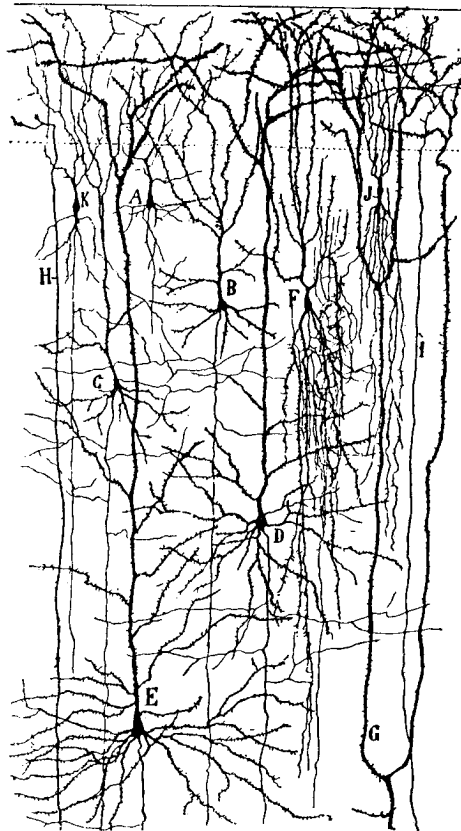


# Information Theory in Neural Networks

Lecture Notes of Course G31/CMNN14

September 2002



ACC Coolen  
Department of Mathematics  
King's College London

# Contents

<b>1</b>	<b>Introduction: Measuring Information</b>	<b>3</b>
<b>2</b>	<b>Elementary Probability Concepts</b>	<b>15</b>
2.1	General Theory . . . . .	15
2.2	Discrete Event Sets . . . . .	20
2.3	Continuous Event Sets . . . . .	22
2.4	Random Variables and Averages . . . . .	24
<b>3</b>	<b>Building Blocks of Shannon’s Information Theory</b>	<b>31</b>
3.1	Entropy . . . . .	31
3.2	Joint Entropy and Conditional Entropy . . . . .	35
3.3	Relative Entropy and Mutual Information . . . . .	39
3.4	Entropy and Mutual Information for Continuous Random Variables . . . . .	44
<b>4</b>	<b>Identification of Entropy as a Measure of Information</b>	<b>51</b>
4.1	Coding Theory . . . . .	51
4.2	Entropy and Optimal Coding . . . . .	57
<b>5</b>	<b>Applications to Neural Networks</b>	<b>63</b>
5.1	Supervised Learning: Boltzmann Machines . . . . .	63
5.2	Maximum Information Preservation . . . . .	69
5.3	Neuronal Specialisation . . . . .	72
5.4	Detection of Coherent Features . . . . .	78
5.5	The Effect of Non-linearities . . . . .	80
5.6	Introduction to Amari’s Information Geometry . . . . .	82
<b>A</b>	<b>Simple Mathematical Tools</b>	<b>93</b>
<b>B</b>	<b>Gaussian Integrals</b>	<b>95</b>
<b>C</b>	<b>The <math>\delta</math>-Distribution</b>	<b>101</b>
<b>D</b>	<b>Inequalities Based on Convexity</b>	<b>103</b>



# Chapter 1

## Introduction: Measuring Information

Neural networks are information processing systems, so it is hardly a surprise that if we aim to quantify the operation of neural networks, or to systematically design appropriate architectures and learning rules, we can use many of the tools that have already been developed for the more conventional information processing systems in engineering. Information theory is a very elegant, well founded and rigorous framework to quantify information and information processing. It is remarkable that this framework was essentially the solo project of one single creative person, Claude Shannon (who launched information theory with a publication in 1948, building on work he did during the war on radar communication), and at the time came almost entirely out of the blue, much like Einstein's general relativity theory.

Measuring information at first sight appears to be a somewhat vague and strange concept. Although especially those of us who are reasonably familiar with using computers will be used to thinking about information content (in terms of *bits*), we will find out that measuring information is not simply counting the number of bits in a data file, but involves somewhat surprisingly probability concepts. Therefore we will start with an introduction aimed at developing some feeling and intuition for which are the relevant issues, why probabilities enter the game, and also play with a number of simple examples, before we dive into formal definitions and proofs.

One can best think about information in a down-to-earth manner, in terms of messages communicated between a sender and a receiver. Before meaningful communication can take place, sender and receiver need to agree on the format of these messages. This could be lines of text, numbers, bleeps representing morse-code, smoke-signals, electric pulses, etc. We write the set of possible messages (agreed upon by sender and receiver) as

$$A = \{a_1, a_2, \dots\} \quad \text{with } |A| \text{ elements} \quad (1.1)$$

Consider the example of a horse race, with five competitors. Sending a message to reveal the outcome of the race just means sending the number of the winning horse, so the set of possible messages is  $A = \{1, 2, 3, 4, 5\}$ , with  $|A| = 5$ . Another example: sending a message consisting of a single word with up to three ordinary characters from the alphabet. Here  $A = \{a, b, \dots, z, aa, ab, ac, \dots, aaa, aab, aac, \dots, zzz\}$ , with  $|A| = 26 + 26^2 + 26^3 = 18,278$ . The set  $A$  could also have an infinite number of elements. If we put no a priori limit on the

number of bleeps used in a telegraphic message, the set  $A$ , being the set of all such possible telegraphic messages will be infinitely large. Now suppose I am being communicated an message  $a \in A$  (I receive a card with something written on it, or I am being told the number of the winning horse in a race). How much information have I received? Is it at all possible to give a sensible unambiguous meaning to such a question?

### 1.0.1 Brute Force: Information Content based on Counting Messages

The first approach we can take is to label all elements in  $A$  with binary numbers (strings of bits). Let us, for simplicity, assume that the number of possible messages is some integer power of two, i.e.  $|A| = 2^n$  so  $A = \{a_1, \dots, a_{2^n}\}$ :

<i>message :</i>	<i>n-bit string :</i>	<i>corresponding number :</i>
$a_1$	000...0	0
$a_2$	100...0	1
$a_3$	010...0	2
$a_4$	110...0	3
$\vdots$	$\vdots$	$\vdots$
$a_{2^n}$	111...1	$2^n - 1$

The correspondence between the elements of  $A$  and the integer numbers  $\{0, 1, \dots, 2^n - 1\}$  is one-to-one; in other words:

$n$  bits are *sufficient* to uniquely describe each  $a \in A$   
 $n$  bits are *necessary* to uniquely describe each  $a \in A$

It appears that a measure of the amount of information communicated by a message can be obtained by simply counting the number of bits  $n$  needed to specify the label. In terms of the number of elements in the set  $|A| = 2^n$  this gives the tentative expression

$$\text{information content of a single message } a \in A = {}^2\log|A| \quad (1.2)$$

This simple result has a nice self-consistency property, which any candidate definition of information content must have. Consider the example of having two sets of messages  $A$  and  $B$ :

$$\begin{aligned} A &= \{a_1, a_2, \dots\} && \text{with } |A| \text{ elements} \\ B &= \{b_1, b_2, \dots\} && \text{with } |B| \text{ elements} \end{aligned}$$

Now imagine that the message being communicated is a pair of variables  $(a, b)$ , with  $a \in A$  and  $b \in B$  (like giving both the name of a winning horse and the current price of a Spice Girls CD). If the two individual variables  $a$  and  $b$  are completely independent we must require the information content of the pair  $(a, b)$  to be the sum of the two individual information contents of  $a$  and  $b$ , giving the value  ${}^2\log|A| + {}^2\log|B|$ . On the other hand we can also apply the above method of labelling messages with bit-strings to the set  $C$  of all pairs  $(a, b)$ :

$$\begin{aligned} C &= \{(a, b) \mid a \in A, b \in B\} \\ C &= \{c_1, c_2, \dots\} && \text{with } |C| = |A||B| \text{ elements} \end{aligned}$$

The information content in the latter case would come out as  ${}^2\log|C|$ . We see that the two outcomes are indeed identical, since  ${}^2\log|C| = {}^2\log|A| + {}^2\log|B|$ .

However, this cannot be the end of the story. There are several ways to show that the above definition of information content, although correct in the case where we just communicate one single message  $a$  from a specified set  $A$ , is not sufficiently powerful to deal with all relevant situations. Firstly: let us go back to the example of communicating a message pair  $(a, b)$ , where we assumed independence of  $a$  and  $b$  (without as yet specifying precisely what this means) and subsequently found

$$a \text{ and } b \text{ independent : } \quad \text{information in message pair } (a, b) = {}^2\log|A| + {}^2\log|B|$$

Alternatively we can choose messages which are clearly related. To consider an extreme choice: suppose  $a$  represents the number of a winning horse and  $b$  the name of its jockey. In this case knowledge of  $a$  implies knowledge of  $b$ , and vice versa. We gain nothing by knowing both, so in this case

$$a \text{ and } b \text{ strongly dependent : } \quad \text{information in message pair } (a, b) = {}^2\log|A|$$

In general one will mostly have a situation in between, with messages relating to data which are correlated to some degree, for example with  $a \in A$  representing the name of the winning horse and  $b \in B$  representing its age (assuming old horses to have on average a reduced chance of winning). Such situations can only be dealt with properly by using probability theory.

Another way to see how probabilities come in is to compare the simple situation of a single set of messages  $a \in A$ , where we found

$$A = \{a_1, a_2, a_3, \dots\} \text{ with } |A| \text{ elements : } \quad \text{information in message } a \in A = {}^2\log|A|$$

with the situation we get when one of the messages, say  $a_1$ , never occurs, giving

$$A' = \{a_2, a_3, \dots\} \text{ with } |A|-1 \text{ elements : } \quad \text{information in message } a \in A' = {}^2\log(|A|-1)$$

So far no problem. However, what happens if we have a situation in between, where element  $a_1$  does occur but rather infrequently? If  $a_1$  can be expected to occur on average only once every 1000,000,000 times (e.g.  $a_1$  happens to be the name of a racing horse with three legs), we would be inclined to say that the information content of a message  $a \in A$  is closer to  ${}^2\log(|A|-1)$  (corresponding to  $a_1$  simply being absent) than to  ${}^2\log|A|$ . Again we need statistical tools to deal with real-world situations, where not all messages are equally likely to occur.

The solution of these problems lies in realising that communicating information is equivalent to *reducing uncertainty*. Before we actually open the envelope and read the message therein, as far as we know the envelope could contain any of the messages of the set  $A$ . By reading the message, however, our uncertainty is reduced to zero. In the case of the combined message  $(a, b)$  where we have already been told what  $a$  is, it is clear that our reduction in uncertainty upon hearing the value of  $b$  is less in the case where  $a$  and  $b$  are strongly correlated than in the case where they are independent. Similarly, knowing that in a race with two horses the four-legged horse has won (at the expense of the three-legged one) does not reduce our uncertainty about the outcome very much, whereas it does in the case of two equally fast contenders.

### 1.0.2 Exploiting Message Likelihood Differences via Coding

We will now demonstrate that the average information content of messages from a message set  $A$  can indeed be less than  ${}^2\log|A|$ , as soon as not all messages are equally likely to occur, by looking at various methods of coding these messages. We will give formal and precise definitions later, but for now it will be sufficient to define binary coding of messages simply as associating with each message  $a \in A$  a unique string of binary numbers (the ‘code words’). We have already played with an example of this procedure; we will now call the bit-string associated with each element its code word, and the full table linking messages to their code words simply a ‘code’:

<i>message :</i>	<i>code word :</i>
$a_1$	000...0
$a_2$	100...0
$a_3$	010...0
$a_4$	110...0
$\vdots$	$\vdots$
$a_{2^n}$	111...1

This particular code, where the elements of the set  $A$  are ordered (enumerated), and where the code word of each element is simply the binary representation of its rank in the list is called an ‘enumerative code’. All code words are of the same length. Clearly, enumerative coding can be used only if  $|A|$  is finite.

The key idea now is to realise that one is not forced to use a set of code words which are all of the same length. In the case where some messages will occur more frequently than others, it might make sense to use *shorter* code words for frequent messages, and to accept longer code words for the infrequent messages. For instance, in Morse code the number of symbols used for infrequent characters such as Q (represented by ‘— — · —’) are deliberately chosen larger than those of frequent characters, such as E (represented by ‘·’). The fact that this alternative way of coding can indeed *on average* reduce the number of bits needed for communicating messages which are not all equally frequent, is easily demonstrated via explicit construction. Let us enumerate the messages in a given finite set  $A$  as  $A = \{a_1, a_2, \dots\}$  and let us write the probability that message  $a$  occurs as  $p(a)$  (if all messages are equally likely then  $p(a) = |A|^{-1}$  for all  $a$ ). Finally, the length of the binary code word used for element  $a$  (the number of bits) will be denoted by  $\ell(a)$  (note that for enumerative codes all  $\ell(a)$  are the same, by construction). For example:

$$A = \{a_1, a_2, a_3, a_4\} \quad p(a_1) = \frac{1}{2}, \quad p(a_2) = \frac{1}{4}, \quad p(a_3) = \frac{1}{8}, \quad p(a_4) = \frac{1}{8}$$

Let us now compare the performance of enumerative coding to that of an alternative coding recipe (a so-called ‘prefix code’<sup>1</sup>), where the lengths of the code-words are adapted to the

---

<sup>1</sup>The name ‘prefix code’ indicates that it is constructed in such a way that no code-word occurs as the prefix of another code-word. Since code-words are no longer guaranteed to have a uniform length, this property is needed in order to ensure that the receiver knows when one code-word ends and the next code-word begins. In the present example we can always be sure that a code-word ends precisely when we either receive a ‘1’ or when we receive the third ‘0’ in a row.

probabilities of occurrence of the four messages:

<i>message :</i>	<i>enumerative code :</i>		<i>prefix code :</i>	
$a_1$	00	$\ell(a_1) = 2$	1	$\ell(a_1) = 1$
$a_2$	10	$\ell(a_2) = 2$	01	$\ell(a_2) = 2$
$a_3$	01	$\ell(a_3) = 2$	001	$\ell(a_3) = 3$
$a_4$	11	$\ell(a_4) = 2$	000	$\ell(a_4) = 3$

Clearly, if we communicate just one individual message it could be that we use more bits in the prefix code ( $a_3 \rightarrow 001$ ,  $a_4 \rightarrow 000$ ) than in the enumerative code ( $a_3 \rightarrow 01$ ,  $a_4 \rightarrow 11$ ) to do so. However, if we calculate the *average number of bits* used, then the picture is interestingly different. If we send a message  $a \in A$  say  $m$  times, and call the actual  $m$  messages thus communicated  $\{a(1), a(2), \dots, a(m)\}$  (where the message  $a(t)$  communicated at ‘time’  $t$  is at each instance drawn at random from  $A$  according to the above probabilities), then the average number of bits is by definition

$$\text{average number of bits} = \frac{1}{m} \sum_{t=1}^m \ell(a(t))$$

We can now define the code length  $L$  as the average number of bits used in the limit  $m \rightarrow \infty$ , i.e.  $L = \lim_{m \rightarrow \infty} \frac{1}{m} \sum_{t=1}^m \ell(a(t))$ . Since all individual messages  $a(t)$  have been drawn at random from the set  $A$ , with the specified probabilities, we can use the property that in the limit  $m \rightarrow \infty$  any average over  $m$  independent trials becomes an average over the underlying probability distribution (this is how probabilities are defined !), so:

$$\text{code length :} \quad L = \lim_{m \rightarrow \infty} \frac{1}{m} \sum_{t=1}^m \ell(a(t)) = \sum_{i=1}^4 p(a_i) \ell(a_i)$$

which for the present example gives:

$$\begin{aligned} \text{enumerative code :} \quad L &= \frac{1}{2} \cdot 2 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 2 + \frac{1}{8} \cdot 2 = 2 \quad \text{bits} \\ \text{prefix code :} \quad L &= \frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 3 + \frac{1}{8} \cdot 3 = 1.75 \quad \text{bits} \end{aligned}$$

We see explicitly in this example that the *average* information content of messages must be smaller than  $^2\log|A| = 2$ , since simply by coding messages cleverly one can on average communicate the messages  $a \in A$  using less than two bits each.

We can easily generalise the construction of the above simple version of a prefix-code to message sets  $A$  of arbitrary size. First we order the messages  $a \in A$  according to decreasing probability of occurrence, i.e.

$$A = \{a_1, a_2, \dots, a_n\} \quad p(a_1) \geq p(a_2) \geq \dots \geq p(a_n)$$

We now assign to each element  $a \in A$  a binary code-word  $C(a) \in \bigcup_{K \geq 1} \{0, 1\}^K$  (the latter



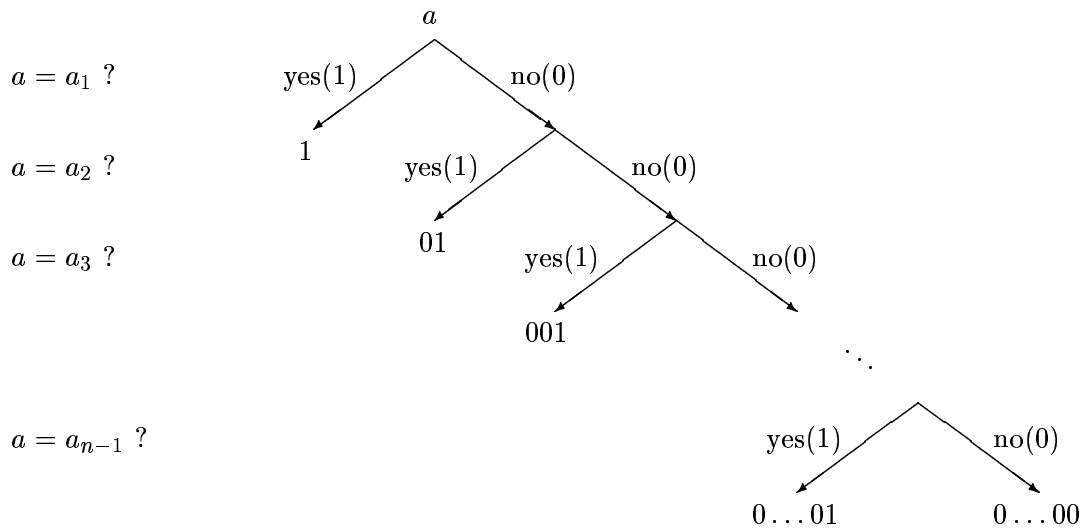
set is the union of all binary strings with one or more symbols) in the following manner:

<i>message :</i>	<i>prefix code :</i>	
$a_1$	1	$\ell(a_1) = 1$
$a_2$	01	$\ell(a_2) = 2$
$a_3$	001	$\ell(a_3) = 3$
$\vdots$	$\vdots$	$\vdots$
$a_{n-1}$	00...01	$\ell(a_{n-1}) = n-1$
$a_n$	00...00	$\ell(a_n) = n-1$

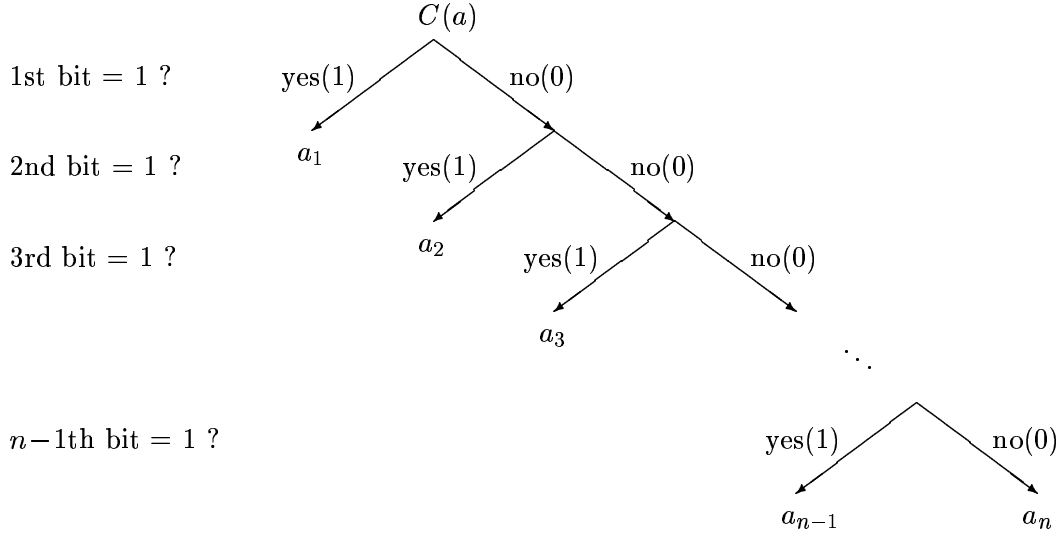
The idea is simple. For  $i < n$  we take  $i-1$  zero's, followed by a one. For  $i = n$  we choose  $n-1$  zero's. Note that the code is again of the 'prefix' type: no code-word occurs as the prefix of another code-word, so the receiver always knows when one message ends and the next message begins (a message terminates either when a one occurs, or when  $n-1$  zero's have occurred in a row). In a formula, which is more suitable for further analysis, we would write:

$$\begin{aligned}
 i < n : C(a_i) &= \overbrace{00 \dots 0}^{i-1 \text{ times}} 1 & \ell(a_i) &= i \\
 i = n : C(a_n) &= \overbrace{00 \dots 00}^{n-1 \text{ times}} & \ell(a_n) &= n-1
 \end{aligned}$$

An alternative way of viewing this particular code, and a simple way to decode a message received, is obtained by interpreting the bits in each code word as representing successive answers in a decision tree. The following tree gives the code word  $C(a)$  for each given message  $a$ :



Conversely, the following tree gives the message  $a$  for each given code word  $C(a)$ :



We can now very easily calculate the code length  $L$  (i.e. the average number of bits needed to communicate a message) for arbitrary ordered message sets  $A = \{a_1, a_2, \dots\}$  with occurrence probabilities  $p(a_1) \geq p(a_2) \geq p(a_3) \geq \dots$ . Let us for now restrict ourselves to finite message sets, so  $|A| < \infty$ :

$$L = \sum_{k \geq 1} p(a_k) \ell(a_k) = \begin{cases} \sum_{k=1}^{|A|-1} p(a_k) k + p(a_{|A|}) (|A|-1) & \text{(prefix code)} \\ {}^2\log |A| & \text{(enumerative code)} \end{cases} \quad (1.3)$$

The value given for the code length  $L$  of the enumerative code, as before, refers to the case where  $|A|$  is some integer power of 2. If this is not true, its code length will be given by the smallest integer following  ${}^2\log |A|$ . Note also that, while the enumerative code can only be used for finite message sets, i.e.  $|A| < \infty$ , the prefix code has no such restriction.

*Example 1.* Let us now turn to a couple of examples to illustrate the above ideas. The first example is a message set  $A$  in which all messages are equally likely to occur:  $p(a) = 1/|A|$  for all  $a \in A$ . Working out the expression in (1.3) for the prefix code gives (with the help of the mathematical tools in Appendix A):

$$L_{\text{pref}} = \frac{1}{|A|} \left\{ \sum_{k=1}^{|A|-1} k + |A| - 1 \right\} = \frac{1}{2} (|A| + 1) - \frac{1}{|A|}$$

For message sets with size equal to some power of two (for simplicity in comparing with the enumerative code), this gives the following results:

$ A $	2	4	8	16	32	$\dots$	$ A  \rightarrow \infty$
$L_{\text{enu}}$	1	2	3	4	5	$\dots$	$\sim {}^2\log  A $
$L_{\text{pref}}$	1	$2\frac{1}{4}$	$4\frac{3}{8}$	$8\frac{7}{16}$	$16\frac{15}{32}$	$\dots$	$\sim \frac{1}{2} A $

It is clear that for this example the prefix code is definitely less efficient, which could have been expected since the prefix code is based on exploiting likelihood differences. In the present example there are no such differences; each message is equally likely to occur.

*Example 2.* As a second example we will choose a message set  $A$  in which the messages are not equally likely to occur, albeit that the likelihood differences are not yet too large:

$$A = \{a_1, \dots, a_{|A|}\} \quad p(a_\ell) = \frac{2}{|A|} \left[ 1 - \frac{\ell}{|A|+1} \right]$$

The elements of the set  $A$  are ordered according to decreasing probability of occurrence; i.e. the message probabilities obey  $p(a_1) \geq p(a_2) \geq \dots \geq p(a_{|A|})$ . They decrease linearly, from  $p(a_1) = 2/(|A|+1)$  to  $p(a_{|A|}) = 2/|A|(|A|+1)$ , such that  $\sum_{\ell=1}^{|A|} p(a_\ell) = 1$  (as it should; probabilities must always add up to 1). Working out the expression in (1.3) for the prefix code now gives (again with the help of the mathematical tools in Appendix A):

$$\begin{aligned} L_{\text{pref}} &= \frac{2}{|A|(|A|+1)} \left\{ (|A|+1) \sum_{k=1}^{|A|-1} k - \sum_{k=1}^{|A|-1} k^2 + |A| - 1 \right\} \\ &= \frac{2}{|A|(|A|+1)} \left\{ \frac{1}{2}|A|(|A|^2-1) - \frac{1}{6}|A|(|A|-1)(2|A|-1) + |A| - 1 \right\} \\ &= \frac{1}{|A|(|A|+1)} \left\{ \frac{1}{3}|A|^3 + |A|^2 + \frac{2}{3}|A| - 2 \right\} = \frac{|A|^3 + 3|A|^2 + 2|A| - 6}{3|A|(|A|+1)} \\ &= \frac{1}{3}|A| + \frac{2}{3} - \frac{2}{|A|(|A|+1)} \end{aligned}$$

For message sets with size equal to some power of two (for simplicity in comparing with the enumerative code), this gives the following results:

$ A $	2	4	8	16	$\dots$	$ A  \rightarrow \infty$
$L_{\text{enu}}$	1	2	3	4	$\dots$	$\sim {}^2\log  A $
$L_{\text{pref}}$	1	$1\frac{9}{10}$	$3\frac{11}{36}$	$5\frac{405}{409}$	$\dots$	$\sim \frac{1}{3} A $

Again the prefix code is generally less efficient, although to a somewhat lesser extent. Apparently for the prefix code to beat the enumerative code the likelihood differences to be exploited need to be significantly larger.

*Example 3.* As a third example we will inspect a message set  $A = \{a_1, \dots, a_{|A|}\}$  in which the message probabilities decrease exponentially:

$$A = \{a_1, \dots, a_{|A|}\} \quad p(a_\ell) = \frac{1}{B} e^{-\lambda \ell} \quad (\lambda > 0)$$

in which the constant  $B$  follows from the normalisation requirement (using the tools in Appendix A to deal with the summations):

$$B = \sum_{\ell=1}^{|A|} e^{-\lambda \ell} = \frac{1 - e^{-|A|\lambda}}{1 - e^{-\lambda}} - 1 = \frac{e^{-\lambda} - e^{-|A|\lambda}}{1 - e^{-\lambda}} = \frac{1 - e^{-\lambda|A|}}{e^{\lambda} - 1} \quad (1.4)$$

Working out expression (1.3), using (1.4), now leads to

$$\begin{aligned}
L_{\text{pref}} &= \frac{1}{B} \left\{ \sum_{k=1}^{|A|-1} e^{-\lambda k} k + e^{-\lambda|A|} (|A|-1) \right\} = \frac{1}{B} \left\{ -\frac{\partial}{\partial \lambda} \sum_{k=1}^{|A|-1} e^{-\lambda k} + e^{-\lambda|A|} (|A|-1) \right\} \\
&= \frac{1}{B} \left\{ -\frac{\partial}{\partial \lambda} [B - e^{-\lambda|A|}] + e^{-\lambda|A|} (|A|-1) \right\} = \frac{1}{B} \left\{ -|A| e^{-\lambda|A|} + e^{-\lambda|A|} (|A|-1) \right\} - \frac{\partial}{\partial \lambda} \log B \\
&= -\frac{1}{B} e^{-\lambda|A|} - \frac{\partial}{\partial \lambda} [\log(1 - e^{-\lambda|A|}) - \log(e^\lambda - 1)] = -e^{-\lambda|A|} \frac{e^\lambda - 1}{1 - e^{-\lambda|A|}} - \left[ \frac{|A| e^{-\lambda|A|}}{1 - e^{-\lambda|A|}} - \frac{e^\lambda}{e^\lambda - 1} \right] \\
&= \frac{e^\lambda}{e^\lambda - 1} - e^{-\lambda|A|} \frac{|A| + e^\lambda - 1}{1 - e^{-\lambda|A|}} = \frac{1}{1 - e^{-\lambda}} - \frac{|A| + e^\lambda - 1}{e^{\lambda|A|} - 1}
\end{aligned}$$

For simplicity we only give the corresponding values for  $|A| = 2$  and  $|A| \rightarrow \infty$  in a table:

$ A $	2	...	$ A  \rightarrow \infty$
$L_{\text{enu}}$	1	...	$\sim {}^2\log A $
$L_{\text{pref}}$	1	...	$\sim 1/(1 - e^{-\lambda})$

Here we see a dramatic difference between enumerative coding and our prefix code. At the same time this convincingly demonstrates that for message sets where the messages are not all equally likely the appropriate measure of information content *cannot be*  ${}^2\log|A|$ . Even in the limit of an infinite number of messages,  $|A| \rightarrow \infty$ , we can still communicate them by using on average only a finite number of bits:

$$\lim_{|A| \rightarrow \infty} L_{\text{pref}} = \frac{1}{1 - e^{-\lambda}}$$

The dependence on  $\lambda$  of this result is consistent with the picture sketched so far. For  $\lambda \rightarrow 0$  (where the messages again tend to become equally likely) we indeed find that the average number of bits needed diverges,  $\lim_{|A| \rightarrow \infty} L_{\text{pref}} = \infty$ . For  $\lambda \rightarrow \infty$  (where  $p_1 \rightarrow 1$  and  $p_{\ell > 1} \rightarrow 0$ , so just one message will be communicated) we find that  $\lim_{|A| \rightarrow \infty} L_{\text{pref}} = 1$ . We just communicate a single bit for messages with zero information content.

### 1.0.3 Questions and Answers

We have seen, by investigating examples of message sets and coding schemes, that the information content of messages must depend on the probabilities of all the various messages that can occur. However, we have not found the precise recipe yet to express the information content in terms of these probabilities. At this stage one is led to the following questions:

**Question 1:** Can we construct a precise and workable definition of information content in terms of codes ?

**Answer:** Yes, we will define the information content of messages from a set  $A$  as the average number of bits used to communicate the messages from set  $A$  if we use *the optimal code* (this optimal code will be different for different message sets).

**Question 2:** What is the resulting expression for the information content of messages from the set  $A = \{a_1, a_2, \dots\}$ ?

**Answer:** The information content of messages from the set  $A$  is given by the *entropy*  $H$ :

$$H = - \sum_{\ell=1}^{|A|} p(a_\ell) {}^2\log p(a_\ell) \quad (1.5)$$

Equation (1.5) is the main statement of information theory. About one half of information theory is concerned with various ways to prove (1.5), whereas the other half deals with working out its consequences. For infinitely large message sets the sum in (1.5) simply extends to infinity. We will prove and investigate this key equation and its consequences in much more detail in subsequent sections. In this introduction I will only simply show what (1.5) predicts for the three specific examples we discussed.

Our first example had uniform message probabilities:  $p(a) = 1/|A|$  for all  $a \in A$ . Here we get precisely the result given by the enumerative code:

$$H = - \sum_{\ell=1}^{|A|} \frac{1}{|A|} {}^2\log \frac{1}{|A|} = {}^2\log |A| \quad (1.6)$$

So the picture is as follows

$ A $	2	4	8	16	32	...	$ A  \rightarrow \infty$
$H$	1	2	3	4	5	...	$\sim {}^2\log  A $
$L_{\text{enu}}$	1	2	3	4	5	...	$\sim {}^2\log  A $
$L_{\text{pref}}$	1	$2\frac{1}{4}$	$4\frac{3}{8}$	$8\frac{7}{16}$	$16\frac{15}{32}$	...	$\sim \frac{1}{2} A $

Our second example was the message set with linearly decreasing probabilities:

$$A = \{a_1, \dots, a_{|A|}\} \quad p(a_\ell) = \frac{2}{|A|} \left[ 1 - \frac{\ell}{|A|+1} \right]$$

Here the entropy (information content according to Shannon) is

$$H = - \frac{2}{|A|} \sum_{\ell=1}^{|A|} \left[ 1 - \frac{\ell}{|A|+1} \right] {}^2\log \left\{ \frac{2}{|A|} \left[ 1 - \frac{\ell}{|A|+1} \right] \right\} \quad (1.7)$$

Asymptotically, i.e. for  $|A| \rightarrow \infty$ , we can replace the sum by an integral, by putting  $\ell = (1-x)|A|$  with  $x \in [0, 1]$  and  $|A|^{-1} \rightarrow dx$ , and use  ${}^2\log z = \log z / \log 2$ :

$$\begin{aligned} \lim_{|A| \rightarrow \infty} \left\{ H - {}^2\log |A| \right\} &= -1 - \lim_{|A| \rightarrow \infty} \frac{2}{|A|} \sum_{\ell=1}^{|A|} \left[ 1 - \frac{\ell}{|A|+1} \right] {}^2\log \left\{ 1 - \frac{\ell}{|A|+1} \right\} \\ &= -1 - \frac{2}{\log 2} \int_0^1 dx \, x \log x = -1 - \frac{2}{\log 2} \int_{-\infty}^0 dz \, z e^{2z} \end{aligned}$$

$$\begin{aligned}
&= -1 + \frac{2}{\log 2} \int_0^\infty dz z e^{-2z} = -1 + \frac{2}{\log 2} \left\{ \left[ -\frac{1}{2} z e^{-2z} \right]_0^\infty + \frac{1}{2} \int_0^\infty dz e^{-2z} \right\} \\
&= -1 + \frac{1}{2 \log 2} = -0.279\dots
\end{aligned}$$

and thus find again that  $H \sim {}^2\log|A|$ .

$ A $	2	4	8	$\dots$	$ A  \rightarrow \infty$
$H$	0.918...	1.846...	2.794...	$\dots$	$\sim {}^2\log A $
$L_{\text{enu}}$	1	2	3	$\dots$	$\sim {}^2\log A $
$L_{\text{pref}}$	1	$1\frac{9}{10}$	$3\frac{11}{36}$	$\dots$	$\sim \frac{1}{3} A $

For our third example, with message probabilities decreasing exponentially, i.e.

$$A = \{a_1, \dots, a_{|A|}\} \quad p(a_\ell) = \frac{e^\lambda - 1}{1 - e^{-\lambda|A|}} e^{-\lambda\ell} \quad (\lambda > 0)$$

we finally get, using  ${}^2\log x = \log x / \log 2$ :

$$\begin{aligned}
H &= -{}^2\log \left[ \frac{e^\lambda - 1}{1 - e^{-\lambda|A|}} \right] + \lambda ({}^2\log e) \frac{e^\lambda - 1}{1 - e^{-\lambda|A|}} \sum_{\ell=1}^{|A|} \ell e^{-\lambda\ell} \\
&= -{}^2\log \left[ \frac{e^\lambda - 1}{1 - e^{-\lambda|A|}} \right] - \frac{\lambda}{\log 2} \frac{e^\lambda - 1}{1 - e^{-\lambda|A|}} \frac{\partial}{\partial \lambda} \sum_{\ell=1}^{|A|} e^{-\lambda\ell} \\
&= -{}^2\log \left[ \frac{e^\lambda - 1}{1 - e^{-\lambda|A|}} \right] - \frac{\lambda}{\log 2} \frac{e^\lambda - 1}{1 - e^{-\lambda|A|}} \frac{\partial}{\partial \lambda} \frac{1 - e^{-\lambda|A|}}{e^\lambda - 1} \\
&= -{}^2\log \left[ \frac{e^\lambda - 1}{1 - e^{-\lambda|A|}} \right] - \frac{\lambda}{\log 2} \frac{\partial}{\partial \lambda} \left[ \log(1 - e^{-\lambda|A|}) - \log(e^\lambda - 1) \right] \\
&= -{}^2\log \left[ \frac{e^\lambda - 1}{1 - e^{-\lambda|A|}} \right] - \frac{\lambda}{\log 2} \left[ \frac{|A| e^{-\lambda|A|}}{1 - e^{-\lambda|A|}} - \frac{e^\lambda}{e^\lambda - 1} \right] \tag{1.8}
\end{aligned}$$

Asymptotically this gives:

$$\lim_{|A| \rightarrow \infty} H = -{}^2\log(e^\lambda - 1) + \frac{\lambda}{\log 2} \frac{e^\lambda}{e^\lambda - 1}$$

This result is shown in figure 1.1, together with the limit  $|A| \rightarrow \infty$  of the code length of the simple prefix code,  $L_{\text{pref}} = 1/(1 - e^{-\lambda})$ , for comparison. We observe that always  $L_{\text{pref}} \geq H$  (as it should). The two graphs appear to touch at  $\lambda = \log 2 = 0.693\dots$ , which is indeed true since

$$\lambda = \log 2 : \quad L_{\text{pref}} = \frac{1}{1 - \frac{1}{2}} = 2 \quad H = -{}^2\log(2-1) + \frac{\log 2}{\log 2} \frac{2}{2-1} = 2$$

Apparently for  $\lambda = \log 2$  the prefix code is optimal.

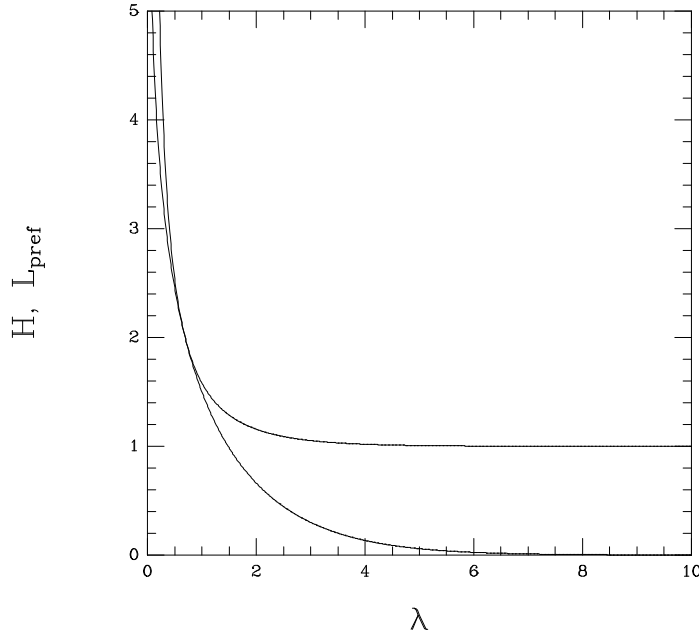


Figure 1.1: Code length  $L_{\text{pref}}$  of simple prefix code (upper graph) versus information content  $H$  (lower graph) of infinitely large message set ( $|A| = \infty$ ) with exponentially decaying message probabilities  $p(a_\ell) \sim e^{-\lambda\ell}$ , as a function of the decay rate  $\lambda$ .

#### 1.0.4 Outlook and References

These lecture notes are organised as follows. I first give a brief revision of the basic concepts and definitions of probability theory, since probabilities and correlations play an important role in information theory, as we have seen. The material covered in this section is brief but self-contained; a more detailed exposé can be found in textbooks like

C.W. Gardiner (1990) *Handbook of Stochastic Methods* (Berlin: Springer)

Next I will cover the basics of Shannon's information theory, based on the following literature (especially the book by Cover and Thomas can be highly recommended):

T.M. Cover and J.A. Thomas (1991) *Elements of Information Theory* (New York: Wiley)

D.J.C. MacKay (1995) *A Short Course in Information Theory* (Cambridge: University Lecture Notes)

Finally I will turn to the applications of information theory to neural network analysis and design. Some of the material to be covered can be found in

G. Deco and D. Obradovic (1996) *An Information-Theoretic Approach to Neural Computing* (London: Springer)

The book by Deco and Obradovic is well-written and clear, but, in contrast with the one by Cover and Thomas, does not give all the proofs of the basic theorems of information theory. However, much is research material which so far has been published in research papers only.

## Chapter 2

# Elementary Probability Concepts

### 2.1 General Theory

Probability theory can be set up in various ways. One route, due to Kolmogorov, starts by defining probabilities in terms of three basic simple axioms, and builds all probability definitions and theorems from there. I will not follow this route in full here, but just show how it works and why it makes sense.

#### 2.1.1 Kolmogorov's Axioms and the Interpretation of Probability

We start off by introducing the notion of 'events'  $\mathbf{x}$ , and the set  $\Omega$  of all possible events. We then associate to each subset  $A \subseteq \Omega$  of events a probability measure  $P(A)$ , which we require to have the following three properties:

(i)  $P(A) \geq 0$  for all  $A \subseteq \Omega$

(ii)  $P(\Omega) = 1$

(iii) If  $\{A_i\}$  ( $i = 1, 2, 3, \dots$ ) is a countable collection of non-overlapping sets  $A_i \subset \Omega$ , i.e.  $A_i \cap A_j = \emptyset$  for all  $i \neq j$ , then  $P(\cup_i A_i) = \sum_i P(A_i)$

There are several advantages to this approach. Firstly, it allows us to deal with discrete variables as well as real variables. The simplest scenario is that of having a discrete (countable) set of events  $\Omega = \{\mathbf{x}_1, \mathbf{x}_2, \dots\}$ . In this case we can equally well restrict ourselves to sets  $A$  which contain single events  $\mathbf{x}$  only, so that instead of the probability measure of a set  $P(A)$  we obtain the more familiar  $p(\mathbf{x})$ . Since  $P(A)$  can be interpreted as the probability of an event having a certain property, namely: to be in the set  $A$ , the quantity  $p(\mathbf{x})$  will be simply the probability that an event *is*  $\mathbf{x}$ . The set of real numbers, on the other hand, is not countable. Therefore for event sets which involve intervals of real numbers, or objects labelled in a continuous way by real numbers, we will have to use sets. Here  $P(A)$  will denote the probability of an event (or its continuous label) having the property of being in a certain *interval*, or higher dimensional continuous set.

Secondly, this set-up allows us to separate that part of probability theory that is exact (i.e. based on precise logical deduction) from that part that is based on our interpretation of probabilities, i.e. on the relation between the mathematical formalism and counting the number of occurrences of real-world events. One can show that the mathematical formalism



can be deduced from the above three axioms, but one can never escape the question ‘but what *is* a probability?’.

In the above set-up the link with real-world observations is usually made via the tentative statement

$$P(A) = \text{the probability that a randomly drawn event } \mathbf{x} \in \Omega \text{ satisfies } \mathbf{x} \in A$$

This, however, just shifts the interpretation problem from giving a meaning to  $P(A)$  to defining what the concepts ‘probability’ and ‘random’ mean. A pragmatic solution is the following. Imagine a system that generates events  $\mathbf{x} \in \Omega$  sequentially (e.g. a physical experiment like throwing a dice, like counting the number of air molecules in an open box, or like counting the daily number of accidents on a particular section of motorway), giving the infinite series  $\mathbf{x}_1, \mathbf{x}_2, \dots$ . We check for each event whether or not it is in set  $A$ . We allow for an arbitrary re-ordering/selection operation  $\pi : \{1, 2, \dots\} \rightarrow \{1, 2, \dots\}$  with  $\pi(n) \neq \pi(\ell)$  for all  $n \neq \ell$ . For the first  $M$  re-ordered elements  $\{\mathbf{x}_{\pi(1)}, \dots, \mathbf{x}_{\pi(M)}\}$  we calculate the fraction  $f_M(A)$  of events that turn out to be in  $A$ :

$$f_M(A) = \frac{1}{M} \sum_{m=1}^M I_A(\mathbf{x}_{\pi(m)}) \quad \begin{cases} I_A(\mathbf{x}) = 1 & \text{if } \mathbf{x} \in A \\ I_A(\mathbf{x}) = 0 & \text{if } \mathbf{x} \notin A \end{cases}$$

We then define *random events* as those generated by a system with the property that for *each* set  $A \subseteq \Omega$  and each ordering  $\pi$  the fraction  $f_M(A)$  tends to a constant in the limit  $M \rightarrow \infty$ . This constant will be then be defined as the ‘probability measure’ of the set  $A$ :

$$\forall A \subseteq \Omega, \forall \pi : \quad \lim_{M \rightarrow \infty} f_M(A) = P(A)$$

It is not clear whether purely random events really exist in nature, but since many event sets meet the above criterion at least to a high degree, probability theory is a powerful tool.

The above interpretation clarifies the need for Kolmogorov’s three axioms. The simple axioms (i) and (ii) come out trivially:

$$\begin{aligned} f_M(A) \geq 0 &\Rightarrow P(A) = \lim_{M \rightarrow \infty} f_M(A) \geq 0 \\ f_M(\Omega) = 1 &\Rightarrow P(\Omega) = \lim_{M \rightarrow \infty} f_M(\Omega) = 1 \end{aligned}$$

To arrive at the third axiom we apply the above interpretation recipe to the situation of having two non-overlapping sets  $A \subset \Omega$  and  $B \subset \Omega$ ,  $A \cap B = \emptyset$ :

$$\begin{cases} I_A(\mathbf{x}) + I_B(\mathbf{x}) = 1 & \text{if } \mathbf{x} \in A \cup B \\ I_A(\mathbf{x}) + I_B(\mathbf{x}) = 0 & \text{if } \mathbf{x} \notin A \cup B \end{cases} \Rightarrow I_A(\mathbf{x}) + I_B(\mathbf{x}) = I_{A \cup B}(\mathbf{x})$$

from which we deduce

$$P(A) + P(B) = \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{m=1}^M [I_A(\mathbf{x}_{\pi(m)}) + I_B(\mathbf{x}_{\pi(m)})] = \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{m=1}^M I_{A \cup B}(\mathbf{x}_{\pi(m)}) = P(A \cup B)$$

We can now immediately deduce  $P(\cup_{i < m} A_i) + P(A_m) = P((\cup_{i < m} A_i) \cup A_m) = P(\cup_{i \leq m} A_i)$ , from which axiom (iii) follows by induction. Clearly axioms (i, ii, iii) are necessary (one can show that further reduction is not possible), Kolmogorov showed that they are also sufficient.

### 2.1.2 Theorems on Sets

Here we give some simple theorems on sets, for future use. Their validity follows either from graphical inspection (using diagrams) or from formal proofs (which is what we will use here). We will often use the abbreviation ‘iff’ for ‘if-and-only-if’:

**theorem:**

$$C \cup (A \cap B) = (C \cup A) \cap (C \cup B) \quad (2.1)$$

**proof:**

$$\begin{aligned} \text{(a) } x \in C &\Rightarrow x \in (C \cup A) \text{ and } x \in (C \cup B) \\ &\Rightarrow x \in (C \cup A) \cap (C \cup B) \\ \text{(b) } x \in (A \cap B) &\Rightarrow x \in (C \cup A) \text{ and } x \in (C \cup B) \\ &\Rightarrow x \in (C \cup A) \cap (C \cup B) \\ \text{(c) } x \notin C \text{ and } x \notin (A \cap B) &\Rightarrow \begin{cases} x \in (C \cup A) \text{ iff } x \in A \\ x \in (C \cup B) \text{ iff } x \in B \end{cases} \\ &\Rightarrow x \in (C \cup A) \cap (C \cup B) \text{ iff } (x \in A \text{ and } x \in B) \\ &\Rightarrow x \notin (C \cup A) \cap (C \cup B) \end{aligned}$$

From (a)+(b) and (c) follow, respectively:

$$\begin{aligned} x \in C \cup (A \cap B) &\Rightarrow x \in (C \cup A) \cap (C \cup B) \\ x \notin C \cup (A \cap B) &\Rightarrow x \notin (C \cup A) \cap (C \cup B) \end{aligned}$$

which completes the proof.  $\square$

**theorem:**

$$(A \cap B) \cup (A \cap C) = A \cap (B \cup C) \quad (2.2)$$

**proof:**

$$\begin{aligned} (A \cap C) \cup (A \cap B) &= ((A \cap C) \cup A) \cap ((A \cap C) \cup B) \quad (\text{put } C \rightarrow A \cap C \text{ in (2.1)}) \\ &= A \cap ((A \cap C) \cup B) \quad (\text{using } (A \cap C) \cup A = A) \end{aligned}$$

We now find:

$$\begin{aligned} x \in A &\Rightarrow x \in (A \cap C) \cup (A \cap B) \text{ iff } x \in (A \cap C) \cup B \\ &\Rightarrow x \in (A \cap C) \cup (A \cap B) \text{ iff } (x \in A \cap C \text{ or } x \in B) \\ &\Rightarrow x \in (A \cap C) \cup (A \cap B) \text{ iff } (x \in C \text{ or } x \in B) \\ &\Rightarrow x \in (A \cap C) \cup (A \cap B) \text{ iff } x \in B \cup C \\ x \notin A &\Rightarrow x \notin (A \cap C) \cup (A \cap B) \end{aligned}$$

from which it follows that

$$(A \cap C) \cup (A \cap B) = \{x \in A \mid x \in B \cup C\} = A \cap (B \cup C)$$

$\square$

**theorem:** If  $\{B_i\}$  are disjoint sets,  $B_i \cap B_j = \emptyset$  if  $i \neq j$ , such that  $\cup_i B_i = \Omega$ , then

$$\cup_i (A \cap B_i) = A \quad (2.3)$$

**proof:**

$$\begin{aligned} (A \cap B_1) \cup (A \cap B_2) &= A \cap (B_1 \cup B_2) && \text{(using (2.2))} \\ (A \cap B_1) \cup (A \cap B_2) \cup (A \cap B_3) &= A \cap (B_1 \cup B_2 \cup B_3) && \text{(using (2.2) and previous line)} \end{aligned}$$

induction :

$$\cup_i (A \cap B_i) = A \cap (\cup_i B_i)$$

By using  $\cup_i B_i = \Omega$  and  $A \cap \Omega = A$  we obtain the theorem.  $\square$

### 2.1.3 Joint Probability, Conditional Probability and Independence

Given two sets  $A \subseteq \Omega$  and  $B \subseteq \Omega$  we define the so-called joint probability as  $P(A \cap B)$ , and the conditional probabilities  $P(A|B)$  and  $P(B|A)$  as

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad P(B|A) = \frac{P(A \cap B)}{P(A)} \quad (2.4)$$

Equivalently:  $P(A \cap B) = P(A|B)P(B) = P(B|A)P(A)$ . Our previous theorems on sets can now be used to establish theorems on joint and conditional probabilities:

**theorem:** If  $\{B_i\}$  are disjoint sets,  $B_i \cap B_j = \emptyset$  if  $i \neq j$ , such that  $\cup_i B_i = \Omega$ , then

$$\sum_i P(A \cap B_i) = P(A) \quad (2.5)$$

**proof:** The sets  $A \cap B_i$  satisfy the conditions of Kolmogorov's axiom (iii), since  $(A \cap B_i) \cap (A \cap B_j) = \emptyset$  for all  $i \neq j$ . Therefore

$$\begin{aligned} \sum_i P(A \cap B_i) &= P(\cup_i (A \cap B_i)) && \text{(axiom (iii))} \\ &= P(A) && \text{(using (2.3))} \end{aligned} \quad \square$$

This theorem (2.5) clearly makes sense. The sets  $B_i$  simply divide the set  $A$  in disjoint sub-sets; the probability measure of  $A$  is then the sum of the sub-set contributions. In terms of the conditional probabilities of (2.4) theorem (2.5) reads:

**theorem:** If  $\{B_i\}$  are disjoint sets,  $B_i \cap B_j = \emptyset$  if  $i \neq j$ , such that  $\cup_i B_i = \Omega$ , then

$$\forall A \subseteq \Omega : \quad \sum_i P(A|B_i)P(B_i) = P(A) \quad (2.6)$$

**proof:** Combination of (2.4) (left equation) and (2.5).  $\square$

**theorem:** If  $\{B_i\}$  are disjoint sets,  $B_i \cap B_j = \emptyset$  if  $i \neq j$ , such that  $\cup_i B_i = \Omega$ , then

$$\forall A \subseteq \Omega : \quad \sum_i P(B_i|A) = 1 \quad (2.7)$$

**proof:** Combination of (2.4) (right equation) and (2.5).  $\square$

We can now define the notion of independent event sets, without as yet using probability interpretations. In the case of having just two sets  $A$  and  $B$  the definition is simple: factorisation of the joint probability distribution. In the case of more than two sets the definition is somewhat more elaborate:

$$\{A, B\} \text{ are independent : } P(A \cap B) = P(A)P(B) \quad (2.8)$$

$$\{A_1, \dots, A_n\} \text{ are independent : } \begin{array}{l} P(A_{\ell_1} \cap A_{\ell_2} \cap \dots \cap A_{\ell_k}) = P(A_{\ell_1})P(A_{\ell_2}) \dots P(A_{\ell_k}) \\ \text{for every set } \{\ell_1, \dots, \ell_k\} \subseteq \{1, \dots, n\} \text{ with } 1 < k \leq n \end{array} \quad (2.9)$$

Note that for  $n = 2$  we recover definition (2.8) as a special case from (2.9), as it should. The implicit statement in definition (2.9) is that requiring the factorisation  $P(A_1 \cap \dots \cap A_n) = P(A_1) \dots P(A_n)$ , although necessary, is not sufficient to have independence as soon as  $n \geq 3$ .

To illustrate the need for definition (2.9), rather than requiring factorisation either of the full joint probability distribution only, or of the various pair probabilities only, let us work out a few simple examples of events representing the outcome of throwing a fair dice. Although having a link with a real experiment is not necessary for the argument, it shows how the theory formalises our intuitive notion of independence.

*Example 1.* We consider the experiment of throwing a fair six-sided dice once. This implies that  $\Omega = \{1, 2, 3, 4, 5, 6\}$  (six sides) and that  $P(A) = \frac{1}{6}|A|$  (the dice being fair). We now investigate the independence of the following event sets:  $A_1 = \{1, 2\}$ ,  $A_2 = \{2, 4, 6\}$ ,  $A_3 = \{1, 2, 3, 4, 5, 6\}$ . To do so we have to check whether the joint probabilities of each selection of  $k$  event sets from the trio  $\{A_1, A_2, A_3\}$  factorise, where  $1 < k \leq 3$ :

$$\begin{array}{ll} k = 2 : & \begin{array}{ll} P(A_1 \cap A_2) = P(\{2\}) = \frac{1}{6} & P(A_1)P(A_2) = \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{6} \\ P(A_1 \cap A_3) = P(A_1) = \frac{1}{3} & P(A_1)P(A_3) = \frac{1}{3} \cdot 1 = \frac{1}{3} \\ P(A_2 \cap A_3) = P(A_2) = \frac{1}{2} & P(A_2)P(A_3) = \frac{1}{2} \cdot 1 = \frac{1}{2} \end{array} \\ k = 3 : & P(A_1 \cap A_2 \cap A_3) = P(\{2\}) = \frac{1}{6} \quad P(A_1)P(A_2)P(A_3) = \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{6} \end{array}$$

Clearly the sets  $\{A_1, A_2, A_3\}$  are independent.

*Example 2.* We again consider the experiment of throwing a fair six-sided dice once, but now with different event sets:  $A_1 = \{1, 2\}$ ,  $A_2 = \{2, 4, 6\}$ ,  $A_3 = \{3, 4\}$ . Now we find:

$$\begin{array}{ll} k = 2 : & \begin{array}{ll} P(A_1 \cap A_2) = P(\{2\}) = \frac{1}{6} & P(A_1)P(A_2) = \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{6} \\ P(A_1 \cap A_3) = P(\emptyset) = 0 & P(A_1)P(A_3) = \frac{1}{3} \cdot \frac{1}{3} = \frac{1}{9} \\ P(A_2 \cap A_3) = P(\{4\}) = \frac{1}{6} & P(A_2)P(A_3) = \frac{1}{2} \cdot \frac{1}{3} = \frac{1}{6} \end{array} \\ k = 3 : & P(A_1 \cap A_2 \cap A_3) = P(\emptyset) = 0 \quad P(A_1)P(A_2)P(A_3) = \frac{1}{3} \cdot \frac{1}{2} \cdot \frac{1}{3} = \frac{1}{18} \end{array}$$

Clearly the sets  $\{A_1, A_2, A_3\}$  are dependent; neither the  $k = 3$  joint probabilities factorise, nor do all  $k = 2$  joint probabilities.

*Example 3.* Let us now move on to an experiment where a fair eight-sided dice is thrown once. Now  $\Omega = \{1, 2, 3, 4, 5, 6, 7, 8\}$  (eight sides) and  $P(A) = \frac{1}{8}|A|$  (the dice being fair). We

investigate the independence of the following event sets:  $A_1 = \{1, 2, 3, 4\}$ ,  $A_2 = \{3, 4, 5, 6\}$ ,  $A_3 = \{1, 2, 5, 6\}$ .

$$\begin{aligned}
 k = 2 : \quad & P(A_1 \cap A_2) = P(\{3, 4\}) = \frac{1}{4} & P(A_1)P(A_2) &= \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \\
 & P(A_1 \cap A_3) = P(\{1, 2\}) = \frac{1}{4} & P(A_1)P(A_3) &= \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \\
 & P(A_2 \cap A_3) = P(\{5, 6\}) = \frac{1}{4} & P(A_2)P(A_3) &= \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \\
 k = 3 : \quad & P(A_1 \cap A_2 \cap A_3) = P(\emptyset) = 0 & P(A_1)P(A_2)P(A_3) &= \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{8}
 \end{aligned}$$

The sets  $\{A_1, A_2, A_3\}$  are not independent, in spite of the factorisation of all joint probabilities involving pairs of sets.

*Example 4.* Our final example is as before the experiment of throwing a fair eight-sided dice once, but now with the event sets  $A_1 = \{1, 2, 3, 4\}$ ,  $A_2 = \{3, 4, 5, 6\}$ ,  $A_3 = \{1, 3, 7, 8\}$ .

$$\begin{aligned}
 k = 2 : \quad & P(A_1 \cap A_2) = P(\{3, 4\}) = \frac{1}{4} & P(A_1)P(A_2) &= \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \\
 & P(A_1 \cap A_3) = P(\{1, 3\}) = \frac{1}{4} & P(A_1)P(A_3) &= \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \\
 & P(A_2 \cap A_3) = P(\{3\}) = \frac{1}{8} & P(A_2)P(A_3) &= \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \\
 k = 3 : \quad & P(A_1 \cap A_2 \cap A_3) = P(\{3\}) = \frac{1}{8} & P(A_1)P(A_2)P(A_3) &= \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{8}
 \end{aligned}$$

The sets  $\{A_1, A_2, A_3\}$  are not independent, in spite of the factorisation the full joint probability  $P(A_1 \cap A_2 \cap A_3)$ .

Apparently it is not sufficient to require factorisation of all joint probabilities involving pairs of event sets, neither is it sufficient to require factorisation of the joint probabilities involving all event sets; the above examples show explicitly that all combinations can occur.

## 2.2 Discrete Event Sets

Discrete event sets are those where the set  $\Omega$  of all possible events is discrete and countable. We will restrict ourselves to events  $\mathbf{x}$  which can be represented by finite-dimensional vectors, i.e.

$$\Omega = \{\mathbf{x}^1, \mathbf{x}^2, \dots\} \subset \mathfrak{R}^n \quad \mathbf{x}^\ell = (x_1^\ell, \dots, x_n^\ell)$$

Since  $\Omega$  is a discrete set, each component  $x_i$  of an event  $\mathbf{x}$  can in turn only assume values from a discrete set  $\Omega_i$ , so  $\Omega \subseteq \Omega_1 \otimes \Omega_2 \otimes \dots \otimes \Omega_n$ . Next we will make explicit choices for the event sets  $A$  in terms of which our fundamental axioms and theorems were formulated. Let us define

$$\begin{array}{lll}
 A_i(x_i) = & \{\mathbf{y} \in \Omega \mid y_i = x_i\} & x_i \in \Omega_i \\
 A_{ij}(x_i, x_j) = & \{\mathbf{y} \in \Omega \mid y_i = x_i \wedge y_j = x_j\} & (x_i, x_j) \in \Omega_i \otimes \Omega_j \\
 A_{ijk}(x_i, x_j, x_k) = & \{\mathbf{y} \in \Omega \mid y_i = x_i \wedge y_j = x_j \wedge y_k = x_k\} & (x_i, x_j, x_k) \in \Omega_i \otimes \Omega_j \otimes \Omega_k \\
 \vdots & \vdots & \vdots \\
 A_{1\dots n}(x_1, \dots, x_n) = & \{\mathbf{x}\} \cap \Omega & \mathbf{x} \in \Omega_1 \otimes \dots \otimes \Omega_n
 \end{array}$$

The probability measure  $P(A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k}))$  associated with such sets has the following interpretation in terms of an associated random process generating an ordered series of

random events  $\mathbf{y}^1, \mathbf{y}^2, \dots$  from  $\Omega$ :

$$\lim_{M \rightarrow \infty} \frac{1}{M} \sum_{m=1}^M I_{A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k})}(\mathbf{y}^m) = P(A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k})) \quad (2.10)$$

In a similar fashion we now obtain the expressions for joint probabilities of two such sets, which, in turn, generate the definition of the conditional probabilities:

$$\begin{aligned} P(A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k}) \cap A_{j_1, \dots, j_\ell}(x_{j_1}, \dots, x_{j_\ell})) &= P(A_{i_1, \dots, i_k, j_1, \dots, j_\ell}(x_{i_1}, \dots, x_{i_k}, x_{j_1}, \dots, x_{j_\ell})) \\ &= P(\{\mathbf{y} \in \Omega \mid y_{i_1} = x_{i_1} \wedge \dots \wedge y_{i_k} = x_{i_k} \wedge y_{j_1} = x_{j_1} \wedge \dots \wedge y_{j_\ell} = x_{j_\ell}\}) \\ P(A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k}) \mid A_{j_1, \dots, j_\ell}(x_{j_1}, \dots, x_{j_\ell})) &= \frac{P(A_{i_1, \dots, i_k, j_1, \dots, j_\ell}(x_{i_1}, \dots, x_{i_k}, x_{j_1}, \dots, x_{j_\ell}))}{P(A_{j_1, \dots, j_\ell}(x_{j_1}, \dots, x_{j_\ell}))} \end{aligned}$$

Next we apply theorem (2.5). The set  $A$  in (2.5) will be one of the  $A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k})$ ; the role of the partitioning family  $\{B_i\}$  in (2.5) will be played by a group of  $A_{j_1, \dots, j_\ell}(x_{j_1}, \dots, x_{j_\ell})$  with various  $(x_{j_1}, \dots, x_{j_\ell}) \in \Omega_{j_1} \otimes \dots \otimes \Omega_{j_\ell}$ , where  $\{i_1, \dots, i_k\} \subseteq \{j_1, \dots, j_\ell\}$ . In words:  $A$  is the set of events for which the components  $\{i_1, \dots, i_k\}$  have specific values, the partitioning sets  $B_i$  are the sets of events where even more components are specified, namely  $\{j_1, \dots, j_\ell\}$ . Note that partitioning sets with different values of  $(x_{j_1}, \dots, x_{j_\ell})$  are non-overlapping (since no event component can have multiple values), and that

$$\text{for any given } \{j_1, \dots, j_\ell\} : \quad \bigcup_{(x_{j_1}, \dots, x_{j_\ell}) \in \Omega_{j_1} \otimes \dots \otimes \Omega_{j_\ell}} A_{j_1, \dots, j_\ell}(x_{j_1}, \dots, x_{j_\ell}) = \Omega$$

(since *any* value of  $(x_{j_1}, \dots, x_{j_\ell})$  will at some point occur in the above union). This also implies (according to the third axiom):

$$\text{for any given } \{j_1, \dots, j_\ell\} : \quad \sum_{(x_{j_1}, \dots, x_{j_\ell}) \in \Omega_{j_1} \otimes \dots \otimes \Omega_{j_\ell}} P(A_{j_1, \dots, j_\ell}(x_{j_1}, \dots, x_{j_\ell})) = 1$$

We conclude that the conditions for theorem (2.5) are met, so that

$$\sum_{(x_{j_1}, \dots, x_{j_\ell}) \in \Omega_{j_1} \otimes \dots \otimes \Omega_{j_\ell}} P(A_{i_1, \dots, i_k, j_1, \dots, j_\ell}(x_{i_1}, \dots, x_{i_k}, x_{j_1}, \dots, x_{j_\ell})) = P(A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k}))$$

It will be clear at this stage that somewhat more compact notation rules would not be unwelcome. If the context is such that no confusion can arise, the following short-hands will be used:

$$P(A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k})) \rightarrow p(x_{i_1}, \dots, x_{i_k}) = P(\{\mathbf{y} \in \Omega \mid y_{i_1} = x_{i_1} \wedge \dots \wedge y_{i_k} = x_{i_k}\})$$

In the case where  $\{i_1, \dots, i_k\} = \{1, \dots, n\}$  we will simply put  $p(x_1, \dots, x_n) = p(\mathbf{x})$ . We will also drop any explicit mentioning of the set  $\Omega_i$  wherever possible: any expression of the form  $\sum_{x_i}$  will mean  $\sum_{x_i \in \Omega_i}$  (and similarly for unions or products). No problems will arise as long as the arguments of  $p(\dots)$  are symbols; as soon as we *evaluate* such expressions for explicit



Next we apply theorem (2.5). The set  $A$  in (2.5) will be  $A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k}; x_{i_1}, \dots, x_{i_k})$ ; the partitioning family  $\{B\}$  in (2.5) will be a group of  $A_{j_1, \dots, j_\ell}(x_{j_1}, \dots, x_{j_\ell}; dx_{j_1}, \dots, dx_{j_\ell})$  with different intervals characterised by  $(x_{j_1}, \dots, x_{j_\ell}; dx_{j_1}, \dots, dx_{j_\ell})$ , and where  $\{i_1, \dots, i_k\} \subseteq \{j_1, \dots, j_\ell\}$ . So  $A$  is the set of events for which the components  $\{i_1, \dots, i_k\}$  are in specific intervals; the partitioning sets  $\{B\}$  are sets of events where even more components are specified to be in certain intervals, namely  $\{j_1, \dots, j_\ell\}$ . We will have to choose the intervals of the partitioning sets such that they are (i) disjunct and (ii) together span the full set  $\Omega$ :

$$\begin{aligned} \mathbf{x} \neq \mathbf{x}' : \quad & A_{j_1, \dots, j_\ell}(x_{j_1}, \dots, x_{j_\ell}; dx_{j_1}, \dots, dx_{j_\ell}) \cap A_{j_1, \dots, j_\ell}(x'_{j_1}, \dots, x'_{j_\ell}; dx'_{j_1}, \dots, dx'_{j_\ell}) = \emptyset \\ & \bigcup_{(x_{j_1}, \dots, x_{j_\ell}; dx_{j_1}, \dots, dx_{j_\ell})} A_{j_1, \dots, j_\ell}(x_{j_1}, \dots, x_{j_\ell}; dx_{j_1}, \dots, dx_{j_\ell}) = \Omega \end{aligned}$$

This also implies (according to the third axiom), at least for our specific choice made for the partitioning intervals:

$$\sum_{(x_{j_1}, \dots, x_{j_\ell}; dx_{j_1}, \dots, dx_{j_\ell})} P(A_{j_1, \dots, j_\ell}(x_{j_1}, \dots, x_{j_\ell}; dx_{j_1}, \dots, dx_{j_\ell})) = 1$$

We conclude that the conditions for theorem (2.5) are met, so that

$$\begin{aligned} & \sum_{(x_{j_1}, \dots, x_{j_\ell}; dx_{j_1}, \dots, dx_{j_\ell})} P(A_{i_1, \dots, i_k, j_1, \dots, j_\ell}(x_{i_1}, \dots, x_{i_k}, x_{j_1}, \dots, x_{j_\ell}; dx_{i_1}, \dots, dx_{i_k}, dx_{j_1}, \dots, dx_{j_\ell})) \\ & = P(A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k}; dx_{i_1}, \dots, dx_{i_k})) \end{aligned}$$

Again somewhat more compact notation rules would not be unwelcome. In addition we want to move from finite intervals to infinitesimally small intervals. If the context is such that no confusion can arise due to the more compact notation, and if the relevant limits exist, the following object will be introduced:

$$p(x_{i_1}, \dots, x_{i_k}) = \lim_{(dx_{i_1}, \dots, dx_{i_k}) \rightarrow \mathbf{0}} \frac{P(A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k}; dx_{i_1}, \dots, dx_{i_k}))}{dx_{i_1} \dots dx_{i_k}} \quad (2.17)$$

It represents a *probability density*, i.e. a probability measure per unit volume in event space  $\Omega$ . The above limit exists if for infinitesimally small hypercubes in  $\Omega$  the probability associated with such hypercubes is proportional to their size. With this compact notation we can again translate our definitions and identities into a more familiar form. Here, however, we have to take extra care in doing the bookkeeping of the infinitesimal interval sizes, in order for the various limits to exist. We simply define

$$\begin{aligned} p(x_{i_1}, \dots, x_{i_k} | x_{j_1}, \dots, x_{j_\ell}) = \\ \lim_{(dx_{i_1}, \dots, dx_{i_k}) \rightarrow \mathbf{0}} \frac{P(A_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k}; dx_{i_1}, \dots, dx_{i_k}) | A_{j_1, \dots, j_\ell}(x_{j_1}, \dots, x_{j_\ell}; dx_{j_1}, \dots, dx_{j_\ell}))}{dx_{i_1} \dots dx_{i_k}} \end{aligned} \quad (2.18)$$



Using the definition of conditional probability for sets, and using the definition (2.17) this subsequently turns out to give the familiar expression

$$p(x_{i_1}, \dots, x_{i_k} | x_{j_1}, \dots, x_{j_\ell}) = \frac{p(x_{i_1}, \dots, x_{i_k}, x_{j_1}, \dots, x_{j_\ell})}{p(x_{j_1}, \dots, x_{j_\ell})}$$

which now, however, involves probability densities. The normalisation of objects such as (2.17) and (2.18) can be derived by application of (2.17) and (2.18) to sets from the partitioning family used in the previous paragraph. Due to their properties of being disjoint and of spanning the event space, in the limit of infinitesimally small interval sizes a union over this family converts in a natural way into an integral, and we find the following identities:

$$\int_{\Omega_{i_1} \otimes \dots \otimes \Omega_{i_k}} dx_{i_1} \dots dx_{i_k} p(x_{i_1}, \dots, x_{i_k}) = 1 \quad (2.19)$$

$$\int_{\Omega_{i_1} \otimes \dots \otimes \Omega_{i_k}} dx_{i_1} \dots dx_{i_k} p(x_{i_1}, \dots, x_{i_k} | x_{j_1}, \dots, x_{j_\ell}) = 1 \quad (2.20)$$

$$\int_{\Omega_{j_1} \otimes \dots \otimes \Omega_{j_\ell}} dx_{j_1} \dots dx_{j_\ell} p(x_{i_1}, \dots, x_{i_k} | x_{j_1}, \dots, x_{j_\ell}) p(x_{j_1}, \dots, x_{j_\ell}) = p(x_{i_1}, \dots, x_{i_k}) \quad (2.21)$$

in which, due to its definition,  $p(x_{i_1}, \dots, x_{i_k})$  is automatically zero when there is no element  $\mathbf{y} \in \Omega$  with the specific components  $(y_{i_1}, \dots, y_{i_k}) = (x_{i_1}, \dots, x_{i_k})$ . Finally we obtain again the usual expression for statistical independence, albeit that here always probability densities are involved, rather than discrete probabilities:

$$\begin{aligned} \{x_1, \dots, x_n\} \text{ are independent :} \quad & p(x_{i_1}, \dots, x_{i_k}) = p(x_{i_1})p(x_{i_2}) \dots p(x_{i_k}) \\ & \text{for every set } \{i_1, \dots, i_k\} \subseteq \{1, \dots, n\} \text{ with } 1 < k \leq n \end{aligned} \quad (2.22)$$

## 2.4 Random Variables and Averages

### 2.4.1 Terminology

Random variables are defined as arbitrary functions  $F(\mathbf{x})$  of random events. For discrete event sets we define *averages* or *expectation values* or *mean values*  $\langle F(\mathbf{x}) \rangle$  of random variables  $F(\mathbf{x})$  as follows:

$$\langle F(\mathbf{x}) \rangle = \sum_{\mathbf{x} \in \Omega} p(\mathbf{x}) F(\mathbf{x}) \quad (2.23)$$

We use the link between probabilities and observations to obtain the following familiar results: for an associated random process generating an ordered series of events  $\mathbf{y}^1, \mathbf{y}^2, \dots$  from  $\Omega$  we find

$$\langle F(\mathbf{x}) \rangle = \lim_{M \rightarrow \infty} \sum_{\mathbf{x} \in \Omega} F(\mathbf{x}) \frac{1}{M} \left[ \text{number of } \mathbf{y} \text{ in } \{\mathbf{y}^1, \dots, \mathbf{y}^M\} \text{ with } \mathbf{y} = \mathbf{x} \right] \quad (2.24)$$

For continuous event sets we define expectation values as

$$\langle F(\mathbf{x}) \rangle = \int_{\Omega} d\mathbf{x} p(\mathbf{x}) F(\mathbf{x}) \quad (2.25)$$

involving a probability density. In order to use the link between probability *densities* and observations, we here have to sample the event space in a discrete way first, followed by sending the sizes  $d\mathbf{x}$  of the bins to zero. For an associated random process generating an ordered series of events  $\mathbf{y}^1, \mathbf{y}^2, \dots$  from  $\Omega$  we then get

$$\langle F(\mathbf{x}) \rangle = \lim_{d\mathbf{x} \rightarrow \mathbf{0}} \sum_{\mathbf{x}} F(\mathbf{x})$$

$$\lim_{M \rightarrow \infty} \frac{1}{M} \left[ \text{number of } \mathbf{y} \text{ in } \{\mathbf{y}^1, \dots, \mathbf{y}^M\} \text{ with } \mathbf{y} \in [x_1, x_1 + dx_1] \otimes \dots [x_n, x_n + dx_n] \right] \quad (2.26)$$

Let us now turn to the definitions and basic properties of a couple of relevant random variables, with events  $\mathbf{x} \in \Omega \subseteq \mathfrak{R}^n$  (discrete or continuous). Note that, except for the special case of having a discrete set  $\Omega$  of just a finite number of events, in general one cannot be sure beforehand (without explicit proof) that the following averages will actually exist (i.e. are finite):

**average:**

$$\mu_i = \langle x_i \rangle \quad (2.27)$$

**variance:**

$$\sigma_i^2 = \langle x_i^2 \rangle - \langle x_i \rangle^2 \quad (2.28)$$

$\sigma_i^2$  is always non-negative, since it can be rewritten as  $\langle x_i^2 \rangle - \langle x_i \rangle^2 = \langle [x_i - \langle x_i \rangle]^2 \rangle$ . This also shows that  $\sigma_i^2 = 0$  implies that  $x_i = x'_i$  for any two events  $\mathbf{x} \in \Omega$  and  $\mathbf{x}' \in \Omega$  with nonzero probabilities.

**covariance matrix:**

$$C_{ij} = \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle \quad (2.29)$$

Note that  $C_{ii} = \sigma_i^2$ . The covariance matrix is symmetric, therefore all eigenvalues are real. It is non-negative definite (so all eigenvalues are non-negative), since it can be written as  $\langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle = \langle (x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) \rangle$ , from which it follows that for any  $\mathbf{z} \in \mathfrak{R}^n$ :

$$\mathbf{z} \cdot \mathbf{C} \mathbf{z} = \left\langle \left[ \sum_{i=1}^n z_i [x_i - \langle x_i \rangle] \right]^2 \right\rangle \geq 0$$

**moments:**

$$\langle x_{i_1}^{m_{i_1}} x_{i_2}^{m_{i_2}} \dots x_{i_k}^{m_{i_k}} \rangle \quad m_{i_t} \in \{0, 1, 2, 3, \dots\} \quad (2.30)$$

**characteristic function:**

$$\forall \mathbf{k} \in \mathfrak{R}^n : \quad \phi(\mathbf{k}) = \langle e^{i\mathbf{k} \cdot \mathbf{x}} \rangle \quad (2.31)$$

The characteristic function always exists, since

$$|\phi(\mathbf{k})| = |\langle e^{i\mathbf{k} \cdot \mathbf{x}} \rangle| \leq \langle |e^{i\mathbf{k} \cdot \mathbf{x}}| \rangle = 1$$

Note that  $\phi(\mathbf{0}) = 1$ . If the moment  $\langle x_{i_1}^{m_{i_1}} \dots x_{i_k}^{m_{i_k}} \rangle$  exists, and  $\phi(\mathbf{k})$  is sufficiently many times continuously differentiable, then

$$\langle x_{i_1}^{m_{i_1}} \dots x_{i_k}^{m_{i_k}} \rangle = \lim_{\mathbf{k} \rightarrow \mathbf{0}} \left[ \left( \frac{1}{i} \frac{\partial}{\partial k_{i_1}} \right)^{m_{i_1}} \dots \left( \frac{1}{i} \frac{\partial}{\partial k_{i_k}} \right)^{m_{i_k}} \right] \phi(\mathbf{k})$$

The proof of this identity only requires substituting the definition of  $\phi(\mathbf{k})$  and working out the various partial derivatives. Provided it exists, one can see  $\phi(\mathbf{k})$  as the Fourier transform of the probability distribution  $p(\mathbf{x})$ . If, for instance, a probability density  $p(\mathbf{x})$  is in the function space  $L^2(\Omega)$  (which means that  $\int_{\Omega} d\mathbf{x} |p(\mathbf{x})|^2 < \infty$ ), the Fourier transform exists, and  $p(\mathbf{x})$  is related to  $\phi(\mathbf{k})$  in a one-to-one manner, via

$$\phi(\mathbf{k}) = \int d\mathbf{x} p(\mathbf{x}) e^{i\mathbf{k} \cdot \mathbf{x}} \quad p(\mathbf{x}) = \int \frac{d\mathbf{k}}{(2\pi)^n} \phi(\mathbf{k}) e^{-i\mathbf{k} \cdot \mathbf{x}}$$

where we have put  $p(\mathbf{x}) = 0$  for all  $\mathbf{x} \notin \Omega$ .

### 2.4.2 Central Limit Theorem

Gaussian probability distributions owe their popularity to the central limit theorem, which states that under a wide range of conditions, a sum of  $n$  independent random variables  $x_i$  will in the limit  $n \rightarrow \infty$  itself become a random variable with a Gaussian probability distribution. Finding out precisely which conditions are sufficient and necessary to arrive at this property would take up an entire one-semester course, I will here only show how the property arises under conditions which in principle are in fact too restrictive (i.e. not all assumptions I will make are strictly speaking necessary).

Let us assume we have  $n$  independent random variables  $(x_1, \dots, x_n)$ , each  $x_i$  described by a probability distribution  $p_i(x_i)$ , with the properties

$$\langle x_i \rangle = \mu_i \quad \langle x_i^2 \rangle - \langle x_i \rangle^2 = \sigma_i^2 \quad \phi_i(k) = \langle e^{ikx_i} \rangle$$

If we calculate average and variance of the sum of these random variables, we find:

$$\langle \sum_{i=1}^n x_i \rangle = \sum_{i=1}^n \mu_i \quad \langle \left[ \sum_{i=1}^n x_i \right]^2 \rangle - \langle \sum_{i=1}^n x_i \rangle^2 = \sum_{i,j=1}^n [\langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle] = \sum_{i=1}^n \sigma_i^2$$

In view of these expressions, we now define the following random variable:

$$x = \frac{1}{\sqrt{n}} \sum_{i=1}^n [x_i - \mu_i] \quad \langle x \rangle = 0 \quad \langle x^2 \rangle = \frac{1}{n} \sum_{i=1}^n \sigma_i^2 \quad (2.32)$$

Since  $x$  is a random variable, it has a probability distribution  $P_n(x)$  associated with it, which must depend in some complicated way on the probability distributions of the  $n$  constituent random variables  $x_i$ .

#### Central Limit Theorem (simple version):

If  $P(x) = \lim_{n \rightarrow \infty} P_n(x) \in L^2(\mathfrak{R})$  and  $(\forall n \geq 1) : P_n \in L^2(\mathfrak{R})$ , and if  $0 < \sigma^2 = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n \sigma_i^2 < \infty$ , and if the characteristic functions  $\phi_i(k)$  are analytical, then:

$$P(x) = \frac{e^{-\frac{1}{2}x^2/\sigma^2}}{\sigma\sqrt{2\pi}} \quad (2.33)$$

**Proof:** For each of the individual variables  $x_i$  we expand the characteristic function:

$$\phi_i(k) = 1 + ik\langle x_i \rangle - \frac{1}{2}k^2\langle x_i^2 \rangle + \mathcal{O}(k^3)$$

$$= 1 + ik\mu_i - \frac{1}{2}k^2[\sigma_i^2 + \mu_i^2] + \mathcal{O}(k^3) = e^{ik\mu_i - \frac{1}{2}k^2\sigma_i^2 + \mathcal{O}(k^3)}$$

We now calculate the characteristic function of  $x$  for a given  $n$ :

$$\begin{aligned} \log \phi(k) &= \log \langle e^{ikx} \rangle = -\frac{ik}{\sqrt{n}} \sum_{i=1}^n \mu_i + \sum_{i=1}^n \log \phi_i\left(\frac{k}{\sqrt{n}}\right) \\ &= -\frac{ik}{\sqrt{n}} \sum_{i=1}^n \mu_i + \sum_{i=1}^n \left\{ \frac{ik\mu_i}{\sqrt{n}} - \frac{1}{2n}k^2\sigma_i^2 + \mathcal{O}(n^{-\frac{3}{2}}) \right\} \\ &= -\frac{k^2}{2n} \sum_{i=1}^n \sigma_i^2 + \mathcal{O}\left(\frac{1}{\sqrt{n}}\right) \end{aligned}$$

Since  $P_n \in L^2(\mathfrak{R})$  it is Fourier transformable, so that the inverse Fourier transform applied to  $\phi(k)$  will produce  $P_n(x)$ :

$$\begin{aligned} P(x) &= \lim_{n \rightarrow \infty} P_n(x) = \lim_{n \rightarrow \infty} \int \frac{dk}{2\pi} e^{-ikx} \phi(k) = \lim_{n \rightarrow \infty} \int \frac{dk}{2\pi} e^{-ikx - \frac{k^2}{2n} \sum_{i=1}^n \sigma_i^2 + \mathcal{O}(n^{-\frac{1}{2}})} \\ &= \int \frac{dk}{2\pi} e^{-ikx - \frac{1}{2}k^2\sigma^2} = e^{-\frac{1}{2}x^2/\sigma^2} \int \frac{dk}{2\pi} e^{-\frac{1}{2}(k+ix/\sigma^2)^2\sigma^2} \\ &= e^{-\frac{1}{2}x^2/\sigma^2} \int_{ix/\sigma - \infty}^{ix/\sigma + \infty} \frac{dz}{2\pi\sigma} e^{-\frac{1}{2}z^2} = e^{-\frac{1}{2}x^2/\sigma^2} \int_{-\infty}^{\infty} \frac{dz}{2\pi\sigma} e^{-\frac{1}{2}z^2} \\ &= \frac{e^{-\frac{1}{2}x^2/\sigma^2}}{\sigma\sqrt{2\pi}} \end{aligned}$$

Here we have used contour integration to change the integration path in the complex plane, and the integral  $\int dz e^{-\frac{1}{2}z^2} = \sqrt{2\pi}$  (see appendix B).  $\square$

As stated earlier, one could cut the number of requirements needed to prove the validity of the CLT quite a bit. For example, a less restrictive condition on the distributions of the random variables  $x_i$ , which can still be shown to be sufficient (although again not always necessary), is the so-called *Lindenberg condition*, which we will just mention here:

$$\text{for all } t > 0 : \quad \lim_{n \rightarrow \infty} \frac{\sum_{i=1}^n \int dx p_i(x) (x - \mu_i)^2 \theta[|x - \mu_i| - t\sqrt{\sum_{i=1}^n \sigma_i^2}]}{\sum_{i=1}^n \int dx p_i(x) (x - \mu_i)^2} = 0 \quad (2.34)$$

in which  $\theta[z]$  denotes the step function:  $\theta[z > 0] = 1$ ,  $\theta[z < 0] = 0$ .

### 2.4.3 Examples

*Example 1.* Let us start with a notorious and nasty example, to take away the impression that smooth distributions will always be well-behaved (and that distributions where moments do not exist are just pathological museum pieces): the Lorentz distribution.

$$p(x) = \frac{1}{\pi} \frac{1}{1+x^2}, \quad x \in \mathfrak{R} \quad (2.35)$$

Let us check normalisation first:

$$\int dx p(x) = \int \frac{dx}{\pi} \frac{1}{1+x^2} = \frac{1}{\pi} [\arctan(x)]_{-\infty}^{\infty} = 1$$

The first few moments  $\langle x \rangle$  and  $\langle x^2 \rangle$ , however, are already ill-defined or infinite:

$$\langle x \rangle = \int dx p(x)x = \int \frac{dx}{\pi} \frac{x}{1+x^2} = \frac{1}{2\pi} [\log(1+x^2)]_{-\infty}^{\infty} = ?$$

$$\langle x^2 \rangle = \int dx p(x)x^2 = \int \frac{dx}{\pi} \frac{x^2}{1+x^2} = \int dx \left[ \frac{1}{\pi} - p(x) \right] = \infty$$

The function  $p(x)$  is in  $L^2(\mathfrak{R})$  (so Fourier transformable), since

$$= \int dx p^2(x) = \int \frac{dx}{\pi} \frac{p(x)}{1+x^2} \leq \int \frac{dx}{\pi} p(x) = \frac{1}{\pi}$$

If  $p \in L^2(\mathfrak{R})$  then also the characteristic function must have this property. If we calculate the characteristic function we find:

$$\phi(k) = \int \frac{dx}{\pi} \frac{e^{ikx}}{1+x^2} = e^{-|k|}$$

The last step follows either from contour integration, or from the following argument (see also appendix C for the definition and properties of the delta function):

$$\left[ \frac{d^2}{dk^2} - 1 \right] \phi(k) = - \int \frac{dx}{\pi} e^{ikx} = -2\delta(k) \quad \begin{cases} x > 0 : & \phi(k) = A_+ e^k + B_+ e^{-k} \\ x < 0 : & \phi(k) = A_- e^k + B_- e^{-k} \end{cases}$$

From  $\phi \in L^2(\mathfrak{R})$  we deduce that  $A_+ = B_- = 0$ . In combination with  $\phi(0) = 1$  this leads to the desired result:  $\phi(k) = e^{-|k|}$ . As a test we can finally try to reconstruct  $p(x)$  from the characteristic function via inverse Fourier transformation:

$$\begin{aligned} p(x) &= \int \frac{dk}{2\pi} \phi(k) e^{-ikx} = \int dk e^{-ikx-|k|} = \int_{-\infty}^0 \frac{dk}{2\pi} e^{k(1-ix)} + \int_0^{\infty} \frac{dk}{2\pi} e^{-k(1+ix)} \\ &= \int_0^{\infty} \frac{dk}{2\pi} \left[ e^{-k(1+ix)} + e^{-k(1-ix)} \right] = -\frac{1}{2\pi} \left[ \frac{e^{-k(1+ix)}}{1+ix} + \frac{e^{-k(1-ix)}}{1-ix} \right]_0^{\infty} \\ &= -\frac{1}{2\pi(1+x^2)} \left[ e^{-k} \left( (1-ix)e^{-ikx} + (1+ix)e^{ikx} \right) \right]_0^{\infty} = \frac{1}{\pi} \frac{1}{1+x^2} \end{aligned}$$

*Example 2.* Our second example is a smooth distribution where in some cases the Fourier transform does not exist, although the moments do:

$$p(x) = (1+\alpha)x^\alpha, \quad x \in [0, 1], \quad \alpha > -1 \quad (2.36)$$

Let us check normalisation first:

$$\int dx p(x) = (1+\alpha) \int dx x^\alpha = \left[ x^{1+\alpha} \right]_0^1 = 1$$

The moments  $\langle x^n \rangle$  are:

$$\langle x^n \rangle = \int dx p(x) x^n = (1+\alpha) \int dx x^{\alpha+n} = \frac{1+\alpha}{1+\alpha+n} \left[ x^{1+\alpha+n} \right]_0^1 = \frac{1+\alpha}{1+\alpha+n}$$

In particular it follows that

$$\mu = \langle x \rangle = \frac{1+\alpha}{2+\alpha} \quad \sigma^2 = \langle x^2 \rangle - \langle x \rangle^2 = \frac{1+\alpha}{(3+\alpha)(2+\alpha)^2}$$

To find out whether  $p(x)$  is in  $L^2[0, 1]$  (Fourier transformable), we have to evaluate

$$\int dx p^2(x) = (1+\alpha)^2 \int dx x^{2\alpha} \quad \begin{cases} \infty & \text{for } -1 < \alpha \leq -\frac{1}{2} \\ \frac{(1+\alpha)^2}{1+2\alpha} & \text{for } \alpha > -\frac{1}{2} \end{cases}$$

So  $p \in L^2[0, 1]$  only for  $\alpha > -\frac{1}{2}$ .

*Example 3.* Let us now turn to an example of a discrete random variable with well-behaved probabilities (contrary to what the name suggests, nothing is fishy), given by the Poisson distribution:

$$P(m) = \frac{1}{m!} a^m e^{-a}, \quad m \in \{0, 1, 2, 3, \dots\}, \quad a > 0 \quad (2.37)$$

Normalisation is again built-in:

$$\sum_{m \geq 0} P(m) = e^{-a} \sum_{m \geq 0} \frac{1}{m!} a^m = e^{-a} \cdot e^a = 1$$

The moments  $\langle m^n \rangle$  ( $n \geq 0$ ) follow from

$$\langle m^n \rangle = \sum_{m \geq 0} P(m) m^n = e^{-a} \sum_{m \geq 0} \frac{1}{m!} m^n a^m = e^{-a} \left[ a \frac{d}{da} \right]^n \sum_{m \geq 0} \frac{1}{m!} a^m = e^{-a} \left[ a \frac{d}{da} \right]^n e^a$$

giving

$$\begin{aligned} n = 1 : & \quad \langle m \rangle = a \\ n = 2 : & \quad \langle m^2 \rangle = a^2 + a \\ n = 3 : & \quad \langle m^3 \rangle = a^3 + 3a^2 + a \\ \dots & \quad \dots \end{aligned}$$

In particular we obtain average and variance:  $\mu = \langle x \rangle = a$ ,  $\sigma^2 = \langle x^2 \rangle - \langle x \rangle^2 = a$ . The characteristic function can be calculated as well:

$$\phi(k) = \langle e^{ikm} \rangle = \sum_{m \geq 0} P(m) e^{ikm} = e^{-a} \sum_{m \geq 0} \frac{1}{m!} [ae^{ik}]^m = e^{-a} \cdot e^{ae^{ik}} = e^{a[\cos(k) + i \sin(k) - 1]}$$

As a test we can use  $\phi(k)$  to generate the moments, using

$$\langle m^n \rangle = \lim_{k \rightarrow 0} \left[ \frac{1}{i} \frac{d}{dk} \right]^n \phi(k)$$

giving

$$\begin{aligned}
n = 1 : \quad \langle m \rangle &= \lim_{k \rightarrow 0} a e^{ik} \phi(k) = a \\
n = 2 : \quad \langle m^2 \rangle &= \lim_{k \rightarrow 0} (a e^{ik} + a^2 e^{2ik}) \phi(k) = a^2 + a \\
n = 3 : \quad \langle m^3 \rangle &= \lim_{k \rightarrow 0} (a e^{ik} + 3a^2 e^{2ik} + a^3 e^{3ik}) \phi(k) = a^3 + 3a^2 + a \\
\dots & \quad \dots
\end{aligned}$$

(as it should).

*Example 4.* Our fourth and final example is the most user-friendly distribution one can imagine (as a result it is always the standard application example): the Gaussian (or ‘normal’) distribution.

$$p(x) = \frac{e^{-\frac{1}{2}(x-\mu)^2/\sigma^2}}{\sigma\sqrt{2\pi}}, \quad x \in \mathfrak{R} \quad (2.38)$$

Normalisation is built-in (see also appendix B):

$$\int dx p(x) = \int \frac{dx}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(x-\mu)^2/\sigma^2} = \int \frac{dz}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} = 1$$

For the Gaussian distribution we also obtain:

$$\langle (x-\mu)^n \rangle = \int \frac{dx}{\sigma\sqrt{2\pi}} (x-\mu)^n e^{-\frac{1}{2}(x-\mu)^2/\sigma^2} = \sigma^n \int \frac{dz}{\sqrt{2\pi}} z^n e^{-\frac{1}{2}z^2}$$

For  $n$  odd the result is zero. For  $n$  even we get:

$$\begin{aligned}
\langle (x-\mu)^{2m} \rangle &= \sigma^{2m} (-1)^m \lim_{y \rightarrow \frac{1}{2}} \frac{d^m}{dy^m} \int \frac{dz}{\sqrt{2\pi}} e^{-yz^2} = \frac{\sigma^{2m}}{\sqrt{2}} (-1)^m \lim_{y \rightarrow \frac{1}{2}} \frac{d^m}{dy^m} y^{-\frac{1}{2}} \\
&= \frac{\sigma^{2m}}{\sqrt{2}} \lim_{y \rightarrow \frac{1}{2}} \left[ \frac{1}{2} \cdot \frac{3}{2} \cdot \dots \cdot \frac{2m-1}{2} \right] y^{-\frac{1}{2}-m} = \sigma^{2m} [1 \cdot 3 \cdot \dots \cdot (2m-1)]
\end{aligned}$$

From these relations, in turn, follow the various moments. For example:

$$\begin{aligned}
\langle (x-\mu) \rangle = 0 &\quad \Rightarrow \quad \langle x \rangle = \mu \\
\langle (x-\mu)^2 \rangle = \sigma^2 &\quad \Rightarrow \quad \langle x^2 \rangle = \mu^2 + \sigma^2 \\
\langle (x-\mu)^3 \rangle = 0 &\quad \Rightarrow \quad \langle x^3 \rangle = \mu^3 + 3\mu\sigma^2 \\
\langle (x-\mu)^4 \rangle = 3\sigma^4 &\quad \Rightarrow \quad \langle x^4 \rangle = 3(\mu^2 + \sigma^2)^2
\end{aligned}$$

Relations such as  $\langle x^4 \rangle = 3(\mu^2 + \sigma^2)^2$  can often serve as a quick test to see whether an unknown distribution could be Gaussian. The Gaussian distribution is in  $L^2(\mathfrak{R})$  (so Fourier transformable), since

$$\int dx p^2(x) = \int \frac{dx}{2\pi\sigma^2} e^{-(x-\mu)^2/\sigma^2} = \int \frac{dz}{2\pi\sigma} e^{-z^2} = \frac{1}{2\sigma\sqrt{\pi}}$$

The characteristic function is again of a Gaussian shape:

$$\begin{aligned}
\phi(k) &= \int dx p(x) e^{ikx} = \int \frac{dx}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(x-\mu)^2/\sigma^2 + ikx} = e^{ik\mu} \int \frac{dz}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2 + ik\sigma z} \\
&= e^{ik\mu - \frac{1}{2}k^2\sigma^2} \int \frac{dz}{\sqrt{2\pi}} e^{-\frac{1}{2}[z-ik\sigma]^2} = e^{ik\mu - \frac{1}{2}k^2\sigma^2} \int_{-ik\sigma-\infty}^{-ik\sigma+\infty} \frac{dz}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} \\
&= e^{ik\mu - \frac{1}{2}k^2\sigma^2} \int \frac{dz}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} = e^{ik\mu - \frac{1}{2}k^2\sigma^2}
\end{aligned}$$

## Chapter 3

# Building Blocks of Shannon's Information Theory

### 3.1 Entropy

The entropy  $H(X)$  of a discrete random variable  $x \in A$ , measured in *bits*, is defined as follows

$$H(X) = - \sum_{x \in A} p(x) {}^2\log p(x) \quad (3.1)$$

We assume  $p(x) > 0$  for all  $x \in A$ , which can always be arranged via the definition of  $A$ , unless stated otherwise (in the latter cases we define  $0 {}^2\log 0 = \lim_{\epsilon \downarrow 0} \epsilon {}^2\log \epsilon = 0$ ). As will be demonstrated in the next chapter, the entropy represents the average information content of messages  $x$  from the set  $A$  with the specified probabilities of occurrence  $p(x)$ .

#### 3.1.1 General Properties

**property 1:**  $H(X) \geq 0$ , with equality if and only if  $x$  is a constant.

**proof:** Use the property  $p(x) \leq 1$  for all  $x \in A$ :

$$H(X) = \sum_{x \in A} p(x) {}^2\log \left[ \frac{1}{p(x)} \right] \geq \sum_{x \in A} p(x) {}^2\log [1] = 0 \quad (3.2)$$

Equality implies  $p(x) = 1$  ( $\forall x \in A$ ). Since also  $\sum_{x \in A} p(x) = 1$  this forces  $x$  to be a constant. Conversely, if  $x$  is a constant then the equality indeed holds.  $\square$

**property 2:**  $H(X) \leq {}^2\log |A|$ , with equality if and only if  $p(x) = |A|^{-1}$  for all  $x \in A$ .

**proof:** Define auxiliary probabilities  $q(x) = |A|^{-1}$  and use the log-sum inequality (D.5):

$$\begin{aligned} {}^2\log |A| - H(X) &= \sum_{x \in A} p(x) \left[ {}^2\log p(x) + {}^2\log |A| \right] = \sum_{x \in A} p(x) \left[ {}^2\log p(x) - {}^2\log q(x) \right] \\ &= \sum_{x \in A} p(x) {}^2\log \left[ \frac{p(x)}{q(x)} \right] \geq \left[ \sum_{x \in A} p(x) \right] {}^2\log \left[ \frac{\sum_{x \in A} p(x)}{\sum_{x \in A} q(x)} \right] = 0 \end{aligned}$$



Equality holds if and only if it holds in the log-sum inequality, so  $(\exists \lambda > 0) : p(x) = \lambda q(x)$ . Since  $\sum_{x \in A} p(x) = \sum_{x \in A} q(x) = 1$  the constant  $\lambda$  must be 1, so  $p(x) = |A|^{-1}$  for all  $x \in A$ .  $\square$

**property 3:** For any pair of random variables  $(x_1, x_2) \in A \subset \mathfrak{R}^2$ :  $H(X_1, X_2) \leq H(X_1) + H(X_2)$ , with equality if and only if  $x_1 \in A_1$  and  $x_2 \in A_2$  are independent.

**proof:** Subtract the two sides of the proposed inequality and use the definitions of the marginal distributions,  $p(x_1) = \sum_{x_2 \in A_2} p(x_1, x_2)$  and  $p(x_2) = \sum_{x_1 \in A_1} p(x_1, x_2)$ :

$$\begin{aligned} & H(X_1, X_2) - H(X_1) - H(X_2) \\ &= \sum_{x_1 \in A_1} p(x_1) {}^2\log p(x_1) + \sum_{x_2 \in A_2} p(x_2) {}^2\log p(x_2) - \sum_{x_1 \in A_1} \sum_{x_2 \in A_2} p(x_1, x_2) {}^2\log p(x_1, x_2) \\ &= \sum_{x_1 \in A_1} \sum_{x_2 \in A_2} p(x_1, x_2) {}^2\log \left[ \frac{p(x_1)p(x_2)}{p(x_1, x_2)} \right] \end{aligned}$$

We now use the log-sum inequality (D.5) and obtain:

$$H(X_1, X_2) - H(X_1) - H(X_2) \leq \left[ \sum_{x_1 \in A_1} \sum_{x_2 \in A_2} p(x_1, x_2) \right] {}^2\log \left[ \frac{\sum_{x_1 \in A_1} \sum_{x_2 \in A_2} p(x_1)p(x_2)}{\sum_{x_1 \in A_1} \sum_{x_2 \in A_2} p(x_1, x_2)} \right] = 0$$

With equality obtained if and only if  $(\exists \lambda > 0) : p(x_1, x_2) = \lambda p(x_1)p(x_2)$  for all  $(x_1, x_2) \in A$ . In the latter case we can sum over all  $(x_1, x_2) \in A$  and obtain  $\lambda = 1$ . We conclude that equality holds if and only if the two random variables  $x_1$  and  $x_2$  are independent.  $\square$

**property 4:**  $H(F(X)) \leq H(X)$ , with equality if and only if  $F$  is invertible.

**proof:** Define the new random variables  $y = F(x) \in \hat{A}$ , with the non-overlapping domains  $D_y = \{x \in A \mid F(x) = y\}$  which obey  $\cup_{y \in \hat{A}} D_y = A$ , and with the associated probabilities

$$\hat{p}(y) = \sum_{x \in D_y} p(x), \quad \sum_{y \in \hat{A}} \hat{p}(y) = \sum_{y \in \hat{A}} \sum_{x \in D_y} p(x) = \sum_{x \in A} p(x) = 1$$

Now work out the difference between the two entropies:

$$\begin{aligned} H(F(X)) - H(X) &= - \sum_{y \in \hat{A}} \hat{p}(y) {}^2\log \hat{p}(y) + \sum_{x \in A} p(x) {}^2\log p(x) \\ &= - \sum_{y \in \hat{A}} \left\{ \hat{p}(y) {}^2\log \hat{p}(y) - \sum_{x \in D_y} p(x) {}^2\log p(x) \right\} = - \sum_{y \in \hat{A}} \sum_{x \in D_y} p(x) {}^2\log \left[ \frac{\sum_{x' \in D_y} p(x')}{p(x)} \right] \\ &= - \sum_{y \in \hat{A}} \sum_{x \in D_y} p(x) {}^2\log \left[ \frac{p(x) + \sum_{x' \in D_y, x' \neq x} p(x')}{p(x)} \right] \leq 0 \end{aligned}$$

Note that equality implies that *each* of the terms  $\sum_{x' \in D_y, x' \neq x} p(x')$  vanishes, which happens if and only if  $(\forall y \in \hat{A}) : |D_y| = 1$ . This is precisely the condition for invertibility of the operation  $F$ .  $\square$

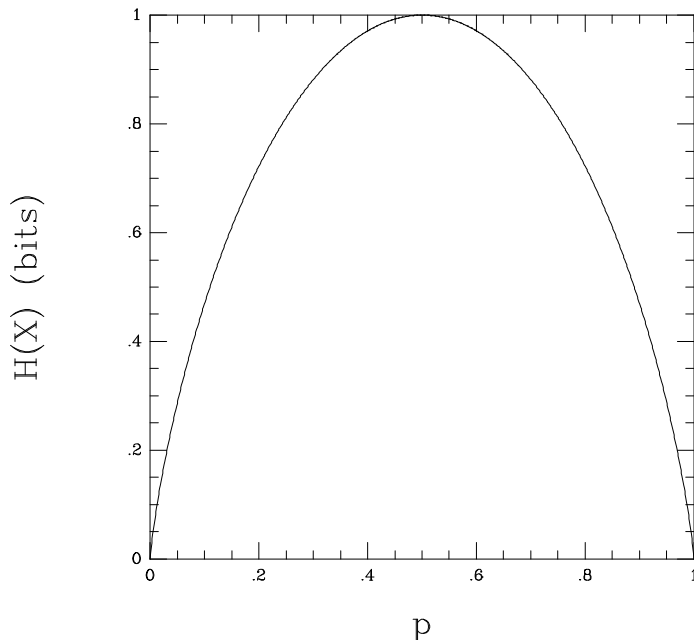


Figure 3.1: Entropy  $H(X)$  for a binary random variable  $x \in \{0, 1\}$ , with probabilities  $p(1) = p$  and  $p(0) = 1 - p$ , as a function of  $p$ . See equation (3.3).

### 3.1.2 Examples

*Example 1.* Let us inspect the simplest non-trivial discrete random variables  $x$ , the ones which can take only two values:  $A = \{0, 1\}$ , with  $p(1) = p$  and  $p(0) = 1 - p$  (with  $p \in [0, 1]$ ). Here we obtain from (3.1):

$$H(X) = -p^2 \log p - (1-p)^2 \log(1-p) \quad (3.3)$$

This expression has the following properties (note:  ${}^2 \log z = \log z / \log 2$ ):

- (i)  $H(X)_p = H(X)_{1-p}$  for all  $p \in [0, 1]$ .
- (ii)  $H(X)_{p=0} = H(X)_{p=1} = 0$ ,  $H(X)_{p=\frac{1}{2}} = 1$
- (iii)  $\frac{d}{dp} H(X) > 0$  for  $p \in [0, \frac{1}{2}]$ ,  $\frac{d}{dp} H(X)_{p=\frac{1}{2}} = 0$ ,  $\frac{d}{dp} H(X) < 0$  for  $p \in [\frac{1}{2}, 1]$ .

This immediately follows from

$$\frac{d}{dp} H(X) = -\frac{1}{\log 2} \frac{d}{dp} [p \log p + (1-p) \log(1-p)] = \frac{1}{\log 2} \log \left[ \frac{1-p}{p} \right] = {}^2 \log \left[ \frac{1}{p} - 1 \right]$$

These properties have a straightforward explanation in terms of information content: (i) states that the information content is invariant under  $0 \leftrightarrow 1$  (alternative coding), (ii) + (iii) state that the information content is maximal for uniform probabilities  $p(1) = p(0)$  and decreases monotonically with increasing probability differences, until it vanishes at  $p \in \{0, 1\}$  (where  $x$  reduces to a constant). Figure 3.1 shows the dependence of  $H(X)$  on  $p$  in a graph.

*Example 2.* Our second example concerns the information content of messages conveying events in a casino, where a coin is thrown until the first head occurs. The random variable  $m$  represents the number of times the coin is thrown (i.e. the number of throws needed for the first head to show up), so  $A = \{1, 2, 3, \dots\}$  with  $|A| = \infty$ . The probabilities  $p(m)$  are obtained by inspection of the various scenarios:

$$\begin{aligned} p(1) &= \text{Prob}(h) &&= \frac{1}{2} \\ p(2) &= \text{Prob}(th) &&= \frac{1}{2} \cdot \frac{1}{2} \\ p(3) &= \text{Prob}(tth) &&= \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \\ &\vdots &&\vdots \\ p(m) &= \text{Prob}(t\dots th) &&= \left(\frac{1}{2}\right)^m \end{aligned}$$

With the help of the simple tools in appendix A to deal with the summation, the entropy (3.1) now comes out as

$$\begin{aligned} H(M) &= - \sum_{m=1}^{\infty} \left(\frac{1}{2}\right)^m {}^2\log\left(\frac{1}{2}\right)^m = \sum_{m=1}^{\infty} m\left(\frac{1}{2}\right)^m = \lim_{z \rightarrow \frac{1}{2}} z \frac{d}{dz} \sum_{m=0}^{\infty} z^m \\ &= \lim_{z \rightarrow \frac{1}{2}} z \frac{d}{dz} \frac{1}{1-z} = \lim_{z \rightarrow \frac{1}{2}} \frac{z}{(1-z)^2} = 2 \text{ bits} \end{aligned} \tag{3.4}$$

On average such messages apparently convey just two bits of information. Note that for this example the prefix code is optimal:

$$C(m) = \underbrace{00\dots 0}_{m-1 \text{ times}}1, \quad \ell(m) = m : \quad L = \sum_{m \geq 1} p(m)\ell(m) = \sum_{m \geq 1} m\left(\frac{1}{2}\right)^m = H$$

*Example 3.* Our third example is an illustration of how the information content of a random variable is reduced if it undergoes a non-invertible operation. Let us choose the event set  $A = \{0, 1, 2, 3, 4, 5, 6, 7\}$  and the following operation:  $F(x) = \cos(\pi x)$ . Equivalently (at least for the present event set):  $F(x) = 1$  for  $x$  even,  $F(x) = -1$  for  $x$  odd. This example is quite typical of real-world communication via imperfect channels (such as lousy telephone lines). Here the only information delivered at the receiving end of the communication channel is whether the original message variable  $x$  was even or odd. The new random variable  $y = F(x)$  has the following set of possible values and associated probabilities

$$y \in \hat{A} = \{-1, 1\}, \quad \begin{aligned} \hat{p}(-1) &= p(1) + p(3) + p(5) + p(7) \\ \hat{p}(1) &= p(0) + p(2) + p(4) + p(6) \end{aligned} \tag{3.5}$$

The information content of the two types of messages, before ( $x$ ) and after ( $y = F(x)$ ) the operation  $F$ , is given by, respectively:

$$H(X) = - \sum_{x \in A} p(x) {}^2\log p(x) \quad H(F(X)) = -\hat{p}(-1) {}^2\log \hat{p}(-1) - \hat{p}(1) {}^2\log \hat{p}(1) \tag{3.6}$$

The outcome of the information reduction  $\Delta H = H(F(X)) - H(X)$  due to the non-invertible operation (the lousy communication medium) will depend on the choice made for the probabilities  $p(x)$ .

In the simplest case where all messages  $x$  are equally likely, i.e.  $p(x) = \frac{1}{8}$  for all  $x \in A$ , we obtain  $\hat{p}(-1) = \hat{p}(1) = \frac{1}{2}$  which results in

$$H(X) = {}^2\log|A| = 3 \text{ bits}, \quad H((F(X))) = {}^2\log|\hat{A}| = 1 \text{ bits}, \quad \Delta H = -2 \text{ bits}$$

Only one third of the original information content survives the operation  $F$ . Alternatively we could inspect a situation where the events in  $A$  have different probabilities of occurrence. Let us choose the following probabilities:

$x$	0	1	2	3	4	5	6	7
$p(x)$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$	$\frac{1}{64}$	$\frac{1}{128}$	$\frac{1}{128}$

According to (3.5) we now have  $\hat{p}(-1) = \frac{43}{128}$  and  $\hat{p}(1) = \frac{85}{128}$ , so that we find for the two entropies (3.6):

$$H(X) = 1 \frac{63}{64} \approx 1.984 \text{ bits}, \quad H(F(X)) = -\frac{43}{128} {}^2\log\left(\frac{43}{128}\right) - \frac{85}{128} {}^2\log\left(\frac{85}{128}\right) \approx 0.921 \text{ bits}$$

$$\Delta H \approx -1.063 \text{ bits}$$

Here about one half of the original information content survives the operation  $F$ . This example illustrates that for a given type of message deterioration (i.e. a given operation  $F$ , which could also involve a probabilistic element) the information loss is dependent on the various probabilities of occurrence of the individual messages.

## 3.2 Joint Entropy and Conditional Entropy

The joint entropy  $H(X, Y)$  of a pair of discrete random variables  $(x, y) \in A$ , measured in *bits*, is defined as follows

$$H(X, Y) = - \sum_{(x,y) \in A} p(x, y) {}^2\log p(x, y) \quad (3.7)$$

The joint entropy represents the average information content of messages  $(x, y)$  from the set  $A$  with the specified probabilities of occurrence  $p(x, y)$ . As before:  $0 \leq H(X, Y) \leq {}^2\log|A|$  (with zero only occurring only when  $(x, y)$  is constant, and  ${}^2\log|A|$  occurring only when  $p(x, y) = |A|^{-1}$ ). The proofs of these statements are identical to those given for  $H(X)$ . Generalisation of the definition of joint entropy to messages consisting of  $n$  random variables,  $H(X_1, \dots, X_n)$ , is straightforward.

The conditional entropy  $H(Y|X)$  of a pair of discrete random variables  $(x, y) \in A$ , measured in *bits*, is defined as follows

$$H(Y|X) = - \sum_{(x,y) \in A} p(x, y) {}^2\log p(y|x) \quad (3.8)$$

The conditional entropy represents the average information content of message component  $y$  of a message  $(x, y)$  from the set  $A$ , given that one knows the other component  $x$  already, with the specified probabilities of occurrence  $p(x, y)$ . Generalisation of the definition of conditional entropy to messages consisting of more than two random variables, giving objects such as  $H(X, Y|Z)$ , is again straightforward.

### 3.2.1 General Properties

**property 1:**  $H(Y|X) \geq 0$ , with equality if and only if there exists a function  $F$  such that  $y = F(x)$  for all  $(x, y)$  with  $p(x, y) > 0$ .

**proof:** Define  $A_y$  as the set of all  $y$  that occur in some pair  $(x, y) \in A$  (likewise  $A_x$ ), and use the following property:

$$p(y|x) = \frac{p(x, y)}{p(x)} = \frac{p(x, y)}{\sum_{y' \in A_y} p(x, y')} = \frac{p(x, y)}{p(x, y) + \sum_{y' \in A_y, y' \neq y} p(x, y')} \leq 1$$

This gives

$$H(Y|X) = \sum_{(x,y) \in A} p(x, y) {}^2\log \left[ \frac{1}{p(y|x)} \right] \geq \sum_{(x,y) \in A} p(x, y) {}^2\log 1 = 0$$

Equality implies  $p(x, y) = p(x)$  for all  $(x, y) \in A$ , so  $\sum_{y' \in A_y} p(x, y') = p(x, y)$ . In other words: for each  $x \in A_x$  there is just one  $y \in A_y$  such that  $p(x, y) \neq 0$ . This means that  $y$  can be written as a function of  $x$  for all allowed pairs  $(x, y)$ .  $\square$

**property 2:**  $H(Y|X) \leq H(Y)$ , with equality if and only if  $x$  and  $y$  are independent.

**proof:** Define  $A_y$  as the set of all  $y$  that occur in some pair  $(x, y) \in A$ , define auxiliary probabilities  $q(x, y) = p(x)p(y)$  and use the log-sum inequality (D.5):

$$\begin{aligned} H(Y) - H(Y|X) &= \sum_{(x,y) \in A} p(x, y) {}^2\log p(y|x) - \sum_{y \in A_y} p(y) {}^2\log p(y) \\ &= \sum_{(x,y) \in A} p(x, y) \left[ {}^2\log p(y|x) - {}^2\log p(y) \right] = \sum_{(x,y) \in A} p(x, y) {}^2\log \left[ \frac{p(x, y)}{q(x, y)} \right] \\ &\geq \left[ \sum_{(x,y) \in A} p(x, y) \right] {}^2\log \left[ \frac{\sum_{(x,y) \in A} p(x, y)}{\sum_{(x,y) \in A} q(x, y)} \right] = 0 \end{aligned}$$

Equality holds if and only if it holds in the log-sum inequality, so  $(\exists \lambda > 0) : p(x, y) = \lambda q(x, y)$ . Since  $\sum_{(x,y) \in A} p(x, y) = \sum_{(x,y) \in A} q(x, y) = 1$  the constant  $\lambda$  must be 1, so  $p(x, y) = q(x, y) = p(x)p(y)$  for all  $(x, y) \in A$ .  $\square$

**property 3 (chain rule):**  $H(X, Y) = H(X) + H(Y|X)$ .

**proof:** Define  $A_x$  as the set of all  $x$  that occur in some pair  $(x, y) \in A$  and use  $p(x, y) = p(y|x)p(x)$ :

$$\begin{aligned} H(X, Y) &= - \sum_{(x,y) \in A} p(x, y) {}^2\log p(x, y) = - \sum_{(x,y) \in A} p(x, y) {}^2\log [p(y|x)p(x)] \\ &= - \sum_{(x,y) \in A} p(x, y) {}^2\log p(y|x) - \sum_{(x,y) \in A} p(x, y) {}^2\log p(x) \\ &= H(Y|X) - \sum_{x \in A_x} p(x) {}^2\log p(x) = H(Y|X) + H(X) \end{aligned}$$

which completes the proof.  $\square$

**property 4:**  $H(X, Y) \geq H(X)$ , with equality if and only if there exists a function  $F$  such that  $y = F(x)$  for all  $(x, y)$  with  $p(x, y) > 0$ .

**proof:** Combination of properties 1 and 3. □

**property 5 (chain rule):**  $H(X, Y|Z) = H(X|Z) + H(Y|X, Z)$ .

**proof:** Define  $A_x$  as the set of all  $x$  that occur in some pair  $(x, y, z) \in A$  (similarly for  $y$  and  $z$ ) and use  $p(x, y|z) = p(y|x, z)p(x|z)$ :

$$\begin{aligned} H(X, Y|Z) &= - \sum_{(x,y,z) \in A} p(x, y, z) \log p(x, y|z) = - \sum_{(x,y,z) \in A} p(x, y, z) \log [p(y|x, z)p(x|z)] \\ &= - \sum_{(x,y,z) \in A} p(x, y, z) \log p(y|x, z) - \sum_{(x,y,z) \in A} p(x, y, z) \log p(x|z) \\ &= H(Y|X, Z) + H(X|Z) \end{aligned} \quad \square$$

**property 6:**  $H(F(X)|X) = 0$ .

**proof:** Included in the proof of property 1. □

**property 7:**  $H(X|F(X)) = 0$  if  $F$  is invertible,  $H(X|F(X)) > 0$  if  $F$  is not invertible.

**proof:** We simply apply property 1 to the random variables  $x$  and  $F(x)$ :  $H(X|F(X)) \geq 0$ , with equality if and only if there exists a function  $G$  such that  $x = G(F(x))$  for all  $x$  with  $p(x) > 0$ . □

### 3.2.2 Examples

*Example 1.* Our first example serves to illustrate the above properties, as well as demonstrate that in general  $H(Y|X) \neq H(X|Y)$ . Consider the stochastic variables  $x$  and  $y$  with  $A_x = A_y = \{1, 2, 3, 4\}$  (and  $A = A_x \otimes A_y$ ), and with joint probabilities  $p(x, y)$  as given in the following table:

$y \ x$	1	2	3	4
1	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$	$\frac{1}{32}$
2	$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{32}$	$\frac{1}{32}$
3	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$
4	$\frac{1}{4}$	0	0	0

The marginal probabilities  $p(x) = \sum_{y \in A_y} p(x, y)$  and  $p(y) = \sum_{x \in A_x} p(x, y)$  come out as

$x$	1	2	3	4
$p(x)$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{8}$

$y$	1	2	3	4
$p(y)$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

If we simply insert the appropriate probabilities in the various definitions, we obtain for the individual entropies  $H(X)$  and  $H(Y)$  and for the joint entropy  $H(X, Y)$  the following results:

$$H(X) = - \sum_{x \in A_x} p(x) {}^2\log p(x) = 1\frac{3}{4} \text{ bits} \quad H(Y) = - \sum_{y \in A_y} p(y) {}^2\log p(y) = 2 \text{ bits}$$

$$H(X, Y) = - \sum_{(x,y) \in A} p(x, y) {}^2\log p(x, y) = -\frac{1}{4} {}^2\log \frac{1}{4} - \frac{2}{8} {}^2\log \frac{1}{8} - \frac{6}{16} {}^2\log \frac{1}{16} - \frac{4}{32} {}^2\log \frac{1}{32} = 3\frac{3}{8} \text{ bits}$$

After careful bookkeeping (summing over all nonzero entries in the table of the joint probabilities) we also obtain the two conditional entropies  $H(Y|X)$  and  $H(X|Y)$ :

$$H(X|Y) = - \sum_{(x,y) \in A} p(x, y) {}^2\log p(x|y) = 1\frac{3}{8} \text{ bits}$$

$$H(Y|X) = - \sum_{(x,y) \in A} p(x, y) {}^2\log p(y|x) = 1\frac{5}{8} \text{ bits}$$

Apparently the average amount of extra information contained in  $x$  if we already know  $y$  need not be identical to the average amount of extra information conveyed by  $y$  if we already know  $x$ . The present example obeys the following inequalities and equalities:

$$0 < H(Y|X) < H(Y) < H(X, Y)$$

$$0 < H(X|Y) < H(X) < H(X, Y)$$

$$H(X, Y) = H(X) + H(Y|X)$$

$$H(X, Y) = H(Y) + H(X|Y)$$

$$H(X) - H(X|Y) = H(Y) - H(Y|X)$$

The first four lines indeed confirm the general properties derived earlier in this section. The bottom line points at a property to be built upon in the subsequent section.

*Example 2.* Our second and trivial example is the one with a non-invertible operation we inspected earlier (example 3 of the previous section)  $F(x) = \cos(\pi x)$  with  $A_x = \{0, 1, 2, 3, 4, 5, 6, 7\}$ . We introduce the variable  $y = F(x)$ , with  $A_y = \{-1, 1\}$ . The joint probability distribution  $p(x, y)$  now becomes

$$p(x, y) = p(x) \delta_{y, F(x)}$$

with  $\delta_{nm} = 1$  if and only if  $n = m$  and zero otherwise. Here we obtain the following expressions for joint and conditional entropies:

$$H(X, Y) = - \sum_{(x,y) \in A_x \otimes A_y} p(x, y) {}^2\log p(x, y) = - \sum_{x \in A_x} p(x) {}^2\log p(x) = H(X)$$

$$\begin{aligned} H(X|Y) &= - \sum_{(x,y) \in A_x \otimes A_y} p(x, y) {}^2\log p(x|y) = \sum_{(x,y) \in A_x \otimes A_y} p(x, y) \{ {}^2\log p(y) - {}^2\log p(x, y) \} \\ &= H(X, Y) - H(Y) = H(X) - H(Y) \end{aligned}$$

$$\begin{aligned} H(Y|X) &= - \sum_{(x,y) \in A_x \otimes A_y} p(x,y) {}^2\log p(y|x) = \sum_{(x,y) \in A_x \otimes A_y} p(x,y) \left\{ {}^2\log p(x) - {}^2\log p(x,y) \right\} \\ &= H(X,Y) - H(X) = 0 \end{aligned}$$

Such results make perfect sense. The additional information conveyed by learning the value of  $y = F(x)$  is zero if we know  $x$  already, so  $H(Y|X) = 0$ . In contrast, learning the precise value of  $x$  does convey additional information if so far we know only  $y = F(x)$  (i.e. whether  $x$  is even or odd).

### 3.3 Relative Entropy and Mutual Information

The relative entropy  $D(p||q)$  of two discrete random variables  $x, x' \in A$ , one described by probability distributions  $p(x)$  and the other by probability distribution  $q(x')$ , measured in *bits*, is defined as

$$D(p||q) = \sum_{x \in A} p(x) {}^2\log \left[ \frac{p(x)}{q(x)} \right] \quad (3.9)$$

with the usual convention  $0 {}^2\log 0 = \lim_{\epsilon \downarrow 0} \epsilon {}^2\log \epsilon = 0$ . The relative entropy is alternatively often called cross-entropy, or ‘Kullback-Leibler distance’, a term which has to be used with some care since  $D(p||q)$  cannot play the role of a true distance due to the possibility that  $D(p||q) \neq D(q||p)$ . As we will demonstrate shortly, it does, however, have the nice property that  $D(p||q) \geq 0$  for any pair of distributions  $p$  and  $q$ , with equality if and only if  $p(x) = q(x)$  for all  $x \in A$ .

It is only natural to also introduce a measure which is similar to the relative entropy but symmetric in the two probability distributions  $p(x)$  and  $q(x)$ :

$$J(p||q) = D(p||q) + D(q||p) \quad (3.10)$$

which is called Jeffreys’ divergence. Its properties of course follow trivially from those of the relative entropy.

An important object, especially in the context of neural networks as we will see, is the so-called mutual information  $I(X, Y)$  of a pair of random variables  $(x, y) \in A$ , defined as

$$I(X, Y) = \sum_{(x,y) \in A} p(x,y) {}^2\log \left[ \frac{p(x,y)}{p(x)p(y)} \right] \quad (3.11)$$

Note that  $I(X, Y) = I(Y, X)$  and that for independent variables  $x$  and  $y$  one would find  $I(X, Y) = 0$ .

Similarly one can define the conditional mutual information  $I(X, Y|Z)$  involving three random variables  $(x, y, z) \in A$  as

$$I(X, Y|Z) = \sum_{(x,y,z) \in A} p(x,y,z) {}^2\log \left[ \frac{p(x,y|z)}{p(x|z)p(y|z)} \right] \quad (3.12)$$

This conditional mutual information has again the properties that  $I(X, Y|Z) = I(Y, X|Z)$  and that for conditionally independent variables  $x$  and  $y$ , where  $p(x, y|z) = p(x|z)p(y|z)$ , one finds  $I(X, Y|Z) = 0$ .



### 3.3.1 General Properties

**property 1:**  $D(p||q) \geq 0$  for any pair of distributions  $p(x)$  and  $q(x)$ , with equality if and only if  $p(x) = q(x)$  for all  $x \in A$ .

**proof:** Use the log-sum inequality (D.5) and the normalisation  $\sum_{x \in A} p(x) = \sum_{x \in A} q(x) = 1$ :

$$D(p||q) = \sum_{x \in A} p(x) {}^2\log \left[ \frac{p(x)}{q(x)} \right] \geq \left[ \sum_{x \in A} p(x) \right] {}^2\log \left[ \frac{\sum_{x \in A} p(x)}{\sum_{x \in A} q(x)} \right] = 0$$

Equality holds if and only if it holds in the log-sum inequality, so if  $(\exists \lambda > 0) : q(x) = \lambda p(x)$  for all  $x \in A$ . Normalisation of  $p$  and  $q$  dictates  $\lambda = 1$ .  $\square$

**property 2:**  $I(X, Y) \geq 0$  for any pair of random variables  $(x, y) \in A$ , with equality if and only if  $x$  and  $y$  are independent.

**proof:** Note that  $I(X, Y) = D(p(x, y)||p(x)p(y))$  and use property 1.  $\square$

**property 3:**  $I(X, Y) = H(X) - H(X|Y)$

**proof:** Substitute  $p(x, y) = p(x|y)p(y)$  in the definition of  $I(X, Y)$ :

$$I(X, Y) = \sum_{(x,y) \in A} p(x, y) {}^2\log \left[ \frac{p(x, y)}{p(x)p(y)} \right] = \sum_{(x,y) \in A} p(x, y) {}^2\log \left[ \frac{p(x|y)}{p(x)} \right] = H(X) - H(X|Y)$$

$\square$

Note that from  $I(X, Y) = I(Y, X)$  it immediately follows that also  $I(X, Y) = H(Y) - H(Y|X)$  (see example 1 in previous section). Note secondly that property 3 allows us to attach a clear meaning to mutual information.  $I(X, Y)$  is the average amount of information in messages  $x$  minus the residual average information that is left after we have learned about  $y$ . Since the reduction was entirely due to the revelation of  $y$  we arrive at:  $I(X, Y)$  is the average amount of information that  $y$  reveals about  $x$ , and vice versa (the 'vice-versa' simply follows from  $I(X, Y) = I(Y, X)$ ). This interpretation follows alternatively from the following statement:

**property 4:**  $I(X, Y) = H(X) + H(Y) - H(X, Y)$

**proof:** Use the chain rule  $H(X, Y) = H(Y) + H(X|Y)$  (property 3) of section 3.2.1 in combination with property 3 above:

$$I(X, Y) = H(X) - H(X|Y) = H(X) - \{H(X, Y) - H(Y)\} = H(X) + H(Y) - H(X, Y)$$

$\square$

**property 5:**  $I(X, F(X)) \leq H(X)$ , with equality if and only if  $F$  is invertible.

**proof:** Combine property 3 above with property 7 of section 3.2.1:

$$I(X, F(X)) - H(X) = -H(X|F(X)) \leq 0$$

with equality (see property 7 in 3.2.1) only if  $F$  is invertible.  $\square$

In particular we find  $I(X, X) = H(X)$ , i.e. the average amount of information that  $x$  conveys about  $x$  is simply the average amount of information in  $x$  (as it should).

**property 6:**  $I(X, Y|Z) \geq 0$ , with equality if and only if  $p(x, y|z) = p(x|z)p(y|z)$  for all  $(x, y, z) \in A$  (i.e. if  $x$  and  $y$  are conditionally independent).

**proof:** Use the log-sum inequality (D.5):

$$\begin{aligned} I(X, Y|Z) &= \sum_{(x,y,z) \in A} p(x, y, z) {}^2\log \left[ \frac{p(x, y|z)}{p(x|z)p(y|z)} \right] = \sum_{(x,y,z) \in A} p(x, y, z) {}^2\log \left[ \frac{p(x, y, z)}{p(x|z)p(y|z)p(z)} \right] \\ &\geq \left[ \sum_{(x,y,z) \in A} p(x, y, z) \right] {}^2\log \left[ \frac{\sum_{(x,y,z) \in A} p(x, y, z)}{\sum_{(x,y,z) \in A} p(x|z)p(y|z)p(z)} \right] \\ &= -{}^2\log \left[ \sum_{(x,y,z) \in A} p(x, z)p(y, z)/p(z) \right] = -{}^2\log \left[ \sum_{z \in A_z} p(z) \right] = 0 \end{aligned}$$

Equality holds only when it holds in the log-sum inequality (D.5), which implies that  $(\exists \lambda > 0) : p(x, y|z) = \lambda p(x|z)p(y|z)$  for all  $(x, y, z) \in A$ . Normalisation forces  $\lambda$  to be one:

$$\begin{aligned} p(x, y, z) = \lambda p(x|z)p(y|z)p(z) &\Rightarrow 1 = \lambda \sum_{(x,y,z) \in A} p(x|z)p(y|z)p(z) \\ 1 = \lambda \sum_{(x,y,z) \in A} p(x, z)p(y, z)/p(z) &= \lambda \end{aligned}$$

So indeed  $p(x, y|z) = p(x|z)p(y|z)$  for all  $(x, y, z) \in A$ .  $\square$

**data processing inequality:** If three random variables  $(x, y, z) \in A$  are related by a Markov chain  $X \rightarrow Y \rightarrow Z$ , i.e.  $p(x, y, z) = p(z|y)p(y|x)p(x)$ , then  $I(X, Y) \geq I(X, Z)$ .

**proof:** Show that  $I(X, Y) = I(X, Z) + I(X, Y|Z)$ , by using the Markov property:

$$\begin{aligned} I(X, Y) - I(X, Z) - I(X, Y|Z) &= \\ \sum_{(x,y,z) \in A} p(x, y, z) &\left\{ {}^2\log \left[ \frac{p(x, y)}{p(x)p(y)} \right] - {}^2\log \left[ \frac{p(x, z)}{p(x)p(z)} \right] - {}^2\log \left[ \frac{p(x, y|z)}{p(x|z)p(y|z)} \right] \right\} \end{aligned}$$

$$\begin{aligned}
&= \sum_{(x,y,z) \in A} p(x,y,z) {}^2\log \left[ \frac{p(x,y)p(x)p(z)p(x|z)p(y|z)}{p(x,z)p(x)p(y)p(x,y|z)} \right] \\
&= \sum_{(x,y,z) \in A} p(x,y,z) {}^2\log \left[ \frac{p(x,z)p(y,z)}{p(x,z)p(y,z)} \right] = 0
\end{aligned}$$

The proof subsequently follows from property 6 (which states that  $I(X, Y|Z) \geq 0$ ):  $I(X, Y) = I(X, Z) + I(X, Y|Z) \geq I(X, Z)$ .  $\square$

The data processing inequality shows explicitly that no processing of the random variable  $y$  (converting it into a new random variable  $z$ ), whether deterministic or probabilistic, can increase the information that it contains about the random variable  $x$  (which obviously makes sense).

### 3.3.2 Examples

*Example 1.* As usual we try to facilitate the digestion of all this via explicit examples. Let us first illustrate the relative entropy by considering the simplest non-trivial discrete variables:  $x \in A = \{0, 1\}$ . We compare two probability distributions,  $p(x)$  and  $q(x)$ , where

$$p(0) = 1-p, \quad p(1) = p \qquad q(0) = 1-q, \quad q(1) = q$$

Upon inserting the above probabilities into the definition (3.9) we obtain the following expressions for the relative entropies  $D(p||q)$  and  $D(q||p)$ :

$$D(p||q) = (1-p) {}^2\log \left[ \frac{1-p}{1-q} \right] + p {}^2\log \left[ \frac{p}{q} \right] \qquad D(q||p) = (1-q) {}^2\log \left[ \frac{1-q}{1-p} \right] + q {}^2\log \left[ \frac{q}{p} \right]$$

We can find the minimum of, for instance,  $D(p||q)$  by calculating its derivative with respect to the parameter  $q$  for a given value of  $p$ :

$$\begin{aligned}
\frac{\partial}{\partial q} D(p||q) &= -\frac{\partial}{\partial q} \left\{ (1-p) {}^2\log(1-q) + p {}^2\log(q) \right\} \\
&= \frac{1}{\log 2} \left\{ \frac{1-p}{1-q} - \frac{p}{q} \right\} = \frac{q-p}{(1-q)q \log 2}
\end{aligned}$$

Since  $\frac{\partial}{\partial q} D(p||q) > 0$  for  $q > p$ ,  $\frac{\partial}{\partial q} D(p||q) < 0$  for  $q < p$  and  $\frac{\partial}{\partial q} D(p||q) = 0$  for  $q = p$ , the minimum of  $D(p||q)$  is obtained for  $p = q$ , giving  $D(p||p) = 0$  (as it should).

The effect of the asymmetry in the definition of  $D(p||q)$  can be illustrated by simply working out the above expressions for  $D(p||q)$  and  $D(q||p)$  for the choice  $p = \frac{1}{2}$  and  $q = \frac{1}{4}$ , where we find

$$p = \frac{1}{2}, \quad q = \frac{1}{4} : \quad D(p||q) = \frac{1}{2} {}^2\log \left[ \frac{1/2}{3/4} \right] + \frac{1}{2} {}^2\log \left[ \frac{1/2}{1/4} \right] = 1 - \frac{1}{2} {}^2\log 3 \approx 0.2075$$

$$p = \frac{1}{2}, \quad q = \frac{1}{4} : \quad D(q||p) = \frac{3}{4} {}^2\log \left[ \frac{3/4}{1/2} \right] + \frac{1}{4} {}^2\log \left[ \frac{1/4}{1/2} \right] = \frac{3}{4} {}^2\log 3 - 1 \approx 0.1887$$

Although both relative entropies are minimised only for  $p = q$  (in which case both are identical zero), in general one clearly has  $D(p||q) \neq D(q||p)$ .

*Example 2.* Our second example serves to illustrate that in general  $I(X, Y) \neq I(X, Y|Z)$ , and that there is even no inequality to order the two. Consider the trio of stochastic variables  $(x, y, z)$  with  $A_x = A_y = A_z = \{0, 1\}$  (so  $A = \{0, 1\}^3$ ) and with joint probabilities  $p(x, y, z)$  as given in the following table:

$(x, y, z)$	(0,0,0)	(1,0,0)	(0,1,0)	(1,1,0)	(0,0,1)	(1,0,1)	(0,1,1)	(1,1,1)
$p(x, y, z)$	1/4	0	0	1/4	0	1/4	1/4	0

By summing over one of the random variables we obtain from these the marginal probabilities  $p(x, y) = \sum_{z \in A_z} p(x, y, z)$ ,  $p(y, z) = \sum_{x \in A_x} p(x, y, z)$  and  $p(x, z) = \sum_{y \in A_y} p(x, y, z)$ , which for this example all come out identical:

$$p(x, y) = p(x, z) = p(y, z) = \frac{1}{4} \quad \text{for all } (x, y, z) \in A$$

The marginal probabilities  $p(x) = \sum_{y \in A_y} p(x, y)$ ,  $p(y) = \sum_{z \in A_z} p(y, z)$  and  $p(z) = \sum_{y \in A_y} p(y, z)$  come out identical as well

$$p(x) = p(y) = p(z) = \frac{1}{2} \quad \text{for all } (x, y, z) \in A$$

If we calculate for the present example the mutual information  $I(X, Y)$  (3.11) and the conditional mutual information  $I(X, Y|Z)$  (3.12) we find:

$$\begin{aligned} I(X, Y) &= \sum_{(x,y) \in A_x \otimes A_y} p(x, y) {}^2\log \left[ \frac{p(x, y)}{p(x)p(y)} \right] = \frac{1}{4} \sum_{(x,y) \in A_x \otimes A_y} {}^2\log [1] = 0 \\ I(X, Y|Z) &= \sum_{(x,y,z) \in A} p(x, y, z) {}^2\log \left[ \frac{p(x, y|z)}{p(x|z)p(y|z)} \right] = \sum_{(x,y,z) \in A} p(x, y, z) {}^2\log \left[ \frac{p(x, y, z)p(z)}{p(x, z)p(y, z)} \right] \\ &= \sum_{(x,y,z) \in A} p(x, y, z) {}^2\log [8p(x, y, z)] = {}^2\log [2] = 1 \end{aligned}$$

At first sight it seems somewhat strange to find simultaneously  $I(X, Y|Z) > 0$  and  $I(X, Y) = 0$ . Apparently  $y$  reveals nothing about  $x$  if we do not take  $z$  into account, but  $y$  does reveal something about  $x$  when in addition  $z$  is known. The explanation is as follows. If we inspect the probability distributions  $p(x, y)$ ,  $p(x)$  and  $p(y)$  we see that  $x$  and  $y$  are independent, so  $I(X, Y) = 0$  (as it should). However, if  $z$  is known the situation changes considerably. From the table of  $p(x, y, z)$  we infer the following: if  $z = 1$  we know that  $(x, y) \in \{(0, 1), (1, 0)\}$  (therefore knowing one means knowing the other), if  $z = 0$  we know that  $(x, y) \in \{(0, 0), (1, 1)\}$  (again knowing one means knowing the other). This explains why  $I(X, Y|Z) > 0$ .

Now we will show that the opposite situation can occur as well, with  $I(X, Y) > 0$  and  $I(X, Y|Z) = 0$ . We change the table of the joint probabilities in the following way:

$(x, y, z)$	(0,0,0)	(1,0,0)	(0,1,0)	(1,1,0)	(0,0,1)	(1,0,1)	(0,1,1)	(1,1,1)
$p(x, y, z)$	0	1/2	0	0	0	0	1/2	0

By summing over one of the random variables we obtain from these the marginal probabilities  $p(x, y) = \sum_{z \in A_z} p(x, y, z)$ ,  $p(y, z) = \sum_{x \in A_x} p(x, y, z)$  and  $p(x, z) = \sum_{y \in A_y} p(x, y, z)$ :

$(x, y)$	(0,0)	(1,0)	(0,1)	(1,1)
$p(x, y)$	0	1/2	1/2	0

$(y, z)$	(0,0)	(1,0)	(0,1)	(1,1)
$p(y, z)$	1/2	0	0	1/2

$(x, z)$	(0,0)	(1,0)	(0,1)	(1,1)
$p(x, z)$	0	1/2	1/2	0

The marginal probabilities  $p(x) = \sum_{y \in A_y} p(x, y)$ ,  $p(y) = \sum_{z \in A_z} p(y, z)$  and  $p(z) = \sum_{y \in A_y} p(y, z)$  again come out identical:

$$p(x) = p(y) = p(z) = \frac{1}{2} \quad \text{for all } (x, y, z) \in A$$

If we now calculate the mutual information  $I(X, Y)$  (3.11) and the conditional mutual information  $I(X, Y|Z)$  we find:

$$I(X, Y) = \sum_{(x,y) \in A_x \otimes A_y} p(x, y) \, {}^2\log \left[ \frac{p(x, y)}{p(x)p(y)} \right] = {}^2\log \left[ \frac{1/2}{(1/2)^2} \right] = {}^2\log[2] = 1$$

$$I(X, Y|Z) = \sum_{(x,y,z) \in A} p(x, y, z) \, {}^2\log \left[ \frac{p(x, y|z)}{p(x|z)p(y|z)} \right] = \sum_{(x,y,z) \in A} p(x, y, z) \, {}^2\log \left[ \frac{p(x, y, z)p(z)}{p(x, z)p(y, z)} \right]$$

$$= \frac{1}{2} \left\{ {}^2\log \left[ \frac{(1/2)^2}{(1/2)^2} \right] + {}^2\log \left[ \frac{(1/2)^2}{(1/2)^2} \right] \right\} = 0$$

Now the situation is reversed ! Apparently  $y$  reveals information about  $x$  if we do not take  $z$  into account, but fails to do so when in addition  $z$  is known. The explanation is as follows. If we inspect the probability distributions  $p(x, y)$ ,  $p(x)$  and  $p(y)$  we see that  $x$  and  $y$  are not independent, so  $I(X, Y) > 0$  (as it should). However, if  $z$  is known the situation changes. From the table of  $p(x, y, z)$  we infer the following: if  $z = 1$  it immediately follows that  $(x, y) = (0, 1)$  (we know everything already, and knowing  $x$  adds nothing to what we know about  $y$ ), if  $z = 0$  it follows that  $(x, y) = (1, 0)$  (again all is already revealed, and knowing  $x$  does not increase our knowledge of  $y$ ). This explains why  $I(X, Y|Z) = 0$ .

### 3.4 Entropy and Mutual Information for Continuous Random Variables

So far we have restricted ourselves to discrete random variables. In the case where the variables are continuous we have to be somewhat careful, since not all properties of our information-theoretic quantities as established for discrete random variables survive the transition to continuous variables.

### 3.4.1 Differential Entropy

Let us consider a discrete random variable  $x$  which can assume the values  $x_k = k\delta x$  (for a given spacing  $\delta x$ ), with  $k = 0, \pm 1, \pm 2, \dots$  and associated probabilities  $\hat{p}(x_k)$ , so  $A = \{\dots, -3\delta x, -2\delta x, -\delta x, 0, \delta x, 2\delta x, 3\delta x, \dots\}$ . Eventually we will consider the limit  $\delta x \rightarrow 0$  which will turn  $x$  into a continuous random variable. As long as  $\delta x$  remains finite, however, the theory developed so far applies, and we obtain

$$H(X) = - \sum_{k=-\infty}^{\infty} \hat{p}(x_k) {}^2\log \hat{p}(x_k), \quad \sum_{k=-\infty}^{\infty} \hat{p}(x_k) = 1 \quad (3.13)$$

In order to pave the way for the continuum limit we now define a probability *density*  $p(x_k) = \hat{p}(x_k)/\delta x$ . Transformation of (3.13) into an expression involving this density gives

$$H(X) = - \sum_{k=-\infty}^{\infty} \delta x p(x_k) {}^2\log p(x_k) + {}^2\log \left[ \frac{1}{\delta x} \right], \quad \sum_{k=-\infty}^{\infty} \delta x p(x_k) = 1$$

Note that

$$\lim_{\delta x \rightarrow 0} \sum_{k=-\infty}^{\infty} \delta x p(x_k) {}^2\log p(x_k) = \int dx p(x) {}^2\log p(x), \quad \lim_{\delta x \rightarrow 0} \sum_{k=-\infty}^{\infty} \delta x p(x_k) = \int dx p(x) = 1$$

(provided these integrals exist). However, we observe that we cannot define the entropy of a continuous random variable simply as the continuum limit  $\delta x \rightarrow 0$  of the entropy of an underlying discrete random variable, since  $\lim_{\delta x \rightarrow 0} {}^2\log[1/\delta x] = \infty$ . The natural adaptation of the discrete definition of the entropy to the case of a continuous random variable described by the probability density  $p(x)$ , with  $\int dx p(x) = 1$ , appears to be restricting ourselves to what is left after we eliminate the offending term  ${}^2\log[1/\delta x]$  from the discrete expression, giving:

$$\tilde{H}(X) = - \int dx p(x) {}^2\log p(x) \quad (3.14)$$

which is called the *differential entropy*. Since in order to arrive at (3.14) we have subtracted a positive term from a discrete entropy  $H(X)$ , we may no longer assume  $\tilde{H}(X) \geq 0$ . Yet, since what has been subtracted is simply a constant which does not depend on the shape of the probability density  $p(x)$ , the differential entropy still has the property that it measures information content (although in a relative rather than an absolute sense; like a thermometer with marks but without an indication of where the zero is).

*Example 1.* Let us calculate the differential entropy for a block-shaped probability density, with  $A = [a - \frac{1}{2}b, a + \frac{1}{2}b]$ :

$$\begin{aligned} p(x) &= b^{-1} & \text{for } a - \frac{1}{2}b \leq x \leq a + \frac{1}{2}b \\ p(x) &= 0 & \text{elsewhere} \end{aligned}$$

For the differential entropy we find:

$$\tilde{H}(X) = -\frac{1}{b} \int_{a-\frac{1}{2}b}^{a+\frac{1}{2}b} dx {}^2\log[1/b] = {}^2\log[b]$$

The result is negative for  $b < 1$  (narrow densities). We conclude that the differential entropy can indeed have negative values, even for simple and well-behaved probability densities. We also observe that for continuous random variables the familiar rule  $\tilde{H}(X) = {}^2\log|A|$  again holds, where  $|A|$  is now defined as the size of the interval of allowed values for  $x$ .

Generalisation to multivariate densities is straightforward. Let us now consider  $\mathbf{x} \in \mathfrak{R}^N$ , with the set  $A$  centered at  $(a_1, \dots, a_N)$ , i.e.  $A = [a_1 - \frac{1}{2}b_1, a_1 + \frac{1}{2}b_1] \otimes \dots \otimes [a_N - \frac{1}{2}b_N, a_N + \frac{1}{2}b_N]$ :

$$p(\mathbf{x}) = \prod_{i=1}^N b_i^{-1} \quad \text{for } a_1 - \frac{1}{2}b_1 \leq x_1 \leq a_1 + \frac{1}{2}b_1 \wedge \dots \wedge a_N - \frac{1}{2}b_N \leq x_N \leq a_N + \frac{1}{2}b_N$$

$$p(\mathbf{x}) = 0 \quad \text{elsewhere}$$

For the differential entropy we find:

$$\tilde{H}(X_1, \dots, X_N) = - \left[ \prod_{i=1}^N b_i^{-1} \right] \int_A d\mathbf{x} \ {}^2\log \prod_{i=1}^N b_i^{-1} = {}^2\log \prod_{i=1}^N b_i = \sum_{i=1}^N {}^2\log b_i$$

The result is negative for  $\prod_{i=1}^N b_i < 1$  (sharply concentrated densities). Again we find  $\tilde{H}(X_1, \dots, X_N) = {}^2\log|A|$ , where  $|A|$  is now defined as the volume of the region of allowed values for  $\mathbf{x}$ .

*Example 2.* Our second example is a probability density with a Gaussian shape (see also appendix B):

$$p(x) = \frac{e^{-\frac{1}{2}[x-\mu]^2/\sigma^2}}{\sigma\sqrt{2\pi}}, \quad A = \langle -\infty, \infty \rangle \tag{3.15}$$

Here the differential entropy is found to be

$$\begin{aligned} \tilde{H}(X) &= - \int \frac{dx}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}[x-\mu]^2/\sigma^2} \ {}^2\log \left[ \frac{e^{-\frac{1}{2}[x-\mu]^2/\sigma^2}}{\sigma\sqrt{2\pi}} \right] \\ &= \int \frac{dx [x-\mu]^2}{2\sigma^3\sqrt{2\pi} \log 2} e^{-\frac{1}{2}[x-\mu]^2/\sigma^2} + {}^2\log [\sigma\sqrt{2\pi}] = \frac{1}{2 \log 2} \left\{ 1 + \log [2\pi\sigma^2] \right\} \end{aligned}$$

We find a negative differential entropy for sufficiently small width  $\sigma$  of the probability density, and an infinite differential entropy for  $\sigma \rightarrow \infty$ .

There is one special property to be mentioned here, which further supports the importance of Gaussian distributions. Let us try to answer the following question: which continuous probability distribution, with fixed average  $\mu$  and variance  $\sigma^2$ , is the one with maximum differential entropy? The constraint of a fixed variance is necessary, since we have already seen that there is no upper bound to the differential entropy (just put  $\sigma \rightarrow \infty$  in the Gaussian distribution). We can solve this functional optimization problem with Lagrange multipliers, defining the three constraints (average, variance and normalization) as

$$C_0(\{p\}) = C_1(\{p\}) = C_2(\{p\}) = 0$$

$$C_0(\{p\}) = \int dx p(x) - 1, \quad C_1(\{p\}) = \int dx p(x)x - \mu, \quad C_2(\{p\}) = \int dx p(x)x^2 - \sigma^2 - \mu^2$$

The solution is then to be solved from:

$$\frac{\delta}{\delta p(x)} \left[ \lambda_0 C_0(\{p\}) + \lambda_1 C_1(\{p\}) + \lambda_2 C_2(\{p\}) - \int dx p(x) \log p(x) \right] = 0$$

$$\forall x \in \mathfrak{R} : \quad \lambda_0 + \lambda_1 x + \lambda_2 x^2 = \log p(x) + 1 \quad \text{so} \quad \forall x \in \mathfrak{R} : \quad p(x) = e^{\lambda_0 - 1 + \lambda_1 x + \lambda_2 x^2}$$

Thus the maximum entropy distribution  $p(x)$  is found to be Gaussian, with the  $\{\lambda_0, \lambda_1, \lambda_2\}$  following from the constraints, which immediately returns us back to (3.15).

Multivariate Gaussian distributions, where  $\mathbf{x} \in \mathfrak{R}^N$ , pose no fundamental problems, although the calculations are slightly more involved. The fundamental properties one always uses are normalisation and the expression for the second order moments:

$$P(\mathbf{x}) = \frac{e^{-\frac{1}{2}(\mathbf{x}-\langle\mathbf{x}\rangle)\cdot\mathbf{A}(\mathbf{x}-\langle\mathbf{x}\rangle)}}{(2\pi)^{N/2}\det^{-\frac{1}{2}}\mathbf{A}}, \quad (\mathbf{A}^{-1})_{ij} = \langle(x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle)\rangle \quad (3.16)$$

Let us denote the (real and non-negative) eigenvalues of the (symmetric) covariance matrix  $\mathbf{A}^{-1}$  as  $\{c_i\}$ . Insertion into the definition (3.14) of the differential entropy gives:

$$\begin{aligned} \tilde{H}(X_1, \dots, X_N) &= - \int d\mathbf{x} \frac{e^{-\frac{1}{2}(\mathbf{x}-\langle\mathbf{x}\rangle)\cdot\mathbf{A}(\mathbf{x}-\langle\mathbf{x}\rangle)}}{(2\pi)^{N/2}\det^{-\frac{1}{2}}\mathbf{A}} \quad {}^2\log \frac{e^{-\frac{1}{2}(\mathbf{x}-\langle\mathbf{x}\rangle)\cdot\mathbf{A}(\mathbf{x}-\langle\mathbf{x}\rangle)}}{(2\pi)^{N/2}\det^{-\frac{1}{2}}\mathbf{A}} \\ &= \frac{1}{2\log 2} \int d\mathbf{x} \frac{e^{-\frac{1}{2}(\mathbf{x}-\langle\mathbf{x}\rangle)\cdot\mathbf{A}(\mathbf{x}-\langle\mathbf{x}\rangle)}}{(2\pi)^{N/2}\det^{-\frac{1}{2}}\mathbf{A}} (\mathbf{x}-\langle\mathbf{x}\rangle) \cdot \mathbf{A}(\mathbf{x}-\langle\mathbf{x}\rangle) + {}^2\log \left[ (2\pi)^{N/2} \det^{-\frac{1}{2}} \mathbf{A} \right] \\ &= \frac{1}{2\log 2} \langle (\mathbf{x}-\langle\mathbf{x}\rangle) \cdot \mathbf{A}(\mathbf{x}-\langle\mathbf{x}\rangle) \rangle + {}^2\log \left[ (2\pi)^{N/2} \sqrt{\prod_i c_i} \right] \end{aligned}$$

(using  $\det^{-1} \mathbf{A} = \det \mathbf{A}^{-1} = \prod_i c_i$ )

$$\begin{aligned} &= \frac{1}{2\log 2} \sum_{ij=1}^N A_{ij} \langle (x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) \rangle + \frac{1}{2} \sum_{i=1}^N {}^2\log [2\pi c_i] \\ &= \frac{1}{2\log 2} \sum_{ij=1}^N \delta_{ij} + \frac{1}{2} \sum_{i=1}^N {}^2\log [2\pi c_i] \end{aligned}$$

(where we have used (3.16)), giving the final result:

$$\tilde{H}(X_1, \dots, X_N) = \frac{1}{2\log 2} \sum_{i=1}^N \{1 + \log [2\pi c_i]\} \quad (3.17)$$

The simplest type of covariance matrix is a diagonal one, describing independent variables  $\{x_i\}$ :  $\langle (x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) \rangle = \sigma_i^2 \delta_{ij}$ , and  $A_{ij} = \sigma_i^{-2} \delta_{ij}$  (with the variances  $\sigma_i$  of the individual variables  $x_i$ ). Here the differential entropy reduces to the sum of the differential entropies of the  $N$  individual  $x_i$ , as it should:

$$\tilde{H}(X_1, \dots, X_N) = \frac{1}{2\log 2} \sum_{i=1}^N \{1 + \log [2\pi \sigma_i^2]\} = \sum_{i=1}^N \tilde{H}(X_i)$$



### 3.4.2 Differential Mutual Information

Let us now consider two discrete random variables  $x$  and  $y$ , which can assume the values  $x_k = k\delta x$  (for a given spacing  $\delta x$ ), with  $k = 0, \pm 1, \pm 2, \dots$ , and  $y_\ell = \ell\delta y$  (for a given spacing  $\delta y$ ), with  $\ell = 0, \pm 1, \pm 2, \dots$ , respectively. The associated joint probabilities are  $\hat{p}(x_k, y_\ell)$ , with marginal probabilities  $\hat{p}(x_k) = \sum_{\ell=-\infty}^{\infty} \hat{p}(x_k, y_\ell)$  and  $\hat{p}(y_\ell) = \sum_{k=-\infty}^{\infty} \hat{p}(x_k, y_\ell)$ . As long as both  $\delta x$  and  $\delta y$  remain finite the theory for discrete random variables applies, and we obtain

$$I(X, Y) = \sum_{k, \ell=-\infty}^{\infty} \hat{p}(x_k, y_\ell) {}^2\log \left[ \frac{\hat{p}(x_k, y_\ell)}{\hat{p}(x_k)\hat{p}(y_\ell)} \right], \quad \sum_{k, \ell=-\infty}^{\infty} \hat{p}(x_k, y_\ell) = 1 \quad (3.18)$$

As in the section dealing with the differential entropy we now define joint and marginal probability *densities*  $p(x_k, y_\ell) = \hat{p}(x_k, y_\ell)/\delta x\delta y$ ,  $p(x_k) = \sum_{\ell=-\infty}^{\infty} \hat{p}(x_k, y_\ell)/\delta x$  and  $p(y_\ell) = \sum_{k=-\infty}^{\infty} \hat{p}(x_k, y_\ell)/\delta y$ . Here the transformation of (3.18) into an expression involving densities only does not generate diverging terms; the spacing parameters  $\delta x$  and  $\delta y$  are cancelled in the argument of the logarithm, and we find

$$I(X, Y) = \sum_{k, \ell=-\infty}^{\infty} \delta x\delta y p(x_k, y_\ell) {}^2\log \left[ \frac{p(x_k, y_\ell)}{p(x_k)p(y_\ell)} \right], \quad \sum_{k, \ell=-\infty}^{\infty} \delta x\delta y p(x_k, y_\ell) = 1$$

Apparently we can now simply define the mutual information of two continuous random variables as the continuum limit  $\delta x \rightarrow 0$ ,  $\delta y \rightarrow 0$  of the mutual information of a pair of underlying discrete random variables:

$$\tilde{I}(X, Y) = \int dx dy p(x, y) {}^2\log \left[ \frac{p(x, y)}{p(x)p(y)} \right] \quad (3.19)$$

(provided the integrals exist).  $\tilde{I}(X, Y)$  is called the *differential mutual information*. Since the differential mutual information can be obtained by taking a limit of a discrete expression for mutual information, we are now guaranteed that  $\tilde{I}(X, Y) \geq 0$ .

*Example 1.* Let us work out the mutual information of two (possibly correlated) Gaussian variables  $x$  and  $y$  with zero averages (for simplicity),  $\langle x \rangle = \langle y \rangle = 0$ , and variances  $\langle x^2 \rangle = \sigma_x^2$  and  $\langle y^2 \rangle = \sigma_y^2$ :

$$p(x, y) = \frac{e^{-\frac{1}{2} \begin{pmatrix} x \\ y \end{pmatrix} \cdot \mathbf{A} \begin{pmatrix} x \\ y \end{pmatrix}}}{2\pi \det^{-\frac{1}{2}} \mathbf{A}}, \quad \mathbf{A}^{-1} = \begin{pmatrix} \sigma_x^2 & \langle xy \rangle \\ \langle xy \rangle & \sigma_y^2 \end{pmatrix}$$

with the marginal probability densities

$$p(x) = \frac{e^{-\frac{1}{2}x^2/\sigma_x^2}}{\sigma_x\sqrt{2\pi}} \quad p(y) = \frac{e^{-\frac{1}{2}y^2/\sigma_y^2}}{\sigma_y\sqrt{2\pi}}$$

For the differential mutual information we find:

$$\tilde{I}(X, Y) = \int dx dy p(x, y) {}^2\log \exp \left\{ \frac{1}{2} \left[ \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} - \begin{pmatrix} x \\ y \end{pmatrix} \cdot \mathbf{A} \begin{pmatrix} x \\ y \end{pmatrix} \right] \right\} + {}^2\log [\sigma_x\sigma_y\sqrt{\det \mathbf{A}}]$$

$$\begin{aligned}
 &= \frac{1}{2 \log 2} \int dx dy p(x, y) \left\{ x^2 \sigma_x^{-2} + y^2 \sigma_y^{-2} - \begin{pmatrix} x \\ y \end{pmatrix} \cdot \mathbf{A} \begin{pmatrix} x \\ y \end{pmatrix} \right\} + {}^2\log [\sigma_x \sigma_y \sqrt{\det \mathbf{A}}] \\
 &= \frac{1}{2 \log 2} \left\{ \langle x^2 \rangle \sigma_x^{-2} + \langle y^2 \rangle \sigma_y^{-2} - \sum_{i,j=1}^2 A_{ij}^{-1} A_{ji} \right\} + {}^2\log [\sigma_x \sigma_y \sqrt{\det \mathbf{A}}] = {}^2\log [\sigma_x \sigma_y \sqrt{\det \mathbf{A}}] \\
 &= {}^2\log \left[ \frac{\sigma_x \sigma_y}{\sqrt{\sigma_x^2 \sigma_y^2 - \langle xy \rangle^2}} \right] = \frac{1}{2} {}^2\log \left[ \frac{\sigma_x^2 \sigma_y^2}{\sigma_x^2 \sigma_y^2 - \langle xy \rangle^2} \right] \\
 &= -\frac{1}{2 \log 2} \log \left[ 1 - \frac{\langle xy \rangle^2}{\sigma_x^2 \sigma_y^2} \right] \tag{3.20}
 \end{aligned}$$

*Example 2.* Our second example is a pair of random variables  $(x, y)$  which are only allowed to take values from the square,  $A_x = A_y = [-1, 1]$ , with joint probability density:

$$\begin{aligned}
 (x, y) \notin [-1, 1]^2 &: p(x, y) = 0 \\
 (x, y) \in [-1, 1]^2 &: p(x, y) = \frac{k + \theta(xy)}{2(2k+1)}
 \end{aligned}$$

with the step function,  $\theta[z > 0] = 1$  and  $\theta[z < 0] = 0$ , and with a parameter  $k \geq 0$  which controls the degree of dependence of the two variables (for  $k \rightarrow \infty$  the two random variables become independent; for  $k = 0$  we find a strong coupling, since  $x$  and  $y$  are forced to have the same sign). The marginal probability densities are obtained by integration:

$$x, y \in [-1, 1] : p(x) = \int_{-1}^1 dy \frac{k + \theta(xy)}{2(2k+1)} = \frac{1}{2}, \quad p(y) = \int_{-1}^1 dx \frac{k + \theta(xy)}{2(2k+1)} = \frac{1}{2}$$

The differential mutual information can be calculated easily:

$$\begin{aligned}
 \tilde{I}(X, Y) &= \int_{-1}^1 dx dy \frac{k + \theta(xy)}{2(2k+1)} {}^2\log \left[ \frac{2[k + \theta(xy)]}{2k+1} \right] \\
 &= \frac{k+1}{2k+1} {}^2\log \left[ \frac{2k+2}{2k+1} \right] + \frac{k}{2k+1} {}^2\log \left[ \frac{2k}{2k+1} \right]
 \end{aligned}$$

This result makes sense. For  $k \rightarrow 0$  we get  $\tilde{I}(X, Y) \rightarrow 1$  (here the two variables are forced to have identical sign, so they do indeed convey exactly one bit of information about one another). For  $k \rightarrow \infty$  we get  $\tilde{I}(X, Y) \rightarrow 0$  (here the two variables are independent).

*Example 3.* As a third and final example we will calculate the differential mutual information of two (possibly correlated) Gaussian variables which are themselves vectors, rather than scalars:  $\mathbf{x} \in \Re^N$  and  $\mathbf{y} \in \Re^M$  with zero averages (for simplicity),  $\langle \mathbf{x} \rangle = \langle \mathbf{y} \rangle = 0$ . The joint probability distribution for the pair  $(\mathbf{x}, \mathbf{y})$  must then be

$$p(\mathbf{x}, \mathbf{y}) = \frac{e^{-\frac{1}{2} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \cdot \mathbf{A} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}}}{(2\pi)^{\frac{N+M}{2}} \det^{-\frac{1}{2}} \mathbf{A}}, \quad \mathbf{A}^{-1} = \begin{pmatrix} \mathbf{C}^{xx} & \mathbf{C}^{xy} \\ \mathbf{C}^{yx} & \mathbf{C}^{yy} \end{pmatrix}$$

with the matrices:

$$C_{ij}^{xx} = \langle x_i x_j \rangle, \quad C_{ij}^{xy} = \langle x_i y_j \rangle, \quad C_{ij}^{yx} = \langle y_i x_j \rangle, \quad C_{ij}^{yy} = \langle y_i y_j \rangle$$

and the marginal densities

$$p(\mathbf{x}) = \frac{e^{-\frac{1}{2}\mathbf{x} \cdot (\mathbf{C}^{xx})^{-1} \mathbf{x}}}{(2\pi)^{N/2} \det^{\frac{1}{2}} \mathbf{C}^{xx}}, \quad p(\mathbf{y}) = \frac{e^{-\frac{1}{2}\mathbf{y} \cdot (\mathbf{C}^{yy})^{-1} \mathbf{y}}}{(2\pi)^{M/2} \det^{\frac{1}{2}} \mathbf{C}^{yy}}$$

For the differential mutual information we find:

$$\begin{aligned} \tilde{I}(\mathbf{X}, \mathbf{Y}) &= \int d\mathbf{x} d\mathbf{y} p(\mathbf{x}, \mathbf{y}) {}^2\log \left[ \frac{p(\mathbf{x}, \mathbf{y})}{p(\mathbf{x})p(\mathbf{y})} \right] \\ &= \int d\mathbf{x} d\mathbf{y} p(\mathbf{x}, \mathbf{y}) {}^2\log e^{\frac{1}{2}\mathbf{x} \cdot (\mathbf{C}^{xx})^{-1} \mathbf{x} + \frac{1}{2}\mathbf{y} \cdot (\mathbf{C}^{yy})^{-1} \mathbf{y} - \frac{1}{2} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \cdot \mathbf{A} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}} + \frac{1}{2} {}^2\log [\det \mathbf{C}^{yy} \det \mathbf{C}^{xx} \det \mathbf{A}] \\ &= \int \frac{d\mathbf{x} d\mathbf{y} p(\mathbf{x}, \mathbf{y})}{2 \log 2} \left[ \mathbf{x} \cdot (\mathbf{C}^{xx})^{-1} \mathbf{x} + \mathbf{y} \cdot (\mathbf{C}^{yy})^{-1} \mathbf{y} - \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \cdot \mathbf{A} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \right] + \frac{1}{2} {}^2\log [\det \mathbf{C}^{yy} \det \mathbf{C}^{xx} \det \mathbf{A}] \\ &= \frac{1}{2 \log 2} \left[ \sum_{ij=1}^N C_{ij}^{xx} (C^{xx})_{ji}^{-1} + \sum_{ij=1}^M C_{ij}^{yy} (C^{yy})_{ji}^{-1} - \sum_{ij=1}^{N+M} A_{ij} A_{ij}^{-1} \right] + \frac{1}{2} {}^2\log [\det \mathbf{C}^{yy} \det \mathbf{C}^{xx} \det \mathbf{A}] \\ &= \frac{1}{2} {}^2\log [\det \mathbf{C}^{yy} \det \mathbf{C}^{xx} \det \mathbf{A}] \end{aligned}$$

Equivalently:

$$\begin{aligned} \tilde{I}(\mathbf{X}, \mathbf{Y}) &= -\frac{1}{2} {}^2\log \det \left[ \begin{pmatrix} (\mathbf{C}^{xx})^{-1} & \emptyset \\ \emptyset & (\mathbf{C}^{yy})^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{C}^{xx} & \mathbf{C}^{xy} \\ \mathbf{C}^{yx} & \mathbf{C}^{yy} \end{pmatrix} \right] \\ &= -\frac{1}{2} {}^2\log \det \begin{pmatrix} \mathbf{I} & (\mathbf{C}^{xx})^{-1} \mathbf{C}^{xy} \\ (\mathbf{C}^{yy})^{-1} \mathbf{C}^{yx} & \mathbf{I} \end{pmatrix} \end{aligned} \quad (3.21)$$

in which  $\mathbf{I}$  denotes the identity matrix. Note that the previous result (3.20) can be recovered as a special case from (3.21) since for  $N = M = 1$  we just obtain  $\mathbf{I} = 1$ ,  $\mathbf{C}^{xx} = \langle x^2 \rangle$ ,  $\mathbf{C}^{yy} = \langle y^2 \rangle$  and  $\mathbf{C}^{xy} = \mathbf{C}^{yx} = \langle xy \rangle$ .

# Chapter 4

## Identification of Entropy as a Measure of Information

In this chapter we will finally anchor the whole framework developed so far, and prove that the entropy (on which all is built) indeed represents the information content of random variables. We will do this by showing that if we use the optimal code to represent the possible values of a given random variable (i.e. the one with the smallest average length of codewords) the average number of bits used per message is equal to the entropy  $H$ .

### 4.1 Coding Theory

For simplicity we will restrict ourselves to binary codes, i.e. to those that use only symbols from the set  $\{0, 1\}$ . Generalisation/adaptation of the theory to families of codes which employ a larger alphabet of symbols is straightforward.

#### 4.1.1 Definitions

First we will give some definitions of codes and their properties:

**definition 1:** A binary code  $C : A \rightarrow \bigcup_{L \geq 1} \{0, 1\}^L$  is a mapping from the set  $A$  of all messages to the set of all binary strings of non-zero length. It associates a codeword  $C(x) \in \bigcup_{L \geq 1} \{0, 1\}^L$  to each message  $x \in A$ .

**definition 2:** A non-singular binary code  $C$  is a binary code with the property that if  $x, x' \in A$ , with  $x \neq x'$ , then  $C(x) \neq C(x')$ . Different messages are always given different codewords, so each individual codeword is uniquely decodable.

**definition 3:** The length  $\ell(x)$  of codeword  $C(x)$  is defined as the number of symbols in the string  $C(x)$ , i.e.  $\ell(x) = \ell$  if and only if  $C(x) \in \{0, 1\}^\ell$ .

**definition 4:** The code-length  $L[C]$  of a binary code  $C$  is the average length of its codewords:  $L[C] = \sum_{x \in A} p(x)\ell(x)$ .

**definition 5:** A prefix code is a non-singular binary code  $C$  with the property that no codeword  $C(x)$  is the prefix of another codeword  $C(x')$ , where  $x, x' \in A$  and  $x \neq x'$ .

**definition 6:** The extension  $C^* : A^n \rightarrow \bigcup_{L \geq 1} \{0, 1\}^L$  of a binary code  $C : A \rightarrow \bigcup_{L \geq 1} \{0, 1\}^L$  is a mapping from the set  $A^n$  of all groups  $(x_1, \dots, x_n)$  of  $n$  messages to the set of all binary strings of non-zero length. The codeword  $C^*(x_1, \dots, x_n)$  which  $C^*$  assigns to a group of  $n$  messages  $(x_1, \dots, x_n) \in A^n$  is defined simply as the concatenation of the codewords  $C(x_i)$  of the  $n$  individual messages:  $C^*(x_1, x_2, \dots, x_n) = C(x_1)C(x_2) \cdots C(x_n)$ .

**definition 7:** A uniquely decodable binary code  $C$  is one with the property that its extension  $C^*$  is non-singular for any  $n$ . Note that this also implies that  $C$  must be a prefix code.

Note that the definition of the extended code  $C^*$  precisely covers the situation where one sends several codewords  $C(x)$ , one after the other.  $C^*$  being non-singular then means that (i) the original code  $C$  is non-singular (so the individual codewords  $C(x)$  are uniquely decodable), and in addition (ii) the receiver of a string of codewords  $C(x)$  can always tell when one codeword ends and the next codeword begins. For example:

$$C(x_1) = 00, \quad C(x_2) = 01 : \quad \begin{cases} C^*(x_1, x_1) = 0000 \\ C^*(x_1, x_2) = 0001 \\ C^*(x_2, x_1) = 0100 \\ C^*(x_2, x_2) = 0101 \end{cases}$$

$$C(x_1) = 00, \quad C(x_2) = 01, \quad C(x_3) = 1 : \quad \begin{cases} C^*(x_1, x_1, x_1) = 000000 \\ C^*(x_1, x_1, x_2) = 000001 \\ C^*(x_1, x_1, x_3) = 00001 \\ C^*(x_1, x_2, x_1) = 000100 \\ C^*(x_1, x_2, x_2) = 000101 \\ C^*(x_1, x_2, x_3) = 00011 \\ C^*(x_1, x_3, x_1) = 00100 \\ C^*(x_1, x_3, x_2) = 00101 \\ C^*(x_1, x_3, x_3) = 0011 \\ C^*(x_2, x_1, x_1) = 010000 \\ C^*(x_2, x_1, x_2) = 010001 \\ C^*(x_2, x_1, x_3) = 01001 \\ \dots \end{cases}$$

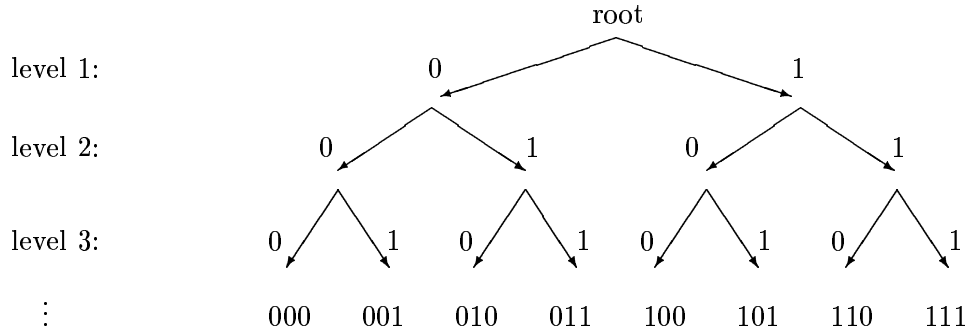
### 4.1.2 The Kraft Inequality

We now define the Kraft inequality for codeword lengths  $\{\ell(x)\}$ , and prove firstly that it will hold for any prefix code, and secondly that if a set of proposed codeword lengths  $\{\ell(x)\}$  satisfies the Kraft inequality, then (conversely) there always exists a prefix code  $C$  with precisely these codeword lengths  $\{\ell(x)\}$ . For a given set of codeword lengths  $\{\ell(x)\}$  the Kraft inequality is defined as

$$\sum_{x \in A} \left(\frac{1}{2}\right)^{\ell(x)} \leq 1 \quad (4.1)$$

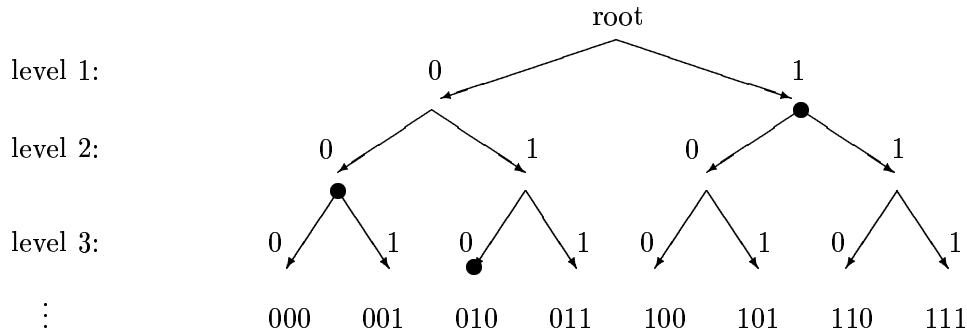
**theorem:** Every prefix code  $C$  satisfies the Kraft inequality (4.1).

**proof:** In order to prove this we first build a tree representation of all possible binary strings, in the following way:



This tree extends downward towards infinity. To each branch point in this tree we associate a binary string, which simply represents all bits one runs into if one follows the directed path (see arrows) from the root to that particular branch point. For example, at level 1 in the tree we find the branch points 0 and 1; at level 2 in the tree we have 00, 01, 10 and 11; at level 3 in the tree one finds 000, 001, 010, 011, 100, 101, 110 and 111, etc.

Next we mark with • those strings that occur among the codewords of our prefix code  $C$  (see graph below, in this example the strings 1, 00 and 010 are codewords).



Since  $C$  is a prefix code we know that once we run into a branch point corresponding to a codeword •, there can be no other codeword further down along that particular branch of the tree. Now imagine performing a directed random walk on this graph, starting at the root. Upon arriving at a given branch point the next step is determined as follows:

- branch point is a codeword : stay there
- branch point is not a codeword : move on,  $\text{Prob}(0) = \text{Prob}(1) = \frac{1}{2}$

After  $n$  iterations the walk will either have proceeded to level  $n$  in the graph (if so far no codeword was found along the way), or will have been terminated at some level  $m < n$  in the graph (at a codeword). The probability  $p(c_1c_2\dots)$  of being at a certain branch point  $c_1c_2\dots$  in the graph after  $n$  iterations (with  $c_i \in \{0, 1\}$ ) is thus given by

- branch point with  $n$  bits :  $p(c_1c_2\dots c_n) = \left(\frac{1}{2}\right)^n$
- branch point with  $m < n$  bits, i.e. codeword  $C(x)$  :  $p(c_1c_2\dots c_m) = \left(\frac{1}{2}\right)^{\ell(x)}$

Consequently, conservation of probability at each iteration step (for we cannot disappear from the graph) gives:

$$\forall n \geq 1 : \sum_{\text{codewords with } \ell(x) < n} \left(\frac{1}{2}\right)^{\ell(x)} + \sum_{\text{paths not yet terminated}} \left(\frac{1}{2}\right)^n = 1$$

so:

$$\forall n \geq 1 : \sum_{\text{codewords with } \ell(x) < n} \left(\frac{1}{2}\right)^{\ell(x)} \leq 1$$

From this result immediately follows, by taking the limit  $n \rightarrow \infty$ :

$$\sum_{x \in A} \left(\frac{1}{2}\right)^{\ell(x)} \leq 1$$

which completes our proof.  $\square$

**theorem:** If a set of codeword lengths  $\{\ell_i\}$  satisfies the Kraft inequality (4.1), then there exists a corresponding prefix code.

**proof:** This is proven by explicit construction. We first order the codeword lengths as follows:

$$\ell_1 \leq \ell_2 \leq \ell_3 \leq \dots$$

We next exploit the fact that each binary string can be interpreted as representing an integer number (here with the most significant bits at the left-hand side of the string), and we denote the integer number represented by the binary string  $c$  as  $[c]$ . For example:

binary string :	integer number :	binary string :	integer number :
0	$[0] = 0$	0000	$[0000] = 0$
1	$[1] = 1$	0001	$[0001] = 1$
		0010	$[0010] = 2$
00	$[00] = 0$	0011	$[0011] = 3$
01	$[01] = 1$	0100	$[0100] = 4$
10	$[10] = 2$	0101	$[0101] = 5$
11	$[11] = 3$	0110	$[0110] = 6$
		0111	$[0111] = 7$
000	$[000] = 0$	1000	$[1000] = 8$
001	$[001] = 1$	1001	$[1001] = 9$
010	$[010] = 2$	1010	$[1010] = 10$
011	$[011] = 3$	1011	$[1011] = 11$
100	$[100] = 4$	1100	$[1100] = 12$
101	$[101] = 5$	1101	$[1101] = 13$
110	$[110] = 6$	1110	$[1110] = 14$
111	$[111] = 7$	1111	$[1111] = 15$

Note that the binary string corresponding to a given integer number is uniquely determined, once we specify the required number of symbols in the string (provided such a





### 4.1.3 Examples

*Example 1.* Let us illustrate the construction of prefix codes, described in the previous proof, from a given set of desired codeword lengths which obey the Kraft inequality. We choose the set  $\{\ell_i\} = \{1, 2, 3, 3\}$ . This set obeys the Kraft inequality, since

$$\sum_i \left(\frac{1}{2}\right)^{\ell_i} = \frac{1}{2} + \left(\frac{1}{2}\right)^2 + 2\left(\frac{1}{2}\right)^3 = 1$$

The construction recipe for the corresponding code generates:

construction :	result :
$C_1 = 0$	$C_1 = 0$
$[C_2] = [0] + 1 = 1, \quad C_2 = 1, \quad C_2 \rightarrow 10$	$C_2 = 10$
$[C_3] = [10] + 1 = 3, \quad C_3 = 11, \quad C_3 \rightarrow 110$	$C_3 = 110$
$[C_4] = [110] + 1 = 7, \quad C_3 = 111$	$C_4 = 111$

This is indeed a prefix code, with the required codeword lengths.

*Example 2.* Next we choose the set  $\{\ell_i\} = \{2, 2, 3, 3, 4, 4\}$ . This set also satisfies the Kraft inequality:

$$\sum_i \left(\frac{1}{2}\right)^{\ell_i} = 2 \left[ \left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^3 + \left(\frac{1}{2}\right)^4 \right] = \frac{7}{8}$$

The construction recipe for the corresponding code generates:

construction :	result :
$C_1 = 00$	$C_1 = 00$
$[C_2] = [00] + 1 = 1, \quad C_2 = 01$	$C_2 = 01$
$[C_3] = [01] + 1 = 2, \quad C_3 = 10, \quad C_3 \rightarrow 100$	$C_3 = 100$
$[C_4] = [100] + 1 = 5, \quad C_4 = 101$	$C_4 = 101$
$[C_5] = [101] + 1 = 6, \quad C_5 = 110, \quad C_5 \rightarrow 1100$	$C_5 = 1100$
$[C_6] = [1100] + 1 = 13, \quad C_6 = 1101$	$C_6 = 1101$

This is again a prefix code, with the required codeword lengths.

*Example 3.* Our third example illustrates how the construction fails when the Kraft inequality is not satisfied. Let us choose the set  $\{\ell_i\} = \{1, 2, 2, 2\}$ . This set violates the Kraft inequality:

$$\sum_i \left(\frac{1}{2}\right)^{\ell_i} = \frac{1}{2} + 3\left(\frac{1}{2}\right)^2 = \frac{5}{4}$$

Now the construction recipe for the corresponding code generates:

construction :	result :
$C_1 = 0$	$C_1 = 0$
$[C_2] = [0] + 1 = 1, \quad C_2 = 1, \quad C_2 \rightarrow 10$	$C_2 = 10$
$[C_3] = [10] + 1 = 3, \quad C_3 = 11,$	$C_3 = 11$
$[C_4] = [11] + 1 = 4, \quad \text{not a 2-bit string !}$	$C_4 = ??$

## 4.2 Entropy and Optimal Coding

### 4.2.1 Bounds for Optimal Codes

We will now prove two important theorems. The first one states that no uniquely decodable code  $C$  will allow us to communicate messages  $x \in A$  in such a way that the average number of bits used is less than the entropy:  $L[C] \geq H(X)$ . The second theorem, on the other hand, states that for every random variable  $x \in A$  there exists a uniquely decodable code that comes very close to this bound:  $L[C] < H(X) + 1$ . Note that each uniquely decodable code is a prefix code.

**theorem:** Every prefix code  $C$  to encode messages  $x \in A$  obeys  $L[C] \geq H(X)$ , with equality if and only if  $\ell(x) = {}^2\log[1/p(x)]$  for each  $x \in A$ .

**proof:** Subtract the two sides of the inequality to be established:

$$\begin{aligned} L[C] - H(X) &= \sum_{x \in A} p(x) \left[ \ell(x) + {}^2\log p(x) \right] = \sum_{x \in A} p(x) {}^2\log \left[ \frac{p(x)}{\left(\frac{1}{2}\right)^{\ell(x)}} \right] \\ &\geq \left[ \sum_{x \in A} p(x) \right] {}^2\log \left[ \frac{\sum_{x \in A} p(x)}{\sum_{x \in A} \left(\frac{1}{2}\right)^{\ell(x)}} \right] = -{}^2\log \left[ \sum_{x \in A} \left(\frac{1}{2}\right)^{\ell(x)} \right] \geq 0 \end{aligned}$$

where the first inequality follows from the log-sum inequality (D.5), and the second inequality follows from the Kraft inequality. Finally, full equality is obtained only if both the log-sum inequality and the Kraft inequality simultaneously reduce to equalities, so:

$$\text{log-sum ineq: } \exists \lambda > 0 : p(x) = \lambda \left(\frac{1}{2}\right)^{\ell(x)}, \quad \text{Kraft ineq: } \sum_{x \in A} \left(\frac{1}{2}\right)^{\ell(x)} = 1$$

Combination gives  $\lambda = 1$ , so indeed  $p(x) = \left(\frac{1}{2}\right)^{\ell(x)}$  for each  $x \in A$ . Equivalently:  $\ell(x) = {}^2\log[1/p(x)]$  for each  $x \in A$ . This completes the proof.  $\square$

**theorem:** For every message set described by a random variable  $x \in A$  there exists a prefix code  $C$  with the property  $L[C] < H(X) + 1$ .

**proof:** The proof is based on explicit construction. It is clear from the previous theorem that efficient codes are those which approach the relation  $\ell(x) = {}^2\log[1/p(x)]$  (since each  $\ell(x)$  must be an integer the ideal situation is not always achievable). We first define for real-valued  $z$ :

$$\text{int}[z] = \text{the smallest integer } \geq z, \quad z \leq \text{int}[z] < z + 1$$

Now we choose the following codeword lengths:  $\ell(x) = \text{int} [ {}^2\log[1/p(x)] ]$ . These lengths obey the Kraft inequality, since

$$\sum_{x \in A} \left(\frac{1}{2}\right)^{\ell(x)} = \sum_{x \in A} \left(\frac{1}{2}\right)^{\text{int} [ {}^2\log(\frac{1}{p(x)}) ]} \leq \sum_{x \in A} \left(\frac{1}{2}\right)^{{}^2\log(\frac{1}{p(x)})} = \sum_{x \in A} 2^{-{}^2\log p(x)} = \sum_{x \in A} p(x) = 1$$

Here we used  $\text{int}[z] \geq z$ . The second theorem in the previous section on the Kraft inequality now guarantees that there exists a prefix code with the codelengths  $\ell(x) = \text{int} \lceil 2 \log [1/p(x)] \rceil$  (it even provides a construction). This code will then have the following code-length:

$$L[C] = \sum_{x \in A} p(x) \text{int} \lceil 2 \log [1/p(x)] \rceil < \sum_{x \in A} p(x) \lceil 2 \log [1/p(x)] + 1 \rceil = H(X) + 1$$

where we used  $\text{int}[z] < z + 1$ . This completes our proof.  $\square$

We can now summarise our present knowledge about the relation between codes and entropy in a compact way. If we define the optimal code (or most efficient code) for a given random variable as that uniquely decodable code  $C$  which has the smallest codelength  $L[C]$ , we get

**theorem:** The optimal code  $C$  to encode messages described by the random variable  $x \in A$  employs an average number of bits per message  $L[C]$  which obeys

$$H(X) \leq L[C] < H(X) + 1 \quad (4.2)$$

**proof:** This theorem follows directly from the previous two theorems (note that uniquely decodable codes are always of the prefix type).  $\square$

Since the term '+1' in the right-hand side is purely due to the fact that the ideal values  $\ell(x) = 2 \log [1/p(x)]$  cannot always be realised in practice (they have to be rounded off to the nearest integer), equation (4.2) is in itself adequate proof that the average information content of messages described by a random variable  $x \in A$  is indeed the entropy  $H(X)$ . However, no self-respecting researcher can resist the temptation to try to eliminate this rounding-off term '+1'. Neither could Shannon. The next section shows, for finite message sets  $|A| < \infty$ , how the rounding-off term can be eliminated by combining messages into sufficiently large groups, leading to Shannon's famous source coding theorem.

## 4.2.2 Killing the Final Bit

*Typical and Untypical Message Sequences.* If we try to construct an efficient code for sending groups of messages  $(x_1, \dots, x_n)$ , we need to know or at least estimate the probabilities with which the various groups can be expected to occur. Our main tool to do this will be the so-called Asymptotic Equipartitioning Property:

**theorem (AEP):** If  $\{x_1, \dots, x_n\}$  are identically distributed independent discrete random variables, each described by probability distribution  $p(x)$  with  $x \in A$ , and  $H(X)$  denotes the entropy of each individual random variable, then:

$$\lim_{n \rightarrow \infty} -\frac{1}{n} 2 \log p(x_1, \dots, x_n) = H(X) \quad (4.3)$$

**proof:** This follows directly from the independence of the variables  $\{x_1, \dots, x_n\}$ :

$$\lim_{n \rightarrow \infty} -\frac{1}{n} 2 \log p(x_1, \dots, x_n) = \lim_{n \rightarrow \infty} -\frac{1}{n} 2 \log \left[ \prod_{i=1}^n p(x_i) \right] = \lim_{n \rightarrow \infty} -\frac{1}{n} \sum_{i=1}^n 2 \log p(x_i)$$

$$= - \sum_{x \in A} p(x) \log p(x) = H(X)$$

which completes the proof.  $\square$

We can rephrase the result (4.3) as follows:

$$(\forall \epsilon > 0)(\exists n_\epsilon) : \quad \left| -\frac{1}{n} \log p(x_1, \dots, x_n) - H(X) \right| \leq \epsilon \quad (\forall n > n_\epsilon)$$

or:

$$(\forall \epsilon > 0)(\exists n_\epsilon) : \quad 2^{-n[H(X)+\epsilon]} \leq p(x_1, \dots, x_n) \leq 2^{-n[H(X)-\epsilon]} \quad (\forall n > n_\epsilon) \quad (4.4)$$

This equivalent version (4.4) of the Asymptotic Equipartitioning Property (4.3) leads in a natural way to the identification of ‘typical’ and ‘untypical’ sequences  $(x_1, \dots, x_n)$ :

**definition:** The typical set  $A_\epsilon^{(n)}$  is the set of all sequences  $(x_1, \dots, x_n) \in A^n$  with the property

$$2^{-n[H(X)+\epsilon]} \leq p(x_1, \dots, x_n) \leq 2^{-n[H(X)-\epsilon]} \quad (4.5)$$

Clearly, efficient coding relies on exploiting the differences in occurrence probabilities of the various messages to be coded. Here we will exploit the distinction between ‘typical’ sequences (with high probability) and ‘un-typical’ sequences (the rare ones). To do so we need the following properties of the typical set:

**property 1:**  $\lim_{n \rightarrow \infty} \sum_{(x_1, \dots, x_n) \in A_\epsilon^{(n)}} p(x_1, \dots, x_n) = 1$

**proof:** Define an indicator function:  $I_S[(x_1, \dots, x_n)] = 1$  if  $(x_1, \dots, x_n) \in S$ , and zero otherwise (with  $S \subseteq A^n$ ). The AEP (4.3) states that for randomly drawn sequences  $(x_1, \dots, x_n)$ :

$$(\forall \epsilon > 0)(\exists n_\epsilon) : \quad I_{A_\epsilon^{(n)}}[(x_1, \dots, x_n)] = 1 \quad (\forall n > n_\epsilon)$$

If we draw  $\ell$  such random sequences  $(x_1^{(k)}, \dots, x_n^{(k)})$  ( $k = 1 \dots \ell$ ) we get, similarly:

$$(\forall \epsilon > 0)(\exists n_\epsilon) : \quad \frac{1}{\ell} \sum_{k=1}^{\ell} I_{A_\epsilon^{(n)}}[(x_1^{(k)}, \dots, x_n^{(k)})] = 1 \quad (\forall n > n_\epsilon)$$

For  $\ell \rightarrow \infty$  this statement converts into an average over all sequences in  $A^n$ :

$$(\forall \epsilon > 0)(\exists n_\epsilon) : \quad \sum_{(x_1, \dots, x_n) \in A^n} p(x_1, \dots, x_n) I_{A_\epsilon^{(n)}}[(x_1, \dots, x_n)] = 1 \quad (\forall n > n_\epsilon)$$

$$(\forall \epsilon > 0)(\exists n_\epsilon) : \quad \sum_{(x_1, \dots, x_n) \in A_\epsilon^{(n)}} p(x_1, \dots, x_n) = 1 \quad (\forall n > n_\epsilon) \quad \square$$

**property 2:**  $|A_\epsilon^{(n)}| \leq 2^{n[H(X)+\epsilon]}$

**proof:** Just use  $A_\epsilon^{(n)} \subseteq A^n$  and the left-hand side of the inequalities of definition (4.5):

$$\begin{aligned} 2^{n[H(X)+\epsilon]} &= 2^{n[H(X)+\epsilon]} \sum_{(x_1, \dots, x_n) \in A^n} p(x_1, \dots, x_n) \geq 2^{n[H(X)+\epsilon]} \sum_{(x_1, \dots, x_n) \in A_\epsilon^{(n)}} p(x_1, \dots, x_n) \\ &\geq \sum_{(x_1, \dots, x_n) \in A_\epsilon^{(n)}} 1 = |A_\epsilon^{(n)}| \end{aligned} \quad \square$$

*Shannon's Source Coding Theorem.* We can now explicitly construct a code to deal with large messages sequences  $(x_1, \dots, x_n)$  in an efficient way. It turns out that we need only worry about whether a given sequence is 'typical' or not; all other probability variations turn out to be irrelevant for killing the rounding off bit in our previous bound (4.2). We will, for simplicity, restrict our proof to finite message sets  $|A| < \infty$  (the theorem can also be proven for  $|A| = \infty$ ).

Assume  $\{x_1, \dots, x_n\}$  to be identically distributed independent discrete random variables, each described by probability distribution  $p(x)$  with  $x \in A$  and  $|A| < \infty$ .  $H(X)$  denotes the entropy of each individual random variable. Then:

**theorem:** For each  $\epsilon > 0$  there exists a uniquely decodable code  $C_\epsilon : A^n \rightarrow \bigcup_{L \geq 1} \{0, 1\}^L$  with the property

$$\lim_{n \rightarrow \infty} \frac{1}{n} L[C_\epsilon] \leq H(X) + \epsilon \tag{4.6}$$

Since  $C_\epsilon$  codes for  $n$  messages,  $\frac{1}{n} L[C_\epsilon]$  is the average number of bits used per message.

**proof:** We will simply construct a code with the desired properties. For a given  $n$  and a given  $\epsilon$  we divide the set  $A^n$  of all sequences  $(x_1, \dots, x_n)$  into the typical set  $A_\epsilon^{(n)}$  (4.5) and its complement  $\overline{A}_\epsilon^{(n)} = \{(x_1, \dots, x_n) \in A^n \mid (x_1, \dots, x_n) \notin A_\epsilon^{(n)}\}$ . For simplicity we will label the sequences in these two sets in the following way:

sequences in $A_\epsilon^{(n)}$ : $(x_1, \dots, x_n)_1$ $(x_1, \dots, x_n)_2$ $(x_1, \dots, x_n)_3$ ...	sequences in $\overline{A}_\epsilon^{(n)}$ : $(y_1, \dots, y_n)_1$ $(y_1, \dots, y_n)_2$ $(y_1, \dots, y_n)_3$ ...
---------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------

Within each of these two sets we now code the sequences in an enumerative way (i.e. the code-word of a sequence is the binary representation of its rank in the set). As a prefix we will attach an extra bit to indicate of which set the sequence is a member ('1' if in  $A_\epsilon^{(n)}$ , '0' if not), giving

sequences in $A_\epsilon^{(n)}$ : codeword : $(x_1, \dots, x_n)_1$ 10000 ... $(x_1, \dots, x_n)_2$ 11000 ... $(x_1, \dots, x_n)_3$ 10100 ... $(x_1, \dots, x_n)_4$ 11100 ... ...	sequences in $\overline{A}_\epsilon^{(n)}$ : codeword : $(y_1, \dots, y_n)_1$ 00000 ... $(y_1, \dots, y_n)_2$ 01000 ... $(y_1, \dots, y_n)_3$ 00100 ... $(y_1, \dots, y_n)_4$ 01100 ... ...
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

This code is of the prefix type. Once the first bit has revealed whether or not the sequence is typical the rest of the codeword is of known length, and therefore the code is uniquely decodable. Enumerative coding of the elements in a given set  $A$  requires  $\text{int} \lceil 2 \log |A| \rceil \leq 2 \log |A| + 1$  bits per codeword (note: if  $2 \log |A|$  is not an integer one has to round off). Therefore we know the lengths of the codewords:

$$\ell(x_1, \dots, x_n) \leq \begin{cases} 2 \log |A_\epsilon^{(n)}| + 2 & \text{if } (x_1, \dots, x_n) \in A_\epsilon^{(n)} \\ 2 \log |\overline{A}_\epsilon^{(n)}| + 2 & \text{if } (x_1, \dots, x_n) \in \overline{A}_\epsilon^{(n)} \end{cases}$$

(where an extra '+1' has been added due to the bit in the code indicating whether the sequence is typical). We now derive upper bounds for the codeword lengths. For  $(x_1, \dots, x_n) \in A_\epsilon^{(n)}$  we use property 2 of the typical set; for  $(x_1, \dots, x_n) \in \overline{A}_\epsilon^{(n)}$  we just use the crude bound  $|\overline{A}_\epsilon^{(n)}| \leq |A^n| = |A|^n$ :

$$\ell(x_1, \dots, x_n) \leq \begin{cases} n[H(X) + \epsilon] + 2 & \text{if } (x_1, \dots, x_n) \in A_\epsilon^{(n)} \\ n^2 \log |A| + 2 & \text{if } (x_1, \dots, x_n) \in \overline{A}_\epsilon^{(n)} \end{cases}$$

For the codelength  $L[C_\epsilon]$  we thus find:

$$\begin{aligned} L[C_\epsilon] &= \sum_{(x_1, \dots, x_n) \in A_\epsilon^{(n)}} p(x_1, \dots, x_n) \ell(x_1, \dots, x_n) + \sum_{(x_1, \dots, x_n) \in \overline{A}_\epsilon^{(n)}} p(x_1, \dots, x_n) \ell(x_1, \dots, x_n) \\ &\leq 2 + n[H(X) + \epsilon] \sum_{(x_1, \dots, x_n) \in A_\epsilon^{(n)}} p(x_1, \dots, x_n) + n^2 \log |A| \left[ 1 - \sum_{(x_1, \dots, x_n) \in A_\epsilon^{(n)}} p(x_1, \dots, x_n) \right] \end{aligned}$$

We now consider the average number of bits per message  $\frac{1}{n}L[C_\epsilon]$  (note that  $C_\epsilon$  codes sequences of  $n$  messages each) and take the limit  $n \rightarrow \infty$ :

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{1}{n}L[C_\epsilon] &\leq [H(X) + \epsilon] \lim_{n \rightarrow \infty} \sum_{(x_1, \dots, x_n) \in A_\epsilon^{(n)}} p(x_1, \dots, x_n) \\ &\quad + 2 \log |A| \left[ 1 - \lim_{n \rightarrow \infty} \sum_{(x_1, \dots, x_n) \in A_\epsilon^{(n)}} p(x_1, \dots, x_n) \right] \end{aligned}$$

Finally we use property 1 of the typical set:

$$\lim_{n \rightarrow \infty} \frac{1}{n}L[C_\epsilon] \leq H(X) + \epsilon$$

which completes the proof.  $\square$

By combining our messages  $x \in A$  into larger and larger sequences, followed by coding these  $n$ -message sequences in an efficient way, we can effectively distribute the rounding off bit '+1' of individual codewords over the  $n$  messages which the codeword describes, and thus obtain an average number of bits per message as close as we like to the entropy  $H(X)$ .



## Chapter 5

# Applications to Neural Networks

The main contribution of information theory to the field of information processing in natural or artificial neural networks is that it provides exact performance measures. This allows us to compare the performance of different models/algorithms in a rigorous way, and furthermore allows for the development of learning rules for systems with a given architecture, based on the maximisation of these information-theoretic performance measures, which are no longer ad-hoc<sup>1</sup>, and which also apply in unsupervised scenarios. The examples we will discuss here are the Boltzmann Machine Learning rule for recurrent layered neural networks and various types of learning rules (e.g. Maximum Information Preservation) for layered networks.

### 5.1 Supervised Learning: Boltzmann Machines

One of the earliest and most elegant applications of information theory in neural networks is the so-called Boltzmann Machine Learning rule. It gives a recipe for training symmetrically recurrent layered neural networks to perform a given input-output operation.

#### 5.1.1 Definitions and General Properties

*Network Architecture.* We imagine a network composed of  $N + K + M$  binary neurons  $s_i \in \{-1, 1\}$ , which has been partitioned into an input layer (of  $N$  neurons), a so-called hidden layer (of  $K$  neurons), and an output layer (of  $M$  neurons). See figure 5.1. To simplify subsequent notation we denote the states of the neurons in these three layers by the vectors  $\mathbf{x} \in \{-1, 1\}^N$ ,  $\boldsymbol{\sigma} \in \{-1, 1\}^K$  and  $\mathbf{y} \in \{-1, 1\}^M$ , respectively, and the combined state of all three layers by

$$\mathbf{s} = (\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \in \{-1, 1\}^{N+K+M} \quad (5.1)$$

The connectivity of the network is described by a symmetric interaction matrix  $\{J_{ij}\}$ , with  $i, j \in \{1, \dots, M+N+K\}$ . It is assumed to have the following properties:

$$J_{ii} = 0 \text{ for all } i, \quad J_{ij} = J_{ji} \text{ for all } (i, j) \quad (5.2)$$

An absent interaction corresponds to  $J_{ij} = 0$ . Due to the symmetry requirement  $J_{ij} = J_{ji}$

---

<sup>1</sup>For instance, the popular error-backpropagation rule for learning in layered networks is based on the ad-hoc choice to minimise the cumulative squared error  $\sum_{\text{inputs}} [S_{\text{network}} - S_{\text{task}}]^2$ . Note that, instead of choosing the square, one can choose any monotonic function of  $|S_{\text{network}} - S_{\text{task}}|$ .



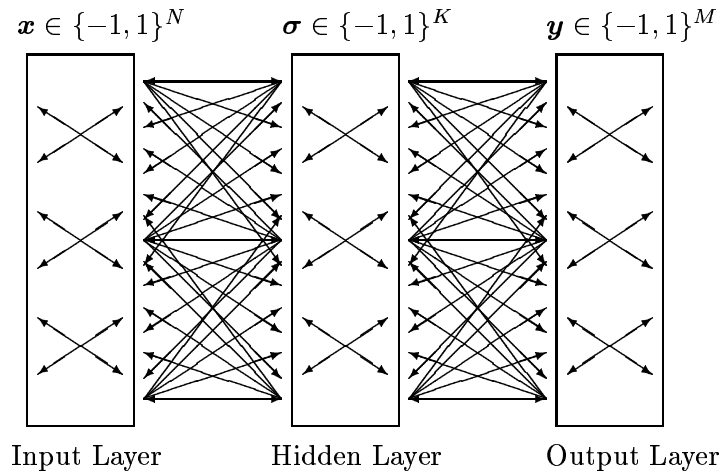


Figure 5.1: Architecture of the Boltzmann Machine, with arrows indicating potential synaptic interactions. All interactions present are required to be symmetric,  $J_{ij} = J_{ji}$ , and self-interactions  $J_{ii}$  are absent (direct synaptic interactions between input- and output layer are also allowed, but have not been drawn in order not to mess up the picture).

the network is recurrent, potentially involving recurrent interactions within all three layers as well as between all three layers, but it need not be fully recurrent (the interaction matrix could be sparse). We will assume as a minimum requirement the presence of non-zero interactions between the input layer and the hidden layer, and between the hidden layer and the output layer, in order to guarantee that input signals can at least reach the output side.

*Neuronal Dynamics.* The neuron states evolve in time in a stochastic manner, however, we have the freedom to allow only a subset  $S \subseteq \{1, \dots, N+K+M\}$  of neurons to actually change their states (the others would remain stationary). At each time step we perform:

1. Choose at random a neuron  $i$  to be updated, from the set  $S \subseteq \{1, \dots, N+K+M\}$
2. Calculate its local field (or postsynaptic potential):  $h_i(\mathbf{s}) = \sum_j J_{ij}s_j + \theta_i$
3. Change its state with probability  $\text{Prob}[s_i \rightarrow -s_i] = \frac{1}{2}[1 - \tanh(\beta s_i h_i(\mathbf{s}))]$

Here the parameters  $\theta_i$  determine the firing thresholds of the neurons, and the parameter  $\beta$  controls the degree of randomness in the dynamics. For  $\beta = 0$  the dynamics just assigns random values to the states of the updated neurons. For  $\beta \rightarrow \infty$ , on the other hand, the candidate neurons align their states strictly to the sign of the local fields:  $s_i \rightarrow 1$  if  $h_i(\mathbf{s}) > 0$ ,  $s_i \rightarrow -1$  if  $h_i(\mathbf{s}) < 0$ .

Since the dynamics is stochastic we can only speak about the probability  $p_t(\mathbf{s})$  to find the system in a certain state  $\mathbf{s}$  at a certain time  $t$ . The nice aspect of the chosen dynamics (in combination with the constraints on the synaptic interaction matrix) is that one can write down an exact expression for the stationary state of the system and prove that it is unique.

**theorem:** The unique stationary probability distribution of the neuronal dynamics is

$$p_\infty(\mathbf{s}) = \frac{1}{Z} e^{-\beta H(\mathbf{s})}, \quad H(\mathbf{s}) = -\frac{1}{2} \sum_{ij} s_i J_{ij} s_j - \sum_i s_i \theta_i \quad (5.3)$$

in which  $Z$  is a normalization constant which depends on the choice made for the set  $S$  of neurons that are allowed to change state.

**proof:** Here we will only show that the distribution (5.3) (the so-called Boltzmann distribution, hence the name of these systems) obeys ‘detailed balance’:

$$\forall i \in S: \quad p_\infty(s_1, \dots, s_i, \dots) \text{Prob}[s_i \rightarrow -s_i] = p_\infty(s_1, \dots, -s_i, \dots) \text{Prob}[-s_i \rightarrow s_i] \quad (5.4)$$

This property states that for each neuron  $i \in S$  the number of transitions  $s_i \rightarrow -s_i$  will equal the number of transitions where  $-s_i \rightarrow s_i$ . The proof that detailed balance indeed implies that  $p_\infty(\mathbf{s})$  is a stationary state, as well as the issues of uniqueness and convergence are beyond the scope of this course (see e.g. the Statistical Mechanics of Neural Networks lecture notes). To prove (5.4) we use the identities  $1 - \tanh[z] = e^{-z} \cosh^{-1}[z]$  and  $\cosh[-z] = \cosh[z]$  as well as the fact that  $h_i(\mathbf{s})$  does not depend on  $s_i$  (due to  $J_{ii} = 0$ ):

$$\begin{aligned} & p_\infty(s_1, \dots, s_i, \dots) \text{Prob}[s_i \rightarrow -s_i] - p_\infty(s_1, \dots, -s_i, \dots) \text{Prob}[-s_i \rightarrow s_i] \\ &= \frac{e^{-\beta H(\mathbf{s}) - \beta s_i h_i(\mathbf{s})}}{2Z \cosh[\beta h_i(\mathbf{s})]} \left[ 1 - e^{-\beta \Delta_i H(\mathbf{s}) + 2\beta s_i h_i(\mathbf{s})} \right] \end{aligned} \quad (5.5)$$

in which  $\Delta_i H(\mathbf{s}) = H(s_1, \dots, -s_i, \dots) - H(s_1, \dots, s_i, \dots)$ . Note that the state change  $s_i \rightarrow -s_i$  can be captured by replacing each  $s_k$  by  $(1 - 2\delta_{ik})s_k$  (where  $\delta_{ik} = 1$  if  $i = k$  and  $\delta_{ik} = 0$  otherwise), so with (5.3) we get

$$\begin{aligned} \Delta_i H(\mathbf{s}) &= -\frac{1}{2} \sum_{kl} [1 - 2\delta_{ik}] s_k J_{kl} [1 - 2\delta_{il}] s_l - \sum_k [1 - 2\delta_{ik}] s_k \theta_k + \frac{1}{2} \sum_{kl} s_k J_{kl} s_l + \sum_k s_k \theta_k \\ &= -\frac{1}{2} \sum_{k \neq l} J_{kl} s_k s_l \{ [1 - 2\delta_{ik}] [1 - 2\delta_{il}] - 1 \} - \sum_k \theta_k s_k \{ 1 - 2\delta_{ik} - 1 \} \\ &= \sum_{k \neq l} J_{kl} s_k s_l \{ \delta_{ik} + \delta_{il} \} + 2\theta_i s_i = 2s_i h_i(\mathbf{s}) \end{aligned}$$

where we have used the symmetry  $J_{kl} = J_{lk}$ . Insertion into our previous expression (5.5) completes the proof.  $\square$

We can now give exact expressions for equilibrium state probabilities under various different operation conditions. The differences between the various choices to be made for the set  $S$  of ‘free’ neurons only affect the normalization factor  $Z$  in (5.3), for instance:

$$(A) \text{ all neurons free: } \quad p_\infty^A(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) = \frac{e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})}}{\sum_{\mathbf{x}', \boldsymbol{\sigma}', \mathbf{y}'} e^{-\beta H(\mathbf{x}', \boldsymbol{\sigma}', \mathbf{y}')}} \quad (5.6)$$

(B) hidden & output neurons free :

$$p_{\infty}^B(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) = p_{\infty}(\boldsymbol{\sigma}, \mathbf{y}|\mathbf{x})p(\mathbf{x}), \quad p_{\infty}(\boldsymbol{\sigma}, \mathbf{y}|\mathbf{x}) = \frac{e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})}}{\sum_{\boldsymbol{\sigma}', \mathbf{y}'} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}', \mathbf{y}')}} \quad (5.7)$$

(C) hidden neurons free :

$$p_{\infty}^C(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) = p_{\infty}(\boldsymbol{\sigma}|\mathbf{x}, \mathbf{y})p(\mathbf{x}, \mathbf{y}), \quad p_{\infty}(\boldsymbol{\sigma}|\mathbf{x}, \mathbf{y}) = \frac{e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})}}{\sum_{\boldsymbol{\sigma}'} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}', \mathbf{y})}} \quad (5.8)$$

### 5.1.2 Derivation of the Learning Rule

*Definition of the Target.* We are now in a position to define the aim of the learning process. The system is to learn a given task which is defined in terms of a prescribed target joint input-output probability distribution  $q(\mathbf{x}, \mathbf{y})$ . It has learnt the task when  $p_{\infty}(\mathbf{x}, \mathbf{y})$  (the equilibrium input-output probability distribution of the network) equals  $q(\mathbf{x}, \mathbf{y})$ . An information-theoretic measure is used to quantify the ‘distance’ between  $q(\mathbf{x}, \mathbf{y})$  and  $p_{\infty}(\mathbf{x}, \mathbf{y})$ : the relative entropy, or Kullback-Leibler distance (3.9):

$$D(q||p_{\infty}) = \sum_{\mathbf{x}\mathbf{y}} q(\mathbf{x}, \mathbf{y})^2 \log \left[ \frac{q(\mathbf{x}, \mathbf{y})}{p_{\infty}(\mathbf{x}, \mathbf{y})} \right] \quad (5.9)$$

$D(q||p_{\infty})$  is minimal (and identical zero) only when  $p_{\infty}(\mathbf{x}, \mathbf{y})$  and  $q(\mathbf{x}, \mathbf{y})$  are identical. We aim to minimise  $D(q||p_{\infty})$  by changing the network parameters (the synaptic interactions  $J_{ij}$  and the thresholds  $\theta_i$ ) via a ‘gradient descent’ rule:

$$\Delta J_{ij} = -\epsilon \frac{\partial D(q||p_{\infty})}{\partial J_{ij}} \quad \Delta \theta_i = -\epsilon \frac{\partial D(q||p_{\infty})}{\partial \theta_i}, \quad 0 < \epsilon \ll 1 \quad (5.10)$$

which guarantees

$$\begin{aligned} \Delta D(q||p_{\infty}) &= \sum_{ij} \frac{\partial D(q||p_{\infty})}{\partial J_{ij}} \Delta J_{ij} + \sum_i \frac{\partial D(q||p_{\infty})}{\partial \theta_i} \Delta \theta_i + \mathcal{O}(\epsilon^2) \\ &= -\epsilon \left\{ \sum_{ij} \left[ \frac{\partial D(q||p_{\infty})}{\partial J_{ij}} \right]^2 + \sum_i \left[ \frac{\partial D(q||p_{\infty})}{\partial \theta_i} \right]^2 \right\} + \mathcal{O}(\epsilon^2) \end{aligned}$$

For sufficiently small modification sizes  $\epsilon$  the ‘distance’  $D(q||p_{\infty})$  decreases monotonically until a stationary state is reached (which could, but need not be  $D(q||p_{\infty}) = 0$ ). For any parameter  $\lambda$  in our system (whether synaptic interaction or threshold) we get:

$$\frac{\partial}{\partial \lambda} D(q||p_{\infty}) = -\frac{1}{\log 2} \sum_{\mathbf{x}\mathbf{y}} q(\mathbf{x}, \mathbf{y}) \frac{\partial}{\partial \lambda} \log p_{\infty}(\mathbf{x}, \mathbf{y}) \quad (5.11)$$

The details of the subsequent calculation will now depend on the network operation conditions for which we want to minimise  $D(q||p_{\infty})$  (i.e. the choice made for the set  $S$ ), since these determine  $p_{\infty}(\mathbf{x}, \mathbf{y})$ . We will first analyse the case where all neurons evolve freely, secondly we will analyse the case where the input neurons are always prescribed and only the hidden- and output neurons are free to evolve. Both cases give a similar result.

*Operation With All Neurons Freely Evolving.* Here the relevant expression with which to calculate  $p_\infty(\mathbf{x}, \mathbf{y})$  is equation (5.6), which gives

$$p_\infty(\mathbf{x}, \mathbf{y}) = \sum_{\boldsymbol{\sigma}} p_\infty(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) = \frac{\sum_{\boldsymbol{\sigma}} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})}}{\sum_{\mathbf{x}' \boldsymbol{\sigma}' \mathbf{y}'} e^{-\beta H(\mathbf{x}', \boldsymbol{\sigma}', \mathbf{y}')}}$$

Derivatives of the type (5.11) are found to be

$$\begin{aligned} \frac{\partial}{\partial \lambda} D(q||p_\infty) &= -\frac{1}{\log 2} \sum_{\mathbf{x} \mathbf{y}} q(\mathbf{x}, \mathbf{y}) \frac{\partial}{\partial \lambda} \left[ \log \sum_{\boldsymbol{\sigma}} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})} - \log \sum_{\mathbf{x}' \boldsymbol{\sigma}' \mathbf{y}'} e^{-\beta H(\mathbf{x}', \boldsymbol{\sigma}', \mathbf{y}')} \right] \\ &= \frac{\beta}{\log 2} \sum_{\mathbf{x} \mathbf{y}} q(\mathbf{x}, \mathbf{y}) \left[ \frac{\sum_{\boldsymbol{\sigma}} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})} \partial H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) / \partial \lambda}{\sum_{\boldsymbol{\sigma}} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})}} - \frac{\sum_{\mathbf{x}' \boldsymbol{\sigma}' \mathbf{y}'} e^{-\beta H(\mathbf{x}', \boldsymbol{\sigma}', \mathbf{y}')} \partial H(\mathbf{x}', \boldsymbol{\sigma}', \mathbf{y}') / \partial \lambda}{\sum_{\mathbf{x}' \boldsymbol{\sigma}' \mathbf{y}'} e^{-\beta H(\mathbf{x}', \boldsymbol{\sigma}', \mathbf{y}')}} \right] \\ &= \frac{\beta}{\log 2} \left[ \sum_{\mathbf{x} \boldsymbol{\sigma} \mathbf{y}} p_\infty^C(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \frac{\partial}{\partial \lambda} H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) - \sum_{\mathbf{x} \boldsymbol{\sigma} \mathbf{y}} p_\infty^A(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \frac{\partial}{\partial \lambda} H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \right] \\ &= \frac{\beta}{\log 2} \left[ \langle \frac{\partial}{\partial \lambda} H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \rangle_+ - \langle \frac{\partial}{\partial \lambda} H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \rangle_- \right] \end{aligned} \quad (5.12)$$

Here averages indicated with '+' are those where the system is only allowed to change the states of hidden neurons (the case described by (5.8)). The states of the input and output neurons are enforced upon the system, with statistics given by the task distribution  $q(\mathbf{x}, \mathbf{y})$ :

$$\langle f(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \rangle_+ = \sum_{\mathbf{x} \boldsymbol{\sigma} \mathbf{y}} f(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) p_\infty^C(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) = \sum_{\mathbf{x} \boldsymbol{\sigma} \mathbf{y}} f(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) p_\infty(\boldsymbol{\sigma} | \mathbf{x}, \mathbf{y}) q(\mathbf{x}, \mathbf{y}) \quad (5.13)$$

Averages indicated with '-' are those where the system evolves freely (as described by (5.6)):

$$\langle f(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \rangle_- = \sum_{\mathbf{x} \boldsymbol{\sigma} \mathbf{y}} f(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) p_\infty^A(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \quad (5.14)$$

What remains is to calculate the derivatives of  $H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})$ , and use (5.12) to evaluate (5.10):

$$\frac{\partial}{\partial J_{ij}} H(\mathbf{s}) = -s_i s_j \quad \frac{\partial}{\partial \theta_i} H(\mathbf{s}) = -s_i \quad (5.15)$$

$$\begin{aligned} \frac{\partial}{\partial J_{ij}} D(q||p_\infty) &= -\frac{\beta}{\log 2} [\langle s_i s_j \rangle_+ - \langle s_i s_j \rangle_-] & \frac{\partial}{\partial \theta_i} D(q||p_\infty) &= -\frac{\beta}{\log 2} [\langle s_i \rangle_+ - \langle s_i \rangle_-] \\ \Delta J_{ij} &= \frac{\epsilon \beta}{\log 2} [\langle s_i s_j \rangle_+ - \langle s_i s_j \rangle_-] & \Delta \theta_i &= \frac{\epsilon \beta}{\log 2} [\langle s_i \rangle_+ - \langle s_i \rangle_-] \end{aligned} \quad (5.16)$$

Each modification step thus involves:

(i) operate the neuronal dynamics with input and output neuron states  $(\mathbf{x}, \mathbf{y})$  fixed until equilibrium is reached, and measure the averages  $\langle s_i s_j \rangle_+$  and  $\langle s_i \rangle_+$ ; repeat this for many combinations of  $(\mathbf{x}, \mathbf{y})$  generated according to the desired joint distribution  $q(\mathbf{x}, \mathbf{y})$ .

(ii) operate the neuronal dynamics with all neurons evolving freely until equilibrium is reached, and measure the averages  $\langle s_i s_j \rangle_-$  and  $\langle s_i \rangle_-$ .

(iii) Insert the results of (i) and (ii) into (5.16) and execute the rule (5.16).

*Operation With Hidden and Output Neurons Freely Evolving.* Now we consider the situation where the states of the input neurons are prescribed, with statistics given by  $p(\mathbf{x}) = \sum_{\mathbf{y}} q(\mathbf{x}, \mathbf{y})$ . Here the relevant expression with which to calculate  $p_{\infty}(\mathbf{x}, \mathbf{y})$  is equation (5.7), which gives

$$p_{\infty}(\mathbf{x}, \mathbf{y}) = \sum_{\boldsymbol{\sigma}} p_{\infty}(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) = \sum_{\boldsymbol{\sigma}} p_{\infty}(\boldsymbol{\sigma}, \mathbf{y}|\mathbf{x})p(\mathbf{x}) = p(\mathbf{x}) \frac{\sum_{\boldsymbol{\sigma}} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})}}{\sum_{\boldsymbol{\sigma}', \mathbf{y}'} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}', \mathbf{y}')}}$$

Derivatives of the type (5.11) are now found to be

$$\begin{aligned} \frac{\partial}{\partial \lambda} D(q||p_{\infty}) &= -\frac{1}{\log 2} \sum_{\mathbf{x}, \mathbf{y}} q(\mathbf{x}, \mathbf{y}) \frac{\partial}{\partial \lambda} \left[ \log \sum_{\boldsymbol{\sigma}} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})} - \log \sum_{\boldsymbol{\sigma}', \mathbf{y}'} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}', \mathbf{y}')} \right] \\ &= \frac{\beta}{\log 2} \sum_{\mathbf{x}, \mathbf{y}} q(\mathbf{x}, \mathbf{y}) \left[ \frac{\sum_{\boldsymbol{\sigma}} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})} \partial H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) / \partial \lambda}{\sum_{\boldsymbol{\sigma}} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})}} - \frac{\sum_{\boldsymbol{\sigma}', \mathbf{y}'} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}', \mathbf{y}')} \partial H(\mathbf{x}, \boldsymbol{\sigma}', \mathbf{y}') / \partial \lambda}{\sum_{\boldsymbol{\sigma}', \mathbf{y}'} e^{-\beta H(\mathbf{x}, \boldsymbol{\sigma}', \mathbf{y}')}} \right] \\ &= \frac{\beta}{\log 2} \left[ \sum_{\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}} p_{\infty}^{\text{C}}(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \frac{\partial}{\partial \lambda} H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) - \sum_{\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}} p_{\infty}^{\text{B}}(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \frac{\partial}{\partial \lambda} H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \right] \\ &= \frac{\beta}{\log 2} \left[ \langle \frac{\partial}{\partial \lambda} H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \rangle_+ - \langle \frac{\partial}{\partial \lambda} H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \rangle_- \right] \end{aligned} \quad (5.17)$$

Averages indicated with '+' are again those where the system is only allowed to change the states of hidden neurons (the case described by (5.8)). The states of the input and output neurons are enforced upon the system, with statistics given by the task distribution  $q(\mathbf{x}, \mathbf{y})$ :

$$\langle f(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \rangle_+ = \sum_{\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}} f(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) p_{\infty}^{\text{C}}(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) = \sum_{\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}} f(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) p_{\infty}(\boldsymbol{\sigma}|\mathbf{x}, \mathbf{y}) q(\mathbf{x}, \mathbf{y}) \quad (5.18)$$

Averages indicated with '-', however, now describe a system where only hidden and output neurons evolve freely (as described by (5.7)), with the states of the input neurons as before enforced upon the system, with probabilities  $p(\mathbf{x}) = \sum_{\mathbf{y}} q(\mathbf{x}, \mathbf{y})$ :

$$\langle f(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) \rangle_- = \sum_{\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}} f(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) p_{\infty}^{\text{B}}(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) = \sum_{\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}} f(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y}) p_{\infty}(\boldsymbol{\sigma}, \mathbf{y}|\mathbf{x}) p(\mathbf{x}) \quad (5.19)$$

Note that the derivatives of  $H(\mathbf{x}, \boldsymbol{\sigma}, \mathbf{y})$  are still given by (5.15). Using (5.17), which differs from (5.12) only in the definition of the average (5.19), our learning rule (5.10) indeed acquires the same form as the previous one:

$$\Delta J_{ij} = \frac{\epsilon \beta}{\log 2} [\langle s_i s_j \rangle_+ - \langle s_i s_j \rangle_-] \quad \Delta \theta_i = \frac{\epsilon \beta}{\log 2} [\langle s_i \rangle_+ - \langle s_i \rangle_-] \quad (5.20)$$

Each modification now involves:

(i) operate the neuronal dynamics with input and output neuron states  $(\mathbf{x}, \mathbf{y})$  fixed until equilibrium is reached, and measure the averages  $\langle s_i s_j \rangle_+$  and  $\langle s_i \rangle_+$ ; repeat this for many combinations of  $(\mathbf{x}, \mathbf{y})$  generated according to the desired joint distribution  $q(\mathbf{x}, \mathbf{y})$ .

(ii) operate the neuronal dynamics with all hidden and output neurons evolving freely until

equilibrium is reached, and measure the averages  $\langle s_i s_j \rangle_-$  and  $\langle s_i \rangle_-$ ; repeat this for many input configurations  $\mathbf{x}$  generated according to the desired distribution  $p(\mathbf{x}) = \sum_{\mathbf{y}} q(\mathbf{x}, \mathbf{y})$ .

(iii) Insert the results of (i) and (ii) into (5.16) and execute the rule (5.16).

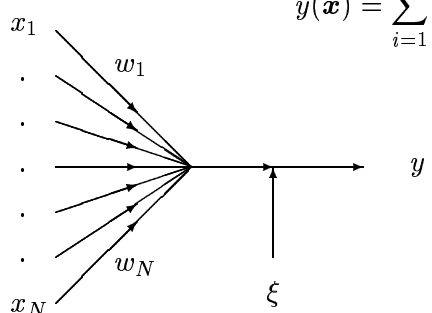
The advantages of the Boltzmann machine learning rule are that it aims to optimise a well-defined and sensible performance measure, and that it can deal in a clean way with probabilistic data (which is the more natural situation in real-world information processing). The disadvantage is that it is usually very slow; each infinitesimal parameter modification requires equilibration of a recurrent stochastic system, which can take a lot of CPU time.

## 5.2 Maximum Information Preservation

As a second class of applications of information theory we will discuss unsupervised learning in layered neural systems. For simplicity we will consider linear neurons only. The techniques and strategies we will discuss can also be used and followed in the more general case of arbitrary (not necessarily linear) neuronal transfer functions, which we will show in a subsequent section. Our neurons will have to adapt their synaptic interactions according to unsupervised rules, i.e. there is no task signal available which can be used as a reference or target.

### 5.2.1 Linear Neurons with Gaussian Output Noise

Imagine a single linear neuron  $y : \mathfrak{R}^N \rightarrow \mathfrak{R}$ , the output  $y(\mathbf{x})$  of which is corrupted by a Gaussian noise source  $\xi$  in the following way:

$$y(\mathbf{x}) = \sum_{i=1}^N w_i x_i + \xi \quad p(\xi) = \frac{e^{-\frac{1}{2}\xi^2/\sigma^2}}{\sigma\sqrt{2\pi}} \quad (5.21)$$


with synaptic weights  $\{w_i\}$  (the system's adjustable parameters). The strength of the noise is measured by  $\sigma^2 = \langle \xi^2 \rangle$ . We assume the input signals to obey  $\langle x_i \rangle = 0$  and to be statistically independent of the noise source. We also assume the uncorrupted signal  $z(\mathbf{x}) = \sum_{i=1}^N w_i x_i$  to have a Gaussian probability distribution; this is true for any  $N$  if the inputs  $x_i$  are themselves Gaussian random variables, and it is true for  $N \rightarrow \infty$  if the inputs  $x_i$  are independent (under some weak conditions on the parameters  $\{w_i\}$ ). As a result we can be sure that the pair  $(y, z)$  is described by a Gaussian joint probability distribution.

Since we have no reference (teacher) signal to compare the output of our neuron with, the application of learning rules for updating the parameters  $\{w_i\}$  which are based on error reduction is ruled out (there is no such thing as an 'error'). This situation is quite common in the primary stages of (biological or artificial) sensory information processing, where one

does not yet know how the incoming information should be used, but one still wants to extract the maximum amount of information from the data. Alternatively one can view the problem as that of data compression: try to put as much information as is possible about the input vectors  $\mathbf{x} \in \mathfrak{R}^N$  in a single variable  $y(\mathbf{x})$ . We here apply the principle of Maximum Information Preservation (InfoMax): we try to maximise the differential mutual information (3.11) between the corrupted signal  $y(\mathbf{x})$  and the uncorrupted signal  $z(\mathbf{x})$ , i.e. we maximise the amount of information that  $y(\mathbf{x})$  reveals about  $z(\mathbf{x})$ . Since the relevant variables have a Gaussian joint probability distribution, and since  $\langle z(\mathbf{x}) \rangle = \sum_{i=1}^N w_i \langle x_i \rangle = 0$  and  $\langle y(\mathbf{x}) \rangle = \langle z(\mathbf{x}) \rangle + \langle \xi \rangle = 0$ , we can express the differential mutual information  $\tilde{I}(Y, Z)$  in terms of the second order moments (see (3.20)):

$$\begin{aligned} \langle z^2 \rangle &= \sum_{ij=1}^N w_i w_j \langle x_i x_j \rangle \\ \langle yz \rangle &= \langle [\sum_i w_i x_i][\sum_j w_j x_j + \xi] \rangle = \sum_{ij=1}^N w_i w_j \langle x_i x_j \rangle + \sum_{i=1}^N w_i \langle x_i \xi \rangle = \sum_{ij=1}^N w_i w_j \langle x_i x_j \rangle \\ \langle y^2 \rangle &= \langle [\sum_i w_i x_i + \xi][\sum_j w_j x_j + \xi] \rangle = \sum_{ij=1}^N w_i w_j \langle x_i x_j \rangle + \langle \xi^2 \rangle = \sum_{ij=1}^N w_i w_j \langle x_i x_j \rangle + \sigma^2 \end{aligned}$$

Substitution into  $\tilde{I}(Y, Z)$  (3.20) gives:

$$\begin{aligned} \tilde{I}(Y, Z) &= -\frac{1}{2} {}^2\log \left[ 1 - \frac{\langle yz \rangle^2}{\langle y^2 \rangle \langle z^2 \rangle} \right] = -\frac{1}{2} {}^2\log \left[ 1 - \frac{\sum_{ij=1}^N w_i w_j \langle x_i x_j \rangle}{\sum_{ij=1}^N w_i w_j \langle x_i x_j \rangle + \sigma^2} \right] \\ &= -\frac{1}{2} {}^2\log \left[ \frac{\sigma^2}{\sum_{ij=1}^N w_i w_j \langle x_i x_j \rangle + \sigma^2} \right] = \frac{1}{2} {}^2\log \left[ 1 + \frac{\sum_{ij=1}^N w_i w_j \langle x_i x_j \rangle}{\sigma^2} \right] \end{aligned} \quad (5.22)$$

We conclude that we can make this expression (5.22) for the differential information as large as we like, simply by boosting all synapses according to  $w_i \rightarrow \lambda w_i$ , which gives

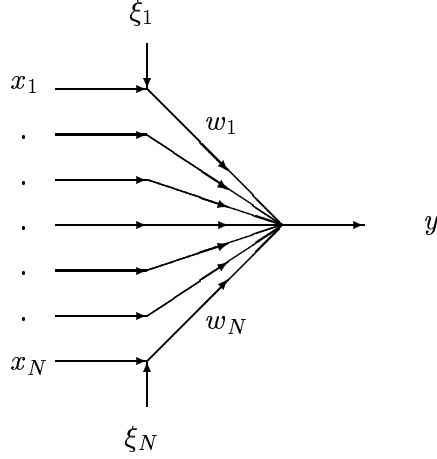
$$\tilde{I}(Y, Z) \rightarrow \frac{1}{2} {}^2\log \left[ 1 + \lambda^2 \sigma^{-2} \sum_{ij=1}^N w_i w_j \langle x_i x_j \rangle \right]$$

(note that  $\sum_{ij} w_i w_j \langle x_i x_j \rangle = \langle [\sum_i w_i x_i]^2 \rangle \geq 0$ ). The differential mutual information has no upper bound. This makes sense: since the output is simply the sum of a signal term (with strength partly controlled by the synaptic weights) and a noise term (of constant strength), the signal/noise ratio can be improved arbitrarily by increasing the strength of the signal via a boost of the weights.

### 5.2.2 Linear Neurons with Gaussian Input Noise

Imagine next a single linear neuron  $y : \mathfrak{R}^N \rightarrow \mathfrak{R}$  of which not the output, but rather the inputs  $\{x_i\}$  are corrupted with mutually independent Gaussian noise sources  $\{\xi_i\}$  in the following way:

$$y(\mathbf{x}) = \sum_{i=1}^N w_i (x_i + \xi_i) \quad p(\xi_i) = \frac{e^{-\frac{1}{2}\xi_i^2/\sigma_i^2}}{\sigma_i \sqrt{2\pi}} \quad (5.23)$$



with synaptic weights  $\{w_i\}$ . The strengths of the  $N$  noise sources  $\xi_i$  are measured by  $\sigma_i^2 = \langle \xi_i^2 \rangle$ . We again assume the input signals to obey  $\langle x_i \rangle = 0$  and to be statistically independent of the noise sources, and that the uncorrupted signal  $z(\mathbf{x}) = \sum_{i=1}^N w_i x_i$  has a Gaussian probability distribution. We can now again be sure that the pair  $(y, z)$  is described by a Gaussian joint probability distribution with  $\langle y(\mathbf{x}) \rangle = \langle z(\mathbf{x}) \rangle = 0$ . The second order moments are given by

$$\begin{aligned} \langle z^2 \rangle &= \sum_{i,j=1}^N w_i w_j \langle x_i x_j \rangle \\ \langle yz \rangle &= \langle [\sum_i w_i x_i] [\sum_j w_j (x_j + \xi_j)] \rangle = \sum_{i,j=1}^N w_i w_j [\langle x_i x_j \rangle + \langle x_i \xi_j \rangle] = \sum_{i,j=1}^N w_i w_j \langle x_i x_j \rangle \\ \langle y^2 \rangle &= \langle [\sum_i w_i (x_i + \xi_i)] [\sum_j w_j (x_j + \xi_j)] \rangle = \sum_{i,j=1}^N w_i w_j [\langle x_i x_j \rangle + \langle \xi_i \xi_j \rangle] \\ &= \sum_{i,j=1}^N w_i w_j \langle x_i x_j \rangle + \sum_{i=1}^N w_i^2 \sigma_i^2 \end{aligned}$$

Substitution into  $\tilde{I}(Y, Z)$  (3.20) gives:

$$\begin{aligned} \tilde{I}(Y, Z) &= -\frac{1}{2} \ 2 \log \left[ 1 - \frac{\langle yz \rangle^2}{\langle y^2 \rangle \langle z^2 \rangle} \right] = -\frac{1}{2} \ 2 \log \left[ 1 - \frac{\sum_{i,j=1}^N w_i w_j \langle x_i x_j \rangle}{\sum_{i,j=1}^N w_i w_j \langle x_i x_j \rangle + \sum_{i=1}^N w_i^2 \sigma_i^2} \right] \\ &= -\frac{1}{2} \ 2 \log \left[ \frac{\sum_{i=1}^N w_i^2 \sigma_i^2}{\sum_{i,j=1}^N w_i w_j \langle x_i x_j \rangle + \sum_{i=1}^N w_i^2 \sigma_i^2} \right] = \frac{1}{2} \ 2 \log \left[ 1 + \frac{\sum_{i,j=1}^N w_i w_j \langle x_i x_j \rangle}{\sum_{i=1}^N w_i^2 \sigma_i^2} \right] \end{aligned} \quad (5.24)$$

Now we find a completely different situation. Note that (with  $v_i = w_i \sigma_i$ ):

$$\max_{\mathbf{w} \in \mathbb{R}^N} \frac{\sum_{i,j=1}^N w_i w_j \langle x_i x_j \rangle}{\sum_{i=1}^N w_i^2 \sigma_i^2} = \max_{\mathbf{v} \in \mathbb{R}^N} \frac{\sum_{i,j=1}^N v_i \langle (x_i / \sigma_i) (x_j / \sigma_j) \rangle v_j}{\sum_{i=1}^N v_i^2} = \Lambda < \infty$$

where  $\Lambda$  is the largest eigenvalue of the (non-negative) matrix with entries

$$L_{ij} = \sigma_i^{-1} \langle x_i x_j \rangle \sigma_j^{-1}$$

The maximum differential mutual information between uncorrupted and corrupted signal is obtained when the vector  $\mathbf{v}$  is chosen to be an eigenvector of the matrix  $\{L_{ij}\}$  with eigenvalue  $\Lambda$ . In terms of  $\mathbf{w}$  this means

$$\mathbf{w}^{\text{opt}} = \text{solution of } \sum_{j=1}^N \langle x_i x_j \rangle w_j = \Lambda \sigma_i^2 w_i \quad \text{with largest } \Lambda \quad (5.25)$$



Expression (5.24) for the differential information is no longer affected by a simple boost of all synapses. This would simply increase both the signal and the noise, without any net effect on the signal/noise ratio.

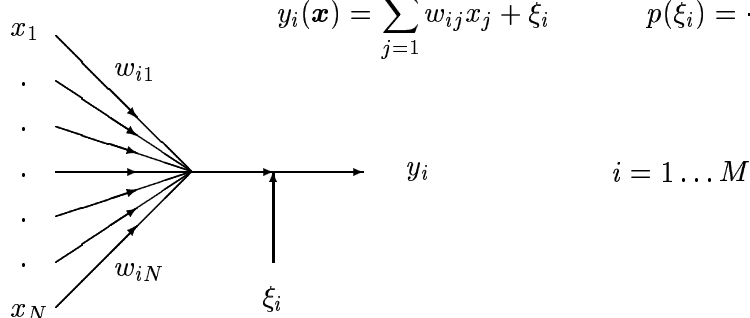
Let us finally choose the simplest example, where the input signals  $x_i$  are mutually independent so that  $\langle x_i x_j \rangle = S_i^2 \delta_{ij}$ . The matrix elements  $L_{ij}$  become  $L_{ij} = [S_i^2 / \sigma_i^2] \delta_{ij}$ . The largest eigenvalue is

$$\Lambda = \max_i S_i^2 / \sigma_i^2$$

Let us denote the set of all channels  $i$  with the largest signal/noise ratio by  $S = \{i \mid S_i / \sigma_i = \sqrt{\Lambda}\}$ . The optimal synaptic weights  $\mathbf{w}$  are now those where the system ‘tunes’ into these particular input channels:  $w_i^{\text{opt}} = 0$  for  $i \notin S$ ,  $w_i^{\text{opt}} \neq 0$  for  $i \in S$ . If there is precisely one input channel with the largest signal/noise ratio, then the optimal network will only be connected to this single channel.

### 5.3 Neuronal Specialisation

In the first case we studied in this section, a single neuron with Gaussian output noise, a simple boost of all synaptic weights allowed the neuron to obtain any desired value of the differential mutual information between the uncorrupted and corrupted signals. However, in practice one usually has constraints on the possible values of the weights which forbid unlimited boosting. Here we will inspect such cases in the context of a population of  $M$  linear neurons  $y_i : \mathfrak{R}^N \rightarrow \mathfrak{R}$  ( $i = 1 \dots M$ ), the outputs  $y_i(\mathbf{x})$  of which are corrupted by Gaussian noise sources  $\xi_i$ :

$$y_i(\mathbf{x}) = \sum_{j=1}^N w_{ij} x_j + \xi_i \quad p(\xi_i) = \frac{e^{-\frac{1}{2}\xi_i^2/\sigma_i^2}}{\sigma_i \sqrt{2\pi}} \quad (5.26)$$


with the synaptic weights  $\{w_{ij}\}$ . The strengths of the noise channels are given by  $\sigma_i^2 = \langle \xi_i^2 \rangle$ . We assume the input signals to obey  $\langle x_i \rangle = 0$  and to be statistically independent of the noise sources, and we assume the uncorrupted signals  $z_i(\mathbf{x}) = \sum_{j=1}^N w_{ij} x_j$  to have a Gaussian joint probability distribution. As a result we can be sure that the pair  $(\mathbf{y}, \mathbf{z})$  is described by a Gaussian joint probability distribution, where  $\mathbf{y} = (y_1, \dots, y_M)$  and  $\mathbf{z} = (z_1, \dots, z_M)$ , with  $\langle \mathbf{y} \rangle = \langle \mathbf{z} \rangle = 0$ , so that the mutual information between the uncorrupted signals  $\mathbf{z}$  and the corrupted signals  $\mathbf{y}$  is given by equation (3.21). The various covariance matrices are

$$\begin{aligned} C_{ij}^{zz} &= \langle z_i z_j \rangle = \sum_{kl=1}^N w_{ik} w_{jl} \langle x_k x_l \rangle \\ C_{ij}^{yz} &= C_{ij}^{zy} = \langle y_i z_j \rangle = \sum_{kl=1}^N w_{ik} w_{jl} \langle x_k x_l \rangle \\ C_{ij}^{yy} &= \langle y_i y_j \rangle = \sum_{kl=1}^N w_{ik} w_{jl} \langle x_k x_l \rangle + \langle \xi_i \xi_j \rangle = \sum_{kl=1}^N w_{ik} w_{jl} \langle x_k x_l \rangle + \sigma_i^2 \delta_{ij} \end{aligned}$$

In view of the prominent role of just two matrices, we use the short-hands  $C_{ij} = C_{ij}^{zz}$  and  $D_{ij} = \sigma_i^2 \delta_{ij}$ , with which we can write equation (3.21) in the form

$$\tilde{I}(\mathbf{Y}, \mathbf{Z}) = -\frac{1}{2} {}^2\log \det \begin{pmatrix} \mathbf{I} & (\mathbf{C} + \mathbf{D})^{-1} \mathbf{C} \\ \mathbf{I} & \mathbf{I} \end{pmatrix} \quad (5.27)$$

where  $\mathbf{I}$  denotes the  $M \times M$  identity matrix.

From this stage onwards we will restrict ourselves for simplicity to the simple situation where the input signals  $x_i$  are statistically independent, so  $\langle x_i x_j \rangle = S_i^2 \delta_{ij}$ , where the different noise signals are of uniform strength, so  $\sigma_i^2 = \sigma^2$ , and where there are just two neurons,  $M = 2$ . We also assume for simplicity that all input variances  $S_i^2$  are different (so that in subsequent calculations we will have no degenerate eigenspaces). We define

$$\begin{aligned} \epsilon_1 &= \sigma^{-2} \sum_{k=1}^N w_{1k}^2 S_k^2 \\ \epsilon_2 &= \sigma^{-2} \sum_{k=1}^N w_{2k}^2 S_k^2 \\ \epsilon_{12} &= \sigma^{-2} \sum_{k=1}^N w_{1k} w_{2k} S_k^2 \end{aligned} \quad (5.28)$$

Since  $\mathbf{C}$  and  $\mathbf{D}$  now commute, the matrix  $(\mathbf{C} + \mathbf{D})^{-1} \mathbf{C}$  in (5.27) can now be written as

$$(\mathbf{C} + \mathbf{D})^{-1} [\mathbf{C} + \mathbf{D} - \mathbf{D}] = \mathbf{I} - (\mathbf{C} + \mathbf{D})^{-1} \mathbf{D} = \mathbf{I} - \mathbf{D}^{-1} (\mathbf{I} + \mathbf{C} \mathbf{D}^{-1})^{-1} \mathbf{D} = \mathbf{I} - \begin{pmatrix} 1 + \epsilon_1 & \epsilon_{12} \\ \epsilon_{12} & 1 + \epsilon_2 \end{pmatrix}^{-1}$$

Furthermore we can use the rule (which can be verified directly for  $M = 2$ ):

$$\det \begin{pmatrix} \mathbf{I} & \mathbf{I} - \mathbf{K} \\ \mathbf{I} & \mathbf{I} \end{pmatrix} = \det \mathbf{K}$$

which gives:

$$\tilde{I}(\mathbf{Y}, \mathbf{Z}) = \frac{1}{2} {}^2\log \det \begin{pmatrix} 1 + \epsilon_1 & \epsilon_{12} \\ \epsilon_{12} & 1 + \epsilon_2 \end{pmatrix} = \frac{1}{2} {}^2\log [1 + \epsilon_1 + \epsilon_2 + \epsilon_1 \epsilon_2 - \epsilon_{12}^2] \quad (5.29)$$

We are interested in finding out which is the optimal choice for the weights in the the case where they are prevented from unbounded growing via a constraint. For this constraint we choose the so-called spherical one:

$$\sum_{k=1}^N w_{1k}^2 = \sum_{k=1}^N w_{2k}^2 = 1 \quad (5.30)$$

Now we will inspect various regime for the noise level  $\sigma$ , where we will observe completely different optimal weight arrangements.

*High noise levels,  $\sigma \gg 1$ .* Since each term in (5.28) is of order  $\mathcal{O}(\sigma^{-2})$  the terms which are quadratic in the  $\epsilon$ 's are of vanishing order compared to the ones linear in the  $\epsilon$ 's. Thus we get, upon using  $\log(1+x) = x + \mathcal{O}(x^2)$ :

$$\tilde{I}(\mathbf{Y}, \mathbf{Z}) = \frac{1}{2 \log 2} \log [1 + \epsilon_1 + \epsilon_2 + \mathcal{O}(\sigma^{-4})] = \frac{1}{2 \sigma^2 \log 2} \sum_{k=1}^N S_k^2 [w_{1k}^2 + w_{2k}^2] + \mathcal{O}(\sigma^{-4})$$

We now calculate the maximum of the leading term in this expression under the constraints (5.30) using Lagrange multipliers, which gives the following equations for the extrema:

$$\frac{\partial \tilde{I}}{\partial w_{1k}} = \gamma_1 w_{1k} \quad \frac{\partial \tilde{I}}{\partial w_{2k}} = \gamma_2 w_{2k} \quad \sum_{k=1}^N w_{1k}^2 = \sum_{k=1}^N w_{2k}^2 = 1$$

(in which  $\gamma_1$  and  $\gamma_2$  are the Lagrange multipliers). Working out the derivatives in the leading order in  $\sigma$  gives:

$$S_k^2 w_{1k} = \gamma_1 \sigma^2 \log 2 w_{1k} \quad S_k^2 w_{2k} = \gamma_2 \sigma^2 \log 2 w_{2k} \quad \sum_{k=1}^N w_{1k}^2 = \sum_{k=1}^N w_{2k}^2 = 1$$

Since all  $S_k$  are assumed different, the only solutions are those where both weight vectors have just one non-zero component:  $(\exists i_1)(\forall k \neq i_1) : w_{1k} = 0$ , and  $(\exists i_2)(\forall k \neq i_2) : w_{2k} = 0$ . The non-zero components obey (normalisation):  $w_{1i_1}^2 = w_{2i_2}^2 = 1$ . The corresponding value for the leading order of the differential mutual information is

$$\tilde{I}(\mathbf{Y}, \mathbf{Z}) = \frac{S_{i_1}^2 + S_{i_2}^2}{2\sigma^2 \log 2}$$

Its maximum is obtained when both non-zero weight components are those coupled to the strongest input signal:  $i_1 = i_2 = i^*$ , where  $i^*$  is defined as  $\max_i S_i^2 = S_{i^*}^2$ . Apparently for high noise levels we make the best use of our available hardware if we force our neurons to team up and both tune into the strongest input channel.

*Low noise levels,  $\sigma \ll 1$ .* Since all  $\epsilon$ 's are of order  $\mathcal{O}(\sigma^{-2})$ , in this case the dominant terms in (5.29) are those which are quadratic in the  $\epsilon$ 's:

$$\begin{aligned} \tilde{I}(\mathbf{Y}, \mathbf{Z}) &= \frac{1}{2} {}^2\log \left[ \epsilon_1 \epsilon_2 - \epsilon_{12}^2 + \mathcal{O}(\sigma^{-2}) \right] \\ &= \frac{1}{2} {}^2\log \left[ \frac{1}{\sigma^4} \left( \sum_{k=1}^N w_{1k}^2 S_k^2 \right) \left( \sum_{k=1}^N w_{2k}^2 S_k^2 \right) - \frac{1}{\sigma^4} \left( \sum_{k=1}^N w_{1k} w_{2k} S_k^2 \right)^2 + \mathcal{O}(\sigma^{-2}) \right] \end{aligned}$$

We now have to calculate the maximum of the leading term in the argument of the logarithm, under the constraints (5.30) using Lagrange multipliers, which gives the following equations:

$$2S_k^2 \left[ w_{1k} \left( \sum_{k=1}^N w_{2k}^2 S_k^2 \right) - w_{2k} \left( \sum_{k=1}^N w_{1k} w_{2k} S_k^2 \right) \right] = \gamma_1 w_{1k} \quad (5.31)$$

$$2S_k^2 \left[ w_{2k} \left( \sum_{k=1}^N w_{1k}^2 S_k^2 \right) - w_{1k} \left( \sum_{k=1}^N w_{1k} w_{2k} S_k^2 \right) \right] = \gamma_2 w_{2k} \quad (5.32)$$

(as before to be solved for each  $k$ , in combination with the constraint equations (5.30)). We now put

$$\tilde{\epsilon}_1 = \sum_{k=1}^N w_{1k}^2 S_k^2 \quad \tilde{\epsilon}_2 = \sum_{k=1}^N w_{2k}^2 S_k^2 \quad \tilde{\epsilon}_{12} = \sum_{k=1}^N w_{1k} w_{2k} S_k^2$$

so that eqns (5.31,5.31) acquire the compact form

$$\begin{pmatrix} 2S_k^2\tilde{\epsilon}_2 - \gamma_1 & -2S_k^2\tilde{\epsilon}_{12} \\ -2S_k^2\tilde{\epsilon}_{12} & 2S_k^2\tilde{\epsilon}_1 - \gamma_2 \end{pmatrix} \begin{pmatrix} w_{1k} \\ w_{2k} \end{pmatrix} = 0$$

so

$$\forall k : \quad (w_{1k}, w_{2k}) = 0 \quad \text{or} \quad \left[ S_k^2\tilde{\epsilon}_2 - \frac{1}{2}\gamma_1 \right] \left[ S_k^2\tilde{\epsilon}_1 - \frac{1}{2}\gamma_2 \right] = S_k^4\tilde{\epsilon}_{12}^2$$

Since all  $S_k$  are assumed different, and since the right-hand side of this equation when solved for  $S_k^2$  can have at most two solutions, we are forced to conclude that each weight vector can have at most two nonzero-components, with identical indices:  $(\exists i_1, i_2)(\forall k \neq i_1, i_2) : w_{1k} = w_{2k} = 0$ . Normalisation dictates  $w_{i_1}^2 + w_{i_2}^2 = w_{2i_1}^2 + w_{2i_2}^2 = 1$ . The corresponding value for the leading order of the differential mutual information is now

$$\begin{aligned} i_1 = i_2 : \quad & \tilde{I}(\mathbf{Y}, \mathbf{Z}) = \frac{1}{2} {}^2\log \left[ \mathcal{O}(\sigma^{-2}) \right] \\ i_1 \neq i_2 : \quad & \tilde{I}(\mathbf{Y}, \mathbf{Z}) = \frac{1}{2} {}^2\log \left[ \frac{1}{\sigma^4} \left( w_{i_1}^2 S_{i_1}^2 + w_{i_2}^2 S_{i_2}^2 \right) \left( w_{2i_1}^2 S_{i_1}^2 + w_{2i_2}^2 S_{i_2}^2 \right) \right. \\ & \quad \left. - \frac{1}{\sigma^4} \left( w_{1i_1} w_{2i_1} S_{i_1}^2 + w_{1i_2} w_{2i_2} S_{i_2}^2 \right)^2 + \mathcal{O}(\sigma^{-2}) \right] \\ & = \frac{1}{2} {}^2\log \left[ \frac{1}{\sigma^4} S_{i_1}^2 S_{i_2}^2 \left( w_{1i_1} w_{2i_2} - w_{1i_2} w_{2i_1} \right)^2 + \mathcal{O}(\sigma^{-2}) \right] \end{aligned}$$

Clearly the maximum is obtained for  $i_1 \neq i_2$ . To work out the last maximisation step we use the constraints and write

$$w_{1i_1} = \cos \phi_1, \quad w_{1i_2} = \sin \phi_1, \quad w_{2i_1} = \cos \phi_2, \quad w_{2i_2} = \sin \phi_2$$

giving

$$\begin{aligned} \tilde{I}(\mathbf{Y}, \mathbf{Z}) &= \frac{1}{2} {}^2\log \left[ \frac{1}{\sigma^4} S_{i_1}^2 S_{i_2}^2 \left( \cos \phi_1 \sin \phi_2 - \sin \phi_1 \cos \phi_2 \right)^2 + \mathcal{O}(\sigma^{-2}) \right] \\ &= \frac{1}{2} {}^2\log \left[ \frac{1}{\sigma^4} S_{i_1}^2 S_{i_2}^2 \sin^2(\phi_1 - \phi_2) + \mathcal{O}(\sigma^{-2}) \right] \end{aligned}$$

The maximum is obtained for  $\phi_1 = \phi_2 + \frac{1}{2}\pi + n\pi$ , where

$$\begin{pmatrix} w_{1i_1} \\ w_{1i_2} \end{pmatrix} \cdot \begin{pmatrix} w_{2i_1} \\ w_{2i_2} \end{pmatrix} = \begin{pmatrix} \cos(\phi_2 + \frac{\pi}{2} + n\pi) \\ \sin(\phi_2 + \frac{\pi}{2} + n\pi) \end{pmatrix} \cdot \begin{pmatrix} \cos \phi_2 \\ \sin \phi_2 \end{pmatrix} = \begin{pmatrix} (-1)^{n+1} \sin(\phi_2) \\ (-1)^n \cos(\phi_2) \end{pmatrix} \cdot \begin{pmatrix} \cos \phi_2 \\ \sin \phi_2 \end{pmatrix} = 0$$

The optimal non-zero weight vectors of our two neurons (with two non-zero components each) are mutually orthogonal, and we get the resulting value

$$\tilde{I}(\mathbf{Y}, \mathbf{Z}) = \frac{1}{2} {}^2\log \left[ \frac{1}{\sigma^4} S_{i_1}^2 S_{i_2}^2 + \mathcal{O}(\sigma^{-2}) \right]$$

which is maximal if the two input channels  $i_1$  and  $i_2$  are chosen to be the strongest two (note that we are forbidden to choose  $i_1 = i_2$  here), so

$$S_{\max}^2 = S_{i_1}^2 > S_{i_2}^2 > \dots, \quad \text{or} \quad S_{\max}^2 = S_{i_2}^2 > S_{i_1}^2 > \dots$$

Apparently for low noise levels the best strategy is no longer for our two neurons to team up, but rather to let them specialise and form an orthogonal basis in the space of the two strongest non-identical channels.

*The Transition to Specialisation.* So far we just checked the two extreme cases of very high and very low noise levels. The results hint at the existence of a transition, at some critical noise level, where specialisation sets in. To simplify notation we arrange the input channels in such a way that

$$S_1 > S_2 > S_3 > \dots$$

Knowing that optimum configurations in both extreme cases  $\sigma \rightarrow 0$  and  $\sigma \rightarrow \infty$  exhibit at most two non-zero weight components for each neuron, we can now inspect intermediate noise levels by putting

$$(w_{11}, w_{12}) = (\cos \phi_1, \sin \phi_1), \quad (w_{21}, w_{22}) = (\cos \phi_2, \sin \phi_2)$$

In terms of the angles  $(\phi_1, \phi_2)$  maximising the differential mutual information (5.29) amounts to maximising the following quantity:

$$\begin{aligned} L &= \sigma^4 [\epsilon_1 + \epsilon_2 + \epsilon_1 \epsilon_2 - \epsilon_{12}^2] = \sigma^2 [\cos^2 \phi_1 S_1^1 + \sin^2 \phi_1 S_2^2 + \cos^2 \phi_2 S_1^1 + \sin^2 \phi_2 S_2^2] \\ &+ (\cos^2 \phi_1 S_1^2 + \sin^2 \phi_1 S_2^2) (\cos^2 \phi_2 S_1^2 + \sin^2 \phi_2 S_2^2) - (\cos \phi_1 \cos \phi_2 S_1^2 + \sin \phi_1 \sin \phi_2 S_2^2)^2 \\ &= \sigma^2 [2S_2^2 + (S_1^2 - S_2^2)(\cos^2 \phi_1 + \cos^2 \phi_2)] + S_1^2 S_2^2 \sin^2(\phi_1 - \phi_2) \\ &= \sigma^2 \left[ S_1^2 + S_2^2 + \frac{1}{2}(S_1^2 - S_2^2)(\cos(2\phi_1) + \cos(2\phi_2)) \right] + \frac{1}{2} S_1^2 S_2^2 - \frac{1}{2} S_1^2 S_2^2 \cos(2\phi_1 - 2\phi_2) \quad (5.33) \end{aligned}$$

The weight constraints (5.30) have now been built-in, so that extremisation of (5.33) reduces to simply putting two derivatives to zero:

$$\begin{aligned} \frac{\partial L}{\partial \phi_1} = 0 : \quad & \sigma^2 (S_1^2 - S_2^2) \sin(2\phi_1) = S_1^2 S_2^2 \sin(2\phi_1 - 2\phi_2) \\ \frac{\partial L}{\partial \phi_2} = 0 : \quad & \sigma^2 (S_1^2 - S_2^2) \sin(2\phi_2) = -S_1^2 S_2^2 \sin(2\phi_1 - 2\phi_2) \end{aligned} \quad (5.34)$$

Addition of the two equations in (5.34) shows that (5.34) can be replaced by the equivalent set

$$\begin{aligned} \sin(2\phi_1) &= -\sin(2\phi_2) \\ \sigma^2 (S_1^2 - S_2^2) \sin(2\phi_1) &= S_1^2 S_2^2 \sin(2\phi_1 - 2\phi_2) \end{aligned} \quad (5.35)$$

The set (5.35) admits two types of solutions (modulo irrelevant multiples of  $2\pi$ ):

$$\begin{aligned} 2\phi_2 = -2\phi_1 : \quad & \sigma^2 (S_1^2 - S_2^2) \sin(2\phi_1) = S_1^2 S_2^2 \sin(4\phi_1) \\ & \sin(2\phi_1) [\sigma^2 (S_1^2 - S_2^2) - 2S_1^2 S_2^2 \cos(2\phi_1)] = 0 \\ & 2\phi_1 \in \{0, \pi\} \quad \text{or} \quad \cos(2\phi_1) = \frac{1}{2} \sigma^2 (S_1^2 - S_2^2) / S_1^2 S_2^2 \\ 2\phi_2 = 2\phi_1 + \pi : \quad & \sin(2\phi_1) = 0 \quad \text{so} \quad 2\phi_1 \in \{0, \pi\} \end{aligned}$$

We can now simply list the various extrema with their corresponding value for the quantity  $L$  we wish to maximise. First we have the simplest ones,  $2\phi_1 \in \{0, \pi\}$  with  $2\phi_2 = -2\phi_1$ , where both neurons tune onto the same channel:

$$\begin{array}{ccccccc}
\cos(2\phi_1) & \sin(2\phi_1) & \cos(2\phi_2) & \sin(2\phi_2) & (w_{11}, w_{12}) & (w_{21}, w_{22}) & L : \\
1 & 0 & 1 & 0 & (\pm 1, 0) & (\pm 1, 0) & 2\sigma^2 S_1^2 \\
-1 & 0 & -1 & 0 & (0, \pm 1) & (0, \pm 1) & 2\sigma^2 S_2^2
\end{array} \quad (5.36)$$

Then we have extrema,  $2\phi_1 \in \{0, \pi\}$  with  $2\phi_2 = 2\phi_1 + \pi$ , where each neuron tunes into a single but different input channel:

$$\begin{array}{ccccccc}
\cos(2\phi_1) & \sin(2\phi_1) & \cos(2\phi_2) & \sin(2\phi_2) & (w_{11}, w_{12}) & (w_{21}, w_{22}) & L : \\
1 & 0 & -1 & 0 & (\pm 1, 0) & (0, \pm 1) & \sigma^2(S_1^2 + S_2^2) + S_1^2 S_2^2 \\
-1 & 0 & 1 & 0 & (0, \pm 1) & (\pm 1, 0) & \sigma^2(S_1^2 + S_2^2) + S_1^2 S_2^2
\end{array} \quad (5.37)$$

Finally there is a non-trivial solution, where each neuron tunes into a specific combination of the two strongest channels:

$$\cos(2\phi_2) = \cos(2\phi_1), \quad \sin(2\phi_2) = -\sin(2\phi_1), \quad \cos(2\phi_1) = \frac{\sigma^2(S_1^2 - S_2^2)}{2S_1^2 S_2^2} \quad (5.38)$$

$$L = \sigma^2(S_1^2 + S_2^2) + S_1^2 S_2^2 + \frac{\sigma^4(S_1^2 - S_2^2)^2}{4S_1^2 S_2^2} \quad (5.39)$$

The solution (5.38) only comes into existence for small noise levels (since  $\cos(2\phi_i) \leq 1$ ):

$$\sigma \leq \sigma_c = \left[ \frac{2S_1^2 S_2^2}{S_1^2 - S_2^2} \right]^{\frac{1}{2}} \quad (5.40)$$

Since the solution (5.38) obeys  $\phi_2 = -\phi_1 + n\pi$ , we can directly calculate the angle between the weight vectors  $(w_{11}, w_{12})$  and  $(w_{21}, w_{22})$  of our two neurons:

$$\begin{aligned}
\begin{pmatrix} w_{11} \\ w_{12} \end{pmatrix} \cdot \begin{pmatrix} w_{21} \\ w_{22} \end{pmatrix} &= \cos \phi_1 \cos(n\pi - \phi_1) + \sin \phi_1 \sin(n\pi - \phi_1) = (-1)^n (\cos^2 \phi_1 - \sin^2 \phi_1) \\
&= (-1)^n \cos(2\phi_1) = (-1)^n \sigma^2 / \sigma_c^2
\end{aligned}$$

(where we have used (5.38) and (5.40)). This expression shows that the solution (5.38) starts off at  $\sigma = \sigma_c$  as two (anti-)parallel weight vectors (as in (5.36), followed by a continuous ‘unfolding’ of the two weight vectors as the noise further decreases, until they form a perfect orthogonal basis for  $\sigma = 0$ . Comparison of (5.39) with (5.37) immediately shows that the solution (5.38), provided it exists, has a larger value for  $L$ . Finally we check the condition for the solution (5.38) to give the maximum mutual information (i.e. for  $L$  in (5.39) to be also larger than  $2\sigma^2 S_1^2$ ):

$$\sigma^2(S_1^2 + S_2^2) + S_1^2 S_2^2 + \frac{\sigma^4(S_1^2 - S_2^2)^2}{4S_1^2 S_2^2} > 2\sigma^2 S_1^2$$

$$\sigma^4 - 2\sigma^2 \left[ \frac{2S_1^2 S_2^2}{S_1^2 - S_2^2} \right] + \left[ \frac{2S_1^2 S_2^2}{S_1^2 - S_2^2} \right]^2 > 0$$

Since this condition is identical to  $(\sigma^2 - \sigma_c^2)^2 > 0$  (see (5.40)) we conclude that as soon as the solution (5.38) exists it will give the maximum mutual information.

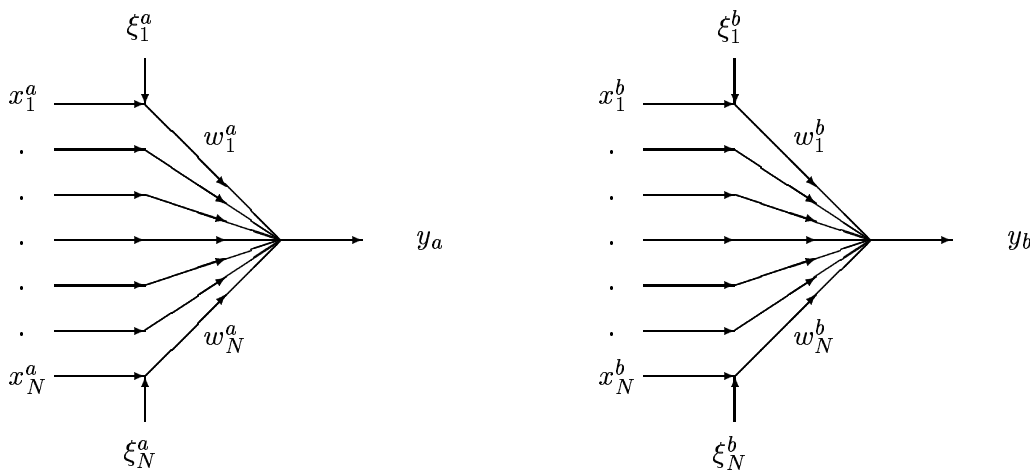
If the solution (5.38) does not exist, i.e. when  $\sigma > \sigma_c$ , we have to find the maximum of  $L$  by comparing the values (5.36) and (5.37). The condition  $\sigma > \sigma_c$  is then found to translate into the statement that the value in (5.36) is largest. In conclusion, all this implies that the optimal weight arrangement indeed shows a specialisation transition at the critical noise level given in (5.40). For  $\sigma > \sigma_c$  both neurons operate identical rules; for  $\sigma < \sigma_c$  the two neurons specialise and start to operate different rules. This example illustrates how heuristic strategies like doing detailed data fitting only in cases where there is not much noise in the data can be rigorously quantified with information-theoretic tools. Here the theory tells us exactly when to specialise and when not to.

## 5.4 Detection of Coherent Features

We now turn to a different problem. Imagine two types of input channels,  $\{x_i^a\}$  and  $\{x_i^b\}$ , each corrupted by independent Gaussian noise sources  $\{\xi_i^a\}$  and  $\{\xi_i^b\}$ , and each feeding into a linear neuron,  $y_a : \mathfrak{R}^N \rightarrow \mathfrak{R}$  and  $y_b : \mathfrak{R}^N \rightarrow \mathfrak{R}$ , respectively:

$$y_a(\mathbf{x}^a) = \sum_{i=1}^N w_i^a (x_i^a + \xi_i^a), \quad p(\xi_i^a) = \frac{e^{-\frac{1}{2}(\xi_i^a)^2/\sigma^2}}{\sigma\sqrt{2\pi}}$$

$$y_b(\mathbf{x}^b) = \sum_{i=1}^N w_i^b (x_i^b + \xi_i^b), \quad p(\xi_i^b) = \frac{e^{-\frac{1}{2}(\xi_i^b)^2/\sigma^2}}{\sigma\sqrt{2\pi}}$$



with the two types of synaptic weights  $\{w_i^a\}$  and  $\{w_i^b\}$ . The uniform strength of the  $2N$  noise sources  $\xi_i^{a,b}$  is measured by  $\sigma^2 = \langle (\xi_i^{a,b})^2 \rangle$ . Note:  $\langle \xi_i^a \xi_j^b \rangle = 0$ .

Our problem will be to extract from the two input streams  $\mathbf{x}^a$  and  $\mathbf{x}^b$  only the information which the two have in common. This is a familiar problem in sensory information processing

in biology, where various sensory systems each provide only partial and often messy information. The total information stream is integrated and/or filtered in such a way that only the underlying regularities, that are coherent across the various sensory systems, are retained (i.e. merging visual images to produce stereo vision, the integration of the various sensors that measure muscle stretch in order to build a correct internal representation of the position of the various limbs, etc). In order to study such a situation let us assume here (the simplest scenario) that the original uncorrupted input signals are related a deterministic way, for instance via a unitary transformation (a rotation in  $\mathfrak{R}^N$ ):

$$\mathbf{x}^b = \mathbf{U}\mathbf{x}^a \quad \mathbf{U}\mathbf{U}^\dagger = \mathbf{U}^\dagger\mathbf{U} = \mathbf{I}$$

with  $(\mathbf{U}^\dagger)_{ij} = U_{ji}$  and  $\mathbf{I}_{ij} = \delta_{ij}$  (the identity matrix). The appropriate strategy here is to maximise the differential mutual information between the two neuron outputs  $y_a$  and  $y_b$ , since this quantity is precisely generated by the information that is coherent across the two input streams.

As before we assume all input signals to obey  $\langle x_i^{a,b} \rangle = 0$  and to be statistically independent of the noise sources, and that the uncorrupted signals  $z_a(\mathbf{x}_a) = \sum_{i=1}^N w_i^a x_i^a$  and  $z_b(\mathbf{x}_b) = \sum_{i=1}^N w_i^b x_i^b$  have a Gaussian joint probability distribution, so that the same must be true for the pair  $(y_a, y_b)$ . We write the identity matrix as  $\mathbf{I}$ , the two weight vectors as  $\mathbf{w}^a = (w_1^a, \dots, w_N^a)$  and  $\mathbf{w}^b = (w_1^b, \dots, w_N^b)$ , and we define the covariance matrix  $\mathbf{C}$ :  $C_{ij} = \langle x_i^a x_j^a \rangle$ . The second order moments are given by

$$\begin{aligned} \langle y_a^2 \rangle &= \sum_{ij=1}^N w_i^a w_j^a [\langle x_i^a x_j^a \rangle + \langle \xi_i^a \xi_j^a \rangle] = \mathbf{w}^a \cdot (\mathbf{C} + \sigma^2 \mathbf{I}) \mathbf{w}^a \\ \langle y_a y_b \rangle &= \sum_{ij=1}^N w_i^a w_j^b \langle x_i^a x_j^b \rangle = \sum_{ijk=1}^N w_i^a w_j^b U_{jk} \langle x_i^a x_k^a \rangle = \mathbf{w}^a \cdot \mathbf{C} \mathbf{U}^\dagger \mathbf{w}^b \\ \langle y_b^2 \rangle &= \sum_{ij=1}^N w_i^b w_j^b [\langle x_i^b x_j^b \rangle + \langle \xi_i^b \xi_j^b \rangle] = \sum_{ijkl=1}^N w_i^b w_j^b U_{ik} U_{jl} \langle x_k^a x_l^a \rangle + \sigma^2 (\mathbf{w}^b)^2 \\ &= \mathbf{w}^b \cdot \mathbf{U} (\mathbf{C} + \sigma^2 \mathbf{I}) \mathbf{U}^\dagger \mathbf{w}^b \end{aligned}$$

Substitution into  $\tilde{I}(Y, Z)$  (3.20) gives:

$$\tilde{I}(Y_a, Y_b) = -\frac{1}{2} {}^2\log \left[ 1 - \frac{(\mathbf{w}^a \cdot \mathbf{C} \mathbf{U}^\dagger \mathbf{w}^b)^2}{[\mathbf{w}^a \cdot (\mathbf{C} + \sigma^2 \mathbf{I}) \mathbf{w}^a] [\mathbf{w}^b \cdot \mathbf{U} (\mathbf{C} + \sigma^2 \mathbf{I}) \mathbf{U}^\dagger \mathbf{w}^b]} \right] \quad (5.41)$$

We now try to maximise (5.41), which is equivalent to maximising the fraction in the argument of the logarithm. This fraction is greatly simplified by the symmetrising transformation  $\mathbf{w} = \mathbf{w}^a$ ,  $\mathbf{v} = \mathbf{U}^\dagger \mathbf{w}^b$  (so  $\mathbf{w}^b = \mathbf{U}\mathbf{v}$ ), which reduces our problem to calculating

$$L = \max_{\mathbf{w}, \mathbf{v} \in \mathfrak{R}^N} \frac{(\mathbf{w} \cdot \mathbf{C} \mathbf{v})^2}{[\mathbf{w} \cdot (\mathbf{C} + \sigma^2 \mathbf{I}) \mathbf{w}] [\mathbf{v} \cdot (\mathbf{C} + \sigma^2 \mathbf{I}) \mathbf{v}]}$$

Extremisation by putting the various derivatives  $\frac{\partial L}{\partial w_i}$  and  $\frac{\partial L}{\partial v_i}$  to zero gives the two vector equations:

$$\left[ \mathbf{w} \cdot (\mathbf{C} + \sigma^2 \mathbf{I}) \mathbf{w} \right] \mathbf{C} \mathbf{v} = (\mathbf{w} \cdot \mathbf{C} \mathbf{v}) (\mathbf{C} + \sigma^2 \mathbf{I}) \mathbf{w} \quad \left[ \mathbf{v} \cdot (\mathbf{C} + \sigma^2 \mathbf{I}) \mathbf{v} \right] \mathbf{C} \mathbf{w} = (\mathbf{v} \cdot \mathbf{C} \mathbf{w}) (\mathbf{C} + \sigma^2 \mathbf{I}) \mathbf{v}$$



which can be written as

$$\mathbf{v} = \Lambda_1(\mathbf{I} + \sigma^2 \mathbf{C}^{-1})\mathbf{w}, \quad \mathbf{w} = \Lambda_2(\mathbf{I} + \sigma^2 \mathbf{C}^{-1})\mathbf{v} \quad (5.42)$$

$$\Lambda_1 = \frac{\mathbf{w} \cdot \mathbf{C}\mathbf{v}}{\mathbf{w} \cdot (\mathbf{C} + \sigma^2 \mathbf{I})\mathbf{w}}, \quad \Lambda_2 = \frac{\mathbf{v} \cdot \mathbf{C}\mathbf{w}}{\mathbf{v} \cdot (\mathbf{C} + \sigma^2 \mathbf{I})\mathbf{v}}, \quad L = \Lambda_1 \Lambda_2 \quad (5.43)$$

Combining the two equations in (5.42) gives the eigenvalue problem  $\mathbf{w} = L(\mathbf{I} + \sigma^2 \mathbf{C}^{-1})^2 \mathbf{w}$ , the solutions of which are just the eigenvectors of the covariance matrix  $\mathbf{C}$ . If we denote its eigenvalues by  $\lambda$ , we arrive at:

$$\mathbf{C}\mathbf{w}_\lambda = \lambda\mathbf{w}_\lambda, \quad \mathbf{C}\mathbf{v}_\lambda = \lambda\mathbf{v}_\lambda, \quad L = \left[ \frac{\lambda}{\lambda + \sigma^2} \right]^2$$

The maximum in (5.41) is obtained if we choose for  $\lambda$  the largest eigenvalue  $\lambda_{\max}$  of  $\mathbf{C}$ :

$$\max \tilde{I}(Y_a, Y_b) = -\frac{1}{2} {}^2\log \left[ 1 - \frac{\lambda_{\max}^2}{(\lambda_{\max} + \sigma^2)^2} \right] \quad (5.44)$$

In the simplest case where the  $\lambda_{\max}$  is not degenerate the optimal weight configuration for which the maximum (5.44) is achieved is, in terms of the original variables:

$$\mathbf{w}_{\text{opt}}^a = k_a \mathbf{w}_{\lambda_{\max}}, \quad \mathbf{w}_{\text{opt}}^b = k_b \mathbf{U} \mathbf{w}_{\lambda_{\max}} \quad (5.45)$$

with  $k_a$  and  $k_b$  arbitrary constants, and where we may choose  $|\mathbf{w}_{\lambda_{\max}}| = 1$ . Insertion into the original operation rules of the two neurons, and using  $\mathbf{x}^b = \mathbf{U} \mathbf{x}^a$ , shows that in this optimal set-up the system apparently operates a rule in which (apart from an overall constant) the signal terms of the two neuron outputs have become identical:

$$y_a(\mathbf{x}^a) = k_a \mathbf{w}_{\lambda_{\max}} \cdot \mathbf{x}^a + k_a \mathbf{w}_{\lambda_{\max}} \cdot \boldsymbol{\xi}^a$$

$$y_b(\mathbf{x}^b) = k_b (\mathbf{U} \mathbf{w}_{\lambda_{\max}}) \cdot \mathbf{x}^b + k_b (\mathbf{U} \mathbf{w}_{\lambda_{\max}}) \cdot \boldsymbol{\xi}^b = k_b \mathbf{w}_{\lambda_{\max}} \cdot \mathbf{x}^a + k_b \mathbf{w}_{\lambda_{\max}} \cdot \mathbf{U}^\dagger \boldsymbol{\xi}^b$$

There is an additional bonus. If we now go back and work out for the present example our previous result (5.25) for the weight arrangements of our two neurons that maximise for each the mutual information between the corrupted and uncorrupted signals, we find exactly the same result (5.45). The weight configuration that maximises the mutual information between  $y_a$  and  $y_b$  apparently at the same time maximises the amount of information about the uncorrupted signals that reach the output! What is nice about this feature is that the latter is a quantity that one can never observe at the output side of the system (since one would need to know the uncorrupted signals), whereas  $\tilde{I}(y_a, y_b)$  only involves the statistics of the output signals and can thus be maximised using available information only (for instance by a simple stochastic search procedure).

## 5.5 The Effect of Non-linearities

Finally we will here give the justification of a previous remark that the our analysis performed for various arrangements involving linear neurons of the type  $y(\mathbf{x}) = \sum_{i=1}^N w_i x_i$ , possibly extended with noise sources, carry over directly to the more general case of neurons with arbitrary (usually non-linear) invertible transfer functions:  $y(\mathbf{x}) = f[\sum_{i=1}^N w_i x_i]$ . The reason is that the differential mutual information, on which our analysis was built, is not sensitive to invertible transformations:

**property:** If two random variables  $u \in A \subseteq \mathfrak{R}$  and  $y \in \mathfrak{R}$  are related by a continuously differentiable and invertible transformation  $f : \mathfrak{R} \rightarrow A$ , i.e.  $u = f(y)$ , and  $z$  is an arbitrary third random variable, then

$$\tilde{I}(U, Z) = \tilde{I}(Y, Z) \quad (5.46)$$

**proof:** First we construct the joint distribution  $p(u, z)$  from the original distribution  $p(y, z)$ , using the  $\delta$ -distribution (see appendix C), and the marginal distribution  $p(u)$ :

$$\begin{aligned} p(u, z) &= \int dy p(u|y, z)p(y, z) = \int dy \delta[u - f(y)]p(y, z) \\ p(u) &= \int dz p(u, z) = \int dy \delta[u - f(y)] \int dz p(y, z) = \int dy \delta[u - f(y)]p(y) \end{aligned}$$

The differential mutual information for the pair  $(u, z)$  then becomes

$$\begin{aligned} \tilde{I}(U, Z) &= \int dudz p(u, z) {}^2\log \left[ \frac{p(u, z)}{p(u)p(z)} \right] \\ &= \int dudydz p(y, z) \delta[u - f(y)] {}^2\log \left[ \frac{\int dy' \delta[u - f(y')]p(y', z)}{p(z) \int dy' \delta[u - f(y')]p(y')} \right] \\ &= \int dydz p(y, z) {}^2\log \left[ \frac{\int dy' \delta[f(y) - f(y')]p(y', z)}{p(z) \int dy' \delta[f(y) - f(y')]p(y')} \right] \end{aligned}$$

Now we use identity (C.5) of appendix C, which gives

$$\begin{aligned} \tilde{I}(U, Z) &= \int dydz p(y, z) {}^2\log \left[ \frac{\int dy' \delta[y - y']p(y', z)}{p(z) \int dy' \delta[y - y']p(y')} \right] \\ &= \int dydz p(y, z) {}^2\log \left[ \frac{p(y, z)}{p(z)p(y)} \right] = \tilde{I}(Y, Z) \end{aligned}$$

which completes the proof.  $\square$

Application of this result to both random variables under consideration allows us to write more generally:

$$F, G \text{ invertible : } \quad \tilde{I}(F(Y), G(Z)) = \tilde{I}(Y, Z) \quad (5.47)$$

Similarly we can prove that such statements are true for random variables which are themselves vectors rather than scalars. This shows that all of our previous statements on linear neurons apply also to neurons with arbitrary non-linear (but invertible) transfer functions.

## 5.6 Introduction to Amari's Information Geometry

In this final section we will briefly touch the area of information geometry and its applications to information processing in neural networks. We will be considering the general scenario of a neural network whose operation is parametrised by a vector  $\boldsymbol{\theta} \in \mathfrak{R}^L$  (representing weights and/or thresholds), and whose input/output characteristics are described by a conditional probability distribution  $p_{\boldsymbol{\theta}}(\mathbf{x}_{\text{out}}|\mathbf{x}_{\text{in}})$ , in which  $\mathbf{x}_{\text{in}} \in \mathfrak{R}^N$  and  $\mathbf{x}_{\text{out}} \in \mathfrak{R}^M$  denote input and output vectors, respectively. The performance of this network on a given input  $\mathbf{x}_{\text{in}}$ , leading to a corresponding network response  $\mathbf{x}_{\text{out}}$ , is measured by some error function  $\mathcal{E}(\mathbf{x}_{\text{in}}, \mathbf{x}_{\text{out}})$ . If the probability of an input  $\mathbf{x}_{\text{in}}$  to be encountered is defined as  $p(\mathbf{x}_{\text{in}})$ , the global error made by a network with parameters  $\boldsymbol{\theta}$  is given by

$$E(\boldsymbol{\theta}) = \sum_{\mathbf{x}_{\text{in}}} \sum_{\mathbf{x}_{\text{out}}} \mathcal{E}(\mathbf{x}_{\text{in}}, \mathbf{x}_{\text{out}}) p_{\boldsymbol{\theta}}(\mathbf{x}_{\text{out}}|\mathbf{x}_{\text{in}}) p(\mathbf{x}_{\text{in}})$$

We write  $\mathbf{x} = (\mathbf{x}_{\text{in}}, \mathbf{x}_{\text{out}}) \in A = \mathfrak{R}^{N+M}$  and define  $p_{\boldsymbol{\theta}}(\mathbf{x}) = p_{\boldsymbol{\theta}}(\mathbf{x}_{\text{out}}|\mathbf{x}_{\text{in}}) p(\mathbf{x}_{\text{in}})$  (which now combines both the parametrised properties of the network and the likelihood of input data):

$$E(\boldsymbol{\theta}) = \sum_{\mathbf{x} \in A} p_{\boldsymbol{\theta}}(\mathbf{x}) \mathcal{E}(\mathbf{x}) \quad (5.48)$$

This set-up is as yet sufficiently general to cover the majority of neural learning scenarios (although not all). The goal of learning is to find an efficient iterative recipe for modifying the parameters  $\boldsymbol{\theta}$  in order to minimize  $E(\boldsymbol{\theta})$  as quickly as possible.

### 5.6.1 Gradient Descent

*Drawbacks of Ordinary Gradient Descent.* The most commonly used dynamical rule for the parameters  $\boldsymbol{\theta}$  to systematically minimize the error in (5.48) is the ‘gradient descent’ procedure:

$$\frac{d}{dt} \boldsymbol{\theta} = -\eta \nabla_{\boldsymbol{\theta}} E(\boldsymbol{\theta})$$

In spite of the fact that this appears to be a sensible choice (choosing the ‘steepest direction’ in order to get to the ‘nearest valley’), and that it ensures

$$\frac{d}{dt} E = \nabla_{\boldsymbol{\theta}} E(\boldsymbol{\theta}) \cdot \frac{d}{dt} \boldsymbol{\theta} = -\eta [\nabla_{\boldsymbol{\theta}} E(\boldsymbol{\theta})]^2 \leq 0 \quad (5.49)$$

it is in fact generally far from optimal, and the source of the plateau phases that haunt learning rules such as error back-propagation.

Note, firstly, that one can insert an arbitrary positive definite (even parameter dependent)  $L \times L$  matrix  $\mathbf{M}(\boldsymbol{\theta})$  in front of the gradient, without loss of the desirable properties of the gradient descent rule:

$$\frac{d}{dt} \boldsymbol{\theta} = -\eta \mathbf{M}(\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} E(\boldsymbol{\theta}) \quad \frac{d}{dt} E = -\eta [\nabla_{\boldsymbol{\theta}} E(\boldsymbol{\theta}) \cdot \mathbf{M}(\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} E(\boldsymbol{\theta})] \leq 0 \quad (5.50)$$

Gradient descent just corresponds to the simplest choice  $\mathbf{M}(\boldsymbol{\theta}) = \mathbf{I}$ . Furthermore, one can easily convince oneself by experimentation with such matrices that  $\mathbf{M}(\boldsymbol{\theta}) = \mathbf{I}$  is usually not optimal. For example: in error back-propagation one finds that using different learning rates

for different network layers, dependent on the fan-in of these layers, speeds up convergence; this is an example of inserting a simple (diagonal) type of positive definite matrix. We are now automatically led to the question: which is the optimal choice for  $\mathbf{M}(\boldsymbol{\theta})$  ?

We secondly have to realize that the way in which we choose to parametrize our network is rather arbitrary. For example, instead of a parameter component  $\theta_i \in \mathfrak{R}$  one could have also inserted  $\theta_i^5$ , or any other monotonic function of  $\theta_i$ , without affecting the range of possible operations  $p_{\boldsymbol{\theta}}(\mathbf{x})$  the network could perform. More generally, the space of possible operations of the form  $p_{\boldsymbol{\theta}}(\mathbf{x})$  is invariant under arbitrary invertible transformations  $f : \mathfrak{R}^L \rightarrow \mathfrak{R}^L$  of the parameter vector  $\boldsymbol{\theta}$ ; there is no 'preferred' choice for  $f$ . Yet, the gradient descent learning rule is highly sensitive to such parameter transformations. Just consider two equivalent choices of parameters,  $\boldsymbol{\theta} \in \mathfrak{R}^L$  and  $\boldsymbol{\xi} \in \mathfrak{R}^L$ , which are related by an invertible transformation:  $\boldsymbol{\theta} = f(\boldsymbol{\xi})$ . A gradient descent dynamics derived in the language of  $\boldsymbol{\xi}$  gives (written explicitly in components):

$$\frac{d}{dt}\xi_i = -\eta \frac{\partial}{\partial \xi_i} E(f(\boldsymbol{\xi})) = -\eta \sum_{j=1}^L \left[ \frac{\partial \theta_j}{\partial \xi_i} \right] \left[ \frac{\partial E(\boldsymbol{\theta})}{\partial \theta_j} \right]$$

If we now work out what this implies for the evolution of the alternative parameters  $\boldsymbol{\theta}$  we find:

$$\frac{d}{dt}\theta_i = \sum_{j=1}^N \left[ \frac{\partial \theta_i}{\partial \xi_j} \right] \frac{d}{dt}\xi_j = -\eta \sum_{jk=1}^L \left[ \frac{\partial \theta_i}{\partial \xi_j} \right] \left[ \frac{\partial \theta_k}{\partial \xi_j} \right] \frac{\partial E(\boldsymbol{\theta})}{\partial \theta_k}$$

i.e.

$$\frac{d}{dt}\boldsymbol{\theta} = -\eta \mathbf{M}(\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} E(\boldsymbol{\theta}) \quad M_{ik}(\boldsymbol{\theta}) = \sum_j \left[ \frac{\partial \theta_i}{\partial \xi_j} \right] \left[ \frac{\partial \theta_k}{\partial \xi_j} \right]$$

This matrix is positive definite, so still  $\frac{d}{dt}E \leq 0$ . However, we recover a gradient descent equation in terms of the parameters  $\boldsymbol{\theta}$  only if  $M_{ik}(\boldsymbol{\theta}) = \delta_{ik}$ , which in general will not be true. We conclude that the error evolution we obtain is highly dependent on how we choose to parametrize our network in the first place. We are now automatically led to the question: which is the optimal choice of (alternative) parametrization ?

*Derivation of Gradient Descent.* In order to understand better the properties of the gradient descent rule, with a view to ultimately replacing it, let us first see how it can be derived from an extremization procedure. We try to extremize  $E(\boldsymbol{\theta} + \Delta\boldsymbol{\theta})$  by variation of  $\Delta\boldsymbol{\theta}$ , subject to the constraint that the magnitude of the change  $\boldsymbol{\theta}$  is fixed:  $|\Delta\boldsymbol{\theta}| = \Delta$ . The solution (extremisation of a given function under a constraint) is easily obtained with the method of Lagrange:

$$\frac{\partial E(\boldsymbol{\theta} + \Delta\boldsymbol{\theta})}{\partial (\Delta\boldsymbol{\theta})_i} = \lambda \frac{\partial}{\partial (\Delta\boldsymbol{\theta})_i} (\Delta\boldsymbol{\theta})^2 \quad (\Delta\boldsymbol{\theta})^2 = \Delta^2$$

We can eliminate  $\lambda$  by taking the inner product with  $\Delta\boldsymbol{\theta}$  in both sides of the above equation, giving  $\lambda = \frac{1}{2}\Delta^{-2}[\Delta\boldsymbol{\theta} \cdot \nabla E(\boldsymbol{\theta})] + \mathcal{O}(\Delta^0)$ . For small changes, i.e.  $\Delta\boldsymbol{\theta} = dt(d\boldsymbol{\theta}/dt) + \mathcal{O}(dt^2)$  with  $0 < dt \ll 1$ , we then find

$$\nabla E(\boldsymbol{\theta}) = \left[ \frac{(d\boldsymbol{\theta}/dt) \cdot \nabla E(\boldsymbol{\theta})}{(d\boldsymbol{\theta}/dt)^2} \right] \frac{d}{dt}\boldsymbol{\theta} + \mathcal{O}(dt)$$

Taking the limit  $dt \rightarrow 0$  finally gives

$$\frac{d}{dt}\boldsymbol{\theta} = -\eta \nabla E(\boldsymbol{\theta}) \quad \eta = \frac{(d\boldsymbol{\theta}/dt)^2}{(d\boldsymbol{\theta}/dt) \cdot [-\nabla E(\boldsymbol{\theta})]}$$

Note that the right equation can be dropped, as it simply follows from the left one by taking the inner product with  $\frac{d}{dt}\boldsymbol{\theta}$ . We have thereby recovered the ordinary gradient descent rule.

### 5.6.2 Metrics in Parameter Space

*Alternative Distance Measures.* Let us now inspect the effect of introducing an alternative distance measure  $D[\boldsymbol{\theta}, \boldsymbol{\theta}']$  in parameter space, not necessarily given by the Euclidean recipe  $D^2[\boldsymbol{\theta}, \boldsymbol{\theta}'] = |\boldsymbol{\theta} - \boldsymbol{\theta}'|^2 = \sum_i (\theta_i - \theta'_i)^2$ . We obviously require

$$D[\boldsymbol{\theta}, \boldsymbol{\theta}'] \geq 0, \quad D[\boldsymbol{\theta}, \boldsymbol{\theta}] = 0, \quad D[\boldsymbol{\theta}, \boldsymbol{\theta}'] = D[\boldsymbol{\theta}', \boldsymbol{\theta}] \quad \text{for all } \boldsymbol{\theta}, \boldsymbol{\theta}' \quad (5.51)$$

as well as the triangular inequality

$$D[\boldsymbol{\theta}, \boldsymbol{\theta}'] + D[\boldsymbol{\theta}', \boldsymbol{\theta}'''] \geq D[\boldsymbol{\theta}, \boldsymbol{\theta}'''] \quad \text{for all } \boldsymbol{\theta}, \boldsymbol{\theta}', \boldsymbol{\theta}''' \quad (5.52)$$

Since we are mostly interested in small changes in the parameter vector, we can at first consider  $\boldsymbol{\theta}' = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$  with  $|\Delta\boldsymbol{\theta}| \ll 1$ . If our distance measure is well-behaved we can expand

$$D^2[\boldsymbol{\theta}, \boldsymbol{\theta} + \Delta\boldsymbol{\theta}] = \sum_i \Delta\theta_i \frac{\partial D^2[\boldsymbol{\theta}, \boldsymbol{\theta}']}{\partial \theta'_i} \Big|_{\boldsymbol{\theta}'=\boldsymbol{\theta}} + \frac{1}{2} \sum_{ij} \Delta\theta_i \Delta\theta_j \frac{\partial^2 D^2[\boldsymbol{\theta}, \boldsymbol{\theta}']}{\partial \theta'_i \partial \theta'_j} \Big|_{\boldsymbol{\theta}'=\boldsymbol{\theta}} + \mathcal{O}(|\Delta\boldsymbol{\theta}|^3)$$

The zero-th order term is absent due to  $D[\boldsymbol{\theta}, \boldsymbol{\theta}] = 0$ . In view of (5.51) we know that the term linear in  $\Delta\boldsymbol{\theta}$  must also be zero, since otherwise we could always violate  $D[\boldsymbol{\theta}, \boldsymbol{\theta}'] \geq 0$  by choosing  $\Delta\theta_i = -\epsilon \frac{\partial D^2[\boldsymbol{\theta}, \boldsymbol{\theta}']}{\partial \theta'_i} \Big|_{\boldsymbol{\theta}'=\boldsymbol{\theta}}$  with  $\epsilon$  sufficiently small. Thus any well behaved distance measure is locally of the form

$$D^2[\boldsymbol{\theta}, \boldsymbol{\theta}'] = \sum_{ij} (\theta_i - \theta'_i) g_{ij}(\boldsymbol{\theta}) (\theta_j - \theta'_j) + \mathcal{O}(|\boldsymbol{\theta} - \boldsymbol{\theta}'|^3) \quad (5.53)$$

in which the  $L \times L$  matrix  $\mathbf{g}(\boldsymbol{\theta})$  is symmetric, i.e.  $g_{ij}(\boldsymbol{\theta}) = g_{ji}(\boldsymbol{\theta})$ , and positive definite due to the non-negativity of  $D$  (since in the case of negative eigenvalues we could choose our  $\Delta\boldsymbol{\theta}$  proportional to the corresponding eigenvector and violate  $D \geq 0$ ). The Euclidean metric is just the simplest case  $g_{ij}(\boldsymbol{\theta}) = \delta_{ij}$  for all  $\boldsymbol{\theta}$ . Note that in (5.53) we could equally well put  $g_{ij}(\boldsymbol{\theta}) \rightarrow g_{ij}(\boldsymbol{\theta}')$ , since this would only generate (irrelevant) higher order terms.

We now show that any metric of the form (5.53), with positive definite  $\mathbf{g}(\boldsymbol{\theta})$ , satisfies the triangular inequality. For any trio of vectors  $\{\boldsymbol{\theta}, \boldsymbol{\theta}', \boldsymbol{\theta}''\}$ , with  $\boldsymbol{\theta} - \boldsymbol{\theta}' = \epsilon\mathbf{v}$  and  $\boldsymbol{\theta}' - \boldsymbol{\theta}'' = \epsilon\mathbf{w}$  and  $0 < \epsilon \ll 1$ , we obtain

$$\begin{aligned} & \{D[\boldsymbol{\theta}, \boldsymbol{\theta}'] + D[\boldsymbol{\theta}', \boldsymbol{\theta}''']\}^2 - D^2[\boldsymbol{\theta}, \boldsymbol{\theta}'''] = D^2[\boldsymbol{\theta}, \boldsymbol{\theta}'] + D^2[\boldsymbol{\theta}', \boldsymbol{\theta}'''] + 2D[\boldsymbol{\theta}, \boldsymbol{\theta}']D[\boldsymbol{\theta}', \boldsymbol{\theta}'''] - D^2[\boldsymbol{\theta}, \boldsymbol{\theta}'''] \\ & = \epsilon^2 \sum_{ij} g_{ij}(\boldsymbol{\theta}) [v_i v_j + w_i w_j - (v_i + w_i)(v_j + w_j)] + 2\epsilon^2 \left[ \sum_{ij} v_i g_{ij}(\boldsymbol{\theta}) v_j \right]^{\frac{1}{2}} \left[ \sum_{ij} w_i g_{ij}(\boldsymbol{\theta}) w_j \right]^{\frac{1}{2}} + \mathcal{O}(\epsilon^3) \\ & = 2\epsilon^2 \left[ \sum_{ij} v_i g_{ij}(\boldsymbol{\theta}) v_j \right]^{\frac{1}{2}} \left[ \sum_{ij} w_i g_{ij}(\boldsymbol{\theta}) w_j \right]^{\frac{1}{2}} - 2\epsilon^2 \sum_{ij} v_i g_{ij}(\boldsymbol{\theta}) w_j + \mathcal{O}(\epsilon^3) \end{aligned}$$

(where we have used the symmetry of  $\mathbf{g}(\boldsymbol{\theta})$ ). We next switch to the basis where  $\mathbf{g}(\boldsymbol{\theta})$  is diagonal (the eigenvalues of  $\mathbf{g}(\boldsymbol{\theta})$  are written as  $g_n$ ), whereby  $(\mathbf{v}, \mathbf{w}) \rightarrow (\hat{\mathbf{v}}, \hat{\mathbf{w}})$ :

$$\{D[\boldsymbol{\theta}, \boldsymbol{\theta}'] + D[\boldsymbol{\theta}', \boldsymbol{\theta}'']\}^2 - D^2[\boldsymbol{\theta}, \boldsymbol{\theta}''] = 2\epsilon^2 \left[ \sum_n g_n \hat{v}_n^2 \right]^{\frac{1}{2}} \left[ \sum_n g_n \hat{w}_n^2 \right]^{\frac{1}{2}} - 2\epsilon^2 \sum_n g_n \hat{v}_n \hat{w}_n + \mathcal{O}(\epsilon^3)$$

Note that the eigenvalues  $\{g_n\}$  and the vectors  $\hat{\mathbf{v}}$  and  $\hat{\mathbf{w}}$  will depend on  $\boldsymbol{\theta}$ , due to the dependence of  $\mathbf{g}(\boldsymbol{\theta})$  on  $\boldsymbol{\theta}$ . Finally we define the new vectors  $\mathbf{x}$  and  $\mathbf{y}$  (which are again  $\boldsymbol{\theta}$ -dependent), with components  $x_n = \hat{v}_n \sqrt{g_n}$  and  $y_n = \hat{w}_n \sqrt{g_n}$ :

$$\{D[\boldsymbol{\theta}, \boldsymbol{\theta}'] + D[\boldsymbol{\theta}', \boldsymbol{\theta}'']\}^2 - D^2[\boldsymbol{\theta}, \boldsymbol{\theta}''] = 2\epsilon^2 \{|\mathbf{x}||\mathbf{y}| - \mathbf{x} \cdot \mathbf{y}\} + \mathcal{O}(\epsilon^3)$$

which, due to the Schwarz inequality  $|\mathbf{x} \cdot \mathbf{y}| \leq |\mathbf{x}||\mathbf{y}|$ , completes the proof that locally the triangular inequality  $D[\boldsymbol{\theta}, \boldsymbol{\theta}'] + D[\boldsymbol{\theta}', \boldsymbol{\theta}''] \geq D[\boldsymbol{\theta}, \boldsymbol{\theta}'']$  indeed holds.

Given the local metric (5.53), one obtains the length  $A$  of a path  $\boldsymbol{\theta}(t)$  through parameter space, with  $t_0 \leq t \leq t_1$ , simply by integrating over the locally defined distance:

$$A = \int_{t_0}^{t_1} dt L \left[ \boldsymbol{\theta}(t), \frac{d}{dt} \boldsymbol{\theta}(t) \right] \quad L \left[ \boldsymbol{\theta}(t), \frac{d}{dt} \boldsymbol{\theta}(t) \right] = \left\{ \frac{d}{dt} \boldsymbol{\theta}(t) \cdot \mathbf{g}(\boldsymbol{\theta}(t)) \frac{d}{dt} \boldsymbol{\theta}(t) \right\}^{\frac{1}{2}} \quad (5.54)$$

Any finite distance  $D[\boldsymbol{\theta}, \boldsymbol{\theta}']$  is now defined as the length  $A$  of the *shortest* path  $\boldsymbol{\theta}(t)$  with  $\boldsymbol{\theta}(t_0) = \boldsymbol{\theta}$  and  $\boldsymbol{\theta}(t_1) = \boldsymbol{\theta}'$ . This (special) path, which is one of the so-called 'geodesics', is calculated by extremisation of the expression in (5.54) by functional variation of the path  $\boldsymbol{\theta}(t)$ , subject to the constraints that  $\delta\boldsymbol{\theta}(t_0) = \delta\boldsymbol{\theta}(t_1) = 0$ :

$$\begin{aligned} \delta A &= \sum_i \int_{t_0}^{t_1} dt \left\{ \delta\theta_i(t) \frac{\delta L}{\delta\theta_i(t)} + \frac{d}{dt} \delta\theta_i(t) \frac{\delta L}{\delta(d\theta_i(t)/dt)} \right\} \\ &= \sum_i \int_{t_0}^{t_1} dt \delta\theta_i(t) \frac{\delta L}{\delta\theta_i(t)} + \sum_i \left[ \delta\theta_i(t) \frac{\delta L}{\delta(d\theta_i(t)/dt)} \right]_{t_0}^{t_1} - \sum_i \int_{t_0}^{t_1} dt \delta\theta_i(t) \frac{d}{dt} \frac{\delta L}{\delta(d\theta_i(t)/dt)} \\ &= \sum_i \int_{t_0}^{t_1} dt \delta\theta_i(t) \left\{ \frac{\delta L}{\delta\theta_i(t)} - \frac{d}{dt} \left[ \frac{\delta L}{\delta(d\theta_i(t)/dt)} \right] \right\} \end{aligned}$$

Therefore, the extremal path is a solution of the equation

$$\frac{\delta L}{\delta\theta_i(t)} = \frac{d}{dt} \left[ \frac{\delta L}{\delta(d\theta_i(t)/dt)} \right] \quad (5.55)$$

with the functional  $L$  as given in (5.54). It is a trivial exercise to show that in the case of Euclidean geometry,  $g_{ij}(\boldsymbol{\theta}) = \delta_{ij}$ , one finds the shortest path always to be the Euclidean straight line  $\boldsymbol{\theta}(t) = \frac{1}{t_1 - t_0} [(t_1 - t)\boldsymbol{\theta}(t_0) + (t - t_0)\boldsymbol{\theta}(t_1)]$ .

*Derivation of Natural Gradient Descent.* We now again try to extremize  $E(\boldsymbol{\theta} + \Delta\boldsymbol{\theta})$  by variation of  $\Delta\boldsymbol{\theta}$ , subject to the constraint that the magnitude of the change  $\boldsymbol{\theta}$  is fixed. However, now the magnitude of the the change is calculated with the general metric (5.53), rather than the Euclidean one, i.e.  $\sum_{ij} \Delta\theta_i g_{ij}(\boldsymbol{\theta}) \Delta\theta_j = \Delta^2$  (with  $0 \leq \Delta \ll 1$ ). The solution is again obtained with the method of Lagrange:

$$\frac{\partial E(\boldsymbol{\theta} + \Delta\boldsymbol{\theta})}{\partial(\Delta\boldsymbol{\theta})_i} = \lambda \frac{\partial}{\partial(\Delta\boldsymbol{\theta})_i} D^2[\boldsymbol{\theta} + \Delta\boldsymbol{\theta}, \boldsymbol{\theta}] \quad D^2[\boldsymbol{\theta} + \Delta\boldsymbol{\theta}, \boldsymbol{\theta}] = \Delta^2$$

$$\frac{\partial E(\boldsymbol{\theta} + \Delta\boldsymbol{\theta})}{\partial \theta_i} = 2\lambda \left\{ \sum_j g_{ij}(\boldsymbol{\theta}) \Delta\theta_j + \mathcal{O}(\Delta^2) \right\} \quad D^2[\boldsymbol{\theta} + \Delta\boldsymbol{\theta}, \boldsymbol{\theta}] = \Delta^2$$

As before we eliminate  $\lambda$  by taking the inner product with  $\Delta\boldsymbol{\theta}$  in both sides of the above equation, giving  $\lambda = \frac{1}{2}\Delta^{-2}[\Delta\boldsymbol{\theta} \cdot \nabla E(\boldsymbol{\theta})] + \mathcal{O}(\Delta^0)$ . We next consider small changes only, i.e. we write  $\Delta\boldsymbol{\theta} = dt(d\boldsymbol{\theta}/dt) + \mathcal{O}(dt^2)$  with  $0 < dt \ll 1$ . Consequently the constraint equation becomes  $\Delta^2 = dt^2[(d\boldsymbol{\theta}/dt) \cdot \mathbf{g}(\boldsymbol{\theta})(d\boldsymbol{\theta}/dt)] + \mathcal{O}(dt^3)$ , and

$$\nabla E(\boldsymbol{\theta}) = \left[ \frac{(d\boldsymbol{\theta}/dt) \cdot \nabla E(\boldsymbol{\theta})}{(d\boldsymbol{\theta}/dt) \cdot \mathbf{g}(\boldsymbol{\theta})(d\boldsymbol{\theta}/dt)} \right] \mathbf{g}(\boldsymbol{\theta}) \frac{d}{dt} \boldsymbol{\theta} + \mathcal{O}(dt)$$

Taking the limit  $dt \rightarrow 0$  produces

$$\frac{d}{dt} \boldsymbol{\theta} = -\eta \mathbf{g}^{-1}(\boldsymbol{\theta}) \nabla E(\boldsymbol{\theta}) \quad \eta = \frac{(d\boldsymbol{\theta}/dt) \cdot \mathbf{g}(\boldsymbol{\theta})(d\boldsymbol{\theta}/dt)}{(d\boldsymbol{\theta}/dt) \cdot [-\nabla E(\boldsymbol{\theta})]}$$

The right equation can be dropped, as it follows from the left one by taking the inner product with  $\mathbf{g}(\boldsymbol{\theta}) \frac{d}{dt} \boldsymbol{\theta}$ . We have thereby derived the so-called *natural* gradient descent rule:

$$\frac{d}{dt} \boldsymbol{\theta} = -\eta \mathbf{g}^{-1}(\boldsymbol{\theta}) \nabla E(\boldsymbol{\theta}) \quad (5.56)$$

Only in the case where our metric is Euclidean, i.e.  $\mathbf{g}(\boldsymbol{\theta}) = \mathbf{I}$ , will natural gradient descent be identical to ordinary gradient descent. Note that we can now also interpret the meaning of inserting a positive definite matrix into the ordinary gradient descent rule, as in (5.50), and identify the optimal choice for such a matrix according to (5.56): the optimal choice is to choose  $\mathbf{M}(\boldsymbol{\theta})$  in (5.50) as the inverse of the metric  $\mathbf{g}(\boldsymbol{\theta})$ . It thus depends crucially on the choice we make for the distance  $D[\boldsymbol{\theta}, \boldsymbol{\theta}']$  in parameter space.

### 5.6.3 The Proper Metric

What remains is to choose the appropriate metric in parameter space. Our aim is to base this choice on the premise that the natural distance between two parameter vectors  $\boldsymbol{\theta}$  and  $\boldsymbol{\theta}'$  should be determined by to what extent the corresponding networks, encoded in the two distributions  $p_{\boldsymbol{\theta}}(\mathbf{x})$  and  $p_{\boldsymbol{\theta}'}(\mathbf{x})$ , are similar. The more different these two distributions, the larger should be the natural distance between  $\boldsymbol{\theta}$  and  $\boldsymbol{\theta}'$ .

*Expansion of Kullback-Leibler Distance.* One object that measures the distance between two probability distributions is the Kullback-Leibler distance:

$$D(p_{\boldsymbol{\theta}} || p_{\boldsymbol{\theta}'}) = \sum_{\mathbf{x} \in A} p_{\boldsymbol{\theta}}(\mathbf{x}) \log \left[ \frac{p_{\boldsymbol{\theta}}(\mathbf{x})}{p_{\boldsymbol{\theta}'}(\mathbf{x})} \right] \quad (5.57)$$

(where we have replaced  ${}^2\log \rightarrow \log$  in order to eliminate the irrelevant prefactor  $\log 2$  which would otherwise be generated in subsequent calculations). This expression can itself not serve as our distance  $D[\boldsymbol{\theta}, \boldsymbol{\theta}']$ , since it violates  $D[\boldsymbol{\theta}, \boldsymbol{\theta}'] = D[\boldsymbol{\theta}', \boldsymbol{\theta}]$ . However, locally it turns out to give a well-behaved distance measure in the sense of (5.53). To see this we put  $\boldsymbol{\theta}' = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$ , with  $|\Delta\boldsymbol{\theta}| \ll 1$ , and expand (5.57) in powers of  $\Delta\boldsymbol{\theta}$ :

$$D(p_{\boldsymbol{\theta}} || p_{\boldsymbol{\theta} + \Delta\boldsymbol{\theta}}) = \sum_{\mathbf{x} \in A} p_{\boldsymbol{\theta}}(\mathbf{x}) \left\{ \log p_{\boldsymbol{\theta}}(\mathbf{x}) - \log p_{\boldsymbol{\theta} + \Delta\boldsymbol{\theta}}(\mathbf{x}) \right\}$$

$$\begin{aligned}
&= \sum_{\mathbf{x} \in A} p_{\boldsymbol{\theta}}(\mathbf{x}) \left\{ \log p_{\boldsymbol{\theta}}(\mathbf{x}) - \log \left[ p_{\boldsymbol{\theta}}(\mathbf{x}) + \sum_i \Delta_i \boldsymbol{\theta} \frac{\partial p_{\boldsymbol{\theta}}(\mathbf{x})}{\partial \theta_i} + \frac{1}{2} \sum_{ij} \Delta_i \boldsymbol{\theta} \Delta_j \boldsymbol{\theta} \frac{\partial^2 p_{\boldsymbol{\theta}}(\mathbf{x})}{\partial \theta_i \partial \theta_j} + \dots \right] \right\} \\
&= - \sum_{\mathbf{x} \in A} p_{\boldsymbol{\theta}}(\mathbf{x}) \log \left[ 1 + \sum_i \frac{\Delta_i \boldsymbol{\theta}}{p_{\boldsymbol{\theta}}(\mathbf{x})} \frac{\partial}{\partial \theta_i} p_{\boldsymbol{\theta}}(\mathbf{x}) + \frac{1}{2} \sum_{ij} \frac{\Delta_i \boldsymbol{\theta} \Delta_j \boldsymbol{\theta}}{p_{\boldsymbol{\theta}}(\mathbf{x})} \frac{\partial^2}{\partial \theta_i \partial \theta_j} p_{\boldsymbol{\theta}}(\mathbf{x}) + \dots \right] \\
&= - \left[ \sum_i \Delta_i \boldsymbol{\theta} \frac{\partial}{\partial \theta_i} + \frac{1}{2} \sum_{ij} \Delta_i \boldsymbol{\theta} \Delta_j \boldsymbol{\theta} \frac{\partial^2}{\partial \theta_i \partial \theta_j} \right] \sum_{\mathbf{x} \in A} p_{\boldsymbol{\theta}}(\mathbf{x}) \\
&\quad + \frac{1}{2} \sum_{\mathbf{x} \in A} p_{\boldsymbol{\theta}}(\mathbf{x}) \sum_{ij} \Delta_i \boldsymbol{\theta} \Delta_j \boldsymbol{\theta} \left[ \frac{\partial \log p_{\boldsymbol{\theta}}(\mathbf{x})}{\partial \theta_i} \right] \left[ \frac{\partial \log p_{\boldsymbol{\theta}}(\mathbf{x})}{\partial \theta_j} \right] + \mathcal{O}(|\Delta \boldsymbol{\theta}|^3) \\
&= \sum_{ij} \Delta_i \boldsymbol{\theta} g_{ij}(\boldsymbol{\theta}) \Delta_j \boldsymbol{\theta} + \mathcal{O}(|\Delta \boldsymbol{\theta}|^3)
\end{aligned}$$

with the metric

$$g_{ij}(\boldsymbol{\theta}) = \sum_{\mathbf{x} \in A} p_{\boldsymbol{\theta}}(\mathbf{x}) \left[ \frac{\partial \log p_{\boldsymbol{\theta}}(\mathbf{x})}{\partial \theta_i} \right] \left[ \frac{\partial \log p_{\boldsymbol{\theta}}(\mathbf{x})}{\partial \theta_j} \right] \quad (5.58)$$

The matrix in (5.58) is called the Fisher Information matrix. It plays an important role in measuring the average amount of information that can be extracted from a single observation  $\mathbf{x}$  about the values of the parameters  $\boldsymbol{\theta}$  of a parametrized distribution  $p_{\boldsymbol{\theta}}(\mathbf{x})$ . The metric (5.58) satisfies all our requirements. It generates a general definition of a distance  $D[\boldsymbol{\theta}, \boldsymbol{\theta}']$  via the route described in the previous sub-section (based on the shortest path connecting  $\boldsymbol{\theta}$  and  $\boldsymbol{\theta}'$ , given by a geodesic).

*Invariance of Natural Gradient Descent Under Re-Parametrization.* Due to the fact that the distance  $D[\boldsymbol{\theta}, \boldsymbol{\theta}']$  generated by the metric (5.58) is based on measuring the mismatch between two probability distributions (rather than on properties of the underlying parametrization), one finds, in contrast to the ordinary gradient descent rule, that the corresponding natural gradient descent equation is *invariant* under parametrization changes. The proof of this important statement is elementary. Just consider two equivalent choices of parameters,  $\boldsymbol{\theta} \in \mathfrak{R}^L$  and  $\boldsymbol{\xi} \in \mathfrak{R}^L$ , which are related by an invertible transformation  $f: \mathfrak{R}^L \rightarrow \mathfrak{R}^L$ :

$$\boldsymbol{\theta} = f(\boldsymbol{\xi}) : \quad p_{\boldsymbol{\xi}}(\mathbf{x}) = \hat{p}_{\boldsymbol{\theta}}(\mathbf{x}), \quad E(\boldsymbol{\xi}) = \hat{E}(\boldsymbol{\theta}), \quad g_{ij}(\boldsymbol{\xi}) = \hat{g}_{ij}(\boldsymbol{\theta})$$

The metric  $g_{ij}(\boldsymbol{\xi})$  (5.58), derived in the language of  $\boldsymbol{\xi}$ , can be related to the metric  $\hat{g}_{ij}(\boldsymbol{\theta})$ , derived in the language of  $\boldsymbol{\theta}$ , in the following way:

$$\begin{aligned}
g_{ij}(\boldsymbol{\xi}) &= \sum_{\mathbf{x} \in A} p_{\boldsymbol{\xi}}(\mathbf{x}) \left[ \frac{\partial \log p_{\boldsymbol{\xi}}(\mathbf{x})}{\partial \xi_i} \right] \left[ \frac{\partial \log p_{\boldsymbol{\xi}}(\mathbf{x})}{\partial \xi_j} \right] \\
&= \sum_{\mathbf{x} \in A} \hat{p}_{\boldsymbol{\theta}}(\mathbf{x}) \sum_{kl} \left[ \frac{\partial \theta_k}{\partial \xi_i} \frac{\partial \log \hat{p}_{\boldsymbol{\theta}}(\mathbf{x})}{\partial \theta_k} \right] \left[ \frac{\partial \theta_l}{\partial \xi_j} \frac{\partial \log \hat{p}_{\boldsymbol{\theta}}(\mathbf{x})}{\partial \theta_l} \right] \\
&= \sum_{kl} K_{ij,kl}(\boldsymbol{\theta}) \hat{g}_{kl}(\boldsymbol{\theta}) \quad \text{with} \quad K_{ij,kl}(\boldsymbol{\theta}) = \left[ \frac{\partial \theta_k}{\partial \xi_i} \right] \left[ \frac{\partial \theta_l}{\partial \xi_j} \right]
\end{aligned}$$



We can now use this relation to derive the dynamical equation which the parameters  $\boldsymbol{\theta}$  will obey, given that the evolution of the parameters  $\boldsymbol{\xi}$  is defined by natural gradient descent (5.56) with the metric  $\mathbf{g}(\boldsymbol{\xi})$ :

$$\begin{aligned} \sum_j g_{ij}(\boldsymbol{\xi}) \frac{d}{dt} \xi_j + \eta \frac{\partial}{\partial \xi_i} E(\boldsymbol{\xi}) &= 0 \\ \sum_{jkl} K_{ij,kl}(\boldsymbol{\theta}) \hat{g}_{kl}(\boldsymbol{\theta}) \frac{d}{dt} \xi_j + \eta \sum_k \frac{\partial \theta_k}{\partial \xi_i} \frac{\partial}{\partial \theta_k} \hat{E}(\boldsymbol{\theta}) &= 0 \\ \sum_k \left[ \frac{\partial \theta_k}{\partial \xi_i} \right] \left[ \sum_{jl} \left[ \frac{\partial \theta_l}{\partial \xi_j} \right] \hat{g}_{kl}(\boldsymbol{\theta}) \frac{d}{dt} \xi_j + \eta \frac{\partial}{\partial \theta_k} \hat{E}(\boldsymbol{\theta}) \right] &= 0 \end{aligned}$$

Since the matrix with entries  $J_{ki}(\boldsymbol{\theta}) = \partial \theta_k / \partial \xi_i$  is just the Jacobian of the transformation  $f$ , which is invertible, it immediately follows that

$$\begin{aligned} \sum_l \hat{g}_{kl}(\boldsymbol{\theta}) \sum_j \frac{\partial \theta_l}{\partial \xi_j} \frac{d}{dt} \xi_j + \eta \frac{\partial}{\partial \theta_k} \hat{E}(\boldsymbol{\theta}) &= 0 \\ \sum_l \hat{g}_{kl}(\boldsymbol{\theta}) \frac{d}{dt} \theta_l + \eta \frac{\partial}{\partial \theta_k} \hat{E}(\boldsymbol{\theta}) &= 0 \end{aligned}$$

This is again the natural gradient descent equation, but now expressed in the language of  $\boldsymbol{\theta}$ . Thus, whatever parametrization one chooses for the network, if one uses the natural gradient descent rule with the metric (5.58) one always generates exactly the same dynamics.

It can be rigorously proven that, apart from an irrelevant prefactor, the metric (5.58) is the only proper metric to be used for parametrized probability distributions, although this is not trivial and will not be done in these notes. Here we have only derived intuitive evidence for this statement, by showing that it is locally equivalent to the Kullback-Leibler distance (which indeed measures the mismatch between distributions in a parametrization-independent way), and by showing that the corresponding natural gradient descent dynamics is invariant under re-parametrizations.

In practice, explicit calculation of the inverse of the matrix (5.58) will usually not be possible, and thus exact execution of the natural gradient descent rule will usually be out of the question. However, at least one knows what the optimal rule is, so that improving upon gradient descent can now be done in a systematic way, via approximation of the inverse of (5.58), rather than in an alchemical manner. There are many ways in which the inverse can be approximated. If  $\mathbf{g}(\boldsymbol{\theta})$  is close to the unity matrix  $\mathbf{I}$ , for instance, one could use the expansion  $\mathbf{g}^{-1}(\boldsymbol{\theta}) = \sum_{n \geq 0} [\mathbf{I} - \mathbf{g}(\boldsymbol{\theta})]^n$ . An alternative is to manipulate general identities for symmetric matrices such as

$$g_{ij}^{-1} = \frac{\int d\mathbf{y} y_i y_j e^{-\frac{1}{2} \mathbf{y} \cdot \mathbf{g} \mathbf{y}}}{\int d\mathbf{y} e^{-\frac{1}{2} \mathbf{y} \cdot \mathbf{g} \mathbf{y}}}$$

### 5.6.4 Simple Applications

*Example 1: Single Neuron Without Inputs.* Our first example is the simplest possible scenario of a noisy binary neuron, without inputs or weights, but with a threshold  $\theta$ , i.e.

$$x_{\text{out}} = \text{sgn}[\xi - \theta] \in \{-1, 1\}$$

which is being trained to permanently generate the output  $x_{\text{out}} = 1$  via adaptation of its threshold  $\theta$ . This obviously requires lowering the threshold to  $\theta = -\infty$ . The noise source  $\xi \in \mathfrak{R}$  has probability density  $w(\xi)$ , where  $w(\xi) = w(-\xi)$  for all  $\xi \in \mathfrak{R}$ . We also define  $W(u) = 2 \int_0^u d\xi w(\xi) \in [-1, 1]$ .

We first have to establish contact with the formalism of this chapter. Since there are no inputs we just have  $p_\theta(x) = \text{Prob}(x_{\text{out}} = x)$ . Thus  $p_\theta(1) = \text{Prob}(x_{\text{out}} = 1) = \int_\theta^\infty d\xi w(\xi) = \frac{1}{2}[1 - W(\theta)]$ . The error measure  $\mathcal{E}(x)$  is required to simply signal the occurrence of a forbidden output value  $x = -1$ , so  $\mathcal{E}(x) = \delta_{x,-1}$ . In combination we thus find, with the definition  $E(\theta) = \sum_x p_\theta(x) \mathcal{E}(x)$ :

$$p_\theta(x) = \frac{1}{2}[1 - xW(\theta)] \quad E(\theta) = \frac{1}{2}[1 + W(\theta)] \quad (5.59)$$

Ordinary gradient descent learning would give for this example:

$$\frac{d}{dt}\theta|_{\text{GD}} = -\eta \frac{\partial}{\partial \theta} E(\theta) = -\frac{1}{2}\eta \frac{\partial}{\partial \theta} W(\theta) = -\eta w(\theta)$$

Let us now compare this to the recipe prescribed by natural gradient descent. The present example has only a single adjustable parameter  $\theta$ , so the Fisher matrix (5.58) reduces to a (time-dependent)  $1 \times 1$  matrix  $g(\theta)$ :

$$\begin{aligned} g(\theta) &= \sum_{x=\pm 1} p_\theta(x) \left[ \frac{\partial}{\partial \theta} \log[1 - xW(\theta)] \right]^2 = \sum_{x=\pm 1} p_\theta(x) \frac{4w^2(\theta)}{[1 - xW(\theta)]^2} \\ &= 2w^2(\theta) \left[ \frac{1}{1 - W(\theta)} + \frac{1}{1 + W(\theta)} \right] = \frac{4w^2(\theta)}{1 - W^2(\theta)} \end{aligned}$$

Natural gradient descent thus reduces to:

$$\frac{d}{dt}\theta|_{\text{NGD}} = -\eta \frac{1 - W^2(\theta)}{4w(\theta)}$$

Let us now make a specific choice for the noise distribution in order to push the analysis further:  $w(\xi) = \frac{1}{2}[1 - \tanh^2(\xi)]$ , so  $W(\theta) = \int_0^\theta d\xi [1 - \tanh^2(\xi)] = \tanh(\theta)$ . This results in

$$\frac{d}{dt}\theta|_{\text{GD}} = -\frac{1}{2}\eta [1 - \tanh^2(\theta)] \quad \frac{d}{dt}\theta|_{\text{NGD}} = -\frac{1}{2}\eta$$

It is already clear that the decrease of the threshold  $\theta$  is slower in the case of gradient descent learning compared to natural gradient descent learning, especially as  $|\theta| \rightarrow \infty$ . This becomes even more striking if we translate the laws for the evolution of  $\theta$  into equations involving the error  $E(\theta)$  only, using the simple relation  $E(\theta) = \frac{1}{2}[1 + \tanh \theta]$ :

$$\frac{d}{dt}E|_{\text{GD}} = -4\eta E^2(1 - E)^2 \quad \frac{d}{dt}E|_{\text{NGD}} = -\eta E(1 - E)$$

In both cases the error decreases monotonically to zero. Asymptotically, i.e. for  $t \rightarrow \infty$ , we can thus expand these equations in powers of  $E$ . Both can then be solved by making an ansatz of the form  $E \sim At^{-\gamma}$ . The result is:

$$\begin{aligned} \frac{d}{dt}E|_{\text{GD}} &= -4\eta E^2 + \dots & \text{with solution} & \quad E \sim \frac{1}{4\eta t} \quad (t \rightarrow \infty) \\ \frac{d}{dt}E|_{\text{NGD}} &= -\eta E + \dots & \text{with solution} & \quad E \sim e^{-\eta t} \quad (t \rightarrow \infty) \end{aligned}$$

This clearly illustrates the potential of natural gradient descent.

*Example 2: Single Neuron With Inputs.* Our second example is a noisy binary neuron with non-zero inputs and weights, but without a threshold:

$$x_{\text{out}} = \text{sgn}\left[\sum_{i=1}^L \theta_i x_i^{\text{in}} + \xi\right] \in \{-1, 1\} \quad \mathbf{x}_{\text{in}} = (x_1^{\text{in}}, \dots, x_L^{\text{in}}) \in \{-1, 1\}^L$$

which is being trained to execute the operation  $\sigma : \{-1, 1\}^L \rightarrow \{-1, 1\}$  via adaptation of the synaptic vector  $\boldsymbol{\theta} \in \mathfrak{R}^L$ . The noise source  $\xi \in \mathfrak{R}$  again has probability density  $w(\xi)$ , where  $w(\xi) = w(-\xi)$  for all  $\xi \in \mathfrak{R}$ , and we define as before  $W(u) = 2 \int_0^u d\xi w(\xi) \in [-1, 1]$ . Thus

$$p_{\boldsymbol{\theta}}(x_{\text{out}}|\mathbf{x}_{\text{in}}) = \frac{1}{2}[1 + x_{\text{out}}W(\boldsymbol{\theta} \cdot \mathbf{x}_{\text{in}})] \quad p_{\boldsymbol{\theta}}(\mathbf{x}_{\text{in}}, x_{\text{out}}) = p_{\boldsymbol{\theta}}(x_{\text{out}}|\mathbf{x}_{\text{in}})p(\mathbf{x}_{\text{in}})$$

The error measure  $\mathcal{E}(\mathbf{x}_{\text{in}}, x_{\text{out}})$  is required to signal the occurrence of  $x_{\text{out}} \neq \sigma(\mathbf{x}_{\text{in}})$ , giving  $\mathcal{E}(\mathbf{x}_{\text{in}}, x_{\text{out}}) = \frac{1}{2}[1 - x_{\text{out}}\sigma(\mathbf{x}_{\text{in}})]$ . In combination we thus find for the average (or global) error, with the definition  $E(\boldsymbol{\theta}) = \sum_{\mathbf{x}_{\text{in}}} \sum_{x_{\text{out}}} p_{\boldsymbol{\theta}}(\mathbf{x}_{\text{in}}, x_{\text{out}})\mathcal{E}(\mathbf{x}_{\text{in}}, x_{\text{out}})$ :

$$E(\boldsymbol{\theta}) = \frac{1}{4} \sum_{\mathbf{x}_{\text{in}}} p(\mathbf{x}_{\text{in}}) \sum_{x_{\text{out}}} [1 + x_{\text{out}}W(\boldsymbol{\theta} \cdot \mathbf{x}_{\text{in}})][1 - x_{\text{out}}\sigma(\mathbf{x}_{\text{in}})] = \frac{1}{2} - \frac{1}{2} \sum_{\mathbf{x}_{\text{in}}} p(\mathbf{x}_{\text{in}})\sigma(\mathbf{x}_{\text{in}})W(\boldsymbol{\theta} \cdot \mathbf{x}_{\text{in}}) \quad (5.60)$$

The metric (5.58) can now be calculated:

$$\begin{aligned} g_{ij}(\boldsymbol{\theta}) &= \sum_{\mathbf{x}_{\text{in}}} \sum_{x_{\text{out}}} p(\mathbf{x}_{\text{in}})p_{\boldsymbol{\theta}}(x_{\text{out}}|\mathbf{x}_{\text{in}}) \left[ \frac{\partial \log p_{\boldsymbol{\theta}}(x_{\text{out}}|\mathbf{x}_{\text{in}})}{\partial \theta_i} \right] \left[ \frac{\partial \log p_{\boldsymbol{\theta}}(x_{\text{out}}|\mathbf{x}_{\text{in}})}{\partial \theta_j} \right] \\ &= \sum_{\mathbf{x}_{\text{in}}} \sum_{x_{\text{out}}} \frac{p(\mathbf{x}_{\text{in}})}{p_{\boldsymbol{\theta}}(x_{\text{out}}|\mathbf{x}_{\text{in}})} [x_i^{\text{in}} x_{\text{out}} w(\boldsymbol{\theta} \cdot \mathbf{x}_{\text{in}})] [x_j^{\text{in}} x_{\text{out}} w(\boldsymbol{\theta} \cdot \mathbf{x}_{\text{in}})] \\ &= \sum_{\mathbf{x}_{\text{in}}} p(\mathbf{x}_{\text{in}}) x_i^{\text{in}} x_j^{\text{in}} w^2(\boldsymbol{\theta} \cdot \mathbf{x}_{\text{in}}) \left[ \frac{2}{1 + W(\boldsymbol{\theta} \cdot \mathbf{x}_{\text{in}})} + \frac{2}{1 - W(\boldsymbol{\theta} \cdot \mathbf{x}_{\text{in}})} \right] \\ &= \sum_{\mathbf{x}_{\text{in}}} p(\mathbf{x}_{\text{in}}) x_i^{\text{in}} x_j^{\text{in}} \frac{w^2(\boldsymbol{\theta} \cdot \mathbf{x}_{\text{in}})}{\frac{1}{4}[1 - W^2(\boldsymbol{\theta} \cdot \mathbf{x}_{\text{in}})]} \end{aligned} \quad (5.61)$$

If we now make the same choice for the noise distribution as in the first example,  $w(\xi) = \frac{1}{2}[1 - \tanh^2(\xi)]$  giving  $W(\boldsymbol{\theta}) = \tanh(\boldsymbol{\theta})$ , and in addition choose uniformly distributed input vectors  $\mathbf{x}_{\text{in}} \in \{-1, 1\}^L$ , we find the following transparent expressions:

$$E(\boldsymbol{\theta}) = \frac{1}{2} [1 - \langle \sigma(\mathbf{x}) \tanh(\boldsymbol{\theta} \cdot \mathbf{x}) \rangle] \quad g_{ij}(\boldsymbol{\theta}) = \delta_{ij} - \langle x_i x_j \tanh^2(\boldsymbol{\theta} \cdot \mathbf{x}) \rangle \quad (5.62)$$

with the abbreviation  $\langle f(\mathbf{x}) \rangle = 2^{-L} \sum_{\mathbf{x} \in \{-1,1\}^L} f(\mathbf{x})$ . The two learning procedures, gradient descent and natural gradient descent, would now generate the following learning dynamics:

$$\frac{d}{dt}\theta_i|_{\text{GD}} = \frac{\eta}{2} \langle x_i \sigma(\mathbf{x}) [1 - \tanh^2(\boldsymbol{\theta} \cdot \mathbf{x})] \rangle \quad \frac{d}{dt}\theta_i|_{\text{NGD}} = \frac{\eta}{2} \sum_j g_{ij}^{-1}(\boldsymbol{\theta}) \langle x_j \sigma(\mathbf{x}) [1 - \tanh^2(\boldsymbol{\theta} \cdot \mathbf{x})] \rangle$$

In general one cannot easily invert the matrix in (5.62) analytically. For small values of  $L$  it can, however, be done. Let us work out the case  $L = 2$  (two inputs) as an explicit example. If the task  $\sigma$  is realizable, there exists a vector  $\mathbf{B} \in \mathfrak{R}^2$  such that  $\sigma(\mathbf{x}) = \text{sgn}[B_1 x_1 + B_2 x_2]$ . We now define  $\sigma_{\pm} = \text{sgn}[B_1 \pm B_2] \in \{-1, 1\}$ , as well as  $\theta_{\pm} = \theta_1 \pm \theta_2$ . All this results in:

$$E(\theta_1, \theta_2) = \frac{1}{2} \left[ 1 - \frac{1}{2} \sigma_+ \tanh(\theta_+) - \frac{1}{2} \sigma_- \tanh(\theta_-) \right] \quad (5.63)$$

$$\mathbf{g}(\boldsymbol{\theta}) = \frac{1}{2} [1 - \tanh^2(\theta_+)] \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} + \frac{1}{2} [1 - \tanh^2(\theta_-)] \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \quad (5.64)$$

One can easily convince oneself that the relevant inverse is

$$\mathbf{g}^{-1}(\boldsymbol{\theta}) = \frac{1}{2[1 - \tanh^2(\theta_+)]} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} + \frac{1}{2[1 - \tanh^2(\theta_-)]} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

We now obtain the following learning rules:

$$\begin{aligned} \frac{d}{dt}\boldsymbol{\theta}|_{\text{GD}} &= \frac{1}{4} \eta \sigma_+ [1 - \tanh^2(\theta_+)] \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \frac{1}{4} \eta \sigma_- [1 - \tanh^2(\theta_-)] \begin{pmatrix} 1 \\ -1 \end{pmatrix} \\ \frac{d}{dt}\boldsymbol{\theta}|_{\text{NGD}} &= \frac{1}{4} \eta \sigma_+ \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \frac{1}{4} \eta \sigma_- \begin{pmatrix} 1 \\ -1 \end{pmatrix} \end{aligned}$$

In terms of  $\theta_{\pm}$  these expressions simplify to

$$\frac{d}{dt}\theta_{\pm}|_{\text{GD}} = \frac{1}{2} \eta \sigma_{\pm} [1 - \tanh^2(\theta_{\pm})] \quad \frac{d}{dt}\theta_{\pm}|_{\text{NGD}} = \frac{1}{2} \eta \sigma_{\pm}$$

Clearly the gradient descent rule will show a saturation slowing down as  $|\theta_{\pm}| \rightarrow \infty$ , which is absent (or rather, being compensated for automatically) in the case of natural gradient descent learning. If  $\boldsymbol{\theta}(t=0) = \mathbf{0}$ , we find in both cases  $\theta_{\pm}(t) = \sigma_{\pm} \phi(t)$ , with  $\phi(0) = 0$  and

$$E = \frac{1}{2} [1 - \tanh(\phi)] \quad \frac{d}{dt}\phi|_{\text{GD}} = \frac{1}{2} \eta [1 - \tanh^2(\phi)] \quad \frac{d}{dt}\phi|_{\text{NGD}} = \frac{1}{2} \eta$$

This implies that, as with the previous example, we can now convert our equations directly into equations for the error  $E$ , which turn out to be identical to those of the previous example:

$$\frac{d}{dt}E|_{\text{GD}} = -4\eta E^2(1-E)^2 \quad \frac{d}{dt}E|_{\text{NGD}} = -\eta E(1-E)$$

Thus we find again a significant convergence speed-up, with asymptotically

$$E|_{\text{GD}} \sim \frac{1}{4\eta t} \quad E|_{\text{NGD}} \sim e^{-\eta t} \quad (t \rightarrow \infty)$$



# Appendix A

## Simple Mathematical Tools

Here I just list, without proof, a couple of useful identities for summations, used in working out the various problem examples:

$$\sum_{k=1}^n k = \frac{1}{2}n(n+1) \quad (\text{A.1})$$

$$\sum_{k=1}^n k^2 = \frac{1}{6}n(n+1)(2n+1) \quad (\text{A.2})$$

$$\sum_{k=0}^n z^k = \frac{1-z^{n+1}}{1-z} \quad (z \neq 1) \quad (\text{A.3})$$



# Appendix B

## Gaussian Integrals

In this appendix we derive some properties of symmetric positive definite matrices  $\mathbf{A}$ , and their associated Gaussian integrals in  $\mathfrak{R}^N$ :

$$I = \int d\mathbf{x} f(\mathbf{x}) e^{-\frac{1}{2}\mathbf{x} \cdot \mathbf{A} \mathbf{x}}$$

(for simple functions  $f$ ), as well as calculate explicitly some integrals of this form, with specific choices of the matrix  $\mathbf{A}$ , that we encounter in the lectures.

*Real, Symmetric, Positive Definite Matrices.* The symmetric  $N \times N$  matrix  $\mathbf{A}$  is assumed to be positive definite, i.e.  $\mathbf{x} \cdot \mathbf{A} \mathbf{x} > 0$  for all  $\mathbf{x} \in \mathfrak{R}^N$  with  $|\mathbf{x}| \neq 0$ . The eigenvalue polynomial  $\det[\mathbf{A} - \lambda \mathbf{I}] = 0$  is of order  $N$ , so  $\mathbf{A}$  will have  $N$  (possibly complex) solutions  $\lambda$  (some may coincide) of the eigenvalue problem

$$\mathbf{A} \mathbf{x} = \lambda \mathbf{x}, \quad \mathbf{x} \neq 0 \tag{B.1}$$

We denote complex conjugation of complex numbers  $z$  in the usual way:  $z = a + ib$ ,  $z^* = a - ib$  ( $a, b \in \mathfrak{R}$ ), and  $|z|^2 = z^* z \in \mathfrak{R}$ . We denote the unit matrix in  $\mathfrak{R}^N$  with  $\mathbf{I}$ , so  $\mathbf{I}_{ij} = \delta_{ij}$ .

**Fact 1:** All eigenvalues of the matrix  $\mathbf{A}$  are real.

**Proof:** In (B.1) we take the inner product with the conjugate vector  $\mathbf{x}^*$ , which gives

$$\sum_{i,j=1}^N x_i^* A_{ij} x_j = \lambda \sum_{i=1}^N |x_i|^2$$

We use the symmetry of  $\mathbf{A}$ , and substitute  $A_{ij} \rightarrow \frac{1}{2}[A_{ij} + A_{ji}]$ :

$$\lambda = \frac{1}{2} \frac{\sum_{ij} x_i^* [A_{ij} + A_{ji}] x_j}{\sum_{i=1}^N |x_i|^2} = \frac{1}{2} \frac{\sum_{ij} A_{ij} [x_i^* x_j + x_i x_j^*]}{\sum_{i=1}^N |x_i|^2}$$

Since  $[x_i^* x_j + x_i x_j^*]^* = x_i x_j^* + x_i^* x_j = x_i^* x_j + x_i x_j^*$ , the above fraction is entirely real-valued, so  $\lambda \in \mathfrak{R}$ .



**Fact 2:** All eigenvectors can be chosen real-valued.

**Proof:** For a given eigenvalue  $\lambda$  the corresponding eigenvectors  $\mathbf{x}$  are the solutions of (B.1). We separate real and imaginary parts of every eigenvector:

$$\mathbf{x} = \operatorname{Re}\mathbf{x} + i\operatorname{Im}\mathbf{x} \quad \operatorname{Re}\mathbf{x} = \frac{1}{2}[\mathbf{x} + \mathbf{x}^*] \quad \operatorname{Im}\mathbf{x} = \frac{1}{2i}[\mathbf{x} - \mathbf{x}^*]$$

with  $\operatorname{Re}\mathbf{x} \in \mathfrak{R}^N$  and  $\operatorname{Im}\mathbf{x} \in \mathfrak{R}^N$ . Taking the complex conjugate of equation (B.1) gives  $\mathbf{A}\mathbf{x}^* = \lambda\mathbf{x}^*$  (since  $\lambda$  is real). Apparently, if  $\mathbf{x}$  is an eigenvector with eigenvalue  $\lambda$ , so is  $\mathbf{x}^*$ . By adding/subtracting the conjugate equation to/from the original equation (B.1) it follows, in turn: if  $\mathbf{x}$  and  $\mathbf{x}^*$  are eigenvectors, so are  $\operatorname{Re}\mathbf{x}$  and  $\operatorname{Im}\mathbf{x}$ . Complex eigenvectors always come in conjugate pairs, and, since the space spanned by  $\mathbf{x}$  and  $\mathbf{x}^*$  is the same as the space spanned by  $\operatorname{Re}\mathbf{x}$  and  $\operatorname{Im}\mathbf{x}$ , we are always allowed to choose the equivalent real-valued pair  $\operatorname{Re}\mathbf{x}$  and  $\operatorname{Im}\mathbf{x}$ .

**Fact 3:** All eigenvalues  $\lambda$  are positive.

**Proof:** From the eigenvalue equation (B.1) we derive this property by taking the inner product with  $\mathbf{x}$ :  $\lambda = (\mathbf{x} \cdot \mathbf{A}\mathbf{x})/(\mathbf{x}^2) > 0$ , since  $\mathbf{A}$  is positive definite and  $\mathbf{x}$  is real and nonzero.

**Fact 4:** For every linear subspace  $L \subseteq \mathfrak{R}^N$  the following holds:

$$\text{if } \mathbf{A}L \subseteq L \text{ then also } \mathbf{A}L^\perp \subseteq L^\perp$$

in which  $L^\perp$  denotes the orthogonal complement, i.e.  $\mathfrak{R}^N = L \otimes L^\perp$ .

**Proof:** For each  $\mathbf{x} \in L$  we find  $(\mathbf{x} \cdot \mathbf{A}\mathbf{y}) = (\mathbf{y} \cdot \mathbf{A}\mathbf{x}) = 0$  (since  $\mathbf{A}\mathbf{x} \in L$  and  $\mathbf{y} \in L^\perp$ ). Therefore  $\mathbf{A}\mathbf{y} \in L^\perp$ , which completes the proof.

**Fact 5:** We can construct a complete orthogonal basis in  $\mathfrak{R}^N$  of  $\mathbf{A}$ -eigenvectors.

**Proof:** Consider two eigenvectors  $\mathbf{x}_a$  and  $\mathbf{x}_b$  of  $\mathbf{A}$ , corresponding to different eigenvalues:

$$\mathbf{A}\mathbf{x}_a = \lambda_a\mathbf{x}_a \quad \mathbf{A}\mathbf{x}_b = \lambda_b\mathbf{x}_b \quad \lambda_a \neq \lambda_b$$

We form:

$$\begin{aligned} 0 &= (\mathbf{x}_a \cdot \mathbf{A}\mathbf{x}_b) - (\mathbf{x}_a \cdot \mathbf{A}\mathbf{x}_b) = (\mathbf{x}_a \cdot \mathbf{A}\mathbf{x}_b) - (\mathbf{x}_b \cdot \mathbf{A}\mathbf{x}_a) \\ &= \lambda_b(\mathbf{x}_a \cdot \mathbf{x}_b) - \lambda_a(\mathbf{x}_b \cdot \mathbf{x}_a) = (\lambda_a - \lambda_b)(\mathbf{x}_a \cdot \mathbf{x}_b) \end{aligned}$$

Since  $\lambda_a \neq \lambda_b$  it follows that  $\mathbf{x}_a \cdot \mathbf{x}_b = 0$ . Eigenspaces corresponding to different eigenvalues are mutually orthogonal. If all eigenvalues are distinct, this completes the proof, there being  $N$  eigenvalues with corresponding eigenvectors  $\mathbf{x} \neq 0$ . Since these

are proven orthogonal, after normalisation  $\mathbf{x} \rightarrow \mathbf{x}/|\mathbf{x}|$  they form a complete orthogonal basis.

To deal with degenerate eigenvalues we need Fact 4. For every symmetric  $N \times N$  matrix we know: if  $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ , then  $\forall \mathbf{y}$  with  $\mathbf{x} \cdot \mathbf{y} = 0$ :  $(\mathbf{A}\mathbf{y}) \cdot \mathbf{x} = 0$ . Having found such an eigenvector for a given eigenvalue  $\lambda$  (not unique in the case of a degenerate eigenvalue), a new reduced  $(N-1) \times (N-1)$  matrix can be constructed by restricting ourselves to the subspace  $\mathbf{x}^\perp$ . The new matrix is again symmetric, the eigenvalue polynomial is of order  $N-1$  (and contains all the previous roots except for one corresponding to the eigenvector just eliminated), and we can repeat the argument. This shows that there *must* be  $N$  orthogonal eigenvectors, which we can normalise and use as a basis in  $\mathfrak{R}^N$ .

Final result: there exist a set of vectors  $\{\hat{\mathbf{e}}^i\}$  ( $\lambda = 1, \dots, N$ ) with the properties:

$$\mathbf{A}\hat{\mathbf{e}}^i = \lambda_i \hat{\mathbf{e}}^i \quad \lambda_i \in \mathfrak{R}, \lambda_i > 0 \quad \hat{\mathbf{e}}^i \in \mathfrak{R}^N, \hat{\mathbf{e}}^i \cdot \hat{\mathbf{e}}^j = \delta_{ij} \quad (\text{B.2})$$

We can now bring  $\mathbf{A}$  onto diagonal form by a simple unitary transformation  $\mathbf{U}$ , which we construct from the components of the normalised eigenvectors  $\hat{\mathbf{e}}$ :  $U_{ij} = \hat{\mathbf{e}}_i^j$ . We denote the transpose of  $\mathbf{U}$  by  $\mathbf{U}^\dagger$ ,  $U_{ij}^\dagger = U_{ji}$ , and show that  $\mathbf{U}$  is indeed unitary, i.e.  $\mathbf{U}^\dagger \mathbf{U} = \mathbf{U} \mathbf{U}^\dagger = \mathbf{I}$ :

$$\begin{aligned} \sum_j (\mathbf{U}^\dagger \mathbf{U})_{ij} x_j &= \sum_{jk} U^{ki} U_{kj} x_j = \sum_{jk} \hat{\mathbf{e}}_k^i \hat{\mathbf{e}}_k^j x_j = \sum_j \delta_{ij} x_j = x_i \\ \sum_j (\mathbf{U} \mathbf{U}^\dagger)_{ij} x_j &= \sum_{jk} U^{ik} U_{jk} x_j = \sum_{jk} \hat{\mathbf{e}}_i^k \hat{\mathbf{e}}_j^k x_j = \sum_k \hat{\mathbf{e}}_i^k (\hat{\mathbf{e}} \cdot \mathbf{x}) = x_i \end{aligned}$$

(since  $\{\hat{\mathbf{e}}^\ell\}$  forms a complete orthogonal basis). From  $\mathbf{U}$  being unitary it follows that  $\mathbf{U}$  and  $\mathbf{U}^\dagger$  leave inner products, and therefore also lengths, invariant:

$$\mathbf{U}\mathbf{x} \cdot \mathbf{U}\mathbf{y} = \mathbf{x} \cdot \mathbf{U}^\dagger \mathbf{U}\mathbf{y} = \mathbf{x} \cdot \mathbf{y} \quad \mathbf{U}^\dagger \mathbf{x} \cdot \mathbf{U}^\dagger \mathbf{y} = \mathbf{x} \cdot \mathbf{U} \mathbf{U}^\dagger \mathbf{y} = \mathbf{x} \cdot \mathbf{y}$$

We can see explicitly that  $\mathbf{U}$  indeed brings  $\mathbf{A}$  onto diagonal form:

$$(\mathbf{U}^\dagger \mathbf{A} \mathbf{U})_{ij} = \sum_{kl=1}^N U_{ik}^\dagger A_{kl} U_{lj} = \sum_{kl=1}^N \hat{\mathbf{e}}_k^i A_{kl} \hat{\mathbf{e}}_l^j = \lambda_j \sum_{k=1}^N \hat{\mathbf{e}}_k^i \hat{\mathbf{e}}_k^j = \lambda_j \delta_{ij} \quad (\text{B.3})$$

Note that the inverse  $\mathbf{A}^{-1}$  of the matrix  $\mathbf{A}$  exists, and can be written as follows:

$$(\mathbf{A}^{-1})_{ij} = \sum_{k=1}^N \lambda_k^{-1} \hat{\mathbf{e}}_i^k \hat{\mathbf{e}}_j^k \quad (\text{B.4})$$

To prove that this is indeed the inverse of  $\mathbf{A}$ , we just work out for any  $\mathbf{x} \in \mathfrak{R}^N$  the two expressions

$$(\mathbf{A} \mathbf{A}^{-1} \mathbf{x})_i = \sum_{kj=1}^N A_{ik} \sum_{\ell=1}^N \lambda_\ell^{-1} \hat{\mathbf{e}}_k^\ell \hat{\mathbf{e}}_j^\ell x_j = \sum_{\ell=1}^N \hat{\mathbf{e}}_i^\ell (\hat{\mathbf{e}}^\ell \cdot \mathbf{x}) = x_i$$

(again since  $\{\hat{\mathbf{e}}^\ell\}$  forms a complete orthogonal basis), and

$$(\mathbf{A}^{-1} \mathbf{A} \mathbf{x})_i = \sum_{kj=1}^N \sum_{\ell=1}^N \lambda_\ell^{-1} \hat{\mathbf{e}}_i^\ell \hat{\mathbf{e}}_k^\ell A_{kj} x_j = \sum_{\ell=1}^N \hat{\mathbf{e}}_i^\ell (\hat{\mathbf{e}}^\ell \cdot \mathbf{x}) = x_i$$

*Gaussian Integrals.* We now turn to the associated Gaussian integrals

$$I = \int d\mathbf{x} f(\mathbf{x}) e^{-\frac{1}{2}\mathbf{x}\cdot\mathbf{A}\mathbf{x}} \quad (\text{B.5})$$

The simplest such integral is

$$\int dx e^{-\frac{1}{2}x^2} = \sqrt{2\pi}$$

Proof:

Write the square of the integral as a single integral in  $\mathfrak{R}^2$ , and switch to polar coordinates, ( $z_1 = r \cos \phi$ ,  $z_2 = r \sin \phi$ ). The Jacobian of this coordinate transformation is simply  $r$ .

$$I^2 = \int dz_1 dz_2 e^{-\frac{1}{2}z^2} = \int_0^{2\pi} d\phi \int_0^\infty dr r e^{-\frac{1}{2}r^2} = 2\pi \left[ -e^{-\frac{1}{2}r^2} \right]_0^\infty = 2\pi$$

Therefore  $I = \sqrt{2\pi}$ .

For  $f(\mathbf{x}) = 1$  we can do the integral (B.5) by using the previous results on the diagonalisability of the matrix  $\mathbf{A}$ . We put  $\mathbf{x} = \mathbf{U}\mathbf{z}$  (since  $\mathbf{U}$  leaves inner products invariant:  $d\mathbf{x} = d\mathbf{z}$ ):

$$\begin{aligned} \int d\mathbf{x} e^{-\frac{1}{2}\mathbf{x}\cdot\mathbf{A}\mathbf{x}} &= \int d\mathbf{z} e^{-\frac{1}{2}\mathbf{z}\cdot\mathbf{U}^\dagger\mathbf{A}\mathbf{U}\mathbf{z}} = \prod_{\ell=1}^N \left[ \int dz e^{-\frac{1}{2}\lambda_\ell z^2} \right] \\ &= \left[ \prod_{\ell=1}^N \frac{1}{\sqrt{\lambda_\ell}} \right] \left[ \int dz e^{-\frac{1}{2}z^2} \right]^N = \frac{(2\pi)^{N/2}}{\sqrt{\det \mathbf{A}}} \end{aligned} \quad (\text{B.6})$$

(note: the determinant of  $\mathbf{A}$  is unvariant under rotations, so it can be evaluated with  $\mathbf{A}$  on diagonal form, which gives the product of the  $N$  eigenvalues).

Due to the symmetry of the integrand in (B.5) under reflection  $\mathbf{x} \rightarrow -\mathbf{x}$ , the integral reduces to zero for  $f(\mathbf{x}) = x_i$ . For  $f(\mathbf{x}) = x_i x_j$  we find:

$$\begin{aligned} \int d\mathbf{x} x_i x_j e^{-\frac{1}{2}\mathbf{x}\cdot\mathbf{A}\mathbf{x}} &= \lim_{\mathbf{b} \rightarrow 0} \frac{\partial^2}{\partial b_i \partial b_j} \int d\mathbf{x} e^{-\frac{1}{2}\mathbf{x}\cdot\mathbf{A}\mathbf{x} + \mathbf{b}\cdot\mathbf{x}} \\ &= \lim_{\mathbf{b} \rightarrow 0} \frac{\partial^2}{\partial b_i \partial b_j} \int d\mathbf{z} e^{-\frac{1}{2}\mathbf{z}\cdot\mathbf{U}^\dagger\mathbf{A}\mathbf{U}\mathbf{z} + \mathbf{z}\cdot\mathbf{U}^\dagger\mathbf{b}} \\ &= \lim_{\mathbf{b} \rightarrow 0} \frac{\partial^2}{\partial b_i \partial b_j} \prod_{\ell=1}^N \left[ \int dz e^{-\frac{1}{2}\lambda_\ell z^2 + z(\mathbf{U}^\dagger\mathbf{b})_\ell} \right] \\ &= \lim_{\mathbf{b} \rightarrow 0} \frac{\partial^2}{\partial b_i \partial b_j} \prod_{\ell=1}^N \left[ \int dz e^{-\frac{1}{2}\lambda_\ell [z - (\mathbf{U}^\dagger\mathbf{b})_\ell \lambda_\ell^{-1}]^2 + \frac{1}{2}\lambda_\ell^{-1} (\mathbf{U}^\dagger\mathbf{b})_\ell^2} \right] \\ &= \lim_{\mathbf{b} \rightarrow 0} \frac{\partial^2}{\partial b_i \partial b_j} e^{\frac{1}{2} \sum_{ij\ell=1}^N \lambda_\ell^{-1} U_{i\ell} b_i U_{j\ell} b_j} \prod_{\ell=1}^N \left[ \int dz e^{-\frac{1}{2}\lambda_\ell z^2} \right] \end{aligned}$$

$$\begin{aligned}
&= \frac{(2\pi)^{N/2}}{\sqrt{\det \mathbf{A}}} \lim_{\mathbf{b} \rightarrow 0} \frac{\partial^2}{\partial b_i \partial b_j} e^{\frac{1}{2} \sum_{ij=1}^N b_i b_j \sum_{\ell=1}^N \lambda_\ell^{-1} \hat{e}_i^\ell \hat{e}_j^\ell} \\
&= (A^{-1})_{ij} \frac{(2\pi)^{N/2}}{\sqrt{\det \mathbf{A}}}
\end{aligned} \tag{B.7}$$

In particular, by combining the last two results, we find a powerful (and completely general) relation for Gaussian probability distributions with zero mean  $\mathbf{x} = 0$  (the latter one can always achieve by a simple translation):

$$\frac{\int d\mathbf{x} x_i x_j e^{-\frac{1}{2} \mathbf{x} \cdot \mathbf{A} \mathbf{x}}}{\int d\mathbf{x} e^{-\frac{1}{2} \mathbf{x} \cdot \mathbf{A} \mathbf{x}}} = (A^{-1})_{ij} \tag{B.8}$$

If we know that a given distribution is Gaussian, with zero average, we apparently only need to calculate the correlations  $\langle x_i x_j \rangle$  to know the full distribution:

$$P(\mathbf{x}) \text{ Gaussian, with } \langle \mathbf{x} \rangle = 0 \quad \Rightarrow \quad P(\mathbf{x}) = \frac{e^{-\frac{1}{2} \mathbf{x} \cdot \mathbf{A} \mathbf{x}}}{(2\pi)^{N/2} \det^{-\frac{1}{2}} \mathbf{A}}, \text{ with } (A^{-1})_{ij} = \langle x_i x_j \rangle \tag{B.9}$$



# Appendix C

## The $\delta$ -Distribution

*Definition.* There are several ways of introducing the  $\delta$ -distribution. Here we will go for an intuitive definition first, and a formal one later. We define the  $\delta$ -distribution as the probability distribution  $\delta(x)$  corresponding to a random variable in the limit where the randomness in the variable vanishes. If  $x$  is ‘distributed’ around zero, this implies

$$\int dx f(x)\delta(x) = f(0) \quad \text{for any function } f$$

The problem arises when we want to actually write down an expression for  $\delta(x)$ . Intuitively one could think of writing something like

$$\delta(x) = \lim_{\Delta \rightarrow 0} G_{\Delta}(x) \quad G_{\Delta}(x) = \frac{1}{\Delta\sqrt{2\pi}} e^{-\frac{1}{2}x^2/\Delta^2} \quad (\text{C.1})$$

This is not a true function in a mathematical sense;  $\delta(x)$  is zero for  $x \neq 0$  and  $\delta(0) = \infty$ . The way to interpret and use expressions like (C.1) is to realise that  $\delta(x)$  only has a meaning when appearing inside an integration. One then takes the limit  $\Delta \rightarrow 0$  *after* performing the integration. Upon adopting this convention, we can use (C.1) to derive the following properties (for sufficiently well-behaved and differentiable functions  $f^1$ ):

$$\int dx \delta(x)f(x) = \lim_{\Delta \rightarrow 0} \int dx G_{\Delta}(x)f(x) = \lim_{\Delta \rightarrow 0} \int \frac{dx}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} f(\Delta x) = f(0)$$

$$\begin{aligned} \int dx \delta'(x)f(x) &= \lim_{\Delta \rightarrow 0} \int dx \left\{ \frac{d}{dx} [G_{\Delta}(x)f(x)] - G_{\Delta}(x)f'(x) \right\} \\ &= \lim_{\Delta \rightarrow 0} [G_{\Delta}(x)f(x)]_{-\infty}^{\infty} - f'(0) = -f'(0) \end{aligned}$$

both can be summarised in and generalised to the single expression:

$$\int dx f(x) \frac{d^n}{dx^n} \delta(x) = (-1)^n \lim_{x \rightarrow 0} \frac{d^n}{dx^n} f(x) \quad (n = 0, 1, 2, \dots) \quad (\text{C.2})$$

---

<sup>1</sup>The conditions on the so-called ‘test-functions’  $f$  can be properly formalised; this being not a course on distribution theory, here we just concentrate on the basic ideas and properties

Equivalently we can take the result (C.2) as our definition of the  $\delta$ -distribution.

*Representations, Relations, Generalisations.* We can use the definitions of Fourier transforms and inverse Fourier transforms to obtain an integral representation of the  $\delta$ -distribution:

$$\begin{aligned}\mathcal{F} : f(x) &\rightarrow \hat{f}(k) & \hat{f}(k) &= \int dx e^{-2\pi i k x} f(x) \\ \mathcal{F}^{-1} : \hat{f}(k) &\rightarrow f(x) & f(x) &= \int dk e^{2\pi i k x} \hat{f}(k)\end{aligned}$$

In combination these relations give the identity:

$$f(x) = \int dk e^{2\pi i k x} \int dy e^{-2\pi i k y} f(y)$$

Application to  $f(x) = \delta(x)$  gives:

$$\delta(x) = \int dk e^{2\pi i k x} = \int \frac{dk}{2\pi} e^{i k x} \quad (\text{C.3})$$

A second useful relation is the following one, which relates the  $\delta$ -distribution to the step-function:

$$\delta(x) = \frac{d}{dx} \theta(x) \quad (\text{C.4})$$

This we prove by showing that both have the same effect inside an integration (with an arbitrary test-function):

$$\begin{aligned}\int dx \left[ \delta(x) - \frac{d}{dx} \theta(x) \right] \phi(x) &= \phi(0) - \lim_{\epsilon \rightarrow 0} \int_{-\epsilon}^{\epsilon} dx \left\{ \frac{d}{dx} [\theta(x)\phi(x)] - \phi'(x)\theta(x) \right\} \\ &= \phi(0) - \lim_{\epsilon \rightarrow 0} [\phi(\epsilon) - 0] + \lim_{\epsilon \rightarrow 0} \int_0^{\epsilon} dx \phi'(x) = 0\end{aligned}$$

Thirdly we can inspect the effect of performing a continuously differentiable and invertible transformation  $f$  on a variable that occurs inside a  $\delta$ -distribution, giving rise to the following identity:

$$\delta[f(x) - f(a)] = \frac{\delta(x - a)}{f'(a)} \quad (\text{C.5})$$

Again this is proved by showing that both sides have the same effect inside an integration (with an arbitrary test-function):

$$\begin{aligned}\int dx \phi(x) \left\{ \delta[f(x) - f(a)] - \frac{\delta(x - a)}{f'(a)} \right\} &= \int dx f'(x) \left[ \frac{\phi(x)}{f'(x)} \right] \delta[f(x) - f(a)] - \frac{\phi(a)}{f'(a)} \\ &= \int dk \frac{\phi(f^{-1}(k))}{f'(f^{-1}(k))} \delta[k - f(a)] - \frac{\phi(a)}{f'(a)} = \frac{\phi(f^{-1}(f(a)))}{f'(f^{-1}(f(a)))} - \frac{\phi(a)}{f'(a)} = 0\end{aligned}$$

Finally, the following generalisation is straightforward:

$$\mathbf{x} \in \mathcal{R}^N : \quad \delta(\mathbf{x}) = \prod_{i=1}^N \delta(x_i) \quad (\text{C.6})$$

## Appendix D

# Inequalities Based on Convexity

We first define what we mean by convexity and strict convexity.

**definition:**  $f$  is called convex on the open interval  $\langle a, b \rangle$  if

$$(\forall x_1, x_2 \in \langle a, b \rangle, x_1 \neq x_2)(\forall \lambda \in [0, 1]) : f[(1-\lambda)x_1 + \lambda x_2] \leq (1-\lambda)f[x_1] + \lambda f[x_2] \quad (\text{D.1})$$

**definition:**  $f$  is called strictly convex on the open interval  $\langle a, b \rangle$  if

$$(\forall x_1, x_2 \in \langle a, b \rangle, x_1 \neq x_2)(\forall \lambda \in (0, 1)) : f[(1-\lambda)x_1 + \lambda x_2] < (1-\lambda)f[x_1] + \lambda f[x_2] \quad (\text{D.2})$$

Next we show how that for twice differentiable functions convexity is a direct consequence of a strictly non-negative second derivative. When applicable, this is usually a far quicker way to demonstrate that a given function is (strictly) convex than by actually proving the inequalities (D.1,D.2):

**theorem:** If  $f$  is twice differentiable on the open interval  $\langle a, b \rangle$  and if  $f''[x] \geq 0$  for all  $x \in \langle a, b \rangle$ , then  $f$  is convex. If  $f''[x] > 0$  for all  $x \in \langle a, b \rangle$ , then  $f$  is strictly convex.

**proof:** Without loss of generality we choose  $a < x_1 < x_2 < b$ , and evaluate for  $0 < \lambda < 1$ :

$$\begin{aligned} f[(1-\lambda)x_1 + \lambda x_2] - (1-\lambda)f[x_1] - \lambda f[x_2] &= f[x_1 + \lambda(x_2 - x_1)] - f[x_1] - \lambda(f[x_2] - f[x_1]) \\ &= \int_{x_1}^{x_1 + \lambda(x_2 - x_1)} dy f'[y] - \lambda \int_{x_1}^{x_2} dy f'[y] \\ &= \lambda(x_2 - x_1) \int_0^1 dz f'[x_1 + z\lambda(x_2 - x_1)] - \lambda(x_2 - x_1) \int_0^1 dz f'[x_1 + z(x_2 - x_1)] \\ &= -\lambda(x_2 - x_1) \int_0^1 dz \{f'[x_1 + z(x_2 - x_1)] - f'[x_1 + z\lambda(x_2 - x_1)]\} \\ &= -\lambda(x_2 - x_1) \int_0^1 dz \int_{x_1 + z\lambda(x_2 - x_1)}^{x_1 + z(x_2 - x_1)} dy f''[y] \leq 0 \end{aligned}$$

We conclude that  $f$  is convex. Finally, if the stronger statement  $f''[x] > 0$  for all  $x \in \langle a, b \rangle$  is true, than we can replace ' $\leq 0$ ' by '< 0' in the last step, so  $f$  is strictly convex. This completes the proof.  $\square$



The converse statement is also true, but we will not use it in this course. Finally we turn to the main subject of this appendix, two important inequalities based on convexity: Jensen's inequality and the so-called log-sum inequality.

**Jensen's inequality:** If  $f$  is convex on the open interval  $\langle a, b \rangle$ , and  $x \in \langle a, b \rangle$  is a random variable, then

$$\langle f[x] \rangle \geq f[\langle x \rangle] \quad (\text{D.3})$$

Furthermore: if  $f$  is strictly convex on the open interval  $\langle a, b \rangle$ , then the equality in (D.3) holds if and only if  $x$  is a constant.

**proof for discrete random variables:** We use induction with respect to the number  $n$  of values the random variable  $x$  can take, i.e.  $x \in A = \{x_1, \dots, x_n\}$ . Since  $f$  is convex and since  $\sum_{i=1}^n p(x_i) = 1$  the statement (D.3) is clearly true for  $n = 2$ :

$$n = 2 : \quad \langle f[x] \rangle = p(x_1)f[x_1] + (1-p(x_1))f[x_2] \geq f[p(x_1)x_1 + (1-p(x_1))x_2] = f[\langle x \rangle]$$

Next we assume (D.3) to be true for a given value of  $n$  and prove that it then must also be true for the value  $n+1$ :

$$\langle f[x] \rangle = \sum_{i=1}^{n+1} p(x_i)f[x_i] = \sum_{i=1}^n p(x_i)f[x_i] + p(x_{n+1})f[x_{n+1}]$$

We define for  $i \in \{1, \dots, n\}$  the auxiliary probabilities  $\hat{p}(x_i)$ :

$$\hat{p}(x_i) = \frac{p(x_i)}{\sum_{j=1}^n p(x_j)}, \quad \hat{p}(x_i) \in [0, 1], \quad \sum_{i=1}^n \hat{p}(x_i) = 1$$

with which we obtain, using convexity of  $f$ , validity of (D.3) for  $|A| = n$ , as well as the normalisation  $\sum_{i=1}^{n+1} p(x_i) = 1$ :

$$\begin{aligned} \langle f[x] \rangle &= \left[ \sum_{j=1}^n p(x_j) \right] \sum_{i=1}^n \hat{p}(x_i)f[x_i] + p(x_{n+1})f[x_{n+1}] \\ &\geq \left[ \sum_{j=1}^n p(x_j) \right] f \left[ \sum_{i=1}^n \hat{p}(x_i)x_i \right] + \left[ 1 - \sum_{j=1}^n p(x_j) \right] f[x_{n+1}] \\ &\geq f \left[ \left( \sum_{j=1}^n p(x_j) \right) \sum_{i=1}^n \hat{p}(x_i)x_i + \left( 1 - \sum_{j=1}^n p(x_j) \right) x_{n+1} \right] \\ &= f \left[ \sum_{i=1}^{n+1} p(x_i)x_i \right] = f[\langle x \rangle] \end{aligned}$$

Finally, suppose  $f$  is strictly convex and suppose that we find an equality in (D.3). Then we know that at each step in the above derivation where the convexity-inequality

of (D.1) was used the corresponding value of  $\lambda \in [0, 1]$  in (D.1) must have been either 0 or 1. Inspection of the occurrences of the relevant inequalities in the above induction procedure shows that for all  $i \in \{1, \dots, n\}$ :  $p(x_i) \in \{0, 1\}$ . Since probabilities are normalised we find that precisely one of the  $n$  probabilities  $p(x_i)$  is 1 and the others are 0; this means that  $x$  is a constant, which completes the proof.  $\square$

**proof for continuous random variables:** We will here treat averages involving continuous random variables as limits of averages involving discrete ones (whereby integrals are written as limits of summations):

$$\begin{aligned} \langle f[x] \rangle - f[\langle x \rangle] &= \int_a^b dx p(x) f[x] - f \left[ \int_a^b dx p(x) x \right] \\ &= \lim_{n \rightarrow \infty} \left\{ \frac{b-a}{n} \sum_{i=1}^n p(x_i) f(x_i) - f \left[ \frac{b-a}{n} \sum_{i=1}^n p(x_i) x_i \right] \right\} \quad x_i = a + (i-1) \frac{b-a}{n} \end{aligned}$$

Again we define suitable auxiliary probabilities with built-in and  $n$ -independent normalisation:

$$\hat{p}(x_i) = \frac{p(x_i)}{\sum_{j=1}^n p(x_j)}, \quad \hat{p}(x_i) \in [0, 1], \quad \sum_{i=1}^n \hat{p}(x_i) = 1$$

so that we can use the validity of (D.3) for discrete random variables:

$$\begin{aligned} \langle f[x] \rangle - f[\langle x \rangle] &= \lim_{n \rightarrow \infty} \left\{ \frac{b-a}{n} \left[ \sum_{i=1}^n p(x_i) \right] \sum_{j=1}^n \hat{p}(x_j) f(x_j) - f \left[ \frac{b-a}{n} \sum_{i=1}^n p(x_i) x_i \right] \right\} \\ &\geq \lim_{n \rightarrow \infty} \left\{ \frac{b-a}{n} \left[ \sum_{i=1}^n p(x_i) \right] f \left[ \sum_{j=1}^n \hat{p}(x_j) x_j \right] - f \left[ \frac{b-a}{n} \left[ \sum_{i=1}^n p(x_i) \right] \sum_{i=1}^n \hat{p}(x_i) x_i \right] \right\} = 0 \end{aligned}$$

In the last step we used the properties  $\lim_{n \rightarrow \infty} \frac{b-a}{n} \sum_{i=1}^n p(x_i) = \int_a^b dx p(x) = 1$  and

$$\lim_{n \rightarrow \infty} \sum_{i=1}^n \hat{p}(x_i) x_i = \lim_{n \rightarrow \infty} \frac{\frac{b-a}{n} \sum_{i=1}^n p(x_i) x_i}{\frac{b-a}{n} \sum_{j=1}^n p(x_j)} = \frac{\lim_{n \rightarrow \infty} \frac{b-a}{n} \sum_{i=1}^n p(x_i) x_i}{\lim_{n \rightarrow \infty} \frac{b-a}{n} \sum_{j=1}^n p(x_j)} = \langle x \rangle$$

Finally, if  $f$  is strictly convex and we obtain an equality in (D.3) we know from the discrete case that for any finite value of  $n$  the random variable  $x$  must be a constant, which must then also be true if we take the above continuum limit  $n \rightarrow \infty$ . This completes our proof.  $\square$

The log-sum inequality, which we will turn to now, is a direct consequence of Jenssen's inequality. The only preparatory work needed is introducing the convention

$$0 \log 0 = \lim_{\epsilon \downarrow 0} \epsilon \log \epsilon = 0 \tag{D.4}$$

and showing that the function  $f(x) = \log(1/x)$  is strictly convex on the interval  $\langle 0, \infty \rangle$ :

$$x \in \langle 0, \infty \rangle : \quad \frac{d^2}{dx^2} f(x) = \frac{1}{x^2} > 0 \quad \Rightarrow \quad f \text{ is strictly convex on } \langle 0, \infty \rangle$$

In particular we can now apply Jenssen's theorem (D.3) to  $f$ .

**log-sum inequality:** If  $a_i, b_i \in [0, \infty)$ , with  $\sum_{i=1}^n a_i > 0$  and  $\sum_{i=1}^n b_i > 0$ , then:

$$\sum_{i=1}^n a_i \log \left[ \frac{a_i}{b_i} \right] \geq \left[ \sum_{i=1}^n a_i \right] \log \left[ \frac{\sum_{j=1}^n a_j}{\sum_{j=1}^n b_j} \right] \quad (\text{D.5})$$

with equality if and only if  $(\exists \lambda > 0) : b_i = \lambda a_i$  ( $\forall i \in \{1, \dots, n\}$ ).

**proof:** We define the variables  $x_i = b_i/a_i$ , with associated probabilities

$$p(x_i) = \frac{a_i}{\sum_{j=1}^n a_j}, \quad p(x_i) \in [0, 1], \quad \sum_{i=1}^n p(x_i) = 1$$

Note that statement (D.5) is trivially true as soon as  $(\exists i) : b_i = 0$  and  $a_i/b_i \neq 0$  (since in that case the left-hand side of the inequality diverges, whereas the right-hand side remains finite). Therefore we may restrict ourselves to those cases where  $b_i = 0$  always implies  $a_i/b_i = 0$ , so that  $a_i \log(a_i/b_i) = 0$ . According to (D.3) the convexity of the function  $f(x) = \log(1/x)$  allows us to write

$$\begin{aligned} \sum_{i=1}^n a_i \log \left[ \frac{a_i}{b_i} \right] &= \left[ \sum_{i=1}^n a_i \right] \sum_{i=1}^n p(x_i) \log(1/x_i) \geq \left[ \sum_{i=1}^n a_i \right] \log \left[ \frac{1}{\sum_{j=1}^n p(x_j)x_j} \right] \\ &= \left[ \sum_{i=1}^n a_i \right] \log \left[ \frac{\sum_{j=1}^n a_j}{\sum_{i=1}^n a_i (b_i/a_i)} \right] = \left[ \sum_{i=1}^n a_i \right] \log \left[ \frac{\sum_{j=1}^n a_j}{\sum_{i=1}^n b_i} \right] \end{aligned}$$

Finally, if we find an equality in (D.5) we know that the variables  $x_i = b_i/a_i$  in the above derivation must all be identical (since  $\log(1/x)$  is strictly convex). In other words:  $(\exists \lambda) : b_i = \lambda a_i$  for all  $i$ . Since both  $a_i \geq 0$  and  $b_i \geq 0$  (with  $\sum_i a_i > 0$  and  $\sum_i b_i > 0$ ) this constant  $\lambda$  must be positive. This completes the proof.  $\square$