27

TSBK02

Image and Audio Coding

Exercises

March 23, 2017

Contents

Introduction

This collection of exercises was conceived for the course *Image Coding and Data Compression* given at Linköping University, Sweden. Most of the problems are taken from old exams.

Below are some swedish notions translated to english. Some of them might be abbreviated.

Swedish	***English***	***Abbreviation***
Autokorrelationsfunktion	Auto correlation function; usually denoted $R_{XX}(k)$ or $R_{XX}(\tau)$.	*a.c.f.*
Spektraltäthet	Power spectral density; usually denoted $\Phi(\theta)$ or $\Phi(f)$.	*p.s.d.*
Täthetsfunktion	Probability density function.	*p.d.f.*

Reconstruction Error Measures

The distortion between a given set of vectors $\{\boldsymbol{x}_i\}$ and their reconstructions $\{\hat{\boldsymbol{x}}_i\}$ $i = 1 \ldots n$ is usually measured as *MSE (Mean Square Error)* . MSE is defined,

$$\text{MSE} \;=\; \frac{1}{n}\sum_{i=1}^{n} |\boldsymbol{x}_i - \hat{\boldsymbol{x}}_i|^2$$

where euclidian norm is used. When $\boldsymbol{x}$ is stochastic we also speak about distortion and most often mean the expected square error, usually denoted D,

$$D \;=\; \mathrm{E}\left\{ |\boldsymbol{X} - \hat{\boldsymbol{X}}|^2 \right\}$$

However, it is most common to express the distortion as a logarithm of the ratio between the signal variance $\sigma^2 = \mathrm{E}\left\{ |\boldsymbol{X} - \mathrm{E}\{ \boldsymbol{X} \}|^2 \right\}$ and D. The SNR (Signal to Noise Ratio) is defined as,

$$\text{SNR} \;=\; 10 \log_{10} \frac{\sigma^2}{D}$$

It is also possible to speak about SNR when having a deterministic situation, but then of course using $\frac{1}{n}\sum |\boldsymbol{x}_i - \boldsymbol{m}_x|^2$ for σ^2 and MSE for D.

Part I

Problems

1 Information Theoretic Concepts

1.1 The random variable X takes values in the alphabet $\{1, 2, 3, 4\}$. The probabilities for the different symbols are

$$P_X(1) = P_X(2) = P_X(3) = P_X(4) = 0.25$$

Calculate $H(X)$.

1.2 The random variable Y takes values in the alphabet $\{1, 2, 3, 4\}$. The probabilities for the different symbols are

$$P_Y(1) = 0.5,\ P_Y(2) = 0.25,\ P_Y(3) = P_Y(4) = 0.125$$

Calculate H(Y).

1.3 Suppose that X and Y from 1.1 and 1.2 are independent. Consider the joint source which generates pairs (X, Y).

a) Determine $P_{XY}(x, y)$. I.e., Prob$(X = x, Y = y)$ for all x, y!

b) Calculate $H(X, Y)$.

c) Show that $H(X, Y) = H(X) + H(Y)$ holds whenever X and Y are independent!

d) (*Generalization*) Show that,

$$H(X_1, X_2, \ldots, X_n) = H(X_1) + \ldots + H(X_n)$$

holds as long as the variables are mutually independent!

1.4 Let Z take values in the alphabet $\{1, 2, 3, 4\}$

a) Give an example of a distribution P_Z which maximizes $H(Z)$. Is P_Z unique?

b) Give an example of a distribution P_Z which minimizes $H(Z)$. Is P_Z unique?

1.5 Let U take values in the alphabet $\{0, 1, 2, \ldots, \infty\}$, with symbol probabilities $P_U(i) = q^i(1-q)$.

a) Check that $\sum_{i=0}^{\infty} P_U(i) = 1$.

b) Calculate $H(U)$.

c) Calculate the average value of U.

1.6 Show that,

$$I(X,Y) \triangleq H(X) - H(X \mid Y) \geq 0$$

Tips: Use the inequality $\ln x \leq x - 1$.

1.7 a) Show that,

$$H_\infty(S) = \sum_k w_k H(S_n | S_{n-1} = s_k)$$

holds for a stationary discrete Markov source where we by $H_\infty(S)$ mean the *entropy-rate* of the state sequence, w_k the asymptotic probabilities and s_k the different states.

b) Form a *hidden Markov* model $U_i = \phi(S_i)$. Let $H_\infty(U)$ denote the entropy-rate of this source and assume unifilarity. Show that

$$H_\infty(U) = H_\infty(S)$$

1.8 A binary memory-less source with $P = \{p, 1-p\}$ have entropy H. Express the entropies for the described sources below in terms of H!

a) A source with four symbols with probabilities: $P = \{\frac{p}{2}, \frac{1-p}{2}, \frac{1-p}{2}, \frac{p}{2}\}$

b) A ternary source: $P = \{\frac{p}{2}, 1-p, \frac{p}{2}\}$

c) The second order extension: $P^2 = \{p^2, p(1-p), (1-p)p, (1-p)^2\}$.

d) The n:th extension.

1.9 Let X be a discrete stochastic variable and f be an arbitrary function of the source-alphabet. Show that,

a) $H(f(X) \mid X) = 0$

b) $H(X, f(X)) = H(X)$

c) $H(f(X)) \leq H(X)$

1.10 Let X and Y be two independent discrete stochastic variables. Show that,

$$H(X) \;\leq\; H(X+Y) \;\leq\; H(X,Y)$$

1.11 Show that,

$$H(X\,|\,Z) \;\leq\; H(X\,|\,Y) \;+\; H(Y\,|\,Z)$$

Tips: Start with $H(X\,|\,Y) \;\geq\; H(X\,|\,YZ)$ and use the chain-rule.

1.12 Show that,

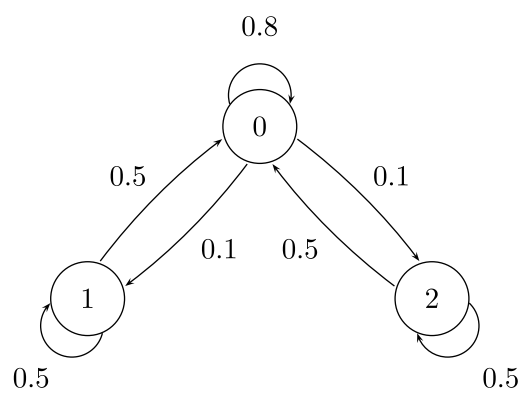
$$H(X_1,\ldots,X_{n+1}) \;=\; H(X_1,\ldots,X_n) + H(X_{n+1}\,|\,X_1,\ldots X_n)$$

1.13 A uniformly distributed random variable X takes values in the alphabet $\{0000, 0001, 0010, \ldots, 1011\}$ (the numbers 0 to 11 written as four bit binary numbers).

a) What is the entropy of each bit?

b) What is the entropy of X?

1.14 A Markov source of order 1 $\{X_i\}$ has state transition probabilities according to the figure below. Calculate the stationary probabilities for the states!



1.15 Consider the source in 1.14.

a) Calculate the memoryless entropy $H(X_i)$.

b) Calculate the block-entropy for pairs of symbols, i.e. $H(X_i, X_{i+1})$. Compare with $H(X_i) + H(X_{i+1}) = 2 \cdot H(X_i)$.

c) Calculate the conditional entropy $H(X_{i+1}\,|\,X_i)$.

1.16 A partly unknown binary stationary source generates alternating runs of zeros and ones. The run-lengths r are known to be independent and have the following distributions:

$$\begin{aligned} P(r \mid zeros) &= q_0^{r-1}(1-q_0) \\ P(r \mid ones) &= q_1^{r-1}(1-q_1) \end{aligned}$$

a) What are the expected run-lengths r_0 and r_1 [symbols/run]?

b) What are the entropies H_0 and H_1 of the runs?

c) Give a bound for the entropy-rate of the source-symbols! Suggest a source-model which can generate the run-lengths!

Discussion: Might the underlying source have even lower entropy than your bound?

2 Entropy Coding

2.1 A suggested binary code for the alphabet $\mathcal{A} = \{1, \ldots, 8\}$ has the code-word lengths $l_1 = 2$, $l_2 = 2$, $l_3 = 3$, $l_4 = 4$, $l_5 = 4$, $l_6 = 5$, $l_7 = 5$ and $l_8 = 6$. Can a prefix code with these lengths be constructed?

2.2 Show that if Kraft's inequality is satisfied (binary case) for a set of codeword lengths you can construct a binary prefix code with those lengths!

2.3 A memoryless source has the infinite alphabet $\mathcal{A} = \{1, 2, 3, \ldots,\}$ and symbol probabilities $P = \{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \ldots\}$, ie $P(i) = 2^{-i},\ i \in \mathcal{A}$.

Construct an optimal binary prefix code for the source and calculate the expected data rate R in bits/symbol.

2.4 A memoryless source has the alphabet $\mathcal{A} = \{x, y, z\}$ and symbol probabilities

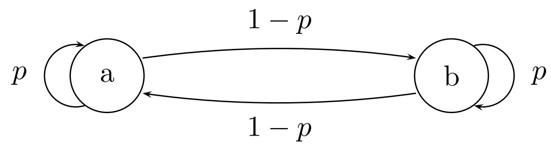
$$P(x) = 0.6, \quad P(y) = 0.3, \quad P(z) = 0.1$$

a) What is the entropy of the source?

b) Construct a Huffman code for single symbols from the source and calculate the rate of the code in bits/symbol.

c) Construct a Huffman code for pairs of symbols from the source and calculate the rate of the code in bits/symbol.

2.5 The following Markov source is given where $p = 0.9$:



Construct Huffman codes for coding two and three symbols with each codeword. Calculate the rates (in bits/symbol) for the two codes. Which code is best?

2.6 Consider the source in problem 2.5. It produces runs of a and b. Instead of coding symbols, we can code the length of each run. We thus create a new source Y which has an infinite alphabet of run lengths $\mathcal{B} = \{1, 2, 3, \ldots\}$.

a) What is the probability of a run of length r?

b) What is the average run length (in symbols/run) ?

c) What is the entropy of Y (in bits/run)?

d) What is the entropy rate of the original source (in bits/symbol) ?

2.7 We now want to make a simple systematic code for the run lengths of the source in 2.5.

a) Construct a four bit fixed length code for the run lengths 1 to 15, ie:

run length	codeword
1	0000
2	0001
3	0010
4	0011
5	0100
...	...
14	1101
15	1110

Longer runs are coded as 1111 followed by the codeword for a run of length-15, ie the run length 16 is coded as 1111 0000, the run length 17 is coded as 1111 0001, the run length 42 is coded as 1111 1111 1011 and so on.

Calculate the rate of the code in bits/symbol.

b) Change the codeword length to five bits and calculate the rate of the code.

2.8 We want to send documents using a fax. The fax can handle the colours black and white. It has been noted that text areas and picture areas of the document have different statistical properties. Documents are coded row-wise according to a time discrete stationary process with the following conditioned probabilities, which has been estimated from a large set of data

Colour of current pixel	*Probability of colour of next pixel*			
	Text area		*Picture area*	
	black	white	black	white
black	0.5	0.5	0.7	0.3
white	0.1	0.9	0.2	0.8

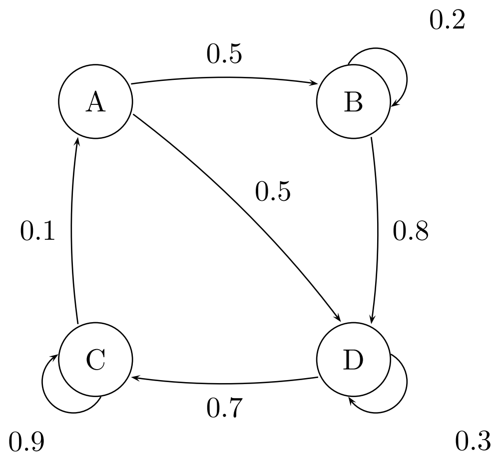
The probability of being in a text area is $\frac{4}{5}$ and the probability of being in a picture area is $\frac{1}{5}$.

Assume that these estimated probabilities are correct and answer the questions below.

a) Assume that we can ignore the cost of coding which areas of the document are text areas and which are picture areas. Give the best upper bound for the theoretically lowest mean bitrate that documents can be coded with. [bits/pixel]

b) Construct a huffman code for the text areas that has a rate of 0.65 [bits/pixel] or less.

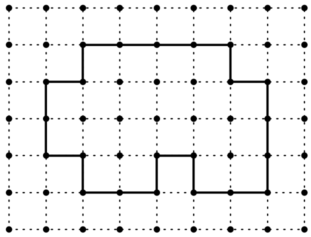
2.9 Consider the Markov source below (output = state)



a) Show that it is possible to encode the output from the source at a mean rate less than 0.6 bits/symbol.

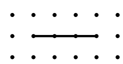
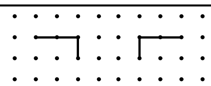
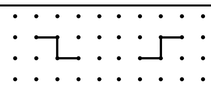
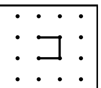
b) Extend the source to two symbols. Construct optimal FIVO tree codes for the original source as well as for the extended source. Calculate the codeword mean lengths.

2.10 An efficient coding method for contours of objects is wanted. A contour consists of a chain of *contour segments*. A contour segment is the one pixel long border between two adjacent pixels. To the right a contour made out of 22 segments is shown.



A natural way of coding a contour is to define an alphabet $\mathcal{A} = \{L, S, R\}$ where L stands for "turn left", S for "straight ahead" and R for "turn right". Note that by this choice of alphabet one has taken into account the memory of the "contour source" in that the possibility to turn back into itself have no corresponding symbol.

However, a more efficient scheme is wanted and a coding method is suggested that also takes into account the segment two steps behind. From a large amount of test data the following distributions for the four principal ways of attaching three sements into a contour was derived.

Contour				
Frequency	0.24	2×0.20	2×0.16	0.04

Reflections (as well as rotations) of the same principal contour have equal probability. Take for instance the second type of contour; it generates the symbol sequence SR and RS or LS and SL depending on from which end you start and they have the same probability.

a) Construct a Huffman-code (blocklength=1) for the contour-source (with alphabet $\mathcal{A}$). Calculate the codeword mean length.

b) The probabilities above implicitely defines a Markov model. What is the lowest mean rate this model can be coded at?

2.11 A memoryless source has alphabet $\mathcal{A} = \{1, 2, 3\}$ and $P = \{0.6, 0.2, 0.2\}$. The sequence 3112 is to be arithmetically encoded. Assume infinite precision in the registers and that they are reset before coding of each block of length n. Determine the code for,

a) $n = 2$, ie code the two first symbols into one codeword and the next two symbols into another codeword.

b) $n = 4$, ie code all four symbols into one codeword.

2.12 A system for transmission of simple colour pictures can transmit white, black, red, blue, green and yellow. The source is modelled as an order-1 Markov process with the following transition probabilities:

State	the prob. for next state					
	white	black	red	blue	green	yellow
white	0.94	0.02	0.01	0.01	0.01	0.01
black	0.05	0.50	0.15	0.10	0.15	0.05
red	0.03	0.02	0.90	0.01	0.01	0.03
blue	0.02	0.02	0.02	0.90	0.03	0.01
green	0.02	0.02	0.01	0.03	0.90	0.02
yellow	0.03	0.01	0.03	0.01	0.03	0.89

We use arithmetic coding to code finite sequences of pixels. (I.e., we flush the coder from time to time.) An in-sequence will be mapped into a unique inverval in [0 1). In each step the encoder uses the conditional probabilities given the previous pixel. The colours are ordered in the encoder according to the table with white corresponding to the interval closest to 0.

a) An in-sequence begins red,white,white. Derive the interval corresponding to this sequence when the last pixel in the previous sequence was red!

b) We are waiting for a new sequence. The bitstream 110100111100001100... arrives. What are the two first colours of this sequence? The last pixel in the previous sequence was black.

2.13 A stationary binary source with memory is to be encoded. The following pair probabilities $P(x_i, x_{i+1})$ have been estimated with great accuracy and can be assumed to be the true ones:

$$P(0,0) = 1/7$$
$$P(0,1) = 1/7$$
$$P(1,0) = 1/7$$
$$P(1,1) = 4/7$$

Derive a codeword for the sequence $1, 1, 0, 1$ using arithmetic coding. The code should be based on the conditional probabilities. The previous symbol is assumed to be 1. Also assume that we have infinite precision and that all four symbols are coded into one codeword.

2.14 We can use *universal source coding* when the source statistics is unknown. A binary memory-less source $P = \{p, 1-p\}$ is coded in the following way: Pick a block of length n bits and count the number of ones, w. Encode the number w as a binary number and then code the index of the block in an enumeration of all possible blocks of length n containing exactly w ones.

a) How many bits are needed to code the sequence x=000000000000100?

b) How many bits are needed to code a block of w ones and $n - w$ zeros?

c) What is the rate when $n \to \infty$?

2.15 A source has the alphabet $\mathcal{A} = \{a, b\}$. Code the sequence

$$ababbaaababbbbaaabaaaaaababba...$$

using the LZ77 algorithm, with a search buffer of length 15 and a look-ahead buffer of length 15.

You can check your solution by decoding your codewords without looking at the original sequence.

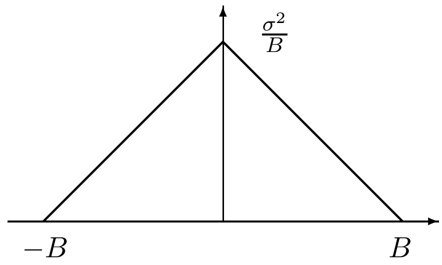
2.16 Code the sequence in problem 2.15 with LZ78.

You can check your solution by decoding your codewords without looking at the original sequence.

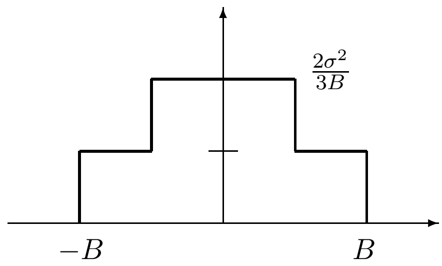
2.17 Code the sequence in problem 2.15 with LZW

3 Rate-Distortion

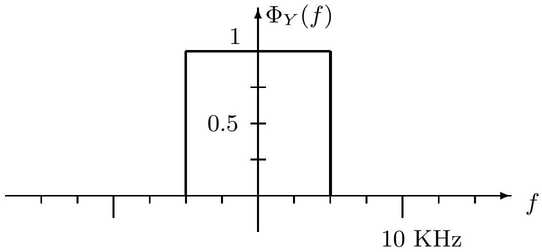
3.1 Determine the rate-distortion function for a gaussian process with p.s.d.,

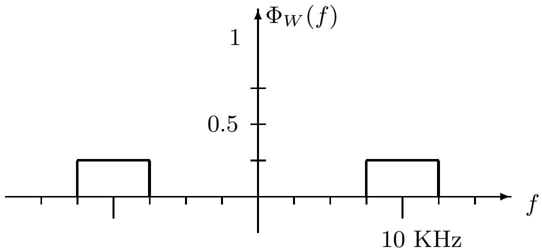


3.2 Determine the rate-distortion function $R(D)$ for a gaussian process with p.s.d.,



3.3 A signal is modelled as a stochastic process $X(t)$ which can be written as the sum of two independent stationary gaussian processes $Y(t)$ and $W(t)$ with power spectras depicted below.





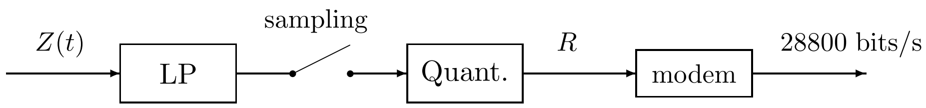
The independence implies $\Phi_X(f) = \Phi_Y(f) + \Phi_W(f)$.

Assume that we have an encoder that is theoretically optimal. Derive the largest possible signal-to-noise ratio (expressed in dB) for $X(t)$ for which $W(t)$ can be completely ignored when we encode $X(t)$ with the optimal encoder. In other words, for which SNR-range of $X(t)$ is it enough to just encode $Y(t)$?

3.4 We have a video signal that we want to digitize and transfer via an ordinary modem with the data rate 28800 bits/s. The video signal is modelled as an analog gaussian process $Z(t)$ with p.s.d.

$$\Phi_Z(f) = \frac{1}{2A} e^{-\frac{|f|}{A}}$$

where $A = 300$



The signal $Z(t)$ is band limited with an ideal low pass filter with bandwidth $\frac{f_s}{2}$ and then sampled with sampling frequency f_s. The sampled signal is Max quantized to R bits/sample. The receiver (not included in the figure) reconstructs the quantized samples and creates a time continuous signal $\hat{Z}(t)$ using pulse code modulation.

a) How shall the sampling frequency f_s and the bit rate R be chosen so that the distortion between $Z(t)$ and $\hat{Z}(t)$ is minimized? What is the resulting signal-to-noise ratio? All assumptions and simplifications must be motivated.

b) What is theoretically the highest signal-to-noise ratio that we can achieve if we code $Z(t)$ to the rate 28800 bits/s?

4 Quantization

4.1 Let X be a continuous uniformly distributed random variable over $(-T\,,\,T)$. Determine the reconstruction points for a three-level Max-quantizer and calculate the expected distortion.

4.2 Let X be a continuous random variable with a laplacian p.d.f.:

$$f(x) = \frac{1}{2a} e^{-\frac{|x|}{a}}$$

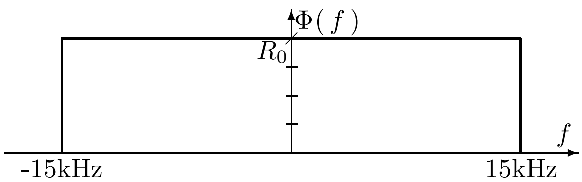
a) Determine the reconstruction points and regions for a two-level Lloyd-Max-quantizer and calculate the expected distortion,

$$D = \mathrm{E}\left\{ (X - \hat{X})^2 \right\}$$

Tips: Simplify by assuming certain symmetries.

b) Like problem a, but using three reconstruction levels.

4.3 A time-continuos electrical signal $X(t)$ is to be sampled and scalar quantized. $X(t)$ has p.s.d.,



and a uniformly distributed amplitude. Design a coder so that the signal-to-nosie ratio is at least 30 dB and the bit-rate is kept low. State any assumptions made.

4.4 A zero-mean gaussian stochastic variable with variance $\sigma^2 = 5$ is to be quantized to two levels.

a) What is the distortion? Use the formula collection.

b) What is the entropy of the output?

c) Compare with $R(D)$ at the same distortion.

4.5 A gaussian stochastic variable with mean 0 and variance σ^2 is represented with two bits. Which values can the distortion take?

4.6 A zero-mean gaussian stochastic variable X is uniformly quantized (fine quantization assumed) with step Δ and then entropy coded. Express the obtained rate R_U as a function of the distortion D. Compare with the rate-distortion function.

4.7 A laplacian variable X with variance σ^2 is uniformly quantized and entropy coded. Express the distortion D as a function of the rate R and σ^2 when fine quantization is assumed.

The p.d.f. is:

$$f_X(x) \;=\; \frac{1}{\sqrt{2}\sigma}e^{-\frac{\sqrt{2}|x|}{\sigma}}$$

4.8 A random variable X with a triangular distribution is uniformly quantized and entropy coded. Express the distortion D as a function of the rate R and the variance σ^2 when fine quantization is assumed.

The probability density function is:

$$f_X(x) = \begin{cases} -\frac{1}{a^2}x + \frac{1}{a} & ;\ 0 \le x \le a \\ \frac{1}{a^2}x + \frac{1}{a} & ;\ -a \le x \le 0 \\ 0 & ;\ \text{otherwise} \end{cases}$$

4.9 Calculate the mean distortion as a function of the rate when using *source adapted quantization* (ie fine Lloyd-Max quantization) of a Laplacian stochastic variable with zero mean and variance σ^2.

The Laplacian probability density function:

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

4.10 On an ordinary audio CD, sound is represented with 16 bits/sample. The quantizer that is used is uniform in the interval $[-a, a]$, where a has been chosen such that $a = 6\sigma$. No entropy coding is done.

If we assume that the audio samples come from a gaussian random process with the variance σ^2 and mean 0, how much could be gained (in SNR) if we use Lloyd-Max quantization instead of uniform quantization?

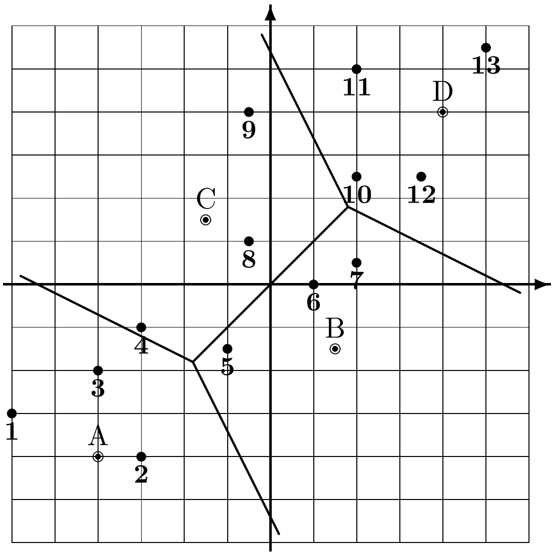
4.11 This problem exemplifies the main steps of the *LBG*-algorithm for training of a so called *codebook* for vector-quantization.

Assume we want to jointly quantize two adjacent pixels at a time to one of four reconstruction points A, B, C or D in $\mathcal{R}^2$. We pick 13 sampled pairs (see figure) as test-data.

a) Calculate the MSE of the given training data set with respect to reconstruction points and regions in the figure.

b) Suggest how to modify the position of the reconstruction points to decrease the MSE to a minimum. Verify by calculating the new MSE!

c) Consider these new reconstruction points and suggest how the reconstruction regions should be modified to minimize the MSE!



Point	X	Y
1	-60	-30
2	-30	-40
3	-40	-20
4	-30	-10
5	-10	-15
6	10	0
7	20	5
8	-5	10
9	-5	40
10	20	25
11	20	50
12	35	25
13	50	55
A	-40	-40
B	15	-15
C	-15	15
D	40	40

4.12 A 1024×1024 image is divided into blocks which are then *vector quantized* so that the resulting rate is 1.5 bits/pixel.

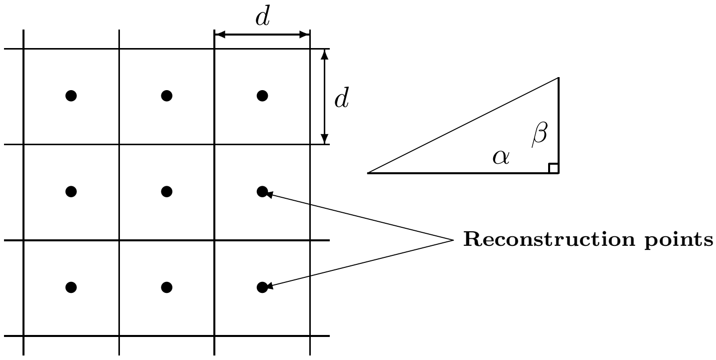
Assume that the quantizer uses *full sequential search* to find the closest match in the codebook for each block. Furthermore, assume that the time it takes to compute the distance between two blocks is 1 ns/dimension.

a) How long will it take to code the image if blocks of size 8×8 are used?

b) What is the maximum number of pixels/block that can be used if we want to code the picture in under one second?

c) What is the maximum number of pixels/block that can be used if we want to code the picture in under one minute?

4.13 A stationary signal is uniformly quantized (Fine quantization is assumed). We can view this as a special case of two-dimensional vector quantization where the reconstruction points form a regular square lattice:



a) How many dB do we gain if we move every second row of reconstruction points $0.5d$ to the left and quantizes in two dimensions (a so called *quincunx* lattice)? New optimal decision-boundaries are assumed.

   Use the following pre-calculated integral over the triangle in the figure above,

$$\int_{x=0}^{\alpha} \int_{y=0}^{\beta x/\alpha} x^2 + y^2 \, dx \, dy = \frac{\alpha\beta}{12}(3\alpha^2 + \beta^2)$$

b) How many dB do we gain if we instead place the reconstruction points in a uniform triangular mesh so that the reconstruction regions form a hexagonal pattern and the bit-rate is unchanged?

   Tips: The reconstruction regions can be assumed to be of the same area as before if the bit-rate is to be unchanged. This follows from the fine quantization assumption.

5 Linear Predictive Coding

5.1 A time-discrete stationary gaussian signal X_i with mean 0 has the following a.c.f. in the points 0, 1 and 2: $R_{XX}(0) = 1.0,\ R_{XX}(1) = 0.9,\ R_{XX}(2) = 0.7$

a) Determine the linear predictor of order 1 that minimizes the prediction error variance. Calculate the prediction error variance.

b) Determine the linear predictor of order 2 that minimizes the prediction error variance. Calculate the prediction error variance.

5.2 A time-discrete signal X_i has mean 0 and a.c.f. $R_{XX}(k) = \rho^{|k|}, \ |\rho| < 1$.

a) Determine the linear predictor of order 1 that minimizes the prediction error variance. Calculate the prediction error variance.

b) Determine the linear predictor of order 2 that minimizes the prediction error variance. Calculate the prediction error variance.

5.3 A time-discrete signal X_i with mean 0 and an a.c.f. satisfying $R_{XX}(1) = \rho \cdot R_{XX}(0)$ is to be coded with a linear order-1 predictor $p_i = a \cdot \hat{X}_{i-1}$. The prediction error is uniformly quantized with stepsize Δ.

a) Determine the linear predictor which minimizes the variance of the prediction error when the quantizer noise is taken into account. (Hint: Assume that $\hat{X}_i = X_i + \varepsilon_i$ where ε_i is the quantization error, and that the quantization error is uncorrelated with the signal.)

b) What will the variance of the prediction error be?

5.4 An audio mono signal is modelled as stationary zero mean time discrete gaussian process X_n. From a large set of data we have estimated the following auto correlation function $R_{XX}(k) = E\{X_n X_{n+k}\}$.

$$R_{XX}(0) = 2.32, \quad R_{XX}(1) = 1.17, \quad R_{XX}(2) = -0.86, \quad R_{XX}(3) = -1.78$$

Design a linear predictive coder for the audio signal that gives a SNR of at least 37 dB at the data rate of 6 bits/sample.

5.5 An image is modelled as a stationary twodimensional normally distributed process $X_{i,j}$ (i and j are coordinates in the image) with the following statistics

$$E\{X_{i,j}\} = 0$$

$$E\{X_{i,j} \cdot X_{k,l}\} = 0.92^{|i-k|} \cdot 0.95^{|j-l|}$$

Construct a linear predictive coder that codes the image with a rate of at most 6 bits/pixel and gives a signal to noise ratio higher than 45 dB.

6 Transform Coding

6.1 A zero-mean time discrete gaussian process X_n has the following estimated auto correlation function $R_{XX}(k) = E\{X_n \cdot X_{n+k}\}$:

$$R_{XX}(0) = 4.63, \quad R_{XX}(1) = 2.34, \quad R_{XX}(2) = -1.72, \quad R_{XX}(3) = -3.55$$

We want to code the signal with a 4-dimensional Hadamard transform and Max quantize it to give an average rate of 1 bit/sample.

Calculate the resulting SNR and compare it to the SNR we would get if we quantized the signal directly without first transforming.

6.2 An audio signal is modelled as a time discrete stationary gaussian process X_n with auto correlation function $R_{XX}(k)$ and mean 0.

$$R_{XX}(k) = E\{X_n \cdot X_{n+k}\} = 0.91^{|k|}$$

The signal is transform coded using a 3 point DCT. The transform components are quantized uniformly and then coded by a memoryless arithmetic coder. The stepsizes of the three quantizers are chosen so that the average rate is 4.5 bits/sample and the average distortion is minimzed.

What is the resulting rate for each of the three transform components?

What is the resulting signal to noise ratio?

What signal to noise ratio would we get if we used a KLT instead of a DCT?

6.3 An image is modelled as a stationary twodimensional zero mean normally distributed process $X_{i,j}$ (i and j are coordinates in the image). From a large set of data, the auto correlation function $R_{XX}(k,l) = E\{X_{i,j} \cdot X_{i+k,j+l}\}$ has been estimated as

$$R_{XX}(0,0) = 1.70, \quad R_{XX}(0,1) = 1.58$$

$$R_{XX}(1,0) = 1.54, \quad R_{XX}(1,1) = R_{XX}(1,-1) = 1.52$$

We want to code the image by taking small blocks of size 2×2 pixels and perform a separable Hadamard transform of the blocks.

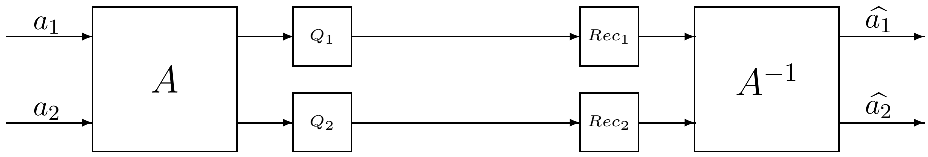
We want to code the blocks using Lloyd-Max quantization, giving an average rate of 1.5 bits/pixel.

What is the bit allocation that minimizes the distortion?

What is the resulting signal-to-noise ratio (in dB) of the coder?

6.4 In the system below, A is a matrix of the form

$$A = \begin{bmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{bmatrix}$$



The distortion of the system is

$$D \triangleq \frac{1}{2}\Big[\mathrm{E}\left\{\,(\hat{a_1} - a_1)^2\,\right\} + \mathrm{E}\left\{\,(\hat{a_2} - a_2)^2\,\right\}\Big]$$

and the average data rate is $R = \dfrac{1}{2}(R_1 + R_2)$
where R_1 and R_2 are the rates of quantizers Q_1 and Q_2 respectively.

Find the optimum allocation of bits (how to choose R_1 and R_2) and the optimal transform (choice of φ). Also calculate the minimal distortion when entropy coding is not used. R can be assumed to be large. The signals a_1 and a_2 are jointly gaussian with

$$\begin{aligned} \mathrm{E}\{\,a_1\,\} &= \mathrm{E}\{\,a_2\,\} = 0 \\ \mathrm{E}\{\,a_1^2\,\} &= \mathrm{E}\{\,a_2^2\,\} = 1 \\ \mathrm{E}\{\,a_1 a_2\,\} &= \rho \end{aligned}$$

and

$$\left|\,\log\frac{1+\rho}{1-\rho}\,\right| \le 4R$$

6.5 A one-dimensional stationary gaussian process X_n is transform coded with a two-point Karhunen-Loève transform. The resulting variances of the two transform components are

$$\sigma_0^2 = 5.85 \;\;;\;\; \sigma_1^2 = 0.15$$

a) The transform components are quantized uniformly and then entropy coded so that the resulting average rate is 5 bits/sample.

   Calculate the resulting signal-to-noise ratio (in dB), assuming that the bits are distributed to minimize the average distortion.

b) We now want to apply predictive coding to X_n instead. Construct an optimal (given the information you have) linear predictor for X_n and quantize the prediction error so that the resulting bit-rate is the same as in a) above. What signal-to-noise ratio is obtained?

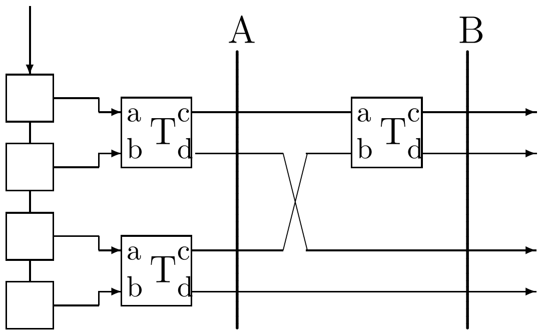
6.6 A time-discrete signal is modelled as a stationary gaussian stochastic process X_n with zero mean and a.c.f.

$$R_{XX}(k) = 0.9^{|k|}$$

The signal is to be transform coded at 2 bits/sample. At one's disposition are "transform units" T which as input take two in-samples (a, b) and produces two transform components according to

$$\begin{cases} c &= \frac{1}{\sqrt{2}}(a+b) \\ d &= \frac{1}{\sqrt{2}}(a-b) \end{cases}$$

Two ways of combining the transform units are to be studied. In both cases four consecutive samples are read in and four transform components are generated according to the figure below. How much do the signal-to-ratio (in dB) increase by using the components at B instead of A? The same Lloyd-Max quantization and bit allocation strategy is assumed for both cases.

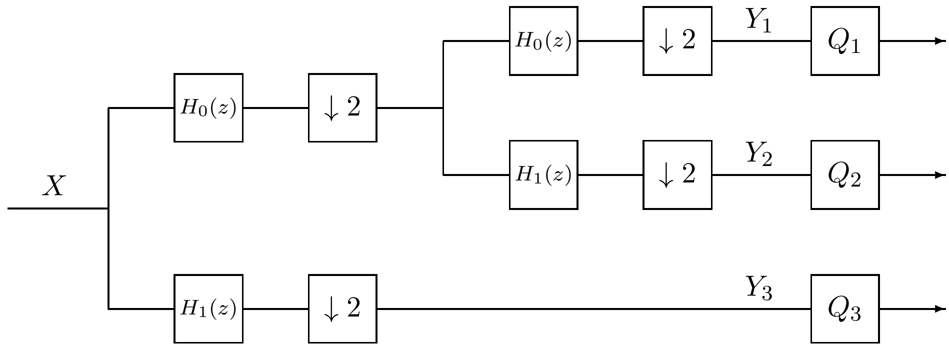


6.7 An audio signal is modelled as a stationary, one-dimensional gaussian process X_n,

$$\mathrm{E}\{X_n\} = 0$$

$$\mathrm{E}\{X_n X_m\} = 2 \cdot 0.85^{|n-m|}$$

We now want to subband-split the audio signal and Lloyd-Max quantize so that the resulting rate is 2 bits/sample.



The filters which are used are $H_0(z) = \frac{1}{\sqrt{2}}(1 + z^{-1})$ and $H_1(z) = \frac{1}{\sqrt{2}}(1 - z^{-1})$, i.e., the normalized sum and difference respectively. The symbol "$\downarrow 2$" stands for down-sampling, i.e., every second sample is thrown.

The filters H_0 och H_1 are orthogonal, normalized and the input signal can be perfectly reconstructed from the down-sampled output.

a) Allocate bits to the quantizers Q_1, Q_2 and Q_3 so that the rate requirement is satisfied and the distortion is minimized!

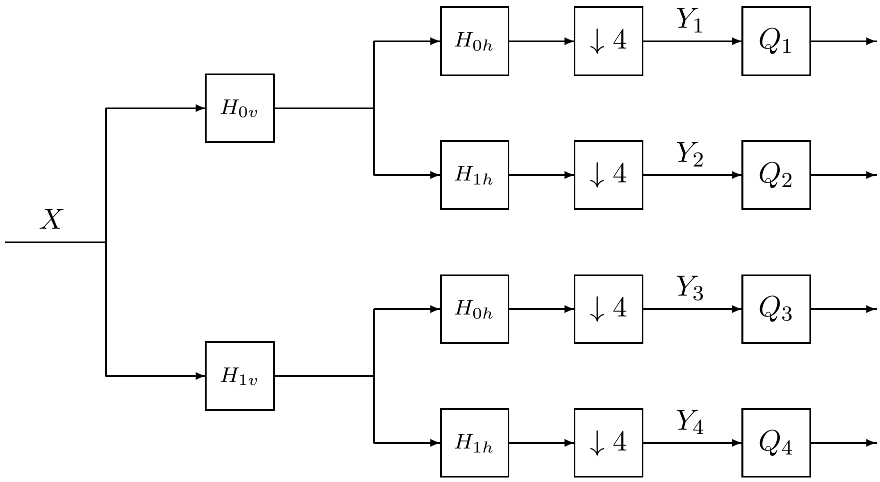
b) Calculate the expected distortion and compare with the distortion if we used Lloyd-Max quantization directly on the signal!

6.8 An image is modelled as an twodimensional gaussian process $X_{i,j}$ (i and j are coordinates in the image) with the statistics

$$E\{X_{i,j}\} = 0$$

$$E\{X_{i,j}X_{k,l}\} = 0.9^{\sqrt{(i-k)^2+(j-l)^2/2}}$$

We want to code the image with a two-dimensional subband coder and Max quantize to get an average rate of 2 bits/pixel.



The signal is first filtered vertically with the filters H_{0v} (lowpass) and H_{1v} (highpass) and thereafter horisontally with the filters H_{0h} (lowpass) och H_{1h} (highpass). The filters are simple sum and difference filters, i.e.,

$$h_{0v}[k,l] = (\delta[k,l] + \delta[k,l+1])/\sqrt{2}$$
$$h_{1v}[k,l] = (\delta[k,l] - \delta[k,l+1])/\sqrt{2}$$
$$h_{0h}[k,l] = (\delta[k,l] + \delta[k+1,l])/\sqrt{2}$$
$$h_{1h}[k,l] = (\delta[k,l] - \delta[k+1,l])/\sqrt{2}$$

The symbol "$\downarrow 4$" means down-sampling by a factor of 4, i.e., only 1/4 of the samples are kept after filtering.

Allocate bits to the four quantizers $Q_1, \ldots, Q_4$ so that the average distortion is minimized. Calculate the resulting distortion.

Part II

Solutions

1.1 $H(X) = -\sum_{i=1}^{4} P_X(i) \cdot \log P_X(i) = -4 \cdot \frac{1}{4} \cdot \log \frac{1}{4} = 2$

1.2 $H(Y) = -\sum_{i=1}^{4} P_Y(i) \cdot \log P_Y(i) = -\frac{1}{2} \cdot \log \frac{1}{2} - \frac{1}{4} \cdot \log \frac{1}{4} - 2 \cdot \frac{1}{8} \cdot \log \frac{1}{8} = 1.75$

1.3 a) Independence gives $P_{XY}(x, y) = P_X(x) \cdot P_Y(y)$.

P_{XY}	y			
	1	2	3	4
1	0.125	0.0625	0.03125	0.03125
x 2	0.125	0.0625	0.03125	0.03125
3	0.125	0.0625	0.03125	0.03125
4	0.125	0.0625	0.03125	0.03125

b) $H(X, Y) = 3.75$

c)
$$\begin{aligned} H(X,Y) &= -\sum_i \sum_j P_{XY}(i,j) \log P_{XY}(i,j) \\ &= -\sum_i \sum_j P_X(i) P_Y(j) (\log P_X(i) + \log P_Y(j)) \\ &= -\sum_i \left(\sum_j P_Y(j) \right) P_X(i) \log P_X(i) - \sum_j \left(\sum_i P_X(i) \right) P_Y(j) \log P_Y(j) \\ &= H(X) + H(Y) \end{aligned}$$

d) Consider $(X_1, \ldots, X_{n-1})$ as separate random variable and construct an induction proof.

1.4 a) $P_Z = \{\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\}$ gives maximal entropy.

b) $P_Z = \{1, 0, 0, 0\}$ gives $H(Z) = 0$. Since entropy is non-negative this is minimal. The solution is not unique.

1.5 a) Use $\displaystyle\sum_{i=0}^{\infty} q^i = \frac{1}{1-q}, \quad |q| < 1$.

b) $\displaystyle H(U) = \frac{H_b(q)}{1-q} = \frac{-q \log q - (1-q) \log(1-q)}{1-q}$

Use $\displaystyle\sum_{i=0}^{\infty} iq^i = \frac{q}{(1-q)^2}, \quad |q|<1.$

c) $E\{U\} = \dfrac{q}{1-q}$

1.6 $$-\ln 2\ I(X;Y) \ \triangleq\ \ln 2\Big(H(X\mid Y) - H(X)\Big) \ =\ \ln 2\Big(H(X,Y) - H(X) - H(Y)\Big)$$
$$=\ \mathrm{E}\left\{\ln\frac{P(X)P(Y)}{P(X,Y)}\right\} \ \le\ \mathrm{E}\left\{\frac{P(X)P(Y)}{P(X,Y)}\right\} - 1 \ =\ 0$$
since $\mathrm{E}\left\{\dfrac{P(X)P(Y)}{P(X,Y)}\right\} \ \triangleq\ \displaystyle\sum_{x,y} P(x,y)\frac{P(x)P(y)}{P(x,y)} = 1.$

1.7 a) $$H_\infty(S) \ \triangleq\ \lim_{n\to\infty}\frac{H(S_nS_{n-1}\dots S_0)}{n} \ \overset{(1)}{=}\ \lim_{n\to\infty} H(S_n|S_{n-1}S_{n-2}\dots S_1)$$
$$\overset{(2)}{=}\ \lim_{n\to\infty} H(S_n|S_{n-1}) \ =\ \sum_k P(S_{n-1}=s_k)H(S_n|S_{n-1}=s_k)$$
where (1) is a consequence of the chain-rule and the stationarity and (2) uses the Markov property.

b) $$H(S_0S_1\dots S_n) \ \overset{(1)}{=}\ H(S_0) + H(S_1S_2\dots S_n|S_0) \ \overset{(2)}{=}\ H(S_0) + H(U_1U_2\dots U_n|S_0)$$
$$=\ H(S_0U_1U_2\dots U_n) \ =\ H(U_1U_2\dots U_n) + \underbrace{H(S_0|U_1U_2\dots U_n)}_{(3)}$$
where the chain-rule is used in (1), (2) follows from the unifilarity in that there for large enough n is a 1-1 relationship between the state-sequence and output-sequnce and (3) is limited by $\log r$ where r is the number of states. Divide the above by n and let n approach infinity and the proof is finished.

1.8 a) $H+1$

b) $H+p$

c) $2H$

d) nH

1.9 a) $$H(f(X)\,|\,X) \ =\ /Z=f(X)/ \ =\ H(Z\,|\,X) \ \triangleq$$
$$\triangleq\ -\sum_{b_j\in A_Z}\sum_{a_i\in A_X} \underbrace{P_{XZ}(b_j,a_i)}_{=0,f(a_i)\neq b_j} \log \underbrace{P_{Z|X}(b_j\,|\,a_i)}_{=1,f(a_i)=b_j}$$
$$=\ 0$$
Note that by convention $0\cdot\log 0 = \lim_{p\to 0} p\log p = 0$.

b) $$H(X,f(X)) \ =\ /\text{chain-rule}/$$
$$=\ H(X) + \underbrace{H(f(X)\,|\,X)}_{=0 \text{ according to a)}}$$

c) $H(X) - H(f(X)) \; = \; /\text{according to b)}/$
$= \; H(X, f(X)) - H(f(X)) \; = \; /\text{chain-rule}/$
$= \; H(X \mid f(X)) \; \geq \; 0$

1.10 According to 1.9, a function of a stochastic variable has equal or less entropy than the variable itself. $\rightarrow$ right inequality.

For the left inequality we let $Z = X + Y$ and show that,

$$\begin{aligned}
H(Z \mid Y) &\overset{\triangle}{=} \\
&= -\sum_z \sum_y P_{ZY}(z,y) \log \frac{P_{ZY}(z,y)}{P_Y(y)} = /x{=}z{-}y/ \\
&= -\sum_x \sum_y P_{XY}(x,y) \log \frac{P_{XY}(x,y)}{P_Y(y)} \\
&\overset{(1)}{=} -\sum_x \sum_y P_X(x) P_Y(y) \log \frac{P_X(x)P_Y(y)}{P_Y(y)} \\
&= -\left(\sum_y P_Y(y)\right)\left(\sum_y P_X(x) \log P_X(x)\right) \; = \; H(X)
\end{aligned}$$

In (1) we use the independence. We can now write,

$$H(X+Y) \; \geq \; H(X+Y \mid Y) \; = \; H(X)$$

See 1.6 for proof that conditioning cannot increase the entropy.

Note that what we have shown is that convolving a distribution with another cannot decrease the entropy!

1.11

$$\begin{aligned}
H(X \mid Y) + H(Y \mid Z) &\overset{(1)}{\geq} \\
&\geq \; H(X \mid YZ) + H(Y \mid Z) \\
&= \; \underbrace{H(X \mid YZ) + H(Y \mid Z) + H(Z)}_{\text{apply chain-rule } = H(XYZ)} - H(Z) \\
&= \; H(XYZ) - H(Z) \\
&\overset{(2)}{=} \; H(X \mid Z) + H(Y \mid XZ) \; \geq \; H(X \mid Z)
\end{aligned}$$

where we in (1) use the fact that conditioning does not increase the entropy and in (2) we again use the chain-rule: $H(XYZ) = H(Z) + H(X \mid Z) + H(Y \mid XZ)$.

1.12 The chain-rule is proved by the following,

$$H(X,Y) \; \overset{\triangle}{=} \; -\mathrm{E}\{\log P_{XY}(X,Y)\} \; = \; \mathrm{E}\{-\log P_{X|Y}(X|Y)p_Y(Y)\}$$
$$= \; \mathrm{E}\{-\log P_{X|Y}(X|Y)\} + \mathrm{E}\{-\log P_Y(Y)\} \; \overset{\triangle}{=} \; H(X \mid Y) \; + \; H(Y)$$

Now let $X = X_{n+1}$ and $Y = (X_1, \ldots, X_n)$.

1.13 a) Let (B_8, B_4, B_2, B_1) be four random variables describing the bits. Then the entropies will be

$$H(B_8) = -\frac{1}{3} \cdot \log \frac{1}{3} - \frac{2}{3} \cdot \log \frac{2}{3} \approx 0.9183$$

$$H(B_4) = -\frac{1}{3} \cdot \log \frac{1}{3} - \frac{2}{3} \cdot \log \frac{2}{3} \approx 0.9183$$
$$H(B_2) = -\frac{1}{2} \cdot \log \frac{1}{2} - \frac{1}{2} \cdot \log \frac{1}{2} = 1$$
$$H(B_1) = -\frac{1}{2} \cdot \log \frac{1}{2} - \frac{1}{2} \cdot \log \frac{1}{2} = 1$$

b)
$$H(X) = \log 12 \approx 3.5850$$

Note that this is smaller than the sum of the entropies for the different bits, since the bits aren't independent of each other.

1.14 The transition matrix P of the source is

$$P = \begin{pmatrix} 0.8 & 0.1 & 0.1 \\ 0.5 & 0.5 & 0 \\ 0.5 & 0 & 0.5 \end{pmatrix}$$

The stationary distribution $\bar{w} = (w_0,\ w_1,\ w_2)$ is given by the equation system $\bar{w} = \bar{w} \cdot P$. Replace one of the equations (any one will do) with the equation $w_0 + w_1 + w_2 = 1$ and solve the system. This gives the solution

$$\bar{w} = \frac{1}{7}(5,\ 1,\ 1) \approx (0.714,\ 0.143,\ 0.143)$$

1.15 a) $H(X_i) = -\frac{5}{7} \cdot \log \frac{5}{7} - 2 \cdot \frac{1}{7} \cdot \log \frac{1}{7} \approx 1.1488$ [bit/sym].

b) The block probabilities are given by

symbol pair	probability
00	$5/7 \cdot 0.8 = 8/14$
01	$5/7 \cdot 0.1 = 1/14$
02	$5/7 \cdot 0.1 = 1/14$
10	$1/7 \cdot 0.5 = 1/14$
11	$1/7 \cdot 0.5 = 1/14$
12	0
20	$1/7 \cdot 0.5 = 1/14$
21	0
22	$1/7 \cdot 0.5 = 1/14$

$H(X_i, X_{i+1}) = -\frac{8}{14} \cdot \log \frac{8}{14} - 6 \cdot \frac{1}{14} \cdot \log \frac{1}{14} \approx 2.0931$ [bits/pair]. ($\rightarrow$ 1.0465 [bits/symbol]). This is less than two times the memoryless entropy.

c) According to the chain rule:
$H(X_{i+1} \mid X_i) = H(X_i, X_{i+1}) - H(X_i) \approx 0.9442$

1.16 a) $r_0 = \dfrac{1}{1-q_0}$ and $r_1 = \dfrac{1}{1-q_1}$.

b) $H_0 = r_0 H_b(q_0)$ and $H_1 = r_1 H_b(q_1)$.

c) $H \leq \dfrac{H_0 + H_1}{r_0 + r_1}$

A Markov source of order 1 with $\mathbf{P} = \begin{pmatrix} q_0 & 1-q_0 \\ 1-q_1 & q_1 \end{pmatrix}$ could be the source that generated these sequences.

There can be no other source generating these run-lengths and at the same time have a lower source-symbol entropy-rate. One way to realize this is to think in terms of an optimal entropy-code of the run-lengths. This simply is the best we can do, since there are no more dependancies to exploit. So the bound is the real entropy-rate.

2.1 Yes, since Kraft's inequality is fulfilled:

$$\sum_{i=1}^{8} 2^{-l_i} = \frac{53}{64} < 1$$

You can of course show this by constructing a prefix code too.

2.2 Let the codeword lengths l_i be sorted,

$$l_1 \leq l_2 \leq \ldots \leq l_m$$

Create a full binary tree of depth l_m. Follow the following steps to prune the tree to get the prefix code:

1. Set $i = 1$.
2. Choose an arbitrary, unused node at depth l_i and cut away the children. If no such node exist $\rightarrow$ STOP. (This algorithm doesn't work.)
3. If $i = m \rightarrow$ STOP, otherwise $i = i + 1$, go to 2.

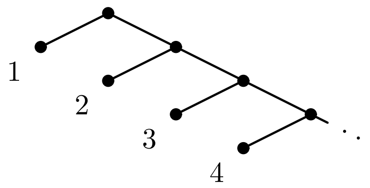
We have to show that the algorithm won't stop in 2 above. This is done with induction.

Clearly there will be a free node at depth l_1 at the start. Assume we have assigned all the lengths up to but not including l_j. At this stage there has to be nodes left at the maximum depth l_m since the total amount of nodes cut at this depth so far is:

$$2^{l_m} - \left(2^{l_m - l_1} + 2^{l_m - l_2} + \ldots + 2^{l_m - l_{j-1}}\right) = 2^{l_m}\left(1 - \sum_{i=1}^{j-1} 2^{-l_j}\right) > 0$$

where the inequality comes from the fact that Kraft's inequality was satisfied. Since there are nodes at depth l_m there surely are nodes at depth l_j to pick. Also note that since the codeword lengths were sorted there is no risk of pruning away an old codeword!

2.3 Since the probabilities are dyadic, it is possible to construct a code with the codeword lengths $l_i = \log P(i) = i$. This code will have a rate that is equal to the entropy of the source and will thus be optimal.



$$R = 2 \text{ [bits/symbol]}.$$

2.4 a) $-0.6 \cdot \log 0.6 - 0.3 \cdot \log 0.3 - 0.1 \cdot \log 0.1 \approx 1.2955$

b) Codeword lengths and an example of codewords:

symbol	length	codeword
x	1	0
y	2	10
z	2	11

The mean codeword length is 1.4 bits/codeword and the average rate is 1.4 bits/symbol.

c) Codeword lengths (not unique for this distribution) and an example of codewords:

symbols	length	codeword
xx	1	0
xy	3	100
xz	4	1100
yx	3	101
yy	4	1110
yz	5	11110
zx	4	1101
zy	6	111110
zz	6	111111

The mean codeword length is 2.67 bits/codeword and the average rate is 1.335 bits/symbol.

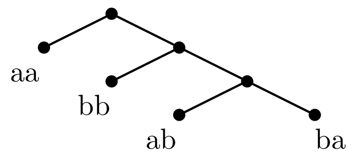
2.5 The stationary distribution of the source is $P(a) = P(b) = 0.5$. This can easily be seen since the source is symmetric. You can of course also find this by solving the equation system

$$\begin{cases} P(a) = 0.9 \cdot P(a) + 0.1 \cdot P(b) \\ P(a) + P(b) = 1 \end{cases}$$

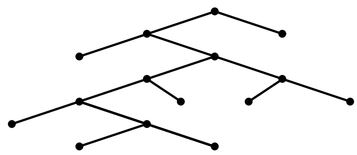
The probabilities of pairs of symbols is given by $P(x_i, x_{i+1}) = P(x_i) \cdot P(x_{i+1}|x_i)$, which gives the probabilities $\{0.45, 0.05, 0.05, 0.45\}$

The probabilities for triples is given by $P(x_i, x_{i+1}, x_{i+2}) = P(x_i) \cdot P(x_{i+1}|x_i) \cdot P(x_{i+2}|x_{i+1})$, which gives the probabilities {0.405, 0.045, 0.005, 0.045, 0.045, 0.005, 0.045, 0.405}

The code trees for the two codes look like



$\bar{l} = 1.65 \Rightarrow R = \frac{\bar{l}}{2} = 0.825$



$\bar{l} = 2.04 \Rightarrow R = \frac{\bar{l}}{3} = 0.68$

It is better to code three symbols with each codeword.

2.6 a) $P(r) = p^{r-1} \cdot (1-p)$

b) $\bar{r} = \sum_{r=1}^{\infty} r \cdot (1-p) \cdot p^{r-1} = \frac{1}{1-p} = \frac{1}{0.1} = 10$ [symbols/run]

c)
$$\begin{aligned} H(Y) &= -\sum_{r=1}^{\infty}(1-p) \cdot p^{r-1} \cdot \log((1-p) \cdot p^{r-1}) \\ &= -(1-p) \cdot \log(1-p) \cdot \sum_{r=1}^{\infty} p^{r-1} - (1-p) \cdot \log p \cdot \sum_{r=1}^{\infty}(r-1) \cdot p^{r-1} \\ &= -(1-p) \cdot \frac{1}{1-p} \cdot \log(1-p) - (1-p) \cdot \frac{p}{(1-p)^2} \cdot \log p \\ &= \frac{H_b(p)}{1-p} \approx 4.690 \quad \text{[bits/run]} \end{aligned}$$

d) $\frac{H(Y)}{\bar{r}} = H_b(p) \approx 0.4690$ [bits/symbol]
We would of course have gotten the same answer by calculating the entropy rate of the original source directly.

2.7 a) $\bar{l} = \mathrm{E}\{\,\text{codeword length/run}\,\} =$

$$4\sum_{r=1}^{15}(1-p)p^{r-1}+8\sum_{r=16}^{30}(1-p)p^{r-1}+12\sum_{r=31}^{45}(1-p)p^{r-1}+\ldots=$$
$$4\sum_{r=1}^{\infty}(1-p)p^{r-1}+4p^{15}\sum_{r=1}^{\infty}(1-p)p^{r-1}+4p^{30}\sum_{r=1}^{\infty}(1-p)p^{r-1}+\ldots=$$
$$4\sum_{r=1}^{\infty}(1-p)p^{r-1}\cdot\sum_{i=0}^{\infty}p^{15i}=4\cdot 1\cdot\frac{1}{1-p^{15}}\approx 5.0371\ \text{[bits/run]}$$

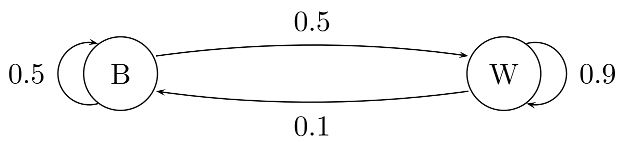
From problem 2.6 we know that $\bar{r}=10$ [symbols/run], thus the rate is $\frac{\bar{l}}{\bar{r}}\approx$ 0.50371 [bits/symbol]. Compare this to the entropy rate of the source.

b) Just as in a) we get

$$\bar{l}=5\cdot 1\cdot\frac{1}{1-p^{31}}\approx 5.1983\ \text{[bits/run]}$$

and the rate $\approx$ 0.51983 [bits/symbol].

2.8 a) The theoretical limit is given by the entropy. The best model we can use, given the information we have, is an order 1 Markov model. In the text area this looks like



The stationary probabilities for this source is $w_B=\frac{1}{6}$ and $w_W=\frac{5}{6}$ and the entropy rate when we're in a text area is therefore

$$H_t=w_B\cdot H_b(0.5)+w_W\cdot H_b(0.9)\approx 0.5575$$

For the picture areas, the entropy rate is similarly

$$H_p\approx 0.7857$$

The total entropy rate for the source is therefore

$$H=\frac{4}{5}H_t+\frac{1}{5}H_p\approx\underline{0.60313}$$

This is the best estimate we can make, the true entropy rate of the source might be lower than this number, if the memory is longer than one symbol as we assumed.

b) The wanted bitrate can be achieved by coding blocks of 3 symbols. The probabilities of the 8 different blocks are given by

$$P(x_i,x_{i+1},x_{i+2})=P(x_i)\cdot P(x_{i+1}|x_i)\cdot P(x_{i+2}|x_{i+1})$$

where the first probability on the right hand side is the stationary probabilities calculated in a), and de conditioned probabilities are the ones given in the problem. The eight probabilities are

$$\boldsymbol{P} = \frac{1}{120}\{1 \quad 5 \quad 5 \quad 5 \quad 5 \quad 9 \quad 9 \quad 81\}$$

The Huffman code is constructed using the Huffman algorithm, giving an average codeword length of <u>0.625</u> bits/symbol.

2.9 a) When allowing codeword lengths to grow it is always possible to achieve a codeword mean length arbitrary close to the entropy rate of the source. The entropy rate of a Markov source is given by the *conditional entropy*,

$$H(S_{n+1}|S_n) = w_A H(S_{n+1}|S_n = A) + w_B H(S_{n+1}|S_n = B) + w_C H(S_{n+1}|S_n = C) + w_D H(S_{n+1}|S_n = D)$$

where w_A etc. denote the stationary probabilities of the states. The stationary probabilities can be calculated by solving the following underdecided equation system together with the fact that the probabilities should add to one:

$$(w_A \; w_B \; w_C \; w_D) \cdot \begin{pmatrix} 0 & 0.5 & 0 & 0.5 \\ 0 & 0.2 & 0 & 0.8 \\ 0.1 & 0 & 0.9 & 0 \\ 0 & 0 & 0.7 & 0.3 \end{pmatrix} = (w_A \; w_B \; w_C \; w_D)$$

$$\Rightarrow \quad (w_A \; w_B \; w_C \; w_D) = \frac{1}{731}(56 \; 35 \; 560 \; 80)$$

and the conditional entropy,

$$H(S_{n+1}|S_n) = \tfrac{1}{731}(56 \cdot H_b(0.5) + 35 \cdot H_b(0.2) + 560 \cdot H_b(0.1) + 80 \cdot H_b(0.3)) \approx 0.5669.$$

b) Optimal FIVO tree-codes can be constructed using the Huffman algorithm. Using the stationary probabilities we get the following suggestion for a code in the memoryless case:

symbol	*probability*	*codeword*	*length*
A	56/731	110	3
B	35/731	111	3
C	560/731	0	1
D	80/731	10	2

yielding rate and mean codeword length $R_1 = \bar{l}_1 = \frac{993}{731} \approx 1.3584$ bits/symbol

When extending the source to two symbols we need the stationary probabilities for the possible sequences of two subsequent symbols. These can easily be calculated using $P(x_i, x_{i+1}) = P(x_i) \cdot P(x_{i+1}|x_i)$ where the conditional probabilities are taken from the graph. The Huffman algorithm can result in the following optimal code:

symbols	*probability*	*codeword*	*length*
AB	28/731	1010	4
AD	28/731	1110	4
BB	7/731	10110	5
BD	28/731	1111	4
CA	56/731	100	3
CC	504/731	0	1
DC	56/731	110	3
DD	24/731	10110	5

With mean codeword length $\bar{l}_2 = \dfrac{1331}{731} \approx 1.8208$ bits/codeword and rate $R_2 = \dfrac{\bar{l}_2}{2} \approx 0.9104$ bits/symbol.

2.10 a) The probabilites for blocks are:

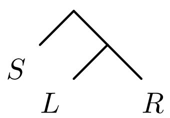
SS	0.24
SR, RS, LS, SL	0.10
LR, RL	0.16
LL, RR	0.02

We want to code one symbol at a time and therefore need the stationary probabilites for each symbol. They can be calculated as below:

$P(L) = P(LL) + P(LS) + P(LR) = 0.28$

$P(S) = P(SL) + P(SS) + P(SR) = 0.44$

$P(R) = P(RL) + P(RS) + P(RR) = 0.28$

The Huffman tree becomes 

.

Codeword mean length: $\bar{l} = 1 + 0.56 = 1.56$

b) The best mean rate for a Markov-1 source is given by the conditional entropy $H(X_i|X_{i-1})$ and can for instance be calculated using the chain-rule:

$H(X_i|X_{i-1}) = H(X_{i-1}, X_i) - H(X_{i-1}) \approx 2.8947 - 1.5496 = 1.3451$

2.11 a) codeword for 31:

The corresponding interval will be $[0.8,\ 0.92)$. At least $\lceil -\log_2(0.92-0.8)\rceil = 4$ bits will be required to specify the interval. The smallest 4 bit binary number larger than 0.8 is $(0.1101)_2 = 0.8125$. To see if 4 bits will be enough, check the number $(0.11011111\ldots)_2 = (0.1110)_2 = 0.875 < 0.92$. Thus, 4 bits will be enough, and the codeword is **1101**.

codeword for 12:

The interval is $[0.36,\ 0.48)$. The same kind of reasoning as above gives that the codeword is **0110**.

b) codeword for 3112:

The interval is $[0.8432,\ 0.8576)$ which leads us to the codeword **1101100**

2.12 a) Assuming that the order of the different colours is the same all the time, the interval will be $[0.05\ ,\ 0.07538)$. (If you changed the colour order, you should at least get the same interval length.)

b) green, green.

2.13 The conditional probabilities can be calculated by using $P(x_i, x_{i+1}) = P(x_i) \cdot P(x_{i+1}|x_i)$ where $P(x_i)$ can befound as the marginal probabilities:

$$P(0) = P(0,0) + P(0,1) = 2/7, \quad P(1) = P(1,0) + P(1,1) = 5/7$$

The interval corresponding to the given sequence is $[0.424,\ 0.488)$ with interval size 0.064.

Next express 0.424 in binary. $0.424 = 0.01101100\ldots$ Since $\lceil -\log 0.064\rceil = 4$ we retrieve our first codeword candidate $(0.)0111 = 0.4375$. When checking the tail we note that $(0.)1000 = 0.5 > 0.488 \rightarrow$ we need one more bit for a correct codeword. Hence the final codeword will be (0.)**01110**

2.14 a) In this case $n = 15$. To code how many ones there are, we need 4 bits (we need to send a number between 0 and 15). To code where the single one is located, we need 4 more bits (we need to code a position between 1 and 15). Thus, in total we need 8 bits to code the given bitstring.

b) w ones can be placed in $\binom{n}{w}$ different ways in a block of length n. Thus we will need $\lceil \log(n+1)\rceil\ +\ \lceil \log \binom{n}{w}\rceil$ bits to code the block.

c) The number of bits/symbol is given by

$$\bar{l} = \frac{1}{n}\sum_{w=0}^{n}\binom{n}{w}p^{n-w}(1-p)^{w}(\lceil \log(n+1)\rceil\ + \lceil \log \binom{n}{w}\rceil) <$$
$$\frac{1}{n}(2 + \log(n+1) + \sum_{w=0}^{n}\binom{n}{w}p^{n-w}(1-p)^{w}\log\binom{n}{w})$$

where we used the fact that $\lceil x \rceil < x + 1$

$$\sum_{w=0}^{n} \binom{n}{w} p^{n-w}(1-p)^w \log \binom{n}{w} =$$

$$\sum_{w=0}^{n} \binom{n}{w} p^{n-w}(1-p)^w \log(\binom{n}{w} p^{n-w}(1-p)^w) - \quad (1)$$

$$\sum_{w=0}^{n} \binom{n}{w} p^{n-w}(1-p)^w \log p^{n-w} - \quad (2)$$

$$\sum_{w=0}^{n} \binom{n}{w} p^{n-w}(1-p)^w \log(1-p)^w \quad (3)$$

(1) is the negative entropy of a binomial distribution, thus it is ≤ 0.

$$(3) = \sum_{w=0}^{n} \binom{n}{w} p^{n-w}(1-p)^w \log(1-p)^w =$$

$$\log(1-p) \sum_{w=0}^{n} \binom{n}{w} p^{n-w}(1-p)^w w =$$

$$n \cdot \log(1-p) \cdot (1-p) \sum_{w=1}^{n} \binom{n-1}{w-1} p^{n-w}(1-p)^{w-1} = n \cdot (1-p) \cdot \log(1-p)$$

where we used the fact that the last sum is just the sum of a binomial distribution, which of course is 1. In the same way we can show that $(2) = n \cdot p \cdot \log p$. Since the source is memoryless, the entropy $H_b(p)$ is a lower bound for $\bar{l}$. Thus we get

$$H_b(p) \leq \bar{l} < \frac{1}{n}(2 + \log(n+1) - n \cdot p \cdot \log p - n \cdot (1-p) \cdot \log(1-p)) =$$

$$H_b(p) + \frac{1}{n}(2 + \log(n+1))$$

So when $n \to \infty$ the rate $\bar{l}$ will approach the entropy of the source, showing that the coding method is universal.

2.15 Assuming that the rightmost position in the search buffer is denoted by 0, we get the codewords:

(offset, length, new symbol)	Binary codeword
(0,0,a)	0000 0000 0
(0,0,b)	0000 0000 1
(1,2,b)	0001 0010 1
(2,1,a)	0010 0001 0
(6,5,b)	0110 0101 1
(8,6,a)	1000 0110 0
(0,4,b)	0000 0100 1
...	...

2.16 The coded sequence of pairs <index, new symbol> is:

$$< 0, a > < 0, b > < 1, b > < 2, a > < 1, a > < 4, b > < 2, b >$$
$$< 4, a > < 3, a > < 5, a > < 5, b > < 3, b > \ \dots$$

If we assume that the dictionary has the size 16, we will need 4+1 bits to code each pair.

After coding, the dictionary looks like:

index	sequence	index	sequence	index	sequence	index	sequence
0	-	4	*ba*	8	*baa*	12	*abb*
1	*a*	5	*aa*	9	*aba*		
2	*b*	6	*bab*	10	*aaa*		
3	*ab*	7	*bb*	11	*aab*		

2.17 The coded sequence of <index> is:

$$< 0 > < 1 > < 2 > < 3 > < 0 > < 2 > < 4 > < 1 >$$
$$< 5 > < 7 > < 6 > < 12 > < 3 > < 9 > \ \dots$$

If we assume that the dictionary has the size 16, we will need 4 bits to code each index.

After coding, the dictionary looks like:

index	sequence	index	sequence	index	sequence	index	sequence
0	*a*	4	*abb*	8	*abbb*	12	*aaa*
1	*b*	5	*baa*	9	*bb*	13	*aaab*
2	*ab*	6	*aa*	10	*baaa*	14	*bab*
3	*ba*	7	*aba*	11	*abaa*	15	*bba*

3.1 $$R(D) = \frac{B}{\ln 2}\left(\ln \frac{1}{1-\sqrt{1-\frac{D}{\sigma^2}}} - \sqrt{1-\frac{D}{\sigma^2}}\right)$$

3.2 $$R(D) = \begin{cases} B \log \frac{2\sigma^2\sqrt{2}}{3D} & ; 0 \leq D \leq \frac{2\sigma^2}{3} \\ \frac{1}{2} B \log \frac{2\sigma^2}{3D-\sigma^2} & ; \frac{2\sigma^2}{3} \leq D \leq \sigma^2 \end{cases}$$

3.3 The theoretical optimum is given by the Rate-Distortion function, $R(D)$ which for a time-continuous gaussian process with spectral density $\Phi(f)$ can be found from

$$\begin{cases} D(\gamma) = \int_{-\infty}^{\infty} \min[\Phi(f), \gamma] df \\ R(\gamma) = \int_{-\infty}^{\infty} \max[0, \frac{1}{2} \log \frac{\Phi(f)}{\gamma}] df \end{cases}$$

$W(t)$ can be ignored for $\gamma \geq \max[\Phi_W(f)] = 0.25$. We get the lowest distortion (and thus the largest SNR) when $\gamma = 0.25$.

$$D(0.25) = \int_{-\infty}^{\infty} \min[\Phi(f), 0.25] df = 2 \cdot 0.25 \cdot 5000 + 0.25 \cdot 10000 = 5000$$

The variance σ_X^2 is given by

$$\sigma_X^2 = \int_{-\infty}^{\infty} \Phi_X(f) df = 2 \cdot 0.25 \cdot 5000 + 1 \cdot 10000 = 12500$$

The resulting SNR is

$$10 \cdot \log_{10} \frac{\sigma_X^2}{D} = 10 \cdot \log_{10} \frac{12500}{5000} \approx \underline{3.98 \text{ [dB]}}$$

We get the lowest SNR when we ignore the signal completely, which gives $D = \sigma_X^2$ and SNR 0 dB.

3.4 a) Assume that the quantization noise and the signal are uncorrelated. The total distortion is the sum of the quantization distortion and the distortion from the LP-filtering prior to sampling

$$D_{\text{LP}} = 2 \int_{f_s/2}^{\infty} \Phi(f) df = \frac{1}{A} \int_{f_s/2}^{\infty} e^{-f/A} df = e^{-f_s/2A}$$

$$D_{\text{quant}} \approx \sigma^2 \cdot \frac{\pi\sqrt{3}}{2} \cdot 2^{-2R}$$

$$\sigma^2 = 2 \int_0^{f_s/2} \Phi(f) df = 1 - e^{-f_s/2A}$$

$$D_{\text{tot}} = D_{\text{LP}} + D_{\text{quant}}$$

$$R_c = R \cdot f_s = 28800 \Rightarrow f_s = \frac{28800}{R} \Rightarrow D_{\text{tot}} \approx e^{-48/R} + (1 - e^{-48/R}) \frac{\pi\sqrt{3}}{2} 2^{-2R}$$

If we restrict ourselves to integer solutions, D_{tot} will be minimal for $R = 6$.

$$R = 6 \Rightarrow D_{\text{tot}} \approx 9.995 \cdot 10^{-4} \Rightarrow f_s = 4800 \text{ [Hz]}$$

$$\text{SNR} = 10 \cdot \log_{10} \frac{\sigma_Z^2}{D_{\text{tot}}} \approx 10 \cdot \log_{10} \frac{1}{9.995 \cdot 10^{-4}} \approx 30.0 \text{ [dB]}$$

b) The theoretical bound is given by the rate-distortion function. Let f_γ be the frequency where $\Phi(f_\gamma) = \gamma$

$$\begin{aligned} R(\gamma) &= 2\int_0^{f_\gamma} \frac{1}{2}\log\frac{\Phi(f)}{\gamma}df \;=\; /\gamma = \frac{1}{2A}e^{-f_\gamma/A}/ \;= \\ &= \int_0^{f_\gamma} \log e^{\frac{f_\gamma - f}{A}} df \;=\; \frac{1}{\ln 2}\frac{1}{A}\int_0^{f_\gamma}(f_\gamma - f)df \;= \\ &= \frac{f_\gamma^2}{2A\ln 2} \;=\; 28800 \end{aligned}$$

$$\Rightarrow f_\gamma \approx 3461 \Rightarrow \gamma \approx 1.6283 \cdot 10^{-8}$$

$$D(\gamma) = 2f_\gamma\gamma + 2\int_{f_\gamma}^{\infty}\Phi(f)df = 2f_\gamma\gamma + e^{-f_\gamma/A} \approx 1.225 \cdot 10^{-4}$$

$$\text{SNR} \approx 10 \cdot \log_{10}\frac{1}{1.225 \cdot 10^{-4}} \approx 39.1 \text{ [dB]}$$

4.1 Three reconstruction points (y_1, y_2 and y_3) and four decision boundaries (b_0, b_1, b_2 and b_3). Since the probability density function is symmetric around the origin, we have $y_1 = -y_3$ and $y_2 = 0$. The optimal decision boundaries should be located halfway between the reconstruction points (except the two end boundaries which should be located at the endpoints of the distribution). We thus get $b_0 = -T$, $b_1 = \frac{y_1+y_2}{2} = -\frac{y_3}{2}$, $b_2 = \frac{y_2+y_3}{2} = \frac{y_3}{2}$ and $b_3 = T$.

What is left is to find y_3. It should be located in the the centroid of its corresponding decision region, ie

$$y_3 = \frac{\int_{b_2}^{b_3} x \cdot f(x)\, dx}{\int_{b_2}^{b_3} f(x)\, dx}$$

We have

$$\int_{b_2}^{b_3} f(x)\, dx = \int_{y_3/2}^{T}\frac{1}{2T}\, dx = \frac{1}{2T}\left(T - \frac{y_3}{2}\right)$$

and

$$\int_{b_2}^{b_3} x \cdot f(x)\, dx = \int_{y_3/2}^{\infty} x \cdot \frac{1}{2T}\, dx = \frac{1}{4T}\left(T^2 - \left(\frac{y_3}{2}\right)^2\right)$$

which gives us

$$y_3 = \frac{T + \frac{y_3}{2}}{2} \quad\Longrightarrow\quad y_3 = \frac{2T}{3}$$

Reconstruction points: $y_1 = -\frac{2T}{3}$, $y_2 = 0$, $y_3 = \frac{2T}{3}$.

Decision boundaries: $b_0 = -T$, $b_1 = -\frac{T}{3}$, $b_2 = \frac{T}{3}$, $b_3 = T$.

Note that the Lloyd-Max quantizer for a uniform distribution is a uniform quantizer.

The distortion is given by

$$
\begin{aligned}
D &= \int_{-T}^{-T/3} (x + \frac{2T}{3})^2 \frac{1}{2T} \; dx + \int_{-T/3}^{T/3} x^2 \frac{1}{2T} \; dx + \int_{T/3}^{T} (x - \frac{2T}{3})^2 \frac{1}{2T} \; dx = \\
&= \frac{3}{2T} \int_{-T/3}^{T/3} x^2 \; dx = \frac{3}{2T} \left[\frac{x^3}{3} \right]_{-T/3}^{T/3} = \frac{T^2}{27}
\end{aligned}
$$

4.2 We start by calculating the variance of the distribution:

$$
\begin{aligned}
\sigma^2 &= \int_{-\infty}^{\infty} x^2 \cdot f(x) \; dx = 2 \int_0^{\infty} x^2 \cdot \frac{1}{2a} e^{-x/a} \; dx = \\
&= \left[-x^2 e^{-x/a} \right]_0^{\infty} + \int_0^{\infty} 2x e^{-x/a} \; dx = \\
&= \left[-2ax e^{-x/a} \right]_0^{\infty} + \int_0^{\infty} 2a e^{-x/a} \; dx = \left[-2a^2 e^{-x/a} \right]_0^{\infty} = 2a^2
\end{aligned}
$$

a) Two reconstruction points (y_1 and y_2) and three decision boundaries (b_0, b_1 and b_2). Since the distribution is symmetric around the origin, we have $y_1 = -y_2$. The optimal decision boundaries should be located halfway between the reconstruction points (except for the two end boundaries, which should be located at the endpoints of the distribution). We thus get $b_0 = -\infty$, $b_1 = \frac{y_1+y_2}{2} = 0$ and $b_2 = \infty$.

What is left is to find y_2. It should be located in the centroid of its corresponding decision region, ie

$$
y_2 = \frac{\int_{b_1}^{b_2} x \cdot f(x) \; dx}{\int_{b_1}^{b_2} f(x) \; dx}
$$

We have

$$
\int_{b_1}^{b_2} f(x) \; dx = \int_0^{\infty} \frac{1}{2a} e^{-x/a} \; dx = \left[-\frac{1}{2} \; e^{-x/a} \right]_0^{\infty} = \frac{1}{2}
$$

and

$$
\begin{aligned}
\int_{b_1}^{b_2} x \cdot f(x) \; dx &= \int_0^{\infty} x \cdot \frac{1}{2a} e^{-x/a} \; dx = \left[-\frac{x}{2} \; e^{-x/a} \right]_0^{\infty} + \int_0^{\infty} \frac{1}{2} \; e^{-x/a} \; dx = \\
&= \left[-\frac{a}{2} \; e^{-x/a} \right]_0^{\infty} = \frac{a}{2}
\end{aligned}
$$

We thus get the the reconstruction points $y_1 = -a = -\frac{\sigma}{\sqrt{2}}$ and $y_2 = a = \frac{\sigma}{\sqrt{2}}$

The distortion is given by

$$\begin{aligned} D &= \int_{-\infty}^{0} (x+a)^2\ f(x)\ dx + \int_{0}^{\infty} (x-a)^2\ f(x)\ dx = \\ &= \int_{-\infty}^{\infty} x^2\ f(x)\ dx + a^2 \int_{-\infty}^{\infty} f(x)\ dx - 4a \int_{0}^{\infty} x f(x)\ dx = \\ &= 2a^2 + a^2 - 4a \cdot \frac{a}{2} = a^2 = \frac{1}{2}\sigma^2 \end{aligned}$$

Compare to the formula collection!

b) Three reconstruction points (y_1, y_2 and y_3) and four decision boundaries (b_0, b_1, b_2 and b_3). Since the distribution is symmetric around the origin, we have $y_1 = -y_3$ and $y_2 = 0$. The optimal decision boundaries should be located halfway between the reconstruction points (except for the two end boundaries, which should be located at the endpoints of the distribution). We thus get $b_0 = -\infty$, $b_1 = \frac{y_1+y_2}{2} = -\frac{y_3}{2}$, $b_2 = \frac{y_2+y_3}{2} = \frac{y_3}{2}$ and $b_3 = \infty$.

What is left is to find y_3. It should be located in the centroid of its corresponding decision region, ie

$$y_3 = \frac{\int_{b_2}^{b_3} x \cdot f(x)\ dx}{\int_{b_2}^{b_3} f(x)\ dx}$$

We have

$$\int_{b_2}^{b_3} f(x)\ dx = \int_{y_3/2}^{\infty} f(x)\ dx = \left[-\frac{1}{2}\ e^{-x/a}\right]_{y_3/2}^{\infty} = \frac{1}{2}\ e^{-y_3/2a}$$

and

$$\begin{aligned} \int_{b_2}^{b_3} x \cdot f(x)\ dx &= \int_{y_3/2}^{\infty} x \cdot \frac{1}{2a} e^{-x/a}\ dx = \left[-\frac{x}{2}\ e^{-x/a}\right]_{y_3/2}^{\infty} + \int_{y_3/2}^{\infty} \frac{1}{2}\ e^{-x/a}\ dx = \\ &= \frac{y_3}{4} e^{-y_3/2a} + \left[-\frac{a}{2}\ e^{-x/a}\right]_{y_3/2}^{\infty} = \left(\frac{y_3}{4} + \frac{a}{2}\right) \cdot e^{-y_3/2a} \end{aligned}$$

which gives us

$$y_3 = \frac{y_3}{2} + a \implies y_3 = 2a$$

Reconstruction points: $y_1 = -2a$, $y_2 = 0$, $y_3 = 2a$.

Decision boundaries: $b_0 = -\infty$, $b_1 = -a$, $b_2 = a$, $b_3 = \infty$.

The distortion is given by

$$\begin{aligned} D &= \int_{-\infty}^{-a} (x+2a)^2\ f(x)\ dx + \int_{-a}^{a} x^2\ f(x)\ dx + \int_{a}^{\infty} (x-2a)^2\ f(x)\ dx = \\ &= \int_{-\infty}^{\infty} x^2\ f(x)\ dx + 8a^2 \int_{a}^{\infty} f(x)\ dx - 8a \int_{a}^{\infty} x f(x)\ dx = \\ &= 2a^2 + 4a^2 e^{-1} - 8a^2 e^{-1} = 2a^2(1-2e^{-1}) = \sigma^2(1-2e^{-1}) \approx 0.2642 \cdot \sigma^2 \end{aligned}$$

4.3 Sample at double the band-width, I.e., 30kHz. This will lead to that no distortion at the receiver is induced by the sampling step (the cut-off filter is assumed to be ideal.) Assume that the quantizer noise is uncorrelated with the signal itself. Then the p.s.d. of the noise will just add to the signal p.s.d., both before and after the pulse-modulation. Since the variance of a signal doesn't scale when using ideal pulse-modulation (search your favorite signal theory literature) the distortion will only be depend on the particular quantizer used. The natural choice of quantizer for a uniform distribution is of course a uniform quantizer.

Assume that the signal is uniformly distributed between $-T$ och T.

The variance for the signal is then $\sigma^2 = \frac{T^2}{3}$.

Let the number of quantization steps be $M = 2^R$.

The step length of the quantizer is $\Delta = \frac{2T}{M} = \frac{2T}{2^R}$

The distortion for uniform quantization of a uniform distribution is $D = \frac{\Delta^2}{12}$, which gives us

$$D = \frac{\Delta^2}{12} = \frac{(2T)^2}{2^{2R} \cdot 12} = \frac{T^2}{2^{2R} \cdot 3} = \frac{\sigma^2}{2^{2R}}$$

which in turn gives us

$$R = \frac{1}{2} \log_2 \frac{\sigma^2}{D}$$

In order to achive an SNR of at least 30 dB we require $\frac{\sigma^2}{D} \geq 10^3$ which gives (if we limit the rate to integers) that $R \geq 5$ [bits/sample] which corresponds to a rate of at least 150 kbit/s.

4.4 a) $D \approx 5 \cdot 0.3634 = 1.817$

b) Since the two quantization levels will be equally probable, $H = 1$ [bit/symbol]

c) $\frac{1}{2} \log \frac{\sigma^2}{D} \approx \frac{1}{2} \log \frac{5}{1.817} \approx 0.73$ [bit/symbol]

4.5 Assuming that no entropy coding is done, ie the varible is quantized to $2^2 = 4$ levels, $D \geq 0.1175 \cdot \sigma^2$

4.6 Call the quantized variable Y and the reconstructed variable $\hat{X}$. Assume that the entropy coder is perfect, i.e., the rate R is equal to the entropy $H(Y)$. If the quantization step of the quantizer is Δ the distortion is then $D \approx \Delta^2/12$

$$\begin{aligned}
R &= H(Y) = -\sum_i p(y_i) \cdot \log p(y_i) \approx \\
&\approx -\sum_i \Delta f_X(\hat{x}_i) \cdot \log \Delta f_X(\hat{x}_i) \approx \\
&\approx -\log \Delta - \int_{-\infty}^{\infty} f_X(x) \log f_X(x) dx = \\
&= -\log \Delta - \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2} \cdot \log \left(\frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2} \right) dx = \\
&= -\log \Delta + \log \sqrt{2\pi}\sigma + \frac{1}{2} \log e = \\
&= \frac{1}{2} \log \frac{2\pi\sigma^2 e}{\Delta^2} \approx \frac{1}{2} \log \frac{\pi e \sigma^2}{6D} = \\
&= \frac{1}{2} \log \frac{\sigma^2}{D} + \frac{1}{2} \log \frac{\pi e}{6} \approx \underline{R(D) + 0.255}
\end{aligned}$$

The lazy solution is to use the formula collection directly.

4.7 Call the quantized variable Y and the reconstructed variable $\hat{X}$. Assume that the entropy coder is perfect, i.e., the rate R is equal to the entropy $H(Y)$. If the quantization step of the quantizer is Δ the distortion is then $D \approx \Delta^2/12$

$$\begin{aligned}
R &= H(Y) = -\sum_i p(y_i) \cdot \log p(y_i) \approx \\
&\approx -\sum_i \Delta f_X(\hat{x}_i) \cdot \log \Delta f_X(\hat{x}_i) \approx \\
&\approx -\log \Delta - \int_{-\infty}^{\infty} f_X(x) \log f_X(x) dx = \\
&= -\log \Delta - 2\int_{0}^{\infty} \frac{1}{\sqrt{2}\sigma} e^{-\sqrt{2}x/\sigma} \cdot \log \left(\frac{1}{\sqrt{2}\sigma} e^{-\sqrt{2}x/\sigma} \right) dx = \\
&= -\log \Delta + \log \sqrt{2}\sigma + \log e = \\
&= \frac{1}{2} \log \frac{2e^2\sigma^2}{\Delta^2} \approx \frac{1}{2} \log \frac{e^2\sigma^2}{6D}
\end{aligned}$$

which gives us

$$D \approx \frac{e^2}{6} \cdot \sigma^2 \cdot 2^{-2R}$$

4.8 Call the quantized variable Y and the reconstructed variable $\hat{X}$. Assume that the entropy coder is perfect, i.e., the rate R is equal to the entropy $H(Y)$. The variance of X is $\sigma^2 = a^2/6$. If the quantization step of the quantizer is Δ the distortion is then $D = \Delta^2/12$

$$\begin{aligned}
R &= H(Y) = -\sum_i p(y_i) \cdot \log p(y_i) \approx \\
&\approx -\sum_i \Delta f_X(\hat{x}_i) \cdot \log \Delta f_X(\hat{x}_i) \approx \\
&\approx -\log \Delta - \int_{-a}^{a} f_X(x) \log f_X(x) dx = \\
&= -\log \Delta - 2 \int_0^a \frac{1}{a}(1 - \frac{x}{a}) \cdot \log \left(\frac{1}{a}(1 - \frac{x}{a}) \right) dx = \\
&= -\log \Delta - \log \frac{1}{a} + \frac{1}{2} \log e = \\
&= -\frac{1}{2} \log \Delta^2 - \frac{1}{2} \log \frac{1}{a^2} + \frac{1}{2} \log e = \frac{1}{2} \log \frac{a^2 e}{\Delta^2} = \\
&= \frac{1}{2} \log \frac{\sigma^2 e}{2D} \quad \Longleftrightarrow \quad \underline{D \approx \frac{e}{2} \cdot \sigma^2 \cdot 2^{-2R}}
\end{aligned}$$

4.9 The optimal choice of step size is according to the formula collection $\Delta(x) = c(f(x))^{-1/3}$. When putting this into

$$\begin{cases} D & \approx \frac{1}{12} \int_{-\infty}^{\infty} \Delta^2(x) f(x) dx \\ R & \approx \log \int_{-\infty}^{\infty} \frac{dx}{\Delta(x)} \end{cases}$$

we get the parametric form:

$$\begin{cases} D & \approx \frac{c^2}{12} \int_{-\infty}^{\infty} (f(x))^{1/3} dx \\ R & \approx \log \frac{1}{c} \int_{-\infty}^{\infty} (f(x))^{1/3} dx \end{cases}$$

Simple calculations yield $\int_{-\infty}^{\infty} (f(x))^{1/3} dx = 3(\sqrt{2}\sigma)^{2/3}$ for the laplacian distribution and hence we can write

$$D = \frac{c^2 (\sqrt{2}\sigma)^{2/3}}{4} \quad \Rightarrow \quad c^2 = \frac{4D}{(\sqrt{2}\sigma)^{2/3}}$$

and we get

$$R = \frac{1}{2} \log \frac{(\sqrt{2}\sigma)^{2/3}}{4D} 3^2 (\sqrt{2}\sigma)^{4/3} = \frac{1}{2} \log \frac{9}{2} \frac{\sigma^2}{D}$$

which gives us:

$$\underline{D \approx \frac{9}{2} \cdot \sigma^2 \cdot 2^{-2R}}$$

4.10 First we need to make some assumptions and approximations. 16 bits/sample is obviously very fine quantization, so we assume that the distribution is constant in each quantization interval. Furthermore, since $a = 6\sigma$ we can ignore the probability that the quantizer is overloaded (it will be roughly $2 \cdot 10^{-9}$). These approximations will give that the distortion $D \approx \frac{\Delta^2}{12}$. The number of levels in the quantizer is

$$M = 2^{16} = \frac{2a}{\Delta}$$

leading to

$$D_U \approx \frac{\Delta^2}{12} = \frac{a^2}{3} 2^{-2 \cdot 16} = 12 \cdot \sigma^2 \cdot 2^{-2 \cdot 16}$$

If we use Max quantization instead of uniform quantization, the distortion will be

$$D_M \approx \frac{\pi\sqrt{3}}{2} \cdot \sigma^2 \cdot 2^{-2 \cdot 16}$$

The difference in SNR is

$$\begin{aligned} 10 \cdot \log_{10} \frac{\sigma^2}{D_M} - 10 \cdot \log_{10} \frac{\sigma^2}{D_U} = 10 \cdot \log_{10} \frac{D_U}{D_M} \approx \\ \approx 10 \cdot \log_{10} \frac{24}{\pi\sqrt{3}} \approx 6.4 \quad [\text{dB}] \end{aligned}$$

Thus, by using Max quantization instead of uniform quantization we will gain 6.4 dB in SNR.

4.11 a) MSE is approximately 438.

b) Each reconstruction point should be moved to the arithmetic mean of the training data within its region.

A should be placed in $(-\frac{130}{3}, -30)$, B in $(\frac{20}{3}, -\frac{10}{3})$, C in $(-\frac{40}{3}, \frac{40}{3})$ and D in $(31.25, 38.75)$. This will result in a new MSE of approximately 340.

c) To minimize the distorsion each training data point should be rounded to the nearest reconstruction point. This implies that the decision boundaries should be straight lines halfway between the reconstruction points. The new MSE will be approximately 321.

4.12 a) Approximately $2.63 \cdot 10^{18}$ years. The size of the codebook will be immense since $1.5 \cdot 8 \cdot 8 = 96$ bits will be used to address it. Generally, if the size of the vector is L pixels, the time required to code one image is $\frac{1024^2}{L} \cdot 2^{1.5 \cdot L} \cdot L = 2^{1.5 \cdot L + 20}$ $[ns]$.

b) Six pixels/block.

c) Ten pixels/block.

4.13 The average distortion D when using fine quantization is given by

$$D = \frac{1}{2} \sum_i \iint_{C_i} f(\bar{x}) ||\bar{x} - \bar{r}_i||^2 d\bar{x} = \frac{1}{2} \sum_i \frac{p_i}{A_i} \iint_{C_i} ||\bar{x} - \bar{r}_i||^2 d\bar{x}$$

where C_i are the decision regions, $\bar{r}_i$ are the reconstruction points, A_i are the areas of the regions and p_i are the probabilities to be in the regions. In the case of lattice quantization, where each region has the same size and shape, we can simplify to

$$D = \frac{1}{2A} \iint\limits_C ||\bar{x}||^2 d\bar{x} = \frac{1}{2A} \iint\limits_C (x^2 + y^2) dx dy$$

where C is the region centered around the origin and A is the area of the region. For the square lattice, we get

$$D_1 = \frac{1}{2d^2} \cdot \frac{d^4}{6} = \frac{d^2}{12}$$

as we already knew.

a) The area of the fundamental region will still be d^2. The distortion will be

$$D_2 = \frac{1}{2d^2} \cdot \frac{31 \cdot d^4}{192} = \frac{31 \cdot d^2}{32 \cdot 12}$$

And the SNR gain will be

$$10 \cdot \log_{10} \frac{D_1}{D_2} = 10 \cdot \log_{10} \frac{32}{31} \approx 0.138 \quad \text{[dB]}$$

b) As before, the area of the fundamental region is d^2. Calculating the distortion over the regular hexagon, we get

$$D_3 = \frac{1}{2d^2} \cdot \frac{d^4}{3.6 \cdot \sqrt{3}} = \frac{d^2}{7.2 \cdot \sqrt{3}}$$

And the SNR gain will be

$$10 \cdot \log_{10} \frac{D_1}{D_3} = 10 \cdot \log_{10} \frac{7.2\sqrt{3}}{12} \approx 0.167 \quad \text{[dB]}$$

5.1 a) $p_i = 0.9 \cdot X_{i-1}, \quad \sigma_d^2 = 0.19.$

b) $p_i = \frac{27}{19} X_{i-1} - \frac{11}{19} X_{i-2}, \quad \sigma_d^2 \approx 0.1263$

5.2 a) $p_i = \rho \cdot X_{i-1}, \quad \sigma_d^2 = 1 - \rho^2.$

b) $p_i = \rho \cdot X_{i-1} + 0 \cdot X_{i-2}, \quad \sigma_d^2 = 1 - \rho^2.$

i.e., the order-2 predictor does not improve the situation. Will any higher order linear predictor work better?

5.3 a) $a = \rho \cdot \frac{R_{XX}(0)}{R_{XX}0 + \mathrm{E}\{\varepsilon^2\}} \approx \rho \cdot \frac{R_{XX}(0)}{R_{XX}(0) + \frac{\Delta^2}{12}}$

b) $(1+a^2)R_{XX}(0) + a^2\mathrm{E}\left\{\varepsilon^2\right\} - 2a\rho R_{XX}(0)$

5.4 The problem can be solved by using an order 2 linear predictor. A predictor of order 1 won't achieve the wanted SNR.

Quantization to 6 bits/sample can be seen as fine quantization $\Rightarrow$ assume that the predictor works with original samples:

$$p_n = a_1 \cdot \hat{X}_{n-1} + a_2 \cdot \hat{X}_{n-2} \approx a_1 \cdot X_{n-1} + a_2 \cdot X_{n-2}$$

The prediction error variance is

$$\sigma_d^2 = E\{(X_n - p_n)^2\} \approx E\{(X_n - a_1 \cdot X_{n-1} - a_2 \cdot X_{n-2})^2\}$$

The predictor that minimizes σ_d^2 is given by

$$\mathbf{A} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \mathbf{R}^{-1} \cdot \mathbf{P}$$

where

$$\mathbf{R} = \begin{pmatrix} R_{XX}(0) & R_{XX}(1) \\ R_{XX}(1) & R_{XX}(0) \end{pmatrix} \quad \text{and} \quad \mathbf{P} = \begin{pmatrix} R_{XX}(1) \\ R_{XX}(2) \end{pmatrix}$$

This gives the optimal predictor $\mathbf{A}$ and the prediction error variance σ_d^2

$$\mathbf{A} \approx \begin{pmatrix} 0.9270 \\ -0.8382 \end{pmatrix} \qquad \sigma_d^2 \approx 0.5145$$

If the quantizer we use is a Max quantizer, the distortion will be

$$D \approx \frac{\pi\sqrt{3}}{2} \cdot \sigma_d^2 \cdot 2^{-2\cdot 6} \approx 0.00034177$$

Signal to noise ratio:

$$10 \cdot \log_{10} \frac{\sigma_X^2}{D} = 10 \cdot \log_{10} \frac{R_{XX}(0)}{D} \approx 38.3 \quad [\text{dB}]$$

5.5 One suitable predictor (there are several) is

$$p_{i,j} = a_1 \cdot \hat{X}_{i-1,j} + a_2 \cdot \hat{X}_{i,j-1} \approx a_1 \cdot X_{i-1,j} + a_2 \cdot X_{i,j-1}$$

Because of the fine quantization we can do the calculations as if the predictor worked using the original signal, ie we can disregard the effect of the quantization on the prediction. The variance of the prediction error:

$$\sigma_d^2 = E\{(X_{i,j} - p_{i,j})^2\} \approx E\{(X_{i,j} - a_1 X_{i-1,j} - a_2 X_{i,j-1})^2\}$$

a_1 and a_2 that minimize σ_d^2 are given by

$$\begin{pmatrix} 1 & 0.95 \cdot 0.92 \\ 0.95 \cdot 0.92 & 1 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} 0.92 \\ 0.95 \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \approx \begin{pmatrix} 0.3799 \\ 0.6180 \end{pmatrix}$$

$$\Rightarrow \sigma_d^2 \approx 0.0634$$

We use uniform quantization followed by entropy coding to the rate 6 bits/pixel, which gives us the distortion

$$D \approx \sigma_d^2 \frac{\pi e}{6} 2^{-2 \cdot 6} \approx 0.000022039$$

$$\text{SNR} = 10 \cdot \log_{10} \frac{\sigma_X^2}{D} \approx 46.6 \text{ [dB]}$$

(We can also reach 45 dB with the rate 5.74 bits/pixel.)

6.1 Transform matrix (one basis function per row)

$$\mathbf{A} = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

Correlation matrix for transform components, variances in the diagonal

$$\begin{aligned} \mathbf{R}_\theta &= \mathbf{A}\mathbf{R}_X\mathbf{A}^T = \mathbf{A} \cdot \begin{pmatrix} 4.63 & 2.34 & -1.72 & -3.55 \\ 2.34 & 4.63 & 2.34 & -1.72 \\ -1.72 & 2.34 & 4.63 & 2.34 \\ -3.55 & -1.72 & 2.34 & 4.63 \end{pmatrix} \cdot \mathbf{A}^T \\ &\approx \begin{pmatrix} 4.645 & 0 & 0 & -2.945 \\ 0 & 1.175 & 2.945 & 0 \\ 0 & 2.945 & 9.295 & 0 \\ -2.945 & 0 & 0 & 3.405 \end{pmatrix} \end{aligned}$$

Using our favourite bit allocation algorithm, we find that that the first and fourth components should be quantized with 1 bit each, the second component with no bits and the third with 2 bits, giving an average rate of 1 bit/sample. The resulting distortion is

$$D_T \approx \frac{0.3634 \cdot 4.645 + 1 \cdot 1.175 + 0.1175 \cdot 9.295 + 0.3634 \cdot 3.405}{4} \approx 1.2981$$

giving an SNR of

$$\text{SNR}_T \approx 10 \cdot \log_{10} \frac{4.63}{1.2981} \approx 5.52 \text{ [dB]}$$

Without transform, we get the distortion

$$D_Q \approx 0.3634 \cdot 4.63 \approx 1.6825$$

which gives the SNR

$$\text{SNR}_Q \approx 10 \cdot \log_{10} \frac{4.63}{1.6825} \approx 4.40 \text{ [dB]}$$

The transform coding gain is 1.12 dB.

6.2 The transform matrix (basis vectors in the rows) of a 3 point DCT is

$$\mathbf{A} = \sqrt{\frac{2}{3}} \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} \\ \sqrt{3}/2 & 0 & -\sqrt{3}/2 \\ 1/2 & -1 & 1/2 \end{pmatrix} = \begin{pmatrix} 1/\sqrt{3} & 1/\sqrt{3} & 1/\sqrt{3} \\ 1/\sqrt{2} & 0 & -1/\sqrt{2} \\ 1/\sqrt{6} & -2/\sqrt{6} & 1/\sqrt{6} \end{pmatrix}$$

Variances of the three transform components:

$$\begin{aligned}
\sigma_0^2 &= E\{\theta_0^2\} = \frac{1}{3}E\{(X_0 + X_1 + X_2)^2\} = \\
&= \frac{1}{3}\left(3R_{XX}(0) + 4R_{XX}(1) + 2R_{XX}(2)\right) = 2.7654 \\
\sigma_1^2 &= E\{\theta_1^2\} = \frac{1}{2}E\{(X_0 - X_2)^2\} = \\
&= \frac{1}{2}\left(2R_{XX}(0) - 2R_{XX}(2)\right) = 0.1719 \\
\sigma_2^2 &= E\{\theta_2^2\} = \frac{1}{6}E\{(X_0 - 2X_1 + X_2)^2\} = \\
&= \frac{1}{6}\left(6R_{XX}(0) - 8R_{XX}(1) + 2R_{XX}(2)\right) = 0.0627
\end{aligned}$$

Alternatively we can calculate the variances as the diagonal elements of $\mathbf{A} \cdot \mathbf{R}_X \cdot \mathbf{A}^T$, where

$$\mathbf{R}_X = \begin{pmatrix} 1 & 0.91 & 0.8281 \\ 0.91 & 1 & 0.91 \\ 0.8281 & 0.91 & 1 \end{pmatrix}$$

The desired rate is high enough for us to use the fine quantization approximations for optimal bit allocation, ie

$$R_i = R + \frac{1}{2}\log_2 \frac{\sigma_i^2}{(\sigma_0^2 \cdot \sigma_1^2 \cdot \sigma_2^2)^{1/3}}$$

which gives us

$$\begin{aligned} R_0 &\approx 6.08 \\ R_1 &\approx 4.07 \\ R_2 &\approx 3.35 \end{aligned}$$

Fractional rates are allowed, since we are using source coding.

Assuming that the arithmetic coder is a perfect source coder, we get the average distortion

$$D \approx \frac{\pi e}{6} \cdot (\sigma_0^2 \cdot \sigma_1^2 \cdot \sigma_2^2)^{1/3} \cdot 2^{-2\cdot 4.5} \approx 0.0008619$$

The signal to noise ratio is

$$10 \cdot \log_{10} \frac{\sigma_X^2}{D} = 10 \cdot \log_{10} \frac{1}{D} \approx 30.65 \text{ dB}$$

If we use a KLT instead, we get the distortion

$$D_K \approx \frac{\pi e}{6} \cdot \det(\mathbf{R}_X)^{1/3} \cdot 2^{-2\cdot 4.5} \approx 0.0008594$$

and the signal to noise ratio

$$10 \cdot \log_{10} \frac{1}{D_K} \approx 30.66 \text{ dB}$$

As we can see, the gain from using a KLT instead of a DCT is marginal.

6.3 The two-point Hadamard transform is given by

$$\mathbf{H}_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Applying this transform separably on a block of size 2×2 pixels is equivalent to to applying the transform

$$\mathbf{A} = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

to vectors like this

$$\mathbf{x} = \begin{pmatrix} X_{i,j} \\ X_{i,j+1} \\ X_{i+1,j} \\ X_{i+1,j+1} \end{pmatrix}, \qquad \theta = \mathbf{A} \cdot \mathbf{x}$$

The variances of the four transform components are

$$\begin{aligned} \sigma_0^2 &= 6.34 \\ \sigma_1^2 &= 0.14 \\ \sigma_2^2 &= 0.22 \\ \sigma_3^2 &= 0.10 \end{aligned}$$

The bit allocation that minimzes the average distortion and gives an average rate of 1.5 bits/pixel is

$$R_0 = 4, \quad R_1 = 1, \quad R_2 = 1, \quad R_3 = 0$$

which gives the average distortion

$$D \approx \frac{0.009497 \cdot 6.34 + 0.3634 \cdot 0.14 + 0.3634 \cdot 0.22 + 0.10}{4} \approx 0.07276$$

and the signal-to-noise ratio

$$10 \cdot \log_{10} \frac{\sigma_X^2}{D} \approx 13.7 \text{ [dB]}$$

6.4 $\varphi = \frac{\pi}{4}$

$R_1 \approx R + \frac{1}{4} \log \frac{1+\rho}{1-\rho}$

$R_2 \approx R - \frac{1}{4} \log \frac{1+\rho}{1-\rho}$

$D_{\min} \approx \frac{\pi\sqrt{3}}{2} \sqrt{1-\rho^2}\; 2^{-2R}$

6.5 a) 5 bits/sample can be regarded as fine quantization. Fine quantization and optimal bit allocation will give (see the formula collection) the distortion $D \approx c\sigma^2 2^{-2R}$ where $c = \frac{\pi e}{6}$ (uniform quantization and entropy coding of gaussian process) and $\sigma^2 = \sqrt{\sigma_0^2 \sigma_1^2}$. Thus

$$D_{KLT} \approx \frac{\pi e}{6} \sqrt{5.85 \cdot 0.15} \cdot 2^{-2 \cdot 5} \approx 0.0013020$$

The variance of X_n is given by $\sigma_X^2 = \frac{1}{2}(\sigma_0^2 + \sigma_1^2) = 3$. Thus, the signal-to-noise ratio is

$$\text{SNR}_{KLT} = 10 \log_{10} \frac{\sigma_X^2}{D_{KLT}} \approx \underline{33.6} \;\; [dB]$$

b) Since the transform is a KL transform, the correlation matrix $\mathbf{R}_\theta$ for the transform coefficients is diagonal. If the autocorrelation function for X_n is $R_{XX}(k)$, the corresponding correlation matrix $\mathbf{R}_X$ for X_n is given by

$$\mathbf{R}_X = \begin{pmatrix} R_{XX}(0) & R_{XX}(1) \\ R_{XX}(1) & R_{XX}(0) \end{pmatrix} = \begin{pmatrix} 3 & R_{XX}(1) \\ R_{XX}(1) & 3 \end{pmatrix}$$

The KL transform for this correlation matrix is the Hadamard transform

$$\mathbf{A} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

which gives us

$$\mathbf{R}_\theta = \mathbf{A}\mathbf{R}_X\mathbf{A}^T = \begin{pmatrix} 3 + R_{XX}(1) & 0 \\ 0 & 3 - R_{XX}(1) \end{pmatrix} = \begin{pmatrix} 5.85 & 0 \\ 0 & 0.15 \end{pmatrix}$$

Thus, $R_{XX}(1) = 2.85$. Since this is all the information we can deduce, the best predictor we can construct is an order 1 predictor. Since the quantization is fine, we ignore the effect of quantization on the predictor. The order 1 predictor $p_n = a \cdot X_{n-1}$ that minimizes the prediction error variance σ_d^2 is given by

$$\sigma_d^2 = E\{(X_n - aX_{n-1})^2\} = 3(1 + a^2) - 5.7a$$

$$\frac{\partial}{\partial a}\sigma_d^2 = 6a - 5.7 = 0 \quad \Rightarrow \quad a = 0.95 \quad \Rightarrow \quad \sigma_d^2 = 0.2925$$

$$D_{pred} = \frac{\pi e}{6} 0.2925 \cdot 2^{-2 \cdot 5} \ ; \ \mathrm{SNR}_{pred} = 10 \log_{10} \frac{\sigma_X^2}{D_{pred}} \approx \underline{38.7} \ \ [dB]$$

6.6 Expressing the two transforms as matrices we get

$$\mathbf{A}_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{pmatrix} \text{ and } \mathbf{A}_2 = \begin{pmatrix} 1/2 & 1/2 & 1/2 & 1/2 \\ 1/2 & 1/2 & -1/2 & -1/2 \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 & 0 \\ 0 & 0 & 1/\sqrt{2} & -1/\sqrt{2} \end{pmatrix}.$$

Let $\mathbf{R}_X = \begin{pmatrix} 1 & 0.9 & 0.9^2 & 0.9^3 \\ 0.9 & 1 & 0.9 & 0.9^2 \\ 0.9^2 & 0.9 & 1 & 0.9 \\ 0.9^3 & 0.9^2 & 0.9 & 1 \end{pmatrix}$ denote the 4×4 correlation matrix of the source. The variances of the transform components can now be found in the diagonals of $\mathbf{A}_1\mathbf{R}\mathbf{A}_1^T$ and $\mathbf{A}_2\mathbf{R}\mathbf{A}_2^T$ respectively.

Repeatedly adding one bit to the component with the currently largest distortion yields the following tables over distortions. Individual component distortions, for the gaussian Max quantization case, are listed in the formula collection.

Variances	*1.9*	*0.1*	*1.9*	*0.1*
1st bit	0.6905	0.0363	0.6905	0.0363
2nd bit	0.2232		0.2232	
3rd bit	0.0656		0.0656	

Transform at A

Variances	*3.5245*	*0.2755*	*0.1*	*0.1*
1st bit	1.2808	0.1001	0.0363	0.0363
2nd bit	0.4141	0.0324		
3rd bit	0.1217			
4th bit	0.0335			

Transform at B

Since the resulting transforms are orthogonal we can compare the transform component distortions directly,

$$10 \log_{10} \frac{0.0656 + 0.0363 + 0.0656 + 0.0363}{0.0335 + 0.0324 + 0.0363 + 0.0363} \approx 1.68$$

Hence the gain is approximately 1.68 dB.

6.7 a) Resulting variances for the different Y-signals: $\sigma^2_{Y_1} \approx 6.6091$, $\sigma^2_{Y_2} \approx 0.7909$ and $\sigma^2_{Y_3} = 0.3$. 4 bits will be allocated to Y_1, 2 bits to Y_2 and 1 bit to Y_3. Note that the sample-rate of Y_3 is 2 times higher than for Y_1 and Y_2. For every 4 input samples, 2 samples will be output at Y_3 and one each at Y_1 and Y_2. Thus, the bits allocated to Y_3 counts double. The bit-rate will hence be $(4+2+2\cdot 1)/4 = 2$ bits/sample.

b) $D \approx \frac{1}{4}(0.009497 \cdot 6.6091 + 0.1175 \cdot 0.7909 + 2 \cdot 0.3634 \cdot 0.3) \approx 0.09343$
Using a gaussian 4-level Max quantizer directly on the signal would result in the distortion $D \approx 2 \cdot 0.1175 = 0.235$, i.e., 4 dB worse than the subband coder.

6.8 The system is equivalent to a transform coder, using the Hadamard transform on 2×2 blocks. The four transform components are

$$\begin{aligned}
Y_1 &= (X_{i,j} + X_{i,j+1} + X_{i+1,j} + X_{i+1,j+1})/2 \\
Y_2 &= (X_{i,j} + X_{i,j+1} - X_{i+1,j} - X_{i+1,j+1})/2 \\
Y_3 &= (X_{i,j} - X_{i,j+1} + X_{i+1,j} - X_{i+1,j+1})/2 \\
Y_4 &= (X_{i,j} - X_{i,j+1} - X_{i+1,j} + X_{i+1,j+1})/2
\end{aligned}$$

with variances

$$\sigma_1^2 = \text{Var}\{Y_1\} = 3.7071$$

$$\sigma_2^2 = \text{Var}\{Y_2\} = 0.1493$$
$$\sigma_3^2 = \text{Var}\{Y_3\} = 0.0929$$
$$\sigma_4^2 = \text{Var}\{Y_4\} = 0.0507$$

Rate 2 bits/pixel $\Rightarrow$ distribute $4 \cdot 2 = 8$ bits on the four quantizers. Y_1 will be quantized with 4 bits, Y_2 with 2 bits and Y_3 and Y_4 with one bit each. This gives a resulting average distortion

$$D = (0.009497 \cdot \sigma_1^2 + 0.1175 \cdot \sigma_2^2 + 0.3634 \cdot (\sigma_3^2 + \sigma_4^2))/4 \approx 0.0262$$