Probability

$P(A, B) = P(A|B)P(B)$

$P(A) = \sum_B P(A|B)P(B)$

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

Information

Entropy

$I = \log_2 p = -H$

Using a logarithm makes the measure additive and monotonic. The negation reflects the intuition that information should reflect the uncertainty removed by some probability being realised.

An *ensemble* is the set of outcomes of one or more random variables. The outcomes form a probability distribution.

$H = -\sum_i p_i \log p_i$

A *joint ensemble* is an ensemble whose outcomes are pairs drawn from two other ensembles. This defines a two dimensional probability distribution.

$H(X, Y) = \sum_{x,y} p(x, y) \log \frac{1}{p(x,y)}$

$H(X|Y) = \sum_y p(y) \left[\sum_x p(x|y) \log \frac{1}{p(x|y)}\right] = \sum_{x,y} p(x, y) \log \frac{1}{p(x|y)}$

We also have the Chain Rule and a consequence of it:

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

$H(X_1, X_2, \ldots, X_n) \leq \sum_{i=1}^{n} H(X_i)$

Mutual Information

$I(X; Y) = \sum_{x,y} p(x, y) \log \frac{p(x,y)}{p(x)p(y)} \geq 0$

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

Distance

$D(X, Y) = H(X, Y) - I(X; Y)$ (satisfies standard axioms)

$D_{KL}(p||q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$ (not symmetric, but measures inefficiency of assuming distribution is q(x) when it is p(x))

Bounds

$P_e \geq \frac{H(X|Y)-1}{\log|\mathbf{A}|}$ (Fanos Inequality, for outcomes $\mathbf{A}$)

$I(X; Y) \geq I(X; Z)$ (Data Processing Inequality, if X, Y, Z form a Markov chain)

Codes

Consider an information source with memory (i.e. its FSA representation has more than one state). The entropy of each state is as normal based on the weights of the edges leaving it, and the entropy of the system is (given that the probability of occupying state i is P_i):

$H = \sum_i P_i H_i = -\sum_i \sum_j P_i p_i(j) \log p_i(j)$

Fixed Length Codes

For N symbols, we require $R = \lceil \log_2(N) \rceil$ bits per block (known as the *code rate*). Efficiency is given by:

$\eta = \frac{H}{R} \leq 1$

This quantity is only 1 if the N symbols are equiprobable and N is a power of two. We can overcome the second constraint by encoding symbols as blocks of length J, such that:

$R = \frac{\lceil \log N^J \rceil}{J} = \frac{\lceil J \log N \rceil}{J}$

Variable Length Codes

A code is *uniquely decodable* if every output string can be produced by at most one input string. A code is a *prefix* code if there is no code word which is a prefix of a longer code word. A necessary condition for the existence of a prefix code where binary code words lengths are n_i is the Kraft-McMillan inequality:

$\sum_{i=1}^{N} \frac{1}{2^{n_i}} \leq 1$

The *source coding theorem* states that for a discrete memoryless source with finite entropy H; for any positive ϵ it is possible to encode the symbols at an average rate R such that $R = H + \epsilon$.

Discrete Memoryless Channels

Characterize these with an input alphabet $\mathcal{X}$ and output alphabet $\mathcal{Y}$, with corresponding random variables. Further introduce a matrix of transition probabilities $p(y_k|x_j) = P(Y = y_k|X = x_j)$, such that $\sum_{k=1}^{K} p(y_k|x_j) = 1$. If the input symbols are a subset of the output symbols we can define the probability of error as:

$P_e = \sum_{k=1}^{K} \sum_{j=1, j\neq k}^{J} p(y_k|x_j)p(x_j)$

The special case of a binary symmetric channel is a discrete memoryless channel where $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ and where there is a common probability of error p.

Note that we can regard the alphabets as ensembles if we have probability information and hence define mutual information, entropy and so on. We can now define channel capacity as:

$C = \max_{\{p(x_j)\}} I(\mathcal{X}; \mathcal{Y})$

Shannon's *channel coding theorem* states that for a channel of capacity C and a source of entropy H; if $H \leq C$, then for an arbitrarily small ϵ, there exists a coding scheme such that the source is reproduced with a residual error rate of less than ϵ.

Repetition Codes

In this code we transmit $n = 2m+1$ bits per symbol, so that we only get an error if $m+1$ or more bits are received in error. Given a binary symmetric channel, this means:

$P_e = \sum_{i=m+1}^{2m+1} \left(\begin{array}{c} 2m+1 \\ i \end{array} \right) p^i (1-p)^{2m+1-i}$

Hamming Codes

A 7/4 Hamming Code requires that we transmit the 4 bits of the underlying symbol (as bits 3, 5, 6 and 7 of the output) along with some others:

$b_4 = b_5 \oplus b_6 \oplus b_7$

$b_2 = b_3 \oplus b_6 \oplus b_7$

$b_1 = b_3 \oplus b_5 \oplus b_7$

Upon reception the syndromes are computed:

$s_4 = b_4 \oplus b_5 \oplus b_6 \oplus b_7$

$s_2 = b_2 \oplus b_3 \oplus b_6 \oplus b_7$

$s_1 = b_1 \oplus b_3 \oplus b_5 \oplus b_7$

If $s_4 s_2 s_1 = 0$ then there is no error, otherwise that is the index of the bit in error. This uses 3 bits to correct 7 error patterns and transfer 4 useful bits. They are *perfect codes* since in general they use m bits to correct $2^m - 1$ errors. They exist for all pairs $(2^n - 1, 2^{n-1})$. The probability of error is:

$P_e = \sum_{i=2}^{7} \left(\begin{array}{c} 7 \\ i \end{array} \right) p^i (1-p)^{7-i}$

Continuous Information

A naive definition of entropy leads to an integral which is not defined. Hence we define *differential entropy* as follows:

$h(\mathcal{X}) = \int_{-\infty}^{\infty} p(x) \log \left(\frac{1}{p(x)} \right) dx$

All other definitions are analogous to their discrete counterparts. We find that the maximum differential entropy for a given variance is $h(X) = \frac{1}{2}\log(2\pi e \sigma^2)$ which is realized by the Gaussian distribution $p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$

When we come to consider ensembles, we in particular consider those of *functions*.

Channel Capacity

Consider signals with average power P, time limited to T and band limited to W. These may be affected by additive Gaussian white noise N such that $\sigma^2 = N_0 W$ where N_0 is the *power spectral density*. In this case:

$h(Y|X) = h(N) = \frac{1}{2}\log(2\pi e N_0 W)$

And so when we maximize mutual information to find the channel capacity we find that the maximum is achieved when X and Y are Gaussian, X having variance P and Y having variance $P + N_0 W$. Now:

$C = \frac{1}{2}\log(1 + \frac{P}{N_0 W})\text{bits/symbol} = W \log_2(1 + \frac{P}{N_0 W})\text{bits/s}$

The fractional quantity is known as the signal to noise ratio. This statement of capacity is known as Shannon's *channel capacity theorem*. Note that as the SNR increases C increases without limit, but as W increases we reach the limit $C \rightarrow \frac{P}{N_0}\log e$. We can define energy per bit, $E_b = \frac{P}{C}$.

Fourier Analysis

Continuous Transform

Consider *real valued, complex functions* $f(x)$. We will use the cosine and sine series as the basis. Now:

$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n cos(nx) + b_n sin(nx)$

$a_n = \frac{1}{\pi}\int_0^{2\pi} f(x) cos(nx) dx$ where $n \geq 0$

$b_n = \frac{1}{\pi}\int_0^{2\pi} f(x) sin(nx) dx$ where $n \geq 1$

If $f(x) = f(-x)$ then $b_n = 0$, if $f(x) = -f(-x)$ then $a_n = 0$. We can also work with complex numbers:

$f(x) = \sum_{-\infty}^{\infty} c_n e^{inx}$

$c_n = \frac{1}{2\pi}\int_0^{2\pi} f(x) e^{-inx} dx$

The Fourier Series is the best approximation to $f(x)$ in terms of mean squared error for linear combinations of *sin* and *cos* terms. Some interesting properties are:

1. If $f(x)$ is bounded in $(0, 2\pi)$ and piecewise continuous then the Fourier series converges to $\{f(0_+) + f(2\pi_-)\}/2$ at 0 and 2π

2. If $f(x)$ is continuous within $(0, 2\pi)$ the sum of the series is equal to $f(x)$ at all points

3. a_n and $b_n$tend to 0 at least as fast as $\frac{1}{n}$ (we can make finite approximations)

Properties:

1. If $g(x)$ real then $G(-k) = G*(k)$

2. Linearity: $ag_1(x) + bg_2(x) \rightleftharpoons aG_1(x) + bG_2(X)$
3. Time Shift: $g(x-a) \rightleftharpoons e^{-ika}G(k)$
4. Frequency Shift: $g(x)e^{i\lambda x} \rightleftharpoons G(k-\lambda)$
5. Differentiation: $g^{(n)}(x) = (ik)^n G(k)$

Define convolution as $(f * g)(x) = \int_{-\infty}^{\infty} f(y)g(x-y)dy$. Convolution in one domain is multiplication in the other.

Sampling

Define the *ideal sampling function* of interval X as:

$\delta_X(x) = \sum_n \delta(x - nX) \rightleftharpoons \frac{1}{X}\sum_m \delta(kX - 2\pi m)$

$g_X(x) = g(x)\delta_X(x)$ (a sampled function g)

The *Nyquist rate* is now defined as $f_s = 2W$ for a signal band limited to W. This is the minimum sampling rate required to prevent aliasing.

Discrete Transform

Consider a data sequence g_n of length N, possibly sampled from an analogue signal $s(t)$ such that $g_n = s(nT_s)$. Now:

$G_k = \sum_{n=0}^{N-1} g_n e^{-\frac{2\pi i}{N}kn}$

$g_n = \frac{1}{N}\sum_{k=0}^{N-1} G_k e^{\frac{2\pi i}{N}kn}$

Properties:

1. If $g_n$is real then $G_{-k} = G*_k$
2. Since $g_n$is periodic, $G_{\frac{N}{2}-k} = G*_{\frac{N}{2}+k}$
3. Linearity: $ag_n + bh_n$ has DFT $aG_k + bH_k$
4. Shifting: for the rotated sequence g_{n-n_0} the DFT is $G_k e^{-\frac{2\pi i k n_0}{N}}$

Define circular convolution as $(g * h)_k = \sum_{r=0}^{N-1} g_r h_{n-r}$. Again, multiplication in one domain is convolution in the other.

Fast Fourier Transform

Observe that, if $\omega = e^{-\frac{2\pi i}{N}}$:

$G_k = \sum_{n=0}^{2L-1} g_n \omega^{nk} = \sum_{n=0}^{L-1}(g_n + g_{n+L}(-1)^k)\omega^{kn}$

This has split the DFT into two smaller DFTs. The even and odd terms of G can now be separated out. If N is a power of 2, this can be done $\log N$ times to obtain N single point transforms. Each "butterfly" requires one complex multiplication and two additions hence the FFT requires $\frac{N}{2}\log N$ multiplications and $N\log N$ additions. This is $O(N\log N)$ rather than the naive $O(N^2)$ complexity.

To find the location of $G_k$in the FFT output array, we can take k as a binary number of $\log N$ bits, reverse them, and treat that as an index into the array.

We can rewrite an inverse FFT like so:

$Ng*_n = \sum_{k=0}^{N-1} G *_k \omega^{kn}$

Which is just a DFT of the complex conjugates of the Fourier coefficients, hence FFT can be used to implement IFFT.

Quantization

Logan's Theorem

Logan's theorem states that if a signal $f(x)$ is band-limited to one octave or less (i.e. $k_{max} \leq 2k_{min}$) and $f(x)$ contains no complex zeroes in common with its Hilbert transform, then the original signal can be recovered up to an amplitude scaling constant by the set of zero-crossings of $f(x)$ alone. However:

- No stable constructive algorithm for making this work is known, although it exists
- It cannot be easily generalized to higher dimensions due to the octave constraint. If an annulus in the frequency domain was chosen then the projection onto any axis would not be band-limited to an octave, and if disjoint squares were chosen then the transform would be anisotropic. Furthermore there are infinite complex zeroes in such a case

Information Diagrams

The similarity theorem states that $f(ax) \rightleftharpoons \left|\frac{1}{a}\right| F(\frac{k}{a})$. Gabor further proved that $(\Delta x)(\Delta k) \geq \frac{1}{4\pi}$ (i.e. the Gabor-Heisenberg-Weyl Uncertainty Principle), where Δx and Δk are the normalized variances (second-moments) of the time and frequency domain functions respectively.

Areas in the information diagram of this size are called *logons*. The family of signals which achieve this minimal area are the Gabor wavelets:

$f(x) = e^{-\frac{(x-x_0)^2}{a^2}} e^{-ik_0(x-x_0)}$

These are localized at epoch x_0, modulated by frequency k_0 and have size constant a, and may be visualized by decaying spirals in the complex plane which evolve along the x axis. They are self-Fourier, and the transform is as follows:

$F(k) = e^{-(k-k_0)^2a^2} e^{-ix_0(k-k_0)}$

Unfortunately Gabor wavelets are mutually non-orthogonal, although methods do now exist to obtain coefficients for this basis. Note that the parameter a in some sense unifies the time and frequency domains.

Kolmogorov Complexity

The Kolmogorov complexity, K of a string is the length in bits of the shortest program which produces that string. It is approximately equal to the entropy H of the distribution from which the string is a randomly drawn sequence.

A sequence of length n is said to be *algorithmically random* if its Kolmogorov complexity is at least n. An infinite string is defined to be *incompressible* if its Kolmogorov complexity in the limit as the string gets arbitrarily long approaches n. The Strong Law of Large Numbers for Incompressible Sequences asserts that the proportions of 0s and 1s in any incompressible string must be nearly equal.