}\{\, S \,|\, S \in \mathcal{I}_k \,\}$
  - Nearest neighbor condition (MSE): $u_k = (s'_k + s'_{k-1})/2$
- Lloyd quantizer design: Iterate between the two optimization criterions
- High-rate approximation of Lloyd quantizer with fixed-length coding (Panter-Dite approximation)

Next Steps

- Theoretical limits for lossy source coding
- Consider entropy coding in quantizer design

Exercise 1: Implement Lloyd Algorithm

Implement the Lloyd algorithm using a programming language of your choice.

- Test the algorithm (for quantizer sizes of $K = 2, 4, 8, 16, 32$) for
  - a unit-variance Gaussian pdf:
  $$f(s) = \frac{1}{\sqrt{2\pi}}\, e^{-\frac{1}{2}\, s^2}$$
  - a unit-variance Laplacian pdf:
  $$f(s) = \frac{1}{\sqrt{2}}\, e^{-\sqrt{2}\,|s|}$$
- Determine the distortion D for your quantizers.
- Compare the R-D performance of your quantizers (for $K = 2, 4, 8, 16, 32$) to the high-rate approximation for Lloyd quantizers with fixed-length codes.

You can implement the Lloyd algorithm that directly uses the pdf or the Lloyd algorithm that uses a training set (files with 1 000 000 samples in float32 format are provided on the course web site)

Exercise 2: Lloyd Quantizer for MSE Distortion (Alternative)

Given is a stationary source with a zero-mean Laplace pdf $f(x)$ and a symmetric 3-interval quantizer:

$$f(x) = \frac{1}{\sqrt{2\sigma^2}} \, e^{-\sqrt{\frac{2}{\sigma^2}}\,|x|} \qquad \text{and} \qquad Q(x) = \begin{cases} -b & : \quad x < -a \\ 0 & : \quad |x| \leq a \\ b & : \quad x > a \end{cases}$$

(a) Derive the optimal reconstruction value b as a function of the threshold a for MSE distortion.

Express the resulting distortion as function of the threshold a and the variance σ^2.

(b) Determine the decision threshold a in a way that a Lloyd quantizer for MSE distortion is obtained.

Determine the distortion and rate for the Lloyd quantizer by assuming fixed-length coding ($R = \log_2 K$) and compare the obtained R-D point with the high-rate approximation.

(c) Can the derived optimal quantizer for fixed-length coding be improved by adding entropy coding (without changing the decision thresholds and reconstruction levels)?

Exercise 3: Lloyd Quantizer for MAE Distortion (Another Alternative)

Given is a stationary source with a zero-mean Laplace pdf $f(x)$ and a symmetric 3-interval quantizer:

$$f(x) = \frac{1}{2m}\, e^{-\frac{|x|}{m}} \qquad \text{and} \qquad Q(x) = \begin{cases} -b & : \quad x < -a \\ 0 & : \quad |x| \le a \\ b & : \quad x > a \end{cases}$$

(a) Derive the centroid condition and nearest neighbor condition for MAE distortion

$$D = \mathrm{E}\{\, |S - S'|\, \}$$

(b) Derive the optimal reconstruction value b as a function of the threshold a for MAE distortion. Express the resulting distortion as function of the threshold a and the parameter m.

(c) Determine the decision threshold a in a way that a Lloyd quantizer for MAE distortion is obtained. Determine the distortion and rate for the quantizer by assuming fixed-length coding ($R = \log_2 K$).