Information Theory, Pattern Recognition and Neural Networks

Part III Physics exams 2006

1 A channel has a 3-bit input,

$$x \in \{\texttt{000}, \texttt{001}, \texttt{010}, \texttt{011}, \texttt{100}, \texttt{101}, \texttt{110}, \texttt{111}\},$$

and a 2-bit output $y \in \{\texttt{00}, \texttt{01}, \texttt{10}, \texttt{11}\}$. Given an input x, the output y is generated by *deleting* exactly one of the three input bits, selected at random. For example, if the input is $x = \texttt{010}$ then $P(y \,|\, x)$ is $1/3$ for each of the outputs `00`, `10`, and `01`; If the input is $x = \texttt{001}$ then $P(y{=}\texttt{01} \,|\, x) = 2/3$ and $P(y{=}\texttt{00} \,|\, x) = 1/3$.

Write down the conditional entropies $H(Y \,|\, x{=}\texttt{000})$, $H(Y \,|\, x{=}\texttt{010})$, and $H(Y \,|\, x{=}\texttt{001})$. [3]

Assuming an input distribution of the form

$$\begin{array}{ccccccccc} x & \texttt{000} & \texttt{001} & \texttt{010} & \texttt{011} & \texttt{100} & \texttt{101} & \texttt{110} & \texttt{111} \\ P(x) & \frac{1-p}{2} & \frac{p}{4} & 0 & \frac{p}{4} & \frac{p}{4} & 0 & \frac{p}{4} & \frac{1-p}{2}, \end{array}$$

work out the conditional entropy $H(Y \,|\, X)$ and show that

$$H(Y) = 1 + H_2\left(\frac{2}{3}p\right),$$

where $H_2(x) = x \log_2(1/x) + (1-x)\log_2(1/(1-x))$. [3]

Sketch $H(Y)$ and $H(Y \,|\, X)$ as a function of $p \in (0,1)$ on a single diagram. [5]

Sketch the mutual information $I(X;Y)$ as a function of p. [2]

$\left[\ H_2(1/3) \simeq 0.92.\ \right]$

Another channel with a 3-bit input

$$x \in \{\texttt{000}, \texttt{001}, \texttt{010}, \texttt{011}, \texttt{100}, \texttt{101}, \texttt{110}, \texttt{111}\},$$

erases exactly one of its three input bits, marking the erased symbol by a `?`. For example, if the input is $x = \texttt{010}$ then $P(y \,|\, x)$ is $1/3$ for each of the outputs `?10`, `0?0`, and `01?`.

What is the capacity of this channel? Describe a method for reliable communication over it. [7]

2 Three six-sided dice have faces labelled with symbols as follows.

Name of die	Number of faces having symbol					
	`r`	`s`	`t`	`u`	`v`	`w`
A	1	1	1	1	1	1
B	0	3	3	0	0	0
C	2	1	2	1	0	0

For example, die B has 3 faces labelled `s` and 3 faces labelled `t`.

One of the dice is selected at random and is rolled N times, creating a sequence of outcomes $x_1, x_2, \ldots x_N$. [The identity of the selected die, d, is not explicitly revealed.]

(a) What is the probability distribution of the first outcome, x_1? Describe an optimal binary symbol code for encoding the first outcome. [6]

(b) Assume that the first outcome is $x_1 = \texttt{s}$. Given this information, how probable are the alternative theories about which die was chosen, $d = \text{A}$, $d = \text{B}$, $d = \text{C}$? [3]

(c) Given that $x_1 = \texttt{s}$, what is the probability distribution of the second outcome x_2? [4]

(d) Assume the entire sequence of N outcomes $\boldsymbol{x} = x_1, x_2, \ldots x_N$ is compressed by arithmetic coding using the appropriate predictive distribution $P(x_n \,|\, x_1, \ldots, x_{n-1})$. Predict the compressed file's length $l(\boldsymbol{x})$ in detail, including a sketch of the probability distribution of the length. Assume $N \gg 100$. $\left[\ H_2(1/3) \simeq 0.92;\ \log_2 6 \simeq 2.6.\right]$ [7]

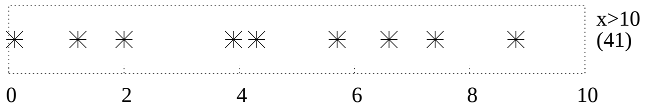
3 Decay events occur at distances $\{x_n\}$ from a source. Each distance x_n has an exponential distribution with characteristic length λ. If λ were known, the probability distribution of x would be

$$P(x \,|\, \lambda) = \frac{1}{\lambda} e^{-x/\lambda}.$$

The locations of decay events that occur in a window from $x = 0$ to $x = b$ are measured accurately. Decay events at locations $x_n > b$ *are also detected*, but the actual *value* of x_n is not obtained in those cases. The probability of such an overflow event is

$$\begin{aligned} P(x > b \,|\, \lambda) &= \int_b^\infty \frac{1}{\lambda} e^{-x/\lambda}\, \mathrm{d}x \\ &= e^{-b/\lambda}. \end{aligned}$$

In an experiment where the right hand side of the window is at $b = 10$ units, a data set of $N = 50$ events is obtained. Of these, $N_< = 9$ events occur in the



window $0 \leq x \leq b$, and $N_{>} = 41$ events have $x_n > b$. The 9 events in the window were at locations

$$\{0.1, 1.2, 2.0, 3.9, 4.3, 5.7, 6.6, 7.4, 8.8\}.$$

(The sum of these numbers is 40.0.)

(a) Write down the likelihood function and find the maximum-likelihood setting of the parameter λ. [7]

(b) Sketch the logarithm of the likelihood function, as a function of $\ln \lambda$. [3]

(c) Find error bars on $\ln \lambda$. [5]

(d) Imagine that we must choose a follow-up experiment to measure λ more accurately. There are two choices, with identical cost: either (A) the window size can be increased from $b = 10$ to $b = 200$ units, and $N' = 250$ new events can be observed; or (B) the window size can be left where it is, at $b = 10$ units, and a greater number, $N' = 5000$, of events can be observed.

Discuss which of these would be the more informative experiment. [5]

END OF PAPER

Solutions:

Channels

$H(Y \mid x{=}\texttt{000}) = 0.$

$H(Y \mid x{=}\texttt{010}) = \log_2 3.$

$H(Y \mid x{=}\texttt{001}) = H_2(1/3).$ [3]

$H(Y \mid X) = pH_2(1/3).$

$P(y = \texttt{00}) = \frac{1}{2}\left[(1-p) + \frac{p}{3}\right].$

$P(y = \texttt{01}) = \frac{1}{3}p.$

$P(y = \texttt{10}) = P(y = \texttt{01}).$

$P(y = \texttt{11}) = P(y = \texttt{00}).$ [3]

$H(Y) = 1 + H_2\left(\frac{2}{3}p\right).$

Sketch of $H(Y|X) = pH_2(1/3)$ is a straight line rising from 0 to 0.92. $H(Y)$ is a concave function rising steeply from 1 at $p = 0$, taking on maximum value of 2 at $p = 3/4$, then decreasing to 1.92 at $p = 1$ with shallower slope. [5]

$I(X;Y)$ is the difference between these curves. It is a concave function with a maximum somewhere to the left of $p = 3/4$ and with value 1 at both $p = 1$ and $p = 0$. The slope at $p = 0$ is steeper; the slope at $p = 1$ is not vertical. . [2]

For the channel with erasures, the optimal input distribution is uniform, by symmetry. The entropy of the input is $H(X) = 3$, and the conditional entropy is $H(X|Y) = 1$. The capacity is 2. Reliable communication at rate 2 can be achieved by using a subset of the inputs such as 000, 011, 101, 110. These are non-confusable inputs. The third bit can be determined from the first two by setting it to their parity. The decoder can recover the erased bit by setting it such that the total parity of all three bits is even. This encoding method achieves capacity. [7]

Compression question

The probability distribution of the first outcome is $(3, 5, 6, 2, 1, 1)/18$. The optimal codewords produced by the Huffman algorithm are:

a_i	p_i	$\log_2 \frac{1}{p_i}$	l_i	$c(a_i)$
r	3/18= 0.1667	2.6	2	00
s	5/18= 0.2778	1.8	2	10
t	6/18= 0.3333	1.6	2	11
u	2/18= 0.1111	3.2	3	010
v	1/18= 0.05556	4.2	4	0110
w	1/18= 0.05556	4.2	4	0111

The probabilities of $d = \text{A}$, $d = \text{B}$, $d = \text{C}$, given $x_1 = \texttt{s}$, are 1/5, 3/5, 1/5. [3]

The probability of the second outcome, given $x_1 = \texttt{s}$, is (marginalizing over d): $(3, 11, 12, 2, 1, 1)/30.$ [4]

The compressed length will be the information content of the outcome; the inference of which die is the die will be very certain after this many tosses, so the information content will be dominated by $N\times$ the conditional entropy of X given

that value of d, whatever it is. If $d = \mathrm{A}$ then the conditional entropy of X is $\log_2 6$, and the information content of every outcome is exactly this; similarly if $d = \mathrm{B}$ then the conditional entropy of X is 1, and the information content of every outcome is exactly this; finally if $d = \mathrm{C}$ then the conditional entropy of X is $1 + H_2(1/3) = 1.92$, and the information content of every outcome is either $\log_2 3$ or $\log_2 6 = 1 + \log_2 3$. It's like a bent coin. The standard deviation of the number of heads is $\sqrt{N\frac{1}{3}\frac{2}{3}}$. The information content is linearly related to the number of heads, with slope 1. The standard deviation of the information content is therefore exactly $\sqrt{N\frac{1}{3}\frac{2}{3}}$ also.
So the information content of the whole outcome has a probability distribution with three peaks, one at N plus a little; one at $N(1 + H_2(1/3)) = 1.92N$ (plus a little); and one at $N \log_2 6 = 2.6N$ (plus a little). The "little" is at most about $\log 3$, the information content of which die was chosen. Each peak has mass $1/3$. The middle peak has standard deviation $\sqrt{N\frac{1}{3}\frac{2}{3}}$. The other two peaks have negligible width. The compressed file's length is one or two bits more than the information content. [7]

Inference question

In this answer, all logs are natural. The log likelihood is

$$\begin{aligned}
\log P(D \mid \lambda) &= \sum_{n\text{`inside'}} \left[\log \frac{1}{\lambda} - \frac{x_n}{\lambda}\right] + N_> \left[-\frac{b}{\lambda}\right] \\
&= -N_< \log \lambda - \frac{N_> b + \sum_n x_n}{\lambda} \\
&= -N_< \log \lambda - \left(N_> b + \sum_n x_n\right) e^{-\log \lambda}
\end{aligned}$$

Differentiating,

$$\begin{aligned}
\frac{\partial}{\partial \log \lambda} \log P(D \mid \lambda) &= -N_< + \left(N_> b + \sum_n x_n\right) e^{-\log \lambda} \\
&= -N_< + \left(N_> b + \sum_n x_n\right) \Big/ \lambda
\end{aligned}$$

$$\frac{\partial^2}{\partial \log \lambda^2} \log P(D \mid \lambda) = -\left(N_> b + \sum_n x_n\right) e^{-\log \lambda}$$

For the maximum likelihood, we set the derivative to zero, getting [7]

$$\lambda = \frac{N_> b + \sum_n x_n}{N_<} = \frac{410 + 40}{9} = \frac{450}{9} = 50.$$

(Sketch) [3]
To get error bars we take the curvature at the maximum, which is $N_<$.

$$\sigma^2_{\log\lambda} = 1 \Bigg/ \frac{\partial^2}{\partial \log\lambda^2} \log P(D \,|\, \lambda)\bigg|_{\lambda=\lambda_{\rm ML}} = 1/N_< = 1/9.$$

So the error bars on $\log\lambda$ are $\pm 1/3$. [5]
The first experiment is expected to yield error bars of about 1/16. (Since we'd get about 250 points inside, and the error bars are always $1/\sqrt{N_{\rm inside}}$.) The second would give error bars about 1/30. (Expected number inside about 900.) So it's best to choose the second experiment, assuming we really believe in our model. [5]

Information Theory, Pattern Recognition and Neural Networks

Part III Physics exam 2005

1

2 Define the capacity and the optimal input distribution of a noisy channel with input x and output y. [3]

Describe how the possibility of reliable communication over a noisy channel is related to the channel's capacity. [5]

3 A noisy channel has a 4-bit input

$$x \in \{\texttt{0000}, \texttt{0001}, \texttt{0010}, \texttt{0011}, \ldots, \texttt{1100}, \texttt{1101}, \texttt{1110}, \texttt{1111}\},$$

and a 3-bit output $y \in \{\texttt{000}, \texttt{001}, \texttt{010}, \texttt{011}, \texttt{100}, \texttt{101}, \texttt{110}, \texttt{111}\}$. Given an input x, the output y is generated by *deleting* exactly one of the four input bits, selected at random. For example, if the input is $x = \texttt{1010}$ then $P(y \,|\, x)$ is $1/4$ for each of the outputs 010, 110, 100, 101; If the input is $x = \texttt{0001}$ then $P(y{=}\texttt{001} \,|\, x) = 3/4$ and $P(y{=}\texttt{000} \,|\, x) = 1/4$.

Assume that the input is selected from a uniform distribution over all 16 possible inputs. What is the entropy of the output, $H(Y)$? [1]

Show that the conditional entropy of Y given X satisfies [2]

$$H(Y|X) < 2\,\text{bits}.$$

Hence show that the capacity of the channel, C, satisfies [1]

$$C > 1\,\text{bit}.$$

Now consider a second input distribution that uses just four of the sixteen inputs:

$$P(x) = \begin{cases} 1/4 & \text{if } x \in \{\texttt{0000}, \texttt{0011}, \texttt{1100}, \texttt{1111}\} \\ 0 & \text{otherwise.} \end{cases}$$

Assuming this input distribution, evaluate the entropy of the input $H(X)$ and the conditional entropy of the input given the output, $H(X|Y)$. [3]

Hence show that the capacity of the channel C satisfies [1]

$$C \geq 2\,\text{bits}.$$

Do you think that the capacity of the channel is exactly 2 bits, or is there a better input distribution? Explain your reasoning. (Further calculations are not expected.) [4]

4 A source emits N symbols from a 17-character alphabet $\{\texttt{a}, \texttt{b}, \texttt{c}, \ldots, \texttt{q}\}$. Successive symbols are *dependent*, as follows. The first symbol, x_1, is equally likely to be any of the 17 symbols. For all $n > 1$, the nth symbol, x_n, is equal to the preceding symbol x_{n-1} with probability 0.99 and different from x_{n-1} with probability 0.01; and if it is different, x_n is equally likely to be any of the 16 symbols that are not x_{n-1}.

Describe an optimal binary symbol code for encoding the *first* symbol, x_1.

If each of the N symbols is encoded using the symbol code you have described, what would be the expected length of the compressed file?

If instead, for all $n > 1$, the nth symbol were encoded using a context-dependent symbol code, namely, a binary code that depends on the value of x_{n-1}, how small could the compressed file be? Describe an optimal context-dependent symbol code for encoding x_n.

What is the conditional entropy of the nth symbol given the $n-1$st, $H(X_n|X_{n-1})$?
$[H_2(0.01) \simeq 0.08.]$

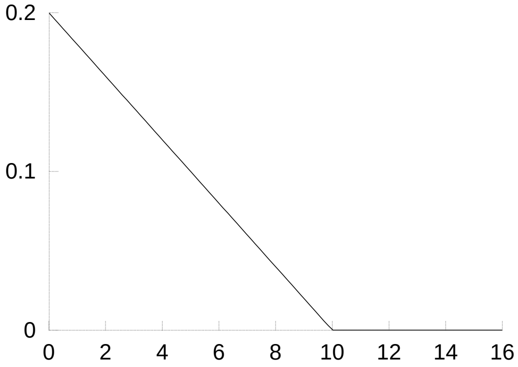
Describe one way in which even better compression could be achieved, assuming the number of source symbols N is large. Estimate how well your method would compress, and estimate the best conceivable compression that *any* compression method could deliver.

5 (a) Explain the terms *likelihood function*, *prior probability distribution*, and *posterior probability distribution*, in the context of the inference of parameters θ from data D. [4]

(b) Random variables x_n come from the triangular distribution with length-scale λ,

$$P(x \mid \lambda) = \begin{cases} \frac{2}{\lambda}\left(1 - \frac{x}{\lambda}\right) & x \in (0, \lambda) \\ 0 & \text{otherwise.} \end{cases}$$

The figure illustrates this density for the case $\lambda = 10$.



A data set $D = \{x_n\}_{n=1}^{N}$ consists of N points from the distribution $P(x \mid \lambda)$.

Sketch the likelihood function, for λ from 0 to 20, assuming the data set is $D = \{2, 1, 4\}$. [Your sketch should indicate roughly the location of the maximum or maxima of this function, but accurate estimates are *not* required.] [5]

Sketch the likelihood function assuming the data set is

$$D = \{2, 2, 2, 2, 1, 1, 1, 1, 4, 4, 4, 4\},$$

relating your answer to your previous sketch in as many ways as possible. [5]

(c) Two random variables c and x are generated as follows. First, a binary class variable c is generated with $P(c{=}0) = 1/2$ and $P(c{=}1) = 1/2$. Second, the length-scale λ is set to λ_c, where $\lambda_0 = 4$ and $\lambda_1 = 8$, and x is generated from the triangular distribution $P(x \mid \lambda)$.

Given the outcome x, we wish to infer the class c. Describe the optimal classifier, including a sketch of the posterior probability of c as a function of x. [6]

END OF PAPER

Answers

1 (a) The capacity is the maximum of the mutual information over all input distributions.

$$C = \max_{P_X} I(X;Y)$$

where

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X).$$

Any probability distribution that achieves this maximum is called an optimal input distribution.

Shannon proved that it is possible to add encoding and decoding systems before and after the noisy channel in such a way that information is communicated reliably at any rate R (in bits per channel use) less than the capacity. More precisely, given any required error probability $\epsilon > 0$ and and rate $R < C$, there exists an encoder with rate $> R$ and a decoder such that the maximal probability of error is $< \epsilon$. [Mention *encoder*, *decoder*, explain *rate*, explain *reliable*, say $R < C$, include figure showing error probability versus rate and the achievable region.]

(b.i) Entropy of output $H(Y) = 3$.

For each input the number of possible outputs is at most 4, and in some cases it is less. So $H(Y|X)$ is the average of several quantities, some equal to $\log_2 4$ and some smaller. So $H(Y|X) < \log_2 4 = 2$.

So using the definition

$$I(X;Y) = H(Y) - H(Y|X),$$

this choice of input distribution achieves $I(X;Y) > 3 - 2 = 1$. So the capacity, which is the maximum possible I, must satisfy $C > 1$.

(b.ii) $H(X) = 2$. These four inputs form a non-confusable subset. Given any of these inputs, the output can be mapped back to the correct input with no uncertainty. 000 → 0000; 001 and 011 → 0011; 110 and 100 → 1100; 111 → 1111. So $H(X|Y) = 0$.

Using

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

we find that this input distribution has $I(X;Y) = 2$. So the capacity must be at least 2.

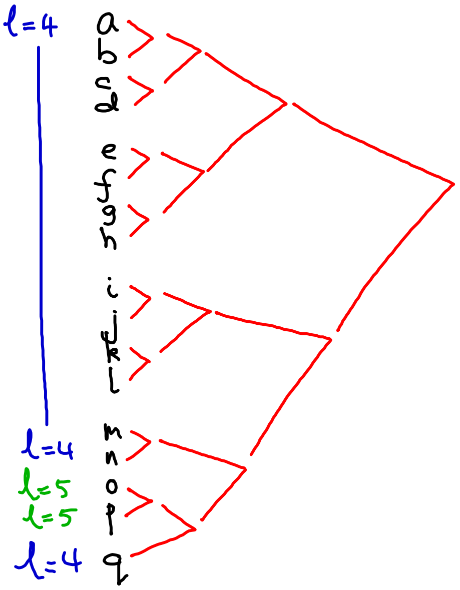
(b.iii) The second input distribution fails to use the outputs 101 and 010. It is a characteristic of the optimal input distribution that every output that is reachable is used with non-zero probability. Therefore this is not an optimal input distribution, and the capacity is greater than 2 bits.

The optimal input distribution will have some symmetry properties: the probabilities of 0000 and 1111 will be the same, and we expect them to be the most probable inputs since they deprive the channel of the chance to inject noise in the output. Similarly, 1001 and 0110 will have the same probability. 1100 and

0011 too. 1110 and 0001 and 1000 and 0111 will have the same probability, but it is not clear whether it's zero. 1010 and 0101 will have the same probability, but it is not clear whether it's zero. One way of guessing the probabilities of the inputs is to rank (inversely) the conditional entropies, so as to guess $P(0000) > P(1110) > P(1100) > P(0110) > P(1010)$. Conditional entropies for these are respectively $H(Y \,|\, x{=}0000) = 0$, $H_2(3/4)$, $H_2(0.5)$, $H(1/2, 1/4, 1/4)$, 4. However, looking at the conditional entropies alone ignores the overlaps between the distributions. The only thing we can be sure of is that the outputs 010 and 101 must be used; and these can be generated with highest probability by 1001 and 0110. So we expect that those two inputs will be used in addition to the four specified in the question. [4]

[In fact the answer found numerically is that only the inputs 1111, 0000, 1100, 0011, 1001, and 0110 are used in the optimal input distribution. Their probabilities are proportional to 4, 4, 3, 3, 2, 2, respectively.]

2 The optimal binary symbol code for the first character can be found by the Huffman algorithm. [1]



The result of this algorithm (see figure above) is that two of the outcomes get [1] codewords of length 5 and the other 15 get codewords of length 4. [1]

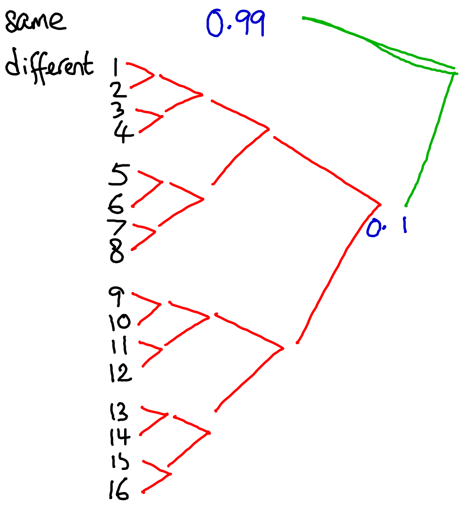
The expected length of the code is $4\frac{2}{17}$ and the expected length of the whole [1] file is

$$4\frac{2}{17}N.$$ [1]

[1]

The optimal binary symbol code for the nth symbol given the preceding one is found by the Huffman algorithm. The result of this algorithm (see figure below) [1] is that the outcome "same again" gets a codeword of length 1 and the other 16 get [1] codewords of length 5.

[1]



The expected length of the code is 1.04 The expected length of the whole file is

$$4\frac{2}{17} + (N-1)1.04.$$

The conditional entropy $H(X_n|X_{n-1})$ is
$H_2(0.01) + 0.01 \times 4 = 0.08 + 0.04 \simeq 0.12$.

So the entropy of the whole file is

$$\log_2 17 + (N-1)0.12.$$

(Notice this is about 7 times better than the previous compression method.) This is the best conceivable compression that any method could deliver. (For large N the first term is negligible, so I don't mind if it's omitted.)

I would compress the outcome using an arithmetic code. This would get within 2 bits of the ideal achievable compression. Arithmetic code works by subdividing the real line in proportion to probabilities and finding a binary string whose interval lies inside the source string's interval.

Alternate answer: I would use a runlength code to encode runs of 'same again' outcomes. The code for the number of 'same again's would represent that number in the form $64m + n$ where $n \in 0, 1, 2, \ldots 63$ is encoded with a string of 6 bits, and $m \in 0, 1, 2, 3, \ldots$ is encoded with a string of m zeros followed by a one. This would get within 1% or so of optimal compression.

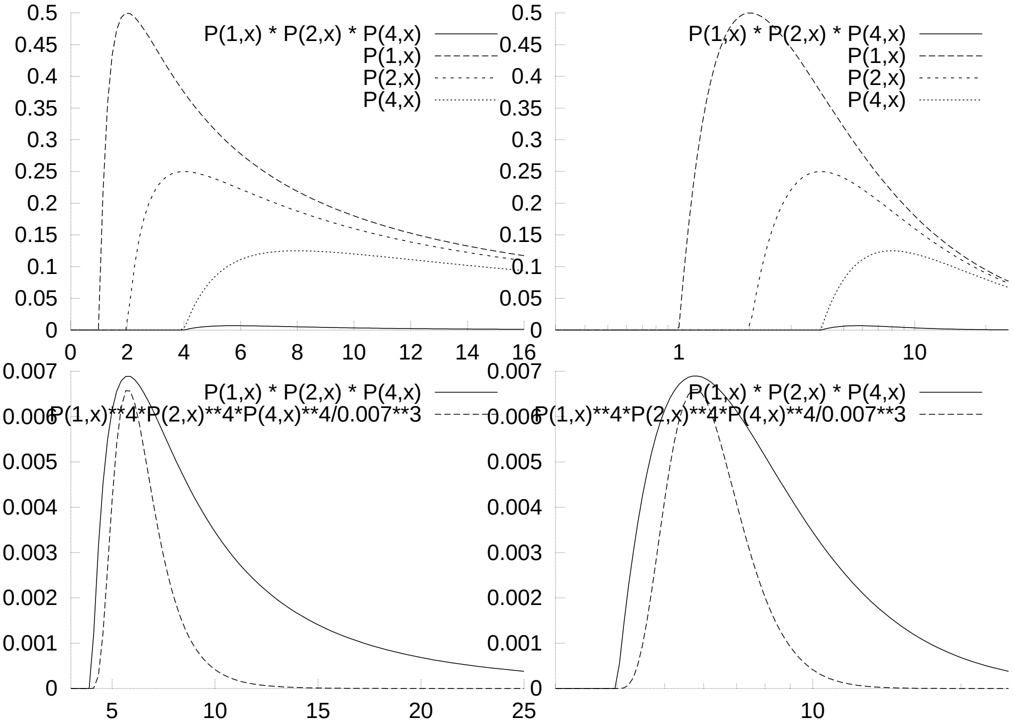
3 (a) The likelihood function is $P(D|\theta)$ (how well parameters θ predicted the data that actually happened), the prior probability distribution is $P(\theta)$, and the posterior probability distribution is given by

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)}.$$

(b) The likelihood function is

$$P(\{x_n\} \mid \lambda) = \prod_n P(x_n \mid \lambda)$$

For the three data points, the likelihood must be zero everywhere to the left of the rightmost point. The likelihood rises from zero there to a peak then falls again for large λ. [The plots below show (top) the three factors in the likelihood and (bottom) the likelihood functions for the two cases. The left plots have x on a linear scale and the right ones use a log scale.]

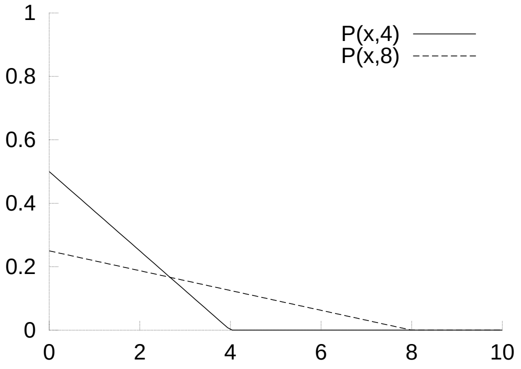


When we have four times as much data, the likelihood is the same function raised to the fourth power. The peak of the likelihood is in exactly the same place. The width of the likelihood is approximately half as wide (because uncertainties reduce as $1/\sqrt{N}$).

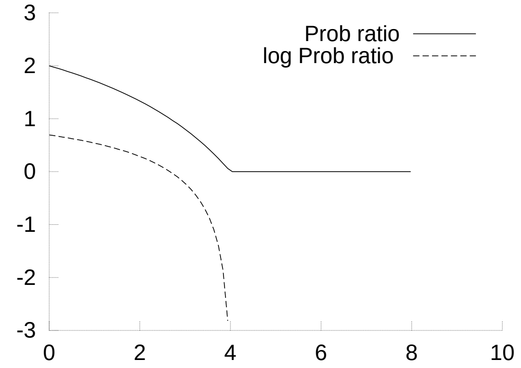
(c) The posterior probability of the class c is

$$P(c = 0 \mid x) = \frac{P(x \mid \lambda = 4)}{P(x \mid \lambda = 4) + P(x \mid \lambda = 8)}$$

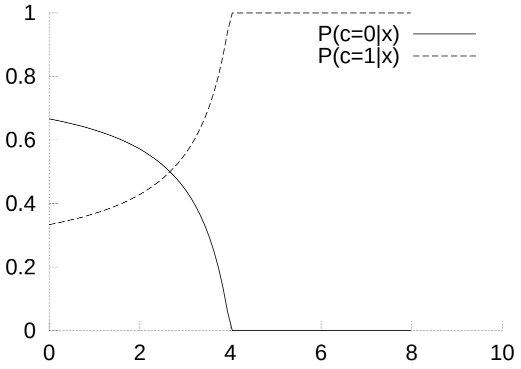
The next sketch shows the two densities



Here are the probability ratio and the log probability ratio



And here's the posterior probability of c. Points to note on the sketch are: the point at which the posterior is 50:50 is the place where the two densities $P(x \mid \lambda)$ intersect, which is at $x = (2/3) \times 4$. The posterior probability at $x = 0$ is in the ratio 2:1, *i.e.*, 2/3:1/3.



Information Theory, Pattern Recognition and Neural Networks

Part III Physics exam 2004

1 Describe an error-correcting code, *other than a repetition code or a Hamming code*, for detecting and correcting errors on a noisy channel such as the binary symmetric channel whose flip probability is $f \simeq 0.01$.

Describe a decoding algorithm for the code, and estimate its probability of error. [15]

Discuss how the performance of the code compares with the theoretical limits for error-correction proved by Shannon. [5]

[$H_2(0.01) \simeq 0.08$.]

2 A bent coin with probability $f = 0.01$ of coming up heads is tossed $N = 10,000$ times, generating a string of outcomes $x_1 x_2 \ldots x_N$. Describe **two** methods for compressing and uncompressing this string of outcomes. Compare and contrast the practical benefits of the two methods.

Estimate the mean compressed length achieved by each method, and compare them with the theoretical best achievable compressed length.

For one of the two methods, discuss how it would perform if, unknown to the designer, the bias of the coin in fact had a value f_{True} different from 0.01. [20]

[$H_2(0.01) \simeq 0.08$.]

3 How would you write a computer program that, given a small number of consecutive characters from the middle of an email message, recognises whether the email is in English? Please assume that the two alternative hypotheses are that the email is in English ($\mathcal{H}_E$), or that it is a random string of characters drawn from the same alphabet ($\mathcal{H}_R$).

Describe how the certainty of your method's decision would depend on the number of consecutive characters provided. Estimate how many characters your method would need in order to work reasonably well.

If instead the two alternative hypotheses are that the file is in English ($\mathcal{H}_E$) or that it is in German, using the same character set as English ($\mathcal{H}_G$), indicate briefly how you would modify your method, and how the number of characters required for good performance would be affected.

(Some facts about English and German are supplied below.)

Letter frequencies of English and German

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
English ($\boldsymbol{e}$)	.07	.01	.03	.03	.10	.02	.02	.05	.06	.001	.006	.03	.02	.06
German ($\boldsymbol{g}$)	.06	.02	.03	.04	.15	.01	.03	.04	.07	.002	.01	.03	.02	.08
	O	P	Q	R	S	T	U	V	W	X	Y	Z	-	
English ($\boldsymbol{e}$)	.06	.02	.0009	.05	.05	.08	.02	.008	.02	.002	.01	.0008	.17	
German ($\boldsymbol{g}$)	.02	.007	.0002	.06	.06	.05	.04	.006	.02	.0003	.0003	.01	.14	

The entropies of these two distributions are $H(\boldsymbol{e}) = 4.1$ bits; $H(\boldsymbol{g}) = 4.1$ bits; and the relative entropies between them are $D_{\rm KL}(\boldsymbol{e}||\boldsymbol{g}) = 0.16$ bits and $D_{\rm KL}(\boldsymbol{g}||\boldsymbol{e}) = 0.12$ bits. The relative entropies between the uniform distribution $\boldsymbol{u}$ and the English distribution $\boldsymbol{e}$ are $D_{\rm KL}(\boldsymbol{e}||\boldsymbol{u}) \simeq 0.6$ bits and $D_{\rm KL}(\boldsymbol{u}||\boldsymbol{e}) \simeq 1$ bits.

END OF PAPER